Soft Prompt Decoding for Multilingual Dense Retrieval

Zhiqi Huang
zhiqihuang@cs.umass.edu
University of Massachusetts Amherst, USA

Hansi Zeng
hzeng@cs.umass.edu
University of Massachusetts Amherst, USA

Hamed Zamani
zamani@cs.umass.edu
University of Massachusetts Amherst, USA

James Allan
allan@cs.umass.edu
University of Massachusetts Amherst, USA

ABSTRACT

In this work, we explore a Multilingual Information Retrieval (MLIR) task, where the collection includes documents in multiple languages. We demonstrate that applying state-of-the-art approaches developed for cross-lingual information retrieval to MLIR tasks leads to sub-optimal performance. This is due to the heterogeneous and imbalanced nature of multilingual collections – some languages are better represented in the collection and some benefit from large-scale training data. To address this issue, we present KD-SPD, a novel soft prompt decoding approach for MLIR that implicitly "translates" the representation of documents in different languages into the same embedding space. To address the challenges of data scarcity and imbalance, we introduce a knowledge distillation strategy. The teacher model is trained on rich English retrieval data, and by leveraging bi-text data, our distillation framework transfers its retrieval knowledge to the multilingual document encoder. Therefore, our approach does not require any multilingual retrieval training data. Extensive experiments on three MLIR datasets with a total of 15 languages demonstrate that KD-SPD significantly outperforms competitive baselines in all cases. We conduct extensive analyses to show that our method has less language bias and better zero-shot transfer ability towards new languages.

CCS CONCEPTS

• **Information systems** → **Multilingual and cross-lingual retrieval**; *Retrieval models and ranking*.

KEYWORDS

Multilingual retrieval; Prompt-based learning; Knowledge distillation; Dense retrieval

ACM Reference Format:
Zhiqi Huang, Hansi Zeng, Hamed Zamani, and James Allan. 2023. Soft Prompt Decoding for Multilingual Dense Retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3539618.3591769

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3591769

1 INTRODUCTION

In Cross-Lingual Information Retrieval (CLIR), a user submits a query in one language, and the system responds by retrieving documents in another language. Hence, in addition to a ranking model, CLIR systems require an extra component of language translation to align the vocabulary in the query language with that of the document language. This translation gap can be bridged by employing dictionaries [77], statistical translation tables [5], machine translations [60], or, more recently, multilingual pre-trained large language models [25].

Motivated by many real-world applications, such as web search, where the retrieval collection includes documents from multiple languages [8, 30], this work focuses on a multilingual retrieval setting, where the query is in one language and the collection is a mixture of languages. We refer to this task as MLIR, and it has been previously explored by [38, 50, 52]. Even though CLIR and MLIR are tightly coupled, effective MLIR models must overcome additional major challenges. For instance, instead of one pair of languages between query and document, the translation component in the MLIR model needs translation knowledge for multiple language pairs. Xu et al. [74] found that in such scenarios, the distribution of relevant documents to a given query often differs in different languages – which highlights the challenges in designing effective MLIR models that also perform fairly across languages.

The advent of multilingual versions of pre-trained Transformer-based language models, such as mBERT [13] and XLM-R [9], provides the possibility of jointly learning representations for many languages. Because tokens in different languages are projected into the same semantic space, the pre-training phase imparts the model with multilingual translation knowledge. Like monolingual retrieval, fine-tuning these models with multilingual retrieval data allows the model to learn the knowledge of query-document matching and perform retrieval tasks under a multilingual setting.

However, we find that this modeling pipeline, which delivers state-of-the-art results in many CLIR tasks [17, 75, 78], suffers from two major shortcomings when applied to MLIR settings. First, to learn query-document matching knowledge on multiple language pairs effectively, this models requires access to multilingual *retrieval* training data that covers the languages present in the target collection. However, many languages suffer from the scarcity of multilingual training data with reliable relevance judgment [31]. Therefore, it is challenging to achieve broad language coverage in training data. For languages not covered in the training data, the model has to retrieve documents in a so-called zero-shot manner, creating a performance gap between the observed and non-observed languages [45]. Second, due to the unbalanced pre-training data in

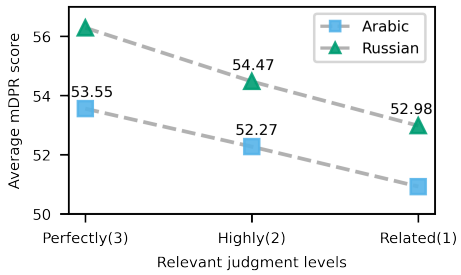


Figure 1: Average score given to parallel documents in Arabic and Russian by mDPR [82]. Queries and relevant judgments are from the TREC 2020 Deep Learning Track. Passages are translated by mMARCO [6]

different languages, the performance of multilingual pre-trained models varies by language in many downstream tasks [69, 71]. MLIR models built on such pre-trained models can inherit language bias, leading to inconsistent ranking results. To demonstrate this case, we pair the test queries from TREC 2020 Deep Learning Track [11] with their relevant passages translated into Arabic and Russian by mMARCO [6]. Then for each language, we score query-document pairs using the multilingual dense passage retriever (mDPR) [82]. Figure 1 illustrates the difference in ranking the same set of relevant documents in these two languages. We observe that mDPR scores Russian documents higher than their Arabic version. We argue that such language bias in MLIR would lead to sub-optimal ranking results, e.g., highly relevant documents in Arabic have lower scores than slightly relevant documents in Russian.

To address these issues, we present KD-SPD,[1] a multilingual dense retrieval model based on knowledge distillation (KD) and soft prompt decoder (SPD) for the MLIR task. KD-SPD does not require any multilingual relevance labels for training, thus automatically solving the data scarcity issue in low-resource languages. Our approach solely requires monolingual retrieval training data in English, which we obtain from MS MARCO [47], and a large collection of parallel and comparable documents. Note that such data is abundant and easily collected through automatic bi-text mining algorithms [15]. We use CCAligned [15] in our experiments.

We first train a monolingual dense retrieval model M, such as ANCE [73], for the English language. Since this model has the relevance matching ability, we freeze its document encoder and then minimize the distance between the representations learned by M for any English document and the representations learned by KD-SPD for its parallel or comparable version in other languages. In fact, M acts as a monolingual teacher model for the multilingual student model KD-SPD. Therefore, our approach implicitly "translates" the representation of documents in different languages into the same language embedding space. We hypothesize that although different languages possess unique properties such as distinct grammar or vocabulary, they also have common traits for expressing similar meanings. To capture these unique and shared features, KD-SPD uses decomposable soft prompt, which is derived as the product of a shared matrix and a low-rank language-specific matrix for each language. Our proposed encoder-decoder architecture transforms documents into contextualized token embeddings and decodes the outputs with language-specific prompts to obtain a final representation. Through joint training across multiple languages, we observe that the learned prompts are capable of reducing language bias and possess the transferable capacity to generalize to unseen languages.

We performed extensive experiments on three MLIR datasets with a total of 15 languages from diverse linguistic families, including both high- and low-resource languages. We also conduct experiments on different relevant distributions with respect to language. In terms of mean average precision (MAP), our proposed method significantly outperforms several strong baseline methods in all multilingual settings, including a 20.2% improvement over mDPR and a 9.6% improvement over a multilingual knowledge distillation method from Sentence-BERT (SBERT) [53]. Further analysis demonstrates that KD-SPD has less language bias and better zero-shot transfer ability toward new languages.

2 RELATED WORK

2.1 Neural Matching Models for MLIR

With respect to language settings, MLIR and CLIR are closely related. CLIR mostly focuses on retrieval between two particular languages, while MLIR considers multiple language pairs between query and document. In general, information retrieval involving a multilingual setting has two sub-tasks: translation and query-document matching. One method involves translating the query into the language of the document set, then using a monolingual retrieval model to evaluate relevance. The translation sub-task can be performed using Statistical Machine Translation (SMT) [5] or Neural Machine Translation (NMT) [56]. The two-step process of translation followed by retrieval is widely used; however, with the advent of bilingual word representation [2, 67] and multilingual pre-trained language models [10, 14], it is possible to bypass the translation step and match the query and document in different languages within a shared representation space.

Multilingual pre-trained language models usually prepend a special token to the input sequence to support downstream applications. Because the special token embedding is contextualized based on other tokens in the sequence, once finetuned, they are effective across various tasks, including retrieval tasks [39, 40, 78, 79]. Named cross-encoder, the model takes the concatenation of the query and document as input. An embedding of the "[CLS]" token is fed into a feed-forward layer to produce a score for the input pair [48]. With multilingual knowledge from pre-training, these language models help bridge the vocabulary between query and document languages. Like monolingual retrieval, multilingual retrieval models based on cross-encoder are computationally expensive and usually rely on a lexical-based sparse retrieval as the first step to finding relevant information. Dense retrieval based on a bi-encoder architecture is proposed to overcome the sparse retrieval bottleneck [28, 32, 43]. With the separation of the query document encoders, dense retrieval has already shown success on monolingual retrieval tasks [18, 29]. By replacing the underlying language model with its multilingual version, dense retrieval is extended to a multilingual setting [82].

[1] KD refers to the model training framework, and SPD refers to the model architecture.

However, the translation gap prevents the multilingual retrieval models from achieving the same level of performance as models in the monolingual (i.e., English-to-English) setting [25]. Supporting the model with abundant multilingual retrieval data is one way to reduce the effect of the translation gap. Sasaki et al. [61] constructed large-scale, weakly supervised CLIR collections based on the linked foreign language articles from Wikipedia pages. Bonifacio et al. [6] built MLIR training data using neural machine translation models. Besides retrieval data, approaches like utilizing external knowledge in language-specific modules are also suggested to close the language gap. Bonab et al. [5] showed that when fine-tuned with retrieval data, dictionary-oriented word embedding could improve the performance of a CLIR model. Huang et al. [25] proposed a mixed attention transformer architecture to learn relevance judgments and word-level translation knowledge jointly. Yang et al. [76] designed a language adapter component to efficiently transfer models based on monolingual data to a cross-lingual setting.

These approaches mostly focus on improving CLIR performance where the query and the target documents are from two particular languages. In this work, we focus on MLIR, a more general setting where the document collection comprises a diverse mix of languages, which is gaining increasing attention recently [35]. While being able to bridge the translation gap between multiple languages, the model for MLIR task also needs to be language-agnostic when ranking documents in different languages. Our approach implicitly "translates" documents in different languages into an embedding space tuned for English retrieval.

2.2 Multi-task & Prompt-based Learning

The goal of multi-task learning is to leverage the shared underlying structure of different tasks to improve the performance of each task [55]. A common approach is to transfer the knowledge from a model fine-tuned on multiple source tasks to the target task [1, 51, 66]. For example, Aghajanyan et al. [1] introduce a pre-finetuning stage that involves multi-task learning steps on diverse NLP tasks. They show that training stability can be improved by applying task-heterogenous batches with task-rebalancing loss scaling. Recent works show that the zero-shot and few-shot performance of pre-trained large language models can be boosted by prompted multi-task learning [41, 58, 68]. For instance, Sanh et al. [58] develop a system that maps any NLP task into a human-readable prompt form where each supervised dataset contains multiple prompts with diverse wording. The experiments imply that the multi-task training on these prompted datasets can improve the zero-shot performance of the pre-trained models. Other works [83] focus on zero-shot classification (ZAC), introducing a meta-tuning training paradigm to optimize the zero-shot classification objective via fine-tuning. They consolidate various classification tasks into a single QA format, compiling a dataset of classification tasks with human-authored prompts for meta-tuning.

Soft Prompt tuning has shown great potential to adapt large language models to downstream tasks [4, 33, 65, 70]. Vu et al. [65] further study the generalizability and transferability of the soft prompts. They first learn a prompt on one or more source tasks and use it as the initialized prompt for a target task. The simple target prompt initialization method can match or outperform full fine-tuning across all model sizes. Asai et al. [4] extend the work by training an attention module to interpolate the source prompts and newly initialized target prompt for each downstream task. During the multi-task training, only the target prompt and attention weights are updated, while the soft prompts and original language model's parameters are frozen. A recent approach [70] learns a transferable shared prompt by applying matrix decomposition and knowledge distillation from multiple source task-specific prompts and using the low-ranking matrix updating for target task adaption.

KD-SPD builds upon the idea of prompt-oriented, parameter-efficient multi-task learning. It treats retrieval in each language as a distinct task while jointly modeling them to capture shared underlying structures. The primary insight is that languages, despite unique properties, share common features and concepts. We utilize decomposable prompts to model these aspects. Unlike conventional parameter-efficient approaches, experiments show that updating prompts jointly with model parameters enhances retrieval performance.

2.3 Knowledge Distillation

Proposed by Hinton et al. [23], knowledge distillation is a method to train a model, called the student, using valuable information provided by the output of another model, called the teacher. This way, the teacher model's knowledge can be transferred into the student model. The idea of knowledge distillation is wildly used in the field of computer vision [36, 72, 80], natural language processing [53, 57] and information retrieval [24, 37, 42, 59, 81].

In the field of information retrieval, it is common for the teacher model to be a complex reranker model with higher capacity but lower efficiency compared to the efficient dual-encoder based student model. Santhanam et al. [59] apply the KL divergence loss to align query-document scores between teacher and student models. Another approach is balanced topic-aware query sampling [24], which shows further improvement on top of the original knowledge distillation loss. To address the performance gap, Zeng et al. [81] propose a curriculum learning based knowledge distillation framework that trains a student model with increasing difficulty. In addition to monolingual retrieval, multilingual distillation frameworks have also been proposed. Li et al. [35] explore using query-document scores as the distillation signals. The cross-lingual token alignment task has also been studied as an optimal transport problem, with Huang et al. [26] proposing a distillation framework to build a CLIR model via bitext data.

Our model training framework is also an extension of knowledge distillation. A typical framework for knowledge distillation relies on a teacher model to solely provide target distributions [21, 44]. Our approach has different sources of knowledge: the major knowledge is from the teacher model, and we also consider the cross-lingual knowledge shared by the prompt matrix. Moreover, from the language perspective, rather than focusing on one CLIR task, our model simultaneously learns retrieval knowledge for multiple CLIR tasks.

3 METHODOLOGY

Our goal is to incorporate the knowledge of query-document matching from a well-trained monolingual retrieval model into a multilingual transformer-based retrieval architecture, such that it is

capable of generating contextual representations under the MLIR setting and thus performing query-document matching in different languages. In this section, we first define the MLIR task and outline our approach. Then we present the key component of our model: a soft prompt-based encoder-decoder architecture. Finally, we introduce the model training via a knowledge distillation framework and build the MLIR model with components from both the teacher and student models. Due to space limitations, we focus on the MLIR case of searching a multilingual collection with an English query as an example to describe our method. It is worth noting that English may also be included in the multiple collection.

3.1 Overview

Given a query q in language X and a target collection $\mathcal{D}_Y$ which contains documents in language set $\boldsymbol{Y} = \{Y_1, Y_2, \dots Y_K\}$, suppose d_{ki}—the i^{th} document in language Y_k—has the ground truth relevance label $Rel(q, d_{ki})$, then the aim is to design an MLIR model f that retrieves a list of documents from $\mathcal{D}_Y$ such that

$$f(q, d_{ki}) \geq f(q, d_{lj}), \quad \forall\, Rel(q, d_{ki}) \geq Rel(q, d_{lj}) \tag{1}$$

where $f(\cdot, \cdot)$ indicates the ranking score calculated by the model. To build model f, we first assume there exists an oracle model g for the retrieval task in language X. Thus, given q and monolingual collection $\mathcal{D}_X$, g satisfies:

$$g(q, d_{xi}) \geq g(q, d_{xj}), \quad \forall\, Rel(q, d_{xi}) \geq Rel(q, d_{xj}) \tag{2}$$

We can achieve (1) with model f' if for any d_* in $\boldsymbol{Y}$ and its translation d_x in X, the model matches the oracle:

$$f'(q, d_*) = g(q, d_x)$$

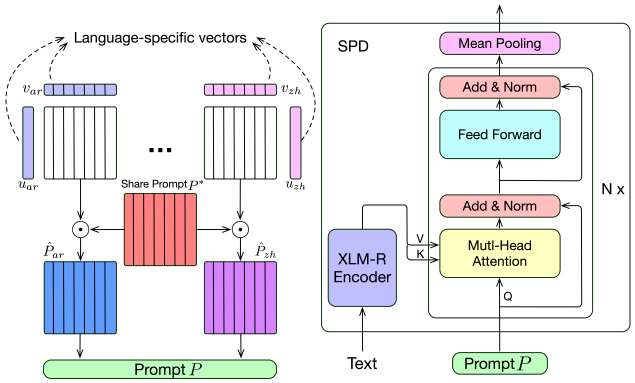
Suppose both f' and g follow the architecture of dense retrieval, the ranking score calculation is the dot-product of the query and document embeddings, thus:

$$f'_E(q) f'_D(d_*)^\top = g_E(q) g_D(d_x)^\top$$

where f'_E and g_E are query encoders; f'_D and g_D are document encoders for f' and g respectively. We then reuse g_E as the query encoder of f'. With $f'_E = g_E$, we have:

$$g_E(q)\big(f'_D(d_*) - g_D(d_x)\big)^\top = 0 \tag{3}$$

It is safe to assume $g_E(q)$ is a nonzero vector. Therefore the goal of finding f' is equivalent to reducing the embedding distance between parallel documents. In our method, we retrain g_D as the teacher model by removing its parameters from the computational graph and train f'_D as the student model. Note that in practice, the oracle model g does not exist. We can use an off-the-shelf English-to-English (monolingual) dense retrieval model as a substitute for g. Because g_D is fixed, the essence of knowledge distillation training is to push multilingual document representations generated by f'_D toward their corresponding English document representations generated by g_D. Moreover, Equation (3) suggests that the training of f'_D does not rely on either query q or ground truth relevant judgment. A group of parallel or comparable sentences from English to any other language involved in the collection is adequate to train f'_D. Parallel or comparable sentences between two languages are often referred as bitext data. Unlike multilingual retrieval data, which often require relevance labels, bitext data are easier to acquire, especially for low-resource languages [22, 63].



(a) **Soft prompt decomposition.** (b) **Multilingual document encoder.**

Figure 2: SPD model architecture.

3.2 Soft Prompt Decoder

We focus on the design of the document encoder of the student model, f'_D, which handles multilingual documents. In general, the function of f'_D is similar to a neural machine translation model. The difference is that f'_D translates the input text into an embedding in the target language rather than natural language text. Thus, we build f'_D based on the encoder-decoder architecture. For the encoder component of f'_D, we exploit multilingual pre-trained language models (i.e., mBERT or XLM-R). The token representation generated by the encoder is then fed to the decoder component. However, unlike the decoder with an autoregressive generation process, we propose a soft prompt-based decoder (SPD) architecture.

Soft Prompt Matrix. We consider f'_D as a multitask model where translating (mapping) each language in the multilingual collection into the target language space is viewed as a single task. Using the language name as the task identifier, a prompt $\mathbf{P}_k \in \mathbb{R}^{l \times d}$ for language Y_k with the same dimension as the token embedding d and vector length as l is used as input to the decoder. Thus, the prompt matrix serves as the language-based decoding initialization vector. Inspired by the prompt decomposition from multitask prompt tuning [70], we decompose $\mathbf{P}_k$ into two parts, as shown in Figure 2a: language-specific low-rank vectors $\mathbf{u}_k \in \mathbb{R}^l$ and $\mathbf{v}_k \in \mathbb{R}^d$ for language Y_k; And a shared prompt $\mathbf{P}^* \in \mathbb{R}^{l \times d}$ across all languages. The language-specific prompt can be parameterized as $\mathbf{W}_k = \mathbf{u}_k \cdot \mathbf{v}_k^\top$, which has the same dimension as the shared prompt $\mathbf{P}^*$. The final prompt $\hat{\mathbf{P}}$ for language Y_k is then formulated as follows.

$$\hat{\mathbf{P}}_k = \mathbf{P}^* \odot \mathbf{W}_k = \mathbf{P}^* \odot (\mathbf{u}_k \cdot \mathbf{v}_k^\top) \tag{4}$$

where $\odot$ denotes the Hadamard product between two matrices. The shared prompt enables efficient knowledge sharing across all source languages and commonalities across translation tasks. Meanwhile, the language-specific vectors still allow each translation task to maintain its own parameters to encode language-specific knowledge. Additionally, prior studies on multitask prompt learning also showed that soft prompt learned from multitask data can be efficiently transferred to a new task [62, 64]. In section 5.5, we show that with a shared prompt, the SPD has a better zero-shot transfer ability toward new languages.

Cross-attention Decoder. The decoder network follows a cross-attention-based multi-layer transformer architecture. Each layer has two sub-layers. The first is a multi-head query-key-value (QKV) cross-attention module, and the second is a position-wise fully connected feed-forward network. We employ residual connection and layer norm around each of the sub-layers.

Let $\mathbf{T}_{d_k} \in \mathbb{R}^{|d_k| \times d}$ denote the token representations generated by the encoder component for document d_k in language Y_k, where $|d_k|$ is the number of tokens in d_k. The first decoder layer applies the cross-attention module between $\mathbf{T}_{d_k}$ and prompt matrix $\hat{\mathbf{P}}_k$. On the mth head, the attention mechanism is defined as follows:

$$\text{Attention}_m = \text{Softmax}\Big(\frac{W_m^q \hat{\mathbf{P}}_k \cdot W_m^k \mathbf{T}_{d_k}}{\sqrt{d/M}}\Big) W_m^v \mathbf{T}_{d_k}$$

where M is the number of heads and W_m^q, W_m^k and W_m^v are matrices with dimension $d/M \times d$. Thus, the prompt matrix has different attention weights over encoder token representations in each subspace projection (head). The output of multi-head QKV cross-attention module is the concatenation of M heads with linear projection:

$$\text{CrossAttention}(\hat{\mathbf{P}}_k, \mathbf{T}_{d_k}) = W^o[\text{Attention}_1, \ldots, \text{Attention}_M]$$

We further define the output of the attention-based sub-layer with the residual connection and layer norm:

$$\mathbf{h}_{d_k} = \text{LN}\big(\hat{\mathbf{P}}_k + \text{CrossAttention}(\hat{\mathbf{P}}_k, \mathbf{T}_{d_k})\big)$$

where $\text{LN}(\cdot)$ denotes the layer norm operation. Because $\hat{\mathbf{P}}_k$ is the query element in the cross-attention module, we use the prompt matrix to query the information from the encoder output and store it in a hidden representation $\mathbf{h}_{d_k}$ which has the same dimension as the prompt matrix. Next, we apply the second sub-layer and generate the output of the first decoder layer for d_k, $\mathbf{H}^1_{d_k} \in \mathbb{R}^{l \times d}$:

$$\mathbf{H}^1_{d_k} = \text{DecoderLayer}_1(\hat{\mathbf{P}}_k, \mathbf{T}_{d_k}) = \text{LN}(\mathbf{h}_{d_k} + \text{FFN}(\mathbf{h}_{d_k}))$$

where $\text{FFN}(\cdot)$ denotes the fully connected feed-forward network with a rectified activation function. Then we use the hidden representation from the previous layer (i.e. $\mathbf{H}^1_{d_k}$) to query the encoder output again in the next layer, that is:

$$\mathbf{H}^{n+1}_{d_k} = \text{DecoderLayer}_{n+1}(\mathbf{H}^n_{d_k}, \mathbf{T}_{d_k})$$

until reaching the maximum layer N designed for the decoder. Finally, we average $\mathbf{H}^N_{d_k}$ over the prompt vector dimension as the document embedding in the target language space. A complete architecture of f'_D is depicted in Figure 2b.

$$f'_D(d_k) = \text{MeanPool}(\mathbf{H}^N_{d_k})$$

3.3 Multilingual Dense Retrieval

Knowledge Distillation Training. Assume that d_{En} is the English version of d_k. From the property of g, we know that the document embedding of d_{En} generated by g_D contains rich knowledge for query-document matching in English. Equation (3) suggests that if we could let f'_D "behave" like g_D, namely, if for any d_k, the output of $f'_D(d_k)$ is close to the output of $g_D(d_{\text{En}})$, then the document embedding generated by f'_D can have a similar retrieval performance as g in the English domain. Therefore, we require the English document encoder g_D as the teacher model and our multilingual document encoder f'_D as the student. During training, we define the distillation loss as the mean square error (MSE) between two embeddings and sample B examples from each language to form a batch.

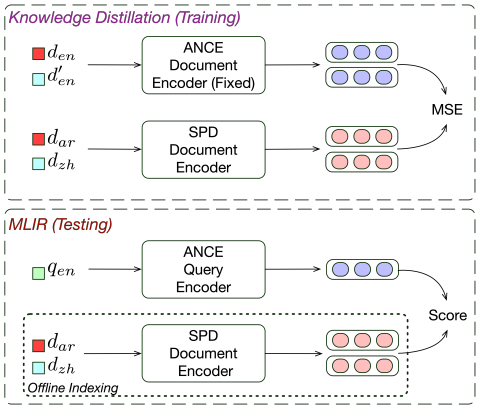


Figure 3: Model building pipeline for MLIR.

Table 1: Summary of MLIR evaluation datasets. Avg. #d$^+$/q denotes the average number of relevant documents per query

Dataset Statistics	CLEF	mTREC	LAReQA
Query size	133	54	1,190
Collection size	241K	35.2M	13,014
Languages in collection	3	4	11
Avg. #d$^+$/q	13.5	66.8	1.0

$$loss := \frac{1}{KB} \sum_{k=1}^{K} \sum_{i=1}^{B} |f'_D(s_{ki}) - g_D(e_{ki})|^2 \tag{5}$$

where s_{ki} is a sentence in language Y_k and e_{ki} is its parallel (translation) in English.

Query-document matching. In this section, we discuss an MLIR task of searching multilingual collections using an English query to introduce the KD-SPD framework. The query encoder in the final retrieval model can be directly copied from the teacher model in the English domain. Specifically, at test time, the matching score of q and d_* is calculated based on the dot-product between g_E and f'_D:

$$f'(q, d_*) = g_E(q) f'_D(d_*)^\top$$

An overview of our MLIR model building pipeline is shown in Figure 3. In fact, we can also apply KD-SPD to other language settings in MLIR task. For example, suppose the task requires searching an English collection using queries in multiple languages. In this case, KD-SPD can be built as a query encoder, and the retrieval model can reuse the teacher's document encoder. More generally, if the MLIR task involves a query language set $\boldsymbol{X}$ and a collection language set $\boldsymbol{Y}$, we can consider English as a bridge to build KD-SPD via two knowledge distillations: $\boldsymbol{X}$ to English for query encoder and $\boldsymbol{Y}$ to English for document encoder.

4 EXPERIMENTAL SETUP

4.1 Dataset

Evaluation data. We focus on retrieval from multilingual collections with English queries. To comprehensively evaluate model performance on this MLIR task, we create three test sets with various combinations of collection size, relevance distribution, and language settings. Note that some multilingual evaluation datasets have separate query sets per language, which does not thoroughly evaluate the MLIR performance. Thus, we focus on a setting where the same set of test queries is evaluated on all languages in the collection. Table 1 shows the statistics of our evaluation datasets.

- **CLEF.** The data is from the Cross-Language Evaluation Forum (CLEF) 2000-2003 campaign for bilingual ad-hoc retrieval tracks [7]. We include documents in French, German, and Italian to build a multilingual collection. The English query is a concatenation of the title and description fields of the topic files. Among the CLEF C001 – C200 topics, we only consider a topic with human-annotated relevant documents in all three languages as a valid query, leading to 133 queries in total.
- **mTREC.** The query and relevance judgments are from the test split of the passage ranking task from the TREC 2020 Deep Learning Track [11]. There are three relevance judgment levels marked by 3,2,1. We build the multilingual collection from mMARCO [6], which is a machine-translated version of the MS MARCO passage collection [47]. We select translated passages in four languages: Arabic, Chinese, Russian, and Indonesian, to form a large-scale multilingual collection. Because translation leads to parallel relevant documents, this evaluation set allows us to study the effect of relevant distribution over languages. We first equally distributed relevant documents on each relevance level among four languages. In section 5.2, we explore biased relevant distribution.
- **LAReQA.** LAReQA [54] is a benchmark for language-agnostic answer retrieval from a multilingual candidate pool. It is built based on two multilingual question-answering datasets: XQuAD [3] and MLQA [34]. The query is formed using the question, and the collection is formed by breaking contextual paragraphs into sentences. Each query (question) appears in 11 different languages[2] and has 11 parallel relevant sentences (answers). To match our MLIR setting, we evaluate English queries on a collection of sentences in 11 languages (including English).

Bitext training data. To support the multilingual knowledge distillation, we use the parallel sentences from the CCAligned dataset [15]. To train one KD-SPD model covering all three evaluation datasets (15 languages[3]), we sample 4 million parallel sentences per language except English. For English, to be consistent with other languages, we sample another 4 million sentences and pair each sentence with itself. Thus, our training data comprises 60 million sentence pairs in 15 languages. We append a language code to each sentence for SPD to identify the language of the input document.

Retrieval fine-tuning data. For a competitive baseline, we further fine-tune mDPR [82] baseline (see section 4.3.2) using cross-lingual triples from mMARCO [6]. We sample 6 million cross-lingual triples per language to form a multilingual training set for languages in CLEF and mTREC. Because languages in LAReQA are not fully covered by mMARCO, we use mDPR on LAReQA without fine-tuning. Note that our KD-SPD model does not use this data.

[2]Languages in LAReQA (ISO code): ar, de, el, en, es, hi, ru, th, tr, vi, zh

[3]List of training languages (ISO code): ar, de, el, en, es, fr, hi, id, it, pt, ru, th, tr, vi, zh

4.2 Implementation Details

We initialize the encoder component of the SPD model using the pre-trained XLM-R model [9] (base-sized) and the decoder component (including prompt matrices) using the Xavier initialization [20]. We train the SPD as a student model using bitext data. To learn the retrieval knowledge in the English domain, we employ the document encoder of ANCE [73] as the teacher. When testing, the query encoder of the final model is also a reuse of the query encoder of ANCE (except in section 5.4, where we investigate the impact of different teachers). For hyper-parameters, we set the length of the prompt token vector $l = 30$ and the number of SPD decoding layers $N = 6$. We truncate the input sequence length at 180 tokens and sample 4 examples per language to build a mini-batch. The model is trained with a learning rate of 2×10^{-5} for one epoch of all bitext data. For evaluation on the CLEF dataset, where the document length is usually longer than 180 tokens, we split long documents into overlapping passages of fixed length with a stride of 90 tokens and compute the score for each query passage pair. Finally, we select a document's maximum passage score as its ranking score [46].

Evaluation. We examine the top 100 ranked documents and report comprehensive metrics, including mean average precision (MAP), normalized discounted cumulative gain (nDCG@10), precision (P@10), mean reciprocal rank (MRR), and recall (R@100). We determine statistical significance using the two-tailed paired t-test with p-value less than 0.05 (i.e., 95% confidence level).

4.3 Compared Methods

From a modeling perspective, we compare KD-SPD with both non-neural and neural approaches. From the system design perspective, we compare KD-SPD with end-to-end solutions and pipeline solutions via rank list merging.

4.3.1 Non-neural baselines. For non-neural baselines, we generally consider a three-step pipeline to address MLIR. First, we break the collection into subsets by language and translate the query to each subset language. Since the translated queries and subset collection are in the same language, we then use a lexical-based sparse retrieval technique (e.g., BM25) to obtain a ranked list for each language. Finally, we merge language-specific ranked lists into a final ranked list. We investigate different strategies of translation and ranked list merging that we elaborate below.

SMT: We translate the query based on a statistical machine translation (SMT) method. Specifically, we first build a translation table from the parallel corpus for each language pair using GIZA++ [49]. Then we select the top 10 translations from the translation table for each query term and apply Galago's[4] *#combine* operator to form a translated query. Finally, we run BM25 with default parameters to retrieve documents in the same language as the query translation.

NMT: We translate the query into collection languages using Google Translation[5] (a neural-based commercial machine translation system). Then we run BM25 with default parameters to retrieve documents from each subset collection using the translated query.

[4]https://www.lemurproject.org/galago.php/

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

Table 2: A comparison of model performance. The highest value is marked with bold text. For KD-SPD, statistically significant improvements are marked by † (over mDPR) and ‡ (over KD-Encoder).

Retrieval Method	CLEF					mTREC					LAReQA				
	MAP	nDCG@10	P@10	MRR	R@100	MAP	nDCG@10	P@10	MRR	R@100	MAP	nDCG@10	P@10	MRR	R@100
SMT+Round Robin	0.1348	0.2540	0.2429	0.4017	0.3732	0.0242	0.0557	0.0630	0.1592	0.0778	0.2678	0.3858	0.2332	0.6610	0.4415
SMT+Score	0.1459	0.2737	0.2421	0.4679	0.3508	0.0187	0.0468	0.0648	0.1060	0.0661	0.2269	0.3407	0.2126	0.6527	0.3506
NMT+Round Robin	0.1783	0.3732	0.3474	0.5793	0.4118	0.0653	0.1735	0.1870	0.3965	**0.1872**	0.5717	0.6178	0.556	0.7139	0.8345
NMT+Score	0.1950	0.3806	0.3474	0.6140	0.4206	0.0522	0.1570	0.1685	0.3970	0.1691	0.5063	0.5671	0.5178	0.7091	0.8002
mDPR+Round Robin	0.1823	0.3412	0.3165	0.5448	0.4330	0.0490	0.1358	0.1537	0.2913	0.1324	0.4935	0.5223	0.5163	0.6493	0.8394
mDPR+Score	0.1941	0.3433	0.3203	0.5364	0.4401	0.0492	0.1459	0.1574	0.3154	0.1300	0.4852	0.5142	0.4462	0.6452	0.8418
mDPR	0.2025	0.3466	0.3195	0.5367	0.4504	0.0549	0.1675	0.1870	0.3954	0.1291	0.4452	0.5031	0.4462	0.7653	0.7970
KD-Encoder	0.1973	0.3883	0.3594	0.5641	0.4315	0.0639	0.2208	0.2293	0.4556	0.1629	0.5931	0.6058	0.5730	0.7673	0.8805
KD-SPD	**0.2200**†‡	**0.4160**†‡	**0.3714**†‡	**0.6356**†‡	**0.4689**‡	**0.0748**†‡	**0.2414**†‡	**0.2556**†‡	**0.5067**†‡	0.1705†	**0.6265**†‡	**0.6316**†‡	**0.6049**†‡	**0.7904**†‡	**0.8912**†

+Round Robin: We merge multiple rank lists in the round-robin style, that is, iteratively extracting the top-ranked document from K languages in random order to be the next K of the final rank list.
+Score: We merge multiple rank lists by ranking scores generated by the retrieval component. Scores within each rank list are first min-max normalized to $[0, 1]$.
The non-neural baselines are the combination of translation with merging strategies: SMT+Round Robin, SMT+Score, NMT+Round Robin, and NMT+Score.

4.3.2 Neural baselines. As a dense retriever, we compare KD-SPD with other dense retrieval methods in the following:
mDPR: Models that follow the dense passage retriever (DPR) paradigm has proven to be effective for many retrieval tasks. Zhang et al. [82] extended DPR to non-English languages by changing the underlying pre-trained language model from BERT to multilingual BERT (mBERT). We adopt the checkpoint of mDPR trained on MS MARCO dataset [47]. For CLEF and mTREC, which have fewer languages in the collections, we further fine-tune mDPR using the mMARCO dataset [6]. We apply mDPR to MLIR in two ways: First, we break the MLIR task into multiple CLIR tasks by language and use mDPR to retrieve documents from subset collections. Then we merge the rank lists from different CLIR tasks, named mDPR+Round Robin and mDPR+Score, respectively. Second, we apply mDPR as an end-to-end solution for MLIR, in which we use it to directly index and search from the multilingual collection.
KD-Encoder: There are methods that can transfer the knowledge from a model built for a monolingual task to a multilingual model, enabling it to address the same task in a multilingual setting. Reimers and Gurevych [53] proposed a knowledge distillation method to create multilingual versions from the same monolingual models. We refer to this idea as the KD-Encoder and apply it to the MLIR task. To compare with our approach, we adopt the same teacher model and train KD-Encoder with the same bitext data.

5 EXPERIMENTAL RESULTS

5.1 Retrieval Performance

Table 2 lists the evaluation results on the three MLIR datasets. Comparing non-neural approaches, given BM25 as the same retrieval component, we can see that methods based on NMT outperform those based on SMT. For document collections with mostly high-resource languages, NMT based method can also achieve higher nDCG, precision, and MRR scores than end-to-end neural approaches (i.e., NMT+Score on CLEF). It highlights that translation quality is an important factor in MLIR.

Usually, for a pipeline approach, the error can accumulate for each step and lead to a sub-optimal result [16, 19]. In MLIR, without evaluating the content with respect to the query, merging rank lists only based on the score or rank within sub-collection will cause errors from multiple languages to accumulate. However, comparing the pipeline with the end-to-end approach of mDPR, we can see that end-to-end mDPR does not show a consistent advantage over the pipeline mDPR. There are two plausible reasons. First, similar to other multilingual models, mDPR based on a multilingual pre-trained language model also inherits the language bias in the pre-training step. Second, the fine-tuning steps of mDPR only focus on ranking documents within the same language space. Even trained with multilingual retrieval data, the candidate documents are still monolingual, and the score comparison is between two particular languages. These two reasons cause the ranking score generated by mDPR to be inconsistent across languages. Moreover, KD-Encoder performs better than mDPR on mTREC and LAReQA, On CLEF, it also scores higher nDCG, precision, and MRR than mDPR. Such results suggest that mapping parallel text from different languages to the same location in the vector space via knowledge distillation can efficiently transfer monolingual retrieval knowledge to multilingual settings. Finally, with the support of soft prompt decoding, KD-SPD achieves the best retrieval performance among all compared methods. In terms of precision-oriented metrics, it consistently and significantly outperforms both mDPR and KD-Encoder.

5.2 Biased Relevant Distribution

In the MLIR task, some queries strongly prefer one language over others and some do not. Thus, different queries tend to have different relevant document distributions among languages. This special feature requires the retrieval system to rank documents independent of their language. In the experiment on mTREC shown in Table 2, relevant documents were distributed equally among four languages for each query. The parallel translations of mTREC allow us to test with different relevant document distributions. In this section, we simulate the language preference in MLIR task: For each query, we first randomly select a language as the primary language and assign 60% of the top relevant documents (sorted by relevance judgment level) to that language. And the other three become the

Table 3: Performance comparison of biased distributed relevant documents in mTREC. Significance tests are marked by † (over mDPR) and ‡ (over KD-Encoder).

Retrieval Method	Biased mTREC				
	MAP	nDCG@10	P@10	MRR	R@100
SMT+Round Robin	0.0134	0.0304	0.0426	0.0759	0.0621
SMT+Score	0.0149	0.0356	0.0426	0.1083	0.0698
NMT+Round Robin	0.0331	0.0939	0.1278	0.2902	0.1500
NMT+Score	0.0438	0.1055	0.1389	0.3430	**0.1751**
mDPR+Round Robin	0.0301	0.0902	0.1074	0.2922	0.1150
mDPR+Score	0.0516	0.1576	0.1778	0.3655	0.1206
mDPR	0.0508	0.1571	0.1759	0.3652	0.1174
KD-Encoder	0.0681	0.2028	0.2078	0.4055	0.1494
KD-SPD	**0.0753**†	**0.2317**†‡	**0.2352**†‡	**0.4579**†‡	0.1684†‡

minor languages for this query, among which we equally distribute the remaining 40% of the relevant documents. Table 3 shows the results on biased distributed relevant documents of mTREC. As expected, the performance of methods based on round-robin merge drop significantly. The reason is that the rank list from minor languages introduces more errors compared to the scenario where languages are uniformly distributed. We can see that KD-SPD is also affected by the change in distribution yet still performs the best among all compared methods.

5.3 Analysis of Knowledge Distillation

To study how SPD behaves after knowledge distillation, we compare the rank distance and score difference of parallel relevant documents in the rank lists generated by different models. In this experiment, again, we take advantage of parallel translations in mTREC and build *duplicate* relevant documents in four languages. Thus, for each query, there are semantically similar relevant documents in different languages. Given a query, we locate all parallel relevant documents in four languages within the top 1,000 candidates from rank lists generated by mDPR, KD-Encoder, and KD-SPD, respectively. Then we compute the maximum rank distance and score difference among the four parallel documents. The equation to compute the score difference is as follows:

$$\mathcal{S} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{|\mathcal{R}_{q_i}|}\sum_{d_{kj}\in\mathcal{R}_{q_i}}\left(\max_{k\in Y} f'(q_i, d_{kj}) - \min_{k\in Y} f'(q_i, d_{kj})\right)$$

where Q is the query set, $\mathcal{R}_{q_i}$ is the set of relevant documents for the query q_i, and Y is the language set. The averaging rank distance can also be obtained in a similar way. Figure 4 shows the results averaged over 54 queries in mTREC. We can see that KD-SPD has the smallest rank distance and score difference over parallel documents. The rank and score of parallel documents reflect the language bias in MLIR models. Thus, KD-SPD is less biased toward languages when ranking documents from a multilingual collection. Moreover, because the query embedding is fixed given the same query, the low mean and standard deviation values indicate that KD-SPD is able to generate similar embeddings for parallel documents in different languages. This matches the model design purpose.

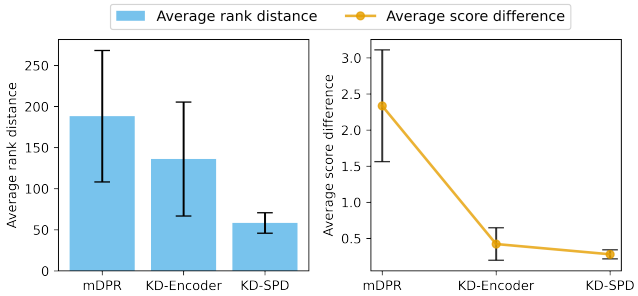


Figure 4: Parallel document analysis for MLIR models.

5.4 Ablation Study

In this section, we conduct experiments on two aspects that could affect the performance of KD-SPD: The number of layers in the decoder and the choice of the teacher model for distillation.

Decoder architecture. Following the idea of weights share in Transformers [12, 27], we replace the multi-layer (6-layer) decoder with a recurrent decoder block. Instead of N distinct layers, a decoder block has the same architecture as one decoder layer and is called recurrently for $N = 12$ steps. The weights of a decoder block are shared between steps. After each step, we add a temporal embedding $\boldsymbol{\tau} \in \mathbb{R}^{l\times d}$ to the hidden states.

$$\mathbf{H}_{d_k}^{n+1} = \boldsymbol{\tau}_n + \text{DecoderBlock}(\mathbf{H}_{d_k}^{n}, \mathbf{T}_{d_k})$$

This approach significantly reduces the size of model parameters. Named universal transformer-based SPD (UTSPD), Table 4 shows its performance, compared to KD-Encoder and KD-SPD. We can see that only with 2.1% more parameters, KD-UTSPD performs better than KD-Encoder. By reducing the parameter size, we show that the performance gain in SPD mainly relies on the prompt design and decoder component based on the cross-attention module. Because reducing parameters limits the model's generalization ability, there is a performance drop from distinct layers to shared weights.

Teacher model The teacher model bounds the retrieval performance of KD-SPD. We hypothesize that a better teacher model in the English domain can lead to a better SPD model for MLIR task. Based on the leaderboard of MS MARCO passage ranking, we replace ANCE [73] with coCondenser [18] for knowledge distillation. To be consistent with coCondenser, we also change the pre-trained multilingual language model used in SPD from XLM-R to mBERT. The evaluation of SPD trained with different teacher models is shown in Table 5. In general, KD-SPD learned from coCondenser performs better than the one learned from ANCE. This suggests that improvements with respect to the retrieval performance in the English domain can be transferred to MLIR task via KD-SPD.

5.5 Zero-shot Transfer

We explore the zero-shot ability of KD-SPD. For documents in languages that are not observed in the training data, we first define the language-specific vectors by averaging all trained language-specific vectors from known languages. Then KD-SPD follows the same steps as other languages to generate a prompt matrix for the new language. Hence, observed languages' knowledge transfers to the new language via the shared prompt matrix.

Table 4: Ablation I: Decoder architecture. The numbers in the bracket are differences in percentage to KD-Encoder.

Model	Parameter Size	CLEF					LAReQA				
		MAP	nDCG@10	P@10	MRR	R@100	MAP	nDCG@10	P@10	MRR	R@100
KD-Encoder	278.6M	0.1973	0.3883	0.3594	0.5641	0.4315	0.5931	0.6058	0.5730	0.7673	0.8805
KD-SPD	320.0M (+14.8)	0.2200 (+11.5)	0.4160 (+7.1)	0.3714 (+3.3)	0.6356 (+12.7)	0.4689 (+8.7)	0.6265 (+5.6)	0.6316 (+4.2)	0.6049 (+5.6)	0.7904 (+3.0)	0.8912 (+1.2)
KD-UTSPD	284.5M (+2.1)	0.2075 (+5.2)	0.4023 (+3.6)	0.3722 (+3.6)	0.5964 (+5.7)	0.4576 (+6.0)	0.6212 (+4.7)	0.6279 (+3.6)	0.5996 (+4.6)	0.7674 (+0.0)	0.8870 (+0.7)

Table 5: Ablation II: Effect of Teacher model. Significance tests with respect to KD-SPD (ANCE) are marked in ▲.

Teacher	CLEF					LAReQA				
	MAP	nDCG@10	P@10	MRR	R@100	MAP	nDCG@10	P@10	MRR	R@100
KD-SPD (ANCE)	0.2200	0.4160	0.3714	0.6356	0.4689	0.6265	0.6316	0.6049	0.7904	0.8912
KD-SPD (coCondenser)	0.2487▲	0.4546▲	0.4008▲	0.6826▲	0.4976▲	0.6501▲	0.6694▲	0.6436▲	0.8012	0.9172

Table 6: Zero-shot CLIR: English-to-Finnish. Significance tests are marked by † (over mDPR) and ‡ (over KD-Encoder).

Retrieval Method	CLEF Finnish				
	MAP	nDCG@10	P@10	MRR	R@100
SMT	0.0739	0.1179	0.0900	0.1390	0.1828
NMT	0.1613	0.2562	0.1560	0.4591	0.4251
mDPR	0.1682	0.2143	0.1300	0.3095	0.5010
KD-Encoder	0.1845	0.2796	0.1920	0.4537	0.5237
KD-SPD	**0.2286**†‡	**0.3321**†‡	**0.2220**†‡	**0.5092**†‡	**0.5958**†‡

Table 7: Zero-shot MLIR. Significance tests are marked by † (over mDPR) and ‡ (over KD-Encoder).

Retrieval Method	CLEF DE-IT-FI				
	MAP	nDCG@10	P@10	MRR	R@100
SMT+Round Robin	0.1099	0.2245	0.208	0.4096	0.2909
SMT+Score	0.1269	0.2242	0.218	0.3726	0.2974
NMT+Round Robin	0.1263	0.2748	0.254	0.5039	0.3384
NMT+Score	0.1447	0.2806	0.258	0.5101	0.344
mDPR+Round Robin	0.1481	0.2734	0.268	0.391	0.3974
mDPR+Score	0.1728	0.3002	0.282	0.4816	0.4083
mDPR	0.1952	0.3377	0.306	0.5175	0.4107
KD-Encoder	0.1963	0.4262	0.382	0.6753	0.4152
KD-SPD	**0.2174**†‡	**0.4494**†‡	**0.4100**†‡	**0.7099**†‡	**0.4545**†‡

In this study, we focus on Finnish as the target language and use a collection of 54,694 Finnish documents from the CLEF dataset. It's worth mentioning that Finnish, a member of the Uralic language family, is distinct from the 15 languages used in training. Among the 133 English queries in the CLEF dataset, 50 have relevant annotations in the Finnish collection, forming a new set of test queries. The results in Table 6 show the performance of KD-SPD in Cross-Language Information Retrieval (CLIR) between English and Finnish, and we observe that KD-SPD significantly outperforms other methods, demonstrating the transferability of knowledge from the prompt matrices to new languages. Next, we expand the evaluation to a more challenging setting, combining Finnish with German and Italian. The resulting collection contains both observed and unobserved languages. Table 7 shows KD-SPD's zero-shot performance in the multilingual information retrieval (MLIR) setting, where it still achieves the best results. This highlights KD-SPD's strong ability to transfer knowledge in a zero-shot scenario.

6 CONCLUSIONS AND FUTURE WORK

In this work, we presented a knowledge distillation (KD) framework based on soft prompt decoding (SPD) to address the multilingual information retrieval (MLIR) task. Using the soft prompt matrix as a task indicator, KD-SPD can implicitly translate documents from multiple languages into the same embedding space as the query language. We proposed prompt decomposition to enable efficient knowledge sharing across all source languages. Our knowledge distillation framework transfers knowledge from a well-trained monolingual retrieval model to KD-SPD, greatly reducing the retrieval data requirements for training MLIR models. Our comprehensive experimental results show that KD-SPD significantly outperforms other baselines on three qualitatively different MLIR evaluation datasets. Further analysis demonstrates that KD-SPD has less language bias and better zero-shot transfer ability toward new languages. For future work, as a general knowledge transfer framework, we are interested in extending KD-SPD to transfer other monolingual task-specific knowledge into the multilingual space. Exploring the applications of KD-SPD to multimodal information retrieval is also an exciting future direction.

ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and This research is based upon work supported in part by the Center for Intelligent Information Retrieval, and in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Contract No. 2019-19051600007 under Univ. of Southern California subcontract no. 124338456. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. 2021. Muppet: Massive Multi-task Representations with Pre-Finetuning. *ArXiv* abs/2101.11038 (2021).

[2] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 451–462.

[3] Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856* (2019).

[4] Akari Asai, Mohammadreza Salehi, Matthew E. Peters, and Hannaneh Hajishirzi. 2022. ATTEMPT: Parameter-Efficient Multi-task Tuning via Attentional Mixtures of Soft Prompts.

[5] Hamed Bonab, Sheikh Muhammad Sarwar, and James Allan. 2020. Training effective neural CLIR by bridging the translation gap. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 9–18.

[6] Luiz Bonifacio, Vitor Jeronymo, Hugo Queiroz Abonizio, Israel Campiotti, Marzieh Fadaee, Roberto Lotufo, and Rodrigo Nogueira. 2021. mmarco: A multilingual version of the ms marco passage ranking dataset. *arXiv preprint arXiv:2108.13897* (2021).

[7] Martin Braschler. 2002. CLEF 2002—Overview of results. In *Workshop of the Cross-Language Evaluation Forum for European Languages*. Springer, 9–27.

[8] SM Chaware and Srikantha Rao. 2009. Information retrieval in multilingual environment. In *2009 Second International Conference on Emerging Trends in Engineering & Technology*. IEEE, 648–652.

[9] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116* (2019).

[10] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

[11] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the TREC 2019 deep learning track. *arXiv preprint arXiv:2003.07820* (2020).

[12] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. 2018. Universal transformers. *arXiv preprint arXiv:1807.03819* (2018).

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

[15] Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. CCAligned: A Massive Collection of Cross-lingual Web-Document Pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*.

[16] Thiago Castro Ferreira, Chris van der Lee, Emiel Van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. *arXiv preprint arXiv:1908.09022* (2019).

[17] Petra Galuščáková, Douglas W Oard, and Suraj Nair. 2021. Cross-language information retrieval. *arXiv preprint arXiv:2111.05988* (2021).

[18] Luyu Gao and Jamie Callan. 2021. Unsupervised corpus aware language model pre-training for dense passage retrieval. *arXiv preprint arXiv:2108.05540* (2021).

[19] Tobias Glasmachers. 2017. Limits of end-to-end learning. In *Asian conference on machine learning*. PMLR, 17–32.

[20] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 249–256.

[21] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision* 129, 6 (2021).

[22] Kevin Heffernan, Onur Çelebi, and Holger Schwenk. 2022. Bitext mining using distilled sentence representations for low-resource languages. *arXiv preprint arXiv:2205.12654* (2022).

[23] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* 2, 7 (2015).

[24] Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. 2020. Improving efficient neural ranking models with cross-architecture knowledge distillation. *arXiv preprint arXiv:2010.02666* (2020).

[25] Zhiqi Huang, Hamed Bonab, Sheikh Muhammad Sarwar, Razieh Rahimi, and James Allan. 2021. Mixed attention transformer for leveraging word-level knowledge to neural cross-lingual information retrieval. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 760–770.

[26] Zhiqi Huang, Puxuan Yu, and James Allan. 2022. Improving Cross-lingual Information Retrieval on Low-Resource Languages via Optimal Transport Distillation. In *The 16th ACM International Conferenceon Web Search and Data Mining (WSDM), 2023*.

[27] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. 2021. Perceiver: General perception with iterative attention. In *International conference on machine learning*. PMLR, 4651–4664.

[28] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906* (2020).

[29] Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 39–48.

[30] Kazuaki Kishida. 2005. Technical issues of cross-language information retrieval: a review. *Information processing & management* 41, 3 (2005), 433–455.

[31] Dawn Lawrie, James Mayfield, Douglas W Oard, and Eugene Yang. 2022. HC4: A new suite of test collections for ad hoc CLIR. In *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I*. Springer, 351–366.

[32] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. *arXiv preprint arXiv:1906.00300* (2019).

[33] Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. *ArXiv* abs/2104.08691 (2021).

[34] Patrick Lewis, Barlas Oğuz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019. MLQA: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475* (2019).

[35] Yulong Li, Martin Franz, Md Arafat Sultan, Bhavani Iyer, Young-Suk Lee, and Avirup Sil. 2022. Learning Cross-Lingual IR from an English Retriever. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Seattle, United States, 4428–4436.

[36] Sihao Lin, Hongwei Xie, Bing Wang, Kaicheng Yu, Xiaojun Chang, Xiaodan Liang, and Gang Wang. 2022. Knowledge Distillation via the Target-Aware Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10915–10924.

[37] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. In-Batch Negatives for Knowledge Distillation with Tightly-Coupled Teachers for Dense Retrieval. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*. Association for Computational Linguistics, Online, 163–173. https://doi.org/10.18653/v1/2021.repl4nlp-1.17

[38] Wen-Cheng Lin and Hsin-Hsi Chen. 2002. Description of NTU Approach to NTCIR3 Multilingual Information Retrieval.. In *NTCIR*. Citeseer.

[39] Robert Litschko, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić. 2018. Unsupervised cross-lingual information retrieval using monolingual data only. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1253–1256.

[40] Robert Litschko, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić. 2018. Unsupervised cross-lingual information retrieval using monolingual data only. In *SIGIR'18*. 1253–1256.

[41] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022. Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning. *ArXiv* abs/2205.05638 (2022).

[42] Wenhao Lu, Jian Jiao, and Ruofei Zhang. 2020. Twinbert: Distilling knowledge to twin-structured compressed bert models for large-scale retrieval. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*.

[43] Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics* 9 (2021), 329–345.

[44] Rongrong Ma, Guansong Pang, Ling Chen, and Anton van den Hengel. 2022. Deep Graph-level Anomaly Detection by Glocal Knowledge Distillation. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*.

[45] Sean MacAvaney, Luca Soldaini, and Nazli Goharian. 2020. Teaching a new dog old tricks: Resurrecting multilingual retrieval using zero-shot learning. In *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42*. Springer, 246–254.

[46] Suraj Nair, Eugene Yang, Dawn Lawrie, Kevin Duh, Paul McNamee, Kenton Murray, James Mayfield, and Douglas W Oard. 2022. Transfer learning approaches for building cross-language dense retrieval models. In *European Conference on Information Retrieval*. Springer, 382–396.

[47] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *choice* 2640 (2016), 660.

[48] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *arXiv preprint arXiv:1901.04085* (2019).

[49] Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics* 29, 1 (2003), 19–51. https://doi.org/10.1162/089120103321337421

[50] Carol Peters, Martin Braschler, and Paul Clough. 2012. *Multilingual information retrieval: From research to practice.* Springer.

[51] Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *ArXiv* abs/1910.10683 (2019).

[52] Razieh Rahimi, Ali Montazeralghaem, and Azadeh Shakery. 2020. An axiomatic approach to corpus-based cross-language information retrieval. *Information Retrieval Journal* 23 (2020), 191–215.

[53] Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv preprint arXiv:2004.09813* (2020).

[54] Uma Roy, Noah Constant, Rami Al-Rfou, Aditya Barua, Aaron Phillips, and Yinfei Yang. 2020. LAReQA: Language-agnostic answer retrieval from a multilingual pool. *arXiv preprint arXiv:2004.05484* (2020).

[55] Sebastian Ruder. 2017. An Overview of Multi-Task Learning in Deep Neural Networks. *ArXiv* abs/1706.05098 (2017).

[56] Shadi Saleh and Pavel Pecina. 2020. Document Translation vs. Query Translation for Cross-Lingual Information Retrieval in the Medical Domain. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 6849–6860.

[57] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2019).

[58] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Rose Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multitask Prompted Training Enables Zero-Shot Task Generalization. *ArXiv* abs/2110.08207 (2021).

[59] Keshav Santhanam, O. Khattab, Jon Saad-Falcon, Christopher Potts, and Matei A. Zaharia. 2021. ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction. In *North American Chapter of the Association for Computational Linguistics*.

[60] Sheikh Muhammad Sarwar, Hamed Bonab, and James Allan. 2019. A Multi-Task Architecture on Relevance-based Neural Query Translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 6339–6344. https://doi.org/10.18653/v1/P19-1639

[61] Shota Sasaki, Shuo Sun, Shigehiko Schamoni, Kevin Duh, and Kentaro Inui. 2018. Cross-lingual learning-to-rank with shared representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. 458–463.

[62] Yusheng Su, Xiaozhi Wang, Yujia Qin, Chi-Min Chan, Yankai Lin, Huadong Wang, Kaiyue Wen, Zhiyuan Liu, Peng Li, Juanzi Li, et al. 2022. On transferability of prompt tuning for natural language processing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 3949–3969.

[63] Weiting Tan and Philipp Koehn. 2022. Bitext Mining for Low-Resource Languages via Contrastive Learning. *arXiv preprint arXiv:2208.11194* (2022).

[64] Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Cer. 2021. Spot: Better frozen model adaptation through soft prompt transfer. *arXiv preprint arXiv:2110.07904* (2021).

[65] Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Matthew Cer. 2021. SPoT: Better Frozen Model Adaptation through Soft Prompt Transfer. In *Annual Meeting of the Association for Computational Linguistics*.

[66] Tu Vu, Tong Wang, Tsendsuren Munkhdalai, Alessandro Sordoni, Adam Trischler, Andrew Mattarella-Micke, Subhransu Maji, and Mohit Iyyer. 2020. Exploring and Predicting Transferability across NLP Tasks. In *Conference on Empirical Methods in Natural Language Processing*.

[67] Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 363–372.

[68] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Maitreya Patel, Kuntal Kumar Pal, M. Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddharth Deepak Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Hannaneh Hajishirzi, Noah A. Smith, and Daniel Khashabi. 2022. Benchmarking Generalization via In-Context Instructions on 1, 600+ Language Tasks. *ArXiv* abs/2204.07705 (2022).

[69] Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending Multilingual BERT to Low-Resource Languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 2649–2656. https://doi.org/10.18653/v1/2020.findings-emnlp.240

[70] Zhen Wang, Rameswar Panda, Leonid Karlinsky, Rogerio Feris, Huan Sun, and Yoon Kim. 2023. Multitask Prompt Tuning Enables Parameter-Efficient Transfer Learning. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=Nk2pDtuhTq

[71] Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? *arXiv preprint arXiv:2005.09093* (2020).

[72] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10687–10698.

[73] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808* (2020).

[74] Jinxi Xu, Ralph Weischedel, and Chanh Nguyen. 2001. Evaluating a probabilistic model for cross-lingual information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. 105–110.

[75] Eugene Yang, Suraj Nair, Ramraj Chandradevan, Rebecca Iglesias-Flores, and Douglas W Oard. 2022. C3: Continued pretraining with contrastive weak supervision for cross language ad-hoc retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2507–2512.

[76] Eugene Yang, Suraj Nair, Dawn Lawrie, James Mayfield, and Douglas W Oard. 2022. Parameter-efficient Zero-shot Transfer for Cross-Language Dense Retrieval with Adapters. *arXiv preprint arXiv:2212.10448* (2022).

[77] Mahsa Yarmohammadi, Xutai Ma, Sorami Hisamoto, Muhammad Rahman, Yiming Wang, Hainan Xu, Daniel Povey, Philipp Koehn, and Kevin Duh. 2019. Robust Document Representations for Cross-Lingual Information Retrieval in Low-Resource Settings. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*. European Association for Machine Translation, Dublin, Ireland, 12–20. https://www.aclweb.org/anthology/W19-6602

[78] Puxuan Yu and James Allan. 2020. A study of neural matching models for cross-lingual IR. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1637–1640.

[79] Puxuan Yu, Hongliang Fei, and Ping Li. 2021. Cross-lingual language model pretraining for retrieval. In *Proceedings of the Web Conference 2021*. 1029–1039.

[80] Mingkuan Yuan and Yuxin Peng. 2019. CKD: Cross-task knowledge distillation for text-to-image synthesis. *IEEE Transactions on Multimedia* 22, 8 (2019), 1955–1968.

[81] Hansi Zeng, Hamed Zamani, and Vishwa Vinay. 2022. Curriculum Learning for Dense Retrieval Distillation. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2022).

[82] Xinyu Zhang, Xueguang Ma, Peng Shi, and Jimmy Lin. 2021. Mr. TyDi: A multilingual benchmark for dense retrieval. *arXiv preprint arXiv:2108.08787* (2021).

[83] Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. 2021. Adapting Language Models for Zero-shot Learning by Meta-tuning on Dataset and Prompt Collections. In *Conference on Empirical Methods in Natural Language Processing*.