TEaR: Improving LLM-based Machine Translation with Systematic Self-Refinement

Zhaopeng Feng [1*] **Yan Zhang** [2*] **Hao Li** [2] **Bei Wu** [2] **Jiayu Liao** [2]
Wenqiang Liu [2] **Jun Lang** [2] **Yang Feng** [3] **Jian Wu** [1] **Zuozhu Liu** [1†]
[1]ZJU-UIUC Institute, Zhejiang University [2]Tencent
[3]Angelalign Research Institute, Angelalign Technology Inc.
{zhaopeng.23,zuozhuliu}@intl.zju.edu.cn
erikyzzhang@global.tencent.com

Abstract

Large Language Models (LLMs) have achieved impressive results in Machine Translation (MT). However, human evaluations reveal that LLM-generated translations still contain various errors. Notably, feeding the error information back into the LLMs can facilitate self-refinement, leading to enhanced translation quality. Motivated by these findings, we introduce **TEaR** (**T**ranslate, **E**stimate, **a**nd **R**efine), a systematic LLM-based self-refinement framework aimed at bootstrapping translation performance. Our key results show that: 1) TEaR framework enables LLMs to improve their translation quality relying solely on self-feedback, measured by both automatic metrics and Multidimensional Quality Metrics (MQM) scores; 2) TEaR autonomously selects improvements, ensuring a robust translation quality baseline while outperforming both internal refinement and external feedback methods. Error analysis and iterative refinement experiments show its ability to continuously reduce translation errors and enhance overall translation quality. Our code and data are publicly available at https://github.com/fzp0424/self_correct_mt.

1 Introduction

The results of the General Machine Translation Task (Kocmi et al., 2023) [1] in WMT23 indicate that LLM-based machine translation systems (Vilar et al., 2023; He et al., 2023; Gao et al., 2023; Wu and Hu, 2023; Peng et al., 2023; Moslem et al., 2023; Liang et al., 2023), especially GPT-4 (Achiam et al., 2023), have taken top positions in the majority of MT subtasks. However, taking the whole 1976 test pairs from the WMT23 Zh-En dataset as a probing study, even with GPT-4 (the best submission in WMT23), only 332 pairs achieved a perfect score (i.e., no errors were identified) according to the Multidimensional Quality Metrics (MQM, Freitag et al., 2021). [2]

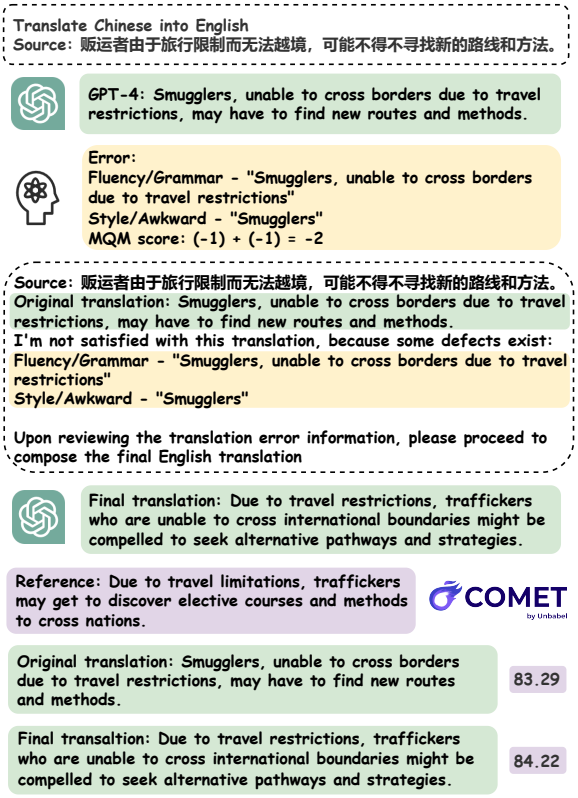

Figure 1: The original translation is the submission of GPT-4 for WMT23. The MQM error is annotated by human experts. We use OpenAI API *gpt-4-0613* to correct the translation. The metric we use is COMET-22 (*wmt22-comet-da*) (Rei et al., 2020).

Interestingly, when we provide GPT-4 with human-generated MQM evaluations and prompt it to correct its initial translation based on that feedback, we observe that many errors are effectively resolved, resulting in an improved metric score, as shown in Figure 1. These findings highlight the potential of using evaluation feedback to refine translations within a single LLM, motivating our

[*]Equally Contributed.
[†]Corresponding author.
[1]WMT 2023 Shared Task: General Machine Translation
[2]https://github.com/google/wmt-mqm-human-evaluation

Findings of the Association for Computational Linguistics:
NAACL 2025, pages 3922–3938

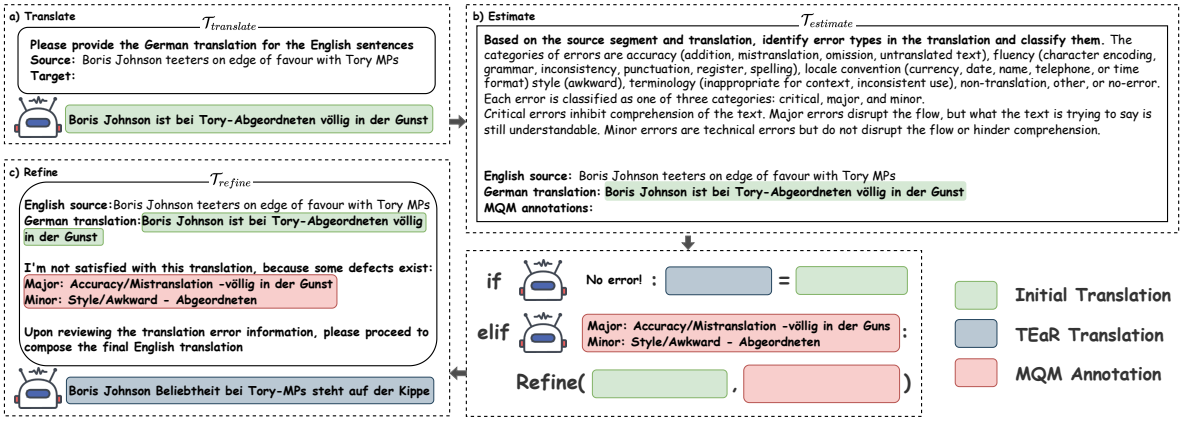

Figure 2: TEaR framework involves three steps: *Translate*, *Estimate*, and *Refine*.

approach to translation bootstrapping.

Involving LLMs to automatically do the self-refinement is increasingly gaining attention (Saunders et al., 2022; Pan et al., 2023; Madaan et al., 2023; Shinn et al., 2023; Li et al., 2023). In the field of MT, Chen et al. (2023) investigate the use of LLMs to rewrite translations by feeding hint words, achieving changes at the lexical and structural levels while maintaining translation quality. Raunak et al. (2023) ask GPT-4 to refine translations from other neural machine translation models based on suggestions from the Chain-of-Thought (CoT) strategy (Kojima et al., 2022), indicating that GPT-4 is adept at translation post-editing. However, these approaches face several challenges: 1) These methods lack an independent evaluation module, resulting in insufficient analysis of translation quality and errors; 2) As a consequence, the refinement process is often unclear and ineffective, providing either vague guidance or redundant information; 3) The self-correction capabilities of LLMs in MT remain largely unexplored, with limited research into their potential for continuous self-refinement.

In this paper, we propose a systematic and plug-and-play self-refinement translation framework, termed **TEaR**: **T**ranslate, **E**stimate, **a**nd **R**efine. **Translate** module uses an LLM to generate translations, ensuring all translations are derived internally. **Estimate** module evaluates the quality of the initial translation, determining whether it should proceed to the **Refine** module. **Refine** module refine the translation based on the feedback from the preceding two modules. Unlike existing self-refinement approaches that rely on external models or human annotations, TEaR operates entirely within a single LLM, eliminating the need for auxiliary training data or additional error-pinpointing models. Our framework leverages the LLM's dual capability as both a generator and a judge, enabling systematic error reduction through self-contained feedback loops. Our comprehensive experiments demonstrate that TEaR is effective to boost translation quality by reducing errors, and iterative refinements lead to consistent, incremental improvements, demonstrating its robustness and scalability across different language pairs and tasks.

2 TEaR

Figure 2 illustrates the workflow of TEaR, which consists of three modules: **Translate**, **Estimate**, and **Refine**. Given an LLM $\mathcal{M}$ and a language pair $\mathcal{X}$-$\mathcal{Y}$ (source-target), we first use $\mathcal{M}$ to generate the initial translation for a segment x with $\mathcal{T}_{translate}$. After obtaining the initial translation y, we apply the same $\mathcal{M}$ with $\mathcal{T}_{estimate}$ for quality estimation. In the **Estimate** step, the LLM follows a predefined MQM typology (see Table 15) to simulate a human annotator's evaluation. If no errors are detected, the initial translation is finalized as the TEaR output. However, if errors are identified, the process proceeds to the **Refine** step, where $\mathcal{M}$ uses the feedback from the estimation to refine the translation with $\mathcal{T}_{refine}$. The templates we used are provided in Appendix B.

Translate. We follow the zero-shot prompt setting outlined by Xu et al. (2023a) and Hendy et al. (2023). For the source sentence x, we use $\mathcal{M}$ to generate the target y using the template $\mathcal{T}_{translate}$. This step is essential for establishing a solid foundation for the subsequent refinement process.

Estimate. Quality Estimation (QE) in MT refers to a method for predicting the quality of a translation. Traditional neural QE models (Rei et al., 2022; Gowda et al., 2023; Juraska et al., 2023) typ-

System	Zh-En		De-En		Ru-En		Cs-En		Avg.	
	COMET-K	XCOMET	COMET-K	XCOMET	COMET-K	XCOMET	COMET-K	XCOMET	COMET-K	XCOMET
IT	79.29	87.22	80.29	95.00	81.29	95.08	82.09	94.34	80.74	92.91
CT	79.25	87.12	80.12‡	95.02	81.30	95.04	82.18	94.14‡	80.71	92.83
	(-0.04)	(-0.10)	(-0.17)	(+0.02)	(+0.01)	(-0.04)	(+0.09)	(-0.20)	(-0.03)	(-0.08)
SCoT	79.38	87.07‡	80.44‡	95.38‡	81.35	95.22	82.26‡	94.30	80.86	92.99
	(+0.09)	(-0.15)	(+0.15)	(+0.38)	(+0.06)	(+0.14)	(+0.17)	(-0.04)	(+0.12)	(+0.08)
TEaR	79.46‡	87.47‡	81.09‡	95.66‡	81.49‡	95.25‡	82.46‡	94.94‡	81.12	93.33
	(+0.17)	(+0.25)	(+0.80)	(+0.66)	(+0.20)	(+0.17)	(+0.37)	(+0.60)	(+0.38)	(+0.42)

Table 1: Results for WMT22 XX→En test set. **IT**: Initial translation; **SCoT**: Structured Chain-of-Thought (Raunak et al., 2023); **CT**: Contrastive Translation (Chen et al., 2023). **TEaR**: TEaR translation using one iteration. Blue indicates the improved scores. Red indicates the decreased scores. ‡: statistically significant at $p < 0.05$.

System	En-Zh		
	COMET-K (↑)	XCOMET (↑)	MQM score (↓)
IT	82.07	87.75	3.70
CT	82.08	87.72	3.63
SCoT	82.13	87.81	3.49
XCOMET-Score (outside)	82.08	87.68	3.57
XCOMET-Span (outside)	82.27	88.01	3.61
TEaR(Iter 1)	82.33	87.99	3.18
TEaR(Iter 8)	**82.44**	**88.31**	**2.40**

Table 2: Comparison with baselines on WMT22 En-Zh test set. *outside* indicates the source of the feedback. **Bold font** and underline indicate the best and second best performance, respectively.

ically provide a single numerical score to represent quality. While models like XCOMET (Guerreiro et al., 2023) and MaTESe (Perrella et al., 2022) can predict error spans, there are still three challenges: 1) They can only provide severity information; 2) Interpreting the numerical representations can be difficult, and predicting error spans may lead to overfitting to the QE model's scores; 3) Incorporating additional models introduces extra deployment costs and requires a substantial amount of annotated MQM data for training. Recent studies have shown that LLMs can annotate translation errors similarly to human (Kocmi and Federmann, 2023; Fernandes et al., 2023). Building on this, we designed a simplified template, $\mathcal{T}_{estimate}$, based on MQM annotator guidelines (Freitag et al., 2021), allowing $\mathcal{M}$ to evaluate translation quality in a human-like manner. Using $\mathcal{T}_{estimate}$, $\mathcal{M}$ makes a clear decision on whether corrections are needed, reducing unnecessary refinements and improving the stability of translation quality. Specifically, if no errors are found, the initial translation y is retained as the final TEaR translation. If errors are detected, the MQM annotations (including error types, severity, and spans) are used as feedback for further refinements.

Refine. Chen et al. (2023) and Raunak et al. (2023) have demonstrated the potential of prompting LLMs to refine translations. However, their approaches lack an explicit estimation module and feedback mechanism, making their refinement processes less transparent. Moreover, they primarily focus on refining translations from external sources rather than systematically exploring LLM self-refinement. To address this, we collect MQM annotations from the **Estimate** module and use it as feedback in $\mathcal{T}_{refine}$ to refine the initial translation.

3 Experiments

3.1 Experimental Setup

Dataset. Our test set comes from WMT22, covering 8 translation directions across 5 languages: English (En), German (De), Czech (Cs), Chinese (Zh), and Russian (Ru). Statistics are in Appendix A.

Models. We use GPT-4o mini as the primary model, which supports the same range of languages as GPT-4o [3]. Model details are in Appendix C.

Baseline. We compare TEaR with three baselines: 1) IT (Initial Translation); 2) SCoT (Structured-CoT), which guides the LLM to suggest improvements based on the original translation (Raunak et al., 2023); and 3) CT (Contrastive Translation), where the word "bad" is added to signal low quality and trigger a contrastive prompt (Chen et al., 2023). We also include XCOMET (Guerreiro et al., 2023), an external estimation model, which provides feedback using both scalar and span forms.

Metrics. We use reference-free metric COMET-K (COMETKiwi, Rei et al. (2022)) and reference-based metric XCOMET (XCOMET-XL, Guerreiro et al. (2023)) as the main metrics. For MQM scores, we follow the assessment method in Freitag et al. (2021), including guidelines to annotators, error category, severity level, and error weighting, using GEMBA-MQM (Kocmi and Federmann, 2023)

[3] https://platform.openai.com/docs/models/gpt-4o-mini

System	CT	SCoT	XCOMET-Score	XCOMET-Span	TEaR
# of refined IT	1854	1694	1740	1417	693
COMET-K	82.07	82.09	81.82	81.47	78.78
XCOMET	87.69	87.41	86.18	84.59	78.64

Table 3: Comparison of refinement robustness on WMT22 En-Zh. The last two rows are the metric scores of the initial translations in the first row.

framework. More details are in Appendix D.

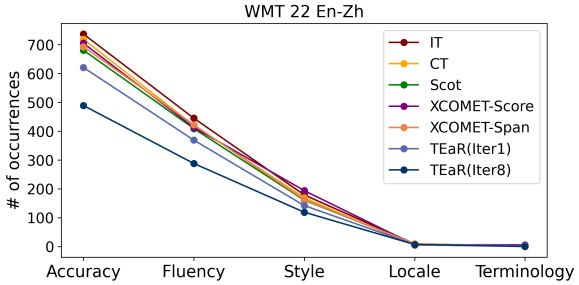

Figure 3: Comparison of errors for different systems. All the translations were re-evaluated in the same setting. **Locale** represents *Locale convention* in Table 16.

3.2 Main Results

Table 1 and 5 show that TEaR successfully improves overall IT quality for all 8 translation directions. Specifically, TEaR achieves an average improvement of +0.38 in COMET-K and +0.42 in XCOMET for XX→En translations. In contrast, SCoT only shows an average increase of +0.12 in COMET-K and +0.08 in XCOMET, with declines observed in the XCOMET scores for the Zh-En and Cs-En directions. CT exhibits an average decline in both COMET-K and XCOMET metrics.

We also compare using the external evaluation as feedback by substituting **Estimate** with XCOMET, as reported in Table 2. We can notice that TEaR outperforms all self-refinement methods and XCOMET-based refinement both in automatic metrics and MQM scores. XCOMET-Span achieves superior performance in XCOMET, which also indicates that fine-grained error estimation can help LLMs enhance translation quality.

3.3 Analysis

TEaR Reduces More Translation Errors. We evaluated the number of translation errors after refinements, using the same settings as the MQM score. As shown in Figure 3, TEaR significantly reduces translation errors, especially in categories like *Accuracy*, *Fluency*, and *Style*. See more language comparison in Figure 5.

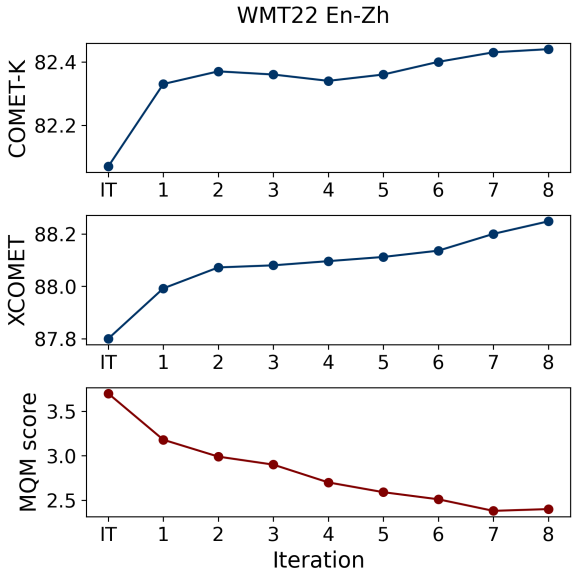

Figure 4: Iterative experiments on WMT22 En-Zh test set. Each iteration includes translations that were deemed "no error" before, as well as those that were refined during the current iteration.

TEaR Is More Selective in Refinement. We analyzed the number of cases refined by each translation system. Table 3 shows that TEaR focuses on improving lower-quality translations, while other systems, such as SCoT and CT, tend to refine most translations regardless of quality. This is expected for systems without independent estimation modules. Notably, when using the external QE model XCOMET for estimation, we simulate TEaR that no refinements were made in cases where XCOMET marked no error spans or gave a score of 1. These results suggest that TEaR effectively identifies lower-quality translations for refinement, enhancing the system's robustness.

Iterative Refinement Yields Additional Improvements. To investigate the potential of iterative refinement of TEaR, in each iteration, we use the refined translation from the last iteration as the new initial translation. Any translation estimated as "no error" in a given iteration will remain unchanged in subsequent iterations. As shown in Figure 4, automatic metric scores improved steadily with each iteration, while MQM scores consistently decreased, indicating a reduction in the total number of errors. This pattern is also reflected in Figure 3.

4 Related Work

LLM-based self-refinement have shown promising results. Methods like Self-Refine (Madaan et al., 2023) and Reflexion (Shinn et al., 2023) use LLMs

to generate outputs, provide feedback, and iteratively refine them. Similarly, CRITIC (Gou et al., 2023) integrates external feedback for output improvement. In the MT domain, Chen et al. (2023) and Raunak et al. (2023) explored self-refinement with LLMs but lacked clear evaluation modules and feedback, limiting transparency and effectiveness. Other works focused on training models for translation refinement (Xu et al., 2023b; Koneru et al., 2023; Alves et al., 2024; Wang et al., 2024; Feng et al., 2024; Treviso et al., 2024), but most rely on external evaluation models and emphasize post-editing rather than comprehensive self-refinement.

5 Conclusion

In this paper, we introduce TEaR, a systematic and plug-and-play framework for self-refinement machine translation. TEaR enables large language models to autonomously refine their own translations based on error feedback, without the need for external supervision. Experimental results demonstrate that TEaR surpasses existing self-refinement MT methods in both metric scores and MQM scores. The iterative nature of TEaR allows for continuous refinement, providing a robust and scalable solution for bootstrapping translation quality.

Limitations

Applying self-refinement in the field of Machine Translation has inherent advantages, as the problem is well-defined, and the feedback information is highly directional and specific. While TEaR improves translation accuracy, its handling of more nuanced linguistic aspects, such as cultural context or domain-specific terminology, remains an area for further research. Moreover, distilling the performance of TEaR from black-box LLMs to smaller-size LLMs is also a promising future work.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 62476241), the Natural Science Foundation of Zhejiang Province, China (Grant No. LZ23F020008), and the Zhejiang University-Angelalign Inc. R&D Center for Intelligent Healthcare.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Duarte M Alves, José Pombal, Nuno M Guerreiro, Pedro H Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, et al. 2024. Tower: An open multilingual large language model for translation-related tasks. *arXiv preprint arXiv:2402.17733*.

Pinzhen Chen, Zhicheng Guo, Barry Haddow, and Kenneth Heafield. 2023. Iterative translation refinement with large language models. *arXiv preprint arXiv:2306.03856*.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Zhaopeng Feng, Ruizhe Chen, Yan Zhang, Zijie Meng, and Zuozhu Liu. 2024. Ladder: A model-agnostic framework boosting llm-based machine translation to the next level. *arXiv preprint arXiv:2406.15741*.

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André FT Martins, Graham Neubig, Ankush Garg, Jonathan H Clark, Markus Freitag, and Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Markus Freitag, Nitika Mathur, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Tom Kocmi, Frédéric Blain, Daniel Deutsch, Craig Stewart, et al. 2023. Results of wmt23 metrics shared task: Metrics might be guilty but references are not innocent. In *Proceedings of the Eighth Conference on Machine Translation*, pages 578–628.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68. Association for Computational Linguistics.

Yuan Gao, Ruili Wang, and Feng Hou. 2023. Unleashing the power of chatgpt for translation: An empirical study. *arXiv preprint arXiv:2304.02182*.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2023. Critic: Large language models can self-correct with tool-interactive critiquing. *ArXiv*, abs/2305.11738.

Thamme Gowda, Tom Kocmi, and Marcin Junczys-Dowmunt. 2023. Cometoid: Distilling strong reference-based machine translation metrics into Even stronger quality estimation metrics. In *Proceedings of the Eighth Conference on Machine Translation*, pages 751–755, Singapore. Association for Computational Linguistics.

Nuno M Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André FT Martins. 2023. xcomet: Transparent machine translation evaluation through fine-grained error detection. *arXiv preprint arXiv:2310.10482*.

Zhiwei He, Tian Liang, Wenxiang Jiao, Zhuosheng Zhang, Yujiu Yang, Rui Wang, Zhaopeng Tu, Shuming Shi, and Xing Wang. 2023. Exploring human-like translation strategy with large language models. *arXiv preprint arXiv:2305.04118*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *arXiv preprint arXiv:2302.09210*.

Juraj Juraska, Mara Finkelstein, Daniel Deutsch, Aditya Siddhant, Mehdi Mirzazadeh, and Markus Freitag. 2023. MetricX-23: The Google submission to the WMT 2023 metrics shared task. In *Proceedings of the Eighth Conference on Machine Translation*, pages 756–767.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, et al. 2023. Findings of the 2023 conference on machine translation (wmt23): Llms are here but not quite there yet. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1–42.

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*, pages 768–775.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Sai Koneru, Miriam Exel, Matthias Huck, and Jan Niehues. 2023. Contextual refinement of translations: Large language models for sentence and document-level post-editing. *arXiv preprint arXiv:2310.14855*.

Miaoran Li, Baolin Peng, and Zhu Zhang. 2023. Self-checker: Plug-and-play modules for fact-checking with large language models. *arXiv preprint arXiv:2305.14623*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.

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

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *arXiv preprint arXiv:2308.03188*.

Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2023. Towards making the most of ChatGPT for machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5622–5633, Singapore. Association for Computational Linguistics.

Stefano Perrella, Lorenzo Proietti, Alessandro Scirè, Niccolò Campolungo, and Roberto Navigli. 2022. MaTESe: Machine translation evaluation as a sequence tagging problem. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 569–577, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

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.

Vikas Raunak, Amr Sharaf, Hany Hassan Awadallah, and Arul Menezes. 2023. Leveraging gpt-4 for automatic translation post-editing. *arXiv preprint arXiv:2305.14878*.

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

Ricardo Rei, Marcos Treviso, Nuno M Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José GC De Souza, Taisiya Glushkova, Duarte Alves, Luísa Coheur, et al. 2022. Cometkiwi: Ist-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of ACL*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Marcos Treviso, Nuno M Guerreiro, Sweta Agrawal, Ricardo Rei, José Pombal, Tania Vaz, Helena Wu, Beatriz Silva, Daan van Stigt, and André FT Martins. 2024. xtower: A multilingual llm for explaining and correcting translation errors. *arXiv preprint arXiv:2406.19482*.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. Prompting PaLM for translation: Assessing strategies and performance. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15406–15427.

Yutong Wang, Jiali Zeng, Xuebo Liu, Fandong Meng, Jie Zhou, and Min Zhang. 2024. TasTe: Teaching large language models to translate through self-reflection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6144–6158, Bangkok, Thailand. Association for Computational Linguistics.

Yangjian Wu and Gang Hu. 2023. Exploring prompt engineering with GPT language models for document-level machine translation: Insights and findings. In *Proceedings of the Eighth Conference on Machine Translation*, pages 166–169, Singapore. Association for Computational Linguistics.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023a. A paradigm shift in machine translation: Boosting translation performance of large language models.

Wenda Xu, Daniel Deutsch, Mara Finkelstein, Juraj Juraska, Biao Zhang, Zhongtao Liu, William Yang Wang, Lei Li, and Markus Freitag. 2023b. Pinpoint, not criticize: Refining large language models via fine-grained actionable feedback. *arXiv preprint arXiv:2311.09336*.

A Dataset Statistics

Table 4 presents the statistics of our test data.

Language	Test set (WMT22)	
	from English	to English
Chinese (Zh)	2037	1875
German (De)	2037	1984
Russia (Ru)	2037	2016
Czech (Cs)	2037	1448

Table 4: The statistics for the test set we used.

B Templates

For $\mathcal{T}_{estimate}$, we follow the setting in Kocmi et al. (2023), using the few-shot (3-shot) multi-lingual MQM annotated examples to ask LLMs to estimate the initial translation in a reference-free scenario. Kocmi and Federmann (2023) have demonstrated that this prompt method ensures the estimation can be executed across any language pairs and can rival metrics models trained on a large amount of MQM annotated data. Table 17, 18, and 19 show the prompts we used in this work. Table 20 and 21 CT and SCoT.For external estimation feedback, the prompts can be found in Table 22 and 23.

C Models

GPT-4o mini surpasses GPT-3.5 Turbo and other smaller models in academic benchmarks while supporting the same range of languages as GPT-4o [4]. We use the version "gpt-4o-mini-2024-07-18" and set the model temperature to 0.7 for all settings.

D MQM Score

Multidimensional Quality Metric (MQM) is a human evaluation framework commonly used in WMT metrics shared tasks as the golden standard (Freitag et al., 2022, 2023). MQM is designed to assess and categorize translation errors. Previous work by Kocmi and Federmann (2023); Freitag et al. (2023) has demonstrated that GPT-4 can accurately identify error spans and achieve state-of-the-art performance in MT evaluation. Therefore, we use GPT-4 Turbo (API version "gpt-4-turbo-2024-04-09") with a model temperature of 0.7 to analyze translation errors generated by the translation systems.

[4] https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

System	En-Zh		En-De		En-Ru		En-Cs		Avg.	
	COMET-K	XCOMET	COMET-K	XCOMET	COMET-K	XCOMET	COMET-K	XCOMET	COMET-K	XCOMET
IT	82.07	87.75	83.48	96.72	83.62	93.03	84.56	93.08	83.43	92.65
CT	82.08	87.72	83.33‡	96.68	83.56	92.99	84.58	93.08	83.39	92.62
	(+0.01)	(-0.03)	(-0.15)	(-0.04)	(-0.06)	(-0.04)	(+0.02)	(+0.00)	(-0.04)	(-0.03)
SCoT	82.13	87.81	83.56	96.78	83.66	93.10	84.63	93.12	83.50	92.70
	(+0.06)	(+0.06)	(+0.08)	(+0.06)	(+0.04)	(+0.07)	(+0.07)	(+0.04)	(+0.07)	(+0.05)
TEaR	82.33‡	87.99‡	83.55	96.76	83.80‡	93.16‡	84.82‡	93.41‡	83.63	92.83
	(+0.26)	(+0.24)	(+0.07)	(+0.04)	(+0.18)	(+0.13)	(+0.26)	(+0.33)	(+0.20)	(+0.18)

Table 5: Results for WMT22 En→XX test set. The marker are the same in Table 1.

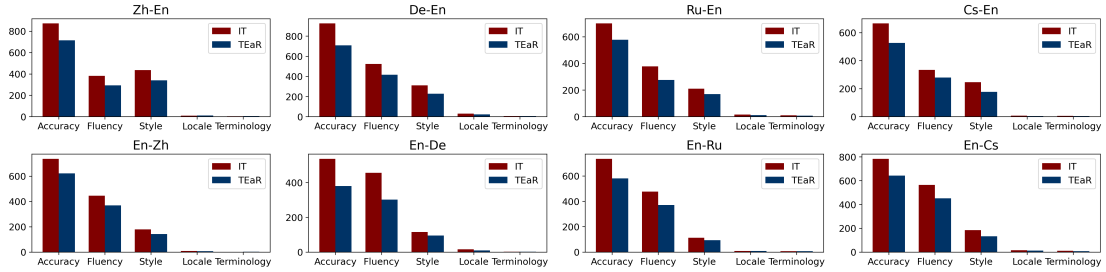

Figure 5: Comparison between initial translation and TEaR translation. TEaR can reduce all types of translation errors across all translation directions.

We follow the GEMBA-MQM (Kocmi and Federmann, 2023) setting to identify translation errors and label their category and severity. After counting the number of major and minor errors, we calculate the final MQM score using the following equation:

$$MQM \text{ score} = w_{\text{major}} n_{\text{major}} + w_{\text{minor}} n_{\text{minor}} \quad (1)$$

where n_{major} and n_{minor} represent the number of major and minor errors, and w_{major} and w_{minor} are the severity weights for major and minor errors, respectively. In line with GEMBA-MQM and WMT Metric Shared Task, w_{major} and w_{minor} are set to 5 and 1, respectively. The maximum score for a single segment is capped at 25. If a critical error is identified, the score for that segment will automatically be 25.

E Applying TEaR in Different Models

To validate the generalizability of the TEaR framework, we selected three models with distinct capabilities: GPT-4-0613, GPT-3.5-turbo-0613, and Mistral-7B (Mistral-7B-Instruct-v0.2). We conducted experiments on a sample of 50 cases from the WMT23 Zh-En dataset, evaluating translation quality using COMET-22 (Rei et al., 2020) and BLEURT-20 (Sellam et al., 2020) metrics. Our results indicate that TEaR positively impacts models of varying sizes and capabilities.

Model	COMET		BLEURT	
	IT	TEaR	IT	TEaR
GPT-4-0613	82.35	82.38 (+0.03)	70.08	70.20 (+0.12)
GPT-3.5-turbo	81.60	82.02 (+0.42)	69.25	69.74 (+0.49)
Mistral-7B	76.54	76.58 (+0.04)	62.88	63.35 (+0.47)

Table 6: Comparison of different models using TEaR. *IT* represents *initial translation*. *TEaR* refers to the version after refining the initial translation using TEaR.

	Kendall (%)	Pearson (%)
zero-shot	11.93	8.91
few-shot	20.22	18.72

Table 7: The Kendall and Pearson correlation between zero/few-shot estimation scores (MQM typology) using GPT-3.5-turbo and gold scores.

F Impact of Estimation Module

To explore the impact of estimation quality in TEaR, we utilized GPT-4-0613 with human assistance as the gold standard evaluation (reference-based). We observed that few-shot estimation notably outperforms zero-shot one when calculating the correlation between their MQM scores and the gold scores, as detailed in Table 7. Figure 6 showcases involving feedback with better estimation quality can help improve the effect of TEaR in COMET. However, the modest correlation scores for few-shot estimation and minor enhancement in TEaR point to the estimation process as a critical

Estimate	Metric	Accuracy/ Mistranslation	Accuracy/ Omission	Accuracy/ Untranslated text	Accuracy/ Addition	Style/ Awkward	Fluency/ Grammar	Terminology/ Inappropriate	Locale Convention/ Name
zero-shot		+0.15	-0.70	+0.45	-0.18	+0.19	/	/	/
few-shot	ΔCOMET	+0.84	-1.16	+18.42	0.00	+1.09	+0.11	-0.10	/
GPT-4 w/ human		+3.34	+4.34	+31.09	/	+1.53	+4.14	+1.37	+1.98

Table 8: Relative COMET score improvements over initial translations (IT) when employing different estimation feedback strategies as in Figure 6. We splited the testing targets based on the classification of errors by estimation strategies. "/" indicates that no testing target was segmented under this error type.

Models	En-Ru		En-De		He-En		Zh-En	
	System(%)	Segment(%)	System(%)	Segment(%)	System(%)	Segment(%)	System(%)	Segment(%)
GPT-3.5-turbo	66.67	23.41	78.79	**34.48**	74.24	19.57	**90.48**	30.36
Gemini-Pro	62.86	19.32	**86.36**	26.64	83.33	30.98	80.95	21.36
Claude-2	**69.52**	**26.49**	83.33	30.33	**92.42**	**33.05**	88.57	**34.75**

Table 9: The system and segment level results of metrics by various LLMs using pairwise accuracy (%) and Kendall correlation (%) with human-annotated MQM scores, respectively. **Bold** results indicate the best in each section.

Models	BLEU	COMET	COMETKiwi	BLEURT	Rank
En-Ru					
GPT-3.5-turbo	36.95	87.94	83.47	75.78	2
Gemini-Pro	34.25	86.34	82.02	74.51	3
Claude-2	**37.72**	**88.90**	**84.15**	**77.04**	1
En-De					
GPT-3.5-turbo	**48.07**	**84.15**	**80.35**	**70.96**	1
Gemini-Pro	44.89	83.08	79.10	70.50	2
Claude-2	45.80	82.69	79.96	68.81	3
He-En					
GPT-3.5-turbo	**47.83**	86.00	82.16	76.00	3
Gemini-Pro	46.46	86.02	82.05	76.13	2
Claude-2	47.37	**86.55**	**82.61**	**76.34**	1
Zh-En					
GPT-3.5-turbo	**26.41**	**80.70**	**80.07**	**67.84**	1
Gemini-Pro	23.63	78.67	77.47	64.82	3
Claude-2	23.06	79.66	79.32	66.03	2

Table 10: The results of testing translation ability of various LLMs under few-shot setting in our sampled WMT23 datasets. **Bold** results indicate the best in each section. We rank different LLMs based on their scores from learned metrics.

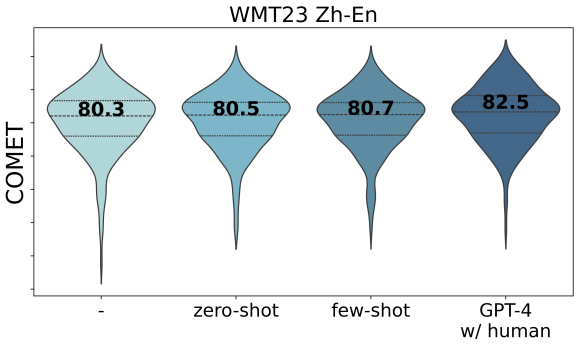

Figure 6: COMET scores for involving various feedback estimation strategies in the TEaR. We use GPT-3.5-turbo to execute the refinement. "-" denotes the initial translation (IT). *zero-shot* and *few-shot* reflect the use of different prompting methods with GPT-3.5-turbo, while *GPT-4 w/ human* indicates estimations made using GPT-4 with human assistance.

bottleneck in optimizing self-correction efficacy. Table 14 offers a typical case study. Table 8 further illustrates the growth in COMET scores after correcting different types of errors. When error types involve *Accuracy*, they usually pertain to higher severity levels of errors and be corrected more. Table 14 showcases different estimation strategies applied to the same initial translation. Weaker models and prompting methods tend to produce hallucinations and overestimate translation errors, leading to poorer downstream refinement.

G Potential Correlation between Translation and Estimation Capability

Traditional neural translation models like NLLB (Costa-jussà et al., 2022) and automatic metrics models COMET (Rei et al., 2020) operate separately, often focusing on singular tasks due to their limited capabilities, while general-purpose LLMs possess the capability to undertake both tasks simultaneously. An intriguing question arises: "Is there a correlation between the language proficiency of general-purpose LLMs and their translation evaluation capabilities?". We apply TEaR with three general-purpose LLMs, including GPT-3.5-turbo[5], Claude-2[6], and Gemini-Pro[7]. We use four automatic metrics: 1) a reference-based neural metric COMET-22 (Rei et al., 2020); 2) a reference-free quality estimation model

[5]We utilize gpt-3.5-turbo-0613 https://platform.openai.com/docs/models/gpt-3-5

[6]https://www.anthropic.com/index/claude-2

[7]https://cloud.google.com/vertex-ai/docs/generative-ai/learn/models

COMETKiwi (Rei et al., 2022); 3) a reference-based trained metric BLEURT-20 (Sellam et al., 2020); 4) a lexical metric SacreBLEU (Post, 2018) for completeness. Table 12 and 13 present details of the sampled test set we used. When comparing the translation estimation capabilities of different LLMs, we consider two dimensions used in WMT Shared Metrics task: system-level and segment-level. For the system-level, we utilize pairwise accuracy of system-ranking (Kocmi et al., 2021),

$$Acc = \frac{[sign(metric\Delta) == sign(human\Delta)]}{|all\ system\ pairs|}$$

where Δ represents the difference between the scores of the two systems. For the segment level, we follow Freitag et al. (2022) to adopt the average of three types of Kendall correlation across all translation pairs.

Table 10 and 9 present the results about how well different general-purpose LLMs do in translation and estimation, using the same prompting strategies and api parameters. We observed that GPT-3.5-turbo performs best in translation for En-De and Zh-En, while Claude-2 excels in En-De and He-En. The average rankings for GPT-3.5-turbo, Gemini-Pro, and Claude-2 are 1.75, 2.5, and 1.75, respectively. We also find that Claude-2 achieves the highest scores in both System-level and Segment-level evaluation for En-Ru and He-En.

	En-Ru	En-De	He-En	Zh-En	Avg
System-level $\mathcal{M}$	1	-0.33	1	1	0.67
Segment-level $\mathcal{M}$	1	0.33	1	0.33	0.67

Table 11: The Kendall correlation between translation and translation evaluation capabilities. System-/Segment-level $\mathcal{M}$ means using evaluation rankings based on System-/Segment-level.

As for En-De, the situation is somewhat complex, The ranking order exhibits significant differences between the system-level and segment-level evaluations. We hypothesize that the current MQM is primarily tailored for shorter sentences, potentially leading to reduced robustness when applied to longer paragraph-level tests. We consider both of these two levels in our subsequent analysis.

We further study the correlation between the translation and estimation capabilities of LLMs. We regard the translation rankings $\mathcal{R}_{\mathcal{M}^{x-y}_{translate}}$ from Table 10 and the estimation rankings $\mathcal{R}_{\mathcal{M}^{x-y}_{estimate}}$ from Table 9 to compute Kendall correlation. Table 11 highlights the consistency of translation and estimation capabilities in En-Ru and He-En, where the Kendall correlation scores are 1. This implies that models performing better in translation also tend to excel in evaluation. What's more, the consistency in En-De is not hypothetical, whether using system-level or segment-level evaluation metrics as a reference. This provides further evidence that using the existing MQM paradigm at paragraph level might not be robust.

Dataset	Language Pair	Domain types	Total Segments	Sampled Segments	#tokens per segment
WMT22	En-Ru	News, E-commerce	2016	200	16.38
WMT23	En-De	Social, News, Meeting notes, E-commerce	557	200	59.46
WMT23	He-En	Social, News	1910	200	21.66
WMT23	Zh-En	Manuals, News, E-commerce	1976	200	22.85

Table 12: Statistics of our testset. *#tokens per segment* indicates the average number of tokens of the translation pairs, calculated based on the sampled text from the English portion of either the source or reference. *Systems*: translation systems that are annotated in the WMT for the given year.

Dataset	Language Pair	Segments	Systems	Total Segments	Systems Selected
WMT22	En-Ru	59	15	885	HuaweiTSC, JDExploreAcademy, Lan-Bridge, M2M100_1.2B-B4, Online-A, Online-B, Online-G, Online-W, Online-Y, PROMT, QUARTZ_TuneReranking, SRPOL, bleu_bestmbr, comet_bestmbr, eTranslation
WMT23	En-De	80	12	960	AIRC, GPT4-5shot, Lan-BridgeMT, NLLB_Greedy, NLLB_MBR_BLEU, ONLINE-A, ONLINE-B, ONLINE-G, ONLINE-M, ONLINE-W, ONLINE-Y, ZengHuiMT
WMT23	He-En	80	12	960	GTCOM_Peter, GPT4-5shot, Lan-BridgeMT, NLLB_Greedy, NLLB_MBR_BLEU, ONLINE-A, ONLINE-B, ONLINE-G, ONLINE-Y, ZengHuiMT, Samsung_Research_Philippines, UvA-LTL
WMT23	Zh-En	80	15	1200	ANVITA, GPT4-5shot, Lan-BridgeMT, NLLB_Greedy, NLLB_MBR_BLEU, ONLINE-A, ONLINE-B, ONLINE-G, ONLINE-M, ONLINE-W, ONLINE-Y, ZengHuiMT, HW-TSC, IOL_Research, Yishu

Table 13: For MQM annotated datasets, we exclude data with missing annotations and sample 80 translation pairs form the former 200 sampled translation pairs, except for En-Ru, where the count is 59. The selected testing systems vary across different language pairs.

Case	
Source	我惊愕了，音质竟然是环绕3D立体！
Reference	I was astounded to discover that the sound quality was surround 3D!
IT	I was amazed that the sound quality was actually 3D surround!
Estimate (reference-free)	
0-shot, GPT-3.5-turbo	Based on the source segment and machine translation, the errors in the translation can be identified and classified as follows: {"critical": accuracy/mistranslation - 'was actually', "major": " "minor":" "}. The critical error is the mistranslated text. This error inhibits comprehension of the text.
3-shot, GPT-3.5-turbo	"critical": no-error, "major": accuracy/mistranslation - 'actually' for '竟然'", "minor": style/awkward - 'I was amazed that'"
GPT-4 w/ human	"critical": no-error, "major": no-error, "minor": style/awkward - 'I was amazed that'"
Refine (with feedback from estimation)	
Fdb (0-shot)	I was amazed that the sound quality was actually 3D surround!
Fdb (3-shot)	I was astonished that the sound quality was actually 3D surround!
Fdb (GPT-4 w/ human)	I was astonished that the sound quality was actually 3D surround!

Table 14: Case study of different estimation feedback. Top: translation defects annotated by human expert (in red); Middle: translation defects estimated successfully by models (in red), overestimated errors (under lines); Bottom: unsuccessfully corrected (in red); successfully corrected (in blue). *Fdb*: source of the estimation feedback.

MQM Annotator Guidelines
You will be assessing translations at the segment level, where a segment may contain one or more sentences. Each segment is aligned with a corresponding source segment, and both segments are displayed within their respective documents. Annotate segments in natural order, as if you were reading the document. You may return to revise previous segments. Please identify all errors within each translated segment, up to a maximum of five. If there are more than five errors, identify only the five most severe. If it is not possible to reliably identify distinct errors because the translation is too badly garbled or is unrelated to the source, then mark a single Non-translation error that spans the entire segment. To identify an error, highlight the relevant span of text, and select a category/sub-category and severity level from the available options. (The span of text may be in the source segment if the error is a source error or an omission.) When identifying errors, please be as fine-grained as possible. For example, if a sentence contains two words that are each mistranslated, two separate mistranslation errors should be recorded. If a single stretch of text contains multiple errors, you only need to indicate the one that is most severe. If all have the same severity, choose the first matching category listed in the error typology (eg, Accuracy, then Fluency, then Terminology, etc). Please pay particular attention to document context when annotating. If a translation might be questionable on its own but is fine in the context of the document, it should not be considered erroneous; conversely, if a translation might be acceptable in some context, but not within the current document, it should be marked as wrong. There are two special error categories: Source error and Non-translation. Source errors should be annotated separately, highlighting the relevant span in the source segment. They do not count against the 5-error limit for target errors, which should be handled in the usual way, whether or not they resulted from a source error. There can be at most one Non-translation error per segment, and it should span the entire segment. No other errors should be identified if Non-Translation is selected.

Table 15: MQM annotator guidelines

Error Category		Description
Accuracy	Addition	Translation includes information not present in the source.
	Omission	Translation is missing content from the source.
	Mistranslation	Translation does not accurately represent the source.
	Untranslated text	Source text has been left untranslated.
Fluency	Punctuation	Incorrect punctuation (for locale or style).
	Spelling	Incorrect spelling or capitalization.
	Grammar	Problems with grammar, other than orthography.
	Register	Wrong grammatical register (e.g., inappropriately informal pronouns).
	Inconsistency	Internal inconsistency (not related to terminology).
	Character encoding	Characters are garbled due to incorrect encoding.
Terminology	Inappropriate for context	Terminology is non-standard or does not fit context.
	Inconsistent use	Terminology is used inconsistently.
Style	Awkward	Translation has stylistic problems.
Locale convention	Address format	Wrong format for addresses.
	Currency format	Wrong format for currency.
	Date format	Wrong format for dates.
	Name format	Wrong format for names.
	Telephone format	Wrong format for telephone numbers.
	Time format	Wrong format for time expressions.
Other		Any other issues.
Source error		An error in the source.
Non-translation		Impossible to reliably characterize the 5 most severe errors.

Table 16: MQM hierarchy.

$\mathcal{T}_{translate}$
Please provide the {tgt_lan} translation for the {src_lan} sentences: Source: {origin} Target:

Table 17: **Translate Prompt** $\mathcal{T}_{translate}$. {tgt_lan}: target language; {src_lan}: source language; {origin}: the source test sentence.

$\mathcal{T}_{estimate}$

Please identify errors and assess the quality of the translation.
The categories of errors are accuracy (addition, mistranslation, omission, untranslated text), fluency (character encoding, grammar, inconsistency, punctuation, register, spelling),
locale convention (currency, date, name, telephone, or time format) style (awkward), terminology (inappropriate for context, inconsistent use), non-translation, other, or no-error.\n
Each error is classified as one of three categories: critical, major, and minor. Critical errors inhibit comprehension of the text. Major errors disrupt the flow, but what the text is trying to say is still understandable. Minor errors are technical errors but do not disrupt the flow or hinder comprehension.
Example1:
Chinese source: 大众点评乌鲁木齐家居商场频道为您提供居然之家地址，电话，营业时间等最新商户信息，找装修公司，就上大众点评
English translation: Urumqi Home Furnishing Store Channel provides you with the latest business information such as the address, telephone number, business hours, etc., of high-speed rail, and find a decoration company, and go to the reviews.
MQM annotations:
critical: accuracy/addition - "of high-speed rail"
major: accuracy/mistranslation - "go to the reviews"
minor: style/awkward - "etc.,"
Example2:
English source: I do apologise about this, we must gain permission from the account holder to discuss an order with another person, I apologise if this was done previously, however, I would not be able to discuss this with yourself without the account holders permission.
German translation: Ich entschuldige mich dafür, wir müssen die Erlaubnis einholen, um eine Bestellung mit einer anderen Person zu besprechen. Ich entschuldige mich, falls dies zuvor geschehen wäre, aber ohne die Erlaubnis des Kontoinhabers wäre ich nicht in der Lage, dies mit dir involvement.
MQM annotations:
critical: no-error
major: accuracy/mistranslation - "involvement"
    accuracy/omission - "the account holder"
minor: fluency/grammar - "wäre"
    fluency/register - "dir"
Example3:
English source: Talks have resumed in Vienna to try to revive the nuclear pact, with both sides trying to gauge the prospects of success after the latest exchanges in the stop-start negotiations.
Czech transation: Ve Vídni se ve Vídni obnovily rozhovory o oživení jaderného paktu, přičemže obě partaje se snaží posoudit vyhlídky na úspěch po posledních výměnách v jednáních.
MQM annotations:
critical: no-error
major: accuracy/addition - "ve Vídni"
    accuracy/omission - "the stop-start"
minor: terminology/inappropriate for context - "partake"

{src_lan} source: {origin}
{tgt_lan} translation: {init_trans}
MQM annotations:

Table 18: **Estimate Prompt** $\mathcal{T}_{estimate}$. {src_lan}: source language; {origin}: the source test sentence; {tgt_lan}: target language; {init_trans}: the initial translation of the source test sentence.

$\mathcal{T}_{refine}$
Please provide the {tgt_lan} translation for the {src_lan} sentences. Source: {raw_src} Target: {raw_mt} I'm not satisfied with this target, because some defects exist: {estimate_fdb} Critical errors inhibit comprehension of the text. Major errors disrupt the flow, but what the text is trying to say is still understandable. Minor errors are technical errors but do not disrupt the flow or hinder comprehension. Upon reviewing the translation examples and error information, please proceed to compose the final {tgt_lan} translation to the sentence: {raw_src}. First, based on the defects information locate the error span in the target segment, comprehend its nature, and rectify it. Then, imagine yourself as a native {tgt_lan} speaker, ensuring that the rectified target segment is not only precise but also faithful to the source segment.

Table 19: **Refine Prompt** $\mathcal{T}_{refine}$. {tgt_lan}: target language; {src_lan}: source language; {raw_src}: the source test sentence; {raw_mt}: the initial translation of the source test sentence; {estimate_fdb}: the estimation feedback.

Contrastive Translation
Please provide the {tgt_lan} translation for the {src_lan} sentences. Source: {raw_src} **Bad** Target: {raw_mt} Please give me a better translation without any explanation.

Table 20: **Refine Prompt CT**. {tgt_lan}: target language; {src_lan}: source language; {raw_src}: the source test sentence; {raw_mt}: the initial translation of the source test sentence.

Structured CoT

Please provide the {tgt_lan} translation for the {src_lan} sentences.
You will work as a machine translation annotator to help assess the quality of translation: Please identify all errors within each translated sentence, up to a maximum of five. If there are more than five errors, identify only the five most severe. To identify an error, specify the relevant span of text, and select a category/sub-category and severity level from the available options. (The span of text may be in the source sentence if the error is a source error or an omission.) When identifying errors, please be as fine-grained as possible. For example, if a sentence contains two words that are each mistranslated, two separate mistranslation errors should be recorded. If a single stretch of text contains multiple errors, you only need to indicate the one that is most severe. If all have the same severity, choose the first matching category listed in the error typology (eg, Accuracy, then Fluency, then Terminology, etc). Be very precise and accurate. If there is an error in translation, identify the severity of the error as follows: Major: Errors that may confuse or mislead the reader due to significant change in meaning or because they appear in a visible or important part of the content. Minor: Errors that don't lead to loss of meaning and wouldn't confuse or mislead the reader but would be noticed, would decrease stylistic quality, fluency or clarity, or would make the content less appealing. Neutral: Use to log additional information, problems or changes to be made that don't count as errors, e.g., they reflect a reviewer's choice or preferred style. If there is an error in translation, try to place it in a category below. If it doesn't match any of those categories, place it as an Other error: 1. Accuracy: there is an error with the translation accuracy, if it matches any of the following categories: Accuracy/Addition: Translation includes information not present in the source. Accuracy/Omission: Translation is missing content from the source. Accuracy/Mistranslation: Translation does not accurately represent the source. Accuracy/Untranslated text: Source text has been left untranslated. 2. Fluency: there is an error with the translation fluency, if it matches any of the following categories: Fluency/Punctuation: Incorrect punctuation (for locale or style). Fluency/Spelling: Incorrect spelling or capitalization. Fluency/Grammar: Problems with grammar, other than orthography Fluency/Register: Wrong grammatical register (e.g., inappropriately informal pronouns). Fluency/Inconsistency: Internal inconsistency. Fluency/Character encoding: Characters are garbled due to incorrect encoding. 3. Terminology: Terminology is inappropriate or inconsistent: Terminology/Inappropriate: Terminology is non-standard or does not fit context. Terminology/Inconsistent: Terminology is used inconsistently. 4. Style: Translation is awkward with stylistic problems. 5. Locale convention: Wrong format for addresses, currency, dates, names, telephone numbers or time expressions. Locale/Address: Wrong format for addresses. Locale/Currency: Wrong format for currency. Locale/Date: Wrong format for dates. Locale/Name: Wrong format for names. Locale/Telephone: Wrong format for telephone numbers. Locale/Time: Wrong format for time expressions. After identifying all the errors, you will produce an improved translation that fixes the identified errors. For the improvements made to the translation, you make sure that the following principles are followed: 1. No corrections are made that add any word or phrase in the translation which are unsupported in the input 2. The capitalizations in the translation strictly follow the input capitalizations, e.g., acronym capitalizations should not be changed 3. The translation contains the appropriate articles and determiners to follow the specifics in the input 4. Do not leave any symbol, word or phrase in the input text untranslated in the final, improved translation 5. Do not add any extraneous words, phrases, clauses or sentences in the translation that is not supported by the input 6. If the input starts with a non capitalized word, the translation starts with a non capitalized word 7. In the case that the translation is severely inadequate, you generate an improved translation from scratch 8. No end punctuations or full stops are added if such punctuations or full stops are not in the input 9. Do not assume that an acronym is a typo, always err on the side of assuming that the presented input words are not typos 10. Do not replace any entities or placeholders in the translation with fictitious (unsupported) entities 11. If the input contains offensive or lewd words, you still translate them faithfully 12. If the translation misses to convey the meaning of a large part of the input sentence, you include the translation for the missing part As an expert translation post editor, your task is to improve the {tgt_lan} translation for the below {src_lan} text: Source: {raw_src}
Target: {raw_mt}
To accomplish this, follow these steps: Step 1: Say "Proposed Improvements:". Then brainstorm and design the improvements that make the {tgt_lan} translation more faithful and fluent. Step 2: Say "Improved Translation:". Then output the {tgt_lan} translation with proposed improvements that increase translation faithfulness and fluency.

Table 21: **Refine Prompt SCoT**. {tgt_lan}: target language; {src_lan}: source language; {raw_src}: the source test sentence; {raw_mt}: the initial translation of the source test sentence.

XCOMET-Score
Please provide the {tgt_lan} translation for the {src_lan} sentences. Source: {raw_src} Target: {raw_mt} Its COMET score is {comet_score}. Please give me the final target translation that might have a higher COMET score.

Table 22: **Refine Prompt XCOMET-Score**. {tgt_lan}: target language; {src_lan}: source language; {raw_src}: the source test sentence; {raw_mt}: the initial translation of the source test sentence; {comet_score}: the XCOMET score.

XCOMET-Span
Please provide the {tgt_lan} translation for the {src_lan} sentences. Source: {raw_src} Target: {raw_mt} I'm not satisfied with this target, because some defects exist: {xcomet_span} Critical errors inhibit comprehension of the text. Major errors disrupt the flow, but what the text is trying to say is still understandable. Minor errors are technical errors but do not disrupt the flow or hinder comprehension. Upon reviewing the translation examples and error information, please proceed to compose the final {tgt_lan} translation.

Table 23: **Refine Prompt XCOMET-Span**. {tgt_lan}: target language; {src_lan}: source language; {raw_src}: the source test sentence; {raw_mt}: the initial translation of the source test sentence; {xcomet_span}: the error spans marked by XCOMET-XL, only severity involved.