

# Enhancing Translation Accuracy of Large Language Models through Continual Pre-Training on Parallel Data

Minato Kondo<sup>1</sup> Takehito Utsuro<sup>1</sup> Masaaki Nagata<sup>2</sup>

<sup>1</sup>Deg. Prog. Sys.&Inf. Eng., Grad. Sch. Sci.&Tech., University of Tsukuba

<sup>2</sup>NTT Communication Science Laboratories, NTT Corporation, Japan

s2320743@u.tsukuba.ac.jp, utsuro@iit.tsukuba.ac.jp,

masaaki.nagata@ntt.com

## Abstract

In this paper, we propose a two-phase training approach where pre-trained large language models are continually pre-trained on parallel data and then supervised fine-tuned with a small amount of high-quality parallel data. To investigate the effectiveness of our proposed approach, we conducted continual pre-training with a 3.8B-parameter model and parallel data across eight different formats. We evaluate these methods on thirteen test sets for Japanese-to-English and English-to-Japanese translation. The results demonstrate that when utilizing parallel data in continual pre-training, it is essential to alternate between source and target sentences. Additionally, we demonstrated that the translation accuracy improves only for translation directions where the order of source and target sentences aligns between continual pre-training data and inference. In addition, we demonstrate that the LLM-based translation model is more robust in translating spoken language and achieves higher accuracy with less training data compared to supervised encoder-decoder models. We also show that the highest accuracy is achieved when the data for continual pre-training consists of interleaved source and target sentences and when tags are added to the source sentences.

## 1 Introduction

In machine translation, transformer encoder-decoder models (Vaswani et al., 2017), such as NLLB-200 (NLLB Team et al., 2022), mT5 (Xue et al., 2021), and mBART (Liu et al., 2020) predominate. The emergence of pre-trained Large Language Models (LLMs) composed solely of the transformer decoder, such as GPT series (Brown et al., 2020; OpenAI, 2023), has prompted the development of pre-trained LLMs, including, PaLM (Chowdhery et al., 2022), and LLaMA (Touvron et al., 2023). When translating with these LLMs, it is common to use in-context few-shot

learning. According to Hendy et al. (2023), GPT-3 demonstrates comparable or superior accuracy to WMT-best for high-resource languages. Furthermore, as reported by Kocmi et al. (2023), GPT-4’s 5-shot surpasses WMT-best’s accuracy in most translation directions. However, Zhu et al. (2024) noted that in 8-shot scenarios, relatively small-scale LLMs (e.g., 7B parameters) exhibit lower accuracy than supervised encoder-decoder models. Therefore, it is necessary to investigate methods capable of achieving translation accuracy equivalent to existing translation models with relatively small-scale LLMs.

On the other hand, in models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), which consist solely of transformer encoders, the effectiveness of continual pre-training, where pre-trained models are further trained on task-specific data such as classification to improve the accuracy of the task, has been reported (Jin et al., 2022; Ke et al., 2022). In the context of LLMs, continual pre-training has been reported to transfer models primarily pre-trained in English, such as LLaMA, to other languages (Cui et al., 2023). Additionally, when building LLM-based translation models, the effectiveness of conducting continual pre-training with either monolingual data, parallel data, or both, followed by supervised fine-tuning, has been reported, mainly when basing the model on primarily English pre-trained models such as LLaMA-2 (Xu et al., 2024a; Alves et al., 2024; Guo et al., 2024).

Although those recent publication in the context of LLMs are closely related to our study, this paper presents research conducted independently of those latest LLM-based translation studies such as Xu et al. (2024a); Alves et al. (2024); Guo et al. (2024). This paper proposes a two-phase training approach: continual pre-training on parallel data crawled from the web and supervised fine-tuning using a small amount of high-quality parallel data created by professional translators. To comprehen-

sively investigate methods for improving translation accuracy through continual pre-training, we conduct continual pre-training across eight data formats for Japanese-to-English (Ja  $\Rightarrow$  En) and English-to-Japanese (En  $\Rightarrow$  Ja) translations using a 3.8B-parameter LLM. We evaluate the translation accuracy on 13 test sets. Our paper’s novelty compared to Xu et al. (2024a); Alves et al. (2024); Guo et al. (2024) lies in the following aspects.

- When conducting continual pre-training on data where source and target sentences appear alternately, the direction of language in which accuracy improves varies depending on the order of source and target sentences.
- LLM-based translation model is more robust in translating spoken language and achieves higher accuracy with less training data compared to supervised encoder-decoder models.
- When indicating the translation direction with tags (“<2en>” etc.) on data for continual pre-training, higher accuracy is achieved compared to simply concatenating source and target sentences.

## 2 Related Work

### Parallel Data in Pre-Training from Scratch

When pre-training LLMs from scratch, it is expected to use monolingual data. However, some reports incorporating bilingual data, such as parallel data, into the pre-training dataset can enhance the accuracy of downstream tasks. Briakou et al. (2023) show that incorporating parallel data into pre-training 1B and 8B parameter LLMs enhances translation accuracy in zero- and five-shot. Separate studies further show that including parallel data in the pre-training of the encoder-decoder model also improved performance in downstream multilingual and cross-lingual tasks (Kale et al., 2021; Schioppa et al., 2023).

**LLMs-Based Translation Models** Zhang et al. (2023) demonstrated that fine-tuning 15 multilingual LLMs using QLoRA for French-to-English translation surpasses the accuracy of both in-context few-shot learning and models trained from scratch. Conversely, Xu et al. (2024a) demonstrated that models predominantly pre-trained on English data, such as LLaMA-2, suffer reduced translation accuracy when translating into non-English target languages. Addressing this issue,

they introduced ALMA, a method that employs fine-tuning monolingual data in the first stage, followed by supervised fine-tuning with a small quantity of high-quality parallel data in the second stage. Furthermore, there exists a report on improving translation accuracy by employing Contrastive Preference Optimization (CPO) for the second stage of supervised fine-tuning in ALMA (Xu et al., 2024b). In addition, the effectiveness of utilizing monolingual and parallel data in the first stage has been reported (Alves et al., 2024; Guo et al., 2024).

LLM-based translation models have only been evaluated on test data from the WMT General Machine Translation Task (Kocmi et al., 2022, 2023) and Flores-200 (NLLB Team et al., 2022). Therefore, their effectiveness compared to conventional supervised encoder-decoder models has not been sufficiently validated across various types of data. Additionally, the impact of continual pre-training data on translation accuracy remains unclear. Our study aims to address these two points.

## 3 Continual Pre-Training and Supervised Fine-Tuning with Parallel Data

We introduce a two-phase training to enhance the accuracy of translation of LLMs. In the first phase, we perform continual pre-training using parallel data crawled from the web, such as ParaCrawl (Bañón et al., 2020). Then, in the second phase, we conduct supervised fine-tuning with a small amount of high-quality parallel data. In LLM fine-tuning, the importance of data quality has been reported (Xu et al., 2024a; Zhou et al., 2023). However, it has also been reported that parallel data crawled from the web may have low data quality (Thompson et al., 2024). Therefore, we used data created by professional translators as high-quality parallel data.

### 3.1 Continual Pre-Training

Continual pre-training involves training on data where the source and target sentences appear alternately. Let the source sentences be denoted as  $\{x_1, \dots, x_n\}$  and the target sentences as  $\{y_1, \dots, y_n\}$ , creating a dataset  $\{x_1, y_1, \dots, x_n, y_n\}$ . With the tokens of the created dataset represented as  $\mathbf{z} = \{z_1, z_2, \dots, z_m\}$ , we train the model parameters  $\theta$  to minimize the following loss:

$$\mathcal{L}_1(\theta) = - \sum_t \log P(z_t | z_{t-c}, \dots, z_{t-1}; \theta) \quad (1)$$

where  $c$  is the number of context lengths representing the maximum input length of LLMs.  $\mathcal{L}_1(\theta)$  is a standard causal language modeling loss, which predicts the next word based on previous words (Radford et al., 2018). Therefore, we train by extracting  $(z_{t-c}, \dots, z_{t-1})$  from  $\mathbf{z}$  in increments of  $c$  tokens, such that the source and target sentences alternate to predict the next word for each token. Extracting  $c$  tokens may result in the input’s start and end being in the middle of the source or target sentence.

In pre-trained models such as LLaMA-2, primarily pre-trained in English, it has been reported that the effectiveness of utilizing monolingual data in continual pre-training, in addition to parallel data, is significant (Guo et al., 2024; Alves et al., 2024). The rationale behind continual pre-training with monolingual data is to acquire the generative ability in languages other than English. Therefore, in models where pre-training with monolingual data has been sufficiently conducted from scratch or where continual pre-training with monolingual data has been conducted, it is optional to conduct continual pre-training with monolingual data.

### 3.2 Supervised Fine-Tuning

After continual pre-training, we perform supervised fine-tuning with a small amount of high-quality parallel data. Let the source sentence be denoted by  $\mathbf{x}$ , the target sentence corresponding to  $\mathbf{x}$  by  $\mathbf{y}$ , and the prompt by  $I(\mathbf{x})$ . We train the model parameters to minimize the following loss:

$$\mathcal{L}_2(\theta) = - \sum_{t=1}^T \log P(\mathbf{y}_t | \mathbf{y}_{<t}, I(\mathbf{x}); \theta) \quad (2)$$

where  $T$  represents the number of tokens in the target sentence, and  $\mathbf{y}_t$  is the  $t$ -th token of the target sentence. While  $\mathcal{L}_2(\theta)$  is also standard causal language modeling loss, it computes the loss only for the output of the target sentence (Xu et al., 2024a; Zhang et al., 2024). Therefore, we combine the prompt and target sentence (e.g., Translate “Good morning” into Japanese: おはよう) and input it into the model. The model predicts the next word for all input words, including the prompt portion. However, this portion is not used during inference and hence excluded from the loss.

## 4 Experiments

We conduct experiments on two NVIDIA RTX A6000 GPUs. Due to severely limited com-

putational resources, we use a 3.8B parameters LLM, rinna/bilingual-gpt-neox-4b (rinna-4b)<sup>1</sup>, which is already pre-trained on Japanese and English data, totaling 524B tokens, with 173B tokens in Japanese and 293B tokens in English. Since rinna-4b has undergone sufficient pre-training from scratch on monolingual data for both Japanese and English, as stated in Section 3, we believe that continual pre-training with monolingual data is unnecessary. Given that the model we employ is pre-trained on Japanese and English, we experiment with Japanese-to-English and English-to-Japanese translation tasks. All experiments utilizing rinna-4b are conducted using the open-source huggingface transformers library.<sup>2</sup>

### 4.1 Dataset

#### 4.1.1 Continual Pre-Training

We utilize JParaCrawl v3.0 (Morishita et al., 2022) as the web-based parallel data comprising 21.8M parallel sentence pairs, the largest and newest dataset of English-Japanese parallel data available. From this dataset of 21.8M parallel sentence pairs, we sample 20.8M sentence pairs using LEALLA-large<sup>3</sup> (Mao and Nakagawa, 2023) for train data. Details on sampling are provided in Appendix A. For dev data, we use the dev and test data from WMT20 (Barrault et al., 2020) and the test data from WMT21 (Akhbardeh et al., 2021).

#### 4.1.2 Supervised Fine-Tuning

We utilize the dev and test data of WMT20 and Flores-200 (NLLB Team et al., 2022), along with the train data from KFTT (Neubig, 2011) as train data, all created by professional translators. The train data for KFTT utilized in experiments consists of 10k instances randomly sampled from 440k samples. The resulting train data comprise 15k samples for both En  $\Rightarrow$  Ja and Ja  $\Rightarrow$  En. For dev data, we utilize the WMT21 test data. We use the prompts written in the following source language, based on the report by Xu et al. (2024a).<sup>4</sup>

#### En $\Rightarrow$ Ja

Translate this from English to Japanese:

English: {source sentence}

Japanese:

<sup>1</sup><https://huggingface.co/rinna/bilingual-gpt-neox-4b>

<sup>2</sup><https://github.com/huggingface/transformers>

<sup>3</sup><https://huggingface.co/setu4993/LEALLA-large>

<sup>4</sup>The reason for writing prompts in the source sentence’s language is that it is more natural to create translation prompts in the source sentence’s language when translating.

## Ja ⇒ En

これを日本語から英語に翻訳してください：

日本語：{source sentence}

英語：

### 4.1.3 Test Sets

We use the test sets employed by [Morishita et al. \(2022\)](#) to evaluate translation accuracy. Since we include the test data from WMT20 and WMT21 in the train and dev data for continual pre-training and supervised fine-tuning, we exclude these and add the test data from WMT22. As a result, there are 13 test sets: 5 for the En ⇒ Ja direction, 3 for the Ja ⇒ En direction, and 5 for both the En ⇒ Ja and En ⇒ Ja directions. For detailed information on the test sets, please refer to Table 6 of Appendix D.

## 4.2 Models

### 4.2.1 Baseline Models

We establish two baseline models as described below. The train data consists of the data described in Section 4.1.1 and Section 4.1.2, and the dev data is the WMT21 test data. Note that the data from JParaCrawl v3.0 is created by randomly sampling 10.4M parallel sentences, which is 50% of the total 20.8M parallel sentences, to be used respectively as the train data for En ⇒ Ja and Ja ⇒ En.

**Transformer** This model is a 1B-parameter transformer trained from scratch. The model architecture is based on mT5-large<sup>5</sup>, with two modifications: reducing the vocab\_size from 250,112 to 65,536, matching that of rinna-4b, and increasing the feed-forward network dimension from 2,816 to 4,096. As a result, the model has 24 layers each for the encoder and decoder, a model dimension of 1,024, 16 attention heads, a feed-forward network with GeGLU activation ([Shazeer, 2020](#)), and a dropout ([Srivastava et al., 2014](#)) of 0.1. The tokenizer is newly created using the sentencepiece library<sup>6</sup> ([Kudo and Richardson, 2018](#)) with the subword method set to unigram, character coverage to 0.9995, and byte-fallback enabled. Training is conducted with a total batch size of 4,096 for 15 epochs (38,160 steps), with validation every 1,000 steps, and it is terminated if the validation loss does not improve for three consecutive validations. We use AdamW optimizer ([Loshchilov and Hutter, 2019](#)), with  $\beta_1 = 0.9$ ,  $\beta_2 = 0.98$ ,  $\epsilon = 1.0 \times 10^{-8}$ . We set the weight decay and label smoothing ([Szegedy](#)

[et al., 2016](#)) to 0.1, and gradient clipping ([Pascanu et al., 2013](#)) to 1.0. The peak learning rate is set to  $1.0 \times 10^{-3}$ , with a warmup ratio 0.1 and an inverse square root scheduler applied. Additionally, bfloat16, gradient checkpointing ([Chen et al., 2016](#)), and the deepspeed<sup>7</sup> ([Rasley et al., 2020](#)) ZeRO stage 2 are applied during training. Training is conducted on two NVIDIA RTX A6000 GPUs with these settings, taking 17 days.

**Direct-SFT** Direct-SFT consists of the rinna-4b directly supervised fine-tuning with parallel data, using LoRA tuning ([Hu et al., 2022](#)). We conduct supervised fine-tuning of this model using the prompts mentioned in Section 4.1.2. Furthermore, to approximate conditions for full-weight tuning, we apply LoRA to the linear layers of self-attention’s query, key, value, and output, as well as the linear layers of the feed-forward network. We set the rank of LoRA to 16, resulting in 25.9M trainable parameters, which constitutes 0.68% of the parameters in rinna-4b.

### 4.2.2 Source and Target Sentences Ordering in Continual Pre-Training

We conduct continual pre-training with 4 patterns, varying the order in which source and target sentences. After continual pre-training with these 4 orders, we undergo supervised fine-tuning using the data and prompts described in Section 4.1.2 with full fine-tuning and LoRA tuning.

**Mono** As stated in Section 3.1, instead of alternating between source and target sentences, the approach involves sequences such as  $(x_1, \dots, x_n), (y_1, \dots, y_n)$ , where only the source or target sentences appear consecutively. Therefore, either Japanese-only or English-only sentences appear consecutively.

**En-Ja** Concatenating a Japanese translation immediately after each English sentence, making it parallel data only in the En ⇒ Ja.

**Ja-En** Concatenating an English translation immediately after each Japanese sentence, making it parallel data only in the Ja ⇒ En.

**Mix** Randomly sampling 10.4M, which is 50% from the total of 20.8M, from the En-Ja and Ja-En without duplication.

<sup>5</sup><https://huggingface.co/google/mt5-large>

<sup>6</sup><https://github.com/google/sentencepiece>

<sup>7</sup><https://github.com/microsoft/DeepSpeed>



Metrics		Baseline models		Continual pre-training + Supervised fine-tuning							
		Transformer	Direct-SFT	Mono		En-Ja		Ja-En		Mix	
				full	LoRA	full	LoRA	full	LoRA	full	LoRA
BLEU	Avg.	13.9	12.2	6.3	5.9	15.4	<b>15.5</b>	7.3	7.2	14.7	14.9
	# Sig.	-	1	0	0	8	9	0	0	7	8
COMET	Avg.	79.0	79.6	75.6	74.8	<b>83.5</b>	83.3	76.9	76.8	82.9	82.9
	# Sig.	-	7	0	0	8	8	0	0	8	8

(a) En  $\Rightarrow$  Ja

Metrics		Baseline models		Continual pre-training + Supervised fine-tuning							
		Transformer	Direct-SFT	Mono		En-Ja		Ja-En		Mix	
				full	LoRA	full	LoRA	full	LoRA	full	LoRA
BLEU	Avg.	<b>17.3</b>	12.5	7.9	7.1	7.8	7.6	17.0	17.0	15.9	15.8
	# Sig.	-	0	0	0	0	0	0	0	0	0
COMET	Avg.	76.4	75.0	70.4	69.7	70.3	70.0	<b>77.8</b>	<b>77.7</b>	77.1	76.9
	# Sig.	-	0	0	0	0	0	7	6	5	5

(b) Ja  $\Rightarrow$  En

Table 1: Results of Baseline models and models continually pre-trained with four orders described in Section 4.2.2 then supervised fine-tuning. ‘‘Avg.’’ represents the average result across 12 test sets, ‘‘# Sig.’’ indicates the number of test sets showing significant differences from Transformer ( $p < 0.05$ ), and full represents full fine-tuning. **Bold numbers** represent the highest scores in each line, and scores that surpass the Transformer are emphasized in green .

### 4.3 Hyperparameters

#### 4.3.1 Continual Pre-Training

We use the AdamW optimizer, with  $\beta_1 = 0.9$ ,  $\beta_2 = 0.95$ ,  $\epsilon = 1.0 \times 10^{-8}$ . The context length is 2048, the same as when pre-training rinna-4b from scratch, and training is conducted for 1 epoch. We perform validation every 100 training steps. We use a cosine learning rate schedule with a warmup ratio of 1% and a peak learning rate of  $1.5 \times 10^{-4}$ . We use a weight decay of 0.1 and gradient clipping of 1.0. We utilize two NVIDIA RTX A6000 GPUs, processing 1 batch on each GPU with a gradient accumulation step of 128, achieving an adequate batch size 256. During training, bfloat16 precision, gradient checkpointing, and deepspeed ZeRO stage 2 are employed. With these configurations, it takes 10 days.

#### 4.3.2 Supervised Fine-tuning

We perform supervised fine-tuning on the model that achieves the minimum validation loss in continual pre-training. We change the AdamW optimizer’s parameter used in Section 4.3.1 only  $\beta_2 = 0.95$  to  $\beta_2 = 0.999$ . Weight decay and gradient clipping are the same as Section 4.3.1. The peak

learning rate is set to  $3.0 \times 10^{-5}$  for full fine-tuning and  $2.0 \times 10^{-4}$  for LoRA tuning, with a warmup ratio of 1% using an inverse square schedule. For LoRA, we set  $r = 16$ ,  $\alpha = 32$ , and dropout to 0.05, applying to the linear layers of query, key, and value in the multi-head attention, resulting in approximately 6.4M trainable parameters corresponding to 0.17% of the rinna-4b’s parameters. We conduct validation every 10% of the total training steps for Direct-SFT only, with 1 epoch and a batch size of 256. For all other cases, validation is performed every 100 training steps, with 5 epoch and the batch size of 64.

#### 4.3.3 Inference

All models use the one with the minimum validation loss for inference, applying bfloat16. The Transformer, which has fewer parameters than the rinna-4b, employs beam search with a beam size of 4 due to its smaller number of parameters. At the same time, the rinna-4b-based models use greedy decoding with the prompt described in Section ?? for inference.

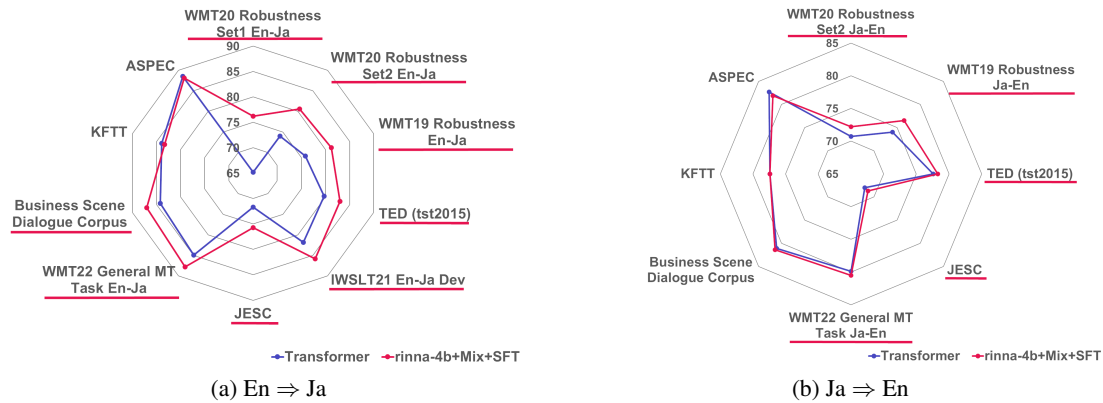


Figure 1: Radar chart of COMET score. Blue line indicates the accuracy of the Transformer, while red line represents the accuracy of the model continually pre-trained with Mix format followed by supervised fine-tuning with full weight. Underlines indicate test sets with a significant difference compared to the Transformer ( $p < 0.05$ ).

#### 4.4 Metrics

We use BLEU<sup>8</sup> (Papineni et al., 2002) and COMET<sup>9</sup> (Rei et al., 2022) as evaluation metrics. We use Unbabel/wmt22-comet-da as COMET model.

### 5 Results

#### 5.1 The Impact of Source and Target Sentences Order

Table 1 presents the results of baseline models compared to models continually pre-trained with three orders described in Section 4.2.2 and then supervised fine-tuning. All results of Table 1 can be found in Table 7 and Table 8 of Appendix D. Direct-SFT, which is directly fine-tuned, and Mono, which was pre-trained with parallel data treated as monolingual data, exhibit lower accuracy than the Transformer. On the other hand, continual pre-training improves accuracy only in the translation direction aligned with the parallel data. Therefore, continual pre-training with data where source and target sentences appear alternately is necessary to achieve high accuracy. Despite data in both the En  $\Rightarrow$  Ja and Ja  $\Rightarrow$  En translation directions, Mix exhibits improved accuracy, even though the input data’s translation direction is inconsistent. This result suggests that LLMs can leverage the knowledge of the translation direction matching the order of the source and target sentences, and they can utilize the knowledge acquired from parallel sentences mixed in the training data.

<sup>8</sup><https://github.com/mjpost/sacrebleu>

<sup>9</sup><https://github.com/Unbabel/COMET>

#### 5.2 Accuracy Comparison Across Test Sets

Figure 1 shows a radar chart of the COMET score of a model in which the Transformer and rinna-4b are continually pre-trained as a Mix, followed by supervised fine-tuning with full weight. In particular, the LLM-based translation model significantly outperforms the Transformer on the WMT19, 20 Robustness Task for the Reddit domain, and on the TED (tst2015), IWSLT21 En-Ja Dev, and JESC for the TED Talk and movie subtitles domains. This result suggests that the LLM-based translation model is more robust than the traditional encoder-decoder model regarding data containing spoken language.

### 6 Discussion

#### 6.1 Data Format in Continual Pre-Training

Mixing data from two translation directions, as in the case of Mix, improves accuracy for both translation directions, allowing one model to be used for both. However, the accuracy is lower than continual pre-training with data from only one translation direction. Therefore, we investigate methods to enhance translation accuracy by explicitly indicating the translation direction for the data used in continual pre-training. Drawing inspiration from studies incorporating parallel data during pre-training from scratch, we conduct experiments on the following four formats.

**Interleaved Translations** This format directly concatenates the source and target sentences (Brikou et al., 2023), identical to the Mix described in Section 4.2.2.

**Prefixed** This format involves inserting the prefix written in the source sentence’s language before the

Metrics	Transformer	Interleaved		Prefix		Tagged		JSON		
		full	LoRA	full	LoRA	full	LoRA	full	LoRA	
BLEU	Avg.	13.9	14.7	14.9	15.1	<b>15.3</b>	15.0	15.1	14.2	14.9
	# Sig.	-	-	-	4	4	<b>6</b>	2	1	1
COMET	Avg.	79.0	82.9	82.9	83.0	<b>83.3</b>	83.2	<b>83.3</b>	82.1	82.9
	# Sig.	-	-	-	0	<b>5</b>	4	4	0	1

(a) En  $\Rightarrow$  Ja

Metrics	Transformer	Interleaved		Prefix		Tagged		JSON		
		full	LoRA	full	LoRA	full	LoRA	full	LoRA	
BLEU	Avg.	16.8	15.9	15.8	16.3	16.1	16.2	<b>16.3</b>	15.5	15.8
	# Sig.	-	-	-	0	0	0	0	0	0
COMET	Avg.	76.4	77.1	76.9	<b>77.4</b>	77.2	77.3	77.2	76.9	76.9
	# Sig.	-	-	-	<b>3</b>	<b>3</b>	<b>3</b>	1	0	0

(b) Ja  $\Rightarrow$  En

Table 2: Results of Transformer and models continually pre-trained with four formats described in Section 6.1 then supervised fine-tuning. “# Sig.” denotes the number of test sets showing significant differences for both Transformer and models continually pre-trained with Interleaved Translations (Interleaved), followed by supervised fine-tuning using the same fine-tuning method ( $p < 0.05$ ). “Avg.,” **bold numbers**, and **green numbers** follow the same conventions as Table 1.

Continual pre-training	Supervised fine-tuning	En $\Rightarrow$ Ja (Average)		Ja $\Rightarrow$ En (Average)	
		BLEU	COMET	BLEU	COMET
×	×	0.6	40.2	0.8	46.0
✓	×	8.2	69.9	9.9	69.3
×	✓	6.5	76.4	8.0	70.9
✓	✓	<b>15.0</b>	<b>83.2</b>	<b>16.2</b>	<b>77.3</b>

Table 3: Results of all combinations of continual pre-training and supervised fine-tuning. Continual pre-training is conducted in Tagged format mentioned in Section 6.1, and supervised fine-tuning is performed with full weight, utilizing the small amount of high-quality data and prompts described in Section 4.1.2. “✓” indicates whether continued pre-training or supervised fine-tuning is conducted. In contrast, “×” indicates the absence of either. **bold numbers** indicate the maximum score in each column. When supervised fine-tuning is conducted, inference is undergone with zero-shot, while inference is performed with five-shot for other cases.

source sentence, followed by the concatenation of the target sentence (Kale et al., 2021). For En  $\Rightarrow$  Ja, the prefix “translate to Japanese: ” is used, while for Ja  $\Rightarrow$  En, “英語に翻訳してください: ” is employed.

**Tagged** This format involves inserting a tag before the source sentence that indicates the target sentence’s language, such as “<2en>” and “<2ja>” (Schioppa et al., 2023).

**JSON** The JSON format is {“L1”: {source}, “L2”: {target}}, where “source” represents the source sentence, “target” represents the target sentence, and “L1”, “L2” are the names of the source and target sentence’s languages written in the

source sentence’s language.<sup>10</sup>

We conduct continual pre-training with these four formats and perform supervised fine-tuning under the same conditions as Section 4. All formats are conducted in the Mix format described in Section 4.2.2, with the continual pre-training data for En  $\Rightarrow$  Ja and Ja  $\Rightarrow$  En fixed to be the same. Table 2 presents the results of the Transformer and models continually pre-trained with the four formats. Among the four formats, Prefix and Tagged showed significant differences in BLEU and COMET metrics compared to the Transformer, and the models are continually pre-trained in the in-

<sup>10</sup>Given that the pre-training data for rinna-4b includes source code, this format aims to transfer the knowledge obtained from the source code to translation.

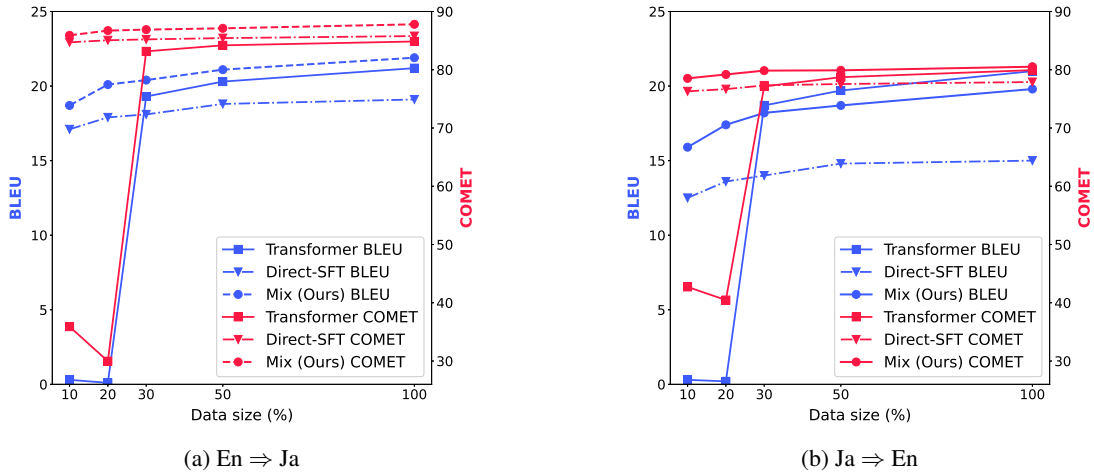


Figure 2: Data curves for BLEU and COMET scores on WMT22 test data for Transformer, Direct-SFT, and Mix. Mix has been evaluated after completing supervised fine-tuning with LoRA tuning following continual pre-training. We experimented with data amounts of 10%, 20%, 30%, 50%, and 100% due to computational resource constraints. For the Transformer, we varied the proportion of data from JParaCrawl v3.0. At the same time, for Direct-SFT and Mix, since training was conducted for only one epoch, we consider the proportion of checkpoints equal to that of the training data and report the accuracy for each checkpoint.

terleaved translations format. This result suggests that the prefixed or tagged format demonstrates higher accuracy than interleaved translation, where source and target sentences are concatenated, indicating that from the convenience perspective, the Tagged format can achieve the highest accuracy most easily. Whether these formats can be applied to other translation directions and models remains a matter of our future work.

## 6.2 Effectiveness of Continual Pre-Training and Supervised Fine-Tuning

As an ablation study, we experiment with all combinations of continual pre-training and supervised fine-tuning. When supervised fine-tuning is conducted, the inference is made with a zero-shot, while for other cases, the inference is performed with a five-shot. We randomly sample five translation examples from the WMT21 test data for five-shot, and the same set of five samples is fixed for all inferences. Continual pre-training is conducted in the Tagged format as described in Section 6.1, and all inferences utilize the prompts described in Section 4.1.2 and employ bfloat16 precision and greedy decoding. Table 3 presents the results, while all results are shown in Table 10 of Appendix D. These results suggest that achieving high accuracy is most feasible when both continual pre-training and supervised fine-tuning are conducted while achieving high accuracy solely through continual pre-training or supervised fine-tuning alone is chal-

lenging.

## 6.3 How Much Parallel Data is Needed?

Figure 2 presents the data curves for these three models at 10%, 20%, 30%, 50%, and 100% data usage on the WMT22 test data. For the Transformer model, only the sampling rate from JParaCrawl v3.0 varies, while other settings remain the same as described in Section 4.2.1. As mentioned in Section 4.3, Direct-SFT performs supervised fine-tuning for one epoch, and Mix also performs continual pre-training for one epoch. Therefore, for these two models, the proportion of training data is equivalent to the proportion of checkpoints, and we report the accuracy for the checkpoints at 10%, 20%, 30%, 50%, and 100%. The Transformer shows very low accuracy, up to 10% and 20%, but there is a significant improvement in accuracy at 30%, after which the increase becomes gradual. When at 20%, the accuracy decreased compared to at 10%, possibly due to the instability in learning caused by the smaller data. On the other hand, Direct-SFT and Mix demonstrate significantly better accuracy at 10% and 20% compared to the Transformer, and like the Transformer, the accuracy increases gradually from 30% onwards. These results suggest that LLM-based translation models can achieve higher accuracy with less training data than supervised encoder-decoder models. Additionally, COMET scores for all three models show a gradual increase in accuracy from 30%, while BLEU scores con-



Source	So, what started as a bit of an inside joke with myself and a willful provocation, as become <b>a thing</b> . ちょっとした自虐ネタで気の利いた挑発をしたつもりが <b>社会現象</b> にまでなっていました
Reference	(What started as a bit of self-deprecating humor and a clever provocation has turned into <b>a social phenomenon</b> .)
Transformer 69.8	だから、私とのちょっとした内輪の冗談と意図的な挑発として始まったことは、 <b>もの</b> になりました。 (So what started as a little inside joke and intentional provocation with me has become <b>a thing</b> .)
Mix (Ours) 77.8	それで、私と故意の挑発でちょっとした内輪の冗談から始まったものが、今では <b>物議をかもすもの</b> になりました。 (So what started as a little inside joke between me and a deliberate provocation has now become <b>a controversial thing</b> .)

(a) IWSLT21 Simultaneous Translation En-Ja Dev

Source	It's a complex topic, so we're just going to <b>dive</b> right in at a complex place: New Jersey. 複雑なトピックですから 前置きはさておき 複雑な所から <b>始めましょう</b> ニュージャージー州です
Reference	(It is a complex topic, so <b>let us</b> skip the introduction and <b>start</b> with the complicated place. New Jersey.)
Transformer 82.4	それは複雑なトピックなので、私たちは複雑な場所に <b>飛び込む</b> つもりです:ニュージャージー。 (It is a complex topic, so we are going to <b>jump into</b> a complex place:New Jersey.)
Mix (Ours) 88.4	複雑な話題なので、まずはニュージャージー州の複雑な場所から <b>始めよう</b> 。 (It is a complex topic, so <b>let us start</b> with the complicated places in New Jersey.)

(b) TED (tst2015)

Table 4: Specific En  $\Rightarrow$  Ja translation results from the two test set comprising TED Talks domains. The numbers under the model names indicate the COMET scores, and the English text below the Japanese sentences shows the back-translations into English. The phrases requiring free translation and the corresponding reference and model output phrases are highlighted in **red** for source sentences. The results for Mix indicate that supervised fine-tuning with full weight is performed after continual pre-training.

tinue to improve even after 30%. This suggests that at least 3M sentence pairs are needed for the translation model to output sentences containing the same meaning as the reference, whereas more parallel data than the 10.4M sentence pairs is required to output sentences containing the exact words as the reference.

## 6.4 Specific Results of Spoken Language

To analyze the differences in translation between the LLM-based model and the encoder-decoder model for spoken language, Table 4 presents En  $\Rightarrow$  Ja translation examples from two test sets comprising TED Talks domain. In these two examples, the LLM-based translation model has achieved higher COMET scores than the Transformer. In Table 4a, the source sentence contains the phrase "a thing," which the reference translates as "社会現象" (a social phenomenon). In contrast, the Transformer translates "a thing" literally as "もの", and the LLM-based model translates it as "物議をかもすもの" (a controversial thing). Additionally, in Table 4b, the source sentence includes the word "dive," which the reference translates as "始めましょう" (let us start). The Transformer translates "dive" literally as "飛び込む" (jump into), whereas the LLM-based model correctly translates it as "始めよう" (let us start). These results suggest that the LLM-based translation model can perform free translation better than the traditional encoder-decoder model.

## 7 Conclusion

We propose a two-phase training approach comprising continual pre-training with interleaved source and target sentence data, followed by supervised fine-tuning using a small amount of high-quality parallel data. Our investigation comprehensively explores methods for enhancing translation accuracy through continual pre-training across eight data formats. Evaluation across 13 test sets reveals that models trained with continual pre-training followed by supervised fine-tuning outperform those supervised fine-tuned solely on parallel data. Furthermore, we observe variations in language direction accuracy improvement during continual pre-training based on the order of source and target sentences. We also demonstrate that LLM-based translation models are more robust in translating sentences containing spoken language, and achieve higher accuracy with less training data, compared to traditional encoder-decoder models. Additionally, augmenting source sentences with tags or using prefixes yields higher accuracy than simple concatenation. In larger LLMs than the rinna-4b model we utilized, such as LLaMA-2 7B and 13B, LoRA enables training with fewer computational resources. LoRA has been reported to be effective in translation tasks (Zhang et al., 2023; Guo et al., 2024). Therefore, it is essential to experiment with LoRA in the future to determine if similar results can be achieved and to investigate if similar results can be obtained with other LLMs.

## 8 Limitations

Our experiments and conclusions are based only on two translation directions (English-to-Japanese, Japanese-to-English) and rinna/bilingual-gpt-neox-4b, which is an LLM pre-trained in English and Japanese. Evaluation for other translation directions and LLMs has yet to be conducted. While in Section 6.3, we demonstrated that continual pre-training requires 3M parallel data, we anticipate that this may vary depending on the translation direction and model. Whether our approach applies to LLMs primarily pre-trained in English, such as XGLM and LLaMA, remains unverified, especially in low resource languages is challenging. Additionally, all the experiments are conducted using only the parameters described in Section 4.3, and an optimal hyperparameter search still needs to be performed. Especially in Direct-SFT, it should be noted that the importance of hyperparameters has been highlighted by Dettmers et al. (2023), and whether full fine-tuning and LoRA tuning demonstrate the same performance varies depending on the model, hyperparameters, and task.

## 9 Ethical statement

We have not conducted verification on significant risks associated with our research. While we propose a method that may enhance translation accuracy using LLMs, it is worth noting that Zhu et al. (2024) have reported GPT-4’s 8-shot translation accuracy to be comparable to or below that of existing methods such as supervised encoder-decoder models. Therefore, even if the proposed method is applied to other LLMs, we do not think that there is a potential risk that the proposed method achieves too high translation accuracy so that it is to be abused.

This study uses a dataset from Morishita et al. (2022), available only for research and development purposes, inheriting potential biases from their datasets. We utilize open-source pre-trained LLM, and our experimental codes also leverage open-source libraries, as mentioned in Section 4. Therefore, this study’s models, data, and tools adhere to the intended usages of those models, data, and tools.

## 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.
- Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. [Tower: An open multilingual large language model for translation-related tasks](#). *arXiv:2402.17733*.
- Antonios Anastasopoulos, Ondřej Bojar, Jacob Breermann, Roldano Cattoni, Maha Elbayad, Marcello Federico, Xutai Ma, Satoshi Nakamura, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Alexander Waibel, Changan Wang, and Matthew Wiesner. 2021. [FINDINGS OF THE IWSLT 2021 EVALUATION CAMPAIGN](#). In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 1–29, Bangkok, Thailand (online). Association for Computational Linguistics.
- Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrías, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaime Zaragoza. 2020. [ParaCrawl: Web-scale acquisition of parallel corpora](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.
- Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussa, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joannis, 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.

- Eleftheria Briakou, Colin Cherry, and George Foster. 2023. [Searching for needles in a haystack: On the role of incidental bilingualism in PaLM’s translation capability](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9432–9452, Toronto, Canada. Association for Computational Linguistics.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. [WIT3: Web inventory of transcribed and translated talks](#). In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation*, pages 261–268, Trento, Italy. European Association for Machine Translation.
- Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. 2016. [Training deep nets with sublinear memory cost](#). *arXiv:1604.06174*.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. [PaLM: Scaling language modeling with pathways](#). *arXiv:2204.02311*.
- Yiming Cui, Ziqing Yang, and Xin Yao. 2023. [Efficient and effective text encoding for chinese llama and alpaca](#). *arXiv:2304.08177*.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. [QLoRA: Efficient finetuning of quantized llms](#). *arXiv:2305.14314*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jiaxin Guo, Hao Yang, Zongyao Li, Daimeng Wei, Hengchao Shang, and Xiaoyu Chen. 2024. [A novel paradigm boosting translation capabilities of large language models](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 639–649, Mexico City, Mexico. Association for Computational Linguistics.
- Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. [How good are GPT models at machine translation? a comprehensive evaluation](#). *arXiv:2302.09210*.
- 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*.
- Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2022. [Lifelong pretraining: Continually adapting language models to emerging corpora](#). In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 1–16, virtual+Dublin. Association for Computational Linguistics.
- Mihir Kale, Aditya Siddhant, Rami Al-Rfou, Linting Xue, Noah Constant, and Melvin Johnson. 2021. [nmT5 - is parallel data still relevant for pre-training massively multilingual language models?](#) In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 683–691, Online. Association for Computational Linguistics.
- Zixuan Ke, Haowei Lin, Yijia Shao, Hu Xu, Lei Shu, and Bing Liu. 2022. [Continual training of language models for few-shot learning](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10205–10216, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- 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, Makoto Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović,



- and Mariya Shmatova. 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.
- Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal Novák, Martin Popel, and Maja Popović. 2022. [Findings of the 2022 conference on machine translation \(WMT22\)](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1–45, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Taku Kudo and John Richardson. 2018. [SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Xian Li, Paul Michel, Antonios Anastasopoulos, Yonatan Belinkov, Nadir Durrani, Orhan Firat, Philipp Koehn, Graham Neubig, Juan Pino, and Hassan Sajjad. 2019. [Findings of the first shared task on machine translation robustness](#). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 91–102, Florence, Italy. Association for Computational Linguistics.
- Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. [Multilingual denoising pre-training for neural machine translation](#). *Transactions of the Association for Computational Linguistics*, 8:726–742.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [RoBERTa: A robustly optimized BERT pretraining approach](#). *arXiv:1907.11692*.
- 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.
- 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.
- Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchi-moto, 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).
- 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 Hefernan, 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](#). *arXiv:2207.04672*.
- OpenAI. 2023. [GPT-4 technical report](#). *arXiv:2303.08774*.
- 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.
- Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. [On the difficulty of training recurrent neural networks](#). In *Proceedings of the 30th International Conference on Machine Learning*, volume 28, pages 1310–1318, Atlanta, Georgia, USA. PMLR.
- Reid Pryzant, Youngjoo Chung, Dan Jurafsky, and Denny Britz. 2018. [JESC: Japanese-English subtitle corpus](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. [Improving language understanding by generative pre-training](#). *Technical report*.
- 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*, 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. 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.
- 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.
- Andrea Schioppa, Xavier Garcia, and Orhan Firat. 2023. [Cross-lingual supervision improves large language models pre-training](#). *arXiv:2305.11778*.
- Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. [Fine-tuned language models are continual learners](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6107–6122, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Noam Shazeer. 2020. [GLU variants improve transformer](#). *arXiv:2002.05202*.
- Lucia Specia, Zhenhao Li, Juan Pino, Vishrav Chaudhary, Francisco Guzmán, Graham Neubig, Nadir Durrani, Yonatan Belinkov, Philipp Koehn, Hassan Sajjad, Paul Michel, and Xian Li. 2020. [Findings of the WMT 2020 shared task on machine translation robustness](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 76–91, Online. Association for Computational Linguistics.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. [Dropout: A simple way to prevent neural networks from overfitting](#). *Journal of Machine Learning Research*, 15(56):1929–1958.
- Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. [Rethinking the inception architecture for computer vision](#). In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Brian Thompson, Mehak Preet Dhaliwal, Peter Frisch, Tobias Domhan, and Marcello Federico. 2024. [A shocking amount of the web is machine translated: Insights from multi-way parallelism](#). *arXiv:2401.05749*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. [Llama 2: Open foundation and fine-tuned chat models](#). *arXiv:2307.09288*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024a. [A paradigm shift in machine translation: Boosting translation performance of large language models](#). In *The Twelfth International Conference on Learning Representations*.
- Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024b. [Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation](#). In *Forty-first International Conference on Machine Learning*.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. [mT5: A massively multilingual pre-trained text-to-text transformer](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.
- Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. 2024. [LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention](#). In *The Twelfth International Conference on Learning Representations*.
- Xuan Zhang, Navid Rajabi, Kevin Duh, and Philipp Koehn. 2023. [Machine translation with large language models: Prompting, few-shot learning, and fine-tuning with QLoRA](#). In *Proceedings of the Eighth Conference on Machine Translation*, pages 468–481, Singapore. Association for Computational Linguistics.
- Chunting Zhou, Pengfei Liu, Puxin Xu, Srinu Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu,

LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. [LIMA: Less is more for alignment](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.

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

Model	En $\Rightarrow$ Ja (Avg.)		Ja $\Rightarrow$ En (Avg.)	
	BLEU	COMET	BLEU	COMET
ALMA-7B-Ja-V2	10.1	80.4	13.0	75.6
Tagged + SFT (full)	<b>15.0</b>	<b>83.2</b>	<b>16.2</b>	<b>77.3</b>

Table 5: Results of BLEU and COMET scores for ALMA-7B-Ja-V2 and the rinna-4b-based translation model. “Tagged + SFT (full)” represents the model continually pre-trained in the Tagged format as described in Section 6.1, followed by supervised fine-tuning with full weight.

## A Sampling of JParaCrawl v3.0

We use 20.8M parallel sentences from JParaCrawl v3.0, initially consisting of 21.8M parallel sentences. We sample sentence pairs using cosine similarity scores between 0.4 and 0.95 based on sentence vector embeddings obtained from LEALLA-large. Parallel sentences with a similarity score below 0.4 are excluded, as a visual inspection revealed a significant presence of inappropriate samples, such as Japanese and English sentences with disproportionate lengths. Additionally, parallel sentences with similarity scores of 0.95 or higher are also excluded, as they consist of Japanese and English sentences that were nearly identical. This sampling results in 1.8B tokens when tokenized with the rinna-4b tokenizer.

## B Comparison with ALMA

We compared with ALMA by using ALMA-7B-Ja-V2<sup>11</sup>, which was trained similarly to ALMA with LLaMA-2 7B but with Russian replaced by Japanese among the languages experimented with ALMA. We compared against ALMA-Ja-V2 using the BLEU and COMET averages of 12 test sets, as shown in the Table 5. From these results, it is evident that the 3.8B LLM-based translation model outperforms the LLaMA-2-based ALMA-7B-Ja-V2. This result aligns with reports suggesting higher accuracy when using parallel data for continual pre-training (Alves et al., 2024; Guo et al., 2024) and the consistency with reports indicating that the influence of parallel data increases with fewer parameters (Kale et al., 2021; Briakou et al., 2023).

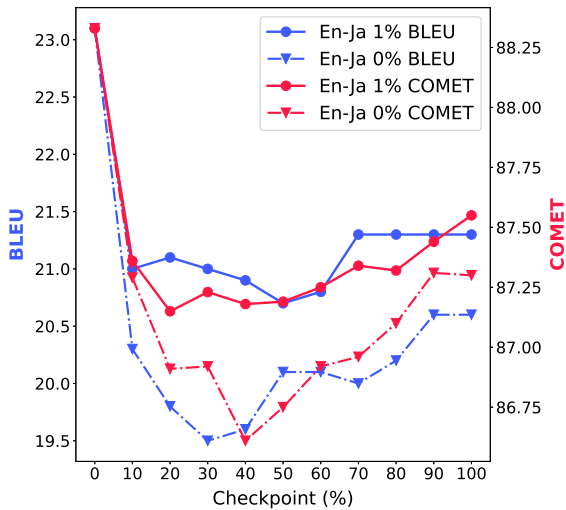


Figure 3: Data Curves for BLEU and COMET scores at each 10% checkpoint of En-Ja2Mix for  $\text{En} \Rightarrow \text{Ja}$  on WMT22 test data. All models at checkpoints have undergone supervised fine-tuning. The 0% on the x-axis represents the accuracy of En-Ja.

### C Analyzing Catastrophic Forgetting

We conducted continual pre-training on En-Ja and then observed catastrophic forgetting by conducting continual pre-training on data, which was in the reverse direction. Based on reports suggesting preventing catastrophic forgetting by mixing data from tasks that should not be forgotten (Scialom et al., 2022), we mixed 1% of En-Ja data. This model is named En-Ja2Mix. Figure 3 shows the data curves for BLEU and COMET scores at each 10% checkpoint for En-Ja2Mix on the WMT22 test data, demonstrating  $\text{En} \Rightarrow \text{Ja}$ . As an ablation study, we also show the data curves for a scenario where the 1% of En-Ja data added to En-Ja2Mix is removed, and continual pre-training is conducted entirely with Ja-En data. From these data curves, it is observed that when conducting continual pre-training with En-Ja data and subsequently with data in the reverse direction, mixing 1% of the first continual pre-training data can mitigate the degradation in accuracy for  $\text{En} \Rightarrow \text{Ja}$ . Therefore, this suggests that in the continual pre-training of LLMs, incorporating a small proportion of data that one does not wish to be forgotten can suppress catastrophic forgetting.

### D Detailed Tables

<sup>11</sup><https://huggingface.co/webbigdata/ALMA-7B-Ja-V2>

Direction	Test set	Domain	# sentences
En ⇌ Ja	ASPEC (Nakazawa et al., 2016)	Scientific Papers	1,812
	JESC (Pryzant et al., 2018)	Movie Subtitles	2,000
	KFTT (Neubig, 2011)	Wikipedia Articles	1,160
	TED (tst2015) (Cettolo et al., 2012)	TED Talk	1,194
	Business Scene Dialogue Corpus (BSD) (Rikters et al., 2019)	Dialogues	2,120
En ⇒ Ja	WMT19 Robustness En-Ja (MTNT2019) (Li et al., 2019)	Reddit	1,392
	WMT20 Robustness Set1 En-Ja (Specia et al., 2020)	Wikipedia Comments	1,100
	WMT20 Robustness Set2 En-Ja (Specia et al., 2020)	Reddit	1,376
	IWSLT21 Simultaneous Translation En-Ja Dev (Anastasopoulos et al., 2021)	TED Talk	1,442
	WMT22 General Machine Translation Task En-Ja (Kocmi et al., 2022)	News, social, e-commerce, dialogue	2,037
Ja ⇒ En	WMT19 Robustness Ja-En (MTNT2019) (Li et al., 2019)	Reddit	1,111
	WMT20 Robustness Set2 Ja-En (Specia et al., 2020)	Reddit	997
	WMT22 General Machine Translation Task Ja-En (Kocmi et al., 2022)	News, social, e-commerce, dialogue	2,008

Table 6: Domain and Number of sentences in test sets. “# sentences” represents the number of sentences on the English side.

Test set	Baseline Models		Continual pre-training + Supervised fine-tuning							
	Transformer	Direct-SFT	Mono		En-Ja		Ja-En		Mix	
			full	LoRA	full	LoRA	full	LoRA	full	LoRA
ASPEC	<b>19.6</b>	15.4	5.1	4.6	19.1	19.0	6.6	6.4	18.4	18.5
JESC	5.8	5.0	3.6	3.4	<b>7.4*</b>	7.3*	4.3	3.9	<b>7.4*</b>	7.0*
KFTT	12.8	8.7	6.7	6.1	<b>15.5*</b>	15.0*	7.1	6.3	14.1*	13.5
TED	12.2	10.9	5.4	5.1	12.7	12.8*	6.7	6.3	12.3	<b>12.9*</b>
BSD	12.5	<b>13.1*</b>	7.5	7.5	14.4*	<b>15.5*</b>	8.6	8.6	14.1*	15.2*
WMT19 R En-Ja	13.1	12.3	6.0	5.5	<b>15.2*</b>	15.1*	6.7	6.7	14.4*	14.7*
WMT20 R Set1 En-Ja	16.9	15.3	7.1	6.5	18.4*	<b>19.5*</b>	7.5	8.0	17.5	18.7*
WMT20 R Set2 En-Ja	12.9	12.1	5.5	4.9	<b>14.8*</b>	<b>14.8*</b>	6.8	6.8	13.9*	13.7*
IWSLT21 En-Ja Dev	12.2	9.8	5.3	5.2	<b>13.2*</b>	<b>13.2*</b>	6.6	7.0	12.8*	12.8*
WMT22 GMT En-Ja	21.2	19.1	11.0	9.8	<b>23.1*</b>	<b>23.1*</b>	11.9	11.7	22.0*	22.1*
Average	13.9	12.2	6.3	5.9	15.4	<b>15.5</b>	7.3	7.2	14.7	14.9
# Sig.	-	1	0	0	8	<b>9</b>	0	0	7	8

(a) BLEU

Test set	Baseline Models		Continual pre-training + Supervised fine-tuning							
	Transformer	Direct-SFT	Mono		En-Ja		Ja-En		Mix	
			full	LoRA	full	LoRA	full	LoRA	full	LoRA
ASPEC	88.5	87.0	78.7	77.1	88.6	<b>88.7</b>	80.5	80.7	88.1	88.2
JESC	71.7	<b>72.8*</b>	71.4	70.8	<b>76.0*</b>	75.7*	72.4	72.4	75.7*	75.8*
KFTT	83.9	79.9	76.4	75.7	<b>84.4</b>	84.3	76.4	76.1	83.3	83.7
TED	79.7	<b>80.6*</b>	76.4	75.1	<b>83.6*</b>	83.2*	78.1	77.7	83.0*	83.0*
BSD	84.2	85.7*	81.3	81.2	87.5*	<b>87.7*</b>	82.9	83.1	87.0*	87.4*
WMT19 R En-Ja	75.8	77.0*	74.2	73.1	<b>81.7*</b>	81.5*	75.3	75.1	81.2*	81.0*
WMT20 R Set1 En-Ja	65.2	<b>68.2*</b>	64.6	63.7	<b>76.8*</b>	76.3*	65.6	66.3	76.2*	75.4*
WMT20 R Set2 En-Ja	74.0	76.1*	72.7	72.2	<b>81.6*</b>	81.1*	74.8	74.6	80.6*	80.4*
IWSLT21 En-Ja Dev	81.8	82.2	79.4	78.8	<b>86.2*</b>	85.9*	81.3	81.2	85.8*	85.8*
WMT22 GMT En-Ja	84.9	<b>85.8*</b>	80.8	79.8	<b>88.3*</b>	<b>88.3*</b>	82.1	81.0	87.8*	87.9*
Average	79.0	<b>79.6</b>	75.6	74.8	<b>83.5</b>	83.3	76.9	76.8	82.9	82.9
# Sig.	-	7	0	0	<b>8</b>	<b>8</b>	0	0	<b>8</b>	<b>8</b>

(b) COMET

Table 7: Results of En ⇒ Ja translation accuracy. Details of baseline models and models continually pre-trained with four orders described in Section 4.2.2 then supervised fine-tuning. **Bold numbers** represent the highest scores in each line, and scores that surpass the Transformer are emphasized in green. “\*” indicates significant differences compared to Transformer, “# Sig.” indicates the number of test sets showing significant differences from Transformer ( $p < 0.05$ ).



Test set	Baseline Models		Continual pre-training + Supervised fine-tuning							
	Transformer	Direct-SFT	Mono		En-Ja		Ja-En		Mix	
			full	LoRA	full	LoRA	full	LoRA	full	LoRA
ASPEC	<b>21.8</b>	16.0	8.8	7.9	9.4	8.5	20.3	20.4	19.1	19.4
JESC	<b>8.9</b>	6.3	4.6	4.0	4.1	4.3	8.5	8.8	7.9	7.7
KFTT	<b>21.0</b>	11.1	9.6	8.4	10.6	9.7	19.9	19.0	18.5	17.4
TED	14.7	10.6	7.4	6.3	6.9	6.6	14.7	<b>15.2</b>	14.3	14.4
BSD	<b>19.8</b>	16.0	9.4	8.8	8.5	9.2	20.1	20.4	18.7	18.6
WMT19 R Ja-En	17.2	14.2	8.3	6.9	8.3	7.1	<b>18.0</b>	17.1	16.4	16.5
WMT20 R Set2 Ja-En	<b>14.3</b>	10.8	5.6	5.4	5.4	5.4	13.9	14.1	13.0	13.0
WMT22 GMT Ja-En	21.0	15.0	9.5	9.4	9.3	9.9	20.8	<b>21.1</b>	19.1	19.3
Average	<b>17.3</b>	12.5	7.9	7.1	7.8	7.6	17.0	17.0	15.9	15.8
# Sig.	-	0	0	0	0	0	0	0	0	0

(a) BLEU

Test set	Baseline Models		Continual pre-training + Supervised fine-tuning							
	Transformer	Direct-SFT	Mono		En-Ja		Ja-En		Mix	
			full	LoRA	full	LoRA	full	LoRA	full	LoRA
ASPEC	<b>82.7</b>	80.4	74.8	74.2	75.3	74.8	82.5	82.5	81.9	82.1
JESC	68.0	67.2	64.3	63.2	64.3	63.5	69.2*	<b>69.3*</b>	68.7*	68.6*
KFTT	77.4	73.5	70.1	69.4	70.5	70.3	<b>78.2*</b>	77.8	77.4	76.6
TED	77.6	75.9	71.2	70.4	71.2	70.9	<b>78.7*</b>	<b>78.7*</b>	78.3*	78.1*
BSD	81.1	79.9	74.4	74.1	74.1	74.2	<b>82.9*</b>	82.0*	81.4	81.1
WMT19 R Ja-En	74.0	74.2	69.6	68.6	69.0	68.4	<b>76.8*</b>	76.3*	76.5*	76.2*
WMT20 R Set2 Ja-En	70.6	70.6	65.7	64.8	65.0	65.0	72.8*	<b>73.0*</b>	72.2*	72.1*
WMT22 GMT Ja-En	79.9	77.9	73.4	72.6	72.7	72.9	81.0*	<b>82.0*</b>	80.5*	80.4*
Average	76.4	73.8	70.4	69.7	70.3	70.0	<b>77.8</b>	77.7	77.1	76.9
# Sig.	-	0	0	0	0	0	<b>7</b>	<b>6</b>	<b>5</b>	<b>5</b>

(b) COMET

Table 8: Results of Ja  $\Rightarrow$  En translation accuracy. **Bold scores**, **green numbers**, “\*”, and “# Sig.” are the same in Table 7.

Test set	Transformer	Interleaved		Prefix		Tagged		JSON	
		full	LoRA	full	LoRA	full	LoRA	full	LoRA
ASPEC	19.6 / 88.5	18.4 / 88.1	18.5 / 88.2	18.8 <sup>†</sup> / 88.5 <sup>†</sup>	18.6 / 88.5 <sup>†</sup>	18.8 <sup>†</sup> / 88.5 <sup>†</sup>	18.7 / 88.6 <sup>†</sup>	18.5 / 88.3 <sup>†</sup>	18.7 / 88.5 <sup>†</sup>
JESC	5.8 / 71.7	7.4* / 75.7*	7.0* / 75.8*	7.8* / 75.7*	6.9* / 75.8*	7.4* / 75.9*	6.8* / 75.8*	7.2* / 75.3*	6.6* / 75.7*
KFTT	12.8 / 83.9	14.1* / 83.3	13.5* / 83.7	<b>14.9*</b> <sup>†</sup> / 83.7	13.9* / 84.0	14.2* / 83.6	14.1* / 83.6	14.0* / 83.3	13.9* / 83.7
TED	12.2 / 79.7	12.3 / 83.0*	12.9* / 83.0*	12.6 / 83.0*	13.0* / 83.2*	<b>12.7*</b> <sup>†</sup> / <b>83.4*</b> <sup>†</sup>	13.1* / 83.2*	<b>12.8*</b> <sup>†</sup> / 83.0*	13.0* / 83.2*
BSD	12.5 / 84.2	14.1* / 87.0*	15.2* / 87.4*	<b>14.7*</b> <sup>†</sup> / 87.2*	15.1* / 87.5*	<b>14.6*</b> <sup>†</sup> / 87.2*	14.9* / 87.5*	14.2* / 86.9*	14.9* / 87.4*
WMT19 R En-Ja	13.1 / 75.8	14.4* / 81.2*	14.7* / 81.0*	<b>15.3*</b> <sup>†</sup> / 81.3*	<b>15.4*</b> <sup>†</sup> / <b>81.7*</b> <sup>†</sup>	<b>15.1*</b> <sup>†</sup> / <b>81.9*</b> <sup>†</sup>	14.9* / <b>81.6*</b> <sup>†</sup>	14.3* / 80.9*	14.5* / 81.3*
WMT20 R Set1 En-Ja	16.9 / 65.2	17.5 / 76.2*	18.7* / 75.4*	17.8* / 76.2*	<b>19.5*</b> <sup>†</sup> / <b>76.7*</b> <sup>†</sup>	<b>18.0*</b> <sup>†</sup> / 76.6*	<b>19.2*</b> <sup>†</sup> / <b>76.8*</b> <sup>†</sup>	12.1 / 69.6*	18.1* / 74.0*
WMT20 R Set2 En-Ja	12.9 / 74.0	13.9* / 80.6*	13.7 / 80.4*	14.0* / 80.7*	<b>14.8*</b> <sup>†</sup> / <b>80.9*</b> <sup>†</sup>	<b>14.3*</b> <sup>†</sup> / <b>81.1*</b> <sup>†</sup>	<b>14.5*</b> <sup>†</sup> / <b>81.1*</b> <sup>†</sup>	13.6* / 80.0*	<b>14.3*</b> <sup>†</sup> / 80.6*
IWSLT21 En-Ja Dev	12.2 / 81.8	12.8* / 85.8*	12.8* / 85.8*	12.7* / 85.9*	13.0* / <b>86.0*</b> <sup>†</sup>	12.7* / 86.0*	12.6 / 86.0*	12.6 / 85.7*	12.7* / <b>86.1*</b> <sup>†</sup>
WMT22 GMT En-Ja	21.2 / 84.9	22.0* / 87.8*	22.1* / 87.9*	<b>22.7*</b> <sup>†</sup> / 88.0*	<b>22.7*</b> <sup>†</sup> / <b>88.2*</b> <sup>†</sup>	<b>22.4*</b> <sup>†</sup> / <b>88.2*</b> <sup>†</sup>	22.4* / <b>88.3*</b> <sup>†</sup>	22.2* / 87.9*	22.4* / 88.1*
Average	13.9 / 79.0	14.7 / 82.9	14.9 / 82.9	15.1 / 83.0	15.3 / 83.3	15.0 / 83.2	15.1 / 83.3	14.2 / 82.1	14.9 / 82.9
# Sig.	-	-	-	4 / 0	4 / 5	6 / 4	2 / 4	1 / 0	1 / 1

(a) En  $\Rightarrow$  Ja

Test set	Transformer	Interleaved		Prefix		Tagged		JSON	
		full	LoRA	full	LoRA	full	LoRA	full	LoRA
ASPEC	21.8 / 82.7	19.1 / 81.9	19.4 / 82.1	19.6 <sup>†</sup> / 82.3 <sup>†</sup>	19.6 / 82.2	19.5 / 82.1	19.6 / 82.1	19.3 / 82.1	19.3 / 82.1
JESC	8.9 / 68.0	7.9 / 68.7*	7.7 / 68.6*	7.9 / <b>69.1*</b> <sup>†</sup>	7.9 / <b>68.9*</b> <sup>†</sup>	8.1 / 68.8*	8.0 / 68.7*	7.9 / 68.6*	7.8 / 68.7*
KFTT	21.0 / 77.4	18.5 / 77.4	17.4 / 76.6	18.9 / 77.6	18.4 <sup>†</sup> / 77.2 <sup>†</sup>	19.0 <sup>†</sup> / 77.4	18.6 <sup>†</sup> / 77.1 <sup>†</sup>	18.7 / 77.3	17.6 / 76.7
TED	14.7 / 77.4	14.3 / 78.3*	14.4 / 78.1*	14.1 / 78.4*	14.4 / 78.3*	14.6 / <b>78.8*</b> <sup>†</sup>	14.2 / 78.3*	13.3 / 78.0*	14.1 / 78.3*
BSD	19.8 / 81.1	18.7 / 81.4	18.6 / 81.1	19.5 <sup>†</sup> / <b>81.6*</b> <sup>†</sup>	19.3 <sup>†</sup> / <b>81.6*</b> <sup>†</sup>	19.0 / 81.6*	18.9 / 81.6*	18.6 / 81.5*	18.8 / 81.3
WMT19 R Ja-En	17.2 / 74.0	16.4 / 76.5*	16.5 / 76.2*	17.2 <sup>†</sup> / 76.7*	16.8 / 76.2*	16.3 / 76.5*	16.9 / 76.5*	14.8 / 75.6*	15.9 / 75.6*
WMT20 R Set2 Ja-En	14.3 / 70.6	13.0 / 72.2*	13.0 / 72.1*	13.0 / 72.6*	13.4 / 72.6*	13.3 / <b>72.6*</b> <sup>†</sup>	13.7 / 72.4*	12.1 / 71.7*	12.9 / 72.2*
WMT22 GMT Ja-En	21.0 / 79.9	19.1 / 80.5*	19.3 / 80.4*	19.8 <sup>†</sup> / <b>80.8*</b> <sup>†</sup>	19.3 / <b>80.8*</b> <sup>†</sup>	20.0 <sup>†</sup> / <b>80.8*</b> <sup>†</sup>	20.3 <sup>†</sup> / <b>81.0*</b> <sup>†</sup>	19.2 / 80.6*	19.8 / 80.6*
Average	17.3 / 76.4	15.9 / 77.1	15.8 / 76.9	16.3 / 77.4	16.1 / 77.2	16.2 / 77.3	16.3 / 77.2	15.5 / 76.9	15.8 / 76.9
# Sig.	-	-	-	0 / 3	0 / 3	0 / 3	0 / 1	0 / 0	0 / 0

(b) Ja  $\Rightarrow$  En

Table 9: Results of translation accuracy (BLEU / COMET). “\*” indicates significant differences compared to Transformer, <sup>†</sup> indicates significant differences compared to the same fine-tuning method as Interleaved Translations (Interleaved), **bold numbers** indicate significant differences in both Transformer and the same fine-tuning method as Interleaved Translations, and “# Sig.” denotes the number of test sets where significant differences is observed in both Transformer and the same fine-tuning method as Interleaved Translations. ( $p < 0.05$ )

Test set	BLEU				COMET			
	rinna-4b	rinna-4b + CPT	rinna-4b + SFT	rinna-4b + CPT + SFT	rinna-4b	rinna-4b + CPT	rinna-4b + SFT	rinna-4b + CPT + SFT
ASPEC	0.3	8.9	5.2	<b>18.8</b>	45.3	72.4	79.0	<b>88.5</b>
JESC	0.2	3.1	3.6	<b>7.4</b>	36.1	64.4	71.9	<b>75.9</b>
KFTT	0.3	4.2	6.8	<b>14.2</b>	41.4	66.8	76.4	<b>83.6</b>
TED	0.3	9.1	5.4	<b>12.7</b>	40.5	72.0	77.2	<b>83.4</b>
BSD	0.4	8.8	7.6	<b>14.6</b>	40.6	77.7	81.7	<b>87.2</b>
WMT19 R En-Ja	0.6	6.5	6.7	<b>15.1</b>	40.2	64.7	75.1	<b>81.9</b>
WMT20 R Set1 En-Ja	2.0	7.1	<b>7.7</b>	18.0	39.3	49.3	66.5	<b>76.6</b>
WMT20 R Set2 En-Ja	0.4	7.4	6.1	<b>14.3</b>	39.5	65.0	74.3	<b>81.1</b>
IWSLT21 En-Ja Dev	0.2	7.6	5.6	<b>12.7</b>	40.7	73.2	80.6	<b>86.0</b>
WMT22 GMT En-Ja	1.4	19.3	10.3	<b>22.4</b>	38.3	83.5	81.3	<b>88.2</b>
Average	0.6	8.2	6.5	<b>15.0</b>	40.2	69.9	76.4	<b>83.2</b>

(a) En  $\Rightarrow$  Ja

Test set	BLEU				COMET			
	rinna-4b	rinna-4b + CPT	rinna-4b + SFT	rinna-4b + CPT + SFT	rinna-4b	rinna-4b + CPT	rinna-4b + SFT	rinna-4b + CPT + SFT
ASPEC	1.0	15.5	8.9	<b>19.5</b>	53.0	77.7	75.0	<b>82.1</b>
JESC	0.3	4.0	4.5	<b>8.1</b>	41.5	62.3	64.6	<b>68.8</b>
KFTT	0.2	9.6	10.0	<b>19.0</b>	44.0	66.7	71.1	<b>77.4</b>
TED	0.3	6.6	7.4	<b>14.6</b>	49.6	66.6	72.0	<b>78.8</b>
BSD	0.3	13.3	9.0	<b>19.0</b>	48.1	76.5	74.6	<b>81.6</b>
WMT19 R Ja-En	1.4	8.3	8.5	<b>16.3</b>	49.9	65.2	69.6	<b>76.5</b>
WMT20 R Set2 Ja-En	0.5	6.7	6.0	<b>13.3</b>	46.4	62.6	65.7	<b>72.6</b>
WMT22 GMT Ja-En	2.0	15.3	9.8	<b>20.0</b>	39.3	76.6	73.4	<b>80.8</b>
Average	0.8	9.9	8.0	<b>16.2</b>	46.0	69.3	70.9	<b>77.3</b>

(b) Ja  $\Rightarrow$  En

Table 10: Results of all combinations of continual pre-training and supervised fine-tuning (BLEU / COMET). **Bold numbers** indicate the highest scores in each line. “+ CPT” indicates continual pre-training in the Tagged format, described in Section 6.1. At the same time, “+ SFT” represents supervised fine-tuning with a small amount of high-quality parallel data, as described in Section 4.1.2. During supervised fine-tuning, zero-shot inference is performed, and five-shot inference is performed for others.