Quality Estimation and Post-Editing Using LLMs For Indic Languages: How Good Is It?

Anushka Singh[1,2] **Aarya Pakhale**[1,4]
Mitesh M. Khapra[1,2] **Raj Dabre**[1,2,3,5]

[1]Nilekani Centre at AI4Bharat [2]Indian Institute of Technology Madras, India
[3]National Institute of Information and Communications Technology, Kyoto, Japan
[4] Indian Institute of Technology Kharagpur, India
[5]Indian Institute of Technology Bombay, India

Abstract

Recently, there have been increasing efforts on Quality Estimation (QE) and Post-Editing (PE) using Large Language Models (LLMs) for Machine Translation (MT). However, the focus has mainly been on high resource languages and the approaches either rely on prompting or combining existing QE models with LLMs, instead of single end-to-end systems. In this paper, we investigate the efficacy of end-to-end QE and PE systems for low-resource languages taking 5 Indian languages as a use-case. We augment existing QE data containing multidimentional quality metric (MQM) error annotations with explanations of errors and PEs with the help of proprietary LLMs (GPT-4), following which we fine-tune Gemma-2-9B, an open-source multilingual LLM to perform QE and PE jointly. While our models attain QE capabilities competitive with or surpassing existing models in both reference-based and reference-free settings, we observe that they still struggle with PE. Further investigation reveals that this occurs because our models lack the ability to accurately identify fine-grained errors in the translation, despite being excellent indicators of overall quality. This opens up opportunities for research in end-to-end QE and PE for low-resource languages. The synthetic dataset and evaluation metrics are publicly accessible online.[1]

1 Introduction

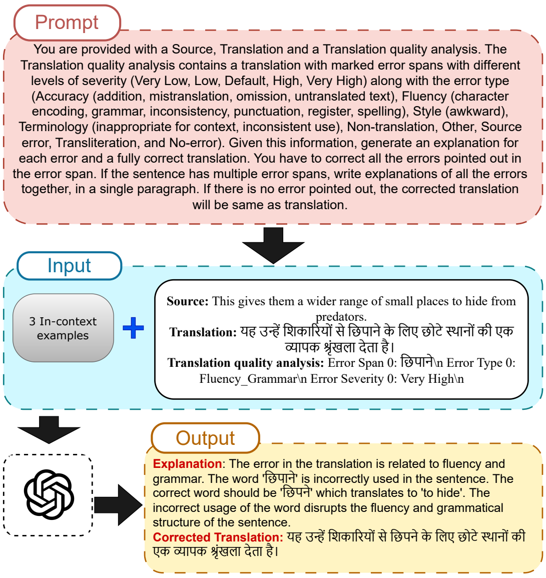
The rapid advancements in Large Language Models (LLMs) (Achiam et al., 2023; Touvron et al., 2023; Riviere et al., 2024) have significantly impacted Machine Translation (MT) leading to state-of-the-art translation quality. This quality is usually measured at the corpus level using a variety of quality estimation (Zerva et al., 2024) metrics among which COMET (supervised) and GEMBA (prompting-based)(Kocmi and Federmann, 2023) are known to be the best. Specifically, COMET has spurred research into language-family specific versions of COMET like in the case of Indic langugaes (Sai B et al., 2023). Closely related is the problem of post-editing where once a poor quality translation has been detected, mistakes in translation need to be suitably fixed (Bhattacharyya et al., 2023).

Recently, Treviso et al. (2024) have shown that it is possible to take error annotations of COMET models and the power of synthetic explanations generated by GPT-4, to develop a system that can post-edit erroneous translations thereby improving translation quality. Their main focus was showing that error explanations in human understandable formats lead to improved post-edits by LLMs. On the other hand, Lu et al. (2025) have leveraged LLMs purely in prompting mode in multiple stages to first annotate errors, choose the most reliable ones, and then post-edit to improve translation quality. However, existing works have two major limitations: a. They do not focus on a singular end-to-end model which does error annotations, error explanations and post-editing in one go. b. They focus on high-resource languages, which makes it difficult to determine the impact on low-resource languages.

In this paper we attempt to fill this gap by focusing on English to Indian languages (En→X) directions – specifically for five Indian languages: Hindi, Gujarati, Marathi, Malayalam and Telugu, which are considered low-resource in the world of quality estimation and post editing. Given the low-resource setting, we ask a simple question: *How good is an all-purpose end-to-end error annotation, explanation and post-editing system for Indian languages in a low-resource setting?*. This leads to 3 specific research questions (RQs):

(RQ1): How well do Large Language Models per-

[1] https://github.com/AI4Bharat/QE-PE-MTEval.git



You are provided with a Source, Translation and a Translation quality analysis. The Translation quality analysis contains a translation with marked error spans with different levels of severity (Very Low, Low, Default, High, Very High) along with the error type (Accuracy (addition, mistranslation, omission, untranslated text), Fluency (character encoding, grammar, inconsistency, punctuation, register, spelling), Style (awkward), Terminology (inappropriate for context, inconsistent use), Non-translation, Other, Source error, Transliteration, and No-error). Given this information, generate an explanation for each error and a fully correct translation. You have to correct all the errors pointed out in the error span. If the sentence has multiple error spans, write explanations of all the errors together, in a single paragraph. If there is no error pointed out, the corrected translation will be same as translation.

Source: This gives them a wider range of small places to hide from predators.
Translation: यह उन्हें शिकारियों से छिपाने के लिए छोटे स्थानों की एक व्यापक श्रृंखला देता है।
Translation quality analysis: Error Span 0: छिपाने\n Error Type 0: Fluency_Grammar\n Error Severity 0: Very High\n

Explanation: The error in the translation is related to fluency and grammar. The word 'छिपाने' is incorrectly used in the sentence. The correct word should be 'छिपने' which translates to 'to hide'. The incorrect usage of the word disrupts the fluency and grammatical structure of the sentence.
Corrected Translation: यह उन्हें शिकारियों से छिपने के लिए छोटे स्थानों की एक व्यापक श्रृंखला देता है।

Figure 1: Overview of the approach used to generate synthetic post-edits and explanations. The figure illustrates the prompt design, input structure, and model-generated output. The prompt specifies how translation quality is analyzed, with error spans and severity levels guiding the generation of explanations and corrected translations

form in evaluating machine translation quality for Indian languages, considering both reference-based and reference-free scenario?

(RQ2): Do explanations of errors and error span detection by LLMs lead to demonstrable improvements in post-editing performance for Indian languages?

(RQ3): Does joint QE and PE, affect QE?

Taking motivation from (Treviso et al., 2024), we augment the Indic MT Evaluation dataset (Sai B et al., 2023) with synthetic explanations and post-edits (see Figure 1) and fine-tune GEMMA2(Riviere et al., 2024) to obtain a single model to generate error annotations (used for computing MQM scores for QE), error explanations and post-edits. On the positive side, we find that QE significantly surpasses all existing models like COMETKiwi, however, unlike previous works, we observe that error annotation and explanation does not often lead to higher translation quality after post-editing. Upon further investigation, we find that this mainly occurs because the limited amount of training data leads to models, which are good at evaluating overall translation quality, but are not always reliable at fine-grained quality estimation. Specifically, they tend to under-detect certain error categories or sometimes misclassify errors, leading to inconsistencies in post-editing corrections. This shows that we are still far away from using LLMs for fine-grained error annotation and use it for post-editing in low-resource settings. Our contributions are:

(i) State-of-the-art quality estimation models for 5 Indian languages in the En→X setting.

(ii) Augmented quality estimation dataset with error explanations and GPT4 post-edits.

(iii) A reality check that LLMs are still unreliable for fine-grained quality estimation and post-editing in low-resource settings.

2 Related Work

Research in the machine translation (MT) evaluation has evolved significantly, driven by the need for more accurate and interpretable metrics. Traditional MT evaluation metrics can be broadly classified into Reference-based and Reference-free approaches. Early metrics, such as BLEU (Papineni et al., 2002) and chrF (Popovic, 2017), primarily relied on lexical overlap between machine translations and human references, often failing to align well with human judgments.

More recent neural based metrics like, COMET (Rei et al., 2020) and BLEURT (Sellam et al., 2020) have shown stronger correlations with humans, but these metrics lack interpretability. These metrics have further improved with the introduction to models like XCOMET (Guerreiro et al., 2024) and COMETKiwi (in reference-free direction) (Rei et al., 2022, 2023) . However, XCOMET primarily detects error spans and their severity without classifying the specific type of error. We aim to explore whether LLMs can capture fine-grained translation errors by identifying their types alongside assessing severity, focusing on Indian languages.

In parallel, the exploration of Large Language Models (LLMs) for MT evaluation has gained momentum (Kocmi and Federmann, 2023; Xu et al., 2023), with research examining their effectiveness in assessing translation quality. While these approaches have been widely explored for high resource languages, their performance for Indian languages, which are notoriously resource poor for quality estimation, remains unexplored.

Additionally, research suggests that fine-grained error analysis and explanations can improve post-editing efficiency (Treviso et al., 2024; Lu et al., 2025). However, our findings indicate that such

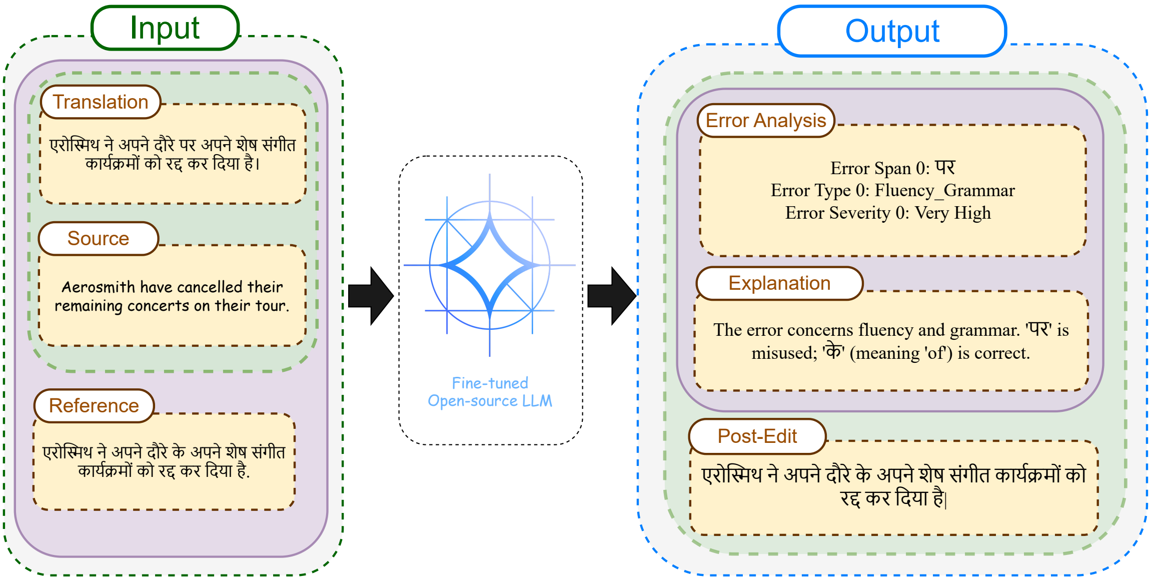

Input
Translation
एरोस्मिथ ने अपने दौरे पर अपने शेष संगीत कार्यक्रमों को रद्द कर दिया है।
Source
Aerosmith have cancelled their remaining concerts on their tour.
Reference
एरोस्मिथ ने अपने दौरे के अपने शेष संगीत कार्यक्रमों को रद्द कर दिया है.
Fine-tuned Open-source LLM
Output
Error Analysis
Error Span 0: पर
Error Type 0: Fluency_Grammar
Error Severity 0: Very High
Explanation
The error concerns fluency and grammar. 'पर' is misused; 'के' (meaning 'of') is correct.
Post-Edit
एरोस्मिथ ने अपने दौरे के अपने शेष संगीत कार्यक्रमों को रद्द कर दिया है|

Figure 2: Overview of fine-tuned LLM models for translation quality assessment. The green box represents the reference-free setting, while the purple box represents the reference-based setting. Given an input consisting of a translation and source (with or without a reference and error analysis), we train models to generate one or more of the *error analysis* (fine-grained MQM style error annotations), *error explanations* and *post-edits* as applicable. Section 4.3 shows all possible model configurations we consider.

benefits may not necessarily extend to low-resource Indian languages, highlighting the need for further investigation into language-specific factors affecting post-editing and evaluation.

3 Methodology

Our approach leverages synthetic explanations and post-edits from LLMs followed by fine-tuning open-source LLMs to enhance a large language model's (LLM) ability to detect, explain, and correct machine translation errors in both reference-based and reference-free settings.

3.1 Error Explanations and Post-Edits

For the tasks of error analysis and post-editing, we generated synthetic explanations and post-edits using a proprietary API based model. Our approach, inspired by (Treviso et al., 2024) is shown in Figure 1. Our initial experiments with zero-shot prompting yielded suboptimal outputs, highlighting the need for more guided generation. To address this, we adopted a 3-shot prompting strategy, incorporating carefully selected in-context examples augmented with explanations and corrections.

The in-context examples were derived from expert annotations provided by bilingual linguists proficient in the target languages. Each linguist was presented with the source sentence, its machine translation, and pre-identified error spans, along with information on error type and severity. They were asked to provide detailed explanations for each error and generate a corresponding post-edited translation that reflects natural and fluent usage. Each expert annotated approximately 10 translation segments per language. From this pool, we manually selected three high-quality examples per language to serve as in-context demonstrations for API based model, enabling it to generate consistent and high-quality explanations and post-edits across the broader dataset.

3.2 Joint Quality Estimation and Post-Editing

Using the original QE data augmented with error explanations and post-edits, we fine-tune an open-source multilingual model in a variety of configurations. Figure 2 gives an overview and Section 4.3.2 details the training setups.

4 Experimental Setup

We now describe specifics of our experimental setup, namely datasets and languages, baselines, model configurations we tested, QE meta-evaluation and PE evaluation approaches.

Metric	**Hindi**		**Malayalam**		**Marathi**		**Tamil**		**Gujarati**		**Average**	
	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
COMET$_{MQM}$	0.441	0.597	0.405	0.516	0.365	0.490	0.498	0.654	0.426	0.487	0.427	0.549
Indic-COMET$_{MQM}$	0.479	0.656	0.441	0.557	0.394	0.538	0.523	0.677	0.473	0.552	0.462	0.596
Base-IndicBERT$_{MQM}$	0.438	0.638	0.443	0.517	0.370	0.512	0.437	0.576	0.487	0.582	0.435	0.565
XCOMET-XL	0.496	0.630	0.471	0.597	0.430	0.557	0.580	$\underline{0.740}$	0.512	0.630	0.498	0.631
XCOMET-XXL	0.597	0.744	**0.642**	**0.696**	0.524	$\underline{0.641}$	**0.602**	**0.747**	**0.610**	$\underline{0.643}$	0.526	**0.694**
MetricX23-XL	0.419	0.401	0.457	0.427	0.388	0.406	0.465	0.396	0.452	0.449	0.436	0.416
MetricX23-XXL	0.439	0.333	0.417	0.391	0.476	0.421	0.323	0.478	0.323	0.533	0.438	0.422
MetricX24-XL	0.409	0.490	0.478	0.544	0.379	0.509	$\underline{0.597}$	0.510	0.532	0.689	0.479	0.550
MetricX24-XXL	0.397	0.360	$\underline{0.486}$	0.520	0.386	0.470	0.438	0.401	0.554	**0.720**	0.452	0.494
ErrSp	**0.776**	**0.778**	0.470	$\underline{0.665}$	$\underline{0.616}$	**0.657**	0.509	0.589	$\underline{0.600}$	0.410	**0.594**	$\underline{0.620}$
ErrSp-Exp	$\underline{0.754}$	$\underline{0.766}$	0.449	0.592	**0.637**	0.602	0.346	0.422	0.596	0.397	$\underline{0.556}$	0.556

Table 1: Segment-level Pearson (ρ) and Kendall tau (τ) scores for evaluation models in the reference-based setting.

4.1 Languages and Dataset Agumentation

For our experiments, we employed the IndicMT Eval dataset(Sai B et al., 2023), which comprises 1,476 examples per language, covering Hindi, Marathi, Malayalam, Tamil, and Gujarati. The dataset was partitioned into training, validation, and test sets containing 1000, 200 and 276 examples, respectively, for each language.

To enrich the dataset with explanations and post-edits, we employed the GPT-4 API to generate synthetic explanations and post-edits using a 3-shot prompting strategy(refer Figure 1). Building upon existing prompt design (Treviso et al., 2024), we incorporated expert-annotated in-context examples to enhance the quality and relevance of the generated explanations and corrections.

While leveraging LLMs for synthetic data generation offers scalability, it also introduces challenges such as generic meta-phrases or contextually irrelevant content. To mitigate these, we iteratively refined prompts, curated in-context examples, and incorporated human verification steps. This meticulous process resulted in well-structured training and validation pairs tailored for error detection, explanation generation, and post-edit prediction.

Additionally, to gauge the quality and utility of the synthetic annotations, we conducted a human evaluation wherein annotators assessed 20 GPT-4-generated explanations per language. The feedback was largely positive, particularly for Hindi, Gujarati, and Marathi. These findings were further corroborated by COMET-22 score comparisons, which showed notable improvements in 76% of Hindi cases, 50% of Marathi, and 44% of Gujarati. Although Tamil (29%) and Malayalam (36%) saw more modest gains, they still reflect incremental improvements attributable to the synthetic data.

4.2 Implementation and Training

We fine-tuned the Gemma-2-9B (Riviere et al., 2024) model on a diverse set of machine translation evaluation tasks, as shown in Figure 2. We initially experimented with fine-tuning LLaMA-3 (Touvron et al., 2023) models; however, their performance was suboptimal compared to Gemma-2, and hence we focused only on the latter. Fine-tuning was conducted with LoRA with a rank of 2 and an alpha value of 16 to optimize memory efficiency while maintaining model performance. For training we used a batch size of 8, a learning rate of 1.5e-4, and BF16 precision. Training was conducted using the open-instruct library[2].

4.3 Models Compared

We describe baselines followed by our various model configurations we tested.

4.3.1 Baselines

All existing baselines we consider only have the capability to do QE and we compare them with the QE capabilities of models we train. We compared our QE results against COMET(MQM) (Rei et al., 2020), IndicCOMET and its variants (Sai B et al., 2023; Singh et al., 2024), MetricX23 (Juraska et al., 2023), MetricX24 (Juraska et al., 2024), XCOMET (in a reference-based setting), and COMETKiwi (for a reference-free setting).

4.3.2 Our Models

We have reference-based models for QE and error explanation and reference-free models for QE, error explanation, and PE. Detailed in Appendix A **Reference-based QE Models** These take in source, translation and a reference and produce:

1. **ErrSp:** Error Annotations (error spans).

[2] https://github.com/allenai/open-instruct

Metric	Hindi		Malayalam		Marathi		Tamil		Gujarati		Average	
	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ	ρ	τ
COMET_QE$_{MQM}$	0.487	0.651	0.354	0.457	0.302	0.416	0.485	0.650	0.359	0.370	0.397	0.509
IndicCOMET$_{MQM}$	0.507	0.675	0.424	0.507	0.349	0.470	**0.526**	**0.680**	0.434	0.428	0.448	0.552
Base-IndicBERT$_{MQM}$	0.439	0.632	0.409	0.520	0.362	0.479	0.476	0.596	0.445	0.547	0.426	0.555
COMET_Kiwi	0.542	0.634	0.458	0.480	0.392	0.475	0.482	0.393	0.494	0.681	0.474	0.533
COMET_Kiwi-XL	0.521	0.586	0.448	0.457	0.405	0.480	0.458	0.287	0.498	0.581	0.466	0.478
COMET_Kiwi-XXL	0.528	0.646	0.448	0.501	0.415	0.526	0.473	0.479	0.451	0.605	0.463	0.551
MetricX23-XL	0.464	0.455	0.423	0.285	0.371	0.300	0.447	0.197	0.443	0.503	0.430	0.348
MetricX23-XXL	0.550	0.417	0.484	0.334	0.424	0.369	0.499	0.241	0.538	0.600	0.499	0.392
MetricX24-XL	0.424	0.593	0.419	0.492	0.326	0.443	0.465	0.486	0.482	0.650	0.423	0.533
MetricX24-XXL	0.461	0.581	0.454	0.501	0.386	0.459	0.399	0.435	0.517	**0.717**	0.443	0.539
ErrSp	0.779	**0.777**	0.641	0.429	0.619	0.634	0.438	0.536	**0.611**	0.403	0.618	0.556
ErrSp-Exp	0.726	0.731	0.594	0.434	**0.621**	**0.644**	0.456	0.374	0.575	0.368	0.594	0.510
ErrSp-Exp-PE	0.754	0.765	0.656	0.457	0.588	0.621	0.370	0.479	0.582	0.374	0.590	0.539
ErrSp-Exp-PE*gpt*	0.753	0.763	0.569	0.452	0.567	0.592	0.443	0.361	0.541	0.343	0.575	0.502
ErrSp-PE	0.753	0.742	**0.697**	**0.560**	0.615	0.642	0.473	0.561	0.604	0.412	**0.628**	**0.583**
ErrSp-PE*gpt*	**0.783**	0.773	0.672	0.506	0.586	0.612	0.455	0.523	0.584	0.368	0.616	0.556

Table 2: Segment-level Pearson (ρ) and Kendall tau (τ) scores for evaluation models in the referenceless setting.

2. **ErrSp-Exp:** 1 + human readable explanations (henceforth explanations).

Reference-free QE and PE Models These take in only source and translation and produce:

1. **ErrSp:** Error Annotations (error spans).
2. **ErrSp-Exp:** 1 + explanations.
3. **PE:** Post-edits with the original reference was used as the post-edit during training.
4. **PE$_{gpt}$:** Post-edits with the GPT generated correction as the post-edit during training.
5. **ErrSp-Exp-PE:** 2+3
6. **ErrSp-Exp-PE$_{gpt}$:** 2+4
7. **ErrSp-PE:** 1+3
8. **ErrSp-PE$_{gpt}$:** 1+4

Additionally, we trained some control models specifically for the purposes of PE, to determine if PE quality improves when the correct error spans are supplied to the model as a part of the prompt (*ip*). To this end, we take the correct error spans as inputs along with the source and translation as a part of the model prompt when training.

9. **ErrSp-ip-PE:** Analogous to 7.
10. **ErrSp-ip-PE$_{gpt}$:** Analogous to 8.
11. **ErrSp-ip-Exp-PE:** Analogous to 5.
12. **ErrSp-ip-Exp-PE$_{gpt}$:** Analogous to 6.

4.4 QE and PE Evaluation

To meta-evaluate the QE capabilities of models, we follow Rei et al. (2020) and compute Pearson and KendallTau correlations of MQM scores computed using predicted MQM error spans against those done by humans. For PE, we compute chrF (Popovic, 2017) and COMET-22 scores of the post-edit generated by the model against the human written reference.

5 Result

In this section, we present the evaluation results of our LLM-based approach for MT quality assessment of Indian languages, addressing the research questions outlined in Section 1. Section 5.1 addresses **RQ1** by evaluating the performance of our models under both reference-based and reference-free settings, comparing them against state-of-the-art MT evaluation systems. Section 5.2 focuses on **RQ2**, investigating whether error annotations and explanations enhance post-editing performance. Additionally, throughout both sections, we explore **RQ3**, analyzing whether joint quality estimation (QE) and post-editing (PE) influence QE performance. By structuring our results around these questions, we provide a comprehensive assessment of LLM capabilities for low-resource MT evaluation.

5.1 LLM-Based MT Evaluation for Indian Languages

Table 1 presents the results of reference-based MT evaluation. Our LLM-based approach achieves competitive performance, comparable to the significantly larger XCOMET-XXL (10.7B) model. Notably, unlike XCOMET-XXL, our method identifies error spans with greater diversity in both category and severity (refer Table 6 for details). Our system demonstrates strong performance for Hindi and Marathi, but we observe comparatively lower

Metric	Hin	Mal	Mar	Tam	Guj
pre-edit	**48.89** / 0.737	47.67 / 0.839	48.47 / 0.729	48.33 / 0.850	50.96 / 0.851
PE	45.24 / 0.733	45.06 / 0.842	42.71 / 0.703	45.80 / 0.854	44.99 / 0.851
PE*gpt*	48.86 / 0.733	48.46 / 0.842	48.89 / 0.703	49.12 /0.854	51.59 / 0.851
ErrSp-PE	45.16 / 0.738	45.69 / 0.840	43.69 / 0.711	47.41 / 0.863	47.05 / **0.859**
ErrSp-PE*gpt*	48.66 / **0.743**	48.03 / 0.832	48.75 / 0.734	48.60 / 0.843	51.17 / 0.846
ErrSp-Exp-PE	47.12 / 0.665	43.54 / 0.736	43.63 / 0.684	44.97 / 0.669	47.45 / 0.761
ErrSp-Exp-PE*gpt*	46.69 / 0.673	44.78 / 0.725	46.12 / 0.698	44.06 / 0.707	47.47 / 0.745
ErrSp-ip-Exp-PE	46.96 / 0.731	45.26 / 0.838	43.53 / 0.717	47.69 / 0.841	45.17 / 0.843
ErrSp-ip-Exp-PE*gpt*	47.43 / 0.714	46.32 / 0.815	48.82 / 0.730	48.40 / 0.815	49.52 / 0.831
ErrSp-ip-PE	44.61 / 0.730	44.85 / 0.837	43.50 / 0.707	46.06 / 0.856	46.90 / 0.855
ErrSp-ip-PE*gpt*	48.70 / **0.743**	**48.92 / 0.845**	**49.00 / 0.737**	**49.99 / 0.858**	**51.71** / 0.854

Table 3: ChrF and COMET scores of model-suggested post-edits vs. reference. Scores are in X/Y format, where X is ChrF and Y is COMET. The "pre-edit" row shows ChrF and COMET scores for MT output vs. reference.

performance for Gujarati and Tamil. This discrepancy suggests language-specific challenges, which require further investigation.

The reference-free evaluation results in Table 2 highlight that our model achieves state-of-the-art performance. Specifically, our model ranks second-best when only predicting error spans but outperforms all models when tasked with both error span detection and post-editing. This underscores the effectiveness of LLMs in evaluating MT quality, particularly when integrating error correction. Consistent with our reference-based findings, the strongest performance is observed for Devanagari-script languages (Hindi and Marathi), reinforcing the notion that script and linguistic features play a crucial role in quality estimation. We also observed that our model got a relatively lower Pearson score; the reason can be the non-linear relationship between model predicted scores and actual MQM scores, the presence of clustered values around certain score ranges (e.g., 0.6, 0.8, and 1.0), and the skewed distribution, which weakens Pearson ability to capture a strong linear correlation despite maintaining a high rank correlation (KendallTau).

5.2 Impact of Error Analysis on Post-Editing

In this section, we analyze the impact of error analysis on post-editing, with a particular focus on **RQ3**, which examines whether joint quality estimation (QE) and post-editing (PE) influence QE performance. Table 3 presents ChrF++ and COMET scores for both original machine translations (pre-edits) and their best post-edited versions. Contrary to prior work suggesting that error explanations significantly improve post-editing quality (Treviso et al., 2024), our results show only marginal gains across Indian languages. Interestingly, while error detection leads to notable improvements in reference-free QE (as shown in Section 5.1), these gains do not consistently carry over to post-editing. The highest ChrF++ and COMET scores are observed when error annotations are available, yet the improvements remain modest, underscoring the limitations of LLM-based post-editing in low-resource settings. Our findings suggest that joint modeling of QE and PE does not consistently enhance QE performance. Although the best results are achieved when combining error analysis with post-editing, the addition of explanations does not yield further benefits. One potential reason for this can be the scarcity of high-quality training data. In contrast to high-resource languages, where fine-grained error analysis and explanations can drive significant improvements, LLMs struggle to generate precise, actionable feedback for low-resource languages. These results indicate that while LLMs show promise in overall MT quality estimation, they remain less reliable for fine-grained quality assessment and post-editing in low-resource scenarios.

6 Conclusion

Our study investigates the role of Large Language Models (LLMs) in machine translation (MT) evaluation for Indian languages, addressing key challenges in fine-grained quality estimation (QE) and post-editing (PE). We leveraged synthetic error explanations and post-edits from GPT-4 and fine-tuned the GEMMA-2-9B model in a variety of settings for reference-based QE and reference-free QE and PE. In reference-based settings we got comparable if not slightly better QE performance against existing strong baselines. On the other hand, in reference-free settings we obtained significantly improved QE performance. However in the case of PE, contrary to previous works in high-resource settings, involving error detection and explanation in the PE framework does not lead to improved post-edited translations. The explanation for this is in the poor fine-grained error detection capabilities of our fine-tuned models due to low-resource settings. This indicates a dire situation but opens avenues for future research on joint QE and PE for low-resource languages.

7 Limitations

This study examined LLM performance on a selection of Indian languages. Future research should broaden this scope to encompass a more diverse set, particularly low-resource languages. Furthermore, even with fine-tuning, LLM post-editing performance for Indian languages requires improvement. To this end, better strategies for low-resource post-editing need to be studied. Another limitation of this work is the limited amount of synthetic data created which should also be a future topic of investigation.

8 Sustainability Statement

Experiments were conducted using a private infrastructure, which has a carbon efficiency of 0.45 kgCO_2eq/kWh. A cumulative of 48 hours of computation was performed on hardware of type A100 PCIe 40GB (TDP of 250W). Total emissions are estimated to be 5.4 kgCO_2eq of which 0 percents were directly offset. Given the low-resource nature of our work, we do not expect our work to have any large negative environmental impact.

Estimations were conducted using the Machine-Learning Impact calculator presented in (Lacoste et al., 2019).

References

OpenAI Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Benjamin Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Sim'on Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Raphael Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Lukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Hendrik Kirchner, Jamie Ryan Kiros, Matthew Knight, Daniel Kokotajlo, Lukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Adeola Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel P. Mossing, Tong Mu, Mira Murati, Oleg Murk, David M'ely, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Ouyang Long, Cullen O'Keefe, Jakub W. Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alexandre Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Pondé de Oliveira Pinto, Michael Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario D. Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin D. Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas A. Tezak, Madeleine Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cer'on Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2023. Gpt-4 technical report.

Pushpak Bhattacharyya, Rajen Chatterjee, Markus Freitag, Diptesh Kanojia, Matteo Negri, and Marco Turchi. 2023. Findings of the WMT 2023 shared task on automatic post-editing. In *Proceedings of the Eighth Conference on Machine Translation*, pages 672–681, 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.

Juraj Juraska, Daniel Deutsch, Mara Finkelstein, and Markus Freitag. 2024. MetricX-24: The Google submission to the WMT 2024 metrics shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 492–504, Miami, Florida, USA. Association for Computational Linguistics.

Juraj Juraska, Mara Finkelstein, Daniel Deutsch, Aditya Siddhant, Mehdi Mirzazadeh, and Markus Freitag. 2023. MetricX-23: The Google Submission to the WMT 2023 Metrics Shared Task. In *Proceedings of the Eighth Conference on Machine Translation*, pages 756–767, Singapore. Association for Computational Linguistics.

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

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.

Qingyu Lu, Liang Ding, Kanjian Zhang, Jinxia Zhang, and Dacheng Tao. 2025. MQM-APE: Toward high-quality error annotation predictors with automatic post-editing in LLM translation evaluators. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5570–5587, Abu Dhabi, UAE. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popovic. 2017. chrf++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation, WMT 2017, Copenhagen, Denmark, September 7-8, 2017*, pages 612–618. Association for Computational Linguistics.

Ricardo Rei, Nuno M. Guerreiro, JosÃ© Pombal, Daan van Stigt, Marcos Treviso, Luisa Coheur, José G. C. de Souza, and André Martins. 2023. Scaling up CometKiwi: Unbabel-IST 2023 submission for the quality estimation shared task. In *Proceedings of the Eighth Conference on Machine Translation*, pages 841–848, 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.

Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022. CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Gemma Team Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, L'eonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ram'e, Johan Ferret, Peter Liu, Pouya Dehghani Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stańczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Boxi Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Christoper A. Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozi'nska, D. Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Pluci'nska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost R. van Amersfoort, Josh Gordon, Josh Lipschultz, Joshua Newlan, Junsong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, L. Sifre, Lena Heuermann, Leti cia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Gorner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Peng chong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Perrin, S'ebastien M. R. Arnold, Se bastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomás Kociský, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeffrey Dean, Demis Hassabis, Koray Kavukcuoglu, Clément Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size. *ArXiv*, abs/2408.00118.

Ananya Sai B, Tanay Dixit, Vignesh Nagarajan, Anoop Kunchukuttan, Pratyush Kumar, Mitesh M. Khapra, and Raj Dabre. 2023. IndicMT eval: A dataset to meta-evaluate machine translation metrics for Indian languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14210–14228, Toronto, Canada. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7881–7892. Association for Computational Linguistics.

Anushka Singh, Ananya Sai, Raj Dabre, Ratish Puduppully, Anoop Kunchukuttan, and Mitesh Khapra. 2024. How good is zero-shot MT evaluation for low resource Indian languages? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 640–649, Bangkok, Thailand. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

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.

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.

Chrysoula Zerva, Frederic Blain, José G. C. De Souza, Diptesh Kanojia, Sourabh Deoghare, Nuno M. Guerreiro, Giuseppe Attanasio, Ricardo Rei, Constantin Orasan, Matteo Negri, Marco Turchi, Rajen Chatterjee, Pushpak Bhattacharyya, Markus Freitag, and André Martins. 2024. Findings of the quality estimation shared task at WMT 2024: Are LLMs closing the gap in QE? In *Proceedings of the Ninth Conference on Machine Translation*, pages 82–109, Miami, Florida, USA. Association for Computational Linguistics.

Model Name	Inputs Provided	Outputs Expected
Reference-Based		
`ErrSp`	Source, Translation, Reference	Error Spans
`ErrSp-Exp`	Source, Translation, Reference	Error Spans + Explanations
`ErrSp-ip-Exp`	Source, Translation, Reference, Error Spans	Explanations
Reference-Free		
`ErrSp`	Source, Translation	Error Spans
`ErrSp-Exp`	Source, Translation	Error Spans + Explanations
`ErrSp-Exp-PE`	Source, Translation	Error Spans + Explanations + Post-Edits
`ErrSp-ip-Exp`	Source, Translation, Error Spans	Explanations
`ErrSp-ip-Exp-PE`	Source, Translation, Error Spans	Explanations + Post-Edits
`ErrSp-ip-PE`	Source, Translation, Error Spans	Post-Edits
`ErrSp-PE`	Source, Translation	Error Spans + Post-Edits
`PE`	Source, Translation	Post-Edits

Table 4: Overview of GEMMA fine-tuning tasks under reference-based and reference-free settings. Each task is defined by the specific inputs provided and the expected outputs the model learns to generate.

Metric	Hin	Mal	Mar	Tam	Guj
Err_Sp Exp	59.46	46.18	55.03	46.78	52.05
Err_Sp Exp PE	58.83	55.11	46.10	49.31	52.97
Err_Sp_Exp PE-gpt	59.26	49.37	41.74	43.71	47.87
Err_Sp_ip Exp	70.17	61.63	55.70	65.47	61.65
Err_Sp_ip Exp PE	70.20	60.94	55.29	64.96	60.81
Err_Sp_ip Exp PE-gpt	70.46	61.67	56.35	65.38	61.47

Table 5: chrF scores of model-suggested explanation vs. GPT generated explanation

A Training Data Preparation

To fine-tune GEMMA-9B for translation quality estimation and post-editing tasks, we constructed a diverse set of input-output training pairs using synthetic error explanations and post-edits. The model was trained under two major settings: *reference-based* (using human reference translations) and *reference-free* (using only the source and machine translation). Table 4 summarizes the task variants explored under each setting.

Figure 2 shows an example prompt for the `ErrSp-Exp-PE` task in the reference-free setting. Other task prompts follow similar structures, differing in the presence or absence of reference translations, error spans, or expected outputs (e.g., explanations, corrections).

For reference-free training, we experimented with two post-edit supervision strategies: one using GPT-4 generated outputs ($\texttt{PE}_{\texttt{gpt}}$), and the other using human references (`PE`). This comparison helps evaluate the reliability of synthetic supervision in low-resource scenarios.

Error Category		**Explanation**
Accuracy	Addition Omission Mistranslation Untranslated text	Translation includes information not present in the source. Translation is missing content from the source. Translation does not accurately represent the source. Source text has been left untranslated
Fluency	Spelling Grammar Register Character Encoding	Incorrect spelling or capitalization. Problems with grammar, other than orthography. Wrong grammatical register (eg, inappropriately informal pronouns). Characters are garbled due to incorrect encoding. Example: Sink ->$ink
Terminology Inappropriate		Terminology is non-standard or does not fit context.
Style Awkward		The style of the text does not feel very apt. (Example: 1. The source sentence feels formal like in a newspaper, but the translation doesn't. 2. Sentences are correct, but simply too long, etc..)
Transliteration		If it transliterates instead of translating words/ phrases, where it should not.
Other		Any other issues.
Source Error		An error in the source.
Non Translation		Impossible to reliably characterize the 5 most severe errors.

Table 6: This table outlines the error categories our models are capable of detecting in machine translation outputs. It includes a comprehensive list of common translation errors, ranging from accuracy issues like additions and omissions to fluency problems such as spelling and grammar mistakes. The categorization is adapted from previous work IndicMT-eval(Sai B et al., 2023)