LexMatcher: Dictionary-centric Data Curation for LLM-based Machine Translation

Yongjing Yin[1,2], **Jiali Zeng**[4], **Yafu Li**[2], **Fandong Meng**[4], **Yue Zhang**[2,3*]
[1]Zhejiang University
[2]School of Engineering, Westlake University
[3]Institute of Advanced Technology, Westlake Institute for Advanced Study
[4]Pattern Recognition Center, WeChat AI, Tencent Inc
{yinyongjing,liyafu}@westlake.edu.cn
{lemonzeng,fandongmeng}@tencent.com
yue.zhang@wias.org.cn

Abstract

The fine-tuning of open-source large language models (LLMs) for machine translation has recently received considerable attention, marking a shift towards data-centric research from traditional neural machine translation. However, the area of data collection for instruction fine-tuning in machine translation remains relatively underexplored. In this paper, we present LexMatcher, a simple yet effective method for data curation, the design of which is driven by the coverage of senses found in bilingual dictionaries. The construction process comprises data retrieval from an existing corpus and data augmentation that supplements the infrequent senses of polysemous words. Utilizing LLaMA2 as our base model, our method outperforms the established baselines on the WMT2022 test sets and also exhibits remarkable performance in tasks related to word sense disambiguation and specialized terminology translation. Our method is also applicable to other pre-trained models, and complements the method of continual pre-training using monolingual data, demonstrating the effectiveness of LexMatcher in enhancing LLM-based machine translation.

1 Introduction

The emergence of large language models (LLMs) (Brown et al., 2020; Touvron et al., 2023b; OpenAI, 2023) has brought about new opportunities for machine translation, and improving the translation performance of smaller-sized LLMs (7B or 13B) has attracted a lot of attention (Jiao et al., 2023; Zeng et al., 2024; Zhang et al., 2023; Xu et al., 2024). Unlike traditional neural machine translation (NMT) which relies heavily on abundant parallel data (Sennrich et al., 2016; Edunov et al., 2018; Gordon et al., 2021; Fernandes et al., 2023), LLMs have demonstrated less dependency on vast amounts of supervised data to achieve competitive performance. Similar to other tasks by LLMs (Zhou et al., 2023; Gunasekar et al., 2023), the quality of fine-tuning data plays a more crucial role in NMT (Zhang et al., 2023; Xu et al., 2024).

Current work primarily focuses on constructing fine-tuning data by leveraging human-written development sets, and creating refined instruction data for special purposes such as contrastive translation pairs and interactive translation (Zeng et al., 2024; Zhang et al., 2023). Despite these advancements, these methods do not fully exploit the potentially valuable information embedded within the existing large parallel corpus. In fact, it has been demonstrated that fine-tuning LLMs with extensive parallel data can impair their inherent translation capabilities (Xu et al., 2024). Furthermore, the quality of data distributions has been emphasized to have a more significant impact on the model performance than quantity alone (Gunasekar et al., 2023; Li et al., 2023), with more uniform data distributions contributing to improved generalization for unseen compositions (Patel et al., 2022).

Motivated by the above observations, we investigate a principled method, LexMatcher, for curating supervised fine-tuning data for LLM-based translation. The objective is to collect a small yet carefully selected dataset that follows a proper distribution for maximizing translation quality. To this end, we leverage a bilingual dictionary as a pivotal resource to ensure comprehensive coverage of word or phrase senses in bilingual contexts. The construction of the dataset involves two steps: data retrieval and data augmentation. In the data retrieval step, we traverse commonly-used corpora (e.g., WMT training data) and **identify** sentence pairs that are guided by the coverage of dictionary senses. Inevitably, however, there may be uncovered senses of polysemous words, representing long-tail knowledge essential for accurate translation. In the data augmentation step, we employ

[*] Corresponding author

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 14767–14779
November 12-16, 2024 ©2024 Association for Computational Linguistics

a commercial LLM (e.g., ChatGPT) to **generate** precise and concise sentence pairs that contain the uncovered senses. Finally, we fine-tune LLMs using a combination of the retrieved and generated data.

We conduct extensive experiments on six language directions including Zh⇔En, En⇔De, and En⇔Ru. By employing LexMatcher, we extract 0.1% of the WMT data, totaling 1 million samples across all six language directions. Results of fine-tuned LLMs on the test sets show the superiority of our method over the baselines in both standard and zero-shot settings. The fine-tuned models also achieve comparable or better performance in terminology translation and translation disambiguation compared to the dedicated or commercial systems. Further analyses of different data collection methods and composition generalization underscore the significance of high-quality data distributions, and the experiment of fine-tuning ALMA indicates the complementarity between our method and the monolingual data curation. The code, data, and models are available at https://github.com/ARIES-LM/Lexmatcher-MT.git.

2 Related Work

Data Selection for NMT. For traditional neural machine translation models, augmenting the volume of parallel data often leads to improvements in performance (Sennrich et al., 2016; Edunov et al., 2018; Gordon et al., 2021; Fernandes et al., 2023). Conversely, there have also been studies exploring data selection to reduce the size of the training corpus. For instance, van der Wees et al. (2017) gradually reduces the training data to a cleaner subset, determined by external scorers. Wang et al. (2018) introduce curriculum-based data selection that employs a trusted clean dataset to assess the noise level of each sample. Kumar et al. (2019) employ reinforcement learning to simultaneously learn a denoising curriculum and improve the NMT model. Mohiuddin et al. (2022) initially train a base NMT model on the entire available data and subsequently fine-tune the base model using selected subsets. Compared to traditional NMT, data curation is more critical for LLM-based MT, for which we make the first investigation by proposing a simple and practical method.

LLMs for MT. The usage of LLM-based MT is significantly different from the conventional NMT. LLMs, particularly large ones like GPT-4, serve as interfaces that can perform translation with simple translation instructions or in-context learning (ICL) (Lin et al., 2022; Hendy et al., 2023; Zhu et al., 2023; Agrawal et al., 2022). For ICL, the influence of data selection methods on model performance is not significantly noticeable (Zhu et al., 2023; Agrawal et al., 2022; Lin et al., 2022). Fine-tuning smaller-sized LLMs such as LLaMA (Touvron et al., 2023a) for translation has garnered increasing attention (Jiao et al., 2023; Zhang et al., 2023), which has the potential to achieve an improved trade-off between quality and efficiency. TIM (Zeng et al., 2024) constructs translation pairs for comparison and introduces an additional preference loss. Bayling (Zhang et al., 2023) automatically generates interactive translation instructions. Mao and Yu (2024) construct an additional cross-lingual discrimination task using word alignment for low-resource languages. Yang et al. (2023) fine-tune LLMs using more than 300 million parallel instances while Xu et al. (2024) indicate that such strategy could potentially impair the translation capabilities of LLMs. Instead, they propose a two-stage process that involves further post-training LLMs using a substantial amount of mixed monolingual data, followed by a subsequent step of fine-tuning with human-written parallel data.

In line with the above efforts, we also aim to improve the open-source LLMs. The difference is that we propose specific parallel data collection methods, following the principle of achieving uniform coverage of semantic units in the dictionary. Moreover, our approach achieves a better balance between efficiency and performance, and we can obtain a high-quality translation model using fewer computational resources compared to continual pre-training.

Bilingual Dictionary for NMT. Bilingual dictionaries have been employed to enhance translation quality, particularly for rare words or domain-specific entities. One approach involves augmenting the training data with pseudo-parallel sentences generated based on the dictionary. For example, Zhao et al. (2020) enhance the parallel corpus with the help of paired entities extracted from multilingual knowledge graphs. Hu et al. (2022) propose denoising entity pretraining for NMT using monolingual data and paired entities. These methods do not consult bilingual dictionaries for translation candidates during the inference stage. Another approach involves leveraging bilingual alignments

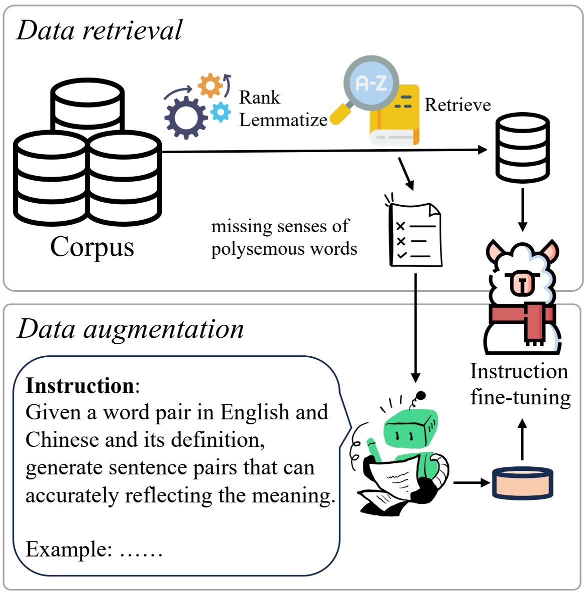

Data retrieval
Rank
Lemmatize
Retrieve
Corpus
missing senses of polysemous words
Data augmentation
Instruction:
Given a word pair in English and Chinese and its definition, generate sentence pairs that can accurately reflecting the meaning.
Example: ……
Instruction fine-tuning

Figure 1: Illustration of our LexMatcher for instruction fine-tuning smaller LLMs (e.g., LLaMA).

as lexical constraints (Li et al., 2022; Wang et al., 2022; Zeng et al., 2023). For LLMs, bilingual dictionaries have been used as a part of prompts (Lu et al., 2023; Ghazvininejad et al., 2023) for the LLMs of more than 100B. In contrast, we aim to improve LLMs' fine-tuning performance on translation tasks. The dictionaries serve as a pivot for data collection and can also be added in prompts when needed.

3 Method

The overview of LexMatcher is illustrated in Figure 1, which takes data retrieval (§3.1) and data augmentation (§3.2) steps for curating a compact parallel dataset for instruction fine-tuning.

3.1 Data Retrieval

Given a dictionary $\Phi = (s, t)$, where $\Phi = \{(s_1, t_1), (s_2, t_2), \ldots, (s_n, t_n)\}$ and each (s_i, t_i) represents a source-target segment pair, we aim to ground each pair in parallel contexts by retrieving data from a bilingual parallel dataset $D = \{(x, y)\}$. The dictionary Φ shares the same source and target languages with D. The segments can be words (e.g., "country"), phrases (e.g., "take over"), or named entities (e.g., "World Trade Organization") in the dictionary. Ideally, the objective is to find a subset $S_r \subseteq D$ such that:

$$\forall (s,t) \in \Phi, \exists (x,y) \in S_r : s \subseteq x \wedge t \subseteq y, \quad (1)$$

where $x = \{x_1, x_2, ..., x_{|x|}\}$ and $y = \{y_1, y_2, ..., y_{|y|}\}$. In practice, however, it is not guaranteed that the existing bilingual corpora can cover all dictionary senses. Therefore, we extract a subset that strives to fulfill this objective.

Algorithm 1 Data retrieval in LexMatcher

```
Input: Parallel dataset $D$, dictionary $\Phi$, threshold $K$
Output: Subset $S_r \subseteq D$
Initialize $S_r \leftarrow \emptyset$, frequency count $C \leftarrow \{\}$
for each $(x, y) \in D$ do
    Found ← false
    for each segment $\hat{x}_i$ in Lemmatize($x$) do
        for each $t_n$ in $\Phi[\hat{x}_i]$ do
            if $C[(\hat{x}_i, t_n)] < K$ and
                $t_n$ in Lemmatize($y$) then
                $C[(\hat{x}_i, t_n)] \leftarrow C[(\hat{x}_i, t_n)] + 1$
                Found ← true
            end if
        end for
    end for
    if Found then
        Add $(x, y)$ to $S_r$
    end if
end for
return $S_r$
```

We traverse the corpus in sequential order and search for potential matches with segment pairs in the dictionary. To prioritize the extraction of high-quality sentence pairs, we rank the corpus with model-based translation quality metrics, e.g., COMET-KIWI (Rei et al., 2022). Specifically, for each segment[1] in a source sentence, we perform a dictionary lookup for all the aligned target words. If one of the aligned target segments exists in the target sentence, we put the sentence pair into the translation candidate subset S_r. We lemmatize each word in the source and target sentence to alleviate the effect of morphological textual variations. In addition, we introduce a threshold K to skip the sentence if all the segment pairs in it have already been matched K times. K enables convenient control over the size of the subset and is used to encourage even distribution of segment pairs. The matching procedure is illustrated in Algorithm 1.

3.2 Data Augmentation

Using a partial set of open-source corpora cannot cover all the senses in the dictionary, which can be rare named entities or low-frequency oc-

[1] We use unigram and bigram excluding stopwords.

currence of distinctive senses of certain words. The translation of rare entities is generally unique and can be solved effectively by prompting LLMs during inference, and the lack of training data for these cases may have minimal impact. However, the senses of polysemous words are context-sensitive and may require specific training data to strengthen the model's understanding and translation of them. To compensate for the missing senses, we leverage ChatGPT[2] to construct translation demonstrations for each sense, thus creating the subset S_c. Concretely, we prompt ChatGPT with a sense expressed in source and target languages and the sense's definition. The prompt is shown in Figure 6 (Appendix B). Only nouns and verbs with more than three senses are considered due to their highly polysemous nature (Campolungo et al., 2022). Note that the subset S_c only takes up a neglectable portion of the whole dataset (e.g., 225 sentence pairs for English-Germen, and the specific numbers are reported in 5).

4 Instruction Fine-tuning LLM for MT

Instruction fine-tuning has become standard practice in LLM-based translation (Zeng et al., 2024; Xu et al., 2024; Zhang et al., 2023). Our instruction-following data is constructed based on $S = S_r \cup S_c$ (§3). Generally, each instance comprises an "instruction" c describing the task the model should perform (e.g., "Translate the sentences from English to Chinese."), an "input" x indicating the source sentence, and a corresponding output y indicating the answer to the instruction, i.e., the target sentence. The LLMs are optimized by minimizing the negative log-likelihood of the output y:

$$L = -\sum_{(x,y)\in S} \frac{1}{|y|} \sum_{i}^{|y|} \log p(y_i|c, x; \theta), \quad (2)$$

where θ is the trainable parameters.

We use two kinds of translation instructions: 1) general translation instructions mainly used to indicate translation directions (e.g., "Translate the following sentence to English"), and 2) constrained translation instructions that specify word translations from a given dictionary or based on specific user requirements. (e.g., 'Translate the following sentence to English using the given reference translations.") For the latter, we randomly sample a small number of sentence pairs to incorporate specified word translations[3]. For each sample, we introduce at most 3 segment pairs matched in the dictionary and orgnize them with a template:

$$c = \text{Template}(\{(s_i, t_i)\}_{i=1}^{N}), \quad (3)$$

where s_i and t_i denote the segment pair following Section 3. We simply use "means" to connect s_i and t_i, and prepend the constraint to the translation instruction. An example is shown in Figure 6(b) (Appendix B). During inference, we can choose whether to use the constrained translation instructions to incorporate translations from the dictionary or terminology, depending on the situation.

Lang	Raw	Dev	Retri			Aug
			K=1	K=2	K=3	
Zh	33M	15k	75k	188k	281k	2.2k
De	278M	14k	93k	233k	351k	0.2k
Ru	227M	15k	98k	246k	367k	0.7k

Table 1: The number of parallel sentences of different data sets. "Retri" and "Aug" are abbreviations for Retrieval and Augmentation, respectively.

5 Experiments

5.1 Setting

For parallel training data, we use the open-source data from WMT22[4] in German⇔English, Chinese⇔English, and Russian⇔English. The detail of data preprocessing is shown in Appendix C. We use bilingual dictionaries provided by Open Multilingual WordNet (Bond et al., 2016)[5]. In addition, we take Wikititles[6] as an entity dictionary. Table 1 presents the number of sentence pairs for each language pair in different subsets, including the original training set, subsets extracted based on different K, and the ChatGPT-generated data. It can be observed that our method achieves a high compression rate. The subset K=3 is used for the main experiment, and the extracted data for Chinese, German, and Russian accounts for only 0.57%, 0.08%, and 0.11% of the original data, respectively. The development sets from the previous WMT competitions are used as the basic fine-tuning data (Jiao et al., 2023; Xu et al., 2024).

[2] GPT-3.5-turbo-0314

[3] The maximum number of sentences under the constrained translation instructions for each direction is set to 10,000.

[4] https://www.statmt.org/wmt22/translation-task.html

[5] https://www.nltk.org/howto/wordnet.html

[6] https://data.statmt.org/wikititles/v3/

Model	Zh⇒En BLEU/COMET	En⇒Zh BLEU/COMET	De⇒En BLEU/COMET	En⇒De BLEU/COMET	Ru⇒En BLEU/COMET	En⇒Ru BLEU/COMET
GPT-3.5†	26.60/82.90	44.90/87.00	33.10/85.50	34.40/87.00	42.40/86.10	34.40/87.00
GPT-4†	27.20/82.79	43.98/87.49	33.87/85.62	35.38/87.44	43.51/86.18	30.45/88.87
NLLB-54B†	16.56/70.70	27.38/78.91	26.89/78.94	34.50/86.45	26.89/78.94	30.96/87.92
LLaMA2-7B†	18.19/75.00	16.97/71.80	30.42/82.74	19.00/76.39	36.02/82.84	16.00/73.24
LLaMA2-7B	23.59/78.54	35.43/84.28	29.04/83.63	28.58/84.09	36.68/83.58	24.23/85.54
Parrot-7B (Jiao et al., 2023)	20.20/75.90	30.30/80.30	27.30/82.40	26.10/81.60	-	-
TIM-7B (Zeng et al., 2024)	24.51/79.71	37.83/85.10	26.12/78.94	20.90/74.91	-	-
ALMA-7B (Xu et al., 2024)	23.52/**79.73**	36.48/85.05	29.49/83.98	30.31/85.59	38.93/**84.81**	27.09/87.17
LexMatcher-7B	**24.81**/79.13	**40.34/86.11**	**32.33/84.29**	**33.56/86.31**	**41.01**/84.43	**28.97/87.23**
LLaMA2-13B†	21.81/78.10	30.00/79.70	31.06/83.01	13.69/75.55	36.50/82.91	0.59/63.84
LLaMA2-13B	24.10/79.29	38.10/85.32	29.73/84.03	30.07/84.65	37.21/83.98	25.64/86.21
DictPrompt-13B (Ghazvininejad et al., 2023)	17.55/74.12	33.75/83.46	30.36/83.31	25.24/80.89	37.70/81.95	21.98/81.00
BigTrans-13B (Yang et al., 2023)	14.16/74.26	28.56/81.31	23.35/80.68	21.48/78.81	26.81/77.80	17.66/78.21
Bayling-13B (Zhang et al., 2023)	20.12/77.72	37.92/84.62	27.34/83.02	25.62/82.69	33.95/82.07	12.77/71.01
ALMA-13B (Xu et al., 2024)	25.46/**80.21**	39.84/85.96	31.14/**84.56**	31.47/85.62	40.27/**85.27**	28.96/87.53
LexMatcher-13B	**26.15**/79.88	**41.13/86.58**	**32.59**/84.55	**34.82/86.45**	**41.53**/84.91	**30.20/87.83**

Table 2: Evaluation results on WMT22 test sets. Higher scores (BLEU and COMET) denote better translation performance. Bold numbers indicate the best scores among models of the same sizes. The numbers with the dagger symbol represent the results from (Xu et al., 2024). LexMatcher-7B outperforms Parrot-7B and ALMA-7B with p-value<0.01, and LexMatcher-13B outperforms ALMA-13B with p-value<0.01.

We use LLaMA2-7B and LLaMA2-13B as the backbones. We mix the data of the different language pairs we have collected to fine-tune a unified multilingual translation model. We fine tune all of our models for 1 epoch with the collected multilingual instruction data. The batch size is 128 and the learning rate is 2e-5. The final checkpoint is used for evaluation, and we use beam search with a beam size of 4 during inference. For automatic evaluations, we use BLEU (Papineni et al., 2002) [7] and COMET[8].

5.2 Main Results

Seen Language Directions. Table 2 presents the translation performance on the WMT22 test sets. The LLaMA2 models fine-tuned on the instruction data collected by LexMatcher significantly outperform their original zero-shot performance, especially for the En⇒xx. Concretely, LexMatcher-7B improves LLaMA2-7B by an average of 17.02 BLEU points and 12.68 COMET points in En⇒xx, and by 4.45 BLEU points and 2.42 COMET points in xx⇒En. LLaMA2-13B performs significantly worse than its 7B counterpart in En⇒xx directions due to severe off-target issues, while LexMatcher-13B improves this performance significantly. We also consider an ICL method DictPrompt (Ghazvininejad et al., 2023) which provides dictionary translations for each source word[9], and the result shows that using dictionary translations as hints yields notable improvements in En⇒xx. In contrast, LexMatcher-13B achieves better performance and is more efficient due to a much shorter context during inference.

LexMatcher demonstrates superior performance compared to other instruction fine-tuned baselines. Specifically, LexMatcher-7B outperforms Parrot-7B and TIM-7B, which construct additional translation pairs and utilize specialized instructions. In the En⇒De translation task, LexMatcher-7B surpasses TIM-7B by more than 10 BLEU and COMET points. Moreover, LexMatcher outperforms BigTrans and ALMA consistently across the En⇒xx tasks, which incorporate a large amount of data for continual pretraining. While LexMatcher-7B still underperforms GPT-3.5[10] and GPT-4[11], the COMET scores for LexMatcher-7B are merely lower than GPT-3.5 within 2 points, and LexMatcher-13B further narrows the gap.

Unseen Language Directions. To evaluate performance in translation directions never seen previously, i.e., zero-shot multilingual capability, we further conduct experiments on Czech-to-English (cs⇒en), Japanese-to-English (ja⇒en), and Ukrainian-to-English (uk⇒en). As depicted in Figure 2, LexMatcher-(*) exhibits superior zero-shot multilingual capability over the LLM baselines, highlighting that better aligning training

[7] https://github.com/mjpost/sacrebleu

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

[9] They use Bloom-176B as the backbone and we re-implement the method on LLaMA2-13B.

[10] GPT-3.5-turbo-0301

[11] GPT-4-0314

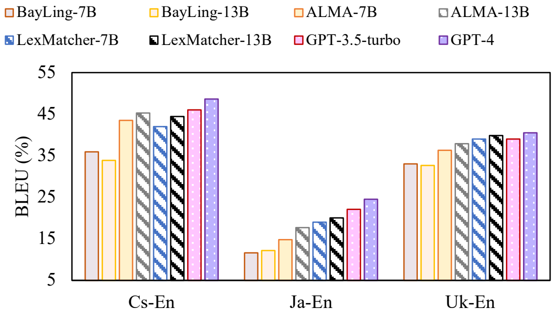


Figure 2: Zero-shot translation.

Model	Zh	De	Ru
DeepL	58.42	**76.64**	67.53
Google-Translate	52.09	67.35	62.03
OPUS	25.94	27.04	28.71
NLLB-54B	48.02	67.97	67.88
LLaMA-7B-ICL(1)	30.61	57.41	60.65
LLaMA-7B-ICL(5)	27.92	55.26	56.83
LLaMA-65B-ICL(1)	44.73	62.05	65.71
LLaMA-65B-ICL(5)	42.49	62.98	66.31
Alpaca-7B	29.63	51.52	55.23
LexMatcher-7B	53.28	63.32	67.72
LexMatcher-13B	**59.09**	66.98	**69.93**

Table 3: Accuracies on the DiBiMT benchmark which is dedicated for evaluating word disambiguation in MT. The number following ICL denotes the number of translation demonstrations.

languages strengthens the alignment of other languages as a by-product.

Disambiguation. By comparing the different senses of a word and multilingual expressions of meaning, the model possibly learns more precise word usage in translation. To investigate it, we submit the models to a challenging disambiguation leaderboard, DiBiMT (Campolungo et al., 2022). It compares the performance of NMT systems when translating sentences with ambiguous words and the performance is evaluated by accuracy. For comparison, we display the performance of top-ranked systems including DeepL[12], Google Translate[13], and NLLB-54B. The results of LLMs are from Iyer et al. (2023).

The result is shown in Table 3. For the LLaMA models, increasing model size improves the performance, and LLaMA-65B matches Google Translate and NLLB-54B with few-shot prompting. Alpaca-7B works well without demonstration (i.e., zero-shot prompting) and significantly outperforms the supervised NMT system OPUS, which indicates its potential for further improvement through fine-tuning on translation data. LexMatcher-7B significantly outperforms Alpaca-7B and surpasses Google Translate in Chinese and Russian disambiguation. With a scale of 13B, it also outperforms the best DEEPL system in Chinese and Russian, achieving accuracy rates of 59.09% and 69.93%, respectively. This result demonstrates the advantage of our data construction principle.

Model	Zh⇒En		De⇒En	
	ChrF/COMET	Suc	ChrF/COMET	Suc
Lingua Custodia	32.6/60.9	74.7	61.8/73.5	62.2
VARCO	40.5/71.5	80.0	-	-
UEDIN$_{LLM}$	**41.2/75.7**	75.3	60.0/81.3	58.8
LexMatcher-7B	38.2/73.2	84.5	64.3/81.9	80.8
LexMatcher-13B	39.1/73.6	**85.6**	**64.5/82.0**	**81.5**

Table 4: Performance on WMT23 terminology translation test sets. “Suc” indicates Terminology Success Rate.

Terminology. On the above experiments, we do not use the instruction which uses dictionary information to guide inference. In this experiment, we evaluate the effectiveness of the instruction on the test sets from WMT23 Terminology Translation Competition[14]. The numbers of sentences on Zh⇒En and De⇒En are 2640 and 2963, respectively. The average numbers of terms per segment on Zh⇒En and De⇒En are 3.8 and 1.1, respectively. The result is shown in Table 4, and we only present the systems achieving the best performance on a specific metric (Semenov et al., 2023). Lingua Custodia and VARCO are specialized Transformer architectures to ensure the appearance of given terminology in the translation, and UEDIN$_{\mathrm{LLM}}$ uses ChatGPT with terminology translation prompts. Compared to them, our models achieve significantly higher terminology success rates, indicating a superior ability to accurately respond to the given domain-specific terminology. On the quality metrics, our models are inferior to UEDIN$_{\mathrm{LLM}}$ on Zh⇒En, and achieve the best results on De⇒En.

6 Analysis

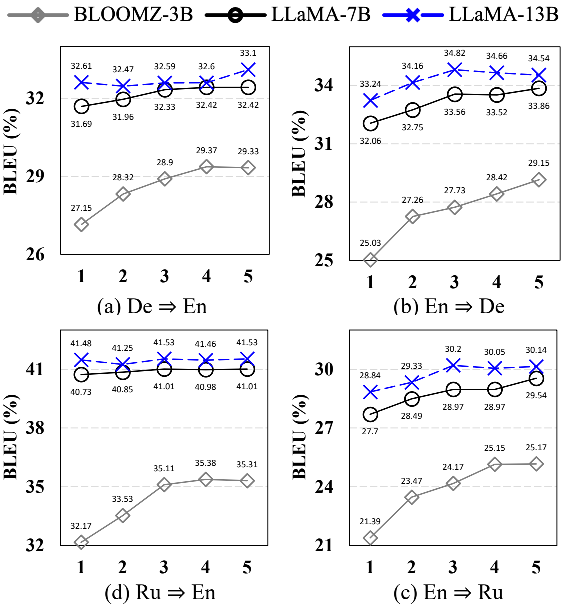
6.1 Effect of K

The maximal number of bilingual contexts of each matched sense is influenced by K. We show the performance of varying Ks across different model sizes on the WMT22 test sets (Figure 3). Regard-

[12] https://www.deepl.com/en/translator

[13] https://translate.google.com

[14] https://wmt-terminology-task.github.io/



(a) De ⇒ En

(b) En ⇒ De

(d) Ru ⇒ En

(c) En ⇒ Ru

Figure 3: BLEU and COMET on the WMT22 test sets with varying K and model sizes.

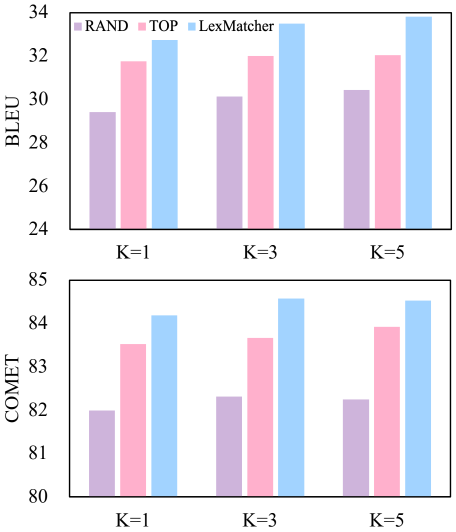


Figure 4: Performance of different selection strategies.

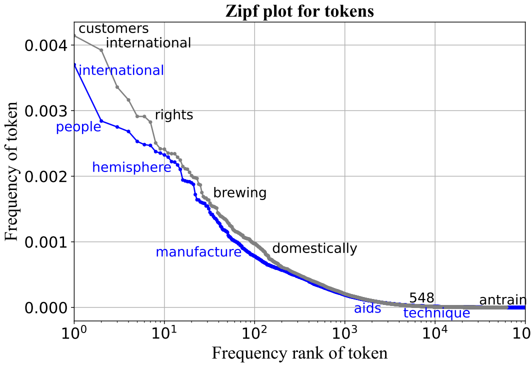


Figure 5: Word frequency distributions. The blue and gray curves denote the distributions calculated on the data selected by LexMatcher (K=1) and randomly selected data, respectively.

less of the amount of training data used, the larger models perform better and require less data for fine-tuning. In addition, the model's performance improves as K increases from 1 to 3. With the addition of more parallel data, the performance gains begin to plateau or even slightly decrease, which aligns with the conclusions of the previous study (Xu et al., 2024). Thanks to the strong few-shot learning capability of the backbones, we do not need to provide as many training examples as before when training the NMT model.

6.2 Alternative Data Selection Strategies

In this experiment, we investigate two intuitive data collection methods: 1) random selection (*RAND*), in which the training data are randomly sampled from the corpus; and 2) quality-based selection (*TOP*), in which the training samples are selected based on the COMET-KIWI scores in descending order. Specifically, we use these two methods to extract the same sample quantity as LexMatcher to mitigate the impact of sample quantity. We use LLaMA2-7B as the backbone, and the result on WMT test sets is shown in Figure 4. The performance of *RAND* is inferior to the other two methods. Random selection ensures a certain degree of diversity but the performance is uncontrollable and non-reproducible. *TOP* performs better than *RAND*, demonstrating the importance of data quality for instruction tuning. *LexMatcher* can simultaneously consider both quality and diversity and achieve the best performance.

Word Frequency Distribution We are interested in whether the collected data has a different word frequency distribution from the general (randomly selected) one. We use the English data of the EN⇒ZH translation task with K=1, and plot the word frequency distributions of the collected data (blue curve) and the corresponding random data (gray curve). As shown in Figure 5, the blue curve tends to be smoother than the gray one, and the blue curve has more flat segments. For words with higher frequency rankings, the word frequency of the data selected based on the dictionary is lower than that of the random data. This phenomenon indicates that the dictionary-based method has generated a less skewed data distribution, which could be

Model	xx⇒En BLEU/COMET	En⇒xx BLEU/COMET	DiBi-Acc
Dev	29.77/82.05	29.41/84.63	55.51
+Augmentation	30.39/82.22	30.10/84.55	55.96
+Retrieval	32.86/82.71	34.13/86.27	59.98
LexMatcher(3)	32.71/82.61	34.29/86.55	61.44

Table 5: Ablation study on different data subsets.

Model	xx⇒En BLEU/COMET	En⇒xx BLEU/COMET
ALMA	30.64/82.84	31.29/85.93
+LexMatcher(1)	32.34/83.11	33.50/86.42
+LexMatcher(2)	31.88/83.07	33.31/86.47
+LexMatcher(3)	33.37/83.32	35.30/87.09
LLaMA3-8B		
+Dev	29.50/81.68	29.16/84.13
+LexMatcher(1)	33.15/83.26	34.20/86.58
+LexMatcher(2)	33.29/83.26	35.12/87.00
+LexMatcher(3)	33.74/83.29	35.38/86.97
Gemma-2B		
+Dev	29.33/81.87	29.20/84.13
+LexMatcher(1)	31.68/82.42	31.01/84.83
+LexMatcher(2)	31.83/82.39	32.13/85.50
+LexMatcher(3)	31.93/82.43	32.33/85.66

Table 6: The performance of LexMatcher combined with different LLMs.

the reason for better fine-tuning performance. Additionally, the dictionary-based data contains 98k unique words while the random data only includes 62k unique words, indicating that the dictionary-based data covers more semantic units, thus diluting the word frequency.

6.3 Ablation Study

The ablation experiment of different data subsets is presented in Table 5 (more detailed results are reported in Table 8). We use LLaMA2-7B as the backbone. Based on the development data, simply incorporating the small amount of synthesized data generated during the data augmentation phase does not have a significant impact on the performance. This is possible because the data is predominantly focused on low-frequency senses, and the model is unable to effectively leverage this knowledge. In comparison, adding the retrieved data leads to a significant performance improvement, and further introducing the synthesized data helps the model learn word disambiguation better, increasing the disambiguation accuracy from 59.98 to 61.44.

6.4 Combination with Other LLMs

In this section, we investigate the performance of our data curation on different LLMs including ALMA-7B (Xu et al., 2024), LLaMA3-8B, and Gemma-2B (Mesnard et al., 2024), and the results are shown in Table 6. ALMA (Xu et al., 2024) is the post-trained LLaMA2 on a large amount of monolingual data mixed by different languages. We find that adding the parallel sentences constructed by LexMatcher further enhance its performance, indicating the compatibility of monolingual continual pretraining and supervised fine-tuning. Although the use of monolingual data during pretraining can reduce the dependency on bilingual data, the direct application of bilingual data for fine-tuning can be more resource-efficient. The size of parallel data collected by LexMatcher is considerably smaller than that of mixed monolingual data, and the training process is only a single stage.

Model	BLEU	Instance	Aggregate
Transformer	59.5	28.4	62.9
Transformer+CReg	61.3	20.2	48.3
LLaMA2-ICL	38.9	68.6	87.4
LLaMA2-SFT	62.4	18.5	43.9
LexMatcher	63.5	15.6	37.3

Table 7: Compound translation error rates (CTERs) on CoGnition. Instance and Aggregate denote the instance-level and aggregate-level CTERs, respectively.

6.5 Compositional Generalization

We investiage the effect of a more balanced atom distribution on CoGnition (Li et al., 2021). The evaluation metrics include instance-level CTER which denotes the translation accuracy of novel compounds, and aggregate-level CTER which measures the translation consistency across different contexts. We use the data retrieval of LexMatcher to obtain 70,272 parallel sentences from the full training data (196,246) with K=50. For LLM, we apply ICL with 8 examples and fine-tune LLaMA2-7B on the randomly sampled training data, of which the size is similar to the retrieved data. The results are shown in Table 7. ICL does not yield good compositional generalization performance, while the fine-tuned LLaMA2 outperforms the previous NMT models significantly. *LexMatcher* achieves lower compound translation error rates than SFT with the same amount of training data, demonstrating the positive effect of the more balanced data distribution.

7 Conclusion

In this paper, we presented LexMatcher, a dictionary-centric data curation method for supervised fine-tuning smaller-sized LLMs to better translation models. We use the bilingual dictionary as the pivot and try to collect limited parallel sentence pairs to cover the senses uniformly. Experiments and analyses validate the effectiveness of LexMatcher from multiple perspectives including zero-shot translation, disambiguation, and terminology translation. One potential avenue for future research involves extending LexMatcher to low-resource scenarios, where the utilization of monolingual data is crucial for achieving satisfactory translation performance.

Acknowledgements

This work has been supported by the China Strategic Scientific and Technological Innovation Cooperation Project Grant No. 2022YFE0204900 (The Macao counterpart project Grant No. FDCT/0070/2022/AMJ).

8 Limitations

This work focuses solely on improving translation performance for medium and high-resource language pairs. For low-resource language pairs that inherently lack parallel data, it is crucial to explore how to optimize LLMs on such translation tasks by integrating dictionaries, monolingual, and possible bilingual data.

References

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2022. In-context examples selection for machine translation. *CoRR*, abs/2212.02437.

Francis Bond, Piek Vossen, John McCrae, and Christiane Fellbaum. 2016. CILI: the collaborative interlingual index. In *Proceedings of the 8th Global WordNet Conference (GWC)*, pages 50–57, Bucharest, Romania. Global Wordnet Association.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher 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 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Niccolò Campolungo, Federico Martelli, Francesco Saina, and Roberto Navigli. 2022. DiBiMT: A novel benchmark for measuring Word Sense Disambiguation biases in Machine Translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4331–4352, Dublin, Ireland. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 489–500. Association for Computational Linguistics.

Patrick Fernandes, Behrooz Ghorbani, Xavier Garcia, Markus Freitag, and Orhan Firat. 2023. Scaling laws for multilingual neural machine translation. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 10053–10071. PMLR.

Marjan Ghazvininejad, Hila Gonen, and Luke Zettlemoyer. 2023. Dictionary-based phrase-level prompting of large language models for machine translation. *CoRR*, abs/2302.07856.

Mitchell A Gordon, Kevin Duh, and Jared Kaplan. 2021. Data and parameter scaling laws for neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5915–5922, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks are all you need. *CoRR*, abs/2306.11644.

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. *CoRR*, abs/2302.09210.

Junjie Hu, Hiroaki Hayashi, Kyunghyun Cho, and Graham Neubig. 2022. DEEP: DEnoising entity pre-training for neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association* *for Computational Linguistics (Volume 1: Long Papers)*, pages 1753–1766, Dublin, Ireland. Association for Computational Linguistics.

Vivek Iyer, Pinzhen Chen, and Alexandra Birch. 2023. Towards effective disambiguation for machine translation with large language models. In *Proceedings of the Eighth Conference on Machine Translation*, pages 482–495, Singapore. Association for Computational Linguistics.

Wenxiang Jiao, Jen-tse Huang, Wenxuan Wang, Zhiwei He, Tian Liang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. ParroT: Translating during chat using large language models tuned with human translation and feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15009–15020, Singapore. Association for Computational Linguistics.

Gaurav Kumar, George Foster, Colin Cherry, and Maxim Krikun. 2019. Reinforcement learning based curriculum optimization for neural machine translation. 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 2054–2061, Minneapolis, Minnesota. Association for Computational Linguistics.

Yafu Li, Yongjing Yin, Yulong Chen, and Yue Zhang. 2021. On compositional generalization of neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4767–4780, Online. Association for Computational Linguistics.

Yafu Li, Yongjing Yin, Jing Li, and Yue Zhang. 2022. Prompt-driven neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2579–2590, Dublin, Ireland. Association for Computational Linguistics.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need II: phi-1.5 technical report. *CoRR*, abs/2309.05463.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona T. Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *EMNLP 2022*, pages 9019–9052.

Hongyuan Lu, Haoyang Huang, Dongdong Zhang, Haoran Yang, Wai Lam, and Furu Wei. 2023. Chain-of-dictionary prompting elicits translation in large language models. *CoRR*, abs/2305.06575.

Zhuoyuan Mao and Yen Yu. 2024. Tuning llms with contrastive alignment instructions for machine translation in unseen, low-resource languages. *CoRR*, abs/2401.05811.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, and Shreya Pathak et al. 2024. Gemma: Open models based on gemini research and technology. *CoRR*, abs/2403.08295.

Tasnim Mohiuddin, Philipp Koehn, Vishrav Chaudhary, James Cross, Shruti Bhosale, and Shafiq Joty. 2022. Data selection curriculum for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1569–1582, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/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. Association for Computational Linguistics.

Arkil Patel, Satwik Bhattamishra, Phil Blunsom, and Navin Goyal. 2022. Revisiting the compositional generalization abilities of neural sequence models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 424–434, Dublin, Ireland. Association for Computational Linguistics.

Ricardo Rei, José G. C. de Souza, Duarte M. Alves, Chrysoula Zerva, Ana C. Farinha, Taisiya Glushkova, Alon Lavie, Luísa 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 2022, Abu Dhabi, United Arab Emirates (Hybrid), December 7-8, 2022*, pages 578–585. Association for Computational Linguistics.

Kirill Semenov, Vilém Zouhar, Tom Kocmi, Dongdong Zhang, Wangchunshu Zhou, and Yuchen Eleanor Jiang. 2023. Findings of the wmt 2023 shared task on machine translation with terminologies. In *Proceedings of the Eight Conference on Machine Translation (WMT)*. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal 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, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Marlies van der Wees, Arianna Bisazza, and Christof Monz. 2017. Dynamic data selection for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1410, Copenhagen, Denmark. Association for Computational Linguistics.

Shuo Wang, Zhixing Tan, and Yang Liu. 2022. Integrating vectorized lexical constraints for neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7063–7073, Dublin, Ireland. Association for Computational Linguistics.

Wei Wang, Taro Watanabe, Macduff Hughes, Tetsuji Nakagawa, and Ciprian Chelba. 2018. Denoising neural machine translation training with trusted data and online data selection. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 133–143, Brussels, Belgium. Association for Computational Linguistics.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024. A paradigm shift in machine translation: Boosting translation performance of large language models. In *ICLR*.

Wen Yang, Chong Li, Jiajun Zhang, and Chengqing Zong. 2023. Bigtrans: Augmenting large language models with multilingual translation capability over 100 languages. *CoRR*, abs/2305.18098.

Jiali Zeng, Fandong Meng, Yongjing Yin, and Jie Zhou. 2024. TIM: teaching large language models to translate with comparison. In *AAAI2024*, pages 19488–19496. AAAI Press.

Zixin Zeng, Rui Wang, Yichong Leng, Junliang Guo, Shufang Xie, Xu Tan, Tao Qin, and Tie-Yan Liu. 2023. Extract and attend: Improving entity translation in neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1697–1710, Toronto, Canada. Association for Computational Linguistics.

Shaolei Zhang, Qingkai Fang, Zhuocheng Zhang, Zhengrui Ma, Yan Zhou, Langlin Huang, Mengyu Bu, Shangtong Gui, Yunji Chen, Xilin Chen, and Yang Feng. 2023. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models. *CoRR*, abs/2306.10968.

Yang Zhao, Jiajun Zhang, Yu Zhou, and Chengqing Zong. 2020. Knowledge graphs enhanced neural machine translation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 4039–4045. ijcai.org.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan 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 *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Jiajun Chen, Lei Li, and Shujian Huang. 2023. Multilingual machine translation with large language models: Empirical results and analysis. *CoRR*, abs/2304.04675.

A Computational Details

We conducted experiments using the Huggingface Transformers. The experiments are performed on NVIDIA A100 GPU, and all the results are run once with the random seed 42. According to the data license of WMT22, the data released for the General MT task can be freely used for research purposes.

B Prompts Used for Manipulating ChatGPT and Terminology Translation

The prompt used to manipulate ChatGPT consists of three parts (Figure 6 (a)). The first part is used to describe the task: generate a pair of parallel sentences, which can reflect the meaning of a given segment pair accurately. The second part is an example to demonstrate the format of the input and output including a segment pair, a definition of the sense, and a sentence pair. The third part is the segment pair requires translation demonstration.

The prompt for terminology translation is shown in Figure 6 (b).

Given a pair of words that are of the same meaning but in different languages, and the definition of the meaning, please generate a pair of sentences in English and Chinese respectively, which can reflect the meaning most accurately.

Example:
  Word Pair: English: “head” - Chinese: “负责人”
  Definition: the person in charge of a group of people or an organization
  Sentence pair:
    English: She resigned as head of department.
    Chinese: 她辞去了部门负责人的职务。

Now, please generate three sentence pairs for the below word pair:
Word Pair: English: “being” - Chinese: “生物”
Definition: a living thing that has (or can develop) the ability to act or function independently
Sentence Pairs:

(a)

“head” means “负责人”; “department” means “部门”.
Translate the following sentence from English to Chinese using the given reference translations.
English: She resigned as head of department.
Chinese:

(b)

Figure 6: Prompts used for (a) manipulating ChatGPT to generate translation demonstrations and (b) terminology translation.

Model	**Zh⇒En** BLEU/COMET	**En⇒Zh** BLEU/COMET	**De⇒En** BLEU/COMET	**En⇒De** BLEU/COMET	**Ru⇒En** BLEU/COMET	**En⇒Ru** BLEU/COMET
Dev	23.59/78.94	35.43/84.28	29.04/83.63	28.58/84.09	36.68/83.58	24.23/85.54
+Augmentation	23.69/79.05	36.50/84.20	29.45/83.82	28.67/83.98	38.03/83.80	25.14/85.49
+Retrieval	25.36/79.46	40.14/86.01	32.37/84.31	33.26/85.77	40.86/84.36	29.00/87.03
LexMatcher(3)	24.81/79.13	40.34/86.11	32.33/84.29	33.56/86.31	41.01/84.43	28.97/87.23
ALMA-7B						
+LexMatcher(1)	24.27/79.82	31.77/84.52	41.00/85.01	38.61/85.83	33.12/86.19	28.77/87.25
+LexMatcher(2)	24.04/79.88	38.27/85.93	31.39/84.32	32.85/86.14	40.61/85.07	28.82/87.34
+LexMatcher(3)	25.20/80.21	41.40/86.59	32.49/84.49	34.44/86.66	42.42/85.28	30.07/88.02
LLaMA3-8B						
+LexMatcher(1)	26.40/80.47	40.30/86.11	32.44/84.52	33.16/86.09	40.63/84.79	29.15/87.54
+LexMatcher(2)	26.33/80.31	42.34/86.94	32.36/84.54	33.68/86.37	41.19/84.93	29.36/87.69
+LexMatcher(3)	26.89/80.51	41.88/86.74	32.95/84.46	34.22/86.49	41.39/84.92	30.04/87.70
Gemma-2B						
+LexMatcher(1)	24.88/79.75	37.89/85.01	31.35/83.77	29.27/83.95	38.81/83.75	25.87/85.53
+LexMatcher(2)	25.19/79.60	39.53/85.92	31.43/83.77	30.35/84.51	38.87/83.81	26.53/86.09
+LexMatcher(3)	24.84/79.55	39.19/85.98	31.77/83.80	30.81/85.04	39.18/83.95	27.00/85.98

Table 8: Detailed results of ablation study and combination with different LLMs.

C Corpus Preprocessing

Since the filtered data of Russian⇔English is significantly less than the other language pairs, we introduce the training set from Tatoeba translation challenge 2021[15]. We filter data with the commonly used rule-based methods and model-based QE. The rules include the following categories: (1) sentence-level deduplication, (2) filter out the sentences longer than 100 words or contain a single word exceeding 40 characters, (3) remove sentence pairs where the ratio of source sentence length to target sentence length is significantly different, i.e., below 1/3 or above 3, (4) filter out the sentences with high repeat ratio, i.e., the proportion of the frequency of the most frequent word in a sentence to the total word frequency greater than 0.3, and

[15] https://github.com/Helsinki-NLP/Tatoeba-Challenge/tree/v2021-08-07/data

(5) filter out the sentences in which the proportion of the content words is between 0.3 and 0.8. In this way, low-quality data can be efficiently filtered out, saving time and resources for the subsequent model-based QE.

We utilize one of the state-of-the-art QE models, COMET-KIWI[16], to obtain sentence-level quality scores. For every sentence pair in the training data, we calculate the QE score for the translation from English to the foreign language. These scores are utilized for both translation directions, as evaluating both directions of the training data can be computationally expensive. We remove sentence pairs with low data quality, e.g., those that have a score below 40. We use spaCy[17] for the lemmatization.

[16] https://huggingface.co/Unbabel/wmt22-cometkiwi-da

[17] https://spacy.io/