
Clarification and User Simulation in Mixed-Initiative Conversational Search

Doctoral Dissertation submitted to the
Faculty of Informatics of the Università della Svizzera italiana
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

presented by
Ivan Sekulić

under the supervision of
Prof. Fabio Crestani

February 2024

Dissertation Committee

Prof. Alippi Cesare Università della Svizzera italiana
Prof. Laura Pozzi Università della Svizzera italiana
Prof. Emine Yilmaz University College London, UK
Prof. Krisztian Balog University of Stavanger, Norway

Dissertation accepted on 27 February 2024

Research Advisor
Prof. Fabio Crestani

PhD Program Director
Prof. Walter Binder

I certify that except where due acknowledgement has been given, the work presented in this thesis is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; and the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

Ivan Sekulić
Lugano, 27 February 2024

Abstract

The primary goal of an information retrieval (IR) system is to satisfy user's information need. Recently, with the recent rise in development of conversational assistants, conversational search, also referred to as conversational information retrieval (CIR), has gained significant attention from research community. While the primary goal of a CIR system is the same, conversational setting, however, poses significant challenges compared to traditional *ad hoc* search. For example, CIR system needs to keep track of the conversational context and previous user utterances, as the user can refer to conversational history at any point in a conversation. Moreover, CIR is often carried out in limited-bandwidth scenarios, such as voice only or mobile search, thus requiring appropriate user interfaces. Mixed initiative, paradigm, where search system can be proactive and take initiative at any point in our conversation, first, some potential solutions to the aforementioned challenges.

This dissertation is presented in two parts. First part deals with modeling specific tasks in mixed-initiative conversational search. Specifically, we address the issue of constructing clarifying questions with the purpose of elucidating user's underlying information need. To this end, we propose a novel method for generating clarifying questions, based on query facets. Further, we analyze the possibility extracting the facets from a list of documents retrieved in response to the initial query. The findings show promising direction for clarifying question construction in conversational search. We additionally address the issue of providing appropriate responses in a conversational setting. To this aim, we propose an entity-based response rewriting approach, which provides explanation of salient entities (or offers the user to learn about them in a follow-up question), thus making the response self-contained.

The second part of dissertation concerns itself with user simulation. Evaluation of conversational search systems is arduous. The challenge arises from the fact that usually expensive and time-consuming user studies are required to properly evaluate our system. User simulation presents itself as a solution to this problem, as the simulator assumes the user role in its interaction with

the system. For example, simulated user should be able to express its information need through queries, answer clarifying questions, and provide feedback to system's response. In this dissertation, we present novel large language model (LLM) based approach to user simulation for conversational search. Specifically, we first design a simulator capable of, given an information need description, answer potential clarifying questions posed by the search system. Moreover, we expand on this approach by proposing an extension, capable of multi-turn interactions and able to provide explicit feedback. We show that our simulated user can be used for interactions with conversational search systems and help in their evaluation. Further, we demonstrate the effectiveness of our approach by conducting human annotation to show that author and says generated by dis-simulator are both useful and natural. Finally, we discuss applications of our approach and possible extensions for future work.

Acknowledgements

My PhD journey has been an incredibly enriching experience, full of life, hard work, and personal growth in all possible directions. During the last four years, I had the privilege of meeting and collaborating with some truly exceptional people, many of whom I hope to call friends for many years to come. In this section, I would like to briefly and sincerely express my gratitude to everyone who accompanied me on this journey.

I would like to thank my advisor, Fabio Crestani, for giving me the opportunity to learn and develop in his lab. I thank him for his continuous support, for granting me the autonomy to pursue my research interests, for guiding me through the academic world, and for connecting me with the research community.

I would like to thank my collaborators, whose support and companionship have made my journey smoother and more fulfilling. First, big thanks to Mohammad Aliannejadi, who accompanied me from the very first week of my PhD, all the way to the end. Thank you for all your time and a very hands-on approach, which greatly propelled the start of my PhD.

I would also like to thank Jeff Dalton and the entire GRILL Lab: Paul Owoicho, Iain Mackie, Sophie Fischer, Carlos Gemell, Alessandro Speggiorin, and Federico Rossetto. Thank you for wholeheartedly accepting me to your group, many great collaborations, and all the enjoyable moments we've shared beyond the office, especially for our epic adventure taking over California.

Thanks to Krisztian Balog, who taught me a great deal about setting boundaries, time management, and focusing on high quality research. Also, thanks to the entire IAI group, Weronika Łajewska, Ivica Kostrić, Nolwenn Bernard, and Trond Linjordet, for all the good times in Stavanger.

Additionally, I want to express my thanks to Telepathy Labs, where I spent six months as a research intern and learned a great deal about the industry's side of research and development. Special thanks to the DM team: Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, André Ferreira Manso, Roland Mathis.

I would also like to express my gratitude to everyone who had an impact on

my development even before starting my PhD, including NLP group at Heidelberg Institute for Theoretical Studies, StatNLP group at Singapore University of Technology and Design, and, especially TakeLab from University of Zagreb. Special thanks to Jan Šnajder, who possibly had a huge impact on the start of my research journey, by setting the high standard for quality, going beyond of what was needed, and enviable dedication to teaching.

Thanks to USI and everyone from the IR and LUXIA groups, to both past and new members: Esteban Andrés Ríssola, Leandro S. Guedes, Sveva Valguarnera, Irene Zanardi, Federico Ravenda, Alessandro Consentino, Navdeep Singh Bedi, Michal Burgunder, Lili Lu, and Maryam. Special thanks to Monica Landoni, for keeping the group grounded and for all of the therapeutic tea sessions.

All of you have, either directly, through collaborations and knowledge transfers, or indirectly, through your unique points of view on research and the world in general, made me a better researcher. I can honestly say I wouldn't have such a fruitful and joyous journey without all of the people listed above.

My gratitude goes to my friends, both old and new, for all the fun adventures, where I was able to completely disconnect from the PhD work. Special thanks to the Sisak crew for staying by my side as my closest friends, despite the Alps and a thousand kilometers keeping us apart. Thanks to my family, for always being there and providing a safe sanctuary for charging my batteries.

Finally, my biggest gratitude goes to Ivna, for unconditional love and support through both the good and the hard times. Your beautiful view on the world, your passion for life, and your amazing virtues, are what inspires me. And of course, thank you Adrian, for showing me what truly matters in life; and thank you Iskra, for taking me out for a walk every day and all the emotional support.

Contents

Contents	vii
List of Figures	xiii
List of Tables	xv
1 Introduction	1
1.1 Conversational Search	2
1.2 Mixed-Initiative Conversational Search	4
1.3 User Simulation	6
1.4 Main Contributions	7
1.5 Dissertation Outline	8
1.6 Publication Overview	9
2 Literature Review and Background	13
2.1 Formalizing Conversational Search	13
2.1.1 Conversation	14
2.1.2 Types of Utterances	14
2.1.3 Search	15
2.2 Conversational Search	16
2.2.1 Defining Conversational Search	16
2.2.2 Query Rewriting	18
2.2.3 Conversational Passage Retrieval	19
2.2.4 Result Representation	21
2.3 Mixed-Initiative	22
2.3.1 Clarification in Search	22
2.3.2 Selecting Clarifying Questions	24
2.3.3 Generating Clarifying Questions	25
2.3.4 Entity-Oriented Search	27
2.4 User Engagement	28

2.5	User Simulation	29
2.5.1	The Challenge of Conversational System Evaluation	29
2.5.2	Automated Metrics	30
2.5.3	User Studies	31
2.5.4	User Simulation in IR	32
2.5.5	Simulation for Conversational Search	32
2.5.6	Simulation for TOD Systems	33
I	Mixed-Initiative Conversational Search	35
3	Clarifying Questions in Conversational Search	37
3.1	Facet-driven Clarifying Question Generation	37
3.1.1	Semantically-Guided Question Generation	39
3.1.2	Dataset Construction	41
3.1.3	Fine-tuning GPT-2	43
3.1.4	Evaluation	43
3.1.5	Limitations and Future Work	49
3.2	Exploiting Document-based Features for Clarification	51
3.2.1	Methodology	53
3.2.2	Evaluation Setting	55
3.2.3	Results and Discussion	57
3.3	Usefulness of Clarifying Questions and Corresponding Answers	62
3.3.1	Problem Statement	63
3.3.2	Results	67
3.3.3	Analysis	68
3.4	Conclusions	70
4	User Engagement Prediction for Clarification in Search	73
4.1	Engagement Level Prediction	75
4.1.1	Data	75
4.1.2	Task Formulation	76
4.1.3	Our approach	77
4.2	Experiments	78
4.2.1	Baselines	79
4.2.2	Results & Discussion	80
4.2.3	Additional Experiments	83
4.3	Conclusions	87

5	Entity-Based Response Rewriting	89
5.1	Understanding Salient Entities in Conversational Information Seeking	92
5.1.1	Problem statement	92
5.1.2	Dataset Acquisition	93
5.1.3	Analysis	95
5.2	Answer Rewriting	98
5.2.1	Inline Entity Descriptions	98
5.2.2	Mixed-initiative Follow-up Prompt	100
5.2.3	Evaluation of Answer Rewrite Strategies	101
5.2.4	Discussion	108
5.3	Conclusions	109
II	User Simulation	111
6	User Simulation for Conversational Search	113
6.1	Motivation and Problem Setting	114
6.1.1	User’s Role in Conversational Search System Evaluation	114
6.1.2	Formal Definition	115
6.1.3	Research Questions	117
6.2	Evaluation Methodology	118
6.2.1	Data	119
6.2.2	NLG Metrics	122
6.2.3	Human Evaluation	123
6.2.4	Impact of Feedback to Retrieval Performance	125
6.2.5	Explicit Feedback	126
6.3	Answering Clarifying Questions via User Simulation	126
6.3.1	Semantically-Controlled Text Generation	127
6.3.2	GPT2-Based Simulated User	128
6.3.3	Evaluation Details	129
6.4	Beyond Clarifying Questions – Providing Feedback	130
6.4.1	In-Context Learning Approach	132
6.4.2	Implementation Details	133
6.4.3	Mixed-Initiative Systems	134
6.4.4	Evaluation Details	137
6.5	Results and Analysis	139
6.5.1	Answering Clarifying Questions	139
6.5.2	Exploiting Simulated Feedback	143

6.5.3	Discussion and Analysis	144
6.6	Qualitative Analysis	145
6.6.1	Examples	148
6.6.2	Response Patterns	149
7	User Simulation for Task-Oriented Dialogue Systems	153
7.1	Generative User Simulator	155
7.1.1	Background	155
7.1.2	Our Approach	156
7.2	Experimental Setting	157
7.2.1	Data sources	157
7.2.2	TOD Systems	158
7.2.3	User Goal Settings	158
7.2.4	Fine-tuning Details	158
7.2.5	Baselines	160
7.3	Evaluation	161
7.3.1	Goal Fulfillment Evaluation Metrics	161
7.3.2	Lexical Diversity of Generated Utterances	162
7.3.3	Qualitative Analysis	162
7.4	Results	162
7.4.1	Goal Fulfillment	163
7.4.2	Lexical Diversity	165
7.4.3	Generalization to Unseen User Tasks	166
7.4.4	Utterance-Level Metrics	166
7.5	Qualitative Analysis	168
7.5.1	Human Evaluation of Generated Dialogues	168
7.5.2	Domain-specific Vocabulary	169
7.5.3	Inter-annotator agreement	170
7.6	Conclusions	170
III	Discussion and Conclusions	173
8	Discussion	175
8.1	Clarification in Conversational Search	175
8.1.1	Generating Clarifying Questions	175
8.1.2	Limitations	177
8.2	LLM-based User Simulation	180
8.2.1	Advantages over Agenda-based Simulation	180

8.2.2	Exploring Conversational Trajectories	181
8.2.3	Performance vs Cost	181
8.2.4	Limitations	182
9	Conclusions	185
9.1	Summary	185
9.2	Main Contributions	187
9.3	Future Directions	189
A	Graph-Based Approaches to IR	193
A.1	Multi-document Graph-based IR	193
A.1.1	Graph Construction	195
A.1.2	Graph Analysis	196
A.1.3	The Graph Model	198
A.1.4	Results and Analysis	199
A.2	Exploring Complex Topics	200
A.3	Extension to Conversational Search	202
A.4	Discussion	203

Figures

1.1	Interaction between a user and a conversational search system . . .	4
3.1	Overlap of extracted features with clarifying questions	58
4.1	An example of a clarification pane	74
4.2	Performance by impression levels and query lengths	83
4.3	Performance by number of search results made available	84
4.4	Performance by question type	85
4.5	Model attention over SERP elements	87
5.1	Possible strategies to making sure the user understands the answer	91
5.2	Examples of changes in salience score through conversational turns	96
5.3	Answer rewriting strategies	99
6.1	Experimental framework with an example interaction between a user simulator and conversational search system	131
6.2	Performance after multiple rounds of feedback	146
7.1	Example conversation between user simulator and TOD system . .	154
A.1	An illustrative subgraph for multi-document retrieval	194
A.2	Pairplot of several aspects of the created graph	197
A.3	Subgraph of entities in conversational search	202

Tables

3.1	Statistics of ClariQ-Fkw dataset	42
3.2	Training triplets from ClariQ-FKw	42
3.3	Evaluation of generated questions against gold standard in ClariQ	45
3.4	Results on naturalness of facet-based question generation methods	48
3.5	Results on usefulness of facet-based question generation methods	48
3.6	Statistical significance of naturalness and usefulness	48
3.7	Examples of generated clarifying questions	49
3.8	Impact of facet extraction methods on retrieval performance	60
3.9	Pairwise comparison of questions generated based on different facets	61
3.10	Examples of annotated subset of ClariQ	65
3.11	Clarifying question relevance on TREC CAsT'22	67
3.12	Performance of mixed-initiative approaches on TREC CAsT'22	67
3.13	Performance gains and losses of MI-All and MI-Clf methods	69
4.1	Dataset statistics for MIMICS-Click	77
4.2	Performance on ELP task in MIMICS	81
4.3	Impact of SERP elements on the model performance	82
4.4	Performance on clarification pane re-ranking task on MIMICS	86
5.1	Excerpts from QReCC with annotations of entity salience scores	94
5.2	Examples of special cases of essential entities	95
5.3	Answer rewrite preference assessed by crowd workers	103
5.4	Answer rewrite preference by top N salient entities rewritten	105
5.5	Reasons for choosing one type of rewrite over another	105
5.6	Examples of LLM rewrites for answer rewriting	107
6.1	Statistics for Qulac and ClariQ datasets	119
6.2	Acquired multi-turn dataset	121
6.3	NLG metrics of different answer generation methods	139
6.4	Results on naturalness and usefulness of responses	140

6.5	Single- and multi-turn results on naturalness and usefulness . . .	142
6.6	Retrieval performance based simulated answers	142
6.7	Performance after asking a clarifying question	144
6.8	Performance on turns where feedback is negative vs. positive . .	145
6.9	Qualitative analysis of answers generated by user simulator	147
6.10	Identified reformulation patterns in generated responses	151
6.11	Prevalence of utterance reformulation types	152
7.1	Dataset statistics after pre-processing	157
7.2	Description of user goals with subtask types	159
7.3	Results of goal fulfillment task	162
7.4	Results of goal-fulfillment task per specific user goal	164
7.5	Performance on MultiWOZ 2.1 within ConvLab2 framework	165
7.6	Percentage of dialogues with successfully identified subtask types	166
7.7	NLG- and NLU-based utterance-level metrics	167
7.8	Percentage of the observed patterns in simulated dialogues	168
7.9	Inter-annotator agreement	170
7.10	Examples of common failures observed through analysis	172
A.1	Results on MS MARCO development set	200
A.2	Ablation study on MS MARCO development	200

Chapter 1

Introduction

The primary goal of an information retrieval (IR) system is to satisfy a user's information need. The same is true for conversational search systems, which aim to meet user's information need through multi-turn dialogue. With the recent rise of conversational assistants, like Siri, Alexa, and Cortana, conversational search has gained in popularity in the research community. However, conversational search poses additional challenges compared to the traditional *ad hoc* IR systems. For example, the system needs to track the state of the conversation, as users can reference to their past utterances at any point of the interaction with the system. Moreover, as conversational search system is usually able to express only limited amount of information to the user, due to the nature of conversations, there is a need for designing appropriate interfaces by constructing concise responses or clarifying the user's information need.

A mixed-initiative paradigm, where the conversational search system is proactive and can at any point in the conversation prompt the user with a question or suggestion, offers potential solutions to the aforementioned challenges [Radlinski and Craswell, 2017]. Specifically, the mixed-initiative conversational search (MICS) system can elucidate the underlying information need by asking clarifying questions to the user [Braslavski et al., 2017]. The first objective of this dissertation is to address the problem of asking clarifying questions, by proposing methods for selecting, generating, and processing clarifying questions. Further, we explore the design of system's responses, with an aim of making them more accessible to users in a text-based conversational setting.

An open research direction in MICS systems, and other conversational systems in general, is their evaluation [Fu et al., 2022]. The challenge arises from the fact that time-consuming and expensive human evaluation of such systems is required for the appropriate assessment of the system's performance [Deriu

et al., 2021]. Recently proposed offline evaluation methods attempt to mitigate the need for human intervention; however, they remain limited to single-turn interactions or predefined conversational trajectories, thus not capturing the particulars of real world scenarios. User simulation aims to address the evaluation, as one of the key bottlenecks in progress of conversational search, by assuming the role of a real user in the interaction with the MICS system [Balog, 2021]. The second objective of the dissertation is to design user simulation methods capable of reliable and natural multi-turn interactions with a variety of MICS systems.

1.1 Conversational Search

Traditional *ad hoc* IR systems aim to satisfy user’s information need by retrieving documents relevant to the user’s query from a collection of such resources. An IR system does so by retrieving relevant documents from a large collection of such resources. The collection can be made up of any types of objects, but in general, and in this document, we consider the collection to be constituted of textual documents. In modern IR systems (e.g., search engines like Google, Bing, and Baidu) users represent their information need with several descriptive keywords, constructing a query. The IR system searches for the documents in it’s collection that are likely to be relevant to the given query. Several main challenges arise in the described *ad hoc* retrieval paradigm. For example, we need to process the documents in our collection and store them in a easily accessible format, formulate a mapping function from a query to a document, and design a user-friendly interface for presenting the retrieved results.

Conversational search is an IR paradigm where users issue queries in a conversational form, either by voice or in a chat-like text [Anand et al., 2020]. The goal of a conversational search system is still to satisfy the underlying user information need, but several new challenges arise compared to *ad hoc* retrieval. For example, the user can at any point reference back some aspects of the conversational history. Thus, the system needs to keep track of the state of the conversation over multi-turn interactions and successfully deal with coreferences [Vakulenko et al., 2021] and topical shifts [Mele et al., 2020], among others.

Conversational search systems are frequently being used for acquiring new knowledge [Gadiraju et al., 2018]. However, while traditional search interfaces offers users to follow hyperlinks, scroll through a list of documents, or consult knowledge panels in search engine results pages in order to learn about certain concepts they might be unfamiliar with [Eickhoff et al., 2014], such opportunity is taken away in conversational setting due to the limited bandwidth interface. In

limited-bandwidth scenarios, such as mobile or voice-only search, there is a need for an appropriate interface with the user [White, 2016], since we are usually able to present only a few, or even only one result to the user. The system's responses to user's queries therefore need to be concise and information-dense, a challenge we address by rewriting system's responses with an aim to make them understandable to wide audiences [Sekulić et al., 2024]. Another prevalent approach to ensuring satisfactory user experience is asking clarifying questions to elucidate users information need, so that the system is able to provide more accurate responses, not overwhelming the user with irrelevant information in the process [Radlinski and Craswell, 2017]. We discuss this challenge in greater detail in the next section.

This dissertation centers primarily on conversational search, also referred to as conversational information retrieval (CIR). While our primary emphasis lies in the topics around conversational search, we touch on various types of conversational AI, highlighting the distinctions between them. This discussion extends to diverse approaches to conversational information access (CIA) [Zamani et al., 2022], such as conversational question-answering (Q&A) [Choi et al., 2018], conversational search, as well as task-oriented dialogue (TOD) systems [Zhang et al., 2020b]. Conversational information access refers to the process of retrieving information through conversation in natural language [Zamani et al., 2022]. It involves leveraging conversational interfaces and techniques to interact with information systems, databases, or other repositories in a manner that mimics human conversation. Detailed distinction between different conversational systems is presented in Chapt. 2.

Figure 1.1 shows an example of user's interaction with a conversational search system, with individual actions taken by each participant noted. In the presented case, the user's information need involves learning about the estimated number of coral reefs left on Earth. After the user initiates a conversation with a query, conversational search system can either provide a response or, as in this case, ask the user a clarifying question with a goal of elucidating their need. The system is further required to take into account user's answers and their explicit feedback, in order to finally provide a satisfactory response.

Through this example, we point out several challenges of the conversational paradigm, compared to the *ad hoc* IR. First and foremost, the system needs to keep track of conversational context and adjust its responses accordingly, leading to a natural evolution of the dialogue. For example, user's third utterance "*That's not what I asked. How many of them are left?*", refers to coral reefs with a pronoun *them*, which requires the system to comprehend the utterance with the conversational history taken into account and therefore deduct that "them"

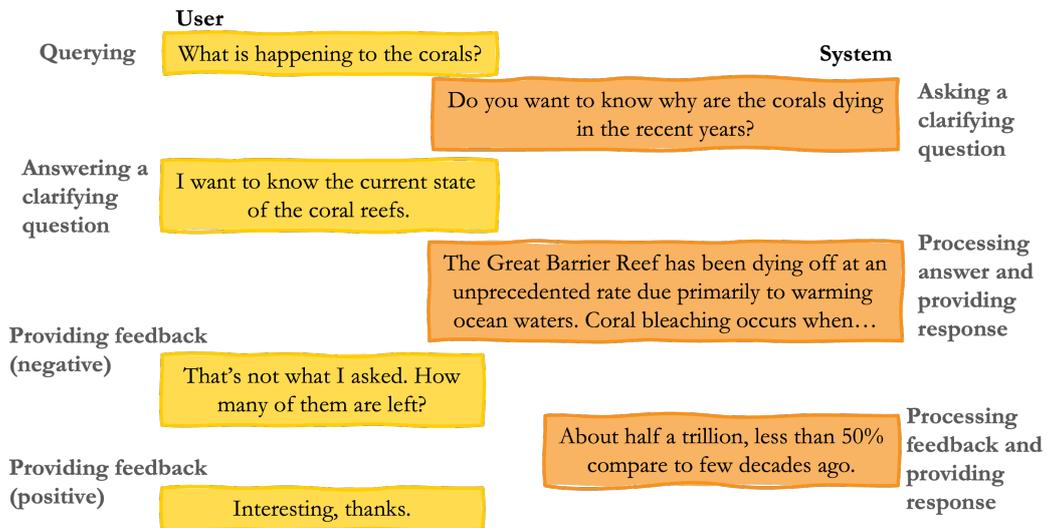


Figure 1.1. Example interaction between a user and a conversational search system.

refers to “coral reefs”. The system then, usually, reformulates the query to an information-dense form with such coreferences resolved (e.g., “*How many coral reefs are left?*”).

Additionally, the system can at any point ask a clarifying question, which we analyze in greater depth in the next section, and thus needs to decide on the appropriate timing and the necessity of asking such a question, as well as on its content. When presenting the information to the user, the system needs to be attentive to the user’s search interface, which, as mentioned, is often of limited bandwidth. Therefore, system’s responses should be concise. Common approaches to achieving this include generating such a response based on the list of documents retrieved from a given collection [Vakulenko et al., 2022, Owoicho et al., 2022]. The response is thus both grounded in existing collection and of appropriate length for a conversational setting. In this dissertation, we present multiple studies, where some include the response generation module, while other do not, thus focusing solely on the task of retrieving the specified list of passages from the collection.

1.2 Mixed-Initiative Conversational Search

A mixed-initiative paradigm of conversational search offers potential solutions to some of the aforementioned challenges [Radlinski and Craswell, 2017]. Under

this paradigm, MICS system can at any point take initiative and prompt the user with a question or a suggestion [Walker and Whittaker, 1990]. One of the most prevalent use cases of mixed-initiative is clarification [Braslavski et al., 2017], as users often submit ambiguous or faceted queries to IR systems. For example, a user who wants to find information about different species of kiwi birds, might issue a query “kiwi” to a modern search engine. However, as “kiwi” can refer to a fruit, a bird, an airline, a New Zealander, and more, it is thus unclear to an IR system which documents to present to the user. *Ad hoc* retrieval in search engines tackle the issue by result diversification [Jiang et al., 2015], i.e., presenting several documents for each of the query facets. However, conversational search system does not have that luxury, and should therefore clarify the user information need by taking initiative and asking clarifying questions. In the example above, an ideal MICS system might ask the user "Are you interested in kiwi fruit, bird, or an airline?". It then needs to comprehend the user’s answer, which would provide the system with additional information about user’s need.

Asking clarifying questions has been shown to improve retrieval effectiveness, as the initial query is enriched by the additional information provided by the user in response to the question [Aliannejadi et al., 2019b]. However, asking the right clarifying question is not straightforward. A line of research attempts to select an appropriate clarifying question from a curated pool of such questions [Aliannejadi et al., 2020b, Owoicho et al., 2022], which remains limited to pre-defined topics, thus poorly mimicking a real-world scenario where user can issue a query on a large variety of topics. Generating clarifying questions therefore imposes itself as a solution to the problem, where researchers, for example, construct template-based questions from facets extracted from query logs [Zamani et al., 2020a]. In this dissertation, we present our work on the usage of generative large language models (LLMs) for the task of generating clarifying questions [Sekulić et al., 2021]. We base these questions in query facets and further provide an analysis of the naturalness and usefulness of the generated questions, as well as the feasibility of extracting appropriate query facets from a list of documents retrieved in response to user’s query.

Recent research focuses not only on clarifying question construction, but also on analyzing their benefit to the user [Zou et al., 2023], their cost in terms of user’s effort [Azzopardi et al., 2022], and their usefulness in terms of information gain [Rosset et al., 2020]. In this dissertation, we do not explicitly model the user’s effort in answering clarifying questions; but rather explore methods for predicting user’s engagement with clarification prompts [Sekulić et al., 2021], with an aim of identifying useful questions, as well as potentially ambiguous queries requiring clarification.

Furthermore, processing the given answer to the question is an additional challenge, as naïve methods that simply concatenate the question and the answer with the initial query can possibly hurt retrieval effectiveness, especially in the case of bad questions and uninformative answers. Thus, estimating usefulness of the posed question and of the given answer is beneficial to the system [Sekulić et al., 2024b], as well as thoughtfully processing user’s explicit feedback [Owoicho et al., 2023].

1.3 User Simulation

Evaluating conversational information access systems is arduous [Penha and Hauff, 2020, Fu et al., 2022]. The challenge arises from expensive and time-consuming user studies required for holistic evaluation of conversational systems [Deriu et al., 2021]. Such studies require real users to interact with the search system in several conversational turns and assess the system’s ability to satisfy their information need. The matter is further complicated under the mixed-initiative paradigm, as system’s proactive behavior, combined with user’s undeterministic answers, can lead to potentially unlimited number of conversational trajectories [Balog, 2021]. Thus, due to the requirement of real users interacting with the systems we want to evaluate, and potentially compare to each other, the approach based on user studies is unscalable.

A relatively simple solution is to conduct offline corpus-based evaluation [Dalton et al., 2020, Aliannejadi et al., 2019b]. However, this approach limits the evaluation to pre-defined conversational trajectories [Dalton et al., 2020] or models mixed-initiative through pre-defined pool of clarifying questions [Aliannejadi et al., 2019b]. As such, the corpus-based approach does not capture the real-world scenario, where, as mentioned, user-system interaction can take on multiple paths. Moreover, such offline evaluation remains limited to single-turn interaction, as the pre-defined questions are associated with corresponding answers and unaware of previous interactions.

User simulation has been proposed to tackle the shortcomings of corpus-based and user-based evaluation methodologies [Balog, 2021]. A simulated user aims to capture the behavior of a real user, i.e., being capable of having multi-turn interactions on unseen data, while still being scalable and inexpensive like other offline evaluation methods [Balog and Zhai, 2023]. For example, the simulated user should therefore assume all of the user’s actions presented in Figure 1.1.

In this dissertation, we outline a thread of studies focusing on user simulation for conversational systems. Specifically, we base our simulation approaches

in state-of-the-art large language models (LLMs), thus utilizing their text generation capabilities. We first propose User Simulator, USi [Sekulić et al., 2022], capable of generating answers to clarifying questions posed by the system. We show that generated utterances are both in line with the given information need and in coherent natural language. However, USi remains limited to single-turn interactions and generating only one type of utterances — answers to clarifying questions. We overcome these challenges with Conversational Simulator, ConvSim [Owoicho et al., 2023]. ConvSim is capable of multi-turn interactions with the system and generating different types of utterances, including explicit feedback to system’s responses. In Chapt. 6, we describe in detail both approaches, evaluate the possibility of their utilization in evaluating conversational search systems, and discuss the advantages and limitations of both.

The challenge of evaluating conversational systems extends to task-oriented dialogue systems as well, where the user is interested in completing a task, such as booking a flight or cooking a specific dish [Gemmell et al., 2022]. Although a set of actions that the user needs to take in order to fulfill their goal is relatively defined, thus enabling construction of agenda-based user simulators [Schatzmann et al., 2007], LLM-based simulators have recently been proposed for the task [Lin et al., 2021a]. Specifically, in-context learning approaches to user simulation have proved successful for simulating users [Terragni et al., 2023, Davidson et al., 2023]. However, such approaches suffer from hallucination, thus impairing the reliability of the simulators to be used for TOD system evaluation. To mitigate this limitation, we propose an approach based in fine-tuning state-of-the-art LLMs on in-domain data consisting of user-system conversations [Sekulić et al., 2024a].

1.4 Main Contributions

Main contributions include modeling clarification in mixed-initiative conversational search systems and LLM-based user simulation methods for conversational systems. The contributions are measured through scientific advancements in terms of published research, as well as through released resources, such as code and data collections.

First, we present a novel approach to generating facet-based clarifying questions, based in LLMs. We additionally analyze in depth several aspects of clarification in search, including facet extraction and user engagement prediction, shedding a new light onto the topic of such a broad and current interest. As mentioned above, designing appropriate search interfaces that provide precise

and concise response is an important challenge in CIA. To this end, we propose a response rewriting approach based in entity salience, with a goal of making the system’s response self-contained and thus understandable by wide audiences.

We design and release user simulation methods for interacting with MICS systems. Specifically, we first release a method capable of assuming the user’s role in answering clarifying questions, followed by its expansion to providing explicit feedback and maintaining multi-turn interactions. We demonstrate its feasibility through a series of experiments involving both automated performance assessments and human annotations. Furthermore, we propose a fine-tuning based approach for simulating users for TOD systems, showing its superiority over out-of-the-box LLM approaches for in-domain data through an extensive set of experiments.

We open-source our code, model weights, and annotated data for a number of the aforementioned contributions, with the links to the resources presented in the corresponding chapters of this document.

1.5 Dissertation Outline

This dissertation is organized in three parts and nine chapters. The next chapter provides relevant background information, defines terminology used throughout the dissertation and reviews literature on the topic of conversational information access and related research fields.

Part I, consisting of three chapters, concerns itself with various topics of mixed-initiative conversational search. Chapter 3 addresses clarifying questions in conversational search systems. We first describe our approach to facet-based clarifying question generation. However, as such facets are not easily obtainable, we provide an analysis for the possibility of acquiring them from the list of documents retrieved in response to the initial query. Moreover, we propose a generative model for automatically extracting appropriate facets from such lists of retrieved documents. Finally, we propose an effective and efficient method for utilizing posed clarifying questions and the corresponding user answers for improving retrieval performance, based on assessing usefulness of the dialogue components. Next, in Chap. 4, we propose a method for predicting users’ engagement with clarification panes in search engines, finding that the list of documents retrieved in response to the initial query provides a strong signal for the predictor. Finally, Chapt. 5 describes our approach for rewriting system’s responses in CIA, with a goal of making them self-contained and thus understandable by wide audiences.

Part II is devoted to user simulation. Chapter 6 describes large language model-based approach to generative user simulation for conversational search. We first propose a user simulator for answering clarifying questions. Next, we expand the proposed model by enabling multi-turn interactions with the search system, as well as adding a feedback-providing feature. Chapter 7, on the other hand, proposes a user simulator for task-oriented dialogue systems. To this end, we fine-tune a large language model on relevant domain data, leading to more reliable and faithful generation.

Part III concludes the dissertation. Chapter 8 is focused on the discussion of the main findings of our studies and their comparison to contemporary research, meanwhile acknowledging the limitations of the presented work. Next, Chapt. 9 concludes the dissertation by outlining its main contributions and promising directions for future research. Finally, Appendix A includes unfinished novel graph-based approaches to conversational search.

1.6 Publication Overview

This dissertation sources from multiple relevant papers, listed below. Some are already published, while others are currently under review.

Chapter 3 is based on:

- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani. "Towards Facet-Driven Generation of Clarifying Questions for Conversational Search." In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR'21*, pages 167-175. 2021.
- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani. "Exploiting Document-Based Features for Clarification in Conversational Search." In *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR'22*, pages 413-427. 2022.
- Sekulić, Ivan, Weronika Łajewska, Krisztian Balog, and Fabio Crestani. "Estimating the Usefulness of Clarifying Questions and Answers for Conversational Search." In *Advances in Information Retrieval - 46th European Conference on IR Research, ECIR'24*, pages 384-392. 2024.

Chapter 4 is based on:

- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani. "User Engagement Prediction for Clarification in Search." In *Advances in Information Re-*

trieval - 43rd European Conference on IR Research, ECIR'21, pages 619–633. 2021.

Chapter 5 is based on:

- Sekulić, Ivan, Balog, Krisztian and Crestani, Fabio. "Towards Self-Contained Answers: Entity-Based Answer Rewriting in Conversational Search." In *Proceedings of the 2024 Conference on Conference Human Information Interaction and Retrieval*, CHIIR'24, pages 209-218, 2024.

Chapter 6 includes:

- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani. "Evaluating Mixed-Initiative Conversational Search Systems via User Simulation." In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM'22, pages 888-896. 2022.
- Owoicho, Paul, Ivan Sekulić, Mohammad Aliannejadi, Jeffrey Dalton, and Fabio Crestani. "Exploiting Simulated User Feedback for Conversational Search: Ranking, Rewriting, and Beyond." In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'23, pages 632-642. 2023.
- Sekulić, Ivan, Mohammad Aliannejadi, and Fabio Crestani. "Analyzing Utterances in LLM-based User Simulation for Conversational Search." In *ACM Transactions on Intelligent Systems and Technology*, TIST, 2024.

Chapter 7 is based on:

- Sekulić, Ivan, Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, André Ferreira Manso, and Roland Mathis. "Reliable LLM-based User Simulator for Task-Oriented Dialogue Systems." In *Workshop on Simulation of Conversational Intelligence in Chat*, SCI-CHAT'24 at EACL'24, 2024.

Other Research Outcomes. Other research outcomes, listed below, are not included in this dissertation. The reasons for exclusions are diverse: insignificant contribution (e.g., a shared task participation), a minor contribution to the publication, research outcomes that do not fit into the topic of the thesis, or early-stage research that required more time to be properly included:

- Sekulić, Ivan, Fabio Crestani, and Mohammad Aliannejadi. "Extending the Use of Previous Relevant Utterances for Response Ranking in Conversational Search." In *Proceedings of the Text REtrieval Conference (TREC)*, NIST, 2020.
- Sekulić, Ivan, Amir Soleimani, Mohammad Aliannejadi, and Fabio Crestani. "Longformer for MS MARCO Document Re-ranking Task." In *Proceedings of the Text REtrieval Conference (TREC)*, NIST, 2020.
- Łajewska, Weronika, Nolwenn Bernard, Ivica Kostrić, Ivan Sekulić, and Krisztian Balog. "The University of Stavanger (IAI) at the TREC 2022 Conversational Assistance Track." In *Proceedings of the Text REtrieval Conference (TREC)*. NIST, 2022.
- Mackie, Iain, Ivan Sekulic, Shubham Chatterjee, Jeffrey Dalton, and Fabio Crestani. "GRM: Generative Relevance Modeling Using Relevance-Aware Sample Estimation for Document Retrieval." In *arXiv preprint arXiv:2306.09938*, 2023.
- Litmimov, Oleg, Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. "Analyzing Coherency in Facet-based Clarification Prompt Generation for Search." In *arXiv preprint arXiv:2401.04524*, 2024.
- Sekulić, Ivan, Lili Lu, Navdeep Singh Bedi, and Fabio Crestani. "Towards Realistic User Simulator for Conversational Search: Parameterizing Behavior Traits via In-Context Learning." Under review.

Unpublished Research. This dissertation additionally includes unpublished research, briefly outlined and discussed as potential future directions in Appendix A:

- Participation in SCAI-QReCC as team *torch*: <https://www.tira.io/task/scai-qrecc#>
- Sekulić, Ivan, Vincent Herrmann, Daniele Grattarola, Fabio Crestani. "Towards Multi-Document Information Retrieval with Graph Neural Networks." *Unpublished, 2021*.
- Mackie, Iain, Ivan Sekulić, Jeffrey Dalton, and Fabio Crestani. "Exploring Complex Topics: Document and Entity Graph Construction for Ranking." *Unpublished, 2021*.

Chapter 2

Literature Review and Background

This dissertation relates to several topics of information retrieval (IR) and natural language processing (NLP). Specifically, we explore several challenges tied to conversational IR (a.k.a. conversational search), with a focus on mixed-initiative paradigm of conversational search. These include conversational passage retrieval, clarifying question generation, user engagement prediction for clarification, processing answers to clarifying questions, and constructing self-contained responses.

Moreover, the dissertation focuses on user simulation approaches for evaluating conversational systems. Thus, we discuss challenges in the evaluation of conversational systems, with user simulation as one of the potential solutions to those challenges. Further, we outline recent research on user simulation for conversational search systems, as well as for task-oriented dialogue (TOD) systems. In this chapter, we review the relevant literature on the aforementioned topics and formally define specific tasks of such conversational systems. First, however, we provide a formal definition of conversational search used in the document and denote commonly used terminology.

2.1 Formalizing Conversational Search

In this section, we provide an overview of the terminology, definitions, and concepts covered throughout this dissertation. We first define a conversation, followed by a definition of conversational information access and, specifically, conversational search. Moreover, we formalize the interactions between users and conversational search systems, listing their roles and functionalities. Finally, we outline the foundations of our experimental setup, including a breakdown of

commonly used retrieval pipelines, as well as our usage of large language models.

2.1.1 Conversation

In this dissertation, we constrain our research on conversations between two participants: a user U and a system S . In other words, we do not model multi-agent conversations, where potentially multiple users interact between each other and with the conversational system. Therefore, we differentiate between user’s utterances u and system’s utterances s . The interaction between the two agents evolves over several conversational turns, where at turn t the utterances are denoted as u^t or s^t . Moreover, we note that we model the conversational turns as adjacency pairs. In linguistics, [Schegloff and Sacks \[1973\]](#) define an adjacency pair as two utterances, uttered by two speakers, where the first one provokes the second, responding, utterance. Thus, user’s utterance u^t is always followed by system’s utterance s^{t+1} . We base our decision for modeling conversations through adjacency pairs in the fact that users tend to seek certain information from the systems and thus prompt it with various questions, queries, and requests. These prompts require a direct response from the system. To conclude, throughout this document, we model conversations between two participants, a user and a system, where each of the participants responds with a single utterance at a time, which is then addressed by the other participant. This exchange formulates the conversational history $H = [u^1, s^2, u^3, s^4, \dots, u^{t-2}, s^{t-1}]$.

2.1.2 Types of Utterances

In user’s interaction with a mixed-initiative conversational search system, we differentiate between different discourse types of the utterances. Specifically, we differentiate between three discourse types of user utterances u :

- (i) queries (or questions) u_q ;
- (ii) answers u_a to clarifying questions posed by the system;
- (iii) explicit feedback u_f to the system’s responses.

Similarly, the system’s utterance s can either be a response s_r aimed at satisfying the user’s information need IN or a clarifying question s_{cq} aimed at elucidating the user’s information need. In the light of adjacency pairs, at turn t , for example, we always expect an answer u_a^t to system’s clarifying question s_{cq}^{t-1} .

2.1.3 Search

Search, as the center of information retrieval, includes retrieving documents relevant to the user’s query from a given collection. The collection C is composed of a large number of documents, where a single document is denoted as d . We note that the collection can also be composed of passages, instead of documents, which are typically longer and composed of multiple passages. The task of the IR component of the conversational search system is thus to retrieve a list of documents $D = [d_1, d_2, \dots, d_i, \dots, d_N]$, where d_i is a document from C and N is the maximum number of documents considered. The documents are sorted by their relevance to u_q in a descending order.

Established approaches to document retrieval, also used in the studies presented in this dissertation, are usually centered around the retrieve & re-rank pipeline. First, computationally inexpensive method, such as BM25 [Robertson and Jones, 1976] or Query Likelihood [Ponte and Croft, 1998], retrieves N documents an indexed collection C . While these methods rely on sparse representations of documents and queries, dense retrieval methods have recently been proposed as a more semantically-rich representation of both the documents and the queries [Zhao et al., 2022]. While both can be used as an initial retrieval step, recent research shows the benefit of a hybrid approach, where the final list of N documents is curated by taking into account both sparse and dense predicted relevance [Luan et al., 2021]. Second, after the first-pass retrieval, documents are re-ranked with a computationally expensive method, such as neural re-ranker or LLM-based re-ranker [Nogueira and Cho, 2019]. Recently, a third step, involving pairwise comparisons of top M documents, where $M \ll N$ (usually tens of documents), has been introduced to improve retrieval effectiveness [Pradeep et al., 2021].

While IR usually concerns itself strictly with retrieving the aforementioned list of documents, search can instead return a ranked list of results $R = [r_1, r_2, \dots, r_i, \dots, r_N]$. Such list can be identical to D , or modified by tailoring the responses to the user or the search setting (e.g., conversational setting). For example, SCAI-QReCC shared task [Vakulenko et al., 2022], in which we participated, required a response s_r fit for a conversational setting, as well as the list D supporting this response. Similarly, TREC CAsT’22 [Owoicho et al., 2022] evaluate both the ranked list of responses R , which can be tailored by any means deemed useful, in addition to the list of documents D retrieved directly from the given collection C .

2.2 Conversational Search

Recently, conversational artificial intelligence has attracted significant attention, from both industry and academia. A contributing factor to this attention is the rise of conversational assistants (e.g., Alexa and Siri), as well as the most recent state-of-the-art large language models (LLMs) (e.g., ChatGPT). In the research community, the attention comes from the fields of natural language processing (NLP) and information retrieval (IR), covering a wide range of different types of conversational systems. Thus, in this section, we make a distinction between the types, focusing on conversational search. Conversational search falls under conversational information access (CIA) [Balog, 2021], also referred to as conversational information seeking [Zamani et al., 2022]. However, CIA additionally encapsulates any conversational systems designed to satisfy user’s information need, such as conversational question and answering (Q&A) systems and conversational recommender systems [Zamani et al., 2022]. In this dissertation, we focus mostly on conversational search, sometimes referred to as conversational IR, and therefore cover the related topics in greater depth.

It should be noticed that one of the first works in conversational IR dates back to 1987 when Croft and Thomson [Croft and Thompson, 1987] proposed I³R, which served as an expert intermediary system by interacting with the user during a search session. Another one of the early works on the topic, Belkin et al. [1995], studies information-seeking strategies for conversational search, utilizing case-based reasoning for offering choices in a search session.

2.2.1 Defining Conversational Search

While definitions of conversational search slightly differ in recent literature [Zamani et al., 2022], we focus on the definitions centered around satisfying user’s information need. Similar to *ad hoc* search, users express their information needs through a query, or a series of queries. The system’s task is then to retrieve documents relevant to the given query from a given collection. However, as opposed to *ad hoc* search, conversational search is conducted in a conversational manner through a series of topically connected utterances in natural language. In this dissertation, we focus on text-based utterances, which are either written directly by the user, or spoken and transcribed. The distinction between conversational Q&A system and conversational search system is primarily in the fact that the latter always contains a retrieval component, that is tasked to compose a ranked list of documents from a given collection, while the former usually provides short answers and does not necessarily utilize an IR component [Zamani et al., 2022].

As mentioned, recent literature defines conversational search through different lenses, stemming from different backgrounds. For example, the typology of conversational search proposed at a recent meeting in Dagstuhl [Anand et al., 2020] base the definition of conversational search systems on three main varieties of systems. Specifically, a conversational search system is defined through functional extensions of IR systems, chatbots, and TOD systems. As such, it extends *ad hoc* IR systems by adding an interactive conversational component; chatbots by adding user task modeling and retrieval; and TOD systems by including a collection of documents and enabling retrieval capabilities.

Another approach to defining conversational search is through its properties and system capabilities. To this end, Radlinski and Craswell [2017] propose a theoretical framework of conversational search, identifying key properties of such systems and focusing on natural and efficient information access through conversations. They define the desired functionalities, including statefulness, user and system revilement, retrieval, and mixed-initiative. Azzopardi et al. [2018] further define a taxonomy of user and systems actions and map them to the aforementioned functionalities. In essence, the user seeks to satisfy its information need through the following set of actions:

- *Reveal* actions, in which a user discloses parts of the underlying information need;
- *Expand* actions, in which a user expands the search space by modifying or removing certain criteria imposed earlier in a conversation;
- *Inquire* actions, in which a user seeks another way of representing the results, e.g., by summarizing the given options, comparing them, or finding similar results;
- *Navigate* actions, in which a user aims to navigate the current search space, by, e.g., revisiting previous options, learning more about them, and similar;
- *Interrupt* actions, in which a user interrupts the system either due to their inquiry about system's response or system's irrelevance.

System actions, complimenting user actions, include:

- *Inquire* actions, in which a system aims to elucidate user's information need by asking for clarifications or eliciting criteria from the user;
- *Reveal* actions, in which a system displays the results to the user;

- *Traverse* actions, in which a system supports navigation through the search space;
- *Suggest* actions, in which a system recommends certain results by hypothesizing about user’s information need;
- *Explain* actions, in which a system reports its understanding of the current information need;
- *Error and Finalization* actions, in which a system is able to recover from faulty responses and is able to finalize the conversation when needed.

Recent research efforts usually focus on modeling one, or a few, set of actions at the time. Similarly, this dissertation is centered around modeling *Inquire* and *Reveal* system actions, through a series of tasks in conversational search, outlined below.

In an attempt to shed light on the conversational search process, [Trippas et al. \[2018\]](#) study spoken interaction between the system and the user. They define three phases of the search process, namely: 1. query formulation; 2. search results exploration; 3. query re-formulation. These high-level phases outline the general search process, out of which we mainly focus on the last one, in the context of mixed-initiative elaborated below.

While some of the challenges remain similar to traditional *ad hoc* search, a significant number of new ones arise under the conversational paradigm. These are surveyed in the recent manuscript of [Zamani et al. \[2022\]](#) and include user intent classification [[Qu et al., 2019](#)], biases in conversational search [[Gerritse et al., 2020](#)], response ranking [[Dalton et al., 2020](#)], and query rewriting [[Yu et al., 2020](#), [Vakulenko et al., 2021](#)]. We review the relevant research on the challenges addressed in this dissertation throughout the following sections.

2.2.2 Query Rewriting

A conversational search system needs to keep track of the conversation and comprehend user’s references to the past utterances. For example, after an initial query “Tell me about yoga.”, user’s following query could be “What are its types?”, thus coreferencing “yoga” from the initial query. While TOD systems usually contain a stateful dialogue state tracking component [[Williams et al., 2016](#)], conversational search systems typically rely on conversational query rewriting (CQR) [[Vakulenko et al., 2021](#)]. Specifically, given a conversational history, containing all of the past queries and system’s responses, the task is to rewrite the

current user’s query with a goal of making it self-contained, resolving potential coreferences and missing information.

Notable approaches include expanding the current query with relevant information from the conversation history, neural-based query rewriting, and dense query representations. For example, [Lin et al. \[2021b\]](#) propose Historical Query Expansion (HQE), a method that aims to quantify the importance of informative terms from conversational history through term frequency-based features. Similarly, [Aliannejadi et al. \[2020a\]](#) identify propose a classifier to assess the usefulness of the past utterances, which are then included to the current query. We follow similar approach in our participation in TREC CAsT’20 [[Sekulić et al., 2020](#)]. However, while the aforementioned methods provide an explainable approach to CQR, as the rewritten query clearly contains useful terms and utterances from the conversational history, its performance is limited.

Neural LLM-based methods achieve significant improvements in the task [[Vakulenko et al., 2021](#)]. [Lin et al. \[2020\]](#) fine-tune several sequence-to-sequence neural models on CANARD dataset [[Elgohary et al., 2019](#)], which contains over 30k training examples consisting of the original query taken from QuAC [[Choi et al., 2018](#)], conversational history, and manually rewritten self-contained query. They find that fine-tuned T5 [[Raffel et al., 2020](#)] significantly outperforms other approaches, such as LSTM-based or BERT-based [[Devlin et al., 2019](#)] methods. The described approach to CQR is frequently used by state-of-the-art conversational search systems across a broad IR research community [[Dalton et al., 2020](#), [Owoicho et al., 2022](#), [Lajewska et al., 2022](#)].

In contrast to the above-described approaches to CQR, which yield an explicit rewrite of the ambiguous utterance, dense retrieval-based approaches aim to embed the current utterance and the conversational history into an information-rich high-dimensional vector. [Yu et al. \[2021\]](#) propose ConvDR, a model that learns contextual embeddings of multi-turn queries, which are then directly used for retrieval.

2.2.3 Conversational Passage Retrieval

Document retrieval is the core of *ad hoc* IR systems. However, as documents can be fairly long, thus potentially exceeding the limited bandwidth available in conversational search, systems are often designed to retrieve passages instead. Conversational Assistant Track (CAsT) [[Dalton et al., 2020](#)] introduces and formalizes the conversational passage retrieval task at the Text REtrieval Conference (TREC), a staple for empirical evaluation in IR. CAsT’19 and CAsT’20, the first and the second editions of the shared task, respectively, formulate the task as a

series of queries on a specific topic. The task is then to retrieve relevant passages from a collection of approx. 18M passages. The performance of each system is then assessed by standard IR metrics, including recall, MRR, nDCG [Harman, 2011].

In both CAsT’19 and CAsT’20, the main challenge is to resolve potential references to the past queries and thus employ the aforementioned CQR methods to obtain a contextualized query rewrite. CAsT’21 makes a step towards more realistic conversational search and introduces canonical responses to each of the queries, which can then be referred to as well, as the conversation progresses. CAsT’22 [Owoicho et al., 2022] makes a step further and allows the system to, for each query, ask a clarifying question, which can then potentially disambiguate the query and add additional useful information.

TREC interactive Knowledge Assistance Track (iKAT) [Aliannejadi et al., 2024], a successor to TREC CAsT, additionally includes user persona description. Thus, the task of passage retrieval is conditioned on the personal contexts stated in the description. This means that, given the same topic, different personas will result in different conversational trajectories when interacting with a search system. The iKAT task is a step towards a realistic real-world scenario.

Once the contextualized rewrite is obtained, most approaches rely on *ad hoc* methods for passage retrieval. Most widely used approach is a two-step retrieve and re-rank approach, where a computationally inexpensive method is first used to retrieve a number of passages, which are then re-ranked with a computationally heavier method. Nogueira and Cho [2019] demonstrate the effectiveness of such an approach, where the first stage retrieval is performed with BM25 method. Next, top 1000 passages retrieved are re-ranked with BERT [Devlin et al., 2019], which is fine-tuned on query-passage pairs with their relevance as the prediction target. Their approach significantly outperforms previous re-ranking methods, achieving state-of-the-art results on multiple benchmarks, and sparks its adoption to several passage ranking tasks, including conversational passage retrieval [Dalton et al., 2020]. An extension of the retrieve and re-rank approach is Expando-Mono-Duo pipeline [Pradeep et al., 2021], where authors employ an additional re-ranking step. Specifically, after the first re-ranking step, a pairwise re-ranking model compares each of the top 10 passages to each other, yielding a more accurate ranked list of passages.

While passage retrieval is an essential component of almost any conversational search system, we do not tackle it explicitly in this dissertation. Rather, we use it as a basic component of our systems and assess the performance of different systems through measured retrieval effectiveness.

2.2.4 Result Representation

Presenting results, i.e., responding to user’s queries, in a conversational setting requires special considerations due to limited bandwidth the search is undertaken in. For example, it is unreasonable to expect a user on a mobile phone to scroll through multiple retrieved passages in pursuit of the relevant information. An alternative to a ranked list of passages is a single passage, which can either be selected from the collection or generated by aggregating information from multiple passages. [Vakulenko et al. \[2022\]](#) construct a dataset of multi-turn interactions between a user and a system, where system’s response is constructed from potentially multiple passages, summarizing them in order to tailor the response directly to the user’s perceived information need. The evaluation of system’s performance is therefore twofold: 1. based on generated system’s response, done by human annotators; 2. based on retrieved list of passages supporting the generated response, done by computing standard IR metrics. [CAsT’22 \[Owoicho et al., 2022\]](#) adapt a similar scenario, with an addition of allowing multiple responses per query. Then, the responses are ranked and assessed accordingly by human annotators. The described approach is recently adapted in the NLP community as well, and is usually referred to as retrieval-augmented generation [[Lewis et al., 2020](#)].

In this dissertation, we do not explicitly focus on summarizing multiple passages to create a concise response, but rather tackle response rewriting with the purpose of making sure a user understands the response to their question, To the best of our knowledge, this direction in CIA has not been explored. However, researchers have studied text rewriting in IR for personalization and text simplification. While text simplification has been shown to improve readability and understanding in medical [[Leroy et al., 2013](#)] and scientific texts [[Ermakova et al., 2022](#)], it is usually done by swapping relatively unfamiliar words with more common alternative words [[Leroy et al., 2013](#)] or leveraging large-scale language models for complete rewriting of the text [[Sheang and Saggion, 2021](#)]. In this setting, a certain degree of information distortion is acceptable, as the text rewritten with such methods might differ from the original due to word substitutions. On the other hand, in [Chapter 5](#), we aim to allow the user to learn about a topic of interest, thus retaining the original terminology [[Sekulić et al., 2024a](#)].

2.3 Mixed-Initiative

One key element of conversational search is mixed-initiative, which is the interaction pattern where both the system and the user have rich forms of interaction. Under the mixed-initiative paradigm, conversational search systems can at any point of conversation take initiative and prompt the user with various questions or suggestions. Mixed-initiative has a long history in dialogue systems with [Walker and Whittaker \[1990\]](#) identifying it as an integral part of conversations and [Horvitz \[1999\]](#) identifying key principles of mixed-initiative interactions. These principles include the necessity to consider uncertainty of user’s goals, employing dialogue to resolve such uncertainties, keeping track of recent interactions, and continuous learning by observing. Moreover, some principles describe the necessity of timely actions, where the system needs to be mindful of user’s attention when taking initiative, while still allowing direct interruption and termination by the user.

While mixed-initiative is a relatively well established concept in the IR community [[Allen et al., 1999](#)], recent advancements in CIA systems have introduced mixed-initiative into conversational search. Specifically, multiple studies have demonstrated the effectiveness of asking clarifying questions with a goal of elucidating the underlying user’s information need [[Braslavski et al., 2017](#)]. In the next section, thus, we focus on clarification in search.

2.3.1 Clarification in Search

Clarifying the user information need is an important aspect of any IR system, while its importance is especially emphasized in a conversational setting. *Ad hoc* IR systems deal with ambiguous and faceted queries by result diversification, where users would be presented with relevant documents for several different aspects of the query, enabling users to scroll through them to find what they need [[Santos et al., 2015](#)]. Recently, [Zamani et al. \[2020c\]](#) report highly positive feedback from users engaged with the clarification features in a modern search engine. In the same line of work, the authors release MIMICS, a large-scale collection of datasets for studying clarification in search. The datasets are composed of real-world queries, taken from Bing search engine, which are associated with the constructed clarification panes and annotated for their quality and perceived usefulness.

In contrast to search engines, in CIA, the emphasis on clarification comes from the fact that conversational search is often carried out in limited-bandwidth scenarios, such as speech-only or mobile interfaces [[Aliannejadi et al., 2019a](#)], thus

making it impossible to present a large range of the results the user. Clarification has thus attracted considerable attention of the research community, including studies on human-generated dialogues in question answering (Q&A) [De Boni and Manandhar, 2003, 2005] and asking clarifying questions [Braslavski et al., 2017].

Asking clarifying questions has been shown to be beneficial for the conversational search system and the user. E.g., Kiesel et al. [2018] studied the impact of voice query clarification on user satisfaction and found that users like to be prompted for clarification. Moreover, Aliannejadi et al. [2019b] proposed an offline evaluation methodology for asking clarifying questions and showed the benefits of clarification in terms of improved performance in document retrieval once question is answered. They construct Qulac, a dataset consisting of question-answer pairs for faceted and ambiguous queries. A similar methodology has been applied in the ClariQ challenge [Aliannejadi et al., 2020b], where the task is to select the most appropriate clarifying question from a pre-defined set of questions. They find that asking the right question can lead to significant improvements in retrieval performance.

Zou et al. [2020] conduct an empirical study on users willingness to respond to clarifying questions and their usefulness perceived by users, concluding that most users are willing to answer 6-10 yes-or-no questions in a commercial setting. Similarly, a “System Ask-User Respond” framework has been proposed by Zhang et al. [2018] for product recommendation in e-commerce, where questions about certain aspects of a query are prompted to the user to clarify their needs. However, Zou et al. [2023] find that prompting the user with poor-quality clarifying questions leads to decrease in user satisfaction and poor search performance. Thus, the authors suggest that it might be better not to ask any clarifying questions, than to ask a poorly-constructed one.

While there are benefits to clarifying questions, there is also cost to the user for these interactions [Azzopardi, 2011, Zou et al., 2023]. Moreover, user can clarify their query through other types of feedback, such as explicit feedback, not just by answering clarifying questions directly [Owoicho et al., 2023]. For example, Azzopardi et al. [2022] propose two economic models for conversational search, based on when user feedback is given and how it impacts the search performance, given the user effort. In this dissertation we mainly focus on the effectiveness of clarifying questions and do not study user costs directly.

In general, two streams of approaches to constructing clarifying questions exist. The first line of approaches aim to select an appropriate question from a pre-defined pool of questions [Aliannejadi et al., 2019b, 2020b, Owoicho et al., 2022, Rosset et al., 2020, Rao and Daumé III, 2018], while the second one aims

to generate the question [Zamani et al., 2020a, Sekulić et al., 2021, Majumder et al., 2021]. We review them in the following sections.

However, despite the abundance of research on clarifying question construction [Rahmani et al., 2023], research aimed at processing users' answers to such questions is scarce. To bridge this gap, Krasakis et al. [2020] conduct an analysis of users' answers and find that they vary in polarity and length. Moreover, they report that incorporating information contained in the questions and the answers often hurts the retrieval effectiveness. We expand on this work in Sekulić et al. [2024b], where we aim to automatically assess the usefulness of both the questions and the answers, with a goal of mitigating this undesired effect.

Zamani et al. [2020a] suggest a taxonomy of clarifying questions, inspired by existing query suggestion taxonomies [Boldi et al., 2011, Jansen et al., 2009]. As such, clarifying questions can serve the purpose of disambiguating the user's query, narrowing its topic, comparing the topic or entity to another one, or identifying preference, such as asking for user's personal, spatial, or temporal information.

2.3.2 Selecting Clarifying Questions

Aliannejadi et al. [2019b] release Qulac, a collection of clarifying questions for a set of faceted and ambiguous queries. Specifically, they utilize crowdsourcing to acquire a selection of clarifying questions for each query and its facet in TREC Web Track 09-12 collections. Further, they formulate a task of ranking clarifying questions from the constructed pool of questions, according to their relevance to the initial query. Aliannejadi et al. [2019b] additionally utilize crowdsourcing to answer these clarifying questions, yielding a valuable resource of question-answer pairs for faceted and ambiguous queries.

In their follow-up work, Aliannejadi et al. [2020b] propose a shared task of selecting clarifying questions. For that purpose, they release ClariQ, a dataset based in Qulac, expanded by introducing unambiguous queries. The shared task includes measuring retrieval effectiveness once the clarifying question was answered. Both Aliannejadi et al. [2019b] and Aliannejadi et al. [2020b] find that retrieval performance increases with the additional information and outline common approaches to clarifying question selection, which are mostly based on pair-wise semantic matching between the query and each of the questions with large transformer-based methods, e.g., BERT. In this dissertation, we rely on both Qulac and ClariQ for a number of studies on clarification in search.

To additionally improve question selection effectiveness, Mass et al. [2022]

propose to utilize information available in the passages retrieved in response to the initial query. This approach was further improved by Hashemi et al. [2020], who propose GuidedTransformer that leverages information from conversation history, retrieved documents, and potential clarifying questions. Their approach yields significant improvements in the question selection task on Qulac.

Rao and Daumé III [2018] propose to select questions on their perceived expected value of perfect information (EVPI), a measure of the value of gathering additional information. To this end, they compute which questions are most likely to elicit an answer that would make the post more informative. On a similar note, Rosset et al. [2020] propose a BERT-based [Devlin et al., 2019] model aimed at selecting useful clarifying questions. They argue that while clarifying question can be on topic of the initial query, it does not necessarily mean it is useful for the search session. For example, questions can miss user intents, be too specific, too broad, or duplicate the query. We adopt the proposed notion of usefulness [Sekulić et al., 2021].

2.3.3 Generating Clarifying Questions

Generating questions has been studied by the NLP community [Rao and Daumé III, 2018, 2019], where the task is to produce questions about a given document, rather to clarify user information need.

In IR, Zamani et al. [2020a] proposed reinforcement learning-based models for generating clarifying questions and the corresponding candidate answers from weak supervision data. They rely on template-based and a sequence-to-sequence generative models and extract query aspects from 1.6 billion query reformulations from Bing logs. As such logs are not publicly available, we assume that query facets can be extracted from the collection itself, by employing a suitable facet extraction method. However, their approach is not entirely fit for conversational search. Specifically, Zamani et al. [2020a] propose clarifying question generation for search engines in a form of clarification panes. Clarification pane consists of a general question and several clickable answers, which makes it unusable in purely conversational setting.

Interaction naturalness has been pointed out to be an important property of a conversational search system [Anand et al., 2020], as interactions in natural language are distinguished from the ones driven by keywords in classical IR. Thus, the generated clarifying question needs to be in coherent and natural language. We address the problem of generating clarifying questions in Chapter 3.

Rosset et al. [2020] tackle the task of question suggestion in a “People Also Ask” search engine setting. They argue that a useful question is not simply re-

lated to the topic of a user’s query, but should also be “conversation leading” and provide meaningful information for the user’s next step. They propose a BERT-based and a generative GPT-2-based model for question suggestion. They find that questions generated by GPT-2 are syntactically correct, but less useful than the ones selected by BERT from a pre-defined pool of questions. The authors suggest that a reason for the inferior performance of GPT-2 might be due to the lack of explicit guidance in semantics. We overcome this shortcoming by grounding our question on query facets.

Natural Language Generation

The rise of large pretrained language models brought significant progress in various tasks of IR and NLP, including natural language generation (NLG). One of the most prominent models is GPT [Radford et al., 2019], with variations as GPT-2, GPT-3, and, more recently, instruction-tuned ChatGPT variations. GPT models are deep autoregressive generative models, trained on large amount of textual data, which makes them an extremely powerful natural language generators. Early work includes a hierarchical recurrent encoder-decoder for generative context-aware query suggestion [Sordoni et al., 2015]. There are also several attempts in generating a more controllable text, e.g., CTRL [Keskar et al., 2019] and Grover [Zellers et al., 2019]. Closer to our work of conditioning language generation on several keywords and a query is PPLM [Dathathri et al., 2019]. PPLM steers the language generated with a decoding scheme using keywords and classifiers. Moreover, Peng et al. [2020] propose a semantically-conditioned GPT for conversational response generation from dialogue acts.

Facet Extraction

Facet extraction has previously been studied in the IR field. Notably, Kong and Allan [Kong and Allan, 2013] developed a graphical-based model for extracting facets from the set of candidate terms. The candidate set is extracted from the documents retrieved in response to a query. Moreover, they formally define the difference between query subtopic/aspect, semantic class, and a facet. Furthermore, Deveaud et al. [2014] proposed a Latent Concept Modeling (LCM) method that aims to understand the conceptual view of user’s information need through modeling the search concepts in a latent space. They base their method on LDA model that identifies specific query-related topics from the top K documents retrieved, where the topics are latent variables. Query facets can also be extracted from search engine query logs [Zamani et al., 2020a], query autocom-

pletion [Salle et al., 2021], or knowledge graphs [Feddoul et al., 2019]. We note that any of these methods can yield query facets suitable for the input of our proposed clarifying question generation model [Sekulić et al., 2021].

In Sect. 3.2, we take ideas from the described LDA-based approach, with a difference that our topic representation needs to become explicit, rather than latent, in order to serve as grounding for clarifying questions. So, we explore various ranking- and document-based features for facet extraction and question generation. Unlike past work, we rely on neither human-generated question banks nor large-scale query logs, but generate clarifying questions automatically from the extracted features. Inspired by the related work on facet extraction, we examine the effect of LDA-based features, as well as others such as noun phrases and entities that appear on the top of the ranked list.

2.3.4 Entity-Oriented Search

Marchionini [2006] categorizes search activities in two broad categories: look-up questions and exploratory search, with the latter requiring carefully curated user interaction [Cámara et al., 2021]. One of the most notable datasets in the space of web search is the Google Natural Questions dataset [Kwiatkowski et al., 2019], which contains queries from real users with manually evaluated responses. During their exploratory web search, users often have the possibility to learn about entities of their interest by following hyperlinks or reformulating their query based on newly seen entities [Eickhoff et al., 2014]. Entity linking and entity-based search are core component in that process [Balog, 2018]. Thus, significant research efforts were put into developing entity linking methods, including entity linking in the Web [Han et al., 2011], in free texts [Piccinno and Ferragina, 2014], and in CIS [Joko et al., 2021].

While documents may contain a large number of entities, some of them are salient, thus central to modeling the aboutness of a document [Paranjpe, 2009], and others are not. Moreover, Gamon et al. [2013] find that only about 5% of the entities in Web pages are salient, while others are often mentioned somewhat sporadically. These salient entities are crucial for the user to be familiar with, in order to satisfy their information need. However, in shorter texts that contain fewer entities, this percentage is anticipated to be higher [Wu et al., 2020]. Answers in CIS are a prime example of such shorter texts. Yet, research on entity salience in CIS is lacking, providing a strong motivation for our work [Sekulić et al., 2024a].

Another aspect of entity salience we aim to explore is how important they

are for the user’s understanding of the texts and readability [Collins-Thompson, 2009]. There is an important distinction to be made between entity salience and entity *relevance* or entity *importance* [Gamon et al., 2013]. For example, *Joe Biden* is objectively an important entity, however, it can be marginal to the document’s topic. Moreover, entity relevance is dependent on the user’s intent and their underlying information need. On the other hand, an entity is salient to a document if it is central and important for the overall topical and informational coherency of the document. Thus, we argue that salient entities are essential to know about for a complete understanding of the provided answers in CIS. In Chapt. 5, we explore their prevalence, characteristics, and ways of improving user experience via answer rewriting around identified salient entities.

2.4 User Engagement

O’Brien and Toms [2008] define user engagement as the quality of user experience in interaction with a system, characterized by various attributes, e.g., positive affect, aesthetic and sensory appeal, attention, novelty, perceived user control. In their recent study [O’Brien et al., 2020], they point user engagement as an important outcome measure in interactive IR research. User engagement has previously been studied in the context of commercial software, social media [Di Gangi and Wasko, 2016], online news [O’Brien, 2017], student engagement with online courses [Dhall et al., 2018], and applications for monitoring health-related signals [Alkhaldi et al., 2016].

User engagement in the aforementioned studies has usually been measured by self-reported questionnaires, facial expression analysis or speech analysis, signal processing methods, or web analytics [Lalmas et al., 2014]. Recently, Zamani et al. [2020b] created a collection of datasets for studying clarification in search by aggregating user interactions with clarification pane in a major commercial search engine, thus falling into the category of measuring the user engagement by web analytics. In this dissertation however, instead of estimating the engagement levels with a goal of advancing search engine clarification feature, we analyze the implicit signals of the interactions that contain valuable information about the ambiguity of the query, diversity of retrieved results, and the quality of the clarifying question. We provide this analysis in Chapt. 4. Mehrotra et al. [2018] stress that learning from user interactions is a valuable approach to user engagement and satisfaction prediction.

Thus, motivated by work on implicit feedback of aggregated users’ click-through logs for ad hoc retrieval [Kelly and Teevan, 2003], we view the en-

engagement levels as implicit evaluation of clarifying questions with respect to the query and search results. Intuitively, the higher the engagement levels with the clarification system, the higher the quality of the prompted clarification, and higher the need for asking for clarification. Zamani et al. [2020c] study the clarifying question selection with respect to user queries, prompted questions and candidate answers in clarification panes of a search engine. However, the retrieved search engine results for a query have not yet been studied. To bridge this gap, in Chapt. 4, we propose a model to predict the user engagement levels, not only from the information in clarification pane, but from the retrieved search results [Sekulić et al., 2021].

2.5 User Simulation

In this section, we present the challenges that arise in evaluating conversational systems. Through these challenges, we motivate user simulation as a potential solution. We first briefly review history of simulation in IR, followed by relevant literature on simulation in conversational search.

2.5.1 The Challenge of Conversational System Evaluation

Deriu et al. [2021] state that the evaluation method in context of conversational systems should be automated, repeatable, correlated to human judgments, able to differentiate between different conversational systems, and explainable. However, evaluating all of these elements in conversational systems is challenging. While various unsupervised and user-based evaluation methods exist [Deriu et al., 2021], there are key trade-offs. Liu et al. [2016] conduct a thorough empirical analysis of unsupervised metrics for conversational system evaluation and conclude that they correlate very weakly with human judgments, emphasizing that reliable automated metrics would accelerate research in conversational systems.

Conversational search has similar evaluation challenges, further complicated by the retrieval of relevant documents from a large collection [Penha and Hauff, 2020]. While traditional Cranfield paradigm fits well for evaluation of *ad hoc* search systems, it is not easily transferable to conversational search [Lipani et al., 2021, Fu et al., 2022]. One of the specific challenges is that the complexity of multi-turn queries and the overall context is ignored by traditional metrics, and requires a more holistic approach [Hassan et al., 2010, Järvelin et al., 2008]. In

their follow-up study, [Fu et al. \[2022\]](#) assess correlation between user satisfaction, as long as the overall context is taken into account.

The additional challenge arises from the fact that multiple conversational trajectories can lead to satisfying the user's underlying information need. The trajectory complexity further increases under mixed-initiative paradigm, as the interactions between the user and the system become more varied and potentially divergent from each other [[Balog, 2021](#)]. Evaluations based on predefined trajectories, where each conversational turn is associated with a set of query relevance judgments and evaluated under the Cranfield paradigm, thus fail to capture full capabilities of mixed-initiative conversational search systems, and are in turn tailored to a narrow set of systems. [Dalton et al. \[2020\]](#) pursue such evaluation setting in TREC CAsT years 1-3. TREC CAsT y4 [[Owoicho et al., 2022](#)] extends this setting by introducing multiple trajectories spanning from the same initial query, covering different aspects of the topic, thus aiming to mimic different users and their interaction with the system. However, the conversational trajectories still remain predefined.

2.5.2 Automated Metrics

Given the complexity of human-computer interactions and natural language, there has been an ongoing discussion in the NLP community about the credibility of automatic evaluation metrics that are based on text overlap [[Novikova et al., 2017](#)]. These metrics, such as BLEU [[Papineni et al., 2002](#)] and ROUGE [[Lin, 2004](#)], try to judge a system's output solely based on how much lexical overlap it has with a reference utterance. While this allows for a scalable and efficient evaluation, as comparisons of system's output to the reference utterances are inexpensive, various studies have shown poor correlation of such metrics with the human judgments [[Sai et al., 2022](#)].

Other line of research extends beyond lexical overlap and aims to design offline evaluation procedures that better capture the performance of a conversational system and are more correlated with user satisfaction. One example of such metrics in traditional IR are measures based on explicit models of user behavior, such as the Rank-Biased Precision (RBP) [[Moffat and Zobel, 2008](#)] and the Expected Reciprocal Rank (ERR) [[Chapelle et al., 2009](#)]. RBP, for example, assumes that users examine retrieved documents in a descending order, based on their relevance assessed by the search system, and that the examination of each of the next documents is depended on user's persistence. While these metrics extend the traditional IR metrics, such as nDCG, by conditioning the performance on a user model, they are not entirely fit for their use in evaluation of conver-

sational systems. To this end, [Lipani et al. \[2019\]](#) propose a session-based RBP (sRBP), which models not only user’s persistence in examining the results, but the trade-off of issuing an additional query and examining a new document in the list. Session search can be cast to the conversational search setting, thus making this metric potentially applicable for assessing CIA system performance.

More recently, [Lipani et al. \[2021\]](#) propose a metric for offline evaluation of conversational search systems based on user interaction model. Their proposed framework mitigates the necessity of having access to user sessions in advance, which was a limitation of sRBP, making it hardly applicable to evaluating new systems.

2.5.3 User Studies

The aforementioned automated metrics cannot capture the performance of the system accurately [[Belz and Reiter, 2006](#)]. Hence, human annotation should be done to evaluate a system’s performance when a generative model is used, in tasks such as summarization and machine translation. Moreover, evaluation of a system becomes even more complex if an ongoing interaction between the user and system exists. Not only must the system evaluate the generated utterance, it should also be able to incorporate a human response. For this reason, researchers adopt human-in-the-loop techniques to mimic human-computer interactions, and further perform human annotation to evaluate the whole system’s performance (in response to human).

On the other hand, [Deriu et al. \[2021\]](#) identify user studies as a more reliable method for evaluating conversational systems, as opposed to automated evaluation through aforementioned procedures. However, the authors stress the fact that such evaluation is both cost- and time-intensive. The reason for the increased cost of user studies is the necessity for acquiring real users and allowing them to interact with the conversational system. Nonetheless, such evaluation procedure is not easily scalable, as evaluating any changes in the system architecture would include the entire process of gathering the users, conducting the user study, and finally analyzing the results.

To mitigate the unscalability of user studies and the rigidity of automated evaluation procedures, [Balog \[2021\]](#) makes the case that user simulation is an important emerging research frontier for conversational search evaluation. In the next section, we briefly review the history of simulation in IR, followed by research related to user simulation for conversational search.

2.5.4 User Simulation in IR

Simulation in IR has long been studied [Cooper, 1973] with the idea of generating pseudo-docs and pseudo-queries to study literature search system performance. The work was then followed by Griffiths et al. [1978], proposing a general framework of simulation for IR systems. Tague et al. [1980] later studied the problems for user simulation in bibliographic retrieval systems. User simulation for evaluation was first proposed in 1990 by Gordon [1990] where the authors proposed a framework for generating simulated queries. This work has been long followed in the literature to study various hypothetical user and system actions (e.g., issuing 100 queries in a session) that cannot be done in a real system [Azzopardi, 2011]. In particular, Azzopardi [2011] proposed to study the cost and gain of user and system actions and studied the effect of different strategies using simulated queries and actions of users (e.g., clicking on relevant documents). Mostafa et al. [2003] studied different dimensions of users' interests and their impact on user modeling and information filtering. Diaz and Arguello [2009] adapted an offline vertical selection prediction model in the presence of user feedback for user simulation. Moreover, Pääkkönen et al. [2017] assess the validity of the use of simulated users in interactive IR and find it justified under a common interaction model.

2.5.5 Simulation for Conversational Search

Applications such as simulating user satisfaction for the evaluation of task-oriented dialogue systems [Sun et al., 2021] and recommender systems [Zhang and Balog, 2020, Afzali et al., 2023], their utilization in mixed-initiative conversational search is limited.

Sun et al. [2021] proposed a simulated user for evaluating conversational recommender systems based on predefined actions and structured response types. Salle et al. [2021] proposed a parametric user simulator for information-seeking conversation where the simulator takes an information need and responds to the system accordingly. In fact, this work is the closest work to ours. However, we would like to draw attention to various limitations of this work. Even though this work takes an information need as input and aims at answering to the system's request according to that, it fails to generate responses. The approach is limited to predicting the relevance of the system's utterance to the user's information need and selecting an appropriate answer from a list of human-generated answers. In this work, we take one step further and generate human-like answers in natural language. Also, the work by Zhang and Balog [2020] that simulates users

for recommender system evaluation, uses structured data and response types. In this work, we propose a simulator that generates natural language responses based on unstructured data.

Balog [2021] proposes several requirements an ideal user simulator should have: A use simulator should capture:

- Personal interests and preferences (and changes over time);
- Persona (personality, educational and socio-economical background);
- Multi-modality of interactions (speech, text, pointing and clicking);
- User’s ability to change their state of knowledge (learn and forget);
- User’s ability to learn how a system works and what its limits are and change expectations and behaviour accordingly.

Salle et al. [2021] make one of the first attempts at user simulation for mixed-initiative conversational search, by designing a simulator that selects an answer to potential clarifying questions posed by the system. However, their approach is limited to pre-defined clarifying questions and pre-defined answers. Given a clarifying question, the simulator selects the most appropriate answer from a pool of pre-constructed answers. As such, simulator’s usability is restricted to a closed collection of such questions and answers. Section 6.3 addresses that issue, as we design *USi* [Sekulić et al., 2022], a simulator capable of generating answers to clarifying questions posed by the system. Generating the answers eliminates the need for pre-defined sets of questions and answers and enables answering any questions posed by the system. *USi*’s answers are in line with a given information need description and in coherent natural language. Nonetheless, their approach remains limited to single-turn interactions and does not take into account conversational context. Moreover, *USi* only addresses clarifying questions that are direct and about a single facet of the query. In our more recent work [Owoicho et al., 2023], presented in Sect.6.4, we propose *ConvSim*, a simulator capable of multi-turn interactions with mixed-initiative conversational search systems. *ConvSim* addresses the challenges of previous work, while also further extending simulator capabilities by being able to provide positive and negative feedback to system’s responses.

2.5.6 Simulation for TOD Systems

The state of the art in user simulation for TOD systems has evolved significantly in the recent years. Initially, Eckert et al. [1997] proposed the Bigram model, which

estimates a user action conditioned on the system actions. Although efficient, this model does not account for the user goal coherence. Rule-based methods like Agenda-based [Schatzmann et al., 2007, Schatzmann and Young, 2009, Keizer et al., 2010] addresses the coherence issue but relies on the manual definition of rules.

Data-driven approaches, leveraging deep learning models [Gür et al., 2018, Asri et al., 2016, Lin et al., 2021a, 2022, 2023], overcome the rule-based constraints but require significant computational resources and annotated data. These methods mandate dialog annotation for user goal fulfillment at each turn. In-context learning approaches [Terragni et al., 2023] have recently gained traction, designing prompts using snippets of example dialogs, the user’s goal (expressed in natural language as in Terragni et al. [2023], or structured format as in Davidson et al. [2023]), and the dialog history. While these approaches demand fewer resources than fine-tuning methods and eschew manual annotation, they underscore limitations of LLMs, including hallucinations, repetitions, and incomplete user goal fulfillment.

Part I

Mixed-Initiative Conversational Search

Chapter 3

Clarifying Questions in Conversational Search

In this chapter, we address clarification in mixed-initiative conversational search. Specifically, in Sect. 3.1, we first propose a facet-driven method for generating clarifying questions. However, we identify a crucial obstacle for scaling our approach: our method relies on query facets, which are not easily obtainable. Thus, in Sect. 3.2, we investigate the possibility of extracting such facets from a retrieved set of documents. Finally, in Sect. 3.3, we assess the usefulness of posed clarifying questions and the given answers, proposing a classifier-based approach for improved retrieval performance.

Work presented in this chapter has been published in [Sekulić et al. \[2021\]](#), [Sekulić et al. \[2022\]](#), and [Sekulić et al. \[2024b\]](#).

3.1 Facet-driven Clarifying Question Generation

As outlined in Chapter 2, clarification in search has been shown to be beneficial both to the user and to the conversational search system. In this section, we describe our approach to generating clarifying questions that are conditioned on a specific aspect of a given query. We first formally define the problem and describe the data acquisition for training such a model. Then, we propose a training method for fine-tuning GPT-2.¹ Finally, we present and analyze the results on the dataset used for fine-tuning in order to quantify its effectiveness with both automatic metrics and human judgments acquired through crowdsourcing.

¹We note that, while current LLMs exceed capabilities of GPT-2, this work was done in 2021, when GPT-2 version state-of-the-art LLM.

Although significant progress in the area of clarifying user intent has been made, current approaches remain limited to either question selection from a predefined pool of questions [Aliannejadi et al., 2019b, 2020b], template-based question generation [Zamani et al., 2020a], or unsuitable for a fully conversational setting due to the format of the question [Zamani et al., 2020a]. Each of the mentioned works make certain assumptions about the data and problem, such as being able to collect all possible clarifying questions [Aliannejadi et al., 2019b], or being able to fit all the questions in a limited number of templates [Zamani et al., 2020a]. We argue that, while making such strong assumptions is necessary as a starting point towards studying the effect of clarifying questions, it does not represent a real-world scenario. Therefore, an ideal IR system should be able to generate any types of clarifying questions.

A good clarifying question needs to steer the conversation towards a clear formulation of the user’s information need, aiming to facilitate retrieval of relevant documents. Two main challenges arise in generating a good clarifying question: i) to decide if the content of the question is about a specific facet of the user’s query or a somewhat more general aspect; ii) to generate the question in coherent and fluent natural language. The two challenges have been explored separately, but the unified solution is still lacking in the research community. In this section, we propose an approach that addresses both problems.

To tackle the first challenge, we propose to generate a clarifying question based on one or more query facets, thus tackling the problem of the lack of semantic guidance of recent question generation approaches [Rosset et al., 2020]. Given a user’s query, the CIR system should first perform a facet extraction step. Query facets can be extracted from the retrieved set of documents [Kong and Allan, 2013], knowledge graphs [Feddou et al., 2019], query logs [Zamani et al., 2020a], or search engine query autocompletion features [Salle et al., 2021]. We propose a question generation model that produces a question based on the initial query and several keywords representing one of the extracted facets. In order to generate a coherent question, thus tackling the second challenge, we propose to employ the large-scale language model GPT-2, which is capable of generating text of near human quality [Radford et al., 2019].

More specifically, in this section we formally define and propose a query- and facet-conditioned clarifying question generation model. To this end, we fine-tune a large-scale language model, specifically GPT-2, conditioned on the user query and one of its facets. Additionally, we construct a dataset of query-facet-question triplets, to use as ground-truth for training our model. The triplets are extracted from the ClariQ dataset [Aliannejadi et al., 2020b] and are intended to simulate a real-world scenario, where facet terms would be automatically extracted

by any of the aforementioned techniques. Moreover, by synthetically creating facet keywords we eliminate the possibility of propagated error from the facet extraction algorithm, ensuring the consistency in training of our models. Experimenting with different facet extraction methods goes beyond the scope of this paper and is left for future work. Instead, we evaluate the proposed method of generating questions with a number of automatic NLG metrics, as well as human annotators. In a crowdsourcing study, we compare the questions generated by our method with those generated by baseline models in terms of *naturalness* and *usefulness*. The results show that query- and facet-conditioned question generation outperforms the template-based and the query-conditioned GPT-2 in both dimensions.

Thus, our contribution is threefold:

- We present a novel approach to generating clarifying questions. We employ large-scale language models driven by query facets for the task.
- We show the plausibility of our approach, by proposing a semantically-controlled generative model. We fine-tune GPT-2 conditioned on an initial query and one of its facets. We release the code for future work and reproducibility purposes.²
- Alongside the automated evaluation of the proposed model, we perform human evaluation. Furthermore, to complete the study, we discuss the limitations, future work, and potential theoretical framework where our approach could be utilized.

3.1.1 Semantically-Guided Question Generation

We define our task of generating clarifying questions as a sequence generation task. Formally, given a facet f and a query q , the model needs to construct a valid clarifying question cq . Facet f is one of the facets taken from the set of extracted facets, as described in Sect. 3.1.5. Query q is issued by a user. A clarifying question is then defined as a function of the query and its facet:

$$cq' = \text{QuestionGenerator}(q, f) \quad (3.1)$$

To the best of our knowledge, we are the first to condition the clarifying question generation on both the query and the facet.

²Github repository: <https://github.com/isekulic/CQ-generation>.

Language Modeling

Current state-of-the-art methods for text generation are based on large-scale auto-regressive language models [Radford et al., 2019, Dai et al., 2019]. The goal of language modeling is to learn probability distribution $p_\theta(\mathbf{x})$, given example sequences $\mathbf{x} = [x_1, x_2, \dots, x_n]$, where n is a sequence length and θ are parameters of our model. In auto-regressive language generation, we decompose language modeling into next-word prediction, factorizing the distribution $p_\theta(\mathbf{x})$ using the chain rule of probability:

$$p_\theta(\mathbf{x}) = \prod_{i=1}^n p_\theta(x_i | x_{<i}) \quad (3.2)$$

We utilize pre-trained GPT-2, an auto-regressive model trained to learn $p_\theta(\mathbf{x})$ as in Equation 3.2. By doing so, we take advantage of the fact that GPT-2 is trained on large amount of text data, which already makes it powerful for language generation, as shown by its performance on various downstream tasks [Radford et al., 2019]. However, as our goal is not just to generate any text sequences, but to generate questions conditioned with the initial query and one of its facets, we essentially model $p_\theta(\mathbf{x} | q, f)$ where q and f are a query and its facet, respectively. Building on top of recent approaches to semantically-controlled language generation [Peng et al., 2020, Keskar et al., 2019], we model:

$$p_\theta(\mathbf{x} | q, f) = \prod_{i=1}^n p_\theta(x_i | x_{<i}, q, f) \quad (3.3)$$

Learning of parameters θ is done by minimizing the negative log-likelihood of the conditional probabilities in Equation 3.3, that is:

$$\mathcal{L}_\theta(D) = - \sum_{j=1}^{|D|} \sum_{i=1}^n \log p_\theta(x_i^j | x_{<i}^j, q^j, f^j) \quad (3.4)$$

where $D = \{(x^j, q^j, f^j)\}_{j=1}^N$ is the dataset of triplets consisting of clarifying questions x , queries q , and facet terms f . We fine-tune our model on D , in order to be able to: 1. generate questions, not just any textual sequence; 2. generate questions about a given facet of a given query.

Inference

In order to generate clarifying questions, we use a combination of state-of-the-art sampling techniques to generate a textual sequence from the trained model.

Namely, we utilize temperature-controlled stochastic sampling with top- k [Fan et al., 2018] and top- p (nucleus) filtering [Holtzman et al., 2020]. By tuning the temperature, we increase the likelihood of high probability words and decrease the likelihood of low probability words, or vice versa. We do so by directly adjusting the softmax over $p_\theta(\mathbf{x})$, making the probability to predict the i -th token from the vocabulary:

$$p_i = \frac{\exp(x_i/T)}{\sum_j \exp(x_j/T)} \quad (3.5)$$

where x_i are the logits for i -th token in the vocabulary, and T is the temperature. Moreover, we restrict the sampling to only the top- k most likely next tokens, redistributing the probability mass over the remaining k tokens. As some tokens can be samples from a sharp distribution, while others from a flat distribution, top- k sampling shows some shortcomings. Parameter k is fixed and set before sampling, so the possibility of sampling an irrelevant token from a sharp distribution increases. At the same time, setting a low k might restrain model’s token variety. To overcome these potential issues, we experiment with top- p (nucleus) sampling [Holtzman et al., 2020], where we consider the minimum number of next possible tokens whose summed probabilities amount to $p \in [0, 1]$. Again, the probability mass is then redistributed among the remaining tokens and standard sampling is performed from the reduced set of tokens. Our final experiments are performed with temperature T set to 0.7, k to 0, and p to 0.9, as this combination of parameters showed promising results in the early analysis, and is supported by previous research [Holtzman et al., 2020].

3.1.2 Dataset Construction

To the best of our knowledge, a dataset suitable for our purpose of training a sequence generation model conditioned on two different segments, i.e., a query and its facet, is not available in the IR/NLP community. Moreover, as we need to generate *questions*, the specificity of our needs increases. Thus, we adapt a simple data filtering to transform ClariQ data samples to the appropriate (q, f, cq) triplets. ClariQ [Aliannejadi et al., 2020b] consists of queries and corresponding clarifying questions, which are acquired from crowdsourcing. However, since questions are not about a specific facet, we extract the facet keywords using the following procedure:

1. We discard the beginning of the question. Since most of the questions fall under a few templates, with 10 different prefixes we cover around 80% of the dataset.

Table 3.1. Statistics of ClariQ-Fkw dataset used for fine-tuning our generative model. Symbol #N represents “number of”.

	train	dev
Number of samples	1756	425
Average #N facet terms	1.9	1.8
Std of #N facet terms	1.0	0.9
Number of unique queries	187	50
Avg #N questions per query	9.39	8.5
Std #N questions per query	2.7	2.6

Table 3.2. Training triplets from ClariQ-FKw as an input to the GPT-2 model, with clarifying question as a language modeling target.

Initial Query	Facet terms	Clarifying Question
Tell me about cass county missouri	list homes sale	are you interested in a list of homes for sale in cass county
What is von Willebrand Disease?	treatments	are you interested in learning about treatments for von willebrand disease
What is von Willebrand Disease?	types	are you interested in the types of von willebrand disease
Tell me about atypical squamous cells	atypical description	are you interested in a description of atypical squamous cells
Tell me about atypical squamous cells	result test	are you interested in atypical squamous cells in a test result
Tell me about atypical squamous cells	urine	are you interested in atypical squamous cells in urine
Tell me more about Rocky Mountain News	archives	are you interested in news archives
Tell me more about Rocky Mountain News	information park national	are you interested in information about the national park
Find me information about the sales tax in Illinois.	state	are you interested in how the illinois state tax is determined

2. We keep only content-bearing words, specifically verbs, nouns, and noun phrases. The part-of-speech tagging is done with NLTK [Bird et al., 2009]. This ensures that our synthetically created facet keywords resemble the output of any of the most typical facet extraction methods.
3. Finally, from the remaining set of words we remove the ones that also appear in the query. This is done in order to keep our dataset as general as possible, as the facets or subtopics in real-world scenario are unlikely to contain words from the initial query.

This procedure gives us a dataset of more than 2000 triplets, whose examples can be seen in Table 3.2. Table 3.1 shows statistics of the created dataset, ClariQ-FKw, where FKw stands for Facet Keywords. We see that the average number of facet terms is 1.9, with standard deviation of 1.0. Thus, for future work, we suggest that facet extraction method follows similar characteristics. Additionally, we notice a high number of questions for the same query. This further enforces our model to consider both the queries and the facet terms when forming a clarifying question.

3.1.3 Fine-tuning GPT-2

We fine-tune the GPT-2 [Radford et al., 2019] model as our clarifying question generation function *QuestionGenerator* from Equation 3.1. Specifically, we form the input to the GPT-2 model as follows:

$$input_seq = f[SEP]q[bos]cq[eos] \quad (3.6)$$

where $[bos]$, $[eos]$, and $[SEP]$ are special tokens indicating the beginning of sequence, the end of sequence, and a separation token, respectively. Query q , facet terms f , and clarifying question cq are tokenized prior to constructing the full input sequence to the model. Additionally, we further feed the model with segment embeddings, which indicate different segments of the input sequence, namely q , f , and cq . All of the text pre-processing and exact formation of the input sequence is available in our Github repository.³

To calculate the language modeling loss \mathcal{L} , we project the hidden-state on the word embedding matrix to get logits and apply a cross-entropy loss:

$$\mathcal{L} = \sum_{i=1}^N p_i \log(\tilde{p}_i) \quad (3.7)$$

where p is the true distribution and \tilde{p} is the predicted distribution of our model. The loss is applied only on the clarifying question (cq) part of the sequence, while preceding tokens of the facet and the query are masked out.

We fine-tune the models with a batch size of 32, learning rate of 5×10^{-5} for 8 epochs. The hyperparameters were chose based on previous research on text generation and a grid search of the optimal learning rate and number of epoch on the development set. The training takes about one hour on GeForce GTX 1080 Ti GPU. We use the HuggingFace [Wolf et al., 2019] implementation of the GPT-2 model. During inference, we omit the clarifying question cq part from the Equation 3.6 of the input sequence to the model. We generate the question by sampling token by token, as described in Sect. 3.1.1.

3.1.4 Evaluation

We evaluate our question generation model using a number of standard NLG metrics. Additionally, as some NLG metrics received heavy criticism from the research community, we make use of crowdsourcing for gathering human judgments. With the evaluation, we aim to assess the plausibility of query- and facet-conditioned GPT-2 for the task of generating clarifying questions.

³Github repository: <https://github.com/isekulic/CQ-generation>.

Baselines

We compare our query and facet-conditioned model (QF-GPT) to two competitive baselines. These are:

Template-based Question Generation (TB) A template-based approach to generating clarifying questions produces a question by simply filling a slot in a pre-defined question. More specifically, we construct the questions by filling the slot *[facet term]* in a question “Are you interested in *[facet term]*” with one of the query facet terms. This specific question was chosen as it is the most common way of constructing clarifying questions in the Qulac dataset [Aliannejadi et al., 2019b]. The template-based approach has been widely used in various IR tasks [Zamani et al., 2020a, Zhang and Balog, 2020].

Query-conditioned GPT-2 (Q-GPT) As our second baseline model, we fine-tune GPT-2 to generate clarifying questions as described in Sect. 3.1.3, but without feeding facet terms as input to the model. This simulates the behavior of most chatbots, as they solely rely on conversational history, rather than the explicit conversational aspect that should be discussed. This model resembles the approach of Rosset et al. [2020], who employ a GPT-2-based model for conversational question suggestion. They train their model on query-question pairs in a pointwise setting. One limitation of that approach, as pointed out by the authors, is the lack of explicit semantic guidance. Our hypothesis is that this baseline model will generate fluent responses. However, clarifying questions are generated solely based on the query and the memorized generic utterances in the weights of the model, rather than being about specific aspect of the query. Moreover, the generated questions have a risk of not being answerable by our collection.

Automated Metrics

We evaluate our generated questions against reference questions from ClariQ. To this aim, we compute a number of standard metrics for evaluating generated language. The first two are widely adopted metrics: BLEU [Papineni et al., 2002] and ROUGE [Lin, 2004], which are based on n-gram overlap between the generated text and the reference text. Additionally, we compute METEOR [Banerjee and Lavie, 2005], which was reported to have higher correlation with human judgments than BLEU and ROUGE [Banerjee and Lavie, 2005]. METEOR mitigates the shortcomings of BLEU and ROUGE by not just counting the overlap of

Table 3.3. Evaluation of generated questions against gold standard in ClariQ. TB, Q-GPT, and QF-GPT stand for templated-based baseline, query-conditioned GPT-2 baseline, and the proposed query- and facet-conditioned GPT-2, respectively. EAC stands for EmbeddingAverageCosine. Bleu-N indicates BLEU metric calculated on N-grams.

Model	Bleu-1	Bleu-2	Bleu-3	EAC	METEOR	ROUGE-L
TB	0.316	0.169	0.101	0.890	0.212	0.394
Q-GPT	0.316	0.210	0.150	0.862	0.165	0.315
QF-GPT	0.320	0.186	0.119	0.906	0.289	0.285

n-grams, but also considering their stems, WordNet synonyms, and paraphrases. Furthermore, we compute the EmbeddingAverage, defined as cosine similarity between the mean of the word embeddings of each token in the generated and the target questions [Kryscinski et al., 2019].

Human judgments

Recent studies have revealed several flaws of the standard heuristic-based NLG metrics [Sai et al., 2022, Liu et al., 2016, Callison-Burch et al., 2006, Ritter et al., 2010, Mairesse et al., 2010]. The criticism comes from the low correlation of the automated metrics with human judgments, thus making the metrics untrustworthy or even misleading. Moreover, Stent et al. [2005] found that several automatic metrics, including BLEU, correlate negatively with human judgments on fluency of generated text. Thus, in order to properly evaluate our generated clarifying questions, we opt for human annotations. As stated before, a good clarifying question should be in a coherent, fluent natural language and relevant to the topic of the conversation. For that reason, we evaluate two different aspects of our generated questions: *naturalness* and *usefulness*, described in the next Section.

Evaluation is done in a pairwise setting, i.e., an annotator is presented with two questions generated by two different models and has to choose which one is more natural, or more useful, depending on the task. We evaluate the performance of the models in a pairwise setting, as it has been shown to be more reliable and more consistent across annotators than for example the Likert scale [Li et al., 2019]. We compare our main facet-guided clarifying question generation model with the two baseline models described in Sect. 3.1.4. Additionally, we compare the two baselines among themselves.

Naturalness. An important feature of a conversational search system is its natural responses in fluent and coherent natural language [Anand et al., 2020]. Inspired by several studies in various tasks of NLG [Sai et al., 2022, Peng et al., 2020], we define *naturalness* as a question being natural, fluent, and likely generated by a human. Similar definitions exist in a wide range of work, including fluency [Callison-Burch et al., 2006, Stent et al., 2005] and humanness [See et al., 2019]. For example, a clarifying question “Would you like to know more about magnesium-rich foods?” is more natural and fluent than “Are you interested in magnesium foods?”.

Usefulness. Rosset et al. [2020] define a *usefulness* metric to describe conversation-leading clarifying questions. They argue that questions can be relevant to the user’s query, but that does not make them necessarily useful. For example, given a query “Tell me about kiwi fruit.”, a question such as “Would you like to know about kiwi?” is arguably relevant to the query, but it is useless, as it is too broad. Moreover, a clarifying question such as “Are you interested in the business model of NZ kiwi fruit company?” is also related, but also useless due to it being far too specific. This definition of usefulness can be related to adequacy [Callison-Burch et al., 2006, Stent et al., 2005] and informativeness [Peng et al., 2020].

Crowdsourcing

We use the crowdsourcing platform Amazon Mechanical Turk⁴ to acquire annotators, i.e., workers, for our evaluation study. We use in total more than 100 different workers, all based in United States, with minimum approval rate of 94% and minimum number of accepted HITs so far of 1000. We limit the number of annotations per worker to 25 question pairs, in order to eliminate tiredness. Moreover, each question pair is judged by three different workers yielding more than 2000 labels in total, across all of the experiments. We use majority voting to decide on the final label. We compute Fleiss’ kappa κ to assess the degree of agreement per annotated pair. The outcome reaches low, fair, and moderate agreement, depending on the compared models. Workers with suspiciously low performance on the manually curated test pair questions were eliminated from the study.

⁴<https://www.mturk.com>

Results

Automated Metrics. Results of our automatic evaluation are presented in Table 6.3. We notice that it is not clear which model is the best based solely on automated metrics. Specifically, QF-GPT yields the best results among the models in terms of Bleu-1, EAC, and METEOR. However, Q-GPT shows the best performance in terms of Bleu-2 and Bleu-3, while TB outperforms all other models in terms of ROUGE-L. This is a well-known issue in evaluating generated text, as the automated metrics rarely highly correlate with the real scenario [Sai et al., 2022]. For that reason, we rely on human judgments to more accurately estimate the performance of the models. With an automated evaluation, we can only add to the large body of work on criticism of automated metrics for NLG.

Human Evaluation. Comparisons of the baseline models, i.e. Template-based and GPT-2-query, with our proposed facet-guided GPT-2 on *naturalness* and *usefulness* are presented in Tables 3.4 and 3.5, respectively. We conducted a binomial test for each of the model comparison that we’ve made. The *p*-values are presented in Table 3.6.

When assessing *naturalness* of questions generated by our model and the baseline models, we notice several key observations:

- GPT-2-based models (Q-GPT and QF-GPT) produce more natural clarifying questions than the template-based model (TB). This was our initial hypothesis and main motivation behind utilizing GPT-2 for the task.
- GPT-2-based models produce questions of similar *naturalness*, as the difference between query-only model (Q-GPT) and query- and facet-guided GPT-2 model is small (51 to 49). This is also expected, as both methods produce fluent questions.

Moreover, from the study on *usefulness* of generated questions, we observe:

- GPT-2-based models outperform the template-based model. The initial hypothesis was that query-aware GPT-2 (Q-GPT) might not perform so well on *usefulness*, as the questions are not grounded in any specific facet of the query. However, careful examination of the annotated questions suggested that even though the model is not guided by an explicit facet, it is still generating questions that are very often facet-based. The crucial difference to the QF-GPT being that we can not control which facet Q-GPT will ask questions about, as it can only ask about the ones that are implicitly saved in GPT-2’s weights.

Table 3.4. Results on Naturalness between query- and facet-guided GPT-2 (QF-GPT), query-only GPT-2 (Q-GPT), and the template-based (TB) models.

		Wins		
		TB	Q-GPT	QF-GPT
Loses	TB	-	69	59
	Q-GPT	31	-	49
	QF-GPT	41	51	-

Table 3.5. Results on Usefulness between query- and facet-guided GPT-2 (QF-GPT), query-only GPT-2 (Q-GPT), and the template-based (TB) models.

		Wins		
		TB	Q-GPT	QF-GPT
Loses	TB	-	63	60
	Q-GPT	37	-	57
	QF-GPT	40	43	-

- QF-GPT outperforms Q-GPT, which confirms our hypothesis that facet-guided question generation is more useful. According to the Table 3.6, the difference is not statistically significant. However, although Q-GPT is capable of producing clarifying questions related to user’s query, we have no control over the content of the questions. Thus, facet-driven QF-GPT poses itself as a stronger choice for the task.

Human judgments confirm our hypothesis that GPT-2 can generate fluent and natural clarifying questions, while allowing explicit semantic guidance, when trained accordingly. Next, we perform qualitative study of actual questions generated by the proposed model.

Table 3.6. The p -values of binomial statistical test for the significance of the comparisons between different models for naturalness and usefulness.

	Naturalness	Usefulness
TB & Q-GPT	0.01	0.01
TB & QF-GPT	0.08	0.05
Q-GPT & QF-GPT	0.92	0.19

Qualitative Study

Table 3.10 shows several examples of questions generated by our proposed facet-guided model and the baselines. We can observe that all questions generated by GPT-2-based models are indeed fluent and coherent. However, a key difference is that query-conditioned GPT-2 generates questions that are often not entirely relevant to the topic of the conversation. This is known as the hallucination of generative models, where generated responses do not correspond to the real-world [Du et al., 2020, Dziri et al., 2021]. By grounding our question generation model in facets, we gain control of the conversation and eliminate the hallucination effect, thus making our model more useful for clarifying the user need.

Table 3.7. Examples of generated clarifying questions given the initial request and one of the facet terms.

Initial request	Tell me about kiwi		
Facet terms	information fruit	biology bird	people background historical
Template-based	Are you interested in information fruit?	Are you interested in biology bird?	Are you interested in people background historical?
Q-GPT	Are you looking for kiwi clothing?	Are you looking for kiwi reviews?	Are you interested in kiwi fitness?
QF-GPT	Are you interested in kiwi fruit?	Are you interested in kiwi birds?	Are you interested in kiwi history?
Initial request	What is von Willebrand Disease?		
Facet terms	treatments	symptoms	types
Templated-based	Are you interested in treatments?	Are you interested in symptoms?	Are you interested in types?
Q-GPT	Are you looking for a specific web page?	Are you looking for a specific medication?	Do you want to know the causes of this disease?
QF-GPT	Do you want to know what treatments are used to treat the von Willebrand disease?	Are you looking for a list of symptoms of von Willebrand?	Are you looking for a list of the diseases?

3.1.5 Limitations and Future Work

Facet Extraction from Retrieved Documents

In this section, we proposed to generate clarifying questions about certain query facets. However, we have dealt only with facets acquired through controlled keyword extraction, leaving automatic facet extraction as a separate part in a conversational system for future work. We now describe several potential approaches for acquiring query facets and formally define the required properties such methods should have in order to be easily included into our model.

Recent work on clarifying question generation extracted query aspects from the search log of a major search engine [Zamani et al., 2020a]. As such logs are not widely available, we suggest extracting the facets from the set of documents retrieved in response to the initial query. Several methods for facet extraction from a set of documents already exist in the literature [Deveaud et al., 2014, Kong and Allan, 2013] and are largely based on clustering and language modeling

approaches. Formally, given a query q , we retrieve a set of documents $\mathcal{D} = \{D\}$ from the collection C . We then extract a set of N facets $\mathcal{F} = \{F\}$, where N is the hyperparameter representing the total number of facets to extract. Each facet F consists of a set of terms (keywords) representing it: $F = \{t\}$. Facet terms F can then be fed into our generative model in order to produce useful clarifying questions. In the following sections, we explore in depth the feasibility of such approach.

Conversation History

Our approach is currently limited to generating clarifying questions from the initial query only. However, one important aspect of conversational search are multi-turn interactions. Extension of our model to multi-turn conversations includes understanding user's answer to our clarifying question and deciding whether to ask a follow up clarifying question, or otherwise to show the user retrieved relevant documents. In order to ask a follow up clarifying question we can simply generate a question about some other facet of a query. However, for better results, we should also consider user's answer, as they often provide additional information and not just a *yes-or-no* answer [Krasakis et al., 2020]. As GPT-2 has been show to be able to capture multiple turns of information-seeking conversations [Peng et al., 2020, Vakulenko et al., 2021], our first attempt of extending the model would be to feed the conversation history together with the initial query and the extracted query facets to the GPT-2-based model.

Multiple Facets

One of the planned extensions of this work is to generate clarifying questions about multiple query facets, rather than just one at the time. Looking at the first example of Table 3.10 and the query *"Tell me about kiwi"*, our current model would generate a question about one of the facets, such as *"Do you want information about kiwi fruit?"*. However, in order to potentially minimise the number of conversational turns needed to satisfy the user information need, we could ask a question in line with *"Are you interested in kiwi fruit, kiwi bird, or New Zealand people?"*. The challenge here is to create a dataset of such questions suitable for training of generative language models, like GPT-2.

3.2 Exploiting Document-based Features for Clarification

In this section, we build on top of the work described in Sect. 3.1, addressing the challenge of acquiring query facets for clarifying question construction. The work presented in this section is published in [Sekulić et al. \[2022\]](#).

The current research line on search clarification relies either on an available question bank [[Aliannejadi et al., 2019b](#)] or large-scale query logs [[Zamani et al., 2020a](#)]. In their proposed offline evaluation methodology, [Aliannejadi et al. \[2021b\]](#) suggested the use of pre-collected clarifying questions for a limited set of topics. While providing a practical ground for the development and offline evaluation of conversational systems, their assumption of having a set of clarifying questions for all possible topics is not realistic. [Zamani et al. \[2020a\]](#), on the other hand, proposed a generative model that learns to generate template-based questions, mining a large-scale query logs from Bing.com. Mining query reformulations of various sessions provides crucial insights into how and why a query can be incomplete and what questions can be asked for clarification. However, it is limited to the existence of such large-scale logs. Moreover, it is not applicable to long-tail queries, or queries with very diverse set of reformulations.

Exploiting the top-k retrieved results has proved to be an effective way of approaching several IR tasks, such as pseudo-relevance feedback [[Cao et al., 2008](#)], query performance prediction [[He and Ounis, 2006](#)], and query facet extraction [[Kong and Allan, 2013](#)]. Pseudo-relevance feedback studies lie on the basic assumption of taking the top-k retrieved results in response to a user query as relevant. It has been shown that these documents contain useful domain knowledge that helps the system specialize the user’s query and improve the performance [[Cao et al., 2008](#)]. Also, a large body of research on query performance prediction [[He and Ounis, 2006](#), [Zhou and Croft, 2007](#)] estimates the difficulty of a given query based on the top-k retrieved documents and their predicted relevance scores.

In this work, we aim to experiment and analyze the effectiveness of top-k retrieved documents in generating clarifying questions. As such, we conduct a set of extensive experiments where we systematically study the effectiveness of various feature sets from different aspects, as well as facet extraction techniques. First, we extract three sets of features, namely, part-of-speech (POS) tags, knowledge graph entities, and Latent Dirichlet Allocation (LDA) topics. In the next step, given that each document would have various features, we propose three approaches to analyze the extracted features on a ranking-based manner and

extract a few keywords that describe a facet. We call this step facet extraction. We propose the three following techniques for facet extraction: (i) a random selection of features; (ii) selecting the features based on entropy of each extracted features; and (iii) selection based on variance of feature’s tf-idf values across the top-N documents.

Regarding the evaluation, we firstly follow [Salle et al., 2021] in simulating users that provide non-cooperative answers to the systems (i.e., only yes and no with no additional information). By doing so, we are able to test the effectiveness of the predicted facets for each query in terms of document retrieval. Hence, we always assume that a user would respond with a “yes” to the system’s posed question. Therefore, following [Aliannejadi et al., 2019b], we re-rank the documents by combining the language models of the original query and the extracted facet and evaluate the effectiveness of a facet in terms of how much it improves the document ranking performance. Moreover, we construct template-based questions with the selected facets and evaluate their usefulness with a crowdsourcing study. Furthermore, we discuss how the retrieval and human-annotation results provide new insights on extraction and selection of ranking-based features.

Among the various findings, we find that facet extraction is a critical part of the process of generating clarifying questions. Comparing the results of the random, entropy-based, and TF-IDF-based facet extraction methods, we observe a difference between the three methods and a margin in performance, indicating the significance of the facet extraction technique where we find that the entropy-based method achieves the best performance. Based on the human annotation, we see that LDA-based and entity-based features lead to more useful questions, compared to POS-based questions.

Our contributions can be summarized as follows:

- We perform detailed experiments on clarifying question generation with document-based features;
- We propose novel facet extraction techniques with the aim of generating clarifying questions;
- We conduct an extensive automatic and human evaluation of our approach.

Our findings show that significant improvements in document retrieval performance can be achieved when the retrieval is performed with a facet-expanded query. This finding, combined with the usefulness assessments of the facet-based clarifying questions, support the usefulness of using entity-based and LDA-based facets for clarification in conversational search.

3.2.1 Methodology

Feature Extraction from Retrieved Documents

In order to extract useful features from the documents and explore their potential in clarifying question generation, we first retrieve a ranked list of relevant documents for the user’s initial queries. For that purpose, we utilize Chatnoir [Bevendorff et al., 2018], a freely accessible Elasticsearch-based search engine with indexes of ClueWeb and CommonCrawl corpora. The Chatnoir service uses BM25 as its main document retrieval model.

For each query, we extract several content-based features from the retrieved list of documents relevant to that query. Formally, for each document D^i in the ranked list of documents $\mathcal{D} = [D^1, D^2, \dots, D^N]$, where N is the maximum number of top ranked documents to consider, we apply a feature extraction function g , yielding a list of features $\mathcal{F}_g = [F_g^1, F_g^2, \dots, F_g^N]$:

$$F_g^i = g(D^i) \quad (3.8)$$

Extracted features for the i -th document in the ranking are represented as feature terms $F^i = [f_1^i, \dots, f_j^i, \dots, f_{M_i}^i]$, where M_i is the number of extracted features for D^i . Notice that we removed the subscript g from F to simplify the notation.

We experiment with different feature extraction methods, aiming to discover which content-based features best capture the essence of clarification in conversational search. More specifically, for each document we extract: 1) nouns; 2) noun phrases; 3) verbs; 4) named entities; 5) entities in a knowledge base; 6) topics with topic modeling. We utilize *spaCy* [Honnibal et al., 2020] for text processing and identifying features 1 through 4. Moreover, in each document, we link the present entities to the ones in the English Wikipedia knowledge base (feature 5) with Radboud Entity Linker (REL) [van Hulst et al., 2020]. The entities acquired by entity linking are expected to yield more precise and reliable entities than *spaCy*’s named entity recognizer. Finally, we employ LDA [Blei et al., 2003] for topic modeling over the document list. We set the number of topics to extract to 5, as it showed the most promising results in the initial experiments of the study. We base our LDA implementation on Stanford’s Mallet topic modeling toolkit [McCallum, 2002].

Facet Extraction

Given the extracted feature list for each retrieved document w.r.t. a single query, the Facet Extraction module aims to select a feature that could be used to generate a useful clarifying question. Such feature would act as a query facet, and

should ideally be informative and allow us to filter out the retrieved document list, advancing towards satisfying the user’s information need. We find motivation for constructing the Facet Extraction module by analyzing precision and recall of the extracted raw facets, as described in Section 3.2.2. We experiment with three different Facet Extraction methods. Namely: *random*, *entropy-based*, and *tf-idf-based* selection. Each of the methods, given a list of extracted features from the document list \mathcal{F} , selects the most discriminative ones based on different criteria to act as a facet.

More specifically, the *random* selection method simply randomly selects any of the features f_i^j from any of the N retrieved documents.

The *entropy-based* selection method computes the entropy for each of the f_j features by taking into account their term-frequency distribution over the documents:

$$score_E(f_j) = H([TF(f_j)^1, TF(f_j)^2, \dots, TF(f_j)^N]) \quad (3.9)$$

where $TF(f_j)^i$ is the term frequency count of the feature j in the i -th document. By computing the entropy of each of the potential facets f_j , we capture their level of uncertainty and amount of carried information [Shannon, 1948]. This means that features with a balanced probability distribution across the retrieved document list will have a higher $score_E$, making them more likely to be extracted as facets.

The score based on term frequency-inverse document frequency (tf-idf) is computed as the standard deviation of tf-idf values for the feature f_j in each of the documents:

$$score_T(f_j) = \sigma([\frac{TF(f_j)^1}{\log DF(f_j)}, \frac{TF(f_j)^2}{\log DF(f_j)}, \dots, \frac{TF(f_j)^N}{\log DF(f_j)}]) \quad (3.10)$$

where the $DF(f_j)$ represents the document frequency of the feature f_j . The score is based on the proved fact that features with a high tf-idf value in a document are highly discriminative for that document. Thus, features with high variance between tf-idf scores across documents should capture that notion of single-document discriminativeness. Intuitively, we want the facet to be discriminative and help in filtering out the result list to tailor it according to the user’s information need, but at the same time be closely related to the topic.

Finally, the facet is selected by taking the one with the maximum score, for each of the methods separately, that is:

$$facet_{\{E,T\}} = \underset{f_j}{\operatorname{argmax}} score_{\{E,T\}}(f_j) \quad (3.11)$$

with an additional requirement that the feature f_j must appear in at least 3 different documents, to avoid overestimation by the entropy-based and tf-idf-based methods.

Facet-based Question Generation

As reviewed in the related work section, several methods exist for question generation. However, options are scarce for facet-guided question generation. In the previous section, [Sekulić et al. \[2021\]](#) proposed a GPT-2 based method for facet-grounded clarifying question generation. Nevertheless, the questions generated by their model are not always about the specific facet, as they can also focus on the query alone. In order to have more control over the experiments, we resort to widely used template-based question construction [[Zamani et al., 2020a](#), [Zhang and Balog, 2020](#)]. Specifically, we substitute the slot in the question pattern “Are you interested in $\{facet\}$?”, with $facet$ extracted through Equation 3.11.

3.2.2 Evaluation Setting

In this section, we describe the user data and explain the evaluation framework of our three-step methodology described in the previous section. Specifically, we describe the precision- and recall-based evaluation of extracted features by computing overlap with the human-generated clarifications. Next, we describe the evaluation of facet extraction methods through a document retrieval-based experiment. Finally, we detail the crowdsourcing-based evaluation of the clarifying questions generated based on the extracted facets.

Evaluation Dataset

For the purpose of evaluating our approach, we focus on the ClariQ dataset [[Aliannejadi et al., 2020b](#)], an extension of the Qulac dataset [[Aliannejadi et al., 2019b](#)], aimed at fostering research in the field of asking clarifying questions in open-domain conversational search. ClariQ was created on top of the TREC Web Track 2009-12 collection, which contains ambiguous and faceted queries that often require clarification when addressed in a conversational setting. Given a topic from the dataset, clarifying questions were collected via crowdsourcing. Then, given a topic and a specific facet of the topic, crowdsource workers were employed to gather answers to these clarifying questions. ClariQ contains more than 200 topics, each associated with an initial user query and the relevant document list for each of the query facets. We extract the feature set \mathcal{F} for each of

the topics from the ranked list of documents retrieved in response to the initial query for the specific topic.

Evaluation of Document-level Features

In order to estimate the effectiveness of various feature extraction methods, we propose a simple, yet informative evaluation procedure based on human-generated clarifying questions in ClariQ. As the human-generated questions capture a large range of query facets, it is justified to evaluate our feature extraction methods by identifying the proportion of the extracted features present in the ClariQ questions. Specifically, for each query, we compute the overlap of the extracted features from the top N ranked documents with the generated questions. Formally, given the feature set \mathcal{F} , extracted by one of the methods g from \mathcal{D} , we compute the precision and recall with the feature set $\mathcal{F}_{\mathcal{C}, \mathcal{Q}}$ extracted from the set of clarifying questions in the ClariQ dataset, as follows:

$$Precision = \frac{|\mathcal{F} \cap \mathcal{F}_{\mathcal{C}, \mathcal{Q}}|}{|\mathcal{F}|} \quad (3.12)$$

and

$$Recall = \frac{|\mathcal{F} \cap \mathcal{F}_{\mathcal{C}, \mathcal{Q}}|}{|\mathcal{F}_{\mathcal{C}, \mathcal{Q}}|} \quad (3.13)$$

We compute precision and recall for a varying numbers of top documents N , aiming to gain insight into the behavior of different feature extraction methods depending on the number of considered documents. The results of this study are presented in Sect. 3.2.3.

Evaluation of Facet Extraction

In order to evaluate the facet selection methods described in Sect. 3.2.1, we adopt the document retrieval-based evaluation methodology used in the Conv-AI3 shared task [Aliannejadi et al., 2020b]. Specifically, we perform document retrieval with an initial query and evaluate its performance. Then, the initial query is expanded by concatenating the facet terms selected by any of the facet extraction methods, and the retrieval is performed again. We then compare the two retrieval performances, with an assumption that if the selected facet terms are useful for identifying the underlying information need, the retrieval performance will improve with the query expanded with those terms. We analyze the results in terms of traditional IR metrics, namely nDCG@k, precision@1, and MRR. The results of the experiment are presented in Sect. 3.2.3.

Evaluation of the Usefulness of Facet-based Clarifying Questions

The impact of facets selected from various feature extraction methods on the clarifying questions is evaluated in terms of question *usefulness*. Rosset et al. [2020] define a conversation-leading clarifying question as *useful*, arguing that questions can be relevant to the user’s query, but not necessarily useful. For example, given a query “Tell me about hotels in Las Vegas.”, a question such as “Would you like to know about Las Vegas?” is arguably relevant to the query, but not useful, as it is too broad and does not help pin-pointing the underlying user’s information need. Notice that *Usefulness* can be related to adequacy [Callison-Burch et al., 2006, Stent et al., 2005] and informativeness [Peng et al., 2020], and has previously been used to evaluate clarifying questions in conversational search [Sekulić et al., 2021].

We perform a crowdsourcing study to assess the *usefulness* of the generated facet-based clarifying questions. We use Amazon MTurk for acquiring workers, based in the US, with at least 95% task approval rate. The study was done in a pair-wise setting, i.e., each worker was presented with a number of question pairs, where each question in a pair was generated based on different facet extraction methods. Their task was then to provide judgment on which question is more useful, with regard to the context, i.e., the initial query.

We compare clarifying questions based on the three different feature extraction methods in a pairwise setting, namely noun phrases-based, entity-based, and LDA-based features. These features are selected based on their performance in previous experiments in order to reduce the number of pairwise comparisons that grows exponentially with the number of methods to compare. Additionally, we compare the questions based on the facet keywords taken from the ClariQ in a similar manner as described in [Sekulić et al., 2021], resulting in four methods to compare, i.e., six pairwise experiments. For each pairwise comparison, we annotate 100 question pairs in terms of *usefulness*, with each pair judged by two crowdsource workers. The questions in pairs have been randomly shuffled to mitigate position bias. We define a *win* for model A if both annotators voted the question generated based on model A as more useful, and *loss* for model A if both voted the question generated by model B as more useful. In case the two workers voted differently on a single question pair, it was defined as a *tie*.

3.2.3 Results and Discussion

In this section, we aim to answer two main research questions: **RQ1:** Which features yield high-quality facets is useful for clarifying questions?; **RQ2:** Which

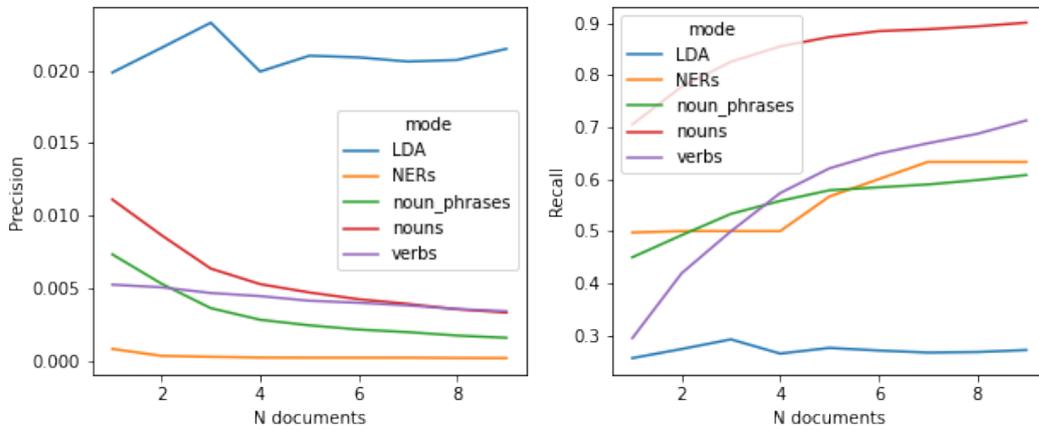


Figure 3.1. Precision and recall measured as overlap of the proposed features and the human-generated clarifying questions from ClariQ.

facet extraction methods result in facets useful for clarification? We discuss RQ1 throughout all of the experiments, while we focus on RQ2 in Section 3.2.3.

Precision and Recall of Extracted Feature Sets

The results in terms of precision and recall of various feature extraction methods, computed as described in Sect. 3.2.2, are presented in Figure 3.1. The performance is shown as a function of the top N documents the top k features were extracted from. Several observations can be made from the experiment. First, as expected, the precision of all of the methods is declining as the number of documents considered increases. Higher number of documents naturally leads to higher number of extracted features, which harms the precision, but improves the recall, as seen in the figure. Also, it suggests that the higher ranked documents lead to more precise set of features, while considering a deeper ranked list provides a broader set of features that improve the recall.

Moreover, we see that the LDA-based extraction method yields higher-precision features overall. However, the high precision comes from the limited number of predicted topics, leading to a low recall. As the number of topics and extracted topic representation terms do not change with the increase of the number of considered documents N , the recall stagnates.

The trend of generally high recall and low precision across all of the methods suggests the need of an additional filtering step in order to identify potentially interesting facets from the large set of features. Obviously, depending on the

facet extraction method, one can study the trade-off between recall and precision depending on the depth of the ranked list that a model can take as input, and its effect on the generated questions. These results strongly motivate the facet extraction module described in Sect. 3.2.1 in our pipeline.

Facet-based Retrieval

The results of facet extraction evaluation through document retrieval are presented in Table 3.8. The first row indicates the performance of the initial query only, while other rows show the performance of queries expanded by various facet extraction methods over the acquired document features. The results reported are for the overall best performing facet for each of the topics. This is to mimic the scenario where the extracted facet would indeed be in line with user’s information need. We observe several patterns in the results. First, queries expanded by facets from facet extraction-based methods outperform query-only baseline in most cases. The exception is *random*-based facet extraction method, which, perhaps unsurprisingly, often even hurts the performance. Second, entity-based features, where entities are linked to a Wikipedia knowledge graph (*Entities* in the table), and LDA-based features seem to perform the best in terms of several metrics presented. We hypothesize this is due to the fact that these type of features are much less noisy, as they are precision oriented, as opposed to recall oriented like noun- or noun phrases-based features. The *tf-idf*- and *entropy*-based facet extraction methods consistently outperform the baselines for those features. However, we note that, after Bonferroni multiple-comparison corrections, neither the *tf-idf*- nor the *entropy*-based method yielded statistically significant improvements over the *random*-based method (with p of 0.05). Similarly, no statistically significant difference was found between features extracted from the text after the Bonferroni correction. This calls for a future study on the additional filtering step aimed towards informed selection of the appropriate facet terms.

Regarding the RQ1, analysis of the results in this experiment suggests that entity- and topic modelling-based features yield the most useful facets. To address the RQ2, both *tf-idf*- and *entropy*-based facet extraction methods show promising results, but they are heavily dependent on the type of input features. Future work will aim to provide a more detailed analysis of relationships between feature and facet extraction methods, as the presented results do not show which method is consistently the best.

Table 3.8. Impact of feature extraction and facet extraction methods on retrieval performance. Symbol † indicate statistically significant difference compared to the query-only baseline. The significance is reported under two-sided t-test with $p < 0.05$.

Features	FacetExtraction	P@1	MRR	nDCG@3	nDCG@10
-	Query-only	0.1800	0.2761	0.1199	0.1412
Nouns	Random	0.2200	0.2957	0.1359	0.1449
	Tf-idf	0.2000	0.2965	0.1500 †	0.1341
	Entropy	0.1867	0.2852	0.1416	0.1260
Verbs	Random	0.2333	0.2987	0.1360	0.1552
	Tf-idf	0.2000	0.2907	0.1384	0.1302
	Entropy	0.2067	0.2920	0.1401†	0.1394
NEs	Random	0.1800	0.2751	0.1362	0.1460
	Tf-idf	0.2067	0.3073	0.1493†	0.1429
	Entropy	0.2467 †	0.3370 †	0.1519 †	0.1591 †
NounPhrases	Random	0.2000	0.2853	0.1153	0.1343
	Tf-idf	0.1733	0.2750	0.1326	0.1194
	Entropy	0.1933	0.2816	0.1419	0.1244
Entities	Random	0.1733	0.2486	0.1135	0.1266
	Tf-idf	0.2302 †	0.3205†	0.1609 †	0.1539
	Entropy	0.2276†	0.3247 †	0.1574†	0.1652 †
LDA	Random	0.1667	0.2666	0.1239	0.1342
	Tf-idf	0.2500 †	0.3131 †	0.1441	0.1543
	Entropy	0.1667	0.2889	0.1657 †	0.1420

Usefulness of Clarifying Questions

Table 3.9. Pairwise comparisons between clarifying questions generated based on different feature extraction methods, as judged by crowdsourcing workers. The * sign indicates statistical significance with $p < 0.05$, calculated by a trinomial test.

Method A	Method B	A Wins	B Wins	Ties	p-value
Entities	HumanF	22%	24%	44%	0.1334
Entities*	NounPhrase	47%	27%	26%	0.0487
LDA*	Entities	46%	14%	40%	0.0076
LDA	HumanF	42%	21%	37%	0.0181
LDA	NounPhrase	30%	26%	44%	0.7844
NounPhrase	HumanF*	17%	52%	31%	$< 10^{-4}$

Table 3.9 shows the results of the crowdsourcing study aimed at estimating the *usefulness* of clarifying questions generated based on selected facets. All of the facets were extracted with an *entropy*-based method described in Sect. 3.2.1, but based on different features. The statistical significance is indicated by * symbol and calculated by the trinomial test, a modification of the sign test that takes into account the ties as well. Paired with manual qualitative analysis of the clarifying questions, we make several observations from the acquired results.

First, LDA-based facets outperform all of the other methods, even the facets based on human-curated clarifying questions *HumanF*. This phenomenon could be explained by the fact that LDA-based features have the lowest level of noise among all of the considered methods. Second, noun phrase-based features are outperformed by all other facet extraction methods. We attribute the poor performance to the high level of noise in the phrases extracted by spaCy from the retrieved documents. Finally, a large number of ties across all of the comparisons suggests both the challenge of the annotation task and the similarity between the extracted facets. However, it is clear that useful questions should be grounded in low-noise-level facets.

Similarly to the previous experiment, we observe the dominance of entity- and topic modeling-based features over POS-based features. Future work on the topic involves the analysis of larger variety of facet-guided clarifying question generation models.

3.3 Usefulness of Clarifying Questions and Corresponding Answers

Previous sections addressed the issue of constructing and generating clarifying questions. However, work aimed at processing and comprehending users' answers to such questions is scarce. Nonetheless, recent research suggests their usefulness by demonstrating improvements in passage retrieval performance after asking a clarifying question and receiving an answer [Aliannejadi et al., 2021b].

To bridge the aforementioned research gap, in this section, we make a first step towards processing the answers given to clarifying questions. We hypothesize that not all information acquired through such interactions with the user would benefit the CS system, i.e., yield improvements in retrieval effectiveness. Thus, the main novelty of our approach is that we do not blindly utilize the questions and the answers, unless they are deemed to be useful. Specifically, we focus on the task of conversational passage retrieval and design a classifier aimed at assessing usefulness of the asked clarifying question and the provided answer. We utilize the question or the answer only if they are deemed useful, by appending them to the conversational history and employing a query rewriting method to attain a more information-dense query. Results on the TREC 2022 Conversational Assistance Track (CASt'22) [Owoicho et al., 2022] demonstrate significant improvements in passage retrieval performance with the use of enhanced query, as opposed to non-mixed-initiative retrieval system (12% and 3% relative improvement in terms of Recall@1000 and nDCG, respectively).

Further, when contrasting our approach to an established method that simply appends the prompted clarifying question and its answer to the original query [Aliannejadi et al., 2021b], we observe differences in performance. Specifically, if neither the question nor the answer are deemed useful, but still used, there is a relative performance decrease of 13% in terms of nDCG@3, compared to non-mixed-initiative baselines. In other words, it is better not to use any information provided by such questions and answers, than to use it wrongly. Thus, our classifier-based approach filters out not useful clarifying questions and answers, reducing the overall noise that can degrade retrieval effectiveness.

Our contributions can be summarized as follows:

- We propose a simple, yet effective, method for processing answers to clarifying questions. The method is based on classifying usefulness of the prompted question and the given answer.
- We identify scenarios where asking clarifying questions resulted in im-

proved passage retrieval, and where it decreased the retrieval performance.

Research presented in this section was published in [Sekulić et al. \[2024b\]](#).

3.3.1 Problem Statement

In this section, we formally define the task of conversational passage retrieval under the mixed-initiative (MI) paradigm and present our methods for each of the components of the task, i.e., query rewriting, clarifying question selection, answer processing, and passage retrieval.

Task Formulation

At a current conversational turn t , given a user utterance u^t and a conversation history $H = [(u^1, s^1), \dots, (u^{t-1}, s^{t-1})]$, the task is to generate a system response s^t . For clarity, we omit the superscript t from the subsequent definitions. In MI conversational search systems, the system’s response s can either be a clarifying question s_{cq} or a ranked list of passages s_D , $D = [d_1, d_2, \dots, d_N]$, where N is the number of passages retrieved and d_i is the i -th passages in the list. Similarly, user utterance u can take form of a query u_q or an answer u_a to system’s question s_{cq} . Modeling other types of user utterances, such as explicit feedback, falls out of the scope of this study. Following prior work on the topic [[Vakulenko et al., 2021](#)], the first task, i.e., query rewriting, is aimed towards resolution of the user query u_q in the context of the conversation history, resulting in $u'_q = \gamma(u_q|H)$, where γ is a query rewriting method.

Following the MI setting introduced at TREC CAsT’22 [[Owoicho et al., 2022](#)], the task is to, at each turn, select the appropriate clarifying question s_{cq} , get the answer u_a to the question, and finally return a ranked list of passages s_D . Therefore, we address the problem of conversational passage retrieval through the following three components: (i) Produce system utterance s_{cq} by selecting an appropriate clarifying question cq from a given pool of questions PQ ; (ii) Process the given answer u_a and incorporate relevant information to the current query, resulting in $u''_q = \theta(u'_q, s_{cq}, u_a)$; (iii) Return a ranked list of passages s_D . Next, we define our approaches to the described components. We note that a clarifying question might be needed only for ambiguous, faceted, or unclear user requests. Thus, for queries not requiring clarification, the system might opt to return a ranked list of passages without asking further questions. However, following the setup enabled by CAsT’22 track, we do not explicitly model clarification need and thus design a system that prompts the user with a clarifying question at each turn.

Clarifying Question Selection

In this subsection, we address the issue of selecting the appropriate clarifying question cq_i from the provided pool of questions PQ as a system’s response s_{cq} . First, however, we rewrite the current user query u_q to take into account the conversational history. Specifically, we utilize T5 fine-tuned on CANARD dataset [Elgohary et al., 2019], available at HuggingFace⁵, as our γ rewriting function, which yields a resolved utterance u'_q . Next, in order to ensure that the selected questions are of high quality, we propose a novel approach that first filters out potentially misleading, unreliable, and faulty questions from the pool (e.g., “What is food like in Nigeria?” resembles queries, while “Would you like to know more about food in Nigeria?” is a clarifying question). To this end, we fine-tune RoBERTa [Liu et al., 2019] (roberta-base⁶) classifier on CAsT’20/21 queries, representing regular questions, and clarifying questions taken from ClariQ [Aliannejadi et al., 2020b]. We then apply the classifier to each clarifying question $cq_i \in PQ$, resulting in the filtered pool PQ_f , containing only questions that are deemed as real clarifying questions by the classifier.

We formulate the task of asking clarifying questions as a ranking task. More specifically, for each query u'_q , we rank the potential candidates cq_i based on their semantic similarity to u'_q . To this end, we use MPNet [Song et al., 2020] from SentenceTransformers [Reimers and Gurevych, 2019], trained for general-purpose semantic matching. We select the clarifying question with the highest score, as predicted by the MPNet: $s_{cq} = \operatorname{argmax}_{cq_i \in PQ_f} \operatorname{MPNet}(u'_q, cq_i)$. Results on the filtering classifier performance and question relevance are reported in Section 3.3.2.

Answer Processing

In this subsection, we describe our novel usefulness-based approach to processing answers given to the asked clarifying questions. To address the issue, we move away from previous approaches that simply append the question and the answer to the original query [Aliannejadi et al., 2019b, 2021b], regardless of the actual information gain. In fact, a recent study by Krasakis et al. [2020] demonstrated that such practice can cause a decrease in retrieval effectiveness. Moreover, they show that multi-word answers are informative (e.g., “yes, I’m looking for info on spiders in Europe”), thus improving retrieval performance. Similarly, short negative answers are not informative (e.g., “no”), while multi-word negative answers

⁵<https://huggingface.co/castorini/t5-base-canard>

⁶<https://huggingface.co/roberta-base>

Table 3.10. Examples of annotated subset of ClariQ, indicating cases when clarifying question, answer, both, or neither are useful.

Query (initial request)	Clarifying Question	Answer	Useful	Prevalence
I'm looking for information on hobby stores.	Do you want to know hours of operation?	No.	<i>None</i>	32%
Tell me information about computer programming.	Are you interested in a coding bootcamp?	No, I want to know what career options programmers have	<i>Answer</i>	53%
Find me map of USA.	Do you want to see a map of US territories?	Yes.	<i>Question</i>	11%
All men are created equal	Would you like to know more about the declaration of independence?	Yes, I'd like to know who wrote it	<i>Question and answer</i>	6%

are (e.g., “no, I’m interested in buying aquarium cleaner”). Thus, we define four possible actions, based on the current resolved utterance u'_q , the clarifying question asked s_{cq} , and the answer u_a :

1. In case the answer is affirmative (e.g., “yes” or “Yes, that is what I’m looking for”), we expand the current utterance by appending the clarifying question asked.
2. In case the answer is deemed useful, i.e., the underlying information need is explained in greater detail, we expand the current utterance by appending the answer.
3. In case the answer is affirmative and it provides additional details, we expand the current utterance with both the clarifying question and the answer.
4. If neither (1), (2), nor (3) is the case, we do not expand the utterance.

Examples of the described cases are presented in Table 3.10 and are all aimed at updating the current utterance by incorporating additional useful information. Formally:

$$u''_q = \begin{cases} u'_q, & \psi(u'_q, s_{cq}, u_a) = 0 \\ \phi(u'_q, s_{cq}), & \psi(u'_q, s_{cq}, u_a) = 1 \\ \phi(u'_q, u_a), & \psi(u'_q, s_{cq}, u_a) = 2 \\ \phi(u'_q, u_a, s_{cq}), & \psi(u'_q, s_{cq}, u_a) = 3 \end{cases} \quad (3.14)$$

where $\psi(u'_q, s_{cq}, u_a)$ is a usefulness classifier, which aims to predict which of the aforementioned actions to take. The labels 0, 1, 2, and 3 correspond to neither s_{cq} or u_a were deemed useful, s_{cq} was deemed useful, u_a was deemed useful, and

both were useful, respectively. Similarly to Section 3.3.1, the function ϕ rewrites the original query given the context, in this case s_{cq} or u_a . We use T5 fine-tuned on CANARD to model the rewriting function ϕ .

Specifically, to model ψ , we fine-tune a large transformer-based model, namely T5 [Raffel et al., 2020], for multi-class classification. To fine-tune the classifier, we manually annotate a portion of ClariQ (150 samples) for the specific aforementioned cases. The annotations were performed by two authors of the paper with an inter-annotator agreement Cohen’s kappa of 0.89. The differences in annotations were discussed and resolved consensually. Examples of annotations are presented in Table 3.10 and classification performance is reported in Section 3.3.2. We dub our novel mixed-initiative classifier-based method *MI-Clf*. Moreover, we assess the prevalence of each of the cases, and find, as presented in Table 3.10, that 68% of interactions contain new, useful information. In the other 32% of the cases, the answer simply negates the prompted clarifying question. While this interaction is as well informative on the user’s information need, current approaches would expand the query by appending the prompted clarifying question and the answer. However, such an expanded query contains terms the user is not interested in, which can potentially degrade retrieval performance. We compare our proposed method to such a baseline, which always extends the query as: $u''_q = \phi(u'_q, scq, u_a)$. This method is dubbed *MI-All*.

Passage Retrieval

Finally, the rewritten utterance u''_q is fed into a standard two stage retrieve and re-rank pipeline [Lajewska and Balog, 2023]. We utilize BM25, fine-tuned on CAsT’20 and CAsT’21 dataset ($k1 = 0.95$, $b = 0.45$), for the initial retrieval step. Additionally, we use pseudo-relevance feedback, where the initial query is extended by RM3 with the highest-weighting terms from top-k scoring passages ($k = 10$ and the number of terms $m = 10$). Next, top 1000 passages are re-ranked with a neural re-ranker. To this end, we use a point-wise monoT5 re-ranker [Nogueira et al., 2020], followed by a pair-wise duoT5 re-ranker [Pradeep et al., 2021] to additionally re-rank top 50 passages. The non-mixed-initiative baseline, dubbed *DuoT5*, uses the same retrieval pipeline. The results of an end-to-end conversational passage retrieval, comparing mixed-initiative methods *MI-Clf* and *MI-All*, and *DuoT5*, are presented in Sect. 3.3.2.

3.3.2 Results

In this section, we report results of our clarifying question selection module and discuss the impact of MI methods on CAsT’22 MI task.

Table 3.11. Clarifying question relevance on TREC CAsT’22.

	nDCG@1	P@1
BM25 [Owoicho et al., 2022]	0.345	0.415
BERT [Owoicho et al., 2022]	0.371	0.454
MPNet	0.492	0.464
MPNet+filter	0.672	0.639

Table 3.12. Performance of the baselines and our mixed-initiative approaches on TREC CAsT’22.

Approach/RunID	R@1000	MAP	MRR	NDCG	NDCG@3	NDCG@5
BM25_T5_automatic	0.3244	0.1498	0.5272	0.2987	0.3619	0.3443
BM25_T5_manual	0.4651	0.2309	0.7155	0.4228	0.5031	0.4831
our_baseline (DuoT5)	0.3846	0.1680	0.4990	0.3392	0.3593	0.3502
+MI-All	0.4441	0.1741	0.5297	0.3594	0.3722	0.3508
MI-Clf	0.4302	0.1776	0.5144	0.3613	0.3697	0.3581

Usefulness Classifier

The proposed usefulness classifier, described in Sect. 3.3.1, achieves an average macro- F_1 score of 0.75 and accuracy of 89% in a stratified 5-fold evaluation on the aforementioned annotated subset of ClariQ. Next, we employ the trained classifier to predict the usefulness of (u'_q, s_{cq}, u_a) at each turn in the CAsT’22 dataset. The question s_{cq} was classified as useful in 28% of turns, while the answer u_a in 37%. In the rest 35% of the cases, neither was predicted to be useful. While the distribution of the predictions is similar to the prevalence in human-annotated data reported in Table 3.10, some differences can be observed. For example, in CAsT’22 28% of the clarifying questions were deemed useful, as opposed to the 13% in ClariQ. This suggests that clarifying questions are appropriate and might be attributed to the strong performance of our clarifying question selection module, discussed in the previous subsection.

Retrieval Performance

Results of the end-to-end conversational passage retrieval task, after the applied mixed-initiative answer processing methods (*MI-All* and *MI-Clf*) are presented in Table 3.12. For reference, we also include the organizer’s baselines in the table. We make several observations from the presented results. First, both methods that utilize mixed-initiative show improvements over the DuoT5 method. This confirms previous findings on the positive impact of clarifications in conversational search. Second, differences between *MI-All* and *MI-Clf* are not statistically significant, across all metrics. However, we note that our classifier-based method utilizes clarifying question or the answer only when deemed useful, which is in about 70% of the cases in CAsT’22. On the contrary, *MI-All* always utilizes both the clarifying question and the answer. The equal performance of the two methods suggests that our usefulness classifier successfully includes only relevant information. Further analysis and discussion of the phenomena is presented in the next section.

Clarifying Question Selection

Our classifier aimed at filtering faulty clarifying questions, described in Section 3.3.1, achieved an accuracy of 97% on a development set comprising 400 questions taken from CAsT’21 and ClariQ, which are not part of the training set. We apply the classifier to the provided candidate question pool PQ and filter out 20% of the questions, resulting in PQ_f . Results on question relevance are reported in Table 3.11. We observe clear improvements of MPNet-based clarifying question selection models, compared to competitive baselines, namely BM25 and BERT. Moreover, the proposed filtering step, aimed at filtering out faulty clarifying questions, combined with MPNet, significantly outperforms all other approaches. We envision the use of such filtering classifiers in similar cases, where selection is performed from a pool of questions, which could potentially be any kind of questions (i.e., queries, clarifying questions, personal questions).

3.3.3 Analysis

In this section, we analyze scenarios where asking a clarifying question resulted in improved passage retrieval performance. Additionally, we identify cases where retrieval performance decreased.

Table 3.13 shows relative changes in performance of mixed-initiative methods (*MI-All* and *MI-Clf*) compared to the *DuoT5* baseline. The changes are in-

Table 3.13. Performance gains and losses of MI-All and MI-Clf methods on CAsT’22 grouped by turns as classified by our usefulness classifier. The gains and losses are computed as relative changes over the non-mixed-initiative baseline DuoT5.

Method	Useful	Recall@1000	nDCG@3
MI-All	<i>None</i>	0.309 (+3.3%)	0.233 (-13.5%)
	<i>Answer</i>	0.509 (+19.8%)	0.389 (+4.7%)
	<i>Question</i>	0.522 (+12.5%)	0.443 (+3.7%)
MI-Clf	<i>None</i>	0.305 (+1.9%)	0.264 (+0.1%)
	<i>Answer</i>	0.533 (+23.4%)	0.412 (+10.0%)
	<i>Question</i>	0.499 (+8.5%)	0.423 (+0.1%)

indicated on CAsT’22 turns, grouped by the predictions of our usefulness classifier. We focus on Recall@1000 to indicate the benefits to the initial retrieval stage, and nDCG@3 to showcase the impact on the re-ranking stage. Several observations can be made from the results. First, in cases where the usefulness classifier predicted that neither the clarifying question s_{cq} nor the answer u_a is useful, we observe a drop of the *MI-All* method’s retrieval performance, in terms of nDCG@3 (−13%). Recall, however, is not impacted by incorporating potentially not useful information and even shows a slight increase (+3.3%). As this method always appends both s_{cq} and u_a to the query u'_q , the performance drop is expected, especially in the re-ranking stage, as the re-ranker might be confused by the additional non-relevant information. The case when both s_{cq} and u_a were predicted as useful by our classifier has been dropped from this analysis, due to the lack of such predictions on CAsT’22 data.

Moreover, for both MI methods, we observe higher performance gains when the answer is useful, compared to cases when the question is useful. This could be explained by the fact that user answers are deemed useful when they are longer and thus provide more detail on the underlying information need [Krasakis et al., 2020]. On the contrary, a clarifying question can be deemed useful even when tangibly addressing user’s need. In other words, a good clarifying question can make a small step towards elucidating user’s information need. However, user’s answer can contain detailed expression of their information need, thus making further gains.

3.4 Conclusions

In this chapter, we first presented a facet-guided model for generating clarifying questions in mixed-initiative conversational search. We showed that large-scale language models, in particular GPT-2, are quite fit for the task, when fine-tuned properly. More specifically, we semantically guide GPT-2 question generation by conditioning the question on the user’s original query and one of the query facets. Human judgments acquired through crowdsourcing show that clarifying questions generated by our proposed model are both natural and useful, compared to competitive baselines. Our results and discussions serve as a preliminary step towards generating clarifying questions from the query facets. Our goal was to demonstrate the capability of large language models for generating clarifying questions, by showing that a model such as GPT-2 can be guided and driven towards a certain topic or goal in a conversation. Our results demonstrated the superiority of generated questions over template-based questions.

Moreover, in Sect. 3.2, we have explored various methods for feature extraction from the list of documents relevant to user’s query and their impact on clarifying question generation. We have identified the importance of the facet extraction module, as the experiments have showed low precision of the initial feature extraction methods and can not be considered useful facets without an appropriate filtering. Thus, we experimented with two facet selection methods, namely, entropy- and tf-idf-based. The results suggest improvements in document retrieval performance, when the retrieval is performed with the facet-expanded query. In particular, our proposed method achieved 38% relative improvement in terms of nDCG@3. This finding, combined with the *usefulness* assessments of the facet-based clarifying questions, indicate the adequacy of entity- and LDA-based facets for clarification in conversational search. However, relatively modest improvements call for an additional facet filtering step, in order to help and pin-point the exact terms representing user’s underlying information need.

Overall, we observe that document-based feature extraction, together with ranking-based facet extraction can lead to significant improvements when used for generating clarifying questions. Compared to the existing work on question selection [Aliannejadi et al., 2021b] and generation based on query logs [Zamani et al., 2020a], we conclude that the proposed method for facet extraction and question generation is effective. Our results can be used to inform question generation models with a final goal of an improved end-to-end document retrieval.

Additionally, in Sect. 3.3, we proposed a classifier-based method, *MI-Clf*, for processing answers to clarifying questions in conversational search. The proposed method extends the original query only when either is deemed useful. The

results on TREC CAsT'22 demonstrate clear improvements of the *MI-Clf* method over non-mixed-initiative baselines (+12% and +3% relative improvement in terms of Recall@1000 and nDCG). Moreover, we showed a drop in performance of established methods that always use both the clarifying question and the answer, in cases where neither are useful (−13% in terms of nDCG@3), thus incorporating noisy information. This study makes the first steps towards improved answer processing methods; we believe there are further gains to be made, e.g., by designing models that can learn how to best utilize user's answers.

In general, a drop in retrieval performance when clarifying question is asked and processed can be due to several reasons:

- No need for clarification: any kind of clarifying question is unnecessary, as user's intention is clear;
- Faulty clarifying question: the question asked does not inquire about relevant aspects of the query;
- Faulty answer utilization: the method for utilizing the given answer is sub-optimal.

Some of the reasons above can be analyzed with the offline datasets used in this study, while some can't (e.g., clarification need) and would require in-depth annotation or extensive user studies.

Chapter 4

User Engagement Prediction for Clarification in Search

Chapter 2 provided a motivation for clarification in search. In this chapter, we address the challenge of user engagement with clarification features in search engines. However, clarification in search proved to be a cumbersome task [Zamani et al., 2020c], posing higher risk of user dissatisfaction. The challenge arises from two main aspects: deciding whether or not it is necessary to ask for clarification, and selecting or generating the appropriate clarifying question. Clarification selection can in fact be formalized as a user engagement prediction problem. User engagement refers to the quality of user experience characterized by, among others, attributes of positive affect, attention, interactivity, and perceived user control [O'Brien and Toms, 2008]. Persistent users' interactions with the clarification mechanism are an indication of a well-designed system. Furthermore, through these interactions users provide implicit feedback about the *necessity* and the *quality* of prompted clarifications.

Recently, modern search engines include various types of clarification components into their systems. An example of such a component in Bing, namely a clarification pane, can be seen on Figure 4.1. Given a user query, a number of Microsoft's internal algorithms propose a clarifying question and offer clickable answers that would filter the retrieved results according to the user's need. The research on the quality of asked clarifying questions and potential answers is still in its early stages [Zamani et al., 2020a]; however, Zamani et al. [2020b] argued that engagement level could be an indicator of the clarification system quality. User engagement prediction has been studied in various domains of IR [O'Brien et al., 2020]. However, studying and modeling user engagement for web search clarification is relatively unstudied.

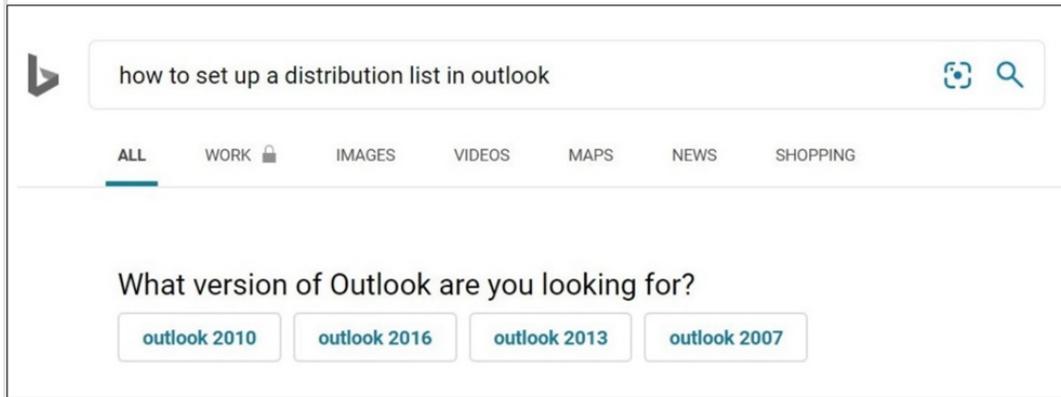


Figure 4.1. An example of Bing clarification pane taken from [Zamani et al., 2020b].

In this chapter, we focus on the task of predicting user engagement level (ELP) on clarification panes. Given an initial query, search results, and clarification pane, ELP aims to estimate how engaged the user would be with the clarification pane. Previous work [Zamani et al., 2020c] studies how engagement levels correlate with the query attributes such as query type and aspects. However, the relationship between SERPs and engagement has not yet been explored. We stress the importance of utilizing retrieved results, as they can contain cues as to how faceted or ambiguous the query is, suggesting how necessary the clarification is in the first place.

Moreover, users' engagement with the system implicitly discloses information about the *necessity* and the *quality* of the asked clarification. The *quality* aspect can be modeled under the assumption that the higher the engagement levels, the better the question and the provided answers are. We make this assumption inspired by a large body of work in the IR community on implicit feedback from aggregated click-through rates for document retrieval [Xue et al., 2004]. Also, we study clarification *necessity* prediction through ELP. Our clarification necessity prediction model takes as input the initial query and the retrieved results list and predicts the level of user engagement with a clarification pane. Although certain attributes of the initial query such as length and ambiguity could indicate the necessity of asking clarifying questions, we show that incorporating other SERP elements such as result titles and snippets play important roles in improved prediction accuracy.

We formulate the task as supervised regression and propose a deep learning-based model for the prediction of the engagement levels. We compare the performance of the model to various central tendency measures and a number of

traditional machine learning algorithms, as well as popular neural models. Our model, based on a Transformer architecture, jointly encodes the user query, the clarification pane, and the SERP elements, outperforming competitive baselines.

We evaluate the performance of our model on a large-scale dataset of search clarification engagements called MIMICS¹ [Zamani et al., 2020b], collected from millions of interaction records of Bing² users. Our extensive experiments establish a strong baseline for the task, while ablation studies and analysis of the model’s inner mechanisms provide guidelines for future research. Our main contributions can be summarized as follows:

- We formally introduce the clarification pane ELP task as supervised regression and propose a transformer-based model to tackle it. We make the code publicly available for reproducibility purposes.³
- We perform ablation studies with respect to the model input data. We find that utilizing retrieved search results greatly benefits the model’s performance.
- We perform detailed analysis of the performance of our model w.r.t. various characteristics of the SERP

To the best of our knowledge, our work is the first to utilize SERP elements for clarification pane engagement prediction. More precisely, we find that utilizing search results in certain ways is highly beneficial for the ELP task, as the performance of our model increases by up to 40% when provided with retrieved results, compared to the query and the clarification pane only.

Large parts of this chapter have been published in Sekulić et al. [2021].

4.1 Engagement Level Prediction

In this section, we first describe the dataset used for engagement level prediction (ELP). Then, we formally introduce the task of ELP and propose a BERT-based model to tackle it.

4.1.1 Data

MIMICS [Zamani et al., 2020b] is a recently proposed large-scale collection of datasets for research on search clarification. It enables the IR community to study

¹<https://github.com/microsoft/MIMICS>

²<http://www.bing.com>

³<https://github.com/isekulic/mimics-EL-benchmark>

various aspects of search clarification, ranging from clarification generation and selection, over re-ranking of candidate answers, to user engagement prediction and click models for clarification. MIMICS consists of three datasets:

MIMICS-Click , including over 400k unique queries, their corresponding clarification panes, and the aggregated user interaction signals.

MIMICS-ClickExplore , consisting of over 60k unique queries, each with multiple clarification panes, and the aggregated interaction signals.

MIMICS-Manual , containing 2k query-clarification pairs, manually labeled for the quality of clarifying questions, candidate answer sets, and landing result pages of each answer.

In this work, we mainly focus on MIMICS-Click, as the largest, most generic one. Each sample in MIMICS-Click consists of the initial query q , the clarification question c , and answers offered as options by the system $A = [a_1, \dots, a_5]$. The sample is associated with user interaction signals as labels. The *impression level* i , a categorical variable where $i \in \{low, medium, high\}$, represents the frequency of the clarification pane being presented to the user for the corresponding query. The *engagement level* $e \in [0, 10]$ shows the level of total engagement received by the users in terms of click-through rate. Each answer is also associated with its conditional click probability.

The authors also released search engine results pages (SERPs) for each query, as retrieved by Bing. In addition to the query meta-data, SERPs contain up to 10 retrieved instances with a title, an URL, and a short snippet of a web document. We denote retrieved results as $R = [r_1, r_2, \dots, r_n]$, where $n \in [0, 10]$. Each of the results r_i consists of a tuple $r_i = (t_i, s_i)$, where t_i and s_i are title and snippet of the i -th result. Table 4.1 shows the average lengths of queries⁴, questions, retrieved titles and snippets, as well as the number of retrieved results in SERPs. We utilize all of the available text and information as input to our models to compose our experiments, as described in Section 4.1.3.

4.1.2 Task Formulation

We formulate the task of user engagement level prediction as a supervised regression. The goal of the regression is to predict the value of the target variable y , given a D -dimensional vector \mathbf{x} of input variables [Bishop, 2006]. Given the

⁴The length was computed by splitting the text on whitespaces.

Table 4.1. Dataset statistics for MIMICS-Click.

	Mean	Std	Median	min-max
Query length	2.66	1.18	2	1 - 12
Question length	6.05	0.47	6	5 - 14
SERP Titles length	7.65	2.71	8	0 - 30
SERP Snippets length	43.47	14.76	45	0 - 149
Answers per query	2.81	1.06	2	2 - 5
Responses per query	9.07	1.19	9	0 - 10

dataset of N observation pairs (\mathbf{x}_n, y_n) , where $n = 1, \dots, N$, the goal is to find a function $f(\mathbf{x})$ whose outputs \hat{y} for new inputs \mathbf{x} produce the predictions for the corresponding values of y . The loss function of the predicted values \hat{y} and the actual values y are model-dependent and described in Sect. 4.1.3.

The target variable y is given in the dataset in the range of 0 to 10, corresponding to the level of user engagement with the clarification pane. We approach ELP as a regression problem as it poses itself as a natural formulation of our task. Compared to classification, false predictions of different value are penalized differently. For example, classification would punish false predictions of $\hat{y} = 7$ and $\hat{y} = 1$ for a sample with $y = 8$ the same, while in reality, the predicted label of 7 is much closer to the actual engagement level. Therefore, even though still wrong, one would prefer a system to predict 7 instead of 1. Moreover, the task of user engagement prediction has been evaluated as regression in various applications such as [Sano et al., 2016, Dhall et al., 2018].

4.1.3 Our approach

We now define our model called **ELBERT** (Engagement Level prediction by **ALBERT**). As mentioned in the previous section, the goal is to predict the engagement level y based on the initial query q , clarification question c , list of candidate answers A , and retrieved results R . We predict the engagement level EL as follows:

$$EL(q, c, A, R) = \psi(\phi_q(q), \phi_c(c), \phi_A(A), \phi_R(R)) \quad (4.1)$$

where $\phi_{\{q,c,A,R\}}$ are high-dimensional representations of q , c , A , and R . The aggregation function ψ outputs the final engagement levels based on the input representations. All of these components can be modelled with numerous methods. In this work, we utilise ALBERT as our encoder for generating $\phi_{\{q,c,A,R\}}$ representations in a joint fashion. More specifically, as ALBERT has been shown to

consistently help downstream tasks with multiple inputs [Lan et al., 2020], we essentially learn the joint representation of query, clarification question, answers, and results as:

$$\Phi(q, c, A, R) = ALBERT(q, c, A, R) \quad (4.2)$$

reducing our Equation 4.1 to:

$$EL(q, c, A, R) = \psi(\Phi(q, c, A, R)). \quad (4.3)$$

Input to the ALBERT component is composed of tokenized query, question, answers, and results, separated by the separation token [SEP], with classification token [CLS] inserted in the beginning of a sequence. Answers a_i are aggregated before feeding them to the model. Similarly, we aggregate SERP information R , with a difference that we experiment with both, titles t_i and snippets s_i as inputs. In either case, texts of titles or of snippets are joined by whitespace prior to being fed to the model. We note that in ablation studies some of the components are left out by simply removing them from Equation 4.2. We use a pretrained ALBERT-base [Lan et al., 2020] as a text encoder and truncate the total input sequence length to a maximum of 512 tokens. Our model has 11M training parameters, making it considerably smaller than other Transformer-based model (e.g., BERT has 110M).

The regression component ψ , that outputs the engagement level, is constructed as follows: last layer hidden-state of the first token of the encoded sequence ([CLS] token) is further processed by a linear layer and a non-linear activation function. We then add another linear layer, with dropout and a non-linear activation function in between, to produce the final 1-dimensional output that corresponds to EL . The model is trained using mean squared error as a loss function for 4 epochs, with a learning rate of 5×10^{-5} , Adam optimizer [Kingma and Ba, 2014] and linear weight decay with warmup.

4.2 Experiments

In this section, we introduce our experimental setup and present main results for the engagement level prediction. Furthermore, we analyse the effect of SERP elements on model’s performance and perform detailed analysis w.r.t. various characteristics of the data.

4.2.1 Baselines

We use central tendency measures as our first baselines for predicting the engagement level. More specifically, we have three different static baselines:

1. *mean* of the data (MeanEngagement);
2. *median* of the data (MedianEngagement);
3. *sampling* from a normal distribution $\mathcal{N}(\mu, \sigma^2)$, where μ and σ are the mean and the standard deviation of the engagement levels in the training data, respectively (NormalEngagement).

To tackle the task of ELP, we experiment with a number of models from traditional machine learning and deep learning. Namely:

Linear Regression. First baseline is a linear regression model, fitted using ordinary least squares approach.

SVR. We employ support vector regression machines [Drucker et al., 1997], a version of support vector machines [Cortes and Vapnik, 1995] for regression. We experiment with the linear, as well as the radial basis function (RBF) kernel.

Random Forests. An ensemble meta-algorithm that uses bootstrap aggregating (bagging) technique to improve the stability of decision trees [Breiman, 2001].

LSTM. Long-short term memory [Hochreiter and Schmidhuber, 1997] are a well-established method for sequence modeling, especially on text data. We experiment with multi-layer bidirectional networks.

The input to traditional ML models are tf-idf weighted bag-of-word features extracted from the input text. LSTM is fed with pretrained GloVe word embeddings [Pennington et al., 2014] of tokenized input text. We use Scikit-learn [Pedregosa et al., 2011], HuggingFace [HuggingFace, 2023], and Pytorch [Paszke et al., 2019] for the implementation of the aforementioned models.

Evaluation Metrics

We evaluate the effectiveness of our models using standard evaluation metrics for the task of supervised regression. The first two are Mean Absolute Error (MAE) and Mean Squared Error (MSE). We also evaluate our regression models with

Coefficient of Determination or R^2 . It is a statistical measurement that examines the proportion of the variance in one variable that is predictable from the second variable, estimating the “goodness of a fit”. It is defined as: $R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2}$, where N is the number of samples, y_i is the actual value in the dataset for the i -th sample, \hat{y} is the predicted value, and \bar{y} is the mean of the actual values.

Experimental Setup

We evaluate our models using a hold-out method, i.e., reserving 20% of the dataset for the test set. We train, and tune traditional ML models in a cross-validation manner [Cawley and Talbot, 2010]. We use 5-fold split of the training set into training and development set, which is used for grid-searching of the best parameters. The extensive grids of parameters include regularization parameter C , the choice of *kernel*, *gamma*, and *epsilon* for SVR, number of estimators and depth of random forest regressor, as well as feature selection process. All of the parameters can be found on our GitHub repository.

For tuning the hyper-parameters of our neural models, we split the training set into training and development sets. Notice that models are retrained on the full training set with the best parameters before being evaluated on the hold-out test set.

We evaluate the models on the full MIMICS-Click dataset, consisting of more than 400k query-clarification-SERP tuples, and on the subset of that dataset, in which only samples with the engagement level larger than zero are selected. The models in this setting were fed all the available data, i.e., the queries, clarification panes, and the SERPs, while the ablation studies in Sect. 4.2.2 go into the analysis of input data.

4.2.2 Results & Discussion

Performance comparison

Here, we compare the performance of our ELBERT model against the baselines on the complete dataset, as well as the subset of data with $EL > 0$. Table 4.2 lists the results in terms of all our evaluation metrics. We can notice that heuristic baselines (i.e., MeanEngagement, MedianEngagement and NormalEngagement) are consistently outperformed by both, the traditional ML models, and the neural models. However, one exception is MedianEngagement, a baseline that always outputs the median of the training set, i.e., EL of 0.0, when evaluated on the full MIMICS-Click by mean absolute error. Since more than 80% of the dataset have

Table 4.2. Performance on the full MIMICS-Click dataset (400k+ samples) and a subset where engagement levels are higher than zero (71k samples). Bold values denote the best results for each metric. Symbols † and ‡ mark statistically significant improvement over central tendency measures and traditional ML models, respectively ($p < 0.01$).

Model	Full MIMICS-Click			EL-only MIMICS-Click		
	MAE	MSE	R^2	MAE	MSE	R^2
Mean	0.1531	0.0546	0.0	0.2426	0.0790	0.0
Median	0.0921 †	0.0531	0.0	0.2412	0.0805	0.0
Normal	0.1896	0.0823	0.0	0.4316	0.2637	0.0
Linear Regression	0.1463	0.0530	0.0359	0.2364	0.0783	0.0083
SVR	0.1462	0.0522	0.0490	0.2318†	0.0736†	0.0676†
RandomForest	0.1477	0.0526	0.0423	0.2301†	0.0729†	0.0775†
BiLSTM	0.1452‡	0.0511‡	0.0606‡	0.2299†	0.0720†	0.0789†
ELBERT	0.1439‡	0.0505 ‡	0.0762 ‡	0.2224 ‡	0.0692 ‡	0.1124 ‡

EL of 0.0, and MAE does not penalize large errors as hard as MSE or R^2 , this is expected. The tide turns swiftly when evaluating on the subset of the data with EL larger than 0.0, where all of the static baselines, including MedianEngagement, are outperformed by all of our models.

Moreover, we see a clear discrepancy in the performance of traditional ML models and neural networks. This is consistent with recent research in various tasks in IR and NLP fields. Moreover, we see that ELBERT significantly outperforms BiLSTM model. Through its powerful encoder, ELBERT is able to capture deeper semantic relations, as it is pretrained on a large body of text. This is also consistent with recent research on deep learning-based models for natural language understanding.

Effect of SERP elements on ELP

In this experiment, we aim to analyze the effect of clarification panes and every SERP element on the performance of our model. Our hypothesis is that each SERP element (e.g., result titles and snippets) provides a complementary set of features that aids the model towards more effective prediction. Therefore, we train our ELBERT model with different combination of SERP elements and clarification panes, and compare the performance of the different models. We report the results in Table 4.3. We see that the relative improvement when utilizing

Table 4.3. Impact of SERP elements available on the model performance. Bold values denote the best performance of each metric. Statistically significant results (with $p < 0.05$) over query setting and query+pane setting are marked with † and ‡, respectively.

#	Setting	Full MIMICS-Click			EL-only MIMICS-Click		
		MAE	MSE	R^2	MAE	MSE	R^2
1	query	0.1500	0.0519	0.0485	0.2275	0.0719	0.0776
2	query+pane	0.1354†	0.0512	0.0626†	0.2257†	0.0714	0.0839†
3	query+titles	0.1335 †‡	0.0436 †‡	0.0814 †‡	0.2229†‡	0.0692 †‡	0.1124 †‡
4	query+snippets	0.1459†	0.0513	0.0606†	0.2255†‡	0.0706†‡	0.0944†‡
5	query+pane+titles	0.1450†	0.0505†	0.0745†‡	0.2224 †‡	0.0692 †‡	0.1124 †‡
6	query+pane+snippets	0.1439†	0.0505†	0.0762†‡	0.2240†‡	0.0704†‡	0.0969†‡

titles from SERPs is up to 35% compared to using query and clarification pane, and more than 45% over query-only setting. The results strongly suggest the advantage of making use of SERP elements for ELP.

An interesting finding is that even though snippets contain more text than titles and thus arguably more information as well, the model does not consistently perform better with snippets as input. In fact, even though results with titles seem better than ones with snippets, we observe no statistically significant difference between the performance of *query+titles* and *query+snippets* on full MIMICS-Click, nor EL-only MIMICS-Click. There are several reasons why snippets do not exceed the performance of titles. First, it might be the quality and type of text shown in snippets. Snippets often show only short excerpts, or even multiple excerpts which are not clearly divided, from a longer document, focusing on query words in the retrieved document. Thus, they might not contain all the semantics of the document, while titles usually do. Second, it might be the maximum input length of our encoder, which is 512 sub-word tokens. As mentioned in Table 6.1, a median length of a title is 8 tokens, while median snippet length is 45. Considering that most of the samples have 9 or more title-snippet pairs in their SERPs, it is evident that some portion of concatenated snippets get left out. The potential limitation of truncating input length in most of BERT-based models is a research direction on its own.

We point out that the necessity of asking the clarification can be estimated from the initial query and retrieved search results, i.e., rows 1, 3, and 4 in Table 4.3. The success of the model to predict EL based on SERPs and the query alone, suggests that this framework can be used for determining whether or not to ask

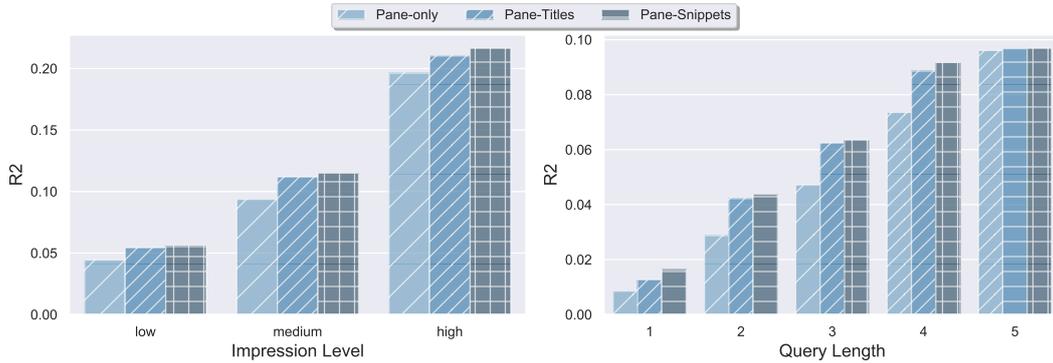


Figure 4.2. Performance by impression levels (left) and query lengths (right) with different input configurations.

a clarifying question. However, we leave this aspect for future work. Instead, in the next subsection we evaluate our model trained on ELP task for clarification pane selection, addressing the pane quality aspect.

4.2.3 Additional Experiments

Here we show ELBERT performance, as measured by R^2 , with respect to various characteristics of the dataset and the input components.

Impression level

Figure 4.2 (left) shows the performance of our model w.r.t. impression levels. We notice that our model performs significantly better on queries with high impression rate, i.e., those whose clarification panes have been shown to users more frequently. The differences between models at each impression level are not statistically significant, while differences between levels are, with $p < 0.01$. As the engagement level labels have been computed by aggregating user click information, this suggests that query-clarification pairs that have been implicitly evaluated by a small number of users, i.e., have low impression level, contain noise.

Query length

Figure 4.2 (right) presents the performance of our model w.r.t. query length. The difference in performance between all query lengths is statistically significant. We notice that longer queries generally lead to better performance. This can be attributed to them being more descriptive, thus allowing the search engine to

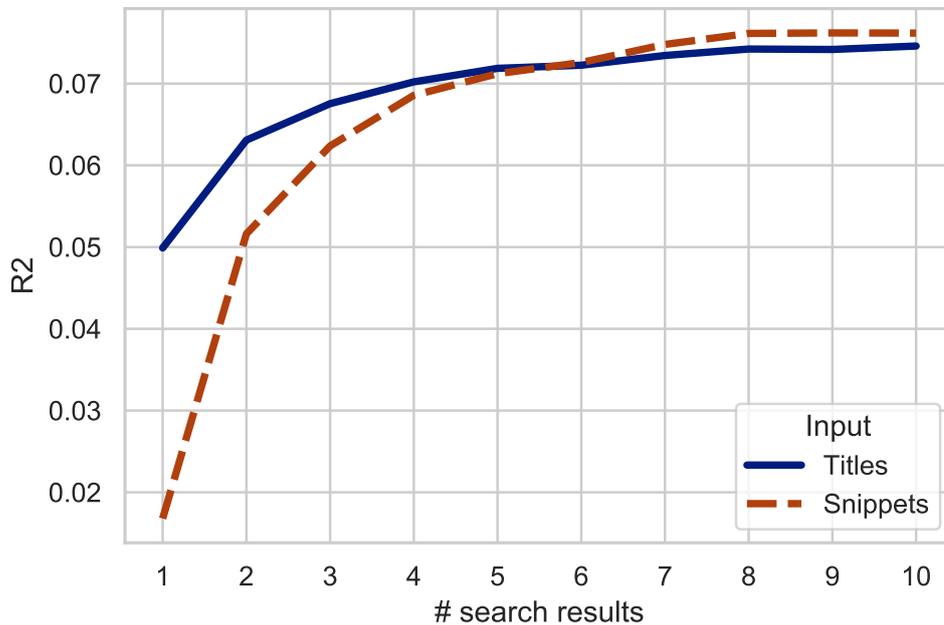


Figure 4.3. Performance by number of search results made available to the model.

retrieve more relevant results. Consequently, our model would utilize SERPs of higher quality, improving the ELP. Highest improvement is seen for a query and pane-only setting. Since the model in that setting does not see any SERP content, it benefits the most out of longer, more descriptive queries.

Number of search results

Since user behavior is mainly biased by the results they see, and they mostly look at top results only, we perform experiments to see how our models behave in a setting with limited number of retrieved results. As mentioned before, MIMICS dataset contains up to 10 retrieved results for each query. We evaluate our model with 1, 2, . . . 10 SERP elements made available to it. Results for both, titles setting and snippets setting, are presented in Figure 4.3. We see a clear improvement in the performance as the number of search results fed to the model rises. This suggests that our model highly utilizes SERP elements for ELP. We notice a saturation after 7 elements, especially in the setting with snippets. This might be due to snippets exceeding the maximum length of input to transformer-based

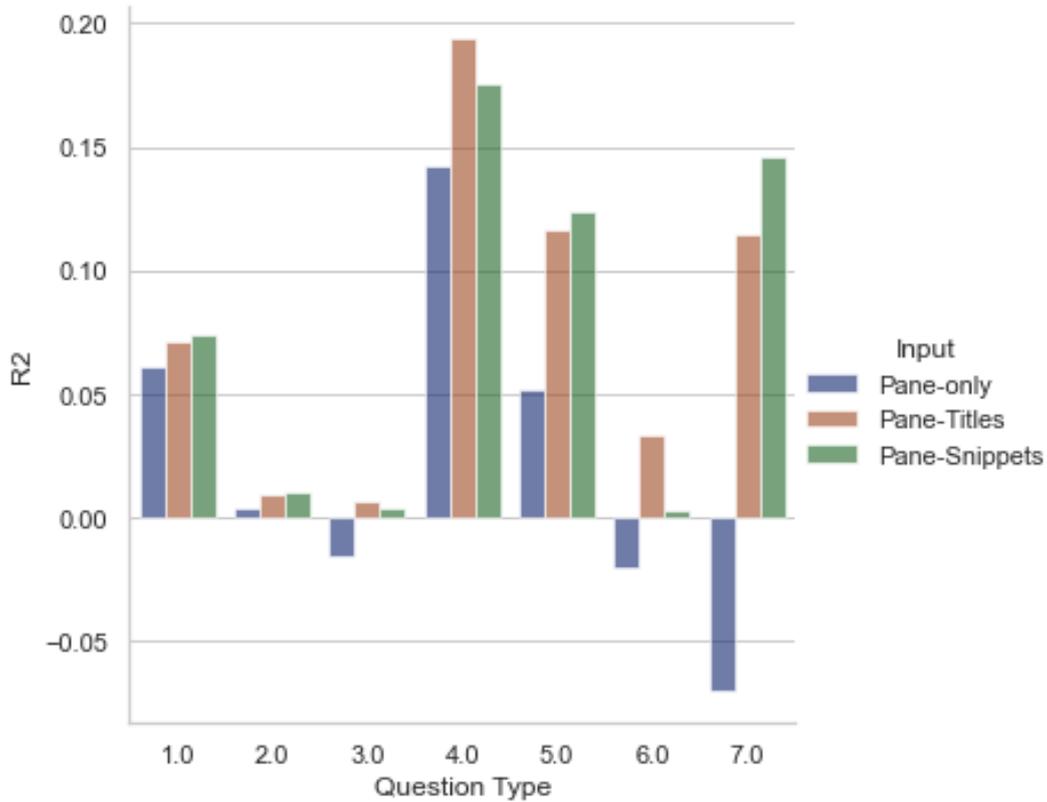


Figure 4.4. Performance by question type.

models, which is 512 subword tokens.

Question type.

Microsoft’s internal algorithms usually output one of the 7 possible clarifying questions in response to user query [Zamani et al., 2020b]. We investigate the performance of our model w.r.t. the clarifying question type. The results in Figure 4.4 suggest that there are differences between them, although as such were not specifically pointed out by Zamani et al. [2020b].

Clarification pane selection

In this subsection, we evaluate our ELBERT model trained for predicting the engagement levels on clarification pane selection task. MIMICS-ClickExplore dataset contains 64k unique queries, each with multiple clarification panes, yielding a total of 168,921 query-clarification pairs. Each pair is associated with cor-

Table 4.4. Performance on clarification pane re-ranking task on MIMICS-ClickExplore. † indicates statistical significance with $p < 0.01$.

	nDCG@1	nDCG@2	nDCG@3	nDCG@5
<i>Worst question</i>	0.0	0.486	0.544	0.586
<i>Random</i>	0.437	0.728	0.759	0.771
ELBERT-panes	0.448†	0.728	0.761†	0.770
ELBERT-titles	0.496†	0.755†	0.785†	0.794†

responding engagement levels, which are generated by distinct groups of users (similar to A/B testing). Therefore, the engagement levels can be directly used as implicit feedback on the quality of clarification.

For each query, we rank their corresponding panes by their engagement level in descending order and consider that as true ranking, similar to [Zamani et al., 2020c]. As our predicted ranking, the model predicts the engagement level for each clarification pane of a given query and we rank the panes by predicted values, again in descending order. The produced ranking is then compared to the true ranking in the dataset by nDCG@K. Results in Table 4.4 show that our model outperforms enlisted baselines, namely *Worst question* and *Random*. However, relatively small improvements over baselines suggest that utilizing a model trained for ELP on clarification pane selection task might not be ideal. Zamani et al [Zamani et al., 2020c] report better results on the task when model is trained directly for question re-ranking, in a pair-wise setting, on manually labeled data.

Transformer attention analysis

As described in 4.1.3, we feed the model with the query, clarifying question, candidate answers, and SERP elements. To inspect the importance of each of the input components, we perform quantitative study of ALBERT’s attention mechanism on the input sequence. Higher attention weights correspond to higher importance for making the final prediction. We average the attention weights of all tokens of a specific component for the last 3 layers of ALBERT encoder, as they have been shown to capture most of the semantic meaning [Jawahar et al., 2019].

Figure 4.5 shows attention of the classification [CLS] token, query tokens, clarifying question tokens, and candidate answers, over the whole input sequence, including the 10 retrieved responses (SERPs). The attention weights are aver-

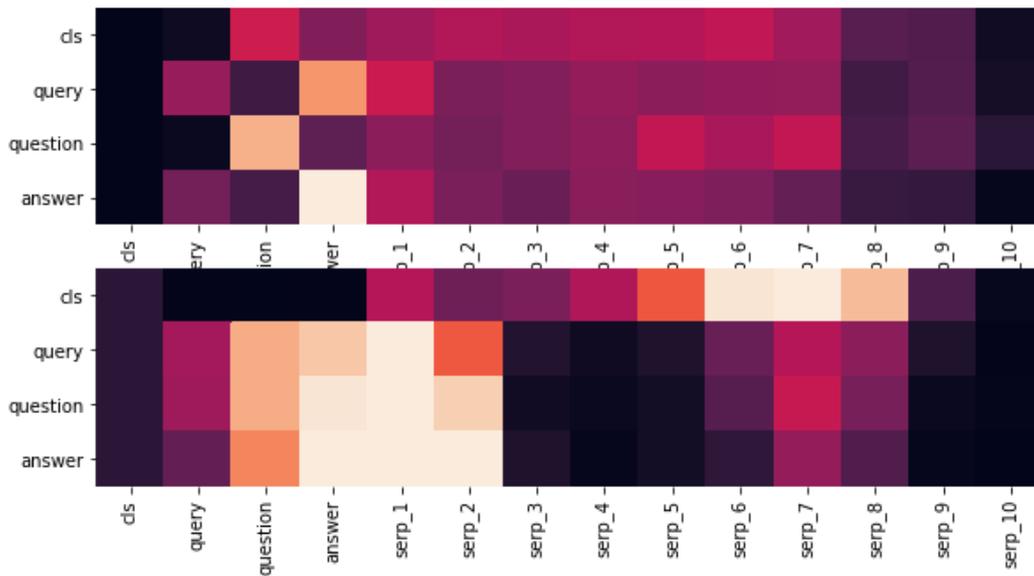


Figure 4.5. Attentions of input elements over queries, clarification panes and SERP elements.

aged over 50 samples in the development set. We notice that SERP elements play an important role for each of the input components, as attention weight distribution is relatively evenly distributed among them. Furthermore, query tokens, question tokens, and answer tokens, attend a lot to themselves.

4.3 Conclusions

In this chapter, we described various experiments on engagement level prediction task for clarification in search. We showed that semantic-rich models, like ALBERT, are much more successful in the task than traditional ML models. Furthermore, we demonstrated the benefit of utilizing information from search engine result pages, such as titles and text snippets of retrieved documents, in the ELP task. Modeling of engagement levels can help guide the system on when and which clarifications to prompt, thus improving the overall user experience. Future work involves deeper analysis of topical changes in the retrieved pages, that could lead to more accurate prediction of engagement levels, and estimating the necessity of asking for clarification.

Chapter 5

Entity-Based Response Rewriting

Search systems are frequently being used for acquiring new knowledge [Marchionini, 2006, Gadiraju et al., 2018], and enabling effective interaction with them has been the focus of a significant body of research [Hearst, 2009, White, 2016]. With the advent of conversational agents, the landscape of search is changing [Zamani et al., 2022], with rapid progress being made in question understanding [Yu et al., 2020, Vakulenko et al., 2021] and result retrieval [Dalton et al., 2020, Yu et al., 2021]. However, little attention has been paid to supporting users according to their knowledge level [Ghafourian, 2022] and ensuring that they can actually understand the answers returned by the system. While traditional web search offers users the possibility to follow hyperlinks or consult knowledge panels in search engine results pages (SERPs) in order to learn about certain concepts they might be unfamiliar with [Eickhoff et al., 2014], such opportunity is taken away in conversational information access (CIA) due to the limited bandwidth interface. For example, while the system’s generated response might be concise and indeed answer the given question, it might mention concepts that the user is unfamiliar with. We argue that CIA systems offer an unique opportunity to proactively assist an individual, and, with this work, we aim to make a step in this direction.

Entities are natural units for organizing information and can improve the user experience throughout the search process [Balog, 2018]. In this chapter, we investigate how to make answers more accessible to users in a text-based conversational setting. The main hypothesis underlying this work is that allowing users to learn more about certain entities mentioned in the answer would lead to an improved user experience. However, not all entities are equally important. Therefore, we utilize the notion of *entity salience* to capture how central a given entity is to understanding the answer returned by the system in response

to a question. Entity salience has been studied in the context of web search, where [Gamon et al. \[2013\]](#) define it as entities being central and prominent, capturing the aboutness of the Web page. While only about 5% entities are salient in Web pages [[Gamon et al., 2013](#)], answers in a conversational setting are short with only a few entities present, therefore yielding a higher ratio of salient entities. However, not knowing those entities might seriously impair the user’s understanding of the answer. Once the top salient entities are identified, we propose two answer rewriting strategies aimed at helping users to understand the system’s response. One approach rewrites the answer to expand it with inline definitions of salient entities, making the answer self-contained. The other approach complements the answer with a follow-up question, offering users the possibility to learn more about specific entities. See [Figure 5.1](#) for an illustration.

The first research question we ask is **(RQ1) What are the characteristics of salient entities in CIA?** To address this question, we conduct an analysis of 360 answers from well-established conversational Q&A datasets using crowdsourcing. Specifically, we extract a number of entities from the answers and ask crowd workers to assess their saliency based on how essential they are to properly understand the answer to the given question. We find that the majority of the answers contain a number of highly salient entities, providing strong motivation for answer rewriting. At the same time, our results also suggest that saliency is highly subjective and is likely influenced by the user’s background knowledge. Additionally, we identify categories of salient entities that do not require further definitions as they belong to common sense knowledge or are already explained in the answer.

The second research question we address is **(RQ2) How to utilize salient entities in answer rewriting for an improved user experience?** We consider two variants of answer expansion by (1) adding definitions from a knowledge base after the entity mention in parentheses, and (2) inserting human-written descriptions in the text in a natural manner. Similarly, we study two options for follow-up generation: (1) asking the user directly whether they want definitions of salient entities, and (2) offering an optional follow-up to learn more about specific entities. An experimental comparison of these four alternatives using crowdsourcing reveals that users generally prefer some type of answer rewrite over the original answer, with inline definitions being generally favored over answers with follow-up questions. As part of our experimental protocol, we also ask crowd workers to provide a free-text justification for their choice of answer rewrite preference. We observe high subjectivity in these responses, with some annotators favoring the original answer for its conciseness, some preferring the one with inline definitions for its comprehensiveness, and others appreciating the

Query:

Why was keto diet originally developed?

Answer:

His trial on a few **epilepsy** patients in 1921 was the first use of the ketogenic diet as a **treatment** for epilepsy.

Answer rewrite with inline definitions:

His trial on a few **epilepsy** (human neurological disease causing seizures) patients in 1921 was the first use of the ketogenic diet as a **treatment** (attempted medical remediation of a health problem) for epilepsy.

Answer rewrite with a follow-up question:

His trial on a few epilepsy patients in 1921 was the first use of the ketogenic diet as a treatment for epilepsy. Would you like to learn more about **epilepsy** or **treatment**?

Figure 5.1. Possible strategies to making sure the user understands the answer in conversational information-seeking.

conversational nature of answers with follow-up questions. Overall, our results provide a strong motivation for future research on personalizing answer rewriting, considering both the background knowledge and interaction preferences of users.

Additionally, we explore the potential of using large language models (LLMs) for the entity-based answer rewriting task, given the recent success of LLMs in a wide array of natural language processing and information retrieval tasks [Brown et al., 2020, Ouyang et al., 2022, Pereira et al., 2023, Gao et al., 2022]. Specifically, we experiment with various way of prompting ChatGPT for end-to-end answer rewriting. Our initial analysis revealed significant shortcomings in terms of knowledge distortion (e.g., rewritten answer contains simpler language, without the original entities the user might want to know about), failure to explain entities, or significantly increasing the answer length, making it unfit for a conversational setting. Taken together, these issues give rise to concerns regarding the lack of control and faithfulness of the rewritten answers, underscoring the need for more controlled answer rewriting strategies that we are proposing.

Our contributions can be summarized as:

- We annotate a sample of 360 question-answer pairs to characterize entity saliency in CIA.
- We propose and evaluate two methods for improving the answers given by the search system: rewriting the answer with inline definitions of salient entities and prompting the user with a follow-up question to allowing them to learn more about salient entities.
- We extensively analyze the feedback on answer rewrite type preference and identify patterns that can help motivate future research.
- We perform an initial exploration of addressing the same task using a state-of-the-art LLM and provide anecdotal evidence for the need for more controlled generation approaches, thereby solidifying the case for the type of methods this chapter is proposing.

This chapter was published in [Sekulić et al. \[2024b\]](#).

5.1 Understanding Salient Entities in Conversational Information Seeking

In this section, we define salient entities in CIA and present several research questions. Then, we describe the dataset acquisition process with crowdsourcing. Finally, we showcase relevant aspects of the created dataset and provide an analysis on special cases of salience.

5.1.1 Problem statement

A salient entity captures the aboutness of the text and is thus central to the given document [[Paranjpe, 2009](#)]. In CIA, answers to user’s questions are usually short, containing from a single to a few sentences with only a few entities present. Identifying salient entities in such answers is thus imperative, as they are essential for the user’s understanding of the given answer. In this work, we inspect the prevalence of entity salience in CIA. We define entity saliency on a graded scale of 0 to 2, i.e., $s(e_i) \in [0, 2]$, with e_i being the i th entity in an answer. A score of 0 corresponds to the entity not being salient at all and 2 to the entity being highly salient.

In this section, we aim to shed light on **RQ1: What are characteristics of salient entities in CIA?** We break this generic question into a series of more specific subquestions:

RQ1.a How prevalent are salient entities in answers in CIA?

RQ1.b How well do users agree on which entities are salient?

RQ1.c Is there empirical evidence that the notion of entity salience is different in conversational answers than in documents?

RQ1.d Are there entities that are salient, but do not require explicit definitions?

5.1.2 Dataset Acquisition

In order to model entity salience in CIA, we extend QReCC [Anantha et al., 2021], an open-domain conversational question answering dataset containing 14k conversations. QReCC is curated from three well-established datasets: TREC CAsT 2019 [Dalton et al., 2020], Google Natural Questions (NQ) [Kwiatkowski et al., 2019], and QuAC [Choi et al., 2018]. TREC CAsT focuses on conversational passage retrieval, while QuAC resolves around conversational Q&A over a Wikipedia text. Contrary, NQ is not conversational in its original form, but has been extended by using its queries as a basis for creating subsequent turns. Excerpts from QReCC with their original sources and saliency annotations are shown in Table 5.1. All of the conversations in the three datasets have been normalized so that they contain multi-turn interactions with manually resolved utterances and manually checked responses. This, together with its diversity, makes QReCC appropriate for our work on entity salience in CIA. In this work, we provide a deep analysis of the dataset in terms of entity salience modeling and thus subsample the original QReCC dataset. We restrict ourselves to the test portion of QReCC, as it contains utterances from all of the three aforementioned datasets. Additionally, in order to annotate as many conversations as possible within reasonable cost, we restrict ourselves to the conversations up to the depth of 3, thereby trading off conversation depth for higher breadth coverage.

We employ a state-of-the-art entity linker, WAT [Piccinno and Ferragina, 2014], to extract entities from the system’s responses. As suggested by the authors, we use a reasonable, slightly precision-oriented confidence threshold of 0.45 for extracting entities from texts. We filter out the entities that appear in the question as well, assuming the user asking the question already knows about them. This procedure results in an entity set E , containing several entities extracted from the given answer A , that do not appear in the question Q .

Table 5.1. Excerpts from QReCC with our crowdsourcing-based annotations of entity salience scores.

Source	Question	Answer	Entity salience score
CAsT	What does it cost to become a physician’s assistant?	Average cost of resident tuition for a 27-month physician assistant program is...	Residency: 1.8 Tuition payments: 1.4
NQ	Why is snow used for igloos?	Snow is used for igloos because the air pockets trapped in it make it an insulator.	Thermal insulation: 1.5
QuAC	What was Sigmund Freud and Wilhelm Fliess’ relationship?	(they)...saw themselves as isolated from the prevailing clinical and theoretical mainstream because of their ambitions to develop radical new theories of sexuality.	Human sexuality: 1.0 Theory: 0.7 Mainstream: 0.4

Now that we have (question, answer, entity_set) triplets, we employ crowdsourcing to annotate which entities from the entity set can be considered salient. Given the question and the answer, the task is to annotate the degree to which a given entity is considered essential for understanding the answer. After an initial analysis of the entities and their importance in understanding the answer, we opted for a graded relevance scale. We adopt an annotation scenario where an entity can be either essential, important, or not important. We draw the similarities of our annotation scheme with well-established graded relevance schemes in IR [Sakai, 2021], where a document can fully satisfy a user’s information need, partially, or be irrelevant. We define the following labels for an entity:

Essential. Knowing about the entity is essential for understanding the answer to the question. It is not possible to comprehend the answer without knowing about (being familiar with) the entity. This label corresponds to a salience score of 2.

Important. Knowing about the entity is important for a deeper and more complete understanding of the answer. However, it is not essential and the user can partially comprehend the answer without knowing about the entity. This label corresponds to a salience score of 1.

Not important. The entity is not important for understanding the answer to the question, nor does its knowledge benefit the user’s knowledge on the topic. This label corresponds to a salience score of 0.

Table 5.2. Examples of special cases of essential entities that do not necessarily require further definitions. Last column indicates the prevalence of such entity types in the expertly-annotated subsampled set of 122 entities across 37 QA pairs.

Special case	Question	Answer	Entity	% in subset
Common sense	Who is Sigmund Freud’s friend Wilhelm Fliess?	During this formative period of his work...his friend Wilhelm Fliess, a Berlin-based ear, nose, and throat specialist.	Human nose	25%
Location/NE	Did Hansie Cronje make any debuts?	Hansie Cronje made his first-class debut for Orange Free State...at Johannesburg.	Johannesburg	12%
Already defined	What are some advantages of using Linux?	One of the main advantages of Linux is that it is an open source operating system, i.e., its source code is easily available for everyone...	open source	2%
Entity is the answer	Who has to push the rock up the hill?	Sisyphus, king of Ephyra, was punished to roll an immense boulder up a hill...	Sisyphus	4%

We use Amazon Mechanical Turk¹ as our annotation platform. All workers are required to have at least 1,000 approved annotations with a minimum 95% overall approval rate and be based in the United States, in order to mitigate the potential language barrier for understanding the task. Each (question, answer, entity) triplet is annotated by five different workers. To insure high quality annotations, we manually curate a test set of (question, answer, entity) triplets that the workers need to annotate correctly in order for their annotations to count towards the final dataset. The size of the test set is 25% of the final dataset size. Additionally, we track workers’ mouse clicks and discard annotations that are done recklessly and quickly. Workers take on average 8.1 ± 12.7 seconds per (question, answer, entity) triplet. To ensure ethical use of crowd workforce, we provide an appropriate compensation of 0.20\$ for 5 annotated entities, resulting in an average of 18\$/h, which is over 250% of the minimum wage in the USA.

5.1.3 Analysis

In this section, we answer our research questions through an extensive analysis of the acquired dataset on entity salience in CIA.

Presence of Salient Entities.

In order to focus on answers with a certain level of complexity, we selected answers with at least 2 entities present (as extracted by WAT), and annotated 120 QA pairs in, containing a of 400 entities. Each (question, answer, entity) pair was assessed by five different workers, resulting in a total of over 2,000 annotations. In the annotated dataset, there are on average 5.06 ± 2.63 entities present in the

¹<https://www.mturk.com>

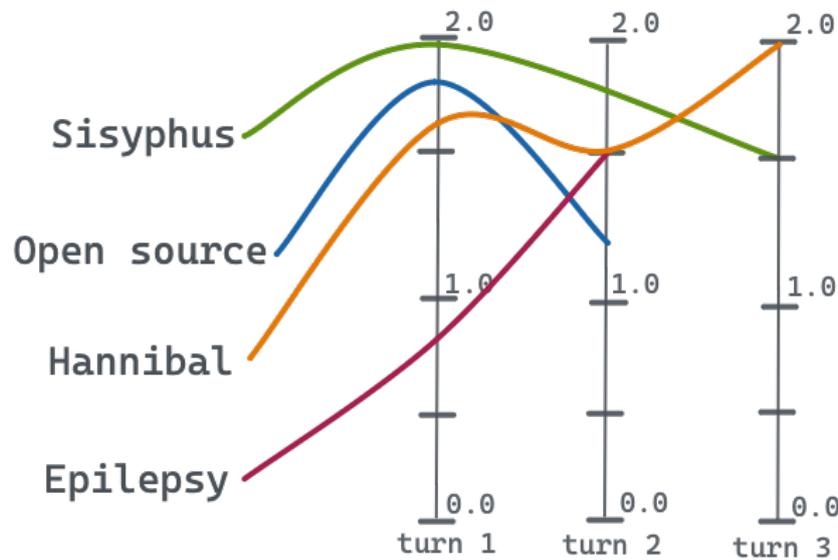


Figure 5.2. Four examples of changes in salience score through conversational turns.

answers. The average salience of those entities, as assessed by crowd workers, is 1.24 ± 0.33 (40% annotated with salience of 2, 53% with 1, and 7% with 0). In response to **RQ1.a**, this means that there are more salient entities than non-salient ones in CIA answers. This finding is further confirmed by averaging the saliency scores for each entity and computing the portion of salient ones (e.g., average saliency score > 1.5) over the total number of entities in the answer. This ratio is 0.63 ± 0.28 , meaning that on average 63% of all entities in CIA can be considered salient entities.

Moreover, we analyze salience throughout the conversation. Figure 5.2 shows examples of the development of an entity salience through three turns of the conversation. The entity *epilepsy* is mentioned sporadically in the answer at turn 1, but becomes considerably more salient in the subsequent turn. Overall, we observe an average change of saliency score between two consecutive turns of 0.36 ± 0.21 , suggesting that an entity might become more or less essential as the focus of the conversation changes. Entities might be sporadically mentioned in earlier turns of the conversation, but with users' further queries they can become central to the topic of the conversation.

Subjectivity in Assessing Entity Saliency

To answer **RQ1.b**, we compute Fleiss' κ [Fleiss, 1971] to measure subjectivity of the annotators assessing the degree of saliency of an entity, i.e., how essential is the entity for a complete understanding the answer. The computed κ is 0.16, suggesting weak inter-annotator agreement and high subjectivity for the task [Viera et al., 2005]. Furthermore, we compute Spearman's rank correlation coefficient ρ between all pairs of workers that annotated a specific QA pair. With this step, we try to assess potential subjectivity level that is due to different perception of scale of essential/important/unimportant entities. For example, two workers might agree on which of the entities is more salient, while their perception of the saliency scale differs slightly. The average Spearman's ρ is 0.45, which suggests a fair agreement and thus a certain level of skewed score subjectivity, which is different from weak agreement measured by κ . Overall, we conclude that the task of assessing which entities are essential for answer understanding is highly subjective. The subjectiveness may come from different user background knowledge, their perception of saliency, but also from personalities. However, having labels collected from five different annotators allows for a robust assessment of entity saliency. The data suggests that there is a lot of potential for dealing with personal preferences and subjectivity when estimating entity saliency.

Entity Saliency in Documents vs. in CIA

We hypothesized that the notion of entity saliency is different in CIA than in Web documents. To assess this hypothesis in the light of **RQ1.c**, we compute the entity saliency score using a state-of-the-art model for saliency prediction in documents, SWAT [Ponza et al., 2019]. For each QA pair, we compute Spearman's ρ over the entities ranked by saliency score from the dataset and the entities ranked by saliency score as predicted by SWAT. The computed ρ averages to 0.25, indicating low to moderate correlation. This suggests that document-level saliency prediction methods are not entirely fit for the task of entity saliency identification in CIA. Moreover, the prevalence of salient entities is significantly higher in CIA answers (63%), as opposed to Web documents (5%), as reported by Gamon et al. [2013].

Special Cases of Salient Entities

Another important finding of the analysis is the case that although most of the answers contain salient entities, which require user's familiarity to comprehend the answer, not all such entities necessarily require definitions. To answer **RQ1.d**,

we take a random subsample of more than a hundred entities from the crowd-annotated answers for analysis with a goal of finding potential patterns. We then perform expert annotation (done by one of the authors of the study) by carefully inspecting entities in the context of a conversation and note whether they would potentially require explicit definitions or not. In our analysis, several special cases of entities arose, which might not require further steps to be taken by the CIA system, even if deemed salient.

Table 5.2 presents the described cases, with an example and their prevalence, as indicated by the percentage of such entities subsampled set. We estimate that around 40% of the entities belong to one of the special cases and potentially do not require definitions, with the biggest category being common-sense knowledge entities.

5.2 Answer Rewriting

We have established that salient entities occur frequently in answers to CIA questions. In this section, we aim at rewriting the answers containing salient entities with the goal to aid users' understanding. To this end, we propose two answer rewriting strategies, depicted in Figure 5.3. The first strategy aims to rewrite the original answer A by inserting inline definitions of the identified salient entities, thus making the answer self-contained. The second strategy makes use of the mixed-initiative CIA paradigm and offers the user to learn more about any of the identified salient entities. Figure 5.3 showcases all rewrite types, further explained in the following sections.

5.2.1 Inline Entity Descriptions

Our first strategy towards ensuring the complete understanding of the answer is based on including the explanations of the identified salient entities in the answer itself. Formally, we rewrite the original answer A by providing inline definitions d_i for each of the salient entities e_i , resulting in the answer rewrite $A\text{-inlinedef}$. The answer $A\text{-inlinedef}$ is thus self-contained, as all of the salient entities are explicitly described. One of the challenges here is to keep the explanations reasonably short and adequate for a conversational setting, as explaining the answer with long definitions would result in a significantly longer answer than the original, thereby overwhelming the user. Thus, we experiment with two alternatives for providing inline definitions.

A-original:

His trial on a few epilepsy patients in 1921 was the first use of the ketogenic diet as a treatment for epilepsy.

A-inlinedef/wiki:

His trial on a few epilepsy (human neurological disease causing seizures) patients in 1921 was the first use of the ketogenic diet as a treatment (attempted medical remediation of a health problem) for epilepsy.

A-inlinedef/human:

His trial on a few epilepsy, a human neurological disease causing seizures, patients in 1921 was the first use of the ketogenic diet as a treatment, referring to attempted medical remediation of a health problem, for epilepsy.

A-followup/offer:

His trial on a few epilepsy patients in 1921 was the first use of the ketogenic diet as a treatment for epilepsy. If you would like to learn more about epilepsy or treatment, feel free to ask.

A-followup/question:

His trial on a few epilepsy patients in 1921 was the first use of the ketogenic diet as a treatment for epilepsy. Would you like to learn more about epilepsy or treatment?

Rewrites with inline definitions

Rewrites with a follow-up

Figure 5.3. Answer rewriting strategies.

Wikibase Entity Descriptions

We utilize a knowledge base to extract definitions of salient entities. Specifically, we consult Wikibase² to retrieve the entry of given entity e_i and get its definition d_i . To construct the final rewritten answer *A-inlinedef/wiki*, we insert d_i in parentheses immediately after the first mention of e_i ($\dots e_1 \dots e_2 \dots e_3 \dots \Rightarrow \dots e_1(d_1) \dots e_2(d_2) \dots e_3(d_3) \dots$).

Manually Curated Entity Description

We hypothesize that entity descriptions inserted into parentheses might appear ineloquent and unnatural for a conversational setting. Therefore, we manually go through the entity descriptions in *A-inlinedef/wiki* answers and rewrite them to sound more natural. The process of manual rewriting involves, among others, avoiding highly technical or too verbose definitions. Moreover, we insert d_i after e_i such that the definition is blended in the text more naturally, i.e., using commas ($\dots e_1 \dots e_2 \dots e_3 \dots \Rightarrow \dots e_1, d_1, \dots e_2, d_2, \dots e_3, d_3, \dots$). This text simplification task could potentially be carried out by a pretrained large-scale language model. However, in our initial experiments with T5 [Raffel et al., 2020], we observed several inaccuracies. As our objective is to measure the usefulness of answer rewrites to users, we opted for human curation to ensure that the findings of this study are not impacted by the imperfections of automatic rewrites. The answer rewritten with this method is referred to as *A-inlinedef/human*.

5.2.2 Mixed-initiative Follow-up Prompt

Under the mixed-initiative paradigm in CIA, the system can at any point take initiative and prompt the user with various elicitation, clarification, or other questions [Allen et al., 1999, Zamani et al., 2022]. As one of the potential limitations of the previously described approach is overwhelming the user with potentially unnecessary entity definitions, we instead ask the user whether they require the explanations of salient entities or not. To this end, we experiment with two different follow-up prompts, described below.

Follow-up Question

The first type of follow-up we propose is a direct question, aimed at asking whether the user is familiar with the salient entities identified in the answer. To

²<https://www.mediawiki.org/wiki/Wikibase>

construct a direct clarifying question, we construct a new answer *A-followup/question* by expanding the original answer *A* with a question “Do you want to learn more about e_1 , e_2 , or e_i ?”, where e_i is in the top N most salient entities identified.

Follow-up Offer

Similarly, an offered follow-up prompt (*A-followup/offer*) is designed by expanding the original answer with “If you wish to learn more about e_1 , e_2 , or e_i , feel free to ask.”

We hypothesize that this strategy offers several benefits over the inline explanation rewrites. First, the user can choose whether they want to learn about the identified salient entities or they are comfortable with moving on with the conversation (they either know enough about the entities or do not care). We note that phrasing the follow-up prompt as a direct question, i.e., “Do you want to learn more about *entity*?” would require the direct answer from the user, potentially disrupting the conversation flow. Instead, our proposed construction of the prompt simply offers the user a possibility for expansion, enabling them to ignore it if they are not interested in learning about the proposed entities. Second, we can learn about the user’s background knowledge by them choosing or not choosing to learn about the salient entities, leading to a potential for personalization of subsequent answers. Third, we encourage engagement with the user by providing potential topics to converse about. While these assumptions intuitively make sense, we formulate specific research questions to assess them empirically.

5.2.3 Evaluation of Answer Rewrite Strategies

In this section, we describe the human-based evaluation procedure for comparing the original answer with the rewritten answers.

Research Questions

The main research question we aim to answer is **RQ2: How to utilize salient entities for answer rewriting for an improved user experience?** Moreover, we also aim to explore what type of rewritten answers users prefer and what methods work the best for generating such rewrites. Thus, we extend our main research question to four more specific questions:

RQ2.a: Do users prefer the rewritten questions over the original ones?

RQ2.b: Which of the two answer rewrite strategies (*A-inlinedef* or *A-followup*) is preferred?

RQ2.c: Is there a preferred way of explaining the salient entities inline (*A-inlinedef/wiki* or *A-inlinedef/human*)?

RQ2.d: Is there a preferred way of offering follow up to the user (*A-followup/question* or *A-followup/offer*)?

RQ2.e: How does the number of salient entities considered in the rewrite (top 1, 2, or 3) affect user preferences?

Experiment Design

We design the evaluation study as a multiple choice inquiry and ask crowd workers to provide their assessments. Given an initial question, the workers need to assume the role of a user and select the answer that they would prefer in an interaction with a conversational assistant. The given options are threefold: an original answer, a rewritten answer with inline explanations, and a rewritten answer with a follow-up prompt in the end. Moreover, to answer research questions RQ2.c and RQ2.d, we vary the methods for inline explanations, as well as the types of questions for the prompt-based rewrite. Note that crowd workers are not aware of those changes and they always have the three mentioned options, without knowing how the rewrites are generated. To ensure consistency, we generate rewrites on the same pool of QA pairs, thus controlling the potential impact of different topics on the rewrite preference. Each question and three answer options, corresponding to original answer, an answer with inline definitions, and an answer with follow-up, is annotated by three different crowd workers.

We ensure the quality and consistency of the annotations by selecting high-quality workers, as described in Section 5.1.2. Moreover, we randomize the order of *A-original*, *A-inlinedef*, and *A-followup* to reduce any potential position bias. In order to gain further insight into the matter, we ask annotators to provide a brief explanation of on why they chose the answer they did. We analyze the provided reasons in depth in the next section. To additionally ensure high quality annotations, we manually inspect all of them, rejecting the crowd workers who carelessly provided nonsensical reasons (e.g., “first one,” “best text,” or simply copy-pasted parts of the answers), and blocking them from further participation in the study. In total, we acquire more than 600 assessments on rewrite type preference with justifications for the choice.

Table 5.3. Answer rewrite preference assessed by crowd workers. Workers are shown three rewritten answers (*A-original*, *A-inlinedef*, and *A-followup*) in a randomized order. Results are broken down by varying the proposed answer rewrite strategies. p -values are reported using a χ^2 test.

Original	<i>A-inlinedef</i>		<i>A-followup</i>		p -value
	/wiki	/human	/offer	/question	
60	66	-	45	-	0.13
56	53	-	-	41	0.28
54	-	60	36	-	0.04
52	-	71	-	27	< 0.01
222	250		149		< 0.01

Results

In this section, we present the results of the crowdsourcing study on answer rewrite type preference and analyze them in the light of the aforementioned research questions.

Original or Rewritten Answer Preference Table 5.3 presents the results of the different combinations of answer rewrites, as explained in Section 5.2.3. To assess whether differences in answer preference are statistically significant, we perform a χ^2 test under the null hypothesis of data being drawn from a uniform probability distribution across the three rewrites (i.e., each row of the table). In response to **RQ2.a**, we observe a preference for one of the answer rewrites, over the original answer (222 for original vs 399 for rewrites, p -value < 0.05). These results suggest that there is a large potential for improving the user experience through answer rewriting. Moreover, the findings suggest a promising direction for further research on answer rewriting in CIA systems, both by providing further inline explanations of certain entities and by offering follow-up clarifications.

Rewrite Type Preference Regarding **RQ2.b**, we observe a preference for answers with inline explanations (*A-inlinedef*) over the answers with a follow up (*A-followup*). Moreover, as indicated in Table 5.3, this preference is prevalent across all combinations of rewrite subtypes. Although not all combinations in Table 5.3 yield statistically significant differences, the overall trend is prevalent across all

of the experiments. This suggests that making the answer self-contained by providing inline entity explanations is more desirable than offering the user to clarify these entities. Contrary to our hypothesis, longer answers obtained by inserting entity descriptions do not seem to overwhelm the majority of the users. However, subjectivity is still important in this scenario, as some users indeed find *A-inlinedef* to be too cluttered, as discussed in the next section.

Rewrite Subtype Preference To address research questions **RQ2.c** and **RQ2.d**, we aggregate the results of different subtypes of answer rewriting. Experiments indicate humanly-curated answer to be slightly more preferred over the Wikibase definitions in parentheses (131 vs. 119), suggesting that more natural rewrites could better help the user understand the answers. This finding is a motivation for the development of answer rewriting methods aimed at defining entities in a more natural manner, compared to entity definitions being inserted into parentheses.

Similarly, *A-followup/offer* is slightly more preferred than *A-followup/question* (81 vs. 68). We hypothesize that a prompt that could be ignored, as opposed to a direct question, would benefit the overall user experience. While both strategies are equally effective in providing the user with desired information, *A-followup/offer* might not impair the flow of the conversation, as it can be ignored if the user does not desire to learn more about proposed entities.

Result Preference by Top N Entities Regarding **RQ2.e**, we report the results on answer preference across top N most salient entities rewritten in Table 5.4. Specifically, we construct the experiment such that the same original answer is rewritten three times, each time with N salient entities taken into account, with $N \in [1, 2, 3]$. Results suggest that the higher the N , i.e., the more entities are defined in the answer, the stronger the user’s preference for *A-inlinedef*. We hypothesize that such answers provide a more complete response to the given question, thus not requiring further explorations of the topic through clarifying prompts.

Analysis

In order to gain further insight into answer rewrite preferences, we manually analyze responses from crowd workers. Recall that workers were asked to justify why they would prefer the answer rewrite they have chosen. We perform a qualitative analysis of the provided reasons by identifying re-occurring reasons for workers’ choices. We find five distinct patterns of the provided reasons, presented in Table 5.5. To estimate the prevalence of each of these, we randomly

Table 5.4. Results on answer rewrite preference by top N salient entities rewritten. The p-value is computed by χ^2 test.

	<i>A</i>	<i>A-inlinedef</i>	<i>A-followup</i>	p-value
Top 1	80	84	65	0.27
Top 2	87	83	42	< 0.01
Top 3	55	83	42	< 0.01
Total	222	250	149	< 0.01

Table 5.5. Reasons for choosing one answer over another, grouped together by observed patterns. The last column presents the prevalence of the pattern in the manually analyzed portion of the dataset.

Reason for choice	Preference	Pattern	Prev.
* "The terms seem to be very specialized in terms of the question and need at least some elucidation to understand" * "not everyone knows how the body functions."	<i>A-inlinedef</i> <i>A-inlinedef</i>	Better explanation	28%
* "The most concise answer. I don't think the recipient would want any more particular information especially about Africa" * "it's the most complete answer that doesn't add unnecessary stuff inside brackets of brackets."	<i>A-original</i> <i>A-original</i>	Concise	28%
* "the other answers have text that shouldn't be there or is too wordy." * "I prefer this one because it doesn't have a question on the end, and because it is the most clear and direct, w/o any parentheticals."	<i>A-original</i> <i>A-original</i>	No clutter	15%
* "invites you to ask for more information about it" * "I choose 1, because I want to learn more about solar energy."	<i>A-followup</i> <i>A-followup</i>	Learn more	14%
* "I prefer the first one because it is the most concise answer. One doesn't need to be told they ask more questions as in answer 2, and answer 3 rambles a bit." * "the topic is very familiar for me and i have some knowledge about chemical energy so i choose this option"	<i>A-original</i> <i>A-inlinedef</i>	Other	10%
* "This doesn't overexplain the compatibility layer aspect, making me feel overwhelmed, and allows me to ask about it if I want instead" * "Apart from answering the question, the assistant is more interactive and continues to ask whether I would like to learn more about the bank of England."	<i>A-followup</i> <i>A-followup</i>	Natural	4%

select 100 responses from the crowd workers and label them using these patterns. In our analysis, we allow for multiple patterns to be associated with a single reason of choice, as for example a worker can pick an answer because it is both *concise* and *natural*. The patterns and their frequency in the analyzed set are shown in Table 5.5. Overall, we find that:

- Users prefer the original answer *A* mostly because it is short, concise, and does not introduce unnecessary clutter.
- Users prefer the answer with inline explanations *A-inlinedef* when they appreciate additional information and think it provides a well-defined answer.
- Users prefer the answer with a follow-up prompt *A-followup* when they think it is the most welcoming of the different answer alternatives, inviting for further conversation, but is not unnecessarily overwhelming with long explanations of entities they perhaps do not require explanation of.

Although all of the workers provided reasonable justifications for their selection, the inter-annotator agreement, as measured by Fleiss' κ , is 0.03, indicating high subjectivity. From the conducted experiments and observed patterns in user's preferences, we find solid evidence that entity-based answer rewriting can lead to an improved conversational user experience. At the same time, we also find that the choice of preferred answer format is highly subjective, which calls for further research on the personalization of such approaches.

LLM-based Answer Rewriting

While entity definitions are currently either taken from a knowledge base or curated by human intervention, the question naturally arises: Could this task not be tackled in an end-to-end manner by a large language model (LLM)? Given promising advancements in text simplification, e.g., with T5 [Sheang and Saggion, 2021], and the broad variety of knowledge and language capabilities demonstrated by ChatGPT, there are reasons to believe that the answer rewriting task could be performed by simply engineering the “right” prompt. Below, we present some anecdotal evidence to the contrary, by presenting results obtained with a state-of-the-art LLM, gpt-3.5-turbo [Brown et al., 2020].

Specifically, we consider two types of prompts: (1) instructing the LLM to rewrite the answer for easier understandability and (2) additionally, including the specific entities that need to be explained. For both settings, we experimented with both zero- and few-shot prompts. Due to space constraints, we only include a few examples in Table 5.6 to illustrate the main limitations we identified:

Table 5.6. Examples of LLM rewrites for answer rewriting with the original answer being: “The lipid bilayer is a thin polar membrane made of two layers of lipid molecules. These membranes are flat sheets that form a continuous barrier around all cells.”

Prompt	Answer rewrite	Limitation
Rewrite the passage below in a simplified language, so it is understandable to everyone.	The lipid bilayer is a very thin and special kind of membrane made up of two layers of lipid molecules. These membranes are <i>like flat sheets</i> that wrap around all cells and create a protective barrier.	Knowledge distortion
Rewrite the passage below so it contains explanations of the main concepts so that it is easier to understand.	The lipid bilayer is a structure consisting of two layers of lipid molecules that form a thin, polar membrane. This membrane serves as a protective barrier encircling every cell in the body. It is composed of flat sheets, ensuring a continuous and consistent covering.	No entity explanations
Rewrite the passage below so it is understandable to wide audiences, by explaining the following entities: "lipid molecules", "polar membrane".	The cells in our body are surrounded by a very thin... head is attracted to water (hydrophilic) while the tail is repelled by it (hydrophobic)... (5 sentences).	Answer lengthening
Rewrite the passage below so it contains brief explanations of "lipid molecules" and "polar membrane".	The lipid bilayer, which is made up of lipid molecules, is a thin and polar membrane. Lipid molecules are special types of fats. This polar membrane consists of two layers and forms flat sheets. It acts as a continuous barrier that encloses and protects all cells.	—

- *Failure to identify salient entities*: When entities that require explanations aren't explicitly stated, the LLM can simply reword the answer, without providing any additional information.
- *Knowledge distortion*: Certain salient entities are removed from the original answer, causing the loss of information by oversimplifying the text.
- *Answer length*: when explicitly stated which entities require inline explanations, the LLM tends to significantly lengthen the original answer (from 1-2 to 5-6 sentences).
- *Inconsistency*: Although hallucination is a known issue in LLMs [Ji et al., 2023], we also observe inconsistency, i.e., a high degree of variation in answer quality, when generating answers to the same prompt multiple times (controllable with parameters to some degree) and across different examples (not controllable).

That said, LLMs can also generate appropriate rewrites, as illustrated by the last example in Table 5.6, which is both concise and natural, while still covering all of the salient entities. However, it is evident that the salient entities had to be explicitly stated and that prompts need to be carefully engineered for the desired outcome.

The main take home message of our study is that entity-based answer rewriting can improve the user experience, but to unlock its full potential, the identification of salient entities as well as the preferred form of answer rewrite need

to be addressed in a personalized manner. These parts require future research. When it comes to the actual generation of the rewritten answer, there is a large potential for utilizing LLMs, provided that they are prompted with the specific entities and the desired format of rewrite.

5.2.4 Discussion

Ours is a novel task in a conversational setting, which makes evaluation inherently challenging. In this section, we reflect on some of the design decisions, acknowledge limitations, and highlight possible future research directions, including potentially revisiting some of the design choices.

Impact of Rewriting on Answer Length

Rewriting answers in CIA by inserting inline definitions of salient entities lengthens the original answer. As observed in our experiments, up to three entity definitions do not seem to hurt the answer rewrite, as such rewrite was often chosen by the crowd workers. However, in case the answer becomes too long due to a large number of salient entities, the amount of them we provide definitions for can be reduced by taking only the top N entities, as ordered by the salience scores.

Text- vs. Voice-based CIA

We hypothesize that results on answer rewrite preference might differ in a voice-only setting, as the user is not able to skim through potentially unnecessary parts of the answer. As such, preference for inline definitions might not be so prevalent, as users could not simply skim through the text and would in fact need to listen to the extended answers. We aim to explore the aforementioned questions in further research.

More Realistic Conversational Setting

Design-wise, we compare answer rewrites turn by turn, rather than evaluating the whole conversations. This is often the case in crowdsourcing-based studies due to the limited availability of users, although recent research points out the benefits of multi-turn dialogue evaluation [Li et al., 2019]. At the same time, utterances in our study are self-contained and do not necessarily require full conversation history for correct assessments. Also, we provide an analysis of the salient entity evolution throughout the conversation. Nevertheless, as part of our

future work, we aim to build multiple CIA systems based on answer rewrite type (e.g., a system that generates answers with inline explanations of salient entities and a system that offers follow-up prompts) and perform a thorough user study to validate the findings of this work.

Salient Entity Annotation

While other automated options for extracting salient entities exist, we opted for annotating salience through crowdsourcing with a goal of acquiring high-quality data. Nevertheless, despite having multiple controls in place for ensuring quality (from the selection of crowd workers to using test questions), the inter-annotator agreement turned out to be relatively low. We attribute this to the high subjectivity of the task, as workers' perception of what is "essential to understand" might differ, in relation to their personal knowledge and their understanding of what "essential" means. We acknowledge the possibility of the annotation task being set up this way to be too open for interpretations, or simply too hard, thus leading to low inter-annotator agreement. In the future we plan to repeat the annotation process as part of a dedicated study, aiming to untangle what role prerequisite knowledge and subjectivity might play here. Nonetheless, we believe the findings of this study on question rewrite strategies and preferences to be sound and useful for the research community.

5.3 Conclusions

In this study, we analyzed the presence of salient entities in conversational information seeking interactions. We found that most of the answers generated by the search system contain some amount of salient entities, required for the complete comprehension of the answer. Moreover, with a goal of ensuring that the user understand these answers, we proposed two strategies for answer rewriting. The first one is based on providing inline definitions of salient entities, while the second one explicitly offers the user to learn more about the entities they might be unfamiliar with. The suggested methods were extensively assessed through human-based evaluation, indicating user preference for answers with inline definitions, over the follow-up prompt-based rewrites. We hope that these findings provide a strong motivation for further research on entity-based answer rewriting.

Part II

User Simulation

Chapter 6

User Simulation for Conversational Search

Evaluation of mixed-initiative conversational search systems is arduous [Penha and Hauff, 2020, Lipani et al., 2021]. The challenge arises from the fact that expensive and time-consuming human-in-the-loop evaluation procedures and user studies are required to properly evaluate conversational systems. That is even more the case with mixed initiatives, as the number of possible conversations is essentially limitless [Balog, 2021]. Such studies require real users to interact with the search system for several conversational turns, provide answers to potential clarifying questions prompted by the system, and provide feedback to system’s responses. A relatively simple solution is to conduct offline corpus-based evaluation [Dalton et al., 2020, Aliannejadi et al., 2019b, Qu et al., 2018]. Existing data collections are mainly built based on online human–human conversations [Qu et al., 2018], synthetic human–system interactions [Dalton et al., 2020], and multiple rounds of crowdsourcing [Aliannejadi et al., 2021b]. However, this limits the system to selecting clarifying questions from a pre-defined set of questions, which does not transfer well to the real-world scenario. No existing data collections, however, feature explicit user feedback extensively in a conversation, thus limiting research in this area. Moreover, such offline evaluation remains limited to single-turn interaction, as the pre-defined questions are associated with corresponding answers and are not aware of any previous interactions. User simulation has been proposed to tackle the shortcomings of corpus-based and user-based evaluation methodologies. The aim of a simulated user is to capture the behavior of a real user, i.e., being capable of having multi-turn interactions on unseen data, while still being scalable and inexpensive like other offline evaluation methods [Salle et al., 2021, Zhang and Balog, 2020].

In this chapter, we further motivate the use of user simulation approaches for conversational search system evaluation and define necessary requirements of such simulators (Sect. 6.1). Section 6.2 describes methodology used for evaluating user simulators. Next, in Sect. 6.3, we describe our proposed LLM-based user simulator for answering clarifying questions in mixed-initiative conversational search [Sekulić et al., 2022]. Given the initial information need description, our simulator is able to answer clarifying questions posed by the system, which mitigates the need for human-in-the-loop evaluation, as was previously the standard [Aliannejadi et al., 2019b, 2020b]. However, the described approach remained limited to answering clarifying questions and did not properly utilize the conversational history (i.e., simulators performance dropped in multi-turn setting). We address these limitations in Sect. 6.4, where we propose a user simulator capable of multi-turn interactions extending beyond answering clarifying questions. Specifically, we enable the simulator to provide feedback, both positive and negative, to the systems responses, in addition to answering posed questions. Section 6.6 then provides an in-depth analysis of the utterances generated by the proposed LLM-based user simulators.

Large parts of this chapter were published in Sekulić et al. [2022], Owoicho et al. [2023], and Sekulić et al. [2024].

6.1 Motivation and Problem Setting

In this section, we explain a user’s role in evaluating conversational search systems. Moreover, we discuss several desired characteristics of a user simulator and formalize the task of simulating a user for conversational search.

6.1.1 User’s Role in Conversational Search System Evaluation

Previous work in task-oriented dialogue systems and conversational search systems mainly evaluate the performance of the systems in an offline setting using a corpus-based approach [Deriu et al., 2021]. The offline evaluation must accurately reflect the nature of conversational systems, as the evaluation is possible only at a single-turn level. Thus, to adequately capture the nature of the conversational search task, it is necessary to involve users in the evaluation procedure [Black et al., 2011, Li et al., 2019]. User involvement allows proper evaluation of multi-turn conversational systems, where users and systems take turns in a conversation. Even with such an approach that most precisely captures the performance of the systems in a real-world scenario, the involvement of users in the

evaluation is tiresome, expensive, and unscalable. To alleviate the evaluation of dialogue systems while still accurately capturing the overall performance, a simulated user approach has been proposed [Zhang and Balog, 2020, Sun et al., 2021]. The simulated user is intended to provide a substitute for real users [Balog, 2021], as it is easily scalable, cheap, fast, and consistent. Next, we formally describe the characteristics of a simulated user for conversational search system evaluation.

6.1.2 Formal Definition

In this section, we formalize the requirements of the mixed-initiative conversational search systems, followed by mapping them to the ones of a user simulator.

Formally, a conversational search session is carried out between a user U and a system S . The search session consists of multiple turns of the user's utterances u and the system's utterances s , forming conversational history $H = [u^1, s^1, \dots, u^{t-1}, s^{t-1}]$, with u^t and s^t corresponding to user's and system's utterance at conversational turn t , respectively. One key factor is that we differentiate between discourse types of user utterances u , namely queries u_q , answers u_a to clarifying questions posed by the system, and explicit feedback u_f to the system's responses. Similarly, the system's utterance s can either be a response s_r aimed at satisfying the user's information need IN or a clarifying question s_{cq} aimed at elucidating the user's information need. One of the inputs to various modules of mixed-initiative systems can as well be the ranked list of results $R = [r_1, r_2, \dots, r_N]$, retrieved in response to u^t , where N is the maximum number of results considered. We note that conversational search systems should also contain modules for contextual query rewriting, document retrieval, re-ranking, and result presentation, but we omit the details for the sake of brevity. More details on these modules can be found in Chapt. 2.

Mixed-Initiative Systems

A conversational search system should be able to effectively conduct contextual query understanding, document retrieval, and response generation. Moreover, under the mixed-initiative paradigm, the system can at any point take initiative and prompt the user with various suggestions or clarifying questions [Radlinski and Craswell, 2017]. Throughout this dissertation, we focus mostly on clarifying questions as a most-widely researched area of the paradigm. Thus, below we formalize the mentioned tasks.

Clarification need. When necessary, e.g., in case of a user’s query being ambiguous, the CS system can ask a clarifying question, or questions, to elucidate the user’s underlying information need. Thus, the first challenge of a mixed-initiative search system is to assess the need for clarification [Aliannejadi et al., 2021b]. Specifically, given the current user’s utterance u^t , the task is to predict whether asking a clarifying question is required, or whether the system should issue a response aimed at answering the user’s question. Thus one of the modules of the search system needs to model a function $clarification_need = f(u^t|H, R)$, where $clarification_need \in \{0, 1\}$, indicating whether not to ask or to ask a clarifying question.

Clarifying questions. As mentioned, asking clarifying questions methods can be broadly categorized into *question selection* and *question generation* [Aliannejadi et al., 2019b, 2020b] methods. In the first approach, given the current user utterance, u^t , and a conversational history H , the task is to select an appropriate clarifying question from a predefined pool of questions $CQ = \{cq_1, cq_2, \dots, cq_n\}$. Formally, we model $s_{cq} = \phi(u_t|H, R, CQ)$ where ϕ is our question selection model. Question generation poses itself as a necessary step in CS, going beyond selection from pre-defined corpora. Formally, the task of the question generation module is to model ψ in $s_{cq} = \psi(u_t|H, R)$.

Processing User Feedback. A CS system needs to be able to process feedback given by the user during the conversation including both answers to clarifying questions and explicit feedback to the system’s response. Therefore, the system, in both cases, needs to update its internal state by refining its representation of the user’s information need. Formally, we define updates to the system’s interpretation of the user’s information need, as query reformulation: $u^{t'} = \gamma(u^t|H)$, where γ is the query rewriting model. We note that, depending on the design choices of mixed-initiative systems, different forms of feedback, i.e., answers to clarifying questions and explicit feedback to the system’s responses, can be modeled differently — e.g., $u^{t'} = \gamma_1(u_q^t|H)$ and $u^{t'} = \gamma_2(u_f^t|H)$. Furthermore, we point out that similar methods might be used to model contextual query reformulation, which aims at resolving current user utterance in the context of conversational history: $u_q^{t'} = \gamma_3(u_q^t|H)$.

User simulation

A user simulator aims to mimic key user’s roles in MI interactions. Although Balog [2021] defines several desired properties of a realistic user simulator, we focus

on the simulator’s ability to capture and communicate aspects of the information need. The simulator should coherently answer any posed clarifying questions, or provide positive/negative feedback to the system’s responses. In other words, the requirements of a user simulator are complementary to the ones of mixed-initiative CS systems. Inspired by [Zhang and Balog \[2020\]](#), we base our user interaction model on the general QRFA model for the conversational information-seeking process [[Vakulenko et al., 2019](#)].

Formally, the user simulator needs to be able to carry out multi-turn interactions with the search system and generate a variety of different utterances: (i) u_q : seek information through querying; (ii) u_a : answer clarifying questions; and (iii) u_f : provide feedback to systems’ responses. All of the utterances generated by the simulator need to be in line with the underlying information need IN . First, a simulator needs to represent its information need by constructing a query utterance $u_q = h(IN)$. Moreover, when prompted with a clarifying question utterance s_{cq} , the user simulator should be able to provide an answer $u_a = \theta_1(s_{cq}|H, IN)$, where θ_1 denotes answer generation model. Similarly, when given a response s_r to its query, it needs to generate feedback $u_f = \theta_2(s_r|H, IN)$, where θ_2 is the response generation function.

6.1.3 Research Questions

Our aim is to evaluate whether our proposed simulated user can replace real users in answering clarifying questions of conversational search systems, which would make the evaluation of such systems significantly less troublesome. Overall, we aim to answer four main research questions:

- RQ1:** To what extent are the generated answers in line with the underlying information need?
- RQ2:** How coherent and natural is the language of the generated answers?
- RQ3:** To what extent does the retrieval model of the conversational search system benefit from the generated answers?
- RQ4:** How do LLM-based simulators behave in multi-turn interactions?
- RQ5:** What are the advantages and disadvantages of either simulation methodologies?

To address these questions, we first compute several natural language generation metrics to compare the generated answers to the oracle human answers

from ClariQ. As several NLG metrics received criticism from the NLP community, especially since they do not correlate well with the coherence of the text, we perform a crowdsourcing study to evaluate the *naturalness* of generated answers. In order to evaluate whether the generated answers are in line with the actual information need, we carry out additional crowdsourcing study, evaluating the *usefulness* of answers. Moreover, we analyze the impact of generated answers to retrieval model performance, by performing a document retrieval before and after answering the prompted clarifying question. Finally, we perform qualitative analysis of generated answers.

We compare our GPT-2-based user simulator to two competitive sequence-to-sequence baselines. The first baseline is a multi-layer bidirectional LSTM encoder-decoder network for sequence-to-sequence tasks [Sutskever et al., 2014].¹ The second baseline is a transformer-based encoder-decoder network, based on Vaswani et al. [Vaswani et al., 2017]. We perform hyperparameter search to select the learning rate, number of layers, and hidden dimension of the models. Both baselines are trained with the same input as our main model.

6.2 Evaluation Methodology

In this section, we describe the evaluation procedure for assessing the performance of a user simulator. Through the chapter, we present several methods for simulating a user, and thus require a method for comparing its performance in the aforementioned task of generating user-like utterances. Our evaluation relies on three main distinct approaches:

- (i) comparison of generated utterances with the gold standard utterances;
- (ii) pairwise comparison of utterances generated by two different simulators (or a simulator and a human) via crowdsourcing-based annotation procedures;
- (iii) comparison of document retrieval performance before and after the simulator-system interactions.

Next, we outline the specifics of the evaluation approaches. Next section, however, first describes the data collections used for evaluation, as well as training of certain models.

¹We use the IBM implementation for our experiments: <https://github.com/IBM/pytorch-seq2seq>

Table 6.1. Statistics for Qulac and ClariQ datasets.

	Qulac	ClariQ
Number of topics	198	237
Number of facets	762	891
Number of questions	2,639	3,304
Number of question-answer pairs	10,277	11,489

6.2.1 Data

We are concerned with mixed-initiative conversational search. Thus, throughout our research, we rely on conversational datasets containing a degree of mixed-initiative, mainly in the form of clarifying questions. In this section, we review two datasets based on clarifications in a single-turn setting, present our extension of such a dataset to a multi-turn scenario, and describe a conversational passage retrieval dataset.

Qulac and ClariQ

For the purpose of training and evaluating the user simulator approaches, we use two publicly available datasets — Qulac [Aliannejadi et al., 2019b] and ClariQ [Aliannejadi et al., 2020b]. The aim of both datasets is to foster research in the field of asking clarifying questions in open-domain conversational search. Qulac was created on top of the TREC Web Track 2009-12 collection. The Web Track collection contains ambiguous and faceted queries, which often require clarification when addressed in a conversational setting. Given a topic from the dataset, clarifying questions were collected via crowdsourcing. Then, given a topic and a specific facet of the topic, workers were employed to gather answers to these clarifying questions. This results in a tuple of (*topic*, *facet*, *clarifying_question*, *answer*). Most of the topics in the dataset are multi-faceted and ambiguous, meaning that the clarifying questions and answers need to be in line with the actual facet. ClariQ is an extension of Qulac created for the ConvAI3 challenge [Aliannejadi et al., 2020b] and contains additional non-ambiguous topics. Relevant statistics of the datasets are presented in Table 6.1.

We utilize these datasets by feeding the corresponding elements as inputs to the simulated user methods. Specifically, *facet* from Qulac and ClariQ represents the description of the underlying information need (*IN*), as it describes in detail what the intent behind the issued *query* is. Moreover, *clarifying_question* rep-

resents the prompted clarifying question s_{cq} , while *answer* (u_a) is our language modeling target or the reference text for automated evaluation.

Curated Multi-turn Dataset

A major drawback of Qulac and ClariQ is that they are both built for single-turn offline evaluation. In reality, a conversational search system is likely to engage in a multi-turn dialog in order to elucidate user need. To bridge the gap between single- and multi-turn interactions, we construct multi-turn data that resembles a more realistic interaction between a user and the system. We use the constructed dataset both for further fine-tuning of our simulator and their evaluation.

To acquire the multi-turn data, we construct a crowdsourcing-based human-to-human interaction. At each conversational turn, crowdsourcing worker is tasked to behave as a search system by asking a clarifying question on the topic of the conversation. Then, another worker is tasked to provide the answer to that question having in mind the underlying information need and the conversation history, imitating the behavior of the real user. We construct in 500 conversations up to depth of three, i.e., we have three sequential question-answers pairs for a topic and its facet.

In order to further study the effects certain clarifying questions have on the whole search experience, we construct several edge cases. In such cases, the clarifying question prompted by the search system is considered faulty, as it is either a repetition, off topic, unnecessary, or completely ignores previous user's answers. We obtain answers to these questions to provide a more realistic data for the training of our model, making our simulated user as human-like as possible. These clarifying questions are intended to simulate a conversational search system of poor quality and provide insight into user's responses to such questions. We employ workers to provide answers to additional 500 clarifying questions of poor quality, up to the depth of two. The specific edge cases and their descriptions with examples are presented in Table 6.2. We publicly release the acquired multi-turn datasets.

TREC CAsT

We additionally use the TREC CAsT [Owoicho et al., 2022] benchmark, designed for the development and evaluation of conversational search systems. CAsT is composed of a series of fixed conversations, each with a pre-determined trajectory and containing a series of topical user utterances and canonical responses.

Table 6.2. Multi-turn dataset acquired through crowdsourcing for fine-tuning a more realistic user simulator. Sample conversations of depth 3 are omitted for space purposes.

Question case	Description	Sample conversation	N
Normal	A good system naturally continues the conversation.	U: I'm looking for information on dieting S: Are you looking for dieting tips? U: Yes and exercise tips as well S: Do you need anything specific in relation to counting calories you consume daily? U: Yes, I would like to know more about that topic.	500
Repeat	System repeats the previous question.	U: Find information on raised gardens. S: Do you need information on materials needed? U: No, I want to find plans S: Do you need information on materials needed? U: I want what I previously asked for.	50
Off-topic	System asks the user an off-topic question.	U: I'm looking for an online world atlas. S: Are you interested in satellite maps? U: No, I want an online world atlas S: Which mountain ski resort would you like information around the pocono area? U: I am not interested in this topic.	50
Similar	System asks a question similar to the previous one, ignoring the user's answer.	U: I'm looking for information about mayo clinic Jacksonville FL S: Would you like to request an appointment? U: yes S: Are you looking for the address of mayo clinic jacksonville fl? U: I just want to request an appointment.	400

We focus on the fourth edition, CAsT'22, because it is the only dataset that includes mixed-initiative interactions.

Since each turn in CAsT does not have an *IN* description, we augment it by adding turn-level information need descriptions. Specifically, two expert annotators independently study each CAsT utterance in the conversation context and describe the full information need in a sentence. We decide on the length of the information, following the typical topic description in the TREC Web track topic list [Clarke et al., 2009]. We instruct the annotators to take into account various sources of information such as the canonical responses and the rewritten queries. The final goal is to generate a self-contained description for each user utterance in CAsT. One could argue that the human rewritten utterances would be sufficient for this aim. In our preliminary analysis, we discover that the re-written utterances miss various contextual information that makes them dependent on the overall conversation context. It is worth noting that we stress on this fact during our annotation process and instruct the annotators to take this aspect into account. We compare the generated information need descriptions by the two annotators. In case of minor differences, we select either of them. However, in cases where the difference is major there is discussion until agreement is reached.

6.2.2 NLG Metrics

We compute several standard metrics for evaluating the generated language. We use two widely adopted metrics based on n-gram overlap between the generated and the reference text. These are BLEU [Papineni et al., 2002] and ROUGE [Lin, 2004]. Additionally, we compute METEOR [Banerjee and Lavie, 2005], which was reported to have higher correlation with human judgments than BLEU and ROUGE [Banerjee and Lavie, 2005]. METEOR mitigates the shortcomings of BLEU and ROUGE by not just counting the overlap of n-grams, but also considering their stems, WordNet synonyms, and paraphrases. Next, we compute the EmbeddingAverage and SkipThought metrics aiming to capture the semantics of the generated text, as they are based on the word embeddings of each token in the generated and the target text. The metric is then defined as a cosine similarity between the means of the word embeddings in the two texts [Kryscinski et al., 2019].

The aforementioned metrics require a reference utterance, that we compare the generated one with. In the context of user simulators, we use answers u_a from ClariQ and Qulac as a reference utterance. Thus, when prompted with a clarifying question, the simulator generates the answer u'_a that is then compared

with the reference one u_a . Formally, the comparison results in a value m that is a function of both utterances: $m = v(u_a, u'_a)$, where v and m are the metric used and the value of the metric, respectively. For all of the computed metrics, higher values of m represent higher similarity of the generated answer to the reference one.

We note that recent studies have revealed several flaws of the standard heuristic-based NLG metrics [Sai et al., 2022, Liu et al., 2016, Callison-Burch et al., 2006, Ritter et al., 2010, Mairesse et al., 2010]. The criticism comes from the low correlation of the automated metrics with human judgments, thus making the metrics untrustworthy or even misleading. Moreover, Stent et al. [2005] found that several automatic metrics, including BLEU, correlate negatively with human judgments on fluency of generated text. Thus, we make use of crowdsourcing for gathering human judgments, as described in the next section.

6.2.3 Human Evaluation

In this section, we describe our annotation-based procedure for evaluating utterances generated by the user simulators. This procedure is utilized throughout the chapter, with its basis describe here, and specifics detailed in the corresponding sections.

Usefulness and Naturalness

In order to simulate a real user, the generated responses need to be fluent and coherent. Thus, we study the *naturalness* of the generated answers. We define *naturalness* as an utterance being natural, fluent, and likely generated by a human. Similarly, fluency [Callison-Burch et al., 2006] and humanness [See et al., 2019] have been used for evaluating generated text.

Moreover, we assess the *usefulness* of the utterances generated by our simulated user. We define *usefulness* as an utterance being in line with the underlying information need and guiding the conversation towards the topic of the information need. This definition of usefulness can be related to similar metrics in previous work, such as adequacy [Stent et al., 2005] and informativeness [Chuklin et al., 2019].

Crowdsourcing

We perform human evaluation of the generated utterances. Specifically, we rely on crowdsourcing and use Amazon Mechanical Turk² for acquiring workers. Moreover, we take several steps to ensure high-quality annotations:

- (i) We select workers based in the United States, in order to mitigate potential language barriers;
- (ii) The selected workers have above 95% lifetime approval rate and at least 5 000 approved HITs;
- (iii) We reject workers with wrong annotations on manually constructed test set;
- (iv) We provide fair compensation of, on average, \$0.25 per HIT, which, with an average completion time of about 30 seconds, results in more than 300% of the minimum wage in the United States.

In the case of deviations from the described crowdsourcing practices, we detail the specifics in the relevant sections explaining the setting of the experiment.

Pairwise Comparison

To assess the relative performance of two simulators, or to compare a simulator to the human-generated data, we perform the annotation in a pairwise setting. Specifically, each worker is presented with a number of utterance pairs, where one of the utterance was generated by the primary method (U_A) and the other utterance was generated by the alternative one (U_B). In case of comparison to the gold standard, one utterance in the pair is generated by our model and the other is generated by a human, taken from the reference collection. Formally, we compare two users, U_A and U_B , where U_A represents the primary user simulator method and U_B represents either an alternative simulator method or a real user.

Crowd worker's task is then to provide judgment on which utterance, one generated by U_A or the one generated by U_B , is more natural or useful, depending on the study. The workers have been provided with the context, i.e., the initial query, facet description, and clarifying question. Moreover, we shuffle the positions of the utterances generated by the primary and alternative methods to mitigate the potential positional bias. We define a *win* for the primary method if both annotators voted its generated utterance as more natural/useful, and *loss*

²<https://www.mturk.com>

for the primary method if both voted the utterance generated by the alternative method (or the utterance taken from the reference collection) as more natural/useful. In case the two workers voted differently on a single utterance pair, we define that as a *tie*.

6.2.4 Impact of Feedback to Retrieval Performance

As the basis for the offline evaluation of open-domain conversational search systems, Aliannejadi et al. [2019b] propose the document retrieval task with the answer to the prompted clarifying question. The initial query is expanded with the text of the clarifying question and the user’s answer and then fed to a retrieval model, such as BM25 or Query Likelihood. The intuition is that if the clarifying question and the answer were both useful, the retrieval model will perform better with them in input, alongside the initial query. In fact, they show significant improvements in retrieval performance with the additional input compared to query-only setting, which is in general a strong motivation for asking clarifying questions in conversational search. The initial retrieval is performed on ClueWeb09b collection, while queries are taken from the ClariQ development set. Each query is associated with the information need (facet) description and several clarifying questions. We then generate answers to these questions and perform additional document retrieval with the initial query expanded with generated answers and corresponding questions.

Formally, we assess retrieval performance at turn $t = 1$, when only the initial query u_q^1 is fed to the retrieval pipeline P . The retrieval yields a ranked list of responses R^1 , which is then compared to the query relevance judgments, yielding the metric value m^1 , at turn $t = 1$. We rely on standard IR metrics, such as mean reciprocal rank (MRR), normalized discounted cumulative gain (nDCG), etc., to compute m^i . Next, the (simulated) user, when prompted with the system utterance s_{cq}^2 or s_r^2 , generates the answer u_a^3 or provides explicit feedback u_f^3 . As denoted in Sect. 6.1.2, the system then utilizes this feedback and provides another ranked list of responses R^3 , which we compare to the relevance judgments and compute m^3 . Finally, we compare m^3 and m^1 . If $m^3 > m^1$, we conclude that the feedback of the user simulator improved the retrieval performance and can be considered helpful and useful. We note that the improvement can become due to a useful system utterance as well, such as a useful clarifying question.

We perform this evaluation both in single- and multi-turn settings. In multi-turn setting, the performance is finally evaluated after an arbitrary number of conversational turns. With this study, we aim to answer research question RQ3, i.e., how beneficial are the generated answers to the retrieval model of a conver-

sational search system. Our hypothesis is that the retrieval performance should increase when the initial query is expanded with the generated feedback, including answers and explicit feedback.

6.2.5 Explicit Feedback

Explicit feedback u_f , generated in response to the systems' responses, needs to be reliable and accurate. To this end, at each turn u_q^t , the system returns response s_r^{t+1} and the simulator generates feedback u_f^{t+2} . At this point, the utterance u_f^{t+2} is externally annotated as positive or negative feedback. Our aim is to measure correlation of retrieval performance at turn u_q^t and type of feedback u_f^{t+2} (positive or negative). Finally, we assess potential differences, as measured by retrieval metrics, between turns that received positive vs negative feedback. Positive feedback should be generated in cases where performance is high, while negative feedback should be given when performance is low.

6.3 Answering Clarifying Questions via User Simulation

In this section, we propose a conversational User Simulator, dubbed USi – a model capable of multi-turn interactions with a general mixed-initiative conversational search system, published in Sekulić et al. [2022]. Given an initial information need, USi interacts with the conversational system by accurately answering clarifying questions prompted by the system. The answers are in line with the underlying information need and help elucidate the given intent. Moreover, USi generate answers in fluent and coherent natural language, making its responses comparable to real users. Previous work on the topic remained limited to retrieving answers from a pre-defined pool of human-generated answers to clarifying questions, e.g., CoSearcher [Salle et al., 2021], or providing feedback with template-based answers in recommender systems [Zhang and Balog, 2020].

We base our proposed user simulator on a large-scale transformer-based language model, namely GPT-2 [Radford et al., 2019], ensuring the near-human quality of generated text. Moreover, USi generates answers to clarifying questions in line with the initial information need, simulating the behavior of a real user. We ensure that through a specific training procedure, resulting in a semantically controlled language model. We evaluate the feasibility of our approach with an exhaustive set of experiments, including automated metrics as well as human judgments.

Following the definitions from Sect. 6.1.2, our simulated user U is initialized with a given information need IN . Simulated user U formulates its need in a form of the initial query u_q , which is then given to the mixed-initiative conversational system S . The aim of the system S is to elucidate the information need IN through a series of clarifying questions cq . We do not go into details of the implementation of such a system, but different approaches have been proposed in recent literature [Aliannejadi et al., 2019b, Hashemi et al., 2020]. Next, the simulated user U needs to provide an answer u_a to the system’s question. The answer u_a needs to be in line with user’s information need IN .

In the next section, we motivate and describe in detail our proposed User Simulator, USi. We make USi semantically-controlled through specific language modeling training. We base our simulated user on a large-scale transformer-based model, namely GPT-2 [Radford et al., 2019].

6.3.1 Semantically-Controlled Text Generation

We define the task of generating answers to clarifying questions as a sequence generation task. Thus, we employ language modeling as our main tool for generating sequences. The goal of a language model (LM) is to learn the probability distribution $p_\theta(x)$ of a sequence of length n : $x = [x_1, x_2, \dots, x_n]$, where θ are the parameters of the LM. Current state-of-the-art language models, such as GPT-2, learn the distribution in an auto-regressive manner, i.e., formulating the task as next-word prediction task:

$$p_\theta(\mathbf{x}) = \prod_{i=1}^n p_\theta(x_i | x_{<i}) \quad (6.1)$$

However, recent research showed that large-scale transformer-based language models, although generating text of near-human quality, are prone to “hallucination” [Dziri et al., 2021] and in general lack semantic guidance [Rosset et al., 2020]. Thus, with a specific fine-tuning technique and careful input arrangement, we fine-tune semantically-conditioned LM. As mentioned, answer generation needs to be conditioned on the underlying information need. To this aim, we model θ from $u_a = \theta(s_{cq} | H, IN)$, defined in Sect. 6.1.2, by learning the probability distribution of generating an answer u_a :

$$p_\theta(\mathbf{u}_a | IN, u_q, s_{cq}) = \prod_{i=1}^n p_\theta(u_a^i | u_a^{<i}, IN, u_q, s_{cq}) \quad (6.2)$$

where u_a^i is the current token of the answer, $u_a^{<i}$ are all the previous ones, while

IN , u_q , and s_{cq} correspond to the information need, the initial query, and the current clarifying question.

6.3.2 GPT2-Based Simulated User

GPT-2 is a large-scale transformer-based language model trained on a dataset of 8 million web pages, capable of synthesizing text of near human quality [Radford et al., 2019]. Moreover, as it is trained on an extremely diverse dataset, it can generate text on various topics, which can be primed with an input sequence. GPT-2 has previously been used for various text generation tasks, including dialogue systems and chatbots [Budzianowski and Vulic, 2019]. Therefore, it is a suitable choice for our task of simulating users through generating answers to clarifying question in a conversational search system.

We base our proposed user simulator USi on the GPT-2 model with language modeling and classification losses, i.e., DoubleHead GPT-2. In this variant, the model not only learns to generate the appropriate sequence through the language modeling loss, but also how to distinguish a correct answer to the distractor one. This has been shown to improve the sequence generation [Radford et al., 2019] and has showed superior performance over only-language loss GPT-2 in the initial stage of experiments. The two losses are linearly combined.

Single-turn Responses

We formulate the input to the GPT-2 model, based on Equation 6.2, as:

$$input_seq = IN[SEP]u_q[SEP]s_{cq}[bos]u_a[eos] \quad (6.3)$$

where $[bos]$, $[eos]$, and $[SEP]$ are special tokens indicating the beginning of sequence, the end of sequence, and a separation token, respectively. Information need IN , initial query u_q , clarifying question s_{cq} , and a target answer u_a are tokenized prior to constructing the full input sequence to the model. Additionally, we construct segment embeddings, which indicate different segments of the input sequence, namely IN , u_q , s_{cq} , and u_a .

When training the DoubleHead variation of the model, we formulate the first part of the input as described above. Additionally, we sample the ClariQ dataset for distractor answers and process them in the same manner as the original answer, based on Equation 6.3. Therefore, the DoubleHead GPT-2 variant accepts as input two sequences, one with the original target answer in the end, and the other with the distractor answer. It then needs to not only learn to model the target answer, but also to distinguish between original and distractor answers

and provide a binary label indicating which of the two answers is the desirable one. We sample the distractor answers from the aforementioned datasets. When possible, we ensure that if the target answer starts with “Yes”, the distractor answers starts with “No”, in order to enforce the connection between the answer, the clarifying question, and the information need. Likewise, if the answer starts with “No”, we sample a distractor answer that starts with “Yes”. Note that USi does not generate answers that begin strictly with a “yes” or a “no”.

Conversation History-aware Model

The conversation history-aware model calls for a different formulation of the input and the training. Specifically, the input to history-aware GPT-2 is constructed as:

$$input_seq = IN[user]u_q[system]s_{cq}^{<t}[user]u_a^{<t}[system]s_{cq}^t[bos]u_a^t[EOS] \quad (6.4)$$

where $[user]$ and $[system]$ are additional special tokens indicating the conversational turns up to the current turn t between the (simulated) user and the conversational system, respectively.

Inference

During inference, we omit the answer u_a from the input sequence, as our goal is to generate this answer to a previously unseen question. In order to generate answers, we use a combination of state-of-the-art sampling techniques to generate a textual sequence from the trained model. Namely, we utilize temperature-controlled stochastic sampling with top- k [Fan et al., 2018] and top- p (nucleus) filtering [Holtzman et al., 2020]. After some initial experiments and consultation with previous work, we fix the parameters of the temperature to 0.7, k to 0, and p to 0.9.

6.3.3 Evaluation Details

We evaluate the feasibility of our approach based on the evaluation procedures outlined in Sect. 6.2. Below, we detail the specifics of these evaluation approaches.

Automated NLG Metrics

We first study the language generation ability of USi and compare it to several baselines. To this end, we perform an automated evaluation of generated utter-

ances through NLG metrics, as described in Sect. 6.2.2. Specifically, the models are trained on ClariQ training set and evaluated on unseen ClariQ development set. We evaluate on ClariQ’s development set since the test set does not contain question-answer pairs. We take a small portion of the training set for our actual development set. The answers generated by USi and the baselines are compared against oracle answers from ClariQ, generated by humans. Results of the automated evaluation are presented in Sect. 6.5.1.

Response Naturalness and Usefulness

We perform a crowdsourcing study to assess the *naturalness* and *usefulness* of generated answers to clarifying questions, as described in Sect. 6.2.3. We require the workers to be based in US and have at least 95% task approval rate. We annotate 230 answer pairs for *naturalness* and 230 answer pairs for *usefulness*, each judged by two crowdsource workers. With this study, we aim to shed light onto research questions RQ1 and RQ2, i.e., whether the generated answers are indeed natural and in line with the underlying information need, compared to human-generated answers. Additionally, we compare Transformer-seq2seq to USi. The results of the study are discussed in Sect. 6.5.1.

6.4 Beyond Clarifying Questions – Providing Feedback

In this section, we address the two main limitations of USi, described in the previous section. To this end, we propose a user simulator approach capable of multi-turn interactions extending beyond answering clarifying questions. Parts of this section were published in Owoicho et al. [2023].

Users can provide explicit feedback on the quality of system’s responses, as well as answer potential questions prompted by the system. Such feedback is beneficial to mixed-initiative conversational search systems and can provide valuable information on user’s needs. Moreover, feedback can have a great effect on how conversation is shaped by, e.g., giving the system the chance to recover from an initial failed attempt [Zou et al., 2023]. Despite its significance, lack of research in this area can be attributed to the difficulty of collecting appropriate data containing user feedback, as mentioned in Sect. 6.1.

To address the vicious circle composed of the lack of research on feedback utilization and the lack of appropriate data, we develop a comprehensive experimental framework based on simulated user–system interactions, as shown in Figure 6.1. The framework allows us to evaluate multiple state-of-the-art mixed-

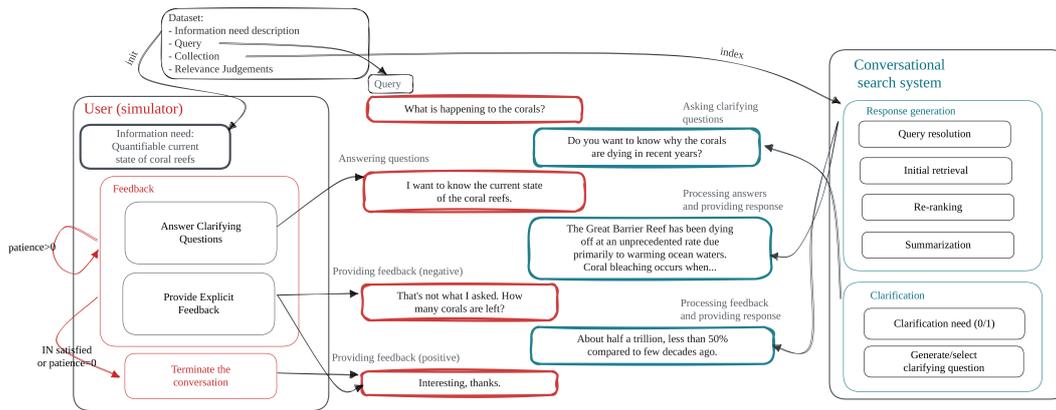


Figure 6.1. Experimental framework with an example interaction between a user simulator (left) and a mixed-initiative conversational search system (right). Functionalities and modules of both are highlighted.

initiative conversational search systems, addressing several challenges, including contextual query resolution, asking clarifying questions, and incorporating user feedback.

Existing work [Aliannejadi et al., 2021a] aims to study the effect of different mixed-initiative strategies on retrieval, however, their findings are limited to a single data collection, and lexical-based retrieval techniques. More recently, work on user simulators for conversational systems aims to address these limitations, however, it remains limited to pre-defined or templated interactions [Zhang and Balog, 2020, Salle et al., 2021] or focus only on one aspect of the search system, e.g., answering clarifying questions [Sekulić et al., 2022]. To address these limitations, we propose a user simulator called ConvSim, capable of multi-turn interactions with mixed-initiative conversational search systems. Given a textual description of the information need, ConvSim answers prompted clarifying questions and provides both positive and negative feedback, as necessary. Recent advancements in LLMs, e.g., GPT-3 [Brown et al., 2020], PALM [Chowdhery et al., 2023], open the possibilities of addressing such nuanced tasks. Thus, we base core functionalities of the proposed simulator on LLMs. Finally, the ConvSim addresses the limitation of pre-built corpora, as the simulator’s behavior adapts to the system’s response.

Our experimental evaluation shows that ConvSim can reliably be used for interacting with mixed-initiative conversational systems. Specifically, we demonstrate that responses generated by the simulator are natural, in line with defined information needs, and, unlike previous work [Sekulić et al., 2022], coher-

ent across multiple conversational turns. The proposed simulator interacts with conversational search systems entirely in natural language, without the need to access the system’s source code or inner mechanisms. Furthermore, the experimental framework, centered around ConvSim, allows for seamless curation of synthetic data on top of existing static IR benchmarks, as the simulator–system interactions can extend over multiple conversational turns.

6.4.1 In-Context Learning Approach

We propose ConvSim, a Conversational search Simulator, capable of multi-turn interactions with the search system in a conversational manner. We design ConvSim to satisfy the requirements defined in Sect. 6.1.2. As such, the simulator needs to encapsulate different behaviors across utterances of various discourse types, including querying u_q , as well as providing feedback u_a and u_f .

We conduct our simulator experiments within the framework of a conversational pipeline that encapsulates the commonly used components in a mixed-initiative conversational search pipeline: query rewriting, passage retrieval, passage re-ranking, clarifying question selection and generation, and response generation. The framework is depicted in Figure 6.1. It enables seamless multi-turn exchange of user simulator utterances u and system’s utterances s . The framework includes a suggested logical exchange of the utterances, i.e., when the system produces a response s_r , the simulator is tasked to provide feedback u_f . Likewise, when posed with a clarifying question s_{cq} the simulator needs to provide an answer u_a . Such interactions continues as long as simulator patience $p > 0$ and IN is not satisfied. Moreover, we design this framework to be flexible, allowing us to easily configure and (re)arrange the steps per our experimental needs. At the heart of this framework is a conversational turn representation that holds all relevant properties about a turn, such as a user query, system response, conversational context, and retrieved documents. This representation flows through every step in the conversational pipeline, each of which modifies the representation in a way that is useful for downstream retrieval and evaluation. We refer the reader to our codebase for the implementation details of this experimental framework.

Specifically, we initialize ConvSim with an information need description IN_t , specific to each turn. This ensures the responses generated by ConvSim are consistent with the user information need and guide the conversation towards the relevant information. Asking too many clarifying questions or providing unsatisfactory responses might impair user’s satisfaction with the search system [Zou et al., 2023]. Thus, a simulator should encapsulate similar behaviors. Following

Salle et al. [2021], we introduce the notion of *patience* $p \in \mathbb{Z}^{0+}$ — a parameter that indicates how many turns of feedback a simulated user willing to provide. Simulator decreases its patience p after each turn in which it has to provide feedback, terminating the conversation once $p = 0$. A conversation is stopped by the simulator either when IN is satisfied or when patience runs out.

We model feedback generation functions θ_1 and θ_2 detailed in Sect. 6.1.2 using LLMs. Given the focus of our experiments, we implement each of the simulator’s possible actions (clarifying question answering for θ_1 , feedback generation for θ_2) as steps in the conversational pipeline framework described below.

An example of a prompt is presented below:

Generate a response to the system question based on the conversation and information needs:

Examples:

Information need: History and evolution of bees.

User: Tell me something interesting about bees.

System: There are lots of exciting things to say about bees. Would you like to know its evolution, characteristics, or sociality?

User: No, tell me something interesting about the history of bees.

6.4.2 Implementation Details

In Owoicho et al. [2023], we build ConvSim on top of OpenAI’s Text-Davinci-003 [Brown et al., 2020] model using few-shot prompting. We stress the fact that other LLMs with similar capabilities can be used as well. We use OpenAI’s completions API endpoint with the following parameter settings based on the author’s guidelines [Brown et al., 2020] and initial empirical exploration:

- ***max_tokens***: 50. This parameter limits the generated response to 50 tokens. This prevents the model from generating overly long responses but is also sufficient enough for the model to generate clarifying questions in addition to negative feedback or to expand a bit on its answers to clarifying questions.
- ***temperature***: 0.5. Temperature scales the logits of the probability distribution used for sampling of the next token. This value is a halfway point between a very conservative, i.e., deterministic, and risky model, i.e., non-deterministic with randomness. While we want creative outputs, we also want the responses to be on topic.

- **frequency_penalty**: 0.2. This value is added to log-probability of a token each time it is generated. It discourages the model from generating previously generated tokens (i.e., repeating itself).
- **presence_penalty**: 0.5. This encourages the model to introduce new tokens by subtracting from the log-probability of a token each time it is generated. In the same way as the *temperature* parameter, this enables fairly novel responses that are always on topic.

For a given turn t , we prompt the model with a task description (i.e., whether to generate an answer to a clarifying question or feedback to system’s response), a description of the information need IN_t , sample transcripts between a user and a system with the desired behavior, and a transcript of the conversational history H between the user and system up to turn t . The exact prompts used can be found in our codebase. We do not explicitly implement the information seeking model $u_q = h(IN)$. Instead, we take the initial query u_q^t directly from the dataset to ensure fair comparisons between non-feedback and feedback utilizing methods described above.

6.4.3 Mixed-Initiative Systems

Compared Methods

We focus our investigations on the effects and ways of using simulated user feedback and answers to clarifying questions for downstream retrieval. In order to analyze the effects of feedback processing modules, we compare their performances against the following non-feedback baselines which do not use any initiative or simulation:

Organizer-auto is a competitive baseline used in the TREC CAsT shared task over the past two years. First, it reformulates the user query with a generative T5 query rewriter fine-tuned on the CANARD dataset³. As context, the rewriter takes in all previous turn queries and system responses as input: $u_q^{t'} = \gamma_3(u_q^t|H)$. No special considerations are made for cases where the input token length exceeds the model’s limit (i.e., 512 tokens). Next, it uses Pyserini’s⁴ BM25 implementation (k1=4.46, b=0.82) to retrieve the top 1 000 documents from the collection and re-ranks it’s constituent passages with a point-wise T5 passage ranker (MonoT5) [Nogueira et al., 2020] trained on MSMARCO [Bajaj et al.,

³<https://www.huggingface.co/castorini/t5-base-canard>

⁴<https://www.github.com/castorini/pyserini>

2016]. Finally, a BART model⁵ summarizes the top 3 passages to output a system response. We run **organizer-manual** on the CAsT benchmark using the manually reformulated queries at each turn for every conversation in the dataset. As these manual rewrites are context-free, this baseline represents an upper bound for retrieval performance without initiative or simulated responses using CAsT’s bag-of-words retrieval and neural ranking methods. We refer the reader to CAsT’21 and CAsT’22 overview papers for more on the implementation details of these baselines.

For incorporating user feedback, we compare against additional baselines built on top of the **organizer-auto** baseline. Formally, we model the following method with the function $u^{t'} = \gamma(u^t|H)$, described in Sect. 6.1.2, aimed at updating the system’s understanding of the user’s information need. However, as parts of the study go beyond the contributions of this dissertation, we point an interested reader to Owoicho et al. [2023] for more details.

Pipeline Components

We implement query rewriting and passage ranking methods to utilize feedback by adapting state-of-the-art systems as follows:

Passage Ranking. We modify the query input of the MonoT5 re-ranker by adding feedback text to it, while keeping the passage input as is. Specifically, we format the input to MonoT5 as follows:

Query [u_q] [u_f] Passage [r_i] Relevant:

where u_q , u_f , and r_i refer to the query, feedback, and passage texts, respectively. Based on empirical investigations, we find this to be more effective in a zero-shot setting than changing the input template to accommodate feedback or using the feedback text in place of the query. We use an automatically rewritten query $u_q^{t'}$ as input, as opposed to the raw, unresolved query. Further, input lengths are restricted to 512 tokens. We refer to our variant of MonoT5-based model as *FeedbackMonoT5*.

Query Rewriting. We use the baseline T5 query rewriter (T5-CQR) to reformulate the feedback utterance based on conversation context (including the user’s raw query). We observe that this makes the rewriter prone to ‘over-rewriting’, especially in the case of positive feedback. For example, ‘Thanks!’ may be rewritten

⁵<https://www.huggingface.co/facebook/bart-large-cnn>

to ‘What types of essential oils should I consider for a scented lotion?’, essentially repeating the user’s query, even after a positive feedback from the user. Given the lack of discourse-aware query rewriters, we examine the effects of mitigating this by also implementing an improved version of the rewriter that only reformulates negative feedback (Discourse-CQR). In both cases, as with the baseline system, the input text is automatically truncated where it exceeds the model’s limit of 512 tokens.

Additionally, we process the answers to clarifying questions following [Aliannejadi et al. \[2019b\]](#). Specifically, we append the answer and the asked clarifying question to the initial query: $u_q^{t'} = u_q^t + s_{cq}^t + u_a^t$. The reformulated utterance is then $u_q^{t'}$ fed through our baseline pipeline *organizer-auto*, without the first step of query rewriting.

Asking Clarifying Questions

We implement several established approaches to asking clarifying questions. While we acknowledge that not all utterances require clarification, as indicated by the *clarification_need* variable described in Sect. 6.1.2, we do not explicitly model it. The clarifying question is thus either not asked at all (*clarification_need* = 0) or asked at each turn (*clarification_need* = 1), depending on the experiment. We focus on both question selection and question generation, implementing the following baselines.

Question Selection. As detailed in Sect. 6.1.2, the aim of this group of models is to select an appropriate clarifying question utterance s_{cq}^t , given the user’s current utterance u_q^t . Therefore, we opt for two ranking-based methods. First, a BM25-based method, termed **SelectCQ-BM25**, which indexes the clarifying question pool CQ and performs retrieval with reformulated user utterance $u_q^{t'}$, specifically: $s_{cq}^t = \arg \max_i (BM25(cq_i | u_q^{t'}))$, $cq_i \in CQ$. A similar approach has been taken in previous works [[Aliannejadi et al., 2019b, 2020b](#)]. Second, a semantic matching-based method, termed **SelectCQ-MPNet**, utilizing MPNet [[Song et al., 2020](#)] to predict a score for each question cq_i from the pool: $s_{cq}^t = \arg \max_i (MPNet(cq_i | u_q^{t'}))$, $cq_i \in CQ$. A similar approach has been adapted for CAsT’22 [[Lajewska et al., 2022](#)]. In both cases, the clarifying question with the highest score is selected, as indicated by the *arg max* function.

Question Generation. We implement entity- and template-based clarifying question generation method, dubbed **GenerateCQ-Entity**. Template-based question

generation has been widely utilized in the research community due to its simplicity and effectiveness [Zamani et al., 2020a, Zhang and Balog, 2020, Sekulić et al., 2021]. With entities being central to the topic of a document, we opt to utilize SWAT [Ponza et al., 2019] to extract salient entities to generate clarifying questions. Specifically, we extract entities above a certain threshold ($\rho > 0.35$, as recommended by the authors) from the top n results in the ranked list. We then sort the entities by their saliency score in descending order, resulting in a list of entities $E = [e_1, e_2, \dots, e_M]$. Finally, the question is constructed by inserting up to m entities (m is set to 3) to the question template “Are you interested in e_1 , e_2 , or e_3 ?” Note that we alter the template according to the number of entities, in case E contains less than 3 entities.

6.4.4 Evaluation Details

We stress the fact that research questions around feedback utilization in conversational search can hardly be answered by existing or pre-built collections. On the other hand, while the questions around leveraging user feedback could be answered through comprehensive user studies, such studies are time-consuming, expensive, and largely limited in the number of experiments we would be able to conduct. Therefore, we not only evaluate our user simulator, but also show the benefit of feedback in multi-turn conversational search.

Mixed-Initiative Search Systems

We use the official measures and methodology from the CAsT benchmark for comparison. We report macro-averaged retrieval effectiveness of all systems at the turn level. We report NDCG@3 to focus on precision at the top ranks as well as standard IR evaluation measures (MAP, MRR, NDCG) to a depth of 1000 and at a relevance threshold of 2 for binary measures. Statistical significance is reported under the two-tailed t-test with the null hypothesis of equal performance. We reject the null hypothesis in favor of the alternative with p -value < 0.05 . We design the experimental framework with the goal of assessing the impact of various conversational search system components on retrieval performance. Specifically, we evaluate the base pipeline, described in Section 6.4.1 for passage retrieval with and without conversational search system components.

Naturalness and Usefulness of Generated Answers

We evaluate ConvSim in terms of naturalness and usefulness, as described in Sect. 6.2.3. To this end, we compare our method to the current state-of-the-art simulator for answering clarifying questions, *USi* [Sekulić et al., 2022], as well as human-generated responses. Following [Sekulić et al., 2022], we conduct a crowdsourcing-based evaluation on the ClariQ dataset [Aliannejadi et al., 2020b]. Specifically, two crowd workers annotate a pair of answers, where one is generated by ConvSim, and the other by *USi* or humans. We instruct them to evaluate the answers in terms *naturalness* and *usefulness*. In this pairwise setting, we count a win for a method if both crowd workers vote that the method’s answer is more natural (or useful), while if the two crowd workers do not agree, we count it a tie. For multi-turn evaluation, we utilize a multi-turn extension of the ClariQ dataset [Sekulić et al., 2022] with human-generated multi-turn conversations. We follow Li et al. [2019] and present full conversations for comparisons. We report statistical significance under the trinomial test [Bian et al., 2011], an alternative to the binomial and Sign tests that takes into account ties. The null hypothesis of equal performance is rejected in favor of the alternative with p -value < 0.05 . We present the results for both single- and multi-turn assessments.

As described in Sect. 6.2.3, we use the MTurk platform for our crowdsourcing-based experiments and take the described steps for ensuring the quality of annotations.

Explicit Feedback

We evaluate the feedback generation capabilities of ConvSim as described in Section 6.2.5. To this end, we generate responses for each turn in the CAsT’22 dataset with the Organizer-auto method, described in Section 6.4.3. Next, we utilize ConvSim to give feedback to the generated responses and manually annotate whether the generated feedback is positive or negative. We consider feedback positive if it is along the lines of “*Thank you, that was helpful.*” and negative if similar to “*That’s not what I asked for.*” We consider it as negative feedback if it includes a more detailed sub-question aimed at eliciting the missing component (e.g., “*Thanks, but what is its impact on climate change in developing countries?*”, since the information need is not entirely satisfied. We compare the system’s responses to the canonical responses present in CAsT to assess whether the information need is satisfied or not.

Table 6.3. Performance of different answer generation methods, measured by automated NLG metrics on the ClariQ development set.

model	BLEU-1	BLEU-2	BLEU-3	ROUGE_L	SkipThoughtCS	EmbeddingAvgCS
LSTM-seq2seq	0.1989	0.1401	0.0988	0.2210	0.3158	0.7012
Transformer-seq2seq	0.2041	0.1352	0.0936	0.2067	0.3666	0.7077
USi [Sekulić et al., 2022]	0.3029	0.2404	0.2054	0.2359	0.4025	0.7322
ConvSim [Owoicho et al., 2023]	0.1949	0.1394	0.1014	0.1898	0.3911	0.6766

6.5 Results and Analysis

Answering CQs, providing feedback, qualitative analysis

In this section, following evaluation methodology presented in Sect. 6.2, we evaluate our two proposed user simulators: USi [Sekulić et al., 2022] and ConvSim [Owoicho et al., 2023]. First, we present the results of their performance for the task of answering clarifying questions through automated natural language generation metrics and human annotation of answer usefulness and naturalness. Next, we present and discuss ConvSim’s performance in the task of providing explicit feedback. Moreover, we evaluate the simulators’ impact on passage retrieval performance, with a focus on the impact of multi-turn interactions, showcasing the benefit of feedback in conversational search.

6.5.1 Answering Clarifying Questions

Automated NLG metrics

Performance of the baseline models and our simulated user models, as evaluated by automated NLG metrics described in Section 6.3.3, is presented in Table 6.3. USi significantly outperforms all baselines by all computed metrics on the ClariQ data. Even though LSTM-seq2seq showed strong performance in various sequence-to-sequence tasks, such as translation [Sutskever et al., 2014] and dialogue generation [Shao et al., 2017], it performs relatively poorly on our task. A similar outcome is observed for Transformer-seq2seq. We hypothesize that the poor performance in this task is due to limited training data, as the success of these seq2seq models on various tasks was conditioned on large training sets. Our GPT2-based model does not suffer from the same problem, as it has been pre-trained on a large body of text, making the fine-tuning enough to capture the essence of the task, which is generating answers to clarifying questions.

An interesting observation is the fact that GPT3-based model, ConvSim [Owoicho et al., 2023] performs worse than the GPT2-based USi. We attribute this

Table 6.4. Results on naturalness and usefulness of responses, USi [Sekulić et al., 2022] vs Transformer-seq2seq baseline [Sekulić et al., 2022] vs human-generated answers [Aliannejadi et al., 2020b]. The † sign indicates statistically significant difference under the trinomial test with p -value < 0.05 .

	USi	Baseline	Ties	USi	Human	Ties
<i>Naturalness</i>	50%†	3%	47%	17%	38%	45%
<i>Usefulness</i>	66%†	3%	31%	22%	27%	51%

result to the aforementioned issues with unreliability of the automated NLG metrics. As such, they capture solely exact matching of the wordings of the generated answer and the gold answers, largely failing to adjust to differences in vocabularies between the two answers, although they might be conveying the same message. Thus, in the next section, we report a more reliable crowdsourcing-based annotations of the generated answers.

Naturalness and Usefulness

USi Evaluation

Table 6.4 presents the results of the crowdsourcing study on *usefulness* and *naturalness*, comparing answers generated by USi and human, as described in Section 6.3.3. Both in terms of *naturalness* and *usefulness*, we observe a large number of *ties*, i.e., the two workers annotating the answer pair did not agree on which one is more natural/beneficial. Since we are comparing answers generated by our GPT2-based simulated user with the answers written entirely by humans, this result favors our proposed model. Moreover, the difference between *losses* and *wins* for our model is relatively tiny (38% losses, 17% wins) for *naturalness* and even smaller in terms of *usefulness* (32% losses, 23% wins). We conduct a trinomial test for statistical significance [Bian et al., 2011], an alternative to the Sign and binomial tests that consider ties.⁶ Regarding *naturalness*, we reject the null hypothesis of equal performance with $p < 0.05$, i.e., human-generated answers are more natural than those generated by USi. Nonetheless, 45% of ties between USi- and human-generated answers suggests the high quality of the generated text. Regarding *usefulness*, we accept the null hypothesis of equal performance with $p = 0.43$, i.e., there is no statistically significant difference between the

⁶Another point-of-view would be to test for equivalent effectiveness Jayasinghe et al. [2015]. However, we refrain from it since it does not consider ties.

performance of human annotators and USi.

Table 6.4 presents the comparison results between the Transformer-seq2seq and USi. We observe a win of the proposed USi over the baseline by a large margin. Our GPT-2-based model significantly outperforms the baseline ($p < 0.05$) both in terms of *naturalness* (50% wins and 3% losses) and *usefulness* (66% wins and 3% losses). This finding is in line with the automated evaluation of generated answers.

Regarding the research questions RQ1 and RQ2, i.e., whether the responses generated by our model align with the underlying information need and, at the same time, coherent and fluent, we arrive at the satisfactory performance of the simulated user. The generated answers to clarifying questions seem to be able to compete with the solutions produced by humans both in terms of *naturalness* and *usefulness*. Moreover, the strong performance of USi over Transformer-seq2seq additionally motivates the use of large-scale pre-trained language models, such as GPT-2, for the task. These results make a strong case for using a user simulator for mixed-initiative conversational search system evaluation.

ConvSim Evaluation

Table 6.5 presents the results in comparison to USi [Sekulić et al., 2022] and human-generated answers to clarifying questions in single- and multi-turn scenarios. We make several observations from the results. First, ConvSim significantly outperforms USi both in terms of naturalness and usefulness in both single- and multi-turn settings. Second, the difference between the performance of ConvSim and USi is especially evident in the multi-turn setting, which is one of USi’s potential limitations indicated by the authors [Sekulić et al., 2022]. The difference is even greater in multi-turn usefulness assessments, which can be attributed to USi’s hallucinations, and thus not staying on topic. Finally, ConvSim in most cases does not significantly outperform human-generated answers, except in single-turn usefulness. Although further analysis is required, we suspect the difference to have come from ConvSim’s precision in answering clarifying questions, while crowd workers sometimes answer them reluctantly and concisely, with no notion of grammar and punctuality (e.g., “no”). The results indicate that ConvSim can be used to answer clarifying questions both in single- and multi-turn settings, outperforming state-of-the-art methods both in terms of naturalness and usefulness.

Table 6.5. Results of crowdsourcing study assessing naturalness and usefulness of generated answers to clarifying questions in single- and multi-turn scenarios. We compare answers generated by ConvSim [Owoicho et al., 2023], USi [Sekulić et al., 2022], and human-generated answers [Aliannejadi et al., 2020b]. Each value indicates the percentage of pairwise comparisons won by the specific model as well as ties. Sign † indicates a significant difference.

		ConvSim	USi [Sekulić et al., 2022]	Ties	ConvSim	Human	Ties
<i>Single</i>	Naturalness	37%†	22%	41%	36%	25%	39%
	Usefulness	44%†	19%	37%	36%†	20%	44%
<i>Multi</i>	Naturalness	45%†	18%	37%	25%	28%	47%
	Usefulness	62%†	12%	26%	26%	16%	58%

Table 6.6. Document retrieval performance based on the answers provided by our simulated user. Percentages in parentheses report relative increase or decrease in performance over Oracle. Symbols † and ‡ indicate statistically significant difference compared to the query-only baseline and the human-generated answers, respectively. The significance is reported under two-sided t-test with $p < 0.01$.

	nDCG@1	nDCG@5	nDCG@20	P@1	MRR@100
Query-only	0.1304 (-3%)	0.1043 (-21%)	0.0852 (-26%)	0.1764 (-4%)	0.2402 (-12%)
LSTM-seq2seq	0.1018‡ (-24%)	0.0899‡ (-31%)	0.0745‡ (-35%)	0.1409‡ (-23%)	0.2131‡ (-22%)
Transformer-seq2seq	0.1124 (-16%)	0.1040‡ (-21%)	0.0847‡ (-26%)	0.1559‡ (-15%)	0.2309‡ (-15%)
USi	0.1355 (+1%)	0.1289† (-2%)	0.1133† (-2%)	0.1862 (+1%)	0.2730† (+0%)
Human (Oracle)	0.1343	0.1312†	0.1154†	0.1839	0.2725†

Document retrieval performance

The comparison of our simulated user and the baselines on the document retrieval performance, before and after answering a clarifying question, is presented in Table 6.6. The first row of the table shows the performance of the BM25 with only the initial query as input. The following rows report the performance of BM25 with input composed of the initial query, clarifying question, and answers generated by each of the models. Answers in the last row of the table are generated by humans and taken directly from ClariQ.

We observe that neither of the baseline models improves over the *query-only* baseline, i.e., we can not reject the null hypothesis of equal performance for neither of the metrics. This suggest that LSTM-seq2seq and Transformer-seq2seq do not yield useful and relevant answers to posed clarifying questions. In most

of the cases, the answers generated by the baselines even have a negative effect on the document retrieval performance, suggesting that the answers confuse the retrieval model.

On the other hand, we observe clear and significant performance increase of our simulated user over both the *query-only* and *seq2seq* baselines. The performance is significant by all of the metrics, except *nDCG@1* and *precision@1*. Similarly, oracle answers significantly outperform both baselines. This confirms the finding previous research, suggesting the document retrieval performance increases with answers to clarifying questions as input [Aliannejadi et al., 2019b].

Interestingly, human-generated answers do not perform better than the answers generated by our model. This finding provides an answer to the research question RQ3, i.e, can the retrieval model benefit from the answers generated by USi. The equal performance of the generated and human answers on this task is in line with previously described analyzes on the *usefulness* of the generated responses. Together, the studies strongly support the possibility of substituting a real user with the user simulator for answering clarifying questions in conversational search.

The results presented in this section do not include ConvSim’s performance, as ConvSim was built on top of TREC CAsT’22 and utilized both for its answers to clarifying questions and its explicit feedback. Therefore, the impact of the utterances generated by ConvSim on retrieval performance is presented in the next section.

6.5.2 Exploiting Simulated Feedback

In this section, we present the results of conversational search systems with varying core components on the passage retrieval task. Specifically, we modify different components of the system, including explicit relevance feedback processing, ranking, and generating clarifying questions. Moreover, we assess the impact of multi-turn simulator–system interactions on performance. Generally, the results demonstrate improvements of feedback-aware methods over the baselines.

In this section, we briefly present the main findings of the study. However, parts of the results extend beyond the individual contributions of this dissertation. Thus, we encourage an interested reader to Owoicho et al. [2023] for a detailed evaluation and analysis of different feedback-aware methods, including a variety of query rewriting methods that incorporate feedback and feedback-aware rerankers.

Table 6.7. Performance after asking a clarifying question constructed by various methods, compared to the baseline.

Method	R	MAP	MRR	nDCG	nDCG@3
organizer-auto	0.348	0.154	0.532	0.311	0.365
+ SelectCQ-BM25	0.433†	0.166	0.625	0.364†	0.411
+ SelectCQ-MPNet	0.413†	0.173†	0.631	0.362	0.409
+ GenerateCQ-Entity	0.409	0.162	0.577	0.348	0.398

Clarification and answer processing Table 6.7 shows performance of three clarifying question construction methods, described in Section 6.4.3. We observe an overall increase in effectiveness across all methods, with *SelectCQ-BM25* and *SelectCQ-MPNet* significantly outperforming the baseline across several metrics. Most gains in performance are in recall, as the original query is expanded by the answer and clarifying question providing additional information to the initial retriever. *GenerateCQ-Entity* does not perform as well as selection-based methods. We attribute this finding to potentially off-topic clarifying questions, as the entities extracted were not necessarily geared towards elucidating user’s need. ConvSim might have responded along the lines of “I don’t know.” or “No thanks.”, thus not helping elucidate the underlying information need.

Generated feedback evaluation Table 6.8 shows the performances of *Organizer-auto* model on CAsT’22 queries broken down by whether feedback given to the system’s response is positive or negative, as described in Section 6.2.5. Results show significant differences between responses with positive and negative feedback. Feedback on the system’s responses generated by ConvSim is useful, as the responses receiving negative feedback correspond to the poor retrieval effectiveness. On the contrary, when the system’s response satisfies the given information need, as demonstrated by higher retrieval performance, the simulator’s feedback is positive. ConvSim is not aware of the system’s retrieval effectiveness and provides feedback solely on the generated response and *IN* description.

6.5.3 Discussion and Analysis

Effect of iterative feedback

We investigate the potential for multiple rounds of feedback in a simulated environment. We run the *organiser-auto+Discourse-CQR* system with *Feedback-*

Table 6.8. Performance on turns where feedback is negative vs. turns where feedback is positive. The “Perc.” column indicates the percentage of such turns in the CAsT’22 dataset. All the differences are significant.

Feedback	Perc.	R	MAP	MRR	nDCG	nDCG@3
Negative	49%	0.073	0.039	0.399	0.091	0.161
Positive	51%	0.185	0.128	0.739	0.239	0.449

MonoT5 passage ranker for 10 rounds of feedback. For efficiency we only apply re-ranking to the first 100 passages retrieved. Figure 6.2 shows consistent improvements in terms nDCG@3 over the *organizer-auto* (round 0) baseline, with slight dips and plateaus between rounds 3 to 5 and rounds 6 to 8. At rounds 6 and above both MRR and nDCG@3 of this system exceed those of the *organizer-manual* system. Recall and MAP at round 8 come within 0.004 and 0.003 points of the manual run, respectively, further highlighting the utility of explicit feedback. Prompting the user for up to 8 or more rounds of feedback is not realistic and motivates the need for more effective feedback models that can learn from fewer rounds of feedback.

Combining clarification and explicit feedback

We analyze the effectiveness of *FeedbackMonoT5* for processing answers to questions selected with *SelectCQ-BM25*. The results suggest an improvement over the *organizer-auto* baseline (nDCG@3 = 0.392; +7% relative improvement), suggesting that *FeedbackMonoT5* can be used for processing answers to clarifying questions. We experiment with a round of clarification and a round of feedback and observe significant boost in Recall (0.448; +29% vs the baseline), but a relatively low improvement in terms of nDCG@3 (0.389; +6%). We hypothesize that both rounds of feedback result in well-defined information need, thus boosting the Recall, but query reformulation methods (i.e., *FeedbackMonoT5*) fail to resolve the complex context, leading to poor re-ranking performance.

6.6 Qualitative Analysis

In this section, we conduct an in-depth analysis of the utterances generated by the aforementioned LLM-based user simulators. The first user simulator, *USi*, is based in fine-tuned GPT-2 [Radford et al., 2019] and able to generate answers to

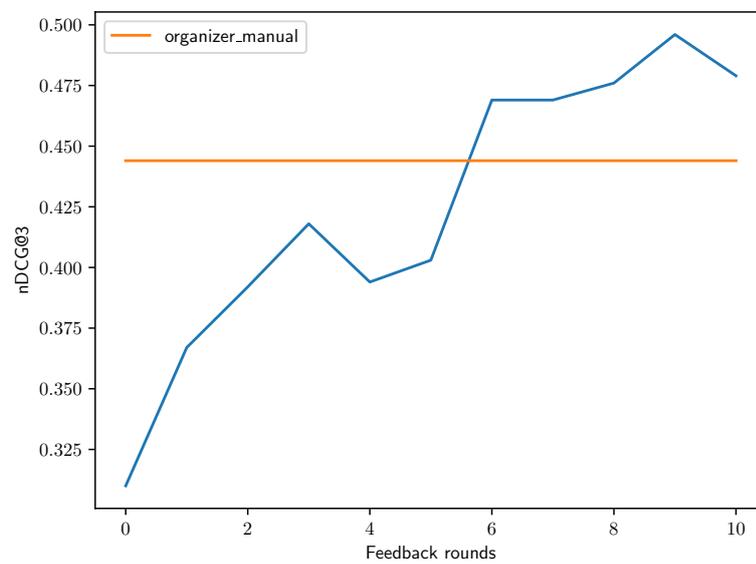


Figure 6.2. Multiple rounds of feedback using the organiser-auto+Discourse-CQR+FeedbackMonoT5 system. The orange line depicts the performance of organizer_manual.

Table 6.9. Qualitative analysis of answers generated by user simulator USi.

Information need	Clarifying question	Generated answer
How do you treat severe angular cheilitis	are you looking for the definition of angular cheilitis?	no i want to know about treatment
What home remedies are there for angular cheilitis	are you looking for the definition of angular cheilitis?	no i am looking for home remedies
Find a list of hobby stores in Maryland	is this for a school craft or play?	no i want to find a list of hobby stores in maryland
What hobby stores carry trains	what location do you want to find a hobby store in?	i want to know where i can purchase trains
What hobby stores carry trains	is this a hobby for an adult?	yes

posed clarifying questions. This method was depicted in Sect. 6.3 and presented in our recent paper [Sekulić et al., 2022]. Second, we use in-context learning approach, a technique made possible with the next generation of LLMs, such as GPT-3 [Brown et al., 2020], LLaMa [Touvron et al., 2023], and Chinchilla [Hoffmann et al., 2022]. A GPT-3 based method, *ConvSim*, was recently proposed in Owoicho et al. [2023] and described in detail in Sect. 6.4.

Both methods generate answers to clarifying questions in line with the initial information needed, simulating part of the behavior of a real user. In the first case, we ensure that through a specific training procedure, resulting in a semantically-controlled language model. With a GPT3-based simulator, we utilize prompting to guide the model into following specific steps to answer started questions. While, as shown earlier in the chapter, *ConvSim* is capable of generating feedback extending beyond answering clarifying questions, in this analysis, we focus on such answers in order to properly compare the two simulation methods.

Therefore, we perform a qualitative analysis of utterance reformulations generated by our LLM-based approaches in response to clarifying questions. We map our findings to recently proposed patterns for conversational recommender systems [Zhang et al., 2022] and find that user simulators tend to rewrite the original query to further explain the underlying information need. However, we note that types of such reformulations highly depend on the training data and the prompts given to the models.

6.6.1 Examples

Single-Turn Analysis of USi

In this section, we analyze several conversation samples of our user simulator with a hypothetical conversational search system. Table 6.9 shows four interaction examples. The user simulator USi is initialized with the information that needs description text. Given an initial query (omitted in the table for space), the conversational search system asks a clarifying question to elucidate USi’s intent. Then, USi generates the answer to the prompted question. The information needed, and the questions for these examples are taken from the ClariQ development set. Most TREC-style datasets contain the information need (facet/topic) description alongside the initial query. Thus, our simulated user can help evaluate conversational search systems on any of such datasets, as it only requires a description for initialization. Then, the system we aim to evaluate can produce clarifying questions and receive answers from USi.

First two examples in Table 6.9 initialize USi with different information needs. However, given the same initial query “How to cure angular cheilitis” and the same prompted clarifying question, USi answers differently, in line with the basic information needed for each case. In the table’s last three rows, we have different information needs for one broad topic of hobby stores. Given the initial query “I’m looking for information on hobby stores”, USi again answers questions in line with the underlying information need. We notice that the text produced by our GPT-2-based user simulator is coherent and fluent and, in the given examples, indeed in line with the underlying information need. Moreover, USi is not bound by answering the question in a “yes” or “no” fashion. Instead, it can produce various answers and even express its uncertainty (e.g., “I don’t know”).

Multi-Turn Analysis of USi

We perform an initial case study on the multi-turn variant of USi. While the initial analysis of multi-turn conversations suggests that *usefulness* and *naturalness* of single-turn interactions transfer into a multi-turn setting, additional evaluation is needed to support that claim strongly. Thus, future work includes a pair-wise comparison of multi-turn conversations inspired by ACUTE-Eval [Li et al., 2019].

Moreover, we aim to observe user simulator behavior in unexpected, edge-case scenarios. For example, initial analysis of the created multi-turn dataset showed that humans tend to repeat their previous answers when the clarifying question is off-topic or repeated. Similarly, our multi-turn USi has been observed to generate answers such as “I already told you what I’m looking for”

when prompted with a repeated question. However, such edge cases need to be clarified for the multi-turn model, which leads to a higher presence of hallucination than in the single-turn variation. This means that the user simulator drifts off the topic of the conversation and starts generating answers outside the basic information needed. This effect is well-documented in recent literature on text generation [Dziri et al., 2021] and should be approached carefully. Although edge cases are also present in the acquired dataset, the GPT2-based model needs additional mechanisms to simulate the behavior of users in such cases. We leave a deeper analysis of the topic for future research.

6.6.2 Response Patterns

In this section, we analyze human- and simulator-generated answers to posed clarifying questions. Specifically, we conduct expert annotation to identify patterns in the given answers, grounding our findings in prior work. To this end, we analyze the answers in light of patterns identified by Krasakis et al. [2020], focusing on the Qulac dataset [Aliannejadi et al., 2019b]. Krasakis et al. [2020] find that users' answers vary in polarity in length. For example, the user can answer with a negative short answer, such as "No.", but also potentially provide a longer answer, e.g., "No, I'm looking for X instead". Naturally, the answer can also be positive polarity depending on the information needed and prompted clarifying questions. Furthermore, we compare the generated answers to patterns identified by Zhang et al. [2022]. Although Zhang et al. [2022] focuses on query reformulations in conversational recommender systems, we find the overlap of the findings high. Thus, we map their proposed query reformulation types to answers in a mixed-initiative conversational search. Finally, we analyze answers to faulty clarifying questions proposed by Sekulić et al. [2022].

Reformulation Types

We analyze answers to prompted clarifying questions in light of previously identified utterance reformulation types [Zhang et al., 2022]. In other words, we map and expand the existing utterance reformulation ontology for conversational recommender systems to answer formulations in conversational search. While specific differences exist between recommender and search systems, our initial analysis suggested that the common conversational setting incites similar user behaviors. In their study, Zhang et al. [2022] analyze how users reformulate their utterances in subsequent turns, given a prompt from the conversational recommendation agent about its lack of understanding of user's needs. Simi-

larly, in conversational search, we have the user’s initial query, clarifying question prompted by the search system, and the user’s answer. Thus, we analyze these answers through the lens of reformulations from the user’s initial query.

Specifically, [Zhang et al. \[2022\]](#) identify seven utterance reformulation behaviors: 1. *start/restart* – users start to present their need; 2. *repeat* – user repeats previous utterance without significant change; 3. *repeat/rephrase* – user repeats last turn with different wording; 4. *repeat/simplify* – user repeats the word with a more straightforward expression, reducing complexity; 5. *clarify/refine* – user clarifies or refines their expression of an information need; 6. *change* – user changes their information need (topic shift); 7. *stop* – user ends the search session. We encourage an interested reader to refer to [Zhang et al. \[2022\]](#) for a more elaborate explanation of the reformulation types. In our analysis, we focus specifically on answers to clarifying questions. Thus, some user utterances must be observed and not discussed in other sections. Specifically, mainly by the design of our research setting, described in Section 7.1, we do not deal with utterance types (1) *start/restart*, (6) *change*, nor (7) *stop*. However, we add two additional categories, mostly to deal with edge cases: (8) *hallucination* – when the provided answer is not in line with the underlying information need; (9) *short answer* – when the answer is just “no” or “yes”. Examples of the observed utterance types are presented in Table 6.10.

Responses to Faulty Clarifying Questions

In order to gain further insight into designing a reliable user simulator for conversational search evaluation, we must adapt it to be resilient to unexpected system responses. For example, if a conversational search system responds with an off-topic clarifying question or an unrelated passage, the simulated user needs to react in a natural, human-like manner. However, to design such a simulator, we first need to learn how real users would react to incorrect responses from the search system. To this end, we acquired a dataset of human responses when prompted with faulty clarifying questions. The published dataset is multi-turn and can thus be used to improve our multi-turn user simulator model.

Examples from the acquired dataset are presented in Table 6.2. The dataset contains several scenarios where a conversational search system asks follow-up clarifying questions. We acquired a dataset of 1000 conversations, with crowd workers assuming the user role and responding to clarifying questions. Initial analysis of the crowd workers’ answers offers several insights. In case of appropriate clarifying questions (*Natural*), users tend to respond naturally by refining their information needs, as expected. However, in case of faulty clarifying ques-

Table 6.10. Identified reformulation patterns in responses generated by our proposed user simulator(s). The patterns are extended from the list proposed by Zhang et al. [2022].

	Explanation	Example
Repeat	Repeat the last utterance without significant change.	U: What are treatments for carotid cavernous fistula? S: Would you like to search for a doctor for a carotid cavernous fistula treatment? U: No, I want to know what treatments are available for carotid cavernous fistula.
Repeat/rephrase	Repeat the last utterance by using different wording.	U: I want to learn about rock art. S: Do you want pictures of rock art? U: Yes, I want information on rock art.
Repeat/simplify	Repeat the last utterance in a simpler language.	U: Find me information about the sales tax in Illinois. S: Do you want to know what the Illinois state tax rate is? U: Yes, I want to know about the taxes in the state.
Clarify/refine	Refine the information need express in last utterance.	IN: Find the homepage for the NASA Voyager mission. U: Tell me about Voyager. S: Are you interested in where the Voyager has been? U: No, I am interested in the website for the mission.
Other	Short response, none above applicable.	U: "all men are created equal" S: Are you looking for declaration of independence at the national archives? U: No.
Hallucination	Response not grounded in information need.	IN: How is workplace diversity achieved and managed? U: Tell me about diversity. S: Would you like the legal definition of diversity? U: Yes, I would like to know about the definition of diversity.

tions (*repeat*, *off-topic*, or *similar*), users either repeat their previous answer (20% of analyzed answers), expand their last reply with more details on their information need (23%), or rephrase previous answer with different wording (37%). Next, we aim to evaluate the resilience of our proposed USi to such faulty questions by analyzing its correspondence to human-generated answers.

Prevalence of Reformulation Types

Table 6.11 shows the prevalence of aforementioned types of utterance reformulations on ClariQ development set. We expertly annotated 150 answers generated by both generative approaches, as well as human answers taken directly from the ClariQ dataset. As indicated in Table 6.11, USi hallucinates in 7% of analyzed cases. The hallucination accounted for in the table is limited to cases when a long answer is generated. However, we observed that USi often needs a better short answer. For example, with an information need related to finding the list

Table 6.11. Prevalence of utterance reformulation types for answers to clarifying questions for human-generated answers [Aliannejadi et al., 2020b], answers generated by USi [Sekulić et al., 2022] and by ConvSim [Owoicho et al., 2023].

	Human	USi	ConvSim
Repeat	2%	0%	3%
Repeat/rephrase	4%	7%	6%
Repeat/simplify	4%	8%	5%
Clarify/refine	63%	37%	83%
Other	25%	40%	3%
Hallucination	2%	7%	0%

of dinosaurs with pictures, when prompted with a clarifying question “Are you looking for pictures of dinosaurs?”, USi answers “No”. Such short answers are mapped under *Other* in Table 6.11, as the focus of the analysis was to capture the extent of the short-vs-long answers. Moreover, we observe the hallucination phenomena in several answers taken from ClariQ, constructed by crowd workers. We attribute this to potentially swift manner the answers were written in, rather than to crowd workers not understanding that their answer isn’t in line with the given information need. On the contrary, the prompt- and GPT3-based *ConvSim* method does not suffer from the mentioned issue.

Moreover, the prevalence of different utterance reformulations differs between human-generated answers, and the answers generated by USi and *ConvSim*. Specifically, we observe a greater frequency of short answers (e.g., “yes”, “no”) in answers generated by GPT2-based USi. On the other hand, GPT3-based *ConvSim* tends to refine and clarify the given information need in the majority of the cases. While both long and short types of answers to clarifying questions are acceptable, as long as they are in line with the information need, certain users have slight preference towards the one or the other. Thus, as discussed in the last section, as a step towards more realistic user simulators, we aim to model users according to their cooperativeness level. In other words, the simulator would be able to generate either concise or long and elaborate answers, depending on the cooperativeness parameter for a specific underlying user model.

Chapter 7

User Simulation for Task-Oriented Dialogue Systems

The field of dialogue systems has seen a notable surge in the utilization of user simulation approaches, primarily for the evaluation and enhancement of conversational search systems [Owoicho et al., 2023] and task-oriented dialogue (TOD) systems [Terragni et al., 2023]. User simulation plays a pivotal role in replicating the nuanced interactions of real users with these systems, enabling a wide range of applications such as synthetic data augmentation, error detection, and evaluation [Wan et al., 2022, Sekulić et al., 2022, Li et al., 2022, Balog and Zhai, 2023].

The significance of user simulation in the development and evaluation of dialogue systems is undeniable. However, the prevailing methodologies often rely on rudimentary rule- and template-based approaches, which can limit their adaptability and effectiveness [Schatzmann et al., 2007, Schatzmann and Young, 2009]. Furthermore, certain user simulation methods require a substantial amount of annotated data [Lin et al., 2021a, 2022, 2023], or a deep understanding of the internal workings of the dialogue system they interact with [Schatzmann et al., 2007, Li et al., 2016].

The recent rise of generative capabilities of LLMs enabled user simulators to generate contextually appropriate responses in natural language, without the need for predefined rules [Terragni et al., 2023, Davidson et al., 2023]. This shift offers distinct advantages over traditional approaches: i) no human effort is needed to construct the rules; ii) it introduces lexical diversity into utterance generation to assess the robustness of downstream natural language understanding and enables testing of system’s robustness to different dialogue paths. However, LLMs are susceptible to hallucinations [Ji et al., 2023, Terragni et al., 2023],

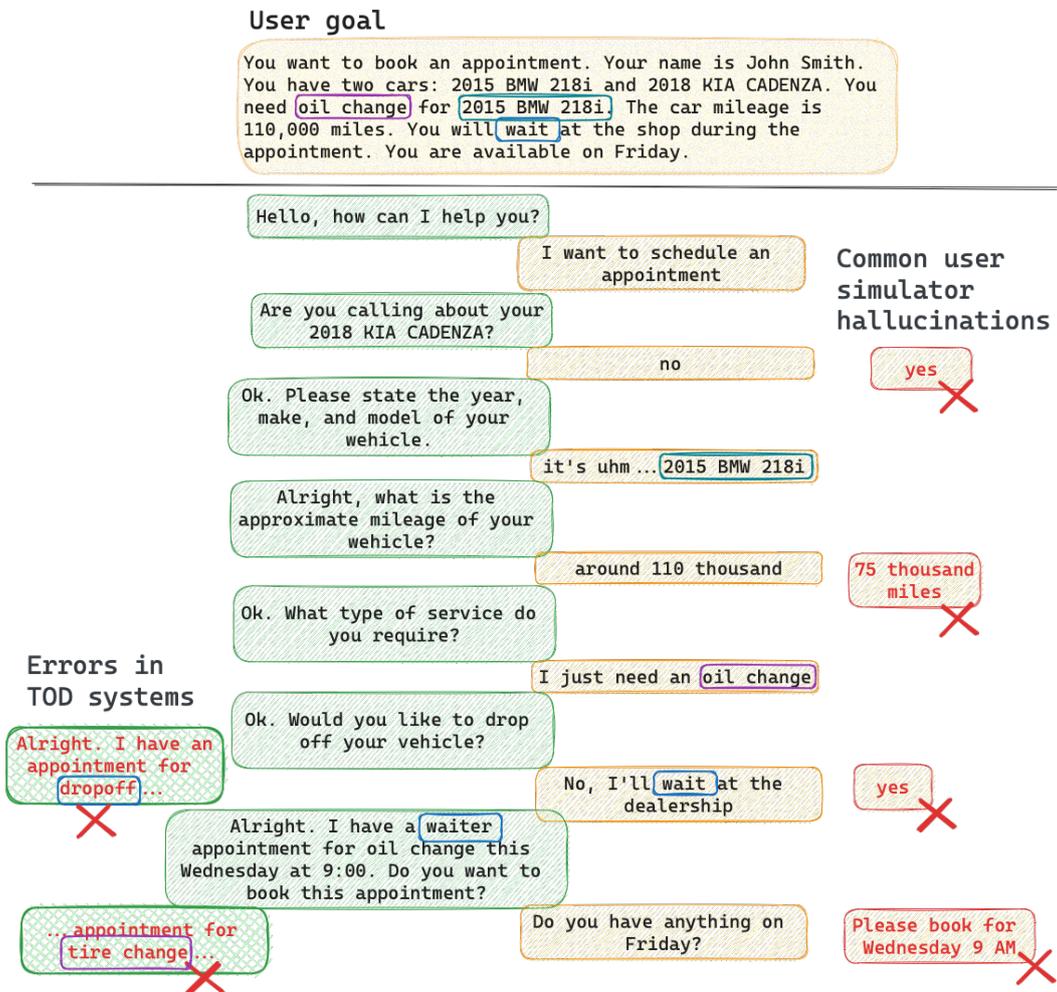


Figure 7.1. Example conversation between user simulator and TOD system. We aim to minimize common simulator's hallucinations (right) and thus ease the detection of TOD system failures (left).

resulting in inconsistency across dialogue turns or the generation of information irrelevant to the user’s goal.

In this chapter, we introduce DAUS, a generative user simulator for TOD systems. As depicted in Figure 7.1, DAUS, once initialized with the user goal description, engages with the system across multiple turns, providing information to fulfill the user’s objectives. Our aim is to minimize the commonly observed user simulator hallucinations (right-hand side of Fig. 7.1), with an ultimate objective of enabling detection of common errors in TOD systems (left-hand side of Fig. 7.1). Our approach is straightforward yet effective: we build upon the foundation of LLM-based simulators [Terragni et al., 2023, Owoicho et al., 2023] and extend such approach by fine-tuning the LLM on in-domain dialogues, annotated with their user goals. Notably, DAUS does not require insights into the inner-workings of the TOD system, its policy, nor system-specific functionalities, as it interacts with the TOD system strictly through natural language.

Our contributions and findings can be summarized as follows:

- **Domain-Specific Adaptation:** DAUS fine-tunes a pre-trained LLM on domain-specific conversational data, enhancing the simulator’s ability to maintain coherent and contextually relevant dialogues in a specific domain.
- **Reducing Simulator Hallucinations:** DAUS mitigates hallucinations originated from in-context learning approaches, causing inconsistencies and irrelevant information in simulator responses. By fine-tuning on domain-specific data, our approach ensures more coherent and contextually relevant simulated dialogues.
- **Balancing Lexical Diversity in User Simulation:** DAUS employs LLMs for user simulation, offering a degree of lexical diversity in generated utterances. While not matching the diversity of in-context learning (partly due to hallucinations), it still provides language variety.

7.1 Generative User Simulator

In this section, we define the task of generative user simulation for TOD systems. Moreover, we describe our approach, based on fine-tuned LLMs.

7.1.1 Background

When interacting with a TOD system, users aim to fulfill their goal, e.g., book a flight, or cancel their reservation in a restaurant. Therefore, a user simulator

(U), designed to imitate a real user, interacts with the TOD system (S) with a given user goal \mathcal{G} . Formally, interactions are a sequence of utterances, where the system’s utterances s and the user’s utterances u take turns, forming a dialogue history $\mathcal{H} = [s_1, u_1, \dots, s_t, u_t, \dots, s_N, u_N]$, with s_t and u_t corresponding to system’s and user’s utterance at turn t , respectively, and N being the total number of exchanged utterances.

We define the user goal \mathcal{G} as all the information the user requires to achieve their aim. An example of user goal is the following: *You want to try an Indian restaurant. The restaurant must be cheap and in the center. Book a table for 2 people at 8PM.* At the end of the dialogue, we expect the user simulator to have fulfilled \mathcal{G} . While the goal \mathcal{G} can be represented either in structural format [Davidson et al., 2023] or in natural language [Terragni et al., 2023], in this work we focus on \mathcal{G} represented in natural language. \mathcal{G} is usually defined by a domain expert or randomly sampled.

7.1.2 Our Approach

We propose Domain-Aware User Simulator (DAUS), a model that relies on learning the specifics of interactions with a TOD system from conversational data. The data needs to contain the goal \mathcal{G} and the dialogue history \mathcal{H} . Typically, such datasets are derived from user conversations with production TOD systems, or created and curated through crowd-sourcing or user studies.

We cast the above-described problem of simulator’s goal fulfillment to an utterance-level generation task. Specifically, the main task of U is to generate the next utterance u_t by modeling:

$$u_t = \phi(\mathcal{G}, \mathcal{H}) \quad (7.1)$$

where ϕ is the function to generate a user utterance. The u_t needs to be aligned with \mathcal{G} and \mathcal{H} , i.e., it needs to be faithful towards the given goal, as well as coherent with the dialogue so far.

Given that both \mathcal{G} and \mathcal{H} are in natural language, we model ϕ from Eq. 7.1 with a language modeling-based approach. Specifically, we first construct a prompt to feed an LLM, by combining \mathcal{G} and \mathcal{H} . We further employ the LLM to generate the u_t in auto-regressive fashion:

$$p_{LLM}(\mathbf{u}_t | \mathcal{G}, \mathcal{H}) = \prod_{i=1}^n p_{LLM}(x_t^i | x_t^{<i}, \mathcal{G}, \mathcal{H}) \quad (7.2)$$

where x_t^i is the i -th token of the utterance at turn t . We break down the dialogue from the data by turn, yielding N data points for each conversation.

Regarding the interaction between our fine-tuned LLM and a TOD system, we follow the same paradigm from Terragni et al. [2023]. DAUS receives a fresh prompt, which comprises the user’s goal for the ongoing dialogue and the cumulative dialogue history. Unlike Terragni et al. [2023], we do not provide any example dialogues to serve as shots. We additionally post-process the generated utterance to ensure that a clean message is passed to TOD systems (i.e., removal of special characters and trailing tokens).

7.2 Experimental Setting

In this section, we describe datasets, implementation details, and experimental setting for simulator-system interactions.

7.2.1 Data sources

Table 7.1. Dataset statistics after pre-processing.

Dataset	Avg # Turns	Avg # Words per User Utterance	Avg # Words per TOD Utterance
MultiWOZ	5.86	13.13	14.86
<i>AutomotiveData</i>	11.20	3.44	12.06

We consider two data sources to evaluate our approach. First, we experiment on internal dialogue data of user-TOD system phone call interactions within the automotive industry, dubbed *AutomotiveData*.¹ Second, we use the well-established dataset of multi-domain TOD systems – MultiWOZ 2.1 [Eric et al., 2019]. Both data sources contain user goal \mathcal{G} in natural language and multi-turn dialogues (compliant with Section 7.1.1). For each dataset, we randomly sample 2,500 dialogues for training, 300 for testing and 300 for validation. The statistics of the resulting datasets are reported in Table 7.1.

¹In order to protect our user’s privacy, we do not release any user data nor models fine-tuned on user’s data. Examples presented throughout the paper are synthetically constructed, whilst preserving realistic user goals. Users have been informed about and have consented to data collection.

7.2.2 TOD Systems

DAUS communicates with TOD systems through natural language, making it system-agnostic. For our user simulator fine-tuned on *AutomotiveData*, we employ an internal TOD system. To evaluate DAUS fine-tuned on MultiWOZ, we use the ConvLab2 framework [Zhu et al., 2020], extended by Terragni et al. [2023], which integrates LLM-based few-shot user simulators.² We use the same TOD the authors used in their original work. We identify a challenge with the default stopping criteria that prematurely end dialogues when users express gratitude. This doesn't always signify the end of the interaction as users may continue with their goals (e.g., “*Thanks for booking my flight. I also need a hotel*” would terminate the conversation). Therefore, we modify the criteria to exclude termination on “*thanks*” intent. We consequently re-run the experiments presented in Terragni et al. [2023]. Moreover, we publicly release the updated framework and the user simulator fine-tuned on MultiWOZ 2.1.³

7.2.3 User Goal Settings

For the MultiWOZ data within ConvLab2 framework, we follow the previous work for construction of the user goals [Zhu et al., 2020, Terragni et al., 2023]. Specifically, the user goals are randomly sampled, conditioned on the domains and entities frequency in the training data. We generate 100 dialogues per user goal.

For evaluation on our internal TOD system, a domain expert manually defined user goals for 8 test cases, detailed in Table 7.2. The test cases vary depending on the complexity and the main task that the simulator has to fulfill. As such, we label the test cases accordingly: *B* for *book* appointment task, *C* for *cancel* appointment task, *R* for *reschedule* appointment task. Moreover, each label is associated with a graded difficulty indicator, i.e., *easy* or *hard*. We generate 100 dialogues per test case (i.e., per user goal).

7.2.4 Fine-tuning Details

We conduct our experiments with the recently released open-source LLM — Llama-2 [Touvron et al., 2023]. The prompt, mentioned in Section 7.1.2, is constructed by concatenating the task description, user goal \mathcal{G} , and the dialog

²<https://github.com/telepathylabsai/prompt-based-user-simulator>.

³GitHub URL placeholder. Code and fine-tuned model weights provided as an attachment to the submission. Upon acceptance, we will release it under the Apache 2.0 Licence.

Table 7.2. Description of user goals with subtask types.

#n	User subtask type	Difficulty	User goal details
1	Book	Easy	New customer; Available: today 4PM; Transport_type: waiter; Service: check engine.
2	Book	Hard	Known customer with 1 appointment and 2 cars; Available: Wednesday; Transport_type: dropoff; Unknown Service.
3	Book	Hard	Known customer with 3 appointments and 2 cars; Available: Wednesday; Transport_type: dropoff; Two services: engine overheating and oil change.
4	Cancel	Easy	Known customer with 1 appointment.
5	Cancel	Hard	Known customer with 3 appointments.
6	Reschedule	Easy	Known customer with 1 appointment; Available: 10 AM; Transport_type: dropoff; Unknown service.
7	Reschedule	Hard	Known customer with 1 appointment; Available: afternoon; Transport_type: waiter; Service: oil change.
8	Reschedule	Hard	Known customer from unknown phone number; With 3 appointments; Available: Wednesday; Transport_type: loaner; Two services: Oil change and engine check

history \mathcal{H} . Moreover, we separate every utterance with a special “<endturn>” token.

We utilize LoRA [Hu et al., 2021] – a parameter-efficient fine-tuning technique, capable of reaching performances comparable to fully fine-tuned models, whilst requiring only a fraction of the computational resources. We adhere to the hyperparameter recommendations and instructions of the recent work on the topic [Hu et al., 2021, He et al., 2021] and use the following LoRA hyperparameters throughout the experiments: rank r of 64, α of 32, and dropout of 0.05. Moreover, we optimize attention layers (query and key matrices) of the Llama-2 model. We use the 13B Llama-2 version for the main experiments, and the 7B version for comparison and the generalization study. We perform hyperparameter grid search for learning rate on the dev sets of our datasets. We settle for $lr = 3e^{-5}$ and the batch size of 12 and 32 for the 13B and 7B versions, respectively.

7.2.5 Baselines

We compare our Llama-2 fine-tuned model with several pre-trained models in zero-shot or few-shot fashion, following [Terragni et al., 2023, ?]. In particular, we consider the following pre-trained models:

- Llama 2 with 13B parameters.
- GPT-3.5 Turbo4 (Chat-GPT), version 0613 [Brown et al., 2020]. For data privacy reasons, we employ this model only for the MultiWOZ experiments.
- Flan-T5 [Chung et al., 2022] with 3B parameters (XL), to reproduce results of Terragni et al. [2023].

In addition to the LLM-based models, we consider an agenda-based simulator (ABUS) [Wen et al., 2015], designed specifically for MultiWOZ within ConvLab2 framework, thus requiring the knowledge of TOD system’s policy. We include two variants of ABUS: the first with template-based NLG and the second with data-driven NLG, dubbed ABUS-T and ABUS-D, respectively. Let us notice that ABUS is a strong baseline, as it is tailored for communicating with the MultiWOZ-based TOD from ConvLab2, therefore it is included as a reference of the potential upper-bound for user goal fulfillment performance. We follow Terragni et al. [2023] and set the temperature for inference to 0.9 for all MultiWOZ experiments, and 0.7 for internal experiments (value chosen through grid search).

7.3 Evaluation

We comprehensively evaluate our method, aiming to assess its ability to achieve designated user goals in dialogues and its impact on lexical diversity when aligning with real user language patterns. Moreover, we perform qualitative analysis of simulated dialogues via human evaluation. In this section, we detail these evaluation procedures.

Additionally, we examine utterance-level metrics, comparing generated utterances with those in the target dataset, using both general natural language generation and domain-specific entity-based metrics. However, we found that these metrics poorly correlate with the simulator’s task completion. Detailed information about these metrics and their results can be found in Appendix 7.4.4.

7.3.1 Goal Fulfillment Evaluation Metrics

Our objective is to evaluate the goal fulfillment at the end of the dialogue. For MultiWOZ experiments, we consider well-known metrics such as Success, Completion and Book rate. These metrics aim to capture how successful was the dialogue in terms of fulfilling specific subtasks from the user goal (e.g., whether the restaurant is booked). We also compute the average precision (P), recall (R) and F_1 scores by matching the entities expressed through the simulated dialogue to the ones in the initial user goal. These metrics aim to assess the simulator’s faithfulness and consistency of entities with the user goal (e.g., whether the correct restaurant *type* was booked). For a comprehensive understanding of the metric definitions, please refer to [Zhu et al. \[2020\]](#) and [Terragni et al. \[2023\]](#).

Regarding our in-house TOD, it is worth noting that we do not differentiate between *book*, *inform* and *request* entities. Therefore, we adapt the mentioned metrics, except for the Book Rate, while considering all entities as inform entities. Moreover, we compute several metrics specific to automotive domain: *user subtask* indicating whether the subtask (*book*, *cancel*, or *reschedule* the appointment) matches the one given in the user goal; *caller info* and *car info* indicating whether user information (name, phone number) and vehicle information (car year, make, and model) match the ones in the goal, respectively; *transport type* assessing the chosen transport type (e.g., dropping of the vehicle, waiting for the service in the dealership).

Table 7.3. Results of goal fulfillment task in simulator interaction with the internal TOD system. The results are averaged across the eight user goals.

Model	Num Shots	Compl Rate	Succ Rate	P	R	F_1	User Subtask	Caller Info	Car Info	Transport Type	UttLen	Unig	MTLD
FlanT5-XL [Terragni et al., 2023]	2	0.46	0.27	0.72	0.86	0.76	70.9	85.5	65.6	39.2	2.8	209	23.4
Llama-2	0	0.35	0.13	0.62	0.87	0.69	50.4	88.8	72.2	12.8	2.4	161	15.5
	1	0.37	0.12	0.67	0.89	0.74	65.6	89.1	81.6	8.0	2.0	149	14.5
	2	0.36	0.15	0.66	0.91	0.74	68.9	90.3	80.2	8.0	2.0	129	13.7
DAUS	0	0.51	0.40	0.91	0.92	0.91	99.5	98.5	99.0	80.7	1.7	112	16.5

7.3.2 Lexical Diversity of Generated Utterances

Lexical diversity (LD) is a measure of word variability and vocabulary size of a given text corpus, in our case, the set of generated user utterances from 100 conversations. We report MTLD scores [McCarthy, 2005], and a number of unigram words (Unig) and average user utterance length (UttLen). LD results are reported in Section 7.4.2.

7.3.3 Qualitative Analysis

During the analysis of the generated simulated dialogues, we observed several re-occurring issues. We categorize them as the simulator’s failure (*hallucination*, *incomplete user goal fulfillment*, or *looping/repeating* utterances across turns) or TOD system’s failure (*NLU misclassification* due to missing user’s intent or entities, *forcing end of dialogue*, or *looping/repeating* utterances). Our aim is to assess the prevalence of these patterns and identify potential limitations of LLM-based user simulators. To this end, we employ three annotators to annotate 45 dialogues generated with an LLM-based baseline and 45 dialogues generated with DAUS within ConvLab2 framework. The annotators are domain-experts and employees of the authors’ institution. We provide guidelines for each of the categories and go through an on-boarding process with the annotators. The labels for each of the dialogues are determined by majority vote. Annotators reach moderate to good agreement, as measured by Fleiss’ κ , detailed in Appendix 7.5.3.

7.4 Results

In this section, we examine our study’s findings across three main threads. First, we investigate the impact of fine-tuning LLMs with domain-specific data on goal

fulfillment in dialog interactions (Sect. 7.4.1). Next, we explore the link between fine-tuning and the lexical diversity of generated utterances (Sect. 7.4.2). Finally, we assess whether the adaptability of LLM-based user simulators to unseen user tasks is influenced by the diversity of subtask types in their training data (Sect. 7.4.3).

7.4.1 Goal Fulfillment

Internal TOD System. Table 7.3 shows results on the goal fulfillment task of DAUS and the baselines detailed in Section 7.2.5, averaged across different user goals. We present the results per each of the eight specific user goals, detailed in Section 7.2.3, in Table 7.4.

As a first remark, DAUS outperforms all of the baselines across all the goal fulfillment metrics. We observe the largest improvements for domain-specific metrics, e.g., precision and recall of relevant entities and accuracy of the transport type. This indicates that fine-tuning on in-domain data improves simulator’s knowledge of the domain-specific terminology. We further expand on this observation in Section 7.5.1.

Regarding the baselines, FlanT5, employing 2 shots as examples, is the second best model. As observed in [Terragni et al., 2023] as well, this instruction fine-tuned model outperforms Llama-2 with 2 shots in most of the cases.

MultiWOZ Data within ConvLab2. We show the goal fulfillment performance of DAUS and the baselines in interaction with ConvLab2’s TOD system on MultiWOZ 2.1 in Table 7.5. As in Section 7.4.1, we observe strong performance of DAUS. Specifically, DAUS outperforms all of the in-context learning approaches in terms of goal fulfillment, including prior state-of-the-art [Terragni et al., 2023]. Moreover, our method outperforms few-shot GPT-3.5, a model significantly larger than ours (estimated 175 billion parameters vs 13 billion). This further suggests the benefits of fine-tuning LLMs on domain-specific conversational data, as stronger performance can be achieved with significantly smaller LLMs, thus reducing the computational requirements of the simulator.

As a general remark, results on both benchmarks, i.e., the ConvLab2 and our internal one, show significant improvements across multiple goal fulfillment metrics. Thus, we conclude that DAUS indeed does lead to more consistent, reliable, and faithful LLM-based user simulators. We will discuss these results more in depth in our qualitative analysis in Section 7.5.1.

Table 7.4. Results of selected baselines and DAUS (the main method based on Llama-2 13B, as well as the 7B version) per specific user goal.

Subtask	Model	N shots	User Task	Compl Rate	Succ Rate	P	R	F1	Service Info	Transport	Car Info	Caller Info
C_{hard}	Llama-2-13b	0	43	100	43	0.74	0.79	0.76				99.5
	Llama-2-13b	2	52	100	44	0.77	0.86	0.8				100
	FlanT5-xxl	0	61	100	57	0.83	0.84	0.83				99.5
	FlanT5-xxl	2	65	100	63	0.84	0.9	0.85				99
	FlanT5-xl	0	67	100	64	0.85	0.89	0.86				98
	FlanT5-xl	2	75	100	73	0.89	0.94	0.9				100
	DAUS-7b	0	93	100	93	0.96	0.96	0.96				100
	DAUS	0	100	100	100	1	1	1				100
B_{hard2}	Llama-2-13b	0	94	23	1	0.59	0.89	0.67	31	4	73	57.5
	Llama-2-13b	2	98	27	1	0.62	0.87	0.71	44	7	84	64
	FlanT5-xxl	0	81	64	19	0.78	0.91	0.81	74	45	78.3	78
	FlanT5-xxl	2	91	72	15	0.77	0.86	0.8	83	42	86	84
	FlanT5-xl	0	81	18	4	0.37	0.77	0.44	36	18	29.3	22.5
	FlanT5-xl	2	95	58	6	0.66	0.81	0.7	75	37	74.6	68.5
	DAUS-7b	0	99	76	29	0.87	0.84	0.85		64	100	88.5
	DAUS	0	100	89	50	0.93	0.85	0.88	90	93	100	99
B_{easy}	Llama-2-13b	0	97	43	23	0.77	0.91	0.82	59	49	77	100
	Llama-2-13b	2	100	43	4	0.76	0.93	0.83	51	15	92.3	100
	FlanT5-xxl	0	90	65	46	0.85	0.93	0.86	70	63	90	99
	FlanT5-xxl	2	98	57	50	0.89	0.88	0.88	60	80	84	100
	FlanT5-xl	0	94	14	14	0.73	0.86	0.78	34	91	40.6	99.5
	FlanT5-xl	2	97	23	22	0.81	0.85	0.82	26	94	54.3	100
	DAUS-7b	0	96	55	22	0.92	0.87	0.89		99	98.7	100
	DAUS	0	100	37	15	0.93	0.89	0.91	38	98	100	98
B_{hard1}	Llama-2-13b	0	65	1	0	0.59	0.9	0.69	19	6	64	100
	Llama-2-13b	2	83	0	0	0.62	0.9	0.71	4	1	71.3	100
	FlanT5-xxl	0	80	10	0	0.82	0.84	0.81	16	71	84	99.5
	FlanT5-xxl	2	56	9	0	0.69	0.86	0.73	35	44	69.3	100
	FlanT5-xl	0	40	2	0	0.6	0.84	0.67	25	28	61.6	100
	FlanT5-xl	2	24	1	0	0.48	0.91	0.6	62	10	48.3	100
	DAUS-7b	0	78	2	0	0.81	0.82	0.8		80	86	100
	DAUS	0	99	15	0	0.84	0.84	0.83	17	84	94	95.5
C_{easy}	Llama-2-13b	0	39	100	37	0.76	0.78	0.76				100
	Llama-2-13b	2	67	100	61	0.85	0.89	0.86				100
	FlanT5-xxl	0	75	100	74	0.91	0.89	0.89				100
	FlanT5-xxl	2	94	100	93	0.98	0.97	0.98				100
	FlanT5-xl	0	73	100	71	0.89	0.87	0.87				100
	FlanT5-xl	2	97	100	97	0.99	0.99	0.99				100
	DAUS-7b	0	100	100	100	1	1	1				100
	DAUS	0	100	100	100	1	1	1				100
R_{easy}	Llama-2-13b	0	14	1	0	0.51	0.91	0.63	2	7	77.6	100
	Llama-2-13b	2	38	2	2	0.54	0.93	0.67	9	6	80	98.5
	FlanT5-xxl	0	60	3	2	0.78	0.91	0.83	16	80	98.3	99.5
	FlanT5-xxl	2	76	28	8	0.84	0.91	0.86	45	82	99.3	100
	FlanT5-xl	0	44	8	3	0.75	0.88	0.8	26	79	92	100
	FlanT5-xl	2	71	30	3	0.81	0.91	0.85	54	70	98.6	100
	DAUS-7b	0	99	10	10	0.97	0.91	0.94		99	100	100
	DAUS	0	99	6	5	0.91	0.93	0.91	9	100	100	100
R_{hard1}	Llama-2-13b	0	25	7	0	0.55	0.88	0.66	27	5	79	93.5
	Llama-2-13b	2	55	13	6	0.61	0.93	0.72	42	15	79.6	91
	FlanT5-xxl	0	20	14	5	0.67	0.87	0.75	88	47	86	85.5
	FlanT5-xxl	2	34	29	10	0.68	0.8	0.73	95	31	81.6	80.5
	FlanT5-xl	0	15	12	3	0.43	0.61	0.49	72	16	40.3	41.5
	FlanT5-xl	2	61	53	11	0.71	0.83	0.74	82	21	74.6	74.5
	DAUS-7b	0	48	22	21	0.72	0.87	0.78		77	84.3	96
	DAUS	0	100	62	46	0.9	0.94	0.91	98	99	100	100
R_{hard2}	Llama-2-13b	0	26	1	0	0.48	0.89	0.59	20	6	62.3	60
	Llama-2-13b	2	58	1	0	0.53	0.94	0.65	23	4	74	69
	FlanT5-xxl	0	26	7	4	0.67	0.93	0.75	67	50	81	79
	FlanT5-xxl	2	42	8	1	0.64	0.83	0.7	49	33	82.3	68.5
	FlanT5-xl	0	9	0	0	0.21	0.43	0.26	17	5	19.3	13.5
	FlanT5-xl	2	47	2	0	0.42	0.69	0.47	14	3	43	41.5
	DAUS-7b	0	36	6	5	0.67	0.83	0.73		67	98	69.5
	DAUS	0	98	1	0	0.78	0.93	0.84	5	10	100	95.5

Table 7.5. Performance on MultiWOZ 2.1 within ConvLab2 framework.

Model	Num Shots	Compl Rate	Succ Rate	Book Rate	P	R	F_1	UttLen	Unig	MTLD
ABUS-T [Wen et al., 2015]	-	0.93	0.83	0.85	0.84	0.94	0.86	17.4	527	46.9
ABUS-D [Wen et al., 2015]	-	0.86	0.60	0.75	0.87	0.90	0.87	9.8	327	28.0
FLanT5-XL [Terragni et al., 2023]	2	0.19	0.13	0.46	0.45	0.39	0.39	13.7	888	41.2
Llama-2	0	0.07	0.04	0.13	0.31	0.21	0.23	8.1	697	30.7
	2	0.09	0.08	0.30	0.46	0.34	0.39	10.0	765	38.8
GPT-3.5	2	0.35	0.19	0.34	0.49	0.52	0.48	16.3	626	38.1
DAUS	0	0.41	0.29	0.66	0.69	0.69	0.67	10.6	789	54.9

7.4.2 Lexical Diversity

Lexical diversity (LD) of generated user utterances from internal TOD system and MultiWOZ experiments is presented in the last 3 columns of Tables 7.3 and 7.5. We observe a drop in LD, as measured by the length of the generated utterances and the total number of unigrams, when DAUS is fine-tuned on *AutomotiveData*. This suggests a limited vocabulary in the training data, which is expected due to the real users often responding with one or two words, especially in the *cancel* task. DAUS had a relatively high MTLD score, because of the correctly generated caller, car and transport entities, which usually have unique values. However, a low unigram score is due to averaging metrics over 8 user tasks, where only 3 of them are the entity-rich *book* task. Meanwhile, the higher LD of FLanT5-based method is due to its prevalent hallucinations, thus falsely inflating the LD scores by generating out-of-context content (see Section 7.5.1).

In MultiWOZ-based experiments, results indicate higher LD than ABUS baselines, as measured by MTLD, while the generated utterances are slightly shorter compared to FLanT5. As such, DAUS does not seem to lose LD during fine-tuning on MultiWOZ, while fine-tuning on *AutomotiveData* seems to reduce it slightly. This can be explained by the fact that *AutomotiveData* contains both specific vocabulary and utterances from real product users, which makes it hard for in-context learning approaches to imitate. On the other hand, fine-tuning procedure enables the model to learn the particulars of such interactions.

Table 7.6. Percentage of dialogues with successfully identified subtask types across the test cases, with models fine-tuned on specific combinations of subtask types.

	DAUS (C+R+B)	C+R	B+R	B+C
B_{easy}	99	100	100	99
B_{hard1}	93	29	85	99
B_{hard2}	99	86	94	97
C_{easy}	96	100	75	99
C_{hard}	100	100	77	96
R_{easy}	88	100	98	34
R_{hard1}	97	50	69	0
R_{hard2}	86	84	56	0

7.4.3 Generalization to Unseen User Tasks

Table 7.6 shows the percentage of successful subtask identifications for four variants of our model: DAUS fine-tuned on the full dataset described in Section 7.2.1, and DAUS fine-tuned on modified datasets by removing certain subtasks (*book* (B), *cancel* (C), or *reschedule* (R)) from the training sets. With this experiment, we aim to assess the generalization abilities of our approach.

Results show a decrease in performance when a model is not shown the specific subtask during training. For example, when we fine-tune DAUS on the combination of *book* and *reschedule* subtasks, we observe a considerable drop in performance on the *cancel* subtask. However, the largest drop is observed in the most complex subtask type, *reschedule*, where the model fine-tuned on $B+C$ data completely fails to successfully communicate its goal for both R_{hard} test cases.

We can conclude that DAUS does not generalize well to unseen user goal subtasks. Nevertheless, the overall performance of the fine-tuned models across all of the subtasks is still comparable to the performance of few-shot based models (e.g., $B+C$ correctly predicts the subtask type, on average, in 66% of the dialogues, while Llama-2 2-shot does it in 69% of the dialogues, on average).

7.4.4 Utterance-Level Metrics

In addition to dialogue-level metrics detailed in Section 7.3, we consider a number of utterance-level metrics. Such metrics are based on comparisons of generated utterances to the target utterance in the test set of the appropriate dataset, described in Section 7.2.1. We consider two main types of utterance-level met-

Table 7.7. NLG- and NLU-based utterance-level metrics.

		BLUE	ROUGE	BERTScore	METEOR	utterance_sim	intent_sim	entities_R	entities_P	entities_F ₁
MultiWOZ	Llama-2-7b	0.12	0.18	0.85	0.16	0.31	0.67	0.22	0.35	0.37
	Llama-2-13b	0.13	0.19	0.85	0.16	0.32	0.67	0.26	0.36	0.38
	Llama-2-13b-fine-tuned	0.12	0.19	0.85	0.16	0.31	0.67	0.26	0.35	0.37
AutomotiveData	Llama-2-7b	0.22	0.24	0.88	0.17	0.52	0.58	0.44	0.18	0.47
	Llama-2-13b	0.42	0.43	0.92	0.26	0.68	0.73	0.42	0.25	0.47
	Llama-2-13b-fine-tuned	0.42	0.43	0.92	0.26	0.68	0.73	0.42	0.25	0.47

rics: 1) natural language generation (NLG) metrics; and 2) natural language understanding-based (NLU) metrics. We compute several well-known NLG metrics: BLEU [Papineni et al., 2002], ROUGE [Lin, 2004], BERTScore [Zhang et al., 2020a], METEOR [Lavie and Agarwal, 2007], as well as cosine similarity between embedded generated and target utterances.

Moreover, we design several domain-specific NLU-based metrics. TOD systems are composed of multiple modules, with NLU module, that aims to understand and parse the given user utterance, being one of the essential modules. Thus, we employ NLU component of the TOD systems to extract user (simulator) intent and mentioned entities, Similarly to NLG-metrics, we compare the intent and entities extracted from the generated utterance, to those in the target utterance. Specifically, we design the following metrics:

- Cosine similarity between the embedded **intents** extracted from the generated utterance and the target utterance. Intents are embedded with RoBERTa model.
- Cosine similarity between the generated and the target utterance, in which the entities were masked. Utterances are embedded with RoBERTa model.
- Precision, Recall, and F_1 of **entities** between the generated and the target utterances.

Table 7.7 shows the results across the described metrics. However, as observed in multiple experiments throughout this dissertation, the automated utterance-level metrics do not show any significant differences between the different approaches. In other works, they do not correlate well with other metrics used to evaluate our approach. Therefore, we do not rely on them for making strong conclusions about overall conclusions about models’ performances.

Table 7.8. Percentage of the observed patterns per sample annotated in simulated dialogues in MultiWOZ.

Label	<i>FlanT5</i>	DAUS
Hallucination	73%	36%
Looping simulator	69%	6%
Incomplete goal	78%	53%
Looping system	20%	22%
NLU misclassification	60%	40%
Forced end	27%	27%

7.5 Qualitative Analysis

In this section, we detail and discuss the findings of our qualitative analysis of simulated dialogues.

7.5.1 Human Evaluation of Generated Dialogues

Table 7.8 presents the prevalence of patterns, described in Section 7.3.3, observed through manual annotation of the simulated MultiWOZ dialogues. We observe consistent decrease in hallucinations, reduced number of dialogues with incomplete goal fulfillment, as well as reduced repetition of utterances in dialogues generated by DAUS, compared to FlanT5-based simulator. Below, we report the main findings from our analysis.

Hallucinations. The percentage of dialogues containing hallucinations drops from 73% for FlanT5-based simulator to 36% with DAUS. We observe that FlanT5 frequently experiences severe failures, mostly because it generates non-specified pieces of information, such as defining a random range of time for a taxi pickup, inventing a location for an attraction or referring to a restaurant that have not been previously mentioned. Such hallucinations lead to dialogue failures, without possibility to recover the conversation. On the other hand, DAUS does not hallucinate nor misinterpret entities from the user goal and the dialogue, but rather sometimes asks for additional information that is not required by the user goal (e.g., asking restaurant’s phone number, even though it is not strictly specified in the goal). Thus, we conclude that not only the prevalence of the hallucinations is reduced, but also their severity.

TOD system failures. We observe that the dialogue frequently stops before all of the subtasks from the user goal are completed (i.e., if the goal requires to book a restaurant and a taxi, perhaps only the restaurant is booked). While this is partly due to the simulator missing certain part of the user goal, it often happens due to the TOD system failures. We observe several issues with the TOD system within the ConvLab2 framework. First, as mentioned in the previous paragraph, the system aims to prematurely complete the dialogue, thus offering no opportunity to the user (simulator) to fully express their goal. Second, the system’s NLU component often misclassifies certain entities, leading to incorrectly booked restaurants or similar. An example dialogue of both prematurely ending the dialogue and NLU misclassification:

CUSTOMER: Can you help me with a robbery that happened? I am the victim.

ASSISTANT: Thank you for using our services . (forcing premature end of conversation)

CUSTOMER: It happened at 9:10

ASSISTANT: Can you tell me which day you ’d like to travel on? What is your destination? Where will you be departing from? (NLU misclassification)

Finally, we observe loops in system’s policy, where the system is stuck in policy states and is not able to continue the conversation.

7.5.2 Domain-specific Vocabulary

During our analysis of conversations generated using our internal TOD system, we observed that DAUS acquires domain-specific vocabulary, which we believe enhances its performance. For instance, when scheduling a vehicle service appointment, customers specify the preferred transport type, usually *dropoff* (the customer drops off the vehicle and returns when the service is complete) or *waiter* (the customer waits at the dealership during the service). This phenomenon, supported by performance improvements for *Transport Type* (see Table 7.3), is evident through qualitative dialogue analysis. Our initial experiments indicate that in-context learning approaches aiming to explain these terms in the prompt do not consistently capture their nuances.

Additionally, we noticed that, when fine-tuned on *AutomotiveData* containing phone call conversations with real users, DAUS tends to generate filler words like “uhm” and “yeah”.

7.5.3 Inter-annotator agreement

Table 7.9 shows inter-annotator agreement, as measured by Fleiss’ κ , per model and per type of pattern.

Table 7.9. Inter-annotator agreement, as measured by Fleiss’ κ for samples from DAUS and FlanT5-XL.

	DAUS	FlanT5-XL
Hallucination	0.365	0.499
Incomplete Goal	0.585	0.754
Looping Simulator	0.319	0.687
NLU Misclassification	0.356	0.308
Forces end of dialogue	0.314	0.367
Looping System	0.640	0.084

Table 7.10 shows three examples of failed simulated dialogues.

7.6 Conclusions

The use of a domain-aware LLM-based user simulator, such as DAUS, shows promising results in multi-turn interactions with TOD systems. It can understand and fulfill user goals by generating consistent and faithful utterances. Compared to previous LLM-based approaches, our method has demonstrated superior performance, as measured by multiple metrics designed to capture the fulfillment of the given goal, as well as faithfulness across the dialogue. This indicates that DAUS is capable of effectively simulating user behavior and can serve as a valuable tool for testing and evaluating TOD systems. Moreover, our approach requires relatively small training dataset (just a few thousand dialogues) and imposes modest computational demands, thanks to parameter-efficient fine-tuning. Consequently, it stands as a feasible choice for wider adoption within the NLP community.

The potential applications of LLM-based user simulators are synthetic data augmentation, supporting reinforcement learning approaches, and TOD system evaluation. DAUS’s reliability and consistency to the user goal make it particularly suitable for TOD system evaluation. As we have seen previously, an incomplete user goal can mainly imply two scenarios: a user simulator who hallucinates or a TOD system that is not able to understand the user’s requirements. Therefore, the presence of a reliable user simulator is crucial: it allows us to identify the TOD system’s errors with high accuracy.

Moreover, we stress that at the center of our approach is an LLM, leading to potentially different generations given the same input, depending on the sampling method. This means that DAUS is more flexible than certain agenda-based simulators, which usually rely on template-based responses. As such, we are able to simulate a dialogue with the same user goal multiple times, which results in multiple different attempts of the simulator to fulfill its goal, going through potentially different conversational paths. Therefore, we are able to test the robustness of the TOD system to different expressions of the same user goal.

We noticed a decrease in performance when certain user subtasks are omitted from the training dataset when we fine-tune DAUS, although the overall performance remains comparable to that of few-shot models. In our analysis, we did not investigate if providing one or two dialog shots would address this performance decrease.

Table 7.10. Examples of common failures observed through qualitative analysis.

Example 1:
 REQUIREMENTS: You are looking forward to trying local restaurants. You are looking for a particular restaurant. Its name is called cote. Once you find a restaurant, make sure you get postcode. Once you find the restaurant you want to book a table for 7 people at 12:00 on tuesday. You are looking for a place to stay. The hotel should be in the cheap price range. The hotel should be in the north. The hotel does not need to include free parking. The hotel should be in the type of guesthouse. If there is no such hotel, how about one that is in the moderate price range. Once you find a hotel, make sure you get postcode, phone number. You also want to book a taxi to commute between the two places. **The taxi should arrive at the restaurant from the hotel by 02:45.** Once you find a taxi, make sure you get car type, phone number
 CONVERSATION:
 ...
 CUSTOMER: Thank you and please wait for the taxi .
 ASSISTANT: What time will you be leaving . What time would you like to arrive ?
 CUSTOMER: I would like to depart at 2 :45 **to return at 5 :15** . (*hallucination*)
 ASSISTANT: Thank you for using our services .
 CUSTOMER: Thank you .
 ASSISTANT: I 'm happy to have been able to help you today .
 CUSTOMER: Thank you for the help the table has been booked and taxi has been booked .
 ASSISTANT: What time would you like to leave ?
 CUSTOMER: I would like to leave at 2 : 45 **to return at 5 :15** . (*hallucination*)
 ASSISTANT: Your booking is complete , a tesla will be picking you up . The contact number is 45067427198 .
 CUSTOMER: Thank you , have a nice day !
 ASSISTANT: Ok , thank you . Have a good day .
 CUSTOMER: Can you check cote restaurant please . (*looping*)
 ASSISTANT: It is in the centre area .
 CUSTOMER: Can you check cote restaurant please . (*looping*)
 ASSISTANT: They serve french . It is in the centre area .
 CUSTOMER: Thank you for your help and have a good day !
 ASSISTANT: Thank you for using our services .
 CUSTOMER: Can you check the cote restaurant . (*looping*)
 ASSISTANT: Okay , how about cote ? That is a french restaurant .
 CUSTOMER: Can you check cote restaurant please .
 ASSISTANT: That is a french restaurant . They are in the expensive price range .
 CUSTOMER: Thank you for your help and have a good day !
 ASSISTANT: You are more than welcome !
 CUSTOMER: Can you please look up cote restaurant for me please . (*looping*)
 ...

Example 2:
 REQUIREMENTS: You are looking for a place to stay. The hotel should have a star of 4. The hotel should be in the centre. The hotel should be in the expensive price range. The hotel does not need to include free parking. If there is no such hotel, how about one that has free parking. Once you find the hotel you want to book it for 5 people on sunday for 3 nights
 CONVERSATION:
 CUSTOMER: I want to find a hotel with a four star rating.
 ASSISTANT: I am sorry I have no listings for any with four stars . (*NLU misclassification since later it does find criteria*)
 CUSTOMER: How about a guesthouse?
 ASSISTANT: Unfortunately , I couldn't find anything with four stars . I can't find any guesthouse that fit your criteria , i'm sorry .
 CUSTOMER: That's fine, but can you try finding an expensive hotel in the centre of town?
 ASSISTANT: There aren't any that match your criteria in the centre . Any other suggestions ? There is none that is expensive . Would you like to change your criteria ?
 I am sorry I have no listings for any with four stars . Sorry there is no hotel fitting the description you asked for.
 CUSTOMER: I am looking for a 4 star hotel that is expensive and in the centre of town with free parking.
 ASSISTANT: How about university arms hotel ? It is in the centre area . It has wifi . It is a hotel . The hotel is 4 stars .
 CUSTOMER: Can you book it for me?
 ASSISTANT: You are more than welcome !
 CUSTOMER: Ok. Book me for 5 people on sunday for 3 nights.

Example 3:
 REQUIREMENTS: You are also looking for a train. The train should depart from cambridge. The train should leave on sunday. The train should go to stansted airport. The train should arrive by 08:45. Once you find the train you want to make a booking for 1 people. You are looking forward to trying local restaurants. The restaurant should be in the centre. The restaurant should be in the expensive price range. The restaurant should serve modern european food. Once you find a restaurant, make sure you get postcode
 CONVERSATION:
 ...
 CUSTOMER: Thanks. I'm also looking for a restaurant.
 ASSISTANT: I have 110 options for you !. How about curry prince ?
 CUSTOMER: I need something in the centre. and I would like to spend some money. Expensive please.
 ASSISTANT: We have 33 such places . I would suggest cote . It is located in the centre .
 CUSTOMER: Is Cote a modern European restaurant?
 ASSISTANT: There no modern european restaurants that I can find right now . Would something else work ? I do n't have anything in the expensive range that fits that criteria . I 'm sorry . i can not find details for cote . I have n't found any in the centre.
 CUSTOMER: I really need a modern european restaurant, preferably expensive. I'll try a different area, if necessary.
 ASSISTANT: I do not have anything in that price range for modern european . Another criteria perhaps ? There are no expensive ones . I 'm sorry . i can not find details for cote . There no such restraurants in centre. (*looping*)
 CUSTOMER: Can you please try one more time? I really need an expensive restaurant, thank you.
 ASSISTANT: I am unable to find any modern european restaurants in town . There are none in expensive , perhaps something else ? I 'm sorry . i can not find details for cote . I did not find any restaurants in centre . (*looping*)
 CUSTOMER: What about expensive modern european?
 ASSISTANT: There are no modern european food places , shall I run another search ? There are none in expensive , perhaps something else ? I 'm sorry . i can not find details for cote . Sorry , there are no restaurants like that in the centre . (*looping*)
 CUSTOMER: Yes please run another search.
 ASSISTANT: Thank you for contacting the help desk . Have a great day . (*forces premature end of conversation*)
 CUSTOMER: What did you find for expensive modern european?
 ASSISTANT: There are no modern european restaurants unfortunately . There are none in expensive , perhaps something else ? I 'm sorry . i can not find details for cote . I am sorry there is none even in the centre. (*looping*)
 CUSTOMER: You are sorry. I am sorry. You are unhelpful. I will find my own place to eat.
 ASSISTANT: Have a good day .
 CUSTOMER: Wait, I'm sorry. What can you suggest for expensive modern European restaurants?
 ASSISTANT: I do not have anything in that price range for modern european . Another criteria perhaps ? I do n't have anything in the expensive range that fits that criteria . I 'm sorry . i can not find details for cote . I am sorry but there are no restaurants that fit that criteria in the centre . (*looping*)
 CUSTOMER: I would really like a restaurant please. Can you search again?
 ...
 ASSISTANT: Have a good day .

Part III

Discussion and Conclusions

Chapter 8

Discussion

While we discussed specific research pieces presented throughout the dissertation in their corresponding chapters, in this chapter, we aim to aggregate the findings of our multiple studies and provide an overview from an holistic point-of-view. This chapter offers discussion on the topics covered throughout the dissertation and acknowledge the limitations of our approaches. We first address the mixed-initiative approaches, followed by the discussion on LLM-based user simulation methods for conversational systems.

8.1 Clarification in Conversational Search

In this section, we reflect on the studies presented throughout the Part I of the dissertation, aggregate their findings, and acknowledge their limitations.

8.1.1 Generating Clarifying Questions

In Chapt. 3, we address the challenge of constructing clarifying questions. To this end, we proposed an LLM approach to facet-based question generation. We note two main discussion points of this approach:

- Extracting appropriate query facet is not straightforward;
- LLMs hallucinate and are a black box.

We discuss both points below.

Facet Extraction for Clarifying Questions

To design our question generation method [Sekulić et al., 2021], we assume the facets needed to construct the question are given. However, this is not the case in real-world scenario, where we need to deal with continuous stream of new queries. To address this issues, we analyze the possibility of extracting the query facets from a list of documents retrieved in response to the initial query [Sekulić et al., 2022]. By comparing the retrieved documents to the already constructed clarifying questions from Qulac [Aliannejadi et al., 2019b], we find that facets from questions indeed do appear in the list of documents. However, their selection is not straightforward. In other words, although the retrieved documents provide a rich source of information, especially in terms of entities, correctly utilizing this information with a goal of selecting facets for clarifying question construction is challenging. The challenge mostly arises from the sheer volume of information available. Specifically, a large number of entities appear in each of the documents, while only a few of them are feasible for a selection as a query facet. We find that frequency-based features are not successful in selecting such facets from the noisy set of entities, and thus call for further research on the described facet-based approaches.

MIMICS [Zamani et al., 2020b], a collection of datasets for studying clarification in search, is, on the other hand, constructed from search engine query logs. Such query logs have two main advantages over retrieved documents, viewed through the task of facet extraction: 1. they are large-scale, offering billions of queries and query reformulations; 2. they are based in real-world queries, generated by real users. However, the sheer volume of queries also carries some disadvantages, mostly in terms of the high noise of the data. Thus, pruning techniques are also necessary to extract precise query facets. Regardless, Zamani et al. [2020b] present and release multiple datasets, ranging from scale-oriented (containing thousands of examples) to precision-oriented datasets (annotated by humans). Thus, facet extraction from query logs is a valuable approach for constructing clarifying questions. However, as mentioned, these query logs are usually not available to wider research audience.

Another line of research attempts to generate query facets for clarification. Recently, Hashemi et al. [2021] propose an LLM-based approach for learning and generating query intents. While the authors report decent success on generating MIMICS [Zamani et al., 2020b] clarification panes, which consist of a question and several query facets (intents), their applicability to conversational search has not been explored. Nonetheless, LLMs' generative capabilities pose a promising direction in both generating query facets [Hashemi et al., 2021] and generating

clarifying questions [Sekulić et al., 2021].

We conclude that significant contributions towards construction of clarifying questions has been made in the last several years. However, approaches presented in this dissertation, based on extracting query facets for clarifying questions from the retrieved list of documents, prove to be difficult for successful utilization. Thus, extracting query facets from query logs [Zamani et al., 2020b] or generating them with LLMs [Hashemi et al., 2021] might result in higher-quality facets, and, consequently, in higher-quality clarifying questions.

LLM-based Question Generation

In recent years, LLMs have reached unprecedented success in text generation. Consequently, they have proved useful in generating clarifying questions. However, LLMs are known to hallucinate [Ji et al., 2023], i.e., generating text not aligned with the underlying task and context. Thus, careful fine-tuning techniques and evaluations are necessary to ensure consistency of the generated text. Nonetheless, it does not guarantee the complete mitigation of hallucinations, as hallucination can as well happen due to overfitting to the training data. On the other hand, template-based question generation [Zamani et al., 2020b] mitigates this issue, as questions are usually directly constructed from the query facets. However, as shown in Sekulić et al. [2021], such questions are less natural than the LLM-generated ones.

Moreover, LLM approaches to generating questions presented in this dissertation [Sekulić et al., 2021] require appropriate training data to be fine-tuned on. We note that these kind of datasets might not always be available. Thus, adapting our approach to slightly different scenarios, e.g., to a scenario where a question about multiple facets is generated, requires to first construct an appropriate dataset for the task. Recent state-of-the-art LLMs, e.g., ChatGPT, offer the possibility of in-context learning—the task is simply explained in the prompt fed to the LLM, thus mitigating the need for large datasets and fine-tuning procedures. However, as discussed in Sect. 8.2.3, such LLMs are significantly larger in size, therefore increasing the operational computational costs. Moreover, in-context learning for clarifying question construction has not yet been explored.

8.1.2 Limitations

In this section, we acknowledge and discuss limitations of our approaches to modeling mixed-initiative conversational search, presented throughout the dissertation.

Timely Interventions

Throughout the dissertation, we propose and discuss several methods related to clarifying questions. However, an important principle of a holistic MICS system is a timely intervention [Horvitz, 1999]. The system should thus be able to decide when to be proactive and potentially prompt the user with a question. Such timely actions can be modeled through clarification need, i.e., whether there is a need for asking a clarifying question, based on the user’s query and context [Aliannejadi et al., 2019b]. Moreover, Zou et al. [2023] find that prompting the user with poor-quality clarifying questions might actually hurt their satisfaction with the search system, as well as result in poor search performance. The system might thus chose not to ask a question, if no high-quality question is available.

In our work, we follow the evaluation settings proposed by the curators of the used datasets [Aliannejadi et al., 2019b, 2020b, Zamani et al., 2020b], which dictate that clarifying question can, or should, be prompted at each conversational turn. However, as we do not explicitly model clarification need, nor other types of timely interventions, we acknowledge this as a limitation that should be addressed in the design of next-generation MICS systems. A recent study towards this direction was done by Meng et al. [2023], who analyze system initiative in CIA systems.

Single- vs Multi-turn Interactions

While the core subject of the dissertation is *conversational* search, thus implying multi-turn interactions, as described in Sect. 2.1, certain studies presented in this document remain limited to one or two conversational turns only. The reason for this is mainly the design of the datasets available for our research, which mostly rely on single-turn evaluation [Aliannejadi et al., 2019b] or predefined conversational trajectories [Dalton et al., 2020]. For example, the main task in Qulac [Aliannejadi et al., 2019b] is to select an appropriate clarifying question based on the initial user’s query. After asking the question and receiving the answer, the system presents relevant results to the user, where the conversation stops. Nonetheless, we acknowledge that these data collections provide a crucial step in the development of MICS systems and our contributions built on top of them should generalize to a true conversational setting, which is still troublesome to achieve [Balog, 2021].

Multi-faceted Clarifying Questions

We proposed and analyzed several methods for composing facet-based clarifying questions [Sekulić et al., 2021, Sekulić et al., 2022, Litvinov et al., 2024]. However, these methods focused only on single-faceted questions, such as “Are you interested in $[X]$?”, where X is a query facet. As the MICS needs to balance the cost, in terms of user’s time and effort, and their benefit [Aliannejadi et al., 2021a], multi-faceted questions might be a better fit in certain scenarios. For example, instead of asking a question three turns in a row, about facets X , Y and Z , the system might simply ask “Are you interested in $[X]$, $[Y]$, or $[Z]$?”. We additionally allude to the fact that selecting these facets, out of a larger group of facets, is a challenging research direction.

Different Types of Initiative

While this dissertation addressed mixed-initiative conversational search mostly through the lens of clarifying questions, mixed-initiative extends to different types, such as prompting the user with suggestions or warnings. While we acknowledge the importance of these types of system initiative, they fall out of scope for the topic of this dissertation, as our work remains limited to modeling clarifying questions.

User Engagement Prediction

In Chapt. 4, we presented our approach for predicting user engagement with clarification panes in a traditional search setting. We showed that transformer-based methods perform significantly better in the task of engagement level prediction than traditional ML models. Furthermore, we demonstrated the benefits of utilizing information from search engine result pages, such as titles and text snippets of retrieved documents. However, while we hypothesize that the benefit arises from the signals indicating ambiguous or faceted queries (e.g., such queries might retrieve a list of results that has been diversified [Jiang et al., 2015]), this hypothesis has not been explored.

Moreover, we do not explore the transferability of the engagement level prediction task to selecting appropriate clarification panes. Intuitively, engagement level of a clarification pane should correlate with its relevance to the query, thus potentially serving as a strong signal for ranking and selecting the panes. However, we do not explore this hypothesis in this dissertation.

8.2 LLM-based User Simulation

Chapters 6 and 7 present LLM-based user simulators for conversational systems, including conversational search systems and task-oriented dialogue systems. We discuss both advantages and limitations of our approach. Our main goal in the development of simulation methods is to ease the *evaluation* of conversational systems. However, we discuss other applications of user simulation.

8.2.1 Advantages over Agenda-based Simulation

LLM-based approaches have several advantages over agenda- and rule-based based approaches to user simulation:

Independent of a System. LLM-based user simulators interact with a conversational system strictly through natural language, as a real user would. Thus, they do not require knowledge of the inner-workings of the system, which is a limitation of the agenda-based approaches that usually rely on specifically defining actions in system’s policy.

Adaptable to Changes in the System. Due to their independence, LLM-based simulators are more easily adaptable to slight changes in the system. For example, an added state and action in the system’s policy would require the implementation of corresponding pair of actions in the agenda-based simulator. On the other hand, LLMs are more robust to such changes, as their interaction is strictly through text.

Transferable Across Systems. Not only are the LLM-based user simulators adaptable to intra-system changes, but also to different systems. As shown in Chapt. 6, our user simulator is able to interact with different varieties of conversational search systems. This enables seamless comparison of different systems.

Faster Development. The implementation of agenda-based simulators involves designing a policy, with its set of actions and states, which corresponds to the system’s policy. Thus, their implementation can be as complex as designing the conversational system we aim to interact with. On the other hand, LLMs offer faster development and enable implementation of the simulators by only fine-tuning them on relevant conversational data. Moreover, the development is especially sped up with in-context learning, where the necessity of fine-tuning is eliminated

and the simulator only requires specifically designed instructions to interact with the system.

8.2.2 Exploring Conversational Trajectories

Large language models offer inherent randomness in text generation. Specifically, generated texts depend on multiple sampling parameters (e.g., temperature, nucleus sampling parameter). These can either be set to be more conservative and produce predictable text or to produce novel text. By allowing a degree of randomness in utterances generated by LLM-based user simulators, we ensure the ability to explore multiple conversational trajectories.

As discussed multiple times through this dissertation, the simulator’s task is to satisfy given information need or given user goal. As the number of conversational trajectories from an initial query to satisfying these goals is potentially infinite [Balog, 2021], meaning each user can arrive at the goal in a slightly different way, large language models offer us the ability to explore these paths.

We explored this hypothesis in Chapt. 7, where we allow the simulator to interact with our system multiple times, with slightly different conversations generated each time. This allows to compute the ratio of successful conversations, i.e. ones where the simulator has reached its goal, and, more importantly, analyze in detail unsuccessful conversations. Often, we observe that slight changes in expression of simulators user goal, result in vastly different conversational trajectories. This allows us to potentially probe task oriented dialogue systems, and automatically detect errors, such as missed intend or loops in the policy. We note that this would not be possible with agenda-based simulators nor with template based text generation, thus emphasizing the advantages of the LLMs.

8.2.3 Performance vs Cost

In Chapt. 6, we introduced two user simulators for conversational search. We note that in-context learning approach has several potential advantages over the previously introduced fine-tuning-based approach. For example, only a couple of examples must be given to the model, thus mitigating the need to create task-specific datasets. As such, prompt-based few-shot learning can adapt to various tasks. Nonetheless, prompting only became a recently valid method due to significant advancements in LLMs. However, the next generation of LLMs requires significantly more processing power and is not feasible to run on single compute nodes. This consequently raises the cost of such methods. Thus, fine-tuning medium-sized LLMs, such as GPT-2, might still be a potentially desirable path.

In this paper, we compare the two methods across several aspects and discuss the potential advantages of one over the other.

While both GPT2- and GPT3-based user simulators can generate natural and valuable answers to clarifying questions, as demonstrated by our experiments presented in Sect. 6.5, GPT3 is still significantly better. Moreover, the difference in performance becomes wider in the multi-turn setting, indicating the overall superiority of the GPT3-based simulator for the task. This was expected, as GPT-3 was trained on a significantly larger dataset (570GB of text) than GPT-2 (40GB of text) and is much more significant in terms of parameters (175 billion for GPT3 vs 1.5 billion for GPT2) [Brown et al., 2020]. However, the increase in performance comes with the rise in cost. Specifically, *Davinci* model used in our experiments costs \$0.0200 per 1K tokens¹. When the cost of pre-training such models is considered, it extends well beyond the cost of pre-training and GPT2-based methods. For example, to run GPT3, we need at least 5 80GB A100 GPUs², while GPT2 runs smoothly on a single 12GB GPU. As such, it would be incredibly beneficial if smaller-scale LLMs could be used on specific tasks with reasonable success. Achieving such performances with smaller-scale LLMs could be the direction towards sustainable AI [Sadat Moosavi et al., 2023].

8.2.4 Limitations

In this section, we outline several limitations of LLM-based approaches to user simulation for conversational search and task-oriented dialogue systems.

Simplified User Model

Balog [2021] argues that persona is an important part of user simulation. However, neither USi and ConvSim explicitly model persona and are not grounded in a specific user model. While this follows the paradigm presented through the series of TREC CAsT [Dalton et al., 2020, Owoicho et al., 2022], where conversations on a topic are presented without any notion of the user, it does not fully encapsulate a realistic scenario. Thus, the next generation of user simulators for MICS should incorporate a user model and adjust to the given user persona. Recently developed TREC iKAT [Aliannejadi et al., 2024] potentially offers the basis for such research, as it models conversational passage retrieval, conditioned on user persona description.

¹Cost at the time of submission: <https://openai.com/pricing>.

²The exact compute required is not available as the model is closed-source. This is the estimate by the AI research community.

Hallucinations

LLMs are prone to hallucination [Ji et al., 2023]. In our analysis in Chapt. 7, we observed instances where our LLM-based simulator exhibit hallucinations. Despite being superior to in-context learning approaches like those presented in Terragni et al. [2023], we still encountered cases of LLM responses that deviated from the expected or coherent output. These hallucinations may lead to unpredictable and potentially inappropriate responses in certain conversational contexts, raising concerns about the reliability and safety of such systems.

It's a Black Box

Another limitation of an LLM-based approach is that it is essentially a black box. The simulator makes decisions based solely on its understanding of the task, conditioned on the weights of the neural network composing the LLM. No expert knowledge, policy, contextualized state, or other rules are taken into account. The advantage of agenda-based simulator is that the steps taken, actions and decisions made are directly traceable. While the reasoning capabilities of LLMs are becoming increasingly better, as demonstrated in this dissertation, future work calls for a necessity of potentially combining rule-based, i.e., agenda-based, approaches with LLMs [Bernard, 2023].

Closed-Source LLMs

The approaches presented in Chapt. 6 and 7 has several inherent limitations, primarily stemming from the use of LLMs. Most notably, GPT-3.5, the model we utilized in our experiments, is not open-source and freely available, which can hinder replicability of the experiments. Another limitation is related to the opaqueness of the model's training and fine-tuning processes. These models undergo pre-training and fine-tuning on diverse datasets, the specifics of which are often undisclosed. Consequently, it is challenging to ascertain whether these models have been exposed to specific datasets, such as MultiWOZ 2.1, or datasets with similar characteristics, which could raise concerns about models performance and potential biases.

Fine-tuning Data

The methodology presented in Chapt. 7 relies on conversational data for fine-tuning LLMs. This reliance introduces additional limitations. Firstly, obtaining

suitable conversational data may be challenging or even unfeasible in some scenarios. Researchers may resort to crowd-sourcing tools to gather dialogue examples or use LLMs themselves to generate synthetic data, which could introduce biases or inaccuracies. Secondly, the quality of the conversational data used for fine-tuning plays a pivotal role in the model's performance. In our study, we utilized well-curated conversational data, but we did not investigate the impact of using noisier or less meticulously curated data. The use of lower-quality data sources may affect the model's performance and raise questions about its reliability and robustness in real-world applications.

Chapter 9

Conclusions

In this chapter, we conclude this dissertation by summarizing the carried out work (Sect. 9.1) and listing our main contributions (Sect. 9.2). Section 9.3 outlines possible continuations of our work.

9.1 Summary

The main motivation for the work carried out in this dissertation was to make steps towards the next generation of conversational systems. It is thus primarily concerned with mixed-initiative conversational search, with a focus on clarification and user simulation.

Part I covered several studies related to mixed-initiative conversational search. First, we proposed an LLM-based clarifying question generation method. The method, given an initial query and query facets, outputs coherent and natural questions, that can then be prompted to the user with a goal of elucidating its information need. However, as discussed in Chapt. 3, acquiring the appropriate query facets is not straightforward. To shed light on the problem, we explored the possibility of extracting query facets from the list of documents retrieved in response to the initial query. We found that, while term- and topic modeling-based methods indeed contain such facets, their extraction from noisy document sets is challenging. We additionally briefly explore the potential of generating query facets via LLM-based methods. To conclude the chapter, we propose a novel method for processing answers to clarifying questions with the purpose of improving the retrieval effectiveness of MICS systems. The method is based in predicting usefulness of the posed question and a given answer, where the initial query is enriched with either (or both) if deemed useful, i.e., if it carries additional information that clarifies the information need.

Second, in Chapt. 4 we proposed a task to predict engagement level with clarification panes in a traditional search setting. We cast the task to a regression problem, where engagement level is indicated by a real number between 0 and 1. Experiments on MIMICS dataset demonstrate the superiority of LLM-based methods over traditional ML methods for the task. Moreover, we find that including information from the documents retrieved in response to the initial query increases the performance, once again indicating their importance.

As motivated throughout this document, conversational information access systems require appropriate interfaces for user interaction. Generating concise system responses is considered an essential component of CIA systems. Chapter 5 thus presented our method for generating self-contained system responses to users' questions. To this end, we based our method in salient entities (i.e., entities crucial for understanding of a given text) and proposed to rewrite the original response by either providing in-line explanations of identified salient entities or offering to clarify them in case they are unfamiliar to the user. Through a crowdsourcing-based experiment, we found strong preference of the rewritten response over the original ones. However, we also identified high subjectivity in the preference of the rewrite type, thus motivating further research on personalized response rewriting.

Part II dealt with user simulation for conversational search and task-oriented dialogue systems. We first motivated the necessity of user simulation for evaluation of conversational systems due to expensive and time-consuming user studies and unrealistic offline evaluation procedures. In Chapt. 6, we first proposed a method capable of assuming a user's role in the interaction with MICS system by answering prompted clarifying questions. We showed that the generated answers are both in line with the given information need description and coherent. However, the proposed method had significant limitations as it was limited to single-turn conversations and only to answering clarifying questions. To mitigate these limitations, we proposed an in-context learning approach to generate answers in a multi-turn setting. Additionally, the proposed method is capable of generating explicit feedback to system's responses. We showcased the benefit of exploiting user's feedback, both in terms of answers to question and explicit feedback, to the system's retrieval effectiveness.

While this dissertation was mostly centered around conversational search, Chapt. 7 instead presented user simulation approach for task-oriented dialogue systems. There, instead of user's information need description, the simulator was given user's goal description, which included e.g., booking an appointment for oil change. We motivated the benefits of simulation approaches for TOD system evaluation and proposed a domain-aware simulator for the task. Specif-

ically, our proposed method relies on state-of-the-art LLMs and is fine-tuned on carefully curated in-domain data. We demonstrated the preeminence of the proposed method compared to previously proposed in-context learning approaches. Our proposed simulator exhibited significantly less hallucinations, thus making it reliable and more useful for evaluating TOD systems.

Finally, in Chapt. 8, we discussed the main findings of the dissertation. Through the discussion, we aggregated the findings from several studies and drew conclusions. We additionally pointed out several limitations of our research on mixed-initiative conversational systems. Moreover, we delineate the advantages and limitations of LLM-based user simulation, comparing the fine-tuning and in-context learning approaches in terms of effectiveness, cost, and scalability.

In the next section, we summarize the main contributions of the dissertation. Section 9.3 finally outlines plausible directions for future research.

9.2 Main Contributions

In Part I, we presented our research on mixed-initiative conversational search. Our contributions can be summarized as follows:

- We proposed a facet-based clarifying question generation method. To this end, we fine-tuned GPT-2 to generate the questions, given the original query and its facet as input. Crowdsourcing-based evaluation showed that our method generates more natural questions than previously proposed template-based methods, while retaining the usefulness of a question (i.e., the model does not hallucinate and the question is on the topic of a given facet).
- We explored the possibility of extracting query facets for constructing clarifying questions from a list of documents retrieved in response to a query. Our approach, based in term frequency and topic modeling methods, shows a promising direction for collection-grounded facet extraction.
- We formally introduce the clarification pane engagement level prediction task as supervised regression and propose a transformer-based model to tackle it. We make the code publicly available for reproducibility purposes and perform detailed analysis of the performance of our model w.r.t. various characteristics of the search engine result page.

We additionally contributed towards the challenge of providing appropriate responses in conversational information access systems:

- We proposed to rewrite the original response in CIA to make it self-contained. We would therefore provide in-line explanations of salient entities, i.e., entities deemed crucial for understanding of the response, or offer the user to learn more about them by asking a follow-up question.
- We performed a crowdsourcing-based study to investigate human preference of the rewritten answers compared to the original answers. We acquired specific reasons for the rewrite type preference and release them to foster future research on the topic.
- We annotated a sample of 360 question-answer pairs to characterize entity saliency in CIA. We released the annotated data.

In Part II, we proposed several LLM-based user simulation approaches for conversational systems. In particular:

- We proposed a user simulator, USi, for conversational search system evaluation, capable of answering clarifying questions prompted by the search system. We release the code and pre-trained USi.¹
- We addressed the limitations of USi, and designed a user simulator capable of multi-turn interactions with mixed-initiative search systems. We released transcripts, code and guidelines² to foster further research.
- We released a dataset of multi-turn interactions acquired through crowdsourcing, that we use to train our multi-turn version of the model. The dataset consists of 1000 conversations of up to three turns, where crowdsourcing workers played the roles of the system that asks clarifying questions and the user seeking information.
- We performed extensive set of experiments to evaluate the feasibility of substituting real users with the simulators, shedding new light on the upper bound of large-scale language models for the task. To this end, we compared our fine-tuning and our in-context learning approaches, and discussed the advantages and limitations of both.
- We proposed a reliable user simulator for task-oriented dialogue systems. Our method, DAUS, utilizes state-of-the-art LLMs with performance-efficient fine-tuning on carefully curated in-domain data, thus outperforming previous in-context learning approaches for the task.

¹<https://github.com/isekulic/USi>

²<https://github.com/grill-lab/ConvSim>

9.3 Future Directions

In this section, we outline the possibilities of extending the work described in this thesis. We first discuss about expanding on mixed-initiative approaches in conversational search, focusing on clarification, but also moving beyond clarifying questions. Next, we describe future work on user simulation, including the expansion to traditional IR datasets, parametrizing user simulators, and error detection.

Beyond Single-faceted Clarifying Questions

In this dissertation, we viewed clarification mostly through the lens of single-faceted clarifying questions. As discussed in Sect. 8.1.2, asking questions about multiple facets at once might help elucidate user's information need faster and in less turns. Thus, future work includes development of multi-faceted clarifying questions for conversational search. Moreover, there is a necessity for studies on selecting appropriate, and an appropriate number of, facets. While certain studies exploring utility and cost-benefit ratio of clarifying questions exist [Azzopardi et al., 2022], a well-designed solution is still lacking.

Reinvigorating Traditional IR Datasets

Our user simulation approaches presented in Chapt. 6 are initialized with information need description. Once initialized, the simulator is capable of multi-turn interactions with the search system. As information need descriptions are often a part of traditional, static IR datasets, such as TREC-style datasets, it is possible to make them conversational via user simulation. This includes adapting the approach discussed in Sect. 6.4 to the new collections and confirming the reliability of the utterances generated by user simulators for the new data.

Making static IR datasets conversational would allow the IR research community to conduct contemporary research whilst making use of decades of well-established collections.

Extracting Information Need from Relevant Documents

Touching on the above-described idea, in absence of information need descriptions, it is possible to extract them from the query relevance judgments. Every IR dataset contains queries and annotated corresponding relevant documents, which satisfy the information need expressed through a query. Thus, designing

a function that assesses whether information need was satisfied by system’s response would involve comparing the response to the documents that are deemed highly relevant. This comparison can be done via LLMs. Moreover, such task can be modeled as a classification task, either binary, i.e., information need was satisfied or not, or graded, indicating the degree to which it was fulfilled. A step further would encapsulate generating explicit feedback aimed to assess which parts of the information need were, or, probably more importantly, weren’t satisfied. This could be modeled by iterating through a list of relevant documents and comparing them to the system’s response.

Parametrized User Simulator

A realistic user simulator should closely follow the behaviors of real users [Balog, 2021]. To achieve that, we need both an underlying user model we design our simulators upon and more control over the types of utterances generated by the simulators. A solution towards this goal lies in parameterized user simulators.

Parameterized user simulators allow for adjustment towards certain types of users. For example, Salle et al. [2021] model cooperativeness, i.e. how lengthy and informative the simulator responses are, and patience, i.e. how many turns of answering clarifying questions is the simulator willing to partake in before giving up. Similarly, Owoicho et al. [2023] propose a GPT3-based simulator with a model for its patience. However, many more parameters could be introduced, allowing for fine-grade user models. Also, parameters’ scalability could be aimed at, as previous approaches solely rely on pre-constructed simulator responses, thus limiting its applicability in real-world scenarios [Salle et al., 2021].

To address the aforementioned limitations, we propose ParaConvSim a user simulator capable of assuming a user’s role with certain behavior traits, namely: patience, cooperativeness, and politeness. Unlike previous work, that assumes users are always at one of the extremes on the spectrum (e.g., always impolite or always polite), at each conversational turn, we draw on a Bernoulli variable based on the set parameter to decide the kind of utterance that should be generated. This ensure the consistency of the generated utterances with in-context learning and state-of-the-art LLMs.

This work has been submitted to SIGIR’24 as a full paper. However, due to time limitations, we do not include the study in the main part of the dissertation.

Error Detection in Conversational Systems

Chapter 7 demonstrated the use of user simulation to detect failures in TOD system. Failures can be detected by allowing multiple interactions between the simulator and the system, and measuring the success of each of the interactions. Then, given a reliable simulator, conversations that do not fulfill given user goal can be considered to fail due to system's failure (e.g., missed intent).

Moreover, we can identify break points in conversational systems by purposely generating faulty utterances, thus probing the search system's robustness. This would additionally test system's ability to recover and continue the conversation in the desired direction. On a similar note, we might design a simulator that performs a sudden topic shift, and indicate what kind of actions would be expected of the search system. For example, widely used contextual query rewriters might struggle with sudden topic shifts.

Synthetic Data and Reinforcement Learning

User simulators can be used for evaluating conversational systems, but also for creating synthetic data to be used for downstream tasks [Meyer et al., 2022]. For example, recently, Abbasiantaeb et al. [2023] utilize LLMs to generate synthetic dialogue data for conversational question answering. Moreover, recent research demonstrated the power of reinforcement learning methods for conversational search [Wang and Ai, 2021]. User simulation can as well play a crucial role in the development of reinforcement learning approaches for conversational systems by providing means to generate training data and evaluate the performance in a simulated environment. Such approaches have been explored by Zhan et al. [2023] in the context of conversational recommender systems, where the system's reward is based on the user simulator's feedback. Future work thus encapsulates exploring the possibilities of training novel methods, both through synthetic data and reinforcement techniques, with the user simulators for conversational search, proposed in this dissertation.

Appendix A

Graph-Based Approaches to IR

In this chapter, we describe a novel, but unfinished, graph-based approach to IR, including graph-based conversational search. It contains parts of the unpublished work done in [Sekulić et al. \[2021\]](#) and [Mackie et al. \[2021\]](#). As detailed in the section, the work, although deemed promising, was abandoned due to unpromising results in the initial experimental phase and the lack of time to properly pursue the complex graph-based approaches for modeling conversational search. Thus, we stopped the pursuit of this direction of my PhD and only outline the vision for graph-based conversational search in Sect. [A.3](#).

A.1 Multi-document Graph-based IR

As mentioned in [Chapt. 2](#), *ad hoc* document retrieval is a central task of IR, where given a query the IR system retrieves a list of documents ordered by its relevance from a large collection. Neural IR models played an important role in the document re-ranking step, significantly outperforming previous approaches [[Mitra et al., 2017](#)]. Recent rise of pretrained transformer-based language models, such as BERT [[Devlin et al., 2019](#)], brought unprecedented success in document re-ranking tasks [[Nogueira and Cho, 2019](#)]. However, such large-scale models are limited by their input size, which restricts them into assessing the relevance of a single document at a time.

To alleviate the aforementioned shortcoming, we propose a novel graph-based approach, able to utilize information from multiple documents at the same time, assessing their relevance simultaneously. The graph is constructed from the user’s query and the documents retrieved by an initial retriever (e.g., BM25). Nodes in a graph are words, entities, or other tokens extracted from the texts. We also include special query and document nodes that aim to capture the impor-

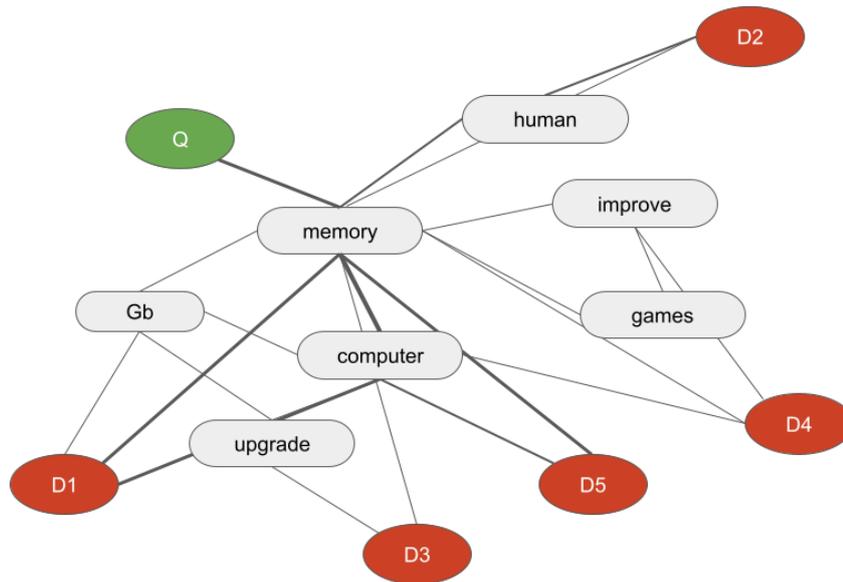


Figure A.1. An illustrative subgraph extracted from the graph created from the user’s query “Tell me about memory” (Q) and the retrieved documents (D_i).

tant high-level information. The edge between two tokens in the graph captures token co-occurrence information, while edges between other types of nodes correspond to a priori relatedness of different concepts, as described in Sect. A.1.1. Graph neural network (GNN) is then applied to the constructed graph to learn the relevance of document nodes. We model document re-ranking task as node classification problem.

We conduct experiments on MS MARCO passage re-ranking task, showcasing the feasibility of our approach. Furthermore, we show the impact of various graph elements, such as node type features and edge weights, on the overall performance. Finally, we discuss potential improvements and applications of the approach to the other areas in IR.

While several approaches to graph construction from text exist in the NLP community, they are often too fine-grained for IR applications, as we deal with tens or hundreds of potentially very large documents at a time. For example, several studies show the benefit of constructing the graph from parsed sentences (e.g., dependency trees) for tasks dealing with sentence-level predictions, or at best with relatively short passages [Cui et al., 2022]. However, due to large noise levels and increased complexity, such a fine-grained approach is not fit for our setting, where the graph is constructed from queries and up to hundreds of text documents, each containing multiple sentences. Thus, we take on a higher-level

approach, where a token appearing in multiple documents is considered as a single node, significantly reducing the number of nodes and edges in a graph, while improving robustness. To do so, we take inspiration from ?, who construct a single large graph from a collection of documents for text classification purposes.

The graph is constructed from tokens from a user’s query and the documents retrieved by an initial retriever, and can be easily expanded with any other relevant information. Nodes in a graph are tokens extracted from the aforementioned texts, while edges represent meaningful connections between the words, weighted by co-occurrence information or tf-idf. We describe our graph construction approach in great detail in the next subsection.

A.1.1 Graph Construction

We propose to create a graph from a user’s query and multiple documents retrieved by an initial retrieval method (e.g., BM25). The nodes in a graph are words, tokens, and entities extracted from these texts, while edges represent co-occurrence-based relations between them. Moreover, we construct special high-level nodes, representing queries and documents.

Text Processing

The aforementioned texts are pre-processed by common NLP pipeline. Specifically, we employ sentence and word tokenizers, followed by stemming and lower-casing. Finally, we acquire a pre-trained word embedding for each of the extracted tokens. Additionally, we experiment with different filters designed to keep only the most informative tokens in the graph. Namely, we propose frequency-based filtering, which disregards tokens with frequency below a certain threshold, and Part-Of-Speech (POS) based filtering. In the latter, we extract only specific potentially content-bearing tokens from the texts, such as nouns or named entities. We employ SpaCy [Honnibal et al., 2020] for all of the text processing.

Nodes

We differentiate three types of nodes:

Token nodes are extracted from texts as described in previous subsection;

Document nodes are connected to all of the tokens extracted from the documents and serve a high-level representation of the documents;

Query nodes represent individual queries and are connected to all of the tokens extracted from the query, as well as to all of the document nodes retrieved from a collection.

All nodes are represented by a N -dimensional vector, regardless of the type. Formally, let \mathbf{u}_i denote contextual representation of a node i , where $\mathbf{u}_i \in \mathbb{R}^N$. The representation \mathbf{u}_i , in case the node i is a token, is a word vector taken from SpaCy *en_core_web_md* pipeline. For other type of nodes, the contextualised representation is computed by averaging representation of all token nodes present in the type of text we are dealing with. For example, to compute the initial representation of a single document node, we average word representations of all the words present in the document.

Edges

Edges between the nodes differ depending on what kind of node types they are connecting. We define the weight between node i and node j as follows:

$$A_{ij} = \begin{cases} PMI(i, j) & i, j \text{ are tokens,} \\ TF\text{-}IDF_{ij} & i \text{ is special node, } j \text{ is word} \\ 1 & i, j \text{ are special nodes} \\ 1 & i = j \\ 0 & \text{otherwise} \end{cases} \quad (\text{A.1})$$

where we define special nodes as either document or query nodes.

We compute point-wise mutual information (PMI) between token nodes to capture word co-occurrence information. Formally, this is defined as:

$$PMI(i, j) = \log \frac{p(i, j)}{p(i)p(j)} \quad (\text{A.2})$$

where $p(i, j) = \frac{\#S(i, j)}{\#S}$ and $p(i) = \frac{\#S(i)}{\#S}$. Function $S(k)$ represents the number of times a token i , or both tokens i and j have appeared in the same sentence, while $\#S$ is a number of total sentences in the considered texts.

A.1.2 Graph Analysis

We analyze several properties of our graph, such as average node degree, number of nodes/edges, and number of cliques, in order to gain further insight into the approach. First, we inspect the size of our graphs depended on the number of

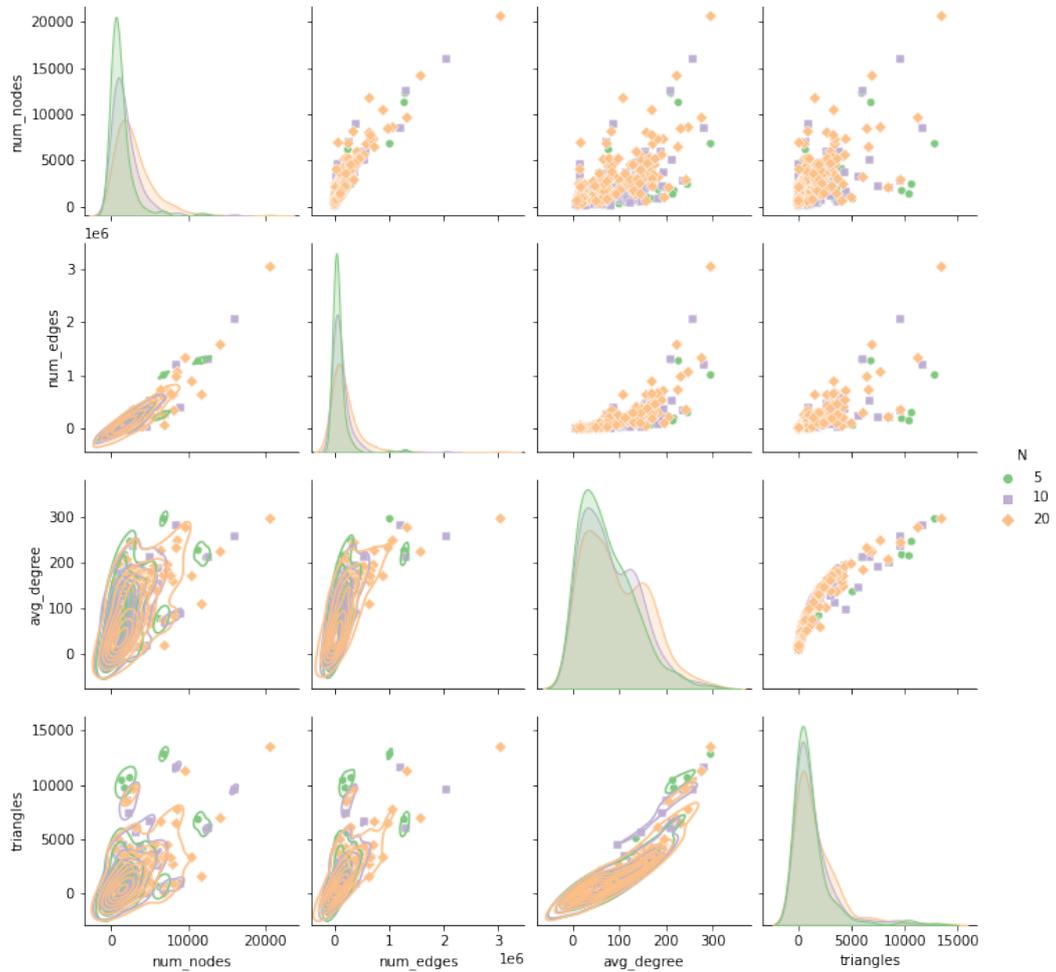


Figure A.2. Pairplot of several aspects of the created graph (number of nodes, number of edges, average degree, number of triangles), varied by the number of documents the graph is created from.

documents we retrieve by our initial retrieval method. Figure A.2 shows that both the number of nodes and the number of edges increases sublinearly with the increasing number of documents we create our graphs from. This is due to the words between documents overlapping, which does not create new nodes in the graph. Consequently, we are able to utilize tens or potentially even hundreds of documents for graph creation at a time, unlike recent transformer-based models where the input size is limited to several hundreds of tokens only, which might not even include a single full document.

Moreover, we observe that the average degree of the nodes, i.e., the average number of neighboring nodes, does not change significantly with the increase in number of documents we create our graph from. Similar observation is made concerning number of triangles (3-vertex cliques) in the graph, i.e., number of three nodes interconnected with each other. Furthermore, clear relations between number of triangles or degree, and number of nodes and edges can be seen.

A.1.3 The Graph Model

Our re-ranking GNN is inspired by the design principles described by ?. The architecture consists of an initial multi-layer perceptron (MLP) applied node-wise, followed by a block of message-passing layers and a final node-level MLP to compute the final node predictions.

The two MLP consist of two layers of 512 units (except for the last layer of the final MLP, which is a simple projection to the target space), with batch normalization and a ReLU activation. The message passing block consists of four layers with 256 channels, batch normalization and ReLU activation. Let $\mathbf{h}_i^l \in \mathbb{R}^{d_l}$ be the feature of the i th node computed by the l th layer, with $\mathbf{h}_i^0 = \mathbf{u}_i$. Each layer works according to the following message-passing scheme:

$$\mathbf{h}_i^{l+1} = \mathbf{h}_i^l + \sum_{j \in \mathcal{N}(i)} (\mathbf{W}^l \mathbf{h}_j^l + \mathbf{b}^l) + (\mathbf{A}_{ji} \mathbf{v}^l + \mathbf{c}^l) \quad (\text{A.3})$$

where $\mathbf{W}^l \in \mathbb{R}^{d_{l+1} \times d_l}$ and $\mathbf{v}^l, \mathbf{b}^l, \mathbf{c}^l \in \mathbb{R}^{d_{l+1}}$ are learnable parameters. For the skip connection of the first layer, we use a learnable linear projection to match the size of the input to the output.

We train the model using Adam with learning rate of 1×10^{-5} with a batch size of 28 and clip the gradients to have norm 0.5. All hyperparameters were found through a random search over a total of 30 configurations, keeping the best model based on the dev loss.

A.1.4 Results and Analysis

Data

Microsoft Machine Reading Comprehension (MS MARCO) [Nguyen et al., 2016] is a collection of large-scale datasets focused on question answering, machine reading comprehension, and passage re-ranking. We focus on the passage re-ranking MS MARCO dataset, which is the largest humanly-labeled dataset for *ad hoc* retrieval to date (Dec 2021). It contains more than 500k queries with associated relevance labels and the collection of 8.8M passages. Given a user’s query q , we need to retrieve and rank documents by decreasing relevance to the query. In this setting, we employ a two-step retrieve and re-rank approach, where we first use a computationally inexpensive method, specifically BM25, followed by our graph-based re-ranker to re-rank the top N retrieved documents. As this initial study aims to make the first steps towards employing GNNs over graphs extracted from multiple documents, we ensure that at least one relevant document is present in the training/dev set. This ensures that the model is indeed able to learn the difference between the relevant and non-relevant documents, as relevant documents are not often in the top ranked ones retrieved by an initial retrieval method.

Results

Table A.1 shows the performance in terms of mean reciprocal rank (MRR) over the baseline on MS MARCO development test. We notice our proposed method outperforms the BM25 baseline when fed with all of the existing features. While the margins are small, they are consistent and statistically significant.

Ablation Studies

With a goal of gaining further insight into our model, we perform a series of ablation studies. First, we explore the impact of different features extracted from the queries and the documents, namely *nouns*, *entities*, and *all* of the tokens. Second, we perform ablation study on the graph-based features made available to our model. Specifically, we experiment with the basic GNN described in A.1.3, as well as its improved versions that utilise node type features and edge weights.

Results presented in Table A.2 suggest that GNNs based on *all* of the extracted tokens from the queries and the documents perform better than the ones trained on *nouns*-only. Although the graphs created with *all* of the tokens significantly exceed the ones created from *nouns* in size, the information contained in them

Table A.1. Results on MS MARCO development set based on different features extracted.

Model	Mean Reciprocal Rank
BM25	0.26
GNN-nouns	0.33
GNN-entities	0.31
GNN-all	0.36

Table A.2. Ablation study on MS MARCO development set. The sign “-” indicates that these features have been removed from the model.

Model	Mean Reciprocal Rank
Full model	0.36
- Node Types	0.33
- Edge Weights	0.34
- Node&Edge	0.31

outweighs the potential noise. Moreover, we observe clear improvement when our proposed GNNs utilised node type features, as described in the Section A.1.3. This information is important to the GNN, as it provides a focus on which nodes are special nodes and which require specific predictions, such as document relevance. Similarly, edge features improve the performance of the GNNs. Edge features provide a weight of the relation, indicating its co-occurrence relevancy between the two nodes.

A.2 Exploring Complex Topics

In this section, we briefly mention our work towards a probabilistic graph construction framework for entity-centric document ranking. Complex information needs require a long-form answer to satisfy the user request, which would contain critical information spanning multiple documents and entities [Mackie et al., 2022]. Unlike factoid questions, where the relevant information is often contained within a single text passage, complex information needs require reasoning and contextualization across many sources and knowledge.

In order to study the complex relations required for answering such topics, our graph construction method utilizes entities from queries and multiple doc-

uments. We analyze various properties of the constructed entity-centric graphs, such as graph density and transitivity, and their impact on graph quality. Moreover, we assess their feasibility for representing document ranking on complex topics, as well as other IR tasks requiring aggregation of information across multiple documents. We find that graphs capture important relations between relevant documents and entities; however, graphs become more sparse as graph size increases and graph construction quality decreases. We develop a probabilistic graph construction framework using the CODEC dataset [Mackie et al., 2022]. CODEC provides complex essay-style topics and aligned document and entity judgments. We use state-of-the-art entity linkers on both documents [van Hulst et al., 2020] and queries [van Hulst et al., 2020, Wu et al., 2020] to provide structured connections for entity-centric document ranking. This allows us to develop query-specific ‘golden graphs’ where nodes are relevant entities and documents, and edges are connections between relevant entities and documents. We simulate the quality of graph construction through probabilistically sampling relevant and non-relevant nodes, which allows us to explore both the size and quality of graphs.

We find that graphs are capable of capturing cross-document complex relations between entities. Yet, we highlight the graph construction challenges IR must tackle, finding that graphs become more sparse as size increases and construction quality reduces. Moreover, we show that certain intrinsic graph properties, such as graph density and transitivity, can be used as features and guidance for future GNN-based applications. For example, node centrality-based pruning can increase the precision of the graph, while keeping only a fraction of the initial nodes and edges. This provides a strong motivation for pursuing graph-based approaches for IR applications.

To conclude, we performed detailed analysis of various graph construction methods and graph properties, with a goal of finding entities central to complex topic of the initial query. While certain properties and pruning techniques indicated promising research trajectories, results on downstream tasks, i.e., document and entity ranking, did not yield satisfactory outcomes. As the pursuit of this graph-based approach required significant time efforts, as well as overcoming the steep learning curve of graph neural network methods, in this dissertation, we do not bring the matter to the final conclusion.

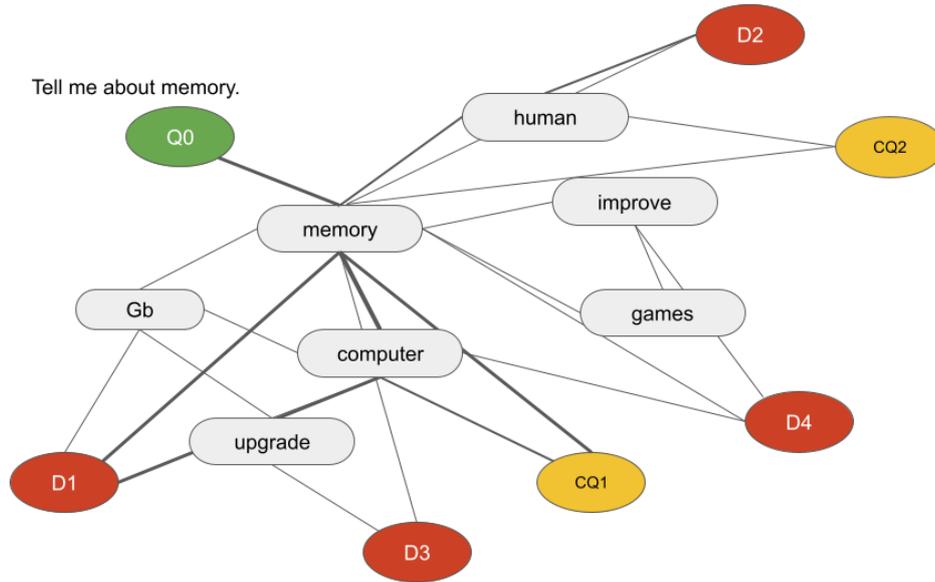


Figure A.3. An illustrative subgraph extracted from the graph created from the initial query “Tell me about memory” (Q_0), the retrieved documents (D_i), and potential clarifying questions (CQ_j).

A.3 Extension to Conversational Search

In this section, we discuss a graph-based approach to several conversational search tasks. The graph is constructed from the user’s query, the documents retrieved by an initial retriever (e.g., BM25), and potential clarifying questions. Graph construction methods rely on the ones outlined in Sect. A.1, where nodes in a graph are words, entities, or other tokens extracted from the aforementioned texts. We also construct higher-level query nodes, document nodes, and clarifying question nodes, that aim to capture the important high-level information. The edge between two tokens in a graph captures token co-occurrence information, while edges between other types of nodes correspond to a priori relatedness of different concepts.

Such holistic graph representation of the conversational search session, would allow us to tackle several tasks:

- Coreference resolution: all of the entities mentioned in the conversational history are represented in a single graph, making it possible to resolve potential ambiguity of the current utterances;
- Topic shift prediction: if the graph designed from the current turn differs

greatly from the graph representing conversational history, it might indicate a topic shift;

- Document relevance assessment: approach similar to the one outlined in Sect. A.1;
- Clarification need prediction: based on graph density, diversity, and other properties, we can decide whether to ask a clarifying question or not;
- Clarifying question selection: appropriate clarifying questions can be selected based on the graph representation, or generated based on certain entities deemed central to the graph.

A strong advantage of our modeling approach is that all of the tasks can be learned jointly on a single graph, benefiting from each other in the process.

A.4 Discussion

While the previous section suggests promising results for document re-ranking with the proposed graph-based method, we believe there is still room for improvement. In this initial study, we explored the space and complexity of graphs created from several documents. However, a more detailed analysis of some aspects of the created graphs is left for a future study. For example, our graph, created from tens of documents, might benefit from node pruning, as perhaps not all of the tokens in the documents are informative for assessing document relevance. We plan to explore node pruning methods based on token frequency or tf-idf features. We hypothesize less nodes would still allow the model to properly learn the relevance of specific document nodes, while further reducing the computational requirements of our already computationally inexpensive method. In fact, the model might perform even better, as there would potentially be less noise present.

Another possible improvement could come from not representing nodes with static pre-trained word embeddings, but rather make them learnable through a proper embedding layer. Similarly, we aim to explore additional upgrades such as LSTM- and BERT-based text encoding layers, with a goal of acquiring the best token representation possible.

Last, but not least, we stress the knowledge-grounding aspect of our proposed model. The graph created from multiple documents can serve as an online-created knowledge graph, which can prove to be beneficial to various aspects of IR, not just for the task of document re-ranking.

While our work in the late 2021 and early 2022 did not result in published papers, we note that graph-based approaches have since been successfully utilized for different tasks in conversational and *ad hoc* search by the research community. For example, [Gao and Lam \[2022\]](#) propose a query-intent-clarification (QIC) graph with an aim of selecting the appropriate clarification prompt in search engines. They show that the graph structure successfully exploits the relations among a given query, its intents, and its clarifications, outperforming relevant baselines.

Close to our work on multi-document IR addressed in Sect. A.1, [Gonçalves et al. \[2023\]](#) propose a method for conversational search based on random walks over entity graphs. They construct an entity graph based on top-ranked passages retrieved by an initial retrieval method. Next, they compute node centrality scores, indicating the importance of each entity, which help re-rank the the passages for improved top precision. As stated by the authors, the improved retrieval performance is modest, but makes up for it by being computationally inexpensive and interpretable, the concepts we advocated for as well.

On a different note, [Schneider et al. \[2023\]](#) propose a knowledge-graph based approach for exploratory conversational search. This work successfully showcases the potential of graphs for the aforementioned conversational state tracking task. Specifically, they design a knowledge-driven conversational system for exploring news articles, where the proposed graph structure allows for effective navigation between different topics.

Bibliography

- Zahra Abbasiantaeb, Yifei Yuan, Evangelos Kanoulas, and Mohammad Aliannejadi. Let the llms talk: Simulating human-to-human conversational qa via zero-shot llm-to-llm interactions. *arXiv preprint arXiv:2312.02913*, 2023.
- Jafar Afzali, Aleksander Mark Drzewiecki, Krisztian Balog, and Shuo Zhang. UserSimCRS: A user simulation toolkit for evaluating conversational recommender systems. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM '23*, pages 1160–1163, 2023.
- Mohammad Aliannejadi, Morgan Harvey, Luca Costa, Matthew Pointon, and Fabio Crestani. Understanding mobile search task relevance and user behaviour in context. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, CHIIR'19*, pages 143–151, 2019a.
- Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19*, pages 475–484, 2019b.
- Mohammad Aliannejadi, Manajit Chakraborty, Esteban Andrés Ríssola, and Fabio Crestani. Harnessing evolution of multi-turn conversations for effective answer retrieval. In *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval*, pages 33–42, 2020a.
- Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. ConvAI3: Generating clarifying questions for open-domain dialogue systems (ClariQ). *arXiv preprint arXiv:2009.11352*, 2020b.
- Mohammad Aliannejadi, Leif Azzopardi, Hamed Zamani, Evangelos Kanoulas, Paul Thomas, and Nick Craswell. Analysing mixed initiatives and search strategies during conversational search. In *CIKM '21: The 30th ACM International*

- Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021, CIKM'21*, pages 16–26, 2021a.
- Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. Building and evaluating open-domain dialogue corpora with clarifying questions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP'21*, pages 4473–4484, 2021b.
- Mohammad Aliannejadi, Zahra Abbasiantaeb, Shubham Chatterjee, Jeffery Dalton, and Leif Azzopardi. Trec ikat 2023: The interactive knowledge assistance track overview. *arXiv preprint arXiv:2401.01330*, 2024.
- Ghadah Alkhalidi, Fiona L Hamilton, Rosa Lau, Rosie Webster, Susan Michie, and Elizabeth Murray. The effectiveness of prompts to promote engagement with digital interventions: a systematic review. *Journal of medical Internet research*, 18(1):e6, 2016.
- James E Allen, Curry I Guinn, and Eric Horvitz. Mixed-initiative interaction. *IEEE Intelligent Systems and their Applications*, 14(5):14–23, 1999.
- Avishek Anand, Lawrence Cavedon, Hideo Joho, Mark Sanderson, and Benno Stein. Conversational search (Dagstuhl Seminar 19461). In *Dagstuhl Reports*, volume 9, 2020.
- Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. Open-domain question answering goes conversational via question rewriting. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL'21*, pages 520–534, 2021.
- Layla El Asri, Jing He, and Kaheer Suleman. A sequence-to-sequence model for user simulation in spoken dialogue systems. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association*, pages 1151–1155, 2016.
- Leif Azzopardi. The economics in interactive information retrieval. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, SIGIR '11*, pages 15–24, 2011.
- Leif Azzopardi, Mateusz Dubiel, Martin Halvey, and Jeffery Dalton. Conceptualizing agent-human interactions during the conversational search process. In *The Second International Workshop on Conversational Approaches to Information Retrieval, CAIR'18*, 2018.

- Leif Azzopardi, Mohammad Aliannejadi, and Evangelos Kanoulas. Towards building economic models of conversational search. In *European Conference on Information Retrieval, ECIR'22*, pages 31–38, 2022.
- Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. MS MARCO: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*, 2016.
- Krisztian Balog. *Entity-oriented search*. Springer Nature, 2018.
- Krisztian Balog. Conversational AI from an information retrieval perspective: Remaining challenges and a case for user simulation. In *Proceedings of the 2nd International Conference on Design of Experimental Search and Information REtrieval Systems, DESIRES '21*, pages 80–90, 2021.
- Krisztian Balog and ChengXiang Zhai. User simulation for evaluating information access systems. *arXiv preprint arXiv:2306.08550*, 2023.
- Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.
- Nicholas J Belkin, Colleen Cool, Adelheit Stein, and Ulrich Thiel. Cases, scripts, and information-seeking strategies: On the design of interactive information retrieval systems. *Expert systems with applications*, 9(3):379–395, 1995.
- Anja Belz and Ehud Reiter. Comparing automatic and human evaluation of nlg systems. In *11th Conference of the European Chapter of the Association for Computational Linguistics, ACL '06*, pages 313–320, 2006.
- Nolwenn Bernard. Leveraging user simulation to develop and evaluate conversational information access agents. *arXiv preprint arXiv:2312.08041*, 2023.
- Janek Bevendorff, Benno Stein, Matthias Hagen, and Martin Potthast. Elastic chatnoir: Search engine for the clueweb and the common crawl. In *Advances in Information Retrieval: 40th European Conference on IR Research, ECIR '18*, pages 820–824, 2018.
- Guorui Bian, Michael McAleer, and Wing-Keung Wong. A trinomial test for paired data when there are many ties. *Mathematics and Computers in Simulation*, 81(6):1153–1160, 2011.

- Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. 2009.
- Christopher M Bishop. *Pattern Recognition and Machine Learning*. 2006.
- Alan W Black, Susanne Burger, Alistair Conkie, Helen Hastie, Simon Keizer, Oliver Lemon, Nicolas Merigaud, Gabriel Parent, Gabriel Schubiner, Blaise Thomson, et al. Spoken dialog challenge 2010: Comparison of live and control test results. In *SIGDIAL, SIGDIAL'11*, pages 2–7, 2011.
- David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.
- Paolo Boldi, Francesco Bonchi, Carlos Castillo, and Sebastiano Vigna. Query reformulation mining: models, patterns, and applications. *Information retrieval*, 14:257–289, 2011.
- Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. What do you mean exactly? Analyzing clarification questions in CQA. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval, CHIIR'17*, pages 345–348, 2017.
- Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- Paweł Budzianowski and Ivan Vulic. Hello, it's gpt-2-how can i help you? towards the use of pretrained language models for task-oriented dialogue systems. *EMNLP-IJCNLP 2019*, page 15, 2019.
- Chris Callison-Burch, Miles Osborne, and Philipp Koehn. Re-evaluating the role of bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics, EACL'06*, 2006.

- Arthur Câmara, Nirmal Roy, David Maxwell, and Claudia Hauff. Searching to learn with instructional scaffolding. In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, CHIIR '21, page 209–218, 2021.
- Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. 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*, SIGIR'08, pages 243–250, 2008.
- Gavin C Cawley and Nicola LC Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *The Journal of Machine Learning Research*, 11:2079–2107, 2010.
- Olivier Chapelle, Donald Metzler, Ya Zhang, and Pierre Grinspan. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and Knowledge Management*, CIKM '09, page 621–630, 2009.
- Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. Quac: Question answering in context. In *EMNLP*, EMNLP'18, pages 2174–2184, 2018.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.
- Aleksandr Chuklin, Aliaksei Severyn, Johanne R Trippas, Enrique Alfonseca, Hanna Silen, and Damiano Spina. Using audio transformations to improve comprehension in voice question answering. In *International Conference of*

- the Cross-Language Evaluation Forum for European Languages, CLEF '19*, pages 164–170, 2019.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022.
- Charles L. A. Clarke, Nick Craswell, and Ian Soboroff. Overview of the TREC 2009 web track. In *TREC*, 2009.
- Kevyn Collins-Thompson. Reducing the risk of query expansion via robust constrained optimization. In *CIKM*, pages 837–846, 2009.
- Michael D Cooper. A simulation model of an information retrieval system. *Information Storage and Retrieval*, 9(1):13–32, 1973.
- Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- W. Bruce Croft and R. H. Thompson. I³r: A new approach to the design of document retrieval systems. *JASIS*, 38(6):389–404, 1987.
- Hejie Cui, Jiaying Lu, Yao Ge, and Carl Yang. How can graph neural networks help document retrieval: A case study on cord19 with concept map generation. *arXiv preprint arXiv:2201.04672*, 2022.
- Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, 2019.
- Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. Trec cast 2019: The conversational assistance track overview. *arXiv preprint arXiv:2003.13624*, 2020.
- Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*, 2019.
- Sam Davidson, Salvatore Romeo, Raphael Shu, James Gung, Arshit Gupta, Saab Mansour, and Yi Zhang. User simulation with large language models for evaluating task-oriented dialogue. *arXiv preprint arXiv:2309.13233*, 2023.

- Marco De Boni and Suresh Manandhar. An analysis of clarification dialogue for question answering. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, page 48–55, 2003.
- Marco De Boni and Suresh Manandhar. Implementing clarification dialogues in open domain question answering. *Natural Language Engineering*, 11(4):343–361, 2005.
- Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echevoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review*, 54(1):755–810, 2021.
- Romain Deveaud, Eric SanJuan, and Patrice Bellot. Accurate and effective latent concept modeling for ad hoc information retrieval. *Document numérique*, 17(1):61–84, 2014.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.
- Abhinav Dhall, Amanjot Kaur, Roland Goecke, and Tom Gedeon. Emotiw 2018: Audio-video, student engagement and group-level affect prediction. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pages 653–656, 2018.
- Paul M Di Gangi and Molly M Wasko. Social media engagement theory: Exploring the influence of user engagement on social media usage. *Journal of Organizational and End User Computing (JOEUC)*, 28(2):53–73, 2016.
- Fernando Diaz and Jaime Arguello. Adaptation of offline vertical selection predictions in the presence of user feedback. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 323–330, 2009.
- Harris Drucker, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. Support vector regression machines. In *Advances in neural information processing systems*, NeurIPS '97, pages 155–161, 1997.

- Yuheng Du, Shereen Oraby, Vittorio Perera, Minmin Shen, Anjali Narayan-Chen, Tagyoung Chung, Anu Venkatesh, and Dilek Hakkani-Tur. Schema-guided natural language generation. *arXiv preprint arXiv:2005.05480*, 2020.
- Nouha Dziri, Andrea Madotto, Osmar Zaiane, and Avishek Joey Bose. Neural path hunter: Reducing hallucination in dialogue systems via path grounding. *arXiv preprint arXiv:2104.08455*, 2021.
- Wieland Eckert, Esther Levin, and Roberto Pieraccini. User modeling for spoken dialogue system evaluation. *1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings*, pages 80–87, 1997.
- Carsten Eickhoff, Jaime Teevan, Ryen White, and Susan Dumais. Lessons from the journey: A query log analysis of within-session learning. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining, WSDM '14*, page 223–232, 2014.
- Ahmed Elgohary, Denis Peskov, and Jordan Boyd-Graber. Can you unpack that? learning to rewrite questions-in-context. In *Empirical Methods in Natural Language Processing, EMNLP '19*, 2019.
- Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, and Dilek Hakkani-Tur. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*, 2019.
- Liana Ermakova, Patrice Bellot, Jaap Kamps, Diana Nurbakova, Irina Ovchinnikova, Eric SanJuan, Elise Mathurin, Sílvia Araújo, Radia Hannachi, Stéphane Huet, et al. Automatic simplification of scientific texts: Simpletext lab at clef-2022. In *European Conference on Information Retrieval, ECIR'22*, pages 364–373, 2022.
- Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*, 2018.
- Leila Feddoul, Sirko Schindler, and Frank Löffler. Automatic facet generation and selection over knowledge graphs. In *International Conference on Semantic Systems*, pages 310–325, 2019.
- Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

- Xiao Fu, Emine Yilmaz, and Aldo Lipani. Evaluating the cranfield paradigm for conversational search systems. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '22, pages 275–280, 2022.
- Ujwal Gadiraju, Ran Yu, Stefan Dietze, and Peter Holtz. Analyzing knowledge gain of users in informational search sessions on the web. In *Proceedings of the 2018 Conference on Human Information Interaction and Retrieval*, CHIIR '18, pages 2–11, 2018.
- Michael Gamon, Tae Yano, Xinying Song, Johnson Apacible, and Patrick Pantel. Identifying salient entities in web pages. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, CIKM '13, page 2375–2380, 2013.
- Chang Gao and Wai Lam. Search clarification selection via query-intent-clarification graph attention. In *European Conference on Information Retrieval*, ECIR '22, pages 230–243, 2022.
- Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*, 2022.
- Carlos Gemmell, Sophie Fischer, Iain Mackie, Paul Owoicho, Federico Rossetto, and Jeff Dalton. Grillbot: A flexible conversational agent for solving complex real-world tasks. *1st Proceedings of the Alexa Prize Taskbot*, 2022.
- Emma J Gerritse, Faegheh Hasibi, and Arjen P de Vries. Bias in conversational search: The double-edged sword of the personalized knowledge graph. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, ICTIR '20, pages 133–136, 2020.
- Yasin Ghafourian. Relevance models based on the knowledge gap. In *European Conference on Information Retrieval*, ECIR '22, pages 488–495, 2022.
- Gustavo Gonçalves, Joao Magalhaes, and Jamie Callan. Conversational search with random walks over entity graphs. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '23, pages 77–85, 2023.
- Michael D Gordon. Evaluating the effectiveness of information retrieval systems using simulated queries. *Journal of the American Society for Information Science*, 41(5):313–323, 1990.

- Jose Griffiths et al. *The computer simulation of information retrieval systems*. PhD thesis, University of London, 1978.
- Izzeddin Gür, Dilek Hakkani-Tür, Gokhan Tür, and Pararth Shah. User modeling for task oriented dialogues. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 900–906, 2018.
- Xianpei Han, Le Sun, and Jun Zhao. Collective entity linking in web text: a graph-based method. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, SIGIR '11*, pages 765–774, 2011.
- Donna Harman. *Information retrieval evaluation*. Morgan & Claypool Publishers, 2011.
- Helia Hashemi, Hamed Zamani, and W Bruce Croft. Guided transformer: Leveraging multiple external sources for representation learning in conversational search. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'20*, pages 1131–1140, 2020.
- Helia Hashemi, Hamed Zamani, and W Bruce Croft. Learning multiple intent representations for search queries. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management, CIKM'21*, pages 669–679, 2021.
- Ahmed Hassan, Rosie Jones, and Kristina Lisa Klinkner. Beyond dcg: user behavior as a predictor of a successful search. In *Proceedings of the third ACM international conference on Web search and data mining, WSDM '10*, pages 221–230, 2010.
- Ben He and Iadh Ounis. Query performance prediction. *Inf. Syst.*, 31(7):585–594, 2006.
- Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021.
- Marti Hearst. *Search user interfaces*. Cambridge University Press, 2009.
- Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration, 2020.
- Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020. URL <https://doi.org/10.5281/zenodo.1212303>.
- Eric Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 159–166, 1999.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.
- HuggingFace. Huggingchat, 2023.
- Bernard J Jansen, Danielle L Booth, and Amanda Spink. Patterns of query reformulation during web searching. *Journal of the American Society for Information Science and Technology*, 60(7):1358–1371, 2009.
- Kalervo Järvelin, Susan L Price, Lois ML Delcambre, and Marianne Lykke Nielsen. Discounted cumulated gain based evaluation of multiple-query ir sessions. In *Advances in Information Retrieval: 30th European Conference on IR Research, ECIR 2008, Glasgow, UK, March 30-April 3, 2008. Proceedings 30*, pages 4–15, 2008.
- Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In *57th Annual Meeting of the Association for Computational Linguistics, ACL '19*, 2019.
- Gaya K Jayasinghe, William Webber, Mark Sanderson, Lasitha S Dharmasena, and J Shane Culpepper. Statistical comparisons of non-deterministic ir systems using two dimensional variance. *Information Processing and Management*, 51(5):677–694, 2015.

- Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12):1–38, 2023.
- Di Jiang, Kenneth Wai-Ting Leung, Lingxiao Yang, and Wilfred Ng. Query suggestion with diversification and personalization. *Knowl.-Based Syst.*, 89:553–568, 2015.
- Hideaki Joko, Faegheh Hasibi, Krisztian Balog, and Arjen P de Vries. Conversational entity linking: Problem definition and datasets. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR’21, pages 2390–2397, 2021.
- Simon Keizer, Milica Gasic, Filip Jurcicek, François Mairesse, Blaise Thomson, Kai Yu, and Steve Young. Parameter estimation for agenda-based user simulation. In *Proceedings of the SIGDIAL 2010 Conference*, SIGDIAL’10, pages 116–123, 2010.
- Diane Kelly and Jaime Teevan. Implicit feedback for inferring user preference: a bibliography. In *Acm Sigir Forum*, volume 37, pages 18–28, 2003.
- Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*, 2019.
- Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. Toward voice query clarification. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1257–1260, 2018.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Weize Kong and James Allan. Extracting query facets from search results. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 93–102, 2013.
- Antonios Minas Krasakis, Mohammad Aliannejadi, Nikos Voskarides, and Evangelos Kanoulas. Analysing the effect of clarifying questions on document ranking in conversational search. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, ICTIR’20, pages 129–132, 2020.

- Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. Neural text summarization: A critical evaluation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, EMNLP'19, pages 540–551, 2019.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.
- Weronika Lajewska and Krisztian Balog. From baseline to top performer: A reproducibility study of approaches at the trec 2021 conversational assistance track. In *European Conference on Information Retrieval, ECIR'23*, pages 177–191, 2023.
- Weronika Lajewska, Nolwenn Bernard, Ivica Kostic, Ivan Sekulic, and Krisztian Balog. The university of stavanger (iai) at the trec 2022 conversational assistance track. 2022.
- Mounia Lalmas, Heather O'Brien, and Elad Yom-Tov. Measuring user engagement. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 6(4): 1–132, 2014.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite BERT for self-supervised learning of language representations. In *proceedings of ICLR*, 2020.
- Alon Lavie and Abhaya Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation, StatMT '07*, page 228–231, 2007.
- Gondy Leroy, James E Endicott, David Kauchak, Obay Mouradi, Melissa Just, et al. User evaluation of the effects of a text simplification algorithm using term familiarity on perception, understanding, learning, and information retention. *Journal of medical Internet research*, 15(7):e2569, 2013.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

- Margaret Li, Jason Weston, and Stephen Roller. Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons. *arXiv preprint arXiv:1909.03087*, 2019.
- Xiujun Li, Zachary C Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. A user simulator for task-completion dialogues. *arXiv preprint arXiv:1612.05688*, 2016.
- Zekun Li, Wenhua Chen, Shiyang Li, Hong Wang, Jing Qian, and Xifeng Yan. Controllable Dialogue Simulation with In-context Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4330–4347, 2022.
- Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.
- Hsien-Chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geishauser, Michael Heck, Shutong Feng, and Milica Gasic. Domain-independent User Simulation with Transformers for Task-oriented Dialogue Systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGdial 2021*, pages 445–456, 2021a.
- Hsien-Chin Lin, Christian Geishauser, Shutong Feng, Nurul Lubis, Carel van Niekerk, Michael Heck, and Milica Gasic. GenTUS: Simulating User Behaviour and Language in Task-oriented Dialogues with Generative Transformers. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL 2022*, SIGDIAL'22, pages 270–282, 2022.
- Hsien-Chin Lin, Shutong Feng, Christian Geishauser, Nurul Lubis, Carel van Niekerk, Michael Heck, Benjamin Ruppik, Renato Vukovic, and Milica Gasić. Emous: Simulating user emotions in task-oriented dialogues. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'23*, pages 2526–2531, 2023.
- Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. Conversational question reformulation via sequence-to-sequence architectures and pretrained language models. *arXiv preprint arXiv:2004.01909*, 2020.
- Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. Multi-stage conversational passage retrieval: An approach to fusing term importance estimation and neural query rewriting. *ACM Transactions on Information Systems (TOIS)*, 39(4):1–29, 2021b.

- Aldo Lipani, Ben Carterette, and Emine Yilmaz. From a user model for query sessions to session rank biased precision (srbp). In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR '19*, pages 109–116, 2019.
- Aldo Lipani, Ben Carterette, and Emine Yilmaz. How am i doing?: Evaluating conversational search systems offline. *ACM Transactions on Information Systems (TOIS)*, 39(4):1–22, 2021.
- Oleg Litvinov, Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Analyzing coherency in facet-based clarification prompt generation for search. *arXiv preprint arXiv:2401.04524*, 2024.
- Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP'16*, pages 2122–2132, 2016.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.
- Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345, 2021.
- Iain Mackie, Ivan Sekulic, Jeffrey Dalton, and Fabio Crestani. Exploring complex topics: Document and entity graph construction for ranking. In *Unpublished work.*, 2021.
- Iain Mackie, Paul Owoicho, Carlos Gemmell, Sophie Fischer, Sean MacAvaney, and Jeffrey Dalton. Codec: Complex document and entity collection. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '22*, pages 3067–3077, 2022.
- François Mairesse, Milica Gasic, Filip Jurcicek, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1552–1561, 2010.

- Bodhisattwa Prasad Majumder, Sudha Rao, Michel Galley, and Julian McAuley. Ask what's missing and what's useful: Improving clarification question generation using global knowledge. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL '21*, pages 4300–4312, 2021.
- Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006.
- Yosi Mass, Doron Cohen, Asaf Yehudai, and David Konopnicki. Conversational search with mixed-initiative - asking good clarification questions backed-up by passage retrieval. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pages 65–71, 2022.
- Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. <http://mallet.cs.umass.edu>, 2002.
- Philip M McCarthy. *An assessment of the range and usefulness of lexical diversity measures and the potential of the measure of textual, lexical diversity (MTLD)*. PhD thesis, The University of Memphis, 2005.
- Rishabh Mehrotra, Ahmed Hassan Awadallah, and Emine Yilmaz. Learnir: WSDM 2018 workshop on learning from user interactions. In *Proceedings of the eleventh ACM international Conference on Web Search and Data Mining, WSDM'18*, pages 797–798, 2018.
- Ida Mele, Cristina Ioana Muntean, Franco Maria Nardini, Raffaele Perego, Nicola Tonello, and Ophir Frieder. Topic propagation in conversational search. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20*, pages 2057–2060, 2020.
- Chuan Meng, Mohammad Aliannejadi, and Maarten de Rijke. System initiative prediction for multi-turn conversational information seeking. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM'23*, pages 1807–1817, 2023.
- Selina Meyer, David Elswiler, Bernd Ludwig, Marcos Fernandez-Pichel, and David E. Losada. Do we still need human assessors? prompt-based gpt-3 user simulation in conversational ai. In *Proceedings of the 4th Conference on Conversational User Interfaces, CUI '22*, 2022.

- Bhaskar Mitra, Fernando Diaz, and Nick Craswell. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web, TheWebConf '17*, pages 1291–1299, 2017.
- Alistair Moffat and Justin Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems (TOIS)*, 27(1): 1–27, 2008.
- Javed Mostafa, Snehasis Mukhopadhyay, and Mathew Palakal. Simulation studies of different dimensions of users' interests and their impact on user modeling and information filtering. *Information Retrieval*, 6(2):199–223, 2003.
- Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *NeurIPS'16*, 2016.
- Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with BERT. *arXiv preprint arXiv:1901.04085*, 2019.
- Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, 2020.
- Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. Why we need new evaluation metrics for nlg. In *EMNLP*, pages 2241–2252, 2017.
- Heather L O'Brien. Antecedents and learning outcomes of online news engagement. *Journal of the Association for Information Science and Technology*, 68(12):2809–2820, 2017.
- Heather L O'Brien and Elaine G Toms. What is user engagement? A conceptual framework for defining user engagement with technology. *Journal of the American society for Information Science and Technology*, 59(6):938–955, 2008.
- Heather L O'Brien, Jaime Arguello, and Rob Capra. An empirical study of interest, task complexity, and search behaviour on user engagement. *Information Processing and Management*, 57(3):102226, 2020.

- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *arXiv*, cs.CL/2203.02155, 2022.
- Paul Owoicho, Jeffery Dalton, Mohammad Aliannejadi, Leif Azzopardi, Johanne R. Trippas, and Svitlana Vakulenko. TREC CAsT 2022: Going beyond user ask and system retrieve with initiative and response generation, 2022.
- Paul Owoicho, Ivan Sekulić, Mohammad Aliannejadi, Jeffrey Dalton, and Fabio Crestani. Exploiting simulated user feedback for conversational search: Ranking, rewriting, and beyond. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'23, pages 632–642, 2023.
- Teemu Pääkkönen, Jaana Kekäläinen, Heikki Keskustalo, Leif Azzopardi, David Maxwell, and Kalervo Järvelin. Validating simulated interaction for retrieval evaluation. *Information Retrieval Journal*, 20:338–362, 2017.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.
- Deepa Paranjpe. Learning document aboutness from implicit user feedback and document structure. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 365–374, 2009.
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8026–8037, 2019.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

- Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. Few-shot natural language generation for task-oriented dialog. *arXiv preprint arXiv:2002.12328*, 2020.
- Gustavo Penha and Claudia Hauff. Challenges in the evaluation of conversational search systems. *KDD Workshop on Conversational Systems Towards Mainstream Adoption*, 2020.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
- Jayr Pereira, Robson Fidalgo, Roberto Lotufo, and Rodrigo Nogueira. Visconde: Multi-document qa with gpt-3 and neural reranking. In *Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, 2023, Proceedings, Part II, ECIR'23*, pages 534–543, 2023.
- Francesco Piccinno and Paolo Ferragina. From tagme to wat: a new entity annotator. In *Proceedings of the first international workshop on Entity recognition and disambiguation, ERD '14*, pages 55–62, 2014.
- Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *SIGIR*, pages 275–281, 1998.
- Marco Ponza, Paolo Ferragina, and Francesco Piccinno. Swat: A system for detecting salient wikipedia entities in texts. *Computational Intelligence*, 35(4): 858–890, 2019.
- Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv*, cs.IR/2101.05667, 2021.
- Chen Qu, Liu Yang, W. Bruce Croft, Johanne R. Trippas, Yongfeng Zhang, and Minghui Qiu. Analyzing and characterizing user intent in information-seeking conversations. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '18*, pages 989–992, 2018.
- Chen Qu, Liu Yang, W Bruce Croft, Yongfeng Zhang, Johanne R Trippas, and Minghui Qiu. User intent prediction in information-seeking conversations. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, CHIIR '19*, pages 25–33, 2019.

- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- Filip Radlinski and Nick Craswell. A theoretical framework for conversational search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, CHIIR '17, page 117–126, 2017.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.
- Hossein A Rahmani, Xi Wang, Yue Feng, Qiang Zhang, Emine Yilmaz, and Aldo Lipani. A survey on asking clarification questions datasets in conversational systems. *arXiv preprint arXiv:2305.15933*, 2023.
- Sudha Rao and Hal Daumé III. Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL'18, pages 2737–2746, 2018.
- Sudha Rao and Hal Daumé III. Answer-based adversarial training for generating clarification questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, NAACL'19, pages 143–155, 2019.
- Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, EMNLP-IJCNLP '19, pages 3982–3992, 2019.
- Alan Ritter, Colin Cherry, and William B Dolan. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL'10, pages 172–180, 2010.
- Stephen E Robertson and K Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3):129–146, 1976.

- Corbin Rosset, Chenyan Xiong, Xia Song, Daniel Campos, Nick Craswell, Saurabh Tiwary, and Paul Bennett. Leading conversational search by suggesting useful questions. In *Proceedings of The Web Conference 2020*, TheWebConf'20, pages 1160–1170, 2020.
- Nafise Sadat Moosavi, Iryna Gurevych, Yufang Hou, Gyuwan Kim, Young Jin Kim, Tal Schuster, and Ameeta Agrawal, editors. *Proceedings of The Fourth Workshop on Simple and Efficient Natural Language Processing (SustaiNLP)*, 2023. Association for Computational Linguistics.
- Ananya B Sai, Akash Kumar Mohankumar, and Mitesh M Khapra. A survey of evaluation metrics used for nlg systems. *ACM Computing Surveys (CSUR)*, 55(2):1–39, 2022.
- Tetsuya Sakai. Graded relevance. In *Evaluating Information Retrieval and Access Tasks*, pages 1–20. 2021.
- Alexandre Salle, Shervin Malmasi, Eugene Agichtein, and Oleg Rokhlenko. Studying the effectiveness of conversational search refinement through user simulation. In *Proceedings of the European Conference on Information Retrieval (ECIR)*, ECIR'21, 2021.
- Shumpei Sano, Nobuhiro Kaji, and Manabu Sassano. Prediction of prospective user engagement with intelligent assistants. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1203–1212, 2016.
- Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. Search result diversification. *Found. Trends Inf. Retr.*, 9(1):1–90, 2015.
- Jost Schatzmann and Steve J. Young. The hidden agenda user simulation model. *IEEE Transactions on Audio, Speech, and Language Processing*, 17:733–747, 2009.
- Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve Young. Agenda-based user simulation for bootstrapping a pomdp dialogue system. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, NAACL'07, pages 149–152, 2007.
- Emanuel A Schegloff and Harvey Sacks. Opening up closings. *Semiotica*, 8(4), 1973.

- Phillip Schneider, Nils Rehtanz, Kristiina Jokinen, and Florian Matthes. From data to dialogue: Leveraging the structure of knowledge graphs for conversational exploratory search. *arXiv preprint arXiv:2310.05150*, 2023.
- Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. What makes a good conversation? how controllable attributes affect human judgments. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, NAACL'19, pages 1702–1723, 2019.
- Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Extending the use of previous relevant utterances for response ranking in conversational search. In *Proceedings of the Twenty-Ninth Text REtrieval Conference, TREC*, 2020.
- Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Towards facet-driven generation of clarifying questions for conversational search. In *Proceedings of the 2021 ACM SIGIR on International Conference on Theory of Information Retrieval, ICTIR'21*, 2021.
- Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. User engagement prediction for clarification in search. In *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR'21*, pages 619–633, 2021.
- Ivan Sekulić, Vincent Herrmann, Daniele Grattarola, and Fabio Crestani. Towards multi-document information retrieval with graph neural networks. In *Unpublished work.*, 2021.
- Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Evaluating mixed-initiative conversational search systems via user simulation. In *International Conference on Web Search and Data Mining, WSDM'22*, 2022.
- Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Exploiting document-based features for clarification in conversational search. In *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR'22*, pages 413–427, 2022.
- Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Analysing utterances in llm-based user simulation for conversational search. *ACM Trans. Intell. Syst. Technol.*, 2024.
- Ivan Sekulić, Krisztian Balog, and Fabio Crestani. Towards self-contained answers: Entity-based answer rewriting in conversational search. In *Proceedings*

- of the 2024 Conference on Conference Human Information Interaction and Retrieval, CHIIR'24, pages 209—218, 2024a.
- Ivan Sekulić, Weronika Lajewska, Krisztian Balog, and Fabio Crestani. Estimating the usefulness of clarifying questions and answers for conversational search. In *Advances in Information Retrieval - 46th European Conference on IR Research, ECIR'24*, pages 384–392, 2024b.
- Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- Yuanlong Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil. Generating high-quality and informative conversation responses with sequence-to-sequence models. In *EMNLP*, pages 2210–2219, 2017.
- Kim Cheng Sheang and Horacio Saggion. Controllable sentence simplification with a unified text-to-text transfer transformer. In *Proceedings of the 14th International Conference on Natural Language Generation, INLG '21*, pages 341–352, 2021.
- Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33:16857–16867, 2020.
- Alessandro Sordani, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM'15*, pages 553–562, 2015.
- Amanda Stent, Matthew Marge, and Mohit Singhai. Evaluating evaluation methods for generation in the presence of variation. In *international conference on intelligent text processing and computational linguistics*, pages 341–351, 2005.
- Weiwei Sun, Shuo Zhang, Krisztian Balog, Zhaochun Ren, Pengjie Ren, Zhumin Chen, and Maarten de Rijke. Simulating user satisfaction for the evaluation of task-oriented dialogue systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'21*, pages 2499–2506, 2021.
- Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NeurIPS'14*, page 3104–3112, 2014.

- Jean Tague, Michael Nelson, and Harry Wu. Problems in the simulation of bibliographic retrieval systems. In *Proceedings of the 3rd Annual ACM Conference on Research and Development in Information Retrieval*, SIGIR '80, page 236–255, 1980.
- Silvia Terragni, Modestas Filipavicius, Nghia Khau, Bruna Guedes, André Manso, and Roland Mathis. In-context learning user simulators for task-oriented dialog systems. *arXiv preprint arXiv:2306.00774*, 2023.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- Johanne R. Trippas, Damiano Spina, Lawrence Cavedon, Hideo Joho, and Mark Sanderson. Informing the design of spoken conversational search: Perspective paper. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, CHIIR'17, pages 32–41, 2018.
- Svitlana Vakulenko, Kate Revoreda, Claudio Di Ciccio, and Maarten de Rijke. QRFA: A data-driven model of information seeking dialogues. In *Advances in Information Retrieval*, pages 541–557, 2019.
- Svitlana Vakulenko, Nikos Voskarides, Zhucheng Tu, and Shayne Longpre. A comparison of question rewriting methods for conversational passage retrieval. In *European Conference on Information Retrieval*, ECIR'21, pages 418–424, 2021.
- Svitlana Vakulenko, Johannes Kiesel, and Maik Fröbe. SCAI-QReCC shared task on conversational question answering. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, LREC '22, pages 4913–4922, 2022.
- Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P. de Vries. Rel: An entity linker standing on the shoulders of giants. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'20, 2020.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NeurIPS'17, page 6000–6010, 2017.

- Anthony J Viera, Joanne M Garrett, et al. Understanding interobserver agreement: the kappa statistic. *Family Medicine*, 37(5):360–363, 2005.
- Marilyn A Walker and Steve Whittaker. Mixed initiative in dialogue: An investigation into discourse segmentation. In *Association for Computational Linguistics, ACL '90*, 1990.
- Dazhen Wan, Zheng Zhang, Qi Zhu, Lizi Liao, and Minlie Huang. A unified dialogue user simulator for few-shot data augmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3788–3799, 2022.
- Zhenduo Wang and Qingyao Ai. Controlling the risk of conversational search via reinforcement learning. In *Proceedings of The Web Conference, TheWebConf'21*, pages 1968–1977, 2021.
- Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP'15*, pages 1711–1721, 2015.
- Ryen W White. *Interactions with search systems*. Cambridge University Press, 2016.
- Jason D Williams, Antoine Raux, and Matthew Henderson. The dialog state tracking challenge series: A review. *Dialogue & Discourse*, 7(3):4–33, 2016.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910, 2019.
- Chuan Wu, Evangelos Kanoulas, and Maarten de Rijke. It all starts with entities: A salient entity topic model. *Natural Language Engineering*, 26(5):531–549, 2020.
- Gui-Rong Xue, Hua-Jun Zeng, Zheng Chen, Yong Yu, Wei-Ying Ma, WenSi Xi, and WeiGuo Fan. Optimizing web search using web click-through data. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 118–126, 2004.

- Shi Yu, Jiahua Liu, Jingqin Yang, Chenyan Xiong, Paul Bennett, Jianfeng Gao, and Zhiyuan Liu. Few-shot generative conversational query rewriting. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'20, page 1933–1936, 2020.
- Shi Yu, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu. Few-shot conversational dense retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'21, pages 829–838, 2021.
- Hamed Zamani, Susan Dumais, Nick Craswell, Paul Bennett, and Gord Lueck. Generating clarifying questions for information retrieval. In *Proceedings of The Web Conference 2020*, TheWebConf'20, pages 418–428, 2020a.
- Hamed Zamani, Gord Lueck, Everest Chen, Rodolfo Quispe, Flint Luu, and Nick Craswell. Mimics: A large-scale data collection for search clarification. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, CIKM'20, pages 3189–3196, 2020b.
- Hamed Zamani, Bhaskar Mitra, Everest Chen, Gord Lueck, Fernando Diaz, Paul N. Bennett, Nick Craswell, and Susan T. Dumais. Analyzing and learning from user interactions for search clarification. page 1181–1190, 2020c.
- Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. Conversational information seeking: Theory and application. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'22, page 3455–3458, 2022.
- Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. Defending against neural fake news. *arXiv preprint arXiv:1905.12616*, 2019.
- Qiusi Zhan, Xiaojie Guo, Heng Ji, and Lingfei Wu. User simulator assisted open-ended conversational recommendation system. In Yun-Nung Chen and Abhinav Rastogi, editors, *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 89–101, 2023.
- Shuo Zhang and Krisztian Balog. Evaluating conversational recommender systems via user simulation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '20, page 1512–1520, 2020.

- Shuo Zhang, Mu-Chun Wang, and Krisztian Balog. Analyzing and simulating user utterance reformulation in conversational recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'22, pages 133–143, 2022.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations*, ICLR'20, 2020a.
- Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W Bruce Croft. Towards conversational search and recommendation: System ask, user respond. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM'18, pages 177–186, 2018.
- Zheng Zhang, Ryuichi Takanobu, Qi Zhu, MinLie Huang, and XiaoYan Zhu. Recent advances and challenges in task-oriented dialog systems. *Science China Technological Sciences*, 63(10):2011–2027, 2020b.
- Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. Dense text retrieval based on pretrained language models: A survey. *arXiv preprint arXiv:2211.14876*, 2022.
- Yun Zhou and W. Bruce Croft. Query performance prediction in web search environments. In *Proceedings of the 30th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pages 543–550, 2007.
- Qi Zhu, Zheng Zhang, Yan Fang, Xiang Li, Ryuichi Takanobu, Jinchao Li, Baolin Peng, Jianfeng Gao, Xiaoyan Zhu, and Minlie Huang. Convlab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.
- Jie Zou, Evangelos Kanoulas, and Yiqun Liu. An empirical study on clarifying question-based systems. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, CIKM'20, pages 2361–2364, 2020.
- Jie Zou, Aixin Sun, Cheng Long, Mohammad Aliannejadi, and Evangelos Kanoulas. Asking clarifying questions: To benefit or to disturb users in web search? *Information Processing & Management*, 60(2):103176, 2023.