Same evaluation, more tokens: On the effect of input length for machine translation evaluation using Large Language Models

Tobias Domhan* **and Dawei Zhu**
Amazon AGI
{domhant,daweizhu}@amazon.com

Abstract

Accurately evaluating machine-translated text remains a long-standing challenge, particularly for long documents. Recent work has shown that large language models (LLMs) can serve as reliable and interpretable sentence-level translation evaluators via MQM error span annotations. With modern LLMs supporting larger context windows, a natural question arises: can we feed entire document translations into an LLM for quality assessment? Ideally, evaluation should be invariant to text length, producing consistent error spans regardless of input granularity. However, our analysis shows that text length significantly impacts evaluation: longer texts lead to fewer error spans and reduced system ranking accuracy. To address this limitation, we evaluate several strategies, including granularity-aligned prompting, Focus Sentence Prompting (FSP), and a fine-tuning approach to better align LLMs with the evaluation task. The latter two methods largely mitigate this length bias, making LLMs more reliable for long-form translation evaluation.

1 Introduction

Historically, the field of Machine Translation has been dominated by a sentence-level paradigm, where individual sentences are translated in isolation. Large Language Models (LLMs), with increasingly long context windows, are able to process hundreds of thousands of tokens of context (Achiam et al., 2023). With the right set of prompts, they can be used to translate full documents (Wu et al., 2024; Briakou et al., 2024), potentially moving beyond the sentence-level paradigm. As machine translation expands to longer texts, a key challenge is developing reliable methods for automatic evaluation. Trained translation metrics have been shown to be able to evaluate long text spans (Vernikos et al., 2022; Raunak et al., 2024), despite being trained on sentence-level data. However, their application to longer texts is often constrained by the context window limitations of their base models, e.g., a 512-token limit (Conneau et al., 2020). At the same time, LLMs have demonstrated SOTA performance in evaluating short-form translations (Freitag et al., 2023, 2024; Kocmi and Federmann, 2023a), when prompted to produce error spans with categories defined by Multidimensional Quality Metrics (MQM) (Lommel et al., 2014). This raises the question: *can we feed increasingly longer translations into LLMs to achieve reliable long-form translation evaluation?* In this work, we define short-form to refer to single or few sentences, while long-form refers to longer units of text, such as multiple paragraphs or documents.

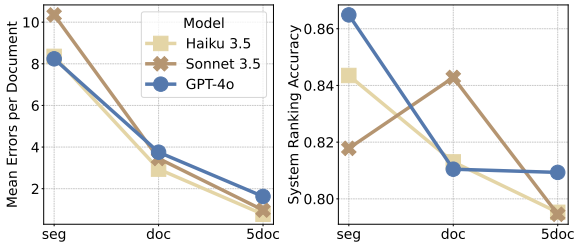

Figure 1: The number of predicted errors (left) and the average accuracy (right) for different input text granularities: segment-level, doc-level, and 5 doc-level.

Ideally, when providing a long document translation for evaluation, LLMs should thoroughly process all sentences and flag all errors present. However, we find that current LLMs are not "length-invariant": they detect significantly fewer errors when assessing an entire document at once, compared to the cumulative number of errors identified when processing the document one segment at a time (Figure 1, left). In other words, many errors are missed when evaluating long documents. Having fewer errors identified reduces the interpretability. Even worse, we find that for Claude 3.5 Haiku and GPT-4o, the average ranking accu-

*Now at Google. Correspondence to: domhant@google.com.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 7929–7947
November 4-9, 2025

racy also decreases as the input length increases (Figure 1, right). This aligns with prior work showing that LLMs struggle with reasoning tasks as the input length increases (Levy et al., 2024). We argue that evaluating long-form translations with LLMs requires more refined approaches. Our core contributions are: **(1)** a comprehensive analysis of length dependence in current LLMs; **(2)** a prompting scheme that ensures stable ranking accuracy and error detection across text granularities; and **(3)** practical guidelines for deploying LLMs in long-form translation evaluation, informed by results from diverse models and settings.

2 Length-invariant translation evaluation

LLMs are shown to be competitive with SOTA translation metrics when used to predict MQM error spans (Fernandes et al., 2023; Kocmi and Federmann, 2023a; Freitag et al., 2023, 2024). For example, Kocmi and Federmann (2023a) propose the GEMBA-MQM prompt to instruct GPT-4 to predict translation error spans, along with their severities. Error weights associated with severities are then summed at the segment or system level to produce a final quality score. However, we find that LLMs become less reliable when evaluating long-form translation outputs (Figure 1). This could be because long responses are less commonly encountered in NLP tasks.

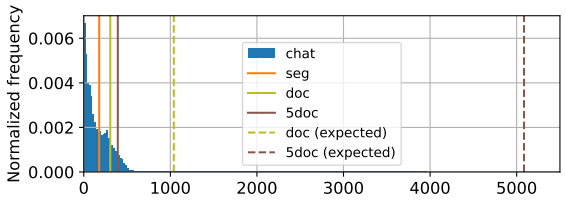

Figure 2: Comparison of chat response lengths (*chat*) compared to Claude 3.5 Sonnet's MQM response length (*seg*, *doc*, *5doc*) on WMT'24 EN-DE metrics task data in GPT-4o tokens. The expected length is based on the concatenation of the segment level responses.

In Figure 2, we contrast the response length of typical user interactions with chat systems, based on Chatbot Arena (Zheng et al., 2023), to the expected response length of Claude for MQM annotation of different text lengths.[1] We observe responses to longer inputs to be substantially shorter than responses concatenated from segment-level responses. For instance, asking Claude to flag all MQM errors in document-level translations yields an average of around 300 tokens (solid green line). However, if we split the same documents into segments, flag errors at the segment level,[2] and then concatenate all flagged errors, we obtain an average of ~1,000 tokens (dashed green line). In fact, 99% of general chat responses are shorter than 516 tokens, and response length required to cover all error spans in a document clearly falls outside this range. In this work, our goal is to develop an evaluation scheme that enables LLMs to reliably assess long-form translation, leading to consistent results across input granularities. To this end, we explore several prompting and fine-tuning strategies.

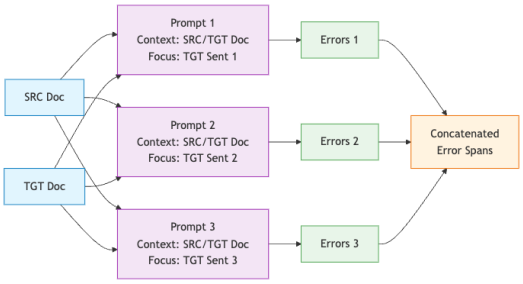

Figure 3: FSP on a three-sentence document.

Focus sentence prompting (FSP) To ensure invariance to input length, we consider individual sentences as the evaluation unit for MQM error span prediction. Specifically, we present the evaluation model with the full source and target documents, but prompt it to evaluate a single sentence at a time. We call this approach Focus Sentence Prompting (FSP). Here, we emphasize that we consider a realistic long-form translation scenario where the entire document is translated in a single pass, which better accommodates discourse phenomena (Maruf et al., 2022). Therefore, the translation may not strictly follow a one-to-one correspondence between source and target sentences. FSP effectively avoids the need for sentence alignment, a process that often introduces noise from wrong alignments. Moreover, providing the full source and translation text enables the model to detect context-dependent errors, such as those related to anaphora resolution. While FSP, by design, requires multiple inference passes, resulting in increased inference cost, this overhead can be mitigated by prompt caching, as all prompts for a document share the same prefix. Figure 3 illustrates the FSP schema, and the full prompt is provided in Appendix C.2.

[1]Three levels of text lengths are considered: *seg*, *doc*, *5doc*, with increased length. Refer to Section 3.1 for the definition.

[2]In WMT'24, parallel data is segment-level with metadata linking segments from the same document, enabling both segment- and document-level evaluation (via concatenation).

Granularity Matching The original GEMBA-MQM prompt uses a fixed set of three sentence-level demonstrations for MQM annotations, which can lead to significant mismatches in text granularity when the test case requires long-form translation evaluation, potentially causing the model to generate fewer error spans. To address this, we use two approaches. One method retains in-context learning (ICL) but selects five demonstrations that roughly match the length of the test example (**GMICL-5**). The other method fine-tunes LLMs on MQM data, similar to Fernandes et al. (2023), but at various text granularities, referred to as **GMFT**. The data source for both GMICL-5 and GMFT comes from the WMT'23 shared task data (Freitag et al., 2023).

Error Span Explanations Inspired by Chain-of-Thought (Wei et al., 2022), our MQM prompts (e.g., FSP and GMICL-5) ask LLMs to predict both error spans and their corresponding explanations unless stated otherwise. The error category and severity are then predicted based on the span-level explanation, aiming for more accurate judgment. In Appendix B.2, we show that adding explanations leads to higher system ranking accuracy.

Direct Assessment (DA) Instead of deriving translation quality scores from predefined error weights, LLMs can be prompted to predict a numerical score for quality assessment based on the identified error spans, a method referred to as Direct Assessment (DA) (Kocmi and Federmann, 2023b).

3 Experiments

3.1 Setup

We use the WMT'24 metrics shared task data for evaluation (Freitag et al., 2024). The data consists of human gold MQM annotations covering three translation directions: English → German (EN-DE), Japanese → Chinese (JA-ZH) and English → Spanish (EN-ES). We use system-level pairwise accuracy (Kocmi et al., 2021) as our evaluation metric. It measures the number of pairs of systems that are ranked correctly when compared to the ranking derived from human annotations. We use the official shared task scripts to access data and compute metrics.[3] Additionally, we measure the number of error spans per document and the character F1 score. The latter is also used by the WMT shared task on quality estimation (Blain et al., 2023) and based on the precision/recall of error spans compared to gold annotations per character with partial credit (0.5) for a mismatch in error severity (more details in Appendix B.5). We evaluate three proprietary LLMs: Claude 3.5 Haiku, Claude 3.5 Sonnet, and GPT 4o, as well as three open weight LLMs: Qwen2.5 14B/32B (Yang et al., 2025), and DeepSeek V3 (DeepSeek-AI et al., 2025), using a temperature of 0 for deterministic decoding.

The WMT'24 metrics shared task data is provided at the segment level, with each segment containing one or a few sentences. These segments originate from documents, which we reconstructed using the provided metadata. Additionally, we concatenate random groups of five documents to simulate extended long-form input. This results in three evaluation settings with different text granularities, referred to as *seg*, *doc*, and *5doc*. The gold MQM annotations for *doc* and *5doc* are derived by concatenating the MQM errors from the segment-level annotations. This leads to an average of 103/507/2713 GPT-4o tokens for the *seg*, *doc* and *5doc* cases. Similarly, we sample text data from WMT'23 with the three granularity levels for demonstrations and training data for GMICL-5 and GMFT. Refer to Appendix B.3 for data construction details. For GMFT, we fine-tune a GPT-4o model.

3.2 Addressing evaluation length dependence

Figure 4 shows the results of the evaluated prompt and fine-tuning settings across different text granularities. Compared to the GEMBA 3-shot baseline, which uses a fixed set of three segment-level MQM demonstrations, the GMICL-5 approach, despite being designed to match the input granularity, results in only marginal increases in error spans and still suffers a substantial drop in system ranking accuracy for longer texts. This suggests that **providing additional test-like demonstrations alone is not sufficient to overcome the response length bias (Figure 2) or to improve accuracy.**

In contrast, FSP effectively stabilizes the number of errors across text granularities and models. It also improves system ranking accuracy, particularly for moderate-sized LLMs and long-document scenarios (e.g., +12% accuracy with Qwen2.5-14B on 5doc compared to GEMBA), suggesting that LLMs can accurately identify the relevant source context for evaluating translation segments, even without explicit alignment. **Overall, FSP, when paired with strong LLM judges, can serve as**

[3] github.com/google-research/mt-metrics-eval

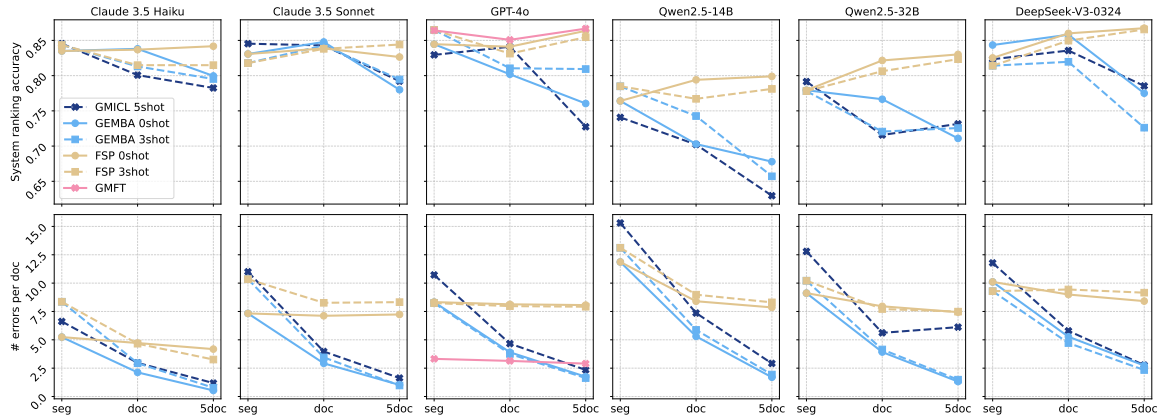

Figure 4: System ranking accuracy and number of error spans per document across methods and input granularities, averaged across translation directions. See Appendix B.4, B.5 for direction-specific results and character F1 scores.

a reliable, long-context, reference-free evaluation metric. For example, FSP with GPT-4o ranks second among official WMT'24 submissions (full ranking in Appendix B.6). Finally, we examine the impact of shot count in GEMBA and FSP on accuracy: no consistent pattern emerges in favor of 3-shot over 0-shot. The most effective setting depends on the specific LLMs and text granularity.

Our fine-tuning experiments with GPT-4o (GMFT) further demonstrate that a small amount of training data may also mitigate length bias. This results in a more consistent error count across text granularities while maintaining high system ranking accuracy, serving as an alternative to FSP. Interestingly, despite the overall stability, the predicted error count is lower at the *seg* and *doc* levels compared to other methods, suggesting that a high number of error spans may not be essential for higher ranking accuracy. The number of errors predicted by GMFT depend on the number of errors in the human annotation training data, which contains fewer, higher quality error spans. We therefore focus on comparing the number of error spans at different granularities for a given model, not comparing across models at a specific granularity.

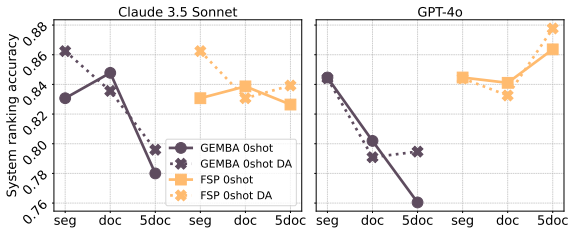

Figure 5: Impact of predicting a quality score via DA compared to computing a weighted MQM error score for GEMBA and FSP prompting.

Additionally, we explore whether supplementing GEMBA with DA could mitigate length dependence. In the DA setting a single quality score is predicted, rather than just identifying errors. Figure 5 reveals that DA, by itself, is still vulnerable to text length variations and does not improve ranking accuracy for long-form translations. Nevertheless, DA integrates well with FSP, likely because the finer-grained error spans provided by FSP help LLMs infer a more accurate quality score.

3.3 FSP inference efficiency

FSP increases input tokens by adding document context to each segment, raising concerns about inference efficiency. In Table 1, we compare the throughput in terms of error spans per second. Despite the increased number of input tokens for FSP, we can see that the throughput is comparable between FSP and GEMBA 3shot. The reason is that for autoregressive models inference time is dominated by the output token generation, whereas FSP only increases the number of input tokens. Additionally, prompt caching, as available in inference frameworks like SGLang (Zheng et al., 2024), is effective for FSP as all evaluation prompts for a given document share the same prefix. Note that the inference time is shorter for GEMBA 3shot due to producing fewer error spans at a lower accuracy.

4 Related work

Kocmi and Federmann (2023a) and Fernandes et al. (2023) show that LLMs can be effective translation evaluators when prompted to predict MQM error spans. Due to the lack of public document-level MQM data for meta-evaluation, their evaluation

	doc			5doc		
	duration	# spans/s	Acc.	duration	# spans/s	Acc.
		Qwen2.5-32B				
GEMBA 3shot	29min	36.6	72.1	13min	29.1	72.6
FSP 3shot	58min	34.4	**80.6**	58min	33.2	**82.3**
		DeepSeek-V3-0324				
GEMBA 3shot	63min	19.4	82.0	46min	13.3	72.6
FSP 3shot	151min	16.3	**85.0**	156min	15.2	**86.6**

Table 1: Inference performance comparison. Models are deployed with SGLang. Duration denotes wall-clock time to complete the WMT'24 metrics shared task.

focuses on sentence-level translations.[4] We overcome this data limitation by combining the data into longer text blocks comprising single or multiple documents. Fernandes et al. (2023) demonstrate that LLMs can be effectively fine-tuned for the MQM error-span prediction task. In the GMFT approach we fine-tune a model on the error span task, extending the setup to texts of different lengths.

For trained, dedicated translation metrics that predict quality scores without fine-grained errors or explanations, it has been shown that sentence-level metrics can be extended to evaluate longer texts (Raunak et al., 2024; Vernikos et al., 2022). xCOMET (Guerreiro et al., 2024) is a recent encoder-only model fine-tuned to be able to predict both a quality score and per-token error severities. As a fine-tuned model, it comes with the downside of not being able to use the latest LLMs, as well as not providing error explanations. The second downside was later overcome by xTower (Treviso et al., 2024), showing that error explanations are useful for error correction. We include error-span explanations, as we find that they improve the ranking accuracy.

5 Conclusion

In this work, we show that SOTA LLMs lack length invariance in translation assessment, detecting fewer error spans at the document level than when evaluating segments individually. The system ranking accuracy also drops with longer inputs. We observed this behavior across a range of open and closed models. To address this, we propose two simple yet effective methods for length-invariant, accurate evaluation.

[4]With the exception of WMT23 EN-DE being paragraph-level data.

Limitations

To ensure transparency and foster future research, we outline several limitations of our study below.

Limited Translation Directions Due to limitations in the publicly available MQM datasets that include document-level metadata, we were able to run experiments on only three language directions. As more MQM datasets become available, we encourage other researchers to replicate our experiments to see whether the findings hold for other language directions. While our work was limited to these three language directions, we have no strong reason to believe that our findings will not generalize to other language combinations.

Evaluation Scope We evaluated the impact of length on long-form translation assessment using the MQM metric. While MQM is a widely accepted standard, we did not extend it to explicitly address document-level phenomena such as anaphora resolution, coherence, or consistency across a document. However, we note the following: (1) achieving length invariance in long-form evaluation is a critical prerequisite that must be addressed before focusing on other important aspects of document-level translation quality; and (2) extending standard metrics falls outside the scope of this paper. Nonetheless, we consider this a promising direction for future research. In particular, future work could involve designing experiments to evaluate the ability of LLMs to identify and assess document-level translation errors, which may require annotated corpora.

Sentence Segmentation Requirement Our work assumes the availability of segment-level data. In practical applications, sentence segmentation would be necessary. However, this should not pose a significant challenge, as sentence segmentation tools are readily available. This assumption allows us to focus on the core issue of the lack of length invariance, which we see as a prerequisite for long-form translation evaluation. Note, that FSP would not able to penalize omissions of entire sentences, which is why advocate GMFT where available. Duplicate sentences, which are rare in practice, might requiring including context or assigning unique identifiers for FSP.

Acknowledgments

We thank Bill Byrne, Felix Hieber, Michael Denkowski, and Raúl Soutelo Quintela for their in-depth discussions and valuable feedback that helped shape this research. We would also like to thank our anonymous reviewers for their constructive feedback.

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*.

Frederic Blain, Chrysoula Zerva, Ricardo Rei, Nuno M Guerreiro, Diptesh Kanojia, José GC de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, et al. 2023. Findings of the wmt 2023 shared task on quality estimation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 629–653.

Eleftheria Briakou, Jiaming Luo, Colin Cherry, and Markus Freitag. 2024. Translating step-by-step: Decomposing the translation process for improved translation quality of long-form texts. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1301–1317, Miami, Florida, USA. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

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 *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083, Singapore. Association for Computational Linguistics.

Markus Freitag, Nitika Mathur, Daniel Deutsch, Chi-Kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Frederic Blain, Tom Kocmi, Jiayi Wang, David Ifeoluwa Adelani, Marianna Buchicchio, Chrysoula Zerva, and Alon Lavie. 2024. Are LLMs breaking MT metrics? results of the WMT24 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 47–81, Miami, Florida, USA. Association for Computational Linguistics.

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 *Proceedings of the Eighth Conference on Machine Translation*, pages 578–628, Singapore. Association for Computational Linguistics.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins.

2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Tom Kocmi and Christian Federmann. 2023a. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation*, pages 768–775, Singapore. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023b. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203, Tampere, Finland. European Association for Machine Translation.

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, Online. Association for Computational Linguistics.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. 2024. Same task, more tokens: the impact of input length on the reasoning performance of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15339–15353, Bangkok, Thailand. Association for Computational Linguistics.

Arle Lommel, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional quality metrics (mqm): A framework for declaring and describing translation quality metrics. *Tradumàtica*, (12):0455–463.

Sameen Maruf, Fahimeh Saleh, and Gholamreza Haffari. 2022. A survey on document-level neural machine translation: Methods and evaluation. *ACM Comput. Surv.*, 54(2):45:1–45:36.

Vikas Raunak, Tom Kocmi, and Matt Post. 2024. SLIDE: Reference-free evaluation for machine translation using a sliding document window. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 205–211, Mexico City, Mexico. Association for Computational Linguistics.

Brian Thompson, Nitika Mathur, Daniel Deutsch, and Huda Khayrallah. 2024. Improving statistical significance in human evaluation of automatic metrics via soft pairwise accuracy. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1222–1234, Miami, Florida, USA. Association for Computational Linguistics.

Marcos V Treviso, Nuno M Guerreiro, Sweta Agrawal, Ricardo Rei, José Pombal, Tania Vaz, Helena Wu, Beatriz Silva, Daan Van Stigt, and Andre Martins. 2024. xTower: A multilingual LLM for explaining and correcting translation errors. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15222–15239, Miami, Florida, USA. Association for Computational Linguistics.

Giorgos Vernikos, Brian Thompson, Prashant Mathur, and Marcello Federico. 2022. Embarrassingly easy document-level MT metrics: How to convert any pretrained metric into a document-level metric. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 118–128, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Minghao Wu, Thuy-Trang Vu, Lizhen Qu, George Foster, and Gholamreza Haffari. 2024. Adapting large language models for document-level machine translation. *arXiv preprint arXiv:2401.06468*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Livia Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E Gonzalez, et al. 2024. Sglang: Efficient execution of structured language model programs. *Advances in Neural Information Processing Systems*, 37:62557–62583.

A Experiment details

A.1 LLM versions

The experiments are based on these versions of Claude, GPT, and DeepSeek V3:

- Sonnet 3.5:
  anthropic.claude-3-5-sonnet-20241022-v2:0
- Haiku 3.5:
  anthropic.claude-3-5-haiku-20241022-v1:0
- GPT-4o:
  gpt-4o-2024-11-20
- DeepSeek V3:
  DeepSeek-V3-0324

B Inference infrastructure

We utilize the official APIs to prompt Claude and GPT models. Qwen and DeepSeek models are deployed through SGLang (Zheng et al., 2024) on NVIDIA H200 GPUs.

B.1 Data statistics

Table 2 contains the number of segments/documents and multi-documents as well as their average length in GPT-4o tokens both per language arc and the average across directions. The number of tokens is computed via the `tiktoken` library[5].

B.2 Error span explanation ablation

Figure 6 compares the GEMBA 3-shot prompt variant once with and once without error span explanations. We see that for both Claude 3.5 Haiku and GPT-4o error span explanations significantly improve system ranking accuracy, while showing the same pattern of decreasing accuracy with longer inputs. For GPT-4o the decrease in system ranking accuracy is even more pronounced when not using error span explanations. For Claude 3.5 Sonnet the accuracy is comparable between the variant with and without error span explanations. The ranking of what works best in terms of granularities is unchanged though.

B.3 Granularity-Matching Data Construction

We constructed a dataset with MQM annotations at various text granularities. Our data is derived from the WMT23 MQM dataset (Freitag et al., 2023), which was originally provided at the segment level for both source texts and machine translations. Similar to the approach used to create the WMT'24-based test data described in Section 3, we joined segments from the same documents to form document-level examples. Subsequently, we concatenated five randomly selected documents from these document-level examples to create *5doc* long examples. In total, our dataset comprises 43307 examples, including 35472, 6530, and 1305 examples for the granularities *seg*, *doc*, and *5doc*, respectively. These examples cover the translation directions English-German, Hebrew-English, and Chinese-English.

For GMICL-5, we randomly select five granularity-matched examples from our constructed dataset. For fine-tuning GPT-4o, we use all 43307 examples and fine-tuned the GPT-4o models for two epochs through OpenAI's API. It is important to note that the translation directions in training and testing differ, with only English-German overlapping. However, we find that fine-tuning still improved GPT-4o's length invariance across all tested directions.

The statistics for the resulting data at different granularities can be found in Table 2.

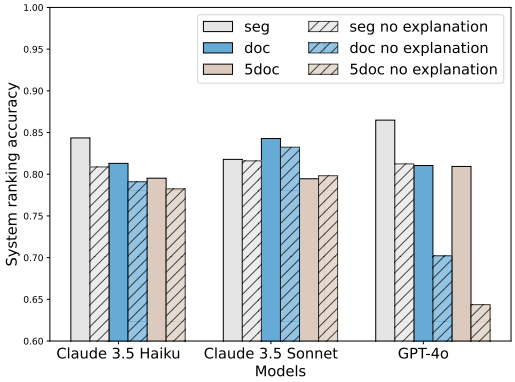

Figure 6: Ablation comparing the GEMBA 3-shot prompt variation with and without error span explanations.

B.4 Translation Direction-Specific Results

In Figure 7, we show the system ranking accuracy, number of error spans, and F1 score at various input granularities for individual translation directions: EN-DE, EN-ES, and JA-ZH. The results demonstrate that FSP generally outperforms other approaches across different translation directions and models, particularly in the 5doc case.

[5] https://github.com/openai/tiktoken

	EN-DE		JA-ZH		EN-ES		average	
Granularity	**# items**	**# tokens**	**# items**	**# tokens**	**# items**	**# tokens**	**# items**	**# tokens**
seg	27,944	90.4	16,606	129.1	23,952	88.0	22,834	102.5
doc	4,788	530.1	4,531	474.4	4,104	516.3	4,474	506.9
5doc	980	2814.2	920	2576.4	840	2747.0	913.3	2712.5

Table 2: WMT'24 metrics shared task evaluation data statistics. The number of tokens is the average number of source and translation text GPT-4o tokens at the given granularity.

B.5 Character-level precision/recall/F1

Implementation Details In order to compute character-level precision/recall/F1 we need to know the location of error spans. The GEMBA-MQM prompts however only result in error span strings without a specific location. While the majority of error spans is unique in the translation string (86.37% of Haiku 3.5's error spans using the FSP prompt on EN-DE data at the *doc5* granularity) there are shorter spans that occur multiple times. To assign a location for them we greedily search for all possible locations and pick the one that is unoccupied by any other error span choosing the one with the highest gold annotation overlap. This optimistically assumes the model refers to the correct location. As this is done equally for all models and systems this does not give an unfair advantage. The chosen location is marked as occupied for each character it spans and we move to the next error span. There may be error spans that can not be matched to a location, which will reduce the precision, while not affecting the recall. Once all error spans have been assigned to locations we can proceed to compute the character-level precision and recall. The precision computes the number of correctly predicted characters, while recall computes the number of gold characters that the model has covered.

Results Figure 8 contains the character-level precision, recall, and F1 under different evaluation setups. As can be seen, FSP not only results in a higher and more stable error distribution but also improves the overlap with gold spans, measured using character F1, when evaluating long documents. This suggests that the error spans predicted by the model more closely align with those identified by human annotators.

B.6 Comparison to WMT'24 shared task submissions

Our goal is to provide an evaluation setting that allows using off-the-shelf LLMs for long-form translation evaluation, not to produce the state-of-the-art in segment-level translation metrics. Nevertheless to see how our prompting setup compares to WMT'24 metrics task submissions we included results in terms of system ranking accuracy in Table 3. We see that, especially when using GPT-4o, our evaluation settings, that includes JSON outputs and error explanations, are competitive with state-of-the-art metrics at the segment level. Note, that WMT'24 moved from system ranking accuracy to a soft variant that takes the uncertainty into account, Soft Pairwise Accuracy (SPA) (Thompson et al., 2024). We chose to continue using system ranking accuracy as SPA requires segment-level scores, which makes it more challenging to compare the same evaluations across different text granularities as we do in this work.

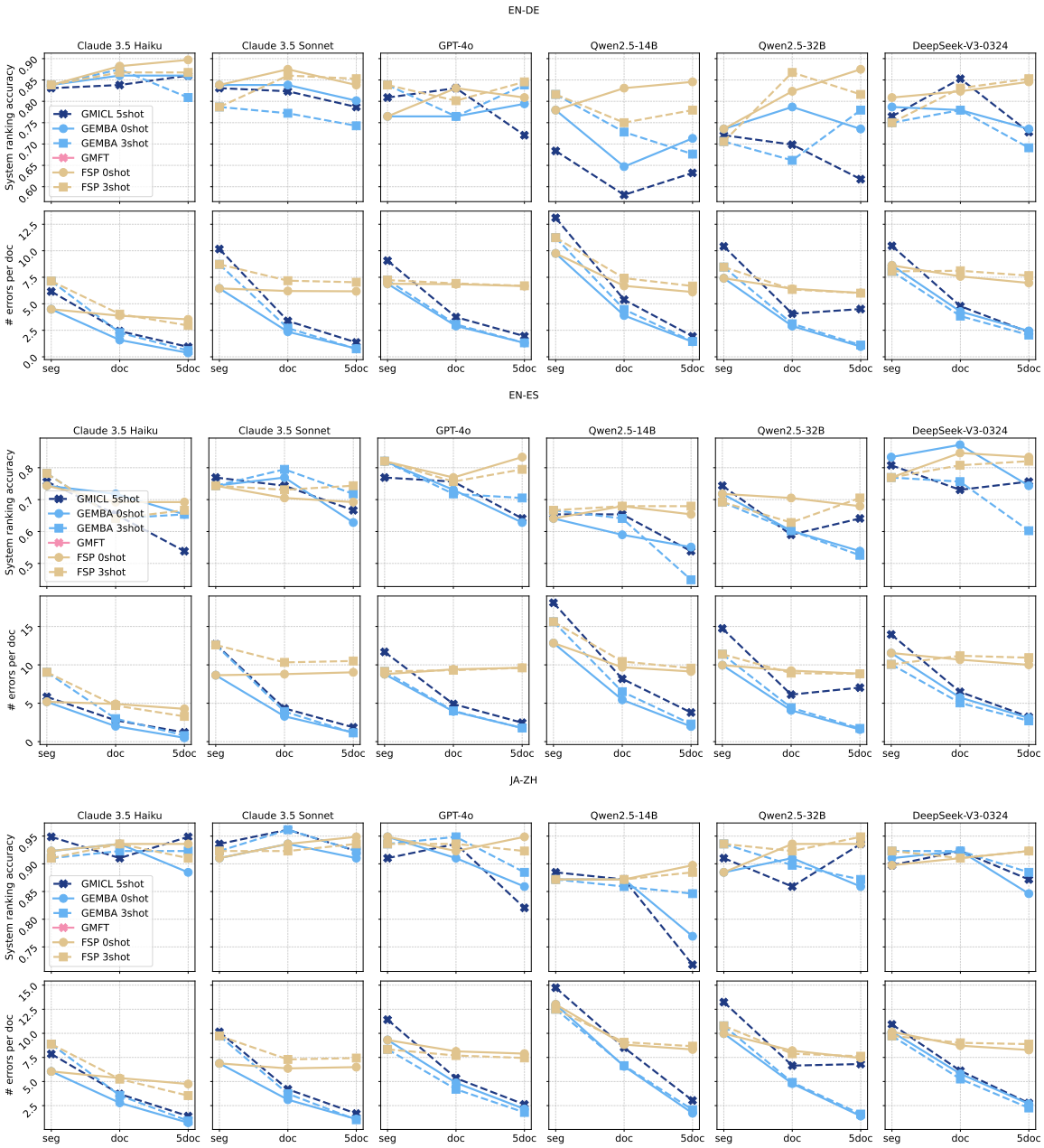

Figure 7: System ranking accuracy and number of error spans at different input granularities for individual translation directions: EN-DE, EN-ES, and JA-ZH.

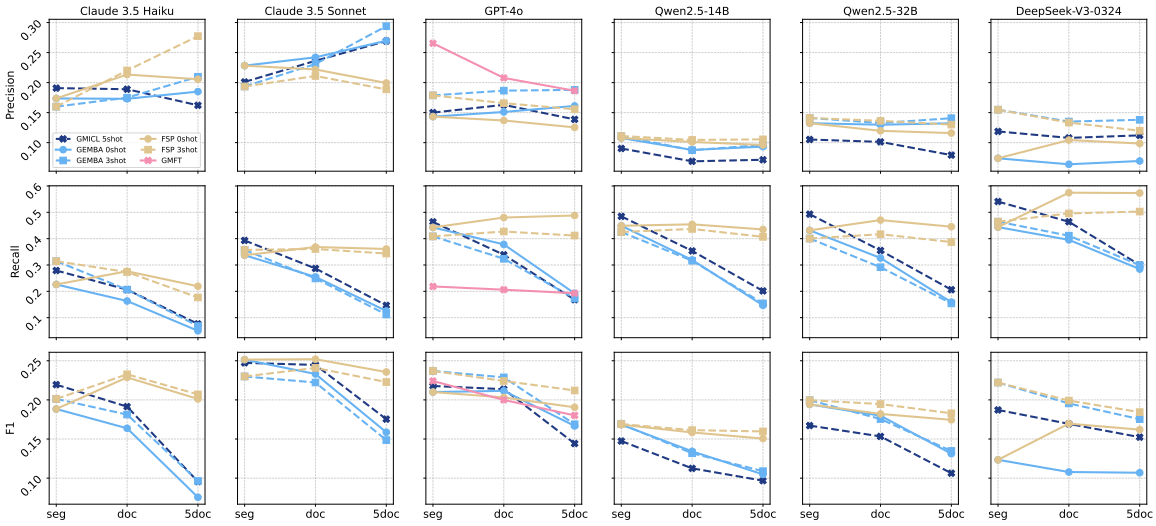

Figure 8: Character precision, recall and F1, averaged across translation directions.

System	ours?	system ranking accuracy
*CometKiwi-XXL[noref]		0.8692
GPT-4o FSP 3shot[noref]	✓	0.8649
*MetricX-24		0.8655
MetricX-24-Hybrid		0.8594
gemba_esa[noref]		0.8520
*metametrics_mt_mqm_same_source_targ		0.8485
metametrics_mt_mqm_hybrid_kendall		0.8485
XCOMET		0.8472
*metametrics_mt_mqm_kendall		0.8460
*MetricX-24-QE[noref]		0.8454
Claude 3.5 Haiku FSP 3shot[noref]	✓	0.8435
MetricX-24-Hybrid-QE[noref]		0.8338
_BLEURT-20		0.8210
Claude 3.5 Sonnet FSP 3shot[noref]	✓	0.8178
DeepSeek V3 0324 FSP 3shot[noref]	✓	0.8141
_COMET-22		0.8063
XCOMET-QE[noref]		0.7996
BLCOM_1		0.7861
Qwen2.5 14B FSP 3shot[noref]	✓	0.7848
bright-qe[noref]		0.7818
*metametrics_mt_mqm_qe_same_source_t[noref]		0.7782
Qwen2.5 32B FSP 3shot[noref]	✓	0.7780
metametrics_mt_mqm_qe_kendall.seg.s[noref]		0.7739
_PrismRefMedium		0.7482
_PrismRefSmall		0.7482
_sentinel-cand-mqm[noref]		0.7428
_YiSi-1		0.7274
damonmonli		0.7267
_CometKiwi[noref]		0.7170
*monmonli		0.7139
_BERTScore		0.7061
MEE4		0.6914
_chrF		0.6889
chrfS		0.6877
_spBLEU		0.6682
_BLEU		0.6273
*BLCOM		0.5507
*XLsimDA[noref]		0.4634
XLsimMqm[noref]		0.4634
_sentinel-ref-mqm		0.0000
_sentinel-src-mqm[noref]		0.0000

Table 3: Comparison to the official WMT'24 submissions in terms of system ranking accuracy averaged across three language directions. [noref] denotes metrics that are reference free.

C Prompts

C.1 GEMBA prompt variation

We modify the GEMBA-MQM (Kocmi and Federmann, 2023a) prompt by changing the output format to JSON for easier parsing and error span explanations, and we present this modified prompt in Figure 9.

C.2 Focus Sentence Prompting (FSP)

We present the complete FSP prompt in Figure 13.

GEMBA 3shot prompt with JSON format and error span explanations

You are an annotator for the quality of machine translation. Your task is to identify errors and assess the quality of the translation using MQM. Based on the source text (in <source></source> tags) and machine translation surrounded (in <translation></translation> tags), identify error types in the translation and classify them. The categories of errors are: accuracy (addition, mistranslation, omission, untranslated text, wrong language), fluency (character encoding, grammar, inconsistency, punctuation, register, spelling), style (awkward), terminology (inappropriate for context, inconsistent use), other. Each error, including omissions or untranslated content, 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 technically errors, but do not disrupt the flow or hinder comprehension. The source text muss be fully covered and any omissions should also be annotated as errors. Please only include errors and no spans that do not contain errors.

Please respond in JSON following this schema:

```
{
  "type": "object",
  "properties": {
    "errors": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "error_span": {
            "type": "string",
            "description": "The relevant input span where the error occurred."
          },
          "explanation": {
            "type": "string",
            "description": "A brief explanation of the error and its impact"
          },
          "error_category": {
            "type": "string",
            "enum": ["accuracy", "fluency", "style", "terminology", "other"],
            "description": "The main category of the error"
          },
          "error_type": {
            "type": "string",
            "description": "The specific type of error within the category"
          },
          "severity": {
            "type": "string",
            "enum": ["critical", "major", "minor"],
            "description": "The severity level of the error"
          },
        },
        "required": ["explanation", "error_category", "error_type", "severity"]
      }
    }
  },
  "required": ["errors"]
}
```

<<Example 1>>

<<Example 2>>

<<Example 3>>

Please score the following input

```
<input>
<source_language>{{ src_lang }}</source_language>
<source>{{ src }}</source>
<target_language>{{ tgt_lang }}</target_language>
<translation>{{ output_seq }}</translation>
</input>
```

Please respond in JSON without any introduction or explanation. Only the JSON response is required.

MQM:

Figure 9: The GEMBA 3-shot prompt augmented with error span explanations. For a clearer presentation, the three examples are shown separately in Figures 10, 11, and 12.

GEMBA 3shot prompt with JSON format and span explanations Example 1

<input>
<source_language>English</source_language>
<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.</source>
<target_language>German</target_language>
<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.</translation>
</input>

MQM:

```
{
  "errors": [
    {
      "error_span": "involvement",
      "explanation": "The word 'involvement' is mistranslated and doesn't fit in the context of the
      German sentence.",
      "error_category": "accuracy",
      "error_type": "mistranslation",
      "severity": "major"
    },
    {
      "error_span": "",
      "explanation": "The phrase 'the account holder' is omitted in the German translation, losing
      important information.",
      "error_category": "accuracy",
      "error_type": "omission",
      "severity": "major"
    },
    {
      "error_span": "wäre",
      "explanation": "The use of 'wäre' (subjunctive) is grammatically incorrect in this context. It should
      be 'bin' (indicative).",
      "error_category": "fluency",
      "error_type": "grammar",
      "severity": "minor"
    },
    {
      "error_span": "dir",
      "explanation": "The use of 'dir' (informal 'you') is inappropriate for the register of this text. It
      should be 'Ihnen' (formal 'you').",
      "error_category": "fluency",
      "error_type": "register",
      "severity": "minor"
    }
  ]
}
```

Figure 10: Example 1 from the GEMBA 3-shot prompt in Jinja format.

GEMBA 3shot prompt with JSON format and span explanations Example 2

<input>
<source_language>English</source_language>
<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.</source>
<target_language>Czech</target_language>
<translation>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.</translation>
</input>

MQM:

```
{
  "errors": [
    {
      "error_span": "ve Vídni se ve Vídni",
      "explanation": "The phrase 've Vídni' (in Vienna) is unnecessarily repeated, adding redundant
      information.",
      "error_category": "accuracy",
      "error_type": "addition",
      "severity": "major"
    },
    {
      "error_span": "",
      "explanation": "The phrase 'the stop-start' is omitted in the Czech translation, losing the
      characterization of the negotiations as intermittent.",
      "error_category": "accuracy",
      "error_type": "omission",
      "severity": "major"
    },
    {
      "error_span": "partaje",
      "explanation": "The word 'partaje' (parties) is inappropriate for this context. A more formal term
      like 'strany' (sides) would be more suitable.",
      "error_category": "terminology",
      "error_type": "inappropriate for context",
      "severity": "minor"
    }
  ]
}
```

Figure 11: Example 2 from the GEMBA 3-shot prompt in Jinja format.

GEMBA 3shot prompt with JSON format and span explanations Example 3

<input>
<source_language>Chinese</source_language>
<source>大众点评乌鲁木齐家居商场频道为您提供高铁居然之家地址,电话,营业时间等最新商户信息, 找装修公司,就上大众点评</source>

<target_language>English</target_language>
<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.</translation>
</input>

MQM:

```
{
  "errors": [
    {
      "error_span": "of high-speed rail",
      "explanation": "The phrase 'of high-speed rail' is incorrectly added to the translation. It's not
      present in the original Chinese text and doesn't make sense in this context.",
      "error_category": "accuracy",
      "error_type": "addition",
      "severity": "critical"
    },
    {
      "error_span": "go to the reviews",
      "explanation": "The phrase 'go to the reviews' is a mistranslation. The original Chinese text refers
      to using Dianping (a review platform), not simply going to reviews.",
      "error_category": "accuracy",
      "error_type": "mistranslation",
      "severity": "major"
    },
    {
      "error_span": "etc.,",
      "explanation": "The use of 'etc.' is inappropriate in this context, as it introduces vagueness to the
      translation. The list should be explicit and complete.",
      "error_category": "style",
      "error_type": "awkward",
      "severity": "minor"
    }
  ]
}
```

Figure 12: Example 3 from the GEMBA 3-shot prompt in Jinja format.

Focus Sentence Prompting (FSP)

You are an annotator for the quality of machine translation. Your task is to identify errors and assess the quality of the translation using MQM. Based on the source text (in <source></source> tags) and machine translation surrounded (in <translation></translation> tags), identify error types in the translation and classify them. The categories of errors are: accuracy (addition, mistranslation, omission, untranslated text, wrong language), fluency (character encoding, grammar, inconsistency, punctuation, register, spelling), style (awkward), terminology (inappropriate for context, inconsistent use), other. Each error, including omissions or untranslated content, 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 technically errors, but do not disrupt the flow or hinder comprehension. The source text muss be fully covered and any omissions should also be annotated as errors. Please only include errors and no spans that do not contain errors. You will be given a full document and its translations, but only score one sentence at a time which is given in <target_segment></target_segment> tags.

Please respond in JSON following this schema:

```
{
  "type": "object",
  "properties": {
    "errors": {
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "error_span": {
            "type": "string",
            "description": "The relevant input span where the error occurred."
          },
          "explanation": {
            "type": "string",
            "description": "A brief explanation of the error and its impact"
          },
          "error_category": {
            "type": "string",
            "enum": ["accuracy", "fluency", "style", "terminology", "other"],
            "description": "The main category of the error"
          },
          "error_type": {
            "type": "string",
            "description": "The specific type of error within the category"
          },
          "severity": {
            "type": "string",
            "enum": ["critical", "major", "minor"],
            "description": "The severity level of the error"
          },
        },
        "required": ["explanation", "error_category", "error_type", "severity"]
      }
    },
    "quality_score": {
      "type": "integer",
      "description": "Overall quality score of the translation. After highlighting all errors, please choose the overall
      quality score. The quality levels associated with numerical scores: 0: No meaning preserved: Nearly all information
      is lost in the translation. 33: Some meaning preserved: Some of the meaning is preserved but significant parts are
      missing. The narrative is hard to follow due to errors. The text may be phrased in an unnatural/awkward way.
      Grammar may be poor. 66: Most meaning preserved and few grammar mistakes: The translation retains most of the
      meaning. It may have some grammar mistakes or minor inconsistencies. 100: Perfect meaning and grammar: The
      meaning and grammar of the translation is completely consistent with the source. The text sounds like native text
      in the the target language without any awkward phrases. Use any number in the range between 0 and 100 for a
      fine-grained quality score."
    }
  },
  "required": ["errors", "quality_score"]
}
```

Please score the following input

```
<input>
<source_language>{{ src_lang }}</source_language>
<source>{{ src }}</source>
<target_language>{{ tgt_lang }}</target_language>
<translation>{{ output_seq }}</translation>
<target_segment>{{ target_segment }}</target_segment>
</input>
```

Please respond in JSON without any introduction or explanation. Only the JSON response is required. Use the full document as context while only scoring the translation segment given in <target_segment></target_segment> tags.

MQM:

Figure 13: The FSP prompt in Jinja format.

C.3 GMICL-5 prompting

We construct *doc* and *5doc* examples from the following WMT'23 metrics shared task gold annotations:

- Doc: `news_guardian.114833:en-de`, System: `NLLB_MBR_BLEU`
- Doc: `mastodon_mathewdiekhake.110349821603822000:en-de`, System: `refA`
- Doc: `userreview_automotive-2-en_0371449-77:en-de`, System: `AIRC`
- Doc: `speech_elitr_minuting-10:en-de`, System: `GPT4-5shot`
- Doc: `news_leadership-en.43063:en-de`, System: `ONLINE-M`
- Doc: `userreview_luggage-2-en_0553796-30:en-de`, System: `NLLB_MBR_BLEU`

The GMICL-5 is presented in Figure 14.

GMICL-5 prompt

You are an annotator for the quality of machine translation. Your task is to identify errors and assess the quality of the translation using MQM. Based on the source text (in <source></source> tags) and machine translation surrounded (in <translation></translation> tags), identify error types in the translation and classify them. The categories of errors are: accuracy (addition, mistranslation, omission, untranslated text, wrong language), fluency (character encoding, grammar, inconsistency, punctuation, register, spelling), style (awkward), terminology (inappropriate for context, inconsistent use), other. Each error, including omissions or untranslated content, 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 technically errors, but do not disrupt the flow or hinder comprehension. The source text muss be fully covered and any omissions should also be annotated as errors. Please only include errors and no spans that do not contain errors.

Please respond in JSON following this schema:
<<SAME AS THE FSP PROMTP>>

Here are some examples:
<<FIVE GRANULARITY MATCHED EXAMPLES>>

Please score the following input
<input>
<source_language>{{ src_lang }}</source_language>
<source>{{ src }}</source>
<target_language>{{ tgt_lang }}</target_language>
<translation>{{ output_seq }}</translation>
</input>

Please respond only in JSON without any introduction. Only the JSON response is required. Unlike the examples you will include error span explanations and a final quality_score.

MQM (with explanation, with quality_score):

Figure 14: The GMICL-5 prompt in Jinja format. For a clearer presentation, we have omitted the JSON schema for the MQM error annotations, which is identical to the one in Figure 13, as well as the content of the five documents and their translations.