GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval

Kexin Wang[1], **Nandan Thakur**[2]*, **Nils Reimers**[3]*, **Iryna Gurevych**[1]
[1] Ubiquitous Knowledge Processing Lab, Technical University of Darmstadt
[2] University of Waterloo, [3] Hugging Face
www.ukp.tu-darmstadt.de

Abstract

Dense retrieval approaches can overcome the lexical gap and lead to significantly improved search results. However, they require large amounts of training data which is not available for most domains. As shown in previous work (Thakur et al., 2021b), the performance of dense retrievers severely degrades under a domain shift. This limits the usage of dense retrieval approaches to only a few domains with large training datasets.

In this paper, we propose the novel unsupervised domain adaptation method *Generative Pseudo Labeling* (GPL), which combines a query generator with pseudo labeling from a cross-encoder. On six representative domain-specialized datasets, we find the proposed GPL can outperform an out-of-the-box state-of-the-art dense retrieval approach by up to 9.3 points nDCG@10. GPL requires less (unlabeled) data from the target domain and is more robust in its training than previous methods.

We further investigate the role of six recent pre-training methods in the scenario of domain adaptation for retrieval tasks, where only three could yield improved results. The best approach, TSDAE (Wang et al., 2021) can be combined with GPL, yielding another average improvement of 1.4 points nDCG@10 across the six tasks. The code and the models are available [1].

1 Introduction

Information Retrieval (IR) is a central component of many natural language applications. Traditionally, lexical methods (Robertson et al., 1994) have been used to search through text content. However, these methods suffer from the lexical gap (Berger et al., 2000) and are not able to recognize synonyms and distinguish between ambiguous words.

Recently, information retrieval methods based on dense vector spaces have become popular to address these challenges. These dense retrieval methods map queries and passages[2] to a shared, dense vector space and retrieve relevant hits by nearest-neighbor search. Significant improvement over traditional approaches has been shown for various tasks (Karpukhin et al., 2020; Xiong et al., 2021). This method is also adapted increasingly by industry to enhance the search functionalities of various applications (Choi et al., 2020; Huang et al., 2020).

However, as shown in Thakur et al. (2021b), dense retrieval methods require a large amount of training data to work well.[3] Most importantly, dense retrieval methods are extremely sensitive to domain shifts: Models trained on MS MARCO perform rather poorly for questions for COVID-19 scientific literature (Wang et al., 2020; Voorhees et al., 2021). The MS MARCO dataset was created before COVID-19, hence, it does not include any COVID-19 related topics and models did not learn how to represent this topic well in a vector space.

In this work, we present *Generative Pseudo Labeling* (GPL), an unsupervised domain adaptation technique for dense retrieval models (see Figure 1). For a collection of paragraphs from the desired domain, we use an existing pre-trained T5 encoder-decoder to generate synthetic queries. These input passages are viewed as the positive passages for the generated queries. For each generated query, we retrieve the most similar paragraphs as the negative passages using an existing dense retrieval model. We term this step *negative mining* and term these negative passages *hard negatives*. Finally, we use an existing cross-encoder to score each (query, passage)-pair and train a dense retrieval

*Contributions made while being employed at UKP Lab.

[1] https://github.com/UKPLab/gpl

[2] We use passage to refer to text of any length.

[3] For reference, the popular MS MARCO dataset (Nguyen et al., 2016) has about 500k training instances; the Natural Questions dataset (Kwiatkowski et al., 2019) has more than 100k training instances.

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2345 - 2360
July 10-15, 2022 ©2022 Association for Computational Linguistics

model on these generated, pseudo-labeled queries using MarginMSE-Loss (Hofstätter et al., 2020).

We use publicly available models for query generation, negative mining, and the cross-encoder, which have been trained on the MS MARCO dataset (Nguyen et al., 2016), a large-scale dataset from Bing search logs combined with relevant passages from diverse web sources. We evaluate GPL on six representative domain-specific datasets from the BeIR benchmark (Thakur et al., 2021b). **GPL improves the performance by up to 9.3 points nDCG@10** compared to state-of-the-art model trained solely on MS MARCO. Compared to the previous state-of-the-art domain-adaptation method QGen (Ma et al., 2021; Thakur et al., 2021b), GPL improves the performance by up to 4.5 nDCG@10 points. Training with GPL is easy, fast, and data efficient.

We further analyze the role of six recent pre-training methods in the scenario of domain adaptation for retrieval tasks. The best approach is TSDAE (Wang et al., 2021), that outperforms the second best approach, Masked Language Modeling (Devlin et al., 2019) on average by 2.5 points nDCG@10. TSDAE can be combined with GPL, yielding another average improvement of 1.4 point nDCG@10.

2 Related Work

Pre-Training based Domain Adaptation. The most common domain adaptation technique for transformer models is domain-adaptive pre-training (Gururangan et al., 2020), which continues pre-training on in-domain data before fine-tuning with labeled data. However, for retrieval it is often difficult to get in-domain labeled data and models are applied in a zero-shot setting on a given corpus. Besides Masked Language Modeling (MLM) (Devlin et al., 2019), different pre-trained strategies specifically for dense retrieval methods have been proposed. Inverse Cloze Task (ICT) (Lee et al., 2019) generates query-passage pair by randomly selecting one sentence from the passage as the query and the other part as the paired passage. ConDensor (CD) (Gao and Callan, 2021) applies MLM on top of the CLS token embedding from the final layer and the other context embeddings from a previous layer to force the model to learn meaningful CLS representation. SimCSE (Gao et al., 2021a; Liu et al., 2021) passes the same input twice through the network with different dropout masks and minimizes the distance of the resulting embeddings, while Contrastive Tension (CT) (Carlsson et al., 2021) passes the input through two different models. TSDAE (Wang et al., 2021) uses a denoising auto-encoder architecture for representation learning: Words from the input text are removed and passed through an encoder to generate a fixed-sized embedding. A decoder must reconstruct the original text without noise. As we show in Appendix E, just using these unsupervised techniques is not sufficient and the resulting models perform poorly.

So far, ICT and CD have only been studied on in-domain performance, i.e. a large in-domain labeled dataset is available which is used for subsequent supervised fine-tuning. SimCSE, CT, and TSDAE have been only studied for unsupervised sentence embedding learning. As our results show in Appendix E, they do not work at all for purely unsupervised dense retrieval.

If these pre-training approaches can be used for unsupervised domain adaptation for dense retrieval was so far unclear. In this work, we transfer the setup from Wang et al. (2021) to dense retrieval and first pre-train on the target corpus, followed by supervised training on labeled data from MS MARCO (Nguyen et al., 2016)[4]. Performance is then measured on the target corpus.

Query Generation. Query generation has been used to improve retrieval performances. Doc2query (Nogueira et al., 2019a,b) expands passages with predicted queries, generated by a trained encoder-decoder model, and uses traditional BM25 lexical search. This performed well in the zero-shot retrieval benchmark BeIR (Thakur et al., 2021b). Ma et al. (2021) proposes QGen, that uses a query generator trained on general domain data to synthesize domain-targeted queries for the target corpus, on which a dense retriever is trained from scratch. As a concurrent work, Liang et al. (2020) also proposes the similar method. Following this idea, Thakur et al. (2021b) views QGen as a post-training method to adapt powerful MS MARCO retrievers to the target domains.

Despite the success of QGen, previous methods only consider the cross-entropy loss with in-batch negatives, which provides coarse-grained relevance and thus limits the performance. In this work, we show that extending this approach by using pseudo-labels from a cross-encoder together with hard neg-

[4] As shown in Wang et al. (2021), training in the reverse order, i.e. first on MS MARCO and then on the target corpus usually performs poorly.

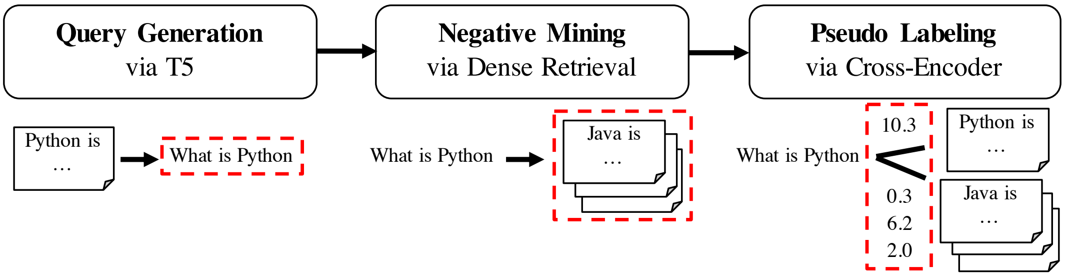

Figure 1: Generative Pseudo Labeling (GPL) for training domain-adapted dense retriever. First, synthetic queries are generated for each passage from the target corpus. Then, the generated queries are used for mining negative passages. Finally, the query-passage pairs are labeled by a cross-encoder and used to train the domain-adapted dense retriever. The output at each step is marked with dashed boxes.

atives can boost the performance by several points nDCG@10.

Other Methods. Recently, Xin et al. (2021) proposes MoDIR to use Domain Adversarial Training (DAT) (Ganin et al., 2016) for unsupervised domain adaptation of dense retrievers. MoDIR trains models by generating domain invariant representations to attack a domain classifier. However, as argued in Karouzos et al. (2021), DAT trains models by minimizing the distance between representations from different domains and such learning objective can result in bad embedding space and unstable performance. For sentiment classification, Karouzos et al. (2021) proposes UDALM based on multiple stages of training. UDALM first applies MLM training on the target domain; and it then applies multi-task learning on the target domain with MLM and on the source domain with a supervised objective. However, as shown in section 5, we find this method cannot yield improvement for retrieval tasks.

Pseudo Labeling and Cross-Encoders: Bi-Encoders map queries and passage independently to a shared vector space from which the query-passage similarity is computed. In contrast, cross-encoders (Humeau et al., 2020) work on the concatenation of the query and passage and predict a relevance score using cross-attention between query and passage. This can be used in a re-ranking setup (Nogueira and Cho, 2019), where the relevancy is predicted for all query-passage-pairs for a small candidate set. Previous work has shown that cross-encoders achieve much higher performances (Thakur et al., 2021a; Hofstätter et al., 2020; Ren et al., 2021) and are less prone to domain shifts (Thakur et al., 2021b). But cross-encoders come with an extremely high computational overhead, making them less suited for a production setting. Transferring knowledge from cross-encoder to bi-encoders have been shown previous for sentence embeddings (Thakur et al., 2021a) and for dense retrieval: Hofstätter et al. (2020) predict cross-encoder scores for (query, positive)-pairs and (query, negative)-pairs and learns a bi-encoder to predict the margin between the two scores. This has been shown highly effective for in-domain dense retrieval.

3 Method

This section describes our proposed *Generative Pseudo Labeling* (GPL) method for the unsupervised domain adaptation of dense retrievers. Figure 1 illustrates the idea of GPL.

For a given target corpus, we generate for each passage three queries (cf. Table 3) using an T5-encoder-decoder model (Raffel et al., 2020). For each of the generated queries, we use an existing retrieval system to retrieve 50 negative passages. Dense retrieval with a pre-existing model was slightly more effective than BM25 lexical retrieval (cf. Appendix A). For each (query, positive, negative)-tuple we compute the margin $\delta = \mathrm{CE}(Q, P^+,) - \mathrm{CE}(Q, P^-)$ with CE the score as predicted by a cross-encoder, Q the query and P^+/P^- the positive / negative passage.

We use the synthetic dataset $D_{\text{GPL}} = \{(Q_i, P_i, P_i^-, \delta_i)\}_i$ with the MarginMSE loss (Hofstätter et al., 2020) for training a domain-adapted dense retriever that maps queries and passages into the shared vector space.

Our method requires from the target domain just an unlabeled collection of passages. Further, we use use pre-existing T5- and cross-encoder models that have been trained on the MS MARCO passages dataset.

Query Generation: To enable supervised training on the target corpus, synthetic queries can be generated for the target passages using a query generator trained on a different, existing dataset like MS MARCO. Previous work QGen (Ma et al., 2021) used the simple MultipleNegativesRanking (MNRL) loss (Henderson et al., 2017; van den Oord et al., 2018) with in-batch negatives[5] to train the model:

$$L_{\text{MNRL}}(\theta) = -\frac{1}{M}\sum_{i=0}^{M-1}\log\frac{\exp\left(\tau\cdot\sigma(f_\theta(Q_i), f_\theta(P_i))\right)}{\sum_{j=0}^{M-1}\exp\left(\tau\cdot\sigma(f_\theta(Q_i), f_\theta(P_j))\right)}$$

where P_i is a relevant passage for Q_i; σ is a certain similarity function for vectors; τ controls the sharpness of the softmax normalization; M is the batch size.

MarginMSE loss: MultipleNegativesRanking loss considers only the coarse relationship between queries and passages, i.e. the matching passage is considered as relevant while all other passages are considered irrelevant. However, the query encoder is not without flaws and might generate queries that are not answerable[6] by the passage. Further, other passages might actually be relevant as well for a given query, which is especially the case if training is done with hard negatives as we do it for GPL.

In contrast, MarginMSE loss (Hofstätter et al., 2020) uses a powerful cross-encoder to soft-label (query, passage) pairs. It then teaches the dense retriever to mimic the score margin between the positive and negative query-passage pairs. Formally,

$$L_{\text{MarginMSE}}(\theta) = -\frac{1}{M}\sum_{i=0}^{M-1}|\hat{\delta}_i - \delta_i|^2 \qquad (1)$$

where $\hat{\delta}_i$ is the corresponding score margin of the student dense retriever, i.e. $\hat{\delta}_i = f_\theta(Q_i)^T f_\theta(P_i) - f_\theta(Q_i)^T f_\theta(P_i^-)$. Here the dot-product is usually used due to the infinite range of the cross-encoder scores.

This loss is a critical component of GPL, as it solves two major issues from the previous QGen method: A badly generated query for a given passage will get a low score from the cross-encoder, hence, we do not expect the dense retriever to put the query and passage close in the vector space. A false negative will lead to a high score from the cross-encoder, hence, we do not force the dense retriever to assign a large distance between the corresponding embeddings. In section 6.3, we show that GPL is a lot more robust to badly generated queries than the previous QGen method.

4 Experiments

In this section, we describe the experimental setup, the datasets used and the baselines for comparison.

4.1 Experimental Setup

We use the MS MARCO passage ranking dataset (Nguyen et al., 2016) as the data from the source domain. It has 8.8M passages and 532.8K query-passage pairs labeled as relevant in the training set. We select six representative datasets from the BeIR benchmark as the data from the target domain (cf. subsection 4.2). As Table 1 shows, a state-of-the-art dense retrieval model, achieving an MRR@10 of 33.2 points on the MS MARCO passage ranking dataset, performs poorly on the six selected domain-specific retrieval datasets when compared to simple BM25 lexical search.

We use the DistilBERT (Sanh et al., 2019) for all the experiments. We use the concatenation of the title and the body text as the input passage for all the models. We use a maximum sequence length of 350 with mean pooling and dot-product similarity by default. For QGen, we use the default setting in Thakur et al. (2021b): 1-epoch training and batch size 75. For GPL, we train the models with 140k training steps and batch size 32. To generate queries for both QGen and GPL, we use the DocT5Query (Nogueira et al., 2019a) generator trained on MS MARCO and generate [7] queries using nucleus sampling with temperature 1.0, $k = 25$ and $p = 0.95$. To retrieve hard negatives for both GPL and the zero-shot setting of MS MARCO training, we use two dense retrievers with cosine-similarity trained on MS MARCO: *msmarco-distilbert-base-v3* and *msmarco-MiniLM-L-6-v3* from Sentence-Transformers[8]. The zero-shot performance of these two dense retrievers is available in Appendix B. We retrieve 50 negatives

[5] In-batch negatives mean that within the same batch, the passages in example i are viewed as the negatives for example j $(i \neq j)$.

[6] For example, some generated queries are statements instead of questions.

[7] We use the script from BeIR at `https://github.com/UKPLab/beir`.

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

using each retriever and uniformly sample one negative passage and one positive passage for each training query to form one training example. For pseudo labeling, we use the *ms-marco-MiniLM-L-6-v2*[9] cross-encoder. For all the pre-training methods (e.g. TSDAE and MLM), we train the models for 100K training steps and with batch size 8.

As shown in Section 6, small corpora require more generated queries and for large corpora, a small down-sampled subset (e.g. 50K) is enough for good performance. Based on these findings, we adjust the number of generated queries per passage $q_{\text{avg.}}$ and the corpus size $|C|$ to make the total number of generated queries equal to a fixed number, 250K, i.e. $q_{\text{avg.}} \times |C| = 250\text{K}$. In detail, we first set $q_{\text{avg.}} >= 3$ and uniformly down-sample the corpus if $3 \times |C| > 250\text{K}$; then we calculate $q_{\text{avg.}} = \lceil 250\text{K}/|C| \rceil$. For example, the $q_{\text{avg.}}$ values for FiQA (original size = 57.6K) and Robust04 (original size = 528.2K) are 5 and 3, resp. and the Robust04 corpus is down-sampled to 83.3K. QGen and GPL share the generated queries for fair comparision.

4.2 Evaluation

As our methods focus on domain adaptation to specialized domains, we selected six domain-specific text retrieval tasks from the BeIR benchmark (Thakur et al., 2021b): FiQA (financial domain) (Maia et al., 2018), SciFact (scientific papers) (Wadden et al., 2020), BioASQ (biomedical Q&A) (Tsatsaronis et al., 2015), TREC-COVID (scientific papers on COVID-19) (Roberts et al., 2020), CQADupStack (12 StackExchange subforums) (Hoogeveen et al., 2015) and Robust04 (news articles) (Voorhees, 2005). These selected datasets each contain a corpus with a rather specific language and can thus act as a suitable test bed for domain adaptation.

The detailed information for all the target datasets is available at Appendix C. We make modification on BioASQ and TREC-COVID. For efficient training and evaluation on BioASQ, we randomly remove irrelevant passages to make the final corpus size to 1M. In TREC-COVID, the original corpus has many documents with a missing abstract. The retrieval systems that were used to create the annotation pool for TREC-COVID often ignored such documents. This leads to a strong annotation bias on text length for these documents, since this dataset contains only titles and abstracts. Hence, we removed all documents with a missing abstract from the corpus. The evaluation results on the original BioASQ and TREC-COVID are available at Appendix D. Evaluation is done using nDCG@10.

4.3 Baselines

Zero-Shot Models: We apply supervised training on MS MARCO or PAQ (Lewis et al., 2021) and evaluate the trained retrievers on the target datasets. (a) **MS MARCO** represents a distilbert-base dense retrieval model trained with MarginMSE on the MS MARCO dataset with batch-size 75 for 70k steps. (b) **PAQ** (Oguz et al., 2021) represents MNRL training on the PAQ dataset. (c) **PAQ + MS MARCO** represents MNRL training on PAQ followed by MarginMSE training on MS MARCO. (d) **TSDAE$_{\textbf{MS MARCO}}$** represents TSDAE (Wang et al., 2021) pre-training on MS MARCO followed by MarginMSE training on MS MARCO. (e) **BM25** system based on lexical matching from Elasticsearch[10].

Previous Domain Adaptation Methods: We include two previous unsupervised domain adaptation methods, UDALM (Karouzos et al., 2021) and MoDIR (Xin et al., 2021). For **UDALM**, we apply MLM training on the target corpus and then apply the multi-task training of MarginMSE training on MS MARCO and MLM training on the target corpus. For **MoDIR**, it starts from the ANCE checkpoint and apply domain adversarial training on MS MARCO and the target dataset. As of writing, the training code of MoDIR is not public, but domain adapted models for 5 out of 6 datasets have been released by the authors.

Pre-Training based Domain Adaptation: We follow the setup proposed in Wang et al. (2021) on domain-adapted pre-training: We pre-train the dense retrievers with different methods on the target corpus and then continue to train the models on MS MARCO with MarginMSE loss. The pre-training methods consist of: (a) **CD** (Gao and Callan, 2021) extracts the hidden representations from an intermediate layer and applies MLM on the CLS token representation and these extracted hidden representations[11]. (b) **SimCSE** (Gao et al., 2021b; Liu et al., 2021) simply encode the same

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

[10] https://www.elastic.co

[11] CD can only be applied with CLS pooling.

Dataset Method	FiQA	SciFact	BioASQ	TRECC.	CQADup.	Robust04	Avg.
Zero-Shot Models							
MS MARCO	26.7	57.1	52.9	66.1	29.6	39.0	45.2
PAQ	15.2	53.3	44.0	23.8	24.5	31.9	32.1
PAQ + MS MARCO	26.7	57.6	53.8	63.4	30.6	37.2	44.9
$\text{TSDAE}_{\text{MS MARCO}}$	26.7	55.5	51.4	65.6	30.5	36.6	44.4
BM25	23.9	66.1	70.7	60.1	31.5	38.7	48.5
Previous Domain Adaptation Methods							
UDALM	23.3	33.6	33.1	57.1	24.6	26.3	33.0
MoDIR	29.6	50.2	47.9	66.0	29.7	–	–
Pre-Training based Domain Adaptation: Target → MS MARCO							
CT	28.3	55.6	49.9	63.8	30.5	35.9	44.0
CD	27.0	62.7	47.7	65.4	30.6	34.5	44.7
SimCSE	26.7	55.0	53.2	68.3	29.0	37.9	45.0
ICT	27.0	58.3	55.3	69.7	31.3	37.4	46.5
MLM	30.2	60.0	51.3	69.5	30.4	38.8	46.7
TSDAE	29.3	62.8	55.5	**76.1**	31.8	39.4	49.2
Generation-based Domain Adaptation (Previous State-of-the-Art)							
QGen	28.7	63.8	56.5	72.4	33.0	38.1	48.8
QGen (w/ Hard Negatives)	26.0	59.6	57.7	65.0	33.2	36.5	46.3
TSDAE + QGen (Ours)	31.4	66.7	58.1	72.6	**35.3**	37.4	50.3
Proposed Method: Generative Pseudo Labeling							
GPL	32.8	66.4	61.0	72.6	34.5	41.4	51.5
TSDAE + GPL	**34.4**	**68.9**	**61.6**	74.6	35.1	**43.0**	**52.9**
Re-Ranking with Cross-Encoders (Upper Bound, Inefficient at Inference)							
BM25 + CE	33.1	67.6	72.8	71.2	36.8	46.7	54.7
MS MARCO + CE	33.0	66.9	57.4	65.1	36.9	44.7	50.7
TSDAE + GPL + CE	36.4	68.3	68.0	71.4	38.1	48.3	55.1

Table 1: Evaluation using nDCG@10. The best results of the single-stage dense retrievers are bold. TRECC. and CQADup. are short for TREC-COVID and CQADupStack. Our proposed GPL significantly outperforms other domain adaptation methods. For the first time, we investigate the TSDAE pre-training in domain adaptation for dense retrieval and find it can significantly improve both QGen and GPL. The results on the full 18 BeIR datasets can be found in Appendix D.

text twice with different dropout masks in combination with MNRL loss. (c) **CT** (Carlsson et al., 2021) is similar to SimCSE but it uses two independent encoders to encode a pair of text. (d) **MLM** (Devlin et al., 2019) uses the default setting in original paper, where 15% tokens in a text are sampled to be masked and are needed to be predicted. (e) **ICT** (Lee et al., 2019) uniformly samples one sentence from a passage as the pseudo query to that passage and uses MNRL loss on the synthetic data. We follow the setting in Lee et al. (2019) and masked out the selected sentence 90% of the time. (f) **TSDAE** (Wang et al., 2021) uses a denoising autoencoder to pre-train the dense retrievers with 60% random tokens deleted in the input texts.

Generation-based Domain Adaptation: We use the training script[12] from Thakur et al. (2021b) to train QGen models with the default setting. Cosine similarity is used and the models are fine-tuned for 1 epoch with MNRL. The default QGen is trained with in-batch negatives. For a fair comparison, we also test QGen with hard negatives as used in GPL, noted as **QGen (w/ Hard Negatives)**. Further, We test the combination of TSDAE and QGen (**TSDAE + QGen**).

Re-Ranking with Cross-Encoders: We also include results of the powerful but inefficient re-ranking methods for reference. Three retrievers for the first-phrase retrieval are tested: BM25 from Elasticsearch, the zero-shot MS MARCO retriever and the GPL retriever enhanced by TSDAE pre-training. We use the cross-encoder *ms-marco-MiniLM-L-6-v2* from Sentence-Transformers, which is also the same model used for pseudo labeling in GPL.

5 Results

Pre-Training based Domain Adaptation:

[12] https://github.com/UKPLab/beir

The results are shown in Table 1. Compared with the zero-shot MS MARCO model, TSDAE, MLM and ICT can improve the performance if we first pre-train on the target corpus and then perform supervised training on MS MARCO. Among them, TSDAE is the most effective method, outperforming the zero-shot baseline by 4.0 points nDCG@10 on average. CD, CT and SimCSE are not able to adapt to the domains in a pre-training setup and achieve a performance worse than the zero-shot model.

To ensure that TSDAE actually learns domain specific terminology, we include $\text{TSDAE}_{\text{MS MARCO}}$ in our experiments: Here, we performed TSDAE pre-training on the MS MARCO dataset followed by supervised learning on MS MARCO. This performs slightly weaker than the zero-shot MS MARCO model.

We also tested the pre-training methods without any supervised training on MS MARCO. We find all of them fail miserably compared to the zero-shot baseline as shown in Appendix E .

Previous Domain Adaptation Methods: We test MoDIR on the datasets except Robust04[13]. MoDIR performs on-par with our zero-shot MS MARCO model on FiQA, TREC-COVID and CQADupStack, while it performs much weaker on SciFact and BioASQ. An improved training setup with MoDIR could improve the results.

We also test UDALM, which first does MLM pre-training on the target corpus, and then runs multitask learning with MLM objective and supervised training on MS MARCO. The results show that UDALM in this case greatly harms the performance by 12.2 points in average, when compared with the MLM-pre-training approach. We suppose this is because unlike text classification, the dense retrieval models usually do not have an additional task head and the direct MLM training conflicts with the supervised training.

Generation-based Domain Adaptation: The results show that the previous best method, QGen, can successfully adapt the MS MARCO models to the new domains, improving the performance on average by 3.6 points. It performs on par with TSDAE-based domain-adaptive pre-training. Combining TSDAE with QGen can further improve the performance by 1.5 points.

When using QGen with hard negatives instead of random in-batch negatives, the performance decreases by 2.5 points in average. QGen is sensitive to false negatives, i.e. negative passages that are actually relevant for the query. This is a common issue for hard negative mining. GPL solves this issue by using the cross-encoder to determine the distance between the query and a passage. We give more analysis in section 7.

Generative Pseudo Labeling (GPL, proposed method): We find GPL is significantly better on almost all the datasets compared to other tested methods, outperforming QGen by up to 4.5 points (on BioASQ) and in average by 2.7 points. One exception is TREC-COVID, but as this dataset has just 50 test queries, this difference can be due to noise.

As a further enhancement, we find that TSDAE-based domain-adaptive pre-training combined with GPL (i.e. TSDAE + GPL) can further improve the performance on all the datasets, achieving the new state-of-the-art result of 52.9 nDCG@10 points in average. It outperforms the out-of-the-box MS MARCO model 7.7 points on average.

For the results of GPL on the full 18 BeIR datasets, please refer to Appendix D. The observations remain the same.

Re-ranking with Cross-Encoders: Cross-encoders perform well in a zero-shot setting and outperform dense retrieval approaches significantly (Thakur et al., 2021b), but they come with a significant computational cost at inference. TSDAE and GPL can narrow but not fully close the performance gap between the single-stage retrievers and the re-ranking methods. Due to the much lower computational costs at inference, the TSDAE + GPL model would be preferable in a production setting. For example, as shown in Thakur et al. (2021b), the retrieval latency on a 1M-sized corpus for one query is 14ms and 450ms for dense retrievers (with the same backbone as ours) and BM25 + CE reranking, resp.

6 Analysis

In this section, we analyze the influence of training steps, corpus size, query generation and choices of starting checkpoints on GPL.

6.1 Influence of Training Steps

We first analyze the influence of the number of training steps on the model performance. We evaluate the models every 10K training steps and end

[13] The original author did not train the model on Robust04 and the code is also not available.

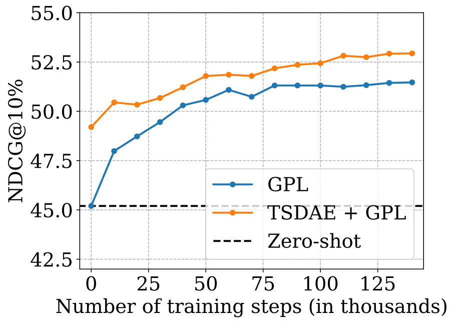

Figure 2: Influence of the number training steps on the averaged performance. The performance of GPL begins to be saturated after 100K steps. TSDAE helps improve the performance during the whole training stage.

Method \ Size	1K	10K	50K	250K	528K
QGen	35.5	36.5	38.7	37.5	37.0
GPL	37.6	41.4	42.5	41.4	41.3
Zero-shot	39.0				

Table 2: Influence of corpus size on performance on Robust04. The full size is 528K. GPL can achieve the best performance with as little as 50K passages.

the training after 140K steps. The results for the change of averaged performance on all the datasets are shown in Figure 2. We find the performance of GPL begins to be saturated after around 100K steps. With the TSDAE pre-training, the performance can be improved consistently during the whole training stage. For reference, training a distilbert-base model for 100k steps takes about 9.6 hours on a single V100 GPU.

6.2 Influence of Corpus Size

We next analyze the influence of different corpus sizes. We use Robust04 for this analysis, since it has a relatively large size. We sample 1K, 10K, 50K and 250K passages from the whole corpus independently to form small corpora and train QGen and GPL on the same small corpus. The results are shown in Table 2. We find with more than 10K passages, GPL can already significantly outperform the zero-shot baseline by 2.4 NDCG@10 points; with more than 50K passages, the performance begins to saturate. On the other hand, QGen falls behind the zero-shot baseline for each corpus size.

6.3 Robustness against Query Generation

Dataset	Method	Queries Per Passage						
		1	2	3	5	10	25	50
SciFact (5.2K)	QGen	56.7	59.6	60.2	59.9	61.5	62.2	63.7
	GPL	61.7	63.2	63.8	64.7	66.8	66.9	67.9
	Zero-shot	57.1						
FiQA (57.6K)	QGen	27.3	28.1	27.8	28.5	29.3	31.1	31.8
	GPL	31.5	32.2	32.3	32.8	33.0	33.5	33.5
	Zero-shot	26.7						
Robust04 (528.2K)	QGen	37.9	38.7	37.0	37.3	38.2	37.7	37.7
	GPL	42.0	41.3	41.4	41.2	40.9	41.2	40.6
	Zero-shot	39.0						

Table 3: Influence of number of generated Queries Per Passage (QPP) on the performance on SciFact, FiQA and Robust04. Corpus size is labeled under each dataset name. Smaller corpora, e.g. SciFact and FiQA require larger QPP to achieve the optimal performance.

Next, we study how the query generation influences the model performance. First, we train QGen and GPL on SciFact, FiQA and Robust04, with 1 up to 50 generated Queries Per Passage (QPP). The results are shown in Table 3. We observe that smaller corpora, e.g. SciFact (size = 5.2K) and FiQA (size = 57.6K) require more generated queries per passage than the large one, Robust04 (size = 528.2K). For example, GPL needs QPP equal to around 50, 5 and 1 for SciFact, FiQA and Robust04, resp. to achieve the optimal performance.

The temperature[14] plays an important role in nucleus sampling, higher values make the generated queries more diverse, but of lower quality. We train QGen and GPL on FiQA with different temperatures: 0.1, 1, 1.3, 3, 5 and 10. Examples of generated queries are available in Appendix F. We generated 3 queries per passage. The results are shown in Figure 3. We find the performance of QGen and GPL both peaks at 1.0. With a higher temperature, the next-token distribution will be flatter and more diverse queries, but of lower quality, will be generated. With high temperatures, the generated queries have nearly no relationship to the passage. QGen will perform poorly in these cases, worse than the zero-shot model. In contrast, GPL performs still well even when the generated queries are of such low quality.

6.4 Sensitivity to Starting Checkpoints

We also analyze the influence of initialization on GPL. In the default setting, we start from a distilbert-model supervised on MS MARCO using MarginMSE loss. We also evaluate to directly fine-tune a distilbert-model using QGen, GPL and TSDAE + GPL. The performance averaged on all the datasets is shown in Table 4. We find the MS MARCO training has relatively small effect on the performance of GPL (with 0.3-point difference in average), while QGen highly relies on the choice

[14] The amplifying coefficient to the raw logits in the Softmax function.

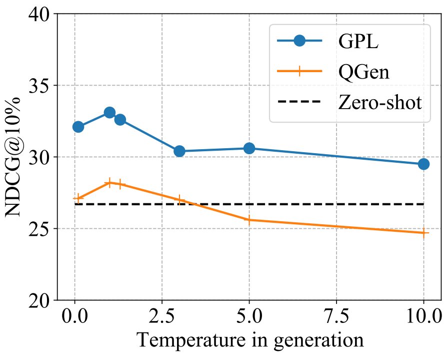

Figure 3: Influence of the temperature in generation on the performance on FiQA. A higher temperature means more diverse queries but of lower quality. GPL can still yield around 3.0-point improvement over the zero-shot baseline with high temperature value of 10.0, where the generated queries have nearly no connection to the passages.

Init. **Method**	**Distilbert**	**MS MARCO**
QGen	46.9	48.8
TSDAE + QGen (Ours)	49.6	50.3
GPL	51.2	51.5
TSDAE + GPL	52.3	52.9
Zero-shot	–	45.2

Table 4: Influence of initialization checkpoint on performance in average. GPL yields similar performance when starting from different checkpoints.

of the initialization checkpoint (with 1.9-point difference in average).

7 Case Study: Fine-Grained Labels

GPL uses continuous pseudo labels from a cross-encoder, which can provide more fine-grained information and is more informative than the simple 0-1 labels as in QGen. In this section, we give a more detailed insight into it by a case study.

One example from FiQA is shown in Table 5. The generated query for the positive passage asks for the definition of “futures contract”. Negative 1 and 2 only mention futures contract without explaining the term (with low GPL labels/scores below 2.0), while Negative 3 gives the required definition (with high GPL label/score 8.2). As an interesting case, Negative 4 gives a partial explanation of the term (with medium GPL label/score 6.9). GPL assigns suitable fine-grained labels to different negative passages. In contrast, QGen simply labels all of them as 0, i.e. as irrelevant. Such difference explains the advantage of GPL over QGen and why using hard negatives harms the performance of QGen in Table 1.

Item	**Text**	**GPL**	**QGen**
Query	what is **futures contract**	–	–
Positive	**Futures contracts** are a member of a larger class of financial assets called derivatives ...	10.3	1
Negative 1	... Anyway in this one example the s&p 500 **futures contract** has an "initial margin" of $19,250, meaning ...	2.0	0
Negative 2	... but the moment you exercise you must have $5,940 in a margin account to actually use the **futures contract** ...	0.3	0
Negative 3	... a **futures contract** is simply a contract that requires party A to buy a given amount of a commodity from party B at a specified price...	8.2	0
Negative 4	... A **futures contract** commits two parties to a buy/sell of the underlying securities, but ...	6.9	0

Table 5: Examples of the labels assigned to different query-passage pairs in FiQA by GPL and QGen. The key term "futures contract" is marked in bold. QGen uses only 0-1 scores. GPL uses raw logits, which can be any value between positive and negative infinity (e.g. $[-12, 11]$ is a typical range).

8 Conclusion

In this work we propose GPL, a novel unsupervised domain adaptation method for dense retrieval models. It generates queries for a target corpus and pseudo labels these with a cross-encoders. Pseudo-labeling overcomes two important short-comings of previous methods: Not all generated queries are of high quality and pseudo-labels efficiently detects those. Further, training with mined hard negatives is possible as the pseudo labels performs efficient denoising.

In this work, we also evaluated different pre-training strategies in a domain-adaptive pre-training setup: We first pre-trained on the target domain, then performed supervised training on MS MARCO. ICT and MLM were able to yield a small improvement (by <=1.5 nDCG@10 points on average), while TSDAE was able to yield a significant improvement of 4 nDCG@10 points on average. Other approaches degraded the performance.

Acknowledgments

This work has been funded by the German Research Foundation (DFG) as part of the UKP-SQuARE project (grant GU 798/29-1).

References

Adam L. Berger, Rich Caruana, David Cohn, Dayne Freitag, and Vibhu O. Mittal. 2000. Bridging the lexical chasm: statistical approaches to answer-finding. In *SIGIR 2000: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 24-28, 2000, Athens, Greece*, pages 192–199. ACM.

Fredrik Carlsson, Amaru Cuba Gyllensten, Evangelia Gogoulou, Erik Ylipää Hellqvist, and Magnus Sahlgren. 2021. Semantic re-tuning with contrastive tension. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Jason Ingyu Choi, Surya Kallumadi, Bhaskar Mitra, Eugene Agichtein, and Faizan Javed. 2020. Semantic product search for matching structured product catalogs in e-commerce. *ArXiv preprint*, abs/2008.08180.

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.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. 2016. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17:59:1–59:35.

Luyu Gao and Jamie Callan. 2021. Condenser: a pre-training architecture for dense retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 981–993, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021a. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021b. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Matthew L. Henderson, Rami Al-Rfou, Brian Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply. *ArXiv preprint*, abs/1705.00652.

Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. 2020. Improving efficient neural ranking models with cross-architecture knowledge distillation. *ArXiv preprint*, abs/2010.02666.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 113–122. ACM.

Doris Hoogeveen, Karin M Verspoor, and Timothy Baldwin. 2015. Cqadupstack: A benchmark data set for community question-answering research. In *Proceedings of the 20th Australasian document computing symposium*, pages 1–8.

Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. Embedding-based retrieval in facebook search. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 2553–2561. ACM.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Constantinos Karouzos, Georgios Paraskevopoulos, and Alexandros Potamianos. 2021. UDALM: Unsupervised domain adaptation through language modeling. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2579–2590, Online. Association for Computational Linguistics.

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.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. PAQ: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115.

Davis Liang, Peng Xu, Siamak Shakeri, Cicero Nogueira dos Santos, Ramesh Nallapati, Zhiheng Huang, and Bing Xiang. 2020. Embedding-based zero-shot retrieval through query generation. *ArXiv preprint*, abs/2009.10270.

Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021. Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. 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.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: Financial opinion mining and question answering. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, page 1941–1942, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with BERT. *ArXiv preprint*, abs/1901.04085.

Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019a. From doc2query to doctttttquery. *Online preprint*.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019b. Document expansion by query prediction. *ArXiv preprint*, abs/1904.08375.

Barlas Oguz, Kushal Lakhotia, Anchit Gupta, Patrick S. H. Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Wen-tau Yih, Sonal Gupta, and Yashar Mehdad. 2021. Domain-matched pre-training tasks for dense retrieval. *ArXiv preprint*, abs/2107.13602.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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.

Kirk Roberts, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, Kyle Lo, Ian Soboroff, Ellen Voorhees, Lucy Lu Wang, and William R Hersh. 2020. TREC-COVID: rationale and structure of an information retrieval shared task for COVID-19. *Journal of the American Medical Informatics Association*, 27(9):1431–1436.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*, volume 500-225 of *NIST Special Publication*, pages 109–126. National Institute of Standards and Technology (NIST).

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv preprint*, abs/1910.01108.

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 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)*.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *ArXiv preprint*, abs/1807.03748.

Ellen Voorhees. 2005. Overview of the trec 2004 robust retrieval track. Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg, MD.

Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R. Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. 2021. TREC-COVID: Constructing a pandemic information retrieval test collection. *SIGIR Forum*, 54(1).

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

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.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Michael Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Nancy Xin Ru Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. CORD-19: The COVID-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul N. Bennett. 2021. Zero-shot dense retrieval with momentum adversarial domain invariant representations. *ArXiv preprint*, abs/2110.07581.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

A Performance of Using Different Retrievers for Negative Mining in GPL

The performance of using different retrievers (BM25, dense, BM25 + dense and single dense retrievers) for mining hard negatives in GPL is shown in Table 6. The results show GPL performs best when using hard negatives mined by both the two dense retrievers.

Dataset Method	FiQA	SciFact	BioASQ	TRECC.	CQADup.	Robust04	Avg.
GPL (w/ BM25 + dense)	32.9	64.4	61.1	68.6	33.8	41.3	50.4
GPL (w/ BM25)	31.1	60.9	57.8	67.5	33.5	35.9	47.8
GPL (w/ dense)	32.8	66.4	61.0	72.6	34.5	41.4	51.5
GPL (w/ msmarco-distilbert-base-v3)	32.1	64.7	60.9	70.8	34.3	41.5	50.7
GPL (w/ msmarco-MiniLM-L-6-v3)	32.7	64.6	61.7	69.7	35	40.4	50.7
MS MARCO	26.7	57.1	52.9	66.1	29.6	39.0	45.2

Table 6: Performance (nDCG@10) of using different retrievers for hard-negative mining in GPL. The scores of the baseline MS MARCO and the scores of GPL with dense retrievers are copied from Table 1. "Dense" represents using both of the two dense retrievers *msmarco-distilbert-base-v3* and *msmarco-MiniLM-L-6-v3*.

B Performance of the Zero-Shot Retrievers in Hard-Negative Mining

The performance of directly using the zero-shot retrievers for hard-negative mining in GPL is shown in Table 7. Compared with the strong baseline (MS MARCO in Table 7) trained with MarginMSE, *msmarco-distilbert-base-v3* and *msmarco-MiniLM-L-6-v3* are much worse in terms of zero-shot generalization on each dataset. This comparison supports GPL can indeed train powerful domain-adapted dense retrievers with minimum reliance on choices of the retrievers for hard-negative mining.

Dataset Method	FiQA	SciFact	BioASQ	TRECC.	CQADup.	Robust04	Avg.
msmarco-distilbert-base-v3	24.0	52.3	45.6	61.1	24.3	30.6	39.7
msmarco-MiniLM-L-6-v3	23.3	48.8	41.9	57.9	24.3	28.5	37.5
MS MARCO	26.7	57.1	52.9	66.1	29.6	39.0	45.2

Table 7: Performance (nDCG@10) of different zero-shot retrievers. *msmarco-distilbert-base-v3* and *msmarco-MiniLM-L-6-v3* are used in GPL for hard-negative mining. The scores of the baseline MS MARCO are copied from Table 1.

C Target Datasets

FiQA is for the task of opinion question answering over financial data. It contains 648 queries and 5.8K passages from StackExchange posts under the Investment topic in the period between 2009 and 2017. The labels are binary (relevant or irrelevant) and there are 2.6 passages in average labeled as relevant for each query.

SciFact is for the task of verifying scientific claims using evidence from the abstracts of the scientific papers. It contains 300 queries and 5.2K passages built from S2ORC (Lo et al., 2020), a publicly-available corpus of millions of scientific articles. The labels are binary and there are 1.1 passages in average labeled as relevant for each query.

BioASQ is for the task of biomedical question answering. It originally contains 500 queries and 15M articles from PubMed[15]. The labels are binary and it has 4.7 passages in average labeled as relevant for each query. For efficient training and evaluation, we randomly remove irrelevant passages to make the final corpus size to 1M.

TREC-COVID is an ad-hoc search challenge for scientific articles related to COVID-19 based on the CORD-19 dataset (Wang et al., 2020). It originally contains 50 queries and 171K documents. The original corpus has many documents with only a title and an empty body. We remove such documents and the

[15] https://pubmed.ncbi.nlm.nih.gov/

final corpus size is 129.2K. The labels in TREC-COVID are 3-level (i.e. 0, 1 and 2) and there are 430.8 passages in average labeled as 1 or 2 in the clean-up version.

CQADupStack is a dataset for community question-answering, built from 12 StackExchange subforums: Android, English, Gaming, Gis, Mathematica, Physics, Programmers, Stats, Tex, Unix, Webmasters and WordPress. The task is to retrieve duplicate question posts with both a title and a body text given a post title. It has 13.1K queries and 457.2k passages. The labels are binary and there are 1.4 passages in average labeled as relevant for each query. As in Thakur et al. (2021b), the average score of the 12 sub-tasks is reported.

Robust04 is a dataset for news retrieval focusing on poorly performing topics. It has 249 queries and 528.2K passages. The labels are 3-level and there are in average 69.9 passages labeled as relevant for each query.

The detailed statistics of these target datasets are shown in Table 8.

Statistics Dataset	Domain	Title	Relevancy	#Queries	#Passages	PPQ	Query Len.	Passage Len.
FiQA	Financial	✗	Binary	648	57.6K	2.6	10.8	132.2
SciFact	Scientific	✓	Binary	300	5.2K	1.1	12.4	213.6
BioASQ	Bio-Medical	✓	Binary	500	1.0M	4.7	8.1	204.1
BioASQ*	Bio-Medical	✓	Binary	500	14.9M	4.7	8.1	202.6
TREC-COVID	Bio-Medical	✓	3-Level	50	129.2K	430.8	10.6	210.3
TREC-COVID*	Bio-Medical	✓	3-Level	50	171.3K	493.5	10.6	160.8
CQADupStack	Forum	✓	Binary	13,145	457.2K	1.4	8.6	129.1
Robust04	News	✗	3-Level	249	528.2K	69.9	15.3	466.4

Table 8: Statistics of the target datasets used in the experiments. Column **Title** indicates whether there is (✓) a title for each passage or not (✗). Column **PPQ** represents number of Passages Per Query. Query/passage lengths are counted in words. Symbol $*$ marks the original version from the BeIR benchmark (Thakur et al., 2021b)

We also evaluate the models trained in this work on the original version of BioASQ and TREC-COVID datasets from BeIR (Thakur et al., 2021b). The results are shown in Table 9.

D Results on full BeIR

We also evaluate the models on all the 18 BeIR datasets. We include DocT5Query (Nogueira et al., 2019a), the strong baseline based on document expansion with the T5 query generator (also used in GPL for query generation) + BM25 (Anserini). We also include the powerful zero-shot model **TAS-B** (Hofstätter et al., 2021), which is trained on MS MARCO with advanced knowledge-distillation techniques into comparison. Viewing TAS-B as the base model and also the negative miner, we apply QGen and GPL on top of them, resulting in **TAS-B + QGen** and **TAS-B + GPL**, resp.

The results are shown in Table 9. We find both DocT5Query and BM25 (Anserini) outperform MS MARCO, TSDAE and QGen, in terms of both average performance and average (performance) rank. QGen struggles to beat MS MARCO, the zero-shot baseline and it even significantly harms the performance on many datasets, e.g. TREC-COVID, FEVER, HotpotQA, NQ. Thakur et al. (2021b) also observes the same issue, claiming that the bad generation quality on these corpora is the key to the failure of QGen. On the other hand, GPL significantly outperforms these baselines above, achieving average performance rank 5.2 and can consistently improve the performance over the zero-shot model on all the datasets. For TSDAE, TSDAE + QGen and TSDAE + GPL, the conclusion remains the same as in the main paper.

For the powerful zero-shot model TAS-B, it outperforms QGen and performs on par with TSDAE + QGen. When building on top of TAS-B, GPL can also yield significant performance gain by up-to 21.5 nDCG@10 points (on TREC-COVID) and 4.6 nDCG@10 points on average. This TAS-B + GPL model performs the best over all these retriever models, achieving the averaged performance rank equal to 3.2. However, when applying QGen on top of TAS-B, it cannot improve the overall performance but also harms the individual performance on many datasets, instead.

Method Dataset	BM25 (Anserini)	DocT5-Query	MS MARCO	TSDAE	QGen	TSDAE + QGen (Ours)	GPL	TSDAE + GPL	TAS-B	TAS-B + QGen	TAS-B + GPL	BM25 + CE (Upperbound)
FiQA	23.6	29.1	26.7	29.3	28.7	31.4	32.8	**34.4**	29.8	30.1	**34.4**	34.7
SciFact	66.5	67.5	57.1	62.8	63.8	66.7	66.4	**68.9**	63.5	65.3	67.4	68.8
BioASQ*	**46.5**	43.1	33.6	37.3	36.9	38.5	41.2	40.9	36.2	38.5	44.2	52.3
TRECC.*	65.6	71.3	66.1	70.8	56.0	58.4	71.8	**74.9**	48.5	56.6	70.0	75.7
CQADup.	29.9	32.5	29.6	31.8	33.0	35.3	34.5	35.1	31.5	33.7	**35.7**	37.0
Robust04	40.8	**43.7**	39.0	39.4	38.1	37.4	41.4	43.0	42.4	39.4	**43.7**	47.5
ArguAna	41.4†	46.9†	33.9	37.5	52.4	54.7	48.3	51.2	43.4	51.8	**55.7**	41.7†
Climate-F.	21.3	20.1	20.0	16.8	22.5	22.6	22.7	22.2	22.1	**24.4**	23.5	25.3
DBPedia	31.3	33.1	34.2	35.4	33.1	33.2	36.1	36.1	**38.4**	32.7	**38.4**	40.9
FEVER	75.3	71.4	76.5	64.0	63.8	64.2	**77.9**	78.6	69.5	63.9	75.9	81.9
HotpotQA	60.3	58.0	55.4	**63.8**	51.4	52.2	56.5	57.2	58.4	52.0	58.2	70.7
NFCorpus	32.5	32.8	27.7	31.2	31.4	33.7	34.2	33.9	31.9	33.4	**34.5**	35.0
NQ	32.9	39.9	45.6	47.1	35.4	34.6	46.7	47.1	46.3	36.3	**48.3**	53.3
Quora	78.9	80.2	81.2	83.3	85.0	**85.7**	83.2	83.1	83.5	85.3	83.6	82.5
SciDocs	15.8	16.2	13.6	15.4	15.5	**17.1**	16.1	16.8	14.9	16.4	16.9	16.6
Signal-1M	**33.0**	30.7	24.4	25.9	26.8	26.8	26.5	27.6	28.9	26.6	27.6	33.8
TRECN.	39.8	42.0	36.0	35.0	36.0	38.3	40.7	41.5	37.7	38.0	**42.1**	43.1
Touché20	**36.7**	34.7	19.6	21.8	17.1	17.2	23.1	23.5	16.2	17.5	25.5	27.1
Avg.	42.9	44.1	40.0	41.6	40.4	41.6	44.5	45.3	41.3	41.2	**45.9**	48.2
Avg. Rank	7.6	6.2	9.8	8.2	8.9	6.5	5.2	4.2	7.8	7.3	**3.2**	2.4

Table 9: Performance (nDCG@10) on all the original 18 BeIR datasets. The results of MS MARCO, TSDAE, QGen, TSDAE + QGen, GPL and TSDAE + GPL on FiQA, SciFact, CQADupStack and Robust04 are copied from Table 1. The results of BM25, DocT5Query and BM25 + CE come from Thakur et al. (2021b). † marks correction over the original scores, where identical IDs between queries and passages are removed. TRECN. is short for TREC-NEWS. Avg. Rank is the average over the rank of the performance on each dataset over the different models (the lower, the better).

E Performance of Unsupervised Pre-Training

The performance of the unsupervised pre-training methods without access to the MS MARCO data is shown in Table 10. We find ICT is the best method, achieving highest scores on all the datasets. However, all the unsupervised pre-training methods cannot directly yield improvement in performance compared with the zero-shot baseline.

Dataset Method	FiQA	SciFact	BioASQ	TRECC.	CQADup.	Robust04	Avg.
CD	6.6	0.6	0.3	9.8	8.1	3.8	4.9
CT	0.2	0.7	0.0	2.5	0.9	0.0	0.7
MLM	5.4	27.8	4.7	16.0	8.5	6.1	11.4
TSDAE	7.8	37.2	6.9	9.4	14.3	10.1	14.3
SimCSE	5.5	25.0	13.1	26.0	14.6	9.8	15.7
ICT	10.2	42.6	39.0	47.5	23.0	16.5	29.8
MS MARCO	26.7	57.1	52.9	66.1	29.6	39.0	45.2

Table 10: Performance (nDCG@10) of unsupervised pre-training methods with only access to the target corpus as the training data. The scores of the zero-shot baseline MS MARCO are copied from Table 1.

F Examples of Generated Queries under Different Temperatures

The generation temperature controls the sharpness of the next-token distribution. The examples for one passage from FiQA are shown in Table 11 Higher temperature results in longer and less duplicate queries under more risk of generating non-sense texts.

Item	**Text**	**Pseudo Label**
Input Passage	You can never use a health FSA for individual health insurance premiums. Moreover, FSA plan sponsors can limit what they are will to reimburse. While you can't use a health FSA for premiums, you could previously use a 125 cafeteria plan to pay premiums, but it had to be a separate election from the health FSA. However, under N. 2013-54, even using a cafeteria plan to pay for indivdiual premiums is effectively prohibited.	–
Temperature 0.1	can you use a cafeteria plan for premiums	9.1
	can you use a cafeteria plan for premiums	9.1
	can you use a cafeteria plan for premiums	9.1
Temperature 1.0	can i use my fsa to pay for a health plan	9.7
	can i use my health fsa for an individual health plan?	9.9
	can fsa pay premiums	9.2
Temperature 3.0	cafe a number cafe plan is used by	-10.5
	what type of benefits do the health savings accounts cover when applying for medical terms health insurance	-7.2
	why can't an individual file medical premium on their insurance account with an fsa plan instead of healthcare policy.	6.0
Temperature 5.0	which one does not apply after an emergency medical	-11.1
	is medicare cafe used exclusively as plan funds (health savings account	-7.2
	how soon to transfer coffee bean fses to healthcare	-11.0
Temperature 10.0	will employer limit premiums reimbursement on healthcare expenses with caeatla cafetaril and capetarians account on my employer ca. plans and deductible accounts a.f,haaq and asfrhnta,	-2.5
	kfi what is allowed as personal health account or ca	-10.2
	do people put funds back to buy plan plans before claiming an deductible without the provider or insurance cover f/f associator funds of the person you elect? healthfin depto of benefit benefits deduct all oe premiumto payer for individual care	-4.5

Table 11: Examples of generated queries under different temperature value for a passage from FiQA.