

Towards Improved Conversation Context Understanding for Effective Conversational Information Retrieval

Praveen Acharya

M.Eng., B. Tech



Supervisor: Prof. Gareth J. F. Jones

A thesis presented for the degree of Doctor of Philosophy

SCHOOL OF COMPUTING
DUBLIN CITY UNIVERSITY

August 2026

Declaration

I hereby certify that this material, which I now submit for assessment on the program of study leading to the award of Doctor of Philosophy is entirely my own work, and that I have exercised reasonable care to ensure that the work is original and have conformed to the regulations on the use and declaration of Generative AI, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: PRAVEEN ACHARYA

(Candidate) ID No.: 21267927

Date: 10 August 2026

Generative AI

Generative AI (genAI) is used in this study for: (i) paraphrasing, spell checks and to verify grammar; and (ii) part of research methodology and experiments (See Chapters 4 and 5).

Tools (T): For textual support, we used the DCU-whitelisted tool Google Gemini 2.5-Flash. For methodological purposes, we experimented with multiple open-source LLMs, including Phi-3-mini-4k-instruct, Phi-3-medium-4k-instruct, Llama-3.1-8B-Instruct and Llama-3.2-1B-Instruct (details and version references are provided within the relevant chapters).

Outputs (O): The outputs generated for proofreading consisted of paraphrased text, grammar suggestions, and spelling corrections. These were manually reviewed, edited, and partially discarded to preserve authorial intent. In the research methodology, genAI tools generated different outputs depending on the tasks (details are provided within the relevant chapters).

Prompts (P): Proofreading prompts were short task-specific queries such as “*Improve grammar and clarity preserving meaning and with minimal changes*”. Methodological prompts followed standardised task definitions and experimental templates (details appear within the relevant chapters).

Iterations (I): Textual outputs were typically refined through 1–2 iterative edits using human judgment. Experimental investigation involved different prompts and test configurations (details appear within the relevant chapters).

Critical Reflection (C): All genAI outputs were critically reviewed for content accuracy, potential hallucination, bias, and suitability. The tools did not contribute conceptual content or interpretations. Please note that analytical reasoning, synthesis, and argumentation remain the author's own. The use of genAI is consistent with DCU's academic integrity guidelines.

Dedicated to my beloved parents, with love and gratitude.

मेरा प्रिय आमाबुबाप्रति, प्रेम र कृतज्ञतासहित समर्पित।

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Prof. Gareth J.F. Jones, for the invaluable guidance, encouragement, and unwavering support provided throughout my PhD journey. Prof. Jones's expertise, insightful feedback, and mentorship have been instrumental in shaping this research and my development as a researcher.

My sincere appreciation goes to my colleagues and friends in the CRT-AI (Centre for Research Training in Artificial Intelligence) and the ADAPT Centre at Dublin City University. I am grateful for the stimulating discussions, collaboration opportunities, and supportive research environment that made this journey both productive and enjoyable.

I would also like to acknowledge the academic and administrative staff of the School of Computing at Dublin City University, especially Vicky Flanagan and Angela Lally, for their assistance and support throughout my studies. Their dedication behind the scenes has contributed greatly to a positive research experience.

I am grateful to my collaborators, whose expertise, discussions, and joint efforts contributed significantly to various aspects of this research.

I would like to acknowledge Research Ireland for funding this research. I am also grateful for the financial support received through CRT-AI (Centre for Research Training in Artificial Intelligence), which enabled this work and provided opportunities for collaboration, dissemination, and professional development.

I would like to express my sincere thanks to the National Institute of Informatics (NII), Japan, for hosting me during my six-month research visit. I am particularly grateful to Prof. Noriko Kando and the members of the research group for their hospitality, mentorship, and engaging discussions that greatly enriched my research experience.

To my friends, thank you for your companionship, encouragement, and for making this journey more enjoyable. Your support, whether academic or personal, has meant a great deal. To everyone who has contributed to this work, directly or indirectly, but whose names may not be mentioned here, I offer my sincere thanks and appreciation.

Finally, I would like to express my deepest gratitude to my family. To my parents, my sister, my brother-in-law, and my nieces, Prisha and Meesha, thank you for your unconditional love, unwavering encouragement, and steadfast belief in me. To the rest of my family, thank you for your patience, understanding, and constant support throughout this journey. This achievement would not have been possible without your love and encouragement, and I am deeply grateful to each of you.

List of Publications

Praveen Acharya, Gareth JF Jones, Xiao Fu, Fabio Crestani, and Noriko Kando. “Report on the 1st Workshop on User Modelling in Conversational Information Retrieval (UM-CIR 2024) at SIGIR-AP 2024 and Workshop on Emerging Trends in Conversational Information Retrieval (ET-CIR 2024) at National Institute of Informatics (NII).” In ACM SIGIR Forum, vol. 59, no. 1, pp. 1-7. New York, NY, USA: ACM, 2025.

Praveen Acharya, Xiao Fu, Noriko Kando, and Gareth JF Jones. “DCU-ADAPT@TREC iKAT 2024: Incorporating Retrieved Knowledge for Enhanced Conversational Search.” In Proceedings of Text REtrieval Conference (TREC 2024), NIST Special Publication. National Institute of Standards and Technology (NIST).

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>

Praveen Acharya, Gareth J. F. Jones, Xiao Fu, Aldo Lipani, Fabio Crestani, and Noriko Kando. “The 1st Workshop on User Modelling in Conversational Information Retrieval (UM-CIR).” In Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region, pp. 315-317. 2024.

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 (CIKM '23). Association for Computing Machinery, New York, NY, USA, 5161–5164. <https://doi.org/10.1145/3583780.3616005> [*Doctoral Consortium*]

Xiao Fu, Navdeep Singh Bedi, **Praveen Acharya**, and Noriko Kando. “NII@ TREC IKAT 2024: LLM-Based Pipelines for Personalized Conversational Information Seeking.” In Proceedings of Text REtrieval Conference (TREC 2024), NIST Special Publication. National Institute of Standards and Technology (NIST).

Table of Contents

List of Publications	vi
List of Figures	xi
List of Tables	xiii
List of Abbreviations	xiv
Abstract	xv
1 Introduction	1
1.1 Focus of this thesis	8
1.1.1 Key Concepts	12
1.2 Hypotheses and Research Questions	13
1.3 Contributions of this Thesis	14
1.4 Structure of the Thesis	15
2 Information Retrieval and Conversational Information Retrieval	18
2.1 An Overview of Information Retrieval	19
2.1.1 Document Representation	20
2.1.2 Query Representation	21
2.1.3 Retrieval Models	21
2.1.4 Neural Modeling Foundations	28
2.1.5 Re-ranking	36
2.2 Overview of Conversational Information Retrieval	37
2.2.1 Conversational Query Understanding	39
2.3 User Knowledge in Information Retrieval	44
2.4 Summary	49
3 Experimental Methodology and Evaluation Framework	51
3.1 Information Retrieval Evaluation	52
3.1.1 Standard Test Collections	52
3.1.2 Evaluation Metrics	56
3.1.3 Significance Tests	59
3.2 Evaluation Test Collections	61
3.2.1 TREC CAsT Test Collections	62
3.2.2 TREC iKAT Test Collection	64
3.3 Framework for Experimental Investigation	68
3.3.1 Task Formulation	68

3.3.2	Experimental Setup	69
3.4	Summary	71
4	Conversational Context Understanding in CIR	73
4.1	Introduction	74
4.2	Investigating the Effect of Historical Conversational Context in CIR .	74
4.2.1	Variations of Historical Conversational Query Context	75
4.2.2	Results	77
4.2.3	Discussion	87
4.3	Similarity-based Related Historical Context Identification	88
4.3.1	Selection Model	89
4.3.2	SBQC Based Historical Conversational Query Formulation . .	90
4.3.3	Hybrid Query Formulation	93
4.3.4	Results	93
4.4	LLMs-based Related Historical Context Identification	101
4.4.1	Results & Discussion	102
4.4.2	Retrieval Performance Based on LLM Identification	104
4.5	Summary	105
5	User Knowledge in Conversational Information Retrieval	115
5.1	Introduction	116
5.2	A Framework for Knowledge Integration in CIR	119
5.2.1	Components	120
5.2.2	Discussion	123
5.3	Keyword-Based Knowledge Integration	124
5.3.1	Various Formulations of Query Enrichment	126
5.3.2	Results and Discussion	128
5.4	Summarization-Based Knowledge Integration	136
5.4.1	Various Formulations of Query Enrichment	136
5.4.2	Results and Discussion	138
5.5	Summary	146
6	Conclusion	149
6.1	Key Findings	149
6.1.1	Exploring the Challenge of Related Historical Context Identification in CIR	150
6.1.2	User Knowledge in CIR	150
6.2	Research Questions Addressed	151
6.2.1	To what extent can the incorporation of historical conversational context impact the overall retrieval effectiveness of a conversational search system?	151
6.2.2	Can we identify and select the historical conversational context in the dialogue so as to be able to support the user in reaching their information goal?	152
6.2.3	Can features of the expected user’s developing knowledge be identified and captured during a conversational search task? .	153
6.2.4	Can we exploit the model of the user’s knowledge to improve the effectiveness of conversational search?	154
6.3	Limitations and Future Directions	155

6.3.1	Limitations	155
6.3.2	Future Directions	158
A	LLM Prompt details	160
A.1	Prompts for the Phi-LLM Model	160
A.2	Prompts for the LLama Models	161

List of Figures

1.1	Classical search model.	3
1.2	A Berrypicking Model of Search (Evolving Search) (Bates 1989). . .	5
1.3	Conversational Information Retrieval.	7
1.4	Query Specificity and User Knowledge range.	10
2.1	Traditional Information Retrieval Process.	20
2.2	Different types of Neural Models. (Modified from (Guo, Fan, Pang, et al. 2020) for consistency in the naming convention)	27
2.3	Transformer Architecture (Vaswani et al. 2017).	29
2.4	Attention Mechanism (Vaswani et al. 2017)	31
2.5	Overall pre-training procedures for BERT. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token. (Devlin et al. 2018)	33
2.6	The general architecture of ColBERT given a query and a docu- ment (Khattab and Zaharia 2020a).)	35
2.7	A general architecture of CIR (Gao et al. 2023).	38
3.1	TREC Document Collection	54
3.2	TREC Query Collection	55
3.3	TREC Relevance Judgement	55
3.4	TREC CAsT 2019 Topic 31	63
3.5	TREC CAsT 2020 Topic 81	64
3.6	iKAT Flow (Aliannejadi, Abbasiantaeb, Chatterjee, Jeffery Dalton, et al. 2024).	66
3.7	TREC iKAT 2023 (User A)	67
3.8	TREC iKAT 2023 (User B)	68
3.9	Overview of the Proposed Framework	69
4.1	Relevant Query Turn Dependencies in the Dialogue History.	79
4.2	Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval. . .	80
4.3	Topic-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval. . .	81
4.4	Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval. . .	81
4.5	Topic-wise ndcg@10 Performance for CAsT 2020 Dense Retrieval. . .	82
4.6	Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval. . .	83
4.7	Topic-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval. . .	83
4.8	Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval. .	84
4.9	Depth-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval. . .	85
4.10	Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval. .	85
4.11	Depth-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval. . .	86

4.12	Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval. . .	86
4.13	Depth-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval. . .	87
4.14	Illustration of the <i>Selection model</i> for related historical query turn identification for Query Formulation.	107
4.15	Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval . .	107
4.16	Topic-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval . .	108
4.17	Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval . .	108
4.18	Topic-wise ndcg@10 Performance for CAsT 2020 Dense Retrieval . .	109
4.19	Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval . .	109
4.20	Topic-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval . .	110
4.21	Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval . .	110
4.22	Depth-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval . .	111
4.23	Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval . .	111
4.24	Depth-wise ndcg@10 Performance for CAsT 2020 Dense Retrieval . .	112
4.25	Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval . .	112
4.26	Depth-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval . .	113
4.27	Average Query Length vs. Conversation Length (Depth) — CAsT 2019	113
4.28	Average Query Length vs. Conversation Length (Depth) — CAsT 2020	114
4.29	Average Query Length vs. Conversation Length (Depth) — iKAT 2023	114
5.1	User’s Knowledge exploitation in Conversational Information Retrieval (CIR) (Acharya, Kando, and G. J. Jones 2024).	118
5.2	Proposed Framework Architecture for Incorporating Expected User Knowledge in CIR (Acharya, Kando, and G. J. Jones 2024).	121
5.3	Illustration of Knowledge Integration for Query Enrichment.	126
5.4	Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval. . .	132
5.5	Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval. . .	133
5.6	Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval. . .	133
5.7	Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval. .	134
5.8	Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval. .	135
5.9	Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval. . .	135
5.10	Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval. . .	143
5.11	Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval. . .	143
5.12	Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval. . .	144
5.13	Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval. .	145
5.14	Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval. .	145
5.15	Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval. . .	146
A.1	Zero-shot Prompt for Phi-LLM	160
A.2	One-shot Prompt for Phi-LLM	160
A.3	Few-shot Prompt for Phi-LLM	161
A.4	Zero-shot Prompt for LLama Model	162
A.5	One-shot Prompt for LLama Model	162
A.6	Few-shot Prompt for LLama Model	163

List of Tables

2.1	Kuhlthau’s Information Search Process (ISP) Stages. (Kuhlthau 1991)	46
4.1	Performance of Sparse and Dense retrieval systems. Bold and <u>underline</u> indicate the best result and the second best result (other than oracle Q_{Gold} and Q_{manual} approach)	77
4.2	Performance of Sparse and Dense retrieval systems. Bold and <u>underline</u> indicate the best result and the second best result (other than Q_{manual} approach.)	78
4.3	Performance of Sparse and Dense retrieval systems. Bold and <u>underline</u> indicate the best result and the second best result (other than oracle Q_{Gold} and Q_{manual} approach). α and β denote no statistically significant and statistically significant improvements over ($Q_{all} + cqr$), respectively. (Paired t-test at $p < 0.05$)	94
4.4	Performance of Sparse and Dense retrieval systems. Bold and <u>underline</u> indicate the best result and the second best result (other than Q_{manual} approach). α and β denote no statistically significant and statistically significant improvements over ($Q_{all} + cqr$), respectively. (Paired t-test at $p < 0.05$)	95
4.5	Identification of Related Historical Query Context using Different Approaches	99
4.6	Identification of Related Historical Query Context using Different Approaches	103
4.7	Performance of Sparse and Dense retrieval systems using different approaches to identify historical conversational query context	105
5.1	Performance of a Sparse Retrieval (BM25) System TREC CAsT 2019, 2020. Bold and <u>underline</u> indicate the best result and the second best result. α and β denote no statistically significant difference over the baselines Q_{raw} and $Q_{raw} + RM3$, respectively. (Paired t-test at $p < 0.05$)	129
5.2	Performance of a Sparse Retrieval (BM25) System TREC iKAT 2023. Bold and <u>underline</u> indicate the best result and the second best result. No statistically significant difference over the baselines Q_{raw} and $Q_{raw} + RM3$. (Paired t-test at $p < 0.05$)	130
5.3	Performance of a Sparse Retrieval (BM25) System TREC CAsT 2019, 2020. Bold and <u>underline</u> indicate the best result and the second best result. α and β denote statistically significant improvements over the baselines Q_{raw} and $Q_{raw} + RM3$, respectively. (Paired t-test at $p < 0.05$)	139

5.4	Performance of a Sparse Retrieval (BM25) System TREC iKAT 2023. Bold and <u>underline</u> indicate the best result and the second best result. No statistically significant difference over the baselines Q_{raw} and Q_{raw} + RM3. (Paired t-test at $p < 0.05$)	140
-----	---	-----

List of Abbreviations

- ASK** Anamolous State of Knowledge.
- BERT** Bidirectional Encoder Representations from Transformers.
- CAsT** Conversational Assistant Track.
- CIR** Conversational Information Retrieval.
- CNN** Convolutional Neural Network.
- ColBERT** Contextualized Late Interaction over BERT.
- FFN** Feed-forward Network.
- iKAT** Interactive Knowledge Assistance Track.
- IR** Information Retrieval.
- ISP** Information Search Process.
- LLM** Large Language Model.
- LM** Language Modeling.
- LSTM** Long Short-term Memory.
- MLM** Masked Language Model.
- NLP** Natural Language Processing.
- NM** Neural Modeling.
- NSP** Next Sentence Prediction.
- PRP** Probability Ranking Principle.
- RNN** Recurrent Neural Network.
- SAL** Searching-as-Learning.
- SBQC** Similarity-Based Historical Query.
- TREC** Text Retrieval Conference.
- VSM** Vector Space Model.

Towards Improved Conversation Context Understanding for Effective Conversational Information Retrieval

Praveen Acharya

Abstract

Conversational Information Retrieval (CIR) has garnered considerable attention in recent years. A key challenge in CIR is formulating effective queries during the dialogue between the searcher and the conversational agent. Unlike traditional information retrieval systems, where users independently formulate a complete query for each search operation, conversational systems offer the opportunity for agents to assist users in an interactive, multi-turn search process. This alleviates the burden on users to express their full information need in a single query by leveraging the iterative nature of conversation to gradually refine and clarify their search.

Conversational context plays a vital role in this query formulation process. In natural language conversations, each turn may implicitly or explicitly refer to previous dialogue turns. This dependency of the turns in the dialogue on conversational history introduces the challenge of identifying and understanding the relevant parts of the dialogue context to correctly interpret the current query.

This PhD project focuses on understanding and leveraging conversational context from two dimensions: conversational history and user knowledge. Conversational history deals with identifying and exploiting the relevant parts from within the conversational historical context to disambiguate and interpret the query, while Knowledge deals with exploiting user knowledge of the subject as well as that accumulated during the conversation

The goal of this thesis is to study conversational context in CIR, with a focus on understanding and modeling these two dimensions. The contributions of this work are twofold: (1) methods for identifying and modeling relevant portions of the conversational history, and (2) approaches for modeling the evolving knowledge of users throughout the dialogue. These contributions aim to improve the effectiveness and responsiveness of conversational information retrieval systems.

Chapter 1

Introduction

Searching for any form of information comes from the inherent need to know more about a particular subject. This need arises from the simple fact that, in relation to the amount of knowledge available in the universe, it is impossible for a human to know everything required to complete their tasks and activities. Furthermore, the entire universe of knowledge is not generally of interest to them or, more specifically, relevant to addressing their current need for information. Therefore, there is a need for efficient means to find relevant information within the body of available knowledge.

Humans have long sought to understand the searching process and have designed and developed various ways, methods, and techniques over the years to access relevant information from a vast number of knowledge sources (Buckland and Z. Liu 1995). The problem of retrieving information from these diverse sources representing the universe of knowledge has been a central focus in Library and Information Sciences and has been extensively studied (Buckland and Z. Liu 1995; Tuomaala, Järvelin, and Vakkari 2014). This field has concentrated on representing knowledge, storing and cataloguing information, understanding the search process, and developing retrieval techniques—all aimed at efficiently and effectively accessing the most relevant information from among the available knowledge sources (D. Harman et al. 2019).

As described, the focus of searching is retrieving information, which is more

formally referred to as the *information retrieval process*. This process begins with what is referred to as a user’s “*information need*”, which is the central concept in Information Retrieval (IR). The goal of an IR process is to satisfy the user’s information need. In simpler terms, an information need refers to a gap or insufficiency in one’s knowledge required to achieve one’s current objective. The objective of IR is for the user to acquire additional information to address the information need in order to complete the current task.

Over the years, a number of approaches to IR have been proposed to achieve this goal of satisfying a user’s information need. Most work on IR to date has focused on what can be referred to as the “classical model”. A high-level overview of the classical model is shown in Figure 1.1. The classical model is a simplified representation of human search behaviour with significant constraints and assumptions. In this model, on one side, we have a user with an information need (I), and on the other side, an IR system that has access to a collection of knowledge sources (the universe of knowledge). The objective of the system is to locate relevant information within the knowledge collection that pertains to (I). To accomplish this, the user must express I in a way that both captures the essence of the need and is interpretable by the system. This description or formulation of their information need is generally referred to as a *Query (Q)*, and is used by the IR system to identify potentially relevant information. Classically, information is taken to be contained in a set of documents.

The classical IR model has been both fundamental and effective in advancing researchers’ understanding of the search process. Its use over the years has contributed significantly to our comprehension of how searching works. However, as our understanding of how people approach search has deepened, the classical model has shown limitations that make it increasingly inadequate. The classical model assumes that users formulate a query that accurately and sufficiently represents their information need to retrieve content deemed relevant and sufficiently satisfy their information need.

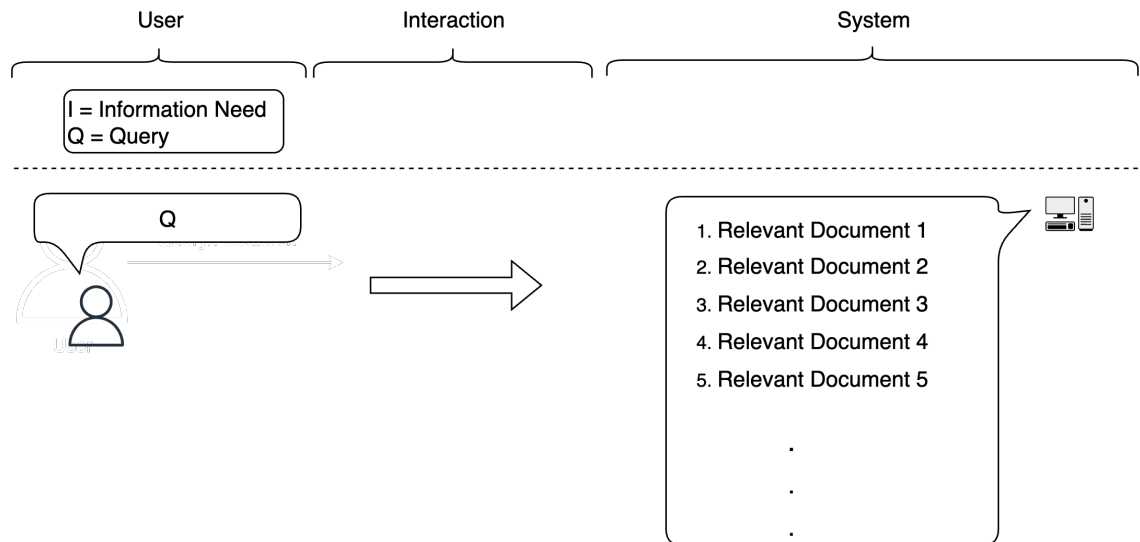


Figure 1.1: Classical search model.

The assumption that the user can accurately describe their information need is problematic. After all, when someone is searching for information, it is usually because they lack sufficient knowledge about the subject. So how can the user be expected to describe something they do not fully understand? This phenomenon, referred to as the *non-specifiability of need problem* (Nicholas J Belkin 1980) is a challenge for IR systems.

In practice, the search process often begins with a broad description of the information need, without the level of granular detail expected by the classical model and needed to satisfy the information need. As the user interacts with the contents retrieved by the first query, they evaluate the retrieved information and gain additional knowledge relevant to their information need. It is rarely the case that the information need is satisfied in a single pass. Instead, the knowledge gained during this interaction leads the user to a better understanding of their information need, which leads to the entry of a revised query, and so on until the information need is satisfied. Therefore, real-world searchers are, in essence, incremental in nature, where an information need is satisfied through an incremental searching of a series of iterative cycles of interaction between the user and the system.

This evolving nature of the searching process is the basis of the “*berypicking*”

model of IR introduced by (Bates 1989). The “*berrypicking*” model argues that queries extend during the search process and are not static but evolve over time, which is a better representation of real-world search behaviour, in contrast to the classical model. Bates (1989) emphasizes that the searcher typically employs a wide variety of search techniques iteratively to gather information and satisfy their information need, a process the classical model does not consider (Bates 1989). A high-level overview of the “*berrypicking*” model is shown in Figure 1.2. In this model, the outermost level is the *Universe of Knowledge*, encompassing all available knowledge in the world. Within this, the user’s search focuses on a subset called the “*Universe of Interest*”, where the search is conducted and where the user’s information need may be satisfied. The search begins and progresses along a directed path reflecting the evolving search process, involving a series of query variations, accessed documents or information, and information gathered through various search techniques until the information need is fulfilled, leading to the conclusion of the search. The classical model, within the context of the *berrypicking* model, can be seen as a specialized case representing the shortest directed path from the start to the end of the search process, lacking the evolving aspect.

Instead of treating search as a single query followed by a single result in the *classical model*, the *berrypicking* model now views search as an ongoing process in which users gradually build their knowledge. This idea is closely related to Search-as-Learning (Vakkari 2016), which suggests that searching is not only about finding information but also about helping users develop a better understanding of a topic. From this perspective, each query reflects the user’s current level of knowledge, and the system’s response helps shape what the user learns next. This idea motivates the two main goals of this thesis: identifying the previous parts of the conversation that are most relevant to the current query, and modelling the knowledge the user has gained from earlier interactions.

While the *berrypicking* model more closely reflects how human search occurs in the real world and addresses some limitations of the classical model, it remains

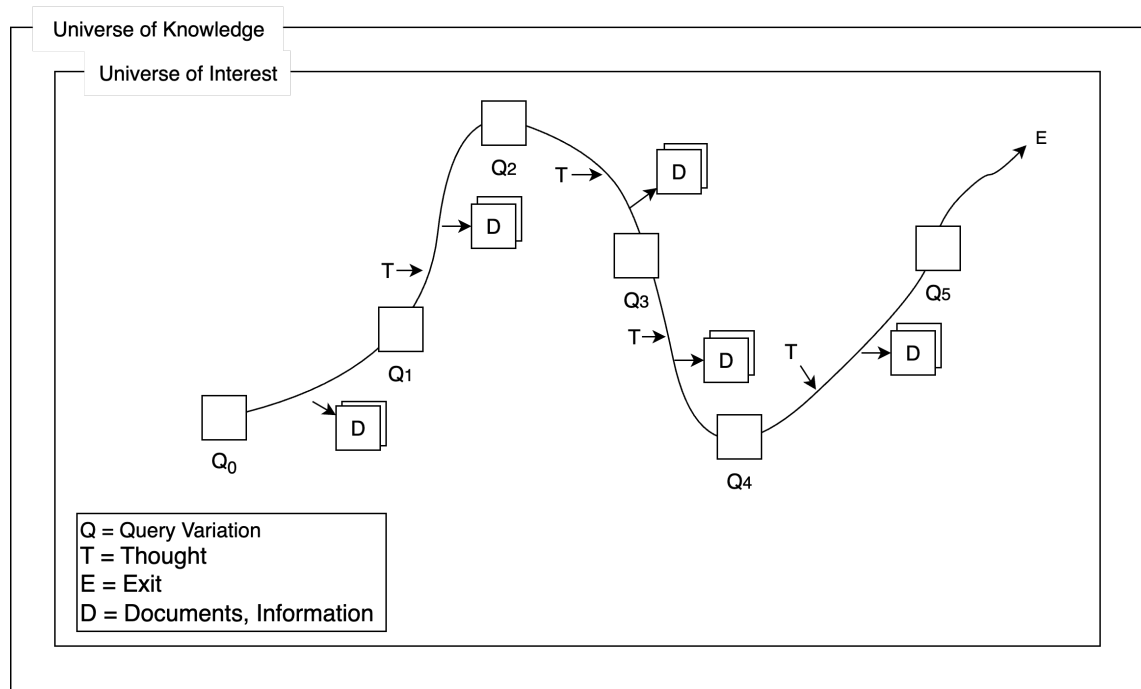


Figure 1.2: A Berry-picking Model of Search (Evolving Search) (Bates 1989).

limited in terms of the interaction between the user and the system. It employs a unidirectional interaction approach, where the flow is from the user to the system, similar to the classical model shown in Figure 1.1.

In both models, the user drives the interaction, and the system is responsible for finding relevant information from the knowledge source. This unidirectional interaction places a high cognitive load on the user to explore the information retrieved by the IR system. Neither the classical nor the *berry-picking* model adequately considers this cognitive burden on the user in satisfying their information need. Although the *berry-picking* model inherently acknowledges the evolving nature of the search process, it still emphasizes the user’s role in managing this cognitive load by applying various search techniques to satisfy their need (Bates 1989). The user might find it difficult to sift through all the retrieved information to identify what truly satisfies their information need. This challenge applies to both the *classical* model and *berry-picking* models. Therefore, an IR system that can assist users in sifting through this information efficiently should be highly beneficial.

Real-world searches are not only evolving in nature but are also collaborative (Morris 2008; Morris 2013; Evans and Chi 2008). The need to search arises from a user’s

recognition of a knowledge deficit, and the pursuit to address this information need may involve seeking assistance from someone believed to have the relevant expertise or the ability to help find the information *e.g* a librarian or a subject expert. The search process begins with an initial query provided by the user to the assistant. Based on this query, the assistant attempts to find relevant information. However, it is rare that they succeed at the first attempt. Instead, a series of interactions occurs between the user and the assistant, during which the assistant seeks to better understand the user’s query and, consequently, the underlying information need, by eliciting additional information, requesting clarifications, asking for feedback, offering suggestions, and providing referrals. This series of interactions reflects both the evolving and collaborative nature of the search process, an aspect not captured by either the *classical* model or *berrypicking* model. This dynamic effectively offloads some of the user’s cognitive load of the search to the assistant and helps them to satisfy their information need through a more natural dialogue setting.

While the *berrypicking* model captures the evolving nature of the search process, it fails to address the collaborative aspect that can help reduce the cognitive load on the user, also overlooked by the classical model. By incorporating interactive elements between the user and the system, a more natural search process can be achieved. This process can be understood as a dialogue between the user and the system that unfolds over a series of turns, guiding the user toward satisfying their information need, as illustrated in Figure 1.3. This approach, known as Conversational Information Retrieval (CIR), improves upon previous IR models by mimicking the more natural, interactive nature of human information seeking (Hannabuss 1989; Wilson 1999; Zamani et al. 2023). CIR builds upon the *classical* and *berrypicking* IR models and extends these models by introducing bi-directional interaction and a collaborative search process.

Figures 1.1, 1.2, and 1.3 show different information retrieval models and provide the motivation for this thesis. Figure 1.1 shows the traditional search model, where the user is responsible for refining their own search. Figure 1.2 extends this, capturing

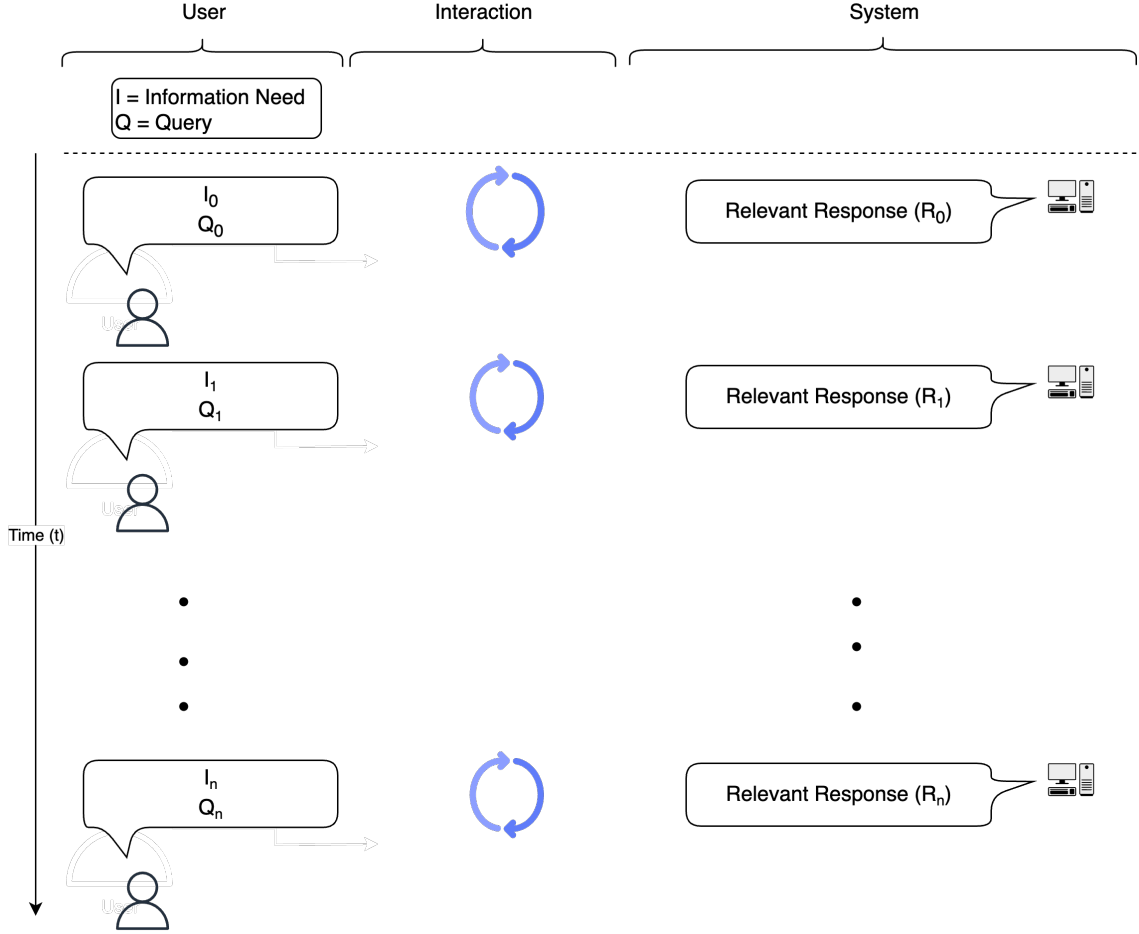


Figure 1.3: Conversational Information Retrieval.

that searching is an ongoing process in which the user’s information needs change over time, although the interaction is still mainly one-way. Figure 1.3 further extends this idea by showing CIR, where the user and system interact through a two-way conversation, allowing the system to help the user refine their information need over multiple turns. The approaches investigated in this thesis are situated within the framework shown in Figure 1.3, and aim to improve how the system understands and uses the conversational context that builds up during the conversation.

Recent years have seen much interest in CIR (Zamani et al. 2023; Mo, Mao, Z. Zhao, et al. 2025). A CIR (also called *conversational search* or *conversational information seeking*) system is an IR system with a conversational interface which allows users to interact with the system to seek information via multi-turn conversations of natural language, in spoken or written form. Traditionally, IR systems

have been completely user-driven, with the system having no way of engaging with the user in the search process beyond simple relevance feedback methods. Relevance feedback methods utilizes information from the retrieved documents that the user finds useful to improve and expand on the user queries (Rocchio Jr 1971). It was one of the first methods developed to make information retrieval more responsive to the user’s changing information needs. However, relevance feedback is still mainly focused on improving the query using information from the retrieved documents. It does not build a broader model of the user’s knowledge or take the full conversation into account. CIR goes further by treating search as a multi-turn conversation. CIR systems incorporate an interactive and collaborative process between all the actors involved. The user is supported, and there is no need to form a standalone single query. CIR systems can involve multiple search stages with iterative development and progression of the query at each turn and the need to track the evolution of the conversation. Being able to understand the conversational context within the search process allows the system to better understand the query, which in turn can lead to retrieving relevant information.

1.1 Focus of this thesis

In natural language conversation, a conversation is a series of interactions between the participants of the dialogue. A single interaction between the participants is defined as a turn, and each turn consists of a pair of utterances exchanged between participants in a dialogue. A conversation consists of a sequence of these turns. Conversational dialogue involves various types of natural language phenomena such as co-reference, omissions, ellipsis, etc. Thus, a user’s utterances are frequently incomplete or underspecified due to the fact that the information may have already been specified in the earlier turns in the conversation. This dependent nature of a turn on the earlier turns of the dialogue raises the challenge of identifying and understanding the conversational context to disambiguate and contextualize the current turn within the dialogue in order to infer the required information, which

helps to understand the intention of the user query. During CIR, users issue a series of queries describing their information need to the IR system, and the system, based on the queries, tries to find the relevant document. As the conversation progresses, the user's queries lose more and more information, because in the natural dialogue setting, it is assumed that the user does not have to specify information time and again that has already been mentioned in the earlier turns in the dialogue. However, these queries are incomplete statements of the information need at the current turn, and the IR system must refine them to include the missing information in order for them to be useful for effective retrieval. In order to infer the necessary information, accurately identifying the relevant earlier conversational turns allows the CIR system to understand the user query, which can, in turn, be utilized to retrieve relevant documents. This may be even more important when the search conversation becomes longer, with features such as evolving information needs, topic shifts, and references running deeper within the conversation. As such, identifying and selecting the appropriate and relevant historical turns from the conversational context is essential.

Furthermore, when searching for information, whether using a traditional or conversational setting, the user will approach the search task with varying degrees of prior knowledge regarding the topic of information that they are looking for (White, Dumais, and Teevan 2009). Regardless of their prior knowledge, they may encounter difficulty in describing their information need arising from the *non-specifiability of need problem* (Nicholas J Belkin 1980). Since whatever their prior knowledge, they are seeking to resolve an information need which they will have difficulty describing. Their query may thus be a poor description of their information need, and they may encounter problems retrieving relevant content. In such cases, the search process can become exploratory depending on the user's knowledge, where the search process begins with a vague query and the user, through an iterative process of interaction with the IR system, is able to refine their query and subsequently their information need. This is in contrast to searches where the information need is well defined, and the user has sufficient knowledge of the topic, enabling them to formulate a clear



Figure 1.4: Query Specificity and User Knowledge range.

and specific query, and where the information need can be satisfied with a single, well-defined answer *e.g.* a factual search or question-answer type information need.

The knowledge disparity influences how different users formulate their search queries, resulting in different degrees of search effectiveness. Queries are a representation of the user’s information need that an IR system can utilize in order to find relevant information. Users can create different queries for the same information need based on their knowledge about that need, and on what they think the system is able to understand in order to identify the relevant sources. For example, an expert (knowledgeable) user can specify their information need in more detail (well-defined query, including correct use of domain-specific vocabulary). This may make it more likely that the search system will retrieve relevant documents. In contrast, a non-expert (ill-informed) user will have difficulty specifying their information need using domain-specific words, leading to under-specified (vague) queries with a lower likelihood of retrieving documents. The range of query specificity, influenced by the user’s knowledge, is illustrated in *Figure 1.4*. It shows how the specificity of a user’s queries changes as their knowledge increases. This idea is closely related to the second hypothesis of this thesis. If a user’s knowledge affects how specific their queries are, then a CIR system that tracks how the user’s knowledge changes over time, instead of treating each query separately, can better handle vague queries early in the conversation and make better use of the more specific queries that appear as the user’s knowledge grows.

We argue that understanding context in conversational search can be addressed by exploiting conversational history and knowledge during the search dialogue.

Conversational history deals with identifying and exploiting the relevant steps

from the conversational historical context to disambiguate and interpret the query. **Knowledge** deals with exploiting user knowledge of the subject as well as that accumulated during the conversation. A user’s knowledge that is acquired during multi-turn interaction with a CIR system can influence their query construction and information need at each step, and have an impact on the effectiveness of the CIR systems. Unlike traditional IR systems, CIR supports a multi-turn, interactive process, where users and systems can collaborate on refining the information need at each turn in the dialogue. By leveraging the knowledge acquired by a user during the dialogue, CIR systems can potentially reduce cognitive load, support comprehension, and generate more relevant and contextually appropriate responses.

Understanding conversational context becomes more important as a search conversation gets longer. This is because there is more previous information that may be relevant, and the user’s knowledge can become very different from what it was when they asked the first query. From the search-as-learning perspective, users learn more about a topic as they interact with the system. As a result, their information needs change over time, and systems that cannot follow these changes may become less effective. These ideas motivate the two main hypotheses of this thesis. The first focuses on identifying which previous turns in the conversation are most relevant to the current query. The second focuses on modeling what the user has learned from those earlier turns and how this knowledge can be used to improve retrieval. Together, these hypotheses view conversational context understanding as involving two connected aspects: the conversation history and the user’s changing knowledge.

Previous works on conversational context understanding (Clarke 2019; Ohsugi et al. 2019; Mehri, Eric, and Hakkani-Tur 2020) have only focused on *conversational history*. Most approaches to the query understanding task that focus on disambiguating and contextualize the current user query with respect to the conversational history do so through query reformulation (Mo, Mao, Z. Zhao, et al. 2025). For effective query reformulation, it is necessary to be able to identify and select the related historical turns to correctly capture the current conversational context.

In order to understand the query, which is typically in a natural language form, the system needs to be able to identify relevant portions from the conversational history. Identifying the relevant historical turns allows the system to infer the required information that is necessary to disambiguate and contextualize the current turn. The effectiveness of query understanding approaches as described above is highly reliant on being able to identify the correct and relevant historical turns. Existing approaches to this turn selection are very simple, ranging from utilizing all historical turns to utilizing a fixed number of the previous-k turns (Jeffrey Dalton, Xiong, Kumar, et al. 2020; Jeffrey Dalton, Xiong, and Callan 2021). An important aspect that has not been explored in previous work is the knowledge that is accumulated during a conversational dialogue. As described earlier, user knowledge can greatly influence their search behaviour, and a system that captures and exploits the knowledge accumulated during the dialogue could help the CIR system to better understand and address the user’s information need.

This PhD project aims to contribute by developing approaches to understanding and leveraging conversational context from the dimensions of context and knowledge, with the goal of improving the performance of CIR systems.

1.1.1 Key Concepts

Conversational Context. refers to all the information available during an ongoing search conversation that helps interpret the user’s current query. This includes previous user queries, system responses, and any information gained during the conversation. It is the broadest of the three concepts defined here, as it represents the overall context in which the current query is understood.

Conversational History. refers to the sequence of previous user queries, $(q_1, \dots, q_{n-1})$, made before the current query (q_n) . It is part of the overall conversational context and focuses only on the user’s earlier queries. As discussed in Chapter 4, not every previous query is useful for understanding the current one.

Identifying which earlier queries are most relevant is a key challenge addressed in this thesis.

User Knowledge. refers to the information and understanding that a user builds up while interacting with a CIR system during a search session. Unlike conversational history, which records the user’s previous queries, user knowledge represents the user’s growing understanding of the topic. This knowledge cannot be observed directly, but it can be estimated from the user’s queries and the documents they view during the search. As the user’s knowledge changes throughout the conversation, it also influences how they ask later queries.

The relationship between these three concepts is hierarchical. *Conversational context* is the broadest concept and includes both *conversational history* and *user knowledge*. Conversational history is the record of the user’s previous queries, while user knowledge represents what the user has learned during the conversation. Figure 3.9 shows how these concepts are used in the proposed experimental framework. An effective CIR system should use both the user’s previous queries to identify relevant context and the user’s changing knowledge to better understand their current information need.

1.2 Hypotheses and Research Questions

As described earlier, conversational context understanding is an important aspect of CIR. It is essential to understand and interpret the current query in the dialogue, in order to understand the current information needs of the user. In this work, we propose to examine modeling and exploitation of conversational context to address the following hypotheses and associated research questions.

Hypothesis 1 (H1): Analysis and exploitation of the dialogue history can help identify contextually relevant dialogue context, which can be used to better under-

stand the current query, e.g. by disambiguating it, leading to a more effective search system.

- ***RQ1:*** *To what extent can the incorporation of historical conversational context impact the overall retrieval effectiveness of a conversational search system?*
- ***RQ2:*** *Can we identify and select the historical conversational context in the dialogue so as to be able to support the user in reaching their information goal?*

Hypothesis 2 (H2): The searcher’s domain knowledge, developed during a search conversation, can potentially be exploited to guide the dialogue progression and effectively support the resolution of the user’s information needs.

- ***RQ1:*** *Can features of the expected user’s developing knowledge be identified and captured during a conversational search task?*
- ***RQ2:*** *Can we exploit model of the user’s knowledge to improve the effectiveness of conversational search?*

1.3 Contributions of this Thesis

This thesis makes the following original contributions to the field of conversational information retrieval.

Empirical analysis of historical conversational context in CIR. We conduct a systematic investigation into how incorporating historical conversational query turns affects retrieval effectiveness in CIR systems. Unlike prior work that either uses all available historical turns or a fixed window of previous turns (Jeffrey Dalton, Xiong, Kumar, et al. 2020; Jeffrey Dalton, Xiong, and Callan 2021), this thesis provides a principled empirical analysis of how the quantity and selection of historical turns influence retrieval performance across multiple datasets and retrieval architectures.

A hybrid approach to related historical turn identification. We propose and evaluate a hybrid approach to identifying and selecting relevant historical conversational query turns, combining simple heuristics with semantic similarity signals. This approach exploits the structural and semantic relationships between user-issued turns. This represents a more realistic and deployable approach to conversational context understanding in CIR.

A framework for user knowledge modeling in CIR. We propose a novel framework for capturing and modeling a user’s evolving knowledge state throughout a conversational search session. This is a distinct contribution from existing CIR approaches, which focus almost exclusively on conversational history and query reformulation. The framework introduces a Knowledge Extractor (KE) and Knowledge Updater (KU) that continuously extract, update, and integrate knowledge from both user queries and retrieved documents across turns. This moves beyond relevance feedback, which uses retrieved documents to reformulate the immediate query, by maintaining a persistent, evolving model of the user’s knowledge state across the entire dialogue.

Knowledge-enriched query formulation. Building on the proposed framework, we investigate and evaluate keyword-based and summarization-based approaches to integrating user knowledge into conversational query formulation. We demonstrate that summarization-based knowledge integration, when selectively applied, can improve retrieval effectiveness under certain conditions.

1.4 Structure of the Thesis

The PhD thesis is organized as follows.

Chapter 2: In this chapter, we provide background information and discuss related work that supports our research. We start with an overview of IR, focusing on how queries and documents are represented. Then, we look at some of the retrieval

models found in existing research. After that, we explore different types of relevance feedback used in IR. We also explain key ideas like indexing, query construction, retrieving results, and re-ranking in IR systems. Next, we give an overview of CIR, followed by a look at existing work on how conversational context and user knowledge are understood in CIR.

Chapter 3: In this chapter, we present the experimental evaluation framework used throughout the thesis. We begin by looking at how evaluation is typically done in IR research. Next, we describe some standard evaluation approaches and discuss widely used test collections, such as the Cranfield and Text Retrieval Conference (TREC) collections. We then explain common evaluation measures in IR, along with methods for significance testing. After this, we introduce the CIR datasets used in our study, examining the details of each one to understand their features and characteristics. We then provide an overview of our proposed experimental framework, followed by a description of how we define our tasks. Finally, we explain the implementation details of our experimental setup, which is used to test our hypotheses and address the research questions.

Chapter 4: In this chapter, we explore how conversational context is understood in CIR. We start by examining the impact of using past conversation history. To do this, we implement several methods for including historical context in CIR and evaluate how well they perform in a full end-to-end system. Next, we test our own proposed method, which uses similarity to identify and include relevant parts of the conversation history, and we evaluate its effectiveness. We also examine the use of Large Language Model (LLM) to identify relevant historical context and assess both their accuracy and how they affect retrieval performance.

Chapter 5: In this chapter, we explore how user knowledge impacts CIR. We begin by explaining the motivation behind using user knowledge in CIR. Then, we introduce a new framework for integrating this knowledge and describe its different components.

After this, we propose a keyword-based approach for knowledge extraction and its integration for query enrichment and carry out experiments to test its effectiveness. We also examine a summarization-based approach for knowledge extraction and its integration for query enrichment and evaluate its performance in a complete end-to-end system. Finally, we combine the conversational context understanding method with the knowledge-based approach and conduct experiments to assess how well this combined method performs.

Chapter 6: In this chapter, we conclude our investigation and summarize the findings of the thesis. We revisit our hypotheses and the associated research questions and summarize how each one of them has been addressed in the context of our research work. We then describe potential avenues of extending our research work, and we also look into the limitations and future directions of the research.

Chapter 2

Information Retrieval and Conversational Information Retrieval

In this chapter, we introduce the concepts of classical information retrieval (IR) and the recently established field of conversational information retrieval (CIR). We begin with an overview of IR, including methods for representing documents and queries. Next, we describe classical IR approaches and discuss several standard retrieval models. We then introduce CIR, exploring key concepts such as conversational context understanding and the role of user knowledge in the retrieval process.

The retrieval models and system components described in this chapter provide the basis for the experiments in this thesis. In particular, the sparse retrieval model BM25 and the neural retrieval model ColBERT (Sections 2.1.3 and 2.1.3) are used as the main retrieval systems in our experiments. The conversational query understanding methods discussed in Section 2.2.1, especially query expansion and query rewriting, provide the background for the proposed methods for identifying relevant conversation history and incorporating user knowledge.

2.1 An Overview of Information Retrieval

Information Retrieval (IR) can be thought of as the process of satisfying a user’s information need by finding relevant information from within a collection of knowledge sources (Manning, Raghavan, and Schütze 2008). An IR system searches the collection of knowledge sources with the goal of retrieving items that satisfy the user’s information need.

Figure 2.1 illustrates the traditional Information Retrieval (IR) process, also referred to as ad-hoc retrieval. It shows the main stages involved in transforming a user’s information need and finding potentially relevant documents to satisfy the information need. The process begins with a document collection (*Document Collection*), which serves as the knowledge source from which the information is to be retrieved. Before the relevant documents can be retrieved, the documents undergo a pre-processing and indexing stage. This typically involves operations such as tokenization, stop-word removal, stemming (*Linguistic Analysis*), for the construction of an index (*Indexed Documents*) that enables efficient search. The purpose of the indexing stage is to transform the raw documents into a representation that can be efficiently compared against user queries.

Similarly, when users (*Users*) submit queries (*Queries*), they are also subjected to corresponding pre-processing operations (*Linguistic Analysis*) to ensure consistency between the representations of documents and queries. As illustrated in Figure 1.1, both the indexed document representations and the processed query (*Query Representation*) are then passed to the retrieval and ranking component.

The retrieval and ranking component (*Retrieval Model*) compares the query representation against the indexed document representations using a retrieval model. Based on this comparison, each document is assigned a relevance score that estimates its likelihood of satisfying the user’s information need. The documents are subsequently ranked according to these scores, and the highest-ranked documents are returned as the retrieval results (*Documents Ranked Based on Relevance*).

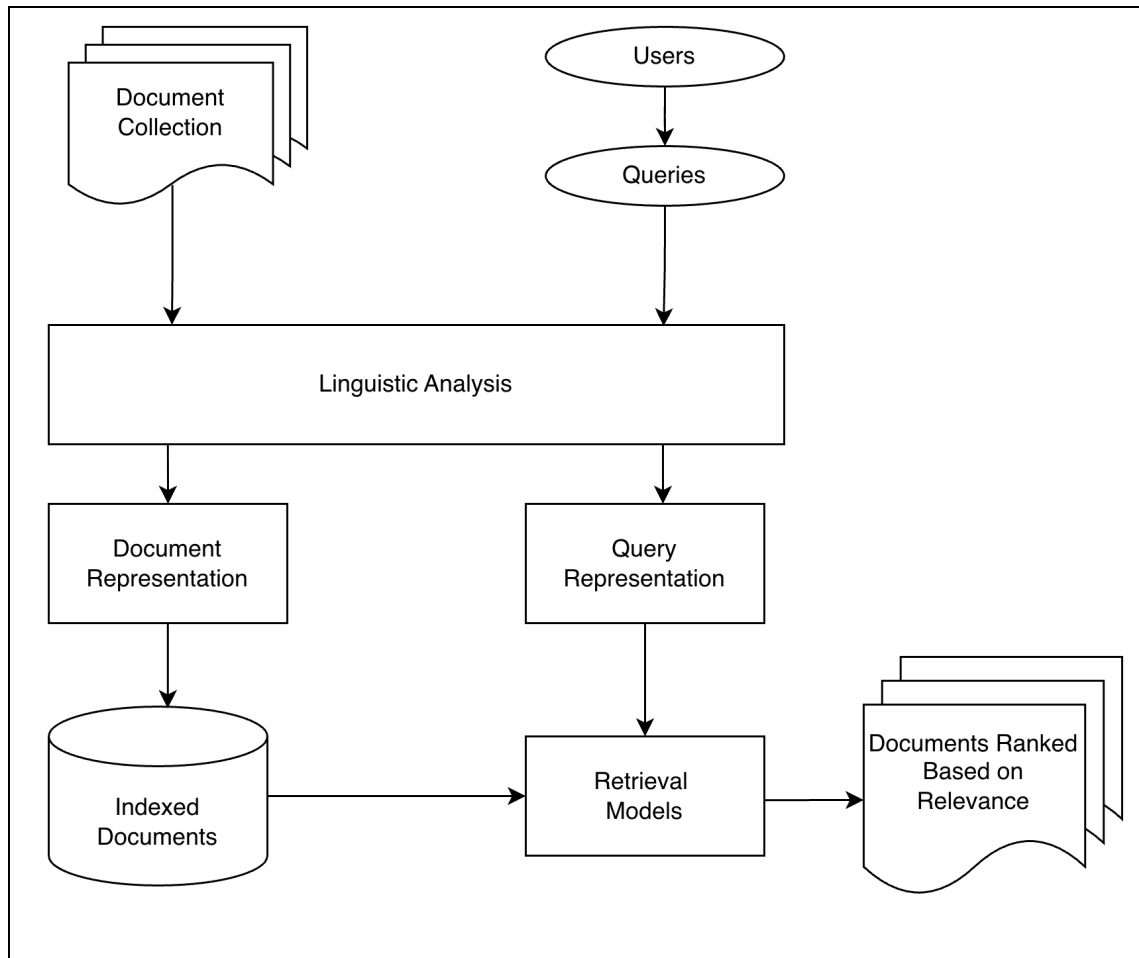


Figure 2.1: Traditional Information Retrieval Process.

2.1.1 Document Representation

As described previously, documents need to be pre-processed for use in a IR system. To achieve this, it is necessary to develop an efficient method for representing the documents. Such representation ensures that when a user submits a query, the IR system can efficiently retrieve potentially relevant documents. The document collection is generally assumed to be static, with new information incorporated only through the addition of new documents. This static characteristic allows for the pre-processing of the information prior to the retrieval phase. This pre-processing of documents within an IR system is referred to as indexing. Indexing is a process in which information is collected, parsed, formatted and stored to facilitate efficient retrieval (H. Huang and B. Zhang 2009). The goal is to create an inverted file data structure that lists, for every word, all documents that contain it and the frequency of

the occurrences in documents, referred to as an inverted index (Manning, Raghavan, and Schütze 2008). Indexing involves several natural language processing techniques, including tokenization, text normalization, stopword removal and stemming

2.1.2 Query Representation

Queries as represented in Figure 2.1 are the description of the information need as created by the *Users*. Queries are submitted to the IR system, and the task of the system is to return documents from the document collection that satisfy the user’s information need. The queries encode the information need (see Ganguly, Leveling, and G. J. Jones 2012, p. 14). These queries must be represented in a form that the search system is able to use to retrieve relevant documents in the collection. Although more complex forms of query representations are possible, the most common way of representing queries is in the form of a textual string, which is used in this thesis, consisting of terms describing the information need. Since the query is generally an incomplete or imprecise statement of the information need, and relevant documents will generally not describe the relevant information in a way that can simply match the query, the retrieval matching process is uncertain, and the IR system seeks to retrieve documents most likely to satisfy the information need.

2.1.3 Retrieval Models

The goal of the retrieval model is to locate and present relevant information need to the user and help them to address their information need as quickly and accurately as possible. The retrieval process, as represented by the *Retrieval Models* in Figure 2.1, compares the query representation to the document representations by a numerical score expressing their likelihood of being relevant. Many retrieval models have been developed over the years, which seek to effectively retrieve relevant lists of documents. In this section, we review some of the most influential information retrieval (IR) models in the literature, with particular emphasis on the models employed in the experiments conducted throughout this thesis.

Boolean Model

The Boolean retrieval model is the earliest and fundamental model in IR (Manning 2008). In this model, queries are represented by a combination of keywords that are connected using *boolean operators* such as AND, OR, and NOT. The document in this model is represented as a set of terms present within the documents, and the model retrieves relevant documents that satisfy the boolean expression. The model performs an exact match with the query when identifying the relevance of a document. An example query in boolean retrieval is given below:

(information AND retrieval) OR (data AND mining)

While the Boolean model is simple, easy to understand, and allows for efficient retrieval, a major drawback is that it does not provide a ranking of the retrieved documents. All matching documents are treated as equally likely to be relevant, which is rarely the case. Furthermore, because the model relies on satisfying Boolean expressions, it does not account for partial relevance in cases where sub-expressions are only partially satisfied. In addition, for the model to effectively retrieve relevant documents, the user must formulate a complex Boolean query that accurately captures their information need. This is a difficult task since users' information needs are often complex and not easily expressed through Boolean logic and many potential users are not familiar with Boolean logic.

Vector Space Model

The Vector Space Model (VSM) (Salton, Wong, and C.-S. Yang 1975) is a ranking-based retrieval model in which documents are ordered according to their potential relevance to the user's query, unlike the Boolean model. In the VSM, both the user query Q and documents in the collection D_j are represented as a vector over a multi-dimensional term space which consists of the vocabulary of the document

collection $V = \{t_1, t_2, \dots, t_m\}$. The document D_j and query Q are represented as:

$$D_j = (w_{1j}, w_{2j}, \dots, w_{mj})$$

$$Q = (w_{1q}, w_{2q}, \dots, w_{mq})$$

where w_{ij} and w_{iq} represent the weight of term t_i in the document D_j or query Q , respectively.

The notion of relevance of a document to a query in the VSM model is based on the degree of similarity between their vector representation. The closer the document vector is to the query vector, the more relevant the document is deemed to be to the query. The similarity in the vector space is determined by calculating the angle between their vectors. The cosine-similarity is the most widely used similarity measure, which calculates the angle between a document and a query vector, which is the dot product of two normalized vectors as shown in Equation 2.1, and is used to determine the relevancy of a document with respect to the query.

$$\text{sim}(D_j, Q) = \cos(\theta) = \frac{D_j \cdot Q}{\|D_j\| \|Q\|} \quad (2.1)$$

Probabilistic Model

The probabilistic model is a more theoretically motivated ranking- retrieval model based on the Probability Ranking Principle (PRP) (Stephen E Robertson 1977), where the model estimates the probability of relevance of a document D_j , given a query Q , $P(R = 1 \mid D_j, Q)$. The higher the probability, the more likely the document is to be relevant. The earliest version of the probabilistic model used a binary independence model (BIM), which assumes that the query term is either present or non-present in the document and the term occurrences are independent. It does not take into account the term frequencies for the number of times the term occurs in the documents, as well as the information about the length of the document. The BM25 (Best Matching 25) (Stephen E Robertson et al. 1995; K. S. Jones, Walker, and

Stephen E. Robertson 2000a), is an elaborated and very widely used retrieval model. This incorporates both term frequency and a length normalization component to address the inherent bias towards higher relevance scores for longer documents when term frequency is incorporated in the term weights. Each document D_j and query Q are compared using the BM25 ranking function defined as in Equation 2.2

$$\text{score}(D_j, Q) = \sum_{t \in Q} \log \frac{N - n_t}{n_t} \cdot \frac{f(t, D_j)(k_1 + 1)}{f(t, D_j) + k_1 \left((1 - b) + b \frac{|D_j|}{\text{avgdl}} \right)} \quad (2.2)$$

where $f(t, D_j)$ is the frequency of term t in document D_j , $|D_j|$ is the length of the document, avgdl is the average document length in the collection, N being the total number of documents in the collection and n_t the number of documents containing the term t and k_1 and b are tuning parameters. The k_1 parameter controls the effect of the frequency of a term t in the document and b controls the degree of document length normalization.

BM25's term-matching properties make it a natural baseline for evaluating query reformulation approaches, which makes it a good baseline for testing query reformulation methods that add useful information from the conversation history. As discussed in Chapter 3, BM25 is used as the main sparse retrieval baseline in all experiments. It provides a strong and consistent reference point for evaluating the proposed methods for using conversational context and user knowledge.

Language Modelling

The Language Modeling (LM) approach is also based on the PRP principle (Hiemstra 2001). In this approach, the LM estimates the probability that a query could be generated from the document. Now, the relevance score of the documents is calculated by computing the maximum log-likelihood of generating the query from the documents, as shown in Equation 2.3 and the documents are ranked in descending order of this likelihood. A higher value of $P(Q | D_j)$ implies that the document D_j

is more likely to be relevant to the query.

$$\text{score}(D_j, Q) = \log P(Q \mid D_j) \quad (2.3)$$

Now, the probability of generating query Q given document D_j is defined in Equation 2.4.

$$P(Q \mid D_j) = \prod_{t \in Q} \lambda P_{\text{MLE}}(t \mid D_j) + (1 - \lambda)P(t \mid D) \quad (2.4)$$

where:

$P_{\text{MLE}}(t \mid D_j)$: maximum likelihood of a term t occurring in D_j ,

$|D|$: total number of document in the Collection.

The maximum likelihood estimate probability of a term t occurring in a document can be estimated using Equation 2.5.

$$P_{\text{MLE}}(t \mid D_j) = \frac{f(t, D_j)}{|D_j|} \quad (2.5)$$

where:

$f(t, D_j)$: frequency of term t in document D_j ,

$|D_j|$: total number of terms in document D_j .

Neural Modeling

The Neural Modeling (NM) approach to IR is based on the principle that representing documents and queries using the terms present in them and computing the potential relevance of documents to the query based on lexical matching and statistical properties of the terms can often be inadequate, since it does not capture contextual semantic information. Query-document comparison based on lexical matching, while often effective, has an inherent limitation that they are often ineffective when there is a vocabulary mismatch where the query and document represent the same concepts using different terms. As users can formulate different queries using different words

for the same information need, and as such, the effectiveness of relying only on lexical matching can depend on the words selected by the user for inclusion in the query. Neural models seek to overcome this limitation by learning semantic representations of both the queries and documents, which allows for identifying relevant documents based upon semantic meaning rather than exact lexical matches. Neural models leverage neural networks to learn semantic representations. Words in the queries and documents are represented in a continuous vector space. These continuous vector space representations are called word embeddings. These word embeddings capture the contextual meaning of words by considering the surrounding text, unlike traditional word representations. Word embeddings are often pre-trained on large corpora of text, creating a word embedding model that allows for the learning of semantic representations. This leads to richer representations of queries and documents in the context of IR.

Both queries and documents are thus encoded into vector representations using the word embeddings. The similarity between these embedded models is then used as a measure of semantic similarity between the query and documents, which determines the relevance of each document. As described earlier for the VSM model (see Section 2.1.3), since embeddings are vector representations, similarity can be determined by calculating the angle between the embeddings, with cosine similarity being the most widely used similarity measure.

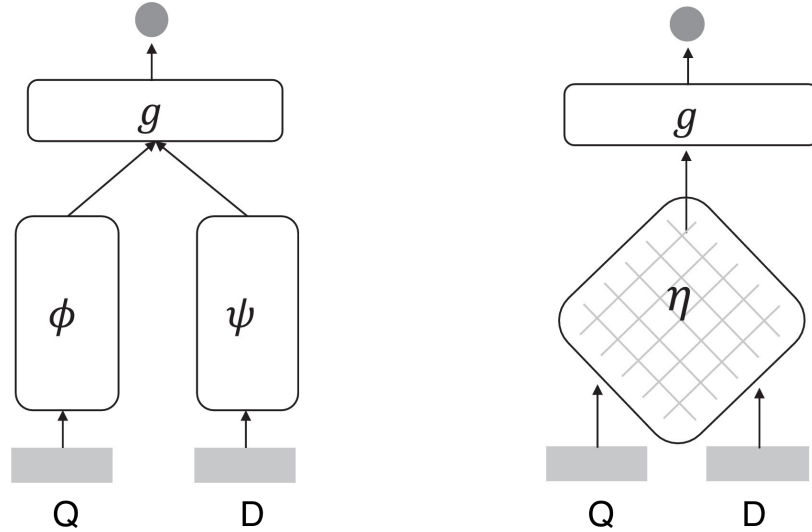
Neural models can be broadly categorized into the following types, as illustrated in Figure 2.2 (Mitra, Craswell, et al. 2018; Guo, Fan, Pang, et al. 2020; Trabelsi et al. 2021):

1. **Representation-focused models:** These models focus on using deep neural networks ϕ and ψ to extract high-level features and represent queries and documents independently in dense vector embeddings. Once the embeddings are obtained, the relevance between a query and a document is computed using similarity metrics such as cosine similarity, which are then used to rank the documents. Different types of deep neural networks have been used for this

purpose, fully-connected networks (P.-S. Huang et al. 2013), Convolutional Neural Network (CNN) (Shen et al. 2014a; Shen et al. 2014b) and Recurrent Neural Network (RNN) (Palangi et al. 2016; Wan et al. 2016).

2. **Interaction-focused models:** In these models, rather than using deep neural networks to extract individual representations of the query and document, as in Representation-focused models, a deep neural network is used to learn the interaction between the query–document pair η . This approach allows the model to capture matching signals between query and document tokens in a fine-grained manner to compute relevance scores (Guo, Fan, Ai, et al. 2016).

Hybrid models can also be constructed which combine both the representation and interaction models as well (Mitra, Diaz, and Craswell 2017; B. Wang, K. Liu, and J. Zhao 2016; S. Wang and Jiang 2016)



(a) Representation-focused

(b) Interaction-focused

Figure 2.2: Different types of Neural Models. (Modified from (Guo, Fan, Pang, et al. 2020) for consistency in the naming convention)

In the following sections, we review the main concepts needed to better understand the neural approaches and techniques.

2.1.4 Neural Modeling Foundations

This section reviews the neural architectures that form the foundation of the neural approach investigated in this thesis. We begin with the Transformer architecture, which introduced the self-attention mechanism that has become central to contemporary language models. We then discuss BERT, a Transformer-based language representation model, before examining ColBERT, a neural retrieval architecture that leverages contextualized token representations for efficient and effective document retrieval, which we use in our experiments described in this thesis.

Transformers

Transformers are a type of neural network model architecture (Vaswani et al. 2017) developed for sequence modelling tasks. These are an advancement over earlier approaches such as, Recurrent Neural Networks (RNN), Long Short-term Memory (LSTM) (Jozefowicz et al. 2016) and gated recurrent neural networks (Chung et al. 2014). RNNs and LSTMs process tokens sequentially and generate a hidden state as a function of previous hidden states and the current input position. This approach compresses all past information into a single hidden state, which limits the richness of the representation as the sequence length becomes longer, and disregards the information that is learned from the earlier tokens in the sequence. Transformer models, however, unlike recurrent models, rely entirely on the attention mechanism without the recurrent aspect, making them capable of modelling long-range dependencies. This attention mechanism, called self-attention relates to different positions of a single sequence to compute the representation of the sequence allowing every token in a sequence to look at all the other tokens to gather contextual information.

The Transformer architecture primarily comprises an encoder stack that maps an input sequence to produce a contextual representation and a decoder stack that, based on this contextual representation, generates an output sequence one element at a time, as illustrated on the left and right of Figure 2.3. The next section provides

a more detailed examination of the architecture.

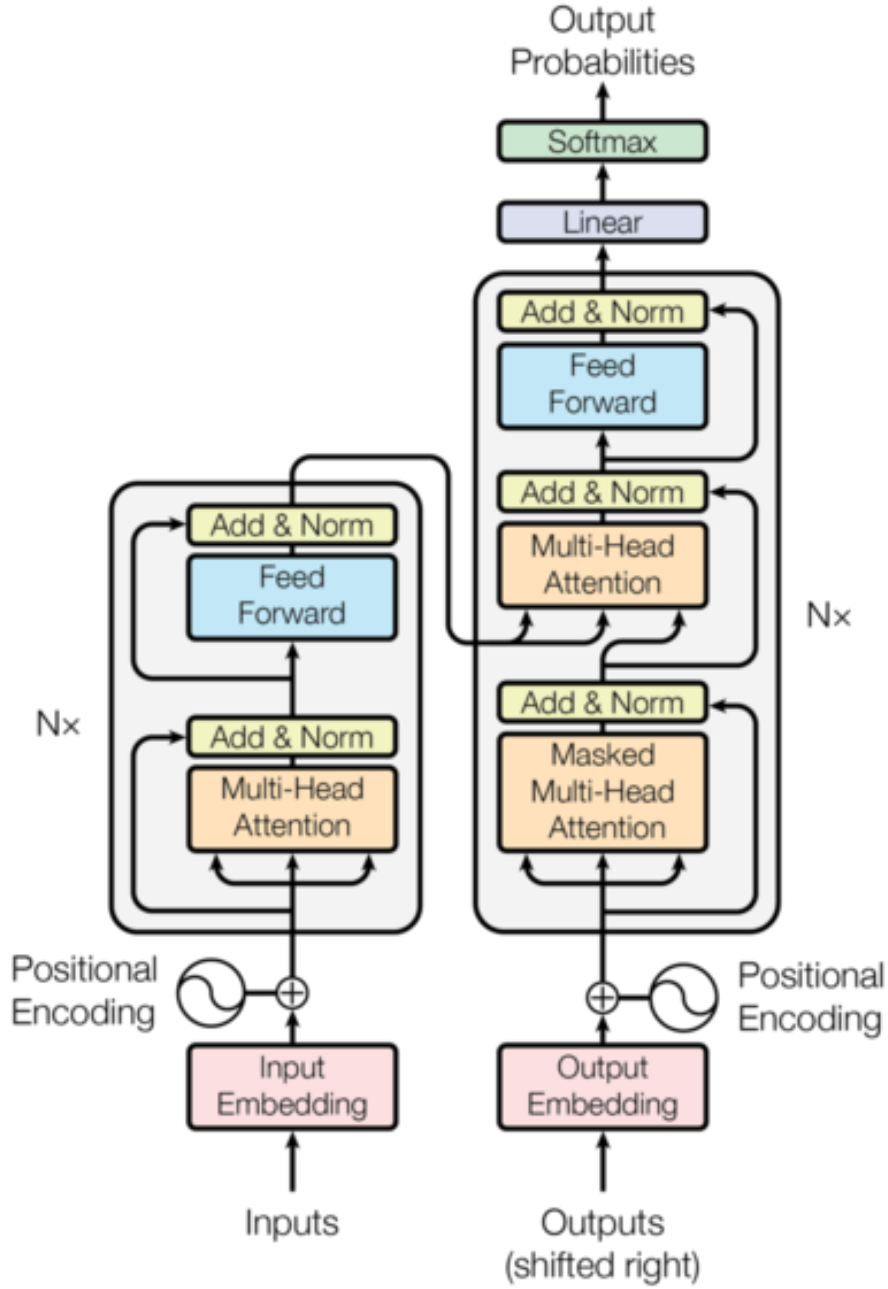


Figure 2.3: Transformer Architecture (Vaswani et al. 2017).

Encoder. The encoder stack is composed of identical layers ($N=6$; (Vaswani et al. 2017)). Each layer is further composed of two sub-layers, namely: (1) multi-headed self-attention and (2) position-wise fully connected feed-forward network.

All sub-layers in the model, as well as the embedding layers, produce output of the same dimension d_{model} . ($d_{model} = 512$ (Vaswani et al. 2017)).

Decoder. The decoder stack is also composed of identical layers as in the encoder layer. The decoder sub-layer consists of three sub-layers, namely: (1) masked multi-headed self-attention, (2) multi-headed self-attention, and (3) position-wise fully connected feed-forward network. The multi-headed self-attention in the decoder (2) performs multi-headed attention over the output of the encoder stack. The masked multi-headed self-attention (1), is a modified multi-head self-attention that uses masking to prevent attending to subsequent positions. This, along with the output embedding being offset by one position, allows the prediction to depend only on the known outputs at previous positions.

Attention Mechanism. The attention mechanism is a mapping for a query and a set of key-value pairs to an output. The *query* represents the current element for which the model is computing attention. The *key* represents each element in the input sequence and measures how relevant it is to the query. The *value* contains the actual information of each input element. The *query*, *key* and *value* enable the model to focus on the most relevant parts of a sequence. The Transformer uses “scaled dot-product attention” as shown in Figure 2.4. A dot product of the query with all the keys is computed and scaled by a factor of $1/\sqrt{d_k}$, after which a softmax function is used to obtain the weights on the value. The attention function is computed on a set of *queries*, *key* and *values*, Q , K and V simultaneously, as shown in Equation 2.6.

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^\top}{\sqrt{d_k}} \right) V \quad (2.6)$$

Transformers use a multi-head attention mechanism, whereby the queries, keys, and values are linearly projected h times using multiple sets of learned (Q , K , V) projections. This allows the model to attend to information from different representation subspaces simultaneously. For each of these learned projections, self-attention is computed in parallel, and the resulting outputs are then concatenated and linearly projected, as in Equation 2.7, yielding the final output of the multi-head attention mechanism, as illustrated in the right half of Figure 2.4.

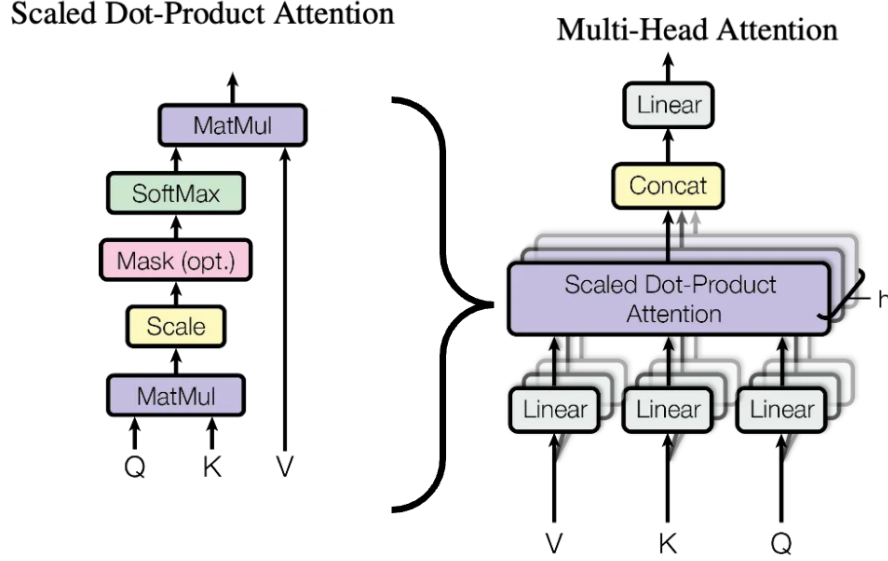


Figure 2.4: Attention Mechanism (Vaswani et al. 2017)

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (2.7)$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$

Point-wise Feed-Forward Network. Each Transformer layer includes a Feed-forward Network (FFN) applied independently to each token. This network adds non-linear transformations, which help the model capture complex representations and relationships between tokens as shown in Figure 2.3.

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2 \quad (2.8)$$

Residual Connection with Layer Normalization. Residual connections (He et al. 2016), facilitate training in deep networks by alleviating the problem of vanishing gradients. After each sub-layer in the Transformer (both encoder and decoder), a residual connection is followed by Layer Normalization (LayerNorm) (Ba, Kiros, and Hinton 2016) as shown in Equation 2.9.

$$\mathbf{y} = \text{LayerNorm}(\mathbf{x} + \text{Sublayer}(\mathbf{x})). \quad (2.9)$$

Since the self-attention mechanism in Transformers does not inherently under-

stand the order of tokens, positional encoding is added to inject token position information. This encoding uses sinusoidal functions to ensure the model can generalize to sequences of varying lengths, improving its ability to capture

Positional Encoding. Since the self-attention mechanism in Transformers does not inherently understand the order of tokens. Positional encoding is added to the input and output embeddings to add the position information of the tokens.

$$PE_{(pos, 2i)} = \sin\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right), \quad PE_{(pos, 2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right) \quad (2.10)$$

where, pos denotes the position index in the input sequence, i represents the dimension within the embedding vector, and d_{model} is the dimensionality of the model’s embeddings.

In the following section, we look into BERT, which uses the Transformer architecture for pre-training language representation.

BERT

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2018) is a language representation model based on Transformer architecture (Vaswani et al. 2017) and uses only the encoder part of the Transformer stack as described in Section 2.1.4. The BERT model addresses the limitation of previous standard language models being unidirectional, i.e left-to-right or right-to-left, (Peters et al. 2018; Radford 2018) when learning the representation by allowing the model to enhance the context representation by incorporating the information available in both directions. This allows it to capture richer contextual dependencies, which helps in understanding the full context of both queries and documents. BERT’s pre-training is based on two tasks that facilitate this bidirectional learning.

1. **Masked Language Model (MLM):** Some of the input tokens are randomly masked (15%), using [MASK] token, where the goal is to predict the masked token word (Taylor 1953). For every position where a [MASK] has been

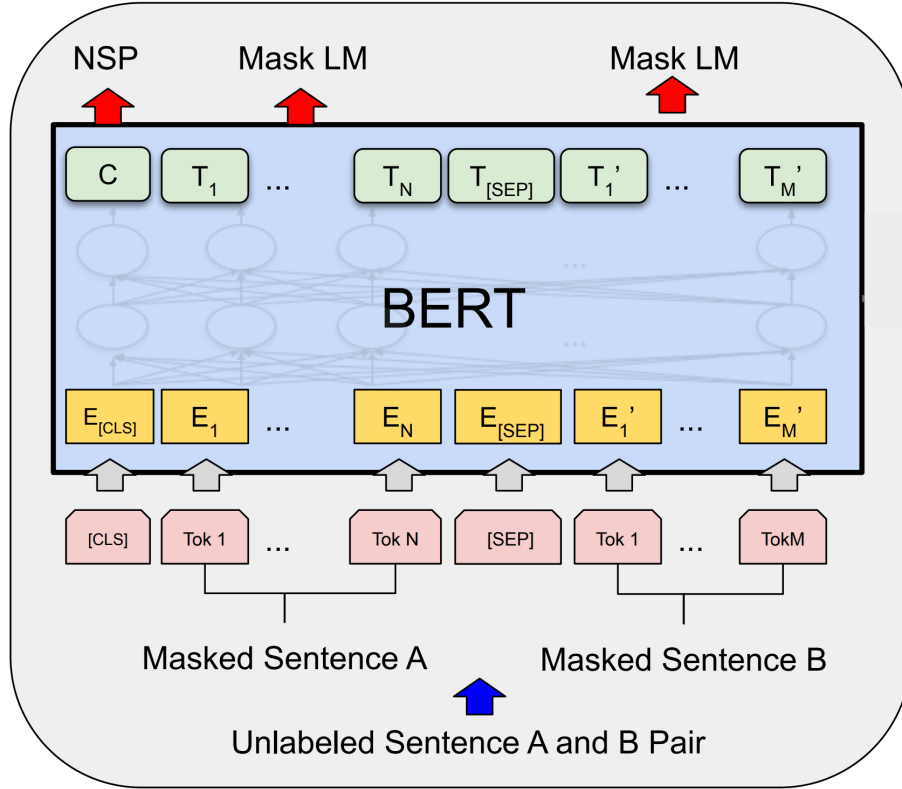


Figure 2.5: Overall pre-training procedures for BERT. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token. (Devlin et al. 2018)

inserted, BERT takes the final hidden state vector and feeds it into a softmax classifier over the entire vocabulary to predict the original (masked) token. This masking procedure allows the language model to learn token representation using the context from both directions, allowing BERT to learn deep contextual embeddings.

2. **Next Sentence Prediction (NSP)**: In this task, the model is given a pair of sentences A and B, and is trained to learn the relationships between the sentences, which is not directly captured by a language model. In the pre-training of the BERT model, the sentences are taken from monolingual corpora, where 50% of the time, sentence B is the sentence following A and 50% of the time, it is a random sentence from the corpus. This allows BERT to model sentence-level relationships.

Figure 2.5 shows the overall pre-training procedures for BERT. The pre-training

corpora used for BERT are the BookCorpus (800M words) (Yukun Zhu et al. 2015) and English Wikipedia (2,500M Words) (Devlin et al. 2018).

The Masked Language Model (MLM) and Next Sentence Prediction (NSP) objectives in the BERT model allow for the pre-training of the model on a large corpus without the need for specific training data, enabling BERT to learn semantically and contextually rich, general-purpose language understanding. This can then be fine-tuned for various Natural Language Processing (NLP) tasks, making it highly effective for a wide range of natural language processing applications, including question answering, sentiment analysis, text classification, and information retrieval, and has been shown to be effective (Devlin et al. 2018). Language models like this, trained on large general text corpora, are called Large Language Models (LLM) and are among the most common approaches in deep learning for NLP-related tasks.

ColBERT

Contextualized Late Interaction over BERT (ColBERT) (Khattab and Zaharia 2020a) is a neural ranking model specifically designed to enhance retrieval performance by leveraging the BERT language model for efficient, interaction-focused ranking. ColBERT introduces a *late interaction* mechanism, where the query and document are independently encoded into contextual embeddings using a BERT-based encoder. Relevance between the query and document is then evaluated after both have been processed. Rather than collapsing each query or document into a single vector as is the case for representation-focused neural models, ColBERT retains token-level embeddings, which allows for token-level similarity computation between the query and document. Figure 2.6 shows the general architecture of ColBERT.

Query Encoder. Given a query q , the query is tokenized $\{q = (q_1, q_2, \dots, q_{|q|})\}$ and encoded independently using BERT, which computes a contextualized representation of each query token.

$$\mathbf{Q} = f_Q(q)$$

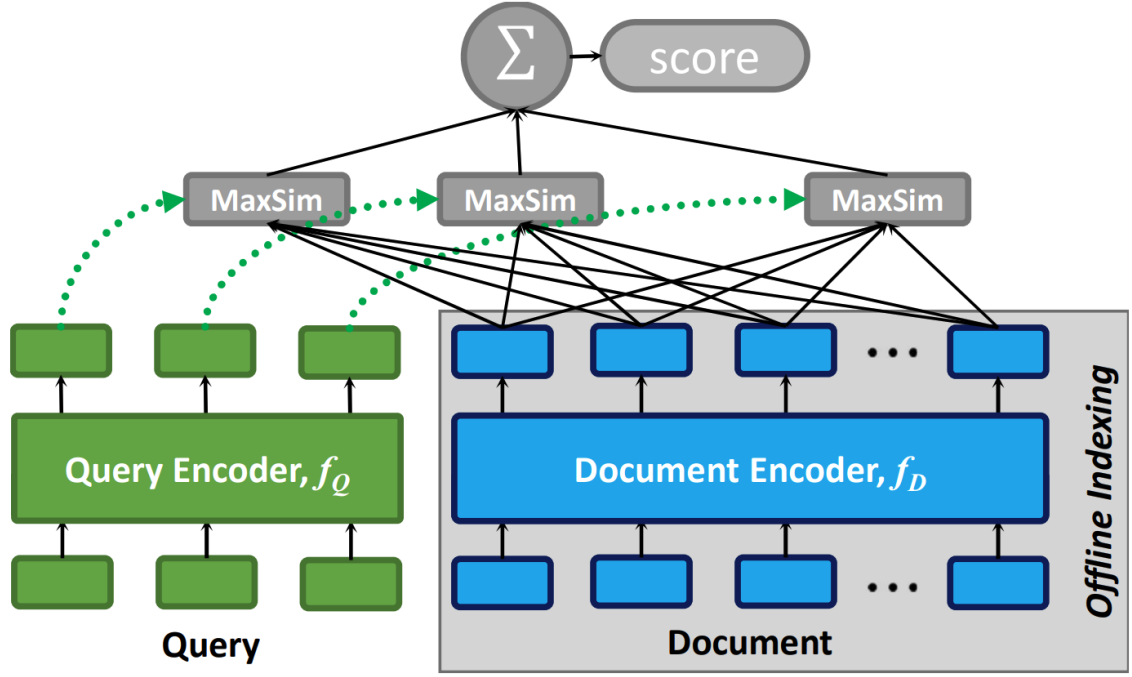


Figure 2.6: The general architecture of ColBERT given a query and a document (Khattab and Zaharia 2020a).

Document Encoder. Similar to the query encoding process, the document encoder also encodes a Document d using BERT to generate a contextualized representation of the document tokens $\{d = (d_1, d_2, \dots, d_{|d|})\}$.

$$\mathbf{D} = f_D(d)$$

Late Interaction. In this mechanism, the similarity between the query Q and Document D is calculated by computing the maximum cosine similarity between each query token embedding q_i and document token embedding d_j as shown in Equation 2.11.

$$\text{MaxSim}(\mathbf{q}_i, \mathbf{D}) = \max_j (\mathbf{q}_i \cdot \mathbf{d}_j) \quad (2.11)$$

The final relevance score is the sum of these maximum similarities.

$$\text{score}(\mathbf{Q}, \mathbf{D}) = \sum_{i=1}^m \text{MaxSim}(\mathbf{q}_i, \mathbf{D})$$

This MaxSim operation effectively preserves token-level contextual interactions

while allowing independent encoding, which enables precomputing and indexing of document embeddings for scalable retrieval.

ColBERT’s late-interaction approach is well-suited to conversational retrieval because it compares queries and documents at the word level. This helps it find more detailed meaning matches, especially when queries are expanded using conversational history or user knowledge. In this thesis, ColBERT is used as the dense retrieval model in the experiments described in Chapter 3. This allows us to test whether the proposed methods work well with both sparse and dense retrieval systems.

2.1.5 Re-ranking

Re-ranking is an important component of many modern IR system pipelines. We describe it here for the sake of completeness as part of the standard retrieval process, although as we do not use re-ranking in our experimentation.

In a typical IR system pipeline, re-ranking follows an initial retrieval step, where a set of documents is first retrieved using a low-cost, efficient model like BM25. This initial step returns a set of candidate documents in a computationally efficient manner and does not take into consideration richer features for query-document relevance. To utilize the richer features of query-document relevance, a more sophisticated model, often a deep learning-based model like, is then applied to re-rank the retrieved documents in the initial retrieval step, which refines the ordering of the documents to ensure more relevant documents are ranked higher.

In re-ranking, the IR system updates the ranking of the retrieved results based on additional context provided by the user or inferred from the relevant documents. This involves refining the initial ranking based on user feedback, clarifications, or implicit signals gathered during the interaction. A user’s query submitted to an IR system may only target a small fraction of the vast collection of items. In this case, only a subset of the items which match the items will be relevant to the user’s information need. Searching through the entire collection for relevant documents can be computationally expensive and may lead to poor recall. Hence, re-ranking

of retrieved items to focus on those which are more likely to be more relevant is employed for more efficient retrieval.

2.2 Overview of Conversational Information Retrieval

Conversational information retrieval (CIR) (also called *conversational search* or *conversational information seeking*) is an IR system with a conversational interface enabling users to seek information via multi-turn natural language dialogues, in spoken or written form, as shown in Figure 2.7 (Gao et al. 2023; Zamani et al. 2023).

The figure illustrates a three-layer CIR architecture comprising a *User Experience Layer*, a *Conversational Search Engine Layer*, and a *Data Layer*. User text or voice queries $Q_1, \dots, Q_N$ are passed through a pipeline that begins with *Contextual Query Understanding*, where query expansion or rewriting models resolve ambiguities inherent in multi-turn dialogue. The reformulated query is then used for *Document Retrieval* using sparse or dense retrieval models, supported by a *Data Layer* consisting of heterogeneous knowledge sources (knowledge bases, indexed document collections, user profiles, and search session logs). Retrieved documents are subsequently re-ranked before a final *Result Generation* stage, producing a device-appropriate response such as a ranked list, a summary, or a direct answer. Together, the contextual query understanding and retrieval stages form the core *Conversational Dense Document Retrieval* sub-component, underscoring the importance of conversational context understanding for effective conversational retrieval.

As outlined in Chapter 1, traditionally, IR systems have been completely user-driven, with the system having no way of engaging with the user in the search process beyond simple relevance feedback methods (Lavrenko and W Bruce Croft 2017; S. Robertson, Zaragoza, et al. 2009). CIR systems seek to address this limitation by incorporating an interactive and collaborative process between all the actors involved. CIR has recently garnered considerable attention (Zamani et al. 2023;

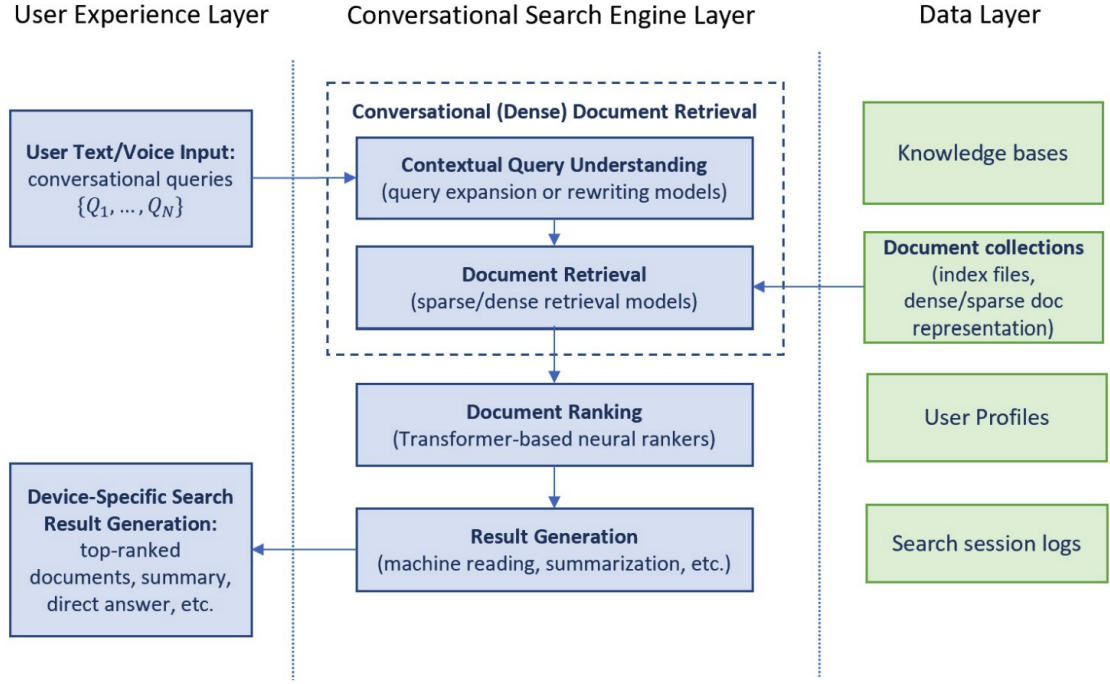


Figure 2.7: A general architecture of CIR (Gao et al. 2023).

Brown et al. 2020) and has been an increasingly popular research area (Anand et al. 2020; Culpepper, Diaz, and Smucker 2018). This has been, to a large extent, enabled by recent advancements in deep learning, which have led to the widespread emergence of conversational personal assistants and systems such as ChatGPT (Brown et al. 2020). While recently popular, interest in conversational search has a longer history. Some of the earliest work investigates considering IR systems as an intermediary in helping users better understand their information need (Brooks and N. J. Belkin 1983). A new approach to designing retrieval systems with the goal of understanding users' information needs was described in (W. B. Croft and Thompson 1987), which envisioned strategies to facilitate query formulation, domain knowledge acquisition, explanation, browsing, retrieval, and evaluation. More recently, Radlinski and Craswell (2017) examined conversational approaches to IR and proposed a set of characteristics they deemed that a conversational IR system should possess. One of their considerations is that a conversation should be adaptive. This feature aligns with our idea that an IR system should be able to adapt as a search conversation progresses, specifically dealing with the conversation history and user knowledge including knowledge developed during the conversation, until the information need

is satisfied.

2.2.1 Conversational Query Understanding

As described earlier, conversational dialogues involves various types of natural language phenomenon, such as, omissions, ellipsis etc (Zamani et al. 2023). During CIR, the user issues a series of queries describing their information need to the IR system and based on these queries, tries to identify the relevant documents from the knowledge sources. As the conversation progresses the user queries lose information or details, because in natural dialogue setting it is assumed that the participants do not have to fully specify all information that has already been included in the earlier turns in the dialogue. As such at any turn during the conversation, the user-issued query may make implicit or explicit reference to information contained in an earlier turn in the conversation. This property of dialogues means the current query often lacks all the details needed for CIR system to be able to retrieve relevant documents. In order to understand the current query, we need to look into the conversational history and try to identify or infer the information missing from the query in order to create a full query statement for entry into the IR system. Most existing query understanding approaches are quite limited. They focus on disambiguating and contextualizing the current query with respect to the conversational history, also known as query reformulation (Mo, Mao, Z. Zhao, et al. 2025; Zamani et al. 2023; Gao et al. 2023), but are restricted to rather conventional query expansion and query rewriting approaches.

Query Expansion. Query expansion is a longstanding method originally developed to improve search effectiveness in traditional single-shot search (Carpineto and Romano 2012). The key idea is to add expansion terms to the input user query to make it a better and fuller statement of the user’s information need with the goal of improving the likelihood of retrieving relevant documents. The original query Q is reformulated into a new query Q' by including additional terms e_i into the original

query as shown in Equation 2.12.

$$Q' = \underbrace{\{q_1, q_2, \dots, q_n\}}_{\text{original query } Q} \cup \underbrace{\{e_1, e_2, \dots, e_m\}}_{\text{expansion terms}} \quad (2.12)$$

In CIR, expansion terms e_i are derived from within documents retrieved in previous turns in the dialogue. In traditional single-shot IR, the expansion terms are selected from terms contained in documents identified as relevant or assumed to be relevant after an initial search operation. In CIR, several approaches to query expansion have been explored. These approaches ranged from heuristic approaches of selecting k previous query turns for expansion (Clarke 2019; Ohsugi et al. 2019; Mehri, Eric, and Hakkani-Tur 2020), utilizing linguistic features (S.-C. Lin, J.-H. Yang, Nogueira, Tsai, C.-J. Wang, and J. J. Lin 2021; Mele et al. 2020) to determine the expansion terms, to machine learning based approaches (Cao et al. 2008; Xiong and Callan 2015; Voskarides et al. 2020) that try to determine the term relevance from the previous query turn for expansion. (Mo, Nie, et al. 2023) expanded the original query by identifying historical turns based on a selection model that identifies related historical query turns based on their usefulness to improve system retrieval performance. (Mele et al. 2021) proposed an approach that classifies user queries based on their ambiguity and uses linguistic analysis to identify missing context from previous query turns based on this classification to expand the query.

In this thesis, query expansion is used as one of the baseline strategies for incorporating conversational context. In particular, we study how choosing specific past turns, instead of using the full history, affects retrieval performance. This leads to our proposed hybrid method for identifying relevant previous turns, described in Chapter 4. This method uses semantic similarity to select the most relevant earlier turns for expanding the query.

Query Rewriting. Query rewriting is the process of reformulating a user’s original query into an improved or alternative version to enhance retrieval effectiveness. The query is reformulated by utilizing the current user query in addition to the historical

conversational context from the dialogue, to create a new query which is intended to be an improved contextualized statement of the information need, as shown in Equation 2.13.

$$Q'_t = f(Q_t, C_{t-1}) \quad (2.13)$$

where:

Q_t : current user query at turn t ,

C_{t-1} : conversation context from previous turns ,

Q'_t : rewritten self-contained query,

f : query rewriting function that incorporates context.

The most commonly used approach for the query rewriting function is to use a pre-trained language model (Devlin et al. 2018) based on a Transformer model (Vaswani et al. 2017) to generate a new query (Vakulenko, Voskarides, et al. 2021; S.-C. Lin, J.-H. Yang, Nogueira, Tsai, C.-J. Wang, and J. Lin 2020). (S.-C. Lin, J.-H. Yang, Nogueira, Tsai, C.-J. Wang, and J. Lin 2020) evaluated different types of pre-trained language models to study their effectiveness for conversational query rewriting, such as GPT-2 (Radford et al. 2019) (pre-trained decoder-only transformer), BERT (Devlin et al. 2018) (pre-trained encoder-only transformer), UniLM (Dong et al. 2019) (BERT architecture pre-trained using unidirectional, bidirectional and sequence-to-sequence language modelling), T5 (Raffel et al. 2020) (an encoder-decoder transformer). These studies showed that pre-trained language models are effective for query rewriting, and the T5-based model demonstrated a better ability to rewrite conversational queries.

Query rewriting using a T5-based model is used as the main method for reformulating queries in our experiments. As described in Chapter 4, we test how different ways of selecting past conversational context, including our proposed hybrid method, and utilizing T5-based rewriting model to reformulate queries, affect the performance of the retrieval systems across different datasets.

There are also hybrid approaches, which combine query expansion with query rewriting approaches. A generative query reformulation method leveraging a pre-trained language model has been proposed to rewrite user queries and expand them with responses generated by the language model (Mo, Mao, Yutao Zhu, et al. 2023). Combining query expansion and query rewriting has been shown to improve upon the individual approaches (Vakulenko, Voskarides, et al. 2021). Neural retrieval models, as described in Section 2.1.3 have also been used for the task of query understanding, without having the need for performing query reformulation. The query and conversation history are encoded into a dense vector representation, which is then used to retrieve relevant documents (Qu et al. 2020; Yu et al. 2021).

Challenges in Conversational Query Understanding

An essential aspect of effective query reformulation is the exploitation of information from previous relevant dialogue turns. Utilizing all historical turns as context (S. Kim and G. Kim 2022; S.-C. Lin, J.-H. Yang, and J. Lin 2021; Yu et al. 2021), assume that all previous queries are helpful to query reformulation for the current turn. However, this might not necessarily be the case, due to the nature of natural dialogue. Also incorporation of queries from all previous turns becomes increasingly inefficient as the length of the search dialogue increases. (Ohsugi et al. 2019) utilizes the previous two turns, (Mehri, Eric, and Hakkani-Tur 2020) considers the previous three turns, (Clarke 2019) consider the first and the previous turn in the conversation and exploits it to reformulate the query based on the observation that the first turn at the start of the conversation represents the topic of the conversational and contains the most information about the information need. (Vakulenko, Longpre, et al. 2021) utilize the previous five dialogue turns. While these heuristic-based approaches for selecting a set of previous turns are simple and efficient, they may overlook deeper semantic relationships and the more complex dialogue context.

Furthermore, recent advances in LLMs are significantly influencing query under-

standing approaches. This allows for inclusion of as many historical turns as possible since LLMs are capable of exploiting and subsequently using them for query rewriting (Mao, Dou, Mo, et al. 2023; S.-C. Lin, J.-H. Yang, Nogueira, Tsai, C.-J. Wang, and J. Lin 2021). Generative approaches are also being used to generate concepts related to the conversational context and using these generated concepts for query reformulation (Rocchietti et al. 2023). (Mao, Dou, Chen, et al. 2023; Galimzhanova et al. 2023) use LLMs to generate query rewrites using prompting techniques to construct a stand-alone search query demonstrating exceptional potential across various IR tasks, including, but not limited to query reformulation (Yutao Zhu, H. Yuan, et al. 2025; Yutao Zhu, P. Zhang, et al. 2024).

Other strategies for identifying relevant historical turns consider it a classification task and build a machine learning model to help with the task. However, training models for this purpose requires a significant amount of data, which is often scarce in conversational search. Alternatives that do not depend on training models typically use previous k-turns as related context and employ LLMs to generate search queries or use linguistic analysis on those turns to extract features to enrich the current query, which can overlook the semantic relationships that can run deeper in the conversational history, as well as miss complex contexts in relation to the dialogue.

Existing methods for identifying relevant past turns in conversational search can be grouped into two main types. First, simple heuristic-based methods (Clarke 2019; Ohsugi et al. 2019; Mehri, Eric, and Hakkani-Tur 2020) select turns based on fixed positional rules. These methods do not capture deeper meaning or more complex patterns in the conversation. Second, machine learning methods treat the task as a classification problem, but they require large amounts of labelled data, which is often not available in conversational search. More recently, large language model (LLM) based methods have shown potential for identifying relevant past turns, but they have not been thoroughly tested in terms of their impact on retrieval performance. Importantly, none of these approaches combines simple structural rules with semantic similarity in a single hybrid method. This gap motivates the first part of this thesis.

We propose and evaluate a hybrid approach that combines simple structural rules with semantic similarity signals to identify relevant past turns. We also study how this affects retrieval performance across different datasets and retrieval models, as described in Chapter 4.

In addition, a key limitation in existing research is that it ignores the knowledge that users gain during a search conversation. Most methods treat the conversation only as a list of queries and responses, without considering how the user’s understanding changes over time. This motivates the second part of this thesis, which is introduced in the next section.

2.3 User Knowledge in Information Retrieval

As introduced in Chapter 1, users approach an IR system with an information need or deficit of knowledge. This is addressed more formally in the concept Anomalous State of Knowledge (ASK) framework introduced by (Nicholas J Belkin 1980). This section introduces consideration of the cognitive processes underlying information-seeking behaviour in IR. The ASK framework describes how people seek information when they recognize a gap or anomaly in their knowledge state, and suggests that IR systems must be viewed based on the user’s knowledge. In the ASK model, when a user encounters an information need that they cannot fully define and recognizes that there is a gap or inadequacy in their current understanding of the information need, they are in an ASK. This anomalous state creates a problem where users are unable to clearly articulate their information need, which can be described as the *non-specifiability of information need* problem as described in Chapter 1. Traditional IR systems assume by default that the user can express their information need sufficiently to achieve effective IR, which is not often the case. In the ASK framework, the emphasis is on the user-centric perspective, suggesting that the user’s knowledge is central to IR and as such IR systems need to be able to support users in addressing their ASK through iterative clarification, exploration, and reformulation of their information need.

Kuhlthau (1991) proposed a user-centered perspective on the Information Search Process (ISP) in information seeking. This study examined various aspects of users' activities as they interact with information, focusing on how these interactions build upon their existing state of knowledge about a topic. The ISP model incorporates three dimensions of user activity: cognitive (thoughts), physical (actions), and affective (feelings). Together, these reflect the user's evolving knowledge state during the search process. According to this model, users engage in a dynamic progression from an initial state of information need to a goal-oriented state of understanding, moving through a series of distinct stages as summarized in Table 2.1. Each of these stages in terms of the cognitive and physical aspects outlined in more detail below.

- **Initiation:** In this stage, the user becomes aware of a lack of knowledge about a search problem or topic. The user's task is to recognize that there is a need for information and to cognitively relate the problem to their prior knowledge in order to describe their information need. This stage relates to the ASK framework (Nicholas J Belkin 1980).
- **Selection:** In this stage, various general topics or ideas relating to the search problem are identified and evaluated against the information goal, so that the information seeking process can begin.
- **Exploration:** In this stage, the information available is explored. The task is to investigate the information on the general topic in an effort to assimilate the information, and try to relate it to the user's existing understanding of the topic and further to gain a better understanding of the information need that will allow for a more focused view of the search problem.
- **Formulation:** In this stage, a more focused view is formed based on identifying and selecting relevant information from the exploration stage, which can then be used to form a narrower or clearer perspective of the topic.
- **Collection:** In this stage, a focused perspective of the topic is used to gather

the information. The action is typically to seek focused information about the topic to extend the user’s knowledge and understanding of the search problem.

- **Presentation:** In this stage, the search task is completed. The user has collected information to improve their understanding of the search problem which builds their knowledge that can satisfy their information need which triggered the search activity.

Stage	Feelings (Affective)	Thoughts (Cognitive)	Actions (Physical)	User Task
<i>Initiation</i>	Uncertainty	General/Vague	Seeking Background Information	Recognize
<i>Selection</i>	Optimism			Identify
<i>Exploration</i>	Confusion/Frustration/Doubt		Seeking Relevant Information	Investigate
<i>Formulation</i>	Clarity	Narrowed/-Clearer		Formulate
<i>Collection</i>	Sense of Direction/Confidence	Increased Interest	Seeking Relevant or Focused Information	Gather
<i>Presentation</i>	Relief/Satisfaction or Disappointment	Clearer or Focused		Complete

Table 2.1: Kuhlthau’s Information Search Process (ISP) Stages. (Kuhlthau 1991)

Kuhlthau (1991)’s ISP model extends on Belkin’s ASK model (Nicholas J Belkin 1980) by describing how users resolve the ASK through successive stages of interaction with information to progressively reduce the ASK as the user constructs new meaning and understanding of the information need.

Vakkari (2016) proposed a Searching-as-Learning (SAL) paradigm for the cognitive nature of information-seeking behaviour. Building on Belkin’s ASK model and Kuhlthau (1991)’s ISP framework, SAL models the information seeking behaviour as an inherently learning oriented activity, beginning with the ASK and following through the various stages of ISP model, through which users acquire, reorganise, and deepen their knowledge as users refine their understanding of the information, resolving their ASK. Drawing on several learning theories of human knowledge,

Vakkari (2016) defines knowledge as a structured network of concepts and their relationships representing a phenomenon. Based on this definition, he further defines learning as a means of acquiring new knowledge, as well as modifying and reinforcing existing knowledge structures.

The degree or extent of learning depends on the user’s prior knowledge and can be categorized as follows Vakkari (2016):

- **Accretion.** The addition of information to the existing knowledge structure without any changes to the structure itself.
- **Tuning.** The modification of the existing knowledge structure without replacing concepts or their relationships.
- **Restructuring.** Changing the existing knowledge structure by replacing concepts and their relationships, resulting in a substantially altered understanding.

In the context of SAL, users engage with retrieval systems with the explicit aim of acquiring knowledge about a specific topic. Retrieval systems should not just focus on efficiency or relevance, but should also support users learning processes, facilitating exploration, reflection, and integration of new knowledge. Through iterative interactions with the retrieved documents, a user progressively accumulates information to satisfy their learning goals. This process entails not only accessing relevant documents but also assimilating and comprehending the information contained within them.

A study focusing on the differences in searching patterns between domain experts and non-experts regarding a shared subject is reported in (White, Dumais, and Teevan 2009). Their findings highlight distinctions in query formulation strategies across various levels of expertise, emphasizing the importance of understanding how different users approach search tasks. Furthermore, the level of domain knowledge influences how the user formulates their queries (O’Brien et al. 2020; Willoughby et al. 2009; Dosso et al. 2024), with non-experts using more keywords than experts and experts producing more new keywords than non-experts (Dosso et al. 2022; Sanchiz,

Chevalier, and Amadiou 2017). Moreover, (X. Zhang, Cole, and N. Belkin 2011) found that insights from data collected during the search process can provide valuable indications of a user’s domain knowledge, suggesting that analyzing user interactions with retrieval systems can yield valuable information about their knowledge. A study by (Hagen et al. 2016) showed that users can learn query terms while engaging in searching and reading activities. This suggests that during the search process, users gradually refine their query formulation and enhance their understanding of the topic through iterative interactions with search results, highlighting the importance of incorporating knowledge mechanisms into the retrieval systems. A search system can assist users in achieving their learning objectives more quickly by estimating how much they learn. For example, it can do this by retrieving documents that match not only their specific query, but also their existing knowledge about the topic.

Several frameworks have been developed to integrate user knowledge into retrieval systems (Hoppe et al. 2018; Syed and Collins-Thompson 2017a; Syed and Collins-Thompson 2017b). (Câmara, El-Zein, and da-Costa-Pereira 2022) introduced a framework focused on representing user knowledge during search sessions. This framework estimates a user’s knowledge about a specific topic by maintaining a representation that continuously updates throughout the session. They achieve this by employing a combination of keyword-based methods and Large Language Models (LLMs) based methods. Their study demonstrates that this representation effectively correlates with users’ actual knowledge levels. Expanding on this work, the framework was extended to incorporate named entities, leveraging the relationships between these entities to better represent and measure user knowledge during a search session (El Zein, Câmara, et al. 2023). This enhancement suggests that utilizing named entities complements earlier approaches, offering a more nuanced estimate of users’ knowledge. Additionally, (Nasser et al. 2024) utilized knowledge graphs to represent both the user’s knowledge and the knowledge goal, demonstrating that the graph-based approach can capture complementary aspects of knowledge.

One of the many challenges when considering the user’s knowledge is its rep-

resentation. Several approaches have been proposed for this purpose. Syed and Collins-Thompson (2017a) uses the number of keyword occurrences read out of the keyword goal, El Zein and Costa Pereira (2021) uses keywords which a user has been exposed to while surfing on the Web.

Although previous work has shown that user knowledge is important in the search process, using it in conversational information retrieval (CIR) is still largely unexplored. Earlier studies on user knowledge, such as those by (Câmara, El-Zein, and da-Costa-Pereira 2022), (El Zein, Câmara, et al. 2023), and (Nasser et al. 2024), consider the development of user knowledge within the context of discrete single-shot searches within search sessions, without considering the interaction between turns, how the user’s understanding changes, or how system responses influence the user. In CIR, however, knowledge plays a central role. Each system response can change or expand what the user understands, which in turn affects how they write future queries. Despite this, existing CIR methods ignore the user’s knowledge entirely. No current CIR method tries to model or use the knowledge a user builds during the conversation to improve retrieval. This is a major gap in the literature, especially given the importance of knowledge in information seeking highlighted by the ASK (Nicholas J Belkin 1980), ISP (Kuhlthau 1991), and SAL (Vakkari 2016) frameworks. This gap motivates the second part of this thesis. We propose a new method for modeling and using the user’s changing knowledge in CIR, described in Chapter 5. This method introduces a Knowledge Extractor and a Knowledge Updater that work across turns to improve query formulation using the user’s accumulated knowledge.

2.4 Summary

In this chapter, we introduced the foundational components of IR, including document and query representation, and reviewed retrieval models such as the Boolean model, the vector space model, probabilistic and language models, as well as modern neural retrieval approaches. We also discussed re-ranking and generation techniques, along

with essential background on transformers, BERT, and ColBERT, which we will use in our investigation. In addition, we explored key concepts in conversational query understanding, such as query expansion and query rewriting, and highlighted the role of user knowledge in IR. In this PhD research, we focus specifically on advancing conversational query understanding and on exploiting user knowledge in CIR to enhance retrieval effectiveness. These aims inform the design of our research goals, research questions, and the methodologies developed in the following chapters.

Chapter 3

Experimental Methodology and Evaluation Framework

In this chapter, we describe the experimental evaluation framework used to conduct the investigation for this research. We begin by introducing information retrieval (IR) evaluation and its various components. This includes an overview of the features of an IR test collection consisting of document sets, query sets and relevance information for IR evaluation. We then describe IR evaluation measures to develop a deeper understanding of the evaluation process. Following this, we describe significance testing, a widely used statistical method for indicating the importance of IR methods.

Next, we introduce the datasets used in our experiments. We describe the document and query collections in these datasets and examine their key features and characteristics. Examples from these datasets are given to provide a better understanding of their structure and content.

Finally, we present the framework employed for our experimental investigation. We provide a brief overview of our framework by outlining the task formulation and the experimental setup designed to address our research questions. We also discuss implementation details to offer a clear understanding of the overall experimental framework.

3.1 Information Retrieval Evaluation

Information retrieval systems are designed to locate relevant information within a knowledge source to address a user’s information need. In the classical IR framework, as noted in Chapter 1, the user submits a query that expresses this need, and the retrieval system’s task is to identify and return the relevant items from the available information set.

Given this objective, various methods have been developed. In order to select the best approach for a retrieval task, it is necessary to measure and compare their effectiveness to determine which one performs best. The standard approach for evaluating the effectiveness of an IR system involves three key components:

- Document Collection – the set of documents or information, typical of the dataset to be searched.
- Query Collection – a set of queries representing users’ information needs, in the form that a user is expected to enter into an operational system.
- Relevance Judgments – assessments indicating documents in the collection judged to be relevant to each individual query in the Query Collection.

Together, these three components constitute what is known as a test collection. Test collections are essential for evaluating and benchmarking the performance of retrieval systems. Over the years, many standard test collections have been developed to enable comparison of retrieval methods between different researchers. Next, we give a review of the background of IR test collections and introduce some significant standardized test collections.

3.1.1 Standard Test Collections

Cranfield Test Collection Format

The standardized form of the IR test collection emerged from studies at Cranfield in the UK (Cleverdon 1991). The Cranfield test collection included 1,398 abstracts

from aerodynamics journal articles (*Document Collection*), a collection of 225 queries (*Query Collection*), and comprehensive relevance assessments for every query, document pair (*Relevance Judgment*). The Cranfield Collection established the general framework for the evaluation of IR systems, which became the foundational basis for IR test collections that have been developed thereafter. This collection framework allows the setup of a controlled lab-like experimental setup with a fixed document and query collection, as well as relevance judgments for evaluating retrieval systems, allowing for generalizable and reproducible evaluation of IR methods.

Even though the collection played an influential role in establishing a framework for evaluation, the small size of the Cranfield collection in terms of the document and query collections quickly made it insufficient for the evaluation of realistic IR systems. Following the development of the Cranfield IR evaluation paradigm, a number of researchers developed their own test collections, which sought to address some of the limitations of the original Cranfield collection by increasing the size of the document collection, focusing on different domains, and, with the increased power of computers and full document content.

The fact that these test collections were either private or informally shared meant that it was not possible to directly compare the performance of new IR methods as they were introduced. The desire to be able to directly compare IR methods led to the establishment of IR evaluation benchmarks. The first and best known of these is the Text Retrieval Conference (TREC) (Voorhees, D. K. Harman, et al. 2005).

Text Retrieval Conference (TREC)

Text Retrieval Conference (TREC) (Voorhees, D. K. Harman, et al. 2005) is an ongoing series of IR evaluation workshops largely based upon the Cranfield paradigm. Each edition of TREC offers a number of IR tasks for evaluation. The datasets in the test collection for these tasks are much larger than any previously developed test collections. The query sets are generally more complex and seek to be representative of real-world user information needs. Relevance judgments for TREC Collections are

carried out over a subset of documents rather than the whole document collection due to its size. The subset (or *pool*) of documents to be judged is determined by aggregating the top-ranked results from multiple retrieval systems submitted by researchers participating in each task. This technique, called ‘*pooling*’ (Spark-Jones 1975), allows for assessing the relevance of a subset of documents instead of the whole document collection.

For our investigation, we use the TREC collections developed for investigating conversational search. We provide details of the task dataset in Section 3.2. In the following section, we provide more details of the components of a typical TREC collection.

Document Collection The document collection is a set of information (items), where the information can be in any format (textual, code, images, videos, etc). This information set is grouped and structured as a set of marked-up documents, shown in Figure 3.1. Each document ‘<DOC>’ is identified by a unique identification ‘<DOCNO>’, and the contents of the document are in the ‘<TEXT>’ field.

A Single Document in the Document Collection

```
<DOC>
  <DOCNO> FT911-1 </DOCNO>
  <TEXT>
    Britain's John Smith delivered a passionate plea ...
  </TEXT>
</DOC>
```

Figure 3.1: TREC Document Collection

Query Collection The query collection is a set of descriptions of information need represented at TREC in a specific format shown in Figure 3.2. The information need is represented as a *topic* (‘<top>’). Each *topic* has the following fields.

- <num>: A unique identifier.

- `<title>`: A short statement of the information need in the form that the user might be expected to input into an information retrieval system.
- `<desc>`: A longer, more descriptive statement of the information need.
- `<narr>`: A detail of the features expected in a retrieved item relevant for the topic.

A Single Query in the Query Collection

```
<top>
  <num> Number: 051 </num>
  <title> Impact of ozone on health </title>
  <desc>
    Describe the health effects of exposure to ozone.
  </desc>
  <narr>
    Relevant documents will discuss either short- or long-
    term impacts of ozone exposure ...
  </narr>
</top>
```

Figure 3.2: TREC Query Collection

Relevance Judgments Relevance judgments are a list of document IDs relevant or non-relevant to the topic, as shown in Figure 3.3. The `<topic-id>` is the query identifier, `<doc-id>` is the document identifier of the document being judged, `<relevance>` is the human-labelled relevance of the document to the query, and `<iteration>` is a placeholder not used for evaluation. Assessments are typically carried out by trained assessors working in a controlled setting to ensure unbiased assessment. *Note: Items not manually judged for relevance to the topic, since they are not present in the assessment pool, are not included in this list for each topic.*

An Entry in the Relevance Judgment File

```
<topic-id> <iteration> <doc-id> <relevance>
```

Figure 3.3: TREC Relevance Judgement

3.1.2 Evaluation Metrics

The most commonly used IR evaluation metrics are *Precision* and *Recall* (Allen et al. 1955).

Precision

Precision is the fraction of retrieved documents that are relevant to the query.

$$Precision = \frac{\text{no. of retrieved documents that are relevant}}{\text{no. of retrieved documents}} \quad (3.1)$$

Recall

Recall is the fraction of relevant documents that are retrieved.

$$Recall = \frac{\text{no. of retrieved documents that are relevant}}{\text{no. of relevant documents}} \quad (3.2)$$

The effectiveness of IR methods is evaluated using a test collection with a range of evaluation metrics. In laboratory-based IR research, these metrics are used to measure the retrieval effectiveness of a selected test collection. A number of evaluation metrics are available which calculate the mean effectiveness over a query set.

Mean Average Precision (MAP)

MAP is a binary relevance-focused measure (Voorhees 1993). It considers the rank position of each relevant document. It is calculated as the mean of the average *Precision* across a collection of queries. (D. Harman 1994).

$$MAP = \frac{1}{Q} \sum_{q=1}^Q AP(q) \quad (3.3)$$

where:

- Q is the total number of queries or documents.

- $AP(q)$ is the Average Precision for query q , computed as:

$$AP(q) = \frac{1}{R_q} \sum_{k=1}^N P(k) \cdot \text{rel}(k) \quad (3.4)$$

where:

- N is the number of retrieved documents,
- R_q is the total number of relevant documents for query q ,
- $P(k)$ is the precision at rank k ,
- $\text{rel}(k)$ is an indicator function equal to 1 if the document at rank k is relevant and 0 otherwise.

Mean Reciprocal Rank (MRR)

Mean reciprocal rank is also a binary relevance-focused measure. It considers the rank position of the first relevant document in the retrieved document list. It is calculated based on *Reciprocal rank*, where *Reciprocal rank* is the inverse of the position of the first relevant item. MRR is the average reciprocal rank across a collection of queries. (Kantor and Voorhees 2000; Kantor and Voorhees 1996)

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \quad (3.5)$$

where:

- $|Q|$ is the total number of queries.
- rank_i is the rank of the first relevant document for query i .

Normalized Discounted Cumulative Gain (nDCG)

Normalized Discounted Cumulative Gain is a graded relevance-focused measure (Järvelin and Kekäläinen 2002). In graded relevance, systems are evaluated based on how effectively they retrieve and rank documents according to their degree of relevance (Kekäläinen 2005). It captures the notion that highly relevant

documents are more useful than marginally relevant documents. Also, the lower the ranked position of a relevant document, the less useful it is for the user. (Manning, Raghavan, and Schütze 2008)

$$\text{NDCG} = \frac{DCG}{IDCG} \quad (3.6)$$

where:

- DCG is the Discounted Cumulative Gain

$$DCG = \sum \frac{rel_i}{\log_2(i+1)}$$

where:

- rel_i is the relevance of the document at rank i .

- IDCG is the Ideal Discounted Cumulative Gain.

$$IDCG = \sum_{i=1}^n \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

where:

- n is the total number of relevant documents in the collection.
- rel_i is the relevance of the document at rank i .

Note: In this case, the relevance judgment must take the form of a graded relevance assessment indicating a degree of relevance at each assessed document.

These metrics were chosen to give a broad view of retrieval performance. MAP measures how well relevant documents are ranked across the entire list of retrieved results, making it useful for evaluating overall ranking quality throughout a conversation. MRR measures the position of the first relevant document, showing how quickly the system returns a useful result. This is especially important in conversational

search, where users expect relevant answers quickly. nDCG is well suited to the datasets used in this thesis (Section 3.2) because they include different levels of document relevance. It gives higher scores when the most relevant documents appear near the top of the results, reflecting what is most useful for users. Together, these three metrics provide a balanced evaluation of retrieval performance from different perspectives, making the results more reliable than relying on a single metric.

3.1.3 Significance Tests

A significance test is a method used to determine whether an observed difference in performance across IR systems is statistically significant and not merely due to chance on some measure of behaviour (Box, J. S. Hunter, and W. G. Hunter 2005). In this thesis, we employ significance testing to determine whether observed performance differences between the proposed approaches and baseline systems are statistically meaningful. Given the relatively small number of evaluated topics across evaluation benchmark datasets (Section 3.2), significance testing is used to assess the reliability and the validity of experimental findings and determine if the difference is meaningful.

There are a number of statistical tests that can be employed in IR evaluation. Which test is most appropriate to a particular task depends on various factors, such as the data size, data type, data distribution and independence of the observations. Furthermore, we also need to have clear null and alternative hypotheses. Depending on the results, we can reject the null hypothesis and consider the results statistically significant, suggesting that the observed effect is not just due to random chance. Interpreting significance testing results requires understanding the p-value and the chosen significance level. The p-value shows how likely it is that the observed results could have happened by chance if there were truly no effect. A small p-value means the results are unlikely to be due to chance. If the p-value is below the significance level α (commonly 0.05 or 0.01), the result is considered statistically significant.

Paired t -Test

For our experiments, we evaluate the performance of two different systems in a pair-wise manner. This is done by comparing the mean performance of both systems. Therefore, we use the paired t -test that looks at the means of two groups and how they are related. This test is particularly applicable when the same subjects (e.g., queries or users) are evaluated under two different conditions or systems, and the observations are thus dependent (Box, J. S. Hunter, and W. G. Hunter 2005).

Let X_i and Y_i represent the performance scores from System A and System B, respectively, for the i -th instance, where $i = 1, 2, \dots, n$. The difference between each paired observation is calculated as:

$$d_i = X_i - Y_i \quad (3.7)$$

The mean of these differences is given by:

$$\bar{d} = \frac{1}{n} \sum_{i=1}^n d_i \quad (3.8)$$

The standard deviation of the differences is computed as:

$$s_d = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (d_i - \bar{d})^2} \quad (3.9)$$

Using the mean and standard deviation, the t -statistic is calculated as:

$$t = \frac{\bar{d}}{s_d / \sqrt{n}} \quad (3.10)$$

This test statistic follows a t -distribution with $n - 1$ degrees of freedom under the null hypothesis.

The paired t -test is the most appropriate choice for our experimental setting because the same set of queries is evaluated under each compared system configuration, making the observations dependent rather than independent. This dependency is accounted for in the paired t -test by directly computing differences at the level

of individual queries. We adopt a significance level of $\alpha = 0.05$ throughout our experiments.

Hypothesis Testing The null hypothesis (H_0) assumes that there is no difference between the mean performances of the two systems, i.e., $\mu_d = 0$. The alternative hypothesis (H_1) posits that there is a statistically significant difference. A result is considered statistically significant if the computed p -value is less than the chosen significance level α . In such cases, the null hypothesis is rejected, indicating that the observed difference is unlikely to have occurred by chance.

In our experiments, we apply the paired t -test to compare the mean performance of two conversational systems across the same set of queries, ensuring that the comparison accounts for the dependency between paired observations.

3.2 Evaluation Test Collections

The test collections used for our experimental investigation are from the TREC Conversational Search tasks, Conversational Assistant Track (CAsT) (Jeffrey Dalton, Xiong, Kumar, et al. 2020; Jeffrey Dalton, Xiong, and Callan 2021) and the Interactive Knowledge Assistance Track (iKAT) (Aliannejadi, Abbasiantaeb, Chatterjee, Jeffrey Dalton, et al. 2024). These collections were selected for several reasons. First, they are the most widely adopted benchmarks in the CIR research community. Second, they collectively span a range of conversational complexity. CAsT 2019 provides a user-query only dependencies, CAsT 2020 introduces system-response dependencies, making it a more realistic representation of conversational search and iKAT 2023 further extends this by incorporating user-specific personas and multi-path dialogue structures, introducing a richer and more challenging evaluation setting. Together, these three collections allow us to evaluate the proposed approaches across progressively more complex conversational scenarios, strengthening the robustness of the experimental findings.

3.2.1 TREC CAsT Test Collections

Document Collection

MS-MARCO: MS-MARCO is a large-scale real-world machine reading and comprehension dataset consisting of passages extracted from web documents retrieved using the Bing search engine, real user search queries, and human-generated answers (Bajaj et al. 2016). The dataset contains 1,010,916 questions, 8,841,823 companion passages extracted from 3,563,535 web documents, and 182,669 editorially generated answers. The main feature of this dataset is that it is based on questions/queries that correspond to actual searches using a real-world search engine. The TREC CAsT tasks use only the passages in this dataset. There are various versions of this dataset available; TREC CAsT uses the passages made available in version 1 (Bajaj et al. 2016).

TREC CAR: The TREC CAR dataset was created as part of the 2017 Complex Answer Retrieval task in TREC (Dietz et al. 2017). The dataset consists of topics, outlines, and paragraphs extracted from English Wikipedia 2016. In order to extract the paragraphs, the Wikipedia articles are split into an outline of the sections and the paragraphs which they contain. Furthermore, all the paragraphs are de-duplicated, resulting in a paragraph corpus containing 29,678,367 unique paragraphs. TREC CAsT uses Version 2 of this paragraph corpus for search.

Topic Collection

CAsT 2019 Topics: This topic set was developed for the TREC 2019 Conversational Assistance Track (CAsT) (Jeffrey Dalton, Xiong, and Callan 2020). It contains statements of information need, also referred to as topics, that are general, exploratory, and inspired by topics from similar TREC datasets (Allan et al. 2017; Carterette et al. 2014). These information needs are intentionally complex, diverse, open-domain, and designed to be answerable. Each topic is structured as a sequence of conversational turns, where each turn is a natural language query mimicking how

Figure 3.4: TREC CAsT 2019 Topic 31

Title:	head and neck cancer
Description:	A person is trying to compare and contrast types of cancer in the throat, esophagus, and lungs
Turn	Utterances
1	What is throat cancer?
2	Is it treatable?
3	Tell me about lung cancer.
4	What are its symptoms?
5	Can it spread to the throat?
6	What causes throat cancer?
7	What is the first sign of it?
8	Is it the same as esophageal cancer?
9	What's the difference in their symptoms?

people ask questions in a conversation. Over the course of these turns, the queries aim to progressively represent the underlying complex information need. The goal is to retrieve a candidate response for each query, where each response is a brief, relevant passage. An example topic from the CAsT-19 is shown in Figure 3.4.

The topics are intentionally structured so that the complex information need cannot be fully addressed in a single interaction. Instead, multiple conversational turns are required to explore and satisfy the information need. The queries in each turn were manually crafted to reflect natural conversational dialogue, incorporating a range of linguistic and contextual features such as ellipsis, implicit references, contextual dependencies, topic shifts, deep dives into subtopics, broad exploration, and changes in focus. These characteristics often make individual turns difficult to interpret in isolation. Importantly, later turns were designed to depend only on previous user utterances, without taking system responses into account. The dataset consists of 50 test topics curated as an information-seeking dialogue with characteristics of a conversation. Relevance assessments are provided for 20 randomly selected test topics, and we utilize these 20 topics in our investigation.

CAsT 2020 Topics: This topic set was developed for the TREC 2020 Conversational Assistance Track (CAsT) (Jeffrey Dalton, Xiong, and Callan 2021). The dataset mainly builds upon the CAsT 2019 dataset, with a few differences. The CAsT 2019 topics follow a static conversational pattern, where at each subsequent turn in the conversation, the questions/queries issued by the user do not take into

Figure 3.5: TREC CAsT 2020 Topic 81

Title:	garage door repair and replacement
Description:	Information on how to replace and repair garage doors.
Turn	Utterances
1	How do you know when your garage door opener is going bad?
2	Now it's stopped working. Why?
3	How much does it cost for someone to fix it?
4	How about replacing it instead?
5	How do I choose a new one?
6	What does a smart one do?
7	What's important for me to know about their safety?
8	How could they be hacked?

consideration the system responses. In fact, the later turns are designed to depend only on previous user utterances, and system responses are disregarded.

CAsT 2020 queries address this limitation by allowing utterances to depend not only on previous user inputs but also on the system responses from earlier turns. This makes the conversational turns a more realistic representation of how search naturally unfolds over the course of a dialogue, enriching the contextual information available during the search process. Topics in this dataset were mined from real users' multi-turn information-seeking sessions on Bing (Rosset et al. 2020). These sessions were manually reviewed to ensure they follow meaningful trajectories and rewritten if necessary. The CAsT 2020 topics are then constructed from these curated sessions, resulting in natural language conversational topics. An example topic is shown in Figure 3.5. The dataset contains 25 evaluation topics and excludes the title and description fields in each query statement.

3.2.2 TREC iKAT Test Collection

Document Collection

ClueWeb22: This dataset is the newest in the ClueWeb datasets series (*The ClueWeb09 Dataset* n.d.; *The ClueWeb12 Dataset* n.d.). ClueWeb22 is a large-scale, real-world collection of spam-filtered, clean web data, designed to mimic the

distribution of web pages in a commercial search system. Clueweb22 contains 10 billion web pages capturing various aspects and information present within these pages (raw HTML, clean text, semantic annotations, rendered visual page and others). The dataset used in the TREC iKAT task is a subset of ClueWeb22 that falls in the B category of ClueWeb22, referred to as ClueWeb22-B. The documents in this collection were segmented into passages, resulting in 116,838,987 passages in the collection (Overwijk, Xiong, and Callan 2022).

Topic Collection

TREC iKAT 2023 topics: This topic set was developed for the 2023 TREC Interactive Knowledge Assistance Track (iKAT) (Aliannejadi, Abbasiantaeb, Chatterjee, Jeffery Dalton, et al. 2024). It builds upon previous years of the TREC Conversational Assistance Track (CAST), with the central motivation being that different users express their information needs in different ways. Furthermore, the relevance of information retrieved by a conversational system can vary depending on the user’s characteristics. These user differences mean that the flow of a conversation can evolve uniquely for each user, even when discussing the same topic. The goal of this task is to explore search over multi-path, multi-turn, and multi-perspective conversations, where the main challenge lies in understanding the user and their specific information needs, while delivering responses tailored to their individual characteristics. An example dialogue flow on the topic of “Finding a University” for two different users is shown in Figure 3.6.

The dataset consists of 36 conversational dialogues over 20 different topics, along with a Personal Text Knowledge Base (PTKB), which is a set of narrative sentences that describe the user. Topics can have multiple different conversational dialogues based on the user’s persona. The PTKB is assumed to be collected from previous conversations of the user with the system and has been created with specific guidelines to allow for a more diverse conversation. Each conversational dialogue consists of a number of user-system turns where the conversational flow

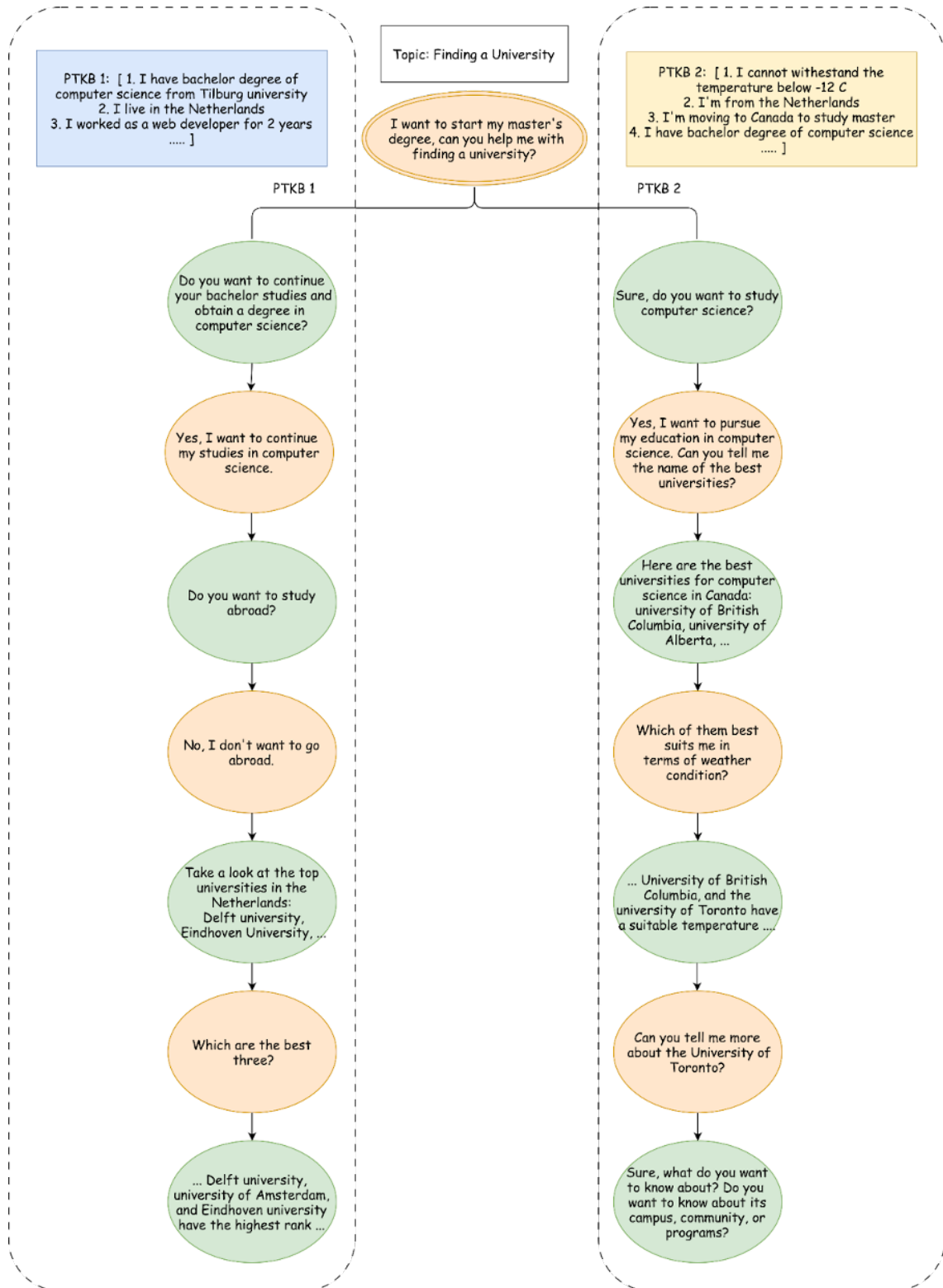


Figure 3.6: iKAT Flow (Aliannejadi, Abbasiantaeb, Chatterjee, Jeffery Dalton, et al. 2024).

is dependent upon the user’s PTKB and takes into consideration user-specifics, such as the user’s background, personal insights and contextual information, enabling a meaningful dialogue. A total of 344 turns with approximately 26,000 passages are provided with assessments of relevance. An example topic with two different users’ PTKB (User A and User B) is shown in Figure 3.7 and Figure 3.8.

Figure 3.7: TREC iKAT 2023 (User A)

Title:	Finding a diet
PTKB:	I don’t like the new spin-off; because I keep comparing the two and it has lower quality. Because of my kidney problem, I have to drink water frequently to stay hydrated. I’m going to change my phone. I can’t exercise too much because of the heart problem that I have. I’m vegetarian. I’m lactose intolerant. I’m allergic to soybeans. I just finished watching the Game of Thrones. I didn’t like how the series ended, especially the war scenes. I’m an Android user.
Turn	Utterances
1	Can you help me find a diet for myself?
2	Ok, good. Can you tell me what diet is the fastest way to lose some weight?
3	What about the DASH diet? I heard it is a healthy diet.
4	I prefer a natural diet, not a pill-based diet. Which of the aforementioned ones is natural?
5	Can you eat fish in any of them?
6	Thanks, but I also want the diet to be maintainable and not very hard to keep up.

However, the test collections have several limitations. First, the relevance judgements are created using a pooling method, which means that documents outside the assessment pool are treated as not relevant, even if they actually are. This can disadvantage new retrieval methods that find relevant documents not included in the pool, which is especially important for the knowledge-based query methods proposed in this thesis. Second, although the topics are designed to represent realistic information needs, they are created under controlled conditions and may not fully reflect the unpredictability and variety of real conversations. Third, the collections are static, meaning the relevance judgments do not change over time. As a result, they cannot capture how a user’s knowledge and information needs change throughout a conversation, which is a key focus of this thesis. An ideal evaluation framework would allow relevance to change as the user’s knowledge changes. Finally,

Figure 3.8: TREC iKAT 2023 (User B)

Title:	Finding a diet
PTKB:	I'm quite into cars. I'm having a social dinner with colleagues tonight. Traveling to Florida next week. I'm 155 cm tall. I have gained weight and now I'm 100 Kg. I have a knee injury. I had a big fight with my wife recently. I'm American. I'm a beer guy. I'm a 56 years old. I don't like surgical solutions.
Turn	Utterances
1	I think I am overweight, how can I measure obesity?
2	I have heard of BMI but what are the other two? Why do I need them?
3	Ok, I heard about BMI, what's a good BMI? What is mine?
4	Ok, I didn't expect it, now what can I do to lose some weight?
5	I can't exercise. So I think my only option is the first one?
6	First help me choose a diet. Later on, I may ask about the first option.
7	Ok I have a lot of options. I don't have diabetes and also I'm not into following celebrities. Hmm, which one is the quickest?
8	Interesting. Tell me more.
9	What is the difference between the two variations of the diet you mentioned?
10	Ok, good info. Now, tell me about the next.
11	No, I mean cardio.
12	Ok, I think for now I'll stick to dieting. By the way, can I eat meat in this diet since you explained a lot about veggies?

the collections contain a relatively small number of topics.

3.3 Framework for Experimental Investigation

In this section, we overview of the details of our experimental setup and the methodology used for our investigation. A brief overview of the proposed framework is shown in Figure 3.9. As described in Section 1.1, we propose a conversational context understanding approach that incorporates conversational history, as well as a knowledge history approach in CIR, with the goal of improved conversational context understanding during the search process.

3.3.1 Task Formulation

For each topic, a multi-turn conversation dialogue corresponding to a sequence of queries $Q = \{q_i\}_{i=1}^n$ is provided. The task goal is to retrieve relevant passages p^+

from a collection of documents D for the query q_n . The query statement at each turn can be ambiguous and is usually context-dependent. Thus, in order to carry out an effective search at each turn, we need to understand the query based on its conversation context C .

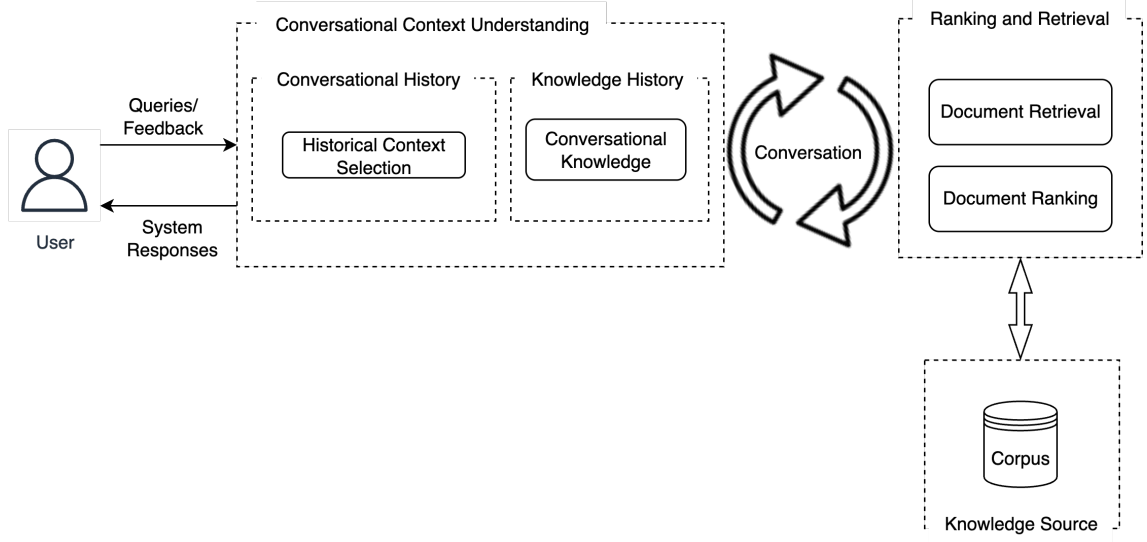


Figure 3.9: Overview of the Proposed Framework

3.3.2 Experimental Setup

We conduct our experiments based on the task formulation and the datasets as described in Section 3.2. We begin by indexing the document collection to be searched and subsequently apply queries to search the dataset. We design our experiments and implement existing approaches to conversational context understanding as described in Section 2.2.1, as well as our proposed approach to conversational history and knowledge history. We evaluate the performance of the retrieval systems using the metrics introduced in Section 3.1.2. All the experiments for our investigation are based on the Pyserini (J. Lin et al. 2021), a Python interface to the Anserini information retrieval toolkit (P. Yang, Fang, and J. Lin 2017).

Query

In the context of this research: *Query* is the utterance issued by the *User*, which is an attempt to describe an information need made by the user to satisfy the information

goal in a conversational setting. We use the queries provided in the datasets for each topic. We conduct our experiments by using historical queries, if available, as part of the conversational history of the conversation.

Retrieval Models

As described in Section 2.1.3, many retrieval models have been developed over the years for the effective retrieval of relevant lists of documents in textual information retrieval. We conduct our experiments employing BM25 (K. S. Jones, Walker, and Stephen E. Robertson 2000b) as the sparse retrieval baseline and ColBERT (Khattab and Zaharia 2020b) as the dense retrieval model. BM25 was selected as it is the most widely used and well-understood sparse retrieval baseline in IR research, providing a strong and reproducible reference point against which the proposed approaches can be compared. ColBERT was selected as the dense retrieval model because it can find documents based on meaning rather than exact keywords, while remaining efficient enough to handle the document collections used in this thesis. Together, BM25 and ColBERT represent the two main approaches used in modern information retrieval: sparse retrieval and dense retrieval. Using both models allows us to evaluate whether the proposed methods work across different retrieval approaches, rather than only with a single retrieval model.

Indexing

Since our primary focus is on textual information, indexing includes pre-processing of documents/corpus to aid in searching and retrieving in a conversational setting. For the different retrieval models described above, we have to process the document collection (corpus) differently.

For the BM25 model, we pre-process the document using the standard indexing pipeline of Pyserini. Text tokenization is done using a grammar-based tokenizer from the Unicode Text Segmentation algorithm (Davis and Iancu 2018). Punctuation marks, symbols and control characters are removed. The tokens are converted to

lowercase, no stemming or lemmatization, and no stopword removal is carried out. After the pre-processing step, an inverted index is created, which is then used to identify relevant documents when the user submits the query. We set the BM25 parameters, $k_1=0.9$ and $b=0.4$, which is the default configuration in Pyserini. The search queries go through the same pre-processing pipeline.

For Colbert model, we use the TCT-ColBERT-v2 model(*castorini/tct_colbert-v2-msmarco*) available in Pyserini where each document is represented by a set of contextualized token embeddings following the ColBERT late-interaction paradigm (Khattab and Zaharia 2020a). The underlying encoder is based on BERT-base (12 layers, hidden size 768). This contextualized embedding is stored using the FAISS library, which allows for an efficient similarity search for large-scale dense vectors (Johnson, Douze, and Jégou 2019). No stopword removal, stemming or lemmatization is carried out. The encoding is done using *castorini/tct_colbert-v2-msmarco* model, and a contextualized encoding of each document is created, which is then used to identify relevant documents when the user submits the query. The queries also go through the same pre-processing and encoding pipeline.

Evaluation Measures

To evaluate the retrieval effectiveness in our experiments, we use Mean Average Precision (MAP), Mean Reciprocal Rank(MRR) and Normalized Discounted Cumulative Gain (nDCG). We evaluate across all the metrics at a described threshold $@k$; where k denotes the rank threshold(cut-off). It only considers *top-k* documents in the retrieved documents list and ignores the rest.

3.4 Summary

In this chapter, we presented the experimental evaluation framework used to guide our investigation. We began with an overview of core concepts in information retrieval evaluation, including document and query collections. Various evaluation measures were introduced, with a focus on commonly used metrics used in our study. We also

introduced the concept of significance testing as a statistical method for comparing system performance. The datasets used in our experiments were described, along with their structure, characteristics, and illustrative examples to support deeper understanding. Finally, we outlined the experimental framework, including the task formulation, experimental setup, and implementation details. This framework forms the basis for the experiments conducted to address our research questions in the following chapters.

Chapter 4

Conversational Context

Understanding in CIR

In this chapter, we investigate historical conversational query context understanding in CIR. We begin by examining the effect of incorporating various types of conversational contextual information and the impact this has on the overall CIR system effectiveness. Exploiting information from historical conversational query context as contextual information in the current query provides a means to integrate relevant information from the dialogue, thereby better contextualizing the user’s current information need and enabling it to be more fully described.

We first explore several standard approaches for integrating historical context into CIR systems. Subsequently, we evaluate the performance of a CIR system with alternative approaches to incorporating contextual information into the current query.

Building on this investigation, we propose a similarity-based related historical context identification approach, which aims to identify and utilize relevant historical query turns to enhance CIR performance. We further explore a hybrid approach that combines existing heuristic methods with the proposed similarity-based approach. Finally, we examine a Large Language Model (LLM)-based method for historical query context identification and analyze its impact on CIR system performance.

4.1 Introduction

One particularly important aspect of query understanding at each turn in CIR is the history of the conversation at this point in the conversation. For example, the context of the query at each turn is essential to disambiguate and interpret it, where there are implicit and explicit references being made to previous turns. These might need to be taken into account in order to understand the information need and create a complete statement of the information need. As such, selecting the turns from the conversation which are relevant to the context of the current query is a vital component of the query construction process. Existing approaches to conversational history selection often rely on relatively simple strategies, such as including a fixed number of the directly preceding turns, without explicitly determining which turns are most relevant to the current query. It is reasonable to assume that the most relevant turn for any given query will be the immediately preceding or a few more previous turns. However, this may not necessarily be true in the case of a longer conversation with features such as topic shifts etc. In the next section, we investigate various historical conversational query context selection approaches and evaluate their effectiveness.

This investigation addresses research questions: **RQ1** and **RQ2**, introduced in Chapter 1 under **Hypothesis 1 (H1)**. Section 4.2 addresses RQ1 by evaluating how the quantity of historical context affects performance. Sections 4.3.4 addresses RQ2 by evaluating whether related turns can be selectively identified rather than exhaustively included, and Section 4.4 extends this investigation using large language models.

4.2 Investigating the Effect of Historical Conversational Context in CIR

Incorporating historical turns as contextual information enables relevant information from the conversation to be incorporated into the current query. To study the effect

of how incorporating historical turns can impact the effectiveness of a CIR system, we carried out an exploration of this topic for the TREC CAsT 2019, 2020 and TREC iKAT 2023 datasets. We experimented with the most widely used heuristic measure of selecting directly preceding k -previous query turns for historical query context incorporation to disambiguate the current query (Clarke 2019; Ohsugi et al. 2019; Mehri, Eric, and Hakkani-Tur 2020).

For our experiments, we use the queries as provided in the datasets without any preprocessing. We experimented with two variations of query reformulation: (1) expansion and (2) rewriting. We performed query expansion by selecting k -previous user queries to the current user query as the context in the order of appearance in the dialogue. We performed query rewriting using a T5 query rewriter (S.-C. Lin, J.-H. Yang, Nogueira, Tsai, C.-J. Wang, and J. Lin 2021) from *HuggingFace*¹ finetuned on the CANARD dataset (Elgohary, Peskov, and Boyd-Graber 2019). The different variations of historical conversational query context utilized in our experiments are detailed below.

4.2.1 Variations of Historical Conversational Query Context

1. $\mathcal{Q}_{raw}$: User-issued query at each turn in the conversation.

$$q = q_n$$

2. $\mathcal{Q}_{all}$: Concatenation of all available user-issued historical queries to the current turn in the conversation.

$$q = q_{1:n-1} \odot q_n$$

3. $\mathcal{Q}_{prev3}$: Concatenation of three previous user-issued historical queries in the conversation.

$$q = q_{n-3:n-1} \odot q_n$$

¹<https://huggingface.co/>

4. $\mathcal{Q}_{Turn1prev3}$: Concatenation of the first user-issued query and three previous user-issued historical queries in the conversation.

$$q = q_1 \odot q_{n-3:n-1} \odot q_n$$

5. $\mathcal{Q}_{manual}$: Contextualized and disambiguated user-issued query at each turn in the conversation that has been manually rewritten by the creator of the datasets.

$$q = \mathcal{QR}_{manual}(q_n)$$

6. $\mathcal{Q}_{Gold}$: Concatenation of user-issued queries from the turns that have been manually labelled to be related to the current turn in the conversation.

$$q = q_{1:n-1}^L \odot q_n, \quad L_n[j] = 1; j \in [1, n-1]$$

7. $\mathcal{Q}_{PRL}$: Concatenation of user-issued queries from the turns that have been labelled as related to the current turn (Relatedness labels L_n) (Mo, Nie, et al. 2023).

$$q = q_{1:n-1}^L \odot q_n, \quad L_n[j] = 1; j \in [1, n-1]$$

Note: For each of the variations of historical conversational query context except for $\mathcal{Q}_{manual}$ that we described above, we also have a variation for query rewriting (denoted by ‘+cqr’).

1. *+cqr*: We take the Historical Conversational Query Context as described above and then perform query rewriting using a T5 query rewriter (S.-C. Lin, J.-H. Yang, Nogueira, Tsai, C.-J. Wang, and J. Lin 2021) LLM, which is fine-tuned on the CANARD dataset (Elgohary, Peskov, and Boyd-Graber 2019)

$$q = \mathcal{QR}(q)$$

Table 4.1: Performance of Sparse and Dense retrieval systems. **Bold** and underline indicate the best result and the second best result (other than oracle Q_{Gold} and Q_{manual} approach)

	Query	CAsT 2019			CAsT 2020		
		nDCG@10	MRR	MAP@1000	nDCG@10	MRR	MAP@1000
Sparse	Q_{raw}	0.1479	0.3304	0.1163	0.0774	0.1787	0.0555
	Q_{all}	0.1645	0.4331	0.1040	0.0805	0.2156	0.0510
	+ cqr	0.3133	0.6324	0.2473	0.1726	0.3738	0.1268
	Q_{prev3}	0.1693	0.4004	0.111	0.0869	0.2268	0.0540
	+ cqr	0.2610	0.5381	0.2087	0.1444	0.3177	0.1075
	$Q_{Turn1prev3}$	0.1633	0.4016	0.1032	0.0949	0.2668	0.0568
	+ cqr	0.3041	0.6198	0.2377	0.1668	0.3651	0.1261
	Q_{PRL}	0.2271	0.5084	0.1691	0.1145	0.2767	0.0774
	+ cqr	0.2842	0.5545	0.2242	0.1413	0.3045	0.1043
	Q_{Gold}	0.2682	0.5518	0.2137	0.1380	0.3253	0.0982
	+ cqr	0.3151	0.6445	0.2525	0.1535	0.3450	0.1163
	Q_{manual}	-	-	-	0.2538	0.5422	0.1957
Dense	Q_{raw}	0.0853	0.2946	0.0143	0.0635	0.1924	0.0162
	Q_{all}	0.0859	0.2935	0.0186	0.0621	0.2042	0.0173
	+ cqr	0.1635	0.4979	0.0341	0.1226	0.3788	0.0340
	Q_{prev3}	0.1065	0.3397	0.0236	0.0672	0.2394	0.0191
	+ cqr	0.1454	0.4543	0.0297	0.1043	0.3172	0.0288
	$Q_{Turn1prev3}$	0.0852	0.2925	0.0195	0.0562	0.1918	0.0147
	+ cqr	<u>0.1592</u>	<u>0.4870</u>	<u>0.0330</u>	<u>0.1205</u>	<u>0.3726</u>	<u>0.0335</u>
	Q_{PRL}	0.1350	0.4156	0.0292	0.1013	0.3170	0.0297
	+ cqr	0.1409	0.4343	0.0289	0.1054	0.3133	0.0304
	Q_{Gold}	0.1661	0.5156	0.0368	0.1155	0.3570	0.0343
	+ cqr	0.1628	0.5087	0.0347	0.1204	0.3655	0.0339
	Q_{manual}	-	-	-	0.1806	0.5291	0.0558

4.2.2 Results

Table 4.1 and Table 4.2 show the performance of a CIR system using sparse (BM25) and dense retrieval (ColBERT) approaches for the CAsT and iKAT datasets, respectively. In general, sparse retrieval performs better than dense retrieval. As expected, Q_{raw} , which is the user-issued query, consistently yielded lower performance in both systems with the exception of the case of sparse retrieval for the iKAT dataset. As we vary the number of previous turns available to the current query turn, there is an increase in the system’s performance, indicating that the more context (available through previous historical query turns), the better the performance of the system

Table 4.2: Performance of Sparse and Dense retrieval systems. **Bold** and underline indicate the best result and the second best result (other than $\mathcal{Q}_{manual}$ approach.)

		iKAT 2023		
Query		nDCG@10	MRR	MAP@1000
Sparse	$\mathcal{Q}_{raw}$	0.1303	0.2828	0.0505
	$\mathcal{Q}_{all}$	0.0310	0.0987	0.0134
	+ cqr	0.1181	0.2501	<u>0.0470</u>
	$\mathcal{Q}_{prev3}$	<u>0.0336</u>	<u>0.0954</u>	<u>0.0163</u>
	+ cqr	0.1147	0.2387	0.0443
	$\mathcal{Q}_{Turn1prev3}$	0.0313	0.1120	0.0133
	+ cqr	<u>0.1189</u>	<u>0.2615</u>	0.0464
	$\mathcal{Q}_{manual}$	0.2888	0.5984	0.1423
Dense	$\mathcal{Q}_{raw}$	0.0273	0.0886	0.0059
	$\mathcal{Q}_{all}$	0.0173	0.0663	0.0049
	+ cqr	0.0515	0.1723	0.0116
	$\mathcal{Q}_{prev3}$	<u>0.0207</u>	<u>0.0683</u>	<u>0.0059</u>
	+ cqr	0.0478	0.1546	0.0105
	$\mathcal{Q}_{Turn1prev3}$	0.0197	0.0628	0.0064
	+ cqr	<u>0.0509</u>	<u>0.1666</u>	<u>0.0113</u>
	$\mathcal{Q}_{manual}$	0.0995	0.2891	0.0234

($\mathcal{Q}_{raw} < \mathcal{Q}_{prev3} < \mathcal{Q}_{Turn1prev3} < \mathcal{Q}_{all}$). The overall best performance is achieved by the ($\mathcal{Q}_{all} + \text{cqr}$) method for both sparse and dense retrieval in the CAsT datasets. In the case of the iKAT dataset, the best performance is achieved by $\mathcal{Q}_{raw}$ for sparse retrieval and ($\mathcal{Q}_{all} + \text{cqr}$) for dense retrieval. The second best approach for both datasets and for both sparse and dense retrieval was achieved by ($\mathcal{Q}_{Turn1prev3} + \text{cqr}$).

Selective incorporation of historical query turns, that are not a heuristic approach of selecting directly preceding k-turns, outperformed the query expansion-based reformulation strategy, as can be seen from the performance of $\mathcal{Q}_{PRL}$ and $\mathcal{Q}_{Gold}$. However, in the rewriting-based reformulation strategy, the $\mathcal{Q}_{all}$ variation outperformed all the other variations. The $\mathcal{Q}_{Gold}$ variation outperformed in both the query reformulation approach of expansion and rewriting in the CAsT 2019 dataset, whereas in the case of CAsT 2020 dataset it outperformed in the expansion strategy and performed comparably in the rewriting-based strategy. The superiority of the

$\mathcal{Q}_{manual}$ approach, in which queries are manually contextualized and disambiguated over the best performing approach of $(\mathcal{Q}_{all} + cqr)$ variation shows that being able to infer relevant information from the historical conversational context is important in order to improve the performance of a retrieval system.

Furthermore, the query rewriting strategy ($+cqr$) consistently outperformed the expansion-based query reformulation strategy in all the variations. Including all the available previous queries $(\mathcal{Q}_{all} + cqr)$ performed the best. In cases where we utilize directly preceding k-turns heuristically, the more context (incorporated through the inclusion of k-preceding turns), the better the performance of the system.

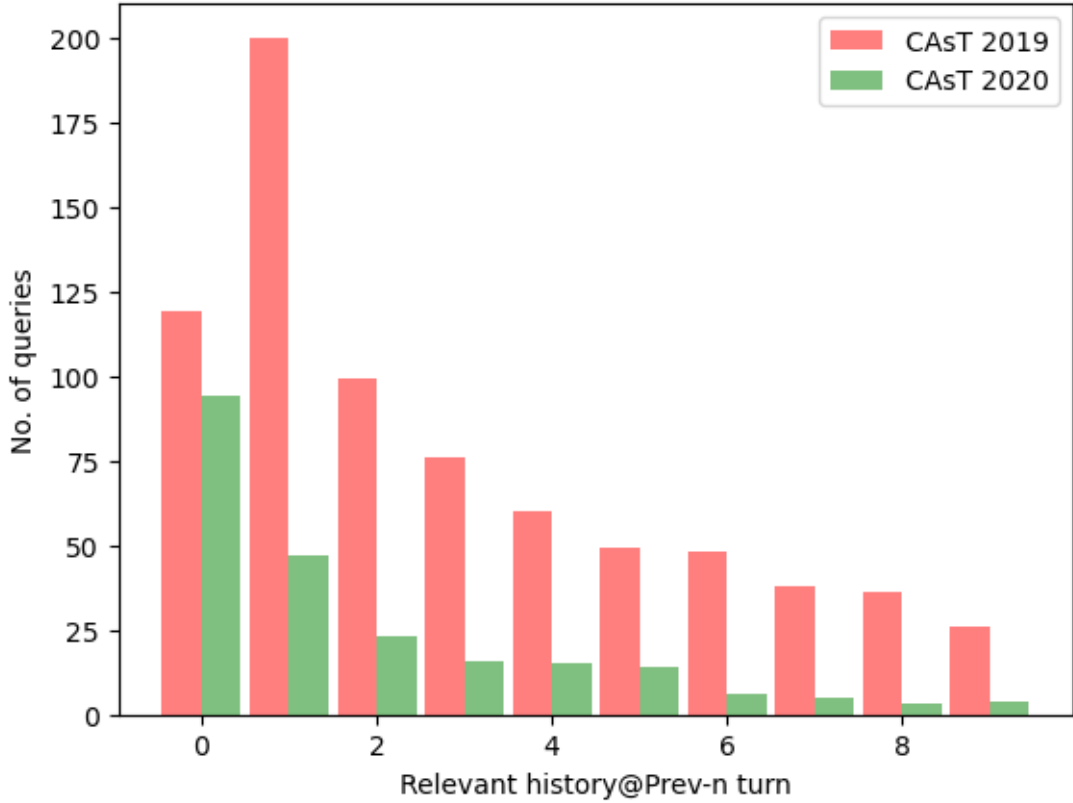


Figure 4.1: Relevant Query Turn Dependencies in the Dialogue History.

Further analysis of the CAsT datasets based on the turns that are manually labelled as related to the current turns (Aliannejadi, Chakraborty, et al. 2020), as seen in Figure 4.1, shows that for a large number of the queries, the most relevant turn is the immediately preceding turn, while for some queries, that is not the case and the relevant turns are earlier in the conversation history. This shows that relevant

turns can run deeper within the dialogue history.

Conversation Topic-Wise Analysis

In this section, we examine the retrieval effectiveness at a topic-level to highlight how different variations of historical conversational query context behave.

CAsT 2019. Figure 4.2 and Figure 4.3 present the topic-wise ndcg@10 performance for the selected variations of historical conversational query context under sparse (BM25) and dense (ColBERT) retrieval systems, respectively. As expected, the Q_{raw} approach consistently yielded the lowest performance across both systems. Although no clear pattern was evident across all topics, there were isolated cases where Q_{raw} slightly outperformed other approaches, such as Topic 54 in both retrieval settings.

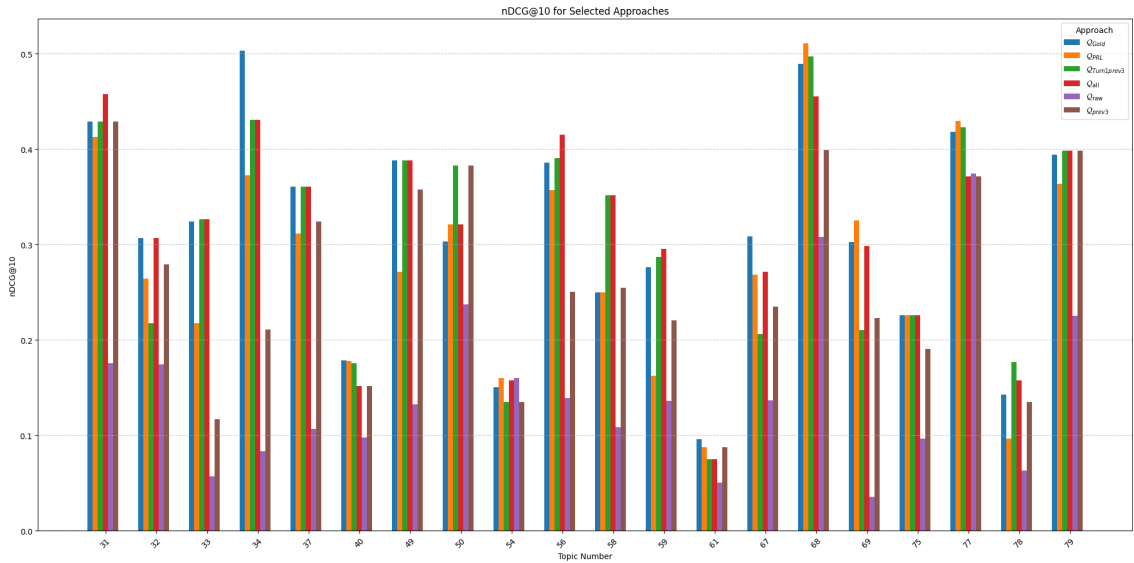


Figure 4.2: Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval.

Overall, Q_{all} achieved the highest performance on average. However, the $Q_{Turn1prev3}$ approach demonstrated comparable effectiveness, occasionally surpassing Q_{all} for certain topics e.g. Topics 40, 50, and 68 in sparse retrieval, and Topics 59 and 67 in dense retrieval. In contrast, Q_{all} consistently outperformed the Q_{prev3} approach across all topics. For Topic 54 in the dense retrieval system, performance remained relatively similar irrespective of the approach.

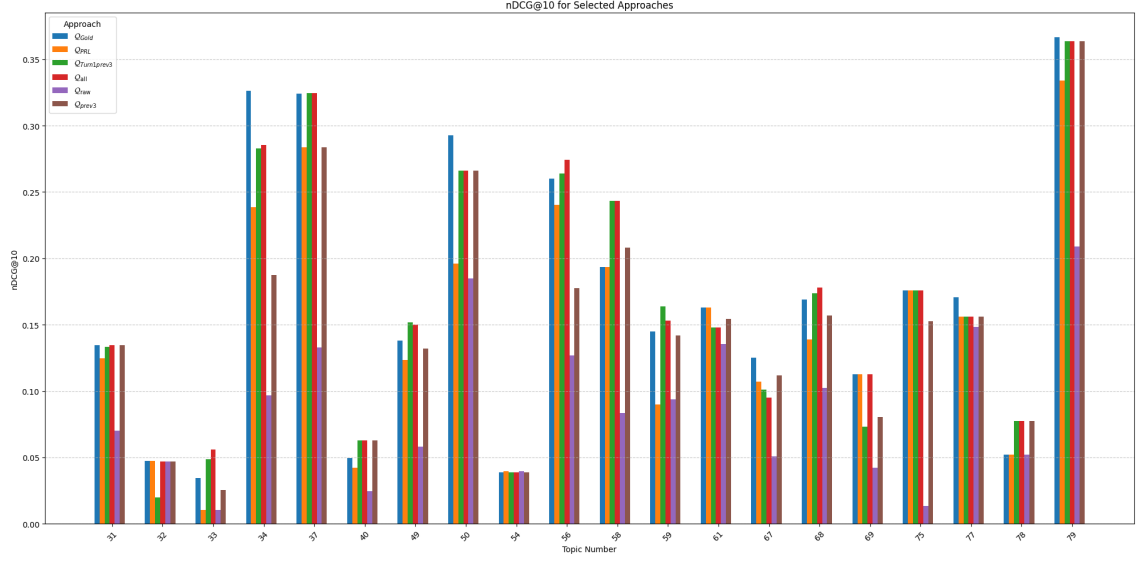


Figure 4.3: Topic-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval.

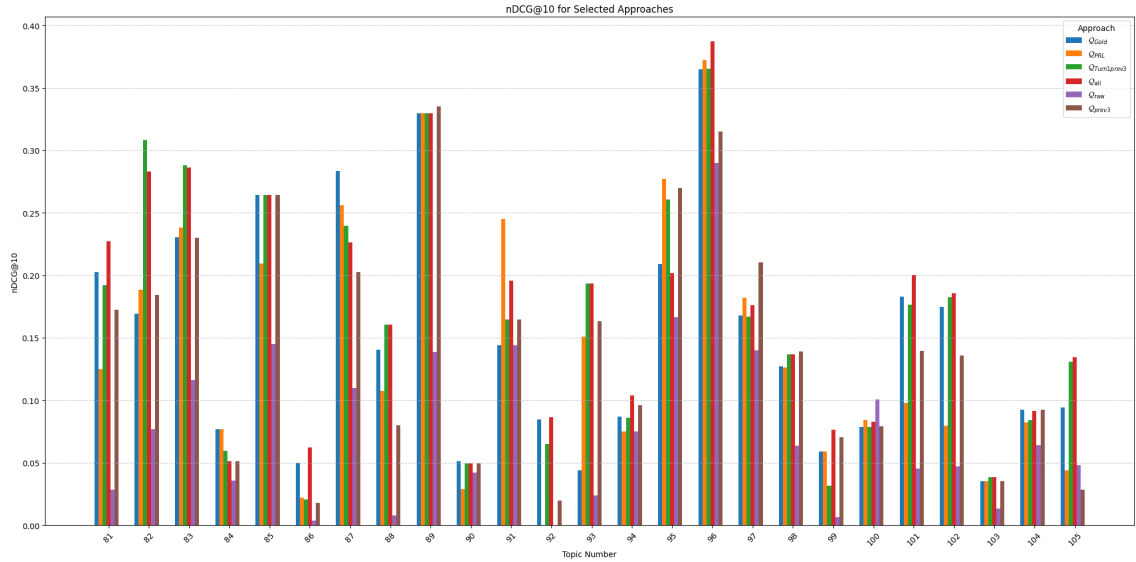


Figure 4.4: Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval.

CAsT 2020. Figure 4.4 and Figure 4.5 present the topic-wise ndcg@10 performance for the selected variations of historical conversational query context under sparse (BM25) and dense (ColBERT) retrieval systems, respectively. As expected, the Q_{raw} approach consistently yielded the lowest performance across both systems. No clear pattern was evident across all topics; there were isolated cases where Q_{raw} outperformed other approaches, such as Topic 100 in both retrieval settings.

Overall, Q_{all} achieved the highest performance on average. However, the $Q_{Turn1prev3}$ approach demonstrated comparable effectiveness, occasionally surpassing

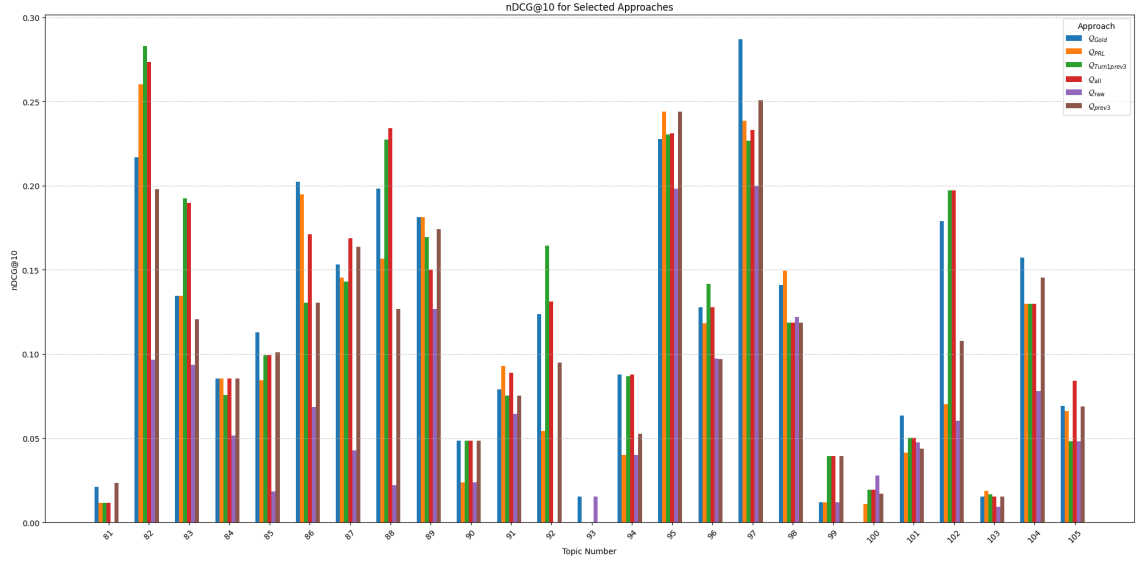


Figure 4.5: Topic-wise ndcg@10 Performance for CAsT 2020 Dense Retrieval.

Q_{all} for certain topics, e.g. Topics 82 and 87 in sparse retrieval, and Topics 82 and 92 in dense retrieval. In contrast, Q_{all} outperformed the Q_{prev3} approach across the majority of topics, with instances (Topics 89, 95 and 97) where it was better in both retrieval systems.

Topic 92 posed a significant challenge for both retrieval systems. When Q_{raw} queries were used for these topics, the systems were unable to retrieve any relevant documents. In contrast, using any other approach managed to retrieve some relevant documents. In the case of Topic 93 in dense retrieval, no approach was able to retrieve relevant documents, and in this case, using Q_{raw} allowed it to retrieve relevant documents at the top-10 rank.

iKAT 2023. Figure 4.6 and Figure 4.7 present the topic-wise ndcg@10 performance for the selected variations of historical conversational query context under sparse (BM25) and dense (ColBERT) retrieval systems, respectively. The Q_{raw} query formulation yielded better performance in the case of sparse retrieval and in certain cases for dense retrieval as well. No clear pattern was evident across all topics; there were isolated cases where Q_{raw} outperformed other approaches, such as Topic 13-1 and 19-1 in both retrieval settings. For Topic 17-3, Q_{raw} was unable to retrieve any relevant documents in either of the retrieval systems, whereas utilizing additional

context allowed for the retrieval of relevant documents.

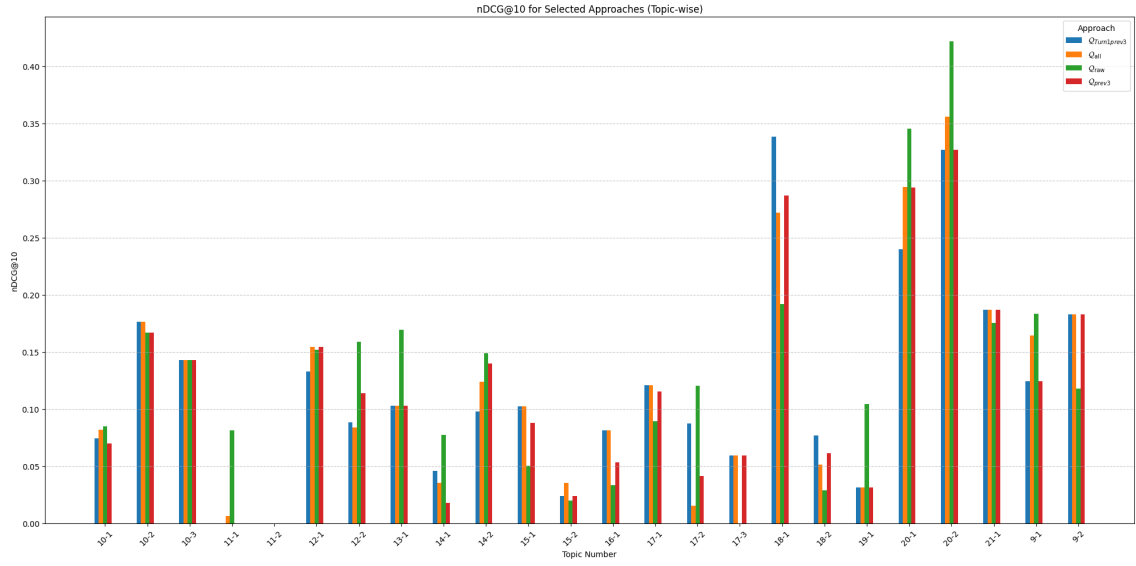


Figure 4.6: Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval.

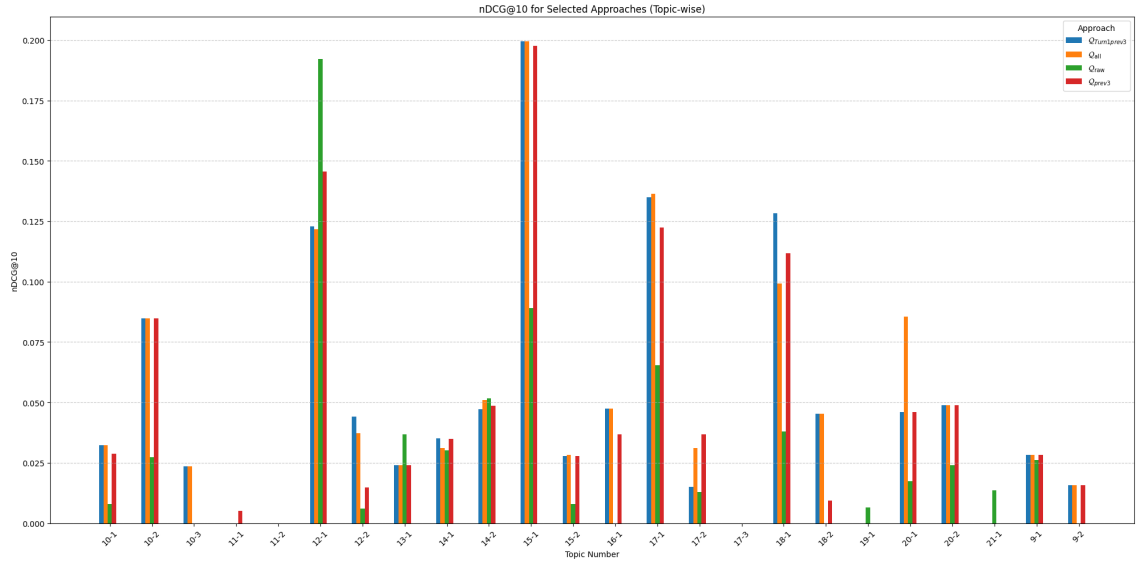


Figure 4.7: Topic-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval.

Overall, $\mathcal{Q}_{all}$ and $\mathcal{Q}_{Turn1prev3}$ approach performed similarly followed by the performance of the $\mathcal{Q}_{prev3}$ approach. Topic 11-2 posed a significant challenge for both retrieval systems. Irrespective of the query formulations, the systems were unable to retrieve any relevant documents. Similarly, Topic 11-1 was also challenging for both systems. In contrast, using any other approach managed to retrieve some relevant documents.

Conversation Depth-Wise Analysis

In this section, we examine retrieval effectiveness across different conversational depths to highlight how different variations of historical conversational query context behave.

CAsT 2019. Figure 4.8 and Figure 4.9 present the performance of sparse(BM25) and dense(ColBERT) retrieval systems across various conversation lengths (depth). These results reveal that the Q_{raw} consistently performs the worst across all conversation lengths. Furthermore, a noticeable decline in performance is observed as conversational depth increases for all retrieval approaches, with the exception of the sparse retrieval system. This trend suggests that, as conversations progress, each subsequent turn may either implicitly or explicitly be dependent on earlier turns. It shows that the ability to retrieve relevant documents diminishes as the conversation becomes longer, indicating the importance of identifying relevant previous turns in a conversation as a source of context to enrich the current query. Overall, all approaches exhibit a similar pattern of gradually decreasing performance as the conversation lengthens.

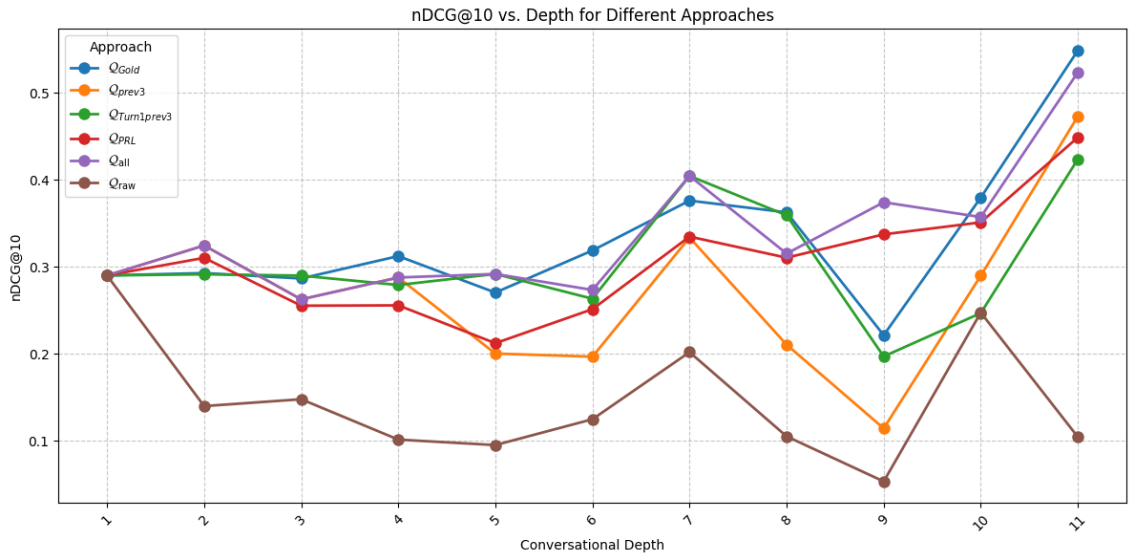


Figure 4.8: Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval.

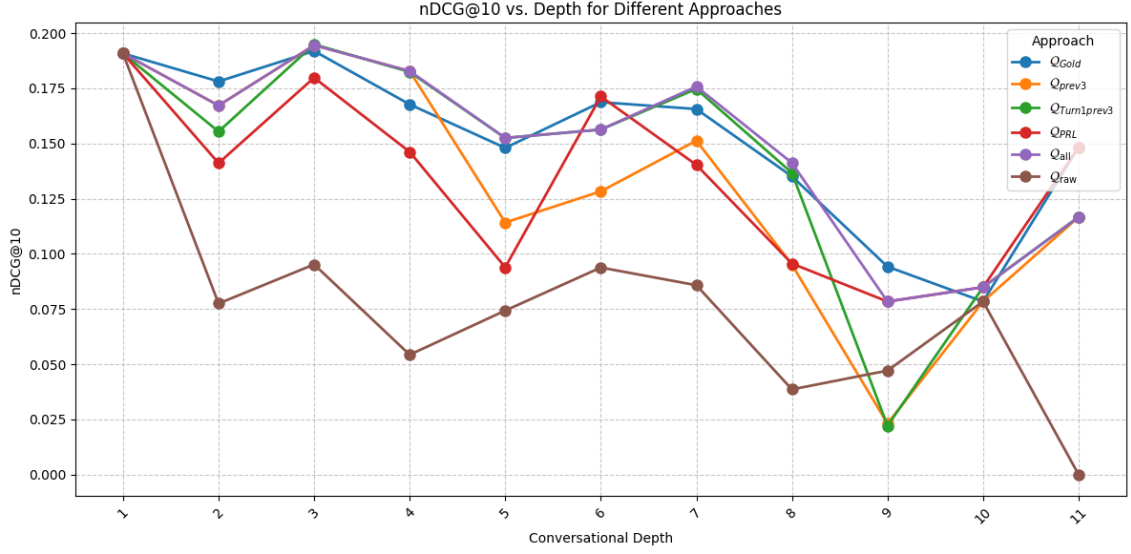


Figure 4.9: Depth-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval.

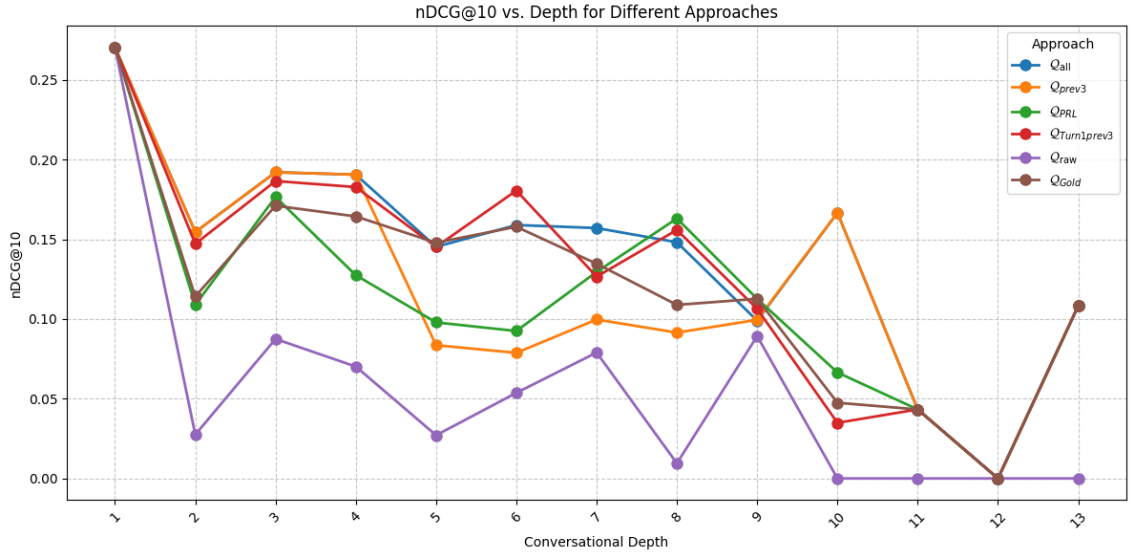


Figure 4.10: Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval.

CAsT 2020. Figure 4.10 and Figure 4.11 illustrate the performance of sparse and dense retrieval systems across varying conversation lengths (depths) in the datasets. The results show that Q_{raw} consistently performs the worst across all conversation lengths. Moreover, all query formulation approaches exhibit a noticeable decline in performance as conversational depth increases. Overall, the performance of all approaches gradually decreases as the conversation lengthens. A significant drop in performance is observed at depth 2 for both retrieval systems. Additionally, the performance gap between Q_{raw} and the other approaches is relatively large, whereas

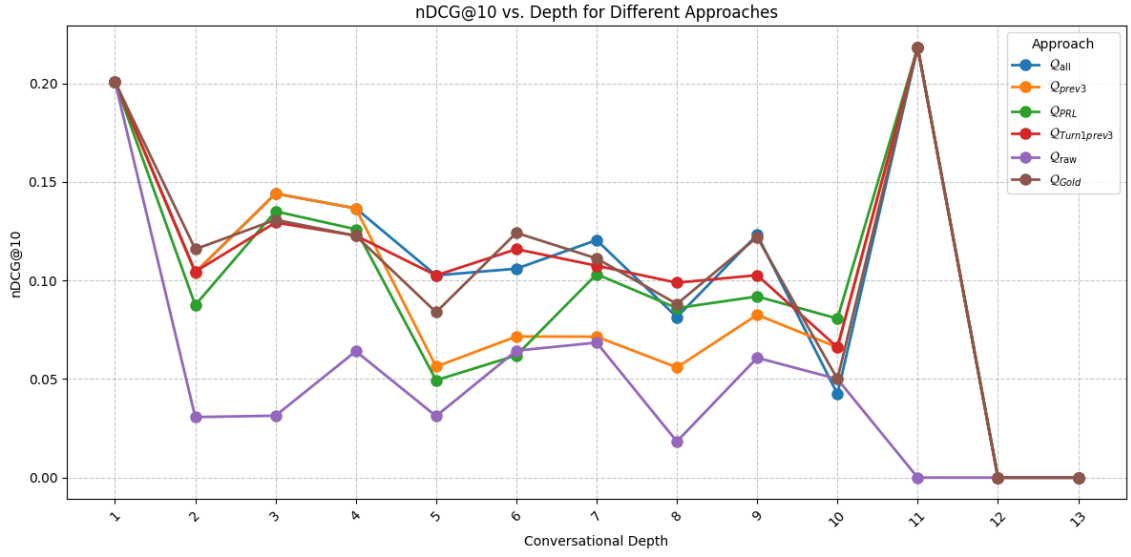


Figure 4.11: Depth-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval.

the differences among the remaining approaches are smaller.

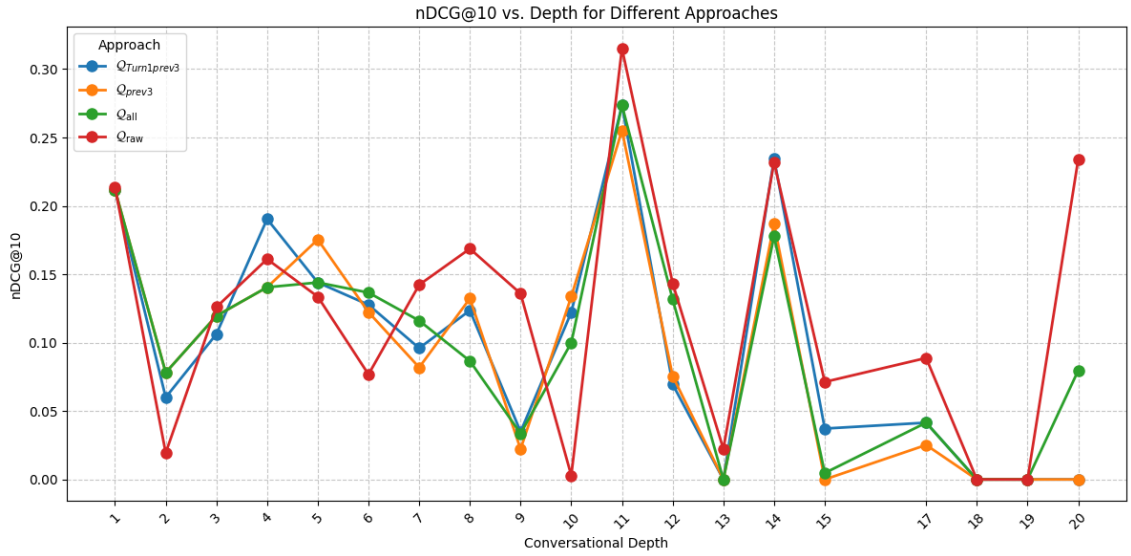


Figure 4.12: Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval.

iKAT 2023. Figure 4.12 and Figure 4.13 illustrate the performance of sparse and dense retrieval systems across varying conversation lengths (depths) in the dataset. The results show that Q_{raw} consistently performs the worst across all conversation lengths. Moreover, all query formulation approaches exhibit a noticeable decline in performance as conversational depth increases. Overall, the performance of all approaches gradually decreases as the conversation lengthens. A significant drop

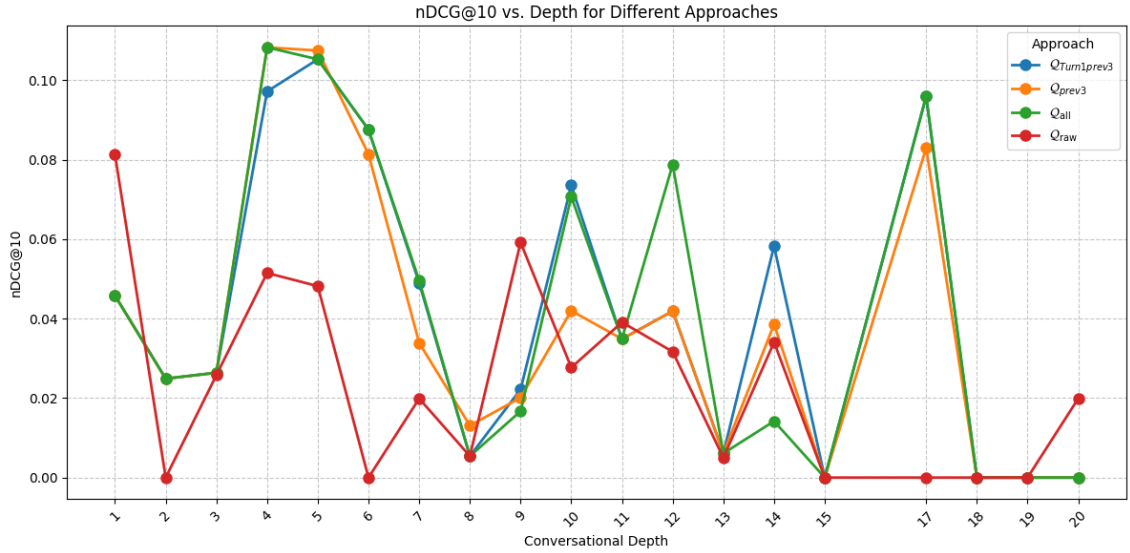


Figure 4.13: Depth-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval.

in performance is observed at depth 2 for both retrieval systems. Additionally, the performance gap between Q_{raw} and the other approaches is relatively large, whereas the differences among the remaining approaches are smaller.

4.2.3 Discussion

The purpose of these experiments was to investigate the effect of historical conversational query context. In these simple query creation methods, more context, in the form of previous turns, increased retrieval effectiveness. As we can see, adding additional context (queries) from the conversation history does not appear to improve performance for the expansion-based query reformulation strategy, whereas, for the rewriting-based approach, the more context, the better. Furthermore, not all previous turns in the history of the conversation are relevant to the current turn, as can be seen from the manually labelled related turns (Aliannejadi, Chakraborty, et al. 2020). Therefore, highlighting the need for a better approach to the selection of relevant turns to form the conversational context. Identifying relevant historical turns in a conversational scenario is non-trivial. It is important to distinguish and identify the relevant *vs* non-relevant turns for the current turn to be able to best interpret the current turn and to support the user in reaching their information goal.

These results answer **RQ1**: using previous conversation turns improves retrieval performance compared to using only the current raw query. However, an important finding is that the $\mathcal{Q}_{all}$ baseline performed very well. Even though it simply combines all previous conversation turns without selecting the most relevant ones, it achieved the best or nearly the best results, especially with query rewriting. This is a notable finding because it suggests that, at least for the conversation lengths present in the CAsT and iKAT datasets, including all previous turns is more helpful than harmful. The T5 rewriting model is able to reformulate a query with all the information, even when the input contains some irrelevant or repeated context. This finding is important for **RQ2**. Any method that selects only relevant history should not only identify the correct turns but also perform as well as or better than the strong $\mathcal{Q}_{all}$ baseline. It should also provide additional benefits, such as shorter queries, lower computational cost, and better handling of longer conversations where adding all previous turns may introduce too much noise. Therefore, the strong performance of $\mathcal{Q}_{all}$ provides a clear and challenging baseline that the proposed selective methods must improve upon.

In the following sections, we investigate some approaches to identify and select related historical conversational query context to the current query turn in the dialogue.

4.3 Similarity-based Related Historical Context Identification

In this section, we introduce our proposed method for identifying historical query context by evaluating the similarity between the current query and each previous query turn in the dialogue. Our specific objective, in terms of conversational context understanding, is to determine which historical query turns should be used to contextualize the current query, capturing only the most informative and non-redundant parts of the dialogue. To achieve this, we identify related historical queries

based on their semantic similarity to the current query and develop a selection model that uses these similarity scores to decide their contextual relevance.

We argue that a CIR system can leverage the similarity of historical turns to the current turn to determine which earlier turns should contribute to the contextualization of the current query, enabling the system to better understand the conversation and potentially improve retrieval. The underlying idea is that if a past query is extremely similar to the current query, it typically does not add new information and thus does not help in the contextualization of the current query. By filtering out such redundant queries, we can preserve a more diverse set of historical information for better conversational context understanding.

We use thresholds to categorize historical queries into different groups based on their similarity scores. Queries with similarity scores above a certain threshold are considered highly similar and are generally excluded from the context. Queries with similarity scores below a certain threshold are treated as distantly related and used for conversational context understanding.

4.3.1 Selection Model

Our method seeks to identify related historical turns to the current query turn based on the measures of:

1. *Similarity* between the previous query turns and the current query turn.
- **Similarity:** Measures how similar two turns are in the conversation. Each turn is first encoded into a vector representation in a shared semantic vector space using the *paraphrase-MiniLM-L6-v2* embedding model. We then compute pairwise *Cosine Similarity* between the embedding of the current turn and the embeddings of all historical turns to measure their semantic similarity.

The Similarity-Based Historical Query (SBQC) procedure for the selection of related historical query turns is described in Algorithm 1. For each historical query in the conversation, we compute a cosine similarity score with the current query. Based on

Algorithm 1: Similarity-Based Historical Query Turn Selection for Query Construction (SBQC)

Input: The n^{th} query turn q_n ; historical queries $h_{1:n-1}$ ($n > 1$), retriever f , similarity thresholds $\{\theta_1, \theta_2\}$

Output: The n^{th} turn relatedness labels L_n

```

1: Initial list of relatedness labels  $L_n = []$ 
2: for  $i$  from 1 to  $n - 1$  do
3:   Similarity  $Sim_{h_i} = g(q_n \circ h_i) \rightarrow$  cosine similarity score
4:   if  $\theta_1 < Sim_{h_i} < \theta_2$  then
5:     Add positive label to  $L_n$ 
6:   else
7:     Add negative label to  $L_n$ 
8:   end if
9: end for

```

this similarity score and a pre-defined threshold (θ_1, θ_2) , the historical query turn is determined to be related or unrelated to the current query turn. By iterating through all previous turns, the algorithm produces a list of relatedness labels that indicate which parts of the historical conversation query should be considered when constructing or expanding the current query. Figure 4.14 illustrates the procedure for the selection model and its subsequent use in query rewriting.

4.3.2 SBQC Based Historical Conversational Query Formulation

The previous turns in the search dialogue were identified as relevant to the construction of the current query statement based on our SBQC algorithm. The algorithm is dependent on choosing an appropriate cut-off threshold for the similarity measure. Determining an appropriate threshold for the cut-off is challenging. We create 3 different bins $\mathcal{B} = \{Low, Mid, High\}$ defined by their lower and upper bounds $[\alpha_1, \alpha_2, \alpha_3] = [0.33, 0.66, 1.0]$ to categorize previous queries, based on their similarity score with the current query.

Relatedness labels

$$L_n = \begin{cases} 1 & \text{if } \theta_1 \leq \text{SimilarityScore}(q_n, q_i) < \theta_2; i \in [1, n-1] \\ 0 & \text{otherwise.} \end{cases}$$

The motivation being that if a previous query turn is highly similar to the current query, then the former has no additional information that the latter turn does not already have, and hence the former query turn can be deemed to have no contextual value.

1. $\mathcal{SIM}_{low}$: User query utterances of turns labelled as related to the current turn from among the previous query turns (Relatedness labels L_n).

$$q = q_{1:n-1}^L \odot q_n \quad L_n[j] = 1$$

where,

$$L_n[j] = \begin{cases} 1 & \text{if } 0 \leq \text{Cosine similarity}(q_1 : q_{n-1}, q_n) < 0.33 \\ 0 & \text{otherwise} \end{cases} \quad \text{for } j \in [1, n-1]$$

2. $\mathcal{SIM}_{mid}$: User query utterances of turns labelled as related to the current turn from among the previous query turns (Relatedness labels L_n).

$$q = q_{1:n-1}^L \odot q_n \quad L_n[j] = 1$$

where,

$$L_n[j] = \begin{cases} 1 & \text{if } 0.33 \leq \text{Cosine similarity}(q_1 : q_{n-1}, q_n) < 0.66 \\ 0 & \text{otherwise} \end{cases} \quad \text{for } j \in [1, n-1]$$

3. $\mathcal{SIM}_{high}$: User query utterances of turns labelled as related to the current turn from among the previous query turns (Relatedness labels L_n).

$$q = q_{1:n-1}^L \odot q_n \quad L_n[j] = 1$$

where,

$$L_n[j] = \begin{cases} 1 & \text{if } 0.66 \leq \text{Cosine similarity}(q_1 : q_{n-1}, q_n) \leq 1.0 \\ 0 & \text{otherwise} \end{cases} \quad \text{for } j \in [1, n-1]$$

To explore the effectiveness of the similarity-based related historical context identification selection model and its effect on the retrieval performance, we compare it to the different variations of historical conversational query context as described in Section 4.2.1, which are listed below.

1. $\mathcal{Q}_{raw}$: User-issued query at each turn in the conversation.

$$q = q_n$$

2. $\mathcal{Q}_{all}$: Concatenation of all available user-issued historical queries to the current turn in the conversation.

$$q = q_{1:n-1} \odot q_n$$

3. $\mathcal{Q}_{manual}$: Contextualized and disambiguated user-issued query at each turn in the conversation that has been manually rewritten by the creator of the datasets.

$$q = \mathcal{QR}_{manual}(q_n)$$

4. $\mathcal{Q}_{Gold}$: Concatenation of user-issued queries from the turns that have been manually labelled to be related to the current turn in the conversation.

$$q = q_{1:n-1}^L \odot q_n, \quad L_n[j] = 1; j \in [1, n-1]$$

5. $\mathcal{Q}_{PRL}$: Concatenation of user-issued queries from the turns that have been labelled as related to the current turn (Relatedness labels L_n) (Mo, Nie, et al. 2023).

$$q = q_{1:n-1}^L \odot q_n, \quad L_n[j] = 1; j \in [1, n-1]$$

4.3.3 Hybrid Query Formulation

In our earlier experiments, we examined a heuristic approach of incorporating directly preceding k turns from the historical conversational query context. Furthermore, we proposed a similarity-based approach to identify related historical conversational query context. The heuristic approach captured the recent information in the dialogue, and the similarity-based approach attempts to capture information from earlier turns in the dialogue with the potential to contribute to the contextualization of the current query. We now combine both these approaches and propose a hybrid query formulation approach.

1. $\mathcal{SIM}_{low} \mathcal{Q}_{prev3}$: Concatenation of use-issued query utterances of turns labelled as related to the current turn using the similarity-based approach and three previous user-issued historical queries in the conversation.
2. $\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$: Concatenation of use-issued query utterances of turns labelled as related to the current turn using the similarity-based approach and the first user-issued query and three previous user-issued historical queries in the conversation.

4.3.4 Results

The retrieval performance of the proposed approach is shown in Table 4.3 and Table 4.4 for the CAsT and iKAT datasets, respectively. It can be observed that identifying all previous queries $\mathcal{Q}_{all}$ as related achieves the highest results and outperforms other approaches for most metrics similar to the results as seen in the previous experiments (Section 4.2.2), followed by the hybrid approach $\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$.

We also conducted experiments to determine if using only a similarity-based measure would be effective in identifying related historical queries. The similarity-based approach $\mathcal{SIM}_{low}$ performed better compared to the $\mathcal{Q}_{prev3}$ approach and comparably to the $\mathcal{Q}_{Turn1prev3}$ approach from the results as seen in the previous experiments (Section 4.2.2). This indicates that semantic similarity is a valuable

Table 4.3: Performance of Sparse and Dense retrieval systems. **Bold** and underline indicate the best result and the second best result (other than oracle Q_{Gold} and Q_{manual} approach). α and β denote no statistically significant and statistically significant improvements over ($Q_{all} + cqr$), respectively. (Paired t-test at $p < 0.05$)

	Query	CAsT 2019			CAsT 2020		
		nDCG@10	MRR	MAP@1000	nDCG@10	MRR	MAP@1000
Sparse	Q_{raw}	0.1479	0.3304	0.1163	0.0774	0.1787	0.0555
	Q_{all}	0.1645	0.4331	0.1040	0.0805	0.2156	0.0510
	+ cqr	0.3133	0.6324	0.2473	<u>0.1726</u>	<u>0.3738</u>	<u>0.1268</u>
	Q_{PRL}	0.2271	0.5084	0.1691	0.1145	0.2767	0.0774
	+ cqr	0.2842	0.5545	0.2242	0.1413	0.3045	0.1043
	SIM_{low}	0.1438	0.3721	0.0985	0.0758	0.1937	0.0484
	+ cqr	0.2793	0.5797	0.2241	0.1495	0.3271	0.1112
	$SIM_{low}Q_{prev3}$	0.1595	0.4073	0.1041	0.0867	0.2291	0.0539
	+ cqr	0.2951	0.6010	0.2348 α	0.1717 α	0.3724 α	0.1249 α
	$SIM_{low}Q_{Turn1prev3}$	0.1639	0.4285	0.1052	0.0883	0.2354	0.0542
	+ cqr	0.3105 α	<u>0.6240α</u>	<u>0.2417α</u>	0.1749α	0.3830α	0.1297β
	Q_{Gold}	0.2682	0.5518	0.2137	0.1380	0.3253	0.0982
	+ cqr	0.3151	0.6445	0.2525	0.1535	0.3450	0.1163
	Q_{manual}	-	-	-	0.2538	0.5422	0.1957
Dense	Q_{raw}	0.0853	0.2946	0.0143	0.0635	0.1924	0.0162
	Q_{all}	0.0859	0.2935	0.0186	0.0621	0.2042	0.0173
	+ cqr	0.1635	0.4979	0.0341	<u>0.1226</u>	<u>0.3788</u>	<u>0.0340</u>
	Q_{PRL}	0.1350	0.4156	0.0292	0.1013	0.3170	0.0297
	+ cqr	0.1409	0.4343	0.0289	0.1054	0.3133	0.0304
	SIM_{low}	0.0940	0.3192	0.0183	0.0588	0.2017	0.0160
	+ cqr	0.1507	0.4656 α	0.0307	0.1129 α	0.3508 α	0.0310 α
	$SIM_{low}Q_{prev3}$	0.0909	0.3108	0.0191	0.0617	0.2128	0.0178
	+ cqr	0.1527	0.4638	0.0318	0.1177 α	0.3675 α	0.0328 α
	$SIM_{low}Q_{Turn1prev3}$	0.0890	0.3004	0.0196	0.0627	0.2124	0.0179
	+ cqr	<u>0.1596α</u>	<u>0.4853α</u>	<u>0.0332α</u>	0.1267β	0.3874α	0.0354α
	Q_{Gold}	0.1661	0.5156	0.0368	0.1155	0.3570	0.0343
	+ cqr	0.1628	0.5087	0.0347	0.1204	0.3655	0.0339
	Q_{manual}	-	-	-	0.1806	0.5291	0.0558

signal for identifying related historical turns. In the case of the CAsT 2020 dataset, all query formulation approaches, regardless of their query selection method, underperformed compared to manually rewritten queries (Q_{manual}). This highlights the potential for improvement in developing more effective selection models for better query construction.

In the case of the iKAT dataset, the best performance was achieved by Q_{raw} for sparse retrieval and ($Q_{all} + cqr$) for dense retrieval. The second-best approach for

Table 4.4: Performance of Sparse and Dense retrieval systems. **Bold** and underline indicate the best result and the second best result (other than $\mathcal{Q}_{manual}$ approach). α and β denote no statistically significant and statistically significant improvements over $(\mathcal{Q}_{all} + \text{cqr})$, respectively. (Paired t-test at $p < 0.05$)

	Query	iKAT 2023		
		nDCG@10	MRR	MAP@1000
Sparse	$\mathcal{Q}_{raw}$	0.1303	0.2828	0.0505
	$\mathcal{Q}_{all}$	0.0310	0.0987	0.0134
	+ cqr	0.1181	0.2501	<u>0.0470</u>
	$\mathcal{SIM}_{low}$	0.0312	0.0983	0.0131
	+ cqr	0.1202 $^{\alpha}$	0.2496 $^{\alpha}$	0.0464 $^{\alpha}$
	$\mathcal{SIM}_{low} \mathcal{Q}_{prev3}$	0.0312	0.0966	0.0131
	+ cqr	0.1165 $^{\alpha}$	0.2469 $^{\alpha}$	0.0467 $^{\alpha}$
	$\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$	0.0305	0.1021	0.0231
	+ cqr	<u>0.1205</u> $^{\alpha}$	<u>0.2584</u> $^{\alpha}$	0.0465 $^{\alpha}$
	$\mathcal{Q}_{manual}$	0.2888	0.5984	0.1423
Dense	$\mathcal{Q}_{raw}$	0.0273	0.0886	0.0059
	$\mathcal{Q}_{all}$	0.0173	0.0663	0.0049
	+ cqr	0.0515	<u>0.1723</u>	0.0116
	$\mathcal{SIM}_{low}$	0.0206	0.0638	0.0056
	+ cqr	0.0473	0.1664 $^{\alpha}$	0.0114 $^{\alpha}$
	$\mathcal{SIM}_{low} \mathcal{Q}_{prev3}$	0.0179	0.0671	0.0052
	+ cqr	<u>0.0506</u> $^{\alpha}$	0.1743 $^{\alpha}$	0.0116 $^{\alpha}$
	$\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$	0.0173	0.0636	0.0050
	+ cqr	0.0515 $^{\alpha}$	0.1694 $^{\alpha}$	<u>0.0115</u> $^{\alpha}$
	$\mathcal{Q}_{manual}$	0.0995	0.2891	0.0234

sparse retrieval was achieved by the hybrid formulation($\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3} + \text{cqr}$), and in the case of dense retrieval, it was also achieved by the hybrid formulation ($\mathcal{SIM}_{low} \mathcal{Q}_{prev3} + \text{cqr}$).

It is important to note that neither the similarity-based method $\mathcal{SIM}_{low}$ nor the hybrid methods consistently outperform $\mathcal{Q}_{all}$, as shown in Table 4.3 and Table 4.4. This helps answer **RQ2**: selecting only relevant history does not improve retrieval effectiveness over using all previous turns on these datasets. This is likely because CAsT and iKAT contain short conversations, so including all history adds little extra

noise. Also, the T5 query rewriting model can handle some irrelevant information. However, the proposed method still has an important advantage. As shown in Section 4.3.4 (*Query Length vs Conversational Length*), it achieves similar retrieval effectiveness to $\mathcal{Q}_{all}$ while using much shorter queries. This shows that conversation history can be reduced without losing retrieval performance, making the approach more efficient, especially for longer conversations.

Conversational Topic-Wise Performance

In this section, we examine the retrieval effectiveness at a topic level to highlight how different variations of historical conversational query context behave.

CAsT 2019. Figure 4.15 and Figure 4.16 present the topic-wise performance of the sparse and dense retrieval systems, respectively. As seen in the results of the previous experiments (Section 4.2.2), the $\mathcal{Q}_{raw}$ approach consistently yielded the lowest performance across both systems. Although no clear pattern was evident across all topics, there were isolated cases where $\mathcal{Q}_{raw}$ slightly outperformed other approaches, such as Topic 54 in both retrieval settings.

Overall, the performance of $\mathcal{Q}_{all}$ and $\mathcal{SIM}_{low}\mathcal{Q}_{Turn1prev3}$ formulation was similar across majority of the topics in both the retrieval systems, which was then closely followed by $\mathcal{SIM}_{low}\mathcal{Q}_{prev3}$. The similarity and the hybrid approach outperformed the $\mathcal{Q}_{all}$ approach in Topics 67 and 78 in the case of a sparse retrieval system.

CAsT 2020. Figure 4.17 and Figure 4.18 present the topic-wise performance of the sparse and dense retrieval systems, respectively. As expected, the $\mathcal{Q}_{raw}$ approach consistently yielded the lowest performance across both systems. No clear pattern was evident across all topics. Overall, the performance of $\mathcal{Q}_{all}$ and $\mathcal{SIM}_{low}\mathcal{Q}_{Turn1prev3}$ formulation was similar across the majority of the topics in both retrieval systems. However, in the case of Topic 82, the similarity as well as the hybrid formulation outperformed $\mathcal{Q}_{all}$ for both retrieval systems and for Topics 92 and 104 in the case of dense retrieval.

Topic 92 posed a significant challenge for both retrieval systems. When $\mathcal{Q}_{raw}$ queries were used for these topics, the systems were unable to retrieve any relevant documents. In contrast, using any other approach managed to retrieve some relevant documents, and in the case of dense retrieval, the similarity as well as the hybrid formulation outperformed $\mathcal{Q}_{all}$. In the case of Topic 93 in dense retrieval, the $\mathcal{SIM}_{low}$ formulation was able to retrieve relevant documents where $\mathcal{Q}_{all}$ failed to retrieve relevant documents at the top-10 rank.

iKAT 2023. Figure 4.19 and Figure 4.20 present the topic-wise performance of the sparse and dense retrieval systems, respectively. The $\mathcal{Q}_{raw}$ query formulation yielded better performance in the case of sparse retrieval and in certain cases for dense retrieval as well. No clear pattern was evident across all topics; there were isolated cases where $\mathcal{Q}_{raw}$ outperformed other approaches, such as Topic 13-1 and 19-1 in both retrieval settings. For Topic 17-3, $\mathcal{Q}_{raw}$ was unable to retrieve any relevant documents in either of the retrieval systems, whereas utilizing additional context allowed for the retrieval of relevant documents only in the sparse system.

For Topics such as 14-2, 17-2 and 20-1, in sparse retrieval and Topics 15-1 and 17-2 in dense retrieval, the hybrid formulation outperformed all the other formulations. Overall, $\mathcal{Q}_{all}$ and $\mathcal{SIM}_{low}\mathcal{Q}_{Turn1prev3}$ approach performed similarly. Topic 11-2 posed a significant challenge for both retrieval systems. Irrespective of the query formulations, the systems were unable to retrieve any relevant documents. Similarly, Topic 11-1 was also challenging for both systems.

Conversational Depth-Wise Performance

In this section, we examine retrieval effectiveness across different conversational depths to highlight how different variations of historical conversational query context behave.

CAsT 2019. Figure 4.21 and Figure 4.22 illustrate the performance of sparse and dense retrieval systems across various conversation lengths (depth) in the

datasets. These results reveal that the $\mathcal{Q}_{raw}$ consistently performs the worst across all conversation lengths. Furthermore, a noticeable decline in performance is observed as conversational depth increases for all retrieval approaches, with the exception of the sparse retrieval system. The similarity-based approach as well as the hybrid approaches followed a similar trend as conversational depth increases, as in the case of other formulations. Overall, all approaches exhibit a similar pattern of gradually decreasing performance as conversation lengthens.

CAsT 2020. Figure 4.23 and Figure 4.24 illustrate the performance of sparse and dense retrieval systems across varying conversation lengths (depths) in the datasets. The results show that $\mathcal{Q}_{raw}$ consistently performs the worst across all conversation lengths. Moreover, all query formulation approaches exhibit a noticeable decline in performance as conversational depth increases. Overall, the performance of all approaches gradually decreases as the conversation lengthens. A significant drop in performance is observed at depth 2 for both retrieval systems. The similarity-based approach as well as the hybrid approach followed a similar trend as conversational depth increased, as in the case of other formulations. The hybrid approach $\mathcal{SIM}_{low}\mathcal{Q}_{Turn1prev3}$ performs similarly to the $\mathcal{Q}_{all}$ formulation, and at some depths outperforms (Depth 6, 8 and 9).

iKAT 2023. Figure 4.25 and Figure 4.26 illustrate the performance of sparse and dense retrieval systems across varying conversation lengths (depths) in the dataset. The results show that $\mathcal{Q}_{raw}$ performs worse across all conversation lengths in the case of dense retrieval. Moreover, all query formulation approaches exhibit a decline in performance as conversational depth increases, with the exception of later conversational depths. A significant drop in performance is observed at depth 2 for both retrieval systems, and subsequently the performance increases at depth 3 and 4. Overall, the similarity-based approach as well as the hybrid approach followed a similar trend as the conversational depth increased, as in the case of other formulations. The hybrid approach $\mathcal{SIM}_{low}\mathcal{Q}_{Turn1prev3}$ performs similarly to the

$\mathcal{Q}_{all}$ formulation, and at some depth outperforms (Depth 5, 10).

Identifying Related Historical Query Turns

Additionally, we evaluated the effectiveness of different approaches in correctly identifying related historical query turns with respect to the human-annotated *Gold* labels for the CAsT 2019 and CAsT 2020 datasets. The *Gold* label consists of manually annotated related historical query turns for a current turn (Aliannejadi, Chakraborty, et al. 2020; Jeffrey Dalton, Xiong, and Callan 2021). Table 4.5 presents the corresponding F1-scores and accuracy values for both CAsT 2019 and CAsT 2020. As seen in the earlier analysis, among the heuristics-based approaches, $\mathcal{Q}_{Turn1prev3}$ outperforms $\mathcal{Q}_{all}$ and $\mathcal{Q}_{prev3}$, demonstrating that leveraging both the first turn and recent turns provides more reliable contextual cues. The $\mathcal{Q}_{PRL}$ approach (Mo, Nie, et al. 2023), which selects turns based on usefulness, achieves the highest accuracy across both datasets and yields comparatively lower F1-scores on CAsT 2019, indicating that it identifies fewer related turns overall but with high precision.

In the case of the similarity-based approach, the results reveal an important pattern regarding threshold selection. The $\mathcal{SIM}_{low}$ approach produces higher F1-scores than $\mathcal{SIM}_{mid}$ and $\mathcal{SIM}_{high}$, suggesting that more relaxed similarity thresholds better capture semantically related but lexically varied turns, which are the characteristics of a conversational dialogue. While higher thresholds ($\mathcal{SIM}_{mid}$,

Table 4.5: Identification of Related Historical Query Context using Different Approaches

Approach	CAsT 2019		CAsT 2020	
	F1-Score	Accuracy	F1-Score	Accuracy
$\mathcal{Q}_{all}$	0.328	0.196	0.268	0.155
$\mathcal{Q}_{prev3}$	0.311	0.489	0.269	0.458
$\mathcal{Q}_{Turn1prev3}$	0.415	0.489	<u>0.337</u>	0.418
$\mathcal{Q}_{PRL}$	0.243	0.795	0.456	<u>0.831</u>
$\mathcal{SIM}_{low}$	0.313	0.344	0.248	0.274
$\mathcal{SIM}_{mid}$	0.218	0.671	0.211	0.736
$\mathcal{SIM}_{high}$	0.008	<u>0.787</u>	0.0	0.834
$\mathcal{SIM}_{low} \mathcal{Q}_{prev3}$	0.321	0.257	0.266	0.210
$\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$	<u>0.339</u>	0.248	0.278	0.201

especially $\mathcal{SIM}_{high}$) yield high accuracy, their extremely low F1-scores indicate that they identify very few related turns, missing much of the relevant context.

Furthermore, hybrid approach, $\mathcal{SIM}_{low} \mathcal{Q}_{prev3}$ and $\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$, shows slight improvements over using similarity alone but does not surpass the strongest heuristic-only approach $\mathcal{Q}_{Turn1prev3}$. This underscores the importance of incorporating the similarity measure in identifying related queries. The effectiveness of the $\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$ model in our experiments shows that the similarity measure in this context can be used as a complementary signal in identifying related historical query turns.

These classification results give a clearer answer to **RQ2** than the retrieval results because they compare the predicted related turns with human-labelled ground truth. The heuristic method $\mathcal{Q}_{Turn1prev3}$ achieved a higher F1-score than the similarity-based and hybrid methods. This shows that, for the CAsT datasets, the position of a turn in the conversation (the first turn and the most recent turns) is a better way to identify related history than semantic similarity alone. This agrees with the analysis in Section 4.2.2, where the previous turn was found to be the most relevant in most cases (Figure 4.1). The similarity signal is still useful because it improves results when combined with the heuristic approach ($\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$ performs better than $\mathcal{SIM}_{low}$ alone). This may become more useful in longer conversations, where the relevant turn is not always the first or the most recent.

Query Length vs Conversational Length

Examining how query length changes from one turn to the next provides insight into how different query formulation approaches handle growing conversational context. Our analysis of the average query length at each turn shows that, in general, query length increases linearly as the conversation progresses. This pattern is observed across various formulation approaches, with a notable exception for heuristic methods that utilize the k-previous turns, as well as the $\mathcal{Q}_{Gold}$ and $\mathcal{Q}_{PRL}$ approaches. We found that the query lengths produced by approaches that integrate similarity

signals are generally shorter compared to those derived from the $\mathcal{Q}_{all}$ method. The $\mathcal{Q}_{all}$ approach, which incorporates all prior query turns into the current query, often leads to unnecessarily lengthy queries. In contrast, the hybrid approaches, which balance between utilizing relevant previous turns and incorporating similarity metrics, effectively preserve search quality while minimizing excessive query expansion. Figure 4.27, Figure 4.28 and Figure 4.29 illustrate these findings, highlighting the differences in query lengths among the various methods.

4.4 LLMs-based Related Historical Context Identification

In the experiments described in the previous sections, we investigated various approaches to incorporating historical conversational query context in a conversational dialogue and evaluated its effectiveness in retrieval performance as described in Section 4.3 and Section 4.3.3. Furthermore, we investigated the effectiveness of these approaches in correctly identifying related historical query turns as described in Section 4.3.4. We now turn to the use of large language models (LLMs) to identify the related historical turns for the current query. In this investigation, our goal is to evaluate the effectiveness of utilizing LLMs to identify the related historical turns and compare their performance against existing heuristic, similarity-based, and hybrid approaches.

We treat the identification of related historical query turns as a classification problem in which the LLM determines whether a historical turn is related to the current turn in a conversational dialogue. To evaluate the performance of the LLM at identifying the related historical query turn, we provide the LLM with the historical query as well as the current query and instruct the LLM to classify the historical query turn. For the instruction given to the LLM, we employ a zero-shot, one-shot, and few-shot prompting strategy. In the zero-shot strategy, the LLM is given only the task description (instruction) and the query turns, with no labelled

examples. In the one-shot strategy, we provide one labelled example illustrating a related and an unrelated turn. In the few-shot strategy, we provide more than one labelled example demonstrating various types of conversational relationships (related/unrelated) between the query turns. We provide the prompt details in the Appendix A. The zero-shot strategy assesses the LLM’s inherent ability to identify conversational dependency between turns. The one-shot strategy assesses the LLM’s performance when given a single concrete example of the task to guide the LLM. In the few-shot strategy, multiple curated examples are provided to illustrate a range of conversational dependencies between turns, to allow the model to generalize and better capture the conversational dependencies between turns.

4.4.1 Results & Discussion

Table 4.6 presents the performance of existing heuristic, similarity-based, hybrid approaches for identifying related historical query context on the CAsT 2019 and CAsT 2020 datasets, along with the performance of an LLM. The table reports F1-scores and accuracy for heuristic approaches and similarity-based approaches from Table 4.5, and also for the large language models (LLMs) under zero-shot, one-shot, and few-shot settings.

Among the heuristic-based methods, $\mathcal{Q}_{Turn1prev3}$ consistently outperformed $\mathcal{Q}_{all}$ and $\mathcal{Q}_{prev3}$ in F1-score, achieving 0.415 on CAsT 2019 and 0.337 on CAsT 2020. This indicates that incorporating the first turn of the conversation along with the previous three queries is a good heuristic to identify related historical query turns. Interestingly, $\mathcal{Q}_{PRL}$ exhibited the highest accuracy across both datasets (0.795 for CAsT 2019 and 0.831 for CAsT 2020), but a lower F1-score on CAsT 2019 (0.243). This suggests that $\mathcal{Q}_{PRL}$ can correctly identify many historical turns, it often misses actual relevant instances, resulting in low recall and a reduced F1 score.

In the case of similarity-based approaches, $\mathcal{SIM}_{low}$ achieved moderate performance, with $\mathcal{SIM}_{low}\mathcal{Q}_{Turn1prev3}$ slightly improving F1-scores over $\mathcal{SIM}_{low}$. However, these methods generally underperform compared to the heuristic approaches de-

Table 4.6: Identification of Related Historical Query Context using Different Approaches

Approach	CAsT 2019		CAsT 2020	
	F1-Score	Accuracy	F1-Score	Accuracy
$\mathcal{Q}_{all}$	0.328	0.196	0.268	0.155
$\mathcal{Q}_{prev3}$	0.311	0.489	0.269	0.458
$\mathcal{Q}_{Turn1prev3}$	0.415	0.489	0.337	0.418
$\mathcal{Q}_{PRL}$	0.243	0.795	0.456	0.831
$\mathcal{SIM}_{low}$	0.313	0.344	0.248	0.274
$\mathcal{SIM}_{low} \mathcal{Q}_{prev3}$	0.321	0.257	0.266	0.210
$\mathcal{SIM}_{low} \mathcal{Q}_{Turn1prev3}$	0.339	0.248	0.278	0.201
LLMs				
<i>zero-shot</i>				
Phi-3-mini-4k-instruct	0.3296	0.6809	0.3165	0.7459
Phi-3-medium-4k-instruct	0.2618	0.6708	0.2448	0.7023
Llama-3.1-8B-Instruct	0.5472	0.7727	0.4436	0.8200
Llama-3.2-1B-Instruct	0.2810	0.6181	0.1933	0.6565
<i>one-shot</i>				
Phi-3-mini-4k-instruct	0.3245	0.4722	0.2780	0.4623
Phi-3-medium-4k-instruct	0.2942	0.6120	0.2977	0.5670
Llama-3.1-8B-Instruct	0.4569	0.6144	0.3707	0.6565
Llama-3.2-1B-Instruct	0.2600	0.6813	0.2294	0.6918
<i>few-shot</i>				
Phi-3-mini-4k-instruct	0.3488	0.6641	0.2792	0.7023
Phi-3-medium-4k-instruct	0.0927	0.7751	0.1053	0.8200
Llama-3.1-8B-Instruct	0.3814	0.3847	0.3282	0.4365
Llama-3.2-1B-Instruct	0.3289	0.1976	0.2691	0.1565

scribed earlier in both F1-score and accuracy, indicating that similarity measures may not capture the nuanced conversational dependencies present in historical query contexts.

LLMs demonstrated strong performance across both datasets. Notably, Llama-3.1-8B-Instruct achieved the highest F1-scores in the zero-shot setting (0.5472 on CAsT 2019 and 0.4436 on CAsT 2020), outperforming all heuristic-based and similarity-based methods described earlier. This indicates the LLM’s robust ability to understand context and identify related historical queries without explicit training on task-specific examples.

All LLMs performed slightly worse on CAsT 2020 than on CAsT 2019, especially in F1-score. Overall, Llama-3.1-8B gave the best results for identifying related conversation turns, providing the strongest answer to **RQ2**. It shows that LLMs are the most accurate method tested in this thesis. The next best method is the heuristic approach $\mathcal{Q}_{Turn1prev3}$. However, better classification did not lead to better retrieval. As shown in Table 4.7, the hybrid method achieved the best retrieval results across all datasets. This suggests that removing some turns may also remove useful context for retrieval. Therefore, the hybrid method is the most practical approach because it gives the best retrieval performance while being much simpler, faster, and using shorter queries than both the LLM and the $\mathcal{Q}_{all}$ baseline.

4.4.2 Retrieval Performance Based on LLM Identification

Table 4.7 shows the retrieval performance of a sparse(BM25) and a dense (ColBERT) retrieval system based on the LLM-identified historical conversational query context, as well as the similarity-based approach described in Section 4.3 and the hybrid approach described in Section 4.3.3. For sparse retrieval, the hybrid approach discussed in Section 4.3.3 consistently performs best across all datasets. For example, on CAsT 2019, the hybrid approach achieves the highest nDCG@10 (0.3105), improving over both the LLM and similarity-based approaches. A similar pattern appears in CAsT 2020 and iKAT 2023 datasets, where the hybrid approach performs the best with

Table 4.7: Performance of Sparse and Dense retrieval systems using different approaches to identify historical conversational query context

		Approach	nDCG@10	MRR	MAP@1000
Sparse	CAsT 2019	LLM	0.2780	0.5679	0.2190
		Similarity	0.2793	0.5797	0.2241
		Hybrid	<i>0.3105</i>	<i>0.6240</i>	<i>0.2417</i>
	CAsT 2020	LLM	0.1280	0.2809	0.0995
		Similarity	0.1495	0.3271	0.1112
		Hybrid	<i>0.1749</i>	<i>0.3830</i>	<i>0.1297</i>
	iKAT 2023	LLM	0.1115	0.2344	0.0439
		Similarity	0.1202	0.2496	0.0464
		Hybrid	<i>0.1205</i>	<i>0.2584</i>	<i>0.0465</i>
Dense	CAsT 2019	LLM	0.1505	0.4695	0.0307
		Similarity	0.1507	0.4656	0.0307
		Hybrid	<i>0.1596</i>	<i>0.4853</i>	<i>0.0332</i>
	CAsT 2020	LLM	0.0997	0.2989	0.0279
		Similarity	0.1129	0.3508	0.0310
		Hybrid	<i>0.1267</i>	<i>0.3874</i>	<i>0.0354</i>
	iKAT 2023	LLM	0.0380	0.1208	0.0085
		Similarity	0.0473	0.1664	0.0114
		Hybrid	<i>0.0515</i>	<i>0.1694</i>	<i>0.0115</i>

nDCG@10 (0.1749) and nDCG@10 (0.1205), respectively. In dense retrieval, there is a similar trend. The hybrid approach provides slight improvements over the LLM and similarity-based approach across all datasets. Overall, the hybrid approach is the most effective approach for identifying historical conversational query context in order to disambiguate and contextualize the current turn in CIR, improving retrieval performance in both sparse and dense systems and offering the most stable performance across datasets.

4.5 Summary

In this chapter, we investigated conversational context understanding in CIR and answered **RQ1** and **RQ2** under **Hypothesis 1**. We begin by examining the effect of incorporating various types of conversational contextual information. We first

explore several standard approaches for integrating historical context into CIR systems. Subsequently, we evaluate the performance of a CIR system under different methods of incorporating contextual information to understand their respective effects on retrieval effectiveness. Building on this investigation, we propose a similarity-based related historical context identification approach, which aims to identify and utilize relevant historical turns. We further explore a hybrid approach that combines existing heuristic methods with the proposed similarity-based approach, assessing its effectiveness in improving retrieval outcomes. Finally, we examine a Large Language Model (LLM) based related historical context identification approach and analyze its potential for improving CIR system performance. For **RQ1**, the results show that using conversation history improves retrieval compared to using only the current query. The $\mathcal{Q}_{all}$ method gave a very strong baseline, especially with query rewriting. For **RQ2**, the results show that LLMs are the most accurate at finding related conversation turns, followed by the simple $\mathcal{Q}_{Turn1prev3}$ heuristic. However, better identification did not always lead to better retrieval. The proposed hybrid method gave the best retrieval results across all datasets and retrieval models while using much shorter queries and less computation than the LLM and $\mathcal{Q}_{all}$. These results support **Hypothesis 1** and show that identifying related history and improving retrieval are not always the same.

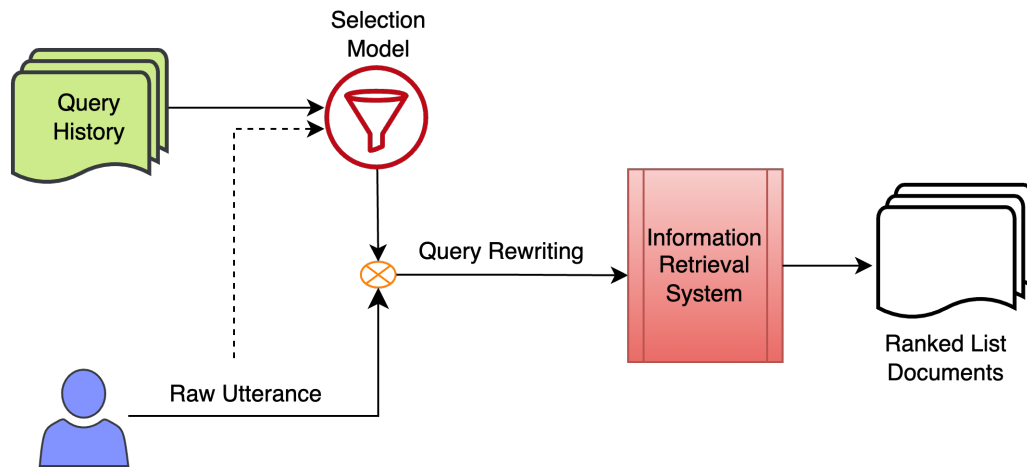


Figure 4.14: Illustration of the *Selection model* for related historical query turn identification for Query Formulation.

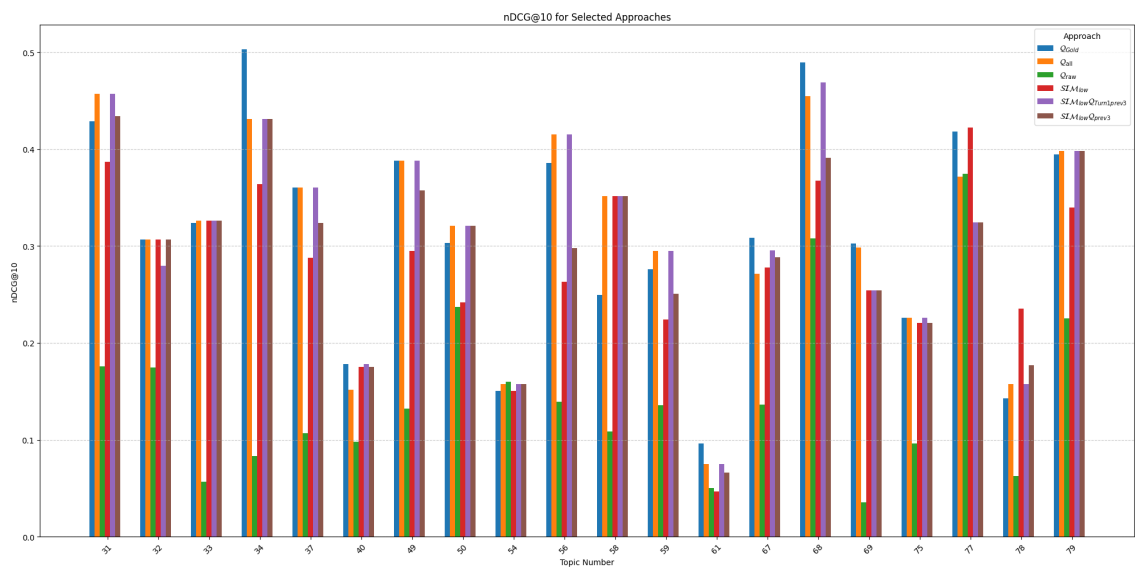


Figure 4.15: Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval

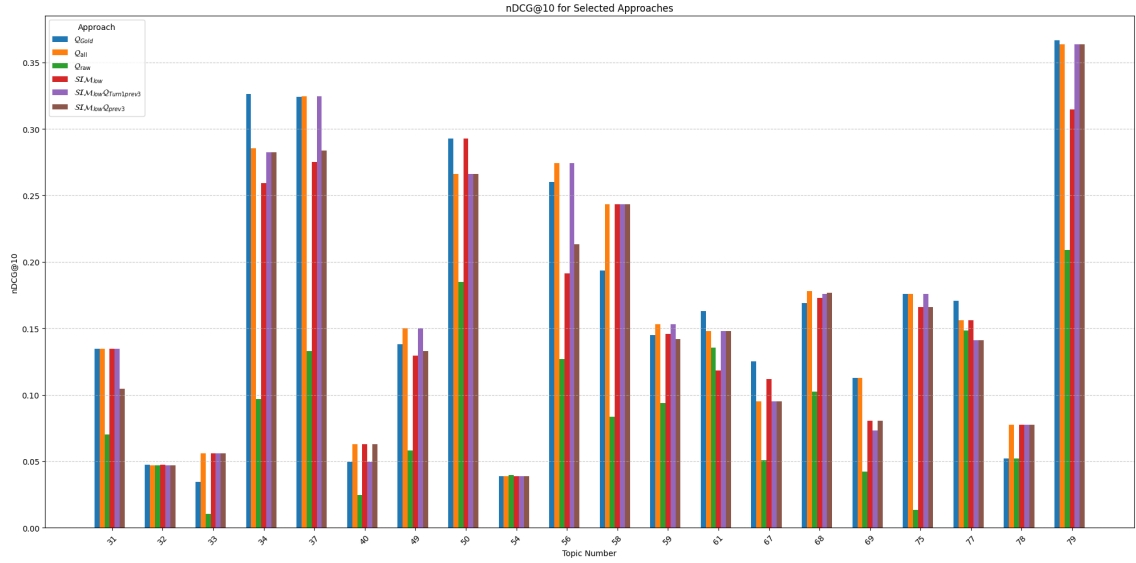


Figure 4.16: Topic-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval

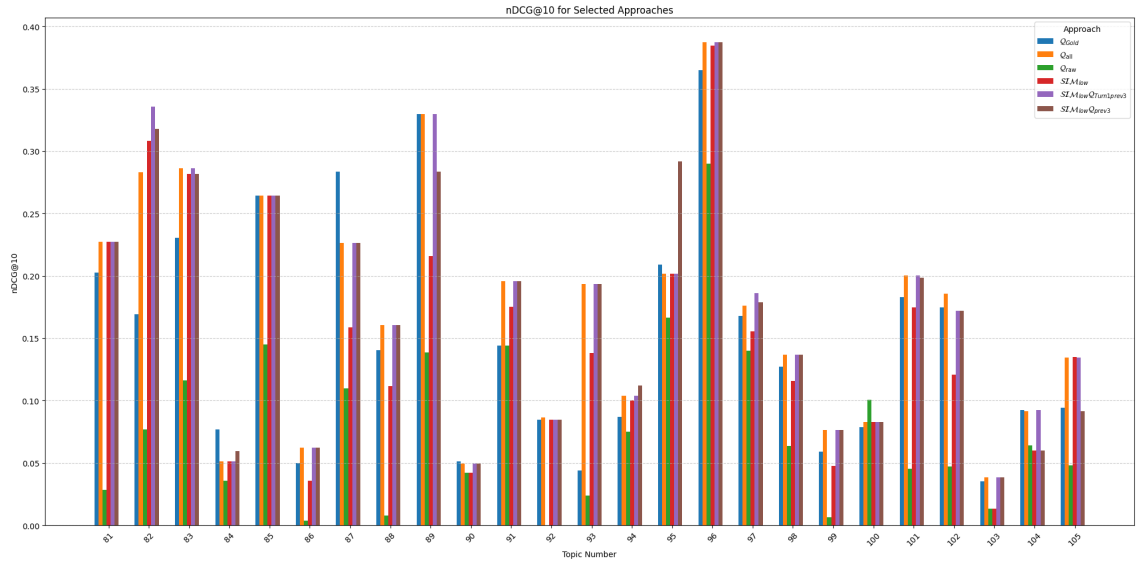


Figure 4.17: Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval

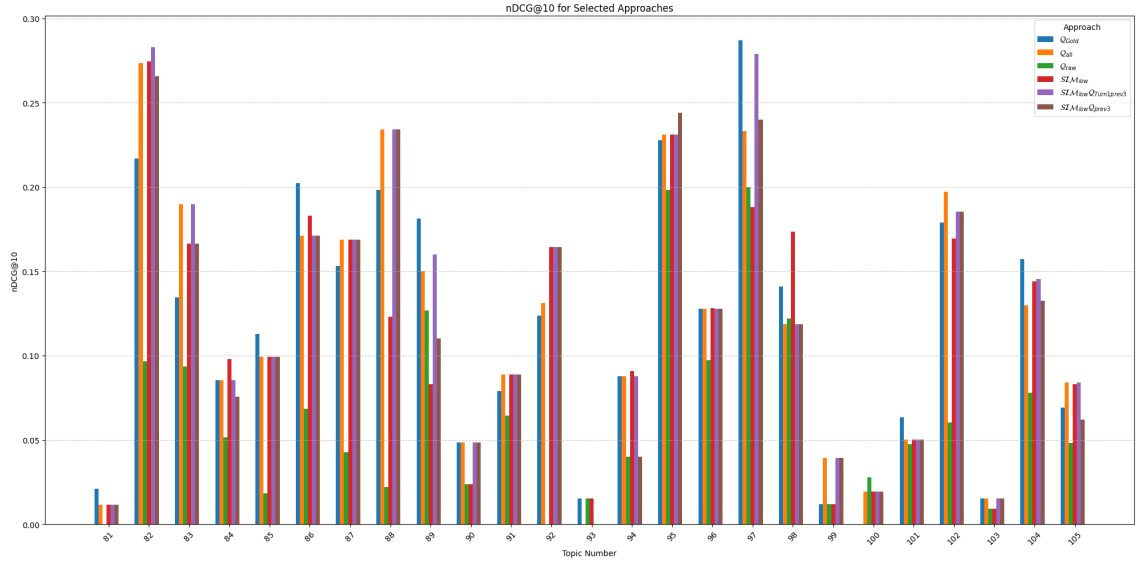


Figure 4.18: Topic-wise ndcg@10 Performance for CAsT 2020 Dense Retrieval

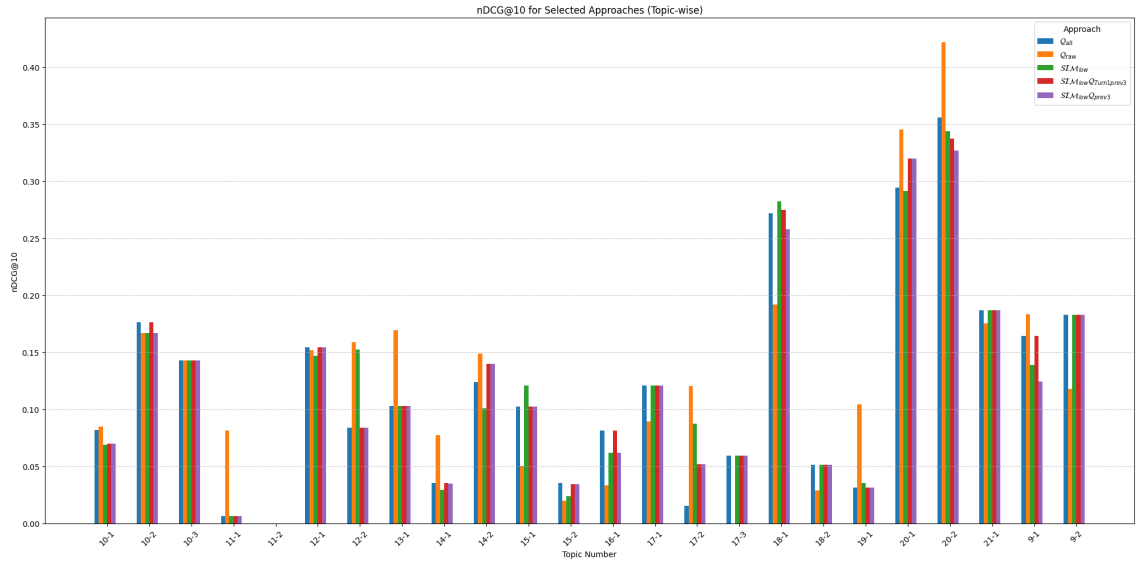


Figure 4.19: Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval

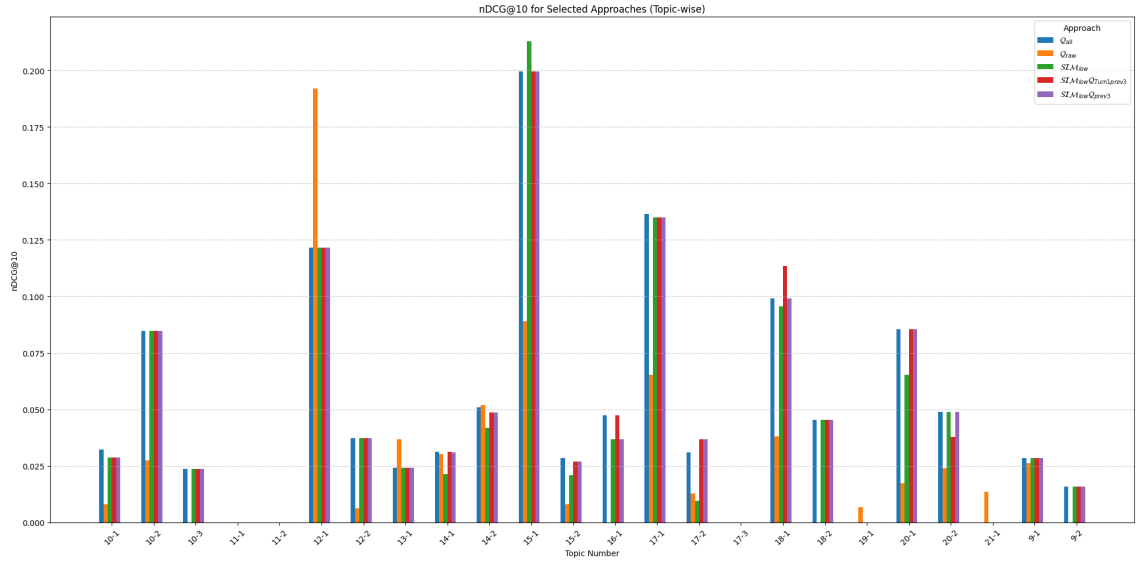


Figure 4.20: Topic-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval

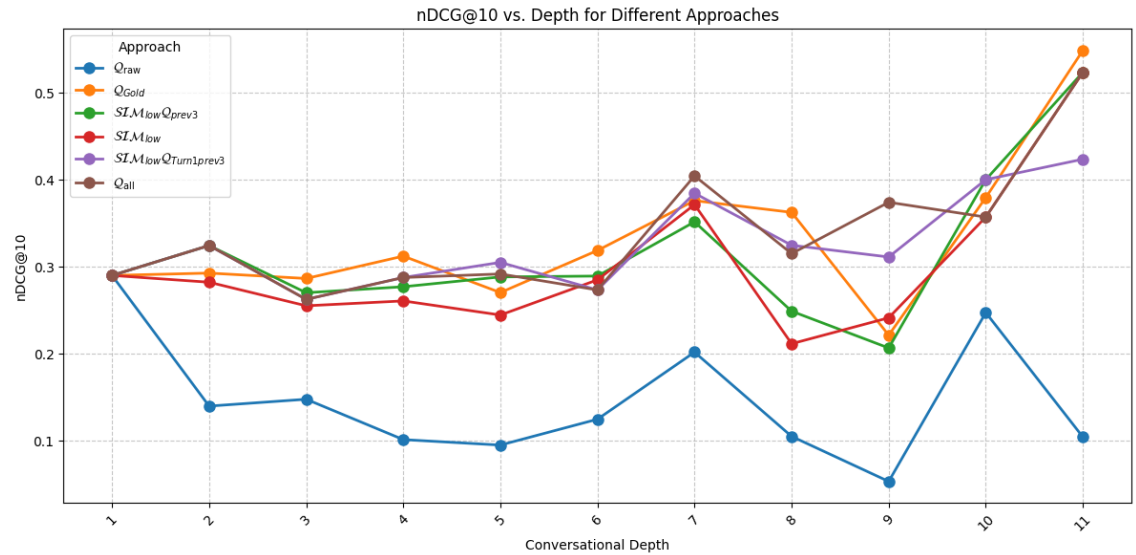


Figure 4.21: Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval

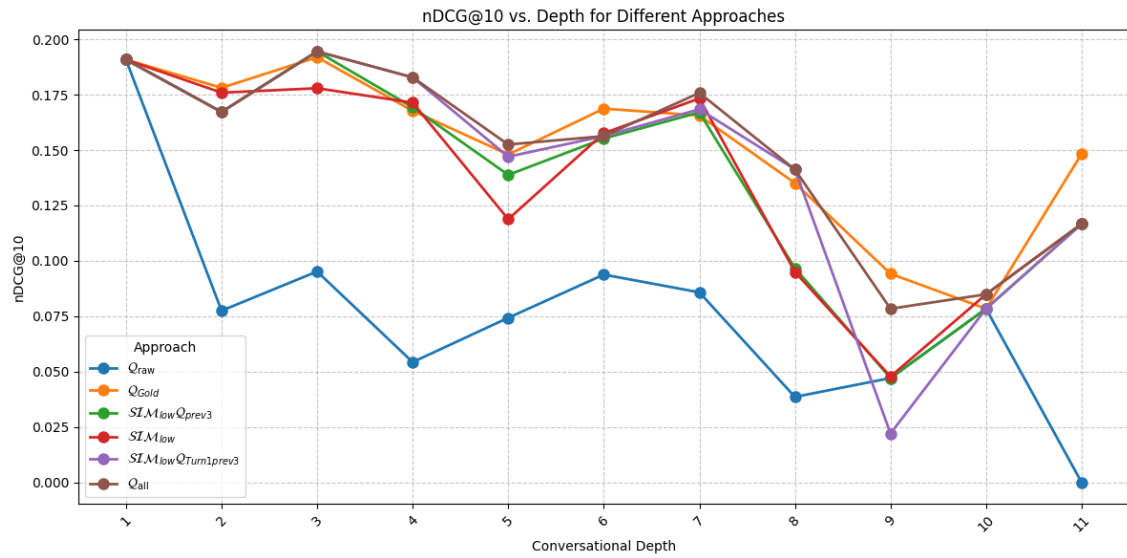


Figure 4.22: Depth-wise ndcg@10 Performance for CAsT 2019 Dense Retrieval

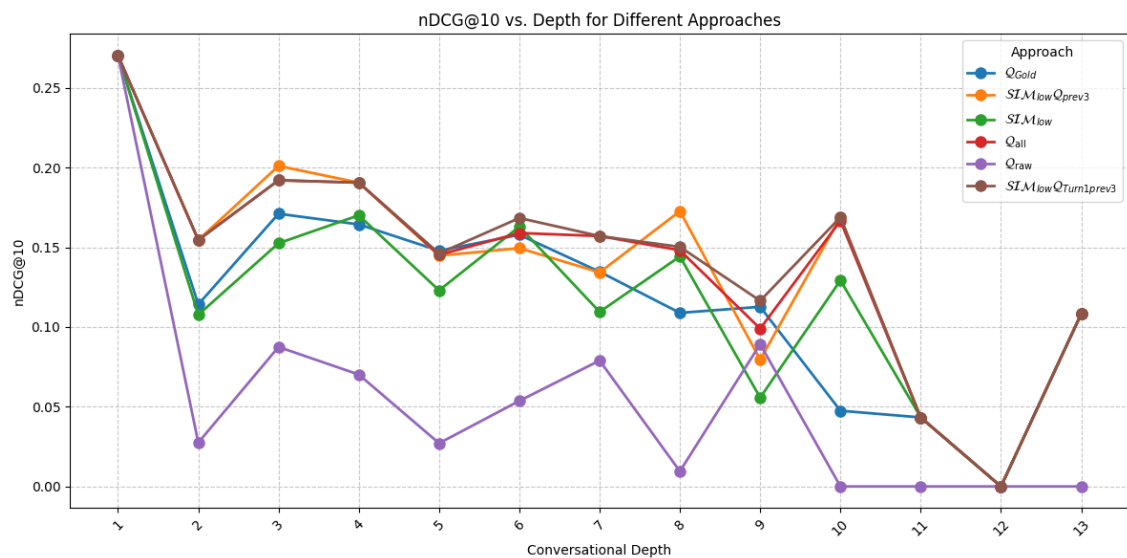


Figure 4.23: Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval

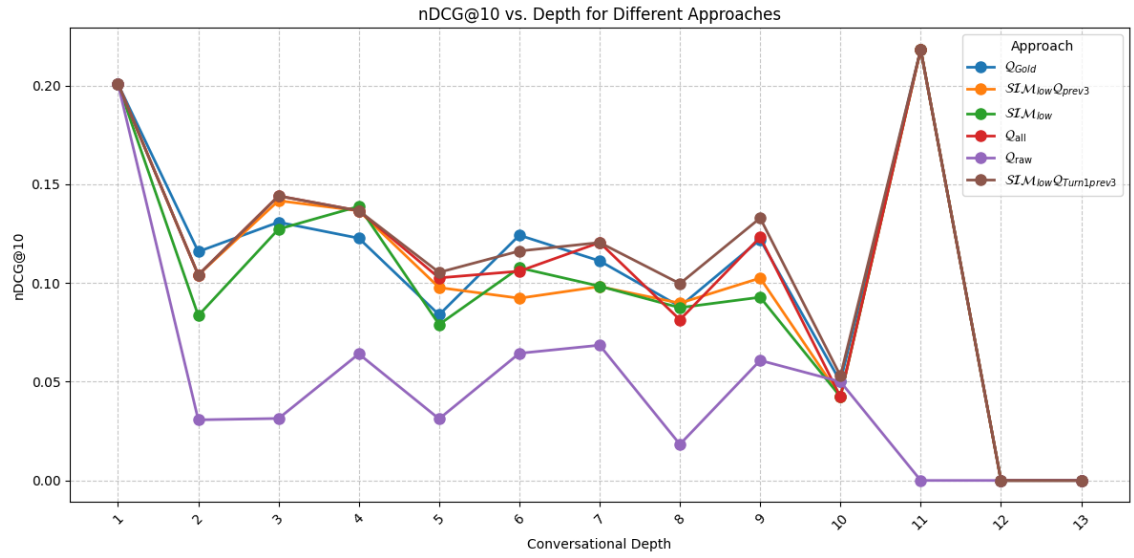


Figure 4.24: Depth-wise ndcg@10 Performance for CAsT 2020 Dense Retrieval

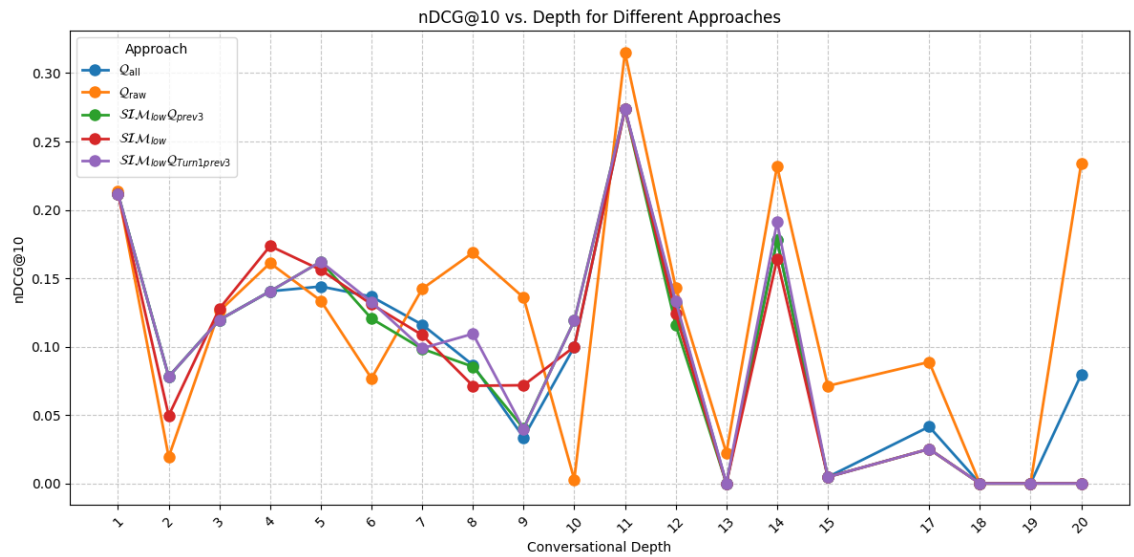


Figure 4.25: Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval

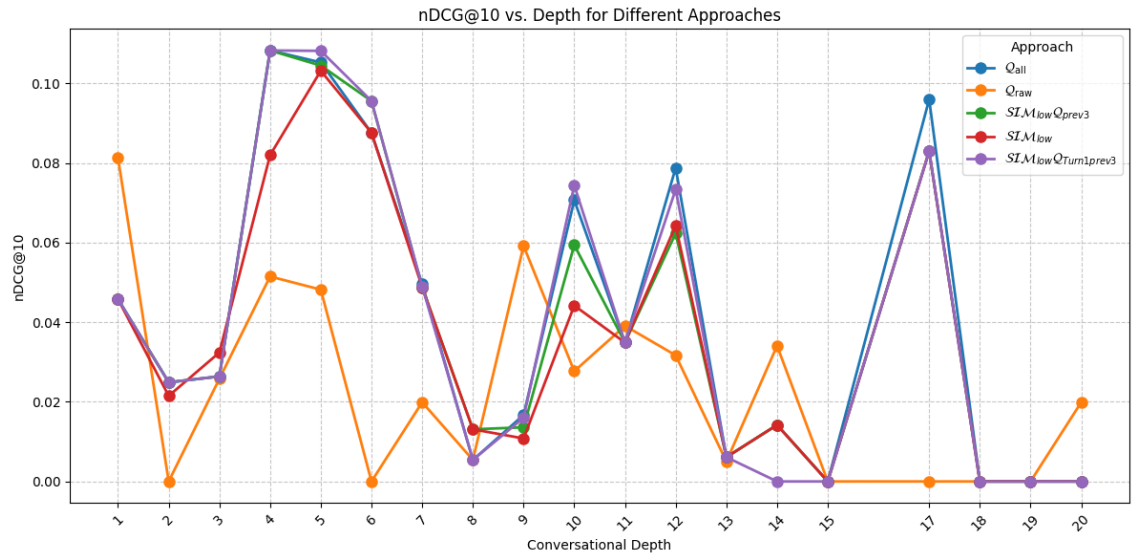


Figure 4.26: Depth-wise ndcg@10 Performance for iKAT 2023 Dense Retrieval

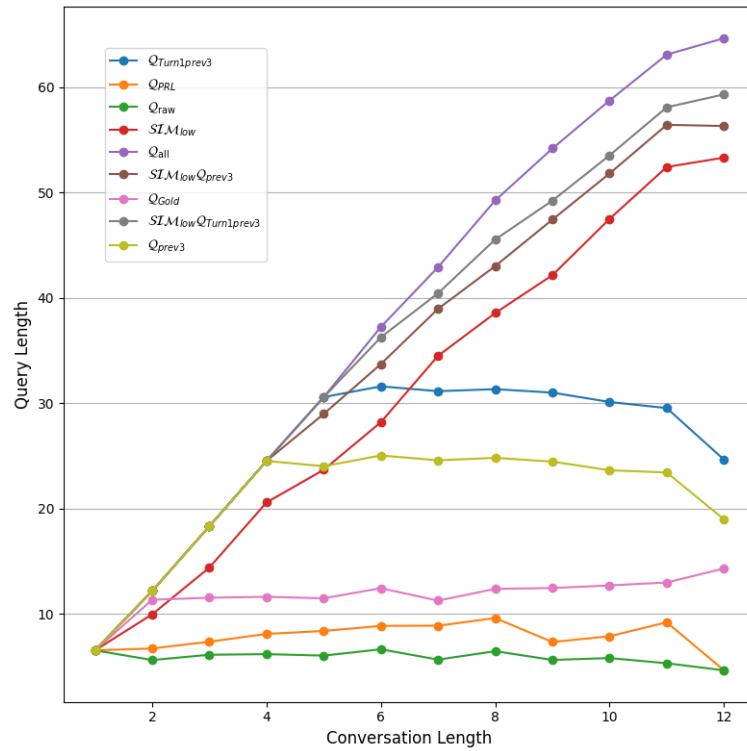


Figure 4.27: Average Query Length vs. Conversation Length (Depth) — CAsT 2019

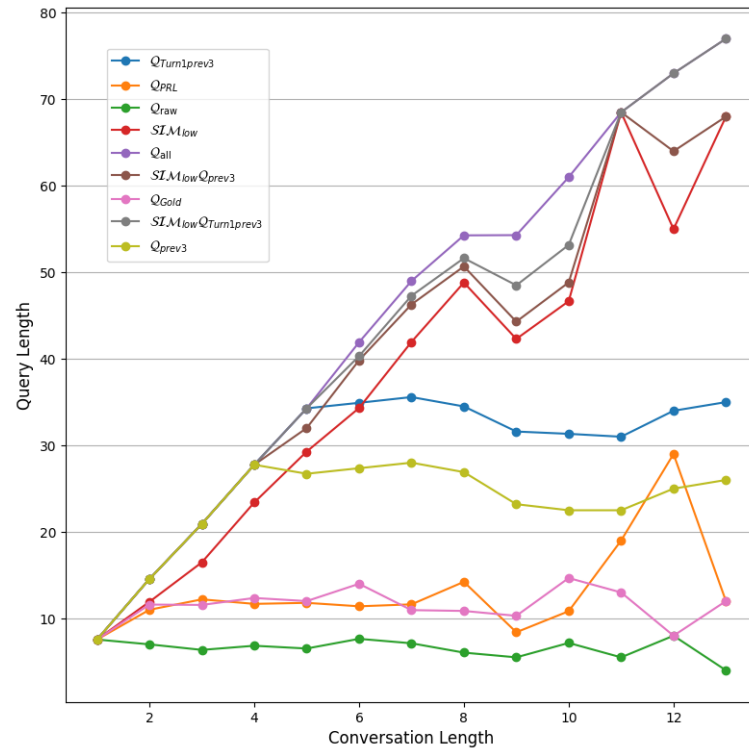


Figure 4.28: Average Query Length vs. Conversation Length (Depth) — CAsT 2020

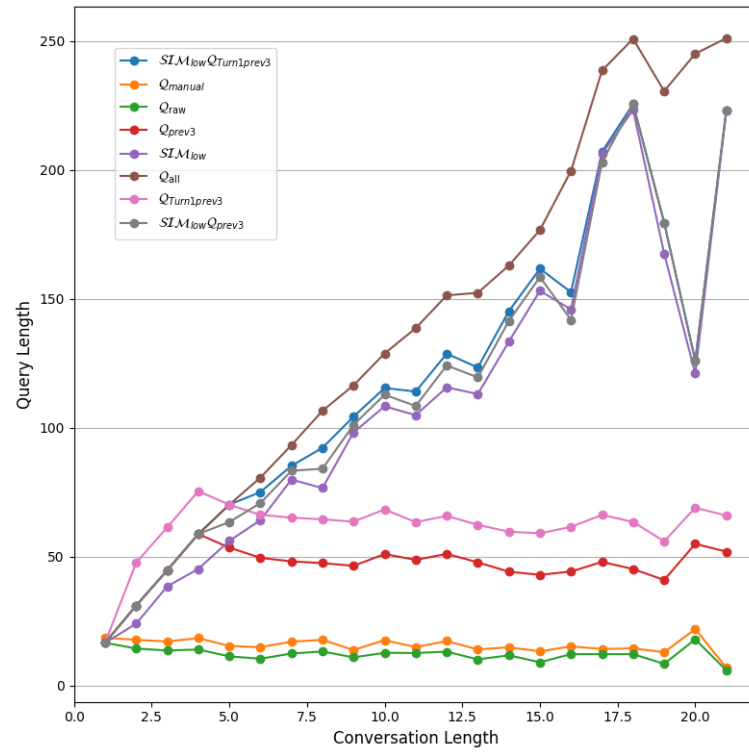


Figure 4.29: Average Query Length vs. Conversation Length (Depth) — iKAT 2023

Chapter 5

User Knowledge in Conversational Information Retrieval

In this chapter, we examine the integration of user knowledge into CIR and present a framework that formalizes this integration within CIR systems. We begin by exploring the role of user knowledge in the IR process. A user’s knowledge acquired during multi-turn interaction with a CIR system can have an impact on the effectiveness of the CIR system. Unlike traditional IR systems, CIR supports a multi-turn, interactive process through a dialogue, where users and systems can collaborate on refining the information need and improving the retrieval outcomes of the CIR system at each turn in the dialogue. By leveraging the knowledge that is acquired by a user during the dialogue, CIR systems can reduce potential cognitive load in the search process, support comprehension, and generate more relevant and contextually appropriate responses.

Despite these potential advantages, current research in CIR systems has not explored explicit mechanisms for representing and utilizing user knowledge during the search dialogue. A structured approach to integrating user knowledge could inform system behaviour, improve the interpretation of evolving information needs, and enable more effective interactions. To address this gap, we introduce a novel framework for integrating knowledge in CIR. We outline the architecture of this

framework and describe its key components, providing a high-level overview of their roles and interactions to support a knowledge-aware CIR system and investigate approaches to incorporate knowledge acquired during the dialogue and study its effectiveness.

This chapter is based on our publication (Acharya, Kando, and G. J. Jones 2024) and addresses **Hypothesis 2 (H2)** and its associated research questions, introduced in Chapter 1. **RQ1** asks whether features of the expected user’s developing knowledge can be identified and captured during a conversational search task. This is addressed through the proposed framework (Section 5.2) and its operationalization via keyword and summarization-based knowledge extraction. **RQ2** asks whether a model of the user’s knowledge can be exploited to improve the effectiveness of conversational search. This is addressed through the query enrichment experiments reported in Sections 5.3 and 5.4, where the extracted knowledge is integrated into query formulation at both turn and conversational levels. Taken together, these investigations constitute the first systematic examination of user knowledge modeling within the CIR dialogue setting, extending prior work on knowledge modeling in single-session IR (Câmara, El-Zein, and da-Costa-Pereira 2022; El Zein, Câmara, et al. 2023; Nasser et al. 2024) to the conversational, multi-turn context.

5.1 Introduction

In both traditional and conversational IR, users engage with an IR system with varying levels of knowledge about the topic of their information needs. They formulate a search query to express their information needs, which are then used by the retrieval systems to seek relevant information. Users with substantial prior domain knowledge about the topic of their information need can create rich queries which include appropriate domain-specific vocabulary, leading to a greater likelihood of retrieval of relevant search results, while those with limited domain knowledge struggle to formulate effective queries. The latter must often refine their queries over multiple search passes as they learn more about the topic from retrieval results which do not

satisfy their information needs. This iterative refinement of queries in the search process imposes a high cognitive load on the user and limits the effectiveness of traditional search systems.

A key limitation of traditional IR systems is the lack of the capability to engage interactively with users. These systems are entirely user-driven, placing a high cognitive load on users, who must formulate progressive queries, examine retrieved results, and determine (without support) whether their information needs are satisfied. If not, the user must repeat the process, which can be cumbersome and inefficient. CIR can potentially address this limitation by incorporating an interactive and collaborative process between the system and the user. As such, CIR systems have the potential to reduce the user's cognitive load by adapting to various user preference parameters, including *knowledge* to provide more personalized assistance, as illustrated in Figure 5.1.

With each interaction in the dialogue, information is progressively accumulated by the user, aiding them in a better understanding of their search topic and improving their knowledge regarding the topic. By representing the user's knowledge acquisition and its continuous refinement, a CIR system might better comprehend and respond to addressing the information needs and support the users in satisfying these needs, resulting in more effective search outcomes. To do this, the CIR system must leverage information gathered during each interaction to enhance CIR system performance by guiding subsequent system actions.

The potential significance of the role of knowledge exploitation in the search process is heightened in CIR systems because knowledge can aid in understanding the user's needs and facilitating effective dialogue with the user, and in tasks such as query expansion, clarification, and information elicitation. (Radlinski and Craswell 2017) delves into conversational approaches to IR and identifies adaptability in conversation as a key characteristic for effective CIR systems. This adaptability allows search systems to dynamically adjust conversational dialogue based on the users' existing knowledge and newly acquired knowledge during the search process,

continuously refining the process until the user’s information needs are met.

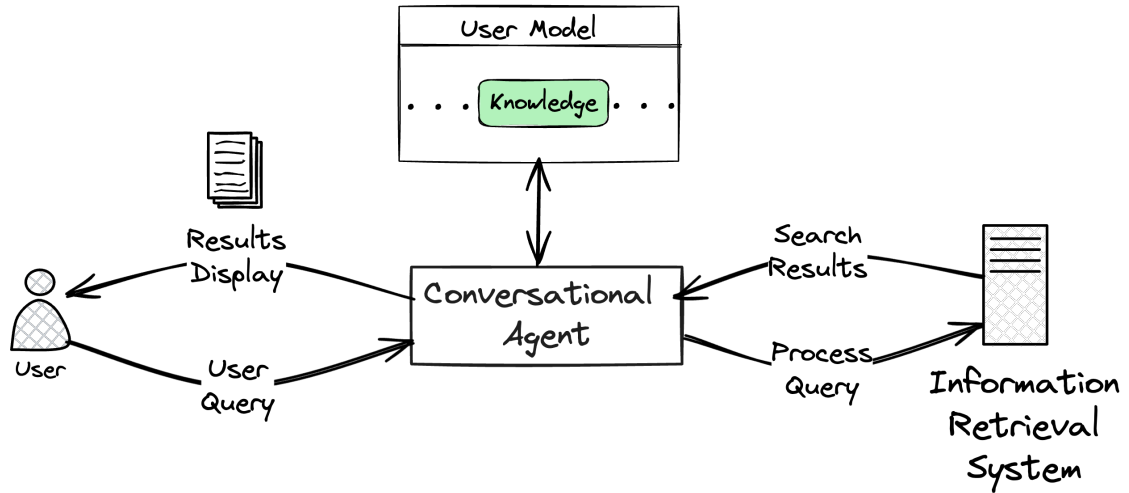


Figure 5.1: User’s Knowledge exploitation in Conversational Information Retrieval (CIR) (Acharya, Kando, and G. J. Jones 2024).

Most IR systems do not consider the user’s knowledge when retrieving relevant information. Hence, irrespective of the fact that every user is unique and therefore has different expectations from the IR systems based on their knowledge of the search topic, existing systems fail to take this into account, leading to suboptimal IR systems. A truly effective IR system should involve minimal effort spent by the user to reach their information goal. We argue that utilizing the user’s knowledge about the search topic can help achieve this goal. This becomes even more imperative in CIR settings, where the system acts as an agent in guiding the user through various processes, and the user’s knowledge can help drive the dialogue by the system to reach the information goal with minimal effort.

To address this proposition, we propose a framework for capturing, modelling, and utilizing user knowledge in CIR. This framework aims to improve the performance and adaptability of conversational search systems, making them more effective and responsive to users’ evolving information needs.

5.2 A Framework for Knowledge Integration in CIR

A CIR system seeks to assist users as they move through various stages of query formulation and refinement in a sequence of interactions. To do this effectively, the CIR system should progressively understand the user’s information need by exploiting knowledge with which the user has engaged during the search process and insights that can be assumed to have been gained from previous search results. However, this potential to enhance the user’s search experience is not a component of existing CIR systems.

One way in which users’ knowledge gained from previous search results might be integrated in the CIR process is by enriching queries to form a more complete statement of the information need. This approach, we hypothesise, allows for a more effective search outcome for each turn in the dialogue.

Information retrieved at each turn of the dialogue exposes the user to information in documents deemed relevant by the search system. As the user interacts with these documents, their understanding of the topic evolves. To capture and model this evolving user knowledge, we propose to extract knowledge from the documents retrieved during each turn, or specifically from the documents that the user engages with. This knowledge can then contribute to refining or expanding the user’s queries in future turns, helping the system to provide the user with more relevant results at subsequent turns.

Knowledge can be categorized into several types, including task knowledge, procedural knowledge, system knowledge, domain knowledge, and topical knowledge, each of which contributes to a user’s overall knowledge. In this study, we focus specifically on domain knowledge and topical knowledge. Domain knowledge influences how users formulate their queries (Dosso et al. 2022; O’Brien et al. 2020), while topical knowledge refers to the query terms and concepts users acquire during the search process (Hagen et al. 2016).

Our conceptualization of how users acquire and refine these forms of knowledge is based on Nicholas J Belkin (1980)’s ASK framework, Kuhlthau (1991)’s (ISP) and Vakkari (2016)’s Searching-as-Learning (SAL) paradigm, which conceptualizes information retrieval as a learning process in which topical acquisitions satisfies the user’s information need and progressively update and deepen user’s domain knowledge as described in Section 2.3.

Research on users’ knowledge in retrieval systems has been explored in several studies. For instance, (Câmara, El-Zein, and da-Costa-Pereira 2022) proposed a keyword-based approach to representing knowledge during a search session. They also introduced a large language model (LLM) 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 incorporated named entities (El Zein, Câmara, et al. 2023) and knowledge graphs (Nasser et al. 2024), providing a more nuanced understanding of users’ knowledge during their search processes.

Our proposed framework for integrating knowledge into CIR is built around three primary components: (1) Knowledge Extractor, (2) Current Knowledge State, and (3) Knowledge Updater. In the following sections, we introduce these components in detail, explaining their roles and functionalities within the framework.

5.2.1 Components

Knowledge Extractor. The Knowledge Extractor (KE) is responsible for identifying and extracting pertinent information from various sources, including queries, documents, and user knowledge history. It analyzes these sources to extract what is deemed relevant knowledge *e.g.* topical knowledge, task knowledge etc...

$$K_{source} = KE(source) \tag{5.1}$$

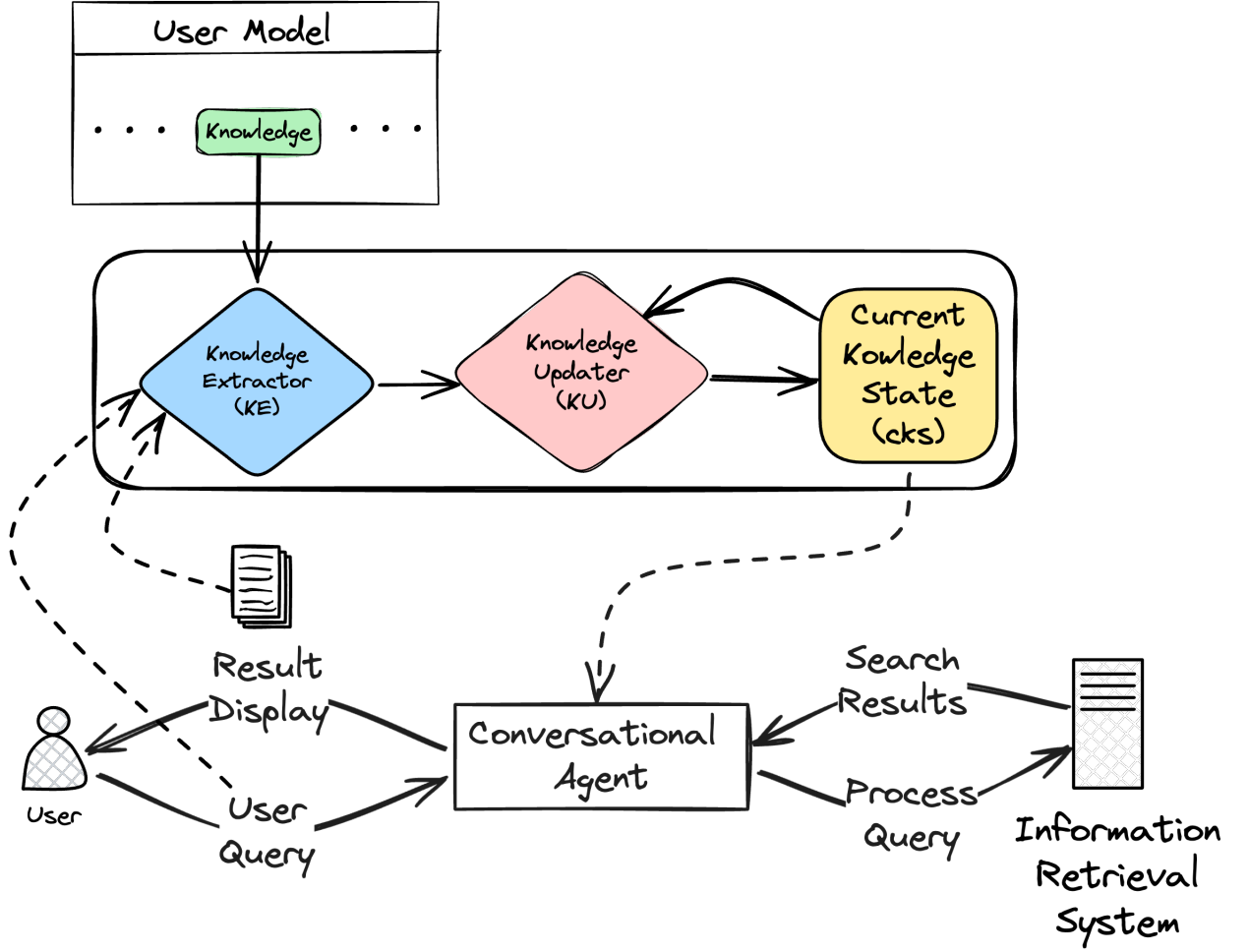


Figure 5.2: Proposed Framework Architecture for Incorporating Expected User Knowledge in CIR (Acharya, Kando, and G. J. Jones 2024).

where *source* refers to any of the following: Query (Q), Document (D), or User Knowledge (U_k).

The **KE** plays a crucial role in various parts of the proposed framework. The user’s interaction with a CIR system begins with the issuance of their query. The **KE** extracts knowledge from the user query (K_{query}) as well as from the existing user knowledge history stored in a user model (K_{U_k}). This initial extraction of knowledge constitutes the initial current knowledge state ($cks_{initial}$) in the CIR process as shown in Equation 5.2.

$$cks_{initial} = \{K_{query}, K_{U_k}\} \quad (5.2)$$

Subsequently, the **KE** is used again after the CIR system retrieves potentially relevant documents. The decision of which documents to extract knowledge from

depends on whether the user interacts with these documents when they are retrieved.

$$K_{\text{Document}} = KE(\text{Documents}) \quad (5.3)$$

Current Knowledge State. The Current Knowledge State (*cks*) reflects the current state of knowledge in the conversation at any given moment. It primarily interacts with the Knowledge Updater (***KU***) introduced in the next section, which is responsible for incorporating new information and is updated at each step of the conversation. As the dialogue progresses, the ***cks*** is continuously updated to accurately represent the user’s information needs and the context of the conversation. This ensures that the conversational system can take various actions (e.g eliciting additional information, requesting clarifications, asking for feedback, offering suggestions) based on strategies tailored to the user’s current knowledge state, facilitating a more natural and fluid progression of the dialogue. Consequently, the system can effectively adapt to the user’s evolving needs throughout the interaction.

Knowledge Updater The Knowledge Updater(***KU***), on the other hand, is responsible for seamlessly integrating the extracted knowledge into the existing knowledge state (*cks*). This component ensures that the system is updated with new knowledge at each turn in the conversation. By continuously incorporating new information, the ***KU*** keeps the knowledge state current and comprehensive, enabling the CIR system to maintain an accurate and up-to-date understanding of the user’s needs and the context of the conversation. This continuous updating process is crucial for the system’s ability to provide relevant and accurate responses and document rankings throughout the interaction.

$$cks_{\text{current}} = KU(cks_{\text{prev}}, K_{\text{Document}}) \quad (5.4)$$

where, cks_{current} is the updated knowledge state (*cks*) after the user’s interaction with documents. The document is represented by K_{Document} , generated by Equation

5.3, and KU is a function that takes cks_{prev} and $K_{Document}$ as inputs, combining them into an updated representation of the knowledge state.

5.2.2 Discussion

The components described above collaborate to integrate and update the relevant knowledge represented by the current knowledge state (***cks***) in the conversation. The ***KE*** interacts with the system by extracting knowledge from retrieved documents at each interaction in the conversation. This extracted knowledge is subsequently used by the ***KU*** to modify and update the ***cks***. The updated knowledge state helps the conversational system understand the user’s current needs and, based on this understanding, employ various strategies to take appropriate actions. Additionally, ***cks*** could potentially be leveraged and used to rank relevant documents according to the user’s current knowledge at each step. Figure 5.2 illustrates how the ***Knowledge Extractor*** and ***Knowledge Updater*** interact within the CIR system.

One significant challenge in dealing with knowledge is how to effectively represent it. Various approaches have been suggested for this, including extracting keywords (El Zein and Costa Pereira 2021; Syed and Collins-Thompson 2017a), using concept maps (Novak 2010), identifying named entities (El Zein, Câmara, et al. 2023), creating knowledge graphs (Nasser et al. 2024) and using LLMs (Câmara, El-Zein, and da-Costa-Pereira 2022). The choice of representation of knowledge has significant implications for the search process. It also affects operational aspects linked to the ***KU*** component, influencing the efficiency and effectiveness of knowledge extraction and application.

Earlier work models user knowledge during a search session, while the proposed framework models knowledge throughout a multi-turn conversation. Its main contribution is that the ***Knowledge Extractor*** and ***Knowledge Updater*** work after every conversation turn. As a result, the ***cks*** reflects not only the documents the user has seen, but also how the user’s knowledge changes during the conversation. Therefore, the experiments test not only whether extracted knowledge improves

retrieval, but also whether knowledge from the conversation provides extra benefit beyond standard relevance feedback.

In the following sections, we examine various approaches to knowledge extraction and its integration within the CIR process, as guided by our proposed Framework 5.2. Although the framework is designed to be general, this investigation is limited to the user’s knowledge as accumulated during the course of the dialogue.

5.3 Keyword-Based Knowledge Integration

Building on traditional keyword-based approaches to knowledge representation (Câmara, El-Zein, and da-Costa-Pereira 2022), we adapt this idea to the CIR setting by extracting words or phrases which are assumed to be the most informative and representative of the document’s content (key terms) relating to the information need of the user. These key terms can then be used to expand future queries, providing additional contextual information.

These key terms serve as a representation of the most relevant information the user has been exposed to, and as such can help to enrich the user’s query in subsequent search turns.

This addresses part of **RQ1** by testing whether features of the user’s changing knowledge can be identified and captured. We use keywords from the retrieved documents as a simple representation of the concepts the user has seen during the conversation. The main assumption is that if a user reads a document with a term, they have been exposed to that concept. This is a simple assumption because it does not show whether the user understood or remembered the information. This limitation is discussed in Chapter 6.

This query enrichment is intended to leverage the user’s evolving understanding of the topic as informed by the documents that they interact with. This process is designed to enable the search system to better capture the user’s shifting information state and needs over multiple turns in a conversation. The overall illustration for this approach is illustrated in Figure 5.3, which shows the steps involved in the knowledge-

integrated query enrichment process, from document retrieval to knowledge extraction and query enrichment for subsequent turns across the conversation.

Following this general framework, we detail the specifics of the knowledge extractor component, which in this case will be the extraction of the key terms from the retrieved documents.

Knowledge Extractor. We use YAKE (R. Campos et al. 2020), an unsupervised and corpus-independent keyword extraction method, to identify salient terms from documents. We describe it in detail in the following section. To extract key terms from the selected top documents in the retrieved list by the search system during each turn of the conversation. For each retrieved document, YAKE was applied to identify the top- n key terms based on their relevance score. These key terms were then incorporated into the user’s next query to refine and enrich their search, capturing the evolving knowledge of the user as they interact with the system, enabling the search engine to deliver more contextually relevant results.

YAKE. Yet Another Keyword Extractor (YAKE) is an unsupervised and corpus-independent keyword extraction method to identify salient terms from documents (R. Campos et al. 2020). The algorithm evaluates the importance of each candidate term (t) in the document using five features. These features include: (1) a normalized term frequency TF_{Norm} , (2) a casing feature T_{Case} , capturing the notion that uppercase terms tend to be more important than lowercase ones, (3) a term position $T_{Position}$, capturing the notion that important keywords tend to appear at the very beginning of a document, (4) a term context relatedness T_{rel} , capturing the notion that the higher the number of different terms that co-occur with the candidate term (t) the less significant term, and (5) a term different sentence $T_{Sentence}$ measure capturing the notion that terms which appear in many different sentences have a higher probability of being important.

These components are combined to yield a single-term importance score $S(t)$ as shown in Equation 5.5, with lower scores indicating higher importance.

$$S(t) = \frac{T_{\text{Rel}} \cdot T_{\text{Position}}}{T_{\text{Case}} + \frac{TF_{\text{Norm}}}{T_{\text{Rel}}} + \frac{T_{\text{Sentence}}}{T_{\text{Rel}}}} \quad (5.5)$$

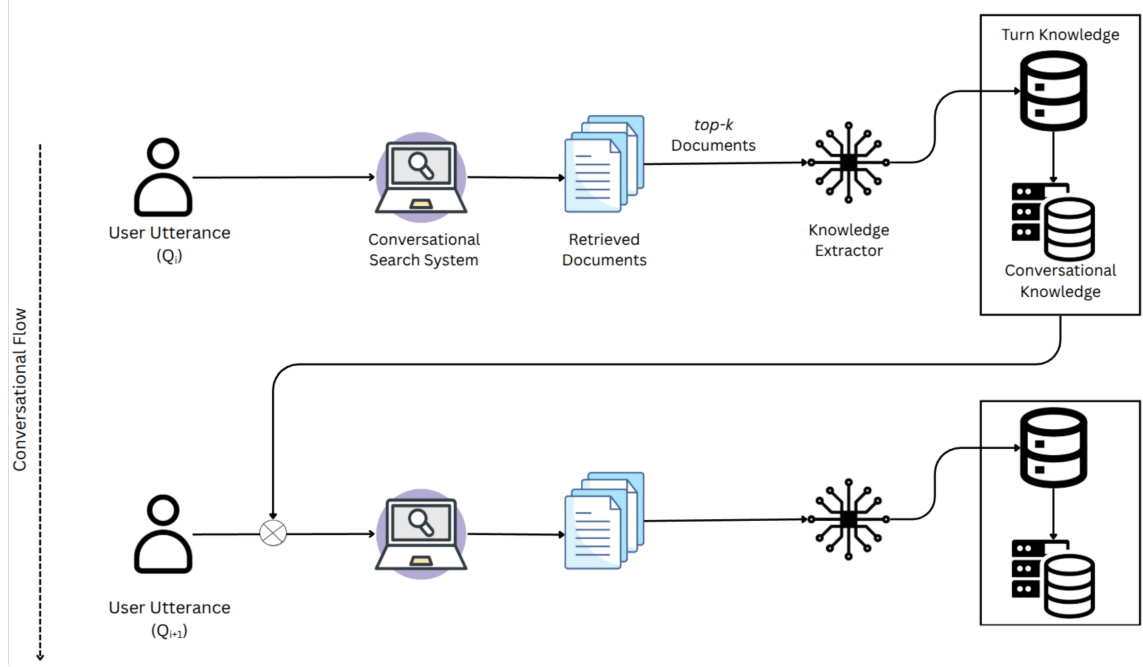


Figure 5.3: Illustration of Knowledge Integration for Query Enrichment.

5.3.1 Various Formulations of Query Enrichment

We explore two different query reformulation strategies. (1) Query expansion and (2) Query Rewriting. In the expansion-based approach, we simply concatenate the extracted knowledge (i.e. summary) to the user query, whereas in the rewriting approach, we utilize the extracted knowledge along with the user query to rewrite the query.

1. $\mathbf{Q}_{raw}$: User-issued query at each turn in the conversation in the conversation.

$$q = (q_n)$$

2. $\mathbf{Q}_{raw} + \text{RM3}$: User-issued query utterance with RM3 pseudo-relevance feedback (Lavrenko and W. Bruce Croft 2001) based relevance-feedback approach for query expansion.

$$q = (q_n \odot \{\text{RM3expansion}\})$$

Levels of Conversational Knowledge. A dialogue consists of a series of multi-turn interactions between the user and the system. These interactions can be categorized into two levels (1) turn-level and (2) conversational-level. In the turn-level interaction, only the current interaction between the user and system is considered without taking the broader conversational context into consideration. This can be seen as a single-shot, ad-hoc retrieval in IR. Whereas, in the conversational-level interaction, the broader conversational context is also taken into consideration. In order to reflect these levels, we consider that knowledge acquisition by the user occurs at both turn and conversational levels and investigate the knowledge integration to enrich queries at these levels as detailed below.

Turn Knowledge ($\mathcal{K}$) Turn-level knowledge extracted from retrieved relevant documents at each turn in the conversational dialogue.

1. $\mathbf{Q}_{raw} + \mathcal{K}_i$: User-issued query utterance concatenated with knowledge extracted at i^{th} turn for query expansion.

$$q = q_i \odot \mathcal{K}_i$$

2. $\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$: Knowledge extracted only from the previous turn in the conversation, concatenated with the user-issued query utterance in the i^{th} turn.

$$q = (\mathcal{K}_{i-1} \odot q_i)$$

Conversational Knowledge ($\mathcal{CK}$) Conversational level knowledge accumulated throughout the dialogue in the search process.

3. $\mathbf{Q}_{\mathcal{CK},all}$: User-issued query utterance with all available conversational knowledge.

$$q = \mathcal{CK}_{1:n-1} \odot q_i$$

4. $\mathbf{Q}_{\mathcal{CK},\text{sim}}$: User-issued query utterance with selected conversational knowledge (based on similarity to the i^{th} user-issued query utterance). We consider the top-3 similar conversational knowledge.

$$q = \mathcal{CK}_{\text{sim}} \odot q_i$$

5. $\mathbf{Q}_{\mathcal{CK},\text{rank}}$: User-issued query utterance with selected conversational knowledge (based on relevance to the user-issued query utterance). We consider the top-3 ranked conversational knowledge based on BM25 ranking.

$$q = \mathcal{CK}_{\text{rank}} \odot q_i$$

5.3.2 Results and Discussion

We present our results for keyword-based knowledge integration for query enrichment at both the turn and conversational levels in the dialogue. We experimented with all combinations of the number of keywords extracted (5 and 10) and the number of documents (5 and 10) to evaluate their impact on performance. The results reported here correspond to the configuration that achieved the best performance in the query expansion baseline ($\mathbf{Q}_{\text{raw}} + \text{RM3}$) which is 10 Documents and 10 Keywords, with the original query weighted to 0.5.

CAsT Datasets. Table 5.1 presents the performance of the sparse retrieval system on the TREC CAsT 2019 and 2020 datasets. Overall, the results show that both datasets remain challenging for sparse retrieval, and the effectiveness of different query reformulation strategies varies considerably.

The user-issued raw queries ($\mathbf{Q}_{\text{raw}}$) achieve an nDCG@10 of 0.1479 on CAsT 2019 and 0.0774 on CAsT 2020. Applying RM3 pseudo-relevance feedback consistently improves all metrics. These results confirm that traditional relevance feedback remains a strong baseline, even in conversational search settings.

In contrast, incorporating turn-level knowledge does not yield improvements over the baselines. Both $((\mathbf{Q}_{\text{raw}} + \mathcal{K}_i))$ and $(\mathcal{K}_{i-1} + \mathbf{Q}_{\text{raw}}))$ lead to a decrease in

Table 5.1: Performance of a Sparse Retrieval (BM25) System TREC CAsT 2019, 2020. **Bold** and underline indicate the best result and the second best result. α and β denote no statistically significant difference over the baselines $\mathbf{Q}_{raw}$ and $\mathbf{Q}_{raw} + \text{RM3}$, respectively. (Paired t-test at $p < 0.05$)

Query	CAsT 2019			CAsT 2020		
	nDCG@10	MRR	MAP@1000	nDCG@10	MRR	MAP@1000
Baselines						
$\mathbf{Q}_{raw}$	<u>0.1479</u>	0.3299	<u>0.1163</u>	<u>0.0774</u>	<u>0.1786</u>	<u>0.0555</u>
$\mathbf{Q}_{raw} + \text{RM3}$	0.1732	<u>0.3493</u>	0.1407	0.0892	0.1873	0.0704
Turn Knowledge						
$\mathbf{Q}_{raw} + \mathcal{K}_i$	0.1090	0.2569	0.0816	0.0587	0.1379	0.0365
$\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$	0.0727	0.1941	0.0563	<u>0.0774</u> $^\alpha$	<u>0.1786</u> $^{\alpha\beta}$	<u>0.0555</u> $^\alpha$
Conversational Knowledge						
$\mathbf{Q}_{CK,all}$	0.1048	0.2891 $^{\alpha\beta}$	0.0763	0.0616 $^\alpha$	0.1594 $^{\alpha\beta}$	0.0419 $^\alpha$
$\mathbf{Q}_{CK,sim}$	0.1349 $^\alpha$	0.3374 $^{\alpha\beta}$	0.1003 $^\alpha$	0.0721 $^{\alpha\beta}$	0.1772 $^{\alpha\beta}$	0.0454 $^\alpha$
$\mathbf{Q}_{CK,rank}$	0.1444 $^{\alpha\beta}$	0.3569 $^{\alpha\beta}$	0.1047 $^\alpha$	0.0697 $^{\alpha\beta}$	0.1762 $^{\alpha\beta}$	0.0448 $^\alpha$

performance across all metrics. For example, on CAsT 2019, nDCG@10 falls to 0.1090 and 0.0727, respectively, compared to 0.1479 for the raw query. Similar declines occur in the CAsT 2020 dataset. These findings suggest that naïvely adding knowledge to queries can introduce noise.

The three conversational-knowledge reformulations produce mixed results. Using all conversational knowledge ($\mathbf{Q}_{CK,all}$) performs the worst among the three, indicating that enriching queries too aggressively harms the performance. However, selectively incorporating conversational knowledge using similarity ($\mathbf{Q}_{CK,sim}$) or rank-based filtering ($\mathbf{Q}_{CK,rank}$) leads to more competitive results. On CAsT 2019, $\mathbf{Q}_{CK,rank}$ reaches $\text{nDCG@10} = 0.1444$ and $\text{MRR} = 0.3569$, nearly matching or slightly outperforming RM3 in MRR. $\mathbf{Q}_{CK,sim}$ also performs better than most turn-level approaches, although it still does not outperform the RM3 approach on any metric. The same pattern holds for CAsT 2020, where the selective conversational-knowledge queries outperform the other knowledge-based methods but remain below the RM3 baseline.

iKAT Dataset. Table 5.2 reports the performance of the sparse retrieval system on the TREC iKAT 2023 dataset. The user-issued queries ($\mathbf{Q}_{raw}$) achieve an nDCG@10 of 0.1302 and a MAP@1000 of 0.0505. These values are the strongest performance among all methods tested. Surprisingly, RM3 pseudo-relevance feedback does not improve retrieval on this dataset. Instead, performance drops across all metrics, with

Table 5.2: Performance of a Sparse Retrieval (BM25) System TREC iKAT 2023. **Bold** and underline indicate the best result and the second best result. No statistically significant difference over the baselines $\mathbf{Q}_{raw}$ and $\mathbf{Q}_{raw} + \text{RM3}$. (Paired t-test at $p < 0.05$)

Query	iKAT 2023		
	nDCG@10	MRR	MAP@1000
<i>Baselines</i>			
$\mathbf{Q}_{raw}$	0.1302	0.2828	0.0505
$\mathbf{Q}_{raw} + \text{RM3}$	<u>0.1052</u>	<u>0.2521</u>	<u>0.0356</u>
Turn Knowledge			
$\mathbf{Q}_{raw} + \mathcal{K}_i$	0.0538	0.1765	0.0168
$\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$	0.0048	0.0185	0.0018
Conversational Knowledge			
$\mathbf{Q}_{CK,all}$	0.0066	0.0219	0.0023
$\mathbf{Q}_{CK,sim}$	0.0150	0.0596	0.0051
$\mathbf{Q}_{CK,rank}$	0.0163	0.0795	0.0059

MAP@1000 falling from 0.0505 to 0.0356. This suggests that the feedback documents retrieved for iKAT may be too noisy, causing RM3 to expand queries in unhelpful directions.

Adding turn-level knowledge decreases retrieval effectiveness. When utilizing the current turn’s knowledge ($\mathbf{Q}_{raw} + \mathcal{K}_i$), nDCG@10 drops to 0.0538. Using the previous turn’s knowledge first ($\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$) produces even worse results across all metrics. These results indicate that naive expansion of turn-specific knowledge introduces substantial noise.

A similar trend is observed with conversational-knowledge query reformulations. Using all available conversational knowledge ($\mathbf{Q}_{CK,all}$) produces low scores on all metrics. The selective approaches ($\mathbf{Q}_{CK,sim}$ and $\mathbf{Q}_{CK,rank}$) perform somewhat better than the turn-level strategies, but they still remain far below the raw user query baseline. While these selective filters reduce noise compared to the full-knowledge approach, they still do not provide useful expansions.

In the CAsT datasets, RM3 is the strongest performer. The effectiveness of turn-level and conversational knowledge integration depends on how the knowledge is selected and incorporated. Utilizing all available knowledge tends to degrade perfor-

mance, whereas selection (similarity or rank-based) narrows down the performance gap but does not exceed it. Compared with the CAsT benchmarks, iKAT is notably more difficult. All methods achieve substantially lower effectiveness. This is expected given iKAT’s more complex, multi-step conversational structure and context needed from the PTKB to interpret each turn. Both pseudo-relevance feedback and all forms of knowledge-based expansion decrease the performance. Unlike the CAsT datasets, where some knowledge-based methods were competitive, in the case of the iKAT dataset, this was not the case.

These results give a mixed answer to **RQ2** for the keyword-based approach. On the CAsT datasets, the knowledge-based methods ($\mathbf{Q}_{\mathcal{CK},\text{sim}}$ and $\mathbf{Q}_{\mathcal{CK},\text{rank}}$) perform close to RM3, but they do not consistently perform better. On the iKAT dataset, none of the methods performs better than the raw user query. The results show that RM3 is still the strongest method on the CAsT datasets. This suggests that keyword-based knowledge adds little new information beyond what RM3 already captures. In some cases, it even adds unnecessary noise. The iKAT results also show that keyword-based knowledge is not enough for more difficult conversations. Therefore, the main contribution of this experiment is to show the limits of keyword-based knowledge and explain why a stronger summarization-based approach is needed.

Conversation Topic-Wise Analysis

In this section, we examine the retrieval effectiveness at a topic-level to highlight how different approaches to keyword-based query enrichment perform.

CAsT 2019. Figure 5.4 presents the topic-wise nDCG@10 performance for the selected query-enrichment approaches for a sparse (BM25) retrieval system. No clear pattern is evident across all topics, the results are mixed. For certain topics, $\mathcal{Q}_{\text{raw}}$ and $\mathcal{Q}_{\text{raw}} + \text{RM3}$ are clear winners (Topics 40, 54 and 58), while for others the conversational-knowledge reformulations ($\mathbf{Q}_{\mathcal{CK}}$) outperform them (Topics 32, 33, 34, 37, 50, 56, 69, and 78), indicating that enriching a query with knowledge accumulated over the conversational history can be valuable. Using all conversational knowledge

($\mathbf{Q}_{CK,all}$) performs the worst among the three, suggesting that overly aggressive query enrichment can harm performance. However, selectively incorporating conversational knowledge using similarity-based filtering ($\mathbf{Q}_{CK,sim}$) or rank-based filtering ($\mathbf{Q}_{CK,rank}$) yields more competitive results.

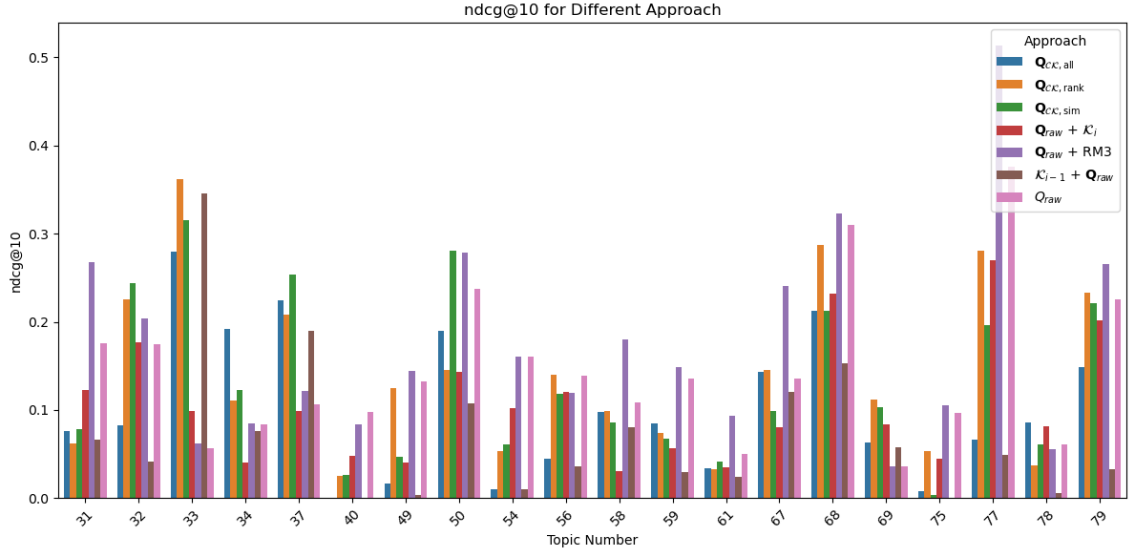


Figure 5.4: Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval.

CAsT 2020. Figure 5.5 presents the topic-wise nDCG@10 performance for the selected query-enrichment approaches for a sparse (BM25) retrieval system. No clear pattern is evident across all topics, the results are mixed. For certain topics, $\mathbf{Q}_{raw}$ and $\mathbf{Q}_{raw} + RM3$ are clear winners (Topics 100, 83 and 85), while for others the conversational-knowledge reformulations ($\mathbf{Q}_{CK}$) outperform them (Topics 101, 82, and 89), indicating that enriching a query with knowledge accumulated over the conversational history can be valuable. Using all conversational knowledge ($\mathbf{Q}_{CK,all}$) performs the worst among the three, suggesting that overly aggressive query enrichment can harm performance. However, selectively incorporating conversational knowledge using similarity-based filtering ($\mathbf{Q}_{CK,sim}$) or rank-based filtering ($\mathbf{Q}_{CK,rank}$) yields more competitive results.

For Topic 92, enriching the query using conversational-knowledge reformulations was able to retrieve relevant documents in the top-10 ranks, whereas the other approaches failed to retrieve any relevant documents within the top 10.

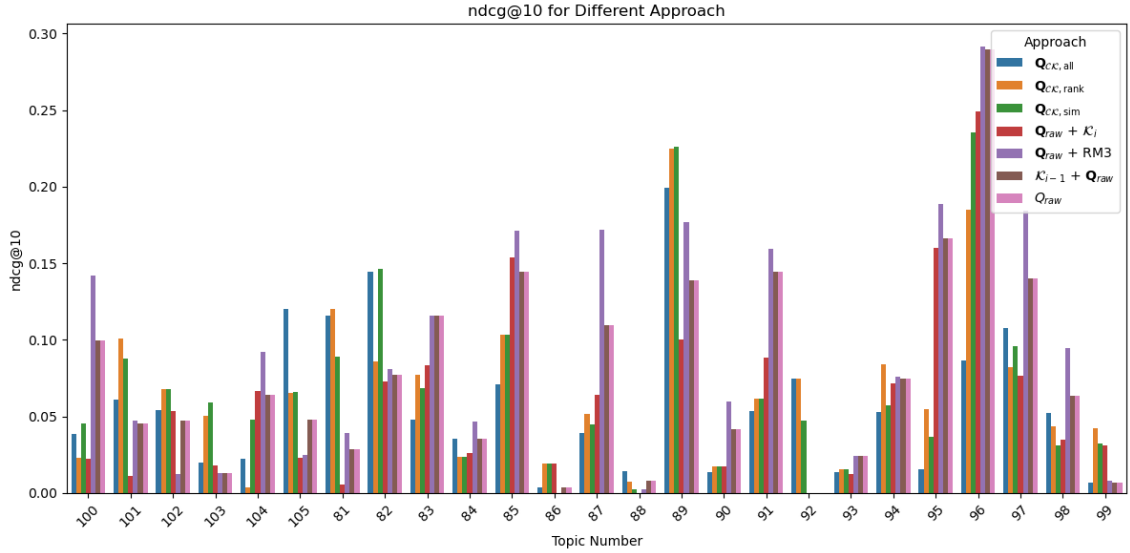


Figure 5.5: Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval.

iKAT 2023. Figure 5.6 presents the topic-wise nDCG@10 performance for the selected query-enrichment approaches for a sparse (BM25) retrieval system. None of the approaches for query enrichment, whether at the turn-level or at the conversational level, was able to outperform Q_{raw} and $Q_{raw} + RM3$ approaches throughout the topics. In general, Q_{raw} and $Q_{raw} + RM3$ performed the best.

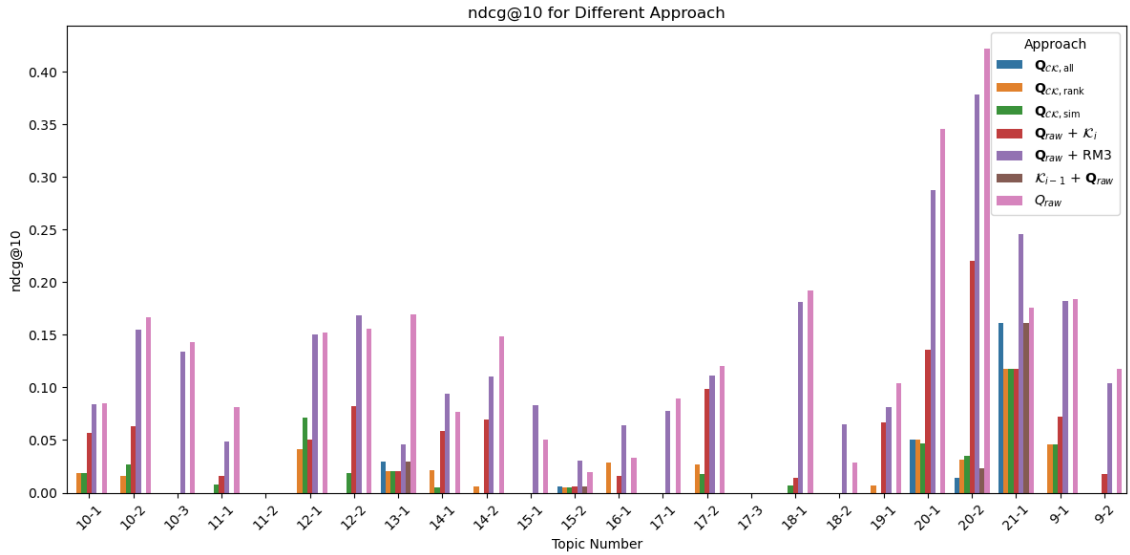


Figure 5.6: Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval.

Conversation Depth-Wise Analysis

In this section, we examine retrieval effectiveness across different conversational depths to highlight how different approaches to keyword-based query enrichment perform.

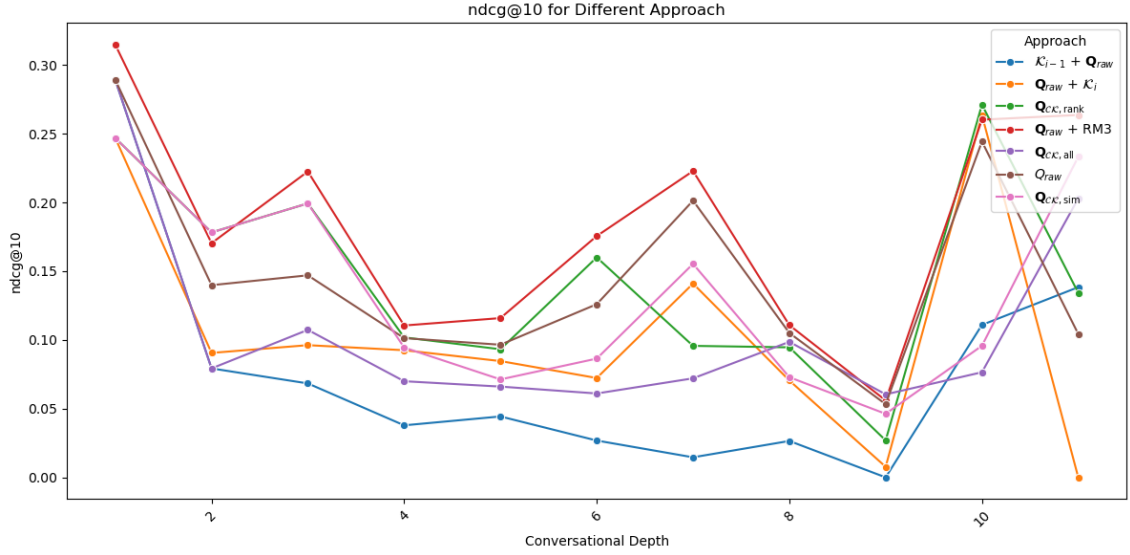


Figure 5.7: Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval.

CAsT 2019. Figure 5.7 presents the nDCG@10 performance of the selected query-enrichment approaches for a sparse (BM25) retrieval system across different conversation lengths (depths). A noticeable decline in performance is observed for all approaches as conversational depth increases, and neither turn-level nor conversational-level enrichment methods outperform the $Q_{raw} + RM3$ throughout the conversation. Overall, all approaches exhibit a similar pattern of gradually decreasing performance as the conversation lengthens.

CAsT 2020. Figure 5.8 presents the nDCG@10 performance of the selected query-enrichment approaches for a sparse (BM25) retrieval system across different conversation lengths (depths). The results show that all query formulation approaches exhibit a noticeable decline in performance as conversational depth increases. Overall, the performance of all approaches gradually decreases as the conversation lengthens. A significant drop in performance is observed at depth 2.

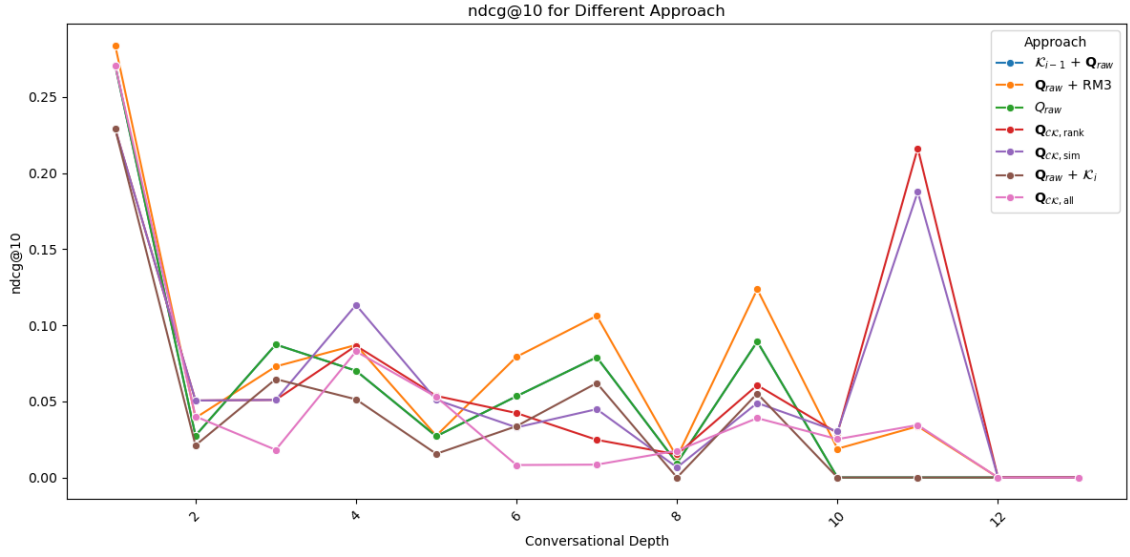


Figure 5.8: Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval.

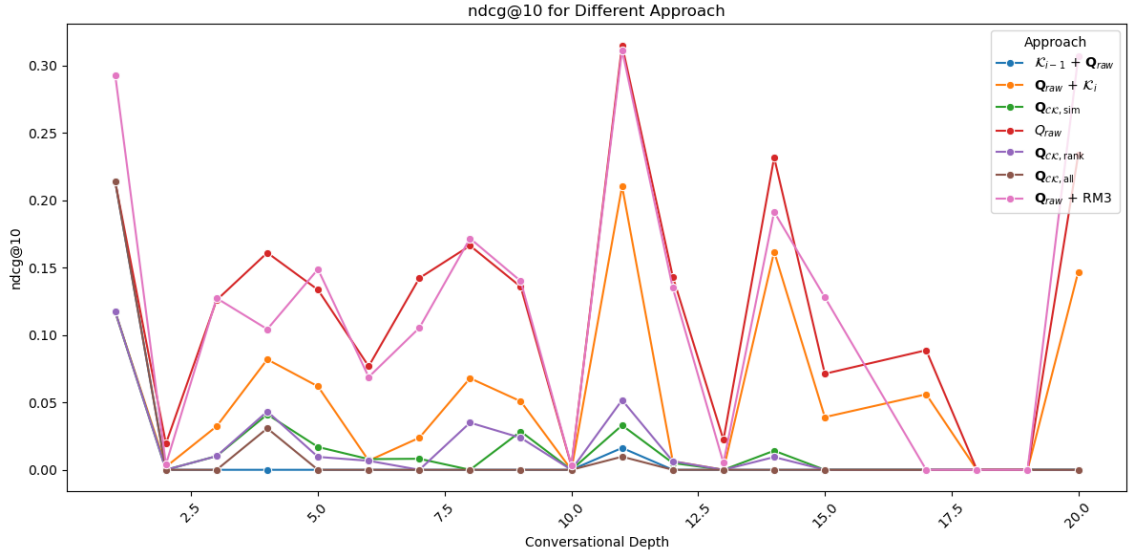


Figure 5.9: Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval.

iKAT 2023. Figure 5.9 presents the nDCG@10 performance of the selected query-enrichment approaches for a sparse (BM25) retrieval system across different conversation lengths (depths) in the dataset. The results show that enriching the query at the turn-level or conversational-level leads to a significant decrease in performance in comparison to the Q_{raw} and $Q_{raw} + RM3$ approaches throughout the conversation. A significant drop in performance is also observed at depth 2 for all the approaches.

5.4 Summarization-Based Knowledge Integration

In this approach, we propose creating a summary of the contents of the retrieved documents and treating this summary as the knowledge that the user has gained during the interaction at that turn in the dialogue. This summary reflects or tries to capture the information the user has been exposed to and is most likely to remember since it is the focus of the documents and can be potentially used to refine or expand the user’s queries in subsequent turns in the dialogue. This is intended to help the system provide more relevant results by leveraging the user’s assumed growing understanding of the topic.

Following the general framework as we described in Figure 5.3, in the following section, we will look into the specifics of the knowledge extractor component, which in this case will be the summary of the retrieved documents at each turn in the dialogue.

Knowledge Extractor. For our experiments, knowledge was extracted using a summarization based approach. The resulting summary is then treated as the knowledge that the user has gained at that turn in the conversation. To extract the summary, we employ a fine-tuned version of the Pegasus model (J. Zhang et al. 2020), specifically adapted for the XSum dataset (Narayan, Cohen, and Lapata 2018), which consists of a single-sentence summary of news articles. Pegasus is an encoder-decoder transformer model for abstractive summarization trained to generate a new sentence (summary) capturing the essential information and not extracting sentences present in the input text. The Pegasus model used has 12 layers for the encoder and decoder, and each layer has 16 attention heads. The hidden size of the model is 1024.

5.4.1 Various Formulations of Query Enrichment

We describe the various formulations of integrating the knowledge (summary) and enriching the user query, and further investigate their effectiveness in CIR following the discussion in Section 5.3.1.

1. $\mathbf{Q}_{raw}$: User-issued query at each turn in the conversation.

$$q = (q_n)$$

2. $\mathbf{Q}_{raw} + \text{RM3}$: User-issued query utterance with RM3 pseudo-relevance feedback (Lavrenko and W. Bruce Croft 2001) based relevance-feedback approach for query expansion.

$$q = (q_n \odot \{\text{RM3expansion}\})$$

Turn Knowledge ($\mathcal{K}$). Turn-level knowledge extracted from retrieved relevant documents at each turn in the conversational dialogue.

1. $\mathbf{Q}_{raw} + \mathcal{K}_i$: User-issued query utterance concatenated with knowledge extracted at i th turn for query expansion.

$$q = q_i \odot \mathcal{K}_i$$

2. $\mathcal{QR}(\mathbf{Q}_{raw} + \mathcal{K}_i)$: User-issued query utterance with knowledge extracted at i th turn and then passed through a query rewriting stage ($\mathcal{QR}$)

$$q = \mathcal{QR}(q_i \odot \mathcal{K}_i)$$

3. $\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$: Knowledge extracted from the previous turn in the conversation, concatenated with the user-issued query utterance in the i th turn.

$$q = (\mathcal{K}_{i-1} \odot q_i)$$

4. $\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw})$: Knowledge extracted from the previous turn in the conversation, concatenated with the user-issued query utterance in the i th turn and then passed through a query rewriting stage ($\mathcal{QR}$).

$$q = \mathcal{QR}(\mathcal{K}_{i-1} \odot q_i)$$

5. $\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$: Knowledge extracted from the previous turn in the conversation with the user-issued query utterance in the current turn, then passed to the query rewriting stage ($\mathcal{QR}$) followed by the relevance-feedback approach for query expansion.

$$q = \mathcal{QR}(\mathcal{K}_{i-1} \odot q_i) \odot \{\text{RM3expansion}\}$$

Conversational Knowledge ($\mathcal{CK}$). Conversational-level knowledge accumulated throughout the dialogue in the search process.

6. $\mathbf{Q}_{\mathcal{CK},\text{all}}$: User-issued query utterance with all available conversational knowledge and then passed to the query rewriting stage ($\mathcal{QR}$).

$$q = \mathcal{QR}(\mathcal{CK}_{1:n-1} \odot q_i)$$

7. $\mathbf{Q}_{\mathcal{CK},\text{sim}}$: User-issued query utterance with selected conversational knowledge (based on similarity to the i th user-issued query utterance) then passed to the query rewriting stage ($\mathcal{QR}$). We consider the top-3 similar conversational knowledge.

$$q = \mathcal{QR}(\mathcal{CK}_{\text{sim}} \odot q_i)$$

8. $\mathbf{Q}_{\mathcal{CK},\text{rank}}$: User-issued query utterance with selected conversational knowledge (based on relevance to the user-issued query utterance) is then passed to the query rewriting stage ($\mathcal{QR}$). We consider the top-3 ranked conversational knowledge based on the BM25 ranking.

$$q = \mathcal{QR}(\mathcal{CK}_{\text{rank}} \odot q_i)$$

5.4.2 Results and Discussion

We present our results for summarization-based knowledge integration for query enrichment at both the turn and conversational levels in the dialogue. The results

Table 5.3: Performance of a Sparse Retrieval (BM25) System TREC CAsT 2019, 2020. **Bold** and underline indicate the best result and the second best result. α and β denote statistically significant improvements over the baselines $\mathbf{Q}_{raw}$ and $\mathbf{Q}_{raw} + \text{RM3}$, respectively. (Paired t-test at $p < 0.05$)

Query	CAsT 2019			CAsT 2020		
	nDCG@10	MRR	MAP@1000	nDCG@10	MRR	MAP@1000
Baselines						
$\mathbf{Q}_{raw}$	0.1479	0.3299	0.1163	0.0774	0.1786	0.0555
$\mathbf{Q}_{raw} + \text{RM3}$	0.1732	0.3493	0.1407	0.0892	0.1873	0.0704
Turn Knowledge						
$\mathbf{Q}_{raw} + \mathcal{K}_i$	0.1439	0.3107	0.0977	0.0805	0.1831	0.0537
$\mathcal{QR}(\mathbf{Q}_{raw} + \mathcal{K}_i)$	0.0728	0.1678	0.0579	0.0288	0.0772	0.0214
$\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$	0.1108	0.2622	0.0761	0.0504	0.1231	0.0374
$\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw})$	<u>0.2016$^\alpha$</u>	<u>0.4288$^{\alpha\beta}$</u>	<u>0.1606$^\alpha$</u>	<u>0.1008$^\alpha$</u>	<u>0.2294$^\alpha$</u>	<u>0.0748$^\alpha$</u>
$\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$	0.2409$^{\alpha\beta}$	0.4612$^{\alpha\beta}$	0.2081$^{\alpha\beta}$	0.1193$^{\alpha\beta}$	0.2529$^{\alpha\beta}$	0.0944$^{\alpha\beta}$
Conversational Knowledge						
$\mathbf{Q}_{CK,all}$	0.1535	0.3402	0.1279	0.0830	0.1793	0.0632
$\mathbf{Q}_{CK,sim}$	0.1890 $^\alpha$	0.4002 $^\alpha$	0.1552 $^\alpha$	0.0863	0.1807	0.0654
$\mathbf{Q}_{CK,rank}$	0.1992 $^\alpha$	0.4050 $^\alpha$	0.1580 $^\alpha$	0.0981 $^\alpha$	0.2090	0.0730 $^\alpha$

correspond to the configuration that achieved the best performance in the query expansion baseline ($\mathbf{Q}_{raw} + \text{RM3}$), which used 10 documents. We also experimented with generating shorter summaries, using a minimum of 10 tokens and a maximum of 20 tokens, but found that summaries with a minimum length of 20 and a maximum of 40 tokens produced better results.

CAsT Datasets. Table 5.3 presents the performance of all query enrichment approaches on the TREC CAsT 2019 and 2020 datasets using a sparse retrieval (BM25). Overall, the results show that incorporating knowledge improves retrieval quality over the baseline queries. ($\mathbf{Q}_{raw} + \text{RM3}$) consistently improves upon the performance of the raw queries ($\mathbf{Q}_{raw}$). For example, on CAsT 2019, nDCG@10 increases from 0.1479 to 0.1732, and MAP rises from 0.1163 to 0.1407. Similar improvements appear in CAsT 2020. Directly utilizing the current turn’s knowledge $\mathcal{K}_i$ in the user query does not lead to improvements.

Applying the query rewriting model $\mathcal{QR}(\mathbf{Q}_{raw} + \mathcal{K}_i)$ at the turn-level further decreases performance. In contrast, using the previous turn’s knowledge ($\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$) is more effective with the query rewriting approach instead of the query expansion approach. $\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw})$ produces the second-best results across all metrics for both datasets. This indicates that knowledge integration using the

Table 5.4: Performance of a Sparse Retrieval (BM25) System TREC iKAT 2023. **Bold** and underline indicate the best result and the second best result. No statistically significant difference over the baselines $\mathbf{Q}_{raw}$ and $\mathbf{Q}_{raw} + \text{RM3}$. (Paired t-test at $p < 0.05$)

Query	iKAT 2023		
	nDCG@10	MRR	MAP@1000
<i>Baselines</i>			
$\mathbf{Q}_{raw}$	0.1302	0.2828	0.0505
$\mathbf{Q}_{raw} + \text{RM3}$	<u>0.1052</u>	0.2521	<u>0.0356</u>
<i>Turn Knowledge</i>			
$\mathbf{Q}_{raw} + \mathcal{K}_i$	0.0561	0.1842	0.0186
$\mathcal{QR}(\mathbf{Q}_{raw} + \mathcal{K}_i)$	0.010	0.0292	0.0025
$\mathcal{K}_{i-1} + \mathbf{Q}_{raw}$	0.0083	0.0263	0.0042
$\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw})$	0.0570	0.1484	0.0240
$\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$	0.0738	0.1729	0.0327
<i>Conversational Knowledge</i>			
$\mathbf{Q}_{CK,all}$	0.0060	<u>0.2544</u>	0.0038
$\mathbf{Q}_{CK,sim}$	0.0050	0.0155	0.0029
$\mathbf{Q}_{CK,rank}$	0.0024	0.0075	0.0014

previous conversational turn is generally more useful than current-turn knowledge.

When RM3 expansion is added on top of $\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw})$, the system achieves the best overall performance. For CAsT 2019, the model reaches $\text{nDCG@10} = 0.2409$ and $\text{MAP} = 0.2081$, outperforming all other methods. A similar pattern is observed in CAsT 2020. These improvements suggest that combining structured previous-turn knowledge with both query rewriting and pseudo-relevance feedback is an effective approach.

The conversational-level knowledge methods, which integrate knowledge gathered across the dialogue, also yield improvements over the baselines ($\mathbf{Q}_{raw}$ and ($\mathbf{Q}_{raw} + \text{RM3}$)). Among these, the similarity-based selection ($\mathbf{Q}_{CK,sim}$) and rank-based selection ($\mathbf{Q}_{CK,rank}$) performed better than simply using all available conversational knowledge ($\mathbf{Q}_{CK,all}$). The ranking-based variant achieves results close to the turn-knowledge methods, reaching $\text{nDCG@10} = 0.1992$ for CAsT 2019 and 0.0981 for CAsT 2020. These findings show that selectively incorporating knowledge is beneficial.

iKAT Dataset. Table 5.4 reports the performance of different query formulations on the TREC iKAT 2023 dataset. Interestingly, $\mathbf{Q}_{raw}$ achieves the best overall performance among all query enrichment approaches. It reaches $\text{nDCG@10} = 0.1302$, $\text{Recall@100} = 0.2828$, and $\text{MAP} = 0.0505$. Adding RM3 pseudo-relevance feedback reduces performance, especially on nDCG@10 and MAP . This suggests that RM3 introduces noisy expansion terms in this dataset, which affects the ranking of the top documents.

All turn-knowledge approaches perform worse. Simply utilizing the current turn’s knowledge $\mathcal{K}_i$ reduces performance. This performance reduction indicates that the extracted knowledge adds a large amount of irrelevant information. Applying the query rewriting model $\mathcal{QR}(\mathbf{Q}_{raw} + \mathcal{K}_i)$ results in an even more substantial decline.

Using previous-turn knowledge $\mathcal{K}_{i-1}$ shows a similar pattern. Query expansion performs poorly, while query rewriting helps but still remains below the baselines. The best performing turn-knowledge variant is $\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$, which achieves $\text{nDCG@10} = 0.0738$ and $\text{MAP} = 0.0327$. Although this is the strongest result within the turn-knowledge group, it still falls short of the raw user query baseline.

The conversational-knowledge methods also fail to outperform the baseline. Using all accumulated conversational knowledge ($\mathbf{Q}_{\mathcal{K},\text{all}}$) results in a higher recall (0.2544), but the nDCG@10 and MAP remain extremely low. This suggests that while ($\mathbf{Q}_{\mathcal{K},\text{all}}$) approach retrieves more potentially relevant documents, they are not ranked near the top. The similarity-based and rank-based selection strategies ($\mathbf{Q}_{\mathcal{K},\text{sim}}$ and $\mathbf{Q}_{\mathcal{K},\text{rank}}$) perform poorly across all metrics. These findings imply that conversational knowledge is not directly beneficial in the iKAT dataset and that the additional context that is used for the reformulation of the query acts as noise.

These results give the strongest answer to **RQ2** in this chapter. On the CAsT datasets, combining previous-turn knowledge, query rewriting, and RM3 ($\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$) gives the best results. It performs better than using RM3 alone. For example, on CAsT 2019, nDCG@10 improves from 0.1732 to 0.2409, and MAP

improves from 0.1407 to 0.2081. This shows that summarized user knowledge can improve conversational search. However, the improvement comes from using knowledge, query rewriting, and RM3 together, so it is difficult to measure the effect of the knowledge summary alone. Also, on the iKAT dataset, none of the knowledge-based methods performs better than the raw user query. This shows that knowledge modeling works better for simple conversations like CAsT than for more complex conversations like iKAT. Compared with keywords, summarization gives a clearer and more meaningful representation of the user’s knowledge, making it easier for the query rewriting model to use. This suggests that richer knowledge representations can further improve conversational search, as discussed in Chapter 6.

Conversation Topic-Wise Analysis

In this section, we examine the retrieval effectiveness at a topic level to highlight how different approaches to summarization-based query enrichment perform.

CAsT 2019. Figure 5.10 presents the topic-wise nDCG@10 performance for the selected query-enrichment approaches for a sparse (BM25) retrieval system. For certain topics, e.g. Topic 77, $\mathcal{Q}_{raw} + \text{RM3}$ performs the best. Turn-level knowledge enrichment ($\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$) generally performed the best across majority of the topic (Topics 31, 32 68 and 61), while for Topics 34, 49, 59 and 56 the conversational-knowledge reformulations ($\mathbf{Q}_{\mathcal{CK}}$) outperforms, indicating that enriching a query with knowledge accumulated over the conversational history can be valuable. Using all conversational knowledge ($\mathbf{Q}_{\mathcal{CK},all}$) performs the worst among the three, suggesting that overly aggressive query enrichment can harm performance. However, selectively incorporating conversational knowledge using rank-based filtering ($\mathbf{Q}_{\mathcal{CK},rank}$) yields more competitive results as compared to similarity-based filtering ($\mathbf{Q}_{\mathcal{CK},sim}$).

CAsT 2020. Figure 5.11 presents the topic-wise nDCG@10 performance for the selected query-enrichment approaches for a sparse (BM25) retrieval system. For

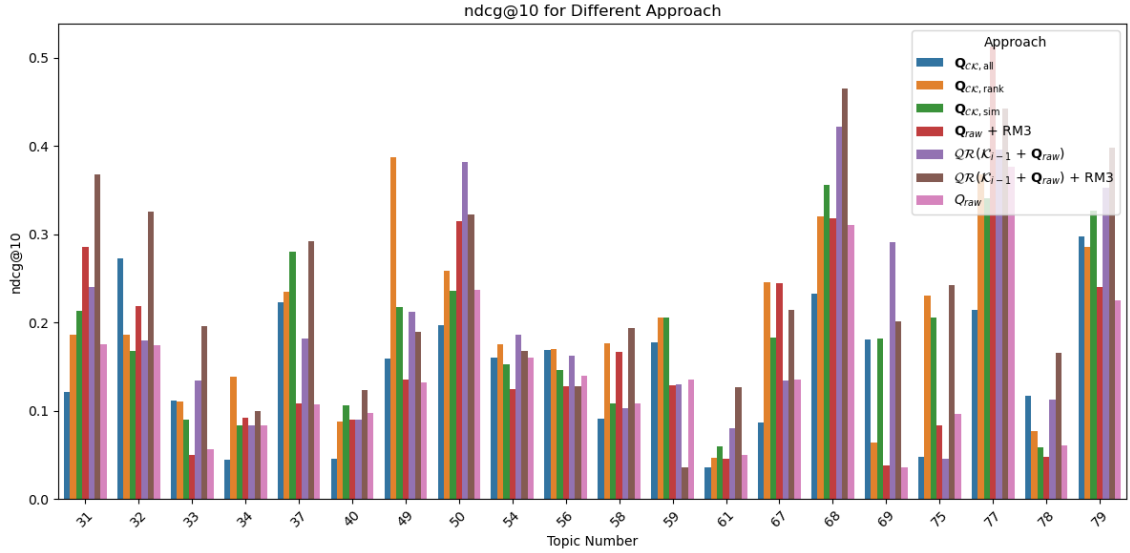


Figure 5.10: Topic-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval.

certain topics (e.g Topic 100 and 97), $Q_{raw} + RM3$ performed the best. Turn-level knowledge enrichment ($Q\mathcal{R}(K_{i-1} + Q_{raw}) + RM3$) generally performed the best across majority of the topic (Topics 101, 104, 82, 85 and 93), while for others the conversational-knowledge reformulations (Q_{cc}) outperforms (Topics 102, 81, 87 and 99), indicating that enriching a query with knowledge accumulated over the conversational history can be valuable.

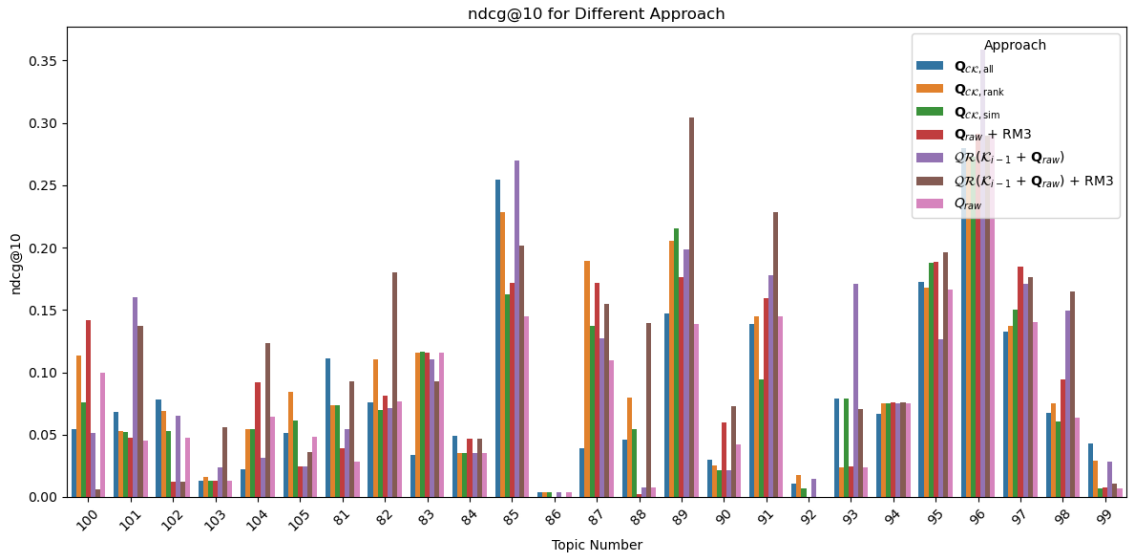


Figure 5.11: Topic-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval.

iKAT 2023. Figure 5.12 presents the topic-wise nDCG@10 performance for the selected query-enrichment approaches for a sparse (BM25) retrieval system. Generally, none of the approaches for query enrichment, whether at the turn-level or at the conversational level, was able to outperform Q_{raw} or $Q_{raw} + RM3$ approaches throughout the topics. However, in the case of Topics 18-2 and 21-1 turn-level knowledge, $(QR(K_{i-1} + Q_{raw}) + RM3)$ performed the best. Conversational-knowledge reformulations failed to perform in comparison. Topics 11-2 and 17-3 posed a significant challenge. Irrespective of the various query enrichment formulations, the system was unable to retrieve any relevant documents in the top 10.

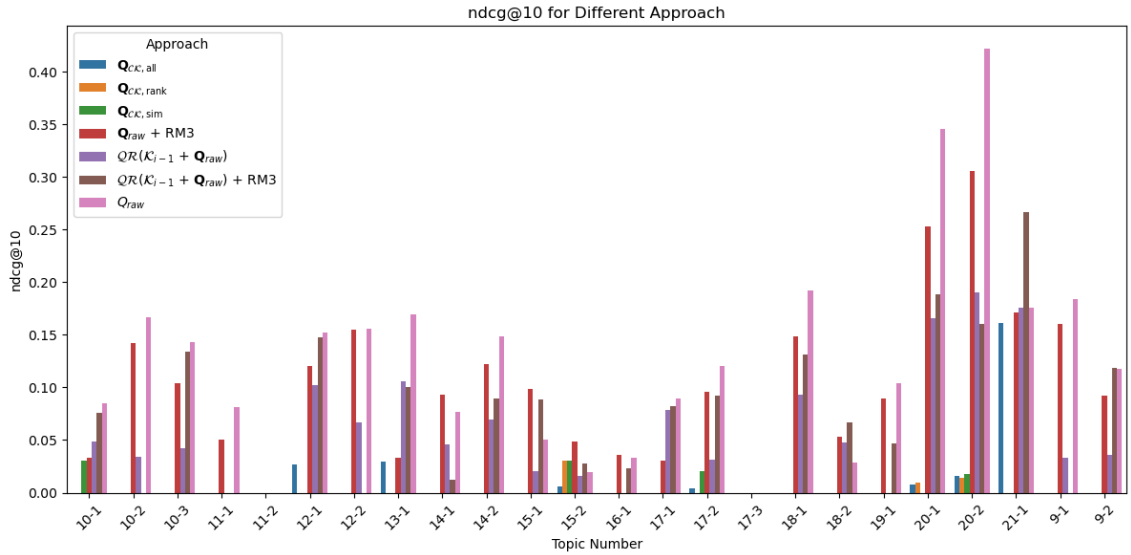


Figure 5.12: Topic-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval.

Conversation Depth-Wise Analysis

In this section, we examine retrieval effectiveness across different conversational depths to highlight how different approaches to keyword-based query enrichment perform.

CAsT 2019. Figure 5.13 presents the nDCG@10 performance of the selected query-enrichment approaches for a sparse (BM25) retrieval system across different conversation lengths (depths). A gradual decline in performance is observed for all approaches as conversational depth increases. The turn-level query enrichment

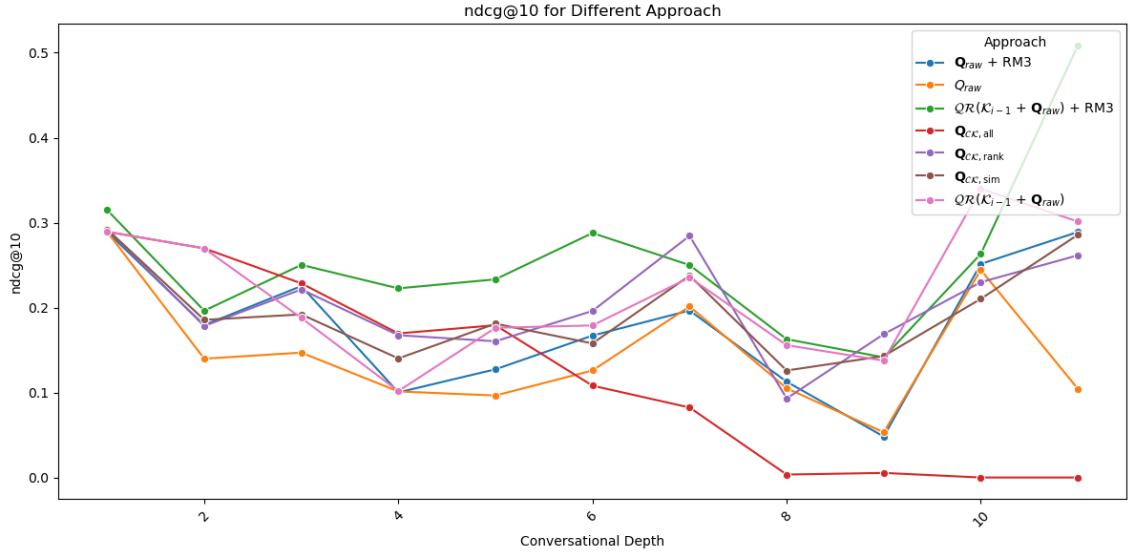


Figure 5.13: Depth-wise ndcg@10 Performance for CAsT 2019 Sparse Retrieval.

$QR(K_{i-1} + Q_{raw}) + RM3$ performed better compared to other approaches. The conversational-level enrichment outperforms the Q_{raw} throughout the conversation depth but is unable to outperform the $QR(K_{i-1} + Q_{raw}) + RM3$ approach.

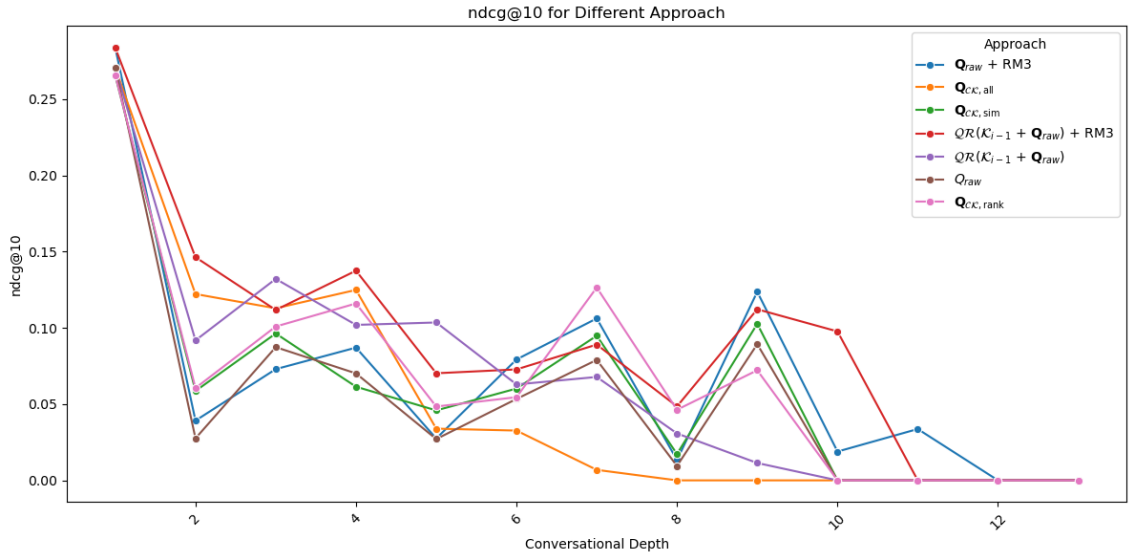


Figure 5.14: Depth-wise ndcg@10 Performance for CAsT 2020 Sparse Retrieval.

CAsT 2020. Figure 5.14 presents the nDCG@10 performance of the selected query-enrichment approaches for a sparse (BM25) retrieval system across different conversation lengths (depths). The results show that all query formulation approaches exhibit a noticeable decline in performance as conversational depth increases. Overall,

the performance of all approaches gradually decreases as the conversation lengthens. A significant drop in performance is observed at depth 2. The turn-level query enrichment $\mathcal{QR}(\mathcal{K}_{i-1} + \mathbf{Q}_{raw}) + \text{RM3}$ performed better compared to other approaches. It was closely followed by the conversation-level query enrichment ($\mathbf{Q}_{\mathcal{CK},all}$) at depths 2, 3 and 4, but the performance degrades significantly beyond depth 4.

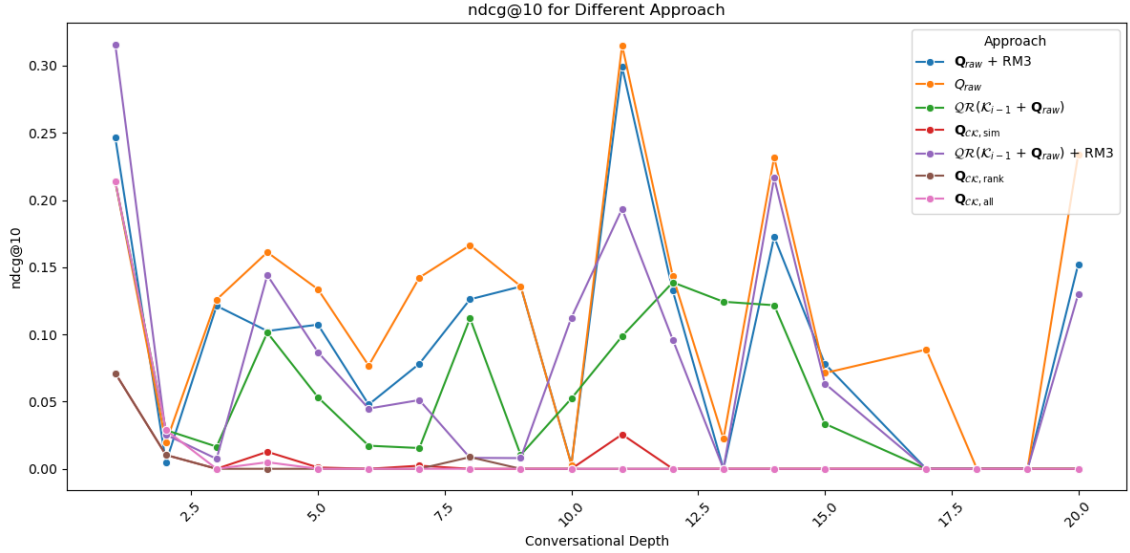


Figure 5.15: Depth-wise ndcg@10 Performance for iKAT 2023 Sparse Retrieval.

iKAT 2023. Figure 5.15 presents the nDCG@10 performance of the selected query-enrichment approaches for a sparse (BM25) retrieval system across different conversation lengths (depths) in the dataset. The results show that enriching the query at the turn-level or conversational-level leads to a significant decrease in performance in comparison to the $\mathbf{Q}_{raw}$ and $\mathbf{Q}_{raw} + \text{RM3}$ approaches throughout the conversation. The turn-level query enrichment is better compared to the conversational-level enrichment. A significant drop in performance is also observed at depth 2 for all the approaches.

5.5 Summary

In this chapter, we presented a framework for integrating knowledge into Conversational Information Retrieval (CIR) systems, addressing the challenge of making

knowledge a central component of information retrieval. By leveraging knowledge accumulated during conversations, the framework aims to improve CIR systems' ability to understand users' information needs, engage in contextually relevant dialogue, and provide more accurate responses. The framework consists of three main components: the ***Knowledge Extractor (KE)***, ***Current Knowledge State (cks)*** and the ***Knowledge Updater (KU)***. The ***KE*** extracts relevant information from user queries, documents, and prior knowledge, while the ***KU*** integrates this knowledge into the current knowledge state, ensuring the system remains up-to-date and comprehensive throughout the interaction represented by the ***cks***.

Furthermore, we explored the enhancement of queries in conversational search by integrating knowledge from retrieved documents. We proposed a keywords-based and summarization-based approach for integrating knowledge to enrich subsequent user queries, and we conducted experiments using the TREC CAsT 2019, 2020 and iKAT 2023 datasets to evaluate its effectiveness. Our results demonstrate that our approach improves the system's retrieval performance in certain scenarios. Additionally, we found that our method can outperform relevance-based feedback techniques, highlighting the potential of a knowledge-based approach to query enrichment in conversational search.

With respect to ***RQ1***, both approaches demonstrate that features of the expected user's developing knowledge can be identified and captured during a conversational search task, through keyword extraction and abstractive summarization of retrieved documents, respectively. The framework's ***Knowledge Extractor*** and ***Knowledge Updater*** components formalise this capture process, constituting the first structured architecture for knowledge modeling within a multi-turn CIR dialogue.

For ***RQ2***, the results show that user knowledge modeling can improve retrieval, but only in some cases. On the CAsT datasets, the summarization-based method, combined with query rewriting and RM3, gives the best results. However, on the more difficult iKAT dataset, none of the knowledge-based methods performs better than the raw user query. The keyword-based method never performs better than

RM3, while the summarization-based method improves results only on CAsT. This shows that how user knowledge is represented is very important. Overall, these results provide partial support for Hypothesis 2. User knowledge can improve conversational search, but better knowledge representations and extraction methods are still needed. These limitations and future work are discussed in Chapter 6.

Chapter 6

Conclusion

In this chapter, we present a summary of our investigation for this research. We describe the key findings of our study. We then revisit the research question introduced in Chapter 1. Following this, we discuss the limitations of our work and identify potential directions for future research and opportunities for further exploration.

The investigations presented in this thesis advance our understanding of how conversational context, historical query turns and evolving user knowledge can be leveraged to improve CIR systems. While these two dimensions were studied separately, their findings are interconnected. Effective retrieval in multi-turn dialogue depends not only on identifying and selecting which prior turns are relevant, but also on understanding how a user’s knowledge state evolves across those turns. Together, these two dimensions point toward a more holistic view of conversational context that future CIR systems must embrace.

6.1 Key Findings

In this PhD work, we examined the role of conversational context within CIR. Specifically, we explored the challenges and opportunities involved in understanding and leveraging conversational context from two key dimensions: (1) conversational history and (2) knowledge. The goal of this thesis is to study and deepen our understanding of these two aspects in CIR by exploring approaches for identifying

related historical turns in the conversational context, and by developing methods for modeling the user’s evolving knowledge throughout the interaction.

6.1.1 Exploring the Challenge of Related Historical Context Identification in CIR

We investigated the effect of historical conversational context in CIR. Our findings show that incorporating historical conversational context is important for improving conversational search, with query rewriting using all prior turns yielding the strongest performance. However, because not all prior turns are related, selective context identification becomes essential. The results of this study were used to propose and evaluate different approaches to selective context identification. Experiments reveal that a hybrid approach of using simple heuristics and semantic similarity signals can improve turn selection by focusing on structurally or semantically important queries, while avoiding using the entire historical conversational context of the dialogue. Large language models (LLMs) are able to identify related historical conversational query turns. However, the retrieval performance of the LLM-identified related turns to contextualize and disambiguate the current turn does not improve compared to the similarity-based approach, as shown in Table 4.7. Overall, the results demonstrate that a hybrid approach leveraging a similarity signal to identify and select related historical conversational query context is the most effective approach with respect to the retrieval effectiveness across datasets and retrieval systems.

6.1.2 User Knowledge in CIR

We proposed a framework for modeling a user’s evolving knowledge in conversational search. Based on this framework, we investigated approaches for knowledge extraction and integration in the form of query enrichment. We investigated a keyword-based knowledge integration approach and a summarization-based knowledge integration. Our findings show that the keyword-based approach is less effective than the summarization-based approach, and in highly complex tasks, the

knowledge integration approaches fail to outperform raw queries. User knowledge modeling can enhance conversational search, but its effectiveness is conditional. Overall, our framework demonstrates that user knowledge modeling has potential for enhancing CIR.

Together, these findings reveal that no single approach, whether historical query turns or extracted knowledge, is sufficient on its own. The most effective strategies are those that selectively and adaptively combine contextual signals. Historical context is most useful when relevant turns are identified, and user knowledge is most beneficial when it is summarized and selectively integrated. Suggesting that future CIR systems should pursue a unified contextual model that jointly represents dialogue history and user knowledge state.

6.2 Research Questions Addressed

We investigated the research questions introduced in Chapter 1. In this section, we revisit the questions and summarize the findings from our investigations.

6.2.1 To what extent can the incorporation of historical conversational context impact the overall retrieval effectiveness of a conversational search system?

In this investigation, we studied how incorporating historical conversational context affects the performance of CIR systems. Our findings showed that including information from previous turns improves retrieval effectiveness. User-issued queries ($\mathbf{Q}_{raw}$) performed the worst, confirming that conversational queries often lack enough information on their own and require context from prior turns to be properly understood.

We found that the performance of the system generally increased as more previous turns were included. We also found that the query-rewriting consistently outperformed expansion-based strategies, with the best results achieved when all

previous queries were used ($\mathcal{Q}_{all} + \text{cqr}$).

Heuristic approach of selecting previous k-turns to incorporate conversational context to contextualize query was helpful, but it did not perform as well compared to using all the available conversational context. However, not all previous turns are equally related and relevant (Aliannejadi, Chakraborty, et al. 2020; Jeffrey Dalton, Xiong, and Callan 2021). Further, our analysis showed that, while the most recent previous turns are the most relevant in the dialogue (Figure 4.1), in longer conversations, relevant information can appear earlier in the conversation. This highlights the challenge of effectively selecting relevant turns, especially as the length of conversational context grows. Overall, incorporating historical conversational query context improves retrieval performance.

6.2.2 Can we identify and select the historical conversational context in the dialogue so as to be able to support the user in reaching their information goal?

In this investigation, we examined similarity-based, hybrid and LLM-based approaches to identifying related historical conversational query turns. Our findings show that it is both possible and effective to select related historical conversational context for conversational search. The analysis of the heuristic approach demonstrates that the approach of selecting the first turn and the previous 3 turns $\mathcal{Q}_{Turn1prev3}$ is more effective than selecting all the previous queries $\mathcal{Q}_{all}$ as a method of identifying related historical conversational query turns (Table 4.6). Similarly, the semantic similarity signal can identify semantically relevant, but lexically varied historical conversational context. Hybrid approach to identify related historical turns ($\mathcal{STM}_{low}\mathcal{Q}_{prev3}$ and $\mathcal{STM}_{low}\mathcal{Q}_{Turn1prev3}$) performed comparably to selecting all the previous queries $\mathcal{Q}_{all}$, avoiding inclusion of unnecessary information while preserving retrieval effectiveness.

LLM-based experiments show that large language models substantially outperform all the other approaches to identifying related historical conversational query context

(Table 4.6).

The retrieval performance that integrates these selection approaches (Table 4.7) demonstrates that the hybrid approach is the strongest performer across all datasets (CAST 2019, CAST 2020, iKAT 2023). The hybrid method consistently outperforms the other approaches, suggesting it is possible to identify historical conversational context while achieving retrieval performance comparable to using the entire historical conversational context.

6.2.3 Can features of the expected user’s developing knowledge be identified and captured during a conversational search task?

As discussed in Chapter 5, understanding and leveraging a user’s evolving knowledge during a conversational search can potentially be utilized for effective information retrieval. We presented a framework that continuously captures and updates a user’s knowledge throughout a search session, demonstrating how such modeling can enhance the system’s ability to support and guide the user’s information-seeking process.

Through the proposed framework, we introduced a method which seeks to identify and capture expected features of a user’s developing knowledge during a conversational search task. This is achieved by continuously extracting, updating, and leveraging information obtained from both the dialogue context and the documents encountered throughout the search process. Recent work has shown that modeling users’ knowledge during search is both feasible and informative (Câmara, El-Zein, and da-Costa-Pereira 2022). Approaches based on keywords, named entities, knowledge graphs, and large language models have each demonstrated the ability to reflect a user’s understanding during a session (Câmara, El-Zein, and da-Costa-Pereira 2022; El Zein, Câmara, et al. 2023; Nasser et al. 2024). Building on these developments, our framework integrates a Knowledge Extractor (KE) and Knowledge Updater (KU) into the CIR workflow, enabling the system to identify concepts, entities, and

relationships from queries, retrieved documents, and accumulated knowledge during the conversational dialogue. The extracted knowledge is then incorporated into the retrieval system so that the system maintains a representation of the user’s evolving information state. By capturing knowledge, the system seeks to refine the user’s query into a more complete and contextually accurate expression of their information need.

These mechanisms aim to allow the model to leverage the user’s growing knowledge to improve retrieval effectiveness in the subsequent turns in the dialogue. Despite this potential, existing CIR research has not sought to integrate the evolving user knowledge during the conversation, leaving a significant opportunity for advancement. Our framework seeks to address this gap. By structuring CIR around continuous extraction, incremental updating, and flexible knowledge representations, we show how a CIR system can adapt to a user’s changing understanding in real time. While our focus is on expected user knowledge that emerges within a single conversational session, the framework establishes a foundation for future efforts that might incorporate longer-term user knowledge models.

6.2.4 Can we exploit the model of the user’s knowledge to improve the effectiveness of conversational search?

In this investigation, we studied keyword-based and summarization-based approaches to integrating knowledge and enriching conversational user queries. The results across both the keyword-based and summarization-based experiments provide a nuanced answer to this question. Overall, the findings show that user-knowledge modeling can improve conversational retrieval effectiveness under specific conditions, but its success depends on how the knowledge is selected and incorporated.

In the keyword-based experiments, similarity-based and rank-based conversational knowledge selection narrowed the gap to the RM3 approach, although they did not exceed it. In the summarization-based experiments, utilizing previous-turn knowledge, when processed through a rewriting model, produces the strongest gains.

This combination achieved the best overall performance on both CAsT datasets. These findings demonstrate that summarized and selective knowledge integration, especially when derived from earlier conversational turns, can be effectively exploited to improve retrieval.

Summarization-based knowledge is more promising than a keyword-based approach. However, in more complex conversational settings, such as in the iKAT dataset, knowledge integration methods fail to outperform the raw user queries, indicating that richer, semantically grounded knowledge representations are needed to adequately capture the complexity of user knowledge in search tasks. This points to a pressing challenge of developing knowledge representations that are expressive enough to generalize across conversational settings of varying complexity in CIR.

6.3 Limitations and Future Directions

While this thesis advances our understanding of historical conversational context and user knowledge modeling in CIR, there are several limitations of this work. These limitations also highlight promising avenues for future research.

6.3.1 Limitations

This work focuses primarily on exploiting historical conversational query context to improve conversational context understanding. We do not explore incorporating the contextual information retrieved from documents or system results from earlier turns in the dialogue for conversational context understanding. However, we explore this avenue in our investigation of incorporating user knowledge in conversational search. While user-issued conversational query turns provide important information for interpreting intent and resolving ambiguity, they capture only one dimension of the evolving interaction. Retrieved documents often provide additional information that can improve the user’s understanding of their information need and influence their subsequent behaviour. Capturing this information can improve query understanding

by revealing how the user’s information need develops across turns. Integrating such information from the retrieved document could therefore enhance the system’s ability to interpret queries more accurately, especially in complex or exploratory information-seeking tasks. By not considering retrieved documents and focusing only on conversational historical query turns, the context modeling examined in this thesis remains limited. This reduces the system’s opportunity to leverage additional contextual information through interaction with the retrieved results. As a result, the findings presented here may underrepresent the potential benefits of a more holistic approach that jointly models both conversational historical query and retrieved documents in the conversational turns.

The conversational scenarios studied in this work do not account for mixed-initiative interactions (Sekulić, Aliannejadi, and Crestani 2022), in which both the user and the system can take the lead in the dialogue, or for clarification-based search behavior (Y. Yuan et al. 2024), where the system, instead of retrieving documents, attempts to elicit additional information from the user by posing a question. Real-world conversational search involves frequent clarification questions (Sekulić, Aliannejadi, and Crestani 2021), preference elicitation (Kostric, Balog, and Radlinski 2024) and mixed-initiation by the participants. By limiting the investigation to user-initiated turns as set out in these datasets, the generalizability of the findings to richer conversational settings is limited.

The approaches investigated for identifying related historical context relied largely on simple semantic similarity signals and heuristic indicators. While these methods proved effective, they represent relatively simple strategies for context selection. Although LLM-based context identification and selection were explored, the work evaluates only a small sample of LLMs, and more advanced or larger contemporary models were not included due to resource constraints. Richer LLM modelling of search conversation may yield improved methods of context modelling and knowledge capture.

In the context of user knowledge modeling, this work employed simple knowledge

extraction techniques, primarily keyword-based and summary-based approaches. Furthermore, we made the simplifying assumption that the user interacts with the top-k retrieved documents, and that the system should extract knowledge from all of these documents by default. This assumption will not reflect actual user behaviour, where users generally view or engage with only a subset of retrieved results, with unequal focus on the documents which they examine. This simplification also limits the expressiveness of the extracted knowledge, since a behaviourally grounded extraction process would likely yield more targeted and meaningful knowledge representations.

This work focuses exclusively on enriching queries with extracted knowledge rather than modeling the knowledge actually gained by the user. Our framework captures knowledge in the search interaction, but does not measure or attempt to truly predict the user’s internal knowledge state. Moreover, the investigation is limited to knowledge emerging within a single conversational session, without accounting for users’ prior domain knowledge or cross-session learning, both of which significantly influence search behaviour and outcomes.

The experiments in this thesis were conducted on a small number of established conversational search datasets, primarily the TREC CAsT 2019, CAsT 2020, and iKAT 2023 datasets. While these are widely used evaluation collections in the CIR community, they represent a narrower landscape of conversational search. These datasets were constructed under specific protocols, with predefined topic structures, and manually assessed relevance judgements. As a result, the conversational scenarios may not fully reflect the real-world information-seeking dialogues. The findings reported in this thesis should therefore be interpreted with this in mind. Future work should evaluate the proposed methods on a broader and more diverse set of benchmarks, including datasets from real user interactions, to more rigorously assess their generalizability.

Several of the approaches investigated in this thesis involve parameters whose values were selected theoretically or based on prior work, including similarity thresh-

olds for hybrid context selection, the number of previous turns k , and the number of top-retrieved documents n from which user knowledge is extracted. The robustness of the reported findings to variation in these parameters across different datasets and retrieval settings has not been exhaustively investigated. For instance, the similarity threshold used in the hybrid selection approach may not be optimal. Similarly, the assumption that users interact with all top- k retrieved documents may overstate actual user behaviour. These parameter dependencies suggest that the proposed methods may require dataset-specific or task-specific tuning, limiting their generalizability. Future work should develop adaptive or self-tuning variants of the proposed approaches that reduce reliance on manually set thresholds.

6.3.2 Future Directions

Future research can extend this thesis in several important and interconnected directions. A key finding of this investigation is the need for richer, more structured representations of conversational context. A key direction for future research is to move beyond treating historical query context and user knowledge as separate signals, and instead develop unified context models that represent both simultaneously. Such a model could dynamically weight historical turns and accumulated knowledge based on their relevance to the current query, enabling a more holistic understanding of the user’s evolving information need.

The findings of this thesis demonstrate that summarisation-based knowledge integration outperforms keyword-based approaches, yet both remain limited in expressiveness. Future research should investigate more structured knowledge representations, including Named Entity Recognition (NER), knowledge graphs, and LLM-derived concept models. Knowledge graphs, in particular, offer the ability to capture relationships between entities across turns, which could substantially improve the system’s ability to model how user understanding evolves. LLM-based knowledge extraction, using models with stronger reasoning capabilities, could further enhance the quality and coverage of extracted knowledge. Importantly, knowledge integration

strategies should be evaluated not only on retrieval effectiveness but also on how accurately they reflect the user’s actual knowledge state.

This thesis assumed that users interact with all top-k retrieved documents, which does not reflect realistic search behaviour. Future work should incorporate implicit feedback signals, such as dwell time and clicks, to identify which documents a user actually engages with, and restrict knowledge extraction accordingly. This would yield more targeted and meaningful knowledge representations and bring knowledge modeling closer to real-world CIR scenarios.

Rather than simply enriching queries with extracted knowledge, future research should investigate how to estimate the knowledge actually gained by the user across turns. This would allow CIR systems to adapt not only to what information the user has encountered, but to how their understanding has developed as a result. Extending this further to cross-session settings, where a user’s prior domain knowledge and past search history are incorporated, would enable substantially more personalised and adaptive conversational search systems.

Developing methods that handle clarification requests and system-initiated suggestions can make context selection and query rewriting more robust and adaptive to real conversational dynamics. Both context modeling and knowledge integration could benefit from clarification turns, which provide explicit signals about the user’s information need and current understanding.

This thesis evaluated a limited sample of LLMs due to resource constraints. Future work should assess more advanced and domain-specialised models to determine whether stronger LLM reasoning yields further improvements in context identification, knowledge extraction, and query rewriting. Such an evaluation would also help clarify the conditions under which LLM-based approaches offer advantages over simpler similarity-based methods.

Appendix A

LLM Prompt details

A.1 Prompts for the Phi-LLM Model

```
Prompt: <|begin_of_text|><|start_header_id|>system<|end_header_id|>
You are a helpful assistant.
<|start_header_id|>user<|end_header_id|> Determine whether the
current conversation turn is related to the previous one.
Previous Turn: {prev_turn}
Current Turn: {current_turn}
Answer with 'related' or 'not related'.
<|start_header_id|>assistant<|end_header_id|>
```

Figure A.1: Zero-shot Prompt for Phi-LLM

```
Prompt: <|begin_of_text|><|start_header_id|>system<|end_header_id|>
You are a helpful assistant.
<|start_header_id|>user<|end_header_id|> Determine whether the
current conversation turn is related to the previous one.
Previous Turn: Do you like horror movies?
Current Turn: I just watched The Conjuring yesterday.
<|start_header_id|>assistant<|end_header_id|>related
<|start_header_id|>user<|end_header_id|>
Previous Turn: {prev_turn}
Current Turn: {current_turn}
<|start_header_id|>assistant<|end_header_id|>
```

Figure A.2: One-shot Prompt for Phi-LLM

```

Prompt: <|begin_of_text|><|start_header_id|>system<|end_header_id|>
You are a helpful assistant.
<|start_header_id|>user<|end_header_id|> Determine whether the
current conversation turn is related to the previous one. Respond
with 'related' or 'not related'.
Previous Turn: What time does the concert start?
Current Turn: It's supposed to rain tonight.
<|start_header_id|>assistant<|end_header_id|>not related
<|start_header_id|>user<|end_header_id|>
Previous Turn: I really enjoyed that movie last night.
Current Turn: Yeah, the acting was fantastic.
<|start_header_id|>assistant<|end_header_id|>related
<|start_header_id|>user<|end_header_id|>
Previous Turn: Do you like Indian food?
Current Turn: I love chicken tikka masala.
<|start_header_id|>assistant<|end_header_id|>related
<|start_header_id|>user<|end_header_id|>
Previous Turn: Are you free tomorrow afternoon?
Current Turn: I need to finish a report by Friday.
<|start_header_id|>assistant<|end_header_id|>not related
<|start_header_id|>user<|end_header_id|>
Previous Turn: {prev_turn}
Current Turn: {current_turn}
<|start_header_id|>assistant<|end_header_id|>

```

Figure A.3: Few-shot Prompt for Phi-LLM

A.2 Prompts for the LLama Models

```
Prompt: <|begin_of_text|><|start_header_id|>system<|end_header_id|>
You are a helpful assistant.
<|start_header_id|>user<|end_header_id|> Determine whether the
current conversation turn is related to the previous one.
Previous Turn: {prev_turn}
Current Turn: {current_turn}
Answer with 'related' or 'not related'.
<|start_header_id|>assistant<|end_header_id|>
```

Figure A.4: Zero-shot Prompt for LLama Model

```
Prompt: <|begin_of_text|><|start_header_id|>system<|end_header_id|>
You are a helpful assistant.
<|start_header_id|>user<|end_header_id|> Determine whether the
current conversation turn is related to the previous one.
Previous Turn: Do you like horror movies?
Current Turn: I just watched The Conjuring yesterday.
<|start_header_id|>assistant<|end_header_id|>related
<|start_header_id|>user<|end_header_id|>
Previous Turn: {prev_turn}
Current Turn: {current_turn}
<|start_header_id|>assistant<|end_header_id|>
```

Figure A.5: One-shot Prompt for LLama Model

Prompt: <|begin_of_text|><|start_header_id|>system<|end_header_id|>
 You are a helpful assistant.
 <|start_header_id|>user<|end_header_id|> Determine whether the
 current conversation turn is related to the previous one. Respond
 with 'related' or 'not related'.
 Previous Turn: What time does the concert start?
 Current Turn: It's supposed to rain tonight.
 <|start_header_id|>assistant<|end_header_id|>not related
 <|start_header_id|>user<|end_header_id|>
 Previous Turn: I really enjoyed that movie last night.
 Current Turn: Yeah, the acting was fantastic.
 <|start_header_id|>assistant<|end_header_id|>related
 <|start_header_id|>user<|end_header_id|>
 Previous Turn: Do you like Indian food?
 Current Turn: I love chicken tikka masala.
 <|start_header_id|>assistant<|end_header_id|>related
 <|start_header_id|>user<|end_header_id|>
 Previous Turn: Are you free tomorrow afternoon?
 Current Turn: I need to finish a report by Friday.
 <|start_header_id|>assistant<|end_header_id|>not related
 <|start_header_id|>user<|end_header_id|>
 Previous Turn: {prev_turn}
 Current Turn: {current_turn}
 <|start_header_id|>assistant<|end_header_id|>

Figure A.6: Few-shot Prompt for LLama Model

Bibliography

- Acharya, Praveen, Noriko Kando, and Gareth JF Jones (2024). “A Framework for Knowledge Integration in Conversational Information Retrieval.” In: *EMTCIR/UM-CIR@ SIGIR-AP*.
- Aliannejadi, Mohammad, Zahra Abbasiantaeb, Shubham Chatterjee, Jeffery Dalton, and Leif Azzopardi (2024). “TREC iKAT 2023: The Interactive Knowledge Assistance Track Overview”. In: *arXiv preprint arXiv:2401.01330*.
- Aliannejadi, Mohammad, 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*, pp. 819–829.
- Aliannejadi, Mohammad, Manajit Chakraborty, Esteban Andrés Ríssola, and Fabio Crestani (2020). “Harnessing Evolution of Multi-Turn Conversations for Effective Answer Retrieval”. In: *Proceedings of 2020 Conference on Human Information Interaction and Retrieval (CHIIR)*. CHIIR ’20. Vancouver, British Columbia, Canada.
- Allan, James, Donna Harman, Evangelos Kanoulas, Dan Li, Christophe Van Gysel, and Ellen M Voorhees (2017). “TREC 2017 Common Core Track Overview.” In: *TREC*.
- Allen, Kent, Madeline M Berry, Fred U Luehrs Jr, and James W Perry (1955). “Machine literature searching VIII. Operational criteria for designing information retrieval systems”. In: *American Documentation (pre-1986)* 6.2, p. 93.

- Anand, Avishek, Lawrence Cavedon, Hideo Joho, Mark Sanderson, and Benno Stein (2020). “Conversational search (dagstuhl seminar 19461)”. In: *Dagstuhl reports* 9.11, pp. 34–83.
- Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E Hinton (2016). “Layer Normalization”. In: *arXiv preprint arXiv:1607.06450*.
- Bajaj, Payal, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. (2016). “Ms marco: A human generated machine reading comprehension dataset”. In: *arXiv preprint arXiv:1611.09268*.
- Bates, Marcia J (1989). “The design of browsing and berrypicking techniques for the online search interface”. In: *Online review* 13.5, pp. 407–424.
- Belkin, Nicholas J (1980). “Anomalous states of knowledge as a basis for information retrieval”. In: *Canadian journal of information science* 5.1, pp. 133–143.
- Box, George EP, J Stuart Hunter, and William G Hunter (2005). *Statistics for experimenters: design, innovation, and discovery*. John Wiley & Sons.
- Brooks, H. M. and N. J. Belkin (1983). “Using Discourse Analysis for the Design of Information Retrieval Interaction Mechanisms”. In: *Proceedings of the 6th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR ’83. Bethesda, Maryland: Association for Computing Machinery, pp. 31–47. ISBN: 0897911075. DOI: 10.1145/511793.511800. URL: <https://doi.org/10.1145/511793.511800>.
- Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. (2020). “Language models are few-shot learners”. In: *Advances in neural information processing systems* 33, pp. 1877–1901.
- Buckland, Michael K and Ziming Liu (1995). “History of information science”. In: *Annual Review of Information Science and Technology (ARIST)* 30, pp. 385–416.
- Câmara, Arthur, Dima El-Zein, and Célia da-Costa-Pereira (2022). “RULK: A Framework for Representing User Knowledge in Search-as-Learning”. In.

- Campos, Ricardo, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt (2020). “YAKE! Keyword extraction from single documents using multiple local features”. In: *Information Sciences* 509, pp. 257–289.
- Cao, Guihong, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson (2008). “Selecting good expansion terms for pseudo-relevance feedback”. In: *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 243–250.
- Carpineto, Claudio and Giovanni Romano (2012). “A survey of automatic query expansion in information retrieval”. In: *Acm Computing Surveys (CSUR)* 44.1, pp. 1–50.
- Carterette, Ben, Evangelos Kanoulas, Mark Hall, and Paul Clough (2014). “Overview of the TREC 2014 session track”. In.
- Chung, Junyoung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio (2014). “Empirical evaluation of gated recurrent neural networks on sequence modeling”. In: *arXiv preprint arXiv:1412.3555*.
- Clarke, Charles LA (2019). “WaterlooClarke at the TREC 2019 Conversational Assistant Track.” In: *TREC*.
- Cleverdon, Cyril W (1991). “The significance of the Cranfield tests on index languages”. In: *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 3–12.
- Croft, W. B. and R. H. Thompson (Nov. 1987). “I3R: A New Approach to the Design of Document Retrieval Systems”. In: *J. Am. Soc. Inf. Sci.* 38.6, pp. 389–404. ISSN: 0002-8231.
- Culpepper, J Shane, Fernando Diaz, and Mark D Smucker (2018). “Research frontiers in information retrieval: Report from the third strategic workshop on information retrieval in lorne (swirl 2018)”. In: *ACM SIGIR Forum*. Vol. 52. 1. ACM New York, NY, USA, pp. 34–90.
- Dalton, Jeffrey, Chenyan Xiong, and Jamie Callan (2020). “TREC CAsT 2019: The Conversational Assistance Track Overview”. In: *ArXiv abs/2003.13624*.

- Dalton, Jeffrey, Chenyan Xiong, and Jamie Callan (2021). “CAsT 2020: The Conversational Assistance Track Overview”. In.
- Dalton, Jeffrey, Chenyan Xiong, Vaibhav Kumar, and Jamie Callan (2020). “CAsT-19: A Dataset for Conversational Information Seeking”. In: SIGIR '20. Virtual Event, China: Association for Computing Machinery, pp. 1985–1988. ISBN: 9781450380164. DOI: 10.1145/3397271.3401206. URL: <https://doi.org/10.1145/3397271.3401206>.
- Davis, Mark and L Iancu (2018). “Unicode text segmentation”. In: *Unicode Standard Annex 29*, p. 65.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2018). “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805*.
- Dietz, Laura, Manisha Verma, Filip Radlinski, and Nick Craswell (2017). “TREC Complex Answer Retrieval Overview.” In: *TREC*.
- Dong, Li, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon (2019). “Unified language model pre-training for natural language understanding and generation”. In: *Advances in neural information processing systems* 32.
- Dosso, Cheyenne, Lynda Tamine, Pierre-Vincent Paubel, and Aline Chevalier (2022). “The impact of expertise on query formulation strategies during complex learning task solving: a study with students in medicine and computer science”. In: *Proceedings of the 21st Congress of the International Ergonomics Association (IEA 2021) Volume V: Methods & Approaches 21*. Springer, pp. 621–627.
- Dosso, Cheyenne, Lynda Tamine, Pierre-Vincent Paubel, and Aline Chevalier (2024). “Navigational and thematic exploration–exploitation trade-offs during web search: effects of prior domain knowledge, search contexts and strategies on search outcome”. In: *Behaviour & Information Technology* 43.10, pp. 2232–2258.
- El Zein, Dima, Arthur Câmara, Célio 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*, pp. 388–393.
- El Zein, Dima and Célia da Costa Pereira (2021). “A cognitive agent framework in information retrieval: Using user beliefs to customize results”. In: *PRIMA 2020: Principles and Practice of Multi-Agent Systems: 23rd International Conference, Nagoya, Japan, November 18–20, 2020, Proceedings 23*. Springer, pp. 325–333.
- Elgohary, Ahmed, Denis Peskov, and Jordan Boyd-Graber (2019). “Can You Unpack That? Learning to Rewrite Questions-in-Context”. In: *Empirical Methods in Natural Language Processing*.
- Evans, Brynn M and Ed H Chi (2008). “Towards a model of understanding social search”. In: *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pp. 485–494.
- Galimzhanova, Elnara, Cristina Ioana Muntean, Franco Maria Nardini, Raffaele Perego, and Guido Rocchietti (2023). “Rewriting conversational utterances with instructed large language models”. In: *2023 IEEE International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. IEEE, pp. 56–63.
- Ganguly, Debasis, Johannes Leveling, and Gareth JF Jones (2012). “Topical relevance model”. In: *Asia Information Retrieval Symposium*. Springer, pp. 326–335.
- Gao, Jianfeng, Chenyan Xiong, Paul Bennett, and Nick Craswell (2023). *Neural approaches to conversational information retrieval*. Vol. 44. Springer.
- Guo, Jiafeng, Yixing Fan, Qingyao Ai, and W Bruce Croft (2016). “A deep relevance matching model for ad-hoc retrieval”. In: *Proceedings of the 25th ACM international on conference on information and knowledge management*, pp. 55–64.
- Guo, Jiafeng, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W Bruce Croft, and Xueqi Cheng (2020). “A deep look into neural ranking models for information retrieval”. In: *Information Processing & Management* 57.6, p. 102067.

- Hagen, Matthias, Martin Potthast, Michael Völske, Jakob Gomoll, and Benno Stein (2016). “How writers search: Analyzing the search and writing logs of non-fictional essays”. In: *Proceedings of the 2016 ACM on conference on human information interaction and retrieval*, pp. 193–202.
- Hannabuss, Stuart (1989). “Dialogue and the search for information”. In: *Aslib proceedings*. Vol. 41. 3. MCB UP Ltd, pp. 85–98.
- Harman, Donna (1994). “Overview of the second text retrieval conference (TREC-2)”. In: *NIST SPECIAL PUBLICATION SP*, pp. 1–1.
- Harman, Donna et al. (2019). “Information retrieval: the early years”. In: *Foundations and Trends® in Information Retrieval* 13.5, pp. 425–577.
- He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.
- Hiemstra, Djoerd (2001). “Using language models for information retrieval”. In.
- Hoppe, Anett, 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*.
- Huang, Haoda and Benyu Zhang (2009). “Text Indexing and Retrieval”. In: *Encyclopedia of Database Systems*. Ed. by LING LIU and M. TAMER ÖZSU. Boston, MA: Springer US, pp. 3055–3058. ISBN: 978-0-387-39940-9. DOI: 10.1007/978-0-387-39940-9_417. URL: https://doi.org/10.1007/978-0-387-39940-9_417.
- Huang, Po-Sen, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck (2013). “Learning deep structured semantic models for web search using click-through data”. In: *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pp. 2333–2338.
- Järvelin, Kalervo and Jaana Kekäläinen (2002). “Cumulated gain-based evaluation of IR techniques”. In: *ACM Transactions on Information Systems (TOIS)* 20.4, pp. 422–446.

- Johnson, Jeff, Matthijs Douze, and Hervé Jégou (2019). “Billion-scale similarity search with GPUs”. In: *IEEE Transactions on Big Data* 7.3, pp. 535–547.
- Jones, K Sparck, Steve Walker, and Stephen E. Robertson (2000a). “A probabilistic model of information retrieval: development and comparative experiments: Part 2”. In: *Information processing & management* 36.6, pp. 809–840.
- Jones, K Sparck, Steve Walker, and Stephen E. Robertson (2000b). “A probabilistic model of information retrieval: development and comparative experiments: Part 2”. In: *Information processing & management* 36.6, pp. 809–840.
- Jozefowicz, Rafal, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu (2016). “Exploring the limits of language modeling”. In: *arXiv preprint arXiv:1602.02410*.
- Kantor, Paul B and Ellen M Voorhees (1996). “Report on the TREC-5 Confusion Track.” In: *TREC*.
- Kantor, Paul B and Ellen M Voorhees (2000). “The TREC-5 confusion track: Comparing retrieval methods for scanned text”. In: *Information Retrieval* 2.2, pp. 165–176.
- Kekäläinen, Jaana (2005). “Binary and graded relevance in IR evaluations—Comparison of the effects on ranking of IR systems”. In: *Information Processing & Management* 41.5, pp. 1019–1033. ISSN: 0306-4573. DOI: <https://doi.org/10.1016/j.ipm.2005.01.004>. URL: <https://www.sciencedirect.com/science/article/pii/S0306457305000075>.
- Khattab, Omar and Matei Zaharia (2020a). “Colbert: Efficient and effective passage search via contextualized late interaction over bert”. In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48.
- Khattab, Omar and Matei Zaharia (2020b). “Colbert: Efficient and effective passage search via contextualized late interaction over bert”. In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48.

- Kim, Sungdong and Gangwoo Kim (2022). “Saving Dense Retriever from Shortcut Dependency in Conversational Search”. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 10278–10287.
- Kostric, Ivica, Krisztian Balog, and Filip Radlinski (2024). “Generating usage-related questions for preference elicitation in conversational recommender systems”. In: *ACM Transactions on Recommender Systems* 2.2, pp. 1–24.
- Kuhlthau, Carol C (1991). “Inside the search process: Information seeking from the user’s perspective”. In: *Journal of the American society for information science* 42.5, pp. 361–371.
- Lavrenko, Victor and W Bruce Croft (2017). “Relevance-based language models”. In: *ACM SIGIR Forum*. Vol. 51. 2. ACM New York, NY, USA, pp. 260–267.
- Lavrenko, Victor and W. Bruce Croft (2001). “Relevance models in information retrieval”. In: *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pp. 120–127.
- Lin, Jimmy, 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*, pp. 2356–2362.
- Lin, Sheng-Chieh, Jheng-Hong Yang, and Jimmy Lin (2021). “Contextualized Query Embeddings for Conversational Search”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 1004–1015.
- Lin, Sheng-Chieh, 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”. In: *arXiv preprint arXiv:2004.01909*.
- Lin, Sheng-Chieh, 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”. In: *ACM Transactions on Information Systems (TOIS)* 39.4, pp. 1–29.
- Lin, Sheng-Chieh, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy J. Lin (2021). “Multi-Stage Conversational Passage Retrieval: An Approach to Fusing Term Importance Estimation and Neural Query Rewriting”. In: *ACM Transactions on Information Systems (TOIS)* 39, pp. 1–29.
- Manning, Christopher D. (2008). *Introduction to information retrieval*. In collab. with Prabhakar Raghavan and Hinrich Schütze. Cambridge: University Press. ISBN: 978-0-511-57336-1.
- Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze (2008). *Introduction to Information Retrieval*. Cambridge University Press.
- Mao, Kelong, Zhicheng Dou, Haonan Chen, Fengran Mo, and Hongjin Qian (2023). “Large Language Models Know Your Contextual Search Intent: A Prompting Framework for Conversational Search”. In: *arXiv preprint arXiv:2303.06573*.
- Mao, Kelong, Zhicheng Dou, Fengran Mo, Jiewen Hou, Haonan Chen, and Hongjin Qian (2023). “Large language models know your contextual search intent: A prompting framework for conversational search”. In: *arXiv preprint arXiv:2303.06573*.
- Mehri, Shikib, Mihail Eric, and Dilek Hakkani-Tur (2020). “Dialoglue: A natural language understanding benchmark for task-oriented dialogue”. In: *arXiv preprint arXiv:2009.13570*.
- Mele, Ida, Cristina Ioana Muntean, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, and Ophir Frieder (2020). “Topic propagation in conversational search”. In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 2057–2060.
- Mele, Ida, Cristina Ioana Muntean, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, and Ophir Frieder (2021). “Adaptive utterance rewriting for conversational search”. In: *Information Processing & Management* 58.6, p. 102682.

- Mitra, Bhaskar, Nick Craswell, et al. (2018). “An introduction to neural information retrieval”. In: *Foundations and Trends® in Information Retrieval* 13.1, pp. 1–126.
- Mitra, Bhaskar, Fernando Diaz, and Nick Craswell (2017). “Learning to match using local and distributed representations of text for web search”. In: *Proceedings of the 26th international conference on world wide web*, pp. 1291–1299.
- Mo, Fengran, Kelong Mao, Ziliang Zhao, Hongjin Qian, Haonan Chen, Yiruo Cheng, Xiaoxi Li, Yutao Zhu, Zhicheng Dou, and Jian-Yun Nie (2025). “A survey of conversational search”. In: *ACM Transactions on Information Systems* 43.6, pp. 1–50.
- Mo, Fengran, Kelong Mao, Yutao Zhu, Yihong Wu, Kaiyu Huang, and Jian-Yun Nie (2023). “ConvGQR: Generative Query Reformulation for Conversational Search”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4998–5012.
- Mo, Fengran, Jian-Yun Nie, Kaiyu Huang, Kelong Mao, Yutao Zhu, Peng Li, and Yang Liu (2023). “Learning to Relate to Previous Turns in Conversational Search”. In: *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. KDD '23.*, Long Beach, CA, USA, Association for Computing Machinery, pp. 1722–1732. DOI: 10.1145/3580305.3599411. URL: <https://doi.org/10.1145/3580305.3599411>.
- Morris, Meredith Ringel (2008). “A survey of collaborative web search practices”. In: *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1657–1660.
- Morris, Meredith Ringel (2013). “Collaborative search revisited”. In: *Proceedings of the 2013 conference on Computer supported cooperative work*, pp. 1181–1192.
- Narayan, Shashi, Shay B Cohen, and Mirella Lapata (2018). “Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization”. In: *arXiv preprint arXiv:1808.08745*.
- Nasser, Hadi, Dima El Zein, Célia da Costa Pereira, Cathy Esczut, 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*, pp. 364–369.
- Novak, Joseph D (2010). *Learning, creating, and using knowledge: Concept maps as facilitative tools in schools and corporations*. Routledge.
- O’Brien, Heather L, Andrea Kampen, Amelia W Cole, and Kathleen Brennan (2020). “The role of domain knowledge in search as learning”. In: *Proceedings of the 2020 conference on human information interaction and retrieval*, pp. 313–317.
- Ohsugi, Yasuhito, Itsumi Saito, Kyosuke Nishida, Hisako Asano, and Junji Tomita (2019). “A simple but effective method to incorporate multi-turn context with BERT for conversational machine comprehension”. In: *arXiv preprint arXiv:1905.12848*.
- Overwijk, Arnold, Chenyan Xiong, and Jamie Callan (2022). “ClueWeb22: 10 billion web documents with rich information”. In: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3360–3362.
- Palangi, Hamid, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab Ward (2016). “Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval”. In: *IEEE/ACM transactions on audio, speech, and language processing* 24.4, pp. 694–707.
- Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer (June 2018). “Deep Contextualized Word Representations”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Ed. by Marilyn Walker, Heng Ji, and Amanda Stent. New Orleans, Louisiana: Association for Computational Linguistics, pp. 2227–2237. DOI: 10.18653/v1/N18-1202. URL: <https://aclanthology.org/N18-1202/>.

- Qu, Chen, Liu Yang, Cen Chen, Minghui Qiu, W Bruce Croft, and Mohit Iyyer (2020). “Open-retrieval conversational question answering”. In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 539–548.
- Radford, Alec (2018). “Improving language understanding with unsupervised learning”. In: *OpenAI Res.*
- Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. (2019). “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8, p. 9.
- Radlinski, Filip and Nick Craswell (2017). “A Theoretical Framework for Conversational Search”. In: CHIIR ’17. Oslo, Norway: Association for Computing Machinery, pp. 117–126. ISBN: 9781450346771. DOI: 10.1145/3020165.3020183. URL: <https://doi.org/10.1145/3020165.3020183>.
- Raffel, Colin, 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”. In: *Journal of machine learning research* 21.140, pp. 1–67.
- Robertson, Stephen, Hugo Zaragoza, et al. (2009). “The probabilistic relevance framework: BM25 and beyond”. In: *Foundations and Trends® in Information Retrieval* 3.4, pp. 333–389.
- Robertson, Stephen E (1977). “The probability ranking principle in IR”. In: *Journal of documentation* 33.4, pp. 294–304.
- Robertson, Stephen E, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. (1995). *Okapi at TREC-3*. British Library Research and Development Department.
- Rocchietti, Guido, Ophir Frieder, Cristina Ioana Muntean, Franco Maria Nardini, and Raffaele Perego (2023). “Commonsense injection in conversational systems: An adaptable framework for query expansion”. In: *2023 IEEE International*

- Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*.
IEEE, pp. 48–55.
- Rocchio Jr, Joseph John (1971). “Relevance feedback in information retrieval”. In:
The SMART retrieval system: experiments in automatic document processing.
- Rosset, Corbin, Chenyan Xiong, Xia Song, Daniel Campos, Nick Craswell, Saurabh
Tiwary, and Paul Bennett (2020). “Leading conversational search by suggesting
useful questions”. In: *Proceedings of the web conference 2020*, pp. 1160–1170.
- Salton, Gerard, Anita Wong, and Chung-Shu Yang (1975). “A vector space model
for automatic indexing”. In: *Communications of the ACM* 18.11, pp. 613–620.
- Sanchiz, Mylene, Aline Chevalier, and Franck Amadieu (2017). “How do older and
young adults start searching for information? Impact of age, domain knowledge
and problem complexity on the different steps of information searching”. In:
Computers in Human Behavior 72, pp. 67–78.
- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani (2021). “Towards facet-
driven generation of clarifying questions for conversational search”. In: *Proceedings
of the 2021 ACM SIGIR international conference on theory of information retrieval*,
pp. 167–175.
- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani (2022). “Evaluating mixed-
initiative conversational search systems via user simulation”. In: *Proceedings of
the Fifteenth ACM International Conference on Web Search and Data Mining*,
pp. 888–896.
- Shen, Yelong, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil (2014a).
“A latent semantic model with convolutional-pooling structure for information
retrieval”. In: *Proceedings of the 23rd ACM international conference on conference
on information and knowledge management*, pp. 101–110.
- Shen, Yelong, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil (2014b).
“Learning semantic representations using convolutional neural networks for web
search”. In: *Proceedings of the 23rd international conference on world wide web*,
pp. 373–374.

- Spark-Jones, Karen (1975). “Report on the need for and provision of an ‘ideal’ information retrieval test collection”. In: *Computer Laboratory*.
- Syed, Rohail and Kevyn Collins-Thompson (2017a). “Optimizing search results for human learning goals”. In: *Information Retrieval Journal* 20, pp. 506–523.
- Syed, Rohail and Kevyn Collins-Thompson (2017b). “Retrieval algorithms optimized for human learning”. In: *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*, pp. 555–564.
- Taylor, Wilson L (1953). ““Cloze procedure”: A new tool for measuring readability”. In: *Journalism quarterly* 30.4, pp. 415–433.
- The ClueWeb09 Dataset* (n.d.). [Accessed 01-09-2025]. URL: <https://lemurproject.org/clueweb09/>.
- The ClueWeb12 Dataset* (n.d.). [Accessed 01-09-2025]. URL: <https://lemurproject.org/clueweb12/>.
- Trabelsi, Mohamed, Zhiyu Chen, Brian D Davison, and Jeff Heflin (2021). “Neural ranking models for document retrieval”. In: *Information Retrieval Journal* 24.6, pp. 400–444.
- Tuomaala, Otto, Kalervo Järvelin, and Pertti Vakkari (2014). “Evolution of library and information science, 1965–2005: Content analysis of journal articles”. In: *Journal of the Association for information science and technology* 65.7, pp. 1446–1462.
- Vakkari, Pertti (2016). “Searching as learning: A systematization based on literature”. In: *Journal of Information Science* 42.1, pp. 7–18.
- Vakulenko, Svitlana, 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*, pp. 355–363.
- Vakulenko, Svitlana, Nikos Voskarides, Zhucheng Tu, and Shayne Longpre (2021). “A comparison of question rewriting methods for conversational passage retrieval”. In: *Advances in Information Retrieval: 43rd European Conference on IR Research*,

- ECIR 2021, Virtual Event, March 28–April 1, 2021, Proceedings, Part II* 43. Springer, pp. 418–424.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017). “Attention is all you need”. In: *Advances in neural information processing systems* 30.
- Voorhees, Ellen M (1993). “On expanding query vectors with lexically related words”. In: *TREC*, pp. 223–232.
- Voorhees, Ellen M, Donna K Harman, et al. (2005). *TREC: Experiment and evaluation in information retrieval*. Vol. 63. MIT press Cambridge.
- Voskarides, Nikos, 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*, pp. 921–930.
- Wan, Shengxian, Yanyan Lan, Jiafeng Guo, Jun Xu, Liang Pang, and Xueqi Cheng (2016). “A deep architecture for semantic matching with multiple positional sentence representations”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 30. 1.
- Wang, Bingning, Kang Liu, and Jun Zhao (2016). “Inner attention based recurrent neural networks for answer selection”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1288–1297.
- Wang, Shuohang and Jing Jiang (2016). “A compare-aggregate model for matching text sequences”. In: *arXiv preprint arXiv:1611.01747*.
- White, Ryen W, Susan T Dumais, and Jaime Teevan (2009). “Characterizing the influence of domain expertise on web search behavior”. In: *Proceedings of the second ACM international conference on web search and data mining*, pp. 132–141.
- Willoughby, Teena, S Alexandria Anderson, Eileen Wood, Julie Mueller, and Craig Ross (2009). “Fast searching for information on the Internet to use in a learning

- context: The impact of domain knowledge”. In: *Computers & Education* 52.3, pp. 640–648.
- Wilson, Tom D (1999). “Models in information behaviour research”. In: *Journal of documentation* 55.3, pp. 249–270.
- Xiong, Chenyan and Jamie Callan (2015). “Query expansion with freebase”. In: *Proceedings of the 2015 international conference on the theory of information retrieval*, pp. 111–120.
- Yang, Peilin, 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*, pp. 1253–1256.
- Yu, Shi, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu (2021). “Few-shot conversational dense retrieval”. In: *Proceedings of the 44th International ACM SIGIR Conference on research and development in information retrieval*, pp. 829–838.
- Yuan, Yifei, Clemencia Siro, Mohammad Aliannejadi, Maarten de Rijke, and Wai Lam (2024). “Asking multimodal clarifying questions in mixed-initiative conversational search”. In: *Proceedings of the ACM Web Conference 2024*, pp. 1474–1485.
- Zamani, Hamed, Johanne R Trippas, Jeff Dalton, Filip Radlinski, et al. (2023). “Conversational information seeking”. In: *Foundations and Trends® in Information Retrieval* 17.3-4, pp. 244–456.
- Zhang, Jingqing, Yao Zhao, Mohammad Saleh, and Peter Liu (July 2020). “PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization”. In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 11328–11339. URL: <https://proceedings.mlr.press/v119/zhang20ae.html>.
- Zhang, Xiangmin, Michael Cole, and Nicholas Belkin (2011). “Predicting users’ domain knowledge from search behaviors”. In: *Proceedings of the 34th international*

ACM SIGIR conference on research and development in information retrieval, pp. 1225–1226.

Zhu, Yukun, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler (2015). “Aligning books and movies: Towards story-like visual explanations by watching movies and reading books”. In: *Proceedings of the IEEE international conference on computer vision*, pp. 19–27.

Zhu, Yutao, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen (2025). “Large Language Models for Information Retrieval: A Survey”. In: *ACM Trans. Inf. Syst.* ISSN: 1046-8188. DOI: 10.1145/3748304. URL: <https://doi.org/10.1145/3748304>.

Zhu, Yutao, Peitian Zhang, Chenghao Zhang, Yifei Chen, Binyu Xie, Zhicheng Dou, Zheng Liu, and Ji-Rong Wen (2024). “INTERs: Unlocking the Power of Large Language Models in Search with Instruction Tuning”. In: *arXiv preprint arXiv:2401.06532*.