BERM: Training the Balanced and Extractable Representation for Matching to Improve Generalization Ability of Dense Retrieval

Shicheng Xu[1,2], **Liang Pang**[1]*, **Huawei Shen**[1,2], **Xueqi Cheng**[1,2]*
[1]Data Intelligence System Research Center,
Institute of Computing Technology, CAS
[2]University of Chinese Academy of Sciences
xschit@163.com {pangliang,shenhuawei,cxq}@ict.ac.cn

Abstract

Dense retrieval has shown promise in the first-stage retrieval process when trained on in-domain labeled datasets. However, previous studies have found that dense retrieval is hard to generalize to unseen domains due to its weak modeling of domain-invariant and interpretable feature (i.e., matching signal between two texts, which is the essence of information retrieval). In this paper, we propose a novel method to improve the generalization of dense retrieval via capturing matching signal called BERM. Fully fine-grained expression and query-oriented saliency are two properties of the matching signal. Thus, in BERM, a single passage is segmented into multiple units and two unit-level requirements are proposed for representation as the constraint in training to obtain the effective matching signal. One is *semantic unit balance* and the other is *essential matching unit extractability*. Unit-level view and balanced semantics make representation express the text in a fine-grained manner. Essential matching unit extractability makes passage representation sensitive to the given query to extract the pure matching information from the passage containing complex context. Experiments on BEIR show that our method can be effectively combined with different dense retrieval training methods (vanilla, hard negatives mining and knowledge distillation) to improve its generalization ability without any additional inference overhead and target domain data.

1 Introduction

Dense retrieval encodes the texts to dense embeddings and efficiently gets the target texts via approximate nearest neighbor search (Johnson et al., 2021). Compared with the traditional word-to-word exact matching methods such as BM25 (Robertson et al., 1995), dense retrieval can capture the relevance at the semantic level of two texts. Because of the excellent performance in efficiency and effectiveness, dense retrieval has been widely used in first-stage retrieval that efficiently recalls candidate documents from the large corpus (Karpukhin et al., 2020; Xiong et al., 2021a).

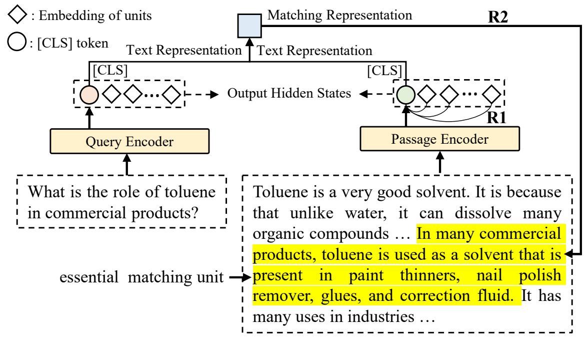


Figure 1: Idea of our method. **R1**: semantic unit balance. **R2**: essential matching unit extractability.

However, recent studies show that the excellent performance of dense retrieval relies on the training on large in-domain datasets. When the trained dense retrieval models are applied to the domains that are inconsistent with the training datasets (i.e., zero-shot setting), the performance of the models drops seriously (Ren et al., 2022; Thakur et al., 2021). The poor generalization limits the application scenarios of dense retrieval because it is common that not enough training samples can be obtained in some domains such as medicine, biology and law that have restrictions on data privacy or require professional knowledge to annotate.

In this work, we point out that according to out-of-domain generalization learning theory (Ye et al., 2021), making the model capture domain-invariant feature (i.e., essence of tasks) is effective in improving generalization ability. As for dense retrieval, matching signal between query and passage is the important domain-invariant feature and reflects the essence of information retrieval (IR). For example, MoDIR (Xin et al., 2022) shows that representation from the interaction-based cross-encoder

*Corresponding authors

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 6620–6635

(more fine-grained description for matching) is much more domain-invariant than it from dense retrieval. Match-Prompt (Xu et al., 2022a), NIR-Prompt (Xu et al., 2022b) and MatchPyramid (Pang et al., 2016) point out the positive significance of matching signals for various IR tasks. The challenge of making dense retrieval model learn to capture matching signal is that in many IR tasks such as open-domain question answering (Chen et al., 2017) and document retrieval (Mitra et al., 2017), the content that matches the query is usually only a unit of the text. The description of matching signal needs to distinguish the matching and not matching information in the text and estimate the overall relevance. This requires the retrieval model to be able to evenly express each unit in the text and dynamically extract matching units through the interaction of the two text representations. However, the requirement on efficiency in first-stage retrieval makes dense retrieval only estimate relevance via vector similarity such as dot product and cosine. Previous training methods based on this architecture lack the above capability because of the coarse-grained training objective and interaction.

In this paper, we propose a novel method called BERM to capture the matching signal between query and passage, which is the domain-invariant feature, to improve the generalization ability of dense retrieval during the training on the single source domain without using the target domain data and other additional modules. First, we introduce a novel concept in dense retrieval, the matching representation. Matching representation is determined by the text representations (output of text encoder) of query and passage, which can reflect the matching information of query and passage. We propose that in the training of dense retrieval models, in addition to using contrastive loss (Xiong et al., 2021b) to optimize the text representation, the information of the matching representation can be used as a constraint to assist the optimization. Based on this, we divide the single passage into multiple units (each sentence is a unit) and propose two requirements on the generalizable dense retrieval models as the constraint in training (shown in Figure 1). One is *semantic unit balance of text representation* (**R1**). The other is *essential matching unit extractability of matching representation* (**R2**). These two requirements can be integrated into different dense retrieval training methods and address the challenge mentioned above. **R1** means the semantics of units in a passage are implicitly aggregated to its text representation and the text representation should evenly and comprehensively express the semantics of each unit. **R2** means that the combination of text representations of query and passage (i.e., matching representation) should extract the information of the matching (i.e, the text chunk in the passage that matches the query and we call it **essential matching unit**) while reducing the overfitting of domain biases. This reflects the ability of the dense retrieval model to determine and score the information that really matches the query in a passage containing complex context, which is the essence of the dense retrieval and domain-invariant. **R1** and **R2** achieve that on the premise that the text representation expresses each unit in a balanced manner, to make essential matching units for different queries be extracted, the semantics of units tend to be orthogonal to each other. In this way, in dot product between representations of query and passage, the semantics of essential matching unit are preserved, while the other units are masked, which is suitable for matching.

Experiments on the standard zero-shot retrieval benchmark (BEIR) show that our method can be effectively combined with different dense retrieval training methods (vanilla, hard negatives mining, and knowledge distillation) to improve the generalization ability without any additional modules, inference overhead, and target domain data. Even in domain adaptation, our method is also effective and performs better than baselines. Code is released at `https://github.com/xsc1234/BERM`.

2 Related Work

Dense retrieval estimates the relevance via representations of two texts. DPR (Karpukhin et al., 2020) combines dense retrieval with pre-trained models for open-domain question answering (Chen et al., 2017). Besides, some methods focus on obtaining more valuable negatives (Qu et al., 2021; Xiong et al., 2021a; Zhan et al., 2021). Some methods use a more powerful reranker for knowledge distillation (Hofstätter et al., 2021; Lin et al., 2021). Recently, the generalization of dense retrieval has received attention. (Ren et al., 2022) performs the examination of the generalization of dense retrieval. BEIR (Thakur et al., 2021) is proposed as the benchmark to evaluate the zero-shot ability of information retrieval models. MoDIR (Xin et al., 2022) uses the data from source and target domains for adversarial training to perform unsupervised domain adaptation. GenQ (Ma et al., 2021) and GPL (Wang et al., 2022) generate queries and pseudo labels for domain adaptation. Contriever (Izacard et al., 2021) uses contrastive pre-training on large corpus (Wikipedia and CC-Net (Wenzek et al., 2020)). COCO-DR (Yu et al., 2022) performs unsupervised pre-training on target domain and introduces distributional robust optimization. GTR (Ni et al., 2021) scales up the model size to improve the generalization. (Huebscher et al., 2022; Formal et al., 2022) introduce sparse retrieval to achieve better generalization.

Improvement of generalization of dense retrieval in previous studies comes from the adaptation of the target domain, knowledge from large pretraining corpus, and assistance of sparse retrieval but not dense retrieval itself. They need to obtain the target domain data in the training or increase the complexity of the system. In this paper, we introduce a novel method to improve the generalization of dense retrieval without target domain data and additional modules via learning the generalizable representation for matching.

One thing must be emphasized that the methods of multi-view dense retrieval (Zhang et al., 2022; Hong et al., 2022) also divide a passage into multiple units, but our method is essentially a completely different method. Multi-view dense retrieval uses multiple representations to fully express a passage from multiple views, which focuses on in-domain retrieval. Our method uses multiple units to make the model learn to extract essential matching unit from the passage containing complex context, which is domain-invariant for generalization. In our method, multiple units are only used as the constraint for optimization in training and only a single representation is used in inference. Learning-based sparse retrieval such as COIL (Gao et al., 2021) and SPLADE (Formal et al., 2021) also aim to express fine-grained token-level semantics but they need multiple vectors to represent tokens in passage (COIL) or sparse-vector of vocabulary size (SPLADE) and calculates the score by token-to-token matching, which is not suitable for dense retrieval that uses single dense vector to perform representation and dot product.

3 Motivation

Dense retrieval is hard to generalize to unseen domains due to its weak modeling of domain-invariant feature (i.e., matching signal between two texts, which is the essence of information retrieval). Fully fine-grained expression (**P1**) and query-oriented saliency (**P2**) are two properties of the matching signal. These two require the passage representation to be able to evenly express each unit in the text, and dynamically extract matching units according to the interaction with different queries. For example, BM25 uses one-hot to evenly express each word of the text, only scores matching words, and ignores not matching words through word-to-word exact matching of the two texts. Cross-encoder uses word embedding to represent the semantics of each token and uses attention to describe the token-to-token semantic matching between texts in a fine-grained manner.

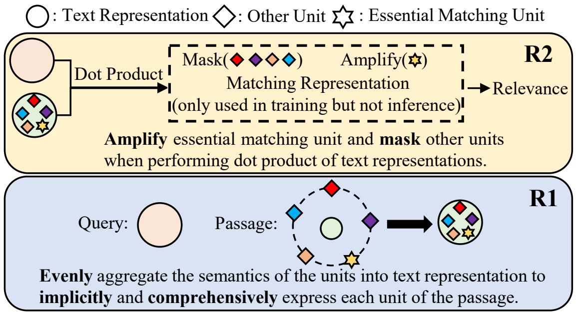


Figure 2: An illustration of the effects of **R1** and **R2**.

In this paper, based on the above two properties, for the training of dense retrieval, we segment a single passage into multiple units and propose two requirements as the constraint in training so that dense retrieval can capture the stronger matching signal and produces a suitable representation for matching. One is *semantic unit balance of text representation* (**R1**), and the other is *essential matching unit extractability of matching representation* (**R2**). Under **R1**, text representation evenly aggregates semantics of the units in the passage to comprehensively express the passage in a fine-grained manner. Besides, **R1** is the premise of **R2**. It is because that matching representation is composed of text representations from passage and query. Unbalanced semantic expression of different units in text representation will affect the identification of essential matching unit in matching representation because it leads to different preferences for different units. Under **R2**, essential matching unit for the query can be extracted from the passage and reflected in matching representation. Unlike using one-hot or word embedding to explicitly express the semantics of each unit and extract matching

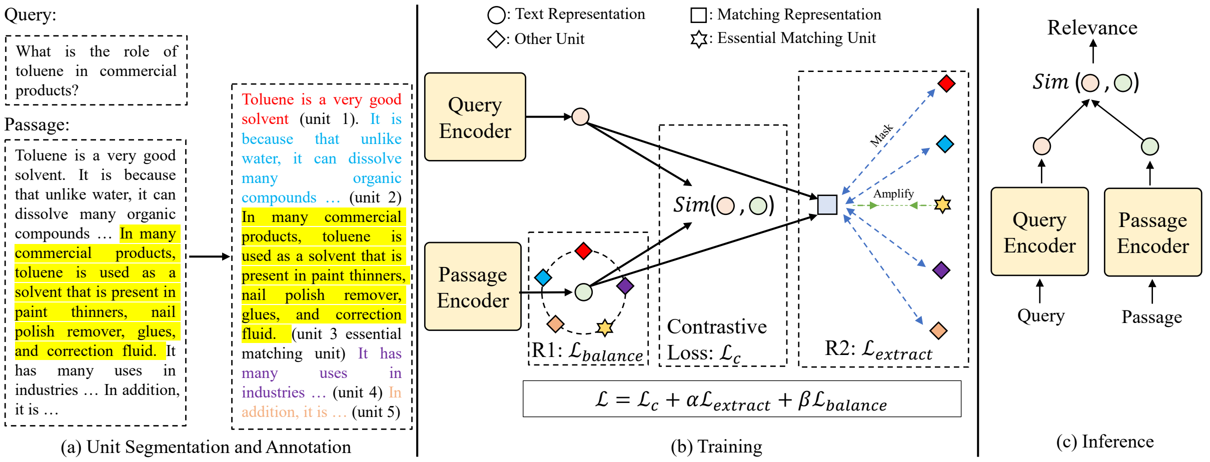


Figure 3: Overview of BERM. (a) Unit Segmentation and Annotation: Segment passage into multiple units (each sentence is a unit) and use the annotator to identify essential matching unit. (b) Training: Training models under the constraints of two requirements on the generalizable dense retrieval for matching. (c) Inference: Same as the mainstream dense retrieval, without introducing additional inference and storage overhead.

information through token-to-token interaction, as shown in Figure 2, **R1** makes the model implicitly aggregate the semantics of each unit into the text representation to satisfy **P1**, and **R2** makes the semantics of units tend to be orthogonal to each other (shown in Table 6). In dot product between representations of query and passage, semantics of essential matching unit are preserved, while the other units are masked, which can satisfy **P2**. Our method unlocks the ability of dense retrieval to capture matching signal without additional interaction.

4 Our Method

This section introduces the implementation of our method (Figure 3). Our method optimizes the relationship between the representations and the units in the passage. Therefore, before training, we perform unit segmentation and annotate the essential matching unit for the datasets. Then, we design loss functions according to the requirements of **R1** and **R2** and combine these functions with task loss of dense retrieval (contrastive loss) for joint training.

4.1 Unit Segmentation and Annotation

Given each positive query-passage pair (q, p_{pos}) in training data, we segment positive passage into multiple units U as shown in Figure 3 (a) (We use the sentence as the segmentation granularity to ensure that each unit has complete semantic information.):

$$p_{pos} \overset{Segment}{\longrightarrow} U = \{u_1, u_2, ..., u_n\}. \qquad (1)$$

For U and q, BM25 is used to compute the word-to-word matching score S_{bm25} between q and $u_i \in U$:

$$S_{bm25} = \{bm25(q, u_1), ..., bm25(q, u_n)\}.$$

For the datasets for question-answering, a trained reader model is additionally introduced to compute the semantic matching score S_{reader} between q and $u_i \in U$. Specifically, reader model computes the probability distribution $A = \{a_1, a_2, .., a_t\}$ of the starting positions of the answer in p_{pos}. a_i indicates the probability that the i-th token in p_{pos} is the starting of the answer to q. For each $u_i \in U$, the semantic matching score from the reader model is:

$$r_i = max(A[s_{u_i} : d_{u_i}]), \qquad (2)$$

where $[s_{u_i} : d_{u_i}]$ are the indexes of tokens in u_i. The hybrid matching score h_i between u_i and q is:

$$h_i = bm25(q, u_i) + \delta r_i,$$

where δ is a hyperparameter. We set δ as 0.1 to give BM25 a higher weight than the reader. It is because the word-to-word exact matching of BM25 is more domain-invariant and conducive to generalization than the semantic matching of reader (Thakur et al., 2021). Then we get matching score list $H = \{h_1, h_2, ..., h_n\}$ for U. The essential matching unit is the unit corresponding to the maximum value in H. For the pair (q, p_{pos}), y_i in label list $Y = \{y_1, ..., y_n\}$ for essential matching unit is that if i is the index corresponding to the maximum value in H, $y_i = 1$, otherwise, $y_i = 0$.

4.2 Training for Generalization

Based on the analysis of properties of matching signal in Section 3, we propose two requirements as the constraints in the training of dense retrieval to get a generalizable representation for matching (shown in Figure 3 (b)). These two requirements enable dense retrieval to extract essential matching information under the premise of balanced expression of each unit, so as to learn domain-invariant feature (i.e., matching signal) for generalization.

Implementation of R1. The first requirement is *semantic unit balance of text representation*, which means that the text representation of the passage encoder can comprehensively express the semantics of each unit in a balanced manner. Given the passage p_{pos}, the text encoder $g(\cdot;\theta)$, output hidden states $\boldsymbol{Z} = g(p_{pos};\theta)$. Text representation $\boldsymbol{t_p}$ of p_{pos} is the embedding of [CLS] token of $\boldsymbol{Z}$. The embeddings $\boldsymbol{E}$ of units in p_{pos} can be obtained from $\boldsymbol{Z}$ as the segmentation in Equ.(1):

$$\boldsymbol{E} = \{\boldsymbol{e_1}, \boldsymbol{e_2}, ..., \boldsymbol{e_n}\}, \tag{3}$$

where $\boldsymbol{e_i}$ is the embedding of the corresponding unit (u_i) and it is the average pooling of the embeddings of tokens $(\boldsymbol{Z}[s_{u_i} : d_{u_i}])$ in the unit, where $[s_{u_i} : d_{u_i}]$ are the indexes of tokens in u_i. Under the constraint of **R1**, the relationship between $\boldsymbol{t_p}$ and $\boldsymbol{E}$ is described by the loss function as:

$$\mathcal{L}_{balance} = D_{KL}[\boldsymbol{b}||sim(\boldsymbol{t_p}, \boldsymbol{E})], \tag{4}$$

where $D_{KL}[\cdot||\cdot]$ is KL-divergence loss, $\boldsymbol{b} = [\frac{1}{n}, ..., \frac{1}{n}]$ is a uniform distribution with equal values and $sim(\boldsymbol{t_p}, \boldsymbol{E}) = \{dot(\boldsymbol{t_p}, \boldsymbol{e_i})|\boldsymbol{e_i} \in \boldsymbol{E}\}$ is a distribution to represent the semantic similarity between $\boldsymbol{t_p}$ and $\boldsymbol{e_i} \in \boldsymbol{E}$, $dot(\cdot,\cdot)$ is dot product.

Implementation of R2. The second requirement is *essential matching unit extractability of matching representation*, which means that under the premise of **R1**, matching representation can saliently represent the unit where the essential matching block is located in. The motivation for this design is discussed in Section 1 and 3. Given the positive query-passage pair (q, p_{pos}), text encoder $g(\cdot;\theta)$, and the text representations for q $(\boldsymbol{t_q})$ and p_{pos} $(\boldsymbol{t_p})$. Matching representation $\boldsymbol{m} \in \mathbb{R}^v$ (v is the dimension of representation) for q and p_{pos} can be obtained by the combination of $\boldsymbol{t_q} \in \mathbb{R}^v$ and $\boldsymbol{t_p} \in \mathbb{R}^v$ as:

$$\boldsymbol{m} = GELU(\boldsymbol{t_q} \odot \boldsymbol{t_p}),$$

where $\odot$ is the element-wise multiplication operator, and $GELU(\cdot)$ is activation function (Hendrycks and Gimpel, 2016) to introduce stochastic regularization. Under the premise of **R1**, $\boldsymbol{t_p}$ can express the semantics of the units in p_{pos} in a balanced manner. In addition, the semantic representation of essential matching unit is more similar to $\boldsymbol{t_q}$ than other units because it really matches the query q. Based on this, the model can be trained to achieve the goal that element-wise multiplication between $\boldsymbol{t_q}$ and $\boldsymbol{t_p}$ can amplify similar patterns (i.e, semantic representation of essential matching unit) and mask the signals of other context units. This design can be supported by convolutional neural networks (LeCun et al., 1998) whose convolution operation can amplify similar patterns in tensors (Girshick et al., 2014). For the p_{pos}, different q amplifies different matching units, which makes $\boldsymbol{m}$ reflect the semantics of the corresponding essential matching unit. Besides, $\boldsymbol{m}$ is obtained by element-wise multiplication between $\boldsymbol{t_q}$ and $\boldsymbol{t_p}$, which is an important part of estimating the relevance of two texts because $dot(\boldsymbol{t_q}, \boldsymbol{t_p}) = sum(\boldsymbol{t_q} \odot \boldsymbol{t_p})$. Thus, the optimization of $\boldsymbol{m}$ can enable the model to obtain the ability to extract essential matching unit according to different queries when estimating relevance. In training, our method utilizes the cross-entropy loss function to optimize the semantic distance between $\boldsymbol{m}$ and each unit to identify the corresponding essential matching units. Given query-passage pair (q, p_{pos}), the embeddings $\boldsymbol{E}$ of the units in p_{pos} as described in Equ. (3), and the label Y for essential matching unit of (q, p_{pos}) as described in Sec. 4.1. Loss function for **R2** is:

$$\mathcal{L}_{extract} = -\sum_{i=1}^{n} y_i \log(dot(\boldsymbol{m}, \boldsymbol{e_i})), \tag{5}$$

where $\boldsymbol{e_i} \in \boldsymbol{E}$, $y_i \in Y$. $\boldsymbol{m}$ is only used as the constraint in training but has important implications for inference. It is because that $\boldsymbol{m}$ is the combination of text representations ($\boldsymbol{t_p}$ and $\boldsymbol{t_q}$). The optimization for $\boldsymbol{m}$ is training the text encoder to output the text representation that is suitable for matching to improve the generalization ability.

Effect of R1 and R2. Table 6 indicates that compared with previous dense retrieval methods, our method makes the semantics of units in text representation tend to be orthogonal to each other. In dot product between two texts, semantics of essential matching unit are preserved, while the other units are masked to capture matching signal.

Total Loss. In addition to $\mathcal{L}_{extract}$ and $\mathcal{L}_{balance}$, contrastive loss is used to train the dense retrieval

Datasets	Jaccard Sim Unigrams	Vanilla DPR	Vanilla DPR+BERM	Knowledge Distillation KD	Knowledge Distillation KD+BERM	Hard Negatives ANCE	Hard Negatives ANCE+BERM
SciFact	22.16	0.478	**0.495**†	0.481	**0.504**†	0.507	**0.511**†
NFCorpus	23.45	0.208	**0.234**†	0.205	**0.242**†	0.237	**0.248**†
TREC-COVID	26.80	0.561	**0.600**†	0.490	**0.505**†	0.654	**0.661**†
SCIDOCS	27.92	0.108	**0.120**†	0.111	**0.115**†	0.122	**0.130**†
DBPedia	30.16	0.236	**0.256**†	0.245	**0.264**†	0.281	**0.293**†
CQADupStack	30.64	**0.281**	0.279	**0.290**	0.281	**0.296**	0.290
HotpotQA	30.87	0.371	**0.386**†	0.427	**0.438**†	0.456	**0.463**†
ArguAna	32.92	0.414	**0.435**†	0.435	**0.437**†	0.415	**0.428**†
Climate-FEVER	34.79	0.176	**0.187**†	0.189	**0.195**†	0.198	**0.201**†
FEVER	34.79	**0.589**	0.585	0.633	**0.664**†	0.669	**0.674**†
FiQA-2018	35.95	**0.275**	0.272	**0.286**	0.285	**0.295**	0.287
Tóuche-2020	37.02	0.208	**0.210**†	0.215	**0.216**†	0.240	**0.248**†
Quora	39.75	0.842	**0.853**†	0.832	**0.836**†	0.852	**0.854**†
NQ	47.27	**0.398**	0.394	**0.420**	0.419	0.446	**0.450**†
Avg	-	0.368	**0.379**	0.376	**0.386**	0.405	**0.410**

Table 1: Zero-shot performance on BEIR (nDCG@10) without any target domain data. **Bold** indicates the better performance in the same training method. †: results with significant performance improvement with p-value ≤ 0.05 compared with baselines. Datasets are ordered by the Jaccard similarity between the source domain (MS-MARCO).

model (Karpukhin et al., 2020) as:

$$\mathcal{L}_c = -\frac{\exp(dot(\boldsymbol{t_q}, \boldsymbol{t_{p+}}))}{\exp(dot(\boldsymbol{t_q}, \boldsymbol{t_{p+}})) + \exp(dot(\boldsymbol{t_q}, \boldsymbol{t_{p-}}))}.$$

So the total loss for training in our method is:

$$\mathcal{L} = \mathcal{L}_c + \alpha \mathcal{L}_{extract} + \beta \mathcal{L}_{balance},$$

where α and β are the hyperparameters.

5 Experiments

This section introduces the experimental setups and analyzes the results.

5.1 Experimental Setups

Datasets. We use MS-MARCO (Nguyen et al., 2016) as the training data (source domain) and choose the 14 publicly available datasets from BEIR [1], a heterogeneous benchmark to evaluate the generalization ability of retrieval models. In addition, we also introduce OAG-QA (Tam et al., 2022) to evaluate the topic generalization ability. Details of datasets are in Appendix A.

Baselines. Our method (BERM) aims to improve the generalization of dense retrieval without any additional modules and target domain data, and it can be combined with different dense retrieval training methods. We select three mainstream dense retrieval training methods including vanilla, hard negatives mining, and knowledge distillation as the baselines. We follow DPR (Karpukhin et al., 2020) to perform vanilla, follow ANCE (Xiong et al., 2021a) to perform hard negatives mining and use a trained cross-encoder as the teacher model to perform knowledge distillation. We compare the change in generalization after combining BERM with these three methods to show the effectiveness of our method. Besides, as previous methods need to obtain target domain data for domain adaptation such as MoDIR (Xin et al., 2022), GenQ (Ma et al., 2021), GPL (Wang et al., 2022) and COCO-DR (Yu et al., 2022), we also compare our method with these methods in domain adaptation setting. Details of baselines are in Appendix B.

Implementation Details. To maintain a fair comparison, we follow (Xiong et al., 2021a) to keep all common hyperparameters (learning rate and batch size, etc.) the same as the three dense retrieval training methods in the baselines. The model is initialized by Roberta$_{base}$ 125M. For the hyperarameters in BERM, δ is 0.1, α is 0.1 and β is 1.0. In domain adaptation, we combine BERM with continuous contrastive pretraining (Yu et al., 2022) to perform unsupervised pre-training on BEIR and use BERM to fine-tune the model on MS-MARCO. We train the model with Pytorch (Paszke et al., 2019) and Hugging Face (Wolf et al., 2020) on 2 Tesla V100 32GB GPUs for about 72 hours.

5.2 Retrieval Performance

Main Results. Table 1 shows the main results on BEIR of different dense retrieval training methods. The results indicate that our method (BERM) can

[1]The left four are unavailable due to copyright restrictions.

Topic	DPR	DPR+BERM
Geometry	0.324	**0.343**†
Mathematical statistics	0.238	**0.246**†
Polynomial	0.174	**0.209**†
Calculus	0.198	**0.207**†
Number theory	0.268	**0.281**†
Matrix	0.259	**0.296**†
Black hole	0.107	**0.143**†
Classical mechanics	0.209	**0.242**†
Physical chemistry	0.154	**0.183**†
Biochemistry	0.306	**0.333**†
Health care	0.389	**0.401**†
Evolutionary biology	0.294	**0.316**†
Cognitive neuroscience	0.303	**0.310**†
Algorithm	0.266	**0.271**†
Neural network	0.179	**0.191**†
Data mining	0.291	**0.336**†
Computer graphics images	0.255	**0.277**†
Optimization	0.230	**0.244**†
Linear regression	0.153	**0.189**†
Economics	0.299	**0.332**†

Table 2: Zero-shot performance on OAG-QA (metric is Top-20). **Bold**: better. †: significant improvement.

be combined with three mainstream dense retrieval training methods (vanilla, knowledge distillation, and hard negatives) to improve the generalization ability without any additional modules and target domain data. For a fair comparison, we combine BERM with the baselines and ensure that their common hyperparameters are consistent. We compute the Jaccard similarity (Ioffe, 2010) between each dataset and MS-MARCO, which can reflect the domain shift between the source and target domain. Table 1 shows that our method is more effective for the datasets with lower Jaccard similarity between MS-MARCO (i.e., domain shift is more significant). This result reflects the ability of our method to capture domain-invariant feature. DPR+BERM and KD+BERM are better than KD, which shows that BERM more effectively enables dense retrieval to learn to capture matching signal than knowledge distillation from cross-encoder.

Topic Generalization. Table 2 shows the generalization performance of DPR and DPR+BERM on different topics of QAG-QA. Topic generalization is important for out-of-domain generalization, which reflects the availability of dense retrieval model for topics with different word distributions. The results show that BERM can significantly improve cross-topic generalization of dense retrieval.

Domain Adaptation. Table 3 shows that BERM achieves the best performance in domain adaptation compared with previous baselines. Specifically, BERM achieves the best average out-of-domain adaptation and in-domain performance. Besides, it gets the best dense retrieval results on seven datasets of BEIR, which is the most of all methods. Our method not only learns the word distribution of the target domain, but also learns the representation suitable for matching for the documents in the target corpus during domain adaptation.

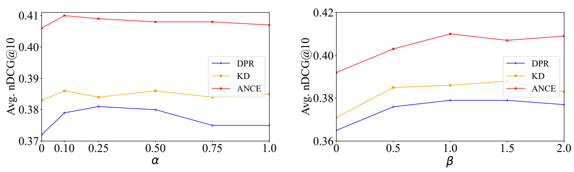
5.3 Ablation Study

Influence of Loss Functions. Table 4 shows the ablation study on the loss functions constrained by **R1** and **R2** via average performance on BEIR. The results indicate that without $\mathcal{L}_{balance}$, $\mathcal{L}_{extract}$ can not improve the generalization, which supports our intuition in Section 3 that only based on the balanced semantic expression of each unit in the text representation, the matching representation is meaningful for extracting the essential semantic unit. This experiment shows that the generalization can be improved significantly when the model is constrained by both **R1** and **R2**.

	DPR+BERM	KD+BERM	ANCE+BERM
	0.379	0.386	0.410
w/o $\mathcal{L}_{balance}$	0.365	0.371	0.392
w/o $\mathcal{L}_{extract}$	0.372	0.383	0.406

Table 4: Ablation study on $\mathcal{L}_{balance}$ and $\mathcal{L}_{extract}$.

Influence of Hyperparameters. Figure 4 shows the average nDCG@10 performance on BEIR with different α and β that are used to tune the weights of different loss functions in training. When α is 0.1 and β is 1.0, our method can achieve the best performance. When α and β are too big, they will interfere with the optimization of the contrastive loss leading to performance degradation.

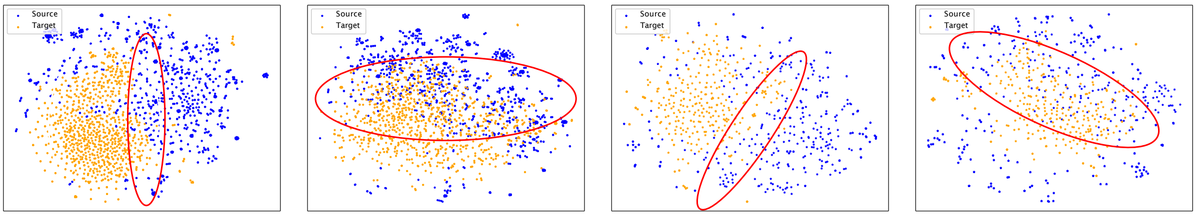


(a) Effect of α. (b) Effect of β.

Figure 4: Performance varies with α and β.

Datasets	Sparse BM25	Late-Inter. ColBERT	Dense MoDIR	Contriever	GenQ	GPL	COCO-DR	BERM (ours)
MS-MARCO	0.228	0.401	0.388	0.407	0.408	-	0.419	0.421
SciFact	0.665	0.671	0.502	0.677	0.644	0.674	0.709	**0.720**†
NFCorpus	0.325	0.305	0.244	0.328	0.319	0.345	0.355	**0.357**†
TREC-COVID	0.656	0.677	0.676	0.596	0.619	0.700	0.789	**0.795**†
SCIDOCS	0.158	0.145	0.124	0.165	0.143	**0.169**	0.160	0.161
DBPedia	0.313	0.392	0.284	**0.413**	0.328	0.384	0.391	0.391
CQADupStack	0.299	0.350	0.297	0.345	0.347	0.357	0.370	**0.374**†
HotpotQA	0.603	0.593	0.462	**0.638**	0.534	0.582	0.616	0.610
ArguAna	0.414	0.233	0.418	0.446	0.493	**0.557**	0.493	0.490
Climate-FEVER	0.213	0.184	0.206	**0.237**	0.175	0.235	0.211	0.220†
FEVER	0.753	0.771	0.680	0.758	0.669	0.759	0.751	**0.760**†
FiQA-2018	0.236	0.317	0.296	0.329	0.308	**0.344**	0.307	0.301
Tóuche-2020	0.367	0.202	**0.315**	0.230	0.182	0.255	0.238	0.235
Quora	0.789	0.854	0.856	0.865	0.830	0.836	0.867	**0.870**†
NQ	0.329	0.524	0.442	0.498	0.358	0.483	0.505	**0.506**
Avg w/o MS-MARCO	0.437	0.444	0.414	0.466	0.425	0.477	0.483	**0.485**

Table 3: nDGC@10 performance on BEIR. **Bold** indicates the best dense retrieval performance. †: results with significant performance improvement with p-value ≤ 0.05 compared with COCO-DR. Dense retrieval models are trained in the domain adaptation setting that unsupervised target domain data can be obtained.



(a) Text Rep. (b) Text Rep. (ours) (c) Matching Rep. (d) Matching Rep. (ours)

Figure 5: T-SNE of the text and matching representations for source and target domains.

5.4 Model Analysis

Domain-invariant Representation. Figure 5 shows that our method is effective in capturing the domain-invariant feature of the representation. We utilize T-SNE to visualize the representations of source and target (SciFact) domains encoded by DPR and DPR+BERM respectively. The results indicate that representations of the two domains encoded by DPR are more separable. After combining our method, the two domains become more difficult to separate, which indicates that our method is more invariant to represent the texts in different domains. More datasets are in Appendix C.

Evaluation of R1 and R2. Table 5 shows the effectiveness of **R1** and **R2**. We randomly sample 100,000 query-passage pairs from the test set. For each passage p, we compute semantic similarity between text representation and each unit via $sim(\boldsymbol{t_p}, \boldsymbol{E}) = \{dot(\boldsymbol{t_p}, \boldsymbol{e_i}) | \boldsymbol{e_i} \in \boldsymbol{E}\}$. We compute the variance of $sim(\boldsymbol{t_p}, \boldsymbol{E})$ and get the average of variance on the sampled set, which can

Metric	DPR Var.	DPR Acc.	KD Var.	KD Acc.	ANCE Var.	ANCE Acc.
Baseline	3.756	0.407	3.891	0.415	3.432	0.450
BERM	0.005	0.778	0.007	0.803	0.003	0.846

Table 5: Variance of the semantic similarity between text representation and units (smaller the better). Accuracy to identify essential matching unit (bigger the better).

reflect the balance of text representation on expressing the semantics of units. Table 5 shows that BERM has a smaller variance (semantic unit balance of text representation) and is more accurate in identifying the essential matching unit (essential matching unit extractability of matching representation) than baselines, which indicates the effectiveness of **R1** and **R2**.

Relationship Between Units. Table 6 shows that our method makes units in a passage more dispersed (tend to be orthogonal), which is more conducive to determining the unit that matches the

	DPR		KD		ANCE	
Axis	X	Y	X	Y	X	Y
Baseline	8.1	6.5	19.5	18.5	27.2	25.9
BERM	248.5	409.5	252.3	410.9	260.4	425.4

Table 6: Dispersion of T-SNE result of representations of units in a passage (measured by the variance of the coordinates on x-axis and y-axis).

query and masking the signals of other units. Our method makes the representation of the passage more suitable for matching, which is the domain-invariant feature for generalization.

6 Conclusion

In this paper, we propose an effective method called BERM to improve the generalization ability of dense retrieval without target domain data and additional modules. The basic idea of BERM is learning the domain-invariant feature, that is, matching signal. To achieve it, we introduce a novel concept of dense retrieval to represent the matching information between two texts, the matching representation. Further, we propose two requirements for matching and text representations as the constraint in the training of dense retrieval to enhance the ability to extract essential matching information from the passage according to different queries under the premise of balanced expression of the text. The two requirements unlock the ability of dense retrieval to capture matching signal without additional interaction. Experimental results show that BERM is a flexible method that can be combined with different dense retrieval training methods without inference overhead to improve the out-of-domain generalization ability. In domain adaptation setting, our method is also effective and performs better than baselines.

Limitations

In this paper, we propose a novel concept of dense retrieval, the matching representation. Based on this, we introduce a novel generalizable dense retrieval training method via training the balanced and extractable representation for matching (BERM). Despite the strong performance of our method in improving the generalization ability of dense retrieval models, more theoretical proof needs to be researched to gain the deeper understanding of generalization improvement. Especially for matching representation, more theoretical analysis and implementation will be discussed in future work. We believe that the deeper study of matching representation will promote the development of dense retrieval, because it not only alleviates the problem that query and passage cannot interact in depth during training, but also describes the essence of retrieval task.

Ethics Statement

Our work innovatively proposes the concept of matching representation in dense retrieval and designs a generalization improvement strategy that can be flexibly combined with different dense retrieval training methods. Our work has important implications for improving the performance of neural information retrieval models. We declare that our work complies with the ACL Ethics Policy.[2]

Acknowledgements

This work was supported by the National Key R&D Program of China (2022YFB3103700, 2022YFB3103704), the National Natural Science Foundation of China (NSFC) under Grants No. 62276248, and the Youth Innovation Promotion Association CAS under Grants No. 2023111.

References

Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. 2020. Overview of Touché 2020: Argument Retrieval. In *Working Notes Papers of the CLEF 2020 Evaluation Labs*, volume 2696 of *CEUR Workshop Proceedings*.

Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A full-text learning to rank dataset for medical information retrieval. In *European Conference on Information Retrieval*, pages 716–722. Springer.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the Conference on ACL 2017*, pages 1870–1879. Association for Computational Linguistics.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 2020 Conference on ACL*, pages 2270–2282, Online. Association for Computational Linguistics.

[2] https://www.aclweb.org/portal/content/acl-code-ethics

Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. CLIMATE-FEVER: A dataset for verification of real-world climate claims. *arXiv preprint arXiv:2012.00614*.

Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2022. From distillation to hard negative sampling: Making sparse neural IR models more effective. In *Proceedings of the 2021 Conference on SIGIR*, pages 2353–2359. ACM.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2288–2292, New York, NY, USA. Association for Computing Machinery.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. COIL: Revisit exact lexical match in information retrieval with contextualized inverted list. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3030–3042, Online. Association for Computational Linguistics.

Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 580–587. IEEE Computer Society.

Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. DBpedia-Entity v2: A test collection for entity search. In *Proceedings of the 2017 Conference on SIGIR*, SIGIR '17, page 1265–1268, New York, NY, USA. Association for Computing Machinery.

Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415.

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 *Proceedings of the 2021 Conference on SIGIR*, pages 113–122. ACM.

Wu Hong, Zhuosheng Zhang, Jinyuan Wang, and Hai Zhao. 2022. Sentence-aware contrastive learning for open-domain passage retrieval. In *Proceedings of the 2022 Conference on ACL*, pages 1062–1074. Association for Computational Linguistics.

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*, ADCS '15, New York, NY, USA. Association for Computing Machinery.

Michelle Chen Huebscher, Christian Buck, Massimiliano Ciaramita, and Sascha Rothe. 2022. Zero-shot retrieval with search agents and hybrid environments. *CoRR*, abs/2209.15469.

Sergey Ioffe. 2010. Improved consistent sampling, weighted minhash and L1 sketching. In *Proceedings of the 2010 Conference on ICDM*, pages 246–255. IEEE Computer Society.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Towards unsupervised dense information retrieval with contrastive learning. *CoRR*, abs/2112.09118.

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

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. 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 EMNLP*, pages 6769–6781. 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*, 7:452–466.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324.

Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. In-batch negatives for knowledge distillation with tightly-coupled teachers for dense retrieval. In *Proceedings of the 6th Workshop on Representation Learning for NLP, RepL4NLP@ACL-IJCNLP 2021, Online, August 6, 2021*, pages 163–173. Association for Computational Linguistics.

Ji Ma, Ivan Korotkov, Yinfei Yang, Keith B. Hall, and Ryan T. McDonald. 2021. Zero-shot neural passage retrieval via domain-targeted synthetic question generation. In *Proceedings of the 2021 Conference on EACL*, pages 1075–1088. 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.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 2017 Conference on WWW*, pages 1291–1299. ACM.

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 2016 Conference on NIPS*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y. Zhao, Yi Luan, Keith B. Hall, Ming-Wei Chang, and Yinfei Yang. 2021. Large dual encoders are generalizable retrievers.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the 2019 Conference on NeurIPS*, pages 8024–8035.

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

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. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qifei Wu, Yuchen Ding, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2022. A thorough examination on zero-shot dense retrieval. *CoRR*, abs/2204.12755.

Stephen E. Robertson, Steve Walker, and Micheline Hancock-Beaulieu. 1995. Large test collection experiments on an operational, interactive system: Okapi at TREC. *Inf. Process. Manag.*, 31(3):345–360.

Weng Lam Tam, Xiao Liu, Kaixuan Ji, Lilong Xue, Xingjian Zhang, Yuxiao Dong, Jiahua Liu, Maodi Hu, and Jie Tang. 2022. Parameter-efficient prompt tuning makes generalized and calibrated neural text retrievers.

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 *Proceedings of the 2021 Conference on NeurIPS Datasets and Benchmarks*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

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).

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 2018. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 2018 Conference on ACL*, pages 241–251, Melbourne, Australia. Association for Computational Linguistics.

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 EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

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

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 4003–4012. European Language Resources Association.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on EMNLP*, pages 38–45. Association for Computational Linguistics.

Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul Bennett. 2022. Zero-shot dense retrieval with momentum adversarial domain invariant representations. In *Proceedings of the 2022 Conference on ACL Findings*, pages 4008–4020. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021a. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *Proceedings of the 2021 Conference on ICLR*. OpenReview.net.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021b. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *Proceedings of the 2021 Conference on ICLR*. OpenReview.net.

Shicheng Xu, Liang Pang, Huawei Shen, and Xueqi Cheng. 2022a. Match-prompt: Improving multi-task generalization ability for neural text matching via prompt learning. In *Proceedings of the 2022 Conference on CIKM*, CIKM '22, page 2290–2300, New York, NY, USA. Association for Computing Machinery.

Shicheng Xu, Liang Pang, Huawei Shen, and Xueqi Cheng. 2022b. Nir-prompt: A multi-task generalized neural information retrieval training framework.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on EMNLP*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Haotian Ye, Chuanlong Xie, Tianle Cai, Ruichen Li, Zhenguo Li, and Liwei Wang. 2021. Towards a theoretical framework of out-of-distribution generalization. In *Proceedings of the 2021 Conference on NeurIPS*, pages 23519–23531.

Yue Yu, Chenyan Xiong, Si Sun, Chao Zhang, and Arnold Overwijk. 2022. COCO-DR: combating distribution shifts in zero-shot dense retrieval with contrastive and distributionally robust learning. *CoRR*, abs/2210.15212.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Optimizing dense retrieval model training with hard negatives. In *Proceedings of the 2021 Conference on SIGIR*, pages 1503–1512. ACM.

Shunyu Zhang, Yaobo Liang, Ming Gong, Daxin Jiang, and Nan Duan. 2022. Multi-view document representation learning for open-domain dense retrieval. In *Proceedings of the 2022 Conference on ACL*, pages 5990–6000. Association for Computational Linguistics.

A Datasets

In our experiment, source domain datsaet used as training data is MS-MARCO and target domain datasets used as testing data are collected from BEIR (Thakur et al., 2021), which is a a heterogeneous benchmark to evaluate the generalization ability of retrieval models. Detials of the datasets are shown in Table 7. In addition, we also introduce OAG-QA (Tam et al., 2022), which is a fine-grained question-answering retrieval dataset consisting of different topics. We select datasets of different topics from 20 disciplines as the testing data to evaluate the generalization ability to different topics with different word distribution. Details of OAG-QA are shown in Table 8.

B Baselines

We introduce the baselines in the main experiment and the domain adaptation experiment respectively.

B.1 Baselines for Main Experiment

In the main experiment, our method is combined with different mainstream dense retrieval training methods to improve its generalization. We consider three training methods including vanilla (DPR (Karpukhin et al., 2020)), knowledge distillation (KD) and hard negatives mining (ANCE (Xiong et al., 2021a)).

- **DPR** trains the dense retrieval model via in-batch negative sampling. Different from (Karpukhin et al., 2020), we train DPR on MS-MARCO to achieve a fair comparison.
- **KD** trains the dense retrieval model under the guidance of the soft labels provided by the teacher model. In the experiment, we use a cross-encoder model trained on MS-MARCO as the teacher model.
- **ANCE** trains the dense retrieval model with hard negatives updated in parallel as described in (Xiong et al., 2021a).

B.2 Baselines for Domain Adaptation

- **MoDIR** uses the data from source and target domains for adversarial training to perform unsupervised domain adaptation.
- **Contriever** performs unsupervised pre-training on Wikipedia and CC-Net (Wenzek et al., 2020).

Task	Domain	Dataset	Test		Avg. Word Lengths	
			#Query	#Corpus	Query	Document
Passage-Retrieval	Misc.	MS-MARCO (Nguyen et al., 2016)	6,980	8,841,823	5.96	55.98
Bio-Medical	Bio-Medical	TREC-COVID (Voorhees et al., 2021)	50	171,332	10.60	160.77
Information Retrieval	Bio-Medical	NFCorpus (Boteva et al., 2016)	323	3,633	3.30	232.26
Open-domain	Wikipedia	NQ (Kwiatkowski et al., 2019)	3,452	2,681,468	9.16	78.88
Question	Wikipedia	HotpotQA (Yang et al., 2018)	7,405	5,233,329	17.61	46.30
Answering	Finance	FiQA-2018 (Maia et al., 2018)	648	57,638	10.77	132.32
Argument	Misc.	ArguAna (Wachsmuth et al., 2018)	1,406	8,674	192.98	166.80
Retrieval	Misc.	Touché-2020 (Bondarenko et al., 2020)	49	382,545	6.55	292.37
Duplicate-Question	StackEx.	CQADupStack (Hoogeveen et al., 2015)	13,145	457,199	8.59	129.09
Retrieval	Quora	Quora (Thakur et al., 2021)	10,000	522,931	9.53	11.44
Entity-Retrieval	Wikipedia	DBPedia (Hasibi et al., 2017)	400	4,635,922	5.39	49.68
Citation-Prediction	Scientific	SCIDOCS (Cohan et al., 2020)	1,000	25,657	9.38	176.19
	Wikipedia	FEVER (Thorne et al., 2018)	6,666	5,416,568	8.13	84.76
Fact Checking	Wikipedia	Climate-FEVER (Diggelmann et al., 2020)	1,535	5,416,568	20.13	84.76
	Scientific	SciFact (Wadden et al., 2020)	300	5,183	12.37	213.63

Table 7: Details of the datasets in BEIR, the table is collected from (Thakur et al., 2021).

Discipline	Topic	#Query	#Corpus
Geometry	Geometry	230	10,000
Statistics	Mathematical Statistics	144	10,000
Algebra	Polynomial	280	10,000
Calculus	Calculus	242	10,000
Number theory	Number theory	274	10,000
Linear algebra	Matrix	130	10,000
Astrophysics	Black hole	160	10,000
Physics	Classical mechanics	115	10,000
Chemistry	Physical chemistry	190	10,000
Biochemistry	Biochemistry	129	10,000
Health Care	Health care	288	10,000
Natural Science	Evolutionary biology	471	10,000
Psycology	Cognitive neuroscience	348	10,000
Algorithm	Algorithm	386	10,000
Neural Network	Neural network	590	10,000
Data Mining	Data mining	131	10,000
Computer Graphics	Computer graphics images	68	10,000
Deep Learning	Optimization	238	10,000
Machine Learning	Linear regression	244	10,000
Economics	Economics	238	10,000

Table 8: Details of QAG-QA.

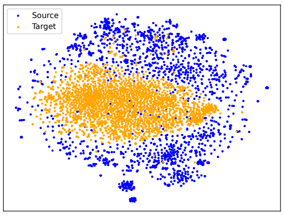
- **GenQ** uses T5 (Raffel et al., 2020) generates 5 queries for each passage in target domain and fine-tunes TAS-B (Hofstätter et al., 2021) on this data.

- **GPL** improves the domain adaptation performance based on GenQ. In addition to generated queries, GPL uses cross-encoder to provide the pseudo-label. GPL fine-tunes multiple backbones on the generated queries and pseudo-labels and we report the best performance that is fine-tuned on TAS-B.

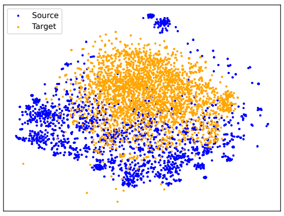
- **COCO-DR** performs unsupervised pre-training on target domain and introduces distributional robust optimization.

C Domain-invariant Representation

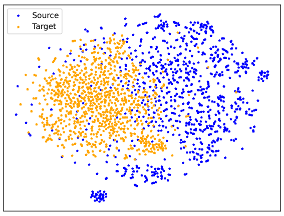
Visualized results of T-SNE of representations of source and target (SCIDOCS, TREC-COVID, NFCorpus and DBpedia) domains encoded by DPR and DPR+BERM respectively are shown in Figure 6.



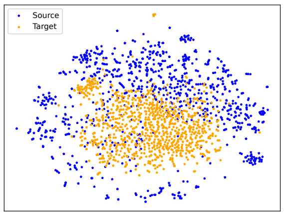
(a) Text Rep.



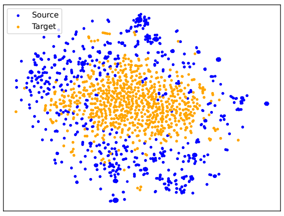
(b) Text Rep. (ours)



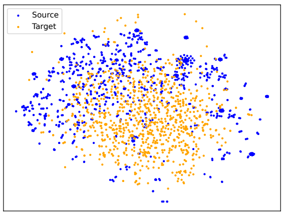
(c) Matching Rep.



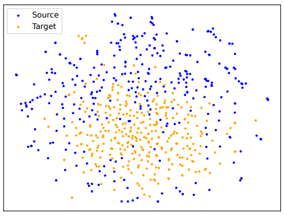
(d) Matching Rep. (ours)



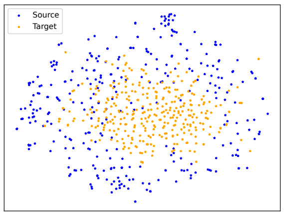
(e) Text Rep.



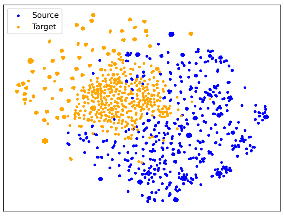
(f) Text Rep. (ours)



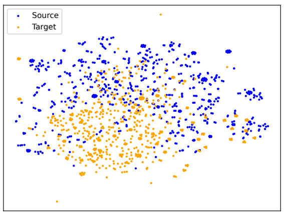
(g) Matching Rep.



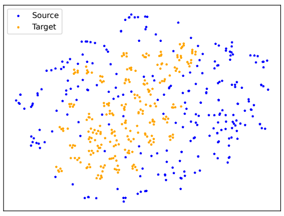
(h) Matching Rep. (ours)



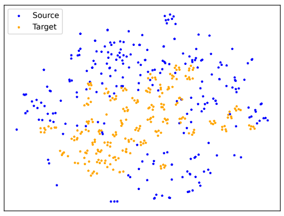
(i) Text Rep.



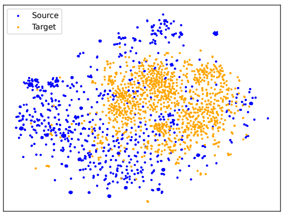
(j) Text Rep. (ours)



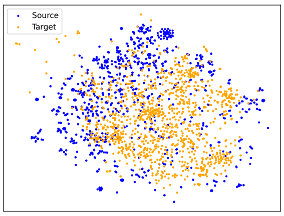
(k) Matching Rep.



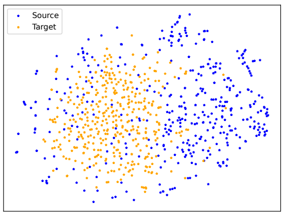
(l) Matching Rep. (ours)



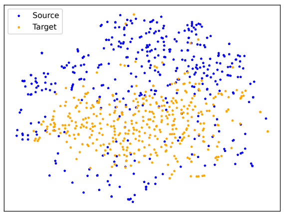
(m) Text Rep.



(n) Text Rep. (ours)



(o) Matching Rep.



(p) Matching Rep. (ours)

Figure 6: T-SNE of the text and matching representations for source and target domains.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
After the Section 6 Conclusion

☑ A2. Did you discuss any potential risks of your work?
Ethics Statement

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 5

☑ B1. Did you cite the creators of artifacts you used?
Section 5.1

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 5.1

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 5.1

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Ethics Statement

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Appendix (only domain)

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix

C ☑ Did you run computational experiments?

Section 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 5.1

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 5

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.