KIKIS at WMT 2025 General Translation Task

Koichi Iwakawa[1], Keito Kudo[1,2], Subaru Kimura[1], Takumi Ito[1], Jun Suzuki[1,2]
[1]Tohoku University, [2]RIKEN

Abstract

We participated in the constrained English–Japanese track of the WMT 2025 General Machine Translation Task. Our system collected the outputs produced by multiple subsystems, each of which consisted of LLM-based translation and reranking models configured differently (e.g., prompting strategies and context sizes), and reranked those outputs. Each subsystem generated multiple segment-level candidates and iteratively selected the most probable one to construct the document translation. We then reranked the document-level outputs from all subsystems to obtain the final translation. For reranking, we adopted a text-based LLM reranking approach with a reasoning model to take long contexts into account. Additionally, we built a bilingual dictionary on the fly from the parallel corpus to make the system more robust to rare words.

1 Introduction

This paper describes KIKIS's submission to the WMT 2025 General Machine Translation Shared Task (Kocmi et al., 2025a,b). We participated in the constrained track for the English–Japanese (En→Ja) direction. Given limited computational resources and the rapid pace of open-source LLM releases, we aimed to build a system that produced high-quality translations without additional training. In particular, we aimed to detect and correct residual errors, such as mismatched numbers or dates, missing key terms, and unnatural phrasing, in otherwise strong translations produced by LLM-based MT systems. To this end, we adopted a multi-stage LLM-based reranking pipeline that selected the best translation from candidate outputs. This paper provides a detailed description of our submitted system. We also report post-evaluation results that demonstrate the effectiveness of each component of our system.

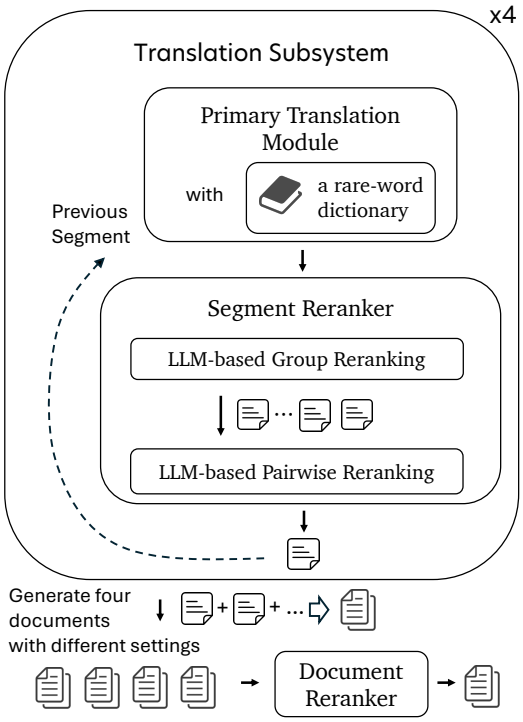

Figure 1: Overview of our final submission system.

2 System overview

Figure 1 provides an overview of our system. Our system consisted of four translation subsystems and a document reranker. We first aggregated outputs from the four subsystems, which differed in configuration (e.g., prompting strategies and context sizes). We then performed document-level reranking to select the best translation from the combined candidates. Within each subsystem, the primary translation module generated multiple segment-level hypotheses. The segment reranker iteratively filtered the candidate set via a tournament-style process to select the most plausible hypothesis. Algorithm 1 shows the pseudocode for the subsystem. Below, we describe three components: the primary translation module (§ 3), the segment reranker (§ 4), and the document reranker (§ 5).

Proceedings of the Tenth Workshop on Machine Translation (WMT), pages 705–722
November 8–9, 2025.

Algorithm 1 Pseudocode for the subsystem.

Require: $s_{1:T}$: T source segments
Require: τ: switch to Pairwise reranking when the number of candidates is $\leq \tau$
Require: g: group size for Group reranking
Require: m: context size
Require: `Partition`(H, n): sequentially partition set H into chunks of size n
function SUBSYSTEM($\{s_t\}_{t=1}^{T}$)
  for $t = 1$ **to** T **do**
    $H_t \leftarrow \mathrm{MT}(s_{t-m:t},\ h^{+}_{t-m:t-1})$ ▷ § 3
    $h^{+}_{t} \leftarrow$ RERANK($H_t,\ s_{t-m:t},\ h^{+}_{t-m:t-1}$) ▷ § 4
  end for
  return $h^{+}_{1:T}$
end function

function RERANK($H_t,\ s_{t-m:t},\ h^{+}_{t-m:t-1}$)
  while $|H_t| > 1$ **do**
    if $|H_t| > \tau$ **then**
      ▷ Group reranking mode (§ 4.1)
      $b \leftarrow g$
      $f_{\text{rerank}} \leftarrow$ GROUPRERANK
    else
      ▷ Pairwise reranking mode (§ 4.2)
      $b \leftarrow 2$
      $f_{\text{rerank}} \leftarrow$ PAIRWISERERANK
    end if
    $H_{\text{next}} \leftarrow [\,]$
    for B **in** PARTITION($H_t,\ b$) **do**
      $\hat{h}_t \leftarrow f_{\text{rerank}}(B,\ s_{t-m:t},\ h^{+}_{t-m:t-1})$
      $H_{\text{next}} \leftarrow H_{\text{next}} \,\|\, [\hat{h}_t]$
    end for
    $H_t \leftarrow H_{\text{next}}$
  end while
  return $H_t[1]$
end function

3 Primary translation module

We used plamo-2-translate (Imajo et al., 2025) as our base model. Plamo-2-translate is an LLM-based translation system with a hybrid architecture that combined Mamba (Gu and Dao, 2024) and Transformers (Vaswani et al., 2017). With this model, we generated 32 hypotheses for each source segment. The decoding hyperparameters are listed in Table 5. To further improve accuracy and naturalness at the document level, we combined three prompting strategies: vocabulary prompting, style prompting, and context prompting. We describe each strategy below.

3.1 Vocabulary prompting

To improve robustness to rare words, we dynamically constructed a bilingual dictionary from parallel corpora and used it as a prompt for the base model.

The dictionary was built in four steps:

- **Term extraction**: Candidate terms (e.g., named entities) were extracted from source sentences using Qwen3-8B (Qwen Team, 2025).
- **Retrieval**: Sentence pairs were retrieved from the parallel corpora whose source side contained the extracted term.
- **Translation-pair extraction**: Qwen3-8B was used to identify the translation of each term in the target sentence, and the term-level pairs were recorded.
- **Cleaning**: Pairs that were likely to be incorrectly extracted were discarded.

When a source segment in the test set contained any extracted terms, the corresponding dictionary entries were included in the prompt to the base model. In total, the dictionary comprised 365 English entries with an average of 1.9 Japanese translations per entry. See appendix C for further details (e.g., list of parallel corpora and filtering criteria).

3.2 Context prompting

To maintain document-level consistency of named entities and overall style, we translated each segment with the preceding context. For the current source segment s_t, the prompt to the base model included the m source segments $s_{t-m:t-1}$ and the previously selected hypotheses $h^{+}_{t-m:t-1}$ from the segment reranker (§ 4). During decoding, we enforced $h^{+}_{t-m:t-1}$ via forced-decoding and then generated the output for s_t.

Formally, context-prompted decoding is defined as:

$$H_t = \left\{ h_t^{(k)} \sim P_\theta\left(\cdot \mid s_{t-m:t}, h^{+}_{t-m:t-1}\right) \right\}_{k=1}^{K}, \quad (1)$$

where H_t is the set of K sampled hypotheses for the current segment, $h_t^{(k)}$ is the k-th sample, and $P_\theta(\cdot \mid s_{t-m:t}, h^{+}_{t-m:t-1})$ denotes the base model's conditional distribution.

3.3 Style prompting

We controlled the translation style based on the domain of the source document. Specifically, we prompted the base model to produce either the polite (“です/ます”) or the plain (“だ/である”) style based on the document domain. We enforced the plain style for literary and news texts and left the style unspecified for social and speech texts. We further included domain-specific instructions to keep the writing appropriate for each domain. [1]

4 Segment Reranker

This module selected plausible hypotheses from the segment-level outputs of the primary translation module via a tournament-style process. We applied two reranking stages with different granularities in sequence. In the early stage, we grouped candidates into batches of at least three and performed coarse filtering within each batch (GROUPRANK in Algorithm 1). After reducing the pool, we performed pairwise comparisons among the remaining candidates to select the final hypothesis (PAIRWISECOMPARE in Algorithm 1). Inspired by (Sun et al., 2023), we adopted a text-based LLM reranking approach using the reasoning model Qwen3-8B (Qwen Team, 2025).

4.1 Group reranking

This module selected a plausible hypothesis from a set of candidate translations. Its role was to roughly filter out low-quality outputs and narrow the candidate set. Concretely, the model received the m previous source segments and the current one, denoted $s_{t-m:t}$, the contexts of confirmed hypotheses from previous iterations $h^{+}_{t-m:t-1}$, and the current subset of candidate hypotheses generated by the primary translation module $H'_t \subseteq H_t$. From these inputs, it selected a plausible hypothesis $\hat{h}_t \in Ht'$. Formally,

$$\hat{h}_t = \mathrm{LLM_g}\left(s_{t-m:t}, h^{+}_{t-m:t-1}, H'_t\right), \qquad (2)$$

where $\mathrm{LLM_g}$ was the LLM instructed to perform group reranking, which returned one hypothesis from H'_t. The prompt template is given in appendix D.

[1] Due to terms and conditions, we cannot include the exact prompt format here. For details about the prompts, please refer to https://translate-demo.plamo.preferredai.jp/contact.

4.2 Pairwise reranking

Given the hypotheses returned by group reranking, this module performed pairwise comparisons to select the most plausible translation. As with group reranking, we used an LLM for hypothesis selection; however, here the LLM compared pairs of hypotheses. To mitigate positional bias (Liu et al., 2024; Wang et al., 2024), we prompted the LLM with each pair in both orders. If the two decisions conflicted, we selected one of the two hypotheses uniformly at random.

Formally, we expressed this as:

$$\hat{h}_t = \arg\max_{h \in H'_t} \sum_{(u,v)\in \mathrm{Perm}_2(H'_t)} \mathbf{1}\left\{\mathrm{LLM_p}\left(s_{t-m:t}, h^{+}_{t-m:t-1}, u, v\right) = h\right\}, \qquad (3)$$

Here, $\mathrm{Perm}_2(H'_t)$ denoted the set of all ordered pairs of distinct elements from H'_t. $\mathrm{LLM_p}$ was the LLM instructed to perform pairwise reranking; it returned one of the two given hypotheses, u or v. $\mathbf{1}$ denoted the indicator function. The prompt template is provided in appendix D.

5 Document reranker

We reranked document-level translation candidates generated by N_{sub} subsystems using Qwen3-8B. We performed pairwise comparisons over all ordered pairs of candidates and selected the most plausible translation. As in the segment reranking, each pair was evaluated in both input orders to deal with positional bias.

We let the set of document-level hypotheses be $\mathcal{D}_{\mathrm{hyp}} = \{D^{(1)}_{\mathrm{hyp}}, \ldots, D^{(N_{\mathrm{sub}})}_{\mathrm{hyp}}\}$, where each $D^{(i)}_{\mathrm{hyp}}$ was a complete translated document produced by a subsystem. Let the source document be $D_{\mathrm{src}} = s_{:}$ (where : denoted all source segments). We selected the final document $\hat{D}_{\mathrm{hyp}}$ by counting pairwise wins:

$$\hat{D}_{\mathrm{hyp}} = \arg\max_{d \in \mathcal{D}_{\mathrm{hyp}}} \sum_{(u,v)\in \mathrm{Perm}_2(\mathcal{D}_{\mathrm{hyp}})} \mathbf{1}\{\mathrm{LLM_d}(D_{\mathrm{src}},\, u,\, v) = d\}, \qquad (4)$$

where $\mathrm{Perm}_2(\mathcal{D}_{\mathrm{hyp}})$ denoted all ordered pairs (u, v) with $u \neq v$, and $\mathrm{LLM_d}$ was the model for document-level reranking that returned one of the two inputs, u or v. The prompt template is given in appendix D.

	Vocab prompt	Context size	Group size	Document reranking	MetricX↓ w/ ref	MetricX↓ w/o ref	XCOMET↑ w/ ref	XCOMET↑ w/o ref
Primary translation module only								
(a)		0			5.62	6.02	0.522	0.502
(b)	✓	0			5.60	6.03	0.520	0.500
(c)		2			5.59	6.01	0.528	0.498
Subsystems								
(d)		2	4		5.50	5.90	0.547	0.519
(e)	✓	2	4		5.40	5.82	0.552	0.521
(f)	✓	4	4		5.58	5.95	0.542	0.508
(g)	✓	2	8		5.49	5.86	0.548	0.520
Final submission system								
(h)	✓	2-4	4-8	✓	5.51	5.92	0.551	0.517

Table 1: Post evaluation results.

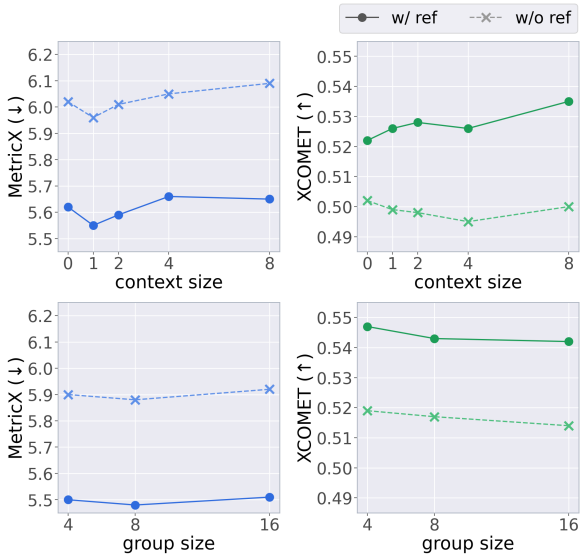

Figure 2: Automatic evaluation scores with varying context sizes (top row) and group sizes (bottom row). The left column shows MetricX results (lower is better) and the right column shows XCOMET results (higher is better). Solid lines indicate reference-based (w/ ref) evaluation, and dashed lines indicate reference-free (w/o ref) evaluation.

6 Post-evaluation

We conducted a post-evaluation to assess the contribution of each component in our system.

6.1 Experimental setup

We reported the performance of our final submission. The final system consisted of four subsystems, and we also reported the performance of each subsystem (before applying document-level reranking). As ablation studies, we evaluated the effects of removing the segment reranker, enabling or disabling vocabulary prompting, and varying the context size (m in § 3.2) and the group size (g in Algorithm 1). In the no-reranking setting (using only the primary translation module), we generated 32 translation candidates for each source segment (as in Section 3) and selected the highest-probability candidate.

We used XCOMET-XL (Guerreiro et al., 2024)[2] and MetricX-24 (XL) (Juraska et al., 2024)[3] as automatic evaluation metrics. We evaluated our approach in both reference-based and reference-free (quality estimation) settings.

6.2 Results and discussion

Table 1 shows the post-evaluation results.

Effect of reranking. Comparing configurations (c) and (d) in Table 1, we observed that segment-level reranking (§ 4) improved performance, suggesting that the LLM selects better translations. By contrast, in our experimental setting, document-level reranking (h)(§ 5) did not surpass configuration (d), which was the best-performing subsystem before reranking. This may be partly due to the much longer input at the document level, which could make the reranking task more challenging for the LLM.

Effect of vocabulary prompting. In Table 1, the comparisons between (a) and (b) and between (d) and (e) showed no consistent effect of vocabulary prompting on evaluation metrics. Qualitatively, however, as shown in Table 2, vocabulary prompting improved translations of domain-specific terms; for example, "facial scrub" is translated correctly.

Effect of context sizes. Figure 2 shows the relationship between context size and performance. According to the automatic evaluation metrics, we

[2] https://huggingface.co/Unbabel/XCOMET-XL
[3] https://huggingface.co/google/metricx-24-hybrid-xl-v2p6

Source	you're gonna cleanse get rid of all the grease and makeup from your face and then you're gonna use your **facial scrub**
Reference	肌の上に。さて、こちらがその**フェイシャルスクラブ**です。このスクラブを使う際に、まず最初に忘れずに顔をクレンジングします。それが第一のステップです。
Translation	
(d) w/o vocab	ここで重要なのは、フェイススクラブを使用する前には必ず洗顔を行うことです。
(e) w/ vocab	まず顔の皮脂やメイクを完全に落とします。その後、このフェイシャルスクラブを使います。

Table 2: Qualitative output evaluation: vocabulary usage ((d) vs. (e)). Incorporating a predefined vocabulary list (e) ensured that the translation matched the reference term "フェイシャルスクラブ", in contrast to the variant "フェイススクラブ" produced without the vocabulary (d).

Source	Segment t:	First job is taking out the floor ⋯ cover that in linoleum. The job is shoddy ⋯ I dunno how to do all that. The roof will remain cold and metal ⋯
	Segment $t+1$:	This chair works very well in the van ⋯ should ideally be fastened better ⋯ flat as a bed.
Reference	Segment t:	「床を外して下の収納スペースにアクセスし、そこにリノリウムを貼ること。…正直言って雑な仕事。…そんな技術はまったくない。…屋根は金属むき出しで冷たいまま。」
	Segment $t+1$:	「床にちゃんと固定できてはいないけど… フラットなベッドになる。」
(a) Context size = 0	Segment t:	「床板を取り外し…敷くことだ。…この作業は粗雑な仕上がりになってしまう。…方法がわからない。…屋根は金属のまま放置することになる。」
	Segment $t+1$:	「椅子は理想的には固定すべきですが…ベッドとしても使える状態になります。」
(c) Context size = 2	Segment t:	「床を撤去し…敷くことです。…この作業は雑な仕上がりです。…方法が分かりません。…屋根は金属のままとなります。」
	Segment $t+1$:	「椅子は固定方法を改善すべきですが…ベッドとしても使用可能です。」

Table 3: Qualitative evaluation of outputs: effect of context size ((a) vs. (c)). (a) Without previously translated hypotheses as context, translations mix polite ("です/ます") and plain ("だ/である") sentence endings across adjacent segments. (c) With previously translated hypotheses provided as context, the sentence-ending style remains consistent, avoiding style shifts between segments.

did not observe a consistent trend in performance as the context size changed. Qualitatively, however, as shown in Table 3, adding previously translated hypotheses as in-document context helped keep a consistent style across the document, either polite ("です/ます") or plain ("だ/である").

Effect of group size. Figure 2 shows how group size affected segment-level reranking. Intuitively, larger groups made it harder to select the best candidate. At the same time, they allowed us to prune more candidates per group, which sped up the system. Across the tested group sizes, XCOMET and MetricX scores dropped by less than 0.05 points. Therefore, there was room to either speed up the final submission system or use the saved time to generate more translation candidates from the primary translation module within the same time budget.

7 Conclusion

We described the KIKIS submission to the WMT 2025 General Machine Translation Shared Task. We participated in the constrained track for the English–Japanese (En→Ja) direction. Our system consisted of four translation subsystems and a document reranker. Each subsystem combined an MT model with an LLM-based segment reranker. We aggregated the outputs from the four subsystems and then applied document-level reranking to select the final translation.

Acknowledgments

We would like to thank the members of the Tohoku NLP Group for their cooperation and feedback throughout the course of this research. This work was supported by Moonshot R&D Grant Number JPMJMS2011-35 (fundamental research), JST BOOST Japan Grant Number JPMJBS2421 and

JSPS KAKENHI Grant Number JP25KJ0615.

Contributions

Koichi Iwakawa conducted hyperparameter search for LLM-based translation models and performed post-evaluations.
Keito Kudo built the foundation of the LLM-based translation system, developed the reranking system, and constructed the bilingual dictionary.
Subaru Kimura was responsible for decoding the test set samples using the submission system. He also investigated translation quality under different prompting strategies.
Takumi Ito contributed to strategic discussions for the WMT submission and provided feedback and advice on translation outputs.
Jun Suzuki supervised and coordinated the entire project.

References

Alfred V. Aho and Margaret J. Corasick. 1975. Efficient string matching: an aid to bibliographic search. *Commun. ACM*, 18(6):333–340.

Inc. Baobab. 2024. baobab_coco_evaluate_caption_24. Initial commit on 2024-11-25; updates on 2024-12-19 and 2024-12-21.

Valeriia Bolotova-Baranova, Vladislav Blinov, Sofya Filippova, Falk Scholer, and Mark Sanderson. 2023. WikiHowQA: A comprehensive benchmark for multi-document non-factoid question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5291–5314, Toronto, Canada. Association for Computational Linguistics.

Bianka Buschbeck and Miriam Exel. 2020. A parallel evaluation data set of software documentation with document structure annotation. In *Proceedings of the 7th Workshop on Asian Translation*, pages 160–169, Suzhou, China. Association for Computational Linguistics.

Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Jan Niehues, Sebastian Stüker, Katsuhito Sudoh, Koichiro Yoshino, and Christian Federmann. 2017. Overview of the IWSLT 2017 evaluation campaign. In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 2–14, Tokyo, Japan. International Workshop on Spoken Language Translation.

Simone Conia, Daniel Lee, Min Li, Umar Farooq Minhas, Saloni Potdar, and Yunyao Li. 2024. Towards cross-cultural machine translation with retrieval-augmented generation from multilingual knowledge graphs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16343–16360, Miami, Florida, USA. Association for Computational Linguistics.

Ona de Gibert, Graeme Nail, Nikolay Arefyev, Marta Bañón, Jelmer van der Linde, Shaoxiong Ji, Jaume Zaragoza-Bernabeu, Mikko Aulamo, Gema Ramírez-Sánchez, Andrey Kutuzov, Sampo Pyysalo, Stephan Oepen, and Jörg Tiedemann. 2024. A new massive multilingual dataset for high-performance language technologies. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1116–1128, Torino, Italia. ELRA and ICCL.

Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. Ccaligned: A massive collection of cross-lingual web-document pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5960–5969, Online. Association for Computational Linguistics.

Ahmed El-Kishky, Adithya Renduchintala, James Cross, Francisco Guzmán, and Philipp Koehn. 2021. Xlent: Mining a large cross-lingual entity dataset with lexical-semantic-phonetic word alignment. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10424–10430, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Vincent J. Felitti, Robert F. Anda, Dale Nordenberg, David F. Williamson, Alison M. Spitz, Valerie Edwards, Mary P. Koss, and James S. Marks. 1998. Relationship of childhood abuse and household dysfunction to many of the leading causes of death in adults: The adverse childhood experiences (ace) study. *American Journal of Preventive Medicine*, 14(4):245–258. The original Adverse Childhood Experiences Study.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2023. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada. Association for Computational Linguistics.

GENIAC Team Ozaki. 2024. Wikihownfqa-ja_cleaned. Created on 2024-05-10; license CC BY 4.0.

Albert Gu and Tri Dao. 2024. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*.

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.

Kazuma Hashimoto, Raffaella Buschiazzo, James Bradbury, Teresa Marshall, Richard Socher, and Caiming Xiong. 2019. A high-quality multilingual dataset for structured documentation translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 116–127, Florence, Italy. Association for Computational Linguistics.

Yuta Hayashibe. 2023. megagonlabs/instruction_ja. https://github.com/megagonlabs/instruction_ja. GitHub repository.

Kentaro Imajo, Masanori Hiano, Kento Nozawa, and Chubachi Kaizabro. 2025. Plamo translate: Development of a large language model specialized for translation (original japanese title: "PLaMo Translate: 翻訳特化大規模言語モデルの開発"). Technical report, Preferred Networks, Inc.

Tatsuya Ishisaka, Masao Utiyama, Eiichiro Sumita, and Kazuhide Yamamoto. 2009. Building a large scale japanese-english open source parallel corpus. *IPSJ SIG Technical Report*, 2009(1):1–6. Also appears in SLP Vol. 2009-SLP-76.

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.

Katsuhiko Toyama. 2009. Japanese law translation project. https://www.kl.i.is.nagoya-u.ac.jp/told/index.html. Accessed: 2025-08-10.

Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakougna, Jessica M. Lundin, Christof Monz, Kenton Murray, and 10 others. 2025a. Findings of the wmt25 general machine translation shared task: Time to stop evaluating on easy test sets. In *Proceedings of the Tenth Conference on Machine Translation*, China. Association for Computational Linguistics.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Natalia Fedorova, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakougna, Jessica Lundin, Kenton Murray, Masaaki Nagata, and 9 others. 2025b. Preliminary ranking of wmt25 general machine translation systems.

Keito Kudo, Hiroyuki Deguchi, Makoto Morishita, Ryo Fujii, Takumi Ito, Shintaro Ozaki, Koki Natsumi, Kai Sato, Kazuki Yano, Ryosuke Takahashi, Subaru Kimura, Tomomasa Hara, Yusuke Sakai, and Jun Suzuki. 2024. Document-level translation with LLM reranking: Team-J at WMT 2024 general translation task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 210–226, Miami, Florida, USA. Association for Computational Linguistics.

Kurohashi-Kawahara Lab. and NICT. 2011. JEC Basic Sentence Data.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Yunmeng Li, Jun Suzuki, Makoto Morishita, Kaori Abe, Ryoko Tokuhisa, Ana Brassard, and Kentaro Inui. 2022. Chat translation error detection for assisting cross-lingual communications. In *Proceedings of the 3rd Workshop on Evaluation and Comparison of NLP Systems*, pages 88–95, Online. Association for Computational Linguistics.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Andrew Merritt, Chenhui Chu, and Yuki Arase. 2020. A corpus for english-japanese multimodal neural machine translation with comparable sentences. *CoRR*, abs/2010.08725.

Mitsua. 2024. Wikidata parallel descriptions (en–ja). Initial commit on 2024-05-13; updated 2024-05-17. Generated from Wikidata dump 2024-05-06; 1,570,685 rows.

Rei Miyata. 2024. Mtpedocs. https://github.com/tntc-project/MTPEdocs. Tntc-project GitHub repository.

Mozilla Contributors. 2005–2025. Mdn web docs. Online. Web platform documentation and learning resource.

Atsushi Nakajima. 2022. fungi_indexed_mycological_papers_japanese. Compiled summaries, tags, reported and compared species from the Daikinrin website.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016. ASPEC: Asian

scientific paper excerpt corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2204–2208, Portorož, Slovenia. European Language Resources Association (ELRA).

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, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Hiroki Nomoto, Kenji Okano, David Moeljadi, and Hideo Sawada. 2018. Tufs asian language parallel corpus (talpco). In *Proceedings of the Twenty-Fourth Annual Meeting of the Association for Natural Language Processing*, pages 436–439, Okayama, Japan. Association for Natural Language Processing. Japanese-based parallel corpus for Burmese, Malay, Indonesian, and English.

Qwen Team. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Matīss Rikters, Ryokan Ri, Tong Li, and Toshiaki Nakazawa. 2019. Designing the business conversation corpus. In *Proceedings of the 6th Workshop on Asian Translation*, pages 54–61, Hong Kong, China. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021. Ccmatrix: Mining billions of high-quality parallel sentences on the web. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6490–6500, Online. Association for Computational Linguistics.

Haiyue Song, Raj Dabre, Atsushi Fujita, and Sadao Kurohashi. 2020. Coursera corpus mining and multistage fine-tuning for improving lectures translation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3640–3649, Marseille, France. European Language Resources Association.

Robyn Speer. 2022. rspeer/wordfreq: v3.0.

Hiroaki Sugiyama, Masahiro Mizukami, Tsunehiro Arimoto, Hiromi Narimatsu, Yuya Chiba, Hideharu Nakajima, and Toyomi Meguro. 2023. Empirical analysis of training strategies of transformer-based japanese chit-chat systems. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 685–691.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937, Singapore. Association for Computational Linguistics.

Yasuhito Tanaka, Jim Breen, and Paul Blay. 2001. Tanaka corpus. Originally compiled at Hyogo University, now part of Tatoeba Project. Japanese-English parallel sentence corpus, maintained by Tatoeba Project.

Tatoeba Community. 2006–2025. Tatoeba: Collection of sentences and translations. Online collaborative platform. Community-driven parallel corpus with over 12.6M sentences in 426 languages.

Ye Kyaw Thu, Win Pa Pa, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Introducing the Asian language treebank (ALT). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1574–1578, Portorož, Slovenia. European Language Resources Association (ELRA).

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA). EUbookshop is part of the OPUS collection.

TLDR Pages Community. 2013–2025. Tldr pages: Collaborative cheatsheets for console commands. GitHub repository. Community-maintained help pages for command-line tools with 56,000+ GitHub stars.

Hayato Tsukagoshi. 2024. hpprc/honyaku. Repository owner: hpprc; dataset card indicates CC BY-SA 4.0.

Masao Utiyama. 2019. Paranatcom — parallel english-japanese abstract corpus made from nature communications articles.

Masao Utiyama and Mayumi Takahashi. 2023. English-japanese translation alignment data. Page last updated on 2016-08-25; dataset originally released in 2003.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ziqi Wang, Hanlin Zhang, Xiner Li, Kuan-Hao Huang, Chi Han, Shuiwang Ji, Sham M. Kakade, Hao Peng, and Heng Ji. 2024. Eliminating position bias of language models: A mechanistic approach. In *NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning*.

Hitomi Yanaka and Koji Mineshima. 2022. Compositional evaluation on Japanese textual entailment and similarity. *Transactions of the Association for Computational Linguistics*, 10:1266–1284.

	Parameters
plamo-2-translate	9.5 B
Qwen3-8B	8.2 B
Total	17.7 B

Table 4: Model parameter sizes.

plamo-2-translate	
temperature	0.8
top-p	0.9
Qwen3-8B	
temperature	0.6
top-p	0.95
top-k	20
min-p	0.0
Output format	structured outputs (json)

Table 5: Decoding hyperparameters.

A Decoding hyperparameters

Table 5 lists the decoding hyperparameters used in our system. For Qwen3-8B decoding, we adopted the officially recommended settings[4]. We used vllm (Kwon et al., 2023) to decode translation candidates and to run the reranking steps.

B Parameter count

Table 4 shows the parameter counts of the models used in our translation system. Our system satisfied the constrained track limit of at most 20B total parameters. We did not fine-tune any model and used the publicly available model parameters as provided.

C Detail of bilingual dictionary construction

This section described the detailed procedure for constructing the bilingual dictionary described in § 3.1.

Term extraction. We first extracted named entities and technical terms from the source sentences in the test set. We used Qwen3-8B (Qwen Team, 2025) to perform term extraction. The prompt we gave to Qwen3-8B for term extraction is shown in List D. After extraction, we filtered out terms that met any of the following conditions:

- The term fell within the top 66% by frequency within the test set.
- The term was tokenized as a single token by the plamo-2-translate tokenizer.
- The term was included in the top 100,000 words of the English word frequency list from `wordfreq` (Speer, 2022).

We applied this filtering because terms that satisfied these conditions were frequent and were expected to be translated correctly by the base model without explicit vocabulary prompts.

Retrieval. We retrieved all parallel sentence pairs from the corpora whose source side contained any of the extracted terms. We used the Aho–Corasick algorithm (Aho and Corasick, 1975) for efficient multi-pattern matching. Table 6 lists the parallel corpora used as sources for the bilingual dictionary. For several corpora, we computed LaBSE (Feng et al., 2022) sentence embeddings and filtered out sentence pairs whose semantic similarity was outside the range [0.7, 0.96]. Pairs with LaBSE similarity below 0.7 were removed due to likely low semantic alignment between source and target. Pairs with LaBSE similarity above 0.96 were removed because they were likely to be nearly identical (copying) or noisy. The Team-J (WMT2024) bitext dataset and development dataset, which we used in last year's submission and which was built from open data sources, was included and was filtered with the same LaBSE-based criterion. See Kudo et al. (2024) for more details.

Translation-pair extraction. We then used Qwen3-8B to extract candidate target terms from the retrieved parallel sentences. The prompt given to the model was shown in List D. This process yielded multiple candidate target-term patterns for each source term.

Cleaning. Some extracted target terms were noisy or incorrect due to model errors. Therefore, for each source term, we retained only the most frequently extracted target terms. Concretely, we ranked the extracted target terms by occurrence frequency and kept the top 30This frequency-based filtering reduced noise and favored stable translations that appeared repeatedly in the parallel data.

[4] https://huggingface.co/Qwen/Qwen3-8B#best-practices

	Filtering	Size
Team-J WMT2024 bitext dataset (Kudo et al., 2024)	✓	22,899,294
Team-J WMT2024 development dataset (Kudo et al., 2024)	✓	18,113
BSD (Rikters et al., 2019)		808
BPersona-chat (Li et al., 2022; Sugiyama et al., 2023)		2,940
MTPEdocs (Miyata, 2024)		1045
CourseraParallelCorpusMining (Song et al., 2020)		53,166
Flickr30kEnt-JP		155,070
JSICK (Yanaka and Mineshima, 2022)		18,854
IWSLT2017 (Cettolo et al., 2017)		9,340
Software Documentation Data Set for Machine Translation (Buschbeck and Exel, 2020)	✓	7,745
localization-xml-mt (Hashimoto et al., 2019)	✓	82,546
Asian Language Treebank Parallel Corpus (Thu et al., 2016)		20,101
ParaNatCom (Utiyama, 2019)		507
JEC Basic Sentence Data (Kurohashi-Kawahara Lab. and NICT, 2011)		4,769
Japanese Law Translation (Katsuhiko Toyama, 2009)		75,930
English–Japanese Translation Alignment Data (Utiyama and Takahashi, 2023)		42,738
Tanaka Corpus (Tanaka et al., 2001)		147,865
honyaku (Tsukagoshi, 2024)		33
TALPCo (Nomoto et al., 2018)		1,372
WikiHowNFQA-ja-en (Bolotova-Baranova et al., 2023; GENIAC Team Ozaki, 2024)		9,584
fungi_indexed_mycological_papers_japanese (Nakajima, 2022)		12,744
baobab_coco_evaluate_caption_24 (Baobab, 2024)		50
ACES (Felitti et al., 1998)		430
MASSIVE (FitzGerald et al., 2023)		11,514
ea-mt-benchmark (Conia et al., 2024)		5,108
JaEnCOCO (Merritt et al., 2020)		461
instruction_ja (Hayashibe, 2023)		669
ASPEC (Nakazawa et al., 2016)	✓	1,465,019
Large Scale Japanese-English Open Source Parallel Corpus (Ishisaka et al., 2009)	✓	180,709
CCAligned (El-Kishky et al., 2020)	✓	11,544,406
CCMatrix (Schwenk et al., 2021)	✓	28,827,886
EUbookshop (Tiedemann, 2012)	✓	81
GNOME (Tiedemann, 2012)	✓	37
HPLT (de Gibert et al., 2024)	✓	14,759,898
KDE4 (Tiedemann, 2012)	✓	81,316
MDN Web Docs (Mozilla Contributors, 2005–2025)	✓	65,621
NLLB (NLLB Team et al., 2022)	✓	28,827,883
PHP (Tiedemann, 2012)	✓	3,109
QED (Tiedemann, 2012)	✓	24,289
Tanzil (Tiedemann, 2012)	✓	53,202
Tatoeba (Tatoeba Community, 2006–2025)	✓	189,386
XLEnt (El-Kishky et al., 2021)	✓	2,577,352
tldr-pages (TLDR Pages Community, 2013–2025)	✓	720
Wikidata parallel descriptions (Mitsua, 2024)	✓	860,742
Total		113,044,452

Table 6: A list of the parallel corpus used for bilingual dictionary construction. Entries with a ✓ in the filtering column indicate that LaBSE-based data filtering was applied; the Size column shows the number of sentence pairs.

D Prompt lists

Term extraction. The prompt template used for term extraction described in § 3.1 was as follows. The source segment from the test set was embedded in <|INPUT_TEXT|>.

```
# Named Entity Recognition and
Technical Term Extraction Instructions

Extract named entities and technical
terms from the following text.

## Processing Steps

1. **Named Entity Extraction**
   Extract expressions that denote
   specific entities belonging to the
   following categories:

   - **PERSON**: Individual names,
   fictional character names
   - **ORGANIZATION**: Companies,
   government agencies, organizations,
   teams, etc.
```

- **LOCATION**: Countries, cities, regions, buildings, natural features, etc.
- **PRODUCT**: Products, services, software, titles of works, etc.
- **EVENT**: Conferences, festivals, competitions, historical events, etc.
- **MISC**: Other specific entities not classified above

Important:
- Exclude commonly known entities that appear frequently in general texts (e.g., "United States", "Japan", "Google", "Microsoft", "China", "Tokyo", etc.)
- Focus on extracting entities that are specific and distinctive to the given text
- **Prioritize entities that may pose translation challenges**, such as:
  * Local or regional entities with cultural significance
  * Organizations with acronyms or abbreviations
  * Location names with specific cultural or historical context
  * Products with brand-specific terminology

※ Numbers and dates should not be included in the extraction

2. **Technical Term Extraction**
Extract domain-specific terminology including:

- **TECHNICAL**: Scientific, medical, engineering, IT, legal, financial, and other field-specific terminology
- **CONCEPT**: Abstract concepts, theories, methodologies, principles specific to certain domains
- **PROCESS**: Specialized procedures, techniques, or methods used in specific fields

Important:
- Exclude commonly used technical terms that are widely known (e.g., "computer", "internet", "software", "database", "algorithm" in IT contexts)
- Focus on specialized or domain-specific terms that provide unique insights
- **Prioritize terms that are challenging for translation**, including:
  * Field-specific jargon with no standard translation
  * Compound terms or phrases unique to the domain
  * Terms requiring deep domain knowledge for accurate translation
  * Newly coined terms or emerging concepts
  * Terms with specific meanings that differ from general usage

3. **Extraction Priority**
Apply the following filtering process for both named entities and technical terms:

1. First, identify all potential entities and terms
2. Filter out generally common/well-known items
3. Keep only distinctive and informative items
4. **Translation Difficulty Priority**: Prioritize entities and terms that are likely to be challenging for translation:
   - Culture-specific concepts with no direct equivalent in other languages
   - Domain-specific terminology requiring specialized knowledge
   - Acronyms, abbreviations, and neologisms
   - Context-dependent expressions
   - Terms with multiple meanings that require disambiguation
   - Compound technical terms unique to specific fields

Output Format

Please provide the output in the following JSON format:

```json
{
  "named_entities": {
    "person": ["List of extracted person names"],
    "organization": ["List of extracted organization names"],
    "location": ["List of extracted location names"],
    "product": ["List of extracted products/services"],
    "event": ["List of extracted events"],
    "misc": ["List of other named entities"]
  },
  "technical_terms": {
    "technical": ["List of field-specific terminology"],
    "concept": ["List of abstract concepts and theories"],
    "process": ["List of specialized procedures and methods"]
  }
}
```

Example

Input example
```
Yesterday, Dr. John Smith from Tohoku University won first place in the WMT2025 machine translation competition held in Suzhou, China. His team's neural translation system outperformed submissions from Meta AI, DeepMind, and Microsoft Research with a BLEU score of 45.7. The competition focused on translation tasks covering more than 10 languages, including low-resource language pairs. The transformer architecture with attention mechanisms proved crucial for handling morphologically complex languages.
```

Output example
```json
{
  "named_entities": {
    "person": ["John Smith"],
    "organization": ["Tohoku University", "Meta AI", "DeepMind", "Microsoft Research"],
    "location": ["Suzhou"],
    "product": [],
    "event": ["WMT2025 machine translation competition"],
    "misc": []
  },
  "technical_terms": {
    "technical": ["neural translation system", "BLEU score", "transformer architecture", "attention mechanisms"],
    "concept": ["low-resource language pairs", "morphologically complex languages"],
    "process": []
  }
}
```
※ Note: "China" and common terms like "machine translation" and "translation tasks" are excluded from the extraction as they are generally well-known

Notes

- Include named entities and technical terms in the list without duplication
- If no items are found for a specific category, output an empty list for that category
- Distinguish between named entities (specific instances) and technical terms (domain concepts)
- A term can be both a product name and a technical term depending on context
- **Exclude entities and terms that are commonly known or frequently used in general discourse** (e.g., major countries, well-known companies, basic technical terms)
- Focus on extracting distinctive, informative, and document-specific entities and terms
- **From a translation perspective, prioritize extraction of:**

```
  * Terms requiring cultural or
  contextual knowledge
  * Domain-specific expressions without
  established translations
  * Ambiguous terms needing
  disambiguation
  * Entities with specific
  local/regional significance

---


## Input
```
<|INPUT_TEXT|>
```
```

Translation-pair extraction. The prompt template used for translation-pair extraction described in appendix C was as follows. The source sentence selected from the parallel corpus was embedded in <<<|SOURCE_TEXT|>>>, and the target sentence selected from the parallel corpus was embedded in <<<|TARGET_TEXT|>>>. Additionally, the term contained in the source sentence that was extracted during the term extraction phase was embedded in <<<|TERM|>>>.

```
# Translation Term Extraction Task

## Task Overview
Extract the corresponding translation
for a specified term from given source
text (original) and target text
(translated) pairs.

## Input Format
The following three elements will be
provided:
- **source text**: The original text
- **target text**: The translated text
- **term**: A specific term contained
in the source text

## Processing Steps
1. Identify the specified term within
the source text
2. Analyze how that term is translated
in the target text
3. Extract the corresponding
translation **exactly as it appears**
from the target text
4. When extracting, use the character
string that actually appears in the
target text without any modifications

## Important Notes
- The source text and target text may
not necessarily have a perfect
translation relationship
- The translation corresponding to the
specified term may not exist in the
target text
- When extracting translations, do not
add speculation or corrections; use
only character strings that actually
exist in the target text
- If no corresponding translation is
found, or if it is determined that no
translation relationship exists,
output null

## Output Format
Output in the following JSON format:

```json
{
  "term": "input term",
  "term_translation": "extracted
  translation" or null
}
```

Examples

Example 1 (Normal extraction)
Input:
- source text: "The artificial
intelligence system processes data
efficiently."
- target text: "その人工知能システムは
データを効率的に処理します。"
- term: "artificial intelligence"

Output:
```json
{
  "term": "artificial intelligence",
  "term_translation": "人工知能"
}
```

Example 2 (Translation not found)
```

````
**Input:**
- source text: "The quantum computer solved the complex problem."
- target text: "その高性能コンピュータは難しい問題を解決した。"
- term: "quantum"

**Output:**
```json
{
  "term": "quantum",
  "term_translation": null
}
```

### Example 3 (Unclear translation relationship)
**Input:**
- source text: "The new policy will be implemented next month."
- target text: "会議は来週開催されます。"
- term: "policy"

**Output:**
```json
{
  "term": "policy",
  "term_translation": null
}
```

## Input
### Source Text
<<<|SOURCE_TEXT|>>>

### Target Text
<<<|TARGET_TEXT|>>>


### Term
<<<|TERM|>>>
````

Pairwise reranking. The prompt template used for pairwise reranking described in § 4.2 was as follows. The surrounding source segments that provided document-level context (e.g., preceding and following segments) were embedded in <<<|SURROUNDING_CONTEXT|>>>. The translations of previously processed source segments, used as context for the current translation, were embedded in <<<|PREVIOUSLY_TRANSLATED_CONTEXT|>>>. The source sentence to be translated (or the source sentence selected from the parallel corpus) was embedded in <<<|SOURCE_TEXT|>>>. A candidate translation for the current source segment (Translation A) was embedded in <<<|TRANSLATION_A|>>>. A candidate translation for the current source segment (Translation B) was embedded in <<<|TRANSLATION_B|>>>.

```
## Task
You will be given the following information as input:
1. Source sentence (original text)
2. Surrounding context (English)
3. Already translated preceding context (Japanese)
4. Two machine translation candidates A and B (Japanese)

Evaluate both candidates and **select the better translation** based on comprehensive quality assessment.

---

### 1. Evaluation Criteria
Determine the ranking according to the following 5 criteria:
1. **Adequacy**: How accurately the meaning of the source text is conveyed
2. **Fluency**: Grammatical correctness and especially the naturalness, readability, and rhythm of the Japanese. Non-literal translations are preferred.
3. **Terminology \& Proper Nouns**: Accuracy and consistency of technical terms and proper nouns. Eliminate any inconsistency in terminology usage and avoid variation in spelling or phrasing of proper nouns.
4. **Style**: Tone, punctuation, and formatting appropriate for the purpose and audience. Choose a tone that aligns with the context—such as conversational for social media posts or literary for narrative writing.
```

````
5. **Contextual Consistency**:
Consistent expression with the source
text, its surrounding context, and the
preceding translated content

---

### 2. Output Format (JSON)
Return only a JSON object following the
schema below.
Do not include any extra keys,
comments, or trailing commas.
```json
{
  "general_comment": "<Describe the
  overall reasoning for the
  selection>",
  "comparison_results": {
    "translation_A": {
      "strengths": "<Key strengths of
      Translation A>",
      "weaknesses": "<Key weaknesses of
      Translation A>"
    },
    "translation_B": {
      "strengths": "<Key strengths of
      Translation B>",
      "weaknesses": "<Key weaknesses of
      Translation B>"
    },
    "selection_reason": "<Brief
    explanation why the selected
    translation is better>",
    "selected_translation": "<A or B>"
  }
}
```
- general_comment: Overall assessment
explaining the comparison and selection
rationale.
- comparison_results
  - translation_A/B: Analysis of each
  translation
    - strengths: Main advantages of
    this translation
    - weaknesses: Main disadvantages of
    this translation
  - selection_reason: Concise
  explanation of why the selected
  translation is superior
  - selected_translation: "A" or "B" -
  the better translation


Even if the translations are very
similar in quality, always make a
definitive selection.

---

### 3. Comparison Procedure
- In the thinking process, please
conduct a detailed comparison by:
  1. Analyzing each translation against
  all evaluation criteria
  2. Identifying specific differences
  between Translation A and Translation
  B
  3. Weighing the relative importance
  of these differences in the given
  context
  4. Making a final judgment based on
  overall quality

---

### 4. Input
Context:
```txt
<<<|SURROUNDING_CONTEXT|>>>
```
Translated Context:
```txt
<<<|PREVIOUSLY_TRANSLATED_CONTEXT|>>>
```
Source:
```txt
<<<|SOURCE_TEXT|>>>
```
Translation A:
```txt
<<<|TRANSLATION_A|>>>
```
Translation B:
```txt
<<<|TRANSLATION_B|>>>
```
````

Group reranking. The prompt template used for group reranking described in § 4.1 was as follows. The surrounding source segments that provided document-level context (e.g., preceding and following segments) were embedded in <<<|SURROUNDING_CONTEXT|>>>.

The translations of previously processed source segments, used as context for the current translation, were embedded in <<<|PREVIOUSLY_TRANSLATED_CONTEXT|>>>. The source sentence to be translated (or the source sentence selected from the parallel corpus) was embedded in <<<|SOURCE_TEXT|>>>.

```
## Task
You will be given the following
information as input:
1. Source sentence (original text)
2. Surrounding context (English)
3. Already translated preceding context
(Japanese)
4. N machine translation candidates
(Japanese)

Evaluate each candidate and **rank them
from highest quality (rank_1) to lowest
quality (rank_N)**.

---

### 1. Evaluation Criteria
Determine the ranking according to the
following 5 criteria:
1. **Adequacy**: How accurately the
meaning of the source text is conveyed
2. **Fluency**: Grammatical
correctness and especially the
naturalness, readability, and rhythm of
the Japanese. Non-literal translations
are preferred.
3. **Terminology \& Proper Nouns**:
Accuracy and consistency of technical
terms and proper nouns. Eliminate any
inconsistency in terminology usage and
avoid variation in spelling or phrasing
of proper nouns.
4. **Style**: Tone, punctuation, and
formatting appropriate for the purpose
and audience. Choose a tone that aligns
with the context—such as
conversational for social media posts
or literary for narrative writing.
5. **Contextual Consistency**:
Consistent expression with the source
text, its surrounding context, and the
preceding translated content

---

### 2. Output Format (JSON)
Return only a JSON object following the
schema below.
Do not include any extra keys,
comments, or trailing commas.

```json
{
  "general_comment": "<Describe the
  overall reasoning for the ranking>",
  "reranking_results": {
    "rank_1": {
      "translation_id": <integer>,
      "score": <0 ~ 100>
    },
    "rank_2": {
      "translation_id": <integer>,
      "score": <0 ~ 100>
    }
    // ⋯
    ,
    "rank_N": {
      "translation_id": <integer>,
      "score": <0 ~ 100>
    }
  }
}
```

- general_comment: Overall assessment
explaining the ranking rationale.
- reranking_results
  - rank_i: rank_1 is the highest
  quality, rank_N is the lowest
  quality.
  - translation_id: The candidate
  number indicated in the input
  (Translation 1 ⇒ 1, Translation 2 ⇒
  2, ⋯).
  - score: Overall score from 0 to 100.
  Higher scores indicate better
  quality.

Even in case of ties, always determine
a definitive order and assign unique
ranks without duplicates.

```

```
### 3. Reranking Procedure
- In the thinking process, please make a final judgment by repeatedly comparing what differences exist between each pair of translation results and comparing which translation result is better.

---


### 4. Input
Context:
```txt
<<<|SURROUNDING_CONTEXT|>>>
```

Translated Context:
```txt
<<<|PREVIOUSLY_TRANSLATED_CONTEXT|>>>
```

Source:
```txt
<<<|SOURCE_TEXT|>>>
```
```

Document reranking. The prompt template used for document reranking described in § 5 was as follows. The source document to be translated was embedded in <<<|SOURCE_TEXT|>>>. A candidate translation for the current source document (Translation A) was embedded in <<<|TRANSLATION_A|>>>. A candidate translation for the current source document (Translation B) was embedded in <<<|TRANSLATION_B|>>>.

```
## High-quality translation result selection task

Given an **English source text** and two **Japanese translation candidates (A, B)**,
compare them and determine which one is superior overall, then respond in the specified format.

---

### 1. Evaluation Criteria

1. **Critical errors or unnaturalness can single-handedly determine the selection.**
2. **Adequacy** - Whether the translation accurately conveys all meaning and information from the source
3. **Fluency \& Style** - Whether it reads naturally in Japanese / Whether the style and tone match the context
4. **Terminology \& Proper Nouns** - Consistency and accuracy of technical terms and proper noun translations
5. **Consistency** - Coherence with surrounding paragraphs/sentences (tense, person, etc.)
6. **Readability** - Whether punctuation, line breaks, word order, etc. are reader-friendly

---

---

### 2. Output Format (one line per input segment)

Please output in the following JSON format:

```json
{
  "selected_translation": "A" | "B",
  "general_comment": "<1-2 sentences explaining the decisive factor>"
}
```

Constraints

* `selected_translation` must be either `"A"` or `"B"`.
* Do not output any characters outside the JSON (no surrounding \`\`\`, etc.).
* `general_comment` should be 2 sentences maximum.
* Always select one even if uncertain.

---

3. Comparison Procedure
```

````
1. Check A / B individually against the
6 evaluation criteria.
2. Prioritize significant differences
(mistranslations, fluency breakdowns,
terminology inconsistencies, etc.).
3. Specify the overall superior choice
as `selected_translation`.
4. Summarize the decisive factor
concisely.



### 4. Input

#### SOURCE:
```txt
<<<|SOURCE_TEXT|>>>
```

#### TRANSLATION A:
```txt
<<<|TRANSLATION_A|>>>
```

#### TRANSLATION B:
```txt
<<<|TRANSLATION_B|>>>
```
````