MQM-MSC: Enhancing Translation Quality Estimation Interpretability with Mask-Driven Self-Correction in Large Language Models

Guanghui Cai[1,2], **Junguo Zhu**[1,2†]
1. Faculty of Information Engineering and Automation,
Kunming University of Science and Technology, Kunming, Yunnan, 650500, China
2. Yunnan Key Laboratory of Artificial Intelligence,
Kunming University of Science and Technology, Kunming, Yunnan, 650500, China
gh.cai.kust@qq.com, jg.zhu.hit@qq.com

Abstract

Large Language Models (LLMs) have demonstrated significant potential in interpretable translation quality estimation by providing both holistic ratings and fine-grained feedback. However, state-of-the-art methods, such as GEMBA-MQM, still suffer from an excessive number of false positives in error prediction, leading to misalignment with human annotations and reducing interpretability. To address this issue, we propose MQM-MSC, a novel training-free framework that employs a mask-driven self-correction (MSC) mechanism. The core of MSC is to use masks to highlight error spans in the initial prediction, enabling the model to re-evaluate these masked portions and verify their correctness. This approach mirrors human cognitive processes: when individuals express inconsistent judgments about the same issue at different times, it often indicates that their initial assessment was flawed. Similarly, MSC exploits contradictions between two evaluations to identify and filter false positives, thereby improving the accuracy and reliability of error annotations. Experimental results show that MQM-MSC effectively reduces false positives across four LLMs and three language pairs, consistently improving the reliability and quality of error annotations in the GEMBA-MQM approach.

1 Introduction

Machine Translation Quality Estimation (QE) assesses translations in real time without requiring reference translations, distinguishing it from traditional metrics such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), TER (Snover et al., 2006), and CHRF, which depend on references. This capability makes QE essential for practical applications where reference translations are unavailable. By providing independent evaluations of translation quality, QE enables users to assess translation accuracy more effectively, helps developers measure system performance reliably. QE plays a critical role in the development, commercialization, and deployment of machine translation.

In recent years, QE mothods based on pre-trained models have achieved significant advancements. However, these models are limited to providing only overall quality scores for sentences and are unable to identify specific translation errors as human evaluators can. Given the high cost of human evaluation, the demand for interpretable QE methods has become increasingly urgent (Leiter et al., 2023; Xu et al., 2023). With the successful application of LLMs in reasoning and generation tasks, leveraging their powerful capabilities to enable interpretable QE has become a promising direction. Researchers have actively explored this area, including pioneering approaches such as EAPrompt (Lu et al., 2023) and GEMBA-MQM (Kocmi and Federmann, 2023a). By incorporating chain-of-thought prompts, these methods construct an evaluation system similar to the Multidimensional Quality Metrics (MQM) (Freitag et al., 2021a), an error-based human evaluation framework. Despite achieving state-of-the-art results at the system level, these methods frequently generate excessive false positives. This discrepancy between LLMs and human annotations undermines the reliability and faithfulness of error annotations as explanations. Additionally, training-dependent methods (Xu et al., 2023; Guerreiro et al., 2024) face high computational costs, limiting their applicability across different models and languages.

[†] Corresponding author

Proceedings of the 24th China National Conference on Computational Linguistics, pages 880-889, Jinan, China, August 11-14, 2025.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

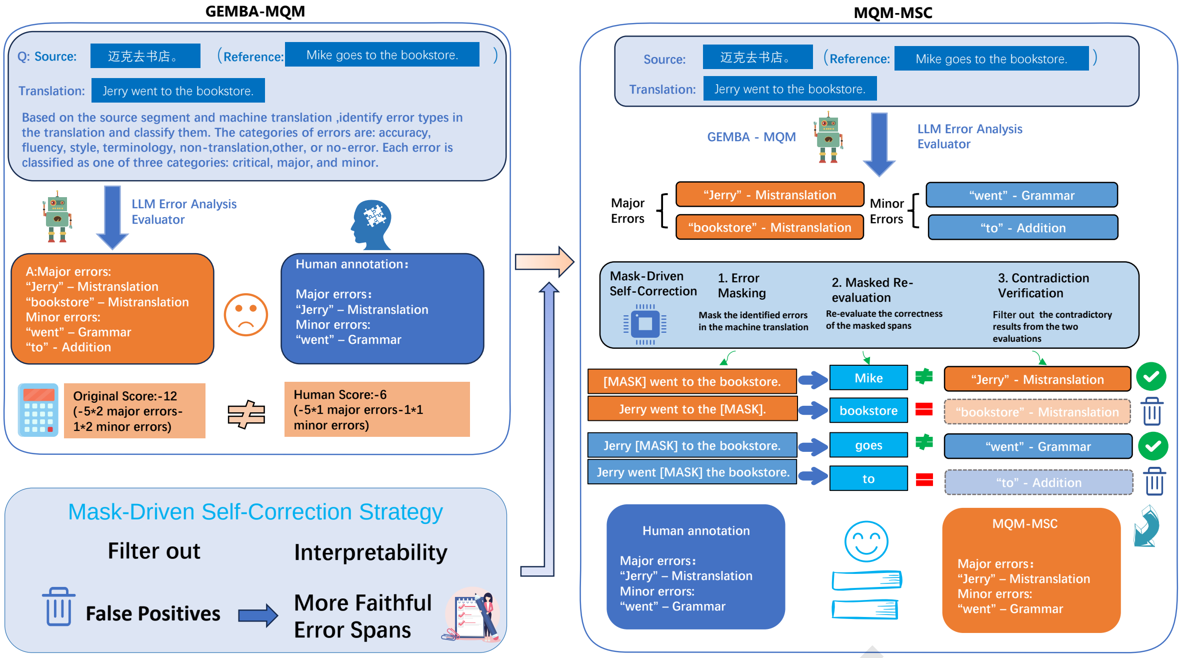
GEMBA-MQM
Q: Source: 迈克去书店。 （Reference: Mike goes to the bookstore. ）
Translation: Jerry went to the bookstore.
Based on the source segment and machine translation ,identify error types in the translation and classify them. The categories of errors are: accuracy, fluency, style, terminology, non-translation,other, or no-error. Each error is classified as one of three categories: critical, major, and minor.
LLM Error Analysis Evaluator
A:Major errors:
"Jerry" – Mistranslation
"bookstore" – Mistranslation
Minor errors:
"went" – Grammar
"to" - Addition
Human annotation：
Major errors：
"Jerry" – Mistranslation
Minor errors:
"went" – Grammar
Original Score:-12 (-5*2 major errors-1*2 minor errors)
≠
Human Score:-6 (-5*1 major errors-1*1 minor errors)
Mask-Driven Self-Correction Strategy
Filter out
Interpretability
False Positives
More Faithful Error Spans
MQM-MSC
Source: 迈克去书店。 （Reference: Mike goes to the bookstore. ）
Translation: Jerry went to the bookstore.
GEMBA - MQM
LLM Error Analysis Evaluator
Major Errors
"Jerry" - Mistranslation
"bookstore" - Mistranslation
Minor Errors
"went" - Grammar
"to" - Addition
Mask-Driven Self-Correction
1. Error Masking
Mask the identified errors in the machine translation
2. Masked Re-evaluation
Re-evaluate the correctness of the masked spans
3. Contradiction Verification
Filter out the contradictory results from the two evaluations
[MASK] went to the bookstore.
Mike
"Jerry" - Mistranslation
Jerry went to the [MASK].
bookstore
"bookstore" - Mistranslation
Jerry [MASK] to the bookstore.
goes
"went" - Grammar
Jerry went [MASK] the bookstore.
to
"to" - Addition
Human annotation
Major errors：
"Jerry" – Mistranslation
Minor errors:
"went" – Grammar
MQM-MSC
Major errors：
"Jerry" – Mistranslation
Minor errors:
"went" – Grammar

Figure 1: **A comparative overview between GEMBA-MQM and our MQM-MSC approach .** MQM-MSC approach introduces four-stage workflow: (1)generate error annotations; (2) masking errors;(3) re-evaluating masked segments;and (4) verifying contradictions to filter false positives.

To address the aforementioned challenges, we innovatively proposes a mask-driven self-correction method and constructs a universal training-free framework: MQM-MSC. The framework activates the intrinsic self-correction capabilities of LLM through a masking mechanism, filters out false positives, and improves the quality of error annotation with minimal additional overhead. Specifically, as shown in Fig. 1, the workflow of the framework is as follows: **(1) Error Analysis Evaluation**—performing an initial evaluation of the translation and generating error annotations; **(2) Error Masking**—masking identified errors in translation to focus the LLM's attention; **(3) Masked Re-evaluation**—providing the source and masked translation for the LLM to re-evaluate the correctness of masked spans; **(4) Contradiction Verification**—comparing initial and re-evaluation results to retain consistent high-confidence errors while filtering inconsistent low-confidence errors.

We conducted extensive experiments on the WMT22 test set using four different LLMs. This test set comprises 106,758 segments from 54 machine translation systems. Our research contributions are as follows:

- MQM-MSC generally outperforms GEMBA-MQM at both the system and segment levels, providing interpretable error annotations that closely align with human annotations.
- MQM-MSC achieves higher consistency with human annotations in both the number and categories of error annotations while introducing acceptable additional costs.
- MQM-MSC validates the feasibility of the mask-driven intrinsic self-correction mechanism for QE tasks, offering novel insights into the intrinsic self-correction of LLMs.

2 Related Work

2.1 Quality Estimation Based on Large Language Models

Quality Estimation (QE) is an essential research area within machine translation, aiming to assess the quality of translation outputs in real-time without relying on reference translations. The rapid advancement of Large Language Models (LLMs) has demonstrated significant potential for QE applications.

GEMBA (Kocmi and Federmann, 2023b), the first QE metric based on LLMs, directly predicts translation quality scores using single-step prompting, achieving outstanding results in system-level evaluations. Nonetheless, GEMBA does not overcome the interpretability limitations inherent in pre-trained models; it primarily focuses on overall quality assessment and lacks the capability to provide detailed error analysis.

To address this deficiency, the Error Analysis Prompt method (Lu et al., 2023) was introduced, combining Chain-of-Thought and Error Analysis to guide LLMs in conducting more fine-grained error analysis through detailed prompts. Similarly, the GEMBA-MQM method (Kocmi and Federmann, 2023a) employs this error analysis approach and specifically optimizes it for the QE domain, using a fixed three-shot prompting technique to generate error annotations that align with MQM standards. Additionally, by using fixed language prompts, this method eliminates the need for manual prompt design for new languages, thus enhancing the scalability of the approach. The GEMBA-MQM method also serves as a baseline and a crucial component of the MQM-MSC.

2.2 Self-Correction

Self-correction, a widely studied approach, leverages LLMs to refine their outputs during inference (Madaan et al., 2023). It has been applied across diverse tasks, including arithmetic reasoning, code generation, and question-answering (Shinn et al., 2023). The simplest form involves prompting LLMs to evaluate and improve their own responses, operating on the premise that error identification is easier than error avoidance. However, recent research (Huang et al., 2023; Gou et al., 2023) challenges the inherent self-correction capabilities of LLMs, demonstrating their limitations in certain tasks.

To address this challenge, Wu et al. (Wu et al., 2024) developed Progressive Correction (PROCO), a self-correction framework for LLMs that employs substitute verification - a process where critical problem conditions are masked and verification questions are generated from the model's initial responses to systematically validate answer correctness. Inspired by this approach, we propose a mask-driven self-correction method. Our method uniquely leverages the divergence between initial annotations and masked re-evaluations as reliability signals, implementing dual-validation consistency checks to achieve precise self-correction.

3 Preliminary: Multidimensional Quality Metrics

The MQM (Multidimensional Quality Metrics) framework is a high-quality human evaluation methodology designed to identify and classify translation errors through detailed error analysis (Lommel et al., 2013; Freitag et al., 2021b). Unlike traditional Direct Assessment (DA), which assigns holistic scores on a 0–100 scale, MQM emphasizes the detection of specific translation errors, categorizing them by severity and type. Human annotators evaluate translations segment by segment, taking into account the broader document context. Each error is assigned a severity level—critical, major, or minor—and labeled according to its error category.

Compared to DA, MQM provides a more fine-grained and structured evaluation. The MQM framework automatically derives quality scores by applying a weighted scheme based on the severity of identified errors. Sentence-level scores range from 0 (perfect translation) to -25 (potentially the worst translation), with the overall score computed as the average across all annotators. For certain use cases, such as metric correlation studies, scores may be reversed to align with other evaluation frameworks. MQM has been shown to align more closely with human judgment than DA, providing a reliable and interpretable framework for assessing machine translation quality (Freitag et al., 2022a; Zhao et al., 2024).

4 Methodology

As shown in Fig 1, our proposed MQM-MSC method follows a four-stage workflow: **1.Error Analysis Evaluation:** evaluate the given source x and translation y, identify errors in y, and provide error annotations $\mathcal{E}$ that include error spans and severities. **2. Error Masking:** We mask the error spans identified in the translation y. **3. Masked Re-evaluation:** We input source x and masked translation $y^{(\text{mask})}$,and prompt the model to re-evaluate the correctness of that masked spans,thereby obtaining the right spans

$\mathcal{R}$. **4. Contradiction Verification:** We filter out false positives based on the contradiction between $\mathcal{E}$ and $\mathcal{R}$. retaining a refined set of errors $\mathcal{E}^* \subseteq \mathcal{E}$. Finally, we score the translation based on the refined error set $\mathcal{E}^*$.

4.1 Error Analysis Evaluation

We employ the GEMBA-MQM method (Kocmi and Federmann, 2023a) to prompt the LLM to perform MQM-like evaluation, identifying errors in translation y of source x. This step can be described as:

$$\mathcal{E} = \text{Evaluation}(x, y) \tag{1}$$

where $\mathcal{E} = \{e_1, e_2, \cdots, e_N\}$, represents the set of errors identified by the evaluation of LLMs, and N denotes the number of errors.

Each error annotation includes three pieces of information: the span of the error, indicating its position in y ; the error category, categorized according to MQM standards (e.g., mistranslation, omission, grammar); and the severity level, labeled as ”Critical”, ”Major”, or ”Minor” to represent descending levels of impact. Translations without identified errors are excluded from subsequent steps.

4.2 Mask-Driven Self-Correction

The mask-driven self-correction method identifies reliable error annotations via a dual-verification process: it performs contradiction verification between the initial evaluation and the masked re-evaluation. Consistent outputs indicate high-confidence error annotations, whereas discrepancies suggest potential false positives. The method primarily consists of the following three components:

4.2.1 Error Masking

We mask the identified error e_i in translation y using regular expressions ,focusing the LLM's attention on the masked span while minimizing potential interference from other sentence elements. For instance,as shown in Fig 1, the span ”Jerry” can be used to mask the translation of ”Jerry went to the bookstore” to ”[MASK] went to the bookstore”. This process can be represented as:

$$y_i^{(\text{mask})} = \text{MASK}(y, e_i), \quad i = 1, 2, \cdots, N \tag{2}$$

where a set of masked translations $\mathcal{Y}_{\text{mask}} = \{y_1^{(\text{mask})}, y_2^{(\text{mask})}, \cdots, y_N^{(\text{mask})}\}$ is produced.

4.2.2 Masked Re-evaluation

We provide the LLM with both the source x and the masked translation $y_i^{(\text{mask})}$, explicitly indicating the original span e_i. The LLM is then prompted to re-evaluate the correctness of these spans. If the original spans are deemed correct, they are output directly; otherwise, the LLM generates corrected spans. Formally, this process can be represented as:

$$\mathcal{R} = \text{Re-evaluation}(x, y_i^{(\text{mask})}, e_i), \tag{3}$$

where a new set of right spans $\mathcal{R} = \{r_1, r_2, \cdots, r_N\}$ is determined by the masked re-evaluation of LLMs,and N denotes the number of spans.

4.2.3 Contradiction Verification

Based on the principle of confidence,we perform contradiction verification between the initial errors set $\mathcal{E}$ and the re-evaluated right spans set $\mathcal{R}$. Inconsistent results ($e_i \in \mathcal{E} = r_i \in \mathcal{R}$)—where the initial evaluation is deemed incorrect but the re-evaluation confirms correctness—are regarded as low-confidence false positives and thus filtered out. Conversely, consistent results ($e_i \in \mathcal{E} \neq r_i \in \mathcal{R}$) identified as high-confidence results and retained. This step is expressed as:

$$\mathcal{E}^* = \{e_i \mid \text{Verification}[(x, y, e_i) \neq (x, y, r_i)]\} \tag{4}$$

where:

$$e_i \in \mathcal{E}, r_i \in \mathcal{R} \tag{5}$$

where a new subset of errors $\mathcal{E}^* \subseteq \mathcal{E}$ that has passed the contradiction verification,with false positives effectively filtered out.

4.3 Post-process: Error-Based Scoring

We follow the MQM weighting framework (Freitag et al., 2021a) to assign human evaluation scores to errors produced by LLMs, in line with prior studies (Kocmi and Federmann, 2023a; Lu et al., 2023). The final score is calculated as the weighted sum of different error types:

$$\text{Score} = -25N_{\text{critical}} - 5N_{\text{major}} - N_{\text{minor}} \tag{6}$$

where $N_{critical}$, N_{major}, and N_{minor} denote the number of critical, major, and minor errors, respectively. To prevent the LLMs from predicting an excessive number of errors that would result in an unreasonably low score, we follow previous work (Lu et al., 2024) by setting a minimum score threshold of -25.

5 Experiments

5.1 Experimental Setup

5.1.1 Dataset

We utilize the test set from the WMT22 shared tasks (Freitag et al., 2022b) in English-German (En-De), English-Russian (En-Ru), and Chinese-English (Zh-En) across 4 different domains - conversational, e-commerce, news, and social, with expert human annotations. This study evaluates 106,758 segments from 54 MT systems. For further information, refer to Table 1.

Dataset	Language Pair	Segments	Systems	Domains
WMT22	En-De	2037	17	conversational, e-commerce, news, social
	En-Ru	2037	17	
	Zh-En	1875	20	

Table 1: **Statistics of testset.** Source and translations are from the WMT22 metrics shared task.

5.1.2 Meta Evaluation

We follow the standard meta-evaluation approach to measure the performance of evaluation metrics (Freitag et al., 2023). At the system level, we use pairwise accuracy across all three language pairs, which calculates the proportion of all possible pairs of MT systems that are ranked the same by the metric and human scores (Kocmi et al., 2021). At the segment level, we apply group-by-item pairwise accuracy with tie calibration (Deutsch et al., 2023). using the acc^*_{eq} variant to compare metric and human score vectors per segment before averaging outcomes. For reproductivity,all meta-evaluations are calculated with MTME, the WMT-recommended evaluation toolkit (Freitag et al., 2022b).

5.2 Baselines and Large Language Models

5.2.1 Baseline Metrics

we compare our method with three model-based QE metrics: COMET-QE (Rei et al., 2020), UniTE-src (Wan et al., 2022), and MaTESe-QE (Perrella et al., 2022). These metrics employ supervised neural networks and multilingual encoders to evaluate machine translation quality without relying on reference translations. Our primary baseline remains GEMBA-MQM (Kocmi and Federmann, 2023a). To evaluate the effectiveness of our method in filtering false positives, we introduce a random error filtering baseline for comparison. Specifically, since performance improvements may stem from simply reducing the number of errors, we compare our method with a baseline that filters errors at random. The number of errors filtered by the baseline is matched to the average number filtered by our method across all systems for the given language pair.

5.2.2 Large Language Models

We employed the following models: Llama3.1-8B-Instruct, GPT-4o-Mini, and Qwen2.5. The Llama3.1-8B-Instruct model is a model optimized for instruction-following tasks, enabling it to better execute user commands. GPT-4o-Mini is a small-sized model within the GPT-4o series. It is fast, cost-effective, and highly capable. We experiment with it using the OpenAI API. Qwen2.5 is a series of large language models recently released by Alibaba Cloud. This series of models provides multilingual support. We test Qwen2.5-7B-Instruct and Qwen2.5-14B-Instruct models.

5.3 Main Results

Models	**Strategy**	**System-Level Acc.**	**Segment-Level Acc***			
		All (3 LPs)	**En-De**	**En-Ru**	**Zh-En**	**Avg**
Baselines	COMET-QE	78.1	55.5	53.4	48.3	52.4
	UniTE-src	75.9	58.2	55.4	50.8	54.8
	MaTESc-QE	74.8	57.2	49.9	49.4	52.2
Qwen2.5-7B-Inst	GEMBA-MQM	80.3	53.9	49.0	44.2	49.0
	Random	75.5	53.7	46.1	44.7	48.2
	MQM-MSC	82.8(+2.5)	54.0	48.8	46.7	49.8(+0.8)
Llama3.1-8b-inst	GEMBA-MQM	76.3	54.3	47.7	45.9	49.3
	Random	73.7	53.7	46.4	45.7	48.6
	MQM-MSC	79.2 (+2.9)	54.3	48.3	46.6	49.7(+0.4)
GPT-4o-mini	GEMBA-MQM	83.6	55.9	53.2	49.3	52.8
	Random	79.9	53.8	49.3	46.8	50.0
	MQM-MSC	87.6 (+4)	55.7	53.7	49.3	52.9(+0.1)
Qwen2.5-14B-Inst	GEMBA-MQM	88.7	56.1	52.5	47.5	52.0
	Random	86.9	54.2	48.7	47.2	50.0
	MQM-MSC	88.3(-0.4)	56.1	52.6	48.7	52.5(+0.5)

Table 2: **The performance of metrics using pairwise accuracy (%) at the system level and pairwise accuracy with tie calibration (%) at the segment level.** All results are compared with human-annotated MQM scores.

The results presented in Table 2 for WMT22, the experimental results demonstrate that in segment-level evaluation, the MQM-MSC outperforms the state-of-the-art GEMBA-MQM method across all models. In system-level evaluation, we method also shows superior performance on three out of the four models. These results validate the broad applicability and stable performance advantages of MQM-MSC across multiple language pairs (En-De, En-Ru, Zh-En). The only exception is the Qwen2.5-14B-Inst model, which shows a decline in system-level performance. Analysis indicates that this is primarily due to the model's subpar performance in evaluating the En-Ru language pair, possibly related to its limited judgment ability for errors in this language pair. In contrast, the random filtering method consistently degrades the performance of GEMBA-MQM. This indicates that MQM-MSC is capable of effectively filtering out false positives while retaining errors that are consistent with human annotations.

Furthermore, compared with model-based baseline metrics, methods based on LLMs consistently outperform the baselines at the system level, which is consistent with previous studies. At the segment level, despite the existing gap between LLMs-based methods and baseline metrics, the error annotations generated by LLMs demonstrate their potential. Our approach aims to enhance the reliability and faithfulness of these error annotations.

6 Analysis

6.1 MQM-MSC enhances alignment between error annotations and human annotations

Models	Category	Total Errors			Average Errors		
		En-De	En-Ru	Zh-En	En-De	En-Ru	Zh-En
Human annotations		16087	23946	31597	946.3	1408.6	1579.9
Qwen2.5-7B-Inst	Original	169029	160063	194297	9942.9	9415.5	9714.9
	Filtered	139858	129359	151759	8226.9	7609.4	7590.0
	Retained	29171	30704	42538	1715.9	1806.1	2126.9
Llama3.1-8b-inst	Original	56612	59562	84405	3330.1	3503.7	4420.3
	Filtered	35817	30746	39830	2106.9	1808.6	1991.5
	Retained	20795	28816	44575	1223.2	1695.1	2228.8
GPT-4o-mini	Original	36267	43633	68511	2133.4	2566.7	3425.6
	Filtered	25077	31561	34767	1475.1	1856.5	1738.4
	Retained	11190	12072	33744	658.2	710.1	1687.2
Qwen2.5-14B-Inst	Original	67118	74792	134610	3948.1	4399.5	6730.5
	Filtered	50277	52661	84089	2957.5	3097.7	4204.5
	Retained	16841	22131	50521	990.65	1301.8	2526.1

Table 3: Comparison of error annotation quantities of various models on three language pairs under the MQM-MSC framework: Statistics on the total and average of original, filtered, and retained errors

We counted the number of error annotations generated by each LLM for each language pair under the GEMBA-MQM method, as well as the number of false positives filtered and the remaining error annotations after applying the MQM-MSC framework. Specifically, we calculated the total and average number of error annotations generated by LLMs and human annotations across all language pairs, with the results detailed in Table 3.

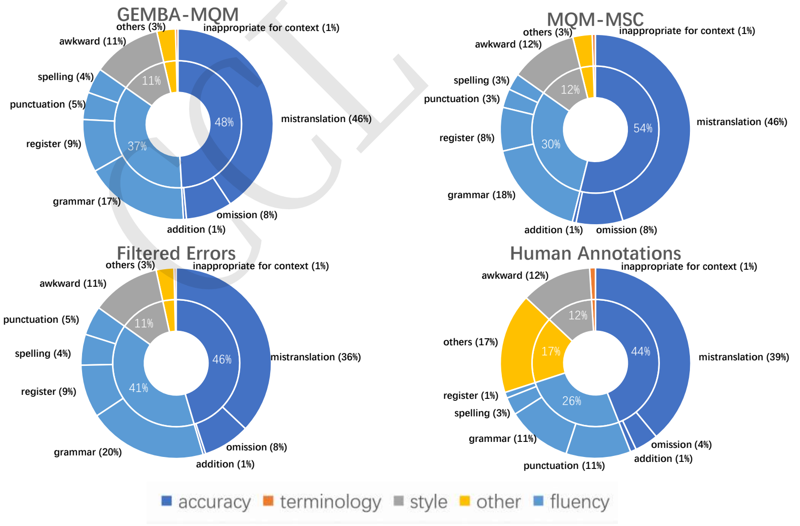

Figure 2: **Distribution of error categories** generated from GEMBA-MQM , MQM-MSC,filtered errors, and human-annotated MQM, respectively.

The results indicate that the number of error annotations generated by LLMs significantly exceeded human annotations, including many false positives. After filtering with the MQM-MSC framework, false positives were reduced by over 50% of the total, bringing the actual number of error annotations closer to the level of human annotations. Moreover, as shown in Fig 2, our method preserved the error categories distribution consistent with human annotations. In the GEMBA-MQM method, the proportion of ”fluency” errors is significantly higher than that human annotations. Our method reduces this proportion to a level that is closer to human annotations. Due to the richness of human annotations, a considerable portion of error types that LLMs fail to predict are categorized as ”other.” Consequently, as the proportion of fluency errors decreases, the proportion of mistranslation errors correspondingly increases.

6.2 MQM-MSC vs. Direct self-correction

Models	**Strategy**	**System-Level Acc.**	**Segment-Level Acc***			
		All (3 LPs)	**En-De**	**En-Ru**	**Zh-En**	**Avg**
Llama3.1-8b-inst	GEMBA-MQM	76.3	54.3	47.7	45.9	49.3
	Direct	75.9(-0.4)	54.3	47.6	46.1	49.3(0.0)
	MQM-MSC	79.2(+2.9)	54.3	48.3	46.6	49.7(+0.4)
GPT-4o-mini	GEMBA-MQM	83.6	55.9	53.2	49.3	52.8
	Direct	82.1(-1.5)	55.8	52.9	48.9	52.5(-0.3)
	MQM-MSC	87.6 (+4)	55.7	53.7	49.3	52.9(+0.1)

Table 4: Comparison of performance between direct self-correction and mask-driven self-correction on WMT22 with human-labeled MQM, evaluated using pairwise accuracy (%) at the system level, pairwise accuracy with tie calibration (%) at the segment level

Models	**Strategy**	**Filtered Errors(%)**		
		En-De	**En-Ru**	**Zh-En**
Llama3.1-8b-inst	Masked	63.3%	51.6%	47.2%
	Direct	19.8%	10.6%	11.7%
GPT-4o-mini	Masked	69.1%	72.3%	50.7%
	Direct	15.5%	12.6%	10.6%

Table 5: Comparison of the proportion (relative to original errors) of false positives filtered by direct self-correction and mask-driven self-correction.

To validate the necessity of the mask-driven mechanism, we conducted an ablation study comparing it with a direct self-correction approach that omits the masking operation. Due to computational constraints, experiments were performed on two large language models (Llama-3.1-8B-Instruct and GPT-4o-mini). As shown in Table 4, the direct self-correction underperforms the mask-driven self-correction at both the system and segment levels. Furthermore, Table 5 reveals that the direct self-correction method is significantly less effective in filtering false positives,filtering significantly fewer false positives compared to the mask-driven method. These results suggest that without positional constraints from masking, the model’s attention becomes dispersed across sentence components, making it prone to interference from other parts of the sentence. This leads to excessive misjudgments.

Our findings demonstrate that in translation quality estimation, the mask-driven mechanism is essential for effective self-correction, whereas direct self-correction fails to achieve reliable performance.

6.3 MQM-MSC Introduces Minimal Inference Overhead Compared to GEMBA-MQM

The GEMBA-MQM method enhances performance by utilizing three-shot prompts and providing detailed instructions for per example. This strategy significantly increases computational overhead. In contrast, the MQM-MSC method proposed in this paper introduces additional computational overhead only during the masked re-evaluation stage, which requires only brief prompt. Therefore, compared to the high computational cost of GEMBA-MQM, the additional computational overhead of MQM-MSC is acceptable.

Strategy	Extra	Input Tokens	Generated Tokens
GEMBA-MQM	–	1168.7	51.2
MQM-MSC	+	156.3	26.3

Table 6: Average number of input and generated tokens per sentence for each strategy. ”+” indicates the additional strategy introduced in MQM-MSC.

7 Conclusion

In this paper, we propose MQM-MSC, a training-free framework that enhances the interpretability of translation quality estimation through mask-driven self-correction. Innovatively, we introduce a masking-driven self-correction approach that combines positional masking with dual-verification checks. This method filters out over 50% of false positives across multiple LLMs and language pairs while preserving error distributions aligned with human annotations. The framework significantly enhances the reliability and faithfulness of error annotations.

Acknowledgements

This work is supported by the Regional Fund Project of National Natural Science Foundation of China (62166022), and Yunnan Xingdian Talent Support Plan Project(KKXX202403023)

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Daniel Deutsch, George Foster, and Markus Freitag. 2023. Ties matter: Meta-evaluating modern metrics with pairwise accuracy and tie calibration. *arXiv preprint arXiv:2305.14324*.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021a. 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, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021b. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474, 12.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022a. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In Philipp Koehn, Loïc Barrault, Ondřej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Tom Kocmi, André Martins, Makoto Morishita, Christof Monz, Masaaki Nagata, Toshiaki Nakazawa, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Marco Turchi, and Marcos Zampieri, editors, *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid), December. Association for Computational Linguistics.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André FT Martins. 2022b. 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.

Markus Freitag, Nitika Mathur, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Tom Kocmi, Frederic 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.

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 preprint arXiv:2305.11738*.

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

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2023. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*.

Tom Kocmi and Christian Federmann. 2023a. Gemba-mqm: Detecting translation quality error spans with gpt-4. *arXiv preprint arXiv:2310.13988*.

Tom Kocmi and Christian Federmann. 2023b. Large language models are state-of-the-art evaluators of translation quality. *arXiv preprint arXiv:2302.14520*.

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. *arXiv preprint arXiv:2107.10821*.

Christoph Leiter, Juri Opitz, Daniel Deutsch, Yang Gao, Rotem Dror, and Steffen Eger. 2023. The eval4nlp 2023 shared task on prompting large language models as explainable metrics. *arXiv preprint arXiv:2310.19792*.

Arle Richard Lommel, Aljoscha Burchardt, and Hans Uszkoreit. 2013. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*, London, UK, November 28-29. Aslib.

Qingyu Lu, Baopu Qiu, Liang Ding, Kanjian Zhang, Tom Kocmi, and Dacheng Tao. 2023. Error analysis prompting enables human-like translation evaluation in large language models. *arXiv preprint arXiv:2303.13809*.

Qingyu Lu, Liang Ding, Kanjian Zhang, Jinxia Zhang, and Dacheng Tao. 2024. Mqm-ape: Toward high-quality error annotation predictors with automatic post-editing in llm translation evaluators. *arXiv preprint arXiv:2409.14335*.

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. *Advances in Neural Information Processing Systems*, 36:46534–46594.

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.

Stefano Perrella, Lorenzo Proietti, Alessandro ScirÃ, Niccolò Campolungo, Roberto Navigli, et al. 2022. Matese: Machine translation evaluation as a sequence tagging problem. In *Workshop on Statistical Machine Translation*, pages 569–577. Association for Computational Linguistics.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652.

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: Technical Papers*, pages 223–231.

Yu Wan, Dayiheng Liu, Baosong Yang, Haibo Zhang, Boxing Chen, Derek F Wong, and Lidia S Chao. 2022. Unite: Unified translation evaluation. *arXiv preprint arXiv:2204.13346*.

Zhenyu Wu, Qingkai Zeng, Zhihan Zhang, Zhaoxuan Tan, Chao Shen, and Meng Jiang. 2024. Large language models can self-correct with minimal effort. In *AI for Math Workshop@ ICML 2024*.

Wenda Xu, Danqing Wang, Liangming Pan, Zhenqiao Song, Markus Freitag, William Yang Wang, and Lei Li. 2023. Instructscore: Explainable text generation evaluation with finegrained feedback. *arXiv preprint arXiv:2305.14282*.

Haofei Zhao, Yilun Liu, Shimin Tao, Weibin Meng, Yimeng Chen, Xiang Geng, Chang Su, Min Zhang, and Hao Yang. 2024. From handcrafted features to llms: A brief survey for machine translation quality estimation. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10. IEEE.