Leveraging Domain Adaptation and Data Augmentation to Improve Qur'anic IR in English and Arabic

Vera Pavlova
rttl.ai
Dubai, UAE
v@rttl.ai

Abstract

In this work, we approach the problem of Qur'anic information retrieval (IR) in Arabic and English. Using the latest state-of-the-art methods in neural IR, we research what helps to tackle this task more efficiently. Training retrieval models requires a lot of data, which is difficult to obtain for training in-domain. Therefore, we commence with training on a large amount of general domain data and then continue training on in-domain data. To handle the lack of in-domain data, we employed a data augmentation technique, which considerably improved results in MRR@10 and NDCG@5 metrics, setting the state-of-the-art in Qur'anic IR for both English and Arabic. The absence of an Islamic corpus and domain-specific model for IR task in English motivated us to address this lack of resources and take preliminary steps of the Islamic corpus compilation and domain-specific language model (LM) pre-training, which helped to improve the performance of the retrieval models that use the domain-specific LM as the shared backbone. We examined several language models (LMs) in Arabic to select one that efficiently deals with the Qur'anic IR task. Besides transferring successful experiments from English to Arabic, we conducted additional experiments with retrieval task in Arabic to amortize the scarcity of general domain datasets used to train the retrieval models. Handling Qur'anic IR task combining English and Arabic allowed us to enhance the comparison and share valuable insights across models and languages.

1 Introduction

Recent advances in Natural Language Processing (NLP) have helped to improve search relevance and retrieval quality. Nevertheless, deep-learning techniques, specifically transformer-based approaches (Vaswani et al., 2017), are hardly employed in Quran'ic NLP (Bashir et al., 2023). In this work,
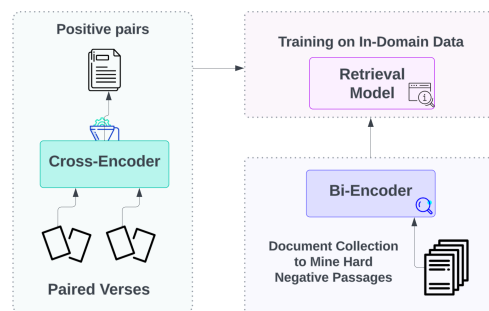

Figure 1: Data augmentation technique that leverages correlation of Qur'anic verses for training retrieval models in-domain.

we will utilize the latest state-of-the-art neural retrieval models to compare what works best for solving the IR task in the Islamic domain. Moreover, we proposed a data-augmentation technique to generate data for in-domain training appropriate for the IR task involving the Holy Qur'an (see Figure 1).

We experimented with Arabic and English languages. Arabic, more precisely one of its variants, Classical Arabic (CA), is the language of the Holy Qur'an and is an integral component in tackling retrieval task using sacred scripture (Bashir et al., 2023). English is another popular language used for search in various domains, including the Islamic domain. Addressing the problem using Arabic and English allows for comparing the solutions and sharing insights across languages. English is a high-resource language with a great choice of corpora and pre-trained LMs for diverse domains. At the same time, depending on the domain, Arabic can be considered a low- or medium-resource language (Xue et al., 2021; Abboud et al., 2022). However, Arabic is in more favorable conditions than English in the Islamic domain; in the case of the Arabic language, there are Islamic corpora like OpenITI (Romanov and Seydi, 2019) and domain-specific LMs (Malhas and Elsayed, 2022; Inoue et al., 2021).

From this perspective, addressing Qur'anic IR in English is more challenging as it requires a number of additional preparations, like preparing an Islamic corpus and pre-training domain-specific LM. This state of affairs with the English language in the Islamic domain necessitates addressing it alongside the Arabic language. Simultaneously, another advantage of handling the problem in English is the abundance of datasets to train for a general domain. Training on general domain data can be a required step to prepare a retrieval model that needs a substantial amount of data for training, where in-domain data is scarce. Experimenting with Qur'anic IR in English will allow us to learn what works best and apply these approaches to Arabic, where general domain data is insufficient.

Our main contributions are:

- We introduce an Islamic corpus and a new language model for the Islamic domain in English.

- We conduct comprehensive experiments with different retrieval models to see what works best for efficient retrieval from the Holy Qur'an in Arabic and English.

- We propose a data-augmentation technique that helped to improve the retrieval models' performance for both languages and set a new state-of-the-art in Qur'anic IR.

The rest of the work is organized as follows: we start with addressing the problem of Qur'anic IR in English. We prepare the Islamic corpus and domain-specific LM (Section 2). Section 3 applies to both languages, English and Arabic, including metrics choice, datasets for training and testing, experimental details, and training procedure of the retrieval models. Section 4 is dedicated to Qur'anic IR in Arabic. Apart from applying successful experiments that worked well with Qur'anic IR in English, we executed more methods of preparing retrieval models for Arabic language, including teacher-student distillation and employing machine translation. Model comparison and Final analysis is performed in Section 5. The prior work done in the field is highlighted in Section 6.
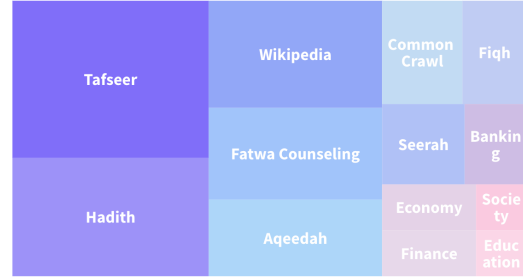

Figure 2: Types of Islamic text that constitute Islamic Corpus.

2 Domain-Specific Language Model as a Backbone of In-Domain IR

2.1 Islamic Corpus in English

Preparing an Islamic Corpus in English is challenging due to the insufficient amount of Islamic Text that is either translated from Arabic or other languages to English or initially written in English. We collect text available online of the following types (see Figure 2):

Islamic literature. These are Islamic books written by Islamic scholars about Tafseer (Qur'an exegesis), Hadith, Seerah, Fiqh (Islamic jurisprudence), and Aqeedah (Islamic creed) (approx. 28M words).

Islamic journals. Journals that are available online and focus on discussing modern issues of Islamic banking, Finance, Economy, and Islamic Education (approx. 5.5 M words).

Fatwa counseling. Fatwas that are available online from Fatwa centers (approx. 4.8M words)

Wikipedia. Articles related to Islam from the Wikipedia Islam portal (approx. 5.6M words).

Common Crawl. We search for keywords and collect files from Common Crawl on Islamic topics. We perform additional filtering and preprocessing of these articles (approx. 2.5 M words).

The total amount of words in the corpus is around 47M words.

2.2 Adaptation of General Domain Language Model for Islamic Domain

Pre-training starting from the existing checkpoint of the model pre-trained for the general domain helps reduce pre-training time (Gururangan et al., 2020; Bommasani et al., 2022; Guo and Yu, 2022). To account for the small size of the pre-training corpus and perform domain adaptation effectively, we continue pre-training the BERT model on the

Islamic corpus. To address the issue of the absence of domain-specific vocabulary during continued pre-training, we trained the WordPiece tokenizer (Song et al., 2021) on the Islamic corpus. We find the intersection between Islamic vocabulary and bert-base-uncased [1], and for the tokens inside this intersection, we assign the weights from bert-base-uncased. For the tokens outside of the intersection (Islamic tokens), we perform contextualized weight distillation following (Pavlova and Makhlouf, 2023) [2].

- In the first step, we find tokens of interest and extract corresponding sentences from the Islamic Corpus. We sample from one to twenty sentences (Bommasani et al., 2020).

- In the second step, we tokenize these sentences using a bert-base-uncased tokenizer. In that case, Islamic tokens are broken into subtokens because they are absent from bert-base-uncased vocabulary. We average the contextualized weights of the corresponding subtokens that the BERT model produces ($t_{distilled}$) and then compute aggregated representation across sentences ($t_{aggregated}$) for a corresponding token of interest from Islamic vocabulary:

$$t_{distilled} = f(t_1, ..., t_k) \atop f \in \{mean\} \tag{1}$$

Where *k* is the number of the subtokens that make up the token of interest.

$$t_{aggregated} = g(t_{distilled}, ..., t_m) \atop g \in \{mean\} \tag{2}$$

And *m* is the number of sentences involved in aggregated representation.

In order to avoid overinflating the vocabulary with new tokens, which would require longer pre-training and be prohibitive in the case of a small corpus, we analyze the frequency of each token in our corpus. Tokens with a count below threshold are filtered out, resulting in 3992 new domain-specific tokens. Moreover, we remove [unused] tokens from the bert-base-uncased vocabulary and add Islamic tokens, resulting in 33511 total tokens in the BPIT model's final vocabulary (BPIT is the abbreviation for BERT Pre-trained on Islamic Text).

2.3 Pre-training Set-up

In order to accommodate the limited size of the pre-training corpus, we schedule two-stage pre-training akin to phases of Curriculum learning (Bengio et al., 2009; Soviany et al., 2022). In the first stage, we start with an easier task of predicting masked tokens/subtokens, with a masking rate of 0.15 and using the "80-10-10" corruption rule (Devlin et al., 2019; Wettig et al., 2023). In the second stage, we increase the complexity of the prediction task by switching to predicting the whole words with the same masking rate and using the corruption rule. It is harder for a language model to predict whole words than to predict tokens or subtokens that might make up the word and give the LM more clues and make the prediction task less challenging (Cui et al., 2021; Dai et al., 2022; Gu et al., 2021). This pre-training approach introduces the LM to a broader scope of language experience and helps to gain more diversified knowledge of textual input (Mitchell, 1997), which is crucial in the case of a small corpus that we use. Pre-training hyperparameters can be found in Appendix A.

3 Preparing Neural IR Model to Retrieve from the Holy Qur'an

3.1 Dataset for Testing Retrieval Models

To test our models, we converted the QRCD (Qur'anic Reading Comprehension Dataset) (Malhas and Elsayed, 2022) to the IR dataset. We use both train and development split as test data. We do not include no-answer questions (Malhas and Elsayed, 2020), which results in 169 queries in total for testing. Queries are accompanied by the corresponding verses from the Holy Qur'an. Each Qur'anic verse is treated as the basic retrieval unit because it presents a more challenging task (see Section 5) and has higher utilization factors. The original dataset is in Arabic and was constructed and annotated by experts in Islamic studies. For our purposes of testing IR systems, we translate queries to English and verify the validity and accuracy of the translation with Islamic scholars. We use the Saheeh International[3] translation of the Holy Qur'an to express specific Qur'anic terms used in query formulation. To retrieve answers, we use the same Sahih International translation as a retrieval collection.

---

[1] https://huggingface.co/bert-base-uncased
[2] https://github.com/rttl-ai/BIOptimus

[3] https://tanzil.net/trans/

3.2 Metrics

Due to the complexity of the language of the Holy Qur'an and the fact that some meanings can be expressed indirectly, the retrieval task using the Holy Qur'an is quite difficult. Therefore, using several metrics to estimate the retrieval model's effectiveness from a different perspective makes sense. We use the MRR@10 (Mean Reciprocal Rate), the official evaluation metric of the MS MARCO dataset (Bajaj et al., 2018) that we extensively use to fine-tune our retrieval models. Furthermore, we add NDCG@5 (Normalized Discounted Cumulative Gain) and Recall@100, used in the BEIR benchmark (Thakur et al., 2021b). This combination of metrics lets us estimate our models with a decision support metric such as Recall, binary rank-aware metrics such as MRR, and metric with a graded relevance judgment such as NDCG (Wang et al., 2013). For evaluation, we use the BEIR framework[4] that utilizes the Python interface of the TREC evaluation tool (Gysel and de Rijke, 2018).

3.3 Baselines

BM25 is a commonly used baseline to compare retrieval systems. It is a sparse lexical retrieval method based on token-matching and uses TF-IDF weights. Though the lexical approaches suffer from the lexical gap (Berger et al., 2000) due to the constraints of retrieving the documents containing exact keywords presented in a query, BM25 remains a strong baseline (Kamalloo et al., 2023). We also include a dense neural retrieval model, trained using a sentence-transformers framework (Reimers and Gurevych, 2019) referred to as SBERT- GD (general domain), a late-interaction model ColBERT (Khattab and Zaharia, 2020) (ColBERT-GD), and Cross-Encoder-GD. All models were trained on the MS MARCO dataset from the bert-base-uncased checkpoint. This approach allows us to evaluate their performance in a zero-shot setting for the Islamic domain and compare them with the retrieval models trained using the domain-specific BPIT model. More details on how SBERT-GD, ColBERT-GD, and Cross-encoder-GD were trained are presented in Section 3.4; hyperparameters details are listed in Appendix A.

3.4 Training a Domain-specific Model on General Domain Data

To prepare the domain-specific model for the IR task, we prepare and compare three approaches.

SBERT-BPIT. We use the sentence-transformers framework, which employs a Siamese network (Bromley et al., 1993) that enables semantic similarity search. We train our BPIT model using the architecture above on the MS MARCO dataset that contains 533k training examples (more details on MS MARCO dataset are in Section 4.4), utilizing Multiple Negative Ranking Loss (MNRL) (Henderson et al., 2017; Ma et al., 2021; van den Oord et al., 2019). MNRL is a cross-entropy loss that treats relevant pairs $\{x^{(i)}, y^{(i)}\}_{i=1}^{M}$ (where M is batch size) as positive labels and other in-batch examples as negative, and formally defined as:

$$J_{\mathrm{MNRL}}(\theta) =$$
$$\frac{1}{M}\sum_{i=1}^{M}\log\frac{\exp\sigma(f_\theta(x^{(i)}), f_\theta(y^{(i)}))}{\sum_{j=1}^{M}\exp\sigma(f_\theta(x^{(i)}), f_\theta(y^{(j)}))}$$

where σ is a similarity function, in our case it is a cosine similarity function; f_θ is the sentence encoder. We use multiple hard negatives; these are negative passages similar to the positive passage but not relevant to the query and mined using cross-encoder scores [5].

Cross-encoder-BPIT. In the case of a cross-encoder, a pair of sentences are simultaneously fed into a transformer-like model, and attention is applied across all tokens to produce a similarity score (Humeau et al., 2020). This approach does not allow end-to-end information retrieval and endure extreme computational overhead. However, in many IR tasks, it performs superior to other methods and can be used for mining hard negatives, data augmentation (Section 3.5), and reranking. The model is trained with triples provided by MS MARCO starting from the BPIT checkpoint under a classification task, using Cross Entropy Loss.

ColBERT-BPIT. ColBERT computes embeddings independently for queries and documents and, at the same time, can also register more fine-grained interactions between tokens. Using the same mined hard negatives constructed for the MS MARCO dataset used to pre-train SBERT-BPIT, ColBERT-BPIT is trained starting from the BPIT

---

checkpoint by optimizing the cross-entropy loss applied to the score of the query and the positive passage against in-batch negatives (Santhanam et al., 2022).

All models with the prefix **BPIT** are counterparts of **GD** models; for a fair comparison, they are trained using the same dataset, objective function, and hyperparameters (see Appendix A) with the only difference that **BPIT** models initialized from the BPIT checkpoint and **GD** models initialized with the bert-base-uncased checkpoint.

3.5 In-domain Training of the Domain-specific Model

The performance of dense retrieval systems worsens when encountering a domain shift (Thakur et al., 2021b); therefore, there is a great benefit in training neural IR models on in-domain data. The lack of domain-specific data is often solved by augmenting training data: generating synthetic data (dos Santos Tanaka and Aranha, 2019), paraphrasing using synonyms (Wei and Zou, 2019), sampling and recombining new training pairs (Thakur et al., 2021a), round-trip translation (Yu et al., 2018; Xie et al., 2020) or involving denoising autoencoders (Wang et al., 2021). These techniques involve data distortion, which is suboptimal when dealing with religious and heritage datasets. We propose a data generation technique for in-domain training advantageous for the retrieval task involving the text of the Holy Qur'an (see Figure 1). Understanding the text of the Holy Qur'an is closely related to the meaning explained in the books of Tafseer written by Islamic Scholars. Tafseer Ibn Kathir, one of the established books of Qur'an exegesis, contains ample verse relations references. Putting this into use allows not only to perform data augmentation but also to intertwine more meaning to Qur'anic verses that need to be more explicit for a LM to learn directly from the text of the verse.

Pairing. Let C_t denote a collection of books of Tafseer by Ibn Kathir. We start with extracting and paring all verse relations mentioned in Tafseer Ibn Kathir. That gives us V_t that contains distinct pairs $\{v_q, v_p\} \in V_t$ and $|V_t| \sim 11k$ pairs.

Filtering. Not all the pairs can be used for training the retrieval model because not all the verse relation pairs will be interpreted by the model as a signal of positive correlation due to meanings that are expressed indirectly. We use the cross-encoder model M_{ce} that was trained on a general domain

to score ayah pairs $s_{ce} = M_C(v_q, v_p)$. We filter out the pairs that were scored below the threshold, leaving us with V_f that contains pairs with strong positive correlations $(q, p^+) \in V_f$ and $|V_f| = 2352$ pairs.

Sampaling hard negatives. To prepare negative passages, we use the text of the Tafseer Ibn Kathir without verses' quotations. The text is split into M passages to form a collection $C^- = \{p_1^-, p_2^-, ..., p_m^-\}$ to sample negative passages. Sampling from the Holy Qur'an text is a less favorable approach. Due to the relatively small size of the Qur'anic corpus, mined negative passages may turn out to be false negatives (Qu et al., 2021). At the same time, sampling from another corpus would create easy negatives that are not beneficial for training (Ren et al., 2021; Karpukhin et al., 2020; Xiong et al., 2021), while the text of the Tafseer Ibn Kathir contains passages that are similar to the positive passages but not precisely relevant to q and are good candidates to play a role of hard negatives. To choose hard negatives, we use a retrieval model trained with a Seamise network M_B and retrieve negative passages $(p_1^-, ..., p_i^-)$ related to $\forall q \in V_f$. We score each pair (q, p^-) with the cross-encoder $s_{ce} = M_C(q, p^-)$, and use these scores in the next stage of training.

Continue training in-domain. We combine the collection of verses from the Holy Quran C^+ and the collection of passages from Tafseer Ibn Kathir C^- into one collection C_{aug} for training, which together with selected positive pairs and mined hard negatives forms new augmented dataset D_{IN} for in-domain training. We continue training SBERT-BPIT and ColBERT-BPIT on new in-domain data following the same procedure described for each model in Section 3.4. The models that come out as a result of this stage of training are SBERT-ID (Islamic Domain) and ColBERT-ID.

3.6 Results and Discussion

The performance of all models on the test dataset is collected in Table 1. All the models steadily outperform the BM25 baseline on every metric. In the category of the **GD** and **BPIT** models, the best-performing model is **ColBERT** for all metrics. In contrast, in the category of **ID** models, **SBERT** shows the best results at **MRR@10**, with a considerable improvement in performance after in-domain training on the augmented dataset (increasing from **0.48** to **0.55**).

	Recall@100	MRR @10	NDCG@5
BM25	0.15	0.27	0.15
SBERT-GD	0.2	0.43	0.23
ColBERT-GD	0.25	0.43	0.26
Cross-encoder-GD	0.19	0.37	0.22
SBERT-BPIT	0.28	0.48	0.28
ColBERT-BPIT	0.32	0.51	0.32
Cross-encoder-BPIT	0.17	0.3	0.16
SBERT-ID	0.32	**0.55**	**0.33**
ColBERT-ID	**0.33**	0.53	**0.33**

Table 1: Performance of retrieval models on the test data (English).

	Recall@100	MRR @10	NDCG@5
SBERT-ID (Saheeh Int.)	0.32	**0.55**	**0.33**
SBERT-ID (Yusuf Ali)	0.31	0.49	0.3
SBERT-ID (al-Hilali)	**0.33**	0.5	0.31
SBERT-ID (Pickthall)	0.29	0.48	0.29
ColBERT-ID (Saheeh Int.)	**0.33**	0.53	**0.33**
ColBERT-ID (Yusuf Ali)	0.28	0.46	0.27
ColBERT-ID (al-Hilali)	0.25	0.5	0.3
ColBERT-ID (Pickthall)	0.27	0.47	0.28

Table 2: Comparison of the performance of the retrieval models on the test data for different translations of the Holy Qur'an into English.

Overall, all **ID** models demonstrate superior performance, proving that training in-domain using our data augmentation technique was beneficial. Moreover, another important observation is consistent progress for **SBERT** and **ColBERT** models when training using the domain-specific model (**BPIT**) coupled with training on in-domain data. We suppose that leveraging domain adaptation of a LM that serves as a backbone for retrieval models and subsequent training of retrieval models on large general domain data before training on in-domain data is an effective approach.

In Table 2, we included a comparison and analysis of the performance of the retrieval models for different translations of the Holy Qur'an into English. We can see no significant degradation of the models' performance. The formulation of the queries contains terms from Saheeh International translation (Section 3.1), which proves that the models can maintain search relevancy with different semantics. With these results and insights, we switch to exploring how to tackle IR tasks for the Holy Quran in the Arabic Language.

4 Preparing a Retrieval Model to Extract Relevant Verses from the Holy Qur'an in Arabic

This section discusses how to address the same problem of designing an efficient neural IR model for extracting relevant verses from the Holy Qur'an in Arabic. Though the goal is essentially the same, the resources to achieve it are quite different in the case of the Arabic Language. The dataset for testing is the same as the one described in Section 3.1. We use the queries as they were initially formulated in Arabic by the authors of QRCD (Malhas and El-sayed, 2022). For the choice of the metrics, refer to Section 3.2.

4.1 Choice of Arabic LM to Tackle IR Task in the Islamic Domain

Due to a lack of manually crafted linguistic resources, Arabic is considered a low- or medium-resource language, depending on the domain of application (Xue et al., 2021; Abboud et al., 2022). Recent advances in Arabic NLP have brought a number of LMs pre-trained on Arabic corpora and new datasets translated into Arabic or initially curated in Arabic. Arabic is the language of the Holy Qur'an and the source language of numerous Islamic scholarly works. Moreover, the multi-institutional initiative has offered the Arabic NLP community an Open Islamicate Texts Initiative OpenITI (Romanov and Seydi, 2019), an excellent source for pre-training a LM for the Islamic domain. These advantageous conditions for the Islamic domain in Arabic let us skip the preliminary stage of corpus preparation and LM pre-training.

However, there is a benefit in comparing how various Arabic LMs can fit as the backbone of the IR system for the Islamic domain. Table 3 compares Arabic LMs' efficiency in tackling IR task in the Islamic domain out-of-the-box. We use a sentence-transformers framework to compare LMs to avoid a costly training stage. We add an averaging pooling layer on top of BERT embeddings and convert it into a fixed-sized sentence embedding (Reimers and Gurevych, 2019). The same model is utilized to create sentence embeddings for both queries and Qur'anic verses, and then answers to the query are found using the cosine similarity measure. The models are not ready to efficiently handle IR tasks without additional training, yet this approach let us to compare LMs' embeddings out-of-the-box. We include in the comparison the bert-base-uncased model and the BPIT model (evaluation is run on the English translation of QRCD).

As we can see from the table, most of the models perform poorly. We can also observe that pre-training on large amounts of data does not neces-

	Number of tokens/ Domain	MRR@10	NDCG@5
bert-base-arabic-camelbert-mix (Inoue et al., 2021)	17.3B/GD	0.01	0.01
bert-base-arabic-camelbert-ca (Inoue et al., 2021)	847M/ID	0.01	0.01
bert-base-arabertv02 (Antoun et al., 2020)	8.6B/GD	0.01	0.01
bert-base-arabic (Safaya et al., 2020)	8.2B/GD	0.06	0.02
bert-base-uncased (Devlin et al., 2019)	3.3B/GD	0.07	0.03
CL-AraBERT (Malhas and Elsayed, 2022)	2.7B+1.05B/GD+ID	**0.11**	**0.06**
BPIT	3.3B+50M/GD+ID	**0.11**	**0.06**

Table 3: Performance of LMs on the test dataset. GD stands for General domain and ID for Islamic domain.

	Recall@100	MRR @10	NDCG@5
Bilingual-distilled	0.12	0.26	0.15
SBERT-AR-NLI	0.21	0.38	0.21
SBERT-AR-MARCO	0.23	0.4	0.23
ColBERT-AR	0.28	0.47	**0.29**
SBERT-AR-ID	0.25	0.45	0.27
ColBERT-AR-ID	**0.29**	**0.48**	**0.29**

Table 4: Performance of retrieval models on the test dataset (Arabic).

sarily lead to better performance in IR task. CL-AraBERT performs significantly better than other Arabic LMs, and its performance is similar to the BPIT model. It is plausible that, as in the case of CL-AraBERT (Malhas and Elsayed, 2022) and the BPIT model, pre-training in a continued approach on a domain-specific corpus with specialized vocabulary starting from the general domain checkpoint helps to tackle IR task in the Islamic domain more efficiently. Another noteworthy observation is that the BPIT model exhibits this performance while pre-trained for a short period and with a small corpus of less than 50M tokens. We assume that contextualized weight distillation might help boost the efficiency during the pre-training stage. The second best performing models are bert-base-uncased and bert-base-arabic. Based on the result of Table 3, we choose CL-AraBERT as a backbone model to conduct subsequent experiments with IR task in Islamic Domain in Arabic.

4.2 Knowledge Distillation Approach to Improve Performance of Arabic LM in IR Task

The lack of manually crafted linguistic resources in low-resource languages can be tackled by knowledge distillation. Reimers and Gurevych (2020) showed that it is possible to improve the performance of sentence embedding models by mimicking the performance of a stronger model. They used parallel corpora to teach the student model to produce sentence embeddings close to the embeddings of the teacher model. Their experiment uses the English SBERT model to initialize the teacher model, and multilingual XLM-RoBERTa (Conneau et al., 2020) is used as a student model. Our experiment uses the SBERT-BPIT (Section 3.4) as

a teacher model and the bilingual EN-AR student model. The student model combines the embedding matrix of the CL-AraBERT for Arabic tokens and the BPIT model for English tokens, and the encoder weights are borrowed from the BPIT model. We use a combination of parallel datasets (EN-AR) available on the OPUS website (Tiedemann, 2012): TED2020, NewsCommentary, WikiMatrix, Tatoeba, and Tanzil, total size of training data is around 1.1M sentences (for hyperparameters details, see Appendix A). Table 4 presents the evaluation results of this approach on the test dataset (Bilingual-distilled-EN-AR model). We can see a significant improvement compared to the results of CL-AraBERT from Table 3, yet the performance is practically twice lower than the performance of the equivalent English model (SBERT-BPIT, Table 1).

4.3 Training on Arabic Natural Language Inference Dataset to Improve Sentence Embeddings

Another approach that can help to improve the quality of the sentence embeddings is training on the Natural Language Inference (NLI) dataset (Reimers and Gurevych, 2019; Bowman et al., 2015; Williams et al., 2018) . Conneau et al. (2018) introduced Cross-lingual Natural Language Inference (XNLI) comprising 7500 examples for development and test sets translated into 15 languages, including Arabic. We train CL-AraBERT on XNLI following Reimers and Gurevych (2019), using 400k machine-translated training examples that accompany XNLI development and test set (more details in Appendix A). The performance of this model (SBERT-AR-NLI, Table 4) is better than Bilingual-distilled-EN-AR, yet lower than SBERT-BPIT (Table 1).

4.4 Employing Machine-Translated Datasets to Overcome The Lack of Large Training Data

Although the quality of the machine-translated dataset is inferior to human translation, the accessibility of machine-translated text helps to generate a considerable training set which is essential for preparing a retrieval model. The experiment with training on the XNLI dataset from section 4.3 showed that training on a machine-translated dataset can achieve competitive performance. This motivates us to extend this experiment further to the MS MARCO dataset. MS MARCO is a large collection of datasets focused on deep learning in search (Bajaj et al., 2018), including the IR dataset that comprises more than half a million queries and is accompanied by a collection of 8.8M passages and 39M triplets for training. Another advantage of using MS MARCO, besides a sizable training set, is that it is more suitable for training IR systems, and we can experiment with both SBERT and ColBERT approaches to prepare retrieval models and compare their performance across languages. Bonifacio et al. (2022) presented a multilingual version of the MS MARCO dataset created using machine translation comprising 13 languages. We use the Arabic translation of MS MARCO and train SBERT-AR-MARCO equivalently to SBERT-BPIT and ColBERT-AR following the training procedure of ColBERT-BPIT (Section 3.4). Table 4 demonstrates that training on MS MARCO can give better results compared to other training approaches described in Sections 4.2 and 4.3.

4.5 In-domain Training of Retrieval Model for Qur'anic IR in Arabic

In the last stage, we perform training on in-domain data and repeat the successful experiment with dataset augmentation in English. The steps to augment dataset are the same (see Section 3.5). We use a cross-encoder trained on machine-translated MS MARCO to score ayah pairs, which results in a slightly different count of selected pairs (2723). We continue training SBERT-AR-MARCO and ColBERT-AR on in-domain data and produce SBERT-AR-ID and ColBERT-AR-ID.

The performance of these retrieval models is included in Table 4, and we can observe further improvement after training on in-domain data. The best-performing model is **ColBERT-AR-ID**, and it is plausible that the retrieval approach of the

	Recall@100	MRR @10	NDCG@5
SBERT-AR-ID	0.25	0.45	0.27
ColBERT-AR-ID	0.29	0.48	0.29
SBERT-AR-ID (passages)	0.7	0.47	0.35
ColBERT-AR-ID (passages)	**0.77**	**0.53**	**0.43**

Table 5: Performance of Arabic retrieval models on the passage retrieval task (Arabic).

ColBERT model that leverages more fine-grained interactions between a query and a verse (Khattab and Zaharia, 2020) is especially advantageous for languages with complex morphological structures, such as Arabic.

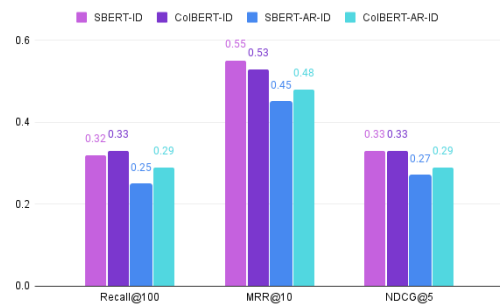## 5 Model comparison and Final analysis

Figure 3: Comparison of the retrieval models for the Islamic domain (ID) for English and Arabic across all metrics.
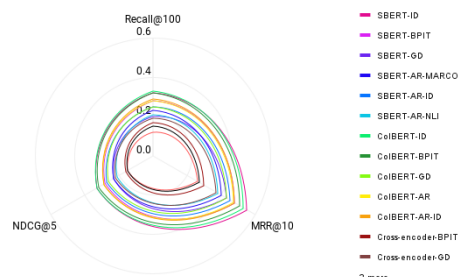

Figure 4: Comparison of the retrieval models for the Islamic domain (ID) for English and Arabic across all metrics.

Figure 3 compares all the retrieval models for the Islamic domain (ID) for English and Arabic across all metrics. A noteworthy observation is that all English retrieval models outperform their Arabic equivalents, which can be explained by the complexity of the Arabic language and the usage of machine-translated data. Nevertheless, the results of Arabic retrieval models are not far apart from English models, and specifically, with the em-

ployment of the ColBERT model, we can see a competitive performance (**0.48** for MRR@10 and **0.29** for Recall@100 and NDCG@5).

The radar chart (Figure 4) shows a more comprehensive comparison across all models. We can see that the radar chart has a tapered shape overall, with an MRR@10 axis being the most prolonged edge, indicating that all models show the best results for this metric. Moreover, NDCG@5 and Recall@100 are more proportionally placed against each other, signifying that the performance for these metrics is similar across all the models. SBERT-ID and ColBERT-ID (magenta and green colors) are located at the edge, showing the best performance. They are followed by ColBERT-BPIT and SBERT-BPIT (English models), and Arabic ColBERT and SBERT models are located in the middle of the chart. In the center, we can see BM25 and the Bilingual-distilled model, these are models with the lowest performance.

In addition, we conducted tests on two models, ColBERT-AR-ID and SBERT-AR-ID (as shown in Table 5), for the passage retrieval task (Malhas, 2023). We did not apply any passage or query expansion heuristics (Malhas, 2023). Our findings indicate that this approach is less challenging and increases the MRR@10 score, especially for the ColBERT model. The NDCG@10 score grows by 0.08 for the SBERT model and by 0.14 for the ColBERT model. Moreover, the Recall@100 grows by almost threefold.

6 Related work

Thakur et al. (2021a) proposed a data augmentation technique to train sentence transformers when little data for in-domain training is available. Wang et al. (2021) and Wang et al. (2022) experimented with domain adaptation techniques for embedding models.

The topic of the choice of hard negatives is discussed in works of: Qu et al. (2021), Ren et al. (2021), Karpukhin et al. (2020), Xiong et al., 2021.

Bashir et al. (2023) wrote a detailed overview of the state of Qur'anic NLP, including the present state of search and retrieval technologies. Most of the approaches described use keywords-based or ontology-driven search. A few works employ semantic search based on deep-learning methods: Alshammeri et al. (2021) use doc2vec; Mohamed and Shokry (2022) utilize word2vec. Malhas and Elsayed (2022) pre-trained CL-Arabert on Open-

ITI (Romanov and Seydi, 2019) starting from the AraBERT checkpoint (Antoun et al., 2020). They also introduced the first Qur'anic Reading Comprehension Dataset (QRCD) that we used as a test data for the Qur'anic IR task.

7 Conclusion

In this paper, we employed state-of-the-art approaches in IR to analyze and compare what works better to improve Qur'anic IR in English and Arabic. The results show that retrieval models in English outperform their Arabic equivalents. The inherent linguistic complexity of the Arabic language may explain this performance gap; nevertheless, transferring successful experiments from English to Arabic, applying large machine-translated datasets, and using the proposed data-augmentation technique helped to enhance the results in Qur'anic IR in Arabic.

One of the possible directions to take in the future is to extend this work to encompass more languages. This would broaden the scope of the semantic search for the Holy Qur'an, making it accessible to a larger audience. Moreover, research conducted in a multilingual environment helps to exchange insights among languages and enhance the results in Qur'anic IR.

Another essential step is to extensively evaluate real-world user queries to analyze models' performance in practice [6].

Limitations

One of the main limitations of our paper is the quality of machine-translated datasets, such as XNLI train set (Conneau et al., 2020) and mMARCO (Bonifacio et al., 2022) translation into Arabic. Using machine translation is a solution to address a lack of data for training models for low or medium-resource languages like Arabic. The quality of

---

[6]A live testing system is deployed at `rttl.ai`

automated translation is constantly improving and has reached a good quality recently; nevertheless, it is not yet equivalent to the high quality of human translation done by experts in the field.

Ethics Statement

We do not anticipate any considerable risks associated with our work. The data and other related resources in this work are publically available, and no private data is involved. We respect previous work done in the field and appropriately cite the methods and datasets we are using. To prevent misuse of pre-trained models, we carefully consider applications and provide access upon request. [7]

References

Khadige Abboud, Olga Golovneva, and Christopher DiPersio. 2022. Cross-lingual transfer for low-resource Arabic language understanding. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 225–237, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Menwa Alshammeri, Eric Atwell, and Mhd ammar Alsalka. 2021. Detecting semantic-based similarity between verses of the quran with doc2vec. *Procedia Computer Science*, 189:351–358.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. Ms marco: A human generated machine reading comprehension dataset.

Muhammad Huzaifa Bashir, Aqil M Azmi, Haq Nawaz, Wajdi Zaghouani, Mona Diab, Ala Al-Fuqaha, and Junaid Qadir. 2023. Arabic natural language processing for qur'anic research: A systematic review. *Artificial Intelligence Review*, 56(7):6801–6854.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

Adam L. Berger, Rich Caruana, David A. Cohn, Dayne Freitag, and Vibhu Mittal. 2000. Bridging the lexical chasm: statistical approaches to answer-finding. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Rishi Bommasani, Kelly Davis, and Claire Cardie. 2020. Interpreting Pretrained Contextualized Representations via Reductions to Static Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4758–4781, Online. Association for Computational Linguistics.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2022. On the opportunities and risks of foundation models.

Luiz Bonifacio, Vitor Jeronymo, Hugo Queiroz Abonizio, Israel Campiotti, Marzieh Fadaee, Roberto Lotufo, and Rodrigo Nogueira. 2022. mmarco: A multilingual version of the ms marco passage ranking dataset.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

---

[7]Contact us at hello@rttl.ai

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1993. Signature verification using a "siamese" time delay neural network. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, page 737–744, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

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*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514.

Yong Dai, Linyang Li, Cong Zhou, Zhangyin Feng, Enbo Zhao, Xipeng Qiu, Piji Li, and Duyu Tang. 2022. "is whole word masking always better for Chinese BERT?": Probing on Chinese grammatical error correction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1–8, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Fabio Henrique Kiyoiti dos Santos Tanaka and Claus Aranha. 2019. Data augmentation using gans.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

Xu Guo and Han Yu. 2022. On the domain adaptation and generalization of pretrained language models: A survey.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval. In *The 41st International ACM SIGIR Conference on Research &amp Development in Information Retrieval*. ACM.

Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun hsuan Sung, Laszlo Lukacs, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Ehsan Kamalloo, Nandan Thakur, Carlos Lassance, Xueguang Ma, Jheng-Hong Yang, and Jimmy Lin. 2023. Resources for brewing beir: Reproducible reference models and an official leaderboard.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert.

Ji Ma, Ivan Korotkov, Yinfei Yang, Keith Hall, and Ryan McDonald. 2021. Zero-shot neural passage retrieval via domain-targeted synthetic question generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1075–1088, Online. Association for Computational Linguistics.

Rana Malhas and Tamer Elsayed. 2020. Ayatec: building a reusable verse-based test collection for arabic question answering on the holy qur'an. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(6):1–21.

Rana Malhas and Tamer Elsayed. 2022. Arabic machine reading comprehension on the holy qur'an using cl-arabert. *Information Processing Management*, 59(6):103068.

Rana R Malhas. 2023. *ARABIC QUESTION ANSWERING ON THE HOLY QUR'AN*. Ph.D. thesis.

Tom M Mitchell. 1997. *Machine learning*, volume 1. McGraw-hill New York.

Ensaf Hussein Mohamed and Eyad Mohamed Shokry. 2022. Qsst: A quranic semantic search tool based on word embedding. *Journal of King Saud University - Computer and Information Sciences*, 34(3):934–945.

Vera Pavlova and Mohammed Makhlouf. 2023. BIOptimus: Pre-training an optimal biomedical language model with curriculum learning for named entity recognition. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 337–349, Toronto, Canada. Association for Computational Linguistics.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. RocketQAv2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2825–2835, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Maxim Romanov and Masoumeh Seydi. 2019. Openiti: a machine-readable corpus of islamicate texts. *Zenodo, URL: https://doi. org/10.5281/zenodo*, 3082464.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059, Barcelona (online). International Committee for Computational Linguistics.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3715–3734, Seattle, United States. Association for Computational Linguistics.

Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. 2021. Fast WordPiece tokenization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2089–2103, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. 2022. Curriculum learning: A survey.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021a. Augmented SBERT: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 296–310, Online. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021b. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS 2021) - Datasets and Benchmarks Track (Round 2)*.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. TSDAE: Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 671–688, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2022. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2345–2360, Seattle, United States. Association for Computational Linguistics.

Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Tie-Yan Liu, and Wei Chen. 2013. A theoretical analysis of ndcg type ranking measures.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Alexander Wettig, Tianyu Gao, Zexuan Zhong, and Danqi Chen. 2023. Should you mask 15% in masked language modeling? In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2985–3000, Dubrovnik, Croatia. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. 2020. Unsupervised data augmentation for consistency training.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen tau Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oğuz. 2021. Answering complex open-domain questions with multi-hop dense retrieval.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension.

A   Appendix

A.1   Hyperparameter details

Computing Infrastructure	2 x NVIDIA RTX 3090 GPU
Hyperparameter	**Assignment**
number of epochs	10
batch size	128
maximum learning rate	0.0005
learning rate optimizer	Adam
learning rate scheduler	None or Warmup linear
Weight decay	0.01
Warmup proportion	0.06
learning rate decay	linear

Table 6: Hyperparameters for continual pre-training of BPIT model.

For training SBERT and ColBERT models, we follow training recommendations implemented by the authors. To ensure fair comparison across models and languages, all the hyperparameters for SBERT models are identical, and the same applies to ColBERT models.

Computing Infrastructure	2 x NVIDIA RTX 3090 GPU
Hyperparameter	**Assignment**
number of epochs	10
batch size	64
learning rate	2e-5
pooling	mean

Table 7: Hyperparameters for training SBERT models.

Computing Infrastructure	2 x NVIDIA RTX 3090 GPU
Hyperparameter	**Assignment**
number of epochs	1
batch size	32
learning rate	1e-5

Table 8: Hyperparameters for training ColBERT models.

Computing Infrastructure	2 x NVIDIA RTX 3090 GPU
Hyperparameter	**Assignment**
number of epochs	1
batch size	32
learning rate	2e-5

Table 9: Hyperparameters for training Cross-encoders.