Guiding Large Language Models to Post-Edit Machine Translation with Error Annotations

Dayeon Ki
Computer Science
University of Maryland
dayeonki@cs.umd.edu

Marine Carpuat
Computer Science, UMIACS
University of Maryland
marine@cs.umd.edu

Abstract

Machine Translation (MT) remains one of the last NLP tasks where large language models (LLMs) have not yet replaced dedicated supervised systems. This work exploits the complementary strengths of LLMs and supervised MT by guiding LLMs to automatically post-edit MT with external feedback on its quality, derived from Multidimensional Quality Metric (MQM) annotations. Working with LLaMA-2 models, we consider prompting strategies varying the nature of feedback provided and then fine-tune the LLM to improve its ability to exploit the provided guidance. Through experiments on Chinese-English, English-German, and English-Russian MQM data, we demonstrate that prompting LLMs to post-edit MT improves TER, BLEU and COMET scores, although the benefits of fine-grained feedback are not clear. Fine-tuning helps integrate fine-grained feedback more effectively and further improves translation quality based on both automatic and human evaluation.[1]

1 Introduction

Machine Translation (MT) remains one of the last NLP tasks where large language models (LLMs) have not yet replaced dedicated supervised systems. LLMs such as ChatGPT (Ouyang et al., 2022) started outperforming commercial MT systems very recently (Vilar et al., 2023; Hendy et al., 2023; Jiao et al., 2023). However, supervised models continue to outperform LLMs in numerous language pairs (Zhu et al., 2023; Kocmi et al., 2023), and the performance of LLMs remains uneven, exhibiting significant variation across models, languages, and translation directions (Bawden and Yvon, 2023; Zhu et al., 2023). This suggests that LLMs and supervised systems possess complementary strengths, and that combining them should offer some benefits.

[1]We release our code, dataset, model checkpoints at https://github.com/dayeonki/mt_feedback.

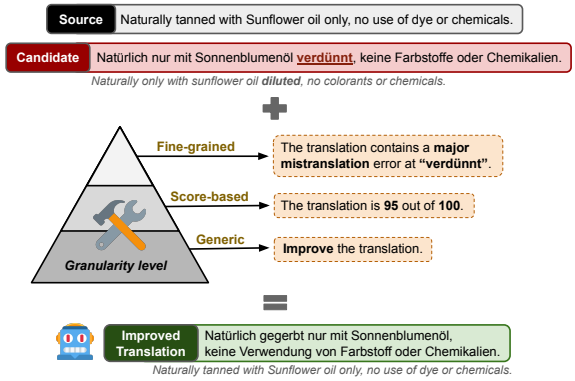

Figure 1: Guiding LLMs with external feedback enhances MT post-editing capabilities. We categorize feedback into different granularity: Generic, Score-based, and Fine-grained. Fine-grained feedback is annotated either by humans or automatic evaluation tools.

In this work, we propose to leverage LLM's text rewriting abilities (Brown et al., 2020; Reif et al., 2022; Raheja et al., 2023; Alves et al., 2024) to improve MT outputs given error annotations. *If we provide an LLM with a source sentence, a MT translation of arbitrary origin, and some feedback on the quality of the MT* (Figure 1), *can we reliably improve the quality of the MT?* This approach can be seen as revisiting the task of MT post-editing (Knight and Chander, 1994; Simard et al., 2007) in the light of recent work highlighting LLMs' ability to refine its own outputs (Madaan et al., 2023; Zeng et al., 2023; Chen et al., 2023). Indeed Chen et al. (2023); Raunak et al. (2023); Xu et al. (2024) recently show the promise of using LLMs for improving MT via refinement. We depart from these three papers by guiding the refinement abilities of LLMs with external feedback rather than self-generated feedback, and by post-editing outputs from arbitrary models rather than improve the LLM's own outputs only. Perhaps most importantly, while they relied exclusively on the largest closed LLMs – GPT3.5 (Brown et al., 2020), GPT4

Findings of the Association for Computational Linguistics: NAACL 2024, pages 4253–4273
June 16-21, 2024 ©2024 Association for Computational Linguistics

(OpenAI, 2023), PaLM-2 (Anil et al., 2023) – we argue that it is also worth exploring to what extent LLMs of more moderate size (e.g., 7B, 13B) can perform post-editing, as such models are less costly to train, run, and deploy in actual applications. This leads us to explore a different set of strategies. We further work with open models facilitating reproducibility of our results and hopefully encourages others to build on this work.

We explore a range of techniques to guide LLaMA-2 models (Touvron et al., 2023) to improve MT outputs using fine-grained feedback derived from Multidimensional Quality Metric (MQM) annotations (Freitag et al., 2021), as shown in Figure 1. Following prior work on refinement, we start with evaluating the impact of such feedback when prompting LLMs in zero-shot and few-shot settings (§5). Different from prior work, we then explore fine-tuning the LLM to advance its ability to improve translations based on the feedback provided in the prompt, in an instruction following style (Taori et al., 2023) (§6).

Through extensive experiments with three language pairs (Chinese-English, English-German, and English-Russian), we show that prompting LLMs to edit MT with feedback reliably improves translation quality as measured by automatic metrics, particularly in the few shot settings where the LLaMA-2 7B model achieves close peformance to the 13B version (§5). However, the models are unable to make the most of the fine-grained feedback which performs roughly on par with generic prompts for improvement. Instruction fine-tuning shows stronger improvements on translation quality based on both automatic and human evaluation (§6). Our analysis reveals that prompting the fine-tuned LLMs with fine-grained feedback not only helps fix the errors highlighted in the prompt (§7), but also leads to more natural outputs.

2 Related Work

MT Error Annotation. An increasing body of work seeks to evaluate MT by providing actionable feedback rather than a single score aggregating diverse dimensions of quality. Freitag et al. (2021) introduce an evaluation methodology based on the multi-dimensional human evaluation (MQM) framework (Lommel et al., 2014) to guide human annotators in identifying spans of translated text that are errors, labeling their types and severity level using a rich taxonomy. Their work inspired automatic approaches to error annotation, building on existing work on automatic evaluation of text generation (Sellam et al., 2020; Fu et al., 2023). These include generating a scalar score to represent MT quality as a whole (Xu et al., 2024; Fu et al., 2023; Fernandes et al., 2023), and more nuanced methods that detail error severity (Kocmi and Federmann, 2023b), error span, and type (Kocmi and Federmann, 2023a), aligning closely with human judgements (Liu et al., 2023). Additionally, learned evaluation metrics have also emerged, pinpointing fine-grained aspects (error span, type, severity level) of MT errors (Guerreiro et al., 2023; Xu et al., 2024) and providing detailed error explanations (Xu et al., 2023). We build on this work by comparing them using human annotated vs. machine annotated errors as feedback to refine MT outputs.

Paper	Model	Feedback	Prompting
Chen et al. (2023)	ChatGPT	Self-generated	Zero-shot
Raunak et al. (2023)	GPT-4	Self-generated	Zero-shot
Xu et al. (2024)	PaLM	Self-generated	Zero-shot
Ours	LLaMA-2	External	Zero-, Few-shot, Fine-tune

Table 1: Smaller models lead us to explore a wider range of settings for post-editing with LLMs.

MT Post-Editing. Recognizing that translation is an iterative process, automatic post-editing originally aimed to improve an original MT provided as input together with the source text (Knight and Chander, 1994; Simard et al., 2007; Chatterjee et al., 2018). Approaches have mirrored progress in MT, starting with statistical phrase-based models (Simard et al., 2007), multi-source neural encoder-decoder models (Junczys-Dowmunt and Grundkiewicz, 2016) and non-autoregressive Transformers (Gu et al., 2019; Wan et al., 2020). Most recent work relies on LLMs, relaxing the requirement for supervised examples of post-editing. Chen et al. (2023) perform refine MT outputs from a wide range of systems and languages using GPT3.5 (Brown et al., 2020), leading to a decrease of string-based quality metrics and comparable if not improved neural metrics. Human evaluation showed that this approach primarily reduces "translationese" in MT outputs. Raunak et al. (2023) frame post-editing as chain-of-thought (Kojima et al., 2023) and show that GPT-4 (OpenAI, 2023) improves COMET scores for MS Translator outputs across language pairs, particularly into English. Finally, in a contemporaneous pre-print, Xu et al. (2024) cast iterative refinement as a search

Category	Prompt
Generic	Improve the translation from English to German without any explanation. English: *The newer items are bagged only.* German: *Neue Gegenstände werden nur mit Gepäck versehen.* Improved German:
Score	Improve the translation from English to German without any explanation. This translation is scored 85 out of 100. English: *The newer items are bagged only.* German: *Neue Gegenstände werden nur mit Gepäck versehen.* Improved German:
Fine-grained	Improve the translation from English to German based on the identified errors without any explanation. (1) There is a major mistranslation error at "mit Gepäck versehen". English: *The newer items are bagged only.* German: *Neue Gegenstände werden nur mit Gepäck versehen.* Improved German:

Table 2: Exemplar prompt template of English-German language pair used for prompting experiments. The part highlighted in orange is the added component from the **Generic** prompt accordingly to each feedback category.

process that takes as input a current MT and automatically generated MQM style error information. Using the PaLM2 LLM (Anil et al., 2023), they show that this search improves the quality of the LLM's original translations on Chinese-English and German-English WMT tasks. Building on these encouraging results obtained with large closed models, we investigate whether smaller open LLMs can also achieve strong post-editing capabilities, which leads to explore a wider range of settings as summarized in Table 1.

LLM Self-Refinement. LLMs have been reported to "self-correct" an initial draft by iteratively refining it based on self-provided feedback for many tasks Pan et al. (2023). Briefly, past work has focused on generation tasks including mathematical program synthesis, lexically-constrained generation, and toxicity control (Welleck et al., 2023), reasoning tasks (Paul et al., 2024), and a range of generation, math reasoning, and code optimization tasks (Madaan et al., 2023), among others. Many works focus on incorporating self-refinement to MT (Chen et al., 2023; Raunak et al., 2023; Xu et al., 2024) where given source and MT translation, LLMs generate feedback and improve upon it. In the same vein, we study MT refinement with an LLM, but incorporate error annotations from various source as feedback to refine MT outputs.

3 Method

We consider two strategies for guiding language models to edit MT error annotations: prompting and fine-tuning with instructions.

3.1 Prompting

We consider zero-shot and few-shot prompting. The specific prompt templates used for each feedback level are outlined in Table 2, and provide a source text, a MT output and depending on the condition some feedback on the quality of the MT. We opt to construct our prompt templates in English, rather than the target language, as they have shown better performance (Lin et al., 2022), likely due to the greater prevalence of English in the pre-training data (Ahuja et al., 2023).

Our study encompasses the following forms of feedback for each model, as illustated in Table 2:

- **Generic**: The model is prompted to improve the initial translation without any specific external feedback.
- **Score**: A single scalar MQM score[2], reflecting the initial translation's overall quality, is provided to the model. We normalize the scores on a range from 0 to 100.
- **Fine-grained**: The model is provided with fine-grained feedback (error span, type, severity level) in the MQM style.

For the **Fine-grained** condition, we consider three distinct sources of error annotation:

- **MQM**: human annotation from the MQM WMT22 dataset (Kocmi et al., 2022).
- **InstructScore**: automatic annotation by InstructScore (Xu et al., 2023), an explainable text

[2]MQM scores are derived automatically from the identified error spans and their categories (Fernandes et al., 2023), based on a weighting scheme illustrated in Appendix Table 6.

generation evaluation metric, which fine-tunes LLaMA (Touvron et al., 2023) to predict MQM style fine-grained error annotations. This metric only supports Chinese-English.

- **xCOMET**: automatic annotation by xCOMET (Guerreiro et al., 2023), an automatic evaluation and quality estimation tool, which fine-tunes XLM-RoBERTa (Conneau et al., 2020) to predict both MQM and Direct Assessment (Graham et al., 2013) annotations of MT quality.

The three methods use different severity level ranges, and xCOMET does not provide error type information. See Appendix A for further details.

3.2 Fine-tuning

In the fine-tuning case, we focus on two types of feedback: generic and fine-grained feedback, to establish the ability of fine-tuning to guide LLMs for post-editing. First, generic and fine-grained feedback consistently shows better performance compared to the score-based baseline. Second, fine-grained feedback uses human annotation thus disentangling error annotation errors from post-editing errors. We leave the exploration of automatically generated feedback to future work.

For fine-grained feedback, we explore two fine-tuning settings: (1) **Bilingual**, where we individually fine-tune for each language pair and (2) **Multilingual**, where we combine three language pairs to fine-tune a single model. We construct fine-tuning datasets from two sources of MT human-annotated with errors: MQM (Freitag et al., 2021) and DEMETR (Karpinska et al., 2022). DEMETR provides MT error annotations in 10 source languages into English direction. Therefore, we use De-En from DEMETR as En-De pair and Ru-En as En-Ru. We reformulate all annotations in an instruction-following style (see Appendix Table 10 for examples). The fine-tuning data statistics are summarized in Table 3. We automatically filter out instances that share identical source or target sentences with those in the test set to ensure a clean train/test separation.

4 Experimental Setup

4.1 Datasets

Data. We experiment with WMT-22 General machine translation task submissions (Kocmi et al., 2022) annotated with MQM dimensions[3]. We focus on three language pairs: Chinese-English (zh-en), English-German (en-de), and English-Russian (en-ru). We evaluate on 1,000 WMT data instances for each language pair. Each sample contains one error span of average length ranging from 9 for En-Ru to 13 for Zh-En. Adequacy errors and minor errors dominate across languages. See Appendix C.1 for further details.

Language pair	# of train	# of dev	# of test
Zh-En	22,373 / 3,200	200	1,000
En-De	13,215 / 3,200	200	1,000
En-Ru	19,450 / 3,200	200	1,000

Table 3: Dataset statistics for fine-tuning instruction datasets. We use DEMETR as train set and split the MQM dataset into train, validation, and test sets. For **# of train** column, we represent as # of train (MQM) / # of train (DEMETR).

Error Annotations. In addition to the manual error annotations described above, we obtain automatic annotations of the same data using InstructScore and xCOMET[4].

To assess how much these different annotations agree with each other, we compute the overlap frequency for each pair of annotation method on a random sample of 200 test cases per language pairs. The overlap frequency measures how often error spans match across two sources of annotations. We observe that the overlap frequency between MQM and xCOMET is 33/200 for En-De and 42/200 for En-Ru. Notably, for Zh-En pair, xCOMET and InstructScore show the highest concordance (51/200), while overlaps with MQM are lower (24/200 with xCOMET and 25/200 with InstructScore). This discrepancy underscores that the automatic annotations are far from perfect. We will test whether they can nevertheless be useful.

4.2 Metrics

We report the traditional BLEU metric (Papineni et al., 2002) with exponential smoothing as implemented in the `sacrebleu` toolkit (Post, 2018), the Translation Edit Rate (TER) (Snover et al., 2006) which is the minimum number of edits

[3] https://github.com/google/wmt-mqm-human-evaluation

[4] We ensure that our data is not in their training set: InstructScore is trained on self-generated dataset from GPT-4 (OpenAI, 2023) and xCOMET is trained on MQM annotations but excluded the WMT-22 General MT task submissions, which they also reserved for testing.

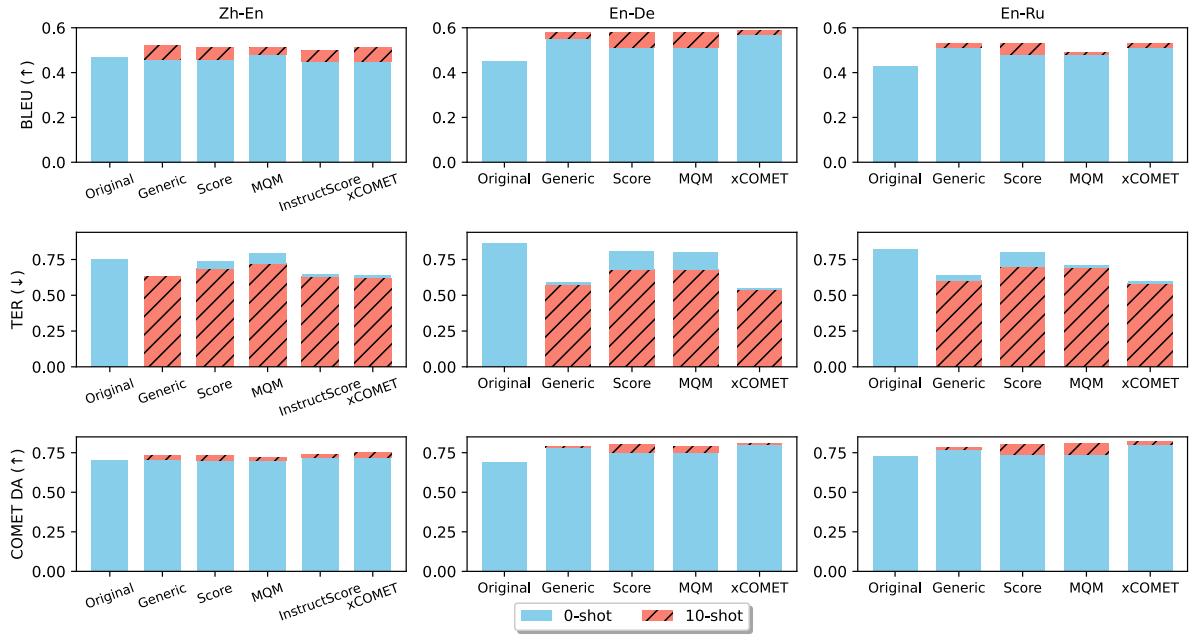

Figure 2: Zero- and 10-shot prompting results for LLaMA-2 7B. *Top*: BLEU scores for Chinese-English (Zh-En), English-German (En-De), English-Russian (En-Ru) pairs; *Middle*: Translation Edit Rate (TER) where zero-shot results show the amount increased compared to that of 10-shot; *Bottom*: COMET_{DA} scores. Note that we only report the supporting language pair (zh-en) results for InstructScore. Numerical results for both 7B and 13B are in Appendix E.1.

needed to change a hypothesis so that it exactly matches one of the references, normalized by the average length of the references, and a modern neural metric, the reference-based COMET_{DA} score (Rei et al., 2020). Scores for all these metrics are reported in the 0-1 range.

4.3 Models

We employ the widely-used open-source LLM LLaMA-2 (Touvron et al., 2023), experimenting with the 7B and 13B variants.[5]

Prompting settings. We set the temperature to 0 for greedy decoding throughout all experiments (Xu et al., 2023). Through this, we ensure to reduce sampling variations of getting inconsistent generations. For 10-shot prompting, the in-context examples are chosen randomly.

Fine-tuning settings. We adopt QLoRA (Dettmers et al., 2023), quantized version of LoRA (Hu et al., 2022), for parameter-efficient fine-tuning. For LoRA configs, we set the LoRA rank to 16, scaling parameter to 32, and dropout probability for layers at 0.05. We fine-tune all of the available training parameters, which is approximately 0.16B (4.4%) of the total parameters. We use the Adam optimizer with an initial learning rate to 2e-4, batch size of 2, gradient accumulation over 4 steps, with a warmup phase of 20 steps. We train over 5 epochs, evaluating the model's performance on 200 MQM validation set instances at the end of each epoch. We implement early stopping to halt the fine-tuning process if there is no improvement in the model performance for 16 consecutive steps.

5 Prompting Results

Figure 2 shows the zero- and 10-shot prompting performance of LLaMA-2 7B across three language pairs. The complete results in table form for both LLaMA-2 7B and 13B can be found in Appendix E.

Zero-Shot. For all language pairs, we observe a *marginal improvement when post-editing with any form of feedback in zero-shot settings*, with small increases in BLEU COMET_{DA} scores, along with reduced TER. Although the score differences between the original and post-edited MT can be small, they are statistically significant ($p \leq 0.05$)

[5]As a sanity check, we prompted the LLaMA models to translate our WMT-22 test set. The resulting translation quality (Appendix E.4) suggests that WMT-22 was not included in pre-training, and is therefore a valid test set.

for all cases. One exception is Zh-En pair, for which BLEU drops by 0.01 to 0.02 points after integrating feedback other than MQM.

Few-Shot. The improvements from zero to 10-shot prompting are shown by hashed lines in Figure 2. The *performance gap between the original and post-edited MT widens with few-shot learning.* We examine a consistent gain in both BLEU and COMET_{DA} scores, which represent the overall MT quality. The average gain across language pairs is +0.04 BLEU (on a 0-1 scale) and +0.03 for COMET_{DA}. TER, which measures the remaining amount of edits to be made also shows -0.03 point improvement for Zh-En, -0.06 point for En-De, and -0.04 point for En-Ru.

7B vs 13B. The 13B model unsurprisingly achieve higher BLEU and COMET_{DA} and lower TER compared to the 7B model in zero-shot settings. However, this performance gap narrows down with the increase in number of few-shot examples. This trend suggests that *few-shot learning helps bridge the performance gap between model sizes* for MT post-editing. We report comprehensive results on LLaMA-2 13B in Appendix E.

Feedback Granularity. We categorize external feedback into three granularity levels: generic, score-based, and fine-grained error annotation. Fine-grained feedback is further divided into human-annotated (MQM) and automatically detected by metrics (xCOMET, InstructScore). We observe that differences in the automatic metrics across different types of feedback are small. Providing fine-grained feedback on errors has limited benefits over a generic feedback while score-based feedback shows to have the least improvement in the MT output. Overall, the performance difference between various granularity of feedback is more evident for zero-shot setting while increasing to 10-shot prompting paints a different picture.

For 10-shot prompting, most forms of our tested feedback, regardless of granularity, converge to a similar performance. However, while the two MT quality metrics, BLEU and COMET_{DA} remains similar for different forms of feedback, there is a clear difference for TER. When providing generic feedback or automatic annotations from xCOMET, TER decreases by approximately 0.15 points for Zh-En and 0.3 points for En-De and En-Ru compared to the original baseline. Score-based feedback remains to show the least increase in performance, but they also decrease 0.1 points for Zh-En and 0.2 points for En-De and En-Ru, which are statistically significant. Nevertheless, prompting does not reveal a marked advantage for using certain type of feedback for post-editing.

Language	Type	BLEU (↑)	TER (↓)	COMET (↑)
Zh-En	Original	0.47	0.75	0.70
	prompt (k=0)	0.48	0.72	0.70
	prompt (k=10)	0.51	0.65	0.72
	FT (Generic)	0.47	0.71	0.72
	FT (Bi)	**0.53**†	0.63†	**0.76**†
	FT (Multi)	**0.53**†	**0.61**†	**0.76**†
En-De	Original	0.45	0.86	0.69
	prompt (k=0)	0.51	0.68	0.75
	prompt (k=10)	0.58	0.56	0.79
	FT (Generic)	0.52	0.62	0.74
	FT (Bi)	0.56†	0.58†	0.79†
	FT (Multi)	**0.59**†	**0.55**†	0.79†
En-Ru	Original	0.43	0.82	0.73
	prompt (k=0)	0.48	0.69	0.74
	prompt (k=10)	0.49	0.67	0.79
	FT (Generic)	0.46	0.67	0.74
	FT (Bi)	0.51†	0.65†	**0.80**†
	FT (Multi)	**0.52**†	**0.63**†	**0.80**†

Table 4: Fine-tuning (FT) results for LLaMA-2 7B. **prompt (k=0)** and **prompt (k=10)** indicate the zero- and 10-shot prompting results of LLaMA-2 7B respectively. **FT (Generic)**: Fine-tuning with generic feedback; **FT (Bi)**: Fine-tuning with fine-grained feedback in bilingual setting, where models are individually fine-tuned for each language pair; **FT (Multi)**: Fine-tuning with fine-grained feedback in multilingual setting, combine 3 language pairs to fine-tune a single model. We test the statistically significance of improvements over the best prompting baseline and † marks results with p-value ≤ 0.05.

6 Fine-Tuning Results

6.1 Automatic Evaluation

We examine the effectiveness of fine-tuning error-annotated translations for MT post-editing. Table 4 shows that *fine-tuning with error annotated translations gives an extra boost in the performance across all metrics.*

Original vs Fine-tuning. We compare the fine-tuning results of each language pair against the original translation quality (indicated as 'Original' in Table 4). Across language pairs, metrics of MT quality all increase for fine-tuning. Translation quality increases steeply by approximately +0.07 BLEU, +0.08 COMET_{DA} and -0.21 TER on average for all language pairs. The multilingual approach mostly outperforms the bilingual one, suggesting that the advantages gleaned from fine-

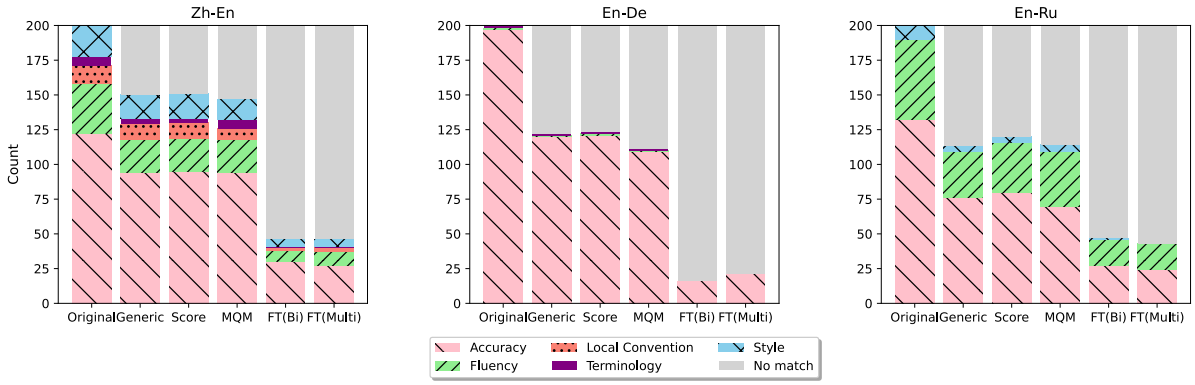

Figure 3: Error analysis for LLaMA-2 7B. We observe how much each error type is resolved by integrating external feedback during post-editing. We classify an error as '*No match*' if the output translation does not contain the specific error span. Across all language pairs, fine-tuning best addresses the errors present in the initial translation. **FT (Bi)**: fine-tuning in bilingual setting; **FT (Multi)**: fine-tuning in multilingual setting. We do not include InstructScore or xCOMET as InstructScore annotates more than 1 error spans, making it difficult for fair comparison and xCOMET does not output error type information.

tuning with diverse language pairs outweigh the benefits of matching the fine-tuning data language consistent to the test language pair. We observe the same trend with LLaMA-2 13B in Appendix Table 7: fine-tuning results improve upon the original baseline results by +0.1 BLEU, +0.08 COMET_{DA} and -0.25 TER points on average.

Prompting vs Fine-tuning. Next, we examine fine-tuning evaluation compared to the zero- and 10-shot prompting results, collected from either LLaMA-2 7B or 13B. Compared to zero-shot prompting, fine-tuning with error annotations always outperform across all metrics and the multilingual approach outperforms 10-shot prompting results for most of the cases.

Feedback granularity. We compare the two distinct types of feedback used for fine-tuning: generic and fine-grained feedback, denoted as 'FT (Generic)' and 'FT (Multi)' respectively in Table 4. While prompting experiments demonstrate no clear preference between levels of feedback granularity, fine-tuning using fine-grained feedback consistently yields superior translation quality compared to fine-tuning with generic feedback with a gap of 4 to 6 BLEU points, 3 to 8 TER, and 4 to 6 COMET. This shows that fine-tuning allows the models to take advantage of the fine-grained feedback more effectively.

As there are few error tokens overall, we first expected to see small edits from our fine-tuned model, thus small score difference. However, surprisingly, fine-tuning results overall show greater improvements, especially for TER, considering that the original MQM dataset only has one error span per sentence. Examining outputs (see Appendix E.5 for examples) suggests that fine-tuning not only edits the targeted error spans but also improve the overall naturalness in the target language, consistent with prior evidence that post-editing with LLMs reduces translationese effects (Chen et al., 2023). To further validate this hypothesis, we turn to human evaluation.

6.2 Human Evaluation

We ask bilingual human annotators to assess the post-edited outputs obtained by fine-tuning in the bilingual setting as it is the stronger approach based on automatic scores. We randomly select 50 instances for each language pair for annotation. Each instance is examined by 3 human annotators. For each instance of source text, original MT with MQM annotation, post-edited MT, the annotator is asked to rate on a 5-point Likert scale (1 strongly disagree to 5 strongly agree) whether the translation quality has improved, and to what extent the annotated errors are actually resolved through post-editing. Ordinal Kripendorff's alpha (Krippendorff, 2011)[6], which measure the inter-annotator agreement is moderate for the *Overall quality*: 0.527, 0.479, 0.421 for Zh-En, En-De, and En-Ru. Annotators are also given the option to provide free form comments. Refer to Appendix F for further details on the annotation set-up.

[6]Kripendorff's alpha ranges from 0 to 1, where 0 means no agreement and 1 means perfect agreement.

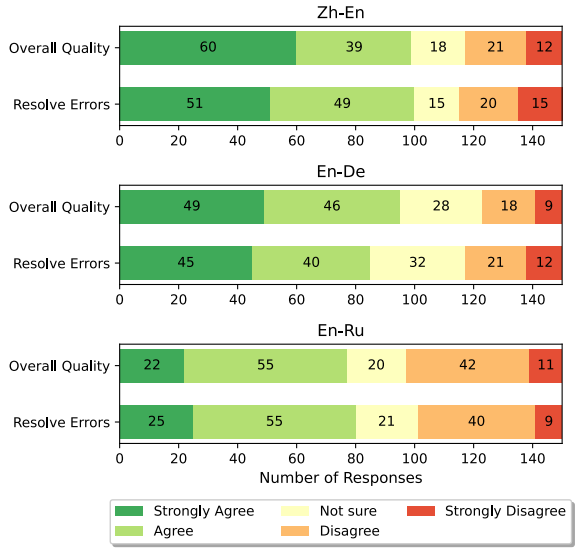

Figure 4: Human evaluation results for 3 language pairs. We collect a total of 150 annotations for each language pair. *Overall Quality*: Output translation from the fine-tuned model is better than the initial translation; *Resolve Errors*: Output translation resolves errors in the initial translation.

As illustrated in Figure 4, our human evaluation results confirm that fine-tuning with error annotations enhances overall translation quality (*Overall Quality*) and effectively resolves errors in the initial translation (*Resolve Errors*). While this improvement is notably evident in Zh-En and En-De pair, for the En-Ru pair, approximately 40/150 annotations lean towards the *Disagree* category. Some of the feedback from En-Ru annotators who choose to *Disagree* state that there are cases when the output translation from the fine-tuned model is more precise in the target language, but loses some of the nuance in the source text.

Further, feedback from the annotators support our own observation that the post-editing via fine-tuning does not only fix targeted errors in the original translation but rewrites for naturalness in the target language. They comment that the fine-tuning translation "*better explains the context*" and "*flows better in the target language*" compared to the original translation which seems to be directly translated without consideration of the context. We list further comments in Appendix Table 20.

7 Analysis by MT Error Categories

Our error analysis aims to pinpoint the types of errors that are most effectively resolved through the integration of external feedback. We evaluate 200 output translations generated by prompting LLaMA-2 7B with each generic, score-based, and MQM feedback. We do not include InstructScore or xCOMET as InstructScore annotates more than 1 error spans making it difficult for fair comparison and xCOMET does not output error type information. We also compare the outputs from our custom fine-tuned models, both bilingual and multilingual version. All of the feedback is based on MQM, thus we categorize the error type as per "Error Category" from MQM detailed in Appendix Table 8.

In Figure 3, we illustrate the extent to which each error type has been resolved by incorporating external feedback. First, we check whether a span annotated as an error in the original translation matches the output after post-editing with feedback. A match increments the count for the error type associated with the span. If there is no match found, the count for the "No match" category is incremented. We observe that using any form of feedback (generic, score, or MQM) increases the portion of "No match" category compared to the original translation. However, there is no distinct trend for any specific error type; all of the errors are addressed in a balanced manner.

Further, by incorporating the output translations from our fine-tuned model, we see a sudden leap in the "No match" category. This suggests that fine-tuning best fixes the targeted error span. This finding is also consistent with the conclusions from Section 6, where we noted that fine-tuning help align LLM behavior with the provided feedback.

8 Post-Editing Correct Outputs

The experiments we have presented so far are focused on post-editing MT hypotheses that are known to leave room for improvement. For completeness, we present in Appendix Table 14 decoding results when zero-shot prompting the LLaMA-2 models to post-edit approaches to 200 WMT hypotheses labeled as "No error" by the WMT human annotators.

As expected, the resulting edits lead to a small drop in automatic metrics, confirming the observation that the nature of edits goes beyond correcting errors to address more stylistic issues such as translationese. Interestingly, the larger LLaMA-2 model and the fine-grained feedback are the least prone to over-editing. We anticipate that different prompts and fine-tuning data are needed for models to jointly consider the task of editing or not, and of

what edits to perform.

9 Conclusion

We explore a range of strategies to guide LLaMA-2 models to improve MT outputs using external feedback, varying in different granularity. We demonstrate that prompting LLM to edit MT with feedback reliably enhances the overall translation quality and post-editing efforts. We further explore instruction fine-tuning LLMs with fine-grained feedback. Through automatic and human evaluation, we demonstrate that fine-tuning shows stronger improvements on enhancing the translation quality, resolving errors in the initial translation, and most notably, generating translations that are more natural (less translationese) in the target language.

Taken together, these results clearly show that post-editing MT output does not require the largest proprietary LLM models and can be done with smaller open-source models. This opens many questions for future work to further explore how to do this well in more diverse settings, while minimizing the reliance on human annotated MT outputs which are expensive to obtain at scale. Building on LLMs fine-tuned for many translation related tasks (Alves et al., 2024) is a promising direction for encouraging transfer learning from limited amounts of annotation.

10 Limitations

We evaluate the impact of post-editing separately on MT outputs that contain one or more errors (§5) and on MT outputs that do not contain any errors (§11). This leaves open the question of how to design a workflow that takes in any MT input and automatically determines whether and how it should be post-edited, possibly selecting among different potential feedback mechanisms, which we leave to future work.

Furthermore, the fine-tuning data is in the same domain as the test data which will not always be the case in practice. While we test on diverse languages and on out-of-English and into-English directions, it remains to be seen how our findings generalize to a wider variety of languages, particularly in low-resource settings.

Finally, our work highlights the effectiveness of using external feedback to resolve errors in translations. Although integrating external feedback is an attractive approach, the scarcity of high-quality feedback remains a significant challenge. This scarcity underscores the demand for the development of automated systems capable of generating high-quality error annotations. In regard to constraints in the currently available external feedback for MT post-editing, our study is constrained in terms of forms of feedback (generic, score, fine-grained) and language pairs. Future works can focus on incorporating automatic systems that can generate consistent, high quality feedback.

11 Acknowledgement

We thank the anonymous reviewers, Shramay Palta, Nishant Balepur, Calvin Bao, Yu Hou and the members of the CLIP lab at University of Maryland for their valuable suggestions and constructive feedback.

This work was supported in part by NSF Fairness in AI Grant 2147292, by the Institute for Trustworthy AI in Law and Society (TRAILS), which is supported by the National Science Foundation under Award No. 2229885, and by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the HIATUS Program contract #2022-22072200006, by NSF grant 2147292. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, NSF or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak

Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. Palm 2 technical report.

Rachel Bawden and François Yvon. 2023. Investigating the Translation Performance of a Large Multilingual Language Model: The Case of BLOOM.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Rajen Chatterjee, Matteo Negri, Raphael Rubino, and Marco Turchi. 2018. Findings of the WMT 2018 Shared Task on Automatic Post-Editing. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 710–725, Belgium, Brussels. Association for Computational Linguistics.

Pinzhen Chen, Zhicheng Guo, Barry Haddow, and Kenneth Heafield. 2023. Iterative translation refinement with large language models.

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.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

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

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. Gptscore: Evaluate as you desire.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2013. Continuous measurement scales in human evaluation of machine translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 33–41, Sofia, Bulgaria. Association for Computational Linguistics.

Jiatao Gu, Changhan Wang, and Jake Zhao Junbo. 2019. Levenshtein transformer. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 1003, pages 11181–11191. Curran Associates Inc., Red Hook, NY, USA.

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

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Wenxiang Jiao, Wenxuan Wang, Jen-tse Huang, Xing Wang, and Zhaopeng Tu. 2023. Is ChatGPT A Good Translator? Yes With GPT-4 As The Engine.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Log-linear Combinations of Monolingual and Bilingual Neural Machine Translation Models for Automatic Post-Editing. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 751–758, Berlin, Germany. Association for Computational Linguistics.

Marzena Karpinska, Nishant Raj, Katherine Thai, Yixiao Song, Ankita Gupta, and Mohit Iyyer. 2022. DEMETR: Diagnosing evaluation metrics for translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9540–9561, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kevin Knight and Ishwar Chander. 1994. Automated postediting of documents. In *Proceedings of the Twelfth AAAI National Conference on Artificial Intelligence*, AAAI'94, pages 779–784, Seattle, Washington. AAAI Press.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Makoto Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović, and Mariya Shmatova. 2023. Findings of the 2023 Conference on Machine Translation (WMT23): LLMs Are Here but Not Quite There Yet. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1–42, Singapore. Association for Computational Linguistics.

Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal Novák, Martin Popel, and Maja Popović. 2022. Findings of the 2022 conference on machine translation (WMT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1–45, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

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.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large Language Models are Zero-Shot Reasoners.

Klaus Krippendorff. 2011. Computing krippendorff's alpha-reliability.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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 *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. 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. *Revista tradumàtica: traducció i tecnologies de la informació i la comunicació*, (12):455–463.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2024. REFINER: Reasoning feedback on

intermediate representations. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1100–1126, St. Julian's, Malta. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Vipul Raheja, Dhruv Kumar, Ryan Koo, and Dongyeop Kang. 2023. CoEdIT: Text Editing by Task-Specific Instruction Tuning.

Vikas Raunak, Amr Sharaf, Yiren Wang, Hany Awadalla, and Arul Menezes. 2023. Leveraging GPT-4 for automatic translation post-editing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12009–12024, Singapore. Association for Computational Linguistics.

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

Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. 2022. A Recipe for Arbitrary Text Style Transfer with Large Language Models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 837–848, Dublin, Ireland. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Michel Simard, Cyril Goutte, and Pierre Isabelle. 2007. Statistical Phrase-Based Post-Editing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 508–515, Rochester, New York. Association for Computational Linguistics.

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

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. Prompting PaLM for Translation: Assessing Strategies and Performance.

David Wan, Chris Kedzie, Faisal Ladhak, Marine Carpuat, and Kathleen McKeown. 2020. Incorporating Terminology Constraints in Automatic Post-Editing. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1193–1204, Online. Association for Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Sean Welleck, Ximing Lu, Peter West, Faeze Brahman, Tianxiao Shen, Daniel Khashabi, and Yejin Choi. 2023. Generating sequences by learning to self-correct. In *The Eleventh International Conference on Learning Representations*.

Wenda Xu, Daniel Deutsch, Mara Finkelstein, Juraj Juraska, Biao Zhang, Zhongtao Liu, William Yang Wang, Lei Li, and Markus Freitag. 2024. Llmrefine: Pinpointing and refining large language models via fine-grained actionable feedback.

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 *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5967–5994, Singapore. Association for Computational Linguistics.

Jiali Zeng, Fandong Meng, Yongjing Yin, and Jie Zhou. 2023. Improving machine translation with large language models: A preliminary study with cooperative decoding.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis.

A Fine-grained Feedback format

In this section, we discuss the details on the format of fine-grained feedback both human-annotated and automatically annotated by InstructScore or xCOMET. We refer to fine-grained feedback in three components: error span position, error type, and error severity level. MQM annotations and InstructScore use the same MQM hierarchy to define error type as shown in Table 8, with InstructScore omitting categories such as "Source error", "Non-translation", and "Other". Unlike these, xCOMET does not provide error type information in their annotation.

The levels of error severity are summarized in Table 5. In our prompting experiments, we eliminate instances annotated as "No-error" in the MQM dataset, as our focus is on understanding the role of external feedback in post-editing **erroneous** translations. However, for fine-tuning, we include all instances, regardless of their error severity level.

Metric	Severity level
MQM	Major, Minor, No-error
InstructScore	Major, Minor
xCOMET	Critical, Major, Minor

Table 5: Error severity levels supported by each metric.

Severity	Category	Weight
Major	Non-translation	25
	All others	5
Minor	Fluency/Punctuation	0.1
	All others	1
Neutral	All	0

Table 6: MQM error weighting. Each score can range from 0 (perfect) to 25 (worst). The final score is the average over scores from all annotators.

B Error Annotation Examples

In Table 9, we present error annotation examples from three sources: MQM, xCOMET, and InstructScore. We obtain automatic annotations of the same evaluation dataset using InstructScore and xCOMET. The consistency of these error annotations across different tools is further discussed in Section 4.1.

C Dataset Details

C.1 MQM Dataset

We analyze 1,000 MQM data instances used for evaluation. We note that the average number of error spans per sentence is 1 as from the original MQM dataset. The average error span length is 13.5 for Zh-En, 11.3 for En-De, and 9.3 for En-Ru. Further, we observe the error type and severity level distribution in Figure 5 and 6. Across all language pairs, "*Accuracy*" errors are the majority (524 for Zh-En, 362 for En-De, 592 for En-Ru), followed by "*Fluency*" (274 for Zh-En, 324 for En-De, 257 for En-Ru). For the severity level, Zh-En has the most "*major*" errors (512/1000), then En-Ru (388/1000) and En-De (202/1000).

D Fine-tuning Details

D.1 Fine-tuning Dataset format

We illustrate the instruction template used for constructing fine-tuning dataset in Table 10. We explicitly include the fine-grained errors in the instruction to guide LLMs on how to leverage them as hints and align in their improved translation outputs. We employ the same instruction format during inference time.

D.2 LLaMA-2 13B Results

We also extend the fine-tuning of LLaMA-2 13B for two settings, mirroring the 7B setup: bilingual and multilingual. We follow the identical experimental setup as in the 7B experiment. We show that similar trend is observed; fine-tuning with error annotations show better performance than the original baseline and the zero- and 10-shot prompting results across all metrics.

E Detailed Results

E.1 LLaMA-2 Original

We show the detailed numerical results for the LLaMA-2 7B and 13B experiments in Table 11 and 12 respectively. In zero-shot prompting, the 13B model outperforms 7B, which can be attributable by larger LLMs having stronger instruction-following capabilities than their smaller counterparts (Wei et al., 2022). Further, 13B shows similar trend observed for 7B, where the use of external feedback in 10-shot prompting significantly improves performance.

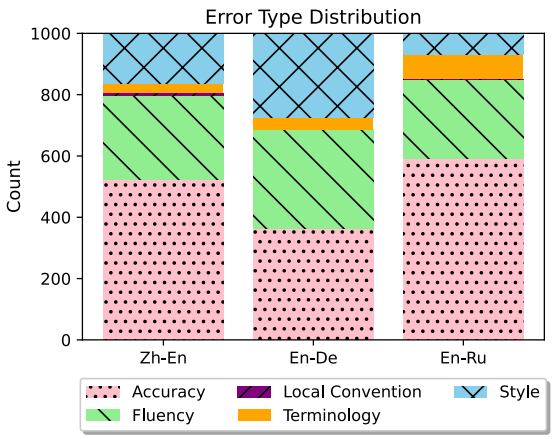

Figure 5: Error type distribution for 3 language pairs. Note that En-Ru dataset from WMT 22 General MT submissions use different names for each error type, thus conduct manual mapping.

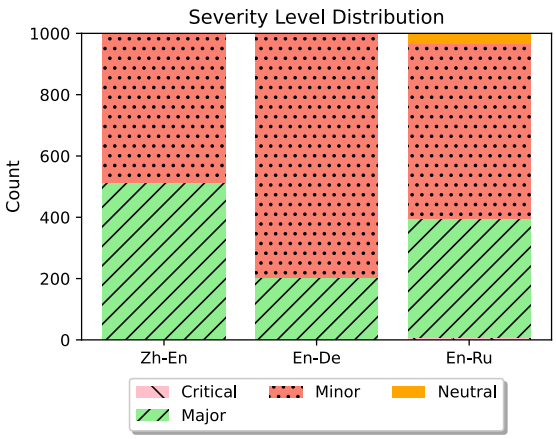

Figure 6: Severity level distribution for 3 language pairs. We note that the En-Ru dataset from WMT 22 General MT submissions use additional severity level category: "*critical*".

Language	Type	BLEU (↑)	TER (↓)	COMET_{DA} (↑)
Zh-En	Original	0.47	0.75	0.70
	prompt (k=0)	0.50	0.74	0.71
	prompt (k=10)	0.50	0.61	0.73
	FT (Bi)	0.51†	0.61	**0.77**†
	FT (Multi)	**0.54**†	**0.58**†	0.76†
En-De	Original	0.45	0.86	0.69
	prompt (k=0)	0.51	0.68	0.75
	prompt (k=10)	0.58	0.56	0.79
	FT (Bi)	0.57	0.55†	**0.80**†
	FT (Multi)	**0.60**†	**0.53**†	**0.80**†
En-Ru	Original	0.43	0.82	0.73
	prompt (k=0)	0.44	0.80	0.73
	prompt (k=10)	0.53	0.56	0.80
	FT (Bi)	0.53	0.57	0.80
	FT (Multi)	**0.54**†	0.56†	**0.81**†

Table 7: Fine-tuning (FT) results for LLaMA-2 13B. **prompt (k=0)** and **prompt (k=10)** indicate the zero- and 10-shot prompting results of LLaMA-2 13B respectively. **FT (Bi)**: Fine-tuning in bilingual setting, where models are individually fine-tuned for each language pair; **FT (Multi)**: Fine-tuning in multilingual setting, combine 3 language pairs to fine-tune a single model. We test the statistically significance of improvements over the best prompting baseline and † marks results with p-value ≤ 0.05.

E.2 LLaMA-2 Chat

We expand our experiments with LLaMA-2-chat, an instruction fine-tuned version of LLaMA-2. Although they are optimized to better follow the instructions that users specify, we show that LLaMA-2 models consistently outperform the chat counterparts in Table 13. Our findings indicate that instruction-following ability of LLMs might not be the only determining factor for successful MT post-editing.

E.3 Fine-grained Components

In Table 15, we observe the impact of each component of the fine-grained feedback: error span position, error type, and severity level with 200 randomly sampled test cases. We examine that while the individual contribution of each error component is trivial, interestingly, providing only the severity level information consistently yields similar or superior results compared to providing all three components simultaneously. This shows that there could be other forms of feedback effective when prompted to LLMs, which we leave for future work.

E.4 Translate from Scratch

We present zero-shot LLaMA-2 translation results in Table 16. We report the scores for 1,000 WMT test instances used in our main evaluation, as translated with LLaMA-2 7B and 13B models with the prompt template as: *"Translate from {source language} to {target language} without any explanation.\n{source language}: {source sentence}\n{target language}:"*. Results show that LLaMA-2 7B is not powerful at translating compared to the baseline hypothesis translations provided by the MQM. LLaMA-2 13B shows comparable results to the original performance except for En-De where it slightly outperforms the original. We would expect much higher scores if the test set had been memorized as part of the LLaMA-2 pre-training data. Further, we notice that translating from scratch with LLaMA-2 7B consistently shows lower performance than post-editing regard-

less of feedback types. For 13B, again translating shows lower performance compared to post-editing with generic or xCOMET feedback but similar to score-based or MQM feedback.

E.5 Qualitative Analysis

In this section, we demonstrate how fine-tuning enhances the alignment of LLM behavior with the external feedback. Tables 17, 18, and 19 illustrate output translations generated by LLaMA-2 7B incorporating different types of feedback. While relatively coarse feedback (generic and score-based) are not able to accurately pinpoint and correct the targeted error spans, fine-grained feedback (MQM, InstructScore, and xCOMET) resolves this issue. Further, even in instances where fine-grained feedback falls short, fine-tuning enables the model to generate translations that more effectively narrow the gap. We also demonstrate that translations from the fine-tuned model not only resolves the errors but also makes it more natural (less translationese) in the target language.

F Human Evaluation

F.1 Evaluation Details

For human evaluation, we employ Qualtrics[7] to design our survey and Prolific[8] to recruit human annotators. We randomly sample 50 instances for each language pair and further divide them into two separate sessions. Each session consists of 25 examples and is estimated to take approximately 30 minutes to complete. For every session and language pair, we engage 3 annotators who are fluent in both source and target languages. For example, when evaluating the Chinese-English pair, we choose annotators fluent in both Chinese and English. Consequently, for each language pair, we recruit a total of 6 annotators, amounting to 18 annotators overall. We offer a compensation of $7 per session, totaling $126 for the entire human evaluation process.

F.2 Annotator Instructions

In Figure 7 and 8, we present the instructions and survey content provided to our annotators. Each annotator reviews 25 set of examples, each consisting of the source text, Translation 1 (the initial translation), and Translation 2 (the output translation from our fine-tuned model). They are tasked with comparing these two translation on a Likert scale ranging from 0 (Strongly disagree) to 5 (Strongly agree). This comparison is based on two criteria: (1) "*Translation 2 is better than Translation 1*" evaluates whether the output translation more effectively conveys the meaning of the source text and exhibits improved fluency in the target language; (2) "*Translation 2 fixes errors that were present in Translation 1*" examines whether the output translation resolves errors present in the original translation. Additionally, we provide a free-form text box alongside each example for any additional feedback or suggestions.

F.3 Feedback from Annotators

In our survey, we provide a text box for each example to collect additional feedback or suggestions from the annotators. We present the feedback per language pair in Table 20. Main reasons for preferring the initial translation over the output translation from our fine-tuned model are that although the output translation may be grammatically and syntactically more precise, the initial translation often better preserves the meaning of the source sentence. Additionally, some annotators note that the initial version is more specific and understandable in some cases.

In contrast, annotators comment that they prefer the output translation from the fine-tuned model over the initial translation because (1) It fixes all the errors that were present in the initial translation; (2) It explains in the context of the target language; (3) It fits well to the actual use of the target language and flows better. All of the comments indicate that fine-tuning error annotations help make the translation more natural.

[7] https://www.qualtrics.com/
[8] https://www.prolific.com/

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

Table 8: MQM hierarchy (Freitag et al., 2021).

Source	现如今绝大多数遇难者的老父母均已谢世，遗孤们也已长大成家就业。
Candidate	Nowadays, the vast majority of the victims' elderly parents have died, and the orphans have grown into family employment.
Reference	Now the old parents of the vast majority of the victims have passed away, and the orphans have also grown up, started working and got married.
MQM	Error span: Nowadays Error type: Accuracy/Mistranslation Severity: Major
InstructScore	Error span: [family employment, Nowadays] Error type: [Incorrect translation is missing content from the correct translation, Incorrect translation has stylistic problems] Severity: [Major, Major]
xCOMET	Error span: [die, have grown into family employment] Severity: [Major, Major]

Table 9: Chinese-English error annotation examples from three sources: MQM dataset, xCOMET, and InstructScore. We input source, candidate translation, and reference sentence to xCOMET and InstructScore. xCOMET does not output error type in their annotation. Both xCOMET and InstructScore returns a list if they detect more than one errors in the candidate translation.

Instruction
English: Memorial meetings were organised at the residence of Sam Stafford, one of the agitators who died, and a playground in Guwahati, with attendees resolving to once again to intensify the stir against the Citizenship (Amendment) Act.\n ### **German**: Gedenkmälerversammlungen wurden in der Residenz von Sam Stafford, einem der gestorbenen Agitatoren, und einem Spielplatz in Guwahati organisiert, wobei die Teilnehmer sich entschlossen hatten, den Aufruhr gegen das Gesetz über die Staatsbürgerschaft (Änderung) erneut zu intensivieren.\n ### Errors: There is a minor accuracy/mistranslation error at "Gedenkmälerversammlungen".\n\n ### Improved **German**: Gedenkveranstaltungen fanden am Wohnsitz von Sam Stafford, einem der getöteten Aktivisten, sowie auf einem Schulhof in Guwahati statt, und die Teilnehmer beschlossen, noch einmal den Protest gegen den CAA zu verstärken.

Table 10: Example of fine-tuning instructions dataset reformulated from English-German error annotated translations. Texts in **bold** represent the placeholders for source and target language. We give fine-grained feedback after ### Errors: .

Language	Shots	BLEU (↑) / TER (↓) / COMET_{DA} (↑)					
		Original	Generic	Score	MQM	InstructScore	xCOMET
Zh-En	0	0.47 / 0.75 / 0.70	0.45 † / 0.63† / 0.71†	0.46 † / 0.69† / 0.7†	0.48† / 0.72† / 0.70†	0.45 † / 0.63† / 0.72†	0.45 † / 0.62† / 0.72†
	10		0.51 † / 0.63† / 0.73†	0.51 † / 0.64† / 0.73†	0.51 † / 0.65† / 0.72†	0.50† / 0.61† / 0.74†	0.51 † / 0.60 † / 0.75 †
En-De	0	0.45 / 0.86 / 0.69	0.55† / 0.57† / 0.78†	0.51† / 0.68† / 0.75†	0.51† / 0.68† / 0.75†	-	0.57 † / 0.54 † / 0.80†
	10		0.58† / 0.55† / 0.79†	0.55† / 0.59† / 0.80†	0.56† / 0.58† / 0.79†	-	0.53† / 0.57† / 0.81 †
En-Ru	0	0.43 / 0.82 / 0.73	0.51† / 0.60† / 0.77†	0.48† / 0.70† / 0.74†	0.48† / 0.69† / 0.74†	-	0.51† / 0.58† / 0.80†
	10		0.53† / 0.56† / 0.79†	0.53 † / 0.60† / 0.79†	0.49† / 0.70‡ / 0.79†	-	0.53 † / 0.56 † / 0.82 †

Table 11: Zero- and 10-shot prompting performance of LLaMA-2 7B model. Original column measures between the source and target sentences from the original MQM dataset. Other columns represents the model performance for different types of feedback: Generic, Score, Fine-grained (MQM, xCOMET, InstructScore). Green: best performance per language pair; Red: worse performance than the original baseline. We test the statistically significance of improvements over the original and † marks results with p-value ≤ 0.05 and ‡ marks results with p-value ≤ 0.1.

Language	Shots	BLEU (↑) / TER (↓) / COMET_{DA} (↑)					
		Original	Generic	Score	MQM	InstructScore	xCOMET
Zh-En	0	0.47 / 0.75 / 0.70	0.50† / 0.66† / 0.73†	0.50† / 0.72† / 0.72	0.50† / 0.74† / 0.71†	0.50† / 0.61† / 0.75†	0.53† / 0.59† / 0.76 †
	10		0.51† / 0.61† / 0.74†	0.51† / 0.61† / 0.74†	0.50† / 0.61† / 0.73†	0.50† / 0.61† / 0.75†	0.54 † / 0.58 † / 0.76 †
En-De	0	0.45 / 0.86 / 0.69	0.58† / 0.55† / 0.80†	0.49† / 0.73† / 0.73†	0.48† / 0.76† / 0.72	-	0.60† / 0.52† / 0.81†
	10		0.58† / 0.54† / 0.80†	0.58† / 0.55† / 0.80†	0.58† / 0.54† / 0.80†	-	0.62 † / 0.51 † / 0.82 †
En-Ru	0	0.43 / 0.82 / 0.73	0.53† / 0.56† / 0.79†	0.46† / 0.74† / 0.74†	0.44† / 0.80 / 0.73	-	0.55† / 0.54† / 0.83†
	10		0.54† / 0.55† / 0.80†	0.54† / 0.56† / 0.80†	0.53† / 0.56† / 0.80†	-	0.57 † / 0.52 † / 0.85 †

Table 12: Zero- and 10-shot prompting performance of LLaMA-2 13B model. Green: best performance per language pair; Red: worse performance than the original baseline. We test the statistically significance of improvements over the original and † marks results with p-value ≤ 0.05.

Language	Shots	BLEU (↑) / TER (↓) / COMET_{DA} (↑)					
		Original	Generic	Score	MQM	InstructScore	xCOMET
LLaMA-2 chat 7B							
Zh-En	0	0.47 / 0.75 / 0.70	0.43† / 0.69† / 0.73†	0.45† / 0.66† / 0.74†	0.40† / 0.70† / 0.70	0.42† / 0.68† / 0.73†	0.41† / 0.67† / 0.73†
	10		0.48 / 0.65† / 0.74†	0.48 / 0.64† / 0.74†	0.48 / 0.64† / 0.73†	0.48† / **0.63** / 0.75†	**0.50** / **0.63**† / **0.76**†
En-De	0	0.45 / 0.86 / 0.69	0.42 / 0.68† / 0.73‡	0.51† / 0.63† / 0.76†	0.54† / 0.62† / 0.76†	-	0.50† / 0.6† / 0.77†
	10		0.54† / 0.60† / 0.78†	0.54† / 0.60† / 0.78†	0.54† / 0.59† / 0.78†	-	**0.56**† / **0.57**† / **0.79**†
En-Ru	0	0.43 / 0.82 / 0.73	0.40† / 0.72† / 0.72‡	0.45† / 0.66† / 0.73	0.48† / 0.64† / 0.74‡	-	0.45† / 0.63† / 0.76†
	10		0.50† / 0.61† / 0.78†	0.49† / 0.62† / 0.77†	0.50† / 0.61† / 0.77†	-	**0.52**† / **0.59**† / **0.81**†
LLaMA-2 chat 13B							
Zh-En	0	0.47 / 0.75 / 0.70	0.40† / 0.71† / 0.73†	0.45† / 0.69† / 0.72†	0.47† / 0.72† / 0.70	0.44 / 0.65† / 0.75†	0.47 / 0.64† / 0.76†
	10		0.48 / 0.65 / 0.74†	0.48‡ / 0.64 / 0.74†	0.49† / 0.62 / 0.74†	0.48† / 0.64 / 0.75†	**0.51** / **0.62** / **0.76**†
En-De	0	0.45 / 0.86 / 0.69	0.53† / 0.59† / 0.78†	0.50† / 0.71† / 0.74†	0.48† / 0.74† / 0.72	-	**0.57**† / **0.54**† / **0.80**†
	10		0.53† / 0.60† / 0.78†	0.55† / 0.58† / 0.70†	0.55† / 0.58† / 0.79†	-	0.55† / 0.58† / **0.80**†
En-Ru	0	0.43 / 0.82 / 0.73	0.47† / 0.61† / 0.77†	0.47† / 0.72† / 0.74†	0.47† / 0.72† / 0.74†	-	**0.51**† / **0.58**† / **0.81**†
	10		0.47† / 0.62† / 0.77†	0.49† / 0.60† / 0.79†	0.49† / 0.60† / 0.78†	-	0.49† / 0.62† / **0.81**†

Table 13: *Top rows*: Prompting performance of LLaMA-2 chat 7B model. *Bottom rows*: LLaMA-2 chat 13B model. **Bold** denotes best performance for each language pair in 7B and 13B. We test the statistically significance of improvements over the original and † marks results with p-value ≤ 0.05 and ‡ marks results with p-value ≤ 0.1.

Language	Size	BLEU (↑) / TER (↓) / COMET_{DA} (↑)			
		Original	Generic	Score	Fine-grained
Zh-En	7B	0.66 / 0.53 / 0.85	0.61 / 0.56 / 0.82	0.62 / 0.55 / 0.82	0.61 / 0.56 / 0.82
	13B		0.62 / 0.56 / 0.82	0.62 / 0.56 / 0.82	0.62 / 0.56 / 0.82
En-De	7B	0.65 / 0.56 / 0.88	0.57 / 0.61 / 0.84	0.52 / 0.65 / 0.81	0.58 / 0.61 / 0.84
	13B		0.64 / 0.56 / 0.88	0.65 / 0.56 / 0.87	0.64 / 0.56 / 0.87
En-Ru	7B	0.62 / 0.58 / 0.92	0.51 / 0.68 / 0.85	0.51 / 0.66 / 0.84	0.56 / 0.64 / 0.87
	13B		0.61 / 0.60 / 0.91	0.62 / 0.58 / 0.91	0.61 / 0.59 / 0.91

Table 14: Zero-shot prompting performance for instances with no error in their hypothesis translations. **Original MT hypothesis**: Translation quality from original MQM dataset. Resulting edits lead to small drop in the metrics but they correct stylistic issues such as translationese.

Language	Component	BLEU (↑)			TER (↓)			COMET_{DA} (↑)		
		MQM	InstructScore	xCOMET	MQM	InstructScore	xCOMET	MQM	InstructScore	xCOMET
Zh-En	All	0.47	0.43	0.41	0.72	0.66	**0.64**	0.70	0.73	**0.72**
	Span	0.47	0.41	0.41	0.71	0.67	0.66	0.71	0.72	0.72
	Type	0.47	0.43	-	0.70	**0.62**	-	0.72	**0.74**	-
	Severity	**0.48**	**0.44**	**0.44**	**0.66**	0.65	**0.64**	0.70	**0.74**	**0.74**
En-De	All	0.47	-	0.54	0.75	-	0.60	0.71	-	0.75
	Span	0.49	-	0.56	**0.71**	-	0.58	**0.72**	-	0.75
	Type	0.49	-	-	0.73	-	-	0.71	-	-
	Severity	**0.50**	-	**0.56**	**0.71**	-	**0.57**	0.71	-	**0.76**
En-Ru	All	0.43	-	0.48	0.77	-	0.62	0.74	-	0.76
	Span	**0.45**	-	0.48	0.75	-	0.62	0.73	-	0.77
	Type	0.44	-	-	0.76	-	-	**0.75**	-	-
	Severity	**0.45**	-	**0.50**	0.76	-	**0.61**	**0.76**	-	**0.78**

Table 15: Zero-shot prompting performance of LLaMA-2 7B when breaking down fine-grained feedback into three components. Note that results are missing as InstructScore only supports Zh-En MT pair and xCOMET does not output error type. While the individual contribution of each component is trivial, providing solely the severity level information outperforms the case of giving all components simultaneously. **Bold** indicates the best performance for each annotation.

Language	Type	BLEU (↑)	TER (↓)	COMET_{DA} (↑)
Zh-En	Original MQM translation	0.47	0.75	0.70
	LLaMA-2 (7B)	0.24	1.54	0.71
	LLaMA-2 (13B)	0.47	0.72	0.73
En-De	Original MQM translation	0.45	0.86	0.69
	LLaMA-2 (7B)	0.32	1.15	0.70
	LLaMA-2 (13B)	0.50	0.68	0.73
En-Ru	Original MQM translation	0.43	0.82	0.73
	LLaMA-2 (7B)	0.23	1.25	0.66
	LLaMA-2 (13B)	0.44	0.73	0.75

Table 16: Zero-shot translation from scratch performance of LLaMA-2 7B and 13B. Results do not support the hypothesis of train/test data contamination.

Type	Example
Source	可以先豁免，到你们调查完再收取我的月费，这样行吗?
Translation	You can first exempt, and then charge my monthly fee until you investigate, okay?
Reference	It can be exempted, and you can charge me with the monthly fee after the investigation, is that Okay?
Generic	Can you first exempt me, and then charge my monthly fee until you investigate, okay?
Score	You can first exempt, and then charge my monthly fee until you investigate, okay?
MQM	You can first waive the fee, and then charge my monthly fee after you investigate, okay?
InstructScore	You can first exempt, and then charge my monthly fee until you investigate, okay?
xCOMET	Can you first exempt, and then charge my monthly fee after you investigate, okay?
FT	It can be waived, and you can charge me the monthly fee after you finish the investigation, can that work?

Table 17: Example of error in Chinese-English translation pair. Underlined part represents the error span in the candidate translation. Each row from Generic to FT (fine-tuning) presents the generations by incorporating each type of feedback. For prompting results, we use 10-shot prompting of LLaMA-2 7B.

Type	Example
Source	Robust Character Creation Tools - Make the character YOU want to play.
Translation	Robust Character Creation Tools - Machen Sie den Charakter, den Sie spielen möchten.
Reference	Robuste Charaktererstellungs-Tools - baue dir die Charaktere, die DU spielen möchtest.
Generic	Robuste Charaktererstellungswerkzeuge - Machen Sie den Charakter, den Sie spielen möchten.
Score	Robuste Charakter-Erstellungswerkzeuge - Erstellt den Charakter, den Sie spielen möchten.
MQM	Robuste Charakter-Erstellungswerkzeuge - Machen Sie den Charakter, den Sie spielen möchten.
xCOMET	Robuste Charaktererstellung - Machen Sie den Charakter, den Sie spielen möchten.
FT	Robuste Charaktererstellungstools - Machen Sie den Charakter, den Sie spielen möchten.

Table 18: Example of major accuracy (mistranslation) error in English-German translation pair. Underlined part represents the error span in the candidate translation.

Type	Example
Source	Have you check on your doorstep or reception area?
Translation	Вы проверяли порог своей двери или приемную?
Reference	Вы проверили свой порог или стойку регистрации?
Generic	Вы проверили, если вы находитесь на пороге или в регистрационной лавке?
Score	Вы проверили свой порог или стойку регистрации?
MQM	Вы проверили ваш порог или регистрационную стойку?
xCOMET	Вы проверили ли свой порог или стойку регистрации?
FT	Проверьте на своем входе или зоне приемных залов?

Table 19: Example of error in English-Russian translation pair. Underlined part represents the error span in the candidate translation.

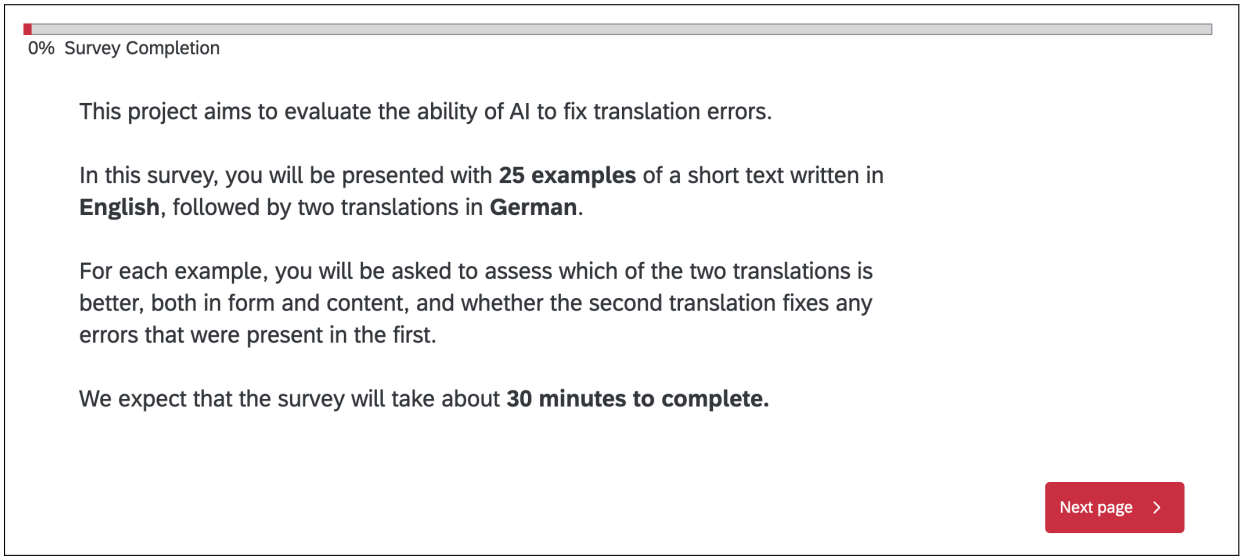

Figure 7: Instructions for human evaluation. This is shown as the first page of our survey to all annotators.

Language	Feedback
Zh-En	(1) Translation 1 is better because it explains more, but Translation 2 corrects the errors that 1 has. (2) Translation 1 is better than Translation 2, is more specific and understandable. (3) Translation 2 is better because it is easy to understand and **explains the context**. (4) Translation 2 **fixes all errors** in Translation 1. (5) Translation 2 is better because there are **no errors** and it is more concrete. (6) Translation 2 is better, but it can improve more.
En-De	(1) Although Translation 1 is more faithful to the original source sentence, it looks like it was directly translated from it. (2) Translation 2 is **more fitting to the actual use of the German language syntax and flow.**
En-Ru	(1) Translation 2 is better because it **avoids the major error** of Translation 1. (2) Translation 2 is more **accurate** and **flows better** in the target language. (3) Translation 2 correctly uses the phrase in the source sentence while Translation 1 has a small error, which is not contextually correct. (4) Translation 1 does not have a Russian translation of the English text. (5) Translation 2 was more emotive than the original text. (6) Translation 1 is misleading whereas Translation 2 **speaks on the actual events.**

Table 20: Feedback from the human annotators. We refer to *Translation 1* as initial translation and *Translation 2* as output translation from our fine-tuned model.

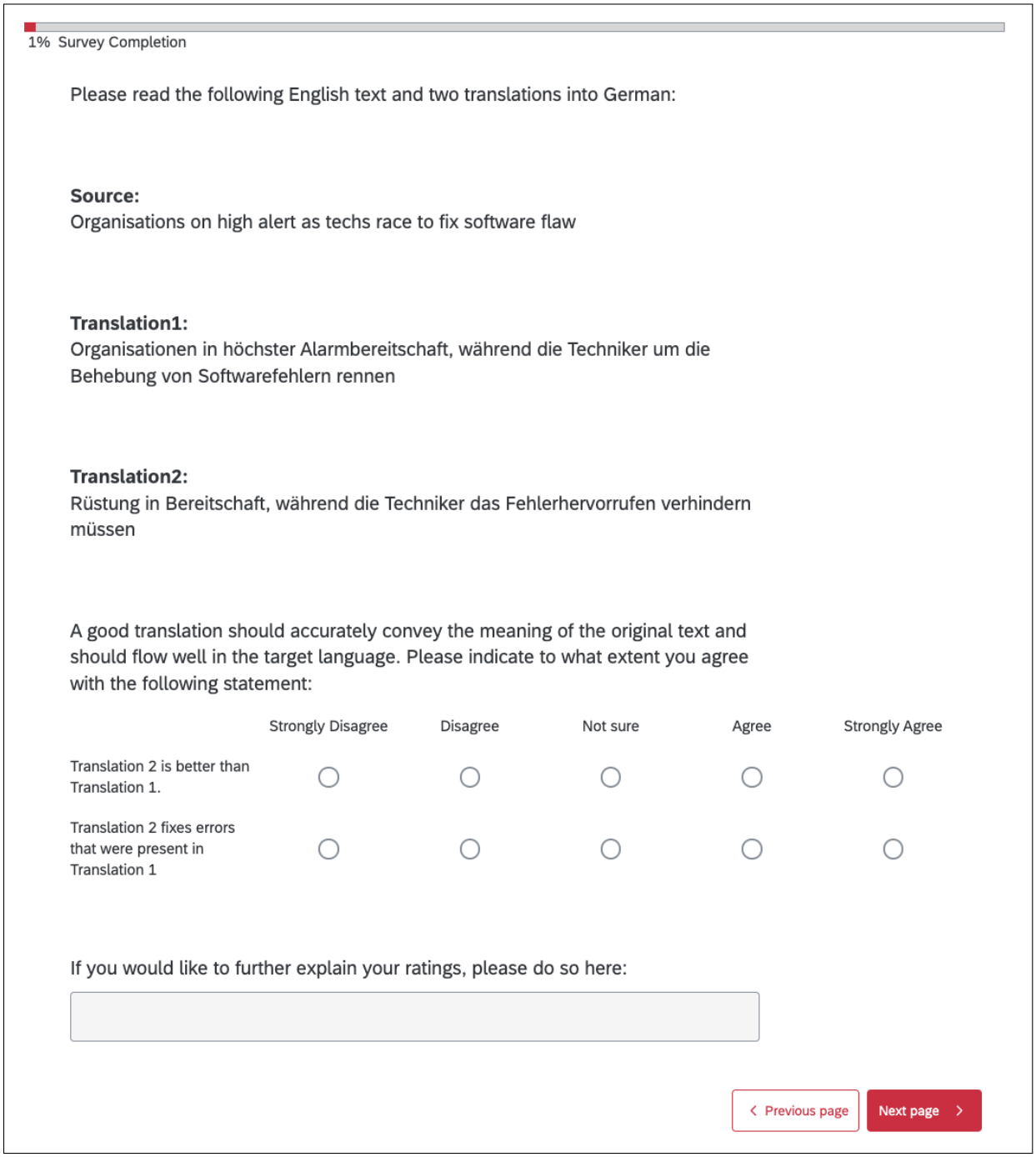

Figure 8: Survey content for human evaluation. Given **Source**, **Translation 1** (original translation), and **Translation 2** (output translation from the bilingual fine-tuned model), annotators are asked to answer 2 questions on a scale from 0 to 5. Extra text box is given for each example for further suggestions.