DCU-ADAPT@TREC iKAT 2024: Incorporating Retrieved Knowledge for Enhanced Conversational Search

Praveen Acharya*
praveen.acharya2@mail.dcu.ie
Dublin City University
Dublin, Ireland

Xiao Fu*
xiao.fu.20@ucl.ac.uk
University College London
London, UK

Noriko Kando
kando@nii.ac.jp
National Institute of Informatics
Tokyo, Japan

Gareth J. F. Jones
gareth.jones@dcu.ie
Dublin City University
Dublin, Ireland

Abstract

Users of search applications often encounter difficulties in expressing their information needs effectively. Conversational search (CS) can potentially support users in creating effective queries by enabling a multi-turn, iterative dialogue between a *User* and the search *Systems*. These dialogues help users to refine and build their understanding of their information need through a series of query-response exchanges. However, current CS systems generally do not accumulate knowledge about the user's information needs or the content with which they have engaged during this dialogue. This limitation can hinder the system's ability to support users effectively. To address this issue, we propose an approach that seeks to model and utilize knowledge gained from each interaction to enhance future user queries. Our method focuses on incorporating knowledge from retrieved documents to enrich subsequent user queries, ultimately improving query comprehension and retrieval outcomes. We test the effectiveness of our proposed approach in our TREC iKAT 2024 participation.

CCS Concepts

• **Information systems** → *Query reformulation*; *Query representation*; • **Human-centered computing** → *Contextual design.*

Keywords

Query Enrichment, Knowledge Integration, Conversational Search

1 Introduction

Users of search applications possess varying levels of familiarity with the topic of their search and will have different levels of knowledge about their information need. This difference in knowledge affects how users construct their search queries, leading to varying levels of query precision and overall search success. For instance, a knowledgeable user can articulate their information need in detail, enabling the search system to retrieve relevant documents effectively. In contrast, an ill-informed user may struggle to clearly define their information need [4], resulting in vague or under-specified queries and, as a result, poor retrieval of relevant content. Consequently, a user's knowledge of the topic significantly impacts the search process, making a search system that can adapt to this varying level of knowledge highly valuable.

Conversational search (CS) approach allows users to meet their information needs without having to create a perfectly defined query. Instead, it offers a process where users can address their needs through a series of iterative query-response exchanges. This process allows users to refine their queries based on feedback from previous search results. As users gather more information, their understanding of the topic improves, helping them craft more accurate queries. This ongoing cycle of query refinement and knowledge accumulation is particularly important for addressing complex information needs. Therefore, a search system built for exploratory information retrieval should support this iterative process, guiding users through multiple stages of query development and refinement. In doing so, such a system helps users navigate large information spaces, gradually narrowing down to the exact information they need.

The role of knowledge is even more critical in CS systems, as it aids in better understanding user needs and facilitates smooth communication. By capturing and applying this knowledge, search systems can adapt the conversation in real-time, taking into account both the user's prior knowledge and the new insights gathered [1]. This continuous adjustment helps refine the interaction, ensuring that the user's information needs are progressively met. Following this argument, in this paper, we investigate the approach of incorporating knowledge in conversational search dialogue and report our submission runs and results for the TREC iKAT 2024 task to study its effectiveness.

2 Related Works

Research on users' knowledge in retrieval systems has been explored in several studies including [6, 10, 15, 16]. For instance, Câmara et al. [5] proposed a keyword-based approach to representing knowledge during a search session. They also introduced a large language model-based method for knowledge representation, demonstrating that both approaches effectively gauge the user's understanding of specific topics. This model continuously updates an internal representation throughout the session, adapting to the user's interactions. Further advancements in this area have

*Work done while at National Institute of Informatics.

TREC '24, November 18–22, 2024, Rockville, MD

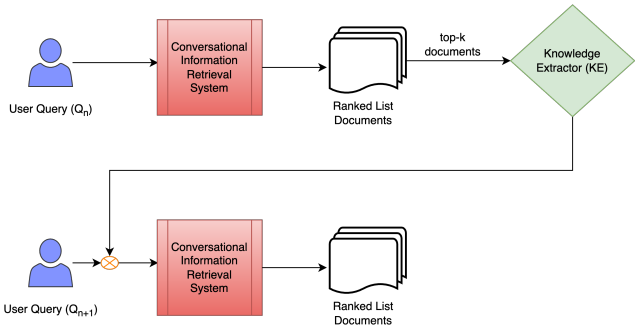

Figure 1: Overview of the proposed methodology of incorporating retrieved knowledge in Conversational Search.

incorporated named entities [8] and knowledge graphs [14], providing a more nuanced understanding of a user's knowledge during their search processes. Although the significance of knowledge in the search process and its effectiveness have been highlighted in previous research, the formal integration of knowledge into conversational search remains surprisingly under-explored. Acharya [1] argue that knowledge should be modelled to support search systems in better understanding the user's information need and propose a framework [2]. Leveraging a user's prior knowledge and their knowledge of the search topic accumulated during a conversational dialogue would appear to have the potential to significantly enhance CS system performance by guiding subsequent actions resulting in more efficient and effective search outcomes.

3 Methodology

In this section, we describe the methodology for our proposed approach. The core idea is that, in CS, each turn of the dialogue exposes the user to documents deemed relevant by the search system. As the user interacts with these documents, their understanding of the topic evolves, which can affect the queries they submit in subsequent dialogue turns. To capture this evolving user knowledge, we propose extracting knowledge from the documents retrieved during each turn, or from specific documents that the user engages with. This knowledge reflects the information the user has been exposed to and can help the system provide more relevant results by leveraging the user's growing understanding of the topic. The overall framework for this approach is illustrated in Figure 1, which outlines the steps involved in the knowledge integration, from document retrieval to knowledge extraction and query enhancement for subsequent turns across the conversation.

We incorporate a ***Knowledge Extractor (KE)*** component within the search process [2]. Its primary function is to extract knowledge from the documents retrieved in response to user queries during each dialogue turn. This extracted knowledge serves as a representation of the most relevant information the user has been exposed to, which may help the CS in subsequent search turns by allowing it to better capture the user's potentially shifting information needs over multiple turns in the conversation.

4 Experimental Investigation

In this section, we report our experimental investigation into the effectiveness of our proposed method of incorporating retrieved knowledge in conversational search.

4.1 Experimental Setup

4.1.1 ***Datasets.*** Our experimental investigation makes use of the iKAT 2024 task datasets. The dataset has been created around multi-turn dialogues between a user and a system, where the system must interpret and respond to a sequence of evolving queries. The primary task of the dataset is to evaluate the ability of a search system to handle multi-turn conversational interactions and how well it adpats to the user's changing query intent while maintaining context across turns. The document collection is the TREC iKAT ClueWeb22-B Passage Collection [3].

4.1.2 ***Implementation Details.*** We implement a two-stage retrieval pipeline in our experiments. The first stage is the retrieval stage which uses sparse retrieval (BM25) followed by a re-ranking stage using a cross-encoder to evaluate the effectiveness of our proposed approach.

Indexing. We index the collection for searching using Pyserini [12]. The passages are processed using the default indexing configuration.

Retrieval. We conduct retrieval using `BM25` [11] with `k1=1.2`, `b=0.75`.

5 Submitted Runs

We submitted 6 runs (4 Automatic runs, 2 Manual runs), for the Passage ranking and PTKB ranking task. We provide a short description of each run below:

5.1 Manual Runs

5.1.1 ***dcu_manual_qe_summ_TopP_3.*** In this run we use BM25 for first-stage retrieval to retrieve top 1K passages followed by second-stage re-ranking using cross-encoder. An abstractive summary is then generated based on the top-3 passages retrieved in the first stage which is considered extracted knowledge and used to enrich the query in the subsequent turn. For re-ranking, we use a pre-trained cross-encoder *ms-marco-MiniLM-L-6-v2*. For rewriting, We use a T5 query rewriter [13] fine-tuned on the CANARD dataset [9]. For abstractive summarizer, we use *pegasus-xsum* [17].

5.1.2 ***dcu_manual_qe_summ_ptkb_TopP_3.*** In this run we use manually rewritten utterances and ground-truth PTKB provenance statements in the dataset. Re-ranking is done using *cross-encoder/ms-marco-MiniLM-L-6-v2* and the abstractive summary is generated using *pegasus-xsum BM25 Clueweb-22*. BM25 is used for first-stage retrieval to retrieve top 1K passages followed by re-ranking using the cross-encoder. An abstractive summary is generated based on the top-3 passages retrieved by BM25 and is used to enrich the user query in the subsequent turn.

5.2 Automatic Runs

5.2.1 ***dcu_auto_qe_key_topP-50_topK-5.*** In this automatic run, BM25 is used in the first stage retrieval to retrieve the top 1K

passages followed by re-ranking using cross-encoder. We extract top-5 key terms from the user utterance and the top-50 passages retrieved in the first stage and use these key terms to enrich the query in the subsequent turn. PTKB provenance ranking is done by selecting the top-3 PTKB with the highest cosine similarity to the enriched query. We use *cross-encoder/ms-marco-MiniLM-L-6-v2* for re-ranking and Sentence-BERT model (paraphrase-MiniLM-L6-v2) to determine PTKB with the highest similarity to the enriched user query for PTKB ranking task. Key term extraction to enrich queries is done using YAKE [7].

5.2.2 ***dcu_auto_qre_sim.*** In this automatic run, related historical queries from the conversational context based on similarity scores to the user utterance are used as additional query context. The user utterance and query context are used to rewrite the query using a T5-based Query rewriter fine-tuned on the CANARD Dataset. BM25 is used to retrieve the top 1K passages followed by re-ranking using cross-encoder. PTKB provenance ranking is done by selecting the top-3 PTKB with the highest cosine similarity to the rewritten query.

5.2.3 ***dcu_auto_qe_summ_TopP_3.*** In this run, the user utterance is used for the query. BM25 is used to retrieve the top 1K passages followed by re-ranking using cross-encoder. We use an abstractive summary from the top 3 passages retrieved from BM25 which is considered as the extracted knowledge and is used to enrich the query in the subsequent turn. PTKB provenance ranking is done by selecting the top-3 PTKB with the highest cosine similarity to the enriched query.

5.2.4 ***dcu_auto_qe_summ_ptkb_TopP_3.*** In this run, the user utterance along with the top-3 PTKB with the highest similarity to the current user query is used. BM25 is used to retrieve the top 1K passages followed by re-ranking using cross-encoder. We use an abstractive summary from the top 3 passages retrieved from BM25 which is considered extracted knowledge and is used to enrich the user query in the subsequent turn. PTKB provenance ranking is done by selecting the top-3 PTKB with the highest cosine similarity to the enriched query.

6 Results and Discussion

Table 1, shows the results of the various retrieval system configurations, based on key evaluation metrics: *ndcg@*5, *ndcg*, *Recall@20*, and *MAP*. The configurations tested in this experiment vary in terms of manual and automatic configuration with a focus on query reformulation by enriching or expanding user queries by extracting key terms or summarizing retrieved passages, to study the impact on retrieval performance.

The configuration `dcu_manual_qe_summ_TopP_3` demonstrated moderate performance, achieving an `ndcg@5` of 0.2174 and an `ndcg` of 0.1966. The `Recall@20` for this run was relatively low at 0.0732, and `MAP` was modest at 0.0783. These results suggest that manual query expansion, combined with summarization controlled by the number of passages parameter, can lead to some improvement in ranking and retrieval effectiveness.

A slight improvement in performance was observed with the configuration `dcu_manual_qe_summ_ptkb_TopP_3`, which achieved an `ndcg@5` of 0.2397, an `ndcg` of 0.2066, and a `MAP` of 0.0867. However, `Recall@20` remained low at 0.0183. The modest increase in `ndcg` and `MAP` indicates that incorporating a PTKB enhances ranking ability, although recall remains constrained.

In contrast, the configuration `dcu_auto_qe_key_topP-50_topK-5` produced the poorest results across all metrics, with an `ndcg@5` of 0.0878, an `ndcg` of 0.0830, `Recall@20` of 0.0267, and `MAP` of 0.0305. These results highlight the limitations of automatic query expansion based on the number of passages (P) and the number of key terms selected (K), which led to a significant decline in performance. The low `ndcg` and `MAP` values suggest the approach struggles to rank relevant documents effectively, while the weak recall emphasizes its inability to retrieve a broad set of relevant results. This indicates that, without further refinements, automatic query expansion with these settings is less effective than manual methods, especially for complex retrieval tasks. The configuration `dcu_auto_qre_sim`, which utilizes similarity-based query expansion, showed moderate improvement over the previous automatic method. With an `ndcg@5` of 0.1632, `ndcg` of 0.1559, `Recall@20` of 0.0491, and `MAP` of 0.0662, this approach improved ranking quality. However, the results still fell short of those from the manual runs. The gains in `ndcg` and `MAP` suggest that similarity-based expansion can enhance the relevance of top-ranked results, though it remains limited in terms of recall and overall retrieval effectiveness.

The configuration `dcu_auto_qe_summ_TopP_3`, which combines automatic query expansion with summarization, performed worse across all runs. It achieved an `ndcg@5` of 0.0443, `ndcg` of 0.0376, `Recall@20` of 0.0111, and `MAP` of 0.0107. These results indicate that automatic query expansion, when paired with summarization, does not improve performance and may even hinder retrieval quality. The low `ndcg` and `MAP` values reflect poor ranking and precision, while the weak recall further suggests that the system is not retrieving enough relevant documents, particularly at higher ranks.

Finally, the configuration `dcu_auto_qe_summ_ptkb_TopP_3` which uses automatic query expansion with PTKB-based summarization, showed even worse results than its manual counterpart, with an `ndcg@5` of 0.0294, an `ndcg` of 0.0227, `Recall@20` of 0.0083, and `MAP` of 0.0052. This configuration performed the poorest across all metrics. The extremely low `ndcg` and `MAP` values indicate poor ranking and precision, and the very low recall further suggests that the system fails to retrieve a sufficient number of relevant documents, even when using PTKB-based summarization. This result highlights the challenges of automatic summarization and query expansion when combined, particularly when not fine-tuned for the task at hand.

Table 2, presents the results for various retrieval system configurations in the PTKB provenance ranking task. The run `dcu_auto_qre_sim` consistently outperforms the other configurations across all metrics. It achieves the highest `ndcg@5` (0.2871) and `ndcg` (0.2755), indicating superior ranking relevance, especially for the top-ranked results. Additionally, `dcu_auto_qre_sim` leads in `Recall@20` (0.2683), suggesting that it retrieves a higher proportion of relevant items within the first 20 results. In comparison, the other runs `dcu_auto_qe_key_topP-50_topK-5`, `dcu_auto_qe_summ_TopP_3`, and `dcu_auto_qe_summ_ptkb_TopP_3`—show relatively similar performance, with `dcu_auto_qe_key_topP-50_topK-5` performing slightly better than

Table 1: Performance of different configurations run on the Passage Ranking task. Bold and *Italic* indicate the best result and the second best result

Submitted Runs	iKAT2024			
	ndcg@5	ndcg	Recall@20	MAP
dcu_manual_qe_summ_TopP_3	*0.2174*	*0.1966*	**0.0732**	*0.0783*
dcu_manual_qe_summ_ptkb_TopP_3	**0.2397**	**0.2066**	0.0183	**0.0867**
dcu_auto_qe_key_topP-50_topK-5	0.0878	0.0830	0.0267	0.0305
dcu_auto_qre_sim	0.1632	0.1559	*0.0491*	0.0662
dcu_auto_qe_summ_TopP_3	0.0443	0.0376	0.0111	0.0107
dcu_auto_qe_summ_ptkb_TopP_3	0.0294	0.0227	0.0083	0.0052

Table 2: Performance of different configurations runs on PTKB Provenance task. Bold and *Italic* indicate the best result and the second best result

Submitted Runs	iKAT2024			
	ndcg@5	ndcg	Recall@20	MAP
dcu_auto_qe_key_topP-50_topK-5	*0.2754*	*0.2622*	0.2416	0.1915
dcu_auto_qre_sim	**0.2871**	**0.2755**	**0.2683**	**0.2044**
dcu_auto_qe_summ_TopP_3	0.2697	0.2582	*0.2494*	0.1939
dcu_auto_qe_summ_ptkb_TopP_3	0.2615	0.2508	0.2418	*0.1966*

the others across all metrics. Despite the relatively close performance among the remaining runs, `dcu_auto_qre_sim` stands out as the most effective configuration overall.

Manual runs utilizing PTKB and integrating knowledge through summarization, consistently outperformed other configurations, particularly in terms of `ndcg` and `MAP`. These methods demonstrated effectiveness in improving ranking quality and relevance in top results, although recall remained a challenge. In contrast, automatic configuration using key terms-based knowledge extraction produced the poorest results, highlighting the limitations of these settings for complex retrieval tasks. Query Enrichment with highly similar historical queries yielded modest improvements over the weaker automatic configurations, suggesting that similarity-based methods could offer a viable alternative for enhancing retrieval performance.

7 Conclusions and Further Work

The experiment highlights the significant impact of query reformulation using various knowledge integration techniques on retrieval performance. Manual methods, particularly when enhanced using PTKB followed by summarization-based knowledge integration, yielded the best results, suggesting that manual interventions remain valuable in complex retrieval tasks. Automatic configuration, while potentially scalable, requires further refinement to achieve comparable performance, especially in terms of recall and ranking quality. Future work should focus on exploring more robust ways of integrating additional knowledge during the search process so that the CS systems can better understand the user information need and subsequently take informed action to enhance both retrieval and relevance.

Acknowledgments

This work was conducted with the financial support of the Science Foundation Ireland Centre for Research Training in Artificial Intelligence (CRT-AI) under Grant No. 18/CRT/6223, the SFI ADAPT Centre at DCU (Grant No. 13/RC/2106_P2) (www.adaptcentre.ie), and the National Institute of Informatics, Japan, Internship programme.

References

[1] Praveen Acharya. 2023. Towards effective modeling and exploitation of search and user context in conversational information retrieval. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 5161–5164.

[2] Praveen Acharya, Noriko Kando, and Gareth J. F. Jones. 2024. A Framework for Knowledge Integration in Conversational Information Retrieval. In *Joint Proceedings of the 1st Workshop on Evaluation Methodologies,Testbeds and Community for Information Access Research (EMTCIR 2024) and the 1st Workshop on User Modelling in Conversational Information Retrieval (UM-CIR 2024) co-located with the 2nd International ACM SIGIR Conference on Information Retrieval in the Asia Pacific (SIGIR-AP 2024), Tokyo, Japan, December 12, 2024 (CEUR Workshop Proceedings, Vol. 3854)*. CEUR-WS.org. https://ceur-ws.org/Vol-3854/um-cir-3.pdf

[3] Mohammad Aliannejadi, Zahra Abbasiantaeb, Shubham Chatterjee, Jeffrey Dalton, and Leif Azzopardi. 2024. TREC iKAT 2023: A Test Collection for Evaluating Conversational and Interactive Knowledge Assistants. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 819–829.

[4] Nicholas J Belkin. 1980. Anomalous states of knowledge as a basis for information retrieval. *Canadian journal of information science* 5, 1 (1980), 133–143.

[5] Arthur Câmara, Dima El-Zein, and Célia da Costa-Pereira. 2022. RULK: A Framework for Representing User Knowledge in Search-as-Learning. (2022).

[6] Arthur Câmara, David Maxwell, and Claudia Hauff. 2022. Searching, learning, and subtopic ordering: A simulation-based analysis. In *European Conference on Information Retrieval*. Springer, 142–156.

[7] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. YAKE! Keyword extraction from single documents using multiple local features. *Information Sciences* 509 (2020), 257–289.

[8] Dima El Zein, Arthur Câmara, Célia Da Costa Pereira, and Andrea Tettamanzi. 2023. RULKNE: Representing User Knowledge State in Search-as-Learning with Named Entities. In *Proceedings of the 2023 Conference on Human Information Interaction and Retrieval*. 388–393.

[9] Ahmed Elgohary, Denis Peskov, and Jordan Boyd-Graber. 2019. Can You Unpack That? Learning to Rewrite Questions-in-Context. In *Empirical Methods in Natural*

Language Processing.

[10] Anett Hoppe, Peter Holtz, Yvonne Kammerer, Ran Yu, Stefan Dietze, and Ralph Ewerth. 2018. Current challenges for studying search as learning processes. In *7th Workshop on Learning & Education with Web Data (LILE2018), in conjunction with ACM Web Science*.

[11] K Sparck Jones, Steve Walker, and Stephen E. Robertson. 2000. A probabilistic model of information retrieval: development and comparative experiments: Part 2. *Information processing & management* 36, 6 (2000), 809–840.

[12] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2356–2362.

[13] 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.

[14] Hadi Nasser, Dima El Zein, Célia da Costa Pereira, Cathy Escazut, and Andrea Tettamanzi. 2024. RULKKG: Estimating User's Knowledge Gain in Search-as-Learning Using Knowledge Graphs. In *Proceedings of the 2024 Conference on Human Information Interaction and Retrieval*. 364–369.

[15] Rohail Syed and Kevyn Collins-Thompson. 2017. Optimizing search results for human learning goals. *Information Retrieval Journal* 20 (2017), 506–523.

[16] Rohail Syed and Kevyn Collins-Thompson. 2017. Retrieval algorithms optimized for human learning. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*. 555–564.

[17] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. arXiv:1912.08777 [cs.CL]