Exploiting Simulated User Feedback for Conversational Search

Ranking, Rewriting, and Beyond

Owoicho, P.; Sekulić, I.; Aliannejadi, M.; Dalton, J.; Crestani, F.

DOI
10.1145/3539618.3591683

Publication date
2023

Document Version
Final published version

Published in
SIGIR '23

Exploiting Simulated User Feedback for Conversational Search: Ranking, Rewriting, and Beyond

Paul Owoicho*
University of Glasgow
Glasgow, Scotland, UK
p.owoicho.1@research.gla.ac.uk

Ivan Sekulić*
Università della Svizzera italiana
Lugano, Switzerland
ivan.sekulic@usi.ch

Mohammad Aliannejadi
University of Amsterdam
Amsterdam, The Netherlands
m.aliannejadi@uva.nl

Jeffrey Dalton
University of Glasgow
Glasgow, Scotland, UK
jeff.dalton@glasgow.ac.uk

Fabio Crestani
Università della Svizzera italiana
Lugano, Switzerland
fabio.crestani@usi.ch

ABSTRACT

This research aims to explore various methods for assessing user feedback in mixed-initiative conversational search (CS) systems. While CS systems enjoy profuse advancements across multiple aspects, recent research fails to successfully incorporate feedback from the users. One of the main reasons for that is the lack of system–user conversational interaction data. To this end, we propose a user simulator-based framework for multi-turn interactions with a variety of mixed-initiative CS systems. Specifically, we develop a user simulator, dubbed *ConvSim*, that, once initialized with an information need description, is capable of providing feedback to system's responses, as well as answering potential clarifying questions. Our experiments on a wide variety of state-of-the-art passage retrieval and neural re-ranking models show that effective utilization of user feedback can lead to 16% retrieval performance increase in terms of nDCG@3. Moreover, we observe consistent improvements as the number of feedback rounds increases (35% relative improvement in terms of nDCG@3 after three rounds). This points to a research gap in the development of specific feedback processing modules and opens a potential for significant advancements in CS. To support further research in the topic, we release over 30 000 transcripts of system-simulator interactions based on well-established CS datasets.

CCS CONCEPTS

• **Information systems** → **Users and interactive retrieval**.

KEYWORDS

user simulation, conversational information seeking, mixed-initiative

ACM Reference Format:
Paul Owoicho*, Ivan Sekulić*, Mohammad Aliannejadi, Jeffrey Dalton, and Fabio Crestani. 2023. 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), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3539618.3591683

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07.
https://doi.org/10.1145/3539618.3591683

1 INTRODUCTION

The primary goal of a conversational search (CS) system is to satisfy the user's information need. However, there are several challenges that arise when it comes to CS, as opposed to traditional *ad-hoc* search. An important tool for addressing these challenges is the use of mixed-initiative techniques. Under the mixed-initiative paradigm, the conversational search system can proactively initiate prompts, such as suggestions, warnings, or questions, at any point in the conversation. In recent years, mixed-initiative conversational search has received significant attention from the information retrieval (IR) research community, leading to advancements in various aspects of this field, including conversational passage retrieval [18, 58], query rewriting in context [54], intent prediction in conversations [39], and asking clarifying questions [5].

Despite the abundance of research on various components of mixed-initiative search systems, little has been done to study the impact of user feedback. 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 CS 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 [65]. Despite its significance, lack of research in this area can be attributed to the difficulty of collecting appropriate data containing user feedback.

Furthermore, evaluation of CS systems is arduous [29, 36]. Typically, it requires the actual users to interact with the system, presenting their information needs, answering potential questions, and providing feedback. Such studies are expensive and time consuming, often requiring a large number of experiments to properly evaluate specific approaches. That is even more the case with mixed initiatives, as the number of possible conversations is essentially limitless [10]. An attempt to address this issue is to compile offline collections aimed at specific challenges in conversational search [4, 18, 38]. Existing data collections are mainly built based on online human–human conversations [38], synthetic human–computer interactions [18], and multiple rounds of crowdsourcing [4]. No existing

*These authors contributed equally to this work

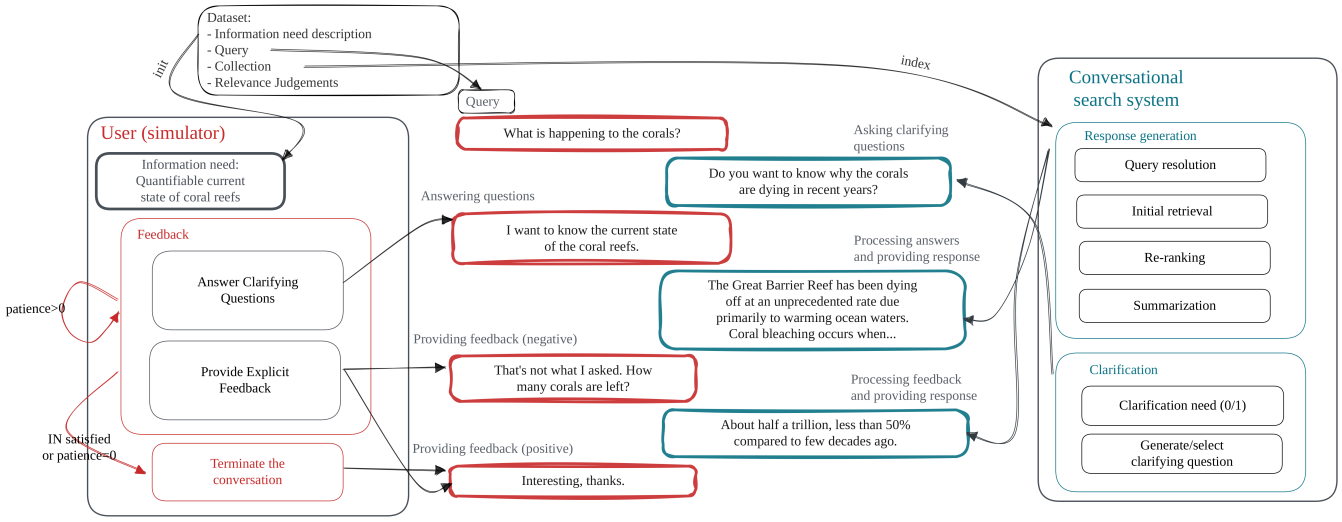

Figure 1: Experimental framework with an example interaction between a user simulator (left) and a mixed-initiative conversational search system (right). Functionalities and modules of both are highlighted.

data collections, however, feature explicit user feedback extensively in a conversation, thus limiting research in this area. Moreover, such corpus-based evaluation paradigms usually remain limited to single-turn interactions and do not take into account the interactive nature of CS, not to mention being limited to non-generative models.

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 1. The framework allows us to evaluate multiple state-of-the-art mixed-initiative CS systems, addressing several challenges, including contextual query resolution, asking clarifying questions, and incorporating user feedback.

Existing work [2] 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 [46, 63] or focus only on one aspect of the search system, e.g., answering clarifying questions [48]. To address these limitations, we propose a user simulator called *ConvSim*, capable of multi-turn interactions with mixed-initiative CS 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 large language models (LLMs), e.g., GPT-3 [13], PALM [15], 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 [48], coherent across multiple conversational turns. The proposed simulator interacts with CS 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.

We stress the fact that research questions around feedback utilization in CS 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.

We find significant improvements in retrieval performance of methods utilizing feedback compared to non-feedback methods, even with only a single turn of feedback. Well-established methods, such as RM3, adapted to handle explicit feedback, demonstrate relative improvement of 11% and 9% in terms of recall and nDCG@3. Further, we identify a shortcoming of standard T5 query rewriter [28] in the task of processing feedback. To address this, we propose a novel adaptation of the T5 method and achieve 10% and 16% improvements in terms of recall and nDCG@3, respectively. Similarly, incorporating answers to clarifying questions yields improvements both in recall (18%) and nDCG@3 (12%). We also find that multiple rounds of simulator–system interactions result in further improvements in retrieval effectiveness (35% relative improvement in terms of nDCG@3 after three rounds). Moreover, we observe that existing methods react poorly to certain types of feedback (e.g., positive feedback "Thanks"), leading to a decrease in performance. This points to a research gap in development of specific feedback processing modules and opens a potential for significant advancements in CS.

Our main contributions are:

- New insights into mixed-initiative CS system design, with a focus on processing users' feedback, including explicit feedback and their answers to clarifying questions.

- A user simulator, capable of multi-turn interactions with mixed-initiative search systems. We release transcripts, code and guidelines[1] to foster further research.

2 RELATED WORK

2.1 Mixed-initiative conversational search

In recent years, conversational search has attracted significant attention both from the IR and natural language processing (NLP) communities [6]. To this end, Radlinski and Craswell [40] propose a theoretical framework of conversational search, identifying key properties of such systems and focusing on natural and efficient information access through conversations. While some of the challenges remain similar to traditional *ad hoc* search, significant new ones arise in the conversational paradigm. These are surveyed in the recent manuscript of Zamani et al. [62]. They include conversational query rewriting [54, 57], conversational retrieval [18, 58] and user intent prediction [39].

One key element of conversational search is mixed-initiative, which is the interaction pattern where both the system 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 [56] identifying it as an integral part of conversations and Horvitz [22] identifying key principles of mixed-initiative interactions. One of the most prominent uses of mixed-initiative is asking clarifying questions with a goal of elucidating the underlying user's information need [5, 12, 51, 60]. The benefits of prompting the user with clarifying questions is found by multiple studies, including improving retrieval performance in conversational search [3, 24, 44, 61, 64]. Clarifying questions are generally either selected from a pre-defined pool of questions [3, 5, 41] or generated [42, 47, 59]. While decent success has been demonstrated by various question selection methods [3], such approaches remain limited to pre-defined conversational trajectories and are not fit for a realistic search scenario. Therefore, generating a clarifying question poses itself as a natural improvement over the selection task, mitigating the need to collect all of the potential questions beforehand. Various question generation methods exist, centered around either template-based questions or LLM-based generation. In this work, we also study clarifying questions and use simulation methods to answer them. While there are benefits to clarifying questions, there is also cost to the user for these interactions [7, 8]. In this work we focus on their effectiveness in a simulation environment and don't study user costs directly.

2.2 Evaluation and user simulation

Deriu et al. [19] 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 [19] there are key trade-offs. Liu et al. [30] 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. Thus, [19] identify user studies as a more reliable method for evaluating conversational systems, stressing the fact that such evaluation is both cost- and time-intensive.

Conversational search has similar evaluation challenges, further complicated by the retrieval of relevant documents from a large collection [36]. While traditional Cranfield paradigm fits well for evaluation of *ad hoc* search systems, it is not easily transferable to conversational search [20, 29]. 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 [21, 23].

Balog [10] makes the case that simulation is an important emerging research frontier for conversational search evaluation. Pääkkönen et al. [34] assess the validity of the use of simulated users in interactive IR and find it justified under a common interaction model. While user simulators are a well-established idea in IR [14, 31], including applications such as simulating user satisfaction for the evaluation of task-oriented dialogue systems [52] and recommender systems [1, 63], their utilization in mixed-initiative conversational search is limited.

To address this, Salle et al. [46] design 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, making its usability restricted to a closed collection of such questions and answers. Sekulić et al. [48] address that issue and design *USi*, a simulator capable of generating answers to clarifying questions posed by the system. Nonetheless, their approach is 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 this work 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.

3 BACKGROUND AND PROBLEM DEFINITION

In this section, we formally define the main task definition of mixed-initiative conversational search systems. We then link these to the requirements of user simulation.

Formally, a search session consists of multiple turns of the user's utterances u and the system's utterances s, forming conversational history $H = [u^1, s^1, \ldots, u^{t-1}, s^{t-1}]$, 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 $R = [r_1, r_2, \ldots, r_N]$,

[1] https://github.com/grill-lab/ConvSim

retrieved in response to u^t, where N is the maximum number of results considered.

3.1 Mixed initiatives

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 CS system can at any point take initiative and prompt the user with various suggestions or clarifying questions [40].

3.1.1 Clarification. 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 [4]. 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 $clarification\_need = f(u^t|H, D)$, where $clarification\_need \in \{0, 1\}$, indicating whether not to ask or to ask a clarifying question.

As mentioned, asking clarifying questions methods can be broadly categorized into *question selection* and *question generation* [3, 5] 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 $CQ = \{cq_1, cq_2, \ldots, cq_n\}$. Formally, we model $s_{cq} = \phi(u_t|H, R, CQ)$ where ϕ is our question selection model. As discussed in Section 2, 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 $s_{cq} = \psi(u_t|H, R)$. In this work, we implement several state-of-the-art question selection and generation models and evaluate their performance. Moreover, we test the robustness of feedback processing modules depending on the type of clarifying question.

3.1.2 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., $u^{t'} = \gamma_1(u^t_a|H)$ and $u^{t'} = \gamma_2(u^t_f|H)$. 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: $u^{t'}_q = \gamma_3(u^t_q|H)$.

3.2 User simulation

A user simulator aims to mimic key user's roles in MI interactions. Although Balog [10] 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 [63], we base our user interaction model on the general QRFA model for the conversational information-seeking process [53].

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. Figure 1 shows a components of the simulator, where θ_1 and θ_2 are utilized at appropriate stages.

Asking too many clarifying questions or providing unsatisfactory responses might impair user's satisfaction with the search system [65]. Thus, a simulator should encapsulate similar behaviors. Following Salle et al. [46], 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.

3.2.1 Naturalness and usefulness of generated answers. In order for simulator's behavior to be similar to real users [10], both answers s_a and feedback s_f need to be relevant, in coherent natural language, and consistent with information need IN. Following Sekulić et al. [48], we assess *naturalness* and *usefulness* of the generated answers to clarifying questions. *Naturalness* refers to the utterance being in fluent natural language and likely generated by humans [35, 45]. We ground our definition of *usefulness* in previous work assessing clarifying questions [44] and their answers [48]. Specifically, it captures whether answers and feedback generated by the simulator are consistent with the provided information need, and can be related to adequacy [50] and informativeness [16]. Moreover, by extending the evaluation to the multi-turn setting, we are also evaluating simulator's context awareness.

3.2.2 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^t_q, the system returns response s^{t+1}_r and the simulator generates feedback u^{t+1}_f. Moreover, the utterance u^{t+1}_f is externally annotated as positive or negative feedback. Our aim is to measure correlation of retrieval performance at turn u^t_q and type of feedback u^{t+1}_f (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.

4 METHODOLOGY

4.1 Proposed simulator framework

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 Section 3.2. 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 reranking, clarifying question selection and generation, and response generation. The framework is depicted in Figure 1. It enables seamless multi-turn exchange of user simulator utterances u and system's utterances s, detailed in Section 3. 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. 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.

We model feedback generation functions θ_1 and θ_2 detailed in Section 3.2 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.

4.1.1 Implementation details. We build *ConvSim* on top of OpenAI's Text-Davinci-003 [13] model using few-shot prompting. We use OpenAI's completions API endpoint with the following parameter settings based on the author's guidelines [13] and initial empirical exploration:

- ***max_tokens*:** 50. 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. This is a halfway point between a very conservative and risky model. While we want creative outputs, we also want the responses to be on topic.
- ***frequency_penalty*:** 0.2. This discourages the model from generating previously generated tokens (i.e., repeating itself).
- ***presence_penalty*:** 0.5. This encourages the model to introduce new topics. 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.

4.2 Evaluation Data

We primarily use the **TREC CAsT** [33] 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. We focus on year 4 because it is the only dataset that includes mixed-initiative interactions.

Because 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 [17]. 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. 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.

4.3 Mixed-initiative systems

4.3.1 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 dataset [2]. 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 Pyeserini's [3] 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) [32] trained

[2]https://huggingface.co/castorini/t5-base-canard
[3]https://github.com/castorini/pyserini

on MSMARCO [9]. Finally, a BART model [4] 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 Section 3.1.2, aimed at updating the system's understanding of the user's information need:

organizer-auto+RM3 uses the user feedback u_f after the BART response generation step. Using the RM3 algorithm [26], we expand the reformulated query u^t with up to 10 terms from the feedback utterance u_f: $u_q^{t'} = u_q^t + RM3(u_f)$. This expanded query is fed through the BM25 and MonoT5 steps, followed by BART response generation. For our experiments, we interpret the number of feedback rounds as a proxy for user patience, detailed in Section 3.2, i.e., the more rounds of feedback a user is willing to give, the more patient they are.

organizer-auto+Rocchio follows the same setup as **organizer-auto+RM3** but uses the Rocchio algorithm [43] for processing explicit feedback: $u_q^{t'} = u_q^t + Rocchio(u_f)$.

organizer-auto+QuReTeC expands the user's query with the QuReTeC model [55] using terms from the conversation history. In our experiments, we adapt QuReTeC to additionally take terms from the explicit simulator feedback into account: $u_q^{t'} = u_q^t + QuReTeC(u_f, H)$.

To assess if feedback utilization works on other systems, we also evaluate three of the strongest automatic submissions to CAsT'22, including *splade_t5mm_ens*, *uis_sparseboat*, and *UWCcano22*. We obtain the run files of these systems from the CAsT'22 organizers.

4.3.2 Utilizing feedback. 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 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. [5]. 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.

4.3.3 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 Section 3.1.1, 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 Section 3.1.1, 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 [3, 5]. Second, a semantic matching-based method, termed **SelectCQ-MPNet**, utilizing MPNet [49] 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 [25]. 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 [47, 59, 63]. With entities being central to the topic of a document, we opt to utilize SWAT [37] 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 $E = [e_1, e_2, \ldots, e_M]$. 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.

[4] https://huggingface.co/facebook/bart-large-cnn

4.4 Evaluation

4.4.1 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 CS system components on retrieval performance. Specifically, we evaluate the base pipeline, described in Section 4.1 for passage retrieval with and without CS system components.

4.4.2 Naturalness and usefulness of generated answers. We evaluate *ConvSim* in terms of naturalness and usefulness, as described in Section 3.2.1. To this end, we compare our method to the current state-of-the-art simulator for answering clarifying questions, *USi* [48], as well as human-generated responses. Following [48], we conduct a crowdsourcing-based evaluation on the ClariQ dataset [3]. 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 [48] with human-generated multi-turn conversations. We follow Li et al. [27] and present full conversations for comparisons. We report statistical significance under the trinomial test [11], 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.

We use the Amazon Mechanical Turk[5] platform for our crowdsourcing-based experiments. 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 $0.25 per HIT, which with an average completion time of about 30 seconds, more than 300% of the minimum wage in the U.S.

4.4.3 Feedback. We evaluate the feedback generation capabilities of *ConvSim* as described in Section 3.2.2. To this end, we generate responses for each turn in the CAsT'22 dataset with the Organizer-auto method, described in Section 4.3.1. 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.

[5] mturk.com

5 RESULTS

In this section we present the empirical evaluation with three core research questions:

RQ1 How can we leverage user feedback and what is its effect on core components of a conversational search pipeline including: explicit relevance feedback processing, ranking and generating clarifying questions, and in core ranking?
RQ2 How does the *ConvSim* model compare with existing approaches for multi-turn simulation in terms of naturalness and usefulness?
RQ3 What is the effect of multiple rounds of simulated feedback when used in ranking?

5.1 Mixed-initiative systems

Tables 1 and 2 list the retrieval results for query reformulation and passage ranking, respectively. Generally, the results demonstrate improvements of feedback-aware methods over the baselines. Below, we discuss the findings in detail.

5.1.1 Query rewriting with feedback. Compared to the baseline system, the addition of the *QuReTeC* results in a 39% decrease in nDCG@3. This is surprising, considering *QuReTeC*'s strong performance on previous editions of the CAsT benchmark. Likewise, *Rocchio* also leads to a decrease in performance, with nDCG@3 going down by 0.151 points (41%). In contrast, the addition of *RM3* improves performance compared to the baseline, significantly outperforming it in terms of Recall, MAP, nDCG, and nDCG@3. Moreover, the results show the *Discourse-CQR* method to outperform the baseline across all metrics, demonstrating the strongest performance among the implemented methods.

Expectedly, high-quality query rewriting/reformulation with feedback enables systems to retrieve more relevant passages in the initial retrieval stage at each turn, as evidenced by the increase in recall for the *RM3* and *Discourse-CQR* methods over the baseline. Not all reformulation methods are effective in all cases, however. Consider a turn where a user provides the following negative feedback without clarification: *"That's not what I asked for. Can you please answer my question?"* Term expansion methods based on explicit feedback alone, such as *RM3* and *Rocchio*, completely fail, given the lack of relevant terms in the feedback utterance. On the other hand, methods that rely on explicit feedback and conversational history stand a better chance, as they have access to more relevant context to arrive at a better expression of the under-specified query.

We note that, without fine-tuning, *T5-CQR* performs competitively as a feedback rewriter, but still underperforms RM3 due to the 'over-rewriting' issues discussed in Section 4.3.1. When we account for this with the *Discourse-CQR* method, we observe boosts across all metrics. This suggests that naively using current models and systems to exploit explicit feedback through query rewriting are failure-prone. As a result, future 'feedback-aware' conversational query rewriters need to take the feedback type into consideration, in order to be effective.

Table 1: Retrieval performance of methods for query reformulation using explicit feedback. Sign † indicates a significant difference compared to the *organizer-auto* baseline.

Method	R	MAP	MRR	nDCG	nDCG@3
organizer-auto	0.348	0.155	0.533	0.311	0.365
+ QuReTeC	0.192	0.088	0.310	0.180	0.223
+ Rocchio	0.195	0.086	0.316	0.174	0.214
+ T5-CQR	0.340	0.131	0.500	0.288	0.329
+ RM3	0.388†	0.167†	0.565	0.343†	0.398†
+ Discourse-CQR	0.384†	0.174†	0.620†	0.348†	0.423†

5.1.2 Passage ranking with feedback. Across the board, we note that passage ranking with feedback leads to additional performance gains when used in a multi-step reranking setup. Specifically, the use of *FeedbackMonoT5* on top of selected participant submissions to TREC CAsT'22 leads to boosts in nDCG@3, nDCG, and MRR scores at various reranking thresholds. Although we only report the results of ranking the top 100 passages in Table 2, we observe similar trends when reranking at depth 10 and 50, and expect that these observations continue beyond the depth of 100. We further note that the magnitude of the improvement explicit feedback brings for retrieval varies between these participant systems, indicating that the effectiveness of explicit feedback may depend on the underlying characteristics of each system.

We note that the addition of *FeedbackMonoT5* leads to an average 6% gain in nDCG@3. These results are consistent for the MRR metric too as *FeedbackMonoT5* provides an average 7% gain. Showing that explicit feedback can be useful in improving the overall retrieval. This is not just due to the quality of the MonoT5 passage ranker but is a result of the additional context from explicit feedback.

We delve deeper into the queries where the delta in nDCG@3 before and after feedback ranking is at least 0.5 points in the *splade_t5mm_ens* run. We observe that passage ranking with feedback hurts performance in cases of positive feedback (*"Thanks,"* and negative feedback without clarification (*"Can you please answer my question?"*), whereas negative feedback with clarification boosts performance (*"That's interesting, but what makes the beef so special?"*). Feedback that introduces more explicit context is more useful. As with query rewriting, this phenomenon suggests that ranking models should be feedback aware.

5.1.3 Clarification and answer processing. Table 3 shows performance of three clarifying question construction methods, described in Section 4.3.3. 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.

Table 2: Retrieval performance of passage ranking using explicit feedback on top of selected CAsT participant systems. This reranking step only reranks the first 100 passages from each system.

System	MAP	MRR	nDCG	nDCG@3
organizer-auto	0.155	0.533	0.311	0.365
+ MonoT5	0.093	0.315	0.257	0.189
+ FeedbackMonoT5	0.152	0.560	0.313	0.387
splade_t5mm_ens	0.217	0.585	0.479	0.411
+ MonoT5	0.221	0.614	0.484	0.417
+ FeedbackMonoT5	0.226	0.632	0.489	0.442
uis_sparseboat	0.187	0.559	0.407	0.383
+ MonoT5	0.177	0.581	0.399	0.381
+ FeedbackMonoT5	0.184	0.611	0.408	0.415
UWCcano22	0.213	0.617	0.441	0.438
+ MonoT5	0.217	0.612	0.443	0.427
+ FeedbackMonoT5	0.217	0.659	0.454	0.454

Table 3: 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

5.2 User simulator

5.2.1 Single- and multi-turn clarifying question answering. Table 4 presents the results in comparison to *USi* [48] 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 [48]. 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.

5.2.2 Generated feedback evaluation. Table 5 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 3.2.2. 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

Table 4: Results of crowdsourcing study assessing naturalness and usefulness of generated answers to clarifying questions in single- and multi-turn scenarios. Each value indicates the percentage of pairwise comparisons won by the specific model as well as ties. Sign † indicates a significant difference.

		ConvSim	*USi* [48]	Ties	*ConvSim*	Human	Ties
Single	Naturalness	37%†	22%	41%	36%	25%	39%
	Usefulness	44%†	19%	37%	36%†	20%	44%
Multi	Naturalness	45%†	18%	37%	25%	28%	47%
	Usefulness	62%†	12%	26%	26%	16%	58%

Table 5: 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

Table 6: Passage ranking using explicit feedback on top of select CAsT participant runs. Runs are evaluated on a subset of queries annotated to require initiative.

System	MAP	MRR	nDCG	nDCG@3
organizer-auto	0.091	0.567	0.251	0.392
+ FeedbackMonoT5	0.101	0.589	0.256	0.404
splade_t5mm_ens	0.168	0.597	0.444	0.424
+ FeedbackMonoT5	0.195	0.659	0.463	0.492
uis_sparseboat	0.137	0.739	0.381	0.445
+ FeedbackMonoT5	0.133	0.733	0.378	0.490
UWCcano22	0.145	0.661	0.388	0.386
+ FeedbackMonoT5	0.150	0.610	0.395	0.393

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 DISCUSSION AND ANALYSIS

Does feedback help where it matters? Section 5 shows that systems that leverage feedback outperform systems that do not use it. We investigate a subset of 24 queries that require initiative as annotated by organizers [33]. These turns require additional user input and are typically open-ended or a branching point. Systems that exploit user input should perform better on these queries than systems that do not. Table 6 shows results of feedback passage ranking method on top of the participant runs introduced in table 6. Using feedback ranking *FeedbackMonoT5* leads to non-significant improvements across most metrics for all runs with an average increase of 7.75% in nDCG@3 with other metrics being similar.

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 *FeedbackMonoT5* passage ranker for 10 rounds of feedback. For efficiency we only apply re-ranking to the first 100 passages retrieved. Figure 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.

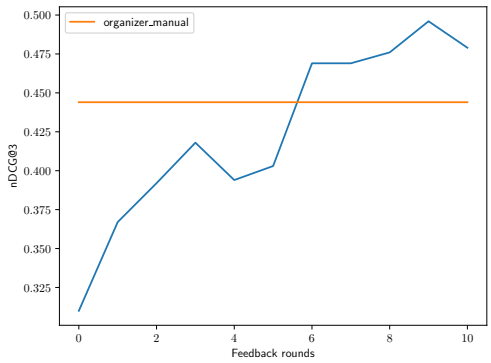

Figure 2: Multiple rounds of feedback using the *organiser-auto+Discourse-CQR+FeedbackMonoT5* system. The orange line depicts the performance of *organizer_manual*.

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.

7 CONCLUSIONS

We study the effectiveness of mixed-initiative conversational search models in combination with simulated user feedback. Specifically, we compare and extend proven models with an aim of incorporating user feedback, including answers to clarifying questions and explicit feedback on system's responses. We propose a new user simulator, *ConvSim*, capable of multi-turn interaction, leveraging LLMs. The results show utilizing feedback consistently improves retrieval across the majority of the methods, resulting in +16% improvement in nDCG@3 after a single turn of feedback. Moreover, we show that several rounds of feedback result in even greater boost (+35% after three rounds). This promises potential for advancements in CS and calls for further work on feedback processing methods.

8 ACKNOWLEDGMENTS

This work is supported by the Engineering and Physical Sciences Research Council (EPSRC) grant EP/V025708/1 and a 2019 Google Research Award.

REFERENCES

[1] Jafar Afzali, Aleksander Mark Drzewiecki, Krisztian Balog, and Shuo Zhang. 2023. UserSimCRS: A User Simulation Toolkit for Evaluating Conversational Recommender Systems. *arXiv preprint arXiv:2301.05544* (2023).

[2] Mohammad Aliannejadi, Leif Azzopardi, Hamed Zamani, Evangelos Kanoulas, Paul Thomas, and Nick Craswell. 2021. 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*. ACM, 16–26.

[3] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2020. ConvAI3: Generating Clarifying Questions for Open-Domain Dialogue Systems (ClariQ). (2020).

[4] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail S. Burtsev. 2021. Building and Evaluating Open-Domain Dialogue Corpora with Clarifying Questions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Association for Computational Linguistics, 4473–4484.

[5] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *SIGIR*. 475–484.

[6] Avishek Anand, Lawrence Cavedon, Hideo Joho, Mark Sanderson, and Benno Stein. 2020. Conversational Search (Dagstuhl Seminar 19461). In *Dagstuhl Reports*, Vol. 9. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[7] Leif Azzopardi. 2011. The economics in interactive information retrieval. In *SIGIR*. ACM, 15–24.

[8] Leif Azzopardi, Mohammad Aliannejadi, and Evangelos Kanoulas. 2022. Towards Building Economic Models of Conversational Search. In *European Conference on Information Retrieval*. Springer, 31–38.

[9] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268* (2016).

[10] Krisztian Balog. 2021. Conversational AI from an information retrieval perspective: Remaining challenges and a case for user simulation. (2021).

[11] Guorui Bian, Michael McAleer, and Wing-Keung Wong. 2011. A trinomial test for paired data when there are many ties. *Mathematics and Computers in Simulation* 81, 6 (2011), 1153–1160.

[12] Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. 2017. What do you mean exactly? Analyzing clarification questions in CQA. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*. 345–348.

[13] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[14] Ben Carterette, Evangelos Kanoulas, and Emine Yilmaz. 2011. Simulating simple user behavior for system effectiveness evaluation. In *CIKM*. 611–620.

[15] 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 Sankaranarayana 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. 2022. PaLM: Scaling Language Modeling with Pathways. https://doi.org/10.48550/ARXIV.2204.02311

[16] Aleksandr Chuklin, Aliaksei Severyn, Johanne R Trippas, Enrique Alfonseca, Hanna Silen, and Damiano Spina. 2019. Using audio transformations to improve comprehension in voice question answering. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, 164–170.

[17] Charles L Clarke, Nick Craswell, and Ian Soboroff. 2009. *Overview of the trec 2009 web track*. Technical Report. WATERLOO UNIV (ONTARIO).

[18] Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2020. TREC CAsT 2019: The conversational assistance track overview. *arXiv preprint arXiv:2003.13624* (2020).

[19] Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echegoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. 2021. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review* 54, 1 (2021), 755–810.

[20] Xiao Fu, Emine Yilmaz, and Aldo Lipani. 2022. Evaluating the Cranfield Paradigm for Conversational Search Systems. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval*. 275–280.

[21] Ahmed Hassan, Rosie Jones, and Kristina Lisa Klinkner. 2010. 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*. 221–230.

[22] Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *CHI*. 159–166.

[23] Kalervo Järvelin, Susan L Price, Lois ML Delcambre, and Marianne Lykke Nielsen. 2008. 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*. Springer, 4–15.

[24] Antonios Minas Krasakis, Mohammad Aliannejadi, Nikos Voskarides, and Evangelos Kanoulas. 2020. 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*. 129–132.

[25] Weronika Lajewska, Nolwenn Bernard, Ivica Kostric, Ivan Sekulic, and Krisztian Balog. 2022. The University of Stavanger (IAI) at the TREC 2022 Conversational Assistance Track. (2022).

[26] Victor Lavrenko and W Bruce Croft. 2009. *A generative theory of relevance*. Vol. 26. Springer.

[27] Margaret Li, Jason Weston, and Stephen Roller. 2019. Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons. *arXiv preprint arXiv:1909.03087* (2019).

[28] Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. 2020. Multi-Stage Conversational Passage Retrieval: An Approach to Fusing Term Importance Estimation and Neural Query Rewriting. https://doi.org/10.48550/ARXIV.2005.02230

[29] Aldo Lipani, Ben Carterette, and Emine Yilmaz. 2021. How Am I Doing?: Evaluating Conversational Search Systems Offline. *ACM TOIS* (2021).

[30] Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation. In *EMNLP*. 2122–2132.

[31] Javed Mostafa, Snehasis Mukhopadhyay, and Mathew Palakal. 2003. Simulation studies of different dimensions of users' interests and their impact on user modeling and information filtering. *Information Retrieval* 6, 2 (2003), 199–223.

[32] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 708–718. https://doi.org/10.18653/v1/2020.findings-emnlp.63

[33] Paul Owoicho, Jeffery Dalton, Mohammad Aliannejadi, Leif Azzopardi, Johanne R. Trippas, and Svitlana Vakulenko. 2022. TREC CAsT 2022: Going Beyond User Ask and System Retrieve with Initiative and Response Generation. (2022).

[34] Teemu Pääkkönen, Jaana Kekäläinen, Heikki Keskustalo, Leif Azzopardi, David Maxwell, and Kalervo Järvelin. 2017. Validating simulated interaction for retrieval evaluation. *Information Retrieval Journal* 20 (2017), 338–362.

[35] Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot Natural Language Generation for Task-Oriented Dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. 172–182.

[36] Gustavo Penha and Claudia Hauff. 2020. Challenges in the Evaluation of Conversational Search Systems. *KDD Workshop on Conversational Systems Towards Mainstream Adoption* (2020).

[37] Marco Ponza, Paolo Ferragina, and Francesco Piccinno. 2019. Swat: A system for detecting salient Wikipedia entities in texts. *Computational Intelligence* 35, 4 (2019), 858–890.

[38] Chen Qu, Liu Yang, W. Bruce Croft, Johanne R. Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and Characterizing User Intent in Information-seeking Conversations. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*. ACM, 989–992.

[39] Chen Qu, Liu Yang, W Bruce Croft, Yongfeng Zhang, Johanne R Trippas, and Minghui Qiu. 2019. User intent prediction in information-seeking conversations. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval (CHIIR '19)*. 25–33.

[40] Filip Radlinski and Nick Craswell. 2017. A theoretical framework for conversational search. In *CHIIR*. 117–126.

[41] Sudha Rao and Hal Daumé III. 2018. Learning to Ask Good Questions: Ranking Clarification Questions using Neural Expected Value of Perfect Information. In *ACL*. 2737–2746.

[42] Sudha Rao and Hal Daumé III. 2019. Answer-based adversarial training for generating clarification questions. *arXiv preprint arXiv:1904.02281* (2019).

[43] Joseph John Rocchio Jr. 1971. Relevance feedback in information retrieval. *The SMART retrieval system: experiments in automatic document processing* (1971).

[44] Corbin Rosset, Chenyan Xiong, Xia Song, Daniel Campos, Nick Craswell, Saurabh Tiwary, and Paul Bennett. 2020. Leading conversational search by suggesting useful questions. In *TheWebConference*. 1160–1170.

[45] Ananya B Sai, Akash Kumar Mohankumar, and Mitesh M Khapra. 2022. A survey of evaluation metrics used for NLG systems. *ACM Computing Surveys (CSUR)* 55, 2 (2022), 1–39.

[46] Alexandre Salle, Shervin Malmasi, Oleg Rokhlenko, and Eugene Agichtein. 2021. Studying the Effectiveness of Conversational Search Refinement Through User Simulation. In *ECIR*. 587–602.

[47] Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. 2021. 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* (Virtual Event) *(ICTIR '21)*. Association for Computing Machinery.

[48] Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. 2022. Evaluating Mixed-initiative Conversational Search Systems via User Simulation. In *WSDM '22: International Conference on Web Search and Data Mining* (Phoenix, AZ).

[49] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems* 33 (2020), 16857–16867.

[50] Amanda Stent, Matthew Marge, and Mohit Singhai. 2005. Evaluating evaluation methods for generation in the presence of variation. In *international conference on intelligent text processing and computational linguistics*. Springer, 341–351.

[51] Svetlana Stoyanchev, Alex Liu, and Julia Hirschberg. 2014. Towards natural clarification questions in dialogue systems. In *AISB symposium on questions, discourse and dialogue*, Vol. 20.

[52] Weiwei Sun, Shuo Zhang, Krisztian Balog, Zhaochun Ren, Pengjie Ren, Zhumin Chen, and Maarten de Rijke. 2021. 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*. 2499–2506.

[53] Svitlana Vakulenko, Kate Revoredo, Claudio Di Ciccio, and Maarten de Rijke. 2019. QRFA: A Data-Driven Model of Information Seeking Dialogues. In *Advances in Information Retrieval*. Springer International Publishing, 541–557.

[54] Svitlana Vakulenko, Nikos Voskarides, Zhucheng Tu, and Shayne Longpre. 2021. A Comparison of Question Rewriting Methods for Conversational Passage Retrieval. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021 (ECIR '21)*. 418–424.

[55] Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. 2020. Query Resolution for Conversational Search with Limited Supervision. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM. https://doi.org/10.1145/3397271.3401130

[56] Marilyn A Walker and Steve Whittaker. 1990. Mixed Initiative in Dialogue: An Investigation into Discourse Segmentation. In *ACL*.

[57] Shi Yu, Jiahua Liu, Jingqin Yang, Chenyan Xiong, Paul Bennett, Jianfeng Gao, and Zhiyuan Liu. 2020. Few-shot generative conversational query rewriting. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 1933–1936.

[58] Shi Yu, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu. 2021. Few-shot conversational dense retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 829–838.

[59] Hamed Zamani, Susan Dumais, Nick Craswell, Paul Bennett, and Gord Lueck. 2020. Generating clarifying questions for information retrieval. In *TheWebConference*. 418–428.

[60] Hamed Zamani, Gord Lueck, Everest Chen, Rodolfo Quispe, Flint Luu, and Nick Craswell. 2020. Mimics: A large-scale data collection for search clarification. In *CIKM*.

[61] Hamed Zamani, Bhaskar Mitra, Everest Chen, Gord Lueck, Fernando Diaz, Paul N Bennett, Nick Craswell, and Susan T Dumais. 2020. Analyzing and Learning from User Interactions for Search Clarification. *arXiv preprint arXiv:2006.00166* (2020).

[62] Hamed Zamani, Johanne R Trippas, Jeff Dalton, and Filip Radlinski. 2022. Conversational information seeking. *arXiv preprint arXiv:2201.08808* (2022).

[63] Shuo Zhang and Krisztian Balog. 2020. Evaluating Conversational Recommender Systems via User Simulation. In *KDD*. 1512–1520.

[64] Jie Zou, Evangelos Kanoulas, and Yiqun Liu. 2020. An Empirical Study on Clarifying Question-Based Systems. In *CIKM*. 2361–2364.

[65] Jie Zou, Aixin Sun, Cheng Long, Mohammad Aliannejadi, and Evangelos Kanoulas. 2023. Asking Clarifying Questions: To benefit or to disturb users in Web search? *Information Processing & Management* 60, 2 (2023), 103176. https://doi.org/10.1016/j.ipm.2022.103176