Improving Conversational Passage Re-ranking with View Ensemble

Anonymous Author(s)*

ABSTRACT

This paper presents ConvRerank, a conversational passage re-ranker that employs a newly developed pseudo-labeling approach. Our proposed view-ensemble method enhances the quality of pseudo-labeled data, thus improving the fine-tuning of ConvRerank. Our experimental evaluation on benchmark datasets shows that combining ConvRerank with a conversational dense retriever in a cascaded manner achieves a good balance between effectiveness and efficiency. Compared to baseline methods, our cascaded pipeline demonstrates lower latency and higher top-ranking effectiveness. Furthermore, the in-depth analysis confirms the potential of our approach to improving the effectiveness of conversational search.

ACM Reference Format:

Anonymous Author(s). 2018. Improving Conversational Passage Re-ranking with View Ensemble. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

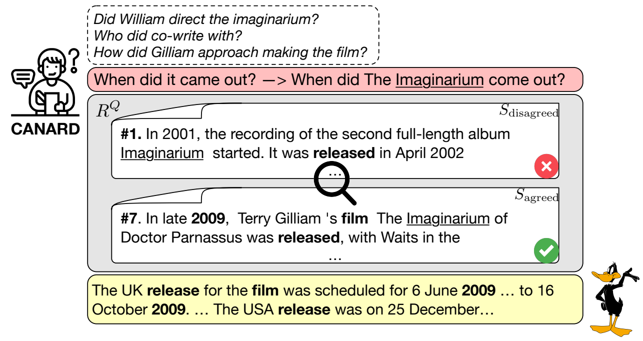

Figure 1: An example of the *view-ensemble* method. R^Q represents the initial ranked list, with #k denoting the top-k relevant passage. While bold words indicate they appear in the ground-truth answer, the words underlined represent those in the question.

1 INTRODUCTION

Conversational search (ConvSearch) [5, 28] has emerged as a rapidly growing research area as the popularity of conversational information seeking systems continues to rise. ConvSearch has the potential to transform the way people search for information, moving from ad-hoc search to interactive search [10, 40]. However, the multi-turn nature of conversations poses significant challenges for information retrieval systems, as users often omit important contexts, particularly in the latter turns of conversations [3, 30]. This creates ambiguity in conversational queries, making it one of the most distinctive challenges for conversational AI systems [1, 3, 26, 30]. To facilitate research in this area, TREC has organized the Conversational Assistance Track (CAsT) [6, 7] to create reliable benchmarks for the evaluation of ConvSearch systems.

Among all the ConvSearch systems, the multi-stage cascaded architecture has proven to be the most effective approach, which addresses the issue of query ambiguity in ConvSearch through the addition of a conversational query reformulation (CQR) module that employs heuristic [34, 37] or neural approaches [1, 15, 20, 33, 35, 38]. Although effective, incorporating CQR modules into the ConvSearch system may increase query latency and complexity, posing challenges for real-world deployment. Conversely, the recently proposed conversational dense retrieval (ConvDR) [14, 18, 21, 39] greatly simplifies the ConvSearch system while demonstrating superior efficiency when compared to cascaded approaches. In ConvDR, a BERT-based query encoder [8, 16] is fine-tuned to encode a user utterance and its dialogue context into a de-contextualized query embedding for dense retrieval without the need for an additional query reformulation module. Despite its efficiency, ConvDR has shown to be less effective than state-of-the-art multi-stage ConvSearch systems, particularly in terms of top-ranking effectiveness (e.g., nDCG@3), as shown by [18].

While the recent work [25, 39] has integrated ConvDR into a cascaded architecture to improve effectiveness through passage re-ranking, we propose that a more optimal re-ranking design in ConvSearch could enhance effectiveness while reducing system complexity. With this in mind, we introduce a conversational passage re-ranker (ConvRerank)[1] that can be more seamlessly integrated with any first-stage retrieval methods for ConvSearch. The main advantages of our proposed ConvRerank over the previous works are two-fold: (1) Unlike Qian and Dou [25], which relies on the separate query reformulation and passage re-ranking, ConvRerank is a single model capable of comprehending conversational queries and assessing their relevance to passages; (2) unlike the previous works of [18, 39] that used human-rewritten queries to construct pseudo-labeled training data, we propose a novel *view-ensemble* pseudo-labeling approach that yields higher-quality data and facilitates fine-tuning of a more robust ConvRerank.

The example in Figure 1 illustrates the motivation behind our view-ensemble method. In this example, a human reformulated the query by replacing the pronoun "it" with "Imaginarium." However, this manual reformulation is still ambiguous for search engines since "Imaginarium" can refer to either an album or a film. As a

[1]Codes have been released at https://anonymous.4open.science/r/ConvRerank-A0D5.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06...$15.00
https://doi.org/XXXXXXX.XXXXXXX

result, the pseudo-relevance labels generated from previous approaches [18, 39] can be misleading (see the top-ranking passage in Figure 1). In this work, we recognize that the annotated answer (see the bottom block in Figure 1) for a given query contains useful information for clarifying the ambiguous human-rewritten query. Based on this intuition, we explore how to better combine the information from a human-rewritten query and its answer and propose a simple yet effective view-ensemble method for generating training data with more accurate pseudo-relevance labels.

Our experiments on TREC CAsT [6, 7] show that ConvRerank, a single model fine-tuned with our created pseudo labels, yields better re-ranking effectiveness than a more cumbersome re-ranking pipeline with a CQR and a re-ranking module. Furthermore, our in-depth analysis regarding the effects of different pseudo labels, first-stage retrieval, and model sizes confirms the robustness of the proposed method. In addition, our ConvRerank can be integrated into any existing conversational retrieval methods, such as [18, 37].

2 PRELIMINARIES

2.1 Task Definition and Notations

The key distinction between conversational search (ConvSearch) and standard ad-hoc search lies in the interaction between queries. While the latter utilizes a standard text query, ConvSearch utilizes a *conversational query*, structured as a series of utterances. Formally, each conversational query, q_i, including the i-th turn utterance along with its corresponding conversational history (e.g., previous utterances), is defined as $q_i = (u_i; u_1, u_2, ..., u_{i-1})$. Given a conversational query q_i, the goal of ConvSearch systems is to retrieve a ranked list of relevant passages, denoted as $R = (p_1, p_2, \ldots, p_k)$. The quality of R can be measured using information retrieval metrics, such as normalized discounted cumulative gain (nDCG).

2.2 Cascaded Architecture for ConvSearch

2.2.1 Conversational Query Reformulation. ConvSearch systems have developed into complex pipelines. In particular, conversational query reformulation (CQR) is widely recognized as the most critical component of such systems [1, 33, 35, 36]. The main goal of CQR is to transform a conversational query q_i into an ad-hoc query q'_i, denoted as

$$q'_i = \mathcal{F}_{\text{CQR}}(u_i; u_1, u_2, ..., u_{i-1})$$

As an example, Yang et al. [37] propose a query expansion approach that appends context-dependant words extracted from historical conversations (i.e., $(u_1, u_2, \ldots, u_{i-1})$) to u_i to form q'_i. Some researchers frame CQR as a term-classification task and utilize BERT models [8] to select tokens from historical conversations [15, 35]. Moreover, some studies utilize transformer-based generative models [27, 29] to rewrite queries through few-shot learning [38] or supervised learning [19, 33] on the CANARD dataset [9].

2.2.2 Multi-stage Pipeline. Many ConvSearch studies have adopted the standard multi-stage passage ranking pipeline in ad-hoc search [24] while using the reformulated query q' as an ad-hoc query. Examples include the works of [15, 20, 33, 35]. Specifically, an effective ConvSearch system consists of a CQR module $\mathcal{F}_{\text{CQR}}$, a first-stage retriever $\mathcal{F}_{\text{RT}}$ and a second-stage passage re-ranker $\mathcal{F}_{\text{RR}}$, as follows:

$$\mathcal{D}_{\text{RT}} = \mathcal{F}_{\text{RT}}(q'; p \in \mathcal{D}),\ R = \mathcal{F}_{\text{RR}}(q'; p \in \mathcal{D}_{\text{RT}}),$$

where $\mathcal{D}$ denotes the entire passage collection, and $\mathcal{D}_{\text{RT}}$ refers to a candidate passage set extracted by the first-stage retriever from $\mathcal{D}$ ($|\mathcal{D}| \gg |\mathcal{D}_{\text{RT}}|$). The passages in $\mathcal{D}_{\text{RT}}$ are then sorted into a ranked list R by the passage re-ranker.

2.2.3 Dense Retrieval. Dense retrieval (DR) using a bi-encoder architecture with a passage encoder and a query encoder has gained attention for its effectiveness and efficiency in many knowledge-intensive tasks, as demonstrated in recent studies [12, 13, 31]. DR works by precomputing representations of passages in a corpus through the passage encoder. During retrieval, only the encoding of the query is performed, allowing for efficient end-to-end retrieval through inner product calculation or approximate nearest neighbor search [11]. The bi-encoder architecture used in DR can be further optimized for conversational dense retrieval (ConvDR) by fine-tuning the model in a few-shot [21, 25, 39] or weakly-supervised [18] manner. Despite its efficiency, ConvDR methods still fall short compared to multi-stage pipelines, particularly in terms of top-ranking effectiveness, as highlighted in [18].

3 METHOD

3.1 Pseudo-Labeling with Ensemble Views

Inspired by [18], we generate a ranked list for the 30K manually rewritten queries q^* in the CANARD dataset [9]. We employ an effective two-stage retrieval pipeline [24] consisting of BM25 search and a monoT5 [23] re-ranker to obtain the ranked list for q^*:

$$R^Q = \text{monoT5}\Big(q^*; p \in \text{BM25}\big(q^*; p \in \mathcal{D}\big)\Big), \quad (1)$$

where R^Q refers to the ranked list consisting of M passages, which are re-ranked from the set of N passage candidates via the BM25 retriever ($M < N$).[2] Note that as in previous work by [18], we adopt the corpus $\mathcal{D}$ from CAsT [6], which includes passages from TREC CAR [22] and MSMARCO [2].

Motivated by previous works [4, 15], we propose further to leverage the *answer* view and use these accurate signals to construct a ranked list with an *ensemble* view. First, to acquire the ranked list with the *answer* view, we concatenate the query q^* with the ground-truth answer a from QuAC [3], which is an initial dataset of CANARD [9]. We then pass it through the same retrieval pipeline as in Eq. (1), obtaining the answer-view ranked list

$$R^A = \text{monoT5}\Big(q^*; p \in \text{BM25}\big(q^* \parallel a; p \in \mathcal{D}\big)\Big), \quad (2)$$

where $\parallel$ denotes the concatenation operator. With the two ranked lists (i.e., R^Q and R^A), we define a filtering function Φ to generate a ranked list with *ensemble* views as

$$R^{\text{EM}(R^Q|R^A)} = \Phi(R^Q, R^A) = S_{\text{agreed}} \parallel S_{\text{disagreed}}, \quad (3)$$

where $R^{\text{EM}(R^Q|R^A)}$ denotes the ranked list with an ensemble view that R^A serves as a filter towards R^Q, consisted of two ordered lists:

$$S_{\text{agreed}} = (p_1^+, p_2^+, \ldots, p_t^+),$$
$$S_{\text{disagreed}} = (p_1^-, p_2^-, \ldots, p_h^-),$$

where p_i^+ denotes the passage agreed by both views (i.e., p_i^+ in both R^Q and R^A), and p_j^- denotes the passage in R^Q but not in R^A. Note

[2] We follow previous works [18] by setting $M = 200$ and $N = 1000$ in our experiments.

Table 1: TREC CAsT statistics.

	CAsT'19 Eval	CAsT'20 Eval
# Queries	173	208
# Topics	20	25
# Judgements	29,571	40,451
# Passages	38M	

that we here keep the original relative order of passages in R^Q for the aforementioned two ordered lists.

In other words, the function Φ reorders the passages in R^Q by pushing the passages agreed by both R^Q and R^A forward and moves the ones only in R^Q backward. The motivation behind this design is that a stand-alone query is often ambiguous for search engines [32]; This ambiguity is even more critical in the context of ConvSearch, as illustrated by the example in Figure 1. As a result, relying solely on R^Q to synthesize pseudo relevance for model training may cause re-rankers to establish unfaithful relations between passages and conversational context. To address this issue, we combine the ranked list with the answer view to reorder passages in R^Q; that is, 1) the resulting passages in S_{agreed} should be more aligned with the user's information need, and 2) the ones in $S_{\text{disagreed}}$ could serve as hard negative to facilitate a more effective training of ConvRerank.

3.2 Fine-tuning Conversational Passage re-rankers with Pseudo-Labeling

For training the proposed ConvRerank, we adopt the ensemble-view ranked list $R^{\text{EM}(R^Q|R^A)}$ to synthesize pseudo relevance. Specifically, for each query, we generate the pseudo labels by treating the top-k results in the ensemble-view ranked list as (pseudo) positive labels and randomly sampling k passages from the top-k to M passages in the same list as (pseudo) negative labels.[3] As for the backbone architecture of ConvRerank, we use T5 models [29] and recast the input format of conversational query passage pairs ($q_i = (u_i; u_1, u_2, ..., u_{i-1}), p$) as a text-to-text format:

Query: u_i Context: $\Omega(u_1, u_2, \ldots, u_{i-1})$ Document: p Relevant:

where Ω indicates the join function with a special unused token in T5 vocabulary "<extra_id_10>" as the separation tokens between each element (i.e., each historical utterance). The objective is the negative log-likelihood loss of generating `true/false` tokens for relevant/irrelevant passages. We compute the relevance scores by taking the probability of `true/false` logit values following the approach of monoT5 [23]. It is worth noting that while our focus is on re-ranking and ConvSearch, the proposed pseudo-labeling method can be applied to ConvDR and other IR tasks as well.

4 EXPERIMENTS

4.1 Data and Experimental Setups

4.1.1 TREC CAsT Evaluation Topics. We used benchmark evaluation data from the TREC Conversational Assistant Track (CAsT): CAsT'19 Eval [7] and CAsT'20 Eval [6]. Each data includes TREC-judged topics; each topic has approximately 8 to 10 turns of questions, and the relevance judgment adopts a five-point scale from 0 to 4. The corpora are composed of MS MARCO [2] and TREC CAR [22]. The data statistics are presented in Table 1.

[3]Note that we set k to 40, which is found to be optimal in our experiments

Table 2: Empirical evaluation on CAsT datasets, including our systems and other baseline multi-stage systems. '†' indicates re-ranking performs on top-500 passage candidates; other systems use top-100 passage candidates.

	Latency	CAsT'19 Eval		CAsT'20 Eval	
	(ms/q)	@3	@100	@3	@100
Our systems					
CQE	-	0.492	0.447	0.319	0.350
CQE → T5-rewrite+monoT5	1910	0.549	0.484	0.418	0.395
CQE → ConvRerank (proposed)	1675	**0.563**	**0.487**	**0.432**	**0.456**
Baseline multi-stage systems					
ConvDR → BERT (RRF) [39]	1900	0.541	-	0.392	-
CRDR [25]	1690	0.553	-	0.381	-
CTS+MVR† [15]	14630	**0.565**	-	-	-
Upper-bound system w/ manual query					
TCT-ColBERT → monoT5	-	0.583	0.545	0.556	0.546

4.1.2 Training, Inference, and Evaluation. We first initialized our ConvRerank with the monoT5 [23] checkpoint, a T5-base re-ranking model that has been fine-tuned on MSMARCO [2].[4] We then fine-tuned the model using our synthesized pseudo labels (see Section 3) with the batch size of 256 for 5 epochs, which is chosen based on the performance on the CAsT'19 train set, within the range of 1 to 5. The other settings for fine-tuning, such as the learning rate and sequence length, are the same as monoT5 [23]. We re-rank the top 100 passage candidates retrieved from CQE [18] and compare their top-ranking and overall effectiveness, as measured by nDCG@3 and @100, respectively. The latency of the re-ranking stage was measured on Google Colab with an A100 GPU. Note that while re-ranking, we set the maximum token length for each document to 384 and the remaining 128 for the query and its context.

4.2 Experimental Results

Table 2 presents our experimental results. We first compared the effectiveness of two passage re-rankers: the monoT5 reranker [23] with a T5-base query rewriting model [19] and the proposed ConvRerank. As shown in the upper panel of the table, ConvRerank outperforms the baseline re-ranker, monoT5 with T5-rewrite, on all evaluation sets, achieving 3.3% and 2.6% improvements in terms of nDCG@3 on CAsT'19 Eval and CAsT'20 Eval, respectively. In terms of efficiency, ConvRerank, which does not require conversational query rewriting, achieves lower overall latency compared to monoT5 with T5-rewrite, making it a more efficient option.

In the lower panel of Table 2, we compared our cascaded approach (CQE → ConvRerank) to other multi-stage systems, including (a) ConvDR → BERT (RRF) [39], which is a rank fusion [4] of few-shot ConvDR and BERT re-ranker, (b) CRDR [25], which integrates ConvDR and a query modification module for further BERT re-ranking, and (c) CTS+MVR [15], which utilizes multiple query views and BERT-base re-ranking to fuse over the views. We observe that our approach yields better efficiency and effectiveness (especially in CAsT'20) compared to these state-of-the-art methods.

[4]We found that fine-tuning from scratch yields a significant effectiveness drop.

Table 3: Using different pseudo labels for fine-tuning.

	Ranked list	CAsT'19 Eval @3	CAsT'19 Eval @100	CAsT'20 Eval @3	CAsT'20 Eval @100
(a)	$R^{\text{EM}(R^Q\|R^A)}$ (proposed)	**0.563**	**0.487**	**0.432**	**0.456**
(b)	R^Q	0.517	0.467	0.396	0.382
(c)	R^A	0.495	0.464	0.392	0.382
(d)	$R^{\text{EM}(R^A\|R^Q)}$	0.519	0.474	0.403	0.389

Table 4: Evaluation on different first-stage (sparse, dense, and hybrid) retrieved passages. '‡' indicates re-ranking performs on top-1000 passage candidates; other models use top-100.

	Retrieval (→ Re-ranking)	CAsT'19 Eval @3	CAsT'19 Eval @100	CAsT'20 Eval @3	CAsT'20 Eval @100
Sparse	HQE [37]	0.261	0.308	0.164	0.204
	HQE → T5-rewrite + monoT5‡	0.553	**0.519**	0.379	0.377
	HQE → ConvRerank‡	**0.558**	0.511	**0.389**	**0.384**
Dense	CQE [18]	0.492	0.447	0.319	0.350
	CQE → T5-rewrite + monoT5	0.549	0.484	0.418	0.395
	CQE → ConvRerank	**0.563**	**0.487**	**0.432**	**0.456**
Hybrid	CQE-HYB [18]	0.498	0.494	0.330	0.368
	CQE-HYB → T5-rewrite + monoT5	0.556	0.531	**0.428**	**0.411**
	CQE-HYB → ConvRerank	**0.584**	**0.534**	0.424	0.410

This result demonstrates the advantages of ConvRerank over the other re-ranking solutions for conversational search. Note that compared to CAST'19, CAsT'20 requires more complex conversational query understanding from the historical utterances and system responses [6]; thus, the larger gap in CAsT'20 between our system and the others indicates that our ConvRerank can address more challenging conversational queries. However, the systems evaluated in our experiments still lag behind the multi-stage pipeline using human-rewritten queries (as shown in the last row of Table 2), indicating there is still room for improvement for future research.

4.3 Effect Analysis

4.3.1 Effect of Different Pseudo Labels. To examine the effect of pseudo labels for fine-tuning ConvRerank, Table 3 compares the effectiveness of models trained on the data with pseudo labels from different ranked lists: (a) $R^{\text{EM}(\text{R}^Q|\text{R}^A)}$, our approach; (b) R^Q in Eq. (1); (c) R^A in Eq. (2); (d) $R^{\text{EM}(\text{R}^A|\text{R}^Q)}$, another ranked list also with the ensemble view by reversing the two lists in Eq. (3). We observe that the re-rankers trained on the pseudo labels generated from the ranked lists with the *ensemble* view (i.e., (a) and (d)) outperform their corresponding single-view variants. (i.e., (b) and (c)) This result demonstrates that R^Q and R^A provide different views for conversational search and can complement each other well. It is worth noting that human-reformulated queries alone (R^Q) generate better training data than those combined with answers (R^A).

4.3.2 Effect of First-stage Retrieval. To examine the robustness of ConvRerank, we also evaluated its performance with two other first-stage retrieval methods: (1) HQE [37], a sparse retriever, built upon BM25 that heuristically concatenates words from the historical conversation; (2) CQE-HYB [18], a hybrid retriever that combines CQE and CQE-sparse.[5] Note that neither of the two approaches requires the use of neural models for reformulating conversational queries, which is consistent with our goal of building a simple yet effective cascaded architecture. Table 4 tabulates the performance with different first-stage retrieval methods, including the originally adopted CQE and the two approaches above. We observe that ConvRerank is able to yield improvement upon different first-stage retrieval methods. Notably, ConvRerank works effectively with HQExp and sometimes performs on par with dense retrieval approaches; for example, on CAsT'19, HQE → ConvRerank achieves a similar nDCG@3 score to CQE-HYB → monoT5 (i.e., 0.558 v.s. 0.556). These results suggest that ConvRerank can provide benefits regardless of the first-stage environments and improve effectiveness even when adopting a simple and non-neural first-stage retrieval method.

Table 5: Scaling up the model sizes.

Re-ranker	Size	CAsT'19 Eval @3	CAsT'19 Eval @100	CAsT'20 Eval @3	CAsT'20 Eval @100
monoT5 (w/T5-rewrite)	large	0.534	0.589	0.449	0.531
ConvRerank	large	**0.572**	**0.610**	**0.487**	**0.550**
monoT5 (w/T5-rewrite)	3B	0.534	0.592	0.470	0.545
ConvRerank	3B	**0.583**	**0.618**	**0.496**	**0.562**

4.3.3 Effect of Model Size. To examine the impact of model size on the performance of ConvRerank, we fine-tine ConvRerank on T5-large and T5-3B[6] with the same procedure and inference setups. As observed from Table 5, our ConvRerank benefits more from scaling model size compared to the monoT5 re-ranker (with T5-rewrite). We hypothesize that the T5-base rewriter bounds the monoT5 re-ranking effectiveness. Thus, to attest to the effectiveness of the multi-stage pipeline (monoT5 w/T5 rewrite), we should scale the size of both the neural re-ranker and re-writer, which potentially results in a higher query latency of the re-ranking stage. In contrast, ConvRerank is a single model and does not suffer from this issue, making it an advantageous choice for ConvSearch.

5 CONCLUSION

We present a novel approach for conversational passage re-ranking, which includes a pseudo-labeling method and our proposed ConvRerank model. Particularly, we design a view-ensemble method to synthesize high-quality pseudo labels that are then used to fine-tune ConvRerank. Moreover, our cascaded approach, the ConvRerank followed by conversational dense retrieval (i.e., CQE) as the first-stage retriever has demonstrated superior performance over other baseline systems on the TREC CAsT datasets in terms of both top-ranking effectiveness and re-ranking latency. Moving forward, we plan to strengthen dependencies between the retriever and re-ranker, for instance, by (1) implementing a co-training framework [26], and (2) adopting first-stage candidate pruning techniques [17], to improve effectiveness and efficiency.

[5] CQE-sparse is a variant of CQE that employs L_2-norm to select words from the historical context as query expansion for BM25 search.

[6] We only fine-tune the model on T5-3B for 2 epochs due to high computational costs.

REFERENCES

[1] Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. 2021. Open-Domain Question Answering Goes Conversational via Question Rewriting. In *Proc. of NAACL-HLT*. 520–534. https://doi.org/10.18653/v1/2021.naacl-main.44

[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. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. https://doi.org/10.48550/arxiv.1611.09268

[3] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question Answering in Context. In *Proc. of EMNLP*. 2174–2184. https://doi.org/10.18653/v1/D18-1241

[4] Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods. In *Proc. of SIGIR*. 758–759. https://doi.org/10.1145/1571941.1572114

[5] J. Shane Culpepper, Fernando Diaz, and Mark D. Smucker. 2018. Research Frontiers in Information Retrieval: Report from the Third Strategic Workshop on Information Retrieval in Lorne. *SIGIR Forum* 52, 1 (2018), 34–90. https://doi.org/10.1145/3274784.3274788

[6] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2019. TREC CAsT 2019: The Conversational Assistance Track Overview. (2019). https://trec.nist.gov/pubs/trec28/papers/OVERVIEW.CAsT.pdf

[7] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2020. CAsT 2020: The Conversational Assistance Track Overview. (2020). https://trec.nist.gov/pubs/trec29/papers/OVERVIEW.C.pdf

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of NAACL-HLT*. 4171–4186. https://doi.org/10.18653/v1/N19-1423

[9] Ahmed Elgohary, Denis Peskov, and Jordan Boyd-Graber. 2019. Can You Unpack That? Learning to Rewrite Questions-in-Context. In *Proc. of EMNLP-IJCNLP*. 5918–5924. https://doi.org/10.18653/v1/D19-1605

[10] Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural Approaches to Conversational AI. In *Proc. of ACL*. 2–7. https://doi.org/10.18653/v1/P18-5002

[11] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale Similarity Search with GPUs. (2017). https://doi.org/10.48550/arxiv.1702.08734

[12] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proc. of EMNLP*. 6769–6781. https://doi.org/10.18653/v1/2020.emnlp-main.550

[13] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proc. of SIGIR*. 39–48. https://doi.org/10.1145/3397271.3401075

[14] Antonios Minas Krasakis, Andrew Yates, and Evangelos Kanoulas. 2022. Zero-Shot Query Contextualization for Conversational Search. In *Proc. of SIGIR*. 1880–1884. https://doi.org/10.1145/3477495.3531769

[15] Vaibhav Kumar and Jamie Callan. 2020. Making Information Seeking Easier: An Improved Pipeline for Conversational Search. In *Proc. of EMNLP (Findings)*. 3971–3980. https://doi.org/10.18653/v1/2020.findings-emnlp.354

[16] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *Proc. of ICLR*. https://openreview.net/forum?id=H1eA7AEtvS

[17] Minghan Li, Xinyu Zhang, Ji Xin, Hongyang Zhang, and Jimmy Lin. 2022. Certified Error Control of Candidate Set Pruning for Two-Stage Relevance Ranking. In *Proc. of EMNLP*. 333–345. https://aclanthology.org/2022.emnlp-main.23

[18] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. Contextualized Query Embeddings for Conversational Search. In *Proc. of EMNLP*. 1004–1015. https://doi.org/10.18653/v1/2021.emnlp-main.77

[19] Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. 2020. Conversational Question Reformulation via Sequence-to-Sequence Architectures and Pretrained Language Models. *arXiv:2004.01909* (2020). https://doi.org/10.48550/arxiv.2004.01909

[20] Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. 2021. Multi-Stage Conversational Passage Retrieval: An Approach to Fusing Term Importance Estimation and Neural Query Rewriting. *ACM Trans. Inf. Syst.* 39, 4, Article 48 (2021), 29 pages. https://doi.org/10.1145/3446426

[21] Kelong Mao, Zhicheng Dou, and Hongjin Qian. 2022. Curriculum Contrastive Context Denoising for Few-shot Conversational Dense Retrieval. In *Proc. of SIGIR*. 176–186. https://doi.org/10.1145/3477495.3531961

[22] Federico Nanni, Bhaskar Mitra, Matt Magnusson, and Laura Dietz. 2017. Benchmark for Complex Answer Retrieval. In *Proc. of ICTIR*. 293–296. https://doi.org/10.1145/3121050.3121099

[23] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Proc. of EMNLP (Findings)*. 708–718. https://doi.org/10.18653/v1/2020.findings-emnlp.63

[24] Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-Stage Document Ranking with BERT. *arXiv:1910.14424* (2019). https://doi.org/10.48550/arxiv.1910.14424

[25] Hongjin Qian and Zhicheng Dou. 2022. Explicit Query Rewriting for Conversational Dense Retrieval. In *Pro. of EMNLP*. 4725–4737. https://aclanthology.org/2022.emnlp-main.311

[26] Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W. Bruce Croft, and Mohit Iyyer. 2020. Open-Retrieval Conversational Question Answering. In *Proc. of SIGIR*. 539–548. https://doi.org/10.1145/3397271.3401110

[27] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models are Unsupervised Multitask Learners.

[28] Filip Radlinski and Nick Craswell. 2017. A Theoretical Framework for Conversational Search. In *Proc. of CHIIR*. 117–126. https://doi.org/10.1145/3020165.3020183

[29] 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 (2020), 1–67. http://jmlr.org/papers/v21/20-074.html

[30] Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A Conversational Question Answering Challenge. *Trans. Assoc. Comput. Linguist.* 7 (2019), 249–266. https://doi.org/10.1162/tacl_a_00266

[31] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proc. of EMNLP-IJCNLP*. 3982–3992. https://doi.org/10.18653/v1/D19-1410

[32] Ruihua Song, Zhenxiao Luo, Ji-Rong Wen, Yong Yu, and Hsiao-Wuen Hon. 2007. Identifying Ambiguous Queries in Web Search. In *Proc. of WWW*. 1169–1170. https://doi.org/10.1145/1242572.1242749

[33] Svitlana Vakulenko, Shayne Longpre, Zhucheng Tu, and Raviteja Anantha. 2021. Question Rewriting for Conversational Question Answering. In *Proc. of WSDM*. 355–363. https://doi.org/10.1145/3437963.3441748

[34] Nikos Voskarides, Dan Li, Andreas Panteli, and Pengjie Ren. 2019. ILPS at TREC 2019 Conversational Assistant Track. In *Proc. TREC*.

[35] Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. 2020. Query Resolution for Conversational Search with Limited Supervision. In *Proc. of SIGIR*. https://doi.org/10.1145/3397271.3401130

[36] Zeqiu Wu, Yi Luan, Hannah Rashkin, David Reitter, Hannaneh Hajishirzi, Mari Ostendorf, and Gaurav Singh Tomar. 2022. CONQRR: Conversational Query Rewriting for Retrieval with Reinforcement Learning. In *Proc. of EMNLP*. 10000–10014. https://aclanthology.org/2022.emnlp-main.679

[37] Jheng-Hong Yang, Sheng-Chieh Lin, Chuan-Ju Wang, Jimmy Lin, and Ming-Feng Tsai. 2019. Query and Answer Expansion from Conversation History. In *Proc. of TREC*. https://trec.nist.gov/pubs/trec28/papers/CFDA_CLIP.C.pdf

[38] Shi Yu, Jiahua Liu, Jingqin Yang, Chenyan Xiong, Paul Bennett, Jianfeng Gao, and Zhiyuan Liu. 2020. Few-Shot Generative Conversational Query Rewriting. In *Proc. of SIGIR*. 1933–1936. https://doi.org/10.1145/3397271.3401323

[39] Shi Yu, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu. 2021. Few-Shot Conversational Dense Retrieval. In *Proc. of SIGIR*. 829–838. https://doi.org/10.1145/3404835.3462856

[40] Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. 2022. Conversational Information Seeking. *arXiv:2201.08808* (2022). https://arxiv.org/abs/2201.08808