Out-of-Domain Semantics to the Rescue! Zero-Shot Hybrid Retrieval Models

Tao Chen, Mingyang Zhang, Jing Lu, Michael Bendersky, and Marc Najork

Google Research, Mountain View, CA 94043, USA,
{taochen,mingyang,ljwinnie,bemike,najork}@google.com

Abstract. The pre-trained language model (*eg,* BERT) based deep retrieval models achieved superior performance over lexical retrieval models (*eg,* BM25) in many passage retrieval tasks. However, limited work has been done to generalize a deep retrieval model to other tasks and domains. In this work, we carefully select five datasets, including two in-domain datasets and three out-of-domain datasets with different levels of domain shift, and study the generalization of a deep model in a zero-shot setting. Our findings show that the performance of a deep retrieval model is significantly deteriorated when the target domain is very different from the source domain that the model was trained on. On the contrary, lexical models are more robust across domains. We thus propose a simple yet effective framework to integrate lexical and deep retrieval models. Our experiments demonstrate that these two models are complementary, even when the deep model is weaker in the out-of-domain setting. The hybrid model obtains an average of 20.4% relative gain over the deep retrieval model, and an average of 9.54% over the lexical model in three out-of-domain datasets.

Keywords: deep retrieval, lexical retrieval, zero-shot learning, hybrid model

1 Introduciton

Traditionally, search engines have used lexical retrieval models (*eg,* BM25) to perform query-document matching. Such models are efficient and simple, but are vulnerable to vocabulary mismatch when queries use different terms to describe the same concept [4]. Recently, deep pre-trained language models (*eg,* BERT) have shown strong ability in modeling text semantics and have been widely adopted in retrieval tasks. Unlike lexical retrievers, deep/dense retrievers[1] capture the semantic relevance between queries and documents in a lower dimensional space, bridging the vocabulary mismatch gaps. Deep retrievers have been successful in many retrieval benchmarks. For instance, the most recent five winners in MS-MARCO passage [2] ranking leaderboard adopt deep retrievers as their first-stage retrieval model.

[1] While we recognize that in some cases the deep retrievers are not necessarily dense, and vice versa, we loosely use these two terms interchangeably throughout the paper.

However, training a deep retrieval model is computationally expensive and a sizable labeled dataset to guide model training is not always available. A natural question then arises, can we train a deep retrieval model in one domain, and then directly apply it to new datasets/domains in a zero-shot setting with no in-domain training data? To answer this question, we carefully select five datasets, including two in-domain, and three out-of-domain datasets with different levels of domain shift. Through comprehensive experiments, we find that a deep retriever model performs well on related domains, but deteriorates when the target domain is distinct from the model source domain. On the contrary, lexical models are rather robust across datasets and domains. Our further analysis shows that lexical and deep models can be complementary to each other, retrieving different sets of relevant documents.

Inspired by this, we propose a zero-shot hybrid retrieval model to combine lexical and deep retrieval models. For simplicity and flexibility, we train a deep retrieval model and a lexical model separately and integrate the two (or more) models via Reciprocal Rank Fusion. This non-parametric fusion framework can be easily applied to any new datasets or domains, without any fine-tuning. Our experiments demonstrate the effectiveness of the hybrid model in both in-domain and out-of-domain datasets. In particular, though the zero-shot deep model is weaker in out-of-domain datasets, the hybrid model brings an average of 20.4% of relative recall gain over the deep retrieval model, and an average of 9.54% gain over lexical model (BM25) in three out-of-domain datasets. It also outperforms a variety of stronger baselines including query and document expansion.

To summarize, in this paper we explore the following research questions:

- **RQ 1**: Can deep retrieval generalize to a new domain in a zero-shot setting?
- **RQ 2** Is deep retrieval complementary to lexical matching and query and document expansion?
- **RQ 3**: Can lexical matching, expansion, and deep retrieval models be combined in a *non-parametric hybrid retrieval* model?

To the best of our knowledge, this paper is the first to propose a hybrid retrieval model that incorporates lexical matching, expansion and deep retrieval in a zero-shot setup. We demonstrate that the proposed hybrid model is simple yet effective in a variety of datasets and domains.

2 Related Work

Information retrieval systems usually contain of two main stages: (a) *candidate retrieval* (b) *candidate re-ranking*. The retrieval stage is aimed at optimizing the recall of relevant documents, while the re-ranking stage optimizes early precision metrics such as NDCG@k or MRR. Prior research (*eg,* [3]) found that the two stages are complementary – gains in retrieval recall often lead to better early precision. Therefore, in this paper, we focus on retrieval recall optimization, with the assumption that the findings can benefit the re-ranking stage as well.

Lexical retriever Traditionally, the first-stage retrieval has been a lexical based model such as BM25 [35], to capture the exact lexical match between queries and documents. Such simple and effective lexical models were the state-of-the-art for decades, and are still widely used in both academia and industry. One key issue with lexical models is the vulnerability to vocabulary mismatch, where queries and documents mention the same concept with different terms. One popular line to alleviate this is to expand terms in queries from pseudo relevance feedback (*e*g, [15] and [1]) or expand terms in documents from related documents (*e*g, [37]). As a result, queries and documents have a higher chance to match each other at the surface form.

Deep LM augmented lexical retriever More recently, pre-trained deep language models (LM) such as BERT [10] have been shown to be powerful in many natural language understanding tasks. The very first application of such models in IR is to augment lexical retrieval models. Dai et al. [7, 9] proposed to learn context-aware term weights by BERT to replace the term frequencies used by lexical models. To remedy the vocabulary gap between queries and documents, Nogueira and Lin [29, 28] employed seq2seq model transformer [39] and later T5 [33] to generate document expansions, which brings significant gains for BM25. In the same vein, Mao et al. [27] adopted seq2seq model BART [20] to generate query expansions, which outperforms RM3 [15], a highly performant lexical query expansion method.

Deep retriever In a separate line of research, deep neural retrieval models adopt LMs to build a new paradigm for first-stage retrieval: instead of performing exact lexical match, they aim at capturing the relevance of queries and documents in a lower dimensional semantic space. This paradigm can largely bridge the vocabulary gap between queries and documents. Since cross-attention models are cost-prohibitive for first-stage retrieval, most works adopt a dual-encoder architecture to learn two single vector representations for the query and the document separately, and then measure their relevance by a simple scoring function (*eg,* dot product or cosine similarity). In this way, finding most relevant documents can be formulated as a nearest neighbor search problem and can be accelerated with quantization techniques [16, 14].

For model training, it is often the case that positive (query, document) pairs are available, while negative pairs need to be sampled from the dataset. Negative sampling strategy plays a crucial role for model performance. Earlier works adopt simple in-batch negative sampling [45, 18], or mine negative pairs from top BM25 results [17]. Recent works propose more sophisticated sampling strategies to identify high-quality hard negatives, such as cross-batch negatives [32], demonised hard negatives [32] and semantic similarity based negatives [23].

Deep retriever model has shown superior performance over lexical models in several passage retrieval tasks (*e*g, MS-MARCO passage ranking [2]). However, training a deep model is expensive computationally but also in terms of labeled data creation. A simple remedy is to directly apply a trained deep retriever model to new domains in a zero-shot setting. However, little work has been conducted to uncover the generalization ability of deep retrievers. One exception is by

Thakur et al. [38] who introduce BEIR, an IR benchmark of 18 datasets with diverse domains and tasks, and evaluate several trained deep models in a zero-shot setup. They found that deep models exhibit a poor generalization ability, and are significantly worse than BM25 on datasets that have a large domain shift compared from what they have been trained on. In our work, we conduct similar studies, and observe the same performance deterioration for a deep model in zero-shot setting. We additionally propose a hybrid model to utilize a lexical model to alleviate the domain shift.

Hybrid retriever Deep retrievers are good at modeling semantic similarity, while could be weaker at capturing exact match or could have capacity issues when modeling long documents [24, 42]. A few recent works attempt to build a hybrid model to take the strength of both deep and lexical retrievers. Most works train a deep model separately and then interpolate its score with a lexical model score [42, 17, 24, 22, 25, 21], or use RM3 built on the top lexical results to select deep retriever results as the final list [18], or simply combine the results of the two models in an alternative way [45]. Gao et al. [13] is the only work that explicitly trains a deep model to encode semantics that lexical model fails to capture. In model inference, they interpolate the scores of deep and lexical models and generate the top retrieval results. While insightful, these prior works limit the model evaluation to a single task and a single domain. It is unclear how such hybrid model performs in a cross-domain setting, without any fine-tuning. Our work aims to fill this research gap, and demonstrates that a zero-shot hybrid retrieval model can be more effective than either of the two models alone.

3 Method

In this section, we describe our zero-shot hybrid retrieval model. For simplicity and flexibility, we train deep and lexical retrieval models separately, and propose a simple yet effective non-parametric framework to integrate the two.

3.1 Hybrid retrieval model

Both traditional lexical retrieval models [35, 30, 1], as well as deep neural retrieval models [13, 18, 41] represent queries and documents using vectors $\mathbf{q}, \mathbf{d} \in \mathbb{R}^N$, and score candidates based on the dot product $< \mathbf{q}, \mathbf{d} >$. Thus, the difference between deep and lexical models stems from how these vectors are constructed.

Lexical models represent queries and documents using sparse weight vectors $\mathbf{q}^{sparse}, \mathbf{d}^{sparse} \in \mathbb{R}^V$, respectively (where V denotes the vocabulary size). The vectors are sparse such that all the entries for vocabulary terms that do not appear in query and document are zeroed out. To combat issues of term mismatch, lexical models often include additional terms in queries and document through some form of expansion (*eg,* based on pseudo-relevance feedback [19]). However, the resulting vectors are still highly sparse, due to the high dimensionality of vocabulary size V.

In contrast, deep neural retrieval models represent queries and documents using dense embedding vectors $\mathbf{q}^{dense}, \mathbf{d}^{dense} \in \mathbb{R}^E$, where $E << V$. While theoretically dense embeddings overcome the term mismatch problem, they do have several shortcomings. First, they require large amounts of data and resources for training [32], and thus may not be directly trained over collections with fewer queries and relevance judgments. Second, they do not capture *exact* query-document matches as well as the sparse lexical scores. Therefore, a lexical and deep model combination is likely to yield the optimal relevance scores.

Most prior works [42, 17, 24, 23] model this combination as a linear interpolation of the scores of deep and lexical retrieval models. This fusion method is sensitive to the score scales and the weights assigned to the different models [42], which needs careful score normalization and weight tuning, especially when multiple models are combined. We expect that the raw scores of the models can vary from one domain/dataset to another, and likewise the interpolation weights.

Since our goal is to build a hybrid model which can be easily applied to a new domain in a zero shot setting (with no in-domain training data), we would like to eliminate such domain-specific normalization and tuning. Therefore, we adopt Reciprocal Rank Fusion (RRF) [5] to generate the final ranking results by considering the *ranking positions* of each candidate generated by different models, instead of fusing their scores. RRF demonstrates robust and effective ensembles in prior works [3, 5] and our experiments. Assuming a set of lexical and deep retrieval models M, we define $\pi^m(q, d)$ as the rank for document d, induced by its score for query q assigned by model $m \in M$. The RRF score is then defined as:

$$RRF(q, d, M) = \sum_{m \in M} \frac{1}{k + \pi^m(q, d)} \tag{1}$$

where $k = 60$, following the definition in the original paper [5].

In the remainder of this paper we demonstrate that this simple non-parametric approach generalizes well across domains, and can make an effective use of out-of-domain semantics of retrieval models trained on a different collection. In the remainder of this section, we describe the lexical and deep retrieval models used to instantiate Equation 1.

3.2 Lexical retrieval model

We adopt **BM25** as the base lexical retrieval model, as it is widely used and shown to be robust [38]. To alleviate the vocabulary mismatch issue, we additionally apply popular query expansion and document expansion techniques to expand the query and the document, forming enhanced lexical models.

BM25+Query expansion Most conventional query expansion approaches follow the pseudo-relevance feedback (PRF) paradigm. It assumes the top K ranked documents for the original query to be relevant, and generates query expansions from these documents. In our work, we experiment with **RM3** [15] (a relevance-based language model) and **Bo1** [1] (a variant of Divergence From Randomness term weighting model), to obtain query expansions from PRF.

BM25+Document expansion Recently, generative models like T5 were shown to generate high-quality document expansions, and bring large gains to the BM25 model on retrieval tasks [29, 28, 31]. Following the **docT5query** approach [28, 31], we fine-tune T5-base with identical setting as the prior works on (query, relevant passage) pairs from the MS-MARCO passage ranking training set, where the query is considered as pseudo document expansion. We adopt the top-k sampling decoder [11] to generate N (a tunable parameter) queries per passage. For each document, we append the expansions to each passage and aggregate them as the document expansion.

3.3 Deep retrieval model

We adopt **NPR** [23], a neural passage retrieval model with improved negative contrast as the deep retrieval model in our framework. Note that our framework is flexible, and NPR can be replaced with any other deep model. Aligned with many popular deep retrievers [17, 32, 43], NPR adopts a dual encoder architecture, learning dense embedding vectors representations, computing the relevance using the dot product $< \mathbf{q}^{dense}, \mathbf{d}^{dense} >$. The training of this model is enhanced with several negative sampling strategies, aiming at obtaining hard and high-quality negative (query, passage) pairs. This model is trained on MS-MARCO passage dataset (detailed in Section 4.1), and achieves a very competitive performance. To adapt NPR to document retrieval setting, we split documents into passages by applying sliding overlapping sentence windows. Following work by Dai and Callan [8], we use the max passage retrieval score as the document level score.

4 Experimental Setup

4.1 Datasets

As we are interested in exploring the performance of the deep retrieval model in a variety of out-of-domain settings, we choose to specifically focus on five datasets in our evaluation (summarized in Table 1).

1. **MS-MARCO passage** [2] dev set is the dataset we use for the *in-domain* model evaluation, as the NPR deep retrieval model, and the docT5query model are trained using the training portion of this dataset (see Section 4.2 for more details). The queries in this dataset are all questions.
2. **MS-MARCO doc** [2] is derived from MS-MARCO passage, but instead the retrieval is done using documents. We use the queries in dev set for evaluation (a subset of MS-MARCO passage dev set). This evaluates the generalization of the model to document retrieval.
3. **ORCAS** [6] is a click dataset based on an intersection of Bing search engine logs and the documents in MS-MARCO dataset. Compared to MS-MARCO, queries in ORCAS exhibit wider topics (not limited to questions) and shorter length (76% of queries have no more than 3 tokens after removing stopwords). Since it has a very large number of queries (10M), we evaluate our model using a stratified sample of 10k queries, based on query length.

Table 1. The five datasets used for model evaluation. "Avg. D/Q" denotes the average number of relevant docs per query.

Dataset	Domain	Task	#Query	#Corpus	Avg. D/Q
MS-MARCO passage [2]	Misc.	Passage retrieval	6980	8.8M	1.1
MS-MARCO doc [2]	Misc	Doc retrieval	5193	3.2M	1.1
ORCAS [6]	Misc.	Doc retrieval	9670	1.4M	1.8
Robust04 [40]	News	Doc retrieval	250	528K	69.9
TREC-COVID [34]	Bio-medical	Doc retrieval	50	191K	493.5

4. **Robust04** [40] is a dataset comprising 528K news stories and 250 queries. Each query consists of three fields, including title (keywords), description (a sentence-length statement of the information needs) and narrative (a paragraph-length text explaining what makes a document relevant). It evaluates how well the retrieval model generalizes to the news domain.
5. **TREC-COVID** [34] is based on the CORD19 [41] collection – PubMed articles and preprints about the COVID-19 pandemic. Each query contains a few keywords, along with a more specific natural language version of question, and a narrative which adds additional clarifications of user intent. As shown by Thakur et al. [38], it is quite distinct from the MS-MARCO dataset, and provides a good test case for whether an out-of-domain retrieval system can be useful in a bio-medical domain.

4.2 Data processing and benchmarking

In following, we detail our experimental setup to ensure the reproducibility of all the reported results.

Deep retrieval model As described in Section 3, we train NPR on the training set of MS-MARCO passage dataset, and apply this model to the other four datasets without any fine-tuning. The documents in the other four datasets are long and may exceed the 512 token length limitation. Following prior work [31], we use a sliding window of ten sentences with a stride of five to split each document into passages. We run NPR on each passage, perform the nearest neighbor search via SCaNN [14] at passage-level and consider the best passage score as its document score. The query used for each dataset is the same as BM25 based lexical model (detailed in Table 2).

Lexical retrieval models For implementing our lexical models, we use the Terrier search engine [26], and apply the default options for stemming and stop word removal provided by Terrier. We employ three fully lexical benchmarks. We carefully tune the parameters, and detail the settings in Table 2.

– BM25 is a commonly used bag-of-words retrieval method. We use the default parameters provided by Terrier, and verify that our results (in terms of MAP) are comparable to other previously reported BM25 benchmarks [44]. We experiment with a few indexing options: 1) full text, 2) passage and 3) abstract for TREC-COVID only.

Table 2. The best setup for lexical retrieval models. "des./narr./ques." denotes description/narrative/question field and "#fk docs/terms" denotes the number of feedback documents/terms.

Model (→)	BM25		Bo1		docT5query
Dataset (↓)	index	query	#fk doc	#fk terms	#expansions
MS-MARCO passage	full text	query	5	10	40
MS-MARCO doc	full text	query	5	5	20/passage
ORCAS	full text	query	10	10	20/passage
Robust04	full text	query+des.+narr.	5	10	10/passage
TREC-COVID	abstract	query+ques.+narr.	20	40	10

- Bo1 is a query expansion package implemented in Terrier. For each dataset, we carefully tune the number of feedback documents ([5, 10, ..., 50]) and the number of feedback terms (*ie*, expansions; [5, 10, ..., 60]). We also experiment with RM3 query expansion package by Terrier and carefully tune the two parameters. However, it yields lower performance than Bo1 in all the five datasets. We thus only report the results of Bo1 in the Section 5.
- docT5query is a T5 based document expansion model. As described in Section 3, we fine-tune T5 model on the MS-MARCO passage training set by strictly following the setup of prior works [28, 31]. We feed each passage length text, namely, passage in the MS-MARCO passage collection, the abstract in TREC-COVID, or split passages of other three datasets, to T5 model and generate N (a tunable parameter; [10, 20, 40]) numbers of expansions. We append the expansions for all the passages to a document.

5 Evaluation

As our work focuses on the first stage retrieval, in this section we adopt Recall@1K as the primary evaluation metric and additionally report MAP score. In our evaluation, we aim to address the research questions posed in Section 1.

5.1 Generalization of the deep retrieval model

We first focus on the results on two in-domain datasets (Table 3). As expected, the deep retrieval model NPR performs very well on MS-MARCO passage on which it is trained. It substantially beats BM25 by an absolute 10.77 (relative 12.35%) and 16.15 (relative 83.59%) in terms of Recall@1K and MAP, respectively. In MS-MARCO doc (the in-domain document retrieval task), NPR also performs well, and betters BM25 by 4.55 (5.0%) and 3.86 (14.57%) at Recall@1K and MAP, respectively. This indicates that a well-trained deep passage retrieval model generalizes well to an in-domain document retrieval task.

In Table 4, we discuss the results of three out-of-domain document retrieval datasets. Compared to MS-MARCO doc, ORCAS dataset has the least domain shift (as the candidate documents stem from MS-MARCO doc albeit with different queries), followed by Robust04 (news domain). TREC-COVID contains

Table 3. Experimental results on two in-domain datasets. The improvements (R@1K) of all hybrid models (5-8) over baselines (1-4) are statistically significant via a paired two-tailed t-test ($p < 0.05$).

Dataset (→)	MS-MARCO passage		MS-MARCO doc	
Model (↓)	R@1K	MAP	R@1K	MAP
1. BM25	87.18	19.32	90.91	26.50
2. BM25+Bo1	88.27	17.95	91.64	22.69
3. BM25+docT5query	94.07	26.09	93.18	30.28
4. NPR	97.95	**35.47**	95.46	30.36
5. RRF(1, 4)	98.31	29.46	96.80	32.09
6. RRF(2, 4)	98.36	28.62	96.90	31.48
7. RRF(3, 4)	**98.65**	32.89	96.86	**33.50**
8. RRF(2, 3, 4)	98.48	29.58	**96.96**	32.48

Table 4. Experimental results on three out-of-domain datasets. The improvements (R@1K) of all hybrid models (5-8) over baselines (1-4) are statistically significant via a paired two-tailed t-test ($p < 0.05$), except 5/7 vs. 2 in Robust04 and TREC-COVID.

Dataset (→)	ORCAS		Robust04		TREC-COVID	
Model (↓)	R@1K	MAP	R@1K	MAP	R@1K	MAP
1. BM25	77.52	27.1	72.84	26.91	49.29	27.86
2. BM25+Bo1	78.85	23.53	79.02	30.83	52.58	30.98
3. BM25+docT5query	79.62	30.28	74.64	28.01	50.66	28.77
4. NPR	81.18	28.29	70.28	28.39	37.58	17.14
5. RRF(1, 4)	85.95	30.33	79.62	33.19	52.32	30.38
6. RRF(2, 4)	86.18	28.36	**82.82**	**34.60**	54.63	32.21
7. RRF(3, 4)	86.44	**31.39**	79.81	33.34	53.01	30.64
8. RRF(2, 3, 4)	**86.49**	29.74	82.65	34.51	**55.66**	**34.22**

COVID-19 specific topics and has the largest domain shift. We observe that NPR has a clear performance drop with the increased domain shift. NPR performs reasonably on ORCAS, and betters BM25 by relative 4.72% and 4.39% at Recall@1K and MAP, respectively. However, it still has an absolute drop of 14.28 and 2.07 in terms of Recall@1K and MAP, compared to its performance on MS-MARCO doc. In the news domain (Robust04 dataset), the performance of NPR is mixed: it outperforms BM25 by 5.5% of relative MAP improvement, but underperforms by relative 3.51% at Recall@1K. In TREC-COVID dataset, BM25 significantly beats NPR by 11.71 (23.76%) and 10.72 (38.48%) in terms of Recall@1K and MAP. This demonstrates that the generalization ability of deep retrieval models is poor, especially when the target domain is dramatically different from its training domain.

5.2 Utility of query and document expansion

Lexical retrieval models are prone to vocabulary mismatch between queries and documents. We examine whether query and document expansion models could

bridge this gap. From Table 3 and Table 4, we see that Bo1 query expansion model consistently brings recall gains, with 1% relative gain on MS-MARCO passage/doc and ORCAS, 8.48% on Robust04 and 6.67% on TREC-COVID.

Recall that docT5query document expansion model is trained on the training set of MS-MARCO passage dataset. In this dataset, it brings very large gains to BM25. In the other four datasets, docT5query shows a consistent, albeit smaller, improvement over BM25 (above 2.5% recall gain), similar to the analysis by Thakur et al. [38].

5.3 Complementarity of lexical and deep retrieval models

As with query/document expansion, deep retrieval model can narrow the vocabulary gap between queries and documents. One natural question is, are these models still complementary to each other? To answer this, we plot the unique relevant documents retrieved by BM25+Bo1, BM25+docT5query and NPR and their overlaps in Figure 1 for Robust04 and TREC-COVID (other three datasets only have around one relevance document per query, ref Table 1). We see that each method is complementary to each other. In general, NPR retrieves the largest number of unique relevant results, though it retrieves less relevant results than the other two methods.

5.4 Effectiveness of the proposed hybrid model

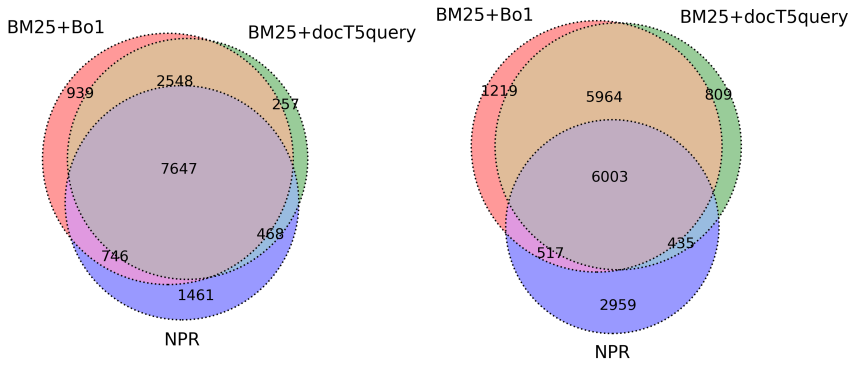
Our proposed hybrid framework provides a flexible mechanism for fusing multiple lexical or deep retrieval models. In Table 3 and Table 4 (row 5-8), we demonstrate the performance of our hybrid model which consistently outperforms either the lexical or deep retrieval model alone. In in-domain MS-MARCO passage dataset, the best performing hybrid model of BM25+docT5query and NPR (#7) obtains a Recall@1K of 98.65, betters BM25 and NPR by relative 12.94% and 0.52%, respectively. This hybrid model outperforms coCondenser (Recall@1K=98.4) [12], the current MS-MARCO leaderboard winner (as of 2021/08/09) in the passage retrieval task.[2] In the in-domain document retrieval task, the best performing hybrid model is the one with all the three methods (Bo1, docT5query and NPR).

In three out-of-domain datasets, the advantage of hybrid model is more evident, given that NPR is weakened in datasets with a large domain shift (ie, TREC-COVID). It consistently improves over BM25 by almost 10% relatively for the three datasets, and substantially outperforms NPR by 6.11%, 14.16% and 44.25% in ORCAS, Robust04, and TREC-COVID, respectively. This demonstrates that our proposed zero-shot hybrid retrieval model is effective and robust across different tasks and domains.

6 Discussion

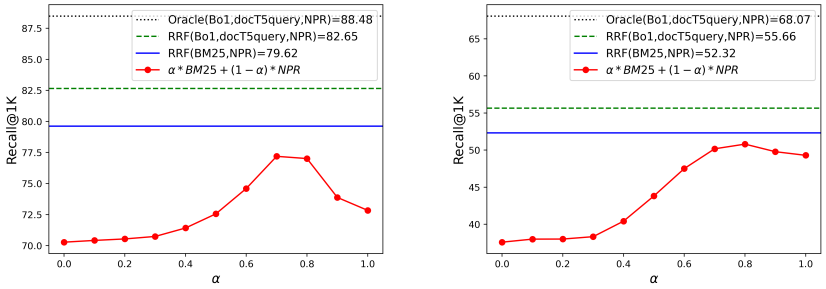
Our zero-shot hybrid model has demonstrated its effectiveness in the experiments. For comparison, we implement the linear interpolation method that

[2] Note that we focus solely on recall, since we do not apply a second re-ranking stage for optimizing early precision.



(a) Robust04 (b) TREC-COVID

Fig. 1. A Venn diagram of relevant results by the Bo1, docT5query, and NPR.



(a) Robust04 (b) TREC-COVID

Fig. 2. The comparisons of our hybrid model, oracle system and interpolation.

most prior works adopted [42, 17, 24, 23], though such model is not zero-shot, and requires weight tuning. As weight tuning complexity increases with the number of models, we only interpolate BM25 and NPR as a case study: $s(d) = \alpha \times s_{BM25}(d) + (1 - \alpha) \times s_{NPR}(d)$. We perform min-max score normalization and carefully tune the weight $\alpha \in [0.1, \ldots, 0.9]$ via grid search for out-of-domain datasets Robust04 and TREC-COVID.

Figure 2 (bottom curve), shows that interpolation is weight-sensitive, and furthermore even the best setting underperforms our simple non-parametric hybrid model RRF(BM25, NPR) by a relative 3% in both datasets. The differences are even larger, when compared with the full RRF model (dashed line). We also explore the hybrid upper bound by fusing the retrieval results of BM25+Bo1, BM25+docT5query and NPR via an oracle, *ie,* merging all relevant results from each method regardless of their ranking positions. Figure 2 (dotted top line) illustrates the large potential headroom for designing an even better fusion model.

Table 5. Mean R@1K result by query length for ORCAS dataset (best result bolded).

Model \Query Length	1	2	3	4	5	6	7	8	9	10
1. BM25	35.0	74.2	80.2	82.0	82.3	82.9	81.1	87.8	85.6	87.5
2. BM25+Bo1	39.5	76.6	81.3	83.1	83.3	83.0	82.6	88.1	**86.3**	**87.9**
3. BM25+docT5query	36.9	77.8	83.0	84.1	86.2	84.9	**83.8**	**88.6**	85.8	87.8
4. NPR	**59.8**	**82.5**	**85.8**	**86.6**	**87.6**	**85.6**	82.8	83.3	80.1	75.9

Similarly to us, Wang et al. [42] found that setting an oracle per-query weight yields better performance than optimizing a global weight. Inspired by this, we hypothesize that the performance of retrieval models relate to query length. We bin the ORCAS queries into 10 groups, based on the number of non-stopword tokens, and show the breakdown results in Table 5. When the queries are very short, NPR largely beats BM25, even with query and document expansion. However, its performance deteriorates for longer queries, with 7 or more tokens.

To gain more insights, we spot-check wins and losses. For single token queries, BM25 performs badly when the query is misspelled (*eg*, "ihpone6") or a compound word (*eg*, "tvbythenumbers"). These words are very likely to be out-of-vocabulary (OOV) in lexical retrieval models. On the contrary, deep retrieval model NPR adopts wordpiece tokenizer, which could still capture the semantics of the OOV from its sub-units. For long queries, NPR performs poorly for those employing complex logic and seeking very specific information, *eg*, "according to piaget, which of the following abilities do children gain during middle childhood?". In this example query, BM25 successfully retrieves relevant documents containing the identical query sentence, while NPR fails. This may indicate that NPR is worse at capturing exact match, consistently with prior work [42, 24].

7 Conclusion

Compared to traditional lexical retrieval models, a deep retrieval model mitigates the vocabulary mismatch by modeling semantic relevance between queries and documents, and has a great success in many retrieval tasks. We show that a deep retrieval model poorly generalizes to a new domain with large domain shift, while lexical matching and expansion models are robust across domains. To address this, we propose a simple non-parametric zero-shot hybrid model to integrate lexical matching, expansion, and deep retrieval models. Our proposed model demonstrates its effectiveness in both in-domain and out-of-domain datasets.

A recent work [36] found that deep retrieval models underperform lexical models for rare entities in an entity-centric QA task. As a future work, we plan to investigate the effectivenss of our hybrid model in this task. Additionally, we plan to parameterize the hybrid retrieval model using query structure, query length, the degree of domain shift, and other signals that may reflect the performance of each individual model. Finally, we plan to explore techniques that improve the utility of out-of-domain deep retrieval models via domain adaptation.

References

1. Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357–389, October 2002.
2. Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. Ms marco: A human generated machine reading comprehension dataset, 2018.
3. Michael Bendersky, Honglei Zhuang, Ji Ma, Shuguang Han, Keith B. Hall, and Ryan T. McDonald. RRF102: meeting the TREC-COVID challenge with a 100+ runs ensemble. *CoRR*, abs/2010.00200, 2020.
4. Adam Berger, Rich Caruana, David Cohn, Dayne Freitag, and Vibhu Mittal. Bridging the lexical chasm: Statistical approaches to answer-finding. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, page 192–199, New York, NY, USA, 2000. Association for Computing Machinery.
5. Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, page 758–759, New York, NY, USA, 2009. Association for Computing Machinery.
6. Nick Craswell, Daniel Campos, Bhaskar Mitra, Emine Yilmaz, and Bodo Billerbeck. Orcas: 20 million clicked query-document pairs for analyzing search. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, page 2983–2989, New York, NY, USA, 2020. Association for Computing Machinery.
7. Zhuyun Dai and Jamie Callan. Context-aware sentence/passage term importance estimation for first stage retrieval. *CoRR*, abs/1910.10687, 2019.
8. Zhuyun Dai and Jamie Callan. Deeper text understanding for ir with contextual neural language modeling. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 985–988, New York, NY, USA, 2019. Association for Computing Machinery.
9. Zhuyun Dai and Jamie Callan. Context-aware term weighting for first stage passage retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1533–1536, New York, NY, USA, 2020. Association for Computing Machinery.
10. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics.
11. Angela Fan, Mike Lewis, and Yann N. Dauphin. Hierarchical neural story generation. *CoRR*, abs/1805.04833, 2018.
12. Luyu Gao and Jamie Callan. Unsupervised corpus aware language model pre-training for dense passage retrieval. *CoRR*, abs/2108.05540, 2021.
13. Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. Complement lexical retrieval model with semantic residual embeddings. In Djoerd Hiemstra, Marie-Francine Moens, Josiane Mothe, Raffaele Perego, Martin

Potthast, and Fabrizio Sebastiani, editors, *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part I*, volume 12656 of *Lecture Notes in Computer Science*, pages 146–160. Springer, 2021.

14. Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating large-scale inference with anisotropic vector quantization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3887–3896. PMLR, 2020.
15. Nasreen Abdul Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah S. Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. Umass at TREC 2004: Novelty and HARD. In Ellen M. Voorhees and Lori P. Buckland, editors, *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004*, volume 500-261 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2004.
16. Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547, 2021.
17. Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 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, November 2020. Association for Computational Linguistics.
18. Saar Kuzi, Mingyang Zhang, Cheng Li, Michael Bendersky, and Marc Najork. Leveraging semantic and lexical matching to improve the recall of document retrieval systems: A hybrid approach. *CoRR*, abs/2010.01195, 2020.
19. Victor Lavrenko and W. Bruce Croft. Relevance based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, page 120–127, New York, NY, USA, 2001. Association for Computing Machinery.
20. Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics.
21. Jimmy Lin and Xueguang Ma. A few brief notes on deepimpact, coil, and a conceptual framework for information retrieval techniques. *CoRR*, abs/2106.14807, 2021.
22. Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 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-2021)*, pages 163–173, Online, August 2021. Association for Computational Linguistics.
23. Jing Lu, Gustavo Hernández Ábrego, Ji Ma, Jianmo Ni, and Yinfei Yang. Multi-stage training with improved negative contrast for neural passage retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6091–6103, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.
24. Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. Sparse, Dense, and Attentional Representations for Text Retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345, 04 2021.

25. Xueguang Ma, Kai Sun, Ronak Pradeep, and Jimmy Lin. A replication study of dense passage retriever. *CoRR*, abs/2104.05740, 2021.
26. Craig Macdonald, Richard McCreadie, Rodrygo LT Santos, and Iadh Ounis. From puppy to maturity: Experiences in developing terrier. *Proc. of OSIR at SIGIR*, pages 60–63, 2012.
27. Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. Generation-augmented retrieval for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4089–4100, Online, August 2021. Association for Computational Linguistics.
28. Rodrigo Nogueira and Jimmy Lin. From doc2query to doctttttquery. Online, 2019.
29. Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. Document expansion by query prediction. *CoRR*, abs/1904.08375, 2019.
30. Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 275–281, New York, NY, USA, 1998. Association for Computing Machinery.
31. Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *CoRR*, abs/2101.05667, 2021.
32. Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online, June 2021. Association for Computational Linguistics.
33. Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.
34. Kirk Roberts, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, Kyle Lo, Ian Soboroff, Ellen Voorhees, Lucy Lu Wang, and William R Hersh. 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, 07 2020.
35. Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at trec-3. In *TREC*, 1994.
36. Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. Simple entity-centric questions challenge dense retrievers. *CoRR*, abs/2109.08535, 2021.
37. Tao Tao, Xuanhui Wang, Qiaozhu Mei, and ChengXiang Zhai. Language model information retrieval with document expansion. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 407–414, New York City, USA, June 2006. Association for Computational Linguistics.
38. Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 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*, 2021.
39. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is all

you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.
40. Ellen Voorhees. Overview of the trec 2004 robust retrieval track, 2005-08-01 2005.
41. 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. CORD-19: The COVID-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online, July 2020. Association for Computational Linguistics.
42. Shuai Wang, Shengyao Zhuang, and Guido Zuccon. Bert-based dense retrievers require interpolation with bm25 for effective passage retrieval. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '21, page 317–324, New York, NY, USA, 2021. Association for Computing Machinery.
43. Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*, 2021.
44. Peilin Yang, Hui Fang, and Jimmy Lin. Anserini: Reproducible ranking baselines using lucene. *J. Data and Information Quality*, 10(4), October 2018.
45. Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. Repbert: Contextualized text embeddings for first-stage retrieval. *CoRR*, abs/2006.15498, 2020.