Exploiting Target Language Data for Neural Machine Translation Beyond Back Translation

Abudurexiti Reheman[1], **Yingfeng Luo**[1], **Junhao Ruan**[1],
Anxiang Ma[1], **Chunliang Zhang**[1,2], **Tong Xiao**[1,2*], and **Jingbo Zhu**[1,2]
[1]School of Computer Science and Engineering, Northeastern University, Shenyang, China
[2]NiuTrans Research, Shenyang, China
{rexiti_neu, luoyf98, rangehow}@outlook.com,
{xiaotong, zhujingbo}@mail.neu.edu.cn

Abstract

Neural Machine Translation (NMT) encounters challenges when translating in new domains and low-resource languages. To address these issues, researchers have proposed methods to integrate additional knowledge into NMT, such as translation memories (TMs). However, finding TMs that closely match the input sentence remains challenging, particularly in specific domains. On the other hand, monolingual data is widely accessible in most languages, and back-translation is seen as a promising approach for utilizing target language data. Nevertheless, it still necessitates additional training. In this paper, we introduce Pseudo-kNN-MT, a variant of k-nearest neighbor machine translation (kNN-MT) that utilizes target language data by constructing a pseudo datastore. Furthermore, we investigate the utility of large language models (LLMs) for the kNN component. Experimental results demonstrate that our approach exhibits strong domain adaptation capability in both high-resource and low-resource machine translation. Notably, LLMs are found to be beneficial for robust NMT systems.

1 Introduction

Neural Machine Translation (NMT) has witnessed significant progress with the adoption of deep learning techniques(Sutskever et al., 2014; Bahdanau et al., 2015), particularly the transformer model(Vaswani et al., 2017). Despite these advancements, challenges still exist in translating uncommon words and adapting NMT systems to different domains(Koehn and Knowles, 2017; Saunders, 2022).

To tackle these challenges, researchers have proposed various methods to incorporate external knowledge into NMT. One such approach involves imposing constraints from terminology dictionaries(Dougal and Lonsdale, 2020; Hasler et al., 2018), or the incorporating fuzzy matches retrieved from translation memories (TMs)(Eriguchi et al., 2019; Xu et al., 2020; Khandelwal et al., 2021; He et al., 2021; Reheman et al., 2023).

These methods enhance NMT systems by leveraging bilingual knowledge. However, due to the limitations in the scale and domain coverage of bilingual data, it is highly challenging to find sentences that closely match the input sentence, especially in specific domains or for low-resource languages. One natural idea is to utilize the vast amount of monolingual data, which can provide a pool of highly relevant sentences in terms of meaning. As a promising method, back translation (Sennrich et al., 2016) has been proven to be helpful for utilizing monolingual data in NMT systems. However, it requires additional training, including training a reverse NMT model and retraining an NMT model with the augmented training data.

In this paper, we introduce pseudo-kNN-MT, a training-free approach that leverages target language data for translation. Specifically, given an input sentence, we retrieve its top-k similar target sentences using a cross-lingual retriever. Our primary objective is to effectively utilize these retrieved sentences. Initially, we pair the retrievals with the input sentence to create pseudo sentence pairs, then perform k-nearest neighbor machine translation (kNN-MT) following Khandelwal et al. (2021). Additionally, LLMs are known for their strong text compression capabilities of mapping texts into the representation space (Brown et al., 2020; Radford et al., 2019), and their training paradigm endows them with good generalization, which might be effective in handling low-frequency patterns. Besides, LLMs also demonstrate strong translation capabilities (Zhang et al., 2023; Zhu et al., 2023a; Xu et al., 2023). Therefore, we take a step further to investigate the potential of utilizing LLMs for the kNN component, where the datastore and the context representation vectors are derived from

[*] Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2024, pages 12216–12228
August 11-16, 2024 ©2024 Association for Computational Linguistics

LLMs rather than the NMT model itself. Furthermore, we explore the integration of LLMs with NMT without relying on target retrieval, focusing on leveraging the translation ability of LLMs and enhancing translation fluency. Experimental results on multi-domain test sets demonstrate that our approach improves the translation results with a great margin, achieving an average improvement of 4.51 sacreBLEU points. In low resource MT scenarios, our method's performance is comparable to or even superior to back-translation in certain domains.

2 Background

In this section, we provide background information on kNN-MT and cross-lingual retrieval.

2.1 kNN-MT

The kNN-MT Khandelwal et al. (2021) is a non-parametric method that utilizes nearest neighbor retrievals from a vector datastore of translation context representation. It involves two main steps: datastore creation and inference.

Datastore Creation The datastore $\mathcal{D}$ comprises a collection of key-value pairs, where the key is a high-dimensional representation of a translation context. This key is computed by an auto-regressive MT decoder, and the value is the corresponding ground-truth target token. Here, the combination of source language tokens and the generated target tokens is called translation context. Let $(\mathcal{X}, \mathcal{Y})$ be a set of bilingual sentences, and let $f(\cdot)$ be a mapping function that transfers the translation context into the high-dimensional representation using a translation model. For all examples in $(\mathcal{X}, \mathcal{Y})$, the key-value datastore is created as:

$$\begin{aligned}\mathcal{D} = \{(f(x, y_{1:t-1}), y_t), \forall y_t \in y| \\ (x, y) \in (\mathcal{X}, \mathcal{Y})\}.\end{aligned} \quad (1)$$

Inference During the inference phase, the translation context representation of each time-step is used as a query, $q = f(x, \hat{y}_{1:t-1})$, to retrieve k-nearest neighbors $\mathcal{N}$ from $\mathcal{D}$, employing vector distance measuring techniques likes $L2$ distance. Subsequently, a probability distribution, $p_{k\mathrm{NN}}$, over the target vocabulary is then generated from $\mathcal{N}$ by applying a softmax with temperature to the negative distances and aggregating the same tokens, defined as:

$$p_{k\mathrm{NN}}(y_t|x, \hat{y}_{1:t-1}) = \frac{\sum_{(k_j,v_j)\in\mathcal{N}} \mathbb{1}_{y_j=v_j} exp(-d(q, k_j)/T)}{\sum_{(k_j,v_j)\in\mathcal{N}} exp(-d(q, k_j)/T)}, \quad (2)$$

where $d(\cdot, \cdot)$ represents a distance function that calculates the distance between the two vectors, specifically the query vector and the retrieved neighbors.

In the end, the final probability distribution is obtained by linear interpolating the two distributions, $p_{k\mathrm{NN}}$ and p_{NMT}, using a tuned hyperparameter λ:

$$\begin{aligned}p(y_t|x, \hat{y}_{1:t-1}) = \lambda p_{k\mathrm{NN}}(y_t|x, \hat{y}_{1:t-1}) + \\ (1-\lambda) p_{\mathrm{NMT}}(y_t|x, \hat{y}_{1:t-1}).\end{aligned} \quad (3)$$

2.2 Cross-lingual Retrieval

Cross-lingual retrieval is the process of retrieving information from multilingual sources (Feng et al., 2022a; Li et al., 2023; Gao et al., 2023). Its core is a pretrained cross-lingual sentence embedding model, which maps the sentences from different languages into a shared semantic space. During application, it returns the embedding of "CLS" token or employs a mean pooling strategy on all token embeddings within the sentence to capture the sentence's representation. This technique proves valuable in various cross-lingual applications, such as information retrieval and machine translation (MT). In this paper, we utilize it to retrieve similar sentences from the target language by taking the source sentence as a query.

3 Methodology

In this section, we will introduce retrieving similar sentences from target dataset (§3.1), as well as the proposed method of pseudo kNN-MT (§3.2) and the large language model integration (§3.3 and §3.4) in detail.

3.1 Retrieving Similar Sentences from Target Language Dataset

Given an input sentence x, a target language dataset $\mathcal{Y} = \{y^1, y^2, ..., y^n\}$, and a cross-lingual sentence embedding model e. First of all, the distributed representation of the target dataset, $h_\mathcal{Y} = \{h_1, h_2, ..., h_n\}$, is obtained by feeding $\mathcal{Y}$ into the cross-lingual model, as:

$$h_\mathcal{Y} = e(\mathcal{Y}). \quad (4)$$

Similarly, we obtain the distributed representation of x as $h_x = e(x)$. Subsequently, we calculate the

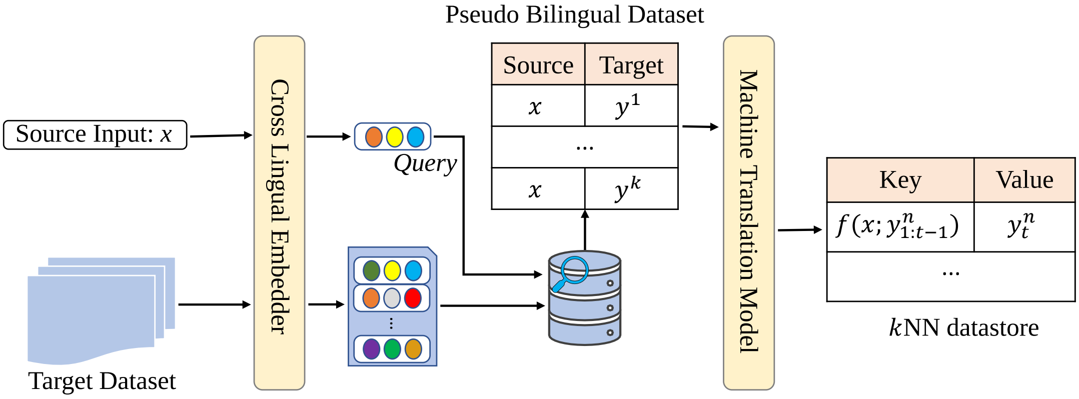

Figure 1: pseudo datastore creation process. Function $f(\cdot)$ returns the last hidden state of MT decoder at every time-step.

distances between each item in $h_\mathcal{Y}$ and h_x using the distance function $d(\cdot)$:

$$D = d(h_x, h_\mathcal{Y}), \tag{5}$$

where $D = \{d_1, d_2, ..., d_n\}$ represents the distance of each sentence in y from x in the vector space. Finally, we acquire the top-k similar sentences by ranking them based on their distances and selecting the k-nearest ones as the final retrieval. Our work is focused on the utilization of these target retrievals.

3.2 kNN-MT with Pseudo Datastore

After obtaining similar sentences from the target dataset, we aim to construct bilingual dataset in order to align with the decoding behavior of the NMT model. Due to the semantic resemblance between the retrieved sentences and the input sentence, we pair them up to form bilingual dataset. After this, we explore whether this pseudo bilingual dataset can effectively facilitate the translation, following the approach of kNN-MT (Khandelwal et al., 2021).

Specifically, we build a key-value datastore $\mathcal{D}_{\text{pse}}$ based on the pseudo bilingual dataset. Suppose $\mathcal{Y}_{\text{sim}} = \{y^1, y^2, ..., y^k\}$ is the target retrieval for the input sentence x, the pseudo bilingual dataset is constructed by pairing x with each sentence in $\mathcal{Y}_{\text{sim}}$, as $(\mathcal{X}, \mathcal{Y})_{\text{pse}} = \{(x, y_i) | y_i \in \mathcal{Y}_{\text{sim}}, i \in [1, k]\}$. The kNN datastore on $(\mathcal{X}, \mathcal{Y})_{\text{pse}}$ is built using the equation 1, as defined below:

$$\begin{aligned} \mathcal{D}_{\text{pse}} \quad = \quad & \{(f(x, y_{1:t-1}), y_t), \forall y_t \in y | \\ & (x, y) \in (\mathcal{X}, \mathcal{Y})_{\text{pse}}\}, \end{aligned} \tag{6}$$

where $f(\cdot)$ also is the mapping function from translation context to the last hidden state of the NMT decoder, and t is the decoding time-step. The creation process of the pseudo datastore is illustrated in Figure 1.

During the inference phase, we construct the target token distribution from $\mathcal{D}_{\text{pse}}$ and interpolate it with the NMT model's distribution, utilizing equation 2 and equation 3 respectively, in the same way as kNN-MT (Khandelwal et al., 2021).

3.3 kNN-MT with LLM Pseudo Datastore

As a dual-model approach, kNN-MT is the combination of an NMT model and a non-parametric kNN translation model sourced from a datastore. Unlike the naive implementation that relies on the NMT model's own hidden states for constructing the key-value datastore and retrieving during inference, any MT model can be used for this process. Here, we explore the potential of using LLMs for the kNN component. Firstly, we construct the key-value datastore using an LLM. Specifically, we feed the pseudo bilingual sentences into the LLM with a specific prompt and extract the hidden states of the translation context at each time step as the key and the corresponding target token as the value. Due to the differences between utilizing LLMs for translation tasks and NMT, where LLMs require instructions to specify the desired translation task, including the support for zero-shot and few-shot learning, the translation context here differs from that in NMT. For zero-shot, the translation context will be:

> Translate this from [source language] into [target language] and return the translation results only.
>
> [source language]: [source sentence]

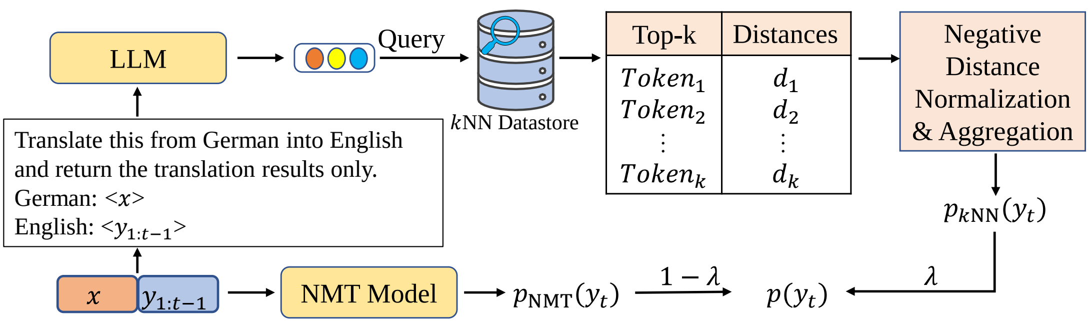

Figure 2: Illustration of decoding using LLM datastore. Here, we take zero shot prompt as an example. The kNN datastore is constructed offline using the LLM on the pseudo bilingual dataset.

[target language]: [previously generated target tokens]

In the few-shot scenario, few-shot examples come after the instruction. It is worth mentioning that when constructing the key-value datastore, we should use the same prompt that was used during the inference to maintain key representation consistency.

At each time step of inference phase, we first construct the translation context using the same prompt as mentioned above. The translation context is then fed into the LLM to extract the hidden state. Subsequently, we take this hidden state as a query to search for the k-nearest neighbors from the datastore and obtain the kNN probability, which is interpolated with NMT probability afterwards. The illustration of the inference phase is given in Figure 2.

3.4 LLM Integration

As a language model, LLMs have strong capabilities in next token prediction. Additionally, they also can process multilingual information, such as machine translation, with proper human instructions. With this knowledge, we further explore LLM integration without additional data. For an input sentence x and previously generated target tokens $y_{1:t-1}$, our method operates as follows.

LLM Translator Interpolation In this method, we make use of the translation abilities of LLMs. At each time-step of inference, we utilize both x and $y_{1:t-1}$ to construct the prompt for the LLM. The prompt here is the same with the translation context for LLM as outlined in Section 3.3. Subsequently, the prompt is fed into the LLM and then the LLM generates its probability distribution for y_t. Finally, we combine the LLM probability p_{LLM} with the NMT probability p_{NMT} through interpolation using a hyperparameter λ:

$$p(y_t|x,\hat{y}_{1:t-1}) = \lambda p_{\text{NMT}}(y_t|x,\hat{y}_{1:t-1}) + (1-\lambda)p_{\text{LLM}}(y_t|x,\hat{y}_{1:t-1},pr), \quad (7)$$

where pr represents the prompt template.

LLM Continuation Generator Fusion In this method, we leverage the LLM's capabilities for generating continuations without relying on the source language information. This means that the generation of the next token is only conditioned on $y_{1:t-1}$. At each time step of inference, we feed $y_{1:t-1}$ into the LLM to obtain its next token probability p_{LLM}. The final translation probability for y_t is calculated by adding this probability with the generation probability of NMT, p_{NMT}, using a hyperparameter λ as:

$$p(y_i|x,\hat{y}_{1:t-1}) = p_{\text{NMT}}(y_t|x,\hat{y}_{1:t-1}) + \lambda p_{\text{LLM}}(y_t|y_i|\hat{y}_{1:t-1}). \quad (8)$$

4 Experiments

In this section, we will introduce our experiments, including the main experiment, LLM integration, low-resource machine translation and a comprehensive analysis from various perspectives.

Datasets and Evaluation Metrics We evaluated the effectiveness of our proposed method on publicly available datasets. For domain adaptation, we performed experiments on IT, Koran, Law, and Medical domains of multi-domain datasets provided by Aharoni and Goldberg (2020). To measure translation quality, we used sarcreBLEU (Post, 2018) and COMET (Rei et al., 2022). The data statistics are given in table 1.

Split	Multi-domain				WMT19
	IT	Koran	Law	Medical	
Train	223K	17K	467K	248K	33M
Valid	2000	2000	2000	2000	6002
Test	2000	2000	2000	2000	2000

Table 1: Statistics of datasets.

Models We use the winner model of the WMT19 De-En news translation task, submitted by Facebook, as the pretrained base NMT model (Ng et al., 2019). For LLM, we used various versions of LLAMA 2 (Touvron et al., 2023), including the base version LLAMA-2-7B, dialogue optimized version LLAMA-2-7B-chat and ALMA-7B (Advanced Language Model-based trAnslator) Xu et al. (2023), a translation optimized model from LLAMA-2-7B, respectively. We encountered difficulties when integrating the NMT model with Llama 2. Facebook's WMT19 De-En model cannot be interpolated with LLM directly, because the two models have differences in tokenization strategy, word granularity, and training data, leading to differences in the dictionary of the two models. Therefore, we trained another NMT model on WMT19 De-En training data, using the dictionary of Llama 2. Additionally, we trained a decoder-only transformer language model (Radford et al., 2019) with 12 layers and a model dimension of 768 on the target data of the WMT19 De-En dataset and Llama 2 dictionary as well to ensure a fair comparison. Before training, We cleaned WMT19 training data by applying punctuation normalization and language identification filtering. Subsequently, we tokenized them using `llama.tokenizer`.

Settings We utilize the cross-lingual embedding model LaBSE (Feng et al., 2022b) to transfer both the source and target language datasets into embedding representations. Subsequently, we employ the dense vector similarity search library, FAISS (Johnson et al., 2021), to perform cross-lingual retrieval. For k-nearest neighbor searching from the kNN datastore, we also rely on FAISS. In all experiments, for retrieve top 32 similar sentences from the target dataset. For kNN model, we retrieve $k = 8$ neighbors from the vector datastore. Regarding the kNN temperature, we followed the optimized settings from Zheng et al. (2021), setting it to 100 for Koran, and 10 for other domains. For the interpolation hyperparameter λ , we search it from $\lambda \in \{0.2, 0.3, 0.4\}$ in LLM datastore method, and other methods searches from $\lambda \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. For decoding, we set the *beam size* to 5, and *length penalty* to 1.0.

We take vanilla NMT (Base-NMT) and vanilla kNN-MT (kNN-MT) as the baselines. To simulate the usage of monolingual data, we use the target language training data as the monolingual dataset. The other methods compared are as follows:

Pseudo-kNN-MT: the method that introduced in Section 3.2.

Retrieve-bt-kNN-MT: a variant of Pseudo-kNN-MT. In this method, the retrieved similar target sentences are back translated into the source language sentences to construct bilingual sentence pairs, which are then used to construct vector datastore.

Mono-bt-kNN-MT: a variant of kNN-MT, whose datastore is created from a bilingual dataset whose source sentences are obtained by translating the target dataset back into source language.

4.1 Main Experiment

In this experiment, we evaluate our method on the test set of the multi-domain dataset. The NMT model is Facebook's WMT19 De-En model, while Facebook's WMT19 En-De model (Ng et al., 2019) is used to translate the target dataset into source language. The experimental results in sacreBLEU scores are presented in Table 2. The COMET scores for this experiment can be found in Appendices B.

The experimental results indicate that although the performance is not as good as the vanilla kNN-MT, our method, Pseudo-kNN-MT, can improve sacreBLEU scores by an average of 4.51 BLEU points compared to the NMT baseline. This improvement seems reasonable intuitively because the pseudo-bilingual sentences are similar or relevant in semantics, although not the exact matches. However, the datastore of kNN-MT is constructed from ground truth bilingual dataset. Compared to Pseudo-kNN-MT, Retrieve-bt-kNN-MT constructs the datastore on machine-translated bilingual dataset, which can further boost average BLEU scores by 0.75 points. Furthermore, Mono-bt-kNN-MT can yield an additional improvement of 0.34 BLEU points. However, this also implies a higher computational cost.

Methods	IT	Koran	Law	Medical	Average
NMT	38.43	17.07	45.99	41.97	35.86
kNN-MT	$46.74_{(0.7)}$	$21.93_{(0.7)}$	$61.92_{(0.9)}$	$56.40_{(0.8)}$	46.75
Pseudo-kNN-MT	$40.63_{(0.3)}$	$18.46_{(0.4)}$	$53.03_{(0.4)}$	$49.36_{(0.5)}$	40.37
Retrieve-bt-kNN-MT	$41.53_{(0.8)}$	$19.44_{(0.8)}$	$54.49_{(0.8)}$	$49.02_{(0.8)}$	41.12
Mono-bt-kNN-MT	$41.58_{(0.5)}$	$20.35_{(0.7)}$	$54.43_{(0.9)}$	$49.47_{(0.7)}$	41.46

Table 2: SacreBLEU scores of Facebook's WMT19 De-En model on the multi-domain test sets. The numbers in the parentheses at the bottom-right indicate that the model yielded the best translation performance when the hyperparameter lambda for interpolation is this value.

4.2 LLM Integration

In this experiment, we validate the efficacy of integrating NMT model and LLM on the multi-domain test sets. To maintain vocabualry consistency between NMT model and LLM for interpolation, we utilize WMT19 De-En Llama-2 dictionary model as the base NMT model, as detaild in Subsection 4. We explore different LLM integration approaches, such as interpolation via kNN-MT with a pseudo datastore constructed by LLM, leveraging the translation capabilities of the Llama model, and fusion using LLM as a continuation generator, on Llama2, Llama2-chat, and ALMA models, respectively. The experimental results are presented in Table 3.

The results from the base models indicate that all three Llama models exhibit weaker translation performance compared to the NMT model, including the translation-optimized ALMA model. Due to the utilization of the Llama dictionary in training the base NMT model, its performance showed an average decrease of 1.67 BLEU points compared to Facebook's WMT19 model. In this study, to ensure a fair comparison with the kNN-MT method using LLM pseudo datastore, we also validate Pseudo-kNN-MT. Pseudo-kNN-MT demonstrate a significant enhancement compared to the base NMT, with an average improvement of 4.37 BLEU points on a slightly less robust NMT model. Retrieve-bt-kNN-MT and Mono-bt-kNN-MT further enhance the performance over Pseudo-kNN-MT.

Although the performance did not reach that of pseudo-kNN-MT, the utilization of LLMs in the kNN component generally outperform its NMT baseline. This validates the effectiveness of kNN-MT with LLM pseudo datastore. Moreover, it is evident that the performance enhancement becomes more significant with "strong" LLMs. This suggests that the performance of the kNN component is related to the translation capability of the LLM; more robust translation models demonstrate superior translation context compressing ability, leading to greater performance enhancements.

Within the interpolation of the LLM translators, all three models can improve NMT translation to varying extents on zero-shot and few-shot scenarios, with such enhancement being notably obvious in the more advanced ALMA model. Concurrently, optimal translation results are achieved on larger λ values for the stronger LLM translators, indicating that latter can provide more translation knowledge to the NMT.

In the experimentation of fusing language models as text continuators, the Llama2 model, owing to its strong generative capability, assists in generating better translations, exhibiting an average improvement of 1.07 BLEU points over the base NMT. Conversely, conventional generative language models decrease the average BLEU score by 0.83 points compared to the base NMT. These results indicate that a language model solely trained for next token generation, if powerful enough, can be directly integrated during decoding and contribute to better translation. Furthermore, fine-tuned language models on validation sets in each domain also demonstrate effectiveness in achieving a similar impact.

4.3 Low Resource Machine Translation

All the experiments mentioned above are conducted using a high-resource NMT model. However, it is well known that monolingual data is more advantageous in low-resource MT scenarios. To evaluate the effectiveness of our method in such scenarios, we carried out the main experiment by replacing the De-En NMT model with a low-resource one. This low resource NMT model was trained on a subset of the training dataset from the WMT21 De-En news translation task. First, the training data was cleaned using language detection operation and by

Methods	**IT**	**Koran**	**Law**	**Medical**	**Average**
Base Models					
NMT	36.39	16.76	44.29	39.34	34.19
+ kNN-MT	45.46$_{(0.7)}$	21.68$_{(0.6)}$	60.24$_{(0.9)}$	55.17$_{(0.8)}$	45.64
+ Pseudo-kNN-MT	38.97$_{(0.3)}$	18.14$_{(0.4)}$	51.14$_{(0.4)}$	47.19$_{(0.5)}$	38.56
+ Retrieve-bt-kNN-MT	39.59$_{(0.5)}$	19.26$_{(0.5)}$	52.14$_{(0.6)}$	46.71$_{(0.6)}$	39.43
+ Mono-bt-kNN-MT	40.22$_{(0.7)}$	20.14$_{(0.6)}$	52.40$_{(0.7)}$	46.85$_{(0.7)}$	39.90
Llama2	34.19	11.71	37.52	33.96	29.35
Llama2-chat	29.03	12.97	28.54	33.83	26.09
ALMA	36.20	15.66	36.25	40.05	32.04
kNN-MT with LLM Pseudeo Datastore					
+Llama2-zero-shot	36.32$_{(0.2)}$	18.08$_{(0.3)}$	45.53$_{(0.4)}$	42.17$_{(0.4)}$	35.53
+Llama2-three-shot	38.38$_{(0.3)}$	17.81$_{(0.2)}$	45.61$_{(0.3)}$	42.17$_{(0.4)}$	35.99
+Llama2-chat-zero-shot	38.85$_{(0.4)}$	17.85$_{(0.3)}$	45.49$_{(0.3)}$	42.46$_{(0.3)}$	36.16
+Llama2-chat-three-shot	38.61$_{(0.3)}$	17.96$_{(0.3)}$	45.49$_{(0.2)}$	42.43$_{(0.4)}$	36.12
+ALMA-zero-shot	38.40$_{(0.3)}$	18.11$_{(0.3)}$	45.95$_{(0.3)}$	42.74$_{(0.4)}$	36.30
+ALMA-three-shot	38.64$_{(0.3)}$	17.95$_{(0.3)}$	45.63$_{(0.3)}$	42.61$_{(0.4)}$	36.20
LLM Translator Interpolation					
+Llama2-zero-shot	37.89$_{(0.2)}$	17.39$_{(0.2)}$	44.87$_{(0.2)}$	39.83$_{(0.1)}$	35.00
+Llama2-three-shot	37.88$_{(0.1)}$	18.06$_{(0.4)}$	44.89$_{(0.2)}$	39.84$_{(0.1)}$	35.17
+Llama2-chat-zero-shot	38.10$_{(0.1)}$	17.16$_{(0.2)}$	44.89$_{(0.1)}$	40.04$_{(0.3)}$	35.05
+Llama2-chat-three-shot	38.39$_{(0.3)}$	17.51$_{(0.2)}$	45.26$_{(0.2)}$	40.39$_{(0.3)}$	35.39
+ALMA-zero-shot	38.93$_{(0.5)}$	17.70$_{(0.4)}$	45.50$_{(0.3)}$	41.78$_{(0.5)}$	35.98
+ALMA-three-shot	38.62$_{(0.5)}$	18.20$_{(0.6)}$	45.25$_{(0.5)}$	41.83$_{(0.5)}$	35.98
Language Model Continuation Generator Fusion					
+Llama2-7B	37.15$_{(0.2)}$	18.38$_{(0.7)}$	45.10$_{(0.3)}$	40.41$_{(0.5)}$	35.26
+LM	34.20$_{(0.1)}$	17.00$_{(0.1)}$	43.37$_{(0.1)}$	38.89$_{(0.1)}$	33.36
+fine-tuned-LM	35.79$_{(0.1)}$	18.35$_{(0.3)}$	47.03$_{(0.2)}$	42.69$_{(0.2)}$	35.96

Table 3: SacreBLEU scores of WMT19 Llama-dictionary De-En model on the test sets of multi-domain data. The numbers in the parentheses at the bottom-right indicate same meaning as in Table 2.

removing too long sentences. Then, we uniformly sampled 500K bilingual data from the cleaned data as the training dataset. The news-test 2019 De-En dataset was used as validation set. To translate the target data into the source language, an En-De NMT model was trained using the same training dataset. The experimental results are given in Table 4, where BT denotes the standard back-translation method proposed in (Sennrich et al., 2016).

Experimental results indicate that Pseudo-kNN-MT outperforms base NMT significantly, with an average increase of 5.36 BLEU points. Surprisingly, its performance is comparable to the strong BT method, with only 1.11 BLEU points lower on average, and even superior to BT in Koran domain. This is mainly because Pseudo-kNN-MT relies more on the retrieval similarity, while BT depends more on the data scale. Despite the Koran dataset containing only 17K sentence pairs, it offers relatively similar sentences, resulting in limited performance improvement for BT. In contrast, the Law dataset, with 467K bilingual domain sentence pairs, greatly enhances BT's performance, while Pseudo-kNN-MT shows less improvement compared to BT. Another interesting finding is that Pseudo-kNN-MT performs better than Retrieve-bt-kNN-MT, as the latter depends on a low-resource reverse NMT model for obtaining pseudo bilingual data. For additional low-resource experiments, please see Appendix A.1.

Methods	IT	Koran	Law	Medical	Average
NMT	28.69	10.68	28.42	30.17	24.49
BT	31.72	12.73	41.32	38.08	30.96
Pseudo-kNN-MT	$30.61_{(0.3)}$	$13.97_{(0.5)}$	$36.75_{(0.4)}$	$38.06_{(0.4)}$	29.85
Retrieve-bt-kNN-MT	$31.91_{(0.4)}$	$12.76_{(0.4)}$	$36.09_{(0.6)}$	$37.00_{(0.6)}$	29.44
Mono-bt-kNN-MT	$32.35_{(0.5)}$	$13.28_{(0.6)}$	$36.65_{(0.7)}$	$37.41_{(0.6)}$	29.92

Table 4: SacreBLEU scores of low resource MT experiment on the multi-domain test sets. The numbers in the parentheses at the bottom-right indicate same meaning as in Table 2.

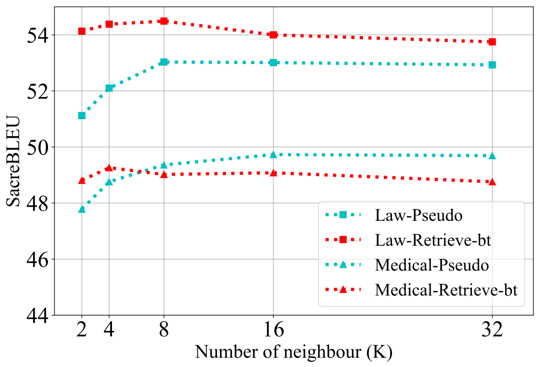

Figure 3: Impact of nearest neighbor numbers on the translation.

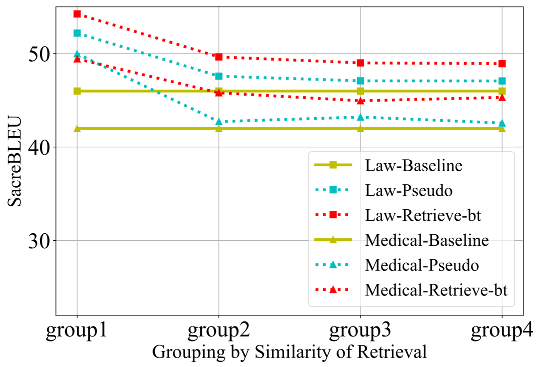

Figure 4: Impact of retrieval similarity on the translation results.

4.4 Influence of Nearest Neighbors Numbers for Per Query

The performance of kNN-MT is sensitive to the value of k, representing the number of nearest neighbors retrieved. To analyze the impact of k on our approach, we conducted experiments on the Medical and Law test sets using different k values. In this experiment, the cross-lingual retrieval remains at 32, while we vary the number of neighbors retrieved from the kNN datastore. The results from the experiment depicted in 3 indicate that both approaches show an initial improvement as k increases, followed by a decline. This trend aligns with the observations in kNN-MT (Khandelwal et al., 2021), suggesting that increasing the number of neighbors appropriately benefits translation, but an excessive number introduces noise and degrades translation quality.

4.5 Influence of Cross-retrieval Similarity on Translation

Except for the experiments in section 4.3, we conducted additional low-resource experiments in Appendix A.1. The findings revealed that Pseudo-kNN-MT did not succeed in enhancing translation quality in the universal domain. These experiments highlighted the importance of similarity between the retrieval and source sentence. To investigate the impact of retrieval similarity on translation, we perform the experiments below on Medical and Law test sets. Specifically, We sort the retrieved 32 target sentences based on their similarity, then divide them into four groups accordingly. The similarity between two vectors is measured using the $L2$ distance from the FAISS library, where closer distances indicate higher similarities. Each group consists of eight sentences, which were used as retrievals for that group. We set the value of k for kNN search to 4, while keeping other experimental settings consistent with the main experiment. The results are presented in Figure 4. The average distances of retrieval from Group 1 to Group 4 are as follows: for Medical (0.5764, 0.6834, 0.7232, 0.7491) and for Law (0.5798, 0.6648, 0.6964, 0.7167). This progression indicated a sequential decrease in similarity from Group 1 to Group 4, suggesting that higher similarity in target language retrieval led to more significant improvements in translation performance.

5 Related Works

As a mature and widely known method, kNN-MT(Khandelwal et al., 2021) has various variants. Zheng et al. (2021) propose adaptive kNN-MT, which can dynamically select k to avoid noisy neighbors. Deguchi et al. (2023) introduce subset kNN-MT, which speeds up inference by retrieving from a small subset based on source similarity. We also leverage subset retrieval while relying cross language similarity. Wang et al. (2022) introduces cluster-based kNN-MT, which adopts a compact network to prune feature datastore extremely. Martins et al. (2022) introduces chuck-based kNN-MT, which changes retrieve granularity from single tokens to chunks. Dai et al. (2023) introduces a fast kNN-MT method, which combines subset kNN-MT and distance-aware λ together. Liu et al. (2023) introduced kNN-TL, which explores the combination of transfer learning method and kNN-MT in low-resource scenarios. Zhu et al. (2023b) introduces INK, a training framework that refines the representation space of an NMT model according to the extracted kNN knowledge to avoid the high inference cost of the kNN-MT. Additionally, Wang et al. (2023) explores how non-parametric kNN-MT method can improve machine translation models at the fine-tuning stage. Cao et al. (2023) introduces a method to address the gap between the upstream NMT model and downstream domains datastores, making kNN-MT more suitable for downstream tasks by reconstructing datastore.

6 Conclusion

In this paper, we propose pseudo-kNN-MT to exploit target language data to NMT. Experimental result show its strong domain adaptation capability on both high-resource and low-resource MT scenarios, validating the effectiveness of incorporating target monolingual data in the kNN-MT. Within this method, we employ a cross-lingual retrieval model to retrieve similar sentences from the target language dataset and pair them with the input sentences to construct pseudo-bilingual data, which is then used to build a key-value datastore. We also explore methods of incorporating LLMs to NMT from various perspectives and find that LLMs are beneficial for robust NMT systems.

7 Limitations

Our proposed pseudo-kNN-MT method is heavily influenced by the similarity of the retrieved target language sentence. If the retrieved target sentence matches the source sentence semantically, it can enhance the translation; otherwise, it may not, and could even degrade translation performance. Therefore, its applicability is limited. Specifically, when translating in a particular domain, the target language data used should also belong to that domain to ensure similarity in retrieval. If this target language data can cover the domain extensively, then our method can perform even better.

Acknowledgements

This work was supported in part by the National Science Foundation of China (No.62276056), the Natural Science Foundation of Liaoning Province of China (2022-KF-16-01), the Fundamental Research Funds for the Central Universities (Nos. N2216016 and N2316002), the Yunnan Fundamental Research Projects (No. 202401BC070021), and the Program of Introducing Talents of Discipline to Universities, Plan 111 (No.B16009).

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zhiwei Cao, Baosong Yang, Huan Lin, Suhang Wu, Xiangpeng Wei, Dayiheng Liu, Jun Xie, Min Zhang, and Jinsong Su. 2023. Bridging the domain gaps in context representations for k-nearest neighbor neural machine translation. *arXiv preprint arXiv:2305.16599*.

Yuhan Dai, Zhirui Zhang, Qiuzhi Liu, Qu Cui, Weihua Li, Yichao Du, and Tong Xu. 2023. Simple and scalable nearest neighbor machine translation. In *The Eleventh International Conference on Learning Representations*.

Hiroyuki Deguchi, Taro Watanabe, Yusuke Matsui, Masao Utiyama, Hideki Tanaka, and Eiichiro Sumita. 2023. Subset retrieval nearest neighbor machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–189, Toronto, Canada. Association for Computational Linguistics.

Duane K. Dougal and Deryle Lonsdale. 2020. Improving NMT quality using terminology injection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4820–4827, Marseille, France. European Language Resources Association.

Akiko Eriguchi, Spencer Rarrick, and Hitokazu Matsushita. 2019. Combining translation memory with neural machine translation. In *Proceedings of the 6th Workshop on Asian Translation*, pages 123–130, Hong Kong, China. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022a. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 878–891. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022b. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 878–891. Association for Computational Linguistics.

Pengzhi Gao, Liwen Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. 2023. Learning multilingual sentence representations with cross-lingual consistency regularization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: EMNLP 2023 - Industry Track, Singapore, December 6-10, 2023*, pages 243–262. Association for Computational Linguistics.

Eva Hasler, Adrià de Gispert, Gonzalo Iglesias, and Bill Byrne. 2018. Neural machine translation decoding with terminology constraints. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 506–512, New Orleans, Louisiana. Association for Computational Linguistics.

Qiuxiang He, Guoping Huang, Qu Cui, Li Li, and Lemao Liu. 2021. Fast and accurate neural machine translation with translation memory. 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 3170–3180.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Trans. Big Data*, 7(3):535–547.

Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2021. Nearest neighbor machine translation. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Ziheng Li, Shaohan Huang, Zihan Zhang, Zhi-Hong Deng, Qiang Lou, Haizhen Huang, Jian Jiao, Furu Wei, Weiwei Deng, and Qi Zhang. 2023. Dual-alignment pre-training for cross-lingual sentence embedding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 3466–3478. Association for Computational Linguistics.

Shudong Liu, Xuebo Liu, Derek F. Wong, Zhaocong Li, Wenxiang Jiao, Lidia S. Chao, and Min Zhang. 2023. kNN-TL: k-nearest-neighbor transfer learning for low-resource neural machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1878–1891, Toronto, Canada. Association for Computational Linguistics.

Pedro Henrique Martins, Zita Marinho, and André F. T. Martins. 2022. Chunk-based nearest neighbor machine translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4228–4245, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yongyu Mu, Abudurexiti Reheman, Zhiquan Cao, Yuchun Fan, Bei Li, Yinqiao Li, Tong Xiao, Chunliang Zhang, and Jingbo Zhu. 2023. Augmenting large language model translators via translation memories. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10287–10299.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation, WMT 2019, Florence, Italy, August 1-2, 2019 - Volume 2: Shared Task Papers, Day 1*, pages 314–319. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 186–191. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Abudurexiti Reheman, Tao Zhou, Yingfeng Luo, Di Yang, Tong Xiao, and Jingbo Zhu. 2023. Prompting neural machine translation with translation memories. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):13519–13527.

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.

Danielle Saunders. 2022. Domain adaptation and multi-domain adaptation for neural machine translation: A survey. *Journal of Artificial Intelligence Research*, 75:351–424.

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.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Dexin Wang, Kai Fan, Boxing Chen, and Deyi Xiong. 2022. Efficient cluster-based k-nearest-neighbor machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2175–2187, Dublin, Ireland. Association for Computational Linguistics.

Jiayi Wang, Ke Wang, Yuqi Zhang, Yu Zhao, and Pontus Stenetorp. 2023. Non-parametric, nearest-neighbor-assisted fine-tuning for neural machine translation. *arXiv preprint arXiv:2305.13648*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models.

Jitao Xu, Josep-Maria Crego, and Jean Senellart. 2020. Boosting neural machine translation with similar translations. In *Annual Meeting of the Association for Computational Linguistics*, pages 1570–1579. Association for Computational Linguistics.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting large language model for machine translation: A case study. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 41092–41110. PMLR.

Xin Zheng, Zhirui Zhang, Junliang Guo, Shujian Huang, Boxing Chen, Weihua Luo, and Jiajun Chen. 2021. Adaptive nearest neighbor 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 2: Short Papers)*, pages 368–374, Online. Association for Computational Linguistics.

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

Wenhao Zhu, Jingjing Xu, Shujian Huang, Lingpeng Kong, and Jiajun Chen. 2023b. Ink: Injecting knn knowledge in nearest neighbor machine translation. *arXiv preprint arXiv:2306.06381*.

A Other Experiments

A.1 Low Resource Settings

To verify the performance of our method in low-resource scenarios, we conducted experiments on the datasets from Is-En and Cs-En news translation tasks of WMT 21. For data selection, we combined all datasets except for the bilingual obtained from machine translation, and then performed uniform sampling on the cleaned bilingual data to obtain a bilingual dataset. The monolingual target language data utilized the news2021 data from `news-crawl/en`. After cleaning, we also used uniform sampling to obtain final monolingual data. In the back-translation method, following Sennrich et al. (2016), we initially trained a reverse NMT model from bilingual data to translate target language monolingual data back into the source language, resulting in 1 million synthetic-bilingual data. Subsequently, we mixed this data with the original bilingual data and trained an NMT model

Split	Is-En	Cs-En	En
Train	500K	500K	1M
Valid	2004	2082	-
Test	1000	1000	-

Table 5: Statistics of datasets for low resource translation scenario.

Split	Is-En	Cs-En
NMT	21.46	21.46
Back-translation	25.69	23.68
Pseudo-kNN-MT	21.20	21.40
Mono-bt-kNN-MT	22.26	22.54
Retrieve-bt-kNN-MT	21.79	21.97

Table 6: SacreBLEU scores of WMT21 low resource NMT models on WMT 21 test sets.

on this combined dataset. Data statistics for this section are presented in Table 5, and experimental results of these low resource NMT models on WMT21 test sets are provided in Table 6. The results indicated that Pseudo-kNN-MT failed to enhance translation quality, while Retrieve-bt-kNN-MT can improve it slightly. As is discussed in Section 4.5, the similarity between the retrieval and source language is crucial. In order to retrieve more similar sentences, we expand target data scale to 5M, 10M, 20M, and calculated their $L2$ distances with the source sentences. The results in Table 7 demonstrate a gradual improvement in our method, although with small margins. Furthermore, we tested our method with the same Cs-En MT model on the JRC-Aquis Cs-En dataset used in (Mu et al., 2023), which exhibits higher retrieval similarity, leading to significant improvements. The target data scale of JRC-Aquis is 681K and average $L2$ distance of the retrievals is 0.5768. These results suggest that low-resource translation can also achieve significant enhancements with highly similar retrievals.

B COMET Scores

Here we present the COMET evaluation results for the main experiment and the LLM integration experiments. Specifically, Table 9 and Table 10 correspond to Table 2 and Table 3 in the main part of the paper, respectively.

Target Data Sacale	1M	5M	10M	20M
NMT	21.46	-	-	-
Pseudo-kNN-MT	21.40	21.31	21.51	21.61
Retrieve-bt-kNN-MT	21.97	22.12	22.08	22.16
Average Distance	0.8785	0.8100	0.7813	0.7542

Table 7: SacreBLEU scores of WMT21 Cs-En low resource model on WMT21 test set with expanding target data scale.

models	SacreBLEU
NMT	32.79
Pseudo-kNN-MT	39.25
Retrieve-bt-kNN-MT	42.29

Table 8: SacreBLEU scores of WMT21 Cs-En low resource model on JRC-Aquis test set.

Methods	**IT**	**Koran**	**Law**	**Medical**	**Average**
NMT	.8246	.7257	.8538	.8316	.8089
kNN-MT	.8489	.7352	.8717	.8486	.8261
Pseudo-kNN-MT	.8251	.7224	.8468	.8243	.8046
Retrieve-bt-kNN-MT	.8264	.7314	.8611	.8384	.8143
Mono-bt-kNN-MT	.8296	.7300	.8596	.8393	.8146

Table 9: COMET scores of Facebook's WMT19 De-En model on the multi-domain test sets.

Methods	**IT**	**Koran**	**Law**	**Medical**	**Average**
Base Models					
NMT	.8236	.7244	.8547	.8335	.8090
+ kNN-MT	.8616	.7342	.8748	.8541	.8311
+ Pseudo-kNN-MT	.8338	.7208	.8492	.8252	.8072
+ Retrieve-bt-kNN-MT	.8346	.7239	.8630	.8409	.8156
+ Mono-bt-kNN-MT	.8354	.7304	.8653	.8428	.8184
Llama2	.7456	.6827	.7678	.8035	.7499
Llama2-chat	.7548	.7773	.7954	.7894	.7792
ALMA	.7700	.7643	.7985	.8049	.7844
kNN-MT with LLM Pseudeo Datastore					
+Llama2-zero-shot	.7739	.7893	.8177	.8080	.7972
+Llama2-three-shot	.7772	.7890	.8177	.8083	.7980
+Llama2-chat-zero-shot	.7750	.7891	.8173	.8074	.7972
+Llama2-chat-three-shot	.7762	.7889	.8177	.8082	.7977
+ALMA-zero-shot	.7769	.7889	.8176	.8075	.7977
+ALMA-three-shot	.7768	.7888	.8174	.8075	.7976
LLM Translator Interpolation					
+Llama2-zero-shot	.7819	.7932	.8189	.8085	.8006
+Llama2-three-shot	.7823	.7933	.8189	.8087	.8008
+Llama2-chat-zero-shot	.7869	.7978	.8204	.8146	.8049
+Llama2-chat-three-shot	.7889	.7975	.8208	.8142	.8054
+ALMA-zero-shot	.7877	.7992	.8206	.8124	.8050
+ALMA-three-shot	.7866	.7977	.8204	.8144	8048
Language Model Continuation Generator Fusion					
+Llama2-7B	.7782	.7913	.8177	.8096	.7992
+LM	.7748	.7856	.8128	.8058	.7947
+fine-tuned-LM	.7772	.7891	.8165	.8093	.7980

Table 10: COMET scores of WMT19 Llama-dictionary De-En model on the test sets of multi-domain data.