Fusion-in-T5: Unifying Variant Signals for Simple and Effective Document Ranking with Attention Fusion

Shi Yu[1†], **Chenghao Fan**[2†], **Chenyan Xiong**[3], **David Jin**[4], **Zhiyuan Liu**[1*], **and Zhenghao Liu**[5*]
[1]NLP Group, DCST, IAI, BNRIST, Tsinghua University, Beijing, China
[2]School of Comp. Sci. & Tech., Huazhong University of Science and Technology, Wuhan, China
[3]Language Technologies Institute, CMU, Pittsburgh, PA, USA
[4]Laboratory for Information and Decision Systems, MIT, Cambridge, MA, USA
[5]Department of Computer Science and Technology, Northeastern University, Shenyang, China
yus21@mails.tsinghua.edu.cn, facicofan@gmail.com, cx@cs.cmu.edu,
jindavid@mit.edu, liuzy@tsinghua.edu.cn, liuzhenghao@mail.neu.edu.cn

Abstract

Common document ranking pipelines in search systems are cascade systems that involve multiple ranking layers to integrate different information step-by-step. In this paper, we propose a novel re-ranker Fusion-in-T5 (FiT5), which integrates text matching information, ranking features, and global document information into one single unified model via templated-based input and global attention. Experiments on passage ranking benchmarks MS MARCO and TREC DL show that FiT5, as one single model, significantly improves ranking performance over complex cascade pipelines. Analysis finds that through attention fusion, FiT5 jointly utilizes various forms of ranking information via gradually attending to related documents and ranking features, and improves the detection of subtle nuances. Our code is open-sourced at https://github.com/OpenMatch/FiT5.

Keywords: document ranking, attention, fusion

1. Introduction

Document ranking in information retrieval (IR) uses signals from many sources: text matching between queries and documents (Nogueira and Cho, 2019), numerical ranking features (Zhang et al., 2021), and pseudo relevance feedback (PRF) from other retrieved documents (Li et al., 2023). These signals capture different aspects of relevance and are currently modeled by different techniques, such as neural ranking (Mitra and Craswell, 2017), learning to rank (LeToR) (Liu et al., 2009), and query expansion (Carpineto and Romano, 2012).

Current search systems often incorporate these ranking techniques by a cascade pipeline with multiple layers of ranking models, each capturing a certain type of ranking signals (Yates et al., 2021; Zhang et al., 2021; Dai et al., 2018). For example, the retrieved documents can be first re-ranked by a BERT ranker for text matching (Nogueira and Cho, 2019), then a LeToR model to combine numerical features (Zhang et al., 2020, 2021, 2022; Dai et al., 2018), and finally re-ranked again by matching with the query enriched by top-ranked global documents (Zheng et al., 2020; Yu et al., 2021; Li et al., 2023). Effective it is, the multi-layered cascade pipeline is complicated to tune, and the barrier between layers inevitably restricts the optimization of search relevance.

In this paper, we introduce Fusion-in-T5 (FiT5), a T5-based (Raffel et al., 2020) ranking model that re-ranks documents within a unified framework using attention fusion mechanism. FiT5 is designed to consolidate multiple signals, including text matching, ranking features, and global document information, into a single, simple model. Specifically, we pack the input to FiT5 using a template that incorporates the document text with the ranking feature. Furthermore, we introduce global attention layers on the representation tokens from the late layers of FiT5 encoders, enabling FiT5 to make comprehensive decisions by considering the collective information across top-ranked documents. With such a design, FiT5 can integrate all aforementioned types of signals and naturally learn a unified model through end-to-end training.

Experimental results on widely-used IR benchmarks MS MARCO (Nguyen et al., 2016) and TREC DL 2019 & 2020 (Craswell et al., 2020, 2021), show that FiT5 exhibits substantial improvements over traditional re-ranking pipelines. On MS MARCO, FiT5 further outperforms Expando-Mono-Duo (Pradeep et al., 2021), a multi-stage re-ranking pipeline by 4.5%. Further analysis reveals that FiT5 effectively leverages ranking features through the attention fusion mechanism. It can better differentiate between similar documents and ultimately produce a better ranking result.

[†] Equal contribution.
[*] Corresponding authors.

LREC-COLING 2024, pages 7556–7561
20-25 May, 2024.

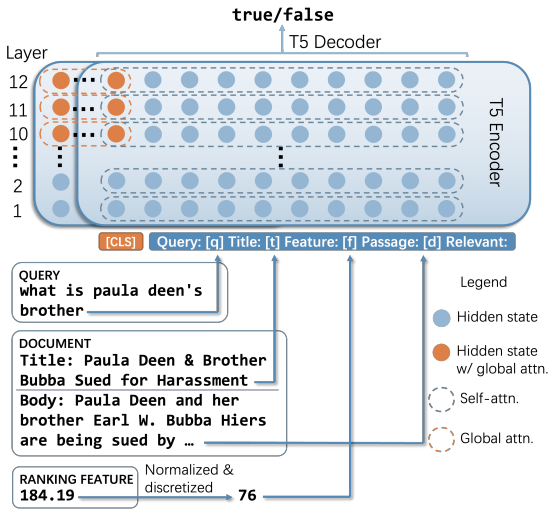


Figure 1: Architecture of Fusion-in-T5. The query, document, and ranking feature are filled in the template to form the input. In this paper, we use the retrieval score as the ranking feature.

2. Related Work

Cascade Document Ranking Pipeline A full cascade ranking pipeline may consist of one/multiple classical/neural ranker(s), LeToR model(s), and a stage of re-ranking with information from top-ranked documents. Rankers include retrievers and re-rankers based on vocabulary matching (e.g. BM25) or neural networks. Neural rankers are composed of deep neural networks (Xiong et al., 2017) or pre-trained language models (Nogueira and Cho, 2019; Yates et al., 2021), optimized with large amounts of data. A Learning-to-Rank (LeToR) model, such as a linear combination model (Metzler and Bruce Croft, 2007) or neural network (Han et al., 2020; Burges et al., 2005), utilizes machine learning to generate a relevance score by considering ranking features extracted from the data or rankers. Documents are finally re-ranked with collective information of all candidate documents, often accomplished by expanding the query with additional information via pseudo relevance feedback (PRF) (Yu et al., 2021; Li et al., 2023) or strengthening document-wise interaction through neural networks (Pradeep et al., 2021; Zhang et al., 2022). Though effective, these cascade methods require careful engineering and may be hard to optimize.

Attention Fusion over Multiple Text Sequences Fusion-in-Decoder (FiD) (Izacard and Grave, 2021) adds a T5 decoder model on top of multiple T5 document encoders to fuse multiple text evidences through the decoder-encoder attention and generate the answer for open-domain QA. Transformer-XH (Zhao et al., 2020) builds eXtra Hop attention across the text evidences inside the BERT layers to model the structure of texts for multi-hop QA. In this paper, we leverage the similar idea from Transformer-XH and propose attention fusion to incorporate variant ranking signals for the document ranking task.

3. Methodology

In this section, we first present the overview of FiT5 in §3.1, then discuss the input and output format in §3.2 and the attention fusion in §3.3.

3.1. Task and Model Overview

Given a query q, a re-ranking model ranks a set of n candidate documents $D = \{d_1, d_2, ..., d_n\}$ from first-stage retrieval by assigning them with a set of scores $S = \{s_1, ..., s_n\}$. Traditional re-ranking model accomplishes the re-ranking task by making point-wise predictions, i.e. $S = \{s_1, ..., s_n\} = \{f(q, d_1), ..., f(q, d_n)\}$, with the query q, the document d_i $(i = 1, ..., n)$, and the model f. FiT5 makes a more comprehensive prediction by deciding globally with more features, i.e. $S = \text{FiT5}(q, D, R)$, where $R = \{r_1, ..., r_n\}$ is the set of ranking features for all documents.

FiT5 is based on the encoder-decoder model T5 (Raffel et al., 2020), as shown in Figure 1. The encoder takes a triple of (q, d_i, r_i) as the input. Attention fusion is introduced in the late layers of the encoder as global attention layers to incorporate signals from other documents in D. The final ranking score s_i is decoded from the decoder.

3.2. Input and Output

We pack (q, d_i, r_i) using a template to form the input to FiT5. The template consists of slots for input data and several prompt tokens, defined as

Query: `[q]` Title: `[t]` Feature: `[f]` Passage: `[d]` Relevant:,

where `[q]`, `[t]` and `[d]` are slots for text features, corresponding to the query q, the title and the body of the document d_i, respectively. `[f]` is the slot for the feature r_i. In this paper, we use the retrieval score as the ranking feature, after min-max normalization and discretization.

The model is fine-tuned to decode the token "true" or "false" according to the input. During inference, the final relevance score is obtained from the normalized probability of the token "true".

3.3. Attention Fusion via Global Attention

The global document set D and its feature set R may contain valuable information for generating the score for every document d_i, which cannot be

Model	Re-ranker PLM(s)	# Params of Re-ranker(s)	MS MARCO MRR@10	MS MARCO MAP@10	TREC DL'19 NDCG@10	TREC DL'19 MRR	TREC DL'20 NDCG@10	TREC DL'20 MRR
First Stage Retrieval								
BM25	n.a.	n.a.	18.7	19.5	50.6	70.4	48.0	65.9
ANCE (2021)	n.a.	n.a.	33.0	–	64.8	–	64.6	–
ANCE-PRF (2021)	n.a.	n.a.	34.4	–	68.1	–	69.5	–
coCondenser (2022)	n.a.	n.a.	38.3	37.6	71.5	86.8	68.0	84.4
*Two-stage Ranking (coCondenser → *)*								
BERT Re-ranker (2019)	BERT-base	110M	39.2^{c}	38.6^{c}	70.1	83.8	69.2	82.3
monoT5 (2020)	T5-base	220M	40.6^{cb}	39.9^{cb}	72.6	84.8	67.7	85.1
FiT5 (ours)	T5-base	227M	**43.9**^{cbm}	**43.3**^{cbm}	**77.6**^{cbm}	**87.4**	**75.2**^{cbm}	**85.5**
Multi(≥3)-stage Ranking (For Reference)								
HLATR-base (2022)	RoBERTa-base	132M	42.5	–	–	–	–	–
HLATR-large (2022)	RoBERTa-large	342M	43.7	–	–	–	–	–
Expando-Mono-Duo (2021)	2×T5-3B	2×3B	42.0	–	–	–	78.4	88.0

Table 1: Overall results on MS MARCO and TREC DL 19 & 20. Superscripts c, b, and m indicate statistically significant improvements over coCondenser, BERT Re-ranker, and monoT5 (permutation test; $p < 0.05$). Inapplicable and unavailable results are marked by "n.a." and "–", respectively.

captured via point-wise inference over the "local" information (q, d_i, r_i) (Yu et al., 2021). To enhance the effectiveness of ranking, we propose attention fusion in FiT5 to enable the model to better comprehend and differentiate these documents with their features in the ranking process.

In FiT5, each (q, d_i, r_i) pair first runs through $l-1$ transformer encoder layers independently, as in vanilla T5. The attention fusion mechanism is enabled in every layer $j \geq l$. The representation of the first token [CLS] (prepended to the input), denoted as $h^j_{i,[\text{CLS}]} \in R^c$, is picked out from the normal self-attention:

$$h^j_{i,[\text{CLS}]}, \hat{\mathbf{H}}^j_i = \text{Transformer}(\mathbf{H}^{j-1}_i), \tag{1}$$

where $\hat{\mathbf{H}}^j_i$ denotes the remaining part of the hidden representation, c is the hidden size and Transformer is the transformer layer. The representations of the first tokens from all n encoders are then fed into a *global* attention layer, allowing fusion over non-local information:

$$\begin{aligned} &\hat{h}^j_{1,[\text{CLS}]}, ..., \hat{h}^j_{n,[\text{CLS}]} \\ =&\text{Global\_Attention}(h^j_{1,[\text{CLS}]}, ..., h^j_{n,[\text{CLS}]}). \end{aligned} \tag{2}$$

Finally, the globally-attended representation $\hat{h}^j_{i,[\text{CLS}]}$ is added back to the hidden representation:

$$\mathbf{H}^j_i = [h^j_{i,[\text{CLS}]} + \hat{h}^j_{i,[\text{CLS}]}; \hat{\mathbf{H}}^j_i]. \tag{3}$$

In this way, the information from other documents in D and features in R is modeled in the representation of the [CLS] token and is then propagated to the following layer(s) in the encoder.

4. Experimental Methodology

In this section, we discuss our experimental setup.

Datasets and Metrics We train FiT5 on MS MARCO passage ranking dataset (Nguyen et al., 2016) and evaluate it on its development set and TREC Deep Learning Tracks (TREC DL) 2019 & 2020 (Craswell et al., 2020, 2021). MS MARCO labels are binary sparse labels (0/1) with often one positive document per query. TREC DL labels are dense judgments on a four-point scale from irrelevant (0) to perfectly relevant (3) and thus are more comprehensive (Craswell et al., 2020, 2021). We report MRR@10, MAP@10 on MS MARCO, and NDCG@10, MRR on TREC DL.

Implementation We use T5-base model (Raffel et al., 2020) as the backbone of our model. Global attention modules are added starting from the third to last layer (i.e. $l = 10$) of the T5 encoder, implemented as standard multi-head attention with 12 attention heads. We re-rank the top 100 retrieved documents from coCondenser (Gao and Callan, 2022) and use coCondenser retrieval score as the ranking feature in the template defined in § 3.2. Specifically, we first normalize the coCondenser scores using min-max normalization and then discretize them into integers in $[0, 100]$ to serve as the input. We first train FiT5 without the features for 400k steps and then train it with the ranking feature for 1.5k steps to obtain the final model.

Baselines We compare FiT5 with typical two-stage retrieve-and-rerank pipelines including BERT Re-ranker (Nogueira and Cho, 2019) and monoT5 (Nogueira et al., 2020). These re-rankers are trained to assign a score for each (q, d_i) text pair individually. The first-stage retrieval for such pipelines is kept the same as it for FiT5. We also report the performance of multi(≥3)-stage ranking pipelines including HLATR (Zhang et al., 2022), a list-aware ranking pipeline and Expando-Mono-Duo (Pradeep et al., 2021), a sophisticated ranking system that employs pairwise comparison. The performance of common first-stage retrieval models are also reported.

	Time	Memory
monoT5	19m35s	6088MiB
FiT5	19m37s	6362MiB

Table 2: Inference time and GPU memory usage of FiT5 and monoT5 on MS MARCO dev set, measured on a single NVIDIA A100 40G GPU with a batch size of 100 question-document pairs per step.

Model	MARCO	DL'19	DL'20
monoT5	40.56	72.55	67.73
monoT5 (w/ feature)	40.95^{m}	72.12	68.73
FiT5 (w/o feature)	42.79^{mw}	74.94^{w}	70.02
FiT5 (linear combination)	43.59^{mwo}	75.41^{mw}	70.95^{mw}
FiT5	**43.93**^{mwo}	**77.63**^{mw}	**75.24**^{mwol}

Table 3: Contribution of attention fusion. The evaluation metric is MRR@10 on MS MARCO and NDCG@10 on TREC DL. (permutation test; $p < 0.05$)

5. Evaluation Results

This section presents the overall results of FiT5, and analyzes its effectiveness.

5.1. Overall Performance

The results of passage ranking on MS MARCO and TREC DL are presented in Table 1. By incorporating multiple types of ranking information, FiT5 greatly improves over the first-stage retrieval model coCondenser, and outperforms typical BERT Reranker and monoT5 that re-rank on top of the same retriever. On MS MARCO, FiT5 further outperforms multi-stage ranking pipelines HLATR-large and Expando-Mono-Duo, which use significantly larger models (RoBERTa-large (Liu et al., 2019) / 2×T5-3B) and more re-ranking stages. Note that Expando-Mono-Duo is extremely computation-expensive as it requires pairwise inference of $n \times (n-1)$ times (Pradeep et al., 2021).

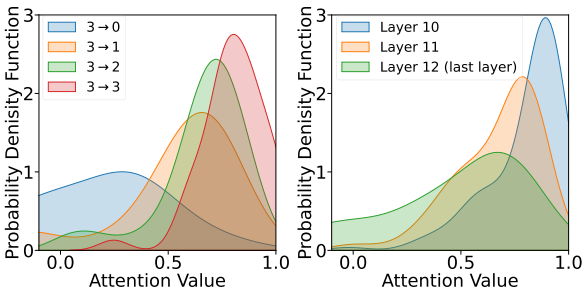
To study the efficiency of FiT5, we measure its inference time and GPU memory usage in comparison to monoT5 on the development set of MS MARCO. As shown in Table 2, compared to monoT5, FiT5 exhibits a mere 4.5% increase in GPU memory usage and only a marginal increase in inference time. This confirms the efficiency of FiT5's architecture, making it well-suited for practical applications.

5.2. Ablation Study

In this section, we first study the contribution of attention fusion in the effectiveness of FiT5. The results are presented in Table 3. When we exclude the feature score (FiT5 (w/o feature)) or global attention (monoT5 (w/ feature)), both scenarios result in a noticeable decline in performance. Notably, monoT5 (w/ feature) does not exhibit a significant performance improvement over monoT5, indicating that the ranking feature can't be effectively captured straightforwardly in a vanilla transformer model. Employing a linear combination of the re-ranker score and the feature still lags behind FiT5, revealing that the use of global attention is the key to effectively integrating the information from the retriever and other documents.

Model	FiT5 (w/o feature)	FiT5
All layers ($l = 1$)	41.23	40.83
Top-6 layers ($l = 7$)	42.49^{an}	43.36^{an}
Top-3 layers ($l = 10$)	42.79^{an}	**43.93**^{a621n}
Top-2 layers ($l = 11$)	**42.95**^{an}	43.43^{an}
Top-1 layer ($l = 12$)	42.78^{an}	43.07^{an}
No global attention	41.49	40.95

Table 4: Performance on MS MARCO with global attention started to introduce at top-k layers. The metric is MRR@10. (permutation test; $p < 0.05$)



(a) Attention between the most relevant passages (3) and passages in every relevance group (0–3).

(b) Attention between the most relevant passages and all others, grouped by layer.

Figure 2: Attention weights distribution on TREC DL 20. (a) depicts the distribution in the last layer (12); 0, 1, 2, and 3 are relevance levels from irrelevant to perfectly relevant. (b) presents the distribution in layer 10, 11, and 12.

We then investigate the impact of the number of global attention layers on performance. We retrain FiT5 with top 1, 2, 3, 6, and 12 transformer layer(s) incorporated with global attention, respectively. Results in Table 4 reveal that starting to integrate global attention from a late layer is an optimal choice. Starting the integration too early may make the optimization harder, whereas starting too late may provide insufficient paths for reasoning.

5.3. Attention Pattern

In this experiment, we investigate the attention patterns within FiT5 and illustrate the distribution of global attention weights in Figure 2. As shown in Figure 2a, within the final layer, the attention values between the most relevant passages (labeled 3) are notably higher than those involving other

passages. As shown in Figure 2b, with increasing layer depth, the general trend reveals a diminishing emphasis in attention values between the most relevant passages and other passages. This attention pattern shows that as data traverses through multiple global attention layers, it fosters a stronger interaction among relevant documents, facilitating the distinction between positive and negative ones.

6. Conclusion

This paper introduces Fusion-in-T5 (FiT5), a unified ranking model that can capture variant information sources. It demonstrates superior or on-par results over cascade ranking systems with more stages. Analysis reveals that the proposed attention fusion mechanism is effective in integrating signals including text matching, ranking features, and global information.

Acknowledgements

This work is supported by the National Key R&D Program of China (No. 2022ZD0116312) and the National Natural Science Foundation of China (No. 62236004, No. 62206042).

7. Bibliographical References

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2005. Learning to rank using gradient descent. In *Proceedings of ICML*, pages 89–96.

Claudio Carpineto and Giovanni Romano. 2012. A survey of automatic query expansion in information retrieval. *ACM Comput. Surv.*, 44(1):1:1–1:50.

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

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of SIGIR*, pages 758–759.

N. Craswell, B. Mitra, E. Yilmaz, and D. Campos. 2021. Overview of the trec 2020 deep learning track. In *TREC*.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the trec 2019 deep learning track. In *TREC*.

W Bruce Croft, Donald Metzler, and Trevor Strohman. 2010. *Search Engines: Information Retrieval in Practice*, volume 520. Addison-Wesley Reading.

Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. 2018. Convolutional neural networks for soft-matching n-grams in ad-hoc search. In *Proceedings of WSDM*, pages 126–134.

Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2019. Trec cast 2019: The conversational assistance track overview. In *Proceedings of TREC*.

Luyu Gao and Jamie Callan. 2022. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of ACL*, pages 2843–2853.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Tevatron: An efficient and flexible toolkit for dense retrieval. *arXiv preprint arXiv:2203.05765*.

Shuguang Han, Xuanhui Wang, Mike Bendersky, and Marc Najork. 2020. Learning-to-rank with bert in tf-ranking. *arXiv preprint arXiv:2004.08476*.

Gautier Izacard and Édouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of EACL*, pages 874–880.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of ACL*, pages 6086–6096.

Hang Li, Ahmed Mourad, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. 2023. Pseudo relevance feedback with deep language models and dense retrievers: Successes and pitfalls. *TOIS*, 41(3):1–40.

Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2020. Fine-grained fact verification with kernel graph attention network. In *Proceedings of ACL*, pages 7342–7351.

Donald Metzler and W Bruce Croft. 2007. Linear feature-based models for information retrieval. *Information Retrieval*, 10:257–274.

Bhaskar Mitra and Nick Craswell. 2017. Neural models for information retrieval. *arXiv preprint arXiv:1705.01509*.

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. *choice*, 2640:660.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. In *Findings of EMNLP*, pages 708–718.

Rodrigo Nogueira and Jimmy Lin. From doc2query to doctttttquery.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*.

Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv preprint arXiv:2101.05667*.

Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the behaviors of bert in ranking. *arXiv preprint arXiv:1904.07531*.

Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W Bruce Croft, and Mohit Iyyer. 2020. Open-retrieval conversational question answering. In *Proceedings of SIGIR*, pages 539–548.

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.

Si Sun, Yingzhuo Qian, Zhenghao Liu, Chenyan Xiong, Kaitao Zhang, Jie Bao, Zhiyuan Liu, and Paul Bennett. 2021. Few-shot text ranking with meta adapted synthetic weak supervision. In *Proceedings of ACL*, pages 5030–5043.

Christopher C Vogt and Garrison W Cottrell. 1999. Fusion via a linear combination of scores. *Information retrieval*, 1(3):151–173.

Shengli Wu. 2009. Applying statistical principles to data fusion in information retrieval. *Expert Systems with Applications*, 36(2):2997–3006.

Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of SIGIR*, pages 55–64.

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

Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. 2021. Pretrained transformers for text ranking: Bert and beyond. In *Proceedings of WSDM*, pages 1154–1156.

HongChien Yu, Chenyan Xiong, and Jamie Callan. 2021. Improving query representations for dense retrieval with pseudo relevance feedback. *arXiv preprint arXiv:2108.13454*.

Kaitao Zhang, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2020. Selective weak supervision for neural information retrieval. In *Proceedings of The Web Conference 2020*, pages 474–485.

Yanzhao Zhang, Dingkun Long, Guangwei Xu, and Pengjun Xie. 2022. Hlatr: enhance multi-stage text retrieval with hybrid list aware transformer reranking. *arXiv preprint arXiv:2205.10569*.

Yue Zhang, ChengCheng Hu, Yuqi Liu, Hui Fang, and Jimmy Lin. 2021. Learning to rank in the age of muppets: Effectiveness–efficiency tradeoffs in multi-stage ranking. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 64–73.

Chen Zhao, Chenyan Xiong, Corby Rosset, Xia Song, Paul Bennett, and Saurabh Tiwary. 2020. Transformer-xh: Multi-evidence reasoning with extra hop attention. In *Proceedings of ICLR*.

Zhi Zheng, Kai Hui, Ben He, Xianpei Han, Le Sun, and Andrew Yates. 2020. Bert-qe: Contextualized query expansion for document re-ranking. In *Findings of EMNLP*, pages 4718–4728.