TRANSLATIONCORRECT: A Unified Framework for Machine Translation Post-Editing with Predictive Error Assistance

Syed Mekael Wasti[*,†] **Shou-Yi Hung**[*,‡] **Christopher Collins**[†] **En-Shiun Annie Lee**[†,‡]
[†]Ontario Tech University [‡]University of Toronto
syedmekael.wasti@ontariotechu.net, sy.hung@mail.utoronto.ca

Abstract

Machine translation (MT) post-editing and research data collection often rely on inefficient, disconnected workflows. We introduce TRANSLATIONCORRECT, an integrated framework designed to streamline these tasks. TRANSLATIONCORRECT combines MT generation using models like NLLB, automated error prediction using models like XCOMET or LLM APIs (providing detailed reasoning), and an intuitive post-editing interface within a single environment. Built with human-computer interaction (HCI) principles in mind to minimize cognitive load, TRANSLATIONCORRECT makes it easier for annotators to perform annotations, as confirmed by a user study using NASA Task Load Indices. For translators, it enables them to correct errors and batch translate efficiently. For researchers, TRANSLATIONCORRECT exports high-quality span-based annotations in the Error Span Annotation (ESA) format, using an error taxonomy inspired by Multidimensional Quality Metrics (MQM). These outputs are compatible with state-of-the-art error detection models and suitable for training MT or post-editing systems. Our user study confirms that TRANSLATIONCORRECT significantly improves translation efficiency and user satisfaction over traditional annotation methods.

1 Introduction

Machine translation (MT) has seen significant advancements with the development of powerful translation models like Meta's No Language Left Behind (Team et al., 2022, NLLB) and evaluation tools such as XCOMET (Guerreiro et al., 2024). However, the current translation and data collection workflows for MT model training remain inefficient. Traditional translation procedures often require human annotators to rely on manual, time-consuming processes involving tools like CSV files or Excel sheets (Federmann, 2018). Typically, a translator must first generate machine translations using an external model, then manually collect and transfer the output into another format for review. Any subsequent error correction must also be performed manually, resulting in an inefficient and error-prone process.

Similar challenges also exist in the data collection process for MT research. Datasets used for training MT systems are often complex to collect, as they have to undergo the tedious manual process mentioned earlier. However, to improve the efficiency of the annotation process, annotation tools like Appraise (Federmann, 2018) have been developed to facilitate the whole process, making MT training data collection easier and standardizing the data collection procedure. However, Appraise remains a platform dedicated to experienced annotators and linguists, enabling them to annotate data for future research and model training, which limits its usage to a specific group of users.

To address these limitations, we introduce TRANSLATIONCORRECT, a framework designed to streamline both translation workflows and MT data collection. For translators, TRANSLATIONCORRECT offers a solution that automatically generates initial translations using a translation model, such as NLLB and identifies potential translation errors using XCOMET or an LLM of choice to provide more insights into the translation errors, enabling efficient post-editing of translations within the same environment. This approach eliminates the need for manual data handling through external tools, improving both translation quality and efficiency. For researchers in the MT community, TRANSLATIONCORRECT also serves as a robust data collection tool, automatically formatting outputs in alignment with state-of-the-art MT dataset standards, supporting outputs that contain Multidimensional Quality Metrics (Burchardt, 2013, MQM) and Error Span Annotation (Kocmi et al., 2024, ESA) information alongside each translation

[*]Equal contribution, corresponding author

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), pages 551–562

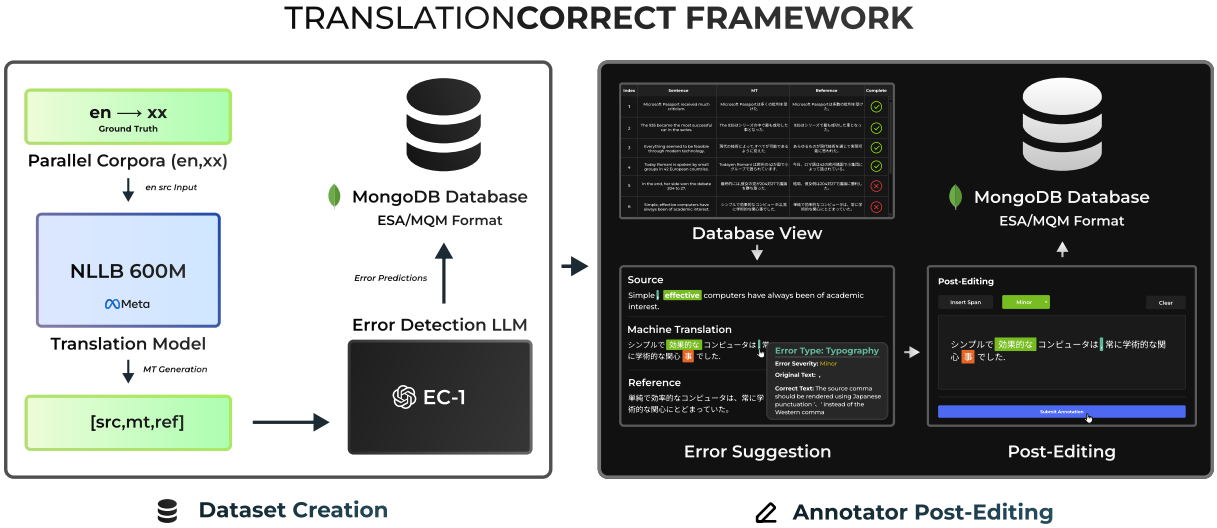

Figure 1: Overview of the TRANSLATIONCORRECT framework. The workflow begins with an annotator fetching data from a previously populated database we create for en→xx language sets. We process our collections using the EC-1 error detection model and analyze the MT output to identify potential errors. The user sees these selected sentences and the relevant predicted errors within the post-edit section, where they can correct the translation based on these suggestions before submitting the final annotated sentence.

source and target pair. This feature enables annotators to generate high-quality datasets that can be used directly for training error correction models like XCOMET or fine-tuning translation systems like NLLB.

Furthermore, our framework is designed with human-computer interaction (HCI) principles in mind, prioritizing ease of use and flexibility for annotators. The user interface is designed to minimize cognitive load and reduce the difficulty typically associated with traditional annotation workflows, such as those relying on manual data processing (Norman, 1983; Hustak et al., 2015) through Microsoft Excel. By integrating MT generation, error prediction, and correction within a single environment, our framework enables translators to focus on the translation task itself, rather than having to work with multiple tools simultaneously. Evaluation results from a user study indicate that our framework significantly outperforms traditional annotation methods, resulting in a considerably lower perceived workload and increased efficiency compared to conventional annotation methods.

Our contributions are summarized as follows:

- TRANSLATIONCORRECT offers an integrated environment that automatically generates initial translations using translation models, predicts potential errors using error detection models or an LLM of choice, and enables efficient corrections.
- The framework supports output formats aligned with state-of-the-art MT dataset standards, including MQM and ESA, enabling researchers and annotators to generate high-quality datasets for training and fine-tuning MT and translation error detection models.
- Designed with HCI principles in mind, TRANSLATIONCORRECT prioritizes ease of use and flexibility, reducing cognitive load for annotators.

Our repository is MIT Licensed and is publicly available on GitHub[1]. Our deployed demo is available on a website[2]. A short demo of our framework is available on YouTube[3].

2 TRANSLATIONCORRECT

An overview of TRANSLATIONCORRECT's workflow is illustrated in Figure 1, outlining the user flow from target language dataset selection to automatic error detection, user post-edit, and data export.

2.1 Database View & MT Generation

When users first enter the TRANSLATIONCORRECT framework, they are presented with a

[1]https://github.com/MekaelWasti/TranslationCorrect
[2]https://translation-correct-annotation-git-27a7e8-mekaelwastis-projects.vercel.app/
[3]https://youtu.be/j2sp13qyeQM

Figure 2: Annotators can use the database view to easily view their target language dataset, select their desired sentences, and monitor completion status

database view interface to input the source text that requires translation. Annotators can load sentences from the view, containing multiple source and MT pairs. The dataset is stored in MongoDB, which holds the precomputed pairs of source sentences and machine-translated sentences. For most of our datasets, we have used a 600M NLLB model[4] to create machine translations; however, as we add increasingly lower-resource languages, we can switch to other models that support them. The source text and translated output are displayed side by side, as shown in Figure 3, enabling the user to compare and assess the translation quality easily. Furthermore, the annotated data is saved to the same database, permitting easy access.

2.2 Error Detection

Following the MT generation, TRANSLATIONCORRECT integrates an error detection model of the user's choice to identify potential errors in the translated output automatically. For our demo, we offer two methods, with the first being XCOMET-XL[5], the 3.5B parameter variant of XCOMET, which will be used as a baseline error detection model, and the other being a custom GPT-4o assistant named EC-1.

2.2.1 Custom GPT-4o EC-1 Assistant

We offer the option to apply a custom GPT-4o assistant, EC-1, to help users identify potential errors with an in-depth explanation, as shown in Figure 3. EC-1 is model-agnostic; other LLMs capable of structured JSON error-span output could be used in place of GPT-4o. However, 4o is cost-effective, fast and performs error detection with consistent accuracy compared to the tested OpenAI models. We leverage this model as an error detection model, using prompt engineering techniques to ensure that the response provided by our custom-crafted EC-1 assistant aligns with our standardized error ruleset, as outlined in Appendix B. As shown in Figure 3, translation errors are highlighted in different colors, present in both the source sentence and the MT output, allowing users to identify potential errors with minimal effort. Furthermore, a detailed explanation of the error is displayed when the user hovers their cursor above the highlighted text. Our human study shows that this provides a more in-depth analysis than using only the XCOMET model.

The EC-1 assistant's response is obtained from an API endpoint, allowing it to be used in minimal client-side and limited computing environments without requiring additional computational resources to run local models; however, API calls to GPT-4o may incur large cloud usage costs depending on usage and the size of input datasets. Implementation details of our custom EC-1 model can be found in Appendix C.

2.3 User Post-Editing

Once errors are identified, users can correct translation errors directly within the system, as shown in Figure 4. If the suggested errors do not match the user's expectations, they are allowed to make fine-grained edits to modify the detected errors and the final translated sentence.

Users are also allowed to insert custom new error spans that the error detection model did not previously highlight.

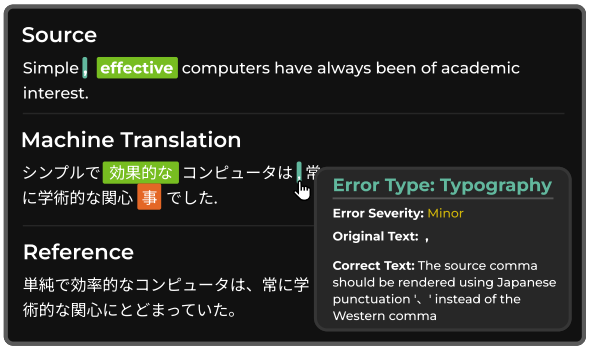

Figure 3: Predicted errors annotated by our error detection model are highlighted in both the source text and the machine translation output, with a detailed description of the error identified and its source-to-MT mapping.

[4] https://huggingface.co/facebook/nllb-200-distilled-600M

[5] https://huggingface.co/Unbabel/XCOMET-XL

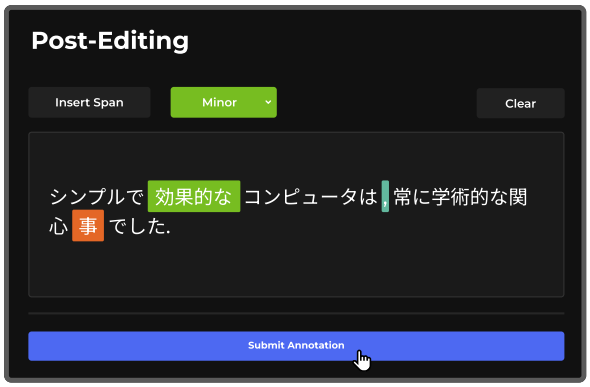

Figure 4: The Post-Edit component allows users to make detailed, fine-grained error edits on top of the potential error spans generated by our error detection model.

2.4 Data Export

Once the post-editing process is completed, the user can export the final translation and annotations into a structured dataset. This feature allows one-click data export in a format compatible with MQM and ESA standards.

The exported data can be downloaded from the interface in multiple formats, including CSV and JSON. It contains information on the source text, MT output, corrected text, error spans, error categories, and error severities.

In addition to annotators manually downloading the data, the server manager can also fetch the annotated data from the MongoDB database connected to the server, allowing for easier management and exportation of the annotated data.

2.5 HCI Considerations

To design an interface that reduces cognitive load, multiple HCI principles must work in tandem. TRANSLATIONCORRECT's interface is simple and clutter-free. This reduces the likelihood of annotators becoming overwhelmed or fatigued by unnecessary content on the screen. We ensured a strict workflow to minimize noise between annotators' submissions.

A dark theme was chosen for the application to reduce visual fatigue from bright colors during long sessions. This was well received and praised by participants in our study. Additionally, vibrant and unique colors were chosen to represent different error categories, allowing users to quickly associate colors with categories, which is especially helpful when viewing error predictions. The interface also provides quick action shortcuts that appear near the annotator's cursor for crucial operations, such as inserting and deleting spans. The local pop-up reduces the distance required for mouse movement and speeds up the annotation process.

A comprehensive user study, further elaborated in the following section, confirms that users find the framework more effective, enjoyable, and efficient than traditional spreadsheet-based annotation workflows.

3 Results and Evaluation

To evaluate the efficacy of TRANSLATIONCORRECT's interface design and the cognitive workload compared to manual annotation methods, a user study was conducted. The **NASA Task Load Index (Hart and Staveland, 1988, TLX)** was used to measure workload across six dimensions: **Mental Demand**, **Physical Demand**, **Temporal Demand**, **Performance**, **Effort**, and **Frustration**. Overall, the results indicate that using TRANSLATIONCORRECT, particularly with our EC-1 error detection model, resulted in significantly lower perceived workload compared to the traditional Excel-based annotation method.

The participant pool comprised 12 annotators across 6 languages (Mandarin, Cantonese, Bengali, French, Japanese, Tamil). All annotators were native speakers of the respective non-English language and participated voluntarily. Details of the data collection process on the user study can be found in Appendix E. The study was conducted under the following conditions:

User Study Conditions Each participant annotated 8 unique sentences, with 2 annotations per condition, under 4 different conditions:

1. **Manual Annotation with Excel:** Participants were provided with a spreadsheet containing source text, machine translation, and reference text. A color guide was used to annotate error categories and severities manually. More details of the instructions provided to participants can be found in Appendix D.

2. **TRANSLATIONCORRECT without Suggestions:** Participants used the TRANSLATIONCORRECT interface with no model-generated error detections.

3. **TRANSLATIONCORRECT with *XCOMET* Suggestions:** Participants received automatic error span suggestions from the XCOMET

Method	Mental (↓)	Physical (↓)	Temporal (↓)	Performance (↑)	Effort (↓)	Frustration (↓)
Excel	4.10 ± 2.51	3.40 ± 2.88	2.70 ± 2.26	7.80 ± 1.55	4.10 ± 2.38	3.50 ± 2.92
No Suggestions	4.17 ± 2.52	2.42 ± 2.57	3.58 ± 2.02	8.58 ± 1.16	3.42 ± 1.16	1.83 ± 2.41
XCOMET	2.92 ± 1.56	**1.58 ± 1.51**	2.50 ± 1.68	**8.67 ± 1.07**	**2.67 ± 1.07**	1.92 ± 2.31
EC-1	**2.67 ± 1.87**	**1.58 ± 1.08**	**2.17 ± 1.59**	8.50 ± 1.00	3.08 ± 1.00	**1.75 ± 2.26**

Table 1: Comparison of NASA TLX dimensions across annotation methods, with Excel annotations done following instructions outlined in Appendix D, and the different error detection settings used within TRANSLATIONCORRECT. Lower is better (↓) for all metrics except Performance (↑). Bold indicates the best score for each metric.

model, which were pre-highlighted in the interface.

4. **TRANSLATIONCORRECT with *EC-1* Suggestions:** Participants used the full system with GPT-4o-based error detection, which included both span highlighting and explanatory tooltips.

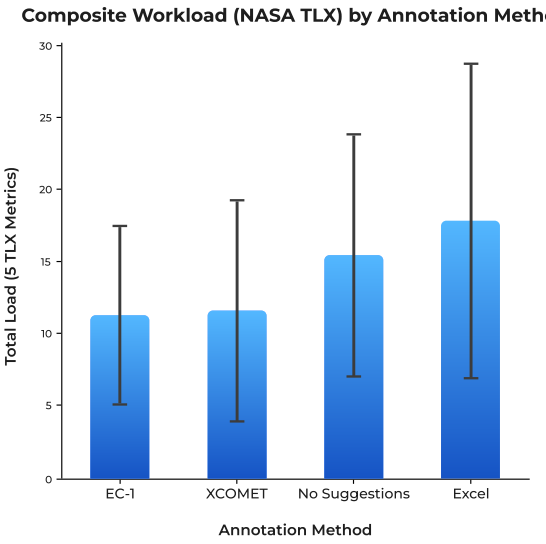

Figure 5: Composite Total Workload across Annotation Methods, calculated as the sum of five NASA TLX dimensions (Mental, Physical, Temporal, Effort, Frustration). Lower scores reflect reduced perceived workload.

Figure 5 presents the composite workload scores across each annotation method in the study. The Excel manual annotation method shows the highest average workload, followed by the "No Suggestions" condition within TRANSLATIONCORRECT. Both error detection models, EC-1 and XCOMET conditions, demonstrated substantially lower average workload scores, indicating a reduction in cognitive burden on users. The error bars indicate considerable variability within workload ratings for the Excel and "No Suggestion" methods, while the EC-1 and XComet conditions exhibited more consistent results.

Composite Workload Calculation. The *Total Load* in Figure 5 is computed as the simple sum of the five TLX dimensions—Mental Demand, Physical Demand, Temporal Demand, Effort, and Frustration—following NASA-TLX guidelines (Hart and Staveland, 1988). We exclude Performance from this composite since it measures perceived success (higher is better), whereas the other five metrics indicate workload (lower is better). No additional weighting or post-processing was applied.

Table 1 presents the results of the user study from an HCI perspective. Across all NASA TLX dimensions, TRANSLATIONCORRECT consistently outperformed the manual Excel-based annotation method, demonstrating significant reductions in **mental demand**, **effort**, and **frustration**, while also improving **perceived performance**. These results demonstrate the effectiveness of our framework in streamlining translation workflows and alleviating the cognitive burden on annotators.

To better understand the internal relationships between different workload factors, we computed Pearson correlation coefficients between TLX dimensions. As shown in Table 2, cognitive and emotional burdens—particularly **Mental Demand**, **Effort**, and **Frustration**—were positively correlated, confirming the internal consistency of the TLX framework in our study. **Perceived Performance** was negatively correlated with most workload dimensions, most notably with **Physical Demand** ($r = -0.44$), suggesting that reducing user effort and fatigue may directly contribute to greater perceived success.

Statistical Analysis

To assess significance across our four annotation methods *(Excel, No Suggestions, XComet, EC-1)*, we applied the **Friedman test** to each NASA TLX

	Mental	Physical	Temporal	Effort	Frustration	Performance
Mental	1.00	0.44	0.46	0.47	0.52	-0.06
Physical	0.44	1.00	0.49	0.34	0.30	-0.44
Temporal	0.46	0.49	1.00	0.37	0.25	-0.20
Effort	0.47	0.34	0.37	1.00	0.60	-0.26
Frustration	0.52	0.30	0.25	0.60	1.00	-0.28
Performance	-0.06	-0.44	-0.20	-0.26	-0.28	1.00

Table 2: Correlation Matrix of the NASA TLX Metrics

dimension. Significant differences were found for **Mental Demand**, **Physical Demand**, and **Frustration** ($\chi^2(3) = 11.09$, $p = .011$; $\chi^2(3) = 10.42$, $p = .015$; $\chi^2(3) = 7.88$, $p = .049$).

For focused comparisons between Excel and EC-1, we ran **Wilcoxon signed-rank tests**, which confirmed that EC-1 **significantly reduced**:

- **Mental Demand** ($W = 2.5$, $p = .010$),
- **Physical Demand** ($W = 2.0$, $p = .041$),
- **Frustration** ($W = 0.0$, $p = .027$).

NASA TLX scores are ordinal and not normally distributed, making non-parametric tests appropriate. We thus used the Friedman test for within-subject comparisons across conditions, and Wilcoxon signed-rank tests for focused pairwise contrasts.

These results corroborate that our predictive-error interface meaningfully lowers the annotator's cognitive and emotional workload compared to a standard spreadsheet baseline. These findings further support the HCI-driven design choices in TRANSLATIONCORRECT, such as predictive error suggestions and minimizing interface friction through quick action buttons corresponding to crucial post-editing tasks intended to reduce cognitive and physical load.

4 Conclusions and Future Work

In this work, we introduced TRANSLATIONCORRECT, a unified framework designed to streamline MT workflows while enhancing data collection for MT research. By integrating MT generation, error prediction, and translation post-editing within a single, user-friendly environment, TRANSLATIONCORRECT significantly improves translation efficiency and user satisfaction while annotating. Our framework also ensures that the annotated data collected from human annotators using our framework can be exported with state-of-the-art MT dataset standards, following MQM and ESA standards. As this paper focuses on annotation tooling, no accompanying dataset has been published. Conducting human annotations is a lengthy process, and we are working on creating a large and quality-assured dataset with TRANSLATIONCORRECT used for annotation. The benefits of our framework assist both translators by offering a seamless post-editing experience and researchers by providing high-quality, standardized datasets for fine-tuning current models, such as XCOMET and NLLB, as well as newer models that will be released in the future.

Empirical evaluation demonstrates that TRANSLATIONCORRECT outperforms traditional translation workflows, such as those annotation workflows based on Excel, in terms of both efficiency and user satisfaction. Our user study indicates that translators find our framework intuitive, efficient, and enjoyable, highlighting the importance of HCI considerations in our framework.

4.1 Continuous fine-tuning

While our framework has already enhanced translation workflows, there is potential to incorporate continuous fine-tuning improvements into the underlying models when using our framework. One promising direction is to collect user-corrected data to fine-tune both the translation model (NLLB) and the error detection model (XCOMET). This additional feature would allow the system to dynamically improve based on the specific translation domain in which users are working, reducing the number of errors in the initial proposed translation and the number of errors detected by the error detection model.

Furthermore, as we have chosen NLLB and similar models as our translation model, alongside XCOMET as our error detection model, we can employ Low-Rank Adaptation (Hu et al., 2022, LoRA) and other parameter-efficient strategies to carry out the fine-tuning process on limited compute. By integrating lightweight fine-tuning techniques, users could personalize their MT pipeline

while maintaining efficiency on a local machine without needing to deploy anything on the cloud.

Nevertheless, the collection of data to carry out the continuous fine-tuning procedures remains difficult, thus, this direction remains a possible extension of our framework in the future.

Multimodal Extensions

While our current framework is focused on text-based machine translation, we envision future extensions to support ASR (speech-to-text) and OCR (image-to-text) modalities. In such cases, the transcribed source (via ASR/OCR) would serve as input to the translation pipeline, followed by the same error detection and post-editing workflow. This would make the framework applicable to low-resource regions or archival content where text is not readily available. We leave implementation and evaluation of this multimodal pipeline for future work.

Limitations

While our evaluation results demonstrate significant gains in translation efficiency and quality, some limitations remain:

- Our user study was limited to 12 translators across 6 languages (Mandarin, Cantonese, Bengali, French, Japanese, Tamil), which may introduce sampling bias and limit generalizability. This evaluation was intended as a usability study to assess the effectiveness of the proposed framework, rather than a large-scale statistical evaluation.

- While TRANSLATIONCORRECT streamlines translation workflows, the final translation quality ultimately still depends on the skill and expertise of human annotators.

- Although TRANSLATIONCORRECT supports low-resource MT models like NLLB, our current evaluation does not validate performance on low-resource languages.

- Our custom GPT-4o assistant might not perform as expected when the source or target language is a low-resource language, as it is not trained intensively in those languages.

- The EC-1 assistant relies on OpenAI's GPT-4o API, which may incur usage costs and raise data privacy concerns. Future work will explore open-weight LLMs such as Mixtral or LLaMA to mitigate these limitations.

By addressing these limitations, TRANSLATIONCORRECT has the potential to become an adaptive, user-driven translation framework, continuously improving through feedback while maintaining high usability and annotation efficiency. We hope that our framework will set a new standard for both translation workflows and MT data collection, bridging the gap between human expertise and machine translation systems.

Ethical Impact Statement

The user study involved requesting the participants to carry out simple translation tasks and MT post-editing tasks, all in the TRANSLATIONCORRECT framework and Microsoft Excel.

We have obtained approval from the Review Ethics Board to conduct the human study for our framework. The user study has active REB approval (File No: 18021).

Acknowledgments

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), Undergraduate Student Research Award (USRA).

References

Luisa Bentivogli, Mauro Cettolo, Marcello Federico, and Christian Federmann. 2018. Machine translation human evaluation: an investigation of evaluation based on post-editing and its relation with direct assessment. In *Proceedings of the 15th International Conference on Spoken Language Translation*, pages 62–69, Brussels. International Conference on Spoken Language Translation.

Frederic Blain, Chrysoula Zerva, Ricardo Rei, Nuno M. Guerreiro, Diptesh Kanojia, José G. C. de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, Constantin Orasan, and André Martins. 2023. Findings of the WMT 2023 shared task on quality estimation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 629–653, Singapore. Association for Computational Linguistics.

Aljoscha Burchardt. 2013. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*, London, UK. Aslib.

Christian Federmann. 2018. Appraise evaluation framework for machine translation. In *Proceedings of the*

27th International Conference on Computational Linguistics: System Demonstrations, pages 86–88, Santa Fe, New Mexico. Association for Computational Linguistics.

Markus Freitag, Nitika Mathur, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Tom Kocmi, Frederic Blain, Daniel Deutsch, Craig Stewart, Chrysoula Zerva, Sheila Castilho, Alon Lavie, and George Foster. 2023. Results of WMT23 metrics shared task: Metrics might be guilty but references are not innocent. In *Proceedings of the Eighth Conference on Machine Translation*, pages 578–628, Singapore. Association for Computational Linguistics.

Christian Girardi, Luisa Bentivogli, Mohammad Amin Farajian, and Marcello Federico. 2014. MT-EQuAl: a toolkit for human assessment of machine translation output. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 120–123, Dublin, Ireland. Dublin City University and 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.

Sandra G. Hart and Lowell E. Staveland. 1988. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In Peter A. Hancock and Najmedin Meshkati, editors, *Human Mental Workload*, volume 52 of *Advances in Psychology*, pages 139–183. North-Holland.

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*.

Xu Huang, Zhirui Zhang, Xiang Geng, Yichao Du, Jiajun Chen, and Shujian Huang. 2024. Lost in the source language: How large language models evaluate the quality of machine translation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3546–3562, Bangkok, Thailand. Association for Computational Linguistics.

Tomas Hustak, Ondrej Krejcar, Ali Selamat, Reza Mashinchi, and Kamil Kuca. 2015. Principles of usability in human-computer interaction driven by an evaluation framework of user actions. In *Mobile Web and Intelligent Information Systems*, pages 51–62, Cham. Springer International Publishing.

Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024. Error span annotation: A balanced approach for human evaluation of machine translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1440–1453, Miami, Florida, USA. Association for Computational Linguistics.

Donald A. Norman. 1983. Design principles for human-computer interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '83, page 1–10, New York, NY, USA. Association for Computing Machinery.

Ricardo Rei, Jose Pombal, Nuno M. Guerreiro, João Alves, Pedro Henrique Martins, Patrick Fernandes, Helena Wu, Tania Vaz, Duarte Alves, Amin Farajian, Sweta Agrawal, Antonio Farinhas, José G. C. De Souza, and André Martins. 2024. Tower v2: Unbabel-IST 2024 submission for the general MT shared task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 185–204, Miami, Florida, USA. 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.

Kirill Semenov, Vilém Zouhar, Tom Kocmi, Dongdong Zhang, Wangchunshu Zhou, and Yuchen Eleanor Jiang. 2023. Findings of the WMT 2023 shared task on machine translation with terminologies. In *Proceedings of the Eighth Conference on Machine Translation*, pages 663–671, Singapore. Association for Computational Linguistics.

Ana Silva, Nikit Srivastava, Tatiana Moteu Ngoli, Michael Röder, Diego Moussallem, and Axel-Cyrille Ngonga Ngomo. 2024. Benchmarking low-resource machine translation systems. In *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, pages 175–185, Bangkok, Thailand. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

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.

Longyue Wang, Chenyang Lyu, Tianbo Ji, Zhirui Zhang, Dian Yu, Shuming Shi, and Zhaopeng Tu. 2023. Document-level machine translation with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16646–16661, Singapore. Association for Computational Linguistics.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, Mexico City, Mexico. Association for Computational Linguistics.

A Related Works

NLLB (Team et al., 2022), a translation model released by Meta AI, addresses translation task challenges by expanding translation capabilities to over 200 languages. NLLB demonstrates a significant advancement in MT performance over multiple metrics included in the WMT Shared Task (Freitag et al., 2023).

To evaluate the translation qualities of MT systems, metrics such as the MQM and ESA offer systematic approaches to analyze translation outputs. MQM (Burchardt, 2013) is a comprehensive framework that categorizes translation errors based on predefined typologies and severity levels. MQM formalizes an analytic evaluation method by assigning translation errors to categories such as accuracy, fluency, and style, allowing for more thorough quality assessments. The framework has been widely adopted in the MT community with multiple variants (Blain et al., 2023; Rei et al., 2020; Guerreiro et al., 2024; Kocmi et al., 2024), serving as one of the most widely used human evaluation metrics for MT tasks.

While MQM offers detailed insights, it is time-consuming and often requires expert annotators, making large-scale evaluations and data collection costly and resource-intensive. To address these limitations, the ESA (Kocmi et al., 2024) framework was introduced as a more efficient alternative. ESA combines elements of Direct Assessment (Bentivogli et al., 2018, DA) with error span marking alongside clear annotation instructions, enabling annotators to highlight specific error spans and assign severity scores. This method retains much of MQM's specificity while reducing the cognitive load on annotators, as it provides clear guidelines to distinguish between different errors, allowing for more efficient and meaningful data collection. Extensive studies by Kocmi et al. show that ESA can match MQM's effectiveness in system ranking while significantly reducing the time and expertise required for annotations.

As the demand for scalable MT evaluation grows, automatic metrics capable of providing interpretable and fine-grained assessments on MT outputs have gained more attention. XCOMET (Guerreiro et al., 2024) represents a significant advancement in this domain by combining traditional sentence-level evaluation with detailed error span detection. Building on the foundations of earlier neural translation metrics like COMET (Rei et al., 2020), which focus on generating a single sentence-level quality score, XCOMET introduces the capability to detect and underline specific translation errors within a sentence. This improvement, specific to XCOMET, enables it to highlight error spans and assess their severity, in addition to a single sentence-level quality score, providing more interpretable evaluations that closely align with human evaluations.

Recently, efforts were also placed into utilizing LLMs to provide a detailed analysis of translation errors. xTower (Treviso et al., 2024) is one such example that gives detailed descriptions and explanations of translation errors spans provided to the model. Treviso et al. have demonstrated that xTower can enhance the interpretability of translation errors identified by XCOMET.

While tools like Appraise (Federmann, 2018) and MT-EQuAl (Girardi et al., 2014) remain widely used in shared tasks and research settings due to their lightweight interface and support for MQM-style annotations, they are limited in functionality. For example, Appraise does not offer predictive error suggestions or integrated LLM-based assistants. In contrast, our framework assists annotators through interactive error span detection and correction workflows powered by models such as XCOMET and GPT-4o, making it more suitable for real-time annotation and educational use.

Although Appraise has long supported structured MT annotation workflows, our study used Excel as a baseline because it reflects a widely used but friction-heavy process many annotators and researchers may adopt due to a lack of access to specialized annotation tools. We selected Excel to represent a realistic baseline for comparison. Future work may evaluate our framework more directly against Appraise and other task-specific tools to assess annotation quality and efficiency in more detail.

B Standardized Error Definition and Ruleset

In our study, we define several error categories to assess the quality of translations. TRANSLATION-CORRECT allows the annotator to categorize any error spans under these given categories, enabling them to easily select one category to associate with an error span.

To avoid having too many categories with similar definitions, and to ensure that each error cat-

egory is distinct and easily identifiable given an error span, we have simplified the existing categories that MQM (Burchardt, 2013) provides into the following:

- Addition, where content that is not present in the target text appears in the source.
- Omission, which refers to content from the source that is missing in the target
- Mistranslation, where the target text inaccurately represents the source content
- Untranslated, where a segment intended for translation is omitted
- Grammar, which covers violations of grammatical rules in the target language
- Spelling, where words are misspelled
- Typography, which addresses visual presentation issues such as incorrect punctuation, inconsistent capitalization, or spacing errors
- Unintelligible, where the text is garbled or incomprehensible

C Custom GPT-4o assistant dubbed EC-1

To supplement traditional error detection models like XCOMET, we implemented a custom GPT-4o assistant named **EC-1**. This assistant is designed to analyze translation outputs with detailed reasoning and character-level span annotations, offering a more interpretable alternative for translation error detection and annotation.

Prompt Design EC-1 is prompted as a professional linguist specializing in machine translation evaluation. Given a source sentence and its corresponding machine translation, EC-1 is instructed to:

- Detect fine-grained translation errors.
- Label each error with a type from a predefined taxonomy: *Addition, Omission, Mistranslation, Untranslated, Grammar, Spelling, Typography, Unintelligible*.
- Assign each error a severity level: *Minor* or *Major*.
- Provide precise, non-overlapping character-level spans in both source and translation texts.
- Justify each detected error with a brief explanation.

The assistant returns structured, ESA-compatible JSON output for each error. This format is directly compatible with our annotation interface and error span alignment.

Example Use A sample input passed to the EC-1 API is structured as follows:

```
Source: "Today Romani is spoken by small
groups in 42 European countries."
MT: "Todayen Romani は欧州の42か国で小
グループで語られています."
```

EC-1 returns:

```
{
  "error_spans": [
    {
      "original_text": "Today",
      "error_type": "Spelling",
      "error_severity": "Minor",
      "start_index_orig": 0,
      "end_index_orig": 5,
      "start_index_translation": 0,
      "end_index_translation": 7,
      "correct_text": "The word 'Today' is
      incorrectly rendered as 'Todayen'..."
    },
    ...
  ]
}
```

Our prompt emphasizes:

- Non-overlapping spans,
- Strict 0-based character indexing,
- A consistent structure aligned with MQM and ESA principles.

EC-1 responses are integrated directly into the TRANSLATIONCORRECT interface, offering users interpretable, guided suggestions for post-editing.

D Excel Annotation Instructions

To assess the efficacy of traditional annotation methods, we have designed a ruleset for the user study participants to annotate on the given test entries.

As shown by Figure 6, we have asked the annotators to highlight text using multiple different colors to indicate different error categories, as outlined in Appendix B. The annotators are also told to use bold font to indicate if the identified error is a Major error, and a non-bold font to indicate that the error is a minor error.

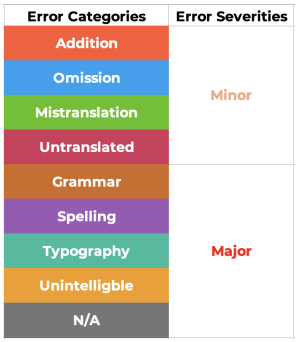

Figure 6: Format that was given to annotators to annotate our test entries with

E User Study Details

We collected our user study data through Google Forms, created the survey using a standard NASA TLX format, and exported user submissions to a CSV format. We then performed statistical analyses on the collected data programmatically using Python and its scientific and numerical packages. A sample of the form [6] used to collect the data in the user study is available. Participants were volunteer student annotators who were native speakers of the respective non-English language they were annotating and were not involved in the authorship of this paper. No monetary compensation was provided.

[6] https://forms.gle/NJcNSPyEBfSUKMVC8