Patent Claim Translation via Continual Pre-training of Large Language Models with Parallel Data

Haruto Azami[1] **Minato Kondo**[1] **Takehito Utsuro**[1] **Masaaki Nagata**[2]
[1]Deg. Prog. Sys.&Inf. Eng., Grad. Sch. Sci.&Tech., University of Tsukuba
[2]NTT Communication Science Laboratories, NTT Corporation, Japan
s2420710_@_u.tsukuba.ac.jp, s2320743_@_u.tsukuba.ac.jp,
utsuro_@_iit.tsukuba.ac.jp,
masaaki.nagata_@_ntt.com

Abstract

Recent advancements in large language models (LLMs) have enabled their application across various domains. However, in the field of patent translation, Transformer encoder-decoder based models remain the standard approach, and the potential of LLMs for translation tasks has not been thoroughly explored. In this study, we conducted patent claim translation using an LLM fine-tuned with parallel data through continual pre-training and supervised fine-tuning. A comparative evaluation against Transformer encoder-decoder based translations showed that the fine-tuned LLM achieved high scores for both BLEU and COMET, demonstrating improvements in addressing issues such as omissions and repetitions. Nonetheless, hallucination errors, which were not observed in traditional models, occurred in some cases and negatively affected translation quality. These findings highlight the promise of LLMs for patent translation while also identifying challenges that warrant further investigation.

1 Introduction

Large language models (LLMs) demonstrate exceptional versatility because of their extensive pre-training, proving highly effective in various natural language processing tasks, such as summarization and question-answering. In the field of machine translation, closed LLMs like GPT-4 have been reported to achieve higher human evaluation scores than existing translation models (Kocmi et al., 2023, 2024). However, in the patent domain, Transformer encoder-decoder based translation models remain the mainstream approach, and the translation capabilities of LLMs have not been sufficiently explored. The translation quality of patent documents has reached a sufficiently high level with conventional neural machine translation (NMT) methods, particularly for the main body of patent texts. However, patent claims remain a notable exception where translation quality is still problematic. Patent claims are known for their extremely long and syntactically complex sentence structures, which pose significant challenges for traditional models. In addition, this study focuses on Japanese-to-English translation, where a major obstacle is the significant difference in word order between the two languages. Such structural divergence further complicates the translation of patent claims, especially in preserving the meaning and consistency across long sequences. In contrast, LLMs are believed to be capable of translating long sequences while maintaining global coherence and consistency. Motivated by this potential, the present study investigates how effectively LLMs can translate patent claims, which represent the most difficult component in patent translation. To this end, we adopt the method proposed by Kondo et al. (2024), utilizing parallel patent data for continual pre-training and supervised fine-tuning (SFT) to construct an LLM specialized in patent claim translation. The performance of this LLM is then compared with that of conventional Transformer-based models, with translation quality evaluated using metrics such as BLEU and COMET. The results demonstrate that the LLM statistically significantly outperforms conventional models, effectively addressing issues such as omissions, repetitions, and terminology inconsistency. However, the study also reveals LLM-specific challenges, such as hallucinations, which are observed in specific cases that do not occur in conventional models. This study evaluates both the potentials and challenges of applying LLMs to patent translation, highlighting their effectiveness and identifying areas requiring further improvement.

2 Related Work

2.1 Translation of Patent Claims

Patent claims are one of the most important parts of a patent document, and they are characterized by strict sentence structures and specialized terminology, making them a significant challenge for machine translation.

Fuji et al. (2015) applied statistical machine translation (SMT) to the translation of English, Chinese, and Japanese patent claims and proposed a method for appropriately transforming claim structures. Their approach utilized manually created synchronous context-free grammar (SCFG) rules to convert the source language structure into the target language structure, thereby addressing the unique descriptive style found in patent claims. However, this method had a limitation: the need for manual rule creation that hindered the flexible adaptation to new descriptive styles.

Additionally, research on patent claim translation has been explored in the NTCIR patent translation task. Conducted by Fujii et al. and Goto et al. from 2008 to 2013, respectively, this task primarily employed SMT, advancing the use of parallel corpora and evaluation methods for patent document translation. In particular, translating lengthy patent claims requires maintaining consistent terminology and proper structural transformations, often supplemented by rule-based approaches.

Subsequently, the patent translation task was incorporated into the Workshop on Asian Translation (WAT), where the neural machine translation (NMT) approach, which had already become dominant in machine translation, was applied to patent translation, as demonstrated by Nakazawa et al. (2016). While NMT improved translation fluency, maintaining the strict structure of patent claims remained a challenge. In recent years, there has been progress in constructing large-scale parallel corpora specifically for patent translation. In 2022, the EuroPat corpus was released by K. Heafield and Wiggins (2022), providing a multilingual parallel dataset based on European patent documents. This resource laid a foundation for research in patent translation, especially among European languages. More recently, in 2024, JaParaPat—a large-scale Japanese-English parallel corpus for patent translation—was introduced (Nagata et al., 2024). Constructed using patent family alignments between Japanese and U.S. patent applications, this resource is utilized in our study as training data for both the continual pretraining and supervised fine-tuning of LLM. The development of such domain-specific resources facilitates research aimed at improving patent translation quality, particularly for the Japanese-English language pair.

2.2 LLM-based Translation

In recent years, LLMs have gained attention in the field of machine translation, demonstrating high accuracy in general text domains such as news articles and dialogues. In particular, the use of QLoRA for fine-tuning LLMs has significantly improved multilingual translation performance (Zhang et al., 2023).

Guo et al. (2024) and Kondo et al. (2024) proposed a method combining continual pre-training on parallel data with SFT to enhance the LLM-based translation performance beyond the traditional Transformer encoder-decoder based models. Their approach involved the continual pre-training using large-scale web-crawled parallel corpora, followed by SFT with high-quality parallel datasets, notably improving translation accuracy. Specifically, Kondo et al. (2024) provided a detailed analysis of the Japanese-English translation, addressing the dataset selection and fine-tuning strategies.

In parallel, recent work has explored domain adaptation methods tailored for LLM-based machine translation. Zheng et al. (2024) conducted a comprehensive comparison of fine-tuning strategies such as full fine-tuning, LoRA, and prompt tuning, demonstrating their effectiveness in adapting LLMs to domain-specific translation tasks. Moslem et al. (2023) proposed an adaptive machine translation framework using LLMs, which integrates context-aware prompting and auxiliary data to improve translation quality in specialized domains. These studies highlight the growing interest in leveraging LLMs for translation in complex, domain-specific settings such as legal or patent language, which motivates our focus on patent claim translation using domain-adapted LLMs.

Recent research also points out key challenges and refinements in LLM-based translation. Xu et al. (2024) demonstrated that models predominantly pre-trained on English data, such as LLaMA-2, suffer reduced translation accuracy when translating into non-English target languages. To address this, they introduced ALMA, a two-stage fine-tuning method: first with monolingual data, then with a small quantity of high-quality parallel data.

Despite these advances, LLM-based translation models have primarily been evaluated on test sets from the WMT General Machine Translation Task (Kocmi et al., 2022, 2023) and Flores-200 (Team et al., 2022), and their effectiveness across diverse domains remains underexplored.

3 Experimental Setup

3.1 Model and Training Procedure

This study follows the approach of Kondo et al. (2024), applying continual pre-training and supervised fine-tuning (SFT) to an open-source LLM, **rinna/llama-3-youko-8b**[1], hereafter referred to as **youko-8b**. youko-8b is a 7B-parameter model initially pre-trained on 22 billion tokens of Japanese and English monolingual data. To adapt the model to the patent translation task, we conducted continual pre-training using parallel patent data, followed by SFT to specialize it for translating patent claims.

3.2 Dataset

We used JaParaPat (Nagata et al., 2024), a large-scale Japanese-English parallel corpus of patent data, for both continual pre-training and supervised fine-tuning. JaParaPat consists of approximately 300 million sentence pairs constructed from patent applications published between 2000 and 2021 by the Japan Patent Office (JPO) and the United States Patent and Trademark Office (USPTO). The dataset was created through document alignment based on patent family information, followed by sentence segmentation and machine-translation-based sentence alignment.

In this study, different subsets of JaParaPat were used depending on the purpose:

- For continual pre-training, we used parallel data from 2016 to 2020, comprising approximately 61 million sentence pairs. From this data, 50,000 sentence pairs were excluded to construct a development set. Sentence similarity was calculated using LaBSE (Feng et al., 2022), and 10,984 pairs with similarity scores between 0.9 and 0.95 were selected as the development set.
- For supervised fine-tuning, we used the 2021 portion of JaParaPat, focusing on patent claims. Sentence pairs were filtered based on similarity scores (0.8 to 0.95), and the selected data was divided into training and development sets. The test set was also constructed from 2021 patent claims by selecting unique sentence pairs with similarity scores between 0.9 and 0.95 and containing more than 100 words.

Usage	Time Period	Data Type	Sentence Pairs	English Words
continual pre-training	2016∼2020	training development	61,364,685 10,984	1.9B 327K
SFT	2021	training development	15,000 1,000	53.6K 36.7K
test set	2021	—	33,923	—

Table 1: Usage and Details of Patent Parallel Data

Table 1 summarizes the breakdown of the data used in each stage.

The input format for continual pre-training was as follows:

```
{Japanese sentence}
{English sentence}
```

For supervised fine-tuning, we used a prompt-based format:

> これを日本語から英語に翻訳してください.
>
> 日本語 (Japanese):Japanese sentence
>
> 英語 (English):English sentence

The English translation of the above prompt is:

> "Translate this from Japanese to English."

We applied both full fine-tuning and LoRA (Hu et al., 2022) for the supervised fine-tuning stage.

3.3 Hyperparameter Settings

The hyperparameters of the continual pre-training are shown in Table 2, and the hyperparameters of the SFT are shown in Table 3. In continual pre-training, bfloat16 and DeepSpeed ZeRO stage 2 (Rasley et al., 2023) were applied during training. The SFT was performed on the model that achieved the lowest validation error during the continual pre-training.

[1] https://huggingface.co/rinna/llama-3-youko-8b

Hyperparameter	Value
optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$)
learning rate schedule	cosine scheduler
warmup ratio	1%
max learning rate	2.5×10^{-5}
weight decay	0.1
gradient Clip	1.0
batch Size	1,024
validate interval updates ratio	10%
epochs	1

Table 2: Hyperparameters for Continual Pre-training

Hyperparameter	Value
optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$)
learning rate schedule	cosine scheduler
warmup ratio	1%
max learning rate	2.5×10^{-6}
weight decay	0.1
gradient Clip	1.0
batch Size	64
epochs	2

Table 3: Hyperparameters for Supervised Fine-Tuning

3.4 Comparative Methods

3.4.1 Baseline

As a baseline, we employed a Transformer encoder-decoder based translation model. The model was trained on the same patent parallel corpus as the LLM-based models, comprising approximately 61M sentence pairs. Specifically, we employed the machine translation software by Fairseq (Ott et al., 2019) and used Transformer Big (Vaswani et al., 2017) as the translation model. The hyperparameters of the Transformer model are shown in Table 4. The training and test data were tokenized using SentencePiece (Kudo and Richardson, 2018), which was trained on a random sample of 10M sentence pairs from the patent parallel corpus. The vocabulary size was set to 32K for both Japanese and English.

Hyperparameter	Value
architecture	Transformer_vaswani_wmt_en_de_big
enc-dec layers	6
optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.98$)
learning rate schedule	Inverse square root decay
warmup steps	4,000
max learning rate	0.001
dropout	0.3
gradient Clip	1.0
batch Size	16K tokens
max number of updates	60K steps
validate interval updates	1K steps

Table 4: Hyperparameters of the Transformer model

3.4.2 LLMs

For comparison, we used models in which youko-8b was continually pre-trained on JParaCrawl v3.0. After the continual pre-training, we performed supervised fine-tuning in two ways: one using the WMT20 test set and other datasets, and the other using patent claims. These models served as baselines in our experiments.

3.4.3 Prompt

When using the prompt format for the inference described in Section 3.2 for the SFT training data, numbers that did not exist in the source sentences appeared at the beginning of the output sentences. Specific examples of this phenomenon are provided in Appendix B. While the exact cause of this issue remains unclear, this phenomenon occurs in Japanese-to-English translations regardless of the data used for the continual pre-training or SFT. Thus, it is hypothesized that this behavior may be attributable to the Japanese continual pre-training process of the youko-8b model. To determine if it is possible to suppress the occurrence of such extraneous numbers in the output, we conducted additional inference experiments by modifying the prompts to the format shown below.

> これを日本語から英語に翻訳してください。ただし文頭に関係のない数字を出さないようにしてください。:
>
> 日本語: {Japanese_text}
>
> 英語:

The English translation of the above prompt is:

> "Translate this from Japanese to English. However, do not start the sentence with an irrelevant number."

3.5 Investigation of Required Data Volume for Continual Pre-training

In this study, approximately 61 million sentence pairs of patent data were used for continual pre-training. To investigate how much data is necessary for effective continual pre-training, we saved checkpoints every 0.1 epoch (i.e., every 6.1M sentence pairs) during the training process. SFT was then applied to each of these intermediate checkpoints, and the translation performance was compared. For reference, the translation accuracy of the model where SFT was applied to youko-8b without any continual pre-training is denoted as the result at "0 sentence pairs".

In addition to the original time-ordered data, we also experimented with two alternative data orderings: reversed chronological order and random order. The same procedure was applied to these variations to examine how the order of training data affects the effectiveness of continual pre-training.

3.6 Evaluation Metrics

For evaluation metrics, we employed BLEU (Papineni et al., 2002) and COMET (Rei et al., 2022). The BLEU scores were calculated using sacreBLEU (Post, 2018), whereas the COMET scores were obtained with the wmt22-comet-da model. Additionally, we analyzed win/lose cases by comparing the baseline translation results and the translation results of the LLM with the highest system-level scores, evaluating them at the sentence level for both BLEU and COMET.

4 Evaluation Results

4.1 Results of Continual Pre-training and SFT

Table 5 shows the translation accuracy achieved through the continual pre-training and SFT. The results indicate that models pre-trained with patent data significantly improved the BLEU scores compared with those pre-trained with JParaCrawl. Specifically, the BLEU score for the model pre-trained with patent data and fine-tuned with patent claims using full fine-tuning reached 50.7, compared with 43.5 for the model pre-trained with JParaCrawl. Similarly, LoRA fine-tuning achieved a BLEU score of 51.3 with patent data, significantly outperforming the 43.8 obtained with JParaCrawl. These results demonstrate that the continual pre-training on patent data effectively enables the model to acquire domain-specific knowledge.

Performing SFT with patent claims resulted in statistically significant improvements ($p < 0.05$) in the BLEU scores over the baseline model, achieving a BLEU score of 50.2. Among the SFT methods, LoRA achieved the highest BLEU score of 51.3, whereas full fine-tuning achieved 50.7. Although LoRA demonstrated superior BLEU scores, the COMET scores favored full fine-tuning, with values of 80.79 for LoRA and 81.25 for full fine-tuning.

When the inference prompt was improved, as described in Section 3.4.3, both the BLEU and COMET scores increased across all SFT methods. After prompt improvements, the BLEU score for LoRA increased to 52.3, and that of full fine-tuning improved to 52.0. Similarly, the COMET scores increased to 82.52 for LoRA and 82.55 for full fine-tuning. The analysis of the outputs revealed that the improved prompt successfully eliminated extraneous numbers at the beginning of sentences, which contributed positively to the translation quality. Examples of outputs before and after prompt modification are provided in Appendix B.

As an additional experiment, we randomly selected 100 test samples and translated them using GPT-4o to compare its performance with the proposed method. The GPT-4o translation was conducted under two conditions: (1) **Zero-shot Translation**, where the model was prompted to generate translations without any additional context, and (2) **Three-shot Translation**, where three example translations were randomly selected from the SFT training data and provided as in-context examples for few-shot translation. This comparison was conducted to provide a reference point for the translation accuracy of commercially available LLMs. Given the results of WMT23, where GPT-4 demonstrated superior translation performance compared to existing models, we aimed to assess how well GPT-4o performs specifically on patent claims. Additionally, we investigated the extent to which its performance improves with a 3-shot prompt and how our proposed approach compares to it. The translation results were evaluated using BLEU and COMET scores and compared against both the baseline and the model that achieved the highest translation accuracy in Table 5, which is referred to as the *Proposed Method* and shown in Table 6. As a result, in terms of BLEU, even with three-shot translation, GPT-4o exhibited a statistically significant drop in scores compared to both the baseline and the proposed method. However, in terms of COMET, no such trend was observed, and the difference was not statistically significant.

4.2 Required Data Volume for Continual Pre-training

4.2.1 Quantitative Evaluation

The BLEU and COMET scores for each data ordering (time-ordered, reversed-order, and random) are compared in Figures 2 and 3, respec-

Training Method	BLEU	COMET
baseline model	50.2	81.92
Continual Pre-training + SFT (Method)		
JParaCrawl + WMT (Full)	38.0	81.42
JParaCrawl + WMT (LoRA)	34.2	80.70
JParaCrawl + patent claims (full)	43.5	81.36
JParaCrawl + patent claims (LoRA)	43.8	81.37
patent + patent claims (full)	50.7∗	81.25
patent + patent claims (LoRA)	51.3∗	80.79
patent + patent claims (full) + prompt improvement	52.0∗	**82.55**∗
patent + patent claims (LoRA) + prompt improvement	**52.3**∗	82.52∗

Table 5: BLEU and COMET scores for each training method. ∗ indicates a significant difference from the baseline ($p < 0.05$).

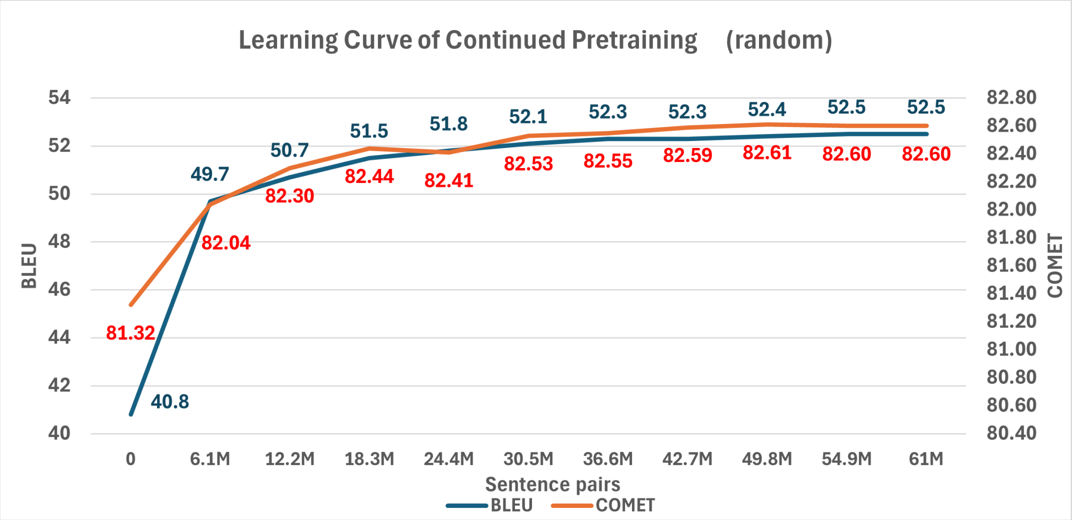

Figure 1: Learning Curve of Continual Pre-training (random)

Models	BLEU	COMET
baseline (Transformer enc-dec)	54.5∗	0.8296
proposed method	59.3∗	0.8345
GPT-4o (zero-shot)	44.8	0.8282
GPT-4o (three-shot)	48.2	0.8324

Table 6: Comparison of translation by GPT-4o. ∗ indicates a significant difference from the GPT-4o (Three-shot) ($p < 0.05$).

tively. Based on these two figures, the randomly ordered data yielded the best overall translation performance. Figure 1 shows the translation evaluation results when supervised fine-tuning (SFT) was conducted at every 6.1M sentence pairs using the randomly ordered data. For reference, the full results and specific values for the time-ordered and reversed-order settings are provided in Appendix C.

At “0 sentence pairs”, i.e., where the base model (youko-8b) was directly fine-tuned with patent claim data, the BLEU score was 40.8. However, by 6.1M sentence pairs, the BLEU score had increased to 49.7, demonstrating that even a small amount of data significantly improved translation accuracy through the continual pre-training.

BLEU and COMET scores showed a substantial increase up to 30.5M sentence pairs, achieving approximately 90% of the total performance gain observed. Beyond this point, BLEU and COMET scores continued to rise, albeit more gradually.

4.2.2 Qualitative Evaluation

As a qualitative evaluation, we compared the translation results of the models subjected to SFT at various stages: before the continual pre-training, and at 24.4M sentence pairs, 42.7M sentence pairs, and 61M sentence pairs of continual pre-training. Specific examples are presented in Table 8. These examples demonstrate significant improvements in translation quality after the continual pre-training compared with that be-

COMET difference	#cases	by Baseline	by LLM
0.1 to 0.2, LLM win	613	0.5728	0.9756
0.1 to 0.2, LLM lose	355	0.9802	0.7785
0.2 or higher, LLM win	203	0.3853	0.9864
0.2 or higher, LLM lose	380	0.7320	0.2618

Table 7: Median Sentence Length Ratios classified by COMET Score Differences and Win/Lose Cases

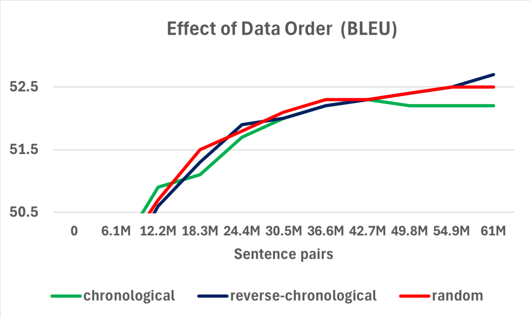

Figure 2: Effect of Data Order (BLEU)

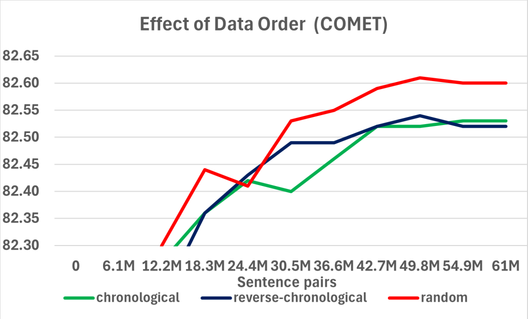

Figure 3: Effect of Data Order (COMET)

fore. This results indicates that the model acquired knowledge related to patents and parallel translations through the continual pre-training. Although the differences between the 24.4M sentence pairs and the completion of continual pre-training appeared minor in this example, variations in expression were observed, and a statistically significant improvement in the sentence BLEU scores was confirmed.

4.3 Analysis

To analyze the tendencies, sentence-level COMET and BLEU scores were calculated, and win/lose sentence sets were divided accordingly. To simplify the analysis and avoid difficulties caused by minor differences, examples with COMET differences between 0.1 and 0.2 (win/lose) and those with differences of 0.2 or higher (win/lose) were extracted, with 50 examples selected for each category. The total number of cases in which these differences occurred, along with the length ratio of the reference sentences to the translation results (baseline and LLM) for the selected 50 examples, is presented in Table 7. The length ratio between the reference and the translated sentences was calculated as a measure because many lose cases showed omissions in the translations, as observed in the selected examples.

The analysis confirmed that, as observed during the initial evaluation, translation outputs in the lose cases generally exhibited a lower length ratio than the reference sentences. This finding indicates that omissions occurred more frequently in the lose cases.

Additionally, manual inspection was conducted to provide a more detailed analysis of the specific errors in translations generated by the baseline model and the LLM.

Based on the manual inspection, among the cases where the LLM outperformed the baseline, 32 out of 50 examples exhibited omissions in the baseline translation, 5 examples showed repetition, and 13 examples contained both omissions and repetitions. Conversely, in cases where the LLM underperformed, 38 out of 50 examples exhibited omissions, 7 examples exhibited both hallucinations and omissions, and 5 examples exhibited repetition.

For example, in one case where the LLM outperformed the baseline, the source sentence described a semi-aromatic polyamide resin including multiple chemical conditions and formula-based constraints. The baseline translation retained only the formulas, such as “10 eq/t AEG+CEG 140 eq/t,” while omitting the entire description of the resin structure. In contrast, the LLM output correctly preserved the chemical structure, including “a structural unit obtained from hexamethylenediamine and terephthalic acid,” and maintained the constraints, indicating a more faithful translation.

In another representative case, the baseline output included severe repetition of the phrase “cantilever shaped” over 60 times, resulting in a clearly failed translation. The LLM translation avoided this repetition entirely, outputting a coherent description such as “with at least one cantilevered

source sentence

各生物学的成分がヌクレオチド配列または微生物株のうちの少なくとも1つである、請求項1から11のいずれか一項に記載のシステム。遺伝子改変を組み込んだ少なくとも1つの目的の産物の遺伝子製造システムにおける産生を制御するためにビルドグラフデータ構造を生成するためのコンピュータ実装方法であって、 生物学的ワークフローの記述であって、生物学的成分の表現を含む記述にアクセスすることと、 前記ワークフロー記述に少なくとも一部は基づいて、ビルドグラフデータ構造をアセンブルすることと を含み、 前記ビルドグラフデータ構造内で、各生物学的成分が、複数のレベルのうちのあるレベルにあるノードによって表され、 前記複数のレベルのうちの所与のレベルにおける1つまたは複数のソースノード、および前記所与のレベルの子レベルにおける宛先ノードが、前記子レベルに対応する、1つまたは複数の反応グループのうちのある反応グループを構成し、 各反応グループが、前記子レベルにおける前記反応グループの前記宛先ノードによって表される生物学的成分を産生するために、前記所与のレベルにおける前記1つまたは複数のソースノードによってそれ自体が表される1つまたは複数の生物学的成分間の反応を表し、 前記子レベルにおける1つまたは複数の宛先ノードが、前記所与のレベルの孫レベルにおける1つまたは複数の反応グループのうちのある反応グループ内の1つまたは複数のソースノードとして作用し、 前記複数のレベルのうちの最終レベルにおける少なくとも1つの宛先ノードが、異なるレベルにおける生物学的成分間の反応によって引き起こされた遺伝子改変を組み込んだ、前記少なくとも1つの目的の産物を表し、 前記ビルドグラフデータ構造を処理することにより、前記少なくとも1つの目的の産物が産生される、方法。前記少なくとも1つの目的の産物がヌクレオチド配列または微生物株を含む、請求項13に記載の方法。

reference translation

access a description of a biological workflow, wherein the description includes representations of biological components; and assemble a build graph data structure based at least in part upon the workflow description, wherein, in the build graph data structure, each biological component is represented by a node that resides at a level of a plurality of levels, wherein one or more source nodes, at a given level of the plurality of levels, and a destination node, at a child level of the given level, constitute a reaction group of one or more reaction groups corresponding to the child level, wherein each reaction group represents a reaction between one or more biological components that are themselves represented by the one or more source nodes at the given level, to produce a biological component represented by the destination node of the reaction group at the child level, wherein one or more destination nodes at the child level act as one or more source nodes in a reaction group of one or more reaction groups at a grandchild level of the given level, and wherein at least one destination node at a final level of the plurality of levels represents the at least one product of interest, which incorporates genetic modifications caused by reactions among biological components at different levels, and wherein processing the build graph data structure results in production of the at least one product of interest.

before continual pre-training (BLEU=0.001, COMET=50.56)

The method of claim 13, wherein the at least one product of interest comprises a nucleotide sequence or a microbial strain.

after continual pre-training with 24.4M sentence pairs (BLEU=68.04, COMET=86.63)

A computer-implemented method for generating a build graph data structure to control production in a genetic manufacturing system of at least one product of interest incorporating a genetic modification, the method comprising: accessing a description of a biological workflow, the description comprising a representation of biological components; and assembling a build graph data structure based at least in part on the workflow description, wherein within the build graph data structure, each biological component is represented by a node at a level of a plurality of levels, and wherein one or more source nodes at a given level of the plurality of levels and a destination node at a child level of the given level constitute a reaction group of one or more reaction groups corresponding to the child level, and wherein each reaction group represents a reaction between one or more biological components represented by the one or more source nodes at the given level to produce a biological component represented by the destination node of the reaction group at the child level, and wherein the one or more destination nodes at the child level act as one or more source nodes in a reaction group at one or more reaction groups at a grandchild level of the given level, and wherein at least one destination node at a final level of the plurality of levels represents the at least one product of interest incorporating a genetic modification caused by a reaction between biological components at different levels, and wherein the at least one product of interest is produced by processing the build graph data structure.

after continual pre-training (with 42.7M sentence pairs) (BLEU=68.94, COMET=86.77)

A computer-implemented method for generating a build graph data structure to control production in a genetic manufacturing system of at least one product of interest incorporating a genetic modification, the method comprising: accessing a description of a biological workflow, the description comprising a representation of biological components; and assembling a build graph data structure based at least in part on the workflow description, wherein within the build graph data structure, each biological component is represented by a node at a level of a plurality of levels, and wherein one or more source nodes at a given level of the plurality of levels and a destination node at a child level of the given level constitute a reaction group of one or more reaction groups corresponding to the child level, and wherein each reaction group represents a reaction between one or more biological components represented by the one or more source nodes at the given level to produce a biological component represented by the destination node of the reaction group at the child level, and wherein the one or more destination nodes at the child level act as one or more source nodes in a reaction group of one or more reaction groups at a grandchild level of the given level, and wherein at least one destination node at a final level of the plurality of levels represents the at least one product of interest incorporating a genetic modification caused by a reaction between biological components at different levels, and wherein the at least one product of interest is produced by processing the build graph data structure.

after completing continual pre-training (with 61M sentence pairs) (BLEU=70.75, COMET=86.8)

A computer-implemented method for generating a build graph data structure to control production of at least one product of interest in a genetic manufacturing system incorporating genetic modifications, the method comprising: accessing a description of a biological workflow, the description comprising a representation of biological components; and assembling a build graph data structure based at least in part on the workflow description, wherein within the build graph data structure, each biological component is represented by a node at a level of a plurality of levels, and wherein one or more source nodes at a given level of the plurality of levels and a destination node at a child level of the given level constitute a reaction group of one or more reaction groups corresponding to the child level, and wherein each reaction group represents a reaction between one or more biological components represented by the one or more source nodes at the given level to produce a biological component represented by the destination node of the reaction group at the child level, and wherein the one or more destination nodes at the child level act as one or more source nodes in a reaction group of one or more reaction groups at a grandchild level of the given level, and wherein at least one destination node at a final level of the plurality of levels represents the at least one product of interest incorporating genetic modifications caused by reactions between biological components at different levels, and wherein the at least one product of interest is produced by processing the build graph data structure.

Table 8: Example (1): Improvements through Continual Pre-training

beam.” This suggests that the LLM reduced unnecessary repetition, contributing to the improved translation scores.

However, when the LLM underperformed, different issues arose. In one example, the source sentence defined a chemical compound using a formula (I) and a detailed list of structural groups such as “X is C1–C6 alkyl...”, “R1 is a halo...”, and “Ar is an aryl or heteroaryl group.” While the baseline output covered all these elements almost verbatim, the LLM output stopped at “A compound of formula (I)...,” omitting all detailed structural components that followed.

A more extreme example of repetition was observed in a case involving a list of agents used to induce a stress response. The original sentence listed items from a) to y), including phrases like “interferon gamma,” “poly(IC),” and “monophosphoryl lipid A.” The baseline correctly stopped at item p) or so. In contrast, the LLM continued well beyond the source list, generating items labeled “z), aa), bb), ... lll),” all filled with repeated phrases like “lipooligosaccharide isolated from gram positive bacteria.” This artificial extension of the list demonstrates a severe repetition pattern unique to LLMs.

Although specific examples are not cited in detail here, it was also observed that in some LLM outputs, lists of detailed items were occasionally collapsed into a single concept. For instance, when the source sentence enumerated specific cancer types, the LLM sometimes generalized this into “cancer” rather than preserving individual names. This abstraction behavior, while possibly acceptable in some domains, represents a unique challenge in the accurate translation of patent claims that demand precision.

For the full outputs corresponding to the examples above, please refer to Appendix A.

5 Conclusion

This study investigated the effectiveness of LLMs for patent claim translation through the application of continual pre-training and SFT with domain-specific parallel data. The results demonstrated that LLMs, fine-tuned with patent-specific datasets, outperformed traditional Transformer encoder-decoder based models in terms of BLEU and COMET scores, thereby highlighting their superior ability to handle the intricate sentence structures and technical terminology characteristic of patent documents. A notable improvement was observed in the reduction of common translation issues such as omissions and repetitions, highlighting the capacity of LLMs to better retain and reproduce the detailed content of the source text. Furthermore, The experimental findings underscore the critical role of prompt design in enhancing translation performance, as improved prompts led to more accurate results. The study further showed the impact of data volume on the continual pre-training, indicating that substantial enhancements in translation performance can be achieved with relatively moderate data sizes. These findings provide a strong foundation for the potential of LLMs as a viable tool for high-quality patent translation tasks, contributing to advancements in the field of specialized machine translation.

References

F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang. 2022. Language-agnostic BERT sentence embedding. In *Proc. 60th ACL*, pages 878–891.

M. Fuji, A. Fujita, M. Utiyama, E. Sumita, and Y. Matsumoto. 2015. Patent claim translation based on sublanguage-specific sentence structure. In *Proc. Machine Translation Summit XV: Papers*, pages 1–16.

A. Fujii, M. Utiyama, M. Yamamoto, T. Utsuro, T. Ehara, H. Echizen-ya, and S. Shimohata. 2008. Overview of the Patent Translation Task at the NTCIR-7 Workshop. In *Proc. NTCIR*, pages 389–400.

I. Goto, Ka-Po Chow, B. Lu, E. Sumita, and Benjamin K Tsou. 2013. Overview of the Patent Machine Translation Task at the NTCIR-10 Workshop. In *Proc. NTCIR*, pages 260–286.

J. Guo, H. Yang, Z. Li, D Wei, H. Shang, and X. Chen. 2024. A novel paradigm boosting translation capabilities of large language models. In *Proc. NAACL 2024*, pages 639–649.

E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proc. 10th ICLR*, pages 1–13.

J. van der Linde G. Ramírez-Sánchez K. Heafield, E. Farrow and D. Wiggins. 2022. The EuroPat Corpus: A parallel corpus of european patent data. In *Proc. 13th LREC*, pages 732–740.

T. Kocmi et al. 2022. Findings of the 2022 conference on machine translation (WMT22). In *Proc. 7th WMT*, pages 1–45.

T. Kocmi et al. 2023. Findings of the 2023 conference on machine translation (WMT23): LLMs are here but not quite there yet. In *In Proc. of 8th WMT*, pages 1–42.

T. Kocmi et al. 2024. Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet. In *Proc. 9th WMT*, pages 1–46.

M. Kondo, T. Utsuro, and M. Nagata. 2024. Enhancing translation accuracy of large language models through continual pre-training on parallel data. In *Proc. 21th IWSLT*, pages 203–220.

T. Kudo and J. Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proc. EMNLP*, pages 66–71.

A. Lacoste, A. Luccioni, V. Schmidt, and T. Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, pages 1–8.

Y. Moslem, R. Haque, J.D. Kelleher, and A. Way. 2023. Adaptive machine translation with large language models. In *Proc. 24th EAMT*, pages 227–237. European Association for Machine Translation.

M. Nagata, M. Morishita, K. Chousa, and N. Yasuda. 2024. JaParaPat: A large-scale Japanese-English parallel patent application corpus. In *Proc. 15th LREC*, pages 9452–9462.

T. Nakazawa, C. Ding, H. Mino, I. Goto, G. Neubig, and S. Kurohashi. 2016. Overview of the 3rd Workshop on Asian Translation. In *Proc. the 3rd Workshop on Asian Translation (WAT2016)*, pages 1–46.

M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proc. NAACL*, pages 48–53.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proc. 40th ACL*, pages 311–318.

M. Post. 2018. A call for clarity in reporting BLEU scores. In *Proc. 3rd WMT*, pages 186–191.

J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He. 2023. DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters. In *Proc. 29th ACM SIGOPS*, pages 3505–3506.

R. Rei, J. C. de Souza, D. Alves, C. Zerva, A. Farinha, T. Glushkova, A. Lavie, L. Coheur, and A. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proc. 7th WMT*, pages 578–585.

NLLB Team et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv*, 2207.04672:1–192.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N Gomez, Ł. Kaiser, and I. Polosukhin. 2017. Attention is all you need. In *Proc. 31st NIPS*, pages 1–11.

H. Xu, K. Young Jin, S. Amr, and A. Hany Hassan. 2024. A paradigm shift in machine translation: Boosting translation performance of large language models. In *Proc. 12th ICLR*, pages 1–21.

X. Zhang, N. Rajabi, K. Duh, and P. Koehn. 2023. Machine translation with large language models: Prompting, few-shot learning, and fine-tuning with QLoRA. In *Proc. 8th WMT*, pages 468–481.

J. Zheng, H. Hong, F. Liu, X. Wang, J. Su, Y. Liang, and S. Wu. 2024. Fine-tuning Large Language Models for Domain-specific Machine Translation.

A Example for Analysis

This appendix provides detailed examples of translation outputs referenced in the section 4.3.

example 2-1 (case of omission)

source sentence

ヘキサメチレンジアミンとテレフタル酸から得られる構成単位、及び 11 －アミノウンデカン酸又はウンデカンラクタムから得られる構成単位を含有し、相対粘度 (RV) が式 (1) の範囲であり、アミノ基末端濃度 (AEG) 、 カルボキシ基末端濃度 (CEG) 及びモノカルボン酸でアミノ基末端を封鎖した末端濃度 (EC) の関係が式 (2) 及び (3) を満たす半芳香族ポリアミド樹脂。1.95 ≦ RV ≦ 3.50 ・ ・ (1) 10eq/t ≦ AEG+CEG ≦ 140eq/t ・ ・ (2) (AEG+CEG) / (AEG+CEG+EC) ≦ 0.50 ・ ・ (3)

reference translation

wherein the resin contains a constituent unit obtained from hexamethylenediamine and terephthalic acid and a constituent unit obtained from 11-aminoundecanoic acid or undecane lactam, wherein a relative viscosity (RV) of the semi-aromatic polyamide resin satisfies the following formula (I): 1.95RV3.50, and wherein a relationship among a concentration of terminal amino groups (AEG), a concentration of terminal carboxyl groups (CEG) and a concentration of terminal amino groups blocked by a monocarboxylic acid (EC) satisfies the following formula (II): 10 eq/tAEG+CEG140 eq/t, and the following formula (III): (AEG+CEG)/(AEG+CEG+EC)0.50.

baseline translation (BLEU=1.8, COMET=40.49)

10 eq/tAEG+CEG140 eq/t (2)(AEG+CEG)/(AEG+CEG+EC)0.50 (3)

LLM translation (BLEU=26.6, COMET=73.74)

A semi-aromatic polyamide resin comprising a structural unit obtained from hexamethylenediamine and terephthalic acid and a structural unit obtained from 11-aminoundecanoic acid or undecane lactam, wherein the semi-aromatic polyamide resin has a relative viscosity (RV) in a range of formula (1), and a relationship between an amino group terminal concentration (AEG), a carboxy group terminal concentration (CEG), and a terminal concentration (EC) obtained by blocking an amino group terminal with a monocarboxylic acid satisfies formulas (2) and (3):

example 2-2 (case of repetition)

source sentence

前記流路遮断バルブは、 内部に前記閉鎖部材を収容し、カンチレバー形状からなる少なくとも 1 つの片持ちばりを備え、円筒形に形成されて、前記連通流路の流入口に挿入されるように設置されるボディー; 及び、 一側は前記片持ちばりから突出形成される係止部により支持され、他側は前記閉鎖部材と接触するように設置されるリング部材; を含み、 前記リング部材は設定された温度以上になると、前記閉鎖部材が中心部を通過するように変形されて、前記閉鎖部材を前記連通流路の内部に向けて移動させる、 ことを特徴とする、請求項 17 に記載のバルブアセンブリ。

reference translation

The valve assembly of claim 17, wherein the flow path blocking valve includes:a body for containing the blocking member therein, and providing at least one cantilever portion formed in a cantilever shape, the body formed in a cylindrical shape and disposed to be inserted into the inlet of the communication flow path; anda ring member having one side supported by a locking portion protruding from the cantilever portion and the other side disposed to contact the blocking member, wherein the ring member moves the blocking member towards the inside of the communication flow path by deforming the blocking member to pass through a central part of the ring member when the internal temperature exceeds the preset temperature.

baseline translation (BLEU=7.7, COMET=34.42)

The valve assembly of claim 17, wherein the flow path shut-off valve comprises:a body configured to accommodate the closing member therein, having at least one cantilever shaped

LLM translation (BLEU=40.5, COMET=82.57)

The valve assembly according to claim 17, wherein the flow path blocking valve comprises:a body, which is formed in a cylindrical shape and is provided with at least one cantilevered beam, which is formed in a cantilever shape and accommodates the closing member therein, and is inserted into the inlet of the communication flow path; anda ring member, one side of which is supported by a locking portion protruding from the cantilevered beam, and the other side of which is in contact with the closing member, wherein the ring member is deformed to allow the closing member to pass through the center portion thereof when the temperature is equal to or higher than a set temperature, so that the closing member is moved toward the inside of the communication flow path.

Table 9: Example (2): LLM Outperforming Baseline

Table 9 presents representative examples in which the LLM-generated translations significantly outperformed those of the baseline model. These include instances where the baseline output exhibits severe omissions (e.g., outputting only numerical expressions and omitting structural descriptions) or extreme repetition (e.g., repeating the phrase “cantilever shaped” more than 60 times). In contrast, the LLM translations tend to preserve semantic structure more faithfully and exhibit fewer surface-level artifacts. BLEU and COMET scores are provided for each translation to support the qualitative analysis.

example 3-1 (case of omission)

source sentence

式(I):(式中、Xは、C1～C6アルキル又はC2～C6アルケニルであり;Yは、 であり;Wは、O又はSであり;R2は、H、アルキル又はアルケニルであり;Zは、ヘテロシクロアルキル又はヘテロアリール基であり、ここで、ヘテロシクロアルキル又はヘテロアリール基は任意選択的に置換されており;R1は、ハロ、シクロアルキル、ヘテロシクロアルキル、アリール又はヘテロアリール基であり、ここで、シクロアルキル、ヘテロシクロアルキル、アリール又はヘテロアリール基は任意選択的に置換されており;Arは、アリール又はヘテロアリール基である)の化合物又はその薬学的に許容可能な塩若しくはプロドラッグ。

reference translation

A compound of formula (I):or a pharmaceutically acceptable salt or prodrug thereof, wherein:X is C1-C6 alkyl or C2-C6 alkenyl;Y isW is O or S;R2 is H, alkyl or alkenyl;Z is heterocycloalkyl or heteroaryl group, which heterocycloalkyl or heteroaryl group is optionally substituted;R1 is a halo, cycloalkyl, heterocycloalkyl, aryl or heteroaryl group, which cycloalkyl, heterocycloalkyl, aryl or heteroaryl group is optionally substituted;Ar is an aryl or heteroaryl group.

baseline translation (BLEU=74.97, COMET=87.9)

A compound of Formula (I):or a pharmaceutically acceptable salt or prodrug thereof, whereinX is C1-C6 alkyl or C2-C6 alkenyl;Y isW is O or S;R2 is H, alkyl, or alkenyl;Z is a heterocycloalkyl or heteroaryl group, wherein the heterocycloalkyl or heteroaryl group is optionally substituted;R1 is a halo, cycloalkyl, heterocycloalkyl, aryl, or heteroaryl group, wherein the cycloalkyl, heterocycloalkyl, aryl, or heteroaryl group is optionally substituted; andAr is an aryl or heteroaryl group.

LLM translation (BLEU=1.64, COMET=42.52)

A compound of formula (I):or a pharmaceutically acceptable salt or prodrug thereof, wherein:

example 3-2 (case of repetition)

source sentence

ストレス応答を誘導することができる前記作用物質が、a) インターフェロンガンマ;b)IVIG;c) 単球条件培地;d) 好中球細胞外トラップに曝露された末梢血単核球からの上清;e) 単球との共培養物;f)IVIGで前処置された単球との共培養物;g)T細胞との共培養物;h)T細胞刺激に曝露されたT細胞との共培養物;i)NK細胞との共培養物;j) グラム陽性菌から単離されたペプチドグリカン;k) グラム陽性菌から単離されたリポテイコ酸;l) グラム陽性菌から単離されたリポタンパク質;m) マイコバクテリアから単離されたリポアラビノマンナン、n) 酵母細胞ウェルから単離されたザイモサン;o) ポリアデニル酸－ポリウリジル酸;p) ポリ(IC); q) リポ多糖;r) モノホスホリルリピドA;s) フラジェリン;t) ガーディキモド;u) イミキモド;v)R848;w)CpGモチーフを含むオリゴヌクレオシド;x)23S リボソームRNA; 及び y) それらの組み合わせからなる群より選択される、請求項131又は136に記載の方法。

reference translation

The method of claim 131 or 136, wherein said agent capable of inducing a stress response is selected from the group consisting of a) interferon gamma; b) IVIG; c) monocyte conditioned media; d) supernatant from neutrophil extracellular trap exposed peripheral blood mononuclear cells; e) co-culture with monocytes; f) co-culture with monocytes that have been pretreated with IVIG; g) co-culture with T cells; h) co-culture with T cells that have been exposed to a T cell stimulus; i) co-culture with NK cells; j) peptidoglycan isolated from gram positive bacteria; k) lipoteichoic acid isolated from gram positive bacteria; l) lipoprotein isolated from gram positive bacteria; m) lipoarabinomannan isolated from mycobacteria, n) zymosan isolated from yeast cell well; o) Polyadenylic-polyuridylic acid; p) poly (IC); q) lipopolysaccharide; r) monophosphoryl lipid A; s) flagellin; t) Gardiquimod; u) Imiquimod; v) R848; w) oligonucleosides containing CpG motifs; x) 23S ribosomal RNA; and y) a combination thereof.

baseline translation (BLEU=55.27, COMET=84.2)

The method of claim 131 or 136, wherein the agent capable of inducing a stress response is selected from the group consisting of: a) interferon gamma; b) IVIG; c) monocyte conditioned medium; d) supernatant from peripheral blood mononuclear cells exposed to a neutrophil extracellular trap; e) co-culture with monocytes; f) co-culture with monocytes pretreated with IVIG; g) co-culture with T cells; h) co-culture with T cells exposed to a T cell stimulus; i) co-culture with NK cells; j) peptidoglycan isolated from Gram-positive bacteria; k) lipoteichoic acid isolated from Gram-positive bacteria; l) lipoprotein isolated from Gram-positive bacteria; m) lipoarabinomannan isolated from mycobacteria, n) zymosan isolated from yeast cell wells; o) polyadenylate-polyuridylate; p) poly(IC);

LLM translation (BLEU=9.37, COMET=6.72)

The method of claim 131 or 136, wherein the agent capable of inducing a stress response is selected from the group consisting of: a) interferon gamma; b) IVIG; c) monocyte conditioned media; d) supernatant from peripheral blood mononuclear cells exposed to neutrophil extracellular trap; e) co-culture with monocytes; f) co-culture with monocytes pre-treated with IVIG; g) co-culture with T cells; h) co-culture with T cells exposed to T cell stimulation; i) co-culture with NK cells; j) peptidoglycan isolated from gram positive bacteria; k) lipoteichoic acid isolated from gram positive bacteria; l) lipoprotein isolated from gram positive bacteria; m) lipooligosaccharide isolated from gram positive bacteria; n) lipooligosaccharide isolated from gram positive bacteria; o) lipooligosaccharide isolated from gram positive bacteria; p) lipooligosaccharide isolated from gram positive bacteria; q) lipooligosaccharide isolated from gram positive bacteria; r) lipooligosaccharide isolated from gram positive bacteria; s) lipooligosaccharide isolated from gram positive bacteria; t) lipooligosaccharide isolated from gram positive bacteria; u) lipooligosaccharide isolated from gram positive bacteria; v) lipooligosaccharide isolated from gram positive bacteria; w) lipooligosaccharide isolated from gram positive bacteria; x) lipooligosaccharide isolated from gram positive bacteria; y) lipooligosaccharide isolated from gram positive bacteria; z) lipooligosaccharide isolated from gram positive bacteria; aa) lipooligosaccharide isolated from gram positive bacteria; · · · iii) lipooligosaccharide isolated from gram positive bacteria; jjj) lipooligosaccharide isolated from gram positive bacteria; kkk) lipooligosaccharide isolated from gram positive bacteria; lll) lipooligosacchar

Table 10: Example (3): LLM Underperforming Baseline

Table 10 contains examples where the LLM underperformed compared to the baseline. These include cases of significant omissions (e.g., truncating a list of chemical constituents) and repetition (e.g., extending a list of items beyond what was present in the source text by repeatedly generating the same phrase, such as “ z), aa), bb), ... lll) ”). These examples illustrate types of degradation unique to LLM outputs, particularly in structured or enumerative patent language. As in the previous table, evaluation scores are provided alongside each translation.

B Prompt

Table 11 shows the output examples from the prompt described in Section 3.2 as well as the improved prompt described in Section 3.4.3. The numbers that appear tend to correspond to the subsequent number following those present in the source text.

source sentence
細胞の単位用量が、規定の数のCD8 + /CCR7 + 細胞、CD4 + /CCR7 + 細胞、CD8 + /CD27 + 細胞、CD4 + /CD27 + 細胞、CD8 + /CCR7 + /CD27 + 細胞、および/またはCD4 + /CCR7 + /CD27 + 細胞を含む、請求項113記載の方法。
reference translation
The method of claim 113, wherein the unit dose of cells comprises a defined number of CD8+/CCR7+ cells, CD4+/CCR7+ cells, CD8+/CD27+ cells, CD4+/CD27+ cells, CD8+/CCR7+/CD27+ cells and/or CD4+/CCR7+/CD27+ cells.
translation by LLM
114. The method of claim 113, wherein the unit dose of cells comprises a defined number of CD8+/CCR7+ cells, CD4+/CCR7+ cells, CD8+/CD27+ cells, CD4+/CD27+ cells, CD8+/CCR7+/CD27+ cells, and/or CD4+/CCR7+/CD27+ cells.
translation by LLM (with improved prompt)
The method of claim 113, wherein the unit dose of cells comprises a defined number of CD8+/CCR7+ cells, CD4+/CCR7+ cells, CD8+/CD27+ cells, CD4+/CD27+ cells, CD8+/CCR7+/CD27+ cells, and/or CD4+/CCR7+/CD27+ cells.

Table 11: Translations Generated by the Prompt in Section 3.2

C Translation Evaluation Results by Data Order (Time-Ordered, Reversed-Order)

This appendix presents the detailed translation evaluation results for the different data orderings—time-ordered and reversed-order—used during the continual pre-training. For each of these settings, both the BLEU and COMET scores are provided. The figure includes a comparison of these scores, highlighting the differences in translation performance across the various data arrangements.

The results for the time-ordered and reversed-order configurations are shown in Figures 4 and 5.

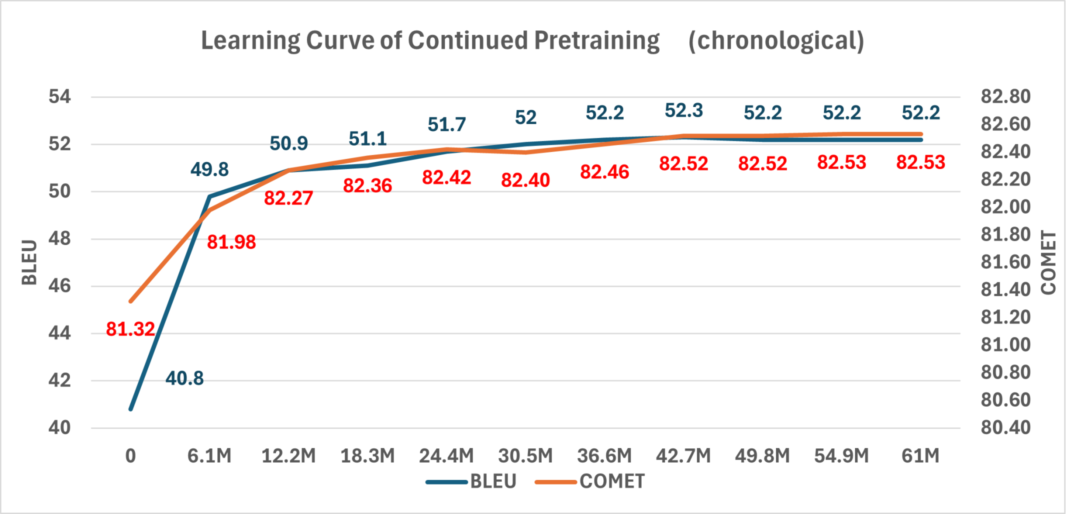

Figure 4: Learning Curve of Continual Pre-training

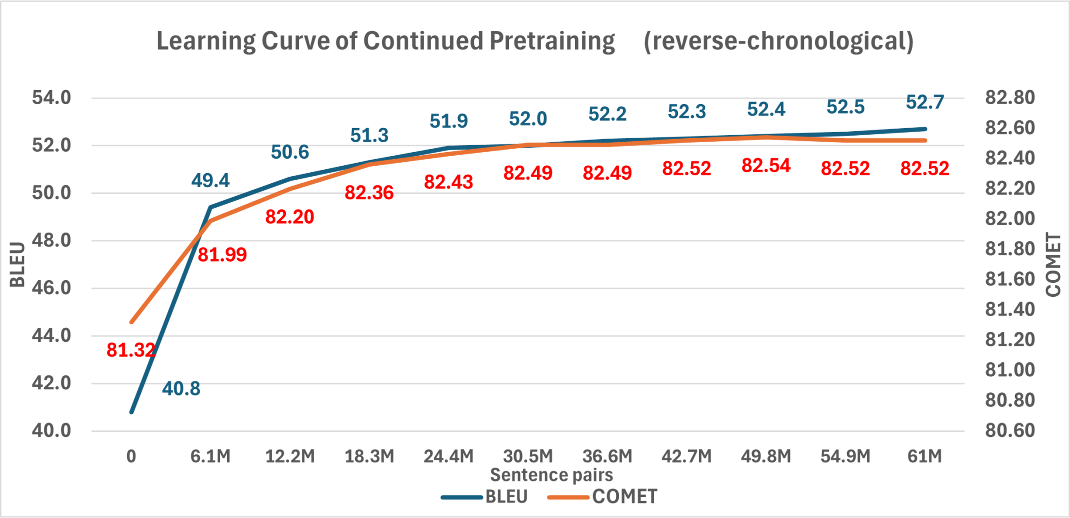

Figure 5: Learning Curve of Continual Pre-training (reverse)

D Sustainability Statement

D.1 CO2 Emission Related to Experiments

Experiments were conducted using a private infrastructure, which has a carbon efficiency of 0.432 kgCO_2eq/kWh. A cumulative of 400 hours of computation was performed on hardware of type A100 SXM4 80 GB (TDP of 400W).

Total emissions are estimated to be 34.56 kgCO_2eq of which 0 percents were directly offset.

Estimations were conducted using the MachineLearning Impact calculator presented in Lacoste et al. (2019).