Combining Lexical and Dense Retrieval for Computationally Efficient Multi-hop Question Answering

Sidiropoulos, G.; Voskarides, N.; Vakulenko, S.; Kanoulas, E.

DOI
10.18653/v1/2021.sustainlp-1.7

Publication date
2021

Document Version
Final published version

Published in
Proceedings of SustaiNLP

License
Article 25fa Dutch Copyright Act

Link to publication

Citation for published version (APA):
Sidiropoulos, G., Voskarides, N., Vakulenko, S., & Kanoulas, E. (2021). Combining Lexical and Dense Retrieval for Computationally Efficient Multi-hop Question Answering. In *Proceedings of SustaiNLP: 2nd Workshop on Simple and Efficient Natural Language Processing (SustaiNLP) : SustaiNLP 2021* (pp. 58–63). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.sustainlp-1.7

Combining Lexical and Dense Retrieval for Computationally Efficient Multi-hop Question Answering

Georgios Sidiropoulos[1], **Nikos Voskarides**[2*], **Svitlana Vakulenko**[1], **Evangelos Kanoulas**[1]
[1] University of Amsterdam, Amsterdam, The Netherlands
[2] Amazon, Barcelona, Spain
g.sidiropoulos@uva.nl, nvvoskar@amazon.com, s.vakulenko@uva.nl, e.kanoulas@uva.nl

Abstract

In simple open-domain question answering (QA), dense retrieval has become one of the standard approaches for retrieving the relevant passages to infer an answer. Recently, dense retrieval also achieved state-of-the-art results in multi-hop QA, where aggregating information from multiple pieces of information and reasoning over them is required. Despite their success, dense retrieval methods are computationally intensive, requiring multiple GPUs to train. In this work, we introduce a hybrid (lexical and dense) retrieval approach that is highly competitive with the state-of-the-art dense retrieval models, while requiring substantially less computational resources. Additionally, we provide an in-depth evaluation of dense retrieval methods on limited computational resource settings, something that is missing from the current literature.

1 Introduction

Multi-hop QA requires retrieval and reasoning over multiple pieces of information (Yang et al., 2018). For instance, consider the multi-hop question: "Where is the multinational company founded by Robert Smith headquartered?". To answer this question we first need to retrieve the passage about Robert Smith in order to find the name of the company he founded (General Mills), and subsequently retrieve the passage about General Mills, which contains the answer the question (Golden Valley, Minnesota). Even though multi-hop QA requires multiple retrieval hops, it is fundamentally different from session search (Yang et al., 2015; Levine et al., 2017) and conversational search (Dalton et al., 2019; Voskarides et al., 2020; Vakulenko et al., 2021), since in multi-hop QA the information need of the user is expressed in a single question, thus not requiring multiple turns of interaction.

QA systems typically consist of (i) a retriever that identifies the passage/document in the underlying collection that contains the answer to the user's question, and (ii) a reader that extracts or generates the answer from the identified passage (Chen et al., 2017). Given that often the answer cannot be found in the top-ranked passage, inference follows a standard beam-search procedure, where top-k passages are retrieved and the reader scores are computed for all k passages (Lee et al., 2019). However, readers are very sensitive to noise in the top-k passages, thus making the performance of the retriever critical for the performance of QA systems (Yang et al., 2019). This is further amplified in multi-hop QA, where multiple retrieval hops are performed; potential retrieval errors get propagated across hops and thus severely harm QA performance.

The majority of current approaches to multi-hop QA use either traditional IR methods (TF-IDF, BM25) (Qi et al., 2019) or graph-based methods for the retriever (Nie et al., 2019; Asai et al., 2020). However, those approaches have serious limitations. The former approaches require high lexical overlap between questions and relevant passages, while the latter rely on an interlinked underlying corpus, which is not always the case. Recently, Xiong et al. (2021) introduced a dense multi-hop passage retrieval model that constructs a new query representation based on the question and previously retrieved passages and subsequently uses the new representation to retrieve the next set of relevant passages. This model achieved state-of-the-art results while not relying on an interlinked underlying corpus.

Even though dense retrieval models achieve state-of-the-art results on multi-hop QA, they are computationally intensive, requiring multiple GPUs to train. Existing work only reports results for the cases where such resources are available; therefore providing no answer on how feasible is to train such models on a low resource setting. In this

[*] Research conducted when the author was at the University of Amsterdam.

Proceedings of the 2nd Workshop on Simple and Efficient Natural Language Processing, pages 58–63
November 10, 2021. ©201 Association for Computational Linguistics

paper, we focus on developing an efficient retriever for multi-hop QA that can be trained effectively in a low computational resource setting. We aim to answer the following research questions:

RQ1 How does the performance of dense retrieval compare to lexical and hybrid approaches?

RQ2 How does the performance degrade in the low computational resource settings?

Our main contributions are the following: (i) we propose a hybrid (lexical and dense) retrieval model which is competitive against its fully dense competitors while requiring eight times less computational power and (ii) we perform a thorough analysis on the performance of dense passage retrieval models on the task of multi-hop QA.[1]

2 Task Description

Let $p \in \mathcal{C}$ denote a passage within a passage collection $\mathcal{C}$, and q a multi-hop question. Given q and $\mathcal{C}$ the task is to retrieve a set of relevant passages $\mathcal{P} = \{p_1, p_2, \ldots, p_n\}$, where $p_i \in \mathcal{C}$. In the multi-hop scenario we consider here, not all relevant passages can be retrieved using the input question q alone. This is due to the fact that there is a low lexical overlap or semantic relationship between question q and one or more of the relevant passages in $\mathcal{P}$. In this case, information from one of the relevant passages p_i is needed to retrieve another relevant passage p_j, where p_j may be lexically/semantically different from question q.

3 Experimental Setup

In this section, we describe the dataset used in our experiments, the metrics we use to answer our research questions and the models we compare against.

3.1 Dataset

For our experiments, we focus on the HotpotQA dataset and particularly the full-wiki setting (Yang et al., 2018). HotpotQA is a large-scale 2-hop QA dataset where the answers to questions must be found in the context of the entire Wikipedia. Questions in HotpotQA fall into one of the following categories: bridge or comparison. In *bridge* questions, the bridge entity that connects the two relevant passages is missing; e.g., 'When did the show that Skeet Ulrich is currently starring in premiere?", where the bridge entity "Riverdale (2017 TV series)" is missing. In *comparison* questions the main two entities (of the two relevant passages) are both mentioned and compared; e.g., "Which has smaller flowers, Campsis or Kalmiopsis?".

3.2 Metrics

Following previous work (Yang et al., 2018; Xiong et al., 2021), we report passage Exact Match (EM) to measure the overall retrieval performance. Exact Match is a metric that evaluates whether both of the ground-truth passages for each question are included in the retrieved passages (then EM=1 otherwise EM=0). Note that metrics such as EM and F1 w.r.t question's answer (Ans) and supporting facts on sentence-level (Sup) do not fit in our experimental setup since we focus on the retrieval part of the pipeline and not on the reading.

3.3 Models

In this section, we describe the models we experiment with.

3.3.1 Single-hop models

Given a question, single-hop models retrieve a ranked list of passages. Thus, they are not aware of the multi-hop nature of the task.

BM25 is a standard lexical retrieval model. We use the default Anserini parameters (Yang et al., 2017).

Rerank is a standard two-stage retrieval model that first retrieves passages with BM25 and then uses BERT to rerank the top-k passages (Nogueira and Cho, 2019). The BERT (base) classifier was trained with a point-wise loss (Nogueira and Cho, 2019). It was fine-tuned on the train split of HotpotQA for 2 epochs. Training took 5 hours with batch size of 8 using a single 12GB GPU. We experimented with k=100 and k=1000, and found that k=100 results in a better reranking performance at the top positions.

DPR is a dense passage retrieval model for simple questions (Karpukhin et al., 2020). Given a question q, a relevant passage p^+ and a set of irrelevant passages $\{p_1^-, p_2^-, \ldots, p_m^-\}$, the model learns to rank p^+ higher via the optimization of the negative log likelihood of the relevant passage. To train DPR on HotpotQA, a multi-hop QA dataset, we follow the procedure described in (Xiong et al., 2021). This model was trained for 25 epochs ($\sim$ 2 days)

[1] Our trained models and data are available at https://github.com/GSidiropoulos/hybrid_retrieval_for_efficient_qa.

on a single 12GB GPU, using a RoBERTa-based encoder.

3.3.2 Multi-hop models

These models are aware of the multi-hop nature of the task. They recursively retrieve new information at each hop by conditioning the question on information retrieved on previous hops (Xiong et al., 2021). In practice, at each hop t the question q and the passage retrieved in the previous hop p_{t-1} gets encoded as the new query $q_t = h(q, p_{t-1})$, where $h(\cdot)$ the question encoder, to retrieve the next relevant passage; when $t = 1$ then we have just the question. Differently from single-hop models, at inference time, given a question, beam search is used to obtain the top-k passage pair candidates. The candidates to beam search at each hop are generated by a similarity function using the query representation at hop t, and the beams are scored by the sum of the individual similarity scores.

MDR is a state-of-the-art dense retriever for multi-hop questions (Xiong et al., 2021). It extends DPR in an iterative fashion by encoding the question and passages retrieved in previous hops as the new query to retrieve the next relevant passages. This model was trained for 25 epochs ($\sim$ 3 days) on a single 12GB GPU, using a RoBERTa-based encoder, without the memory bank mechanism (Wu et al., 2018). The memory bank mechanism is dropped since it is very expensive to compute and its contribution to retrieval performance is limited.

MDR (full) is MDR with the additional memory bank mechanism, trained for 50 epochs on 8×32GB GPUs by Xiong et al. (2021).

Rerank + DPR$_2$ is a hybrid model we propose in this paper. Specifically, for the first hop we rely on the BERT-based re-ranking model described in Section 3.3.1 (Rerank), while for the second hop we train a DPR only on second hop questions (DPR$_2$). To train the latter, we build a variation of HotpotQA where the question gets concatenated with the ground truth passage of the first hop, and the second hop ground truth passage is the only relevant passage to be retrieved. DPR$_2$ was trained for 25 epochs ($\sim$ 1 day) on a single 12GB GPU, using a RoBERTa-based encoder.

4 Results

In this section, we present our experimental results that answer our research questions.

Model	EM@2	EM@10	EM@20
BM25	0.127	0.320	0.395
Rerank	0.314	0.476	0.517
DPR	0.116	0.275	0.336
MDR	0.440	0.581	0.619
Rerank+DPR$_2$	0.599	0.732	0.762
MDR (full)	**0.677**	**0.772**	**0.793**

Table 1: Overall retrieval performance. In the first group we show single-hop models, while in the second group we show multi-hop models.

4.1 Overall performance

Here we aim to answer **RQ1** by comparing the retrieval performance of the models we consider. In Table 1, we observe that the single-hop models perform much worse than the multi-hop models. This is expected since single-hop models are not aware of the multi-hop nature of the task.

As for the multi-hop models, we observe that MDR (full) achieves the best performance at the higher positions in the ranking. It is important to underline here that MDR (full) uses considerably more resources than Rerank+DPR$_2$ and MDR. The last two use relatively limited computational resources and a comparison between them is more fair (see Section 3.3).[2] We observe that our Rerank+DPR$_2$ outperforms MDR on all metrics while is also competitive against MDR (full), especially w.r.t EM@10 and EM@20. This is due to the fact that often questions and their relevant passages are not only semantically related, but also have high lexical overlap. This is also highlighted by Karpukhin et al. (2020), who reported that dense retrieval has performance issues when the question has high lexical overlap with the passages.

4.2 Performance for limited resources

In this section, we answer **RQ2** by comparing the retrieval performance of MDR and DPR as provided in (Xiong et al., 2021), against the same models trained with limited computational resources.

In Table 2 we see that performance drops significantly as we limit resources for both DPR and MDR. This is a result of the training scheme that is used in (Karpukhin et al., 2020) and (Xiong et al.,

[2]Even though the memory bank mechanism is omitted from MDR, the comparison of Rerank+DPR$_2$ and MDR remains fair since this particular mechanism can also be potentially applied to Rerank+DPR$_2$ (in the DPR part).

Model	Encoder	# GPU	# Epochs	Batch size	# Gradient acc. steps	EM@2	EM@10	EM@20
MDR (full)	RoBERTa	8×32GB	50	150	1	0.677	0.772	0.793
MDR	RoBERTa	8×32GB	50	150	1	0.637	0.742	0.772
	RoBERTa	4×24GB	50	28	1	0.606	0.711	0.735
	RoBERTa	4×24GB	25	28	1	0.550	0.668	0.698
	RoBERTa	4×24GB	20	28	1	0.537	0.659	0.687
	RoBERTa	1×12GB	25	4	32	0.440	0.581	0.619
	BERT	1×12GB	25	4	32	0.421	0.560	0.597
DPR	RoBERTa	8×32GB	50	256	1	0.252	0.454	0.521
	RoBERTa	4×24GB	25	128	1	0.223	0.427	0.487
	RoBERTa	1×12GB	25	8	32	0.116	0.275	0.336

Table 2: Analysis of how computational resources affect the performance of MDR and DPR. The MDR(full) configuration is provided by (Xiong et al., 2021). Different beam size can slightly change the results from what was originally reported. MDR and DPR, both trained trained on 8 GPUs, are not available and therefore we report the results as they were reported in (Xiong et al., 2021).

2021). More specifically, DPR and MDR rely on using in-batch negatives both for decreasing the training time (positive passages of a question are reused as negative passages for the rest of the questions in the batch, instead of having to sample new ones beforehand), and for improving accuracy (bigger batch size will produce more in-batch negatives, thus increasing the number of training samples). When we have limited resources, training time gets significantly longer (since we use fewer GPUs), and therefore we have to compromise for fewer training epochs while the batch size is restricted by the GPU memory size. For instance, training for 50 epochs takes $\sim$1 day on $8\times$32GB GPUs, while it takes $\sim$6 days on a single 12GB GPU.

In addition, when comparing MDR trained on $4\times$24GB GPUs against MDR trained on a single 12GB GPU, for 25 epochs each, we observe that even though we can simulate bigger batch sizes by using gradient accumulation,[3] we do not observe an increase in performance. This is a consequence of the fact that the number of in-batch negatives is limited by the real batch size. Note that we also observe a similar trend for DPR.

In summary, computational resources are of vital importance for multi-hop dense retrieval models. Hence, in the case where only limited resources are available, following a hybrid (lexical and dense) approach such as our proposed Rerank+DPR$_2$ seems to be a good choice. As we showed in Tables 1 and 2, Rerank+DPR$_2$ (trained on a single GPU) performs similarly to MDR trained on 4 GPUs and is competitive against MDR trained on 8 GPUs.

[3]Gradient accumulation is a mechanism that accumulates the gradients and the losses over consecutive batches for a specific number of steps (without updating the parameters), and then updates the parameters based on the cumulative gradient (Mnih et al., 2016).

4.3 Error Analysis

We perform qualitative analysis to gain further insights into where the models succeed or fail. More specifically, we compare specific cases where our hybrid model (Rerank+DPR$_2$) retrieves both relevant passages successfully while MDR fails and vice versa. For our analysis we focus on bridge questions since comparison questions are more straightforward to retrieve.

Table 3 shows two typical examples of questions for which Rerank+DPR$_2$ retrieves both relevant passages at the top positions while MDR fails to do so. When there is a high lexical overlap between the question and a relevant passage, our hybrid model can capture this exact n-gram match and improve the performance. In contrast, fully dense models seem incapable of capturing this. In particular, this lexical overlap can be between the question and both relevant passages for the case of comparison questions, or between the question and the first relevant passage for bridge questions.

In bridge questions if the lexical overlap is between the question and the second passage then our hybrid model favors passages in which this phrase appears, and therefore it retrieves an irrelevant first passage; leading to an irrelevant second passage as well. MDR on the other hand manages to retrieve both relevant passages at the top positions. Those are the cases where the lexical overlap is used in the given question in order to disambiguate the final answer. Examples can be found in Table 4.

In the first example, “Golden Globe Award” is

	q **[bridge]: Who was the defense counsel of a German woman who underwent Catholic exorcism rites during the year before her death?**
Rerank+DPR$_2$	**Anneliese Michel**: Anneliese Michel] (21 September 1952 – 1 July 1976) was a German woman who underwent Catholic exorcism rites during the year before her death. Later investigation determined that she was malnourished and dehydrated... **Erich Schmidt-Leichner**: Erich Schmidt-Leichner (14 October 1910 – 17 March 1983) was a German lawyer who made a name as a distinguished defense counsel at the Nuremberg Trials (1945 - 1946). In 1978, he was a defense counsel in the "Klingenberg Case" (Anneliese Michel)...
MDR	**Maria Pauer**: Maria Pauer (October 1734/36 – 1750 in Salzburg), was an alleged Austrian witch. She was the last person to be executed for witchcraft in Austria. **Franz Burkard (died 1539)**: Franz Burkard (died 1539) was a canon lawyer in Ingoldstadt who opposed Lutheranism, particularly in the trial of Andreas Seehofer.
	q **[bridge]: The astronomer who formulated the theory of stellar nucleosynthesis co-authored what landmark paper?**
Rerank+DPR$_2$	**Fred Hoyle**: Sir Fred Hoyle FRS (24 June 1915 – 20 August 2001) was an English astronomer who formulated the theory of stellar nucleosynthesis ... **B2FH paper**: The BFH paper, named after the initials of the authors of the paper, Margaret Burbidge, Geoffrey Burbidge, William Fowler, and Fred Hoyle, is a landmark paper of stellar physics published in "Reviews of Modern Physics" in 1957 ...
MDR	**Sequence hypothesis**: The sequence hypothesis was first formally proposed in the review "On Protein Synthesis" by Francis Crick in 1958 ... **Francis Crick**: Francis Harry Compton Crick (8 June 1916 – 28 July 2004) was a British molecular biologist, biophysicist, and neuroscientist, most noted for being a co-discoverer of the structure of the DNA molecule in ...

Table 3: Two example bridge questions for which Rerank+DPR$_2$ retrieves both relevant passages at the top positions while MDR fails to do so. Lexical overlap is indicated with underlined text while the answer to the question is highlighted.

	q **[bridge]: What Golden Globe Award actor starred in the film Little Fugitive?**
Rerank+DPR$_2$	**Ali MacGraw:** Elizabeth Alice "Ali" MacGraw (born April 1, 1939) is an American actress, model, author, and animal rights activist. She first gained attention with her role in the 1969 film "Goodbye, Columbus", for which she won the Golden Globe Award for Most Promising Newcomer. She reached international fame in 1970's "Love Story", for which she was nominated for an Academy Award for Best Actress and won the Golden Globe Award for Best Actress in a Motion Picture – Drama ... **Little Fugitive**: Little Fugitive (1953) is an American film written and directed by Raymond Abrashkin (as "Ray Ashley"), Morris Engel and Ruth Orkin, that tells the story of a child alone in Coney Island.
MDR	**Little Fugitive (2006 film)**: Little Fugitive is a 2006 remake of the film of the same name. It was directed by Joanna Lipper and produced by Nicholas Paleologos. The film is set in present day Brooklyn and tells the story of 11-year-old Lenny (Nicolas Martí Salgado) who must take care of his 7-year-old brother, Joey (David Castro), while their father (Peter Dinklage) is in jail and their mother work long hours a nursing home ... **Peter Dinklage**: Peter Hayden Dinklage (, born June 11, 1969) is an American actor and film producer. He has received numerous accolades, including a Golden Globe Award and two Primetime Emmy Awards.
	q **[bridge]:Which American professional poker player also starred in the 2015 movie "Extraction"?**
Rerank+DPR$_2$	**Allen Cunningham**: Allen Cunningham (born March 28, 1977) is an American professional poker player who has won five World Series of Poker bracelets. **Extraction (film)**: Extraction is a 2015 American action-thriller film directed by Steven C. Miller and written by Umair Aleem. The film stars Kellan Lutz, Bruce Willis, Gina Carano, D. B. Sweeney, Dan Bilzerian and Steve Coulter ...
MDR	**Extraction (film)**: Extraction is a 2015 American action-thriller film directed by Steven C. Miller and written by Umair Aleem. The film stars Kellan Lutz, Bruce Willis, Gina Carano, D. B. Sweeney, Dan Bilzerian and Steve Coulter ... **Dan Bilzerian**: Dan Brandon Bilzerian (born December 7, 1980) is an American professional poker player.

Table 4: Two example bridge questions for which MDR retrieves both relevant passages at the top positions while Rerank+DPR$_2$ fails to do so. The answer to the question is highlighted.

used in the given question in order to disambiguate the final answer, since in the film “Little Fugitive” there is more than one actor involved. Therefore, “Golden Globe Award” must be used to assist the retrieval of the second passage. Since Rerank+DPR$_2$ builds on top of BM25, it favors passages in which this phrase appears, and therefore it retrieves an irrelevant first passage leading to an irrelevant second passage as well. On the other hand, MDR manages to retrieve both relevant passages at the top positions. Similarly, for the second example, “American professional poker player” is used to specify the actor that starred in the “Extraction” movie, hence supporting the retrieval of the second relevant passage.

5 Conclusion

In this work, we provided insights on the performance of state-of-the-art dense retrieval for multi-hop questions. We showed that Rerank+DPR$_2$ (our hybrid model) outperforms MDR (the state-of-the-art multi-hop dense retrieval model) in the low resource setting, and it is competitive with MDR in the setting where MDR uses considerably more computational resources. Finally, we highlighted that fully dense retrieval models get harmed when using limited computational resources. For future work, we plan to build on our insights to improve the performance of multi-hop models by combining the strengths of lexical and dense retrieval. Also, we aim to develop less computationally expensive multi-hop retrieval models.

Acknowledgements

This research was supported by the NWO Innovational Research Incentives Scheme Vidi (016.Vidi.189.039), the NWO Smart Culture - Big Data / Digital Humanities (314-99-301), the H2020-EU.3.4. - SOCIETAL CHALLENGES - Smart, Green And Integrated Transport (814961). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1870–1879. Association for Computational Linguistics.

Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2019. Trec cast 2019: The conversational assistance track overview. *Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC*.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096.

Nir Levine, Haggai Roitman, and Doron Cohen. 2017. An extended relevance model for session search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 865–868.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR.

Yixin Nie, Songhe Wang, and Mohit Bansal. 2019. Revealing the importance of semantic retrieval for machine reading at scale. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2553–2566.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with BERT. *CoRR*, abs/1901.04085.

Peng Qi, Xiaowen Lin, Leo Mehr, Zijian Wang, and Christopher D. Manning. 2019. Answering complex open-domain questions through iterative query generation. In *2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

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*, pages 355–363.

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*, pages 921–930.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, et al. 2021. Answering complex open-domain questions with multi-hop dense retrieval. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*.

Hui Yang, Dongyi Guan, and Sicong Zhang. 2015. The query change model: Modeling session search as a markov decision process. *ACM Transactions on Information Systems (TOIS)*, 33(4):1–33.

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1253–1256.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with bertserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Demonstrations*, pages 72–77. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.