SPIRIT: Zero Shot Information Retrieval Domain Transfer with Soft Prompts

Ethan Kim*
ethan_kim@college.harvard.edu

Diganta Misra
Mila- Quebec AI Institute
diganta.misra@mila.quebec

ABSTRACT

Dense information retrieval yields strong in-domain performance, but often struggles with out-of-domain generalization, lagging behind unsupervised methods. Retrieval tasks can vary across a number of dimensions including domain, query intent, and language. Using a single dense retrieval model for all tasks often underperforms lexical methods such as BM25. For practical information retrieval systems, it is expensive to deploy a different model for each task. Therefore, our motivation is to develop a cheap and effective information retrieval model that maintains strong performance across different domains while easily adapting to any new domain. Other approaches to domain transfer in information retrieval rely on large auxiliary language models or datasets and create a separate model for each task. In this work, we develop a method utilizing prompt tuning to efficiently adapt dense retrievers with a minimal amount of additional computation. By combining models trained on a variety of different domains, we can effectively boost performance on a target task in a new domain. Specifically, we train dense retrieval models using prompt tuning on a large number of information retrieval tasks across diverse domains and types of query intents. To adapt to a new domain, we create new prompt embeddings by averaging the prompt embeddings from a set of source tasks selected in an unsupervised manner. We evaluate zero-shot transfer performance across a wide variety of information retrieval domains and show competitive performance while leveraging a minimal amount of compute. Notably, our SPIRIT method achieves while being extremely lightweight and practical to deploy in production.

CCS CONCEPTS

• **Information systems → Retrieval models and ranking**.

KEYWORDS

information retrieval, domain transfer, prompt tuning
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Encoding queries and documents using large language models has been shown to yield strong performance on tasks such as information retrieval and open domain question answering [8]. Converting text to dense vectors allows for efficient similarity computation and yields state-of-the-art retrieval results when trained on in-domain data. However lexical approaches such as TFIDF and BM25 typically outperform dense methods when applied to a wide variety of domains and in addition, do not require any training data [4]. Given that we want to enable dense information retrieval for newly published information sources it is important to have techniques that can update dense retrievers to perform well in new domains.

Broadly speaking there are several main categories of approaches to improve the out-of-domain robustness of dense retrievers. First, better self-supervised intermediate pretraining objectives can improve generalization capability [16][7] [9]. Second, supervised intermediate training on related source domains can improve generalization, [8] [14]. Third, domain-specific retrievers can be trained in a supervised manner on synthetic data generated for that domain [4] [24] [3] [21] [1].

Specially constructed pretraining tasks automatically extract training data that mimic query and document pairs. Intermediate training on supervised data sets can boost performance on a target domain. Finally given documents in a target domain we can leverage generative models to produce related queries and train a dense retriever on the synthetic data.

All of these methods have found success; however, relying on synthetic data to train in-domain retrievers has a number of disadvantages. First, generating large numbers of synthetic queries requires expensive inference on large language models (up to 137 billion parameters) [3]. Second, training a new dense retriever per new domain increases the storage and deployment costs associated with using dense retrievers in practice. Thus, our goal is to adapt to a new domain by leveraging learning on existing domains while 1) avoiding reliance on synthetic data, and 2) reusing a single pre-trained backbone model.

To achieve our goal, we train models using prompt tuning [10], which adapts a model by tuning only a small number of prompt embeddings or "pseudo tokens" that are prepended to the model input. The rest of the language model parameters are frozen and do not receive any gradient updates. Although finetuning only a small percentage of the model parameters, prompt tuning can often match the performance of full finetuning. Training dense retrievers with prompt tuning helps us achieve our goals in multiple ways: As a parameter efficient method, prompt tuning provides a regularization effect that increases the generalizability of neural retrievers [19]. Moreover, tuning models can easily be combined or transferred across domains simply by averaging the prompt embeddings [22] [18]. The inherent interpretability of prompt tuning also allows us to examine nearest neighbors in the vocabulary embedding space in order to understand and predict domain transfer performance. Most importantly, prompt tuning can be used to store and deploy multiple models at the same time using in-batch parallel computing.

*

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Thus prompt tuning is extremely practical for the simultaneous real-world deployment of dense retrieval models across many different domains.

By training models with prompt tuning, we can easily create a model for a new domain by averaging the prompt embeddings of the trained models in the most similar domains. We show through experiments on diverse information retrieval tasks including 14 tasks from BEIR [21] and, 25 fine grained tasks from [19] that our method is able to effectively transfer to new domains in a true zero shot manner without any additional training or external models. Overall our contributions advance research in domain transfer for dense retrieval by:

- Introducing a method for domain transfer using prompt tuning and dense retrieval
- Demonstrating the effectiveness of prompt tuning for domain transfer across a wide variety of domains without the need for large amounts of data or compute.
- Creating a simple and extensible model that can be used in concert with other intermediate training or synthetic data techniques to boost out-of-domain performance.

2 RELATED WORK

Dense Retrieval (DR) systems, also known as bi-encoder or dual-encoder models seek to embed queries and documents into an embedding space such that queries are close to relevant documents. Retrieval systems can be deployed in a highly efficient and scalable manner using approximate nearest neighbors search and offline encoding of the document index. A typical setup coverts queries and documents to a vector using a transformer-based encoder and trains using supervised contrastive loss on positive query and document pairs with in-batch negatives. Given and query with positive documents p^+ and negative documents p^-, we optimize the contrastive loss function

$$\mathcal{L} = -\log \frac{e^{\text{sim}q_i,p^+}}{e^{\text{sim}(q_i,p_i^+)} + \sum_{j=1}^{n} e^{\text{sim}(q_i,p_i^-)}}$$

2.1 Better Dense retrievers

Other work focuses on improving dense retrievers through creative pretraining tasks, optimizing the mining of hard negatives, or encoding multiple vectors per document. Self-supervised pretraining seeks to leverage signals such as the proximity of phrases in a document to generate positive pairs for training [7], [25] [23] [16]. Such methods can yield improved results at the expense of increased training or deployment complexity.

2.2 Domain Synthetic Finetuning

The simplest method for zero-shot adaptation of a dense retriever for a new domain is to get synthetic supervised data for that domain if none is available. QGen uses a T5 encoder model finetuned on MSMARCO queries and documents to generate synthetic queries [21]. GPL leverages the QGen model along with distillation from a cross-encoder and hard negative mining for effective zero shot adaptation to a new domain [24]. [14] adapt models to a new domain by choosing specifically matched source datasets including 65 million synthetically generated question-answer pairs [11]. Promptagator takes a few-shot approach, leveraging a 137 Billion parameter language model to generate synthetic queries [3]. Similarly, InPars uses GPT-3 Curie [2] to generate synthetic in domain queries. ART utilizes a question generation model to calculate relevance scores during training [17].

2.3 Parameter Efficent Finetuning

Parameter efficient learning methods seek to adapt a model by updating only a subset of the total parameters. Major approaches include Adapters which inserts a trainable bottleneck layer between transformer layers [5] [15], BitFit which trains only the bias terms [27], LORA which trains a low-rank delta matrix [6]. and Prompt Tuning which inserts trainable prompts into the model input [10]. Inspired by the ability of large language models to adapt to in-context prompts [2], Prompt Tuning optimizes a small number of embedding tokens $P_e \in^{p \times e}$ which are then prepended to the instance input embeddings to get the model input $[P_e; X_e] \in^{(p+n) \times e}$

Prompt tuning and other PE methods have been applied to information retrieval tasks. DPTDR uses a retrieval-oriented intermediate pretraining technique similar to [7] which trains soft prompts that can be used to train downstream models for multitask inference [20]. With optimization, parameter-efficient methods can achieve the same performance on IR tasks as full finetuning [13]. MatchPrompt uses prompts to train cross-encoder models for improved out-of-domain generalization in text ranking [26]. Finally, intermediate pretraining using prompts can improve zero-shot performance, likely due to the prevention of overfitting [19]. Prompt tuned models can transfer across domains as shown in [22] and [18]

3 METHOD

Our SPIRIT method **S**oft**P**rompt **I**nformation **R**etr**I**eval **T**ransfer method combines models trained on a diverse set of source domains to transfer to a new target domain. The goal is to combine the knowledge encoded in the parameters of the source task models. We take inspiration from [12] where model parameters are combined using a weighted average based on transferability to a particular target domain.

3.1 Combining Models for Zero Shot Transfer

Formally, we define a set of source domains D_s we want to transfer from, and a target domain d_t. Given unlabeled documents in each domain, we define a measure of similarity between domains, s. Calculating similarities, we have s_{ij} = similarity score between domains i and j. To transfer to a new domain, we simply set the soft prompt embedding weights as the weighted average of the trained soft prompts from the source domains.

$$\theta_t = \sum_{i=0}^{|s|} \theta_i \alpha_{i,t}$$

We set the weight assigned to the parameters of each source domain to be proportional to its similarity to the target domain.

$$\alpha_{ij} = \frac{s_{ij}}{\sum_{k=0}^{|s|} s_{ik}}$$

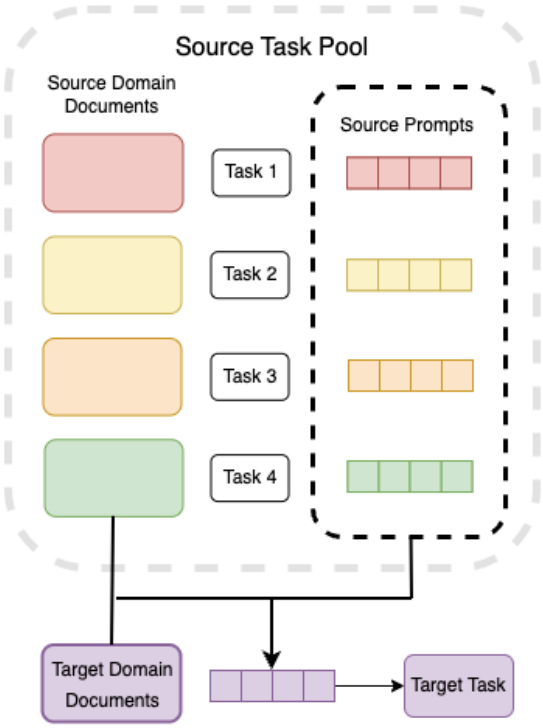

Figure 1: SPIRIT Methods: A pool of source prompts is trained on various tasks. For zero shot transfer source prompts are combined as a weighted average according to domain similarity and transferred to a target task

We experiment with a temperature-weighted average as well as only transferring from the top k most similar source domains.

$$\alpha_{ij} = \frac{\exp(s_{ij}/\tau)}{\sum_{k=0}^{|s|} \exp(s_{ik}/\tau)}$$

3.2 Domain Similarity

We experiment with two different heuristics to calculate the similarity between two domains, inspired by [22]. To compare two domains, we embed n documents from each domain using a pretrained language model encoder. We first experiment with using the cosine similarity of mean pooled document embeddings.

$$s_{ij} = \cos\left(\frac{1}{n}\sum_{k=0}^{n} e_{ik}, \frac{1}{n}\sum_{k=0}^{n} e_{ij}\right)$$

In the second method, we calculate the average pairwise cosine similarity between document embeddings from each domain

$$s_{ij} = \frac{1}{n^2}\sum_{k}\sum_{l}\cos(e_{ik}, e_{il})$$

4 EXPERIMENTS

We train dense retrievers using prompt tuning and contrastive loss with in-batch negatives. We first do an intermediate training phase on 4 datasets (Natural Questions, TriviaQA, Web Questions and Curated TREC) following the DPR-multi setup [8]. For dense retrieval training we use a batch size of 128, a learning rate of 1e-2 and train for up to 40 epochs following the DPR setup. We use bert-base-uncased as our pretrained backbone model. Our training is implemented using the OpenDelta[1] and OpenMatch[2]. To extract a dense vector representation of a document or query, we mean pool the final token hidden state representations excluding the soft prompt tokens. We use 50 soft prompt tokens and use a shared query and passage encoder To calculate domain similarity, we embed document using an off the shelf retriever [3].

4.1 Datasets

To evaluate out-of-domain performance we test zero-shot transfer a diverse set of information retrieval tasks, BEIR [21]. BEIR includes 15 heterogeneous information retrieval tasks. We follow a leave-one-out setup for transfer evaluation; given a target dataset from a set of source datasets, we remove that dataset from the pool and evaluate the performance allowing transfer from every other dataset. We experiment with different settings and find that using mean pooled document embeddings for domain similarity comparison works best. In addition, we use a low temperature of 0.5 to up weight the most similar source domains.

4.2 Results

We compare zero shot out of domain results on BEIR using the normalized cumulative discounted gain @10 metric (ndcg@10). We compare to the baseline generalized prompt tuned retrieval from [19] As a ceiling on zero shot performance we compare to models trained on synthetic data from a 137 Billion parameter model [3]. For OAGQA datasets we compare the zero-shot performance using top 20 retrieval accuracy (success@20) and follow the same grouping of 87 topics into 22 domains.

4.3 Discussion

Zero Shot Performance results are shown in Table 1. Notably using k=5 yielded improved performance over using k=1 (i.e simply using the most similar domain). This indicates that averaging the soft prompts is able to transfer knowledge from multiple domains. Meanwhile k=5 outperformed k=10 suggesting that only the few most relevant datasts should be used for transfer. For BEIR datasets, we observed relatively large variance in results. We hypothesize that due to the high heterogeneity in BEIR datasets in terms of size, domain and type of matching signal the most similar source domains do not transfer well, especially for small datasets.

Our method is designed with practicality in mind. Once source domain models are trained they can be reused for any new information retrieval tasks. Models can also be added and removed from the full as information retrieval needs evolve over time. The flexibility of combining soft prompts can be used to adapt to new search terms

[1] https://github.com/thunlp/OpenDelta
[2] https://github.com/OpenMatch/OpenMatch
[3] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Table 1: Performance comparison of different models

Dataset	BM25	PT V2	SPIRIT	Promptagator
MSMARCO	0.228	0.171	**0.361**	0.727
TREC-COVID	0.656	0.394	–	0.334
NFCorpus	0.325	0.224	**0.227**	0.604
NQ*	0.329	0.479	**0.496**	0.404
HotpotQA	0.603	0.416	**0.644**	0.538
FiQA-2018	0.236	**0.128**	0.105	0.266
ArguAna	0.315	**0.214**	0.194	–
Touche-2020	0.367	**0.207**	0.067	–
CQADupstack	0.299	**0.158**	0.154	–
Quora	0.789	0.509	**0.703**	0.762
DBPedia	0.323	**0.254**	0.090	0.214
SCIDOCS	0.158	**0.099**	0.063	0.623
FEVER	0.753	**0.593**	0.535	–
Climate-FEVER	0.213	**0.194**	0.135	–
SciFact	0.665	0.436	**0.466**	–

NQ dataset: in domain data used.

Table 2: Evaluation of different information retrieval methods using ndcg@10 as the evaluation metric.

and domains as they emerge. For example, if new fields of scientific research were to emerge we bootstrap a model by pulling models trained on the most relevant arxiv categories.

The SPIRIT method incurs minimal extra cost at deployment and training time. Creating a model for a new domain simply requires calculating similarities and averaging prompts which takes under a minute. At deployment time the same backbone model can be leveraged with in-batch parallel computing to serve queries for many models simultaneously.

5 CONCLUSION

In this work, we present a simple approach to domain transfer for information retrieval by using soft prompts. By retrieving the most relevant prompts from the source models we were able to boost zero-shot performance. Notable future improvements to this method include scaling up a very large pool of source prompts to select and evaluating transfer performance across different types of query intents. Adapting language models with minimal computation has many practical use cases.

REFERENCES

[1] Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. InPars: Data Augmentation for Information Retrieval using Large Language Models. https://doi.org/10.48550/arXiv.2202.05144 arXiv:2202.05144 [cs].

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]* (July 2020). http://arxiv.org/abs/2005.14165 arXiv:2005.14165.

[3] Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. 2022. Promptagator: Few-shot Dense Retrieval From 8 Examples. http://arxiv.org/abs/2209.11755 arXiv:2209.11755 [cs].

[4] Revanth Gangi Reddy, Vikas Yadav, Md Arafat Sultan, Martin Franz, Vittorio Castelli, Heng Ji, and Avirup Sil. 2022. Towards Robust Neural Retrieval with Source Domain Synthetic Pre-Finetuning. In *Proceedings of the 29th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 1065–1070. https://aclanthology.org/2022.coling-1.89

[5] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP. http://arxiv.org/abs/1902.00751 arXiv:1902.00751 [cs, stat].

[6] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. http://arxiv.org/abs/2106.09685 arXiv:2106.09685 [cs].

[7] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised Dense Information Retrieval with Contrastive Learning. http://arxiv.org/abs/2112.09118 arXiv:2112.09118 [cs].

[8] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. http://arxiv.org/abs/2004.04906 arXiv:2004.04906 [cs].

[9] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. http://arxiv.org/abs/1906.00300 arXiv:1906.00300 [cs].

[10] Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. *arXiv:2104.08691 [cs]* (Sept. 2021). http://arxiv.org/abs/2104.08691 arXiv: 2104.08691.

[11] 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 (2021), 1098–1115.

[12] Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2022. Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models. http://arxiv.org/abs/2208.03306 arXiv:2208.03306 [cs].

[13] Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, and Xueqi Cheng. 2022. Scattered or Connected? An Optimized Parameter-efficient Tuning Approach for Information Retrieval. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 1471–1480. https://doi.org/10.1145/3511808.3557445 arXiv:2208.09847 [cs].

[14] Barlas Oğuz, Kushal Lakhotia, Anchit Gupta, Patrick 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. http://arxiv.org/abs/2107.13602 arXiv:2107.13602 [cs].

[15] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-Destructive Task Composition for Transfer Learning. http://arxiv.org/abs/2005.00247 arXiv:2005.00247 [cs].

[16] Ori Ram, Gal Shachaf, Omer Levy, Jonathan Berant, and Amir Globerson. 2022. Learning to Retrieve Passages without Supervision. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Seattle, United States, 2687–2700. https://doi.org/10.18653/v1/2022.naacl-main.193

[17] Devendra Singh Sachan, Mike Lewis, Dani Yogatama, Luke Zettlemoyer, Joelle Pineau, and Manzil Zaheer. 2022. Questions are all you need to train a dense passage retriever. *arXiv preprint arXiv:2206.10658* (2022).

[18] Yusheng Su, Xiaozhi Wang, Yujia Qin, Chi-Min Chan, Yankai Lin, Huadong Wang, Kaiyue Wen, Zhiyuan Liu, Peng Li, Juanzi Li, Lei Hou, Maosong Sun, and Jie Zhou. 2022. On Transferability of Prompt Tuning for Natural Language Processing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Seattle, United States, 3949–3969. https://doi.org/10.18653/v1/2022.naacl-main.290

[19] 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. http://arxiv.org/abs/2207.07087 arXiv:2207.07087 [cs].

[20] Zhengyang Tang, Benyou Wang, and Ting Yao. 2022. DPTDR: Deep Prompt Tuning for Dense Passage Retrieval. In *Proceedings of the 29th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 1193–1202. https://aclanthology.org/2022.coling-1.103

[21] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models. http://arxiv.org/abs/2104.08663 arXiv:2104.08663 [cs].

[22] Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Cer. 2022. SPoT: Better Frozen Model Adaptation through Soft Prompt Transfer. http://arxiv.org/abs/2110.07904 arXiv:2110.07904 [cs].

[23] Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. Tsdae: Using transformer-based sequential denoising auto-encoder for unsupervised sentence embedding learning. *arXiv preprint arXiv:2104.06979* (2021).

[24] Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2022. GPL: Generative Pseudo Labeling for Unsupervised Domain Adaptation of Dense Retrieval. *arXiv:2112.07577 [cs]* (April 2022). http://arxiv.org/abs/2112.07577 arXiv: 2112.07577.

[25] Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2022. LaPraDoR: Unsupervised Pretrained Dense Retriever for Zero-Shot Text Retrieval. In *Findings of the Association for Computational Linguistics: ACL 2022*. Association for Computational Linguistics, Dublin, Ireland, 3557–3569. https://doi.org/10.18653/v1/2022.findings-acl.281

[26] Shicheng Xu, Liang Pang, Huawei Shen, and Xueqi Cheng. 2022. Match-Prompt: Improving Multi-task Generalization Ability for Neural Text Matching via Prompt Learning. https://doi.org/10.1145/3511808.3557388 arXiv:2204.02725 [cs].

[27] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. 2022. BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models. http://arxiv.org/abs/2106.10199 arXiv:2106.10199 [cs].