UTSK26 at WAT2026 Patent Claims Translation/Evaluation Tasks

Yingyi Fu, Kosei Buma, Takehito Utsuro
Deg. Prog. Sys.&Inf. Eng., Grad. Sch. Sci.&Tech., University of Tsukuba

Abstract

This paper describes the UTSK26 submission to the WAT2026 Patent Claims Translation/Evaluation Task for the Japanese-to-English (JA–EN) and English-to-Japanese (EN–JA) directions. The task requires participants to annotate translation errors in machine translation (MT) outputs of patent claims and to assign a quality score from 0 to 100 to each translation. For error annotation, we use GEMBA-Patent-Hybrid, which combines an LLM-as-a-Judge method for detecting error spans, categories, and severities using GPT-5.2 with another method for assigning categories and severities to encoder-detected error spans. For quality scoring, we use Patent-Rubric-Checklist, a rubric-based LLM-as-a-Judge method that instructs GPT-5.2 to answer patent-specific checklist items covering five evaluation dimensions and aggregates the answers into a score with predefined weights. On the training data, which consist of the human evaluation results of the WAT2025 task, Patent-Rubric-Checklist achieves the highest Pearson and Spearman correlations with human scores in both directions among the evaluated methods, including COMETKiwi, MetricX-24, GEMBA-DA, and GEMBA-MQM.

1 Introduction

Patent claims are typically written as a single long sentence with complex structures and highly specialized terminology, which makes them difficult to translate. Evaluating their translations is also challenging, because seemingly minor errors, such as the omission of a limitation or an incorrect relationship between claim elements, may substantially change the scope of the claimed invention. In the first Patent Claims Translation Task at WAT2025, Nakazawa et al. (2025) reported that automatic metrics such as COMETKiwi (Rei et al., 2023) and MetricX-24 (Juraska et al., 2024) showed low correlations with human evaluation.

The WAT2026 Patent Claims Translation/Evaluation Task focuses on this evaluation problem. Given patent claims and their MT outputs, participants annotate translation errors with their categories and severities, and assign a quality score to each translation. The ultimate goal of the task is to develop automatic evaluation methods that can accurately assess patent claim translations.

This paper describes the UTSK26 system for both the JA–EN and EN–JA directions. Our system consists of two components:

- **Error annotation:** GEMBA-Patent-Hybrid, which combines two methods: one instructs GPT-5.2 (OpenAI, 2025) to detect error spans, categories, and severities, while the other instructs GPT-5.2 to assign categories and severities to error spans detected by an encoder model.
- **Quality scoring:** Patent-Rubric-Checklist, which instructs GPT-5.2 to evaluate a translation with a rubric of five patent-claim-specific evaluation dimensions and computes the final score from the checklist answers outside the LLM.

2 Task Description

2.1 Task Definition

For each translated claim, participants submit the following evaluation results in the same JSON format as the training data:

- **Error annotation:** Participants mark error spans in the source text or the MT output, and assign each span the lowest-level category of a three-level error hierarchy and a severity of either major or minor. The hierarchy follows the MQM-based criteria (Freitag et al., 2021) adapted for patent translation in WAT2025 (Nakazawa et al., 2025), whose

Proceedings of the 13th Workshop on Asian Translation (WAT 2026), pages 119–133
November 9, 2026 ©2026 Association for Computational Linguistics

	Training		Test	
	JA–EN	EN–JA	JA–EN	EN–JA
Source documents	13	13	2	2
Source claims	28	28	2	3
MT outputs / claim	6	4	5	4
Translated claims	168	112	10	12
Error annotations	1,130	1,017	–	–
Avg. quality score	75.5	66.0	–	–

Table 1: Statistics of the training and test data.

top-level categories are accuracy, fluency, terminology, style, other, source error, and non-translation. For example, an incorrect article is annotated as *Article* under *Accuracy* > *Mistranslation*.

- **Quality scoring:** Participants assign a holistic quality score from 0 to 100 to each translated claim.

2.2 Data

Table 1 shows the statistics of the data. The training data consist of the human evaluation results for a subset of the WAT2025 task data (Nakazawa et al., 2025).[1] For each direction, 28 claims from 13 documents were translated by six (JA–EN) or four (EN–JA) MT systems, and each translation was annotated with error spans and a quality score. The test data consist of 10 JA–EN and 8 EN–JA files, each of which contains a source document and its MT output. The JA–EN files are derived from two single-claim source documents, each paired with five different MT outputs. The EN–JA files are derived from two source documents with three claims in total, each paired with four different MT outputs.

3 System Description

Our system evaluates each translated claim independently. Each line in a text file corresponds to one claim in the test set. Source and MT claims are aligned by line number. The original text is converted into the character-grid format. The grid text is used to construct input to GPT-5.2 for error annotation.

3.1 Error Annotation

We use GPT-5.2 and an encoder-based model for error annotation. For the encoder-based approach, we employ the token-level QE model introduced in our previous work (Buma et al., 2025), using mDeBERTa-v3-base[2] as the multilingual encoder. The model jointly encodes the source sentence and its machine translation, followed by a token-level classification layer that predicts an OK/BAD label for each target token. We then merge consecutive BAD labels into a single error span and record its start and end character offsets. We use three methods for error annotation.

GEMBA-Patent. We input the source and the MT claims to GPT-5.2. The model outputs the error annotations in JSON format, including error sides, tagged error spans, categories, and severities. In particular, for each error, it outputs the full sentence with the error span enclosed by `<ERROR_START>` and `<ERROR_END>` error tags. We then derive the corresponding `[start:end]` offsets from the tag positions. Moreover, we provide patent-specific definitions of the error categories and severities in the prompt. The prompt allows an empty annotation list when no translation error is detected. The detailed prompt for GEMBA-Patent is shown in Figures 1–4 in Appendix A.

Encoder-GEMBA-Patent. We convert the encoder's error spans into the same format as the error spans output by GEMBA-Patent, where for each error, the full sentence is output with the error span enclosed by the error tags. We then input the processed source and MT claims into GPT-5.2. GPT-5.2 assigns an error category and severity to each error. As in GEMBA-Patent, we provide patent-specific definitions of the error categories and severities in the prompt. The detailed prompt for Encoder-GEMBA-Patent is shown in Figures 6–9 in Appendix A.

GEMBA-Patent-Hybrid. For claims whose GEMBA-Patent outputs that do not satisfy our predefined format and validation requirements (invalid outputs), we use the annotations from Encoder-GEMBA-Patent and map the error categories and severities from English to the Japanese labels provided by the task organizers.

Error category. We instruct GPT-5.2 to output error category labels by joining the hierarchy levels with hyphens. We finally map each English error category label to the corresponding lowest-level Japanese category. For example, `"accuracy-mistranslation-numerals/symbols"` is mapped to `"数値・記号"`.

[1] https://github.com/Tzawa/pat-claim-trans

[2] https://huggingface.co/microsoft/mdeberta-v3-base

Labels containing an invalid subcategory combination, such as `"accuracy-mistranslation-inappropriate"`, are mapped to an empty label, `""`.

Severity. For both GEMBA-Patent and Encoder-GEMBA-Patent, we instruct GPT-5.2 to assign either major or minor severity to each error span. A major error changes or obscures the meaning, claim scope, an essential claim element, a technical relationship, or essential claim form, whereas a minor error does not materially affect the meaning or claim scope.

The prompts for these methods were designed mainly with the assistance of ChatGPT. Notably, we provided the patent-specific checklist explained in Section 3.2 to ChatGPT to help design the GEMBA-MQM-based prompts for GEMBA-Patent and Encoder-GEMBA-Patent, including patent-specific definitions of error categories and severities.

3.2 Quality Scoring

For quality scoring, we use Patent-Rubric-Checklist, the same method in our accepted work to appear at WAT2026 (Fu et al., 2026), which is a reference-free LLM-as-a-Judge method based on a rubric specific to patent claim translation evaluation. Inspired by LiTransProQA (Zhang et al., 2025), which evaluates literary translations by having an LLM answer evaluation questions, the method asks an LLM to answer checklist items rather than to predict a score directly.

Rubric. Table 2 summarizes the rubric, which consists of five evaluation dimensions with 16 checklist items in total. **[A] Content Fidelity** evaluates the preservation of the core inventive idea and technical functions or effects, as well as the absence of hallucinated technical content. **[B] Element Coverage** evaluates whether all claim elements are translated and whether element relationships, quantities, ranges, and conditions are preserved. **[C] Claim Scope & Logical Constraints** evaluates the preservation of the claiming scope, the correct attachment of constraint clauses, and the preservation of coordination and modification relations. **[D] Claim Form & Syntactic Requirements** evaluates whether the translation is written as a single-sentence claim, preserves the claim type, maintains a consistent antecedent basis, and avoids narrative or explanatory prose style. **[E] Fluency & Grammaticality** evaluates grammatical correctness, stylistic appropriateness, and readability. The checklist items were mainly developed with the assistance of Copilot, by providing general translation criteria such as semantic equivalence and fluency, as well as patent-specific constraints such as the number of claim elements and their logical relationships, the single-sentence structure of claims, and terminology consistency. The prompt was further refined with ChatGPT.

	Evaluation Dimension	Items	Weight
A	Content Fidelity	3	0.25
B	Element Coverage	3	0.25
C	Claim Scope & Logical Constraints	3	0.25
D	Claim Form & Syntactic Requirements	4	0.15
E	Fluency & Grammaticality	3	0.10

Table 2: Evaluation dimensions of Patent-Rubric-Checklist with the number of checklist items and the weight of each dimension.

Prompt. As shown in Figures 10–12 in Appendix B, the prompt describes the task setting and evaluation procedure, presents the rubric, and specifies the input and output formats. Given a source claim and its MT output, the LLM answers each checklist item with "yes" (no problem), "partial" (minor or weakening problem), or "no" (clear or serious problem), provides a short justification for each answer and an overall explanation, and returns them in JSON format.

Scoring. We map the answers "yes," "partial," and "no" to 1.0, 0.5, and 0.0, respectively. For each evaluation dimension, we average the mapped values of its checklist items to obtain a dimension score, denoted by A through E. The final quality score is then computed outside the LLM with the following weighted formula:

$$\begin{aligned}\text{score} = 100 \times \big(&0.25A + 0.25B \\ &+ 0.25C + 0.15D + 0.10E\big)\end{aligned} \tag{1}$$

The weights in Equation 1 were suggested by Copilot.

3.3 Implementation Details

For both quality scoring and error annotation, newline characters are removed from the model input. For error annotation, the predicted error span offsets are mapped back to the original grid text with newline characters restored.

4 Experiments

4.1 Setup

Since the test data do not include human evaluation results, we validate our system on the training data (Table 1).

Error annotation. We use Encoder-GEMBA-MQM, a GEMBA-MQM-based method, as the comparison method for error annotation. We use the same input format as Encoder-GEMBA-Patent for GPT-5.2. GPT-5.2 then assigns an error category and an error severity to each error. The error categories are modified to match the annotation criteria provided by the task organizers, whereas the error severities are limited to major or minor. The prompt for this method is also designed with the assistance of ChatGPT. The detailed prompt for Encoder-GEMBA-MQM is in Figure 5 in Appendix A.

We compute span-level precision, recall and F1 on the training data, as well as category and severity accuracy over matched spans. A predicted span is considered correct only when its block index, side, and start/end offsets exactly match those of a gold span. Matched spans are counted as true positives, unmatched predicted spans as false positives, and unmatched gold spans as false negatives. Category and severity accuracy are computed only over the matched spans. We compare GEMBA-Patent-Hybrid with Encoder-GEMBA-Patent and Encoder-GEMBA-MQM.

Quality scoring. We compute Pearson's correlation coefficient r and Spearman's rank correlation coefficient ρ between the automatic scores and the human scores of the 168 JA–EN and 112 EN–JA translated claims. We compare Patent-Rubric-Checklist with two reference-free neural metrics, COMETKiwi[3] (Rei et al., 2023) and MetricX-24[4] (Juraska et al., 2024), and two LLM-as-a-Judge methods, GEMBA-DA (Kocmi and Federmann, 2023b) and GEMBA-MQM (Kocmi and Federmann, 2023a). For MetricX-24, which outputs error scores, we negate the scores so that higher is better. GEMBA-DA directly assigns a score from 0 to 100. GEMBA-MQM identifies errors with MQM-style error categories and severities, and we convert its negative raw score s into $100 + 4s$. All LLM-as-a-Judge methods use GPT-5.2. We also conduct permutation tests to compare the correlations of Patent-Rubric-Checklist with those of GEMBA-DA and GEMBA-MQM. In particular, the submitted scores are rounded to the nearest integer, whereas the unrounded scores are used for the correlation analysis.

[3] https://huggingface.co/Unbabel/wmt23-cometkiwi-da-xl

[4] https://huggingface.co/google/metricx-24-hybrid-xl-v2p6

4.2 Results

4.2.1 Results on the Training Set

Error annotation. Table 3 shows the span-level precision, recall and F1 score, as well as category and severity accuracy over matched spans. Because Encoder-GEMBA-Patent and Encoder-GEMBA-MQM use the same error spans, their span-level precision, recall, and F1 are identical. GEMBA-Patent-Hybrid generally shows lower span-level performance than both encoder-based methods in both translation directions, except for precision in JA–EN. We notice that the error annotations of GEMBA-Patent-Hybrid contain a relatively large number of claims with empty annotations. An empty annotation list is allowed by the GEMBA-Patent prompt and indicates that no translation error is detected, as specified in Figure 4. When GEMBA-Patent returns an empty annotation list, the hybrid method treats the output as valid and therefore does not use the annotations assigned by Encoder-GEMBA-Patent. As a result, the hybrid output contains a relatively large proportion of claims with empty annotations. Overall, 33.93% of EN–JA claims and 37.50% of JA–EN claims have empty annotations in GEMBA-Patent-Hybrid, compared with 0.89% and 11.31%, respectively, in the encoder-based methods.

Moreover, in the JA–EN direction, Encoder-GEMBA-MQM achieves the highest category accuracy, while Encoder-GEMBA-Patent achieves the highest severity accuracy. In the EN–JA direction, Encoder-GEMBA-Patent achieves the highest category and severity accuracy.

We also conduct a two-shot experiment to further analyze the effects of adding human-annotated examples and removing the instruction to return empty annotations on GEMBA-Patent. Its setup and results are reported in Appendix C.

Quality scoring. Table 4 shows the correlations between the automatic and human quality scores. Patent-Rubric-Checklist achieves the highest Pearson and Spearman correlations in both directions. For JA–EN, the differences in Spearman correlation

Method	JA–EN Precision	Recall	F1	Category Acc.	Severity Acc.
Encoder-GEMBA-MQM (GPT-5.2)	0.040	**0.028**	**0.033**	**0.406**	0.688
Encoder-GEMBA-Patent (GPT-5.2)	0.040	**0.028**	**0.033**	0.250	**0.719**
GEMBA-Patent-Hybrid	**0.057**	0.011	0.018	0.250	0.667
Method	**EN–JA Precision**	**Recall**	**F1**	**Category Acc.**	**Severity Acc.**
Encoder-GEMBA-MQM (GPT-5.2)	**0.057**	**0.039**	**0.047**	0.375	0.675
Encoder-GEMBA-Patent (GPT-5.2)	**0.057**	**0.039**	**0.047**	**0.500**	**0.700**
GEMBA-Patent-Hybrid	0.108	0.017	0.029	0.353	0.588

Table 3: Error annotation results on the training data.

Method	JA–EN Pearson's r	Spearman's ρ	EN–JA Pearson's r	Spearman's ρ
COMETKiwi	0.2880^{**}	0.2703^{**}	0.1860^{*}	0.1810
MetricX-24 (reference-free)	0.2436^{**}	0.2945^{**}	0.1397	0.1770
GEMBA-DA (GPT-5.2)	0.4685^{**}	0.4210^{**}	0.1996^{*}	0.3017^{**}
GEMBA-MQM (GPT-5.2)	0.5533^{**}	0.4741^{**}	0.2461^{**}	0.3135^{**}
Patent-Rubric-Checklist (GPT-5.2)	$\mathbf{0.6214}^{**}$	$\mathbf{0.5931}^{**}$	$\mathbf{0.3767}^{**}$	$\mathbf{0.4417}^{**}$

$^{*}p < 0.05$, $^{**}p < 0.01$.

Table 4: Correlations between automatic quality scores and human quality scores on the training data.

between Patent-Rubric-Checklist and GEMBA-DA ($p < 0.01$) and between Patent-Rubric-Checklist and GEMBA-MQM ($p < 0.05$) are statistically significant, whereas the differences in Pearson correlation are not. For EN–JA, none of these differences are statistically significant. In addition, all methods show lower correlations for EN–JA than for JA–EN.

4.2.2 Results on the Test Set

The organizers evaluate our submitted system on the test set using gold annotations and gold MQM scores, which are GEMBA-MQM-style scores calculated from gold annotations. Our quality scoring method achieves the highest correlations with human evaluation among the evaluated methods, whereas our error span detection has low recall. Tables 6–7 in Appendix D show the evaluation results of our error annotation method and quality scoring method on the test set as reported by the organizers. For a more detailed evaluation of the submissions, see Nakazawa et al. (2026).

4.3 Submission

We submitted the outputs of the system described in Section 3 for both directions. Because the span-level precision, recall and F1 were calculated after submission, they were not used for system selection. Following GEMBA-MQM, major and minor errors were assigned weights of 5 and 1, respectively, and the negative raw score s was converted to $100 + 4s$. At most five errors per claim were counted, with major errors prioritized over minor errors. The scores used for system selection were computed from the intermediate error annotations before label mapping. We selected GEMBA-Patent-Hybrid because its scores achieved the highest correlation with human scores on the training set among the three evaluated systems. Notably, the choice of scoring process and severity weights was made with the assistance of ChatGPT. Table 8 in Appendix E shows Pearson and Spearman correlations between automatic scores obtained by the three evaluated systems and human scores on the training data.

5 Conclusion

We described the UTSK26 system for the WAT2026 Patent Claims Translation/Evaluation Task. For error annotation, we use GEMBA-Patent-Hybrid, which combines an LLM-as-a-Judge method that detects error spans and assigns categories and severities, with another method that assigns categories and severities to error spans detected by an encoder model. On the training data, it generally showed lower performance than Encoder-GEMBA-Patent and Encoder-GEMBA-MQM. For quality scoring, we used Patent-Rubric-Checklist, which instructs GPT-5.2 to answer patent-claim-specific checklist items and computes a weighted score from the answers. On the training data, it achieves higher correlations with human scores than COMETKiwi, MetricX-24, GEMBA-DA, and GEMBA-MQM in both directions. In future work, we will improve the quality of error annotation, the design of the rubric and checklist items, and optimize the dimension weights to better align with human evaluation.

Limitations

Limited Validation Data We validate our system on the training data, which contain only 28 claims per direction. Therefore, it remains uncertain whether the observed correlations generalize to other patent claims and MT systems.

Encoder-based QE Model The encoder-based QE model used for error annotation was trained on patent-domain data, but its training data did not include patent claims. This may affect the quality of the resulting error annotations.

Stability and Reproducibility We run each prompt only once; therefore, we cannot assess the stability of the LLM outputs. In addition, our quality scoring relies on a proprietary LLM, GPT-5.2, and its performance with open-weight LLMs remains unknown.

Rubric Design The rubric, checklist items, and dimension weights were designed with the assistance of LLM-based assistants and were not optimized on human evaluation data. The rubric is also designed for Japanese–English patent claims and may need to be adapted to other language pairs.

Error Annotation The LLM may output labels containing invalid subcategory combinations, which are mapped to empty labels and may result in missing error category annotations. The GEMBA-Patent-Hybrid method treats an empty annotation list from GEMBA-Patent as a valid output and therefore does not use Encoder-GEMBA-Patent in such cases. As a result, some claims remain without error annotations even when the encoder-based method detects errors. Moreover, because newline characters are removed from the model input (Session 3.3), a space within a predicted error span may be mapped back to a newline character when mapping the span offsets back to the original grid text.

Author Contributions

Yingyi Fu Conducted Patent-Rubric-Chesklist for quality scoring and implemented Encoder-GEMBA-MQM, Encoder-GEMBA-Patent, GEMBA-Patent, and GEMBA-Patent-Hybrid for error annotation.

Kosei Buma Conducted error annotation using an encoder-based model.

Takehito Utsuro Built and managed our team.

References

Kosei Buma, Takehito Utsuro, and Masaaki Nagata. 2025. Two step automatic post editing of patent machine translation based on pre-trained encoder models and LLMs. In *The 14th International Joint Conference on Natural Language Processing and The 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 218–231, Mumbai, India. 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.

Yingyi Fu, Takehito Utsuro, and Masaaki Nagata. 2026. Towards a rubric-based llm-as-a-judge for patent claim translation evaluation. In *Proceedings of the Thirteenth Workshop on Asian Translation (WAT 2026)*. In press.

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.

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.

Toshiaki Nakazawa, Takashi Tsunakawa, Isao Goto, Kazuhiro Kasada, Katsuhito Sudoh, Shoichi Okuyama, Takashi Ieda, and Masaaki Nagata. 2025. Findings of the first patent claims translation task at WAT2025. In *Proceedings of the Twelfth Workshop on Asian Translation (WAT 2025)*, pages 1–15, Mumbai, India. Association for Computational Linguistics.

Toshiaki Nakazawa, Takashi Tsunakawa, Isao Goto, Taiki Ushimaru, Katsuhito Sudoh, Shoichi Okuyama, Takashi Ieda, and Masaaki Nagata. 2026. Findings of the second patent claims translation/evaluation task at WAT2026. In *Proceedings of the Thirteenth Workshop on Asian Translation (WAT 2026)*, Hengqin, China. Association for Computational Linguistics.

OpenAI. 2025. Introducing GPT-5.2. `https://openai.com/index/introducing-gpt-5-2/`.

Ricardo Rei, Nuno M. Guerreiro, José Pombal, Daan van Stigt, Marcos Treviso, Luisa Coheur, José G. C. de Souza, and André F. T. 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.

Ran Zhang, Wei Zhao, Lieve Macken, and Steffen Eger. 2025. LiTransProQA: An LLM-based literary translation evaluation metric with professional question answering. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 29099–29121, Suzhou, China. Association for Computational Linguistics.

A Prompt for Error Annotation

Figures 1–9 show the prompts for error annotation.

B Prompt for Quality Scoring

Figures 10–12 show the prompt for Patent-Rubric-Checklist.

C Few-shot Experiment

Setup. We additionally conduct a two-shot experiment for error annotation using one human-annotated claim from each translation direction in the training set as examples in the prompt. For each translation direction, among documents containing only one claim, we select the document that has the largest number of different error categories in its human annotations. We convert the selected few-shot annotations to the same output format as GEMBA-Patent, excluding annotations with an empty error category or severity. We evaluate on the remaining 27 claims in each direction. We test two prompt versions. GEMBA-Patent-fewshot adds the two examples to the GEMBA-Patent prompt. GEMBA-Patent-fewshot-v2 additionally removes the instruction "If no error is detected, return {"annotations": []}." For both versions, we handle invalid outputs in the same way as GEMBA-Patent-Hybrid. Finally, we compare the results of GEMBA-Patent-fewshot-Hybrid and GEMBA-Patent-fewshot-v2-Hybrid with GEMBA-Patent-Hybrid, Encoder-GEMBA-Patent, and Encoder-GEMBA-MQM. For GEMBA-Patent-Hybrid, Encoder-GEMBA-Patent, and Encoder-GEMBA-MQM, we recompute the metrics after excluding the selected claims.

Results. Table 5 shows the span-level precision, recall and F1 score, as well as category and severity accuracy over matched spans, on the training data excluding the claims used as two-shot examples. Although GEMBA-Patent-fewshot-v2-Hybrid reduces the number of claims with empty annotations, both encoder-based methods achieve the highest recall and F1 in both translation directions. GEMBA-Patent-fewshot-Hybrid achieves the highest precision in the JA–EN direction, whereas GEMBA-Patent-fewshot-v2-Hybrid achieves the highest in the EN–JA direction. GEMBA-Patent-fewshot-Hybrid achieves the highest category and severity accuracies for the JA–EN direction, whereas Encoder-GEMBA-Patent achieves the highest values for the EN–JA direction.

D Evaluation Results on the Test Set

Tables 6–7 show the evaluation results on the test set as reported by the organizers.

E Correlations of Error Annotation Scores with Human Evaluation

Table 8 shows the Pearson and Spearman correlations between human scores and the automatic scores derived from Encoder-GEMBA-MQM, Encoder-GEMBA-Patent, and GEMBA-Patent-Hybrid on the training set.

```
You are an annotator for the quality of patent-claim machine translation. Your task is to identify errors and assess the quality of the translation.
```

Figure 1: System Prompt for GEMBA-Patent

```
Evaluate the following translation:

{source_lang} source:
```text
{source_seg}
```

{target_lang} translation:
```text
{target_seg}
```

Identify and classify translation errors using only the following hierarchy. Form each error label by joining its hierarchy levels with hyphens while preserving spaces and slashes within each category name.

```

Figure 2: User Prompt for GEMBA-Patent, Part 1: Input Format and Task Summary.

| | JA–EN | | | | |
|---|---|---|---|---|---|
| **Method** | **Precision** | **Recall** | **F1** | **Category Acc.** | **Severity Acc.** |
| Encoder-GEMBA-MQM (GPT-5.2) | 0.037 | **0.027** | **0.031** | 0.429 | 0.714 |
| Encoder-GEMBA-Patent (GPT-5.2) | 0.037 | **0.027** | **0.031** | 0.286 | 0.750 |
| GEMBA-Patent-Hybrid | 0.061 | 0.012 | 0.020 | 0.250 | 0.667 |
| GEMBA-Patent-fewshot-Hybrid | **0.067** | 0.015 | 0.024 | **0.467** | **0.933** |
| GEMBA-Patent-fewshot-v2-Hybrid | 0.064 | 0.018 | 0.029 | 0.368 | 0.737 |
| | **EN–JA** | | | | |
| **Method** | **Precision** | **Recall** | **F1** | **Category Acc.** | **Severity Acc.** |
| Encoder-GEMBA-MQM (GPT-5.2) | 0.059 | **0.043** | **0.050** | 0.368 | 0.684 |
| Encoder-GEMBA-Patent (GPT-5.2) | 0.059 | **0.043** | **0.050** | **0.500** | **0.711** |
| GEMBA-Patent-Hybrid | 0.107 | 0.018 | 0.031 | 0.313 | 0.563 |
| GEMBA-Patent-fewshot-Hybrid | 0.080 | 0.018 | 0.030 | 0.250 | 0.438 |
| GEMBA-Patent-fewshot-v2-Hybrid | **0.084** | 0.024 | 0.037 | 0.333 | 0.524 |

Table 5: Error annotation results on the training data, excluding the claims used as two-shot examples.

| **Direction** | **Annotation/claim** | **Detection F1** | **Span F1** | **Severity agreement** | **Pearson's $r$** | **Spearman's $rho$** |
|---|---|---|---|---|---|---|
| JA–EN | 3.1 | 0.111 | 0.097 | 0.36 | 0.65 | 0.73 |
| EN–JA | 4.2 | 0.139 | 0.155 | 0.70 | 0.38 | 0.32 |

Table 6: Evaluation results of our submitted system on the test set.

| **Direction** | $n$ | **Pearson's $r$** | **Spearman's $\rho$** | **MAE** | **Mean (Gold/Submitted)** |
|---|---|---|---|---|---|
| JA–EN | 10 | 0.654 (p=0.0401) | 0.727 (p=0.0172) | 43.3 | 34 / 76 |
| EN–JA | 12 | 0.383 (p=0.2193) | 0.320 (p=0.31) | 28.2 | 54 / 72 |

Table 7: Correlations, MAE, and mean scores for gold MQM scores and submitted scores on the test set.

| | JA–EN | | EN–JA | |
|---|---|---|---|---|
| **Method** | **Pearson's $r$** | **Spearman's $\rho$** | **Pearson's $r$** | **Spearman's $\rho$** |
| Encoder-GEMBA-MQM (GPT-5.2) | $0.2874^{**}$ | $0.3293^{**}$ | $0.2752^{**}$ | $0.3330^{**}$ |
| Encoder-GEMBA-Patent (GPT-5.2) | $0.2992^{**}$ | $0.3562^{**}$ | $0.2040^{*}$ | $0.2689^{**}$ |
| GEMBA-Patent-Hybrid (GPT-5.2) | $\mathbf{0.4927}^{**}$ | $\mathbf{0.4591}^{**}$ | $\mathbf{0.3458}^{**}$ | $\mathbf{0.3746}^{**}$ |

$^{*}p < 0.05$, $^{**}p < 0.01$.

Table 8: Correlations between scores derived from error annotation and human quality scores on the training data.

```
- accuracy (main category)
  - addition (sub category): the translation adds technical content or claim elements not present in the source
  - omission (sub category): the translation omits an essential claim element or other source content
  - untranslated text (sub category): source text has been left untranslated
  - mistranslation (sub category): the translation incorrectly represents the inventive idea, technical function or effect, element relationship, claim scope, claim type, or other source meaning
    - numerals/symbols (sub-sub category): a quantity, numerical range, condition, numeral, or symbol is translated incorrectly
    - article (sub-sub category): an article or antecedent basis, such as "a," "the," or "said," is used incorrectly
    - incorrect dependency (sub-sub category): an element relationship, constraint-clause attachment, coordination, or modification relationship is translated incorrectly
    - unknown dependency (sub-sub category): the dependency or attachment relationship expressed in the source is not maintained and cannot be determined from the translation
    - ambiguity (sub-sub category): the translation introduces ambiguity not present in the source, particularly ambiguity affecting element relationships or claim scope

- fluency (main category)
  - punctuation (sub category): punctuation is incorrect, including improper sentence division when the patent claim should be written as one sentence
  - spelling (sub category): spelling or capitalization is incorrect
  - grammar (sub category): the translation is grammatically incorrect in the target language
  - register (sub category): the grammatical register is inappropriate for a patent claim
  - inconsistency (sub category): the translation contains an internal inconsistency unrelated to terminology
  - character encoding (sub category): characters are garbled because of incorrect encoding

- terminology (main category)
  - inappropriate for context (sub category): a term is non-standard or inappropriate for the technical or patent context
  - inconsistent use (sub category): the same technical or claim term is translated inconsistently

- style (main category)
  - awkward (sub category): the translation is excessively awkward, difficult to read, or written in narrative or explanatory prose rather than patent-claim style

- other (main category): any other translation issue
- source error (main category): an error in the source text
- non-translation (main category): the output does not constitute a translation for which distinct errors can be reliably identified

```

Figure 3: User Prompt for GEMBA-Patent, Part 2: Error Categories.

```
Classify every error as:
- major: an error that changes or obscures meaning, claim scope, an essential claim element, a technical relationship, or essential claim form
- minor: a limited error that does not materially affect meaning or claim scope

Return JSON only in the following format:

{{
  "annotations": [
    {{
      "side": "input",
      "tagged_text": "preceding context <ERROR_START>exact text span<ERROR_END> following context",
      "text": "exact text span",
      "error": "accuracy-mistranslation-article",
      "severity": "major"
    }}
  ]
}}

For every annotation:
- "side" must be either "input" or "output".
- "tagged_text" must contain the complete source or translation claim with exactly one error span enclosed by "<ERROR_START>" and "<ERROR_END>".
- If "side" is "input", create "tagged_text" from the complete source_seg.
- If "side" is "output", create "tagged_text" from the complete target_seg.
- Do not alter any part of the claim except by inserting the error tags.
- "text" must exactly equal the text enclosed by "<ERROR_START>" and "<ERROR_END>".
- "severity" must be either "major" or "minor".
- Output one annotation object per error, with exactly one tagged error in each complete claim.

If no error is detected, return {{"annotations": []}}.
```

Figure 4: User Prompt for GEMBA-Patent, Part 3: Error Severities, Output Format, and Task Request.

```
(system) You are an annotator for the quality of machine translation. Your task is to identify errors and assess the quality of the translation.

(user) {source_lang} source:
```{source_seg}```
{target_lang} translation:
```{target_seg}```

The source segment and machine translation surrounded with triple backticks contain exactly one pair of <ERROR_START> and <ERROR_END> tags. The tags enclose a pre-identified error span. Classify only the error type and severity of this span. Do not identify any additional errors.

The categories of errors are: accuracy (addition, omission, untranslated text, mistranslation (numerals/symbols, article, incorrect dependency, unknown dependency, ambiguity)), fluency (punctuation, spelling, grammar, register, inconsistency, character encoding), terminology (inappropriate for context, inconsistent use), style (awkward), other, source error, or non-translation. Join the applicable category levels with hyphens in the "error" field.

Each error is classified as one of two categories: major and minor. Major errors disrupt the flow, but what the text is trying to say is still understandable. Minor errors are technically errors, but do not disrupt the flow or hinder comprehension.


Output only the following JSON:
{{
  "annotations": [
    {{
      "text": "exact text enclosed by the tags, excluding the tags themselves",
      "error": "accuracy-mistranslation-article",
      "severity": "major"
    }}
  ]
}}
```

Figure 5: Prompt for Encoder-GEMBA-MQM

```
You are an annotator for the quality of patent-claim machine translation. Your task is to assign an error category and severity to each pre-identified error span.
```

Figure 6: System Prompt for Encoder-GEMBA-Patent

```
Evaluate the following translation:

{source_lang} source:
```{source_seg}```
{target_lang} translation:
```{target_seg}```

Text enclosed by <ERROR_START> and <ERROR_END> is a pre-identified error span. The tags are markers and are not part of the source or translation. Classify only the tagged error span. Do not identify additional errors or modify the span. A tagged source span does not necessarily indicate a source error.

Use only the following hierarchy. Form the error label by joining its hierarchy levels with hyphens while preserving spaces and slashes within each category name.

```

Figure 7: User Prompt for Encoder-GEMBA-Patent, Part 1: Input Format and Task Summary.

```
- accuracy (main category)
  - addition (sub category): the translation adds technical content or claim elements not present in the source
  - omission (sub category): the translation omits an essential claim element or other source content
  - untranslated text (sub category): source text has been left untranslated
  - mistranslation (sub category): the translation incorrectly represents the inventive idea, technical function or effect, element relationship, claim scope, claim type, or other source meaning
    - numerals/symbols (sub-sub category): a quantity, numerical range, condition, numeral, or symbol is translated incorrectly
    - article (sub-sub category): an article or antecedent basis, such as "a," "the," or "said," is used incorrectly
    - incorrect dependency (sub-sub category): an element relationship, constraint-clause attachment, coordination, or modification relationship is translated incorrectly
    - unknown dependency (sub-sub category): the dependency or attachment relationship expressed in the source is not maintained and cannot be determined from the translation
    - ambiguity (sub-sub category): the translation introduces ambiguity not present in the source, particularly ambiguity affecting element relationships or claim scope

- fluency (main category)
  - punctuation (sub category): punctuation is incorrect, including improper sentence division when the patent claim should be written as one sentence
  - spelling (sub category): spelling or capitalization is incorrect
  - grammar (sub category): the translation is grammatically incorrect in the target language
  - register (sub category): the grammatical register is inappropriate for a patent claim
  - inconsistency (sub category): the translation contains an internal inconsistency unrelated to terminology
  - character encoding (sub category): characters are garbled because of incorrect encoding

- terminology (main category)
  - inappropriate for context (sub category): a term is non-standard or inappropriate for the technical or patent context
  - inconsistent use (sub category): the same technical or claim term is translated inconsistently

- style (main category)
  - awkward (sub category): the translation is excessively awkward, difficult to read, or written in narrative or explanatory prose rather than patent-claim style

- other (main category): any other translation issue
- source error (main category): an error in the source text
- non-translation (main category): the output does not constitute a translation for which distinct errors can be reliably identified

```

Figure 8: User Prompt for Encoder-GEMBA-Patent, Part 2: Error Categories.

```
Classify every error as:
- major: an error that changes or obscures meaning, claim scope, an essential claim element, a technical relationship, or essential claim form
- minor: a limited error that does not materially affect meaning or claim scope

Return JSON only in the following format:

{{
  "annotations": [
    {{
      "text": "exact text enclosed by the tags, excluding the tags themselves",
      "error": "accuracy-mistranslation-article",
      "severity": "major"
    }}
  ]
}}

The "text" value must exactly match the text enclosed by <ERROR_START> and <ERROR_END>, excluding the tags themselves. The "error" value must be one of the defined error labels, and "severity" must be either "major" or "minor". Do not output any additional fields or text.
```

Figure 9: User Prompt for Encoder-GEMBA-Patent, Part 3: Error Severities and Output Format.

```
You are an expert evaluator of MACHINE TRANSLATION OF PATENT CLAIMS.

Your task is to evaluate the translation quality using a
rubric-based evaluation framework,
implemented explicitly as a CHECKLIST.

This is NOT general prose.
The input is a PATENT CLAIM, which has strict legal and structural constraints.

You must:
(1) Answer ALL checklist questions below using:
    - "yes" (no problem)
    - "partial" (minor or weakening problem)
    - "no" (clear or serious problem)
(2) For each question, provide a short justification.
(3) Provide a brief explanation of the evaluation result.
(4) Evaluate SOURCE vs HYPOTHESIS only; no reference is provided.

```

Figure 10: Prompt for Patent-Rubric-Checklist, Part 1: Task Setting and Evaluation Procedure.

```
--------------------
CHECKLIST (PATENT-CLAIM-SPECIFIC)
--------------------

[A] CONTENT FIDELITY (Meaning Preservation)

A1. Is the core inventive idea (what is being claimed) preserved?
A2. Are the technical functions or effects preserved without distortion?
A3. Is the translation free from hallucinated technical content not present in the
    source?

[B] ELEMENT COVERAGE (Completeness)

B1. Are all essential claim elements translated (no omissions)?
B2. Are relationships between elements (having, connected to, arranged on, etc.)
    preserved?
B3. Are quantities, ranges, or conditions (e.g. "at least one", numerical ranges)
    preserved?

[C] CLAIM SCOPE & LOGICAL CONSTRAINTS (CRITICAL)

C1. Is the claiming scope preserved?
    (e.g., "comprising" vs "consisting of", "characterized in that")
C2. Are constraint clauses correctly attached?
    (e.g., "wherein", "前記◯◯が〜")
C3. Are coordination vs modification relations preserved?
    (e.g., A and B that each have C, NOT A and (B that has C))

[D] CLAIM FORM & SYNTACTIC REQUIREMENTS

D1. Is the translation written as a SINGLE SENTENCE claim?
D2. Is the claim type preserved (apparatus / method / composition, etc.)?
D3. Is antecedent basis consistent?
    (a/the/said, 前記〜)
D4. Does the translation avoid narrative or explanatory prose style?

[E] FLUENCY & GRAMMATICALITY

E1. Is the translation grammatically correct in the target language?
E2. Is it stylistically appropriate for patent claims?
E3. Is the sentence readable and not excessively awkward or ambiguous?

```

Figure 11: Prompt for Patent-Rubric-Checklist, Part 2: Rubric and Checklist Items.

```
--------------------
INPUT FORMAT
--------------------
{source_lang} source patent claim:
"{source_seg}"

{target_lang} translated patent claim:
"{target_seg}"

--------------------
OUTPUT FORMAT (JSON ONLY)
--------------------

{{
  "checklist_answers": {{
    "A1": {{"answer": "yes", "justification": "Short justification."}},
    "A2": {{"answer": "yes", "justification": "Short justification."}},
    "A3": {{"answer": "yes", "justification": "Short justification."}},
    "B1": {{"answer": "yes", "justification": "Short justification."}},
    "B2": {{"answer": "yes", "justification": "Short justification."}},
    "B3": {{"answer": "yes", "justification": "Short justification."}},
    "C1": {{"answer": "yes", "justification": "Short justification."}},
    "C2": {{"answer": "yes", "justification": "Short justification."}},
    "C3": {{"answer": "yes", "justification": "Short justification."}},
    "D1": {{"answer": "yes", "justification": "Short justification."}},
    "D2": {{"answer": "yes", "justification": "Short justification."}},
    "D3": {{"answer": "yes", "justification": "Short justification."}},
    "D4": {{"answer": "yes", "justification": "Short justification."}},
    "E1": {{"answer": "yes", "justification": "Short justification."}},
    "E2": {{"answer": "yes", "justification": "Short justification."}},
    "E3": {{"answer": "yes", "justification": "Short justification."}}
  }},
  "overall_explanation": "Brief explanation of the evaluation result in 1-2 sentences."
}}
```

Figure 12: Prompt for Patent-Rubric-Checklist, Part 3: Input and Output Formats.