SH at WMT25 General Machine Translation Task

Hayate Shiroma
University of the Ryukyus
e225719_@_cs.u-ryukyu.ac.jp

Abstract

We participated in the unconstrained track of the English-to-Japanese translation task at the WMT 2025 General Machine Translation Task. Our submission leverages several large language models, all of which are trained with supervised fine-tuning, and some further optimized via preference learning. To enhance translation quality, we introduce an automatic post-editing model and perform automatic post-editing. In addition, we select the best translation from multiple candidates using Minimum Bayes Risk (MBR) decoding. For MBR decoding, we use COMET-22 and LaBSE-based cosine similarity as evaluation metrics.

1 Introduction

In this paper, we describe the system submitted by Team SH to WMT2025.

We participated in the unconstrained track of the General Machine Translation Task for English-to-Japanese (En-Ja) translation.

Our submission leverages several large language models (LLMs) trained with supervised fine-tuning, and some further optimized via preference learning.

To enhance translation quality, we introduce an automatic post-editing model that performs automatic post-editing.

Additionally, we select the best translation from multiple candidates using Minimum Bayes Risk (MBR) decoding (Fernandes et al., 2022).

For MBR decoding, we use COMET-22 (Rei et al., 2022) and LaBSE-based cosine similarity (Feng et al., 2022) as evaluation metrics.

Our system is designed to translate text on a sentence-by-sentence basis, with each sentence separated by newlines.

Our system is based on two hypotheses. First, we hypothesize that preference learning contributes to improving translation quality, as it can consider both positive and negative examples, encouraging the model to generate better translations. Second, we hypothesize that the combination of automatic post-editing and MBR decoding contributes to improving translation quality. While automatic post-editing can sometimes degrade translations, using MBR decoding allows us to select the best translation from multiple candidates, thereby mitigating the risk of degradation and improving overall translation quality.

2 System Overview

Our system consists of three components: an initial translation model (Section 3), an automatic post-editing model (Section 4), and a reranking step (Section 5). The overall architecture of the system is shown in Figure 1.

The initial translation model produces a translation given the source text as input. The automatic post-editing model takes both the source text and the initial translation as input and generates an improved translation, thereby enhancing the output of the initial translation model.

The initial translation model is trained using supervised fine-tuning (SFT) and preference learning, while the automatic post-editing model is trained using SFT. We denote the model trained with only SFT as $\mathrm{INIT}_{\mathrm{SFT}}$ and the model trained with both SFT and preference learning as $\mathrm{INIT}_{\mathrm{SimPO}}$. Their corresponding automatic post-editing models are denoted as $\mathrm{PEDIT}_{\mathrm{SFT}}$ and $\mathrm{PEDIT}_{\mathrm{SimPO}}$, respectively.

During inference, we generate multiple translations from these models and select the best translation using MBR decoding. Specifically, we construct a candidate set from four pipelines: $\mathrm{INIT}_{\mathrm{SFT}}$ alone, $\mathrm{INIT}_{\mathrm{SimPO}}$ alone, $\mathrm{PEDIT}_{\mathrm{SFT}}$ applied to the output of $\mathrm{INIT}_{\mathrm{SFT}}$, and $\mathrm{PEDIT}_{\mathrm{SimPO}}$ applied to the output of $\mathrm{INIT}_{\mathrm{SimPO}}$.

Proceedings of the Tenth Conference on Machine Translation, pages 644–650
November 8-9, 2025

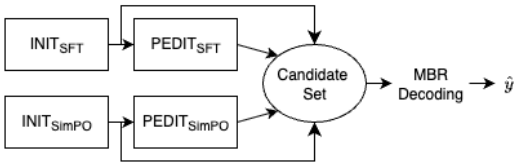

Figure 1: Overall architecture of the system

3 Initial Translation Model

3.1 Datasets

We used the following datasets for supervised fine-tuning: News Commentary (Kocmi et al., 2023), the Kyoto Free Translation Corpus (KFTT) (Neubig, 2011), TED Talks (Barrault et al., 2020), and past WMT General Machine Translation Task test data (WMT20, WMT21, WMT22, WMT23).

For preference learning, we used the same datasets as for supervised fine-tuning, but reformatted them to meet the requirements.

Preference learning requires a triplet consisting of source text, preferred translation, and non-preferred translation. However, the original datasets contain only the source text and the reference translation.

Therefore, we translated the source text using SFT model and used the output as the non-preferred translation.

3.2 Model Selection

We employed the cyberagent/DeepSeek-R1-Distill-Qwen-14B-Japanese (Ishigami, 2025) as a pre-trained model.

This model was further trained on Japanese data based on deepseek-ai/DeepSeek-R1-Distill-Qwen-14B (DeepSeek-AI et al., 2025).

The reason for selecting this model is that it has been trained on large amounts of Japanese and Chinese data in addition to English, making it suitable for the English-to-Japanese translation task. Moreover, it is one of the most recent Japanese LLMs.

3.3 Training Procedure

The training procedure of the initial translation model is shown in Figure 2.

Supervised Fine-Tuning First, We performed supervised fine-tuning using QLoRA (Dettmers et al., 2023).

QLoRA is an efficient fine-tuning method that combines the low-rank adaptation technique

Quantization Settings	
Load in 4-bit	True
Quantization Datatype	4-bit NormalFloat
Double Quantization	True
Compute Datatype	float16

Table 1: Quantization Settings

LoRA Settings	
Target Modules	q_proj, v_proj
Rank / Alpha	4 / 16
Dropout	0.05

Table 2: LoRA Settings

LoRA (Hu et al., 2022) with 4-bit quantization.

We used the BitsAndBytes library (Dettmers et al., 2023) (Dettmers et al., 2022a) (Dettmers et al., 2022b) for quantization and the PEFT library (Mangrulkar et al., 2022) for applying LoRA.

The training was executed using the Trainer class from the Transformers library (Wolf et al., 2020).

Table 1, Table 2, and Table3 show the specific quantization settings, LoRA settings, and hyperparameters used in QLoRA, respectively. Table 4 shows the prompt used for the initial translation model.

Preference Learning We conducted preference learning using QLoRA after supervised fine-tuning.

For preference learning, we adopted the SimPO (Meng et al., 2024) method. SimPO is a method that is efficient while suppressing redundant sentence generation. We used the trl library (von Werra et al., 2020) for implementation.

The quantization settings, LoRA settings, and prompts were the same as those used for supervised fine-tuning. Table 3 shows the hyperparameters used for preference learning.

Hereafter, we refer to the model trained with supervised fine-tuning only as $\mathrm{INIT}_{\mathrm{SFT}}$ and with both supervised fine-tuning and preference learning as $\mathrm{INIT}_{\mathrm{SimPO}}$.

4 Automatic Post-Editing Model

4.1 Datasets

For training the automatic post-editing model, we need a triplet consisting of the source text, preferred translation, and non-preferred translation,

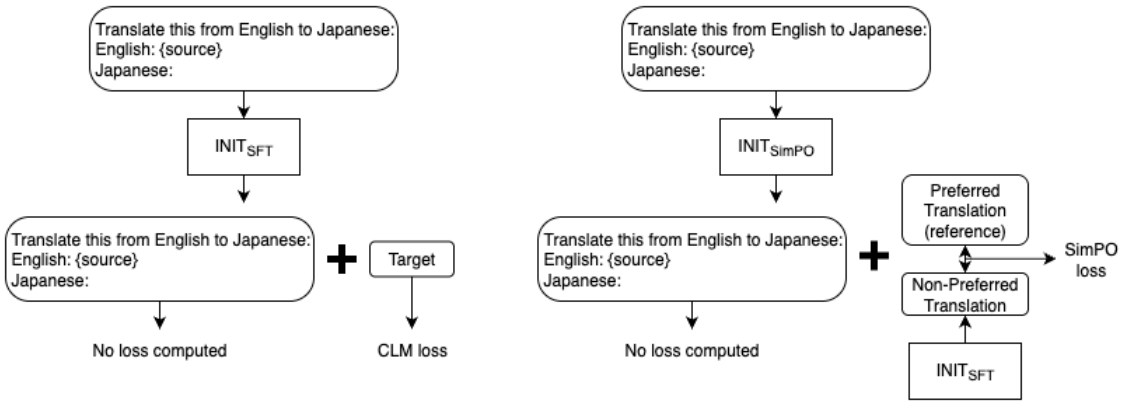

Figure 2: Left: Supervised fine-tuning step of the initial translation model; Right: Preference learning step of the initial translation model.

Common Settings	
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e-08$)
Gradient Clipping	1.0
Batch Size	1
Gradient Accumulation	64
Epochs	1
Supervised Fine-Tuning Settings	
Learning Rate	5e-05
SimPO Settings	
Learning Rate	1e-06
Alpha	1.0
Beta	0.1
Gamma	0.5
Context Length	1024

Table 3: Hyperparameter Settings

Translate this from English to Japanese:
English: {source}
Japanese:

Table 4: Prompt for Initial Translation Model

similar to preference learning. We used the following two datasets:

The first dataset is the same dataset used for preference learning of the initial translation model.

The second dataset is a dataset where the source text and preferred translation are the same as in the first dataset, but the non-preferred translation is not the output of the SFT model, but rather the output of the preference learning model.

4.2 Model Selection

We employed the same pre-trained model as the initial translation model, which is cyberagent/DeepSeek-R1-Distill-Qwen-14B-Japanese.

4.3 Training Procedure

The training procedure of the automatic post-editing model is shown in Figure 3.

We conducted supervised fine-tuning of the automatic post-editing model using QLoRA.

The automatic post-editing model was trained separately on each of the two datasets described above. The quantization settings, LoRA settings, and hyperparameters used for QLoRA were the same as those used for the initial translation model. Table 5 shows the prompt used for the automatic post-editing model.

Hereafter, we refer to the model trained with supervised fine-tuning on the first dataset as $\mathrm{PEDIT}_{\mathrm{SFT}}$ and on the second dataset as $\mathrm{PEDIT}_{\mathrm{SimPO}}$.

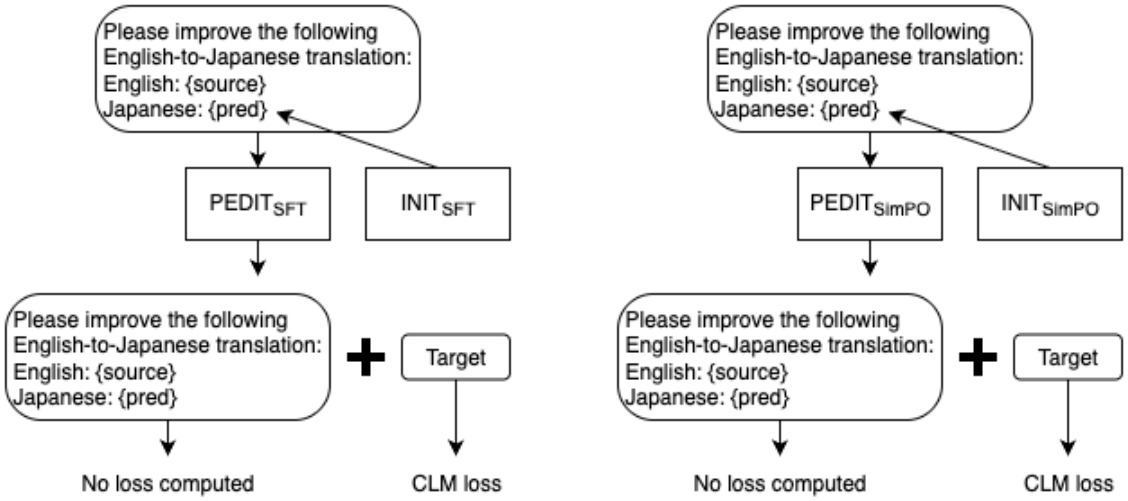

Figure 3: Left: Supervised fine-tuning step of $\mathrm{PEDIT}_{\mathrm{SFT}}$; Right: Supervised fine-tuning step of $\mathrm{PEDIT}_{\mathrm{SimPO}}$

Please improve the following English-to-Japanese translation:
English: {source}
Japanese: {pred}

Table 5: Prompt for Automatic Post-Editing Model

5 Reranking

5.1 Method

MBR decoding is a selection strategy that selects the candidate with the maximum expected utility from a candidate set.

This strategy takes the form of reranking in practice and is formulated as follows:

$$\hat{y}_i = \underset{c_i \in \mathcal{C}}{\operatorname{argmax}} \frac{1}{|\mathcal{C}|} \sum_{j=1}^{|\mathcal{C}|} u(c_i, c_j)$$

where $\hat{y}_i$ is the selected candidate, c_i is a candidate in the candidate set $\mathcal{C}$, and $u(c_i, c_j)$ is the utility function between candidates c_i and c_j.

In machine translation, reference-based evaluation metrics (e.g., BLEU, COMET) are often used as the utility function. That is, MBR decoding is defined as a strategy that selects the candidate with the highest average utility with respect to the other candidates in the set.

5.2 Reranking Procedure

Candidate Generation First, we generated a candidate set. We used four combinations of models: $\mathrm{INIT}_{\mathrm{SFT}}$, $\mathrm{INIT}_{\mathrm{SimPO}}$, $\mathrm{PEDIT}_{\mathrm{SFT}}(\mathrm{INIT}_{\mathrm{SFT}})$, and $\mathrm{PEDIT}_{\mathrm{SimPO}}(\mathrm{INIT}_{\mathrm{SimPO}})$.

Here, $\mathrm{PEDIT}_{\mathrm{SFT}}(\mathrm{INIT}_{\mathrm{SFT}})$ refers to the output of $\mathrm{PEDIT}_{\mathrm{SFT}}$ when given the output of $\mathrm{INIT}_{\mathrm{SFT}}$ as input and $\mathrm{PEDIT}_{\mathrm{SimPO}}(\mathrm{INIT}_{\mathrm{SimPO}})$ refers to the output of $\mathrm{PEDIT}_{\mathrm{SimPO}}$ when given the output of $\mathrm{INIT}_{\mathrm{SimPO}}$ as input.

We generated two translations for each model using two decoding strategies: Greedy decoding and Temperature 0.9 + Top-p 0.6 + Top-k 50.

The Greedy decoding is a decoding strategy that selects the most probable token at each step, while the Temperature 0.9 + Top-p 0.6 + Top-k 50 is a sampling-based decoding strategy that introduces randomness in the selection of tokens.

Therefore, the number of candidates generated for each input is $2+2+2\times2+2\times2 = 12$ because $\mathrm{INIT}_{\mathrm{SFT}}$ and $\mathrm{INIT}_{\mathrm{SimPO}}$ each generate two translations, and $\mathrm{PEDIT}_{\mathrm{SFT}}$ and $\mathrm{PEDIT}_{\mathrm{SimPO}}$ each generate two translations for each of the outputs of $\mathrm{INIT}_{\mathrm{SFT}}$ and $\mathrm{INIT}_{\mathrm{SimPO}}$, respectively.

Reranking Next, we applied MBR decoding to the generated candidate set. The utility function used is a linear combination of the following:

$$0.8 \times \text{COMET-22} + 0.2 \times \text{LaBSE-cos}$$

Where LaBSE-cos is the cosine similarity based on LaBSE. The combination of these utility functions is inspired by the winning system in the WMT24 General Machine Translation Task (Kondo et al., 2024).

6 Experiment and Analysis

We used the WMT24++ (Deutsch et al., 2025) as the evaluation dataset. We evaluated the system using automatic metrics, specifically COMET-22 and BLEU (Papineni et al., 2002). We used sacreBLUE (Post, 2018) for BLEU calculation.

We evaluated zero-shot performance of the base model as a baseline. In addition to the submitted system, we compared the following four model configurations: $\mathrm{INIT}_{\mathrm{SFT}}$, $\mathrm{INIT}_{\mathrm{SimPO}}$, $\mathrm{PEDIT}_{\mathrm{SFT}}(\mathrm{INIT}_{\mathrm{SFT}})$, and $\mathrm{PEDIT}_{\mathrm{SimPO}}(\mathrm{INIT}_{\mathrm{SimPO}})$.

For baseline, $\mathrm{INIT}_{\mathrm{SFT}}$ and $\mathrm{INIT}_{\mathrm{SimPO}}$, we used the same prompt as the one used during the training of the initial translation model. For $\mathrm{PEDIT}_{\mathrm{SFT}}$ and $\mathrm{PEDIT}_{\mathrm{SimPO}}$, we used the same prompt as the one used during the training of the automatic post-editing model.

Since the baseline outputs think tokens, we removed the text enclosed in <think> tags using regular expressions during evaluation.

Except for the submitted system, we used the default decoding strategy of the base model: Temperature 0.6 and Top-p 0.95.

Table 6 shows the results of the automatic evaluation. For BLEU, $\mathrm{INIT}_{\mathrm{SFT}}$ achieved the highest score, followed by the submitted system. For COMET-22, however, the submitted system scored the highest. In both metrics, the baseline had the lowest score.

The baseline achieved a significantly lower BLEU score, possibly because its outputs often contained extraneous information in addition to the translation (see Table 7). On the other hand, after SFT, cases where extraneous information other than the translation was included in the output were rarely observed. Therefore, the reason for the improvement in score after SFT is thought to be that the improvement in output consistency worked favorably for automatic evaluation.

Also, when automatic post-editing was applied, the BLEU score decreased, but the COMET-22 score improved. This may be because there was only one reference sentence when calculating the BLEU score this time, and the automatic post-editing, which generates diverse expressions, led to a decrease in the BLEU score. On the other hand, since COMET-22 is tolerant of diverse expressions, it is thought that the COMET-22 score improved when automatic post-editing was applied. In fact, when we checked some translation examples, we found that there were few cases where the quality significantly deteriorated when automatic post-editing was applied.

BLEU	
Baseline	1.34
$\mathrm{INIT}_{\mathrm{SFT}}$	49.89
$\mathrm{INIT}_{\mathrm{SimPO}}$	48.11
$\mathrm{PEDIT}_{\mathrm{SFT}}(\mathrm{INIT}_{\mathrm{SFT}})$	47.96
$\mathrm{PEDIT}_{\mathrm{SimPO}}(\mathrm{INIT}_{\mathrm{SimPO}})$	46.61
Submitted System	49.45
COMET-22	
Baseline	0.6696
$\mathrm{INIT}_{\mathrm{SFT}}$	0.8368
$\mathrm{INIT}_{\mathrm{SimPO}}$	0.8453
$\mathrm{PEDIT}_{\mathrm{SFT}}(\mathrm{INIT}_{\mathrm{SFT}})$	0.8451
$\mathrm{PEDIT}_{\mathrm{SimPO}}(\mathrm{INIT}_{\mathrm{SimPO}})$	0.8471
Submitted System	0.8761

Table 6: Results of Automatic Evaluation

The submitted system achieved a highest score in COMET-22. This is thought to be due to MBR decoding, which can suppress quality degradation caused by automatic post-editing.

Based on these results, we confirm that supervised fine-tuning, preference learning, and the introduction of the automatic post-editing model are effective.

7 Conclusion

In this paper, we described the system submitted by Team SH to the WMT2025 General Machine Translation Task. We developed the initial translation model through supervised fine-tuning and preference learning, and the automatic post-editing model through supervised fine-tuning. We generated multiple translations from these models and selected the best translation using MBR decoding. The submitted system demonstrated superior performance compared to the baseline in both automatic evaluation metrics, namely BLEU and COMET-22.

References

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

Input:
A final push for female equality
Output:
女性の平等の推進に向けた最後の努力
（※より自然な日本語表現のため、原文の「final push」を「最後の努力」とし、「female equality」を「女性の平等の推進」と訳出しています）

【補足】
- 「final push」を「最後の追い込み」と訳す場合：
「女性の平等のための最後の追い込み」
- 「女性の平等」を「女性の平等実現」とし、よりアクティブな表現にすると：
「女性の平等実現への最後の総力戦」

文脈やトーンに応じて使い分けると自然です。

Table 7: Example of baseline output containing extraneous information

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.

DeepSeek-AI et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022a. Llm.int8(): 8-bit matrix multiplication for transformers at scale. *arXiv preprint arXiv:2208.07339*.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 2022b. 8-bit optimizers via block-wise quantization. *9th International Conference on Learning Representations, ICLR*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Daniel Deutsch, Eleftheria Briakou, Isaac Caswell, Mara Finkelstein, Rebecca Galor, Juraj Juraska, Geza Kovacs, Alison Lui, Ricardo Rei, Jason Riesa, Shruti Rijhwani, Parker Riley, Elizabeth Salesky, Firas Trabelsi, Stephanie Winkler, Biao Zhang, and Markus Freitag. 2025. Wmt24++: Expanding the language coverage of wmt24 to 55 languages & dialects.

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.

Patrick Fernandes, António Farinhas, Ricardo Rei, José G. C. de Souza, Perez Ogayo, Graham Neubig, and Andre Martins. 2022. Quality-aware decoding for neural machine translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1396–1412, Seattle, United States. Association for Computational Linguistics.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Ryosuke Ishigami. 2025. Deepseek-r1-distill-qwen-14b-japanese.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Masaaki Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović, Mariya Shmatova, and Jun Suzuki. 2023. Findings of the 2023 conference on machine translation (WMT23): LLMs are here but not quite there yet. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1–42, Singapore. Association for Computational Linguistics.

Minato Kondo, Ryo Fukuda, Xiaotian Wang, Katsuki Chousa, Masato Nishimura, Kosei Buma, Takatomo Kano, and Takehito Utsuro. 2024. NTTSU at WMT2024 general translation task. In *Proceedings of the Ninth Conference on Machine Translation*, pages 270–279, Miami, Florida, USA. Association for Computational Linguistics.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. PEFT: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

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

Pennsylvania, USA. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

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. 2022. 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.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. 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.