Error Analysis Prompting Enables Human-Like Translation Evaluation in Large Language Models

Anonymous ACL submission

Abstract

Generative large language models (LLMs), e.g., ChatGPT, have demonstrated remarkable proficiency across several NLP tasks, such as machine translation, text summarization. Recent research (Kocmi and Federmann, 2023b) has shown that utilizing LLMs for assessing the quality of machine translation (MT) achieves state-of-the-art performance at the system level but *performs poorly at the segment level*. To further improve the performance of LLMs on MT quality assessment, we conduct an investigation into several prompting designs, and propose a new prompting method called **`Error Analysis Prompting`** (EAPrompt) by combining Chain-of-Thoughts (Wei et al., 2022) and Error Analysis (Lu et al., 2023). This technique emulates the commonly accepted human evaluation framework - Multidimensional Quality Metrics (MQM, Freitag et al. (2021)) and *produces explainable and reliable MT evaluations at both the system and segment level*. Experimental Results from WMT22 metrics shared task validate the effectiveness of EAPrompt on various LLMs, with different structures. Further analysis confirms that EAPrompt effectively distinguishes major errors from minor ones, while also sharing a similar distribution of the number of errors with MQM. These findings highlight the potential of EAPrompt as a human-like evaluator prompting technique for MT evaluation. We will release our code and scripts to facilitate the community.

1 Introduction

Large language models (LLMs), especially Generative Pre-trained Transformer (GPT) models (Radford et al., 2019; Brown et al., 2020) such as ChatGPT (Ouyang et al., 2022; Achiam et al., 2023), have shown remarkable performance in various natural language processing (NLP) tasks (Qin et al., 2023; Zhong et al., 2023). LLMs are capable of integrating multiple NLP tasks and can generate detailed and comprehensive responses to human inquiries. Additionally, they can respond appropriately to follow-up questions and maintain sensitivity throughout several turns of conversation.

Previous research has demonstrated that LLMs can perform as well as or even better than other LLMs in machine translation task (Hendy et al., 2023; Jiao et al., 2023; Peng et al., 2023). Given the high cost and time-intensive nature of human evaluation, there is a growing demand for MT metrics that offer both explainability and reliability. Therefore, LLMs hold promise in serving as ideal evaluators, capable of generating both judgments and explanations for the translations.

Concurrent to our research, GEMBA (Kocmi and Federmann, 2023b) presents an encouraging finding that GPT models can surpass current best MT metrics at the system level quality assessment using straightforward zero-shot standard prompting, confirming the reliability and potential of this technique. However, such prompts exhibit unrealistic performance at the segment level, and cannot offer additional interpretable information regarding translation errors, thus detracting from the goal of achieving a "human-like" evaluation.

To this end, we take the further step by carefully investigating advanced prompting strategies upon various LLMs for MT quality assessment and propose a novel prompting strategy - **Error Analysis Prompting** (EAPrompt), combining the Chain-of-Thought (CoT, Wei et al. (2022)) and Error Analysis (EA, Lu et al. (2023)). We give an example of EAPrompt in Figure 1. The idea is to prompt LLMs to emulate the human evaluation framework - MQM (Freitag et al., 2021) by ❶ *identifying major&minor errors*, and ❷ *scoring the translations according to the severity of these errors*.

We conduct experiments using the test set from the WMT22 metrics shared task, comprising 106,758 segments on 54 MT systems across diverse domains to verify the effectiveness of our approach. Our findings reveal that:

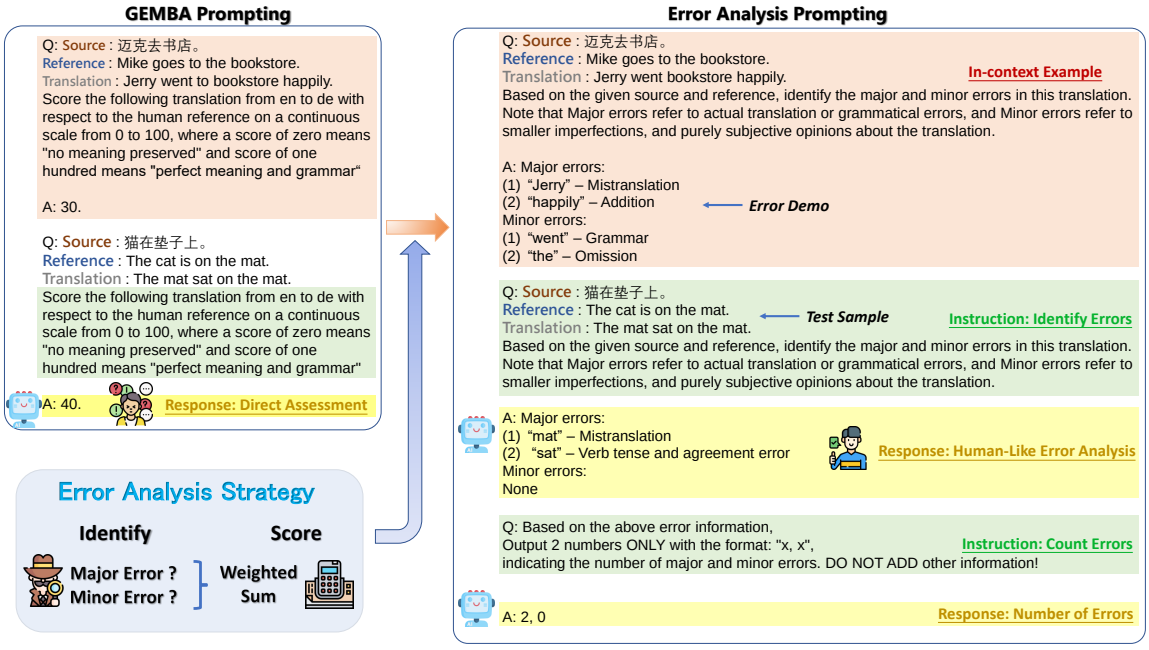

Figure 1: **A comparative overview between GEMBA Prompting and our proposed Error Analysis Prompting** in assessing the MT quality with LLMs.

- EAPrompt significantly enhances the performance of LLMs at the system level. Notably, prompting *GPT-3.5-Turbo* with EAPrompt outperforms all other metrics and prompting strategies, establishing a new state-of-the-art.
- EAPrompt surpasses GEMBA in 8 out of 9 test scenarios across various language models and language pairs, demonstrating superior performance at the segment level.
- The findings regarding EAPrompt's strong performance remain consistent even in reference-less settings, highlighting its suitability for quality estimation tasks.
- When designing prompts, we recommend the EAPrompt variant featuring a 2-step separated prompting approach and itemized error demonstrations.
- Further analysis confirms that EAPrompt adeptly distinguishes major errors from minor ones, closely aligning its error distribution with MQM.
- Optimizing the inference costs of EAPrompt can be achieved by leveraging Regular Expressions instead of counting queries.

This study provides an initial exploration of utilizing error analysis to prompt LLMs as evaluators. EAPrompt can also be extended to benefit other evaluation scenarios within language generation, including summarization and data-to-text tasks.

2 Prompt LLMs with Error Analysis

2.1 Translation Evaluation Metric

Translation evaluation metrics are used to assess the performance of machine translation systems on specific test sets (Freitag et al., 2022; Mathur et al., 2020b). These metrics typically take inputs from three sources: the sentence from source language ("Source"), the reference translation provided by human translators ("Reference"), and the hypothesis being evaluated ("Translation"). In scenarios where reference signals are not provided, this "reference-less" metric can also be utilized for quality estimation purposes (Zerva et al., 2022; Specia et al., 2010; Qiu et al., 2022). The output of the metric is a score or rank indicating the translation quality of each hypothesis.

To verify the reliability of MT metrics, Multidimensional Quality Metric (MQM) has been adopted recently in WMT as a high-quality human evaluation strategy (Freitag et al., 2021). It asks human experts to annotate the errors in the hypothesis and categorize them into "Major" and "Minor" indicating their severity. A detailed example of MQM annotation is presented in Appendix A.

2.2 Prompt LLMs as Evaluation Metrics

When prompting LLMs as evaluation metrics, it is crucial to design appropriate instructions that describe the evaluation task. In this paper, we mainly adopt two prompting strategies: "GEMBA Prompting" and "Error Analysis Prompting".

GEMBA (Kocmi and Federmann, 2023b) is a zero-shot prompting approach that directly asks LLMs to generate a score that reflects the quality of the translation, which shows state-of-the-art performance on GPT models when compared to other model-based metrics. However, they also observe that the performance at the segment level is relatively poorer. This highlights the importance of combining Chain-of-Thought with the Error Analysis Strategy to prompt LLMs in a manner that more closely resembles human evaluation.

2.3 Error Analysis Prompting

Motivated by the MQM framework in human evaluation, the idea of the Error Analysis (EA) paradigm, as introduced by Lu et al. (2023), is to enhance the automatic scoring process by explicitly incorporating error identification, thus providing a more human-like evaluation.

The Chain-of-Thought (CoT) prompting strategy was first proposed by Wei et al. (2022). Instead of directly generating the answer, CoT prompts LLMs to think step-by-step. This approach has shown significant performance improvements on reasoning tasks, such as GSM8K (Cobbe et al., 2021). CoT is an emergent ability of LLMs and has been incorporated in instruction fine-tuning of LLMs (Chung et al., 2022) as well as in benchmarks designed to evaluate LLM capabilities (Suzgun et al., 2022).

In this work, we combine the CoT and EA paradigms, introducing a novel prompting strategy called Error Analysis Prompting (EAPrompt). As shown in Figure 1, EAPrompt divides the scoring process into two stages: First, the LLM is instructed to identify major and minor errors in the translation ("Instruction: Identify Errors"). Subsequently, the number of these two types of errors is counted ("Instruction: Count Errors"). Distinguished from GEMBA prompting, EAPrompt emulates the evaluation process of MQM and produces more explanable and reliable automatic evaluations.

After exploring several prompt contexts in initial experiments, we made the following modifications to EAPrompt as follows:

- we adopt the one-shot learning format (Brown et al., 2020) to enhance the LLMs' understanding of the task (§3.4); different in-context examples are used for different language pairs;
- we employ itemized error demonstration in the template response, enabling clearer identification and quantification of errors (§3.5);
- we partition the evaluation process into two stages to enhance the reliability of metric performance. Additionally, we present a simplified alternative to optimize inference costs by counting errors automatically (§4.3).

2.4 Post-processing of LLM responses

After obtaining the number of major and minor errors, we compute the final score of the translation using the following equation:

$$\text{score} = -w_{\text{major}} n_{\text{major}} - w_{\text{minor}} n_{\text{minor}}, \quad (1)$$

where n_{major} and n_{minor} denotes the number of major and minor errors respectively, while w_{major} and w_{minor} represent the severity weight assigned to major and minor errors. Since different LLMs may apply distinct criteria for major and minor errors, we follow Lu et al. (2023) to adopt a flexible scoring approach by fixing the $w_{\text{minor}} = 1$ while treating w_{major} as a latent variable within EAPrompt. We present an analysis on the influence of this variable in §4.2 and the detailed implementation in experiments is described in Appendix B.

3 Experimental Results

3.1 Experiment Setup

Dataset We utilize the test set from the WMT22 shared tasks (Freitag et al., 2022) in English-German (En-De), English-Russian (En-Ru), and Chinese-English (Zh-En) across 4 different domains - conversational, e-commerce, news, and social. Table 1 provides statistics about our test set.

Human Evaluation We utilize MQM (Freitag et al., 2021) as human judgments, which is annotated by human experts and has been widely adopted in recent WMT metrics shared tasks (Freitag et al., 2022) and quality estimation tasks (Zerva et al., 2022).

Meta Evaluation We follow the standard meta-evaluation approach to measure the performance of MT evaluation metrics (Freitag et al., 2023). At

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

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

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 adopt the group-by-item pairwise accuracy with tie calibration as described by Deutsch et al. (2023). We use the acc^*_{eq} variant to compare vectors of metric and gold scores for each segment, then average the results over segments. All the meta-evaluation are calculated with MTME[1], a metric evaluation tool recommended by WMT (Freitag et al., 2022) to maintain comparability with other metrics.

3.2 Baselines and Large Language Models

Baseline Metrics Given the reported unreliability of BLEU (Papineni et al., 2002), we compare our method with several model-based metrics for MT evaluation. BLEURT (Sellam et al., 2020) and COMET (Rei et al., 2020) are supervised neural metrics fine-tuned on human evaluation. We employ the **BLEURT20** and **COMET-22** for reference-based metrics, and **COMET-QE** for the reference-less metric. **UniTE** (Wan et al., 2022) is also a learnt metric that evaluates MT outputs combining three different evaluation scenarios. We also adopt **UniTE-src** for comparing reference-less metrics. **MetricX-XXL** (Juraska et al., 2023) is a large-scale multi-task metric that fine-tunes LLM checkpoints using diverse human feedback data. For reference-less metrics, we also reproduce **MaTESe-QE** (Perrella et al., 2022), a metric leveraging transformer-based multilingual encoders to identify error spans in translations.

Large Language Models For prioprietary models, we use the OpenAI API to experiment with **GPT-3.5-Turbo** [2]. We also experiment with a human-aligned Llama2-70B series model (Touvron et al., 2023b) fine-tuned with multilingual translation data, noted as "**Llama2-70b-Chat**" in experimental results. We also use a high-quality sparse mixture-of-experts model, Mixtral-8x7b (Jiang et al., 2024). We use a state-of-the-art checkpoint **Mixtral-8x7b-Instruct** which has been optimised through supervised fine-tuning and direct preference optimisation to follow instructions.

3.3 Prompts for LLM evaluators

For GEMBA Prompting, we adopt the GEMBA-DA variant as suggested by (Kocmi and Federmann, 2023b), given its widespread usage and superior performance across three language pairs (Kocmi and Federmann, 2023a).

For Error Analysis Prompting (EAPrompt), we conduct a comparison of various prompting strategies of EAPrompt in §3.5, and use the best-performing variant for other experiments. We show the detailed prompt contexts in Appendix C.

3.4 Experimental Results

We compute system&segment level performance of EAPrompt with LLMs in Table 2. We see that:

(i) At the system level, EAPrompt empowers GPT-3.5-Turbo to surpass all other metrics and achieves state-of-the-art performance. Consistent with the findings of Kocmi and Federmann (2023b), LLMs achieve state-of-the-art performance across all three language pairs at the system level, significantly outperforming traditional metrics ("**Baselines**") by a large margin.

Remarkably, when prompting all LLMs with EAPrompt, the performance notably surpasses GEMBA at the system level, achieving the highest pairwise accuracy of 91.2% on **GPT-3.5-Turbo**, thus establishing a new SOTA.

(ii) At the segment level, EAPrompt outperforms GEMBA in 8 out of 9 tested scenarios. At the segment level, despite previous findings by Kocmi and Federmann (2023b) regarding the weak correlation between LLMs as evaluators and human judgments, prompting with EAPrompt addresses this drawback of LLM evaluators, outperforming GEMBA's performance on nearly all tested LLMs

[1] https://github.com/google-research/mt-metrics-eval

[2] We use the 0613 OpenAI model versions.

Models	Metrics / Prompts	Ref?	System-Level Acc.	Segment-Level Acc*		
			All (3 LPs)	En-De	En-Ru	Zh-En
Baselines	MetricsX-XXL	✓	85.0	60.4	60.6	54.4
	BLEURT20	✓	84.7	56.8	54.0	48.9
	COMET22	✓	83.9	59.4	57.7	53.6
	UniTE	✓	82.8	59.8	57.7	51.7
	COMET-QE	✗	78.1	55.5	53.4	48.3
	UniTE-src	✗	75.9	58.2	55.4	50.8
	MaTESe-QE	✗	74.8	57.2	49.9	49.4
Llama2-70b-Chat	GEMBA	✓	74.1	53.7	48.8	45.4
	EAPrompt	✓	85.4 (+11.3)	55.2(+1.5)	51.4 (+2.6)	**50.2 (+4.8)**
	GEMBA	✗	72.6	54.1	47.8	45.0
	EAPrompt	✗	85.8 (+13.2)	55.0 (+0.9)	51.6 (+3.8)	49.3 (+4.3)
Mixtral-8x7b-Instruct	GEMBA	✓	69.7	54.8	48.3	46.7
	EAPrompt	✓	84.0 (+14.3)	53.8 (-1.0)	50.6 (+2.3)	48.2 (+1.5)
	GEMBA	✗	74.1	54.8	47.5	46.2
	EAPrompt	✗	82.5 (+8.4)	54.1 (-0.7)	49.9 (+2.4)	48.3 (+1.1)
GPT-3.5-Turbo	GEMBA	✓	86.5	55.2	49.5	48.2
	EAPrompt	✓	**91.2 (+4.7)**	**56.7 (+1.5)**	53.3 (+3.8)	50.0 (+1.8)
	GEMBA	✗	86.9	54.7	50.0	47.6
	EAPrompt	✗	89.4 (+2.5)	55.7 (+1.0)	**53.4** (+3.4)	48.8 (+1.2)

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 best results among the same model are highlighted in **bold**. The best results among all metrics are underlined.

and language pairs by a significant margin. The best segment-level results are achieved by **GPT-3.5-Turbo** for En-De (56.7) and En-Ru (53.4), and by **Llama2-70b-chat** for Zh-En (50.2). This validates the effectiveness of our EAPrompt.

The only exception of the result is observed for En-De **Mixtral-8x7b-Instruct**, where the segment-level accuracy is lower than GEMBA by 1.0. This discrepancy might be attributed to the limited capability of identifying translation errors in En-De language pair. Another notable finding is that prompting with LLMs, both with GEMBA and EAPrompt, fails to surpass current best metrics ("**Baselines**") at the segment level. This could be because these baseline metrics have been fine-tuned using extensive translation and human evaluation datasets, while the LLMs employed in our experiment are versatile models guided by few-shot prompts.

(iii) EAPrompt enhances the performance of LLMs as translation evaluators in reference-less scenarios. Our main findings remain consistent with both reference-based and reference-less settings (indicated by "✓" and "✗" in **Ref?**, respectively), where EAPrompt continues to outperform GEMBA across all three tested LLMs at the system level, and in 8 out of 9 scenarios at the segment level. The improvement is slightly lower compared to scenarios with referenced signals.

These results underscore the impressive cross-lingual capabilities of LLMs and their suitability for quality estimation under EAPrompt, even in the absence of reference translations, which poses a significant challenge on MT evaluation.

3.5 Ablation Study of Prompt Variants

Given the crucial significance of the prompt design, we investigate several versions of in-context prompt contexts and present an analysis in Table 3. The prompt contexts used in our experiment are detailed in Appendix C. Due to budget constraints, we utilize two LLMs, **Mixtral-8x7b-Instruct** and **Llama2-70b-Chat**, as the test bed for this ablation study. Our findings indicate that:

(i) Itemized error demonstration is superior to detailed illustration. We assume that when identifying translation errors, providing detailed descriptions may impede the LLM's capability to accurately identify errors and count the number of them. As illustrated in the "**Demo of Errors**" column, employing itemized error demonstrations instead of detailed paragraphs yields improved performance at both the system and segment levels for both tested LLMs.

In our initial study, we observed that generating excessively detailed responses could lead to incorrect error counting or misclassification of error

Prompt	Demo of Errors		Type of Queries		Mixtral-8x7b-Instruct				Llama2-70b-Chat			
	Detailed	Itemized	1-step	2-step	All (3 LPs)	En-De	En-Ru	Zh-En	All (3 LPs)	En-De	En-Ru	Zh-En
GEMBA	-	-	-	-	69.7	**54.8**	48.3	46.7	74.1	53.7	48.8	45.4
EAPrompt	✓		✓		75.2	53.4	50.0	45.0	62.0	53.7	47.0	47.8
	✓			✓	75.5	53.4	47.9	45.5	84.7	53.5	46.9	47.5
		✓	✓		60.2	53.4	45.1	45.6	56.9	53.7	48.4	**50.2**
		✓		✓	**84.0**	53.7	**50.6**	**48.2**	**85.4**	**55.2**	**51.4**	**50.2**

Table 3: **Comparison of the system level** ("All (3 LPs)") **and segment level** ("En-De", "En-Ru", "Zh-En") **performance of LLMs with different variants of prompts for EAPrompt**. We compare itemized or detailed responses to demonstrate identified errors. We also compare the instructions, whether separated into two queries (marked as "2-step", one for identifying errors and another for scoring) or combined into a single query (marked as "1-step"). The best results among all prompt variants are highlighted in **bold**.

severity. Therefore, it is recommended to employ clear and concise error descriptions in a format that is easily processed and comprehended by LLMs.

(ii) Separating the scoring process from error identification with two queries will enhance the performance of LLMs as translation evaluators. Another consideration in prompt design is the division of the evaluation process into error identification and error counting. As depicted in the "**Type of Queries**" column, it is evident that the performance of using a single prompting step is considerably lower than that of employing a 2-step prompting approach. This may be because separating the scoring process allows LLMs to concentrate on a single task in each query, thereby facilitating more accurate judgments and reducing the likelihood of incorrectly counting the number of errors.

(iii) Among the prompting strategies, EAPrompt appears to be more suitable for the LLMs as translation evaluators. When compared with GEMBA prompting strategies, the EAPrompt variant featuring a 2-step separated prompting approach and itemized response achieves superior performance in enhancing LLMs' effectiveness as translation evaluators. Consequently, we recommend employing this particular variant for LLMs as translation evaluators.

4 Analysis

4.1 EAPrompt aligns with human judgment through similar distribution of major and minor errors across most LLMs

To investigate can LLMs align with gold human judgement MQM through similar distributions of major and minor errors, we present the error distribution across various test scenarios in Figure 2.

We can see that, for major errors, all tested LLMs exhibit distributions that closely resemble MQM. Regarding minor errors, Mixtral-8x7b-Instruct appears to produce a slightly higher frequency of such errors compared to other LLMs, while the distribution of other LLMs remains consistent with MQM. This observation further validates the efficacy of EAPrompt.

This finding provides valuable insights into enhancing the reliability of LLMs as translation evaluators. It suggests a potential focus on guiding LLMs to more accurately identify minor errors, such as clarifying the specific categories and severity of minor errors.

4.2 EAPrompt empowers LLMs to distinguish major errors from minor ones

A potential concern on EAPrompt is whether this technique can prompt LLMs to distinguish major errors from minor ones. To address this concern, we adjust the weight assigned to major errors (w_{major}) in the score computation process outlined in §2.4. We visualize the impact of this adjustment on both the system and segment-level performance in Figure 3. If the metric effectively distinguishes major errors from minor ones, we anticipate a noticeable performance decrease when the weight of major errors w_{major} approaches that of minor errors ($w_{\text{minor}} = 1$ in this study).

Our findings reveal that for all three LLMs tested, adjusting $w_{\text{major}} < 3$ results in a substantial performance decline, indicating that prompting error analysis with all tested LLMs possesses the ability to discriminate major errors from minor ones.

Another noteworthy observation from this analysis is that when $w_{\text{major}} \geq 5$, both the system-level and segment level-accuracies exhibit minimal fluctuation, suggesting that the performance of EAPrompt remains nearly unaffected by this latent variable during score computation.

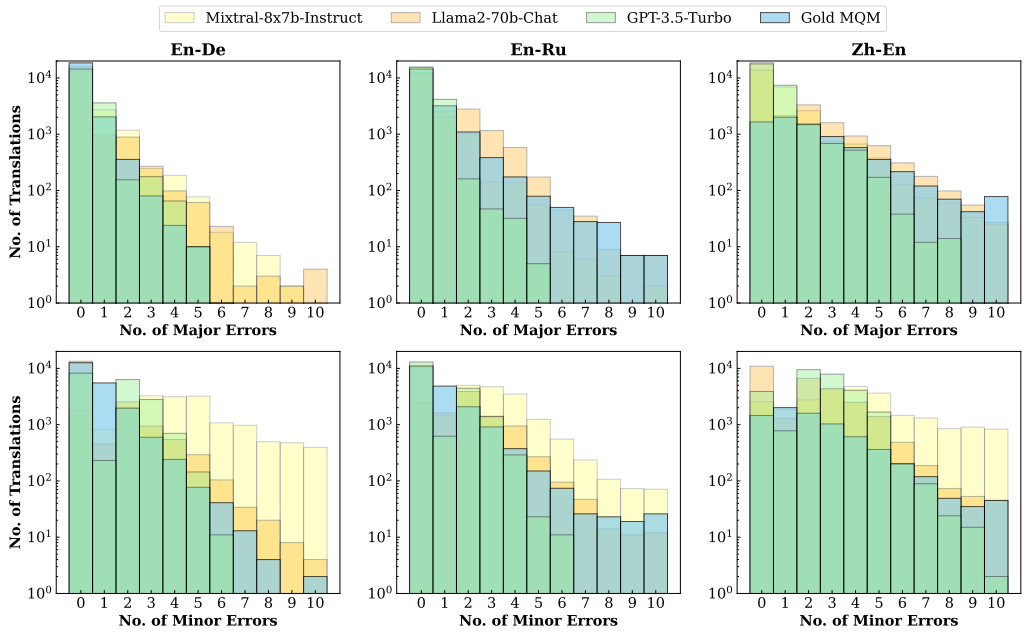

Figure 2: **Distribution of identified error counts** across various LLMs and human evaluation (MQM), for the language pairs En-De, En-Ru and Zh-En, repectively.

		System-Level Acc.	Segment-Level Acc*		
Models	**Repr?**	**All (3 LPs)**	**En-De**	**En-Ru**	**Zh-En**
Llama2-70b-Chat	✓	85.0	55.6	51.5	50.4
	✗	85.4	55.2	51.4	50.2
Mixtral-8x7b-Instruct	✓	82.8	53.7	50.9	47.6
	✗	84.0	53.7	50.6	48.2
GPT-3.5-Turbo	✓	90.1	56.8	53.9	50.0
	✗	91.2	56.7	53.3	50.0

Table 4: **Performance comparison of EAPrompt between utilizing the Regular Expression Matching strategy** ("✓" in **Repr?**) and the counting query strategy ("✗" in **Repr?**) across various LLMs.

4.3 EAPrompt optimizes inference costs by utilizing regular expressions instead of counting queries

Since EAPrompt adopts a two-step prompting strategy, one related question is: can we simplify the query process to reduce inference costs? One potential approach involves substituting the scoring query step with an algorithm that identifies major and minor errors using regular expressions (**Repr**) to detect bullet points or initial numbers. A detailed description of the **Repr** matching strategy is provided in the Appendix. The analysis, as depicted in Table 4, indicates that employing Repr matching strategy, as opposed to the original query for counting errors (indicated by "✗" in **Repr?**), yields minimal performance variation at both system and segment levels. Thus, if inference costs are a concern for this metric, substituting the second query step of EAPrompt with regular expressions could be a viable option. Note that for different LLMs, a tailored regular expression pattern may be necessary to encompass various response structures.

4.4 Case Study

We discuss potential issues encountered by LLMs and their corresponding solutions in Appendix E, including invalid responses, input order bias, etc. We aim to provide insights that should be considered when utilizing LLMs as translation evaluators.

5 Related Work

Translation Evaluation Metrics MT Evaluation metrics are of crucial importance to the development of MT systems (Freitag et al., 2022). Studies have shown that traditional surface-based metrics such as BLEU (Papineni et al., 2002) are

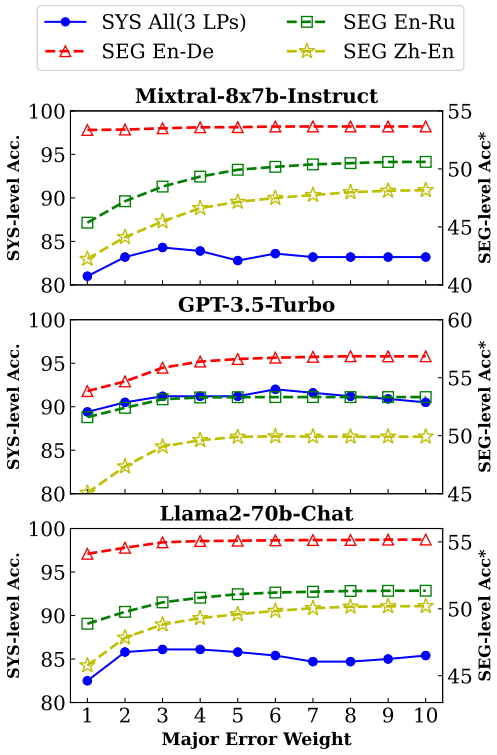

Figure 3: **Effect of varying major error weight** (w_{major}) on EAPrompt across different LLMs at both system and segment levels.

no longer suitable for evaluating high-quality MT systems (Mathur et al., 2020a). Modern metrics like COMET (Rei et al., 2020), MetricsX-XXL (Juraska et al., 2023), BLEURT (Sellam et al., 2020), and UniTE (Wan et al., 2022) leverage human evaluations and high-quality translations for training. While these metrics achieve strong correlation with human judgements such as MQM (Freitag et al., 2021), there is a growing demand for explainability in their evaluation. Despite progress, recent research struggles to strike a balance between the reliability and explainability of these metrics (Lu et al., 2023; Xu et al., 2022; Perrella et al., 2022). In this work, we delve into the potential of LLMs for "human-like" translation evaluation, as they possess the capability to explicitly identify translation errors without further fine-tuning, which resembles the evaluation process of human.

LLMs as Evaluators LLMs refers to language models with hundreds of billion of parameters which are trained on massive textual data (Chang et al., 2024; Zhao et al., 2023). Since the emergence of ChatGPT, LLMs have shown its remarkable proficiency across various NLP tasks (Achiam et al., 2023; Touvron et al., 2023b). A prevalent application of LLMs is harnessing them as evaluators for assessing the performance of Chatbots (Zheng et al., 2023). Recent studies also show LLM's efficacy in evaluating NLG tasks like summarization and dialog generation through multi-step prompting (Liu et al., 2023). GEMBA (Kocmi and Federmann, 2023b) is the pioneering effort in utilizing LLMs as translation evaluators via a zero-shot prompting approach with GPT models. In this work, EAPrompt innovatively combines error analysis (Lu et al., 2023) and chain-of-thought (Wei et al., 2022) to prompt LLMs for achieving human-like translation evaluation.

Subsequent work follows our work to further explore the potential of LLMs as translation evaluators. AutoMQM (Fernandes et al., 2023) parallels our approach, utilizing PaLM-2 model (Anil et al., 2023) as the testbed. GEMBA-MQM (Kocmi and Federmann, 2023a) further improves EAPrompt by employing a few-shot prompting technique using GPT-4, making this approach universally applicable across languages. Another line of research focuses on fine-tuning LLMs to accurately predict error spans in translations. For instance, InstructScore (Xu et al., 2023) fine-tunes a Llama model (Touvron et al., 2023a), while XCOMET (Guerreiro et al., 2023) scales from COMETKiwi (Rei et al., 2023) to achieve this goal.

6 Conclusion

In this paper, we explore the potential of LLMs as a metric for evaluating translations. We design a novel one-shot prompting strategy EAPrompt based on chain-of-thought and error analysis, and show that this strategy significantly improves the evaluation performance on both the system and segment levels. We compare different EAPrompt variants and ultimately opt for a 2-step prompting approach with itemized error demonstrations. Further analysis confirms EAPrompt's proficiency in error identification and its alignment with the commonly accepted human evaluations MQM.

In future work, we would like to experiment with a broader range of LLMs (Barrault et al., 2019; Anastasopoulos et al., 2021; Kocmi et al., 2022; Zan et al., 2022), to make our conclusion more convincing. Lastly, it will be interesting to test the capabilities of LLMs for other MT-related tasks, such as grammatical error correction and automatic post-editing (Wu et al., 2023; Vidal et al., 2022).

Limitations

The limitations of this work are three-fold:

- Potential Test Data Contamination: Although we utilized WMT22 to minimize the risk of test set leakage in the training data of LLMs, it is still possible that some contamination from the test data remains. Therefore, future researchers utilizing these datasets should be cautious and carefully address this issue, as it may affect the availability of the test set for comparison purposes.

- Budget Constraints: Due to limited resources, we were unable to explore more prompt choices comprehensively in our research. The findings presented in this study only reflect our initial experiments. We leave the impact of different prompt choices for further investigation.

- Limited Range of LLMs Tested: In this study, we focused on evaluating a limited number of LLMs that we believed possessed potential and capability as translation evaluators. However, it is important to note that not all existing LLMs can necessarily serve as reliable evaluators under the EAPrompt approach. Future research could explore and experiment with a broader range of LLMs, examining their effectiveness and assessing their suitability as evaluators.

Ethics Statement

We take ethical considerations very seriously, and strictly adhere to the Code of Ethics. All procedures performed in this study are in accordance with the ethical standards. This paper focuses on evaluating the capabilities of LLM as a translation evaluator. Our proposed approach, EAPrompt, does not include statements that induce the model to generate harmful information. Additionally, this method solely extracts and processes the numerical scores from the model's response, thereby further mitigating the potential risks. Both the datasets and models used in this paper are publicly available and have been widely adopted by researchers. Our model will not learn from user inputs or cause potential risks to the NLP community. We ensure that the findings and conclusions of this paper are reported accurately and objectively. Informed consent was obtained from all individual participants included in this study.

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*.

Antonios Anastasopoulos, Ondřej Bojar, Jacob Bremerman, Roldano Cattoni, Maha Elbayad, Marcello Federico, et al. 2021. Findings of the IWSLT 2021 evaluation campaign. In *IWSLT*.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint*.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, et al. 2019. Findings of the 2019 conference on machine translation (WMT19). In *WMT*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, et al. 2020. Language models are few-shot learners. *NeurIPS*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A survey on evaluation of large language models. *ACM*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint*.

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

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *WMT*.

Markus Freitag, George Foster, David Grangier, et al. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *TACL*.

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

Kocmi, Frederic Blain, Daniel Deutsch, Craig Stewart, Chrysoula Zerva, Sheila Castilho, Alon Lavie, and George Foster. 2023. Results of WMT23 metrics shared task: Metrics might be guilty but references are not innocent. In *WMT*.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, et al. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *WMT*.

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*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, et al. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *arXiv preprint*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint*.

Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Xing Wang, and Zhaopeng Tu. 2023. Is chatgpt a good translator? a preliminary study. *arXiv preprint*.

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 *WMT*.

Tom Kocmi, Rachel Bawden, Ondřej Bojar, et al. 2022. Findings of the 2022 conference on machine translation (WMT22). In *WMT*.

Tom Kocmi and Christian Federmann. 2023a. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *WMT*.

Tom Kocmi and Christian Federmann. 2023b. Large language models are state-of-the-art evaluators of translation quality. In *EAMT*.

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 *WMT*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *EMNLP*.

Qingyu Lu, Liang Ding, Liping Xie, Kanjian Zhang, Derek F. Wong, and Dacheng Tao. 2023. Toward human-like evaluation for natural language generation with error analysis. In *ACL*.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020a. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *ACL*.

Nitika Mathur, Johnny Wei, Markus Freitag, Qingsong Ma, and Ondřej Bojar. 2020b. Results of the WMT20 metrics shared task. In *WMT*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

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. *arXiv preprint*.

Stefano Perrella, Lorenzo Proietti, Alessandro Scirè, Niccolò Campolungo, and Roberto Navigli. 2022. MaTESe: Machine translation evaluation as a sequence tagging problem. In *WMT*.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint*.

Baopu Qiu, Liang Ding, Di Wu, Lin Shang, Yibing Zhan, and Dacheng Tao. 2022. Original or translated? on the use of parallel data for translation quality estimation. *arXiv preprint*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

Ricardo Rei, Nuno M. Guerreiro, JosÃ© Pombal, Daan van Stigt, Marcos Treviso, Luisa Coheur, José G. C. de Souza, and André Martins. 2023. Scaling up CometKiwi: Unbabel-IST 2023 submission for the quality estimation shared task. In *WMT*.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *EMNLP*.

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

Lucia Specia, Dhwaj Raj, and Marco Turchi. 2010. Machine translation evaluation versus quality estimation. *Machine translation*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint*.

Blanca Vidal, Albert Llorens, and Juan Alonso. 2022. Automatic post-editing of MT output using large language models. In *AMTA*.

Yu Wan, Dayiheng Liu, Baosong Yang, Haibo Zhang, Boxing Chen, Derek Wong, and Lidia Chao. 2022. UniTE: Unified translation evaluation. In *ACL*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint*.

Haoran Wu, Wenxuan Wang, Yuxuan Wan, Wenxiang Jiao, and Michael Lyu. 2023. Chatgpt or grammarly? evaluating chatgpt on grammatical error correction benchmark. *arXiv preprint*.

Wenda Xu, Yi-Lin Tuan, Yujie Lu, Michael Saxon, Lei Li, and William Yang Wang. 2022. Not all errors are equal: Learning text generation metrics using stratified error synthesis. In *EMNLP*.

Wenda Xu, Danqing Wang, Liangming Pan, Zhenqiao Song, Markus Freitag, William Wang, and Lei Li. 2023. INSTRUCTSCORE: Towards explainable text generation evaluation with automatic feedback. In *EMNLP*.

Changtong Zan, Keqin Peng, Liang Ding, Baopu Qiu, et al. 2022. Vega-MT: The JD explore academy machine translation system for WMT22. In *WMT*.

Chrysoula Zerva, Frédéric Blain, Ricardo Rei, et al. 2022. Findings of the WMT 2022 shared task on quality estimation. In *WMT*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint*.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2023. Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert. *arXiv preprint*.

A Description of MQM

Multidimensional Quality Metric (MQM) is a human evaluation framework commonly used in WMT metrics shared tasks as the golden standard (Freitag et al., 2021, 2023). In this paper, EAPrompt emulates MQM to identify major and minor errors, providing insightful explanations for the translation. Table 6 illustrates an example in detail annotated through MQM framework.

B Post-processing of EAPrompt

As described in §2.4, we treat w_{major} as a latent variable within EAPrompt. In our experiments, we select this latent variable with the best averaging performance for each LLMs denoted as w^*_{major}. The value was reported in Table 5.

Model	$w^*_{\textbf{major}}$
GPT-3.5-Turbo	6
Llama2-70b-Chat	10
Mixtral-8x7b-Instruct	10

Table 5: **Optimal values of $w^*_{\textbf{major}}$ for each LLM**. To ensure fair comparison, we maintain this variable constant across all tested scenarios for every LLM.

C Prompt Contexts of EAPrompt

Figure 4 provides the prompt contexts implemented in EAPrompt, along with the detailed error demonstration and combined query instruction discussed in §3.5 for reproduction of our experiments.

D Counting Errors using Regular Expressions Matching

In Figure 5, we present an overview of our error-matching strategy utilized in §4.3 to automatically identify the number of major and minor errors. The procedure can be listed as follows:

1. Locate "major error" and "minor error" within the response, then segment the response accordingly.

2. Utilize Regular Expression matching to identify the initial numbers of major and minor errors. For implementation, we include three different initial number formats: "1.", "1)" and "(1)" (using "1" as an example);

3. Record the number of major and minor errors.

E Case Study

In Figure 6, we list several typical issues with the case study that should be aware of when using LLMs such as ChatGPT as translation evaluators.

E.1 Potential instability in the responses without temperature control

Issue: When evaluating translations using LLMs, the generated responses may vary significantly. See in **Case 1**, we regenerate several responses with the same input and obtain 3 different scores (98, 95, 100) for the translation.

Solution: We control the temperature parameter to mitigate the variability in LLM judgments. Accordingly, for all experiments detailed in this paper, we set the temperature to 0 for **GPT-3.5-Turbo**. For the other two models, namely **Llama2-70b-Chat** and **Mixtral-8x7b-Instruct**, we opted for a temperature setting of 0.05 since the inference parameter from these two models should be above zero.

E.2 Input order bias when evaluating multiple translations simultaneously

Issue: An alternative prompting strategy is to present multiple translations together as a single input to LLMs for evaluation, reducing the number of queries and potentially saving budget. However, we observe a bias where translations presented earlier tend to get higher scores compared to those presented later. As shown in **Case 2**, we provide 8 translations along with their corresponding source and reference sentences. At the first time, we present the translations sequentially and ask LLM to rank them according to their translation quality. Then, we reverse the order of translations and obtain an entirely different sequence of ranks.

Solution: The contradictory results may be attributed to the auto-regressive nature of the decoder model, which gives more attention to the latter input, potentially leading to greater identification of errors for the translation input later. Therefore, we recommend that researchers input one translation at a time instead of providing multiple translations.

E.3 LLMs may generate invalid answers for all prompting strategies

Issue: We observe that in certain cases, LLMs may not function as translation evaluators that may producing invalid answers with textual explanations. A typical case is illustrated in **Case 3**, where ChatGPT tends to prioritize the BLEU score instead of offering judgments based on its inherent capabilities.

Solution: We follow the method mentioned in Kocmi and Federmann (2023b) for handling invalid answers, where we introduce randomness to LLMs by iteratively increasing the temperature. Subsequently, we take the first response that falls within the expected score range.

System	Online-A.en
Domain	conversational
Doc_id	1
Seg_id	6
Text	**Source:** 请问，订单情况现在是什么样?
	Reference: May I ask what the status of the order is now?
	Translation: Please ask, what is the order situation now?
Human Evaluation	**Major Error:** "Please ask" - Accuracy/Mistranslation
	Minor Error: "situation" - Style/Awkward

Table 6: **An example of MQM**, comprising information of the test sample along with human-annotated major and minor errors.

Figure 4: **The prompt contexts employed in EAPrompt**. We present itemized/detailed responses for error demonstrations and separated/combined instructions for different types of queries.

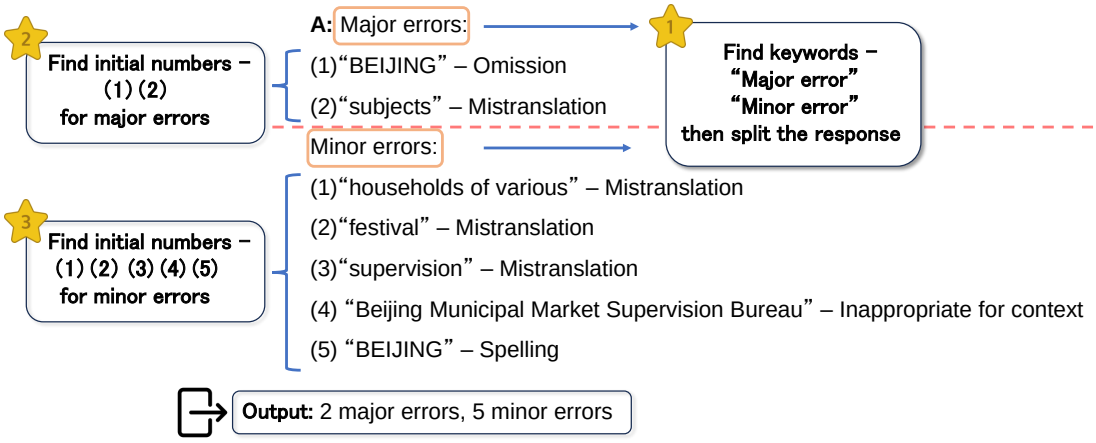

Figure 5: **The regular expression matching strategy** utilized in §4.3 to automatically count the number of major and minor errors in the LLM response.

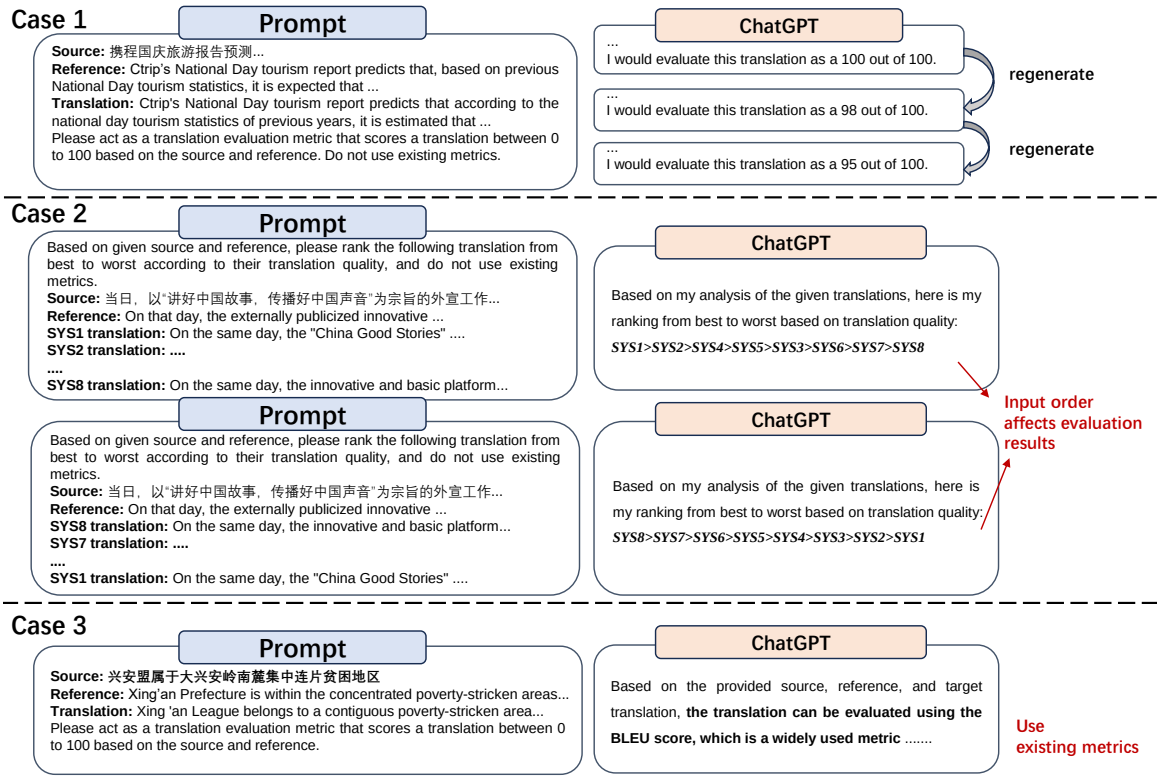

Figure 6: **Case study of potential issues in LLMs**. All three cases are from GPT-3.5-Turbo model ("ChatGPT"). **Top**: LLM exhibits variations in its responses upon multiple regenerations; **Medium**: different input order of samples may affect the judgment of LLM; **Bottom**: LLM sometimes relies on existing metrics during translation evaluation.