NTTSU at WMT2025 General Translation Task

Zhang Yin ♠, **Hiroyuki Deguchi** ♢, **Haruto Azami** ♠, **Guanyu Ouyang** ♠,
Kosei Buma ♠, **Yingyi Fu** ♠, **Katsuki Chousa** ♢, **Takehito Utsuro** ♠
♠University of Tsukuba ♢NTT, Inc.

Abstract

This paper presents the submission of NTTSU for the constrained track of the English–Japanese and Japanese–Chinese at the WMT2025 general translation task. For each translation direction, we build translation models from a large language model by combining continual pretraining, supervised fine-tuning, and preference optimization based on the translation quality and adequacy. We finally generate translations via context-aware MBR decoding to maximize translation quality and document-level consistency.

1 Introduction

We describe our NTTSU translation system in the WMT'25 English–Japanese (En–Ja) and Japanese–Chinese (Ja–Zh) general translation task under the constrained track.

Our translation models are trained on a pretraining large language model (LLM), Qwen3-14B (Qwen Team, 2025). We combine training methods for each translation direction from three training stages: continual pretraining (CPT) (Ke et al., 2023), supervised fine-tuning (SFT) (Zhang et al., 2024), and preference optimization (PO) (Rafailov et al., 2023). In PO, to maximize the translation quality and adequacy, we use two different reward metrics, MetricX-24 (Juraska et al., 2024) and coverage of word alignment between source and target texts (Wu et al., 2024). After training the models, we generate translations using context-aware minimum Bayes risk (MBR) decoding, which maximizes the expected translation quality (Kumar and Byrne, 2004; Eikema and Aziz, 2020) and also utilizes context information of both source and generated target texts, though we use a sentence-level metric (Kudo et al., 2024; Pombal et al., 2024). The following sections show the details of our system.

2 Approaches

2.1 Training

Continual pretraining Continual pretraining (CPT) continues to train LLM models based on the next token prediction as well as pretraining using monolingual corpora (Ke et al., 2023). Let $\mathbf{y} \coloneqq (y_1, y_2, \ldots, y_{|\mathbf{y}|}) \in \mathcal{V}^*$ be a sequence of tokens in a corpus, where $\mathcal{V}^*$ is the Kleene closure of vocabulary $\mathcal{V}$. CPT optimizes the model parameter θ by minimizing the loss function $\mathcal{L}_{\text{CPT}}$ over a monolingual corpus $\mathcal{D}_{\text{CPT}} \coloneqq \{\mathbf{y}_i\}_{i=1}^{|\mathcal{D}_{\text{CPT}}|} \subset \mathcal{V}^*$:

$$\underset{\theta}{\operatorname{argmin}} \sum_{\mathbf{y} \in \mathcal{D}_{\text{CPT}}} \mathcal{L}_{\text{CPT}}(\mathbf{y}; \theta), \tag{1}$$

$$\mathcal{L}_{\text{CPT}}(\mathbf{y}; \theta) \coloneqq -\sum_{t=1}^{|\mathbf{y}|} \log p_\theta(y_t | \mathbf{y}_{<t}). \tag{2}$$

For efficiency, $\mathbf{y}_{[t-c,t)} \coloneqq (y_{t-c}, y_{t-c+1}, \ldots y_{t-1})$ is used instead of $\mathbf{y}_{<t}$ in practice, where $c \in \mathbb{N}$ is a length of a context window. This objective is the same as the pretraining loss of causal language models, i.e., the model is trained to predict the next token y_t under the condition of c context tokens.

Supervised fine-tuning Supervised fine-tuning (SFT) adapts a pretrained model to downstream tasks using labeled data (Zhang et al., 2024). Specifically, given a pretrained model parameter θ, SFT updates it on a labeled dataset $\mathcal{D}_{\text{SFT}} \coloneqq \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{D}_{\text{SFT}}|} \subset \mathcal{V}^* \times \mathcal{V}^*$, where $\mathbf{x}_i$ and $\mathbf{y}_i$ are the input and its corresponding ground-truth output sequence, respectively, as follows:

$$\underset{\theta}{\operatorname{argmin}} \sum_{(\mathbf{x},\mathbf{y}) \in \mathcal{D}_{\text{SFT}}} \mathcal{L}_{\text{SFT}}(\mathbf{x}, \mathbf{y}; \theta), \tag{3}$$

$$\mathcal{L}_{\text{SFT}}(\mathbf{x}, \mathbf{y}; \theta) \coloneqq -\log p_\theta(\mathbf{y} \mid \mathbf{x}). \tag{4}$$

This encourages the model to generate outputs that are consistent with the human-annotated targets.

Proceedings of the Tenth Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 657–665
November 8–9, 2025.

Preference optimization Preference optimization (PO) aims to align a trained model with preferences. One of the major PO algorithms is direct PO (DPO), which uses pairwise comparison data instead of explicit reward models (Rafailov et al., 2023). Let $\mathcal{D}_{\text{PO}} \coloneqq \{(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)\}_{i=1}^{|\mathcal{D}_{\text{PO}}|} \subset \mathcal{V}^* \times \mathcal{V}^* \times \mathcal{V}^*$ be a triplet dataset that consists of a prompt $\mathbf{x}$ and its corresponding output pairs $(\mathbf{y}^+, \mathbf{y}^-)$, where $\mathbf{y}^+$ is preferred over $\mathbf{y}^-$ according to human feedback or a reward function, i.e., $\mathbf{y}^+ \succeq \mathbf{y}^-$. PO tunes a model θ by minimizing a pairwise loss that encourages the model to generate $\mathbf{y}^+$ rather than $\mathbf{y}^-$. We minimize the following objective function that incorporates adaptive rejection (Xu et al., 2025) into SimPO, a variant of DPO (Meng et al., 2024):

$$\operatorname*{argmin}_{\theta} \sum_{(\mathbf{x},\mathbf{y}^+,\mathbf{y}^-)\in\mathcal{D}_{\text{PO}}} \mathcal{L}_{\text{PO}}(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-; \theta), \quad (5)$$

$$\begin{aligned}\mathcal{L}_{\text{PO}}(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-; \theta) \coloneqq \\ -\log \sigma \left(r(\mathbf{x}, \mathbf{y}^+) - \tau_\theta(\mathbf{y}^+, \mathbf{y}^-) r_\theta(\mathbf{x}, \mathbf{y}^-) - \gamma\right) \\ + \alpha \log p_\theta(\mathbf{y}^+|\mathbf{x}), \quad (6)\end{aligned}$$

where $\alpha \in \mathbb{R}$ is a weight of the behavior cloning regularizer, $\gamma \in \mathbb{R}$ is a reward margin between $\mathbf{y}^+$ and $\mathbf{y}^-$. Note that $r_\theta(\mathbf{x}, \mathbf{y})$, $\tau_\theta(\mathbf{y}^+, \mathbf{y}^-)$, and $z_\theta(\mathbf{y}^+, \mathbf{y}^-)$ are defined as follows:

$$r_\theta(\mathbf{x}, \mathbf{y}) \coloneqq \frac{\beta}{|\mathbf{y}|} \log p_\theta(\mathbf{y}|\mathbf{x}), \quad (7)$$

$$\tau_\theta(\mathbf{y}^+, \mathbf{y}^-) \coloneqq \min\left(e^{\eta \cdot z_\theta(\mathbf{y}^+, \mathbf{y}^-)} - 1, 1\right), \quad (8)$$

$$z_\theta(\mathbf{y}^+, \mathbf{y}^-) \coloneqq \left|\frac{\log p_\theta(\mathbf{y}^+|\mathbf{x})}{|\mathbf{y}^+|} - \frac{\log p_\theta(\mathbf{y}^-|\mathbf{x})}{|\mathbf{y}^-|}\right|, \quad (9)$$

where $\beta \in \mathbb{R}$ and $\eta \in \mathbb{R}$ are hyperparameters.

Stepwise preference optimization Stepwise PO (Wachi et al., 2024) is an extension of PO designed to align models with multiple preference metrics. It optimizes the model using multiple preferences sequentially, where each stage focuses on a distinct preference objective. Consequently, by chaining multiple preference optimization stages, the model incrementally aligns with multiple-perspective preferences.

2.2 Decoding

We generate translations via context-aware minimum Bayes risk (MBR) decoding, which leverages sentence-level metrics for MBR decoding (Goel and Byrne, 2000; Kumar and Byrne, 2004; Eikema and Aziz, 2020) yet utilizes both source and generated target context information.

Algorithm 1: Context-aware MBR decoding

Given : Translation model θ, utility function u, the number of hypotheses $|\mathcal{H}|$, and the context size $C \in \mathbb{N}$.
Input : Source document $\mathbf{X} \coloneqq (\mathbf{x}_1, \ldots, \mathbf{x}_{|\mathbf{X}|})$ where $\mathbf{x}_i \in \mathcal{V}^*$ is the i-th source sentence.
Output : Target document $\mathbf{Y} \coloneqq (\mathbf{y}_1, \ldots, \mathbf{y}_{|\mathbf{Y}|})$.

$\mathbf{Y} \leftarrow \phi$
Create queues: $\mathbf{C}_x \leftarrow \phi$ and $\mathbf{C}_y \leftarrow \phi$
for $i \leftarrow 1 \ldots |\mathbf{X}|$ **do**
  Enqueue$(\mathbf{C}_x, \mathbf{x}_i)$
  `// ℋ is a multiset of hypotheses.`
  $\mathcal{H} \leftarrow \{\mathbf{h}_k \sim p(\mathbf{y}_i|\mathbf{C}_x, \mathbf{C}_y; \theta)\}_{k=1}^{|\mathcal{H}|}$
  `// We use the same candidate set for hypotheses and pseudo-references.`
  $\hat{\mathbf{y}}_i \leftarrow \operatorname{argmax}_{\mathbf{h}\in\mathcal{H}} \frac{1}{|\mathcal{H}|} \sum_{k=1}^{|\mathcal{H}|} u(\mathbf{h}, \mathbf{h}_k)$
  `// ∘ denotes concatenetion.`
  $\mathbf{Y} \leftarrow \mathbf{Y} \circ \hat{\mathbf{y}}_i$
  Enqueue$(\mathbf{C}_y, \hat{\mathbf{y}}_i)$
  while $|\mathbf{C}_x| > C$ **do**
    Dequeue$(\mathbf{C}_x)$
  while $|\mathbf{C}_y| > C$ **do**
    Dequeue$(\mathbf{C}_y)$
return $\mathbf{Y}$

MBR decoding The goal of MBR decoding is to find a translation that maximizes the expected utility rather than the output probability (Goel and Byrne, 2000; Kumar and Byrne, 2004). The objective is formally defined as follows:

$$\mathbf{y}^\star_{\text{MBR}} \coloneqq \operatorname*{argmax}_{\mathbf{h}\in\mathcal{V}^*} \mathop{\mathbb{E}}_{\mathbf{y}\sim\Pr(\cdot|\mathbf{x})}[u(\mathbf{h}, \mathbf{y})], \quad (10)$$

where $\Pr(\cdot|\mathbf{x})$ is the true probability of human translation and $u\colon \mathcal{V}^* \times \mathcal{V}^* \to \mathbb{R}$ is a utility function that evaluates a hypothesis under the given reference $\mathbf{y}$ and satisfies $\mathbf{h}^+ \succeq \mathbf{h}^- \iff u(\mathbf{h}^+, \mathbf{y}) \geq u(\mathbf{h}^-, \mathbf{y})$. Since searching over $\mathcal{V}^*$ and calculating the expectation over the output space are infeasible, the objective of MBR decoding is approximated by the Monte Carlo (MC) estimation (Eikema and Aziz, 2020, 2022). We denote a hypothesis set by $\mathcal{H} \subset \mathcal{V}^*$. The MBR decoding with the MC estimation is calculated as follows:

$$\mathbf{y}_{\text{MBR}} \coloneqq \operatorname*{argmax}_{\mathbf{h}\in\mathcal{H}} \frac{1}{|\hat{\mathcal{Y}}|} \sum_{\mathbf{y}\in\hat{\mathcal{Y}}} u(\mathbf{h}, \mathbf{y}), \quad (11)$$

where $\hat{\mathcal{Y}} \coloneqq \{\mathbf{y}_i \sim p_\theta(\mathbf{y}|\mathbf{x})\}_{i=1}^{|\hat{\mathcal{Y}}|}$ is a multiset, a.k.a. bag, of pseudo-references, translation samples drawn from the output probability of translation model θ. Typically, hypotheses are also used as pseudo-references.

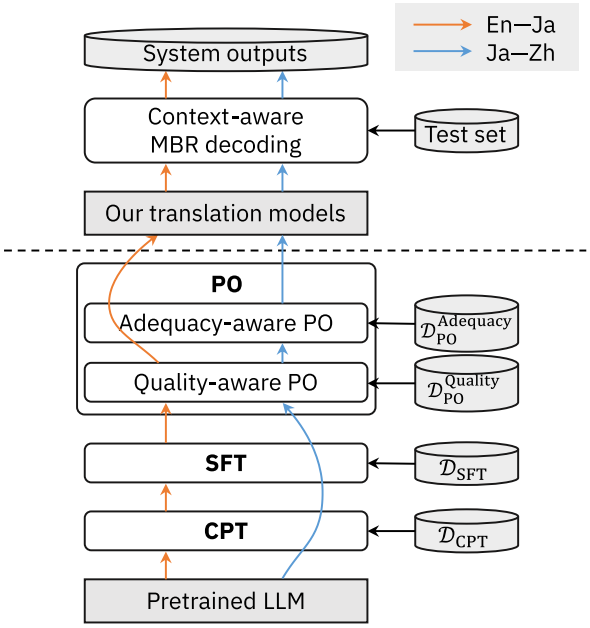

Figure 1: Overview of our translation system.

Method	En–Ja	Ja–Zh
Continual pretraining (CPT)	✓	✗
Supervised fine-tuning (SFT)	✓	✗
Preference optimization (PO)	—	—
Quality-aware PO	✓	✓
Adequacy-aware PO	✗	✓
Context-aware MBR decoding	✓	✓

Table 1: List of methods we employed.

Hyperparameter	CPT	SFT
Optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$) (Loshchilov and Hutter, 2019)	
Learning rate	2.5×10^{-5}	1×10^{-6}
Scheduler	cosine	inverse square root
Warmup ratio	1%	1%
Weight decay	0.1	0.1
Gradient clip	1.0	1.0
Epoch	1	3
Batch size	1,024 chunks	64 sentence pairs
Chunk size	2,048 tokens	N/A
Accelerator	DeepSpeed ZeRO-2 (Rasley et al., 2020)	
Precision	bfloat16	bfloat16

Table 2: Hyperparameters of CPT and SFT.

Context-aware MBR decoding For document-level translation, we extend MBR decoding to a context-aware method. However, most automatic evaluation metrics, which are used for the utility function, are designed for sentence-level metrics. To bridge this gap, we determine output translations for each sentence by MBR decoding with a sentence-level utility and add the generated translation to the context, similar to Kudo et al. (2024) and Pombal et al. (2024). Algorithm 1 shows our decoding algorithm, which autoregressively generates sentence-level translations at each step. Since the WMT'25 general translation task provides the source document without sentence segmentation, we first apply a sentence segmenter before running our decoding algorithm. In Line 5, hypotheses are sampled given the source and target context sentences, i.e., $\mathbf{C}_x$ and $\mathbf{C}_y$, respectively. The source context includes the current source sentence $\mathbf{x}_i$ as well as the preceding ones, while the target context consists only of previously generated target sentences. Accordingly, the model focuses on the current sentence $\mathbf{x}_i$ and generates its corresponding target sentence $\mathbf{y}_i$ under the given contexts, naturally. The hyperparameter $C \in \mathbb{N}$ denotes the size of the context queues. Rather than using a fixed number of sentences, we set it based on the paragraph size, i.e., a variable number of sentences depending on the dataset format.

3 Submission System

We train En–Ja and Ja–Zh translation models from a pretrained LLM, Qwen3-14B (Qwen Team, 2025). According to our preliminary experiments and subjective judgment, we selected the combinations of training methods. Finally, we generate translations via context-aware MBR decoding. We show the system overview in Figure 1 and Table 1.

3.1 Continual pretraining

We perform the bilingual CPT only for the En–Ja model. For the training data of CPT, we use JParaCrawl v3.0 (Morishita et al., 2022) and filter it into 20.8M sentence pairs using LEALLA-large (Mao and Nakagawa, 2023). We create training examples following Kondo et al. (2024). The hyperparameters of CPT are listed in Table 2.

3.2 Supervised fine-tuning

Similar to CPT, we conduct supervised fine-tuning (SFT) only for the En–Ja model. For the training data of SFT, we use the development and test sets of the WMT'20 translation task (Barrault et al., 2020) and FLoRes-200 (NLLB Team et al., 2022), along with the train set of the Kyoto Free Translation Task (KFTT) (Neubig, 2011). For the development set, we use the test set of the WMT'21 translation task (Akhbardeh et al., 2021). The hyperparameters of SFT are also listed in Table 2.

3.3 Preference optimization

To maximize translation quality and adequacy, we perform PO with two reward metrics. For En–Ja, we apply the quality-aware PO on top of the model trained via SFT. For Ja–Zh, we apply both quality- and adequacy-aware PO through stepwise PO. Note that we do not apply CPT and SFT to the Ja–Zh model as listed in Table 1; thus, we directly tune the pretrained Qwen3-14B.

Quality-aware PO To improve translation quality, we employ an automatic evaluation metric that highly correlates with human assessments for creating the preference data. Specifically, we first randomly sample 20,000 source sentences from the NewsCrawl corpus (Kocmi et al., 2024), and generate translations for each source sentence using two LLMs, Qwen3-32B and Aya-Expanse-32B (Dang et al., 2024). To obtain high-quality translations efficiently, we employ COMET[1] (Rei et al., 2022a) -based MBR decoding using 64 hypotheses sampled via epsilon sampling with $\varepsilon = 0.02$ (Freitag et al., 2023). These high-quality translations and the baseline translations, generated via beam search from Qwen3-14B, are then compared using MetricX-24-XXL (Juraska et al., 2024). Among the outputs from the three models, we label the highest-quality translation as the preferred, i.e., chosen, instance and the lowest-quality translation as the non-preferred, i.e., rejected, instance. From these paired instances with each source sentence, we construct the training dataset for quality-aware PO. Finally, we train the model by optimizing Equation (5) on the created preference data with $\alpha = 1.0, \beta = 0.2, \eta = 1.5, \gamma = 0.005$.

Adequacy-aware PO To mitigate hallucination and omission, i.e., overgeneration and undergeneration, we also employ the word alignment-based preference metric (Wu et al., 2024) for Ja–Zh. For the preference data, we randomly sample 10,000 source sentences from CCAligned (El-Kishky et al., 2020), where each sentence has at least 15 characters. To label the preference data, we use the coverage score obtained via word alignment calculated by WSPAlign[2] (Wu et al., 2023). Apart from these two modifications, we follow the same procedure as in the quality-aware PO, but with different hyperparameters: $\alpha = 1.0, \beta = 0.01, \eta = 1.5, \gamma = 0.005$.

[1] https://huggingface.co/Unbabel/wmt22-comet-da

[2] https://huggingface.co/qiyuw/WSPAlign-mbert-base

3.4 Prompt templates

We basically use the following template that turns on the `continue_final_message` (CFM) option defined in the tokenizers of Huggingface transformers (Wolf et al., 2020):

```
<|im_start|>user
Translate this from English to Japanese:
English: ...<|im_end|>
<|im_start|>assistant
<think>
</think>
Japanese:
```

We call this "CFM" template. The CFM template inserts the target language name with a colon into the last of the assistant chat and does not close it. Hence, the model naturally generates a target text following the target language name.

However, in our preliminary Ja–Zh translation experiments, we observed that generated texts with the CFM template are often collapsed due to hallucinations. Thus, we change the inference template to the below "AGP" template, which enables the `add_generation_prompt` (AGP) option instead of the `continue_final_message` option:

```
<|im_start|>user
Translate this from English to Japanese:
English: ...
Japanese:<|im_end|>
<|im_start|>assistant
<think>
</think>
```

Although there is a slight difference between training and inference, we employ this method because hallucinations decrease in Ja–Zh.

To summarize, we train both En–Ja and Ja–Zh models with the CFM template, and generate translations with CFM for En–Ja and AGP for Ja–Zh.

3.5 Decoding

In decoding, we use MetricX-24-XXL (Juraska et al., 2024) for the utility function u. During decoding, we generate 64 translation candidates via epsilon sampling with $\varepsilon = 0.02$ (Freitag et al., 2023) and use them for both hypotheses and pseudo-references. We split the source documents into sentences using segment-any-text[3] (Frohmann et al., 2024). We use at most one previous paragraph as context, i.e., the target context includes a preceding generated paragraph and generated sentences until the current focused sentence.

[3] https://huggingface.co/segment-any-text/sat-12l-sm

CPT	SFT	PO Quality	PO Adequacy	MTX24↓	XCMT↑	KIWI22↑
✗	✗	✗	✗	4.44	79.89	83.04
			✓	4.39	80.28	82.97
		✓	✗	4.32	80.60	83.08
			✓	4.44	79.67	83.08
	✓	✗	✗	4.82	76.58	81.89
			✓	4.67	77.82	82.52
		✓	✗	4.32	80.82	83.05
			✓	4.47	79.31	83.06
✓	✗	✗	✗	Failed	Failed	Failed
			✓	5.35	72.97	81.04
		✓	✗	4.30	81.02	**83.15**
			✓	4.50	79.81	83.09
	✓	✗	✗	4.97	75.82	81.36
			✓	4.79	77.10	82.07
		✓	✗	**4.29**	**81.27**	83.11
			✓	4.54	79.66	82.99

Table 3: Comparisons of training methods on the WMT'24 En–Ja test set. The **bold** font indicates the best scores in each metric. The green highlighted rows indicate the setting of our submission system.

CPT	SFT	PO Quality	PO Adequacy	MTX24↓	XCMT↑	KIWI22↑
✗	✗	✗	✗	3.51	73.55	73.26
			✓	3.44	73.75	73.12
		✓	✗	3.46	74.03	73.12
			✓	**3.43**	**74.36**	**73.38**
✗	✓	✗	✗	4.17	70.20	72.38
			✓	4.08	70.96	72.32
		✓	✗	3.53	73.17	73.10
			✓	3.54	73.26	73.15
✓	✗	✗	✗	Failed	Failed	Failed
			✓	Failed	Failed	Failed
		✓	✗	3.92	66.81	71.73
			✓	4.06	66.71	72.00
✓	✓	✗	✗	5.43	63.97	70.63
			✓	4.38	68.10	71.42
		✓	✗	3.57	71.28	73.07
			✓	3.65	71.14	73.10

Table 4: Comparisons of training methods on the WMT'24 Ja–Zh test set. The **bold** font indicates the best scores in each metric. The green highlighted rows indicate the setting of our submission system.

4 Experiments

4.1 Ablation study of training methods

We investigate the effects of each training method.

Setup We compare the combination of training methods: CPT, SFT, quality-aware PO, and adequacy-aware PO. We train models with the same training data and hyperparameters as our submission system, as described in Section 4.1, except for the differences noted below. In Ja–Zh, we use the same hyperparameters as listed in Table 2 for both CPT and SFT. For the training data of CPT in Ja–Zh, we use the parallel corpora listed in "WMT 2025 Translation Task Training Data"[4]. We filter them to retain only those with CometKiwi-22 (Rei et al., 2022b) scores between 0.5 and 0.88, and then clean them using bifixer (Ramírez-Sánchez et al., 2020). In both En–Ja and Ja–Zh, the source sides of training examples are shared between SFT and adequacy-aware PO. The translation quality is evaluated on MetricX-24-XXL (MTX24) (Juraska et al., 2024), xCOMET-XXL (XCMT) (Guerreiro et al., 2024), and CometKiwi-22 (KIWI22) (Rei et al., 2022b) in the test sets of WMT'24 En–Ja and Ja–Zh translation tasks (Kocmi et al., 2024).

[4] https://www2.statmt.org/wmt25/mtdata/

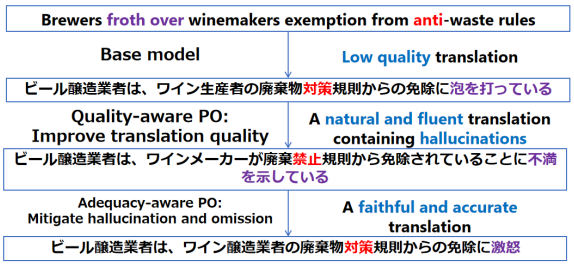

Figure 2: Examples of stepwise preference optimization (PO). Purple and red texts highlight corresponding phrases in the source text and its translations. Blue text provides descriptive labels for each step.

Results The results of automatic evaluation on the WMT'24 test sets are demonstrated in Table 3 and Table 4. In the tables, "Failed" indicates that it failed to generate translations due to hallucinations or critical errors, and it cannot be evaluated. As demonstrated in Table 3 and Table 4, the configuration of our submission system achieved the best MetricX-24-XXL and xCOMET-XXL scores in both En–Ja and Ja–Zh. In addition, we also confirmed through subjective judgment that these models have successfully generated the highest-quality translations compared to other settings. Accordingly, we selected these combinations of training methods for each translation direction.

Figure 2 shows translation examples of the same sentence from the WMT24 test set, generated by

the base model (Qwen3-14B), the model after an initial training step with Quality-aware PO, and the model after a subsequent step with Adequacy-aware PO. As shown in the figure, the initial translation from the base model is relatively low quality, incorrectly translating "froth over." After training with Quality-aware PO, the translation becomes more fluent and natural overall. However, hallucinations occur during translation—for example, rendering "froth over," which expresses anger, as merely expressing dissatisfaction, and interpreting "anti," which denotes countermeasures, as "prohibit." In contrast, the model trained with Adequacy-aware PO produces a translation that is accurate and faithful to the source text.

It is noteworthy that among the three translations, the one from the Quality-aware PO model achieved the best MetricX-24-XXL score. From up to below, the MetricX-24-XXL scores are 6.34, 4.99, and 5.54. This indicates that even advanced metrics such as MetricX-24-XXL may assign better scores to fluent and natural translations that contain hallucinations than to factually accurate but less fluent ones.

4.2 Comparison of decoding strategy

We evaluate our decoding algorithm using the final submission system by comparing it with a baseline context-aware MAP decoding.

Setup We use the WMT'25 En–Ja and Ja–Zh translation task and evaluate the translation quality of decoding methods using reference-free quality estimation (QE) models, MetricX-23-QE-XXL (Juraska et al., 2023), MetricX-24-XXL[5] (Juraska et al., 2024), and CometKiwi-23-XXL (Rei et al., 2023). We compare our decoding algorithm with context-aware MAP decoding, which employs a beam search with a beam size of 5. The context sizes of both methods are at most one previous paragraph, as described in Section 3.5.

For evaluation, we first apply a sentence segmenter[6] (Frohmann et al., 2024) to each source and target paragraph and compute the scores across all pairs of source and target sentences for each paragraph. Then, we compute the score alignment that maximizes the total scores. Finally, the document-level QE scores are calculated by averaging the paragraph-level QE scores.

Direction	Decoding	MTX23↓	MTX24↓	KIWI23↑
En–Ja	MAP	3.4	4.9	74.7
	MBR	**3.0**	**4.2**	**77.2**
Ja–Zh	MAP	4.0	4.9	63.1
	MBR	**3.5**	**4.7**	**64.8**

Table 5: Reference-free quality estimation scores on the WMT'25 test set. The **bold** font indicates the best scores in each translation direction. The green highlighted rows indicate the setting of our submission system.

Results Table 5 demonstrates the translation quality of decoding methods. The table shows that MBR decoding consistently outperformed MAP decoding across all metrics, even though we used only MetricX-24-XXL for the utility function. One reason for these results is that MAP decoding tends to propagate translation errors, including hallucinations, whereas MBR decoding carefully selects translations based on the expected utility computed from the evaluation metric and other translation samples, thereby mitigating the generation of pathological sequences.

5 Conclusion

We built our system on the WMT'25 general translation task in En–Ja and Ja–Zh. Our models were trained with the combinations of CPT, SFT, and stepwise PO based on the quality- and adequacy-aware rewards, for each translation direction. To maximize the translation quality and document-level consistency, we generated translations via context-aware MBR decoding.

In document-level translation, we observed that LLMs are more likely to generate collapsed hallucination texts. To mitigate this issue, we employed adequacy-aware PO. Nevertheless, in some cases, the models still failed to generate translations. We hope to further improve hallucination mitigation in document translation.

Limitations

Metric bias We used MetricX-24-XXL for the preference data creation in PO and the utility function of MBR decoding, which heavily relied on a single metric. Thus, our system may be affected by the metric bias.

Domain adaptation We built a single system for each translation direction, regardless of domains,

[5]MetricX-24 is a hybrid reference-based/-free metric, so we use it as a reference-free QE model in this evaluation.

[6]https://huggingface.co/segment-any-text/sat-12l-sm

while the WMT'25 general translation task contains multiple domains. By considering domain-specific knowledge and preferences, further improvements in translation quality can be expected.

Multimodal translation In the speech domain, original videos are also provided in addition to plain texts transcribed by an automatic speech recognition (ASR), but we did not use them. This means that ours is a cascade-style speech-to-text or video-to-text translation in the speech domain. By utilizing the original videos and audio, we can expect to suppress the propagation of errors caused by an ASR system.

Acknowledgements

This work was done mainly under a collaborative research agreement between NTT and Tsukuba University. Additionally, this work partially used computational resources of Pegasus provided by the Multidisciplinary Cooperative Research Program in the Center for Computational Sciences, University of Tsukuba.

Author Contributions

Zhang Yin applied PO, conducted translation experiments as described in Section 4.1 and other preliminary experiments, and selected the submission system.
Hiroyuki Deguchi conducted context-aware MBR decoding, as described in Section 2.2, and experiments regarding decoding strategies as shown in Section 4.2.
Haruto Azami applied CPT in En–Ja and generated the preference data in En–Ja for PO.
Guanyu Ouyang applied SFT in En–Ja and selected the submission system.
Kosei Buma collected and cleaned up the Ja–Zh dataset for SFT and PO via translation scoring and deduplication.
Yingyi Fu participated in discussions and reviewed system performance.
Katsuki Chousa provided advice on model development and decoding.
Takehito Utsuro built and managed our team.

References

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, Jungo Kasai, Daniel Khashabi, Kevin Knight, Tom Kocmi, Philipp Koehn, Nicholas Lourie, Christof Monz, Makoto Morishita, Masaaki Nagata, Ajay Nagesh, Toshiaki Nakazawa, Matteo Negri, Santanu Pal, Allahsera Auguste Tapo, Marco Turchi, Valentin Vydrin, and Marcos Zampieri. 2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. 2020. Findings of the 2020 conference on machine translation (WMT20). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1–55, Online. Association for Computational Linguistics.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, Acyr Locatelli, Hangyu Lin, Dwarak Talupuru, Bharat Venkitesh, David Cairuz, Bowen Yang, Tim Chung, Wei-Yin Ko, Sylvie Shang Shi, Amir Shukayev, Sammie Bae, Aleksandra Piktus, Roman Castagné, Felipe Cruz-Salinas, Eddie Kim, Lucas Crawhall-Stein, Adrien Morisot, Sudip Roy, Phil Blunsom, Ivan Zhang, Aidan Gomez, Nick Frosst, Marzieh Fadaee, Beyza Ermis, Ahmet Üstün, and Sara Hooker. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier.

Bryan Eikema and Wilker Aziz. 2020. Is MAP decoding all you need? the inadequacy of the mode in neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4506–4520, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Bryan Eikema and Wilker Aziz. 2022. Sampling-based approximations to minimum Bayes risk decoding for neural machine translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10978–10993, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

Markus Freitag, Behrooz Ghorbani, and Patrick Fernandes. 2023. Epsilon sampling rocks: Investigating sampling strategies for minimum Bayes risk decoding for machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9198–9209, Singapore. Association for Computational Linguistics.

Markus Frohmann, Igor Sterner, Ivan Vulić, Benjamin Minixhofer, and Markus Schedl. 2024. Segment any text: A universal approach for robust, efficient and adaptable sentence segmentation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11908–11941, Miami, Florida, USA. Association for Computational Linguistics.

Vaibhava Goel and William J Byrne. 2000. Minimum bayes-risk automatic speech recognition. *Computer Speech & Language*, 14(2):115–135.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

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

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

Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. 2023. Continual pre-training of language models. In *The Eleventh International Conference on Learning Representations*.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popović, Mariya Shmatova, Steinthór Steingrímsson, and Vilém Zouhar. 2024. Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1–46, Miami, Florida, USA. Association for Computational Linguistics.

Minato Kondo, Takehito Utsuro, and Masaaki Nagata. 2024. Enhancing translation accuracy of large language models through continual pre-training on parallel data. In *Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024)*, pages 203–220, Bangkok, Thailand (in-person and online). Association for Computational Linguistics.

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.

Shankar Kumar and William Byrne. 2004. Minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 169–176, Boston, Massachusetts, USA. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Zhuoyuan Mao and Tetsuji Nakagawa. 2023. LEALLA: Learning lightweight language-agnostic sentence embeddings with knowledge distillation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1886–1894, Dubrovnik, Croatia. Association for Computational Linguistics.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems*, volume 37, pages 124198–124235. Curran Associates, Inc.

Makoto Morishita, Katsuki Chousa, Jun Suzuki, and Masaaki Nagata. 2022. JParaCrawl v3.0: A large-scale English-Japanese parallel corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6704–6710, Marseille, France. European Language Resources Association.

Graham Neubig. 2011. The Kyoto free translation task. http://www.phontron.com/kftt.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti

Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Jose Pombal, Sweta Agrawal, and André Martins. 2024. Improving context usage for translating bilingual customer support chat with large language models. In *Proceedings of the Ninth Conference on Machine Translation*, pages 993–1003, Miami, Florida, USA. Association for Computational Linguistics.

Qwen Team. 2025. Qwen3 technical report.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, pages 53728–53741. Curran Associates, Inc.

Gema Ramírez-Sánchez, Jaume Zaragoza-Bernabeu, Marta Bañón, and Sergio Ortiz Rojas. 2020. Bifixer and bicleaner: two open-source tools to clean your parallel data. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 291–298, Lisboa, Portugal. European Association for Machine Translation.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 3505–3506, New York, NY, USA. Association for Computing Machinery.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022a. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

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

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

Akifumi Wachi, Thien Q. Tran, Rei Sato, Takumi Tanabe, and Youhei Akimoto. 2024. Stepwise alignment for constrained language model policy optimization. In *Advances in Neural Information Processing Systems*, volume 37, pages 104471–104520. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Qiyu Wu, Masaaki Nagata, Zhongtao Miao, and Yoshimasa Tsuruoka. 2024. Word alignment as preference for machine translation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3223–3239, Miami, Florida, USA. Association for Computational Linguistics.

Qiyu Wu, Masaaki Nagata, and Yoshimasa Tsuruoka. 2023. WSPAlign: Word alignment pre-training via large-scale weakly supervised span prediction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11084–11099, Toronto, Canada. Association for Computational Linguistics.

Haoran Xu, Kenton Murray, Philipp Koehn, Hieu Hoang, Akiko Eriguchi, and Huda Khayrallah. 2025. X-ALMA: Plug & play modules and adaptive rejection for quality translation at scale. In *The Thirteenth International Conference on Learning Representations*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2024. Instruction tuning for large language models: A survey.