Relevance-aware Diverse Query Generation for Out-of-domain Text Ranking

Jia-Huei Ju[1*] **Huck Chao-Han Yang**[2] **Szu-Wei Fu**[2]
Ming-Feng Tsai[3] **Chuan-Ju Wang**[4]
[1]University of Amsterdam [2]NVIDIA Research
[3]National Chengchi University [4]Academia Sinica
j.ju@uva.nl, {hucky, szuweif}@nvidia.com,
mftsai@nccu.edu.tw, cjwang@citi.sinica.edu.tw

Abstract

Domain adaptation presents significant challenges for out-of-domain text ranking, especially when supervised data is limited. In this paper, we present ReadQG[1] (Relevance-Aware Diverse Query Generation), a method to generate informative synthetic queries to facilitate the adaptation process of text ranking models. Unlike previous approaches focusing solely on relevant query generation, our ReadQG generates diverse queries with continuous relevance scores. Specifically, we propose leveraging soft-prompt tuning and diverse generation objectives to control query generation according to the given relevance. Our experiments show that integrating negative queries into the learning process enhances the effectiveness of text ranking models in out-of-domain information retrieval (IR) benchmarks. Furthermore, we measure the quality of query generation, highlighting the underlying beneficial characteristics of negative queries. Our empirical results and analysis also shed light on potential directions for more advanced data augmentation in IR. The data and code have been released.

1 Introduction

Many domain-specific tasks lack supervised data, posing challenges for many neuarl approaches. Recently, Thakur et al. (2021) introduce an out-of-domain (OOD) information retrieval (IR) benchmark across diverse scenarios and domains. Their findings indicate that many neural text ranking models demonstrate limited effectiveness in such contexts. These tasks primarily struggle with adaptation (Gururangan et al., 2020), highlighting the issues of insufficient task- and domain-specific labeled data.

To address this, one line of research propose utilizing synthetic training data for adapting text ranking models (Ma et al., 2021; Bonifacio et al., 2022; Wang et al., 2022). These approachees employ generative models to first learn document-to-query mapping from rich-resource datasets such as MSMARCO (Bajaj et al., 2018). Subsequently, a document and its generated query can be treated as a relevant pair for fine-tuning text ranking models. Recently, these methods have been further refined with instruction-tuned large language models (LLMs) (Brown et al., 2020a; Chung et al., 2022). Such LLM-driven query generation can produce more informative query through in-context (Jeronymo et al., 2023) or few-shot learning (Dai et al., 2023).

Document
(Title) animals environment general health health general weight philosophy ethics. *(Text)* Being vegetarian helps the environment … Modern farming is …

Relevance-aware Queries
1.0 why do you think meat is bad for a planet earth
0.9 what is the philosophy of vegetarian
…
0.7 what is a vegetarian diet?
0.6 what is deforestation in asian countries
….
0.1 what is an asian diet
0.0 what is the difference between food and a burger

Figure 1: An example of generatie negative query from document inputs by ReadQG.

Compared to these works, in this study, we introduce **Re**levance-**a**ware **D**iverse **Q**uery **G**eneration (ReadQG), aiming to generate more informaive synthetic queries with lightweight generative models. Specifically, we generate both positive and (hard) negative queries from the same document, as illustrated in Figure 1. Our hypothesis is that a set of diverse relevance-aware queries can enhance relevancy representation of texts in unseen domains. To achieve this, we develop the instruction prompt and relevance prompt embeddings. The instruction prompt directs LLMs to generate query, while rel-

[*]Work partially done as an research assistnat at Academia Sinica.
[1]https://github.com/DylanJoo/readqg

Proceedings of the 9th Workshop on Representation Learning for NLP (RepL4NLP-2024), pages 26–36

evance prompt captures and conrols the *relevance dynamic* between docuemnt and multiple queries. Moreover, we develop two strategies to diversify our generated queries: self-contrastive loss and sequence calibration loss (Zhao et al., 2023). These strategies prevent ReadQG from degeneration (i.e., falling back to naive relevant query generation).

Finally, to exploit positive and negative queries genereted by ReadQG, we integrate the query-based objectives into the training process of text ranking models. Our experiments demonstrates that models fine-tuned on our synthetic data outperform the original model in terms of passage re-ranking efftiveness on the BEIR benchmark. In addition, we define and propose two metrics to measure the quality of generated queries. We observe that the query exhibiting both diversity and relevancy provide useful signals for passage re-ranking models to learn, emphasizing the importance of hard negative query.

To sum up, we propose a domain adaptation pipeline with ReadQG, tailored for out-of-domain text ranking. Our empirical results show that hard negative queries could provide useful signals. Further, the domain adaptation pipeline is built with lightweight generators and text ranking models, achieving improved effectiveness but more efficient in terms of inference time and computational costs. More details can be found in our results (Section 6.1) and our analysis (Section 6.2).

2 Backgrounds

Data augmentation in IR. Numerous IR studies have pioneered in the area of data augmentation for domain adaptation. For instance, QGen (Ma et al., 2021) used synthetic query with documents to facilitate the adaptation of bi-encoder as domain-adaptive dense retrieval. This can also be combined with negative mining techniques (Xiong et al., 2020), leading to enhanced effectiveness (Wang et al., 2022). Recent data augmentation techniques in IR have further been improved by the advancements in instruction-tuned large language models (LLMs) (Brown et al., 2020b; Chung et al., 2022). InPars (Bonifacio et al., 2022) showed that specific in-context prompting can enhance the quality of generated queries. Moreover, Promptagator (Dai et al., 2023) introduces few-shot in-context learning to bridge the gap between in-domain and out-of-domain data. Typically, all these methods center aorund augmenting synthetic queries derived from unseen document and utilizing them as additional training data.

Query Generation. Since the documents in out-of-domain corpus are usually available, we in this work focus on the query generation instead of document generation (Gao et al., 2022). Particularly, Nogueira et al. (2019) first explored the role of query generation in IR. Oguz et al. (2022) also showcased that increasing the number of synthetic queries enhances domain adaptation capability, while Lin et al. (2023) further validated diverse queries can bridge the gap between zero-shot and supervised setups. Question generation can also plays a cruicial role in improving robustness of question answering (QA) systems (Bartolo et al., 2021; Lee et al., 2020) and has broader impacts in various NLP applications such as summarization (Lyu et al., 2021) or building retrieval-intensive QA datasets (Min et al., 2020).

Diverse and controllable text generation. We further extend the concept of query generation to controllable text generation in NLP area. Similar to our goal, Cho et al. (2019) propose capturing the one-to-many relationship between texts, such as dcument-to-summaries. However, due to the discrete nature of text generation, controlling sequence diversity is challenging and often required specialized learning settings (Bowman et al., 2016) or model adjustments (See et al., 2017). Text decoding strategies also significantly influence the results (Holtzman et al., 2020). Many ongoing research focus on designing constraints and objectives, such as unlikelihood (Welleck et al., 2020) or additional constrative-like learning signals (Liu et al., 2022; Zhao et al., 2023). We hypothesize recent LLMs could transform the notion of continuous relevance and present them with divese queries, thereby improve the domain adaptation of out-of-domain text ranking.

3 Methodologies

In this work, we utilize synthesized out-of-domain training queries to tackle the domain-mismatch issues. We will first provide an overview of our domain adaptation pipeline in Section 3.1; it is also illustrated in Figure 2. Following this are the details of the two main stages in ths pipeline, including out-of-domain data augmentation and domain-adaptive fine-tuning.

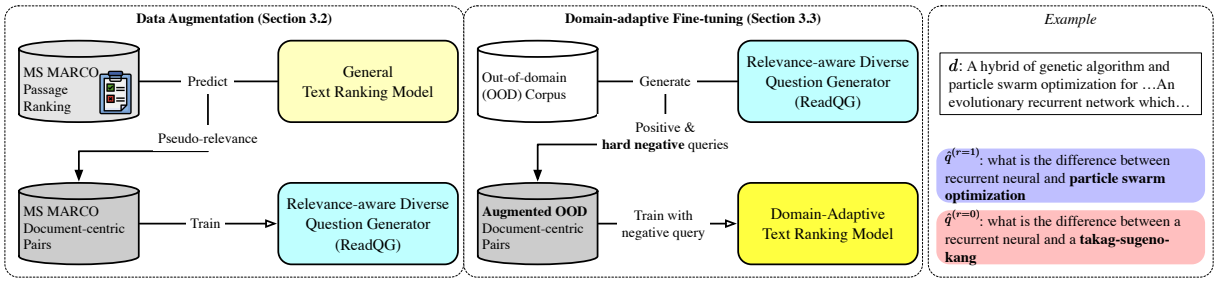

Figure 2: The domain adaptation pipeline for out-of-domain text ranking. The first block is for data augmentation (ReadQG, Section 3.2), and the second block is domain adpative fine-tuning (Section 3.3) with our augmented dataset. The last block is an example pairs we used for training text ranking models.

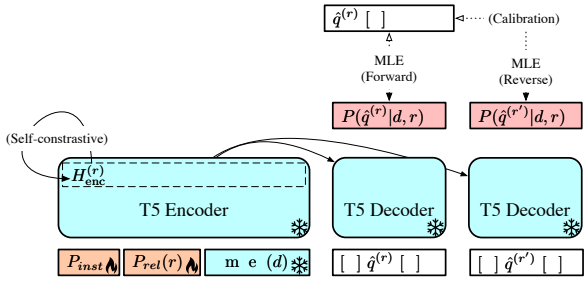

Figure 3: The relevance-aware diverse query generation.

3.1 Overview

As illustrated in the first block of Figure 2, we introduce a novel data augmentation approach, **Re**levance-**a**ware **d**iverse **Q**uery **G**eneration (ReadQG). We generate multiple synthetic queries for each documents in the targeted unseen[2] domains. These queries enables us to construct high-quality docuemnt-centric training pairs (See example in the last block in Fig. 2). By leveraging these training data, we can efficienctly transform general text re-ranking models into domain-adaptive ones as shown in second block of the figure.

3.2 ReadQG: Relevance-aware Diverse Query Generation

Unlike common studies with relevant query generation, we generate a set of relevance-aware queries $Q = [q^{(r_1)}, q^{(r_2)}, ..., q^{(r_n)}]$ for each unseen document $d \in \mathcal{D}$ as illustrated in the Figure 2. Specifically, we aim to generate the positive (relevant) query and the *hard negative* queries.[3] To achieve this, we develop a controllable query generator with diversity text generation objectives.

[2]Here, we regard MSMARCO as the source domain; thus, the retrieval tasks in BEIR are considered as unseen domains.

[3]The term hard negative query refers to less-relevant queries, as oppose to the random (negative) query.

3.2.1 Controllable Query Generation

Here we propose to parameterize such document-to-queries generation process via soft prompt-tuning (Lester et al., 2021). Our focus is specifically on learning such process of *relevance dynamic* from a rich-resource domain (i.e., MS-MARCO) with relatively smaller models instead of directly prompting LLMs. Thus, we leave the attempts of in-context prompting with larger causal LLMs as our future works. Here, we propose to parameterize such a document-to-queries generation process via soft prompt-tuning (Lester et al., 2021). Our focus is specifically on learning the process of *relevance dynamic* from a rich-resource domain (i.e., MSMARCO) with relatively smaller models instead of directly prompting LLMs. Thus, we leave the attempts of in-context prompting with larger causal LLMs as our future work.

Soft prompt-tuning with relevance. As depicted in Figure 3, we employ soft embedding prompts to control the relevance-aware query generation process. These prompts act as the composite input for documents and relevance scores. Simply put, the generator G is expected to generate a query conditioned on the given docuemnt d and also the specified relevance r. To achieve, we include two learnable embedding prompts: an instruction prompt P_{inst} and the relevance prompt $\mathrm{P}_{rel}(r)$. Thus, we can formulate the relevance-aware query generation as:

$$\hat{q}^{(r)} \leftarrow G\big(\mathrm{P}_{inst}; \mathrm{P}_{rel}(r); d\big); \quad \forall r \in [0, 1], \quad (1)$$

where the relevance r is a re-scaled continuous variable score ranging from 0 to 1 (See Section 4 for more details). It is worth noting that we only consider the prompts as trainable parameters, encouraging to leverage the inherent capabiltiy of instruction-tuned language models.

Prompt initialization. To inherit the merits of language model pretraining, we initialize P_{inst} with natural-language instructions.[4] While the relevance prompt $\mathrm{P}_{rel}(r)$ is a function of a continuous relevance score r:

$$\begin{bmatrix} r & 1-r \\ \vdots & \vdots \\ r & 1-r \end{bmatrix} \times \begin{bmatrix} \mathrm{P}^{+}_{rel} \leftarrow E(\texttt{true} \ \ldots) \\ \mathrm{P}^{-}_{rel} \leftarrow E(\texttt{false} \ \ldots) \end{bmatrix}.$$

This is basically a linear combination of P^{+}_{rel} and P^{-}_{rel} with respect to relevance. In addition, we initiliaize them with embeddings of "`true`" and "`false`" tokens before fine-tuning.

Encoder-decoder architecture. As our composite input and expected output are highly correlated syntactically, we choose the encoder-decoder architecture, Flan-T5 (Chung et al., 2022) as our backbone model. Formally, the inner flow of hidden states during training is as follow

$$\begin{aligned} H^{(r)}_{enc} &= G_{enc}\big(\mathrm{P}_{inst}; \mathrm{P}_{rel}(r); d\big); \\ H^{(r)}_{dec} &= G_{dec}\big(q_t | q_{<t}; H^{(r)}_{enc}\big), \end{aligned}$$

where H_{enc} and H_{dec} indicate the hidden states of encoder outputs and decoder outputs respectively. To optimize the parameterized embedding prompts P, we adopt the standard training recipe of teacher-forcing and maximum likelihood estimation (MLE):

$$\mathcal{L}_{mle} = -\log P\Big((H^{(r)}_{dec})^T \mathbf{W}\Big), \qquad (2)$$

where $\mathbf{W}$ is the projection layer of LM head.

3.2.2 Learning to Diverse Generation

As negative relevance of document-to-queries is intricate, we impose two objectives to encourage sequence generation diversity.

In-batch self-contrastive loss. By treating the generator's encoder G_{enc} as an independant (document) encoder, we can leverage the similar softmax objectives with mini-batch like DPR (Karpukhin et al., 2020). We define the hidden states of encoder output $H^{(r)}_{enc}$ itself as positive and the others in mini-batch as random negatives. Thus, the self-contrastive loss of document d_i with relevance r_j is as follow

$$\mathcal{L}_{\text{sc}} = \frac{\exp\Big(\phi(H^{(r_j),i}_{enc}, H^{(r_j),i}_{enc})/\tau\Big)}{\sum_{i' \in \mathbf{B}, j' \in 2m} \exp\Big(\phi(H^{(r_j),i}_{enc}, H^{(r_{j'}),i'}_{enc})/\tau\Big)}, \qquad (3)$$

where $\phi(x, y)$ represents the cosine similarity scores between x and y after average pooling, and τ is the temperature. $2m$ refers to the indices of collected relevance-query samples: $\{(r_j, q^{(r_j)})\}^{2m}_{j}$ for each document, consisting of m positive queries and m negative ones.[5] Intuitively, the semantic distance between arbitrary relevance-aware document representations H_{enc} would propogate gradient to the relevant prompts. Therefore, this loss will guide encoder G_{enc} to comprehend differently across different documents and relevance simultaneously.

Calibrated sequence likelihood. Since negative queries could be infinite, the models will tend to generate random trivial queries or non-scene texts (Welleck et al., 2020; Holtzman et al., 2020). Thus, we specifically control the sequence likelihoods of positive and negative query generation to avoid such degeneration. Inspired by sequence calibration (Zhao et al., 2023), which leverages multiple references to calibrate the sequence likelihood, we treat the relevance-contradicted query as a reference to calibrate the likelihood of the relevance-entailed query, as illustrated in Figure 3.

Specifically, for each composite input of document and relevance, we regard the likelihood of *relevance-entailed* query generation as $\log P_{fw} = \log P(q^{(r)}|d, r)$, indicating the "forward" generation. On the contrary, we calculate the "reverse" likelihood by substituting the decoder input with the contradicted one, denoted as $\log P_{rev} = \log P(q^{(r')}|d, r)$. This implies the likelihood of generating *relevance-contradicted* queries. Both the adjustments can be done efficiently within the batch; we simply swap the decoder inputs between the forward one and the reverse one as demonstrated in Figure 3. The calibration loss for each relevance-aware query generation is as follows:

[4] Among a few preliminary zero-shot tests, we cherry-picked a better one: "`Generate a question for this passage with the labels:`" as initialization.

[5] As we fix the number of sampled queries per document d, we here ignore the docuemnt dependency and replace the notations of r_{ji}, m_i by r_j, m for brevity.

$$\mathcal{L}_{cal} = \sum_{(r,r')} \max\left(0, \epsilon - \log P_{\text{fw}} + \log P_{\text{rev}}\right); \\ (\hat{q}^{(r)}, \hat{q}^{(r')}) \in \left([R_d^+; R_d^-], [R_d^-; R_d^+]\right), \quad (4)$$

where ϵ is a fixed margin that provides tolerances when forward-reverse gap is large enough. R_d^+ and R_d^- are the available postive and negative query samples and their corresponding relevance scores (Section 4). In particular, the intuition behind this loss is to increase the discrepancy between positive and negative query generation along with the given relevance distribution.

3.3 Domain-adaptative Passage Re-ranking

Afterward, as depicted in the second block in Figure 2, we can generate diverse relevance-aware queries $\hat{q}^{(r)}$ via ReadQG by feeding the document with different relevance scores. We then use these queries to construct special synthetic training pairs, each comprising a document d, a positive query $\hat{q}$, and a negative one $\hat{q}^-$.[6] These examples, especially the query-query pair, serve as additional domain-adaptive learning signals for downstream text ranking models.

Cross-encoder for relevance classification. We choose cross-encoder architecture and passage re-ranking task as the experimental testbed. And we use binary cross-entropy (BCE) loss for training cross-encoders, similar to the point-wise ranking (Nogueira and Cho, 2020), In addition, we adopt the common in-batch negative sampling (Karpukhin et al., 2020) to obtain a random negative document d^- and formuate the loss $\mathcal{L}_{bce}$ as:

$$\frac{1}{2|\mathcal{D}_\mathbf{B}|} \sum_{d}^{\mathcal{D}_\mathbf{B}} -\log P_F(\hat{q}, d) + \log P_F(\hat{q}, d^-), \quad (5)$$

where d^- is sampled from documents other than the d (i.e., the positive one) within the same mini-batch $\mathcal{D}_\mathbf{B}$; we choose the one with the highest predicted relevance as the negative document sample. Note that this is not a hard negative mining startegy (Xiong et al., 2020); it is only for avoiding underlying overfiting caused by imbalanced labels.

[6] We treat $\hat{q}^{(r=0)}$ as the hard negative query and leave the exploration of other interpolated ones as our future work.

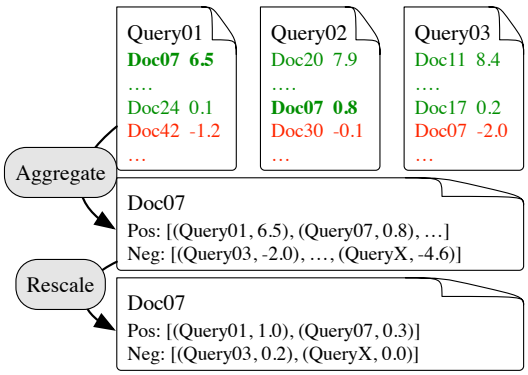

Figure 4: Construct semi-supervised document-centric pairs with MSMARCO and pseudo relevance scores. Once the documents are sorted, we aggregate queries for each document and rescale the relevance scores as document-centric pairs.

Dual learning with query-based objectives. In addition, we include query-based learning with synthetic positive and negative query pairs. The purpose is to enhance domain-specific knowledge through learning from query-query similarity. We hypothesize that the hard negative query could provide a misunderstanding comprehension of the document, offering another view of negative relevance, and thereby steering the ranking model to familiarize itself with unseen domains. Here, we adopt the margin ranking loss with query-query similarity as follows:

$$\mathcal{L}_{mr} = \sum_{d}^{\mathcal{D}_\mathbf{B}} \max\left(0, F(\hat{q}, \hat{q}^-) - F(\hat{q}, d)\right). \quad (6)$$

The intuition is that the relevance of the hard negative query should not be greater than the query-document relevance, providing extra gradient for relevance classification. Finally, we fine-tune domain-adaptive cross-encoders in a few-shot manner with the two objectives in Eq.(5) and Eq. (6).

4 Semi-supervised MSMARCO Document-centric Pairs

To fine-tune ReadQG, we collect training pairs for query generation using the MSMARCO passage ranking dataset (Bajaj et al., 2018). We utilize this dataset, along with the pseudo-relevance of BM25 hard negatives[7], which are predicted by the off-the-shelf ranking model, MiniLM.[8] This cross-

[7] https://huggingface.co/datasets/sentence-transformers/msmarco-hard-negatives

[8] https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-6-v2

encoder is fine-tuned on MS MARCO triplets. The procedure is illustrated in Figure 4.

Query-centric to document-wise aggregation. First, we sort the (query-centric) rank list by pseudo-relevance, as shown at the top of Figure 4. We also define the relevance boundary of positive and negative as 0 since MiniLM was fine-tuned with a regression-like objective. Next, we aggregate queries in a document-wise manner. For example, "Doc07" in Figure 4 appears in three ranking lists. We then re-sort the pseudo-relevance across these three lists and categorize them as positive or negative with the boundary of 0, resulting in semi-supervised document-to-queries pairs.

Re-scaling and sampling. In addition, for each pair, we rescale the relevance scores of the aggregated query set using `MinMaxScaler`.[9] The purpose of this step is to align the scores with the relevance prompt function in Eq. (1). Finally, for each document, we collect positive queries with the top-m highest relevance scores into R_d^+. Conversely, queries with the bottom-m lowest scores are considered negative query samples. Documents with fewer than 2m queries are discarded, resulting in approximately 4.7M document-centric training pairs for ReadQG.

5 Experimental Setups

We will first report the setup of the two stages in the proposed pipeline: data augmnetation using ReadQG (Section 3.2), and domain adaptive passage re-ranking (Section 3.3) fine-tuned on the synthetic training data.

5.1 Training and Inference Setups

Stage I: Data augmentation. Our ReadQG is initialized with Flan-T5 base checkpoint.[10] with only a few tunable parameters of embedding prompts (Section 3.2). We set the length of instruction and relevance prompt as 10 and 5, respectively, ensuring the lightweight training and inference overhead. The generator is then fine-tuned on the semi-supervised training pairs (See Section 4 for details) for 20K steps with a constant learning rate 1e-2. The maximum sequence length of input (document) and target (query) are 128 and 16. We use batch size of 32, comprising 4 documents and $m = 4$, for each positive and negative query samples.

During inference, to control the generation of positive and *hard* negative query, we specify the relevance scores as $r = 1$ and $r = 0$, respectively. We then construct the composite input for positive and negative query generation as described in Eq. (1). The maximum sequence length of input and output are 384 and 64 with top-k ($k = 10$) decoding strategies (Fan et al., 2018). We also analyze greedy and beam search decoding strategy in Section 6.2.

Stage II: Domain-adaptative passage re-ranking Once we have the synthetic training pairs, we use them to fine-tune domain-adaptive passage ranking models. We initialize the models with MiniLM pre-trained on MSMARCO passage ranking, the same model used for pseudo-relevance labels in Section 4. Then, we fine-tune the model with batch size 8 for 2 epoch[11] with learning rate 7e-6. An epoch of training steps is defined as the corpus size divided by batch size, as we only generate one query pair per document. We set the maximum length as 384. Other training hyperparameters follow the default setups of SentenceBERT cross-encoder.[12]

5.2 Evaluation Setups

BEIR benchmark. We experiment on BEIR and select several tasks with corpus size is less than 100K for evaluation, including NFCorpus (NFC, 3.6K), FiQA (FQA, 57.6K), ArguAna (ARG, 8.7K), SCIDOCS (SCD, 25.7K), and SciFact (SCF, 5.2K). For brevity, we will use these abbreviations henceforth. We first validate the domain adaptation capability by out-of-domain text ranking effectiveness nDCG@10, the official metric in BEIR. We used the fixed candidates from BM25 top-100 retrieved results and foucsed on the re-ranking effectiveness for simpler comparison.

Performance metrics. We also investigate the unique charateristic of generated queries directly from an IR perspective. Specifically, by using the off-the-shelf bi-encoders and cross-encoders, we can analyze the useful properties of synthetic query. We introduce the following:

1. **Diversity.** We regard the generated queries Q from an unseen document as different texts.

[9] https://scikit-learn.org

[10] https://huggingface.co/google/flan-t5-base

[11] We found there is no improvement after the second.

[12] https://github.com/UKPLab/sentence-transformers

#	Retrieval + Re-ranking (synthetic data)	Objectives $(\hat{q}, d)$	Objectives $(\hat{q}, d, \hat{q}^-)$	Params (M) Gen./Rank	nDCG@10 NFC	FQA	ARG	SCD	SCF	Avg.
	BM25	-	-	-	32.5	23.6	41.4	15.8	66.5	36.0
(0)	BM25 + MiniLM-MS			- / 0.2M	35.0	34.7	41.7	16.6	68.8	39.4
(1)	BM25 + MiniLM-MS (InPars-v2 data)	✓	✗	6B / 0.2M	35.4	**35.2**	42.3	16.6	69.8	39.8
(2)	BM25 + MiniLM-MS (ReadQG)	✓	✗	220M / 0.2M	35.4	34.0	42.8	15.7	71.4	39.8
(3)	BM25 + MiniLM-MS (ReadQG)	✓	✓	220M / 0.2M	**35.5**	34.4	**49.6**	**16.7**	**71.6**	**41.6**

Table 1: The Out-of-domain text re-ranking effectivenss (nDCG@10) with top-100 candidates retrieved using BM25. The third and forth columns indicate learning objectives of Eq. (3) and Eq. (6).

Thus, we first encode n_Q queries with off-the-shelf bi-encoders[13] E^*. Then, we compute the average pairwise angular distance (Cer et al., 2018) across n query embeddings as follow:

$$\frac{2}{n_Q^2 - n_Q} \sum_{i=1}^{n_Q} \sum_{j=i+1} \Omega\left(E^*(q_i), E^*(q_j)\right),$$

where we set $n_Q = 11$ and indicate relevance scores $r \in \{0, 0.1, ..., 1.0\}$. $\Omega(u, v)$ indicates the angular distance between vectors u and v.

2. **Relevancy.** In addition, we feed the document with our generated positive and negative query into another effective cross-encoder model. We use monoT5-3B-InPars-v2 (Jeronymo et al., 2023) as we assume the predicted scores of the larger model can accurately reflect the true relevance between query and document. These metrics include

$$\begin{aligned} \text{rel}^+ &= F^*(\hat{q}^{(r=1)}, d); \\ \text{rel}^- &= F^*(\hat{q}^{(r=0)}, d); \\ \Delta\text{rel} &= \text{rel}^+ - \text{rel}^-. \end{aligned}$$

Note that all metrics will first be calculated per document and then take the average across documents in our later results.

6 Empirical Results

In this section, we first validate the text ranking effectiveness using the synthetic data constructed by ReadQG. Then, we explore the query generation effectiveness via the aforementioned three performance metrics.

6.1 Main Results

Out-of-domain text ranking. Table 1 shows that the domain-adaptive text ranking models fine-tuned with an additional negative query from ReadQG (condition #(3)) increase the average nDCG@10 by approximately two points compared to the initial zero-shot one (i.e., #(0)). This indicates the positive query together with the *hard negative* query can transfer useful signals during learning. Moreover, condition #(3) outperforms our baseline #(2), the condition used only positive queries. This implies the role of negative relevance in Eq. (6) can guide models to accurately estimate the relevancy of queries and documents.

We also compare with the generated query from InPars-v2 (Jeronymo et al., 2023) by fine-tuning the synthetic pairs with identical settings. Note that we here exclude negative documents in the released data[14] for a fair comparison. We also align the amounts of training pairs by random sampling. By comparing conditions #(1) and #(2), we observe the positive queries generated by ReadQG can perform on par with InPars-v2's[15] with a smaller generator (i.e., 220M parameters), demonstrating an efficient alternative to transfer knowledge from rich-resource MSMARCO dataset (Bajaj et al., 2018).

#	Div.	$(\text{rel}^+/\text{rel}^-/\Delta\text{rel})$	nDCG@10
(a) $\mathcal{L}_{mle}$	.218	(.970/.859/.111)	.707
(b) + $\mathcal{L}_{sc}$	.154	(.957/.938/.019)	.709
(c) + $\mathcal{L}_{cal}$	.269	(.967/.732/.235)	.706
(d) + $\mathcal{L}_{sc} + \mathcal{L}_{cal}$	.219	(.973/.935/.037)	.716

Table 2: Quality of generated query with different diversity learning objectives. We use SCIFACT as example. The reported metrics are diversity (Div.) and relevancy and nDCG@10.

Generation Quality. To better understand generated queries, we compare the variants of our proposed learning objectives in Section 3.2, as shown in different rows in Table 2. We fixed all the settings of query generation, including prompt length

[13] We use GTE encoder (Li et al., 2023) as it has been pre-trained on scientific corpora.

[14] https://huggingface.co/datasets/inpars/generated-data

[15] For a fair comparison, we shuffle the generated queries and sample the same size as ours. And we only used the positive query-document pairs.

$\|P_{\text{rel}}\|$	Div.	$(\text{rel}^+/\text{rel}^-/\Delta\text{rel})$	nDCG@10
1	.204	(.972/.916/.056)	70.4
5	.219	(.973/.935/.037)	71.6
10	.192	(.986/.944/.042)	70.4

Table 3: The impacts of different length of relevance soft prompts. We use the SCIFACT dataset as an example.

Decode	NFC	FQA	ARG	SCD	SCF	Avg.
Beams= 1	35.6	33.8	50.1	16.6	71.6	41.5
Beams= 3	35.5	33.5	52.8	16.6	71.7	42.0
Top-k(10)	35.5	34.4	49.6	16.7	71.6	41.6

Table 4: The impacts of different sequence decoding strategy. The reported scores are nDCG@10.

as 5 and greedy decoding. We observe there is no single metric solely related to the ranking effectiveness. However, one interesting finding is that the condition #(d) (i.e., MLE + two diverse generation losses) and condition #(a) (MLE only) have similar diversity, but their relevance scores of negative queries (i.e., rel^-) differ; condition #(d) has .935 but #(a) is .859. This highlights the unique characteristic of harder negative query – high diversity but also high relevance (Div. $\uparrow$; rel^- $\uparrow$) – with the same document. This also shows that calibration loss with self-contrastive loss can complement each other and produce better relevance-aware diver queries. The high diversity sometimes hurts text ranking effectiveness such as condition #(c), meaning that the negative query is too trivial (i.e., random negative query).

6.2 ReadQG Analysis

Length of relevance prompt. In Table 3, we investigate different lengths of soft relevance prompts as many studies have claimed the impact of prompt length is significant (Li and Liang, 2021; Lester et al., 2021). We train generators with fixed learning objectives and inference with the same greedy decoding. Comparing the first two rows (lengths of 1 and 5), we observe the improvement is attributed to the better expression capability with longer prompts, enabling to parameterize more non-linearity of query-document relevance. However, further increasing prompt length may not significantly increase the diversity of generated queries and result in lower diversity (Div. $\downarrow$). Moreover, the retrieval effectiveness would be limited when the prompt length is too long even though the relevance of the negative query is higher (rel^- $\uparrow$). This finding aligns with our observation in Table 1 that the informative *hard* negative query is meaningful when exhibits both high diversity and relevance.

Decoding strategies. Table 4 demonstrates the different decoding strategies. Intuitively, we consider beam search as the most effective option for negative query generation. However, the top-k sampling is the better strategy considering the efficiency. It can balance diversity and efficiency. However, since we only test the *hard* negative with relevance score $r = 0$, it required more investigation for interpolated query and the corresponding learning design of text ranking. We hypothesize the diverse generated queries can similarly benefit the dense retrieval models like Lin et al. (2023), which we leave it as our future work.

7 Conclusion

In this study, we present relevance-aware diverse queries generation and validate several setups for constructing more informative queries. The generation of negative query can benefit from appropriate soft-prompt tuning and diverse generation constraints, resulting in a more effective learning process of text ranking models. Thus, we consider the negative query generation as a potential research direction. There are several other avenues for future work, including (1) scaling up ReadQG or prompting larger LLMs for negative queries; (2) mining hard negative documents with hard negative query; (3) fine-tuning bi-encoders dense retrieval with an additional negative query; (4) exploreing more complicated learning tehcniques (Ren et al., 2021; Li et al., 2021) that can fully exploit interpolated negative queries. Regarding the domain adaptation, we suspect the query distribution will be another important factor, as seen in promptagator (Dai et al., 2023) boost the performance with few in-domain data. More empirical evaluation on other benchmarks like Massive Textual Evaluation Benchmark (mteb) can also provide deeper insights of the usefulness of hard negative queries.

Acknowledgements

This research was partially supported by the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, https://hybrid-intelligence-centre.nl.

References

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.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. Inpars: Unsupervised dataset generation for information retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 2387–2392, New York, NY, USA. Association for Computing Machinery.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020a. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

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

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Jaemin Cho, Minjoon Seo, and Hannaneh Hajishirzi. 2019. Mixture content selection for diverse sequence generation. 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 3121–3131, Hong Kong, China. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith Hall, and Ming-Wei Chang. 2023. Promptagator: Few-shot dense retrieval from 8 examples. In *The Eleventh International Conference on Learning Representations*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Precise zero-shot dense retrieval without relevance labels.

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.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Vitor Jeronymo, Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, Roberto Lotufo, Jakub Zavrel, and Rodrigo Nogueira. 2023. Inpars-v2: Large language models as efficient dataset generators for information retrieval.

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.

Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, and Sung Ju Hwang. 2020. Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional VAEs. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics, pages 208–224, Online. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Yizhi Li, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2021. More robust dense retrieval with contrastive dual learning. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '21, page 287–296, New York, NY, USA. Association for Computing Machinery.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning.

Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen tau Yih, and Xilun Chen. 2023. How to train your dragon: Diverse augmentation towards generalizable dense retrieval.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. BRIO: Bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903, Dublin, Ireland. Association for Computational Linguistics.

Chenyang Lyu, Lifeng Shang, Yvette Graham, Jennifer Foster, Xin Jiang, and Qun Liu. 2021. Improving unsupervised question answering via summarization-informed question generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4134–4148, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage re-ranking with bert.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document expansion by query prediction.

Barlas Oguz, Kushal Lakhotia, Anchit Gupta, Patrick Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Scott Yih, Sonal Gupta, and Yashar Mehdad. 2022. Domain-matched pre-training tasks for dense retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1524–1534, Seattle, United States. Association for Computational Linguistics.

Ruiyang Ren, Shangwen Lv, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. Pair: Leveraging passage-centric similarity relation for improving dense passage retrieval. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

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

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.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*.

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.

Yao Zhao, Mikhail Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J Liu. 2023. Calibrating sequence likelihood improves conditional language generation. In *The Eleventh International Conference on Learning Representations*.