Query Rewriting with Expansion and Multi-Turn Entity Graphs for Answer Selection

Nour Jedidi[†], Gustavo Gonçalves[†§], Jamie Callan[†]
{njedidi,ggoncalv,callan}@cs.cmu.edu

[†]Language and Technology Institute, Carnegie Mellon University, USA
[§]NOVA LINCS, Universidade NOVA de Lisboa, Portugal

ABSTRACT

Conversational search is challenging in part because often the meaning of the current question cannot be fully understood without contextual information from previous questions and/or answers. This paper describes research on using query reformulation and lightweight reranking based on a multi-turn entity graph to represent and make use of contextual information in the CAsT 2021 track.

1 INTRODUCTION

Conversational search has gathered a lot of interest both by industry with the rise of smart assistants, and subsquent domestic products that include such assistants, as well as by the academic comunity, where the CAsT track is a manifestation of this interest and a joint effort to obtain offline datasets for conversational search. Conversational information seeking (CIS) appears as the next step to drift from the usual keyword-based search paradigm. Both users and systems are formatted for interacting with search through keywords, where queries are short and lack conversational elements, such as context. Conversational search intents introduce novel and difficult challenges to which search engines are not sensible. The introduction of a fine-grained context dependency between utterances introduces problems such as coreference resolution, and lack of commonsense knowledge.

Our work explores the contextual approximation of the user's information need through query reformulation using pre-trained language models. Moreover, we also focus on keeping track of conversational context using the entities that appear throughout the conversation to identify relevant documents that could improve early-precision, by giving more weight to entities that were disregarded during the ranking process.

Section 2 discusses related work. Section 3 formalize our query rewriting and entity centrality approaches. Section 4 discusses our results. Section 5 concludes.

2 RELATED WORK

Our work is related to research in *conversational search* and *entity-driven reranking*.

Conversational Search A common approach for conversational search includes a query rewriting model that first reformulates a raw, de-contextualized, query and then uses a standard search setup such as a BM25 initial ranker followed by a pre-trained language model for reranking such as BERT [14] or T5 [15]. Recent research [7, 12, 22, 24] have used variants of this multi-stage framework by fine-tuning pre-trained language models such as T5 [19] or GPT-2 [18] as the query rewriter. There has also been work that models query rewriting as a classification problem [24], in which, for each term in previous conversational turns, the task is to predict whether it is relevant or not to the current turn query.

Another direction of research has focused on leveraging dense retrieval for conversational search. Lin et al. [11] proposed a single-stage conversational search pipeline which integrates query reformulation into the query encoder of a bi-encoder. Similarly, Yu et al. [26] use a teacher-student approach to train a query encoder, which takes as input the concatenation of the current turn and previous turns, to mimic embeddings of rewritten queries from an ad-hoc dense retriever.

Entity-Driven Reranking Entities are one effective way of modeling what is important about a topic. Moreover, conversations are often about entities. These observations introduce possible ways to improve current context tracking techniques for conversational search. Previous work shows that explicitly tackling named-entities can improve many language modeling tasks [9, 10]. Neural architectures still present room for improvement while carrying conversational history, as they do not fully discriminate the important textual information available [21].

Popular approaches to maintaining a conversational context include using the Transformer architecture to encode context along with information needs [16, 17]. While these approaches leverage on pre-trained language models to obtain semantic context they are limited by the amount of information they can encode.

External knowledge sources provide additional information that may not be explicit in documents. A fundamental aspect is the extraction and linking of named-entities across the conversation turns. Well known entity linking work includes TagMe [8] and DBPediaSpotlight [13], which used knowledge-graphs based on Wikipedia, or DBpedia [1], and their respective entity page information and connections to disambiguate the most likely entities to be linked.

3 PROPOSED APPROACHES

Conversational search introduces context inference issues due to the multi-turn design of the CAsT track. The systems designed for this submission focus on understanding two main issues. The first explores if entities are useful to improve the early precision of the top documents by including information that may have been neglected by pre-trained language models. The second issue is the approximation of the unresolved queries to the manually resolved query set. In this section, we outline our retrieval setup and the entity reranking approach. Then we describe our query reformulation methodology.

3.1 Multi-Turn Entity Graph

Named-entities can be strong signals to estimate a conversation topic, and an information source to overcome semantic limitations introduced by pre-trained language models. Query entities define the initial information intent, hence we consider them to be the strongest signal to drive the conversation. Passage entities also important since they are related to the topic, and help to determine the importance of passages that don't have the required terms to be higher in the ranking.

The conversational history at turn j consists of the current query q_j, previous queries, and current retrieval results. The query sequence $Q_j = \{q_i, \cdots, q_j\}$ consists of previous queries and the current query. Each turn j is assumed to have retrieved up to k passages $P_j = \{p_j^1, \cdots, p_j^k\}$. The conversation entity graph is built from queries Q_j and passages P_j.

The reranking model consists of two stages. The first stage is the text-ranker described in section 3.2. The text ranker produces the P_j ranking. The second stage analyzes P_j to estimate *entity centrality* scores, which are used to rerank passages more effectively. We refer to these two stages as the *text-retrieval* and *entity centrality* stages throughout this notebook.

The set of unique entities E_j, for turn j, is computed from the conversation query history Q_j, and the top retrieved passages P_j of the current query q_j.

This leads to the set of unique entities E_j defined as:

$$E_j = \{e_1, \cdots, e_g, \cdots, e_n\}, \qquad \forall e_g \in Q_j \cup P_j \tag{1}$$

Given the entities E_j and the passages P_j, the system is able to compute the entity-passage occurrence matrix $\mathbf{C_{P_j}} \in \{w_{g,j}\}^{n \times k}$ as the weighted entity occurrences for query q_j. The conversation query history vector $\mathbf{C_{Q_j}} \in \{0,1\}^{n \times 1}$ is defined as the binary vector of entities present in the query history Q_j.

The occurrence matrix $\mathbf{C_{QP_j}} \in \{w_{g,j}\}^{n \times k+1}$ is the concatenation of vector $\mathbf{C_{Q_j}}$ with matrix $\mathbf{C_{P_j}}$.

$$\mathbf{C_{QP_j}} = \left[\begin{array}{c|c} \gamma \cdot \begin{bmatrix} q_{j,e_1} \\ \vdots \\ \vdots \\ q_{j,e_n} \end{bmatrix} & (1-\gamma) \cdot \begin{bmatrix} p_{j,e_1}^1 & \cdots & p_{j,e_1}^k \\ \vdots & \ddots & \vdots \\ p_{j,e_n}^1 & \cdots & p_{j,e_n}^k \end{bmatrix} \end{array} \right]^T \tag{2}$$

Finally, the entity graph is given by the application of the dot product over the occurrence matrix.

$$\mathbf{G_j} = \mathbf{C_{QP_j}} \cdot \mathbf{C_{QP_j}}^T, \quad \mathbf{G_j} \in R^{n \times n} \tag{3}$$

Given the entity graph defined above, the system uses PageRank [3] to calculate entity centrality due to its probabilistic view of centrality, and efficient convergence with a power-iteration implementation [5]. The EC vector of the top passages entities of the conversation turn j is computed as

$$\mathbf{EC_j}^{(t)} = (1-\alpha) \cdot \frac{1}{|EC_j|} + \alpha \cdot \mathbf{G_j} \cdot \mathbf{EC_j}^{(t-1)} \tag{4}$$

where α is the damping factor and each dimension i of EC_j contains the centrality score of entity i.

Formally, the score of each top passage is obtained by computing the dot product between $\mathbf{EC_j}$ scores at turn j, and the entity-passage matrix $\mathbf{C_{P_j}}$.

$$\mathbf{S_j} = \mathbf{EC_j}^T \cdot \mathbf{C_{P_j}} = \begin{bmatrix} EC_{j,e_1} \\ \vdots \\ EC_{j,e_n} \end{bmatrix}^T \cdot \begin{bmatrix} p_{j,e_1}^1 & \cdots & p_{j,e_1}^k \\ \vdots & \ddots & \vdots \\ p_{j,e_n}^1 & \cdots & p_{j,e_n}^k \end{bmatrix} \tag{5}$$

eq. (5) results in a scoring vector $\mathbf{S_j} \in [0,1]^{1 \times k}$ for all of the initial passages in matrix $\mathbf{C_{P_j}}$ derived from the entity centralities. The systems's used in this submission use TagMe [8] to extract entities from both queries and documents. We use a threshold of 0.1 to extract entities.

3.2 Passage Retrieval and Reranking ($T5_{rerank}$)

Our system uses BM25 as a first-stage retrieval ranker followed by a T5 reranker [15], $T5_{rerank}$. Given a query, we retrieve 1000 passages with BM25. With the top 1000 passages, we rerank them using $T5_{rerank}$. We fine-tune $T5_{rerank}$ on the MS-MARCO training dataset of (query, non-relevant passage, relevant passage) triples. Similarly to [15] our text input to the $T5_{rerank}$ is:

$$\text{Query: } q \text{ Passage: } p \text{ Relevant:} \tag{6}$$

3.3 Query Rewriting with Expansion

The query rewriting pipeline consists of a two-step process that first rewrites the query with a query rewriting model ($T5_{rewrite}$) and then expands the rewritten query with a query expansion model ($T5_{expand}$). Once the user query is rewritten and expanded, we feed the it into BM25 + T5 reranker ($T5_{rerank}$). This process is shown in 1.

3.3.1 Query Rewriting Model ($T5_{rewrite}$). Following previous work [12], we fine-tune the text-to-text transformer (T5) model for query rewriting. Given a dataset of (raw query, conversational context, rewritten query) triples, the query rewriting model, $T5_{rewrite}$, takes as input a concatenation of the raw query and conversational context and outputs a rewritten query. The conversational context $c = \{q_1, p_1, q_2, p_2, \cdots, q_i, p_i\}$ consists of previous conversational queries q_i and previous retrieved answers p_i. We use the following text as input to $T5_{rewrite}$:

$$\text{Rewrite Question: } r \text{ Given: } c \tag{7}$$

where r is the raw query and c is the conversational context.

3.3.2 Query Expansion Model ($T5_{expand}$). A comparison of query rewriting methods [23] showed that combining a sequence generation query rewriter with a term classification query rewriter improves performance. Motivated by these results, we add a term classification model to the query rewriting pipeline. The query expansion model, $T5_{expand}$, is another T5 model that, given a raw query and conversational context, is fine-tuned to *predict* expansion terms rather than *rewrite*. To fine-tune $T5_{expand}$, we use a dataset of (raw query, conversational context, expansion terms) triples. As before, the conversational context consists of previous conversational queries q_i and previous retrieved answers p_i. $T5_{expand}$ is

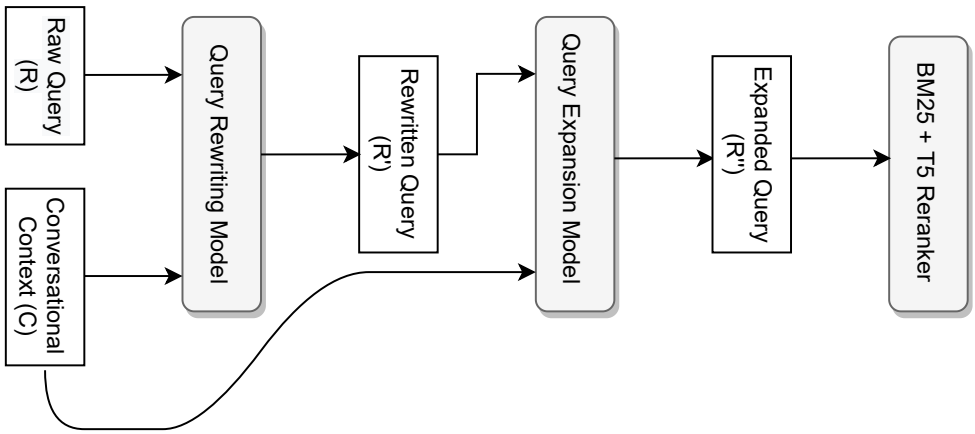

Figure 1: Query rewriting approach. The query rewriting pipeline consists of a two-step process that first rewrites the query with a query rewriting model ($T5_{rewrite}$) and then expands the rewritten query with a query expansion model ($T5_{expand}$). Once the user query is rewritten and expanded, we feed the it into a BM25 + T5 reranker ($T5_{rerank}$).

fine-tuned to produce expansion terms given an input of the raw query and conversational context. The input to $T5_{expand}$ is as such:

$$\text{Previous Turns: } c \text{ Current Turn: } r' \tag{8}$$

where r' is the rewritten query from $T5_{rewrite}$ and c is the conversational context.

4 EXPERIMENTAL RESULTS

In this section, we first describe the datasets used for training our multi-stage pipeline. Next, the training details for the fine-tuning of $T5_{rewrite}$ and $T5_{expand}$ are described. The four proposed systems' are presented alongside their runtags. To conclude the section we compare the results of our CAsT submissions.

4.1 Datasets

The fine-tuning process of $T5_{rewrite}$ and $T5_{expand}$ leverages on the following datasets:

QReCC [2] is a question-answering dataset that includes 14k conversations and 81k question-answer pairs from Natural Questions, TREC CAsT 2019 [6], and QuAC [4]. Each conversation includes a raw query, a conversational context with previous queries and retrieved answers and the rewritten query.

QNLI [25] is a sentence pair classification dataset derived from SQuaD [20]. Given a (question, sentence) pair, the task is predict whether the context sentence contains the answer to the passage or not. QNLI contains 105k training pairs.

4.2 Training

Query Rewriting Model ($T5_{rewrite}$) To fine-tune $T5_{rewrite}$ for query rewriting on QReCC, the raw query is concatenated to the conversational context with the <SEP> token. $T5_{rewrite}$ is trained to produce the rewritten query. In addition, $T5_{rewrite}$ is trained on data from QNLI to take advantage of multi-task learning. We hypothesize that the QNLI task could improve the ability of the query rewriter to infer which context is important for the current utterance. Following Raffel et al. [19], the input to $T5_{rewrite}$ for QNLI is as such:

$$\text{qnli question: } q \text{ sentence: } s \tag{9}$$

where q is the QNLI question and c is the sentence context.

We fine-tune $T5_{rewrite}$ for 10 epochs using the Adafactor optimizer with a learning rate of 2e-5, batch size of 4, and weight decay of 4e-5. We select the epoch with the lowest evaluation loss as our final model for inference.

Query Expansion Model ($T5_{expand}$) When fine-tuning $T5_{expand}$ on QReCC the raw query is concatenated to the conversational context with the <SEP> token. To create expansion terms, we use terms in the rewritten query that are *not* in the raw query. Then the model is trained to produce the expanded terms. We train $T5_{expand}$ is with the same parameters as $T5_{rewrite}$.

4.3 Evaluation

Inference The input used to rewrite and expand queries on CAsT is similar to the input used for training over QReCC. However, CAsT passages are longer than QReCC retrieved answers. Thus, the input to $T5_{rewrite}$ and $T5_{expand}$ is only the *last three* automatic canonical passages. In addition, since $T5_{rewrite}$ was fine-tuned with queries that did not include expansion terms, only the rewritten queries are included in the conversational context. As such the context for a raw query r_i is as follows:

$$ctx(r_i) = \hat{q_1} \oplus \hat{q_2} \oplus \cdots q_{i-3}\hat{} \oplus p_{i-3} \oplus q_{i-2}\hat{} \oplus p_{i-2} \oplus q_{i-1}\hat{} \oplus p_{i-1} \tag{10}$$

where $\hat{q_i}$ is the rewritten query i from $T5_{rewrite}$ and p_i is the automatic canonical passage for q_i.

Official Runs The following describes our official runs submitted to TREC CAsT 2021.

- **LTI-entity-g (section 3.1)**: Is our only manual run, where the manual resolved queries are used to build the conversational entity graph. Query and passage entities are linked, over the top 20 passages, given the $T5_{rerank}$ described in

Runtag	Type	nDCG @3	nDCG @5	nDCG @10	Precision @5	Precision @10
LTI-entity-g	M	0.4618	0.4541	0.4321	0.5848	0.5013
LTI-rewriter-g	A	0.3688	0.3656	0.3593	0.4861	0.4304
LTI-rerwiter-5q	A	0.2955	0.2982	0.2919	0.4076	0.3665
LTI-rerwiter-tc	A	0.3671	0.3651	0.3585	0.4873	0.4291

Table 1: Experimental Results for CAsT 2021 runs. The Type column indicates whether the run is Manual (M), or Automatic (A).

section 3.2. The entities on the query side were manually corrected in case of mislinks to the knowledge base. With the entity graph built over the passages and query we use the entity centrality over the graph to perform a second-step reranking using the PageRank scores.

- **LTI-rewriter-g (section 3.1 and section 3.3.1)**: Is an automatic adaptation of run **LTI-entity-g**, which build the graph with the entities obtained from the rewritten queries, and respective passages obtained with the approach described in section 3.3.1.
- **LTI-rewriter-5q (section 3.3.1)**: Automatic run which uses the top five ranked passages from the *previous* query as input to $T5_{rewrite}$. Rather than appending all five passages into one conversational context, we instead use generate five separate conversation contexts with each of the passages and generate five queries. We then append each *unique* query to each other and feed it into BM25 + $T5_{rerank}$.
- **LTI-rewriter-tc (section 3.3.1 and section 3.3.2)**: Automatic run which first rewrites queries with $T5_{rewrite}$ and then feeds the resolved query into $T5_{expand}$. The rewritten and expanded query is then fed into BM25 + $T5_{rerank}$.

4.4 Results

Table 1 shows the results for each of our submissions to CAsT 2021. The manual run, LTI-entity-g, shows the difference in using the rewritten queries, and indicates that there is room for improvement in our rewriter. Our manual run is slightly below the median, which confirms the strength of $T5_{rerank}$, and how the entity centrality model is unable to detect better documents on lower positions of the ranking.

Across the three automatic runs, we observe that a combination of the $T5_{rewrite}$ and $T5_{expand}$ (LTI-rewriter-tc) outperforms generating multiple queries (LTI-rewriter-5q) – based on different retrieved passages – with $T5_{rewrite}$. We originally hypothesized that generating queries based on multiple top-ranked passages would incorporate additional relevant terms that were missing from the automatic canonical passage. However, given that $T5_{rewrite}$ + $T5_{expand}$ approach only used automatic canonical passages, this difference in performance signifies that using passages that the user did not look at hurts performance.

The entity centrality based system, LTI-rewriter-g, used the queries provided by the system LTI-rewriter-tc to perform the second-stage reranking on top of the $T5_{rerank}$. Hence, the slight improvement when using the entity centrality reranker versus the direct application of the reformulated queries over $T5_{rerank}$. On average the number of relevant documents on the top 5 and top 10 retrieved results stays roughly the same, which indicates that the $T5_{rerank}$ is very strong, and the entity centrality is not able to detect better documents from the top 20 documents that should be placed higher in the ranking.

5 CONCLUSION

This edition of the CAsT track, our models present lower scores when empirically compared to previous editions. This can be due to the increasing dataset difficulty over the past editions. However, we can observe that our query rewritting model trails by a 7-10% gap in terms of precision, which indicates that the query rewriting is a powerful tool to address conversation contextualization, and there is still room for improvement. Moreover, we could identify that explicitly modeling entities is less useful for maintaining context when the pre-trained language model reranker provides a competitive ranking.

REFERENCES

[1] Karl Aberer, Key-Sun Choi, Natasha Noy, Dean Allemang, Kyung-Il Lee, Lyndon Nixon, Jennifer Golbeck, Peter Mika, Diana Maynard, Riichiro Mizoguchi, Guus Schreiber, Philippe Cudré-Mauroux, David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, and Gerhard Weikum (Eds.). 2007. *The Semantic Web: 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007. Proceedings.* Lecture Notes in Computer Science, Vol. 4825. Springer Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-76298-0

[2] Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. 2020. Open-domain question answering goes conversational via question rewriting. *arXiv preprint arXiv:2010.04898* (2020).

[3] Sergey Brin and Lawrence Page. 1998. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems* 30, 1-7 (April 1998), 107–117. https://doi.org/10.1016/S0169-7552(98)00110-X

[4] 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 *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). Association for Computational Linguistics, 2174–2184. https://doi.org/10.18653/v1/d18-1241

[5] Seungjin Choi. 2005. On Variations of Power Iteration. In *Artificial Neural Networks: Formal Models and Their Applications - ICANN 2005, 15th International Conference, Warsaw, Poland, September 11-15, 2005, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 3697)*, Wlodzislaw Duch, Janusz Kacprzyk, Erkki Oja, and Slawomir Zadrozny (Eds.). Springer, 145–150. https://doi.org/10.1007/11550907_24

[6] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2019. CAsT 2019: The Conversational Assistance Track Overview. In *Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC 2019, Gaithersburg, Maryland, USA, November 13-15, 2019 (NIST Special Publication, Vol. 1250)*, Ellen M. Voorhees and Angela Ellis (Eds.). National Institute of Standards and Technology (NIST).

[7] Ahmed Elgohary, Denis Peskov, and Jordan Boyd-Graber. 2019. Can you unpack that? learning to rewrite questions-in-context. *Can You Unpack That? Learning to Rewrite Questions-in-Context* (2019).

[8] Paolo Ferragina and Ugo Scaiella. 2012. Fast and Accurate Annotation of Short Texts with Wikipedia Pages. *IEEE Software* 29, 1 (2012), 70–75. https://doi.org/10.1109/MS.2011.122

[9] Hao Huang, Xiubo Geng, Jian Pei, Guodong Long, and Daxin Jiang. 2021. Reasoning over Entity-Action-Location Graph for Procedural Text Understanding. 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)*. Association for Computational Linguistics, Online, 5100–5109. https://doi.org/10.18653/v1/2021.acl-long.396

[10] Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-Relation Extraction as Multi-Turn Question Answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1340–1350. https://doi.org/10.18653/v1/P19-1129

[11] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. Contextualized Query Embeddings for Conversational Search. *arXiv preprint arXiv:2104.08707* (2021).

[12] 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 Transactions on Information Systems (TOIS)* 39, 4 (2021), 1–29.

[13] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. DBpedia Spotlight: Shedding Light on the Web of Documents. In *Proceedings of the 7th International Conference on Semantic Systems - I-Semantics '11*. ACM Press, Graz, Austria, 1–8. https://doi.org/10.1145/2063518.2063519

[14] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-Ranking with BERT. *CoRR* abs/1901.04085 (2019). arXiv:1901.04085

[15] Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. *arXiv preprint arXiv:2003.06713* (2020).

[16] Chen Qu, Liu Yang, Minghui Qiu, W. Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. 2019. BERT with History Answer Embedding for Conversational Question Answering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). ACM, 1133–1136. https://doi.org/10.1145/3331184.3331341

[17] Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W. Bruce Croft, and Mohit Iyyer. 2019. Attentive History Selection for Conversational Question Answering. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, Wenwu Zhu, Dacheng Tao, Xueqi Cheng, Peng Cui, Elke A. Rundensteiner, David Carmel, Qi He, and Jeffrey Xu Yu (Eds.). ACM, 1391–1400. https://doi.org/10.1145/3357384.3357905

[18] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[19] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683* (2019).

[20] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250* (2016).

[21] Chinnadhurai Sankar, Sandeep Subramanian, Chris Pal, Sarath Chandar, and Yoshua Bengio. 2019. Do Neural Dialog Systems Use the Conversation History Effectively? An Empirical Study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 32–37. https://doi.org/10.18653/v1/P19-1004

[22] Svitlana Vakulenko, Shayne Longpre, Zhucheng Tu, and Raviteja Anantha. 2021. Question rewriting for conversational question answering. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 355–363.

[23] Svitlana Vakulenko, Nikos Voskarides, Zhucheng Tu, and Shayne Longpre. 2021. A comparison of question rewriting methods for conversational passage retrieval. *arXiv preprint arXiv:2101.07382* (2021).

[24] Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. 2020. Query resolution for conversational search with limited supervision. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 921–930.

[25] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461* (2018).

[26] Shi Yu, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu. 2021. Few-Shot Conversational Dense Retrieval. *arXiv preprint arXiv:2105.04166* (2021).