Clarifying Ambiguities: on the Role of Ambiguity Types in Prompting Methods for Clarification Generation

Anfu Tang*
Sorbonne Université, CNRS, ISIR
F-75005 Paris, France
tang@isir.upmc.fr

Laure Soulier
Sorbonne Université, CNRS, ISIR
F-75005 Paris, France
laure.soulier@isir.upmc.fr

Vincent Guigue
AgroParisTech, UMR MIA-PS
Palaiseau, France
vincent.guigue@agroparistech.fr

Abstract

In information retrieval (IR), providing appropriate clarifications to better understand users' information needs is crucial for building a proactive search-oriented dialogue system. Due to the strong in-context learning ability of large language models (LLMs), recent studies investigate prompting methods to generate clarifications using few-shot or Chain of Thought (CoT) prompts. However, vanilla CoT prompting does not distinguish the characteristics of different information needs, making it difficult to understand how LLMs resolve ambiguities in user queries. In this work, we focus on the concept of ambiguity for clarification, seeking to model and integrate ambiguities in the clarification process. Following the reasoning and acting paradigm, we propose a new prompting scheme Ambiguity Type-Chain of Thought (AT-CoT), which enhances the reasoning abilities of LLMs by limiting CoT to first predict ambiguity types that can be interpreted as actions, then generate clarifications correspondingly. Experiments are conducted on various datasets containing human-annotated clarifying questions to compare AT-CoT with multiple baselines. We also perform user simulation to implicitly measure the quality of generated clarifications under various IR scenarios. Our codes are available at: https://github.com/anfutang/ClarifyingAmbiguities/.

CCS Concepts

• **Information systems** → **Information retrieval**.

Keywords

Clarifying Question, Dialogue Search System, Ambiguity Type

ACM Reference Format:

Anfu Tang, Laure Soulier, and Vincent Guigue. 2025. Clarifying Ambiguities: on the Role of Ambiguity Types in Prompting Methods for Clarification Generation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3726302.3729922

*Corresponding author.

SIGIR '25, July 13–18, 2025, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3729922

1 Introduction

Ambiguity in information retrieval (IR) is a common factor that could undermine the quality of the retrieved documents. Indeed, real-world users often provide ambiguous queries to initialize a search without further elaboration [6]. The reasons for this ambiguity can vary [6, 44], such as avoiding the effort of typing lengthy queries, uncertainty about information needs, the tip-of-the-tongue phenomenon [4], etc. The ambiguity of natural language itself could also account for this ambiguity in user queries, such as synonyms or polysemies [33]. Regardless of the different causes, ambiguity is essentially a form of uncertainty, i.e. we cannot discern users' real intents by a single query. To better understand the ambiguities underlying user queries, previous studies have investigated ambiguity types (ATs) and proposed different taxonomies [15, 30, 43] to classify ambiguities.

To navigate users through the ambiguity, we need a method that facilitates users in expressing their needs without compromising their user experience. Previous studies seek to achieve this goal by building proactive search-oriented dialogue systems [49], which can take the initiative to provide information or suggestions to help improve the quality of search results, including providing clarifications. Instead of passively receiving a list of documents, users can actively participate in the search by communicating with the proactive dialogue system through conversations. Early studies on clarifications focus on reformulated queries [12, 34], which seek to provide useful suggestions that may meet the user's need, without explicitly exploiting the user's intent. Recent studies focus more on asking clarifying questions [2, 50], which consists of providing a clarifying question and allowing the user to respond freely. Clarification generation methods have evolved with the development of large language model (LLM), from supervised methods that rely on human-annotated data [3, 30], to LLM prompting methods [56, 61], among which Chain of Thought (CoT) prompting [57] is found to generate better clarifying questions [24, 61] compared to prompts with no generated reasoning. However, previous work mostly uses CoT prompts to freely generate reasoning, without explicitly asking LLMs to distinguish different information needs. We argue that understanding and integrating ambiguities into reasoning is important for the clarification process, since humans may first categorize the scenario of ambiguities, and then decide how to clarify the query properly. To simulate how humans handle ambiguous queries, we seek to first analyze the concept of ambiguity from the perspective of ambiguity types, and then integrate them into reasoning for clarification generation. To achieve this, we combine ambiguity types with CoT prompting to build Ambiguity Type-Chain of Thought (AT-CoT), which prompts LLMs to predict Ambiguity

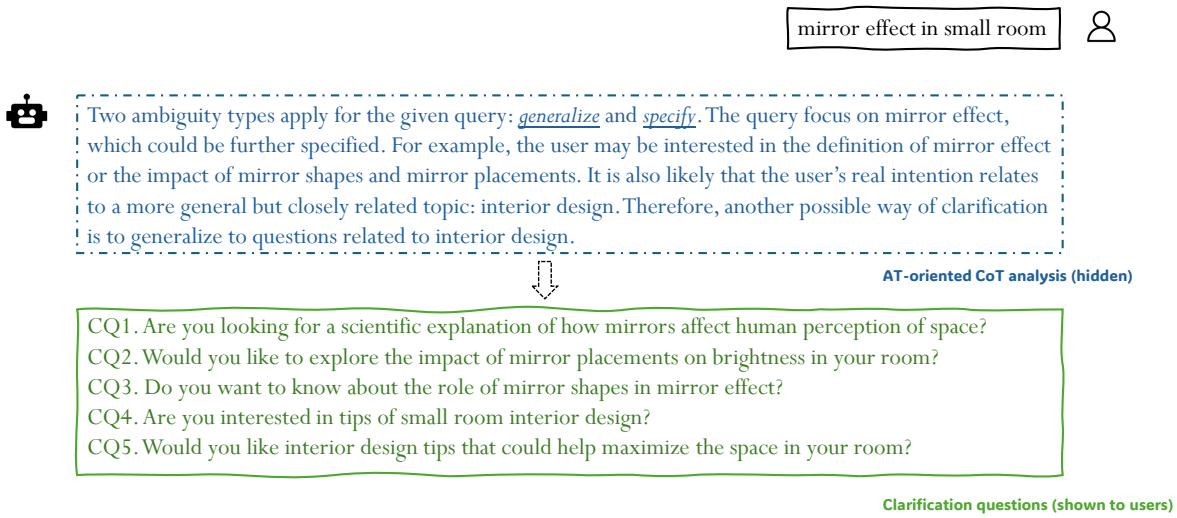

Figure 1: Illustration of AT-CoT: Unlike vanilla CoT, LLM-generated reasoning is limited to predict possible ambiguity types.

Types (ATs) that underlie a given query before generating clarifications correspondingly. To make AT-CoT properly work, we distill an action-based AT taxonomy from existing studies. Each AT in our taxonomy serves not only the purpose of helping LLMs understand ambiguity causes, but can also be interpreted as an instruction for LLMs to generate clarifications. Figure 1 illustrates AT-CoT. Provided a query *mirror effect in small room*, our method first predicts two ATs, then performs the corresponding actions to generate clarifying questions (CQs): CQ4 and CQ5 *generalize* the query; CQ1, CQ2, and CQ3 *specify* the query.

To validate the effectiveness of our method, experiments are carried out on both intrinsic and extrinsic tasks (resp. clarification generation and IR) on numerous datasets including Qulac [2], ClariQ [1], and TREC IR collections [15–18]. For the IR task, following previous work [2, 29, 63], we perform user simulation to generate multi-turn conversations and then transform the generated conversations to reformulated queries. We compare different clarification interaction scenarios such as proposing query reformulations for users to select (*select*) and asking a single clarifying question for users to respond (*respond*). To summarize, our main contribution is as follows:

- We analyze ambiguities from the perspective of ambiguity types, comprehensively investigate the impact of integrating ambiguities and reasoning in LLM prompting methods for clarification.
- We validate the effectiveness of our method through experiments on clarification generation and IR tasks.

2 Related Work

2.1 Ambiguity in User Queries

While there is a lack of a widely accepted taxonomy of ambiguities, ambiguous queries have been long studied in the IR community [15, 59, 61]. Previous work on ambiguity types (ATs) can be categorized into three types. The first group of studies formulates an AT taxonomy by analyzing queries in specific datasets [3, 15, 30, 43]. For instance, Guo et al. [30] proposed a taxonomy based on ambiguous questions in Abg-CoQA [30] with four ambiguity types: *Coreference Resolution* (unclear reference of pronouns), *Time Dependency* (the interpretation of question depends on time), *Answer Types* (multiple answer possibilities) and *Event References* (an entity in the question corresponds to multiple events). However, taxonomies in these studies are proposed more for analytical purposes and contain very specific ATs (e.g. *Entity References* [43] and *Coreference Resolution* [30] both correspond to a specific type of semantic ambiguity). Unlike these studies, we seek in this paper to formulate an AT taxonomy containing mutually exclusive ATs that can help LLMs generate better clarifications, rather than analyzing in detail why a query is ambiguous. Another group of studies focuses on the relations between queries by mining query logs [8, 31, 36, 59], mostly based on sampled query reformulations from query logs. Although these studies may not be directly related to clarification generation, their findings provide useful insights into clarification patterns. For instance, two common query reformulation patterns observed [8, 31] are *Generalization* and *Specialization*. While the latter is widely considered in studies related to clarification, the need for generalization is less investigated. We argue that generalization can also help specify the information needs of users in certain scenarios. It corresponds to an important dimension of ambiguity, reflecting the possibility that user queries may fail to accurately convey user intent. The last group consists of studies in the post-LLM era. We notice a recent work [61] that proposed a well-organized taxonomy with a special focus on ambiguities specific to LLMs, such as misaligned interpretations of queries between LLMs and humans. Our work differs from theirs: their taxonomy is used more as a tool to evaluate LLM performances in handling ambiguous queries, while our work focuses on integrating ambiguity types into reasoning for clarification generation. In a nutshell, previous work mostly exploit ATs for analysis, without searching to enhance the reasoning ability of LLM prompting by integrating

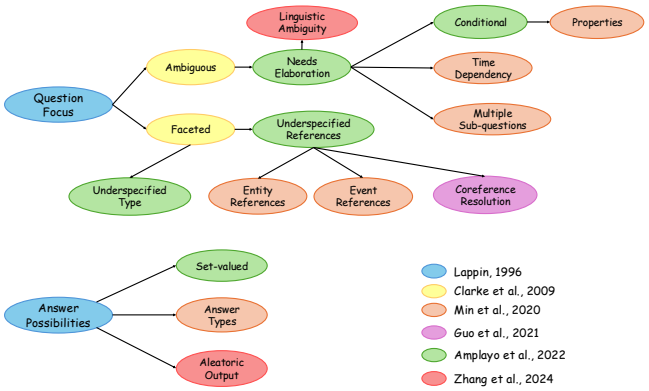

Figure 2: Ambiguity type taxonomies for analytical purposes in previous work.

ATs. To help better understand previous work, we organize ATs in existing taxonomies and present them in Figure 2.

2.2 Clarification in Information Retrieval

Clarification serves the purpose of eliciting the user's information need [59] by exchanging with the user and exploiting the user's feedback. The clarification form could be diverse, either by proposing reformulated queries to diversify the retrieval results or by asking clarifying questions to induce users to clarify. Early attempts of clarification generation focus on automatic query expansion [12], whereby users' original queries are rewritten or augmented. For example, Chirita et al. [13] proposed to expand user queries with terms collected from user data to handle ambiguities of short keyword queries in web searches. Other studies [11, 23, 42] investigate query suggestions by exploring different user-specific data sources such as landing pages, clicks, or hitting time. Both query expansion and query suggestions can be regarded as a form of proposing reformulated queries to users and collecting users' feedback. Recent studies concentrate more on asking clarifying questions [2, 37, 50, 51, 58]. The common approach consists of using the conversation history as input to a generative language model to generate CQs.

Before the era of LLM, clarification generation methods mostly consist of training sequence-to-sequence neural models (e.g. seq2seq [53]) using labeled data. For example, Guo et al. [30] fine-tune BART [38] to generate clarifying questions with ambiguous questions provided as input; Xu et al. [58] investigated knowledge-based clarifying question generation and concatenated entity texts and the current question as input to a Seq2seq [5] model. Recent studies in the post-LLM era have increasingly focused on LLM prompting methods, such as using few-shot prompting [37, 61], Chain of Thought (CoT) prompting [24, 61]. Our work extends existing studies on LLM prompting methods for clarification generation by integrating ambiguity types into CoT reasoning.

2.3 Conversation Simulation in Search

In IR, user simulation consists of creating artificial conversations based on hypotheses about user behaviors, often used to automatically test the performance of dialogue systems without performing real user tests [14, 26, 27]. The common approach is to instantiate a user agent to communicate with the dialogue system according to certain strategies [9, 28, 41]. Hypotheses about user behavior are made to control how user agents respond, depending on the purpose of the simulation. For example, to test the quality of reformulation in IR systems, Erbacher et al. [28] assumed that the user agent is greedy and fully cooperative, thus always selecting the reformulations most similar to the user intent. In another work [29], user agents are allowed to only respond 'yes' or 'no' to augment IR datasets with multi-turn conversations. Some studies involve simulation of more complex user behaviors, in which user agents are initialized with different variables, each corresponding to a specific type of users. Recent studies have increasingly focused on LLM-based conversation simulation. For example, Owoicho et al. [47] built a user simulator for mixed-initiative multi-turn conversation systems by prompting LLMs. Following previous work on LLM-based conversation simulation in IR, in our work, we instantiate our user agent using LLMs and simulate user responses by few-shot LLM prompting.

3 Methodology

We present here the outline of our methodology. We focus on the following research questions:

RQ1. What is an appropriate taxonomy of ambiguities for generating clarifying questions that is compatible with LLM prompting methods?

RQ2. How to integrate ambiguity and reasoning in LLM prompting methods for clarification?

3.1 Ambiguity Type Taxonomy

To respond to RQ1, we first seek to exploit ambiguity types to concretize the concept of ambiguity. The goal is to establish a taxonomy that can be used to enhance LLM reasoning ability in terms of handling ambiguous queries. Previous work [3, 30, 43] proposed various ambiguity type (AT) taxonomies for the analytical purpose. However, from the perspective of helping LLMs understand ambiguities and better instructing LLMs to generate clarifications, we find existing taxonomies redundant and unsuitable for LLM prompting methods. Firstly, as evidenced by Zhang et al. [61], existing taxonomies were mostly proposed before the era of LLMs, some ATs lack clear definitions and ATs are not mutually exclusive. Secondly, ATs in existing taxonomies can be reduced to two actions that LLMs can take: *Determine the Query Interpretation* or *Further Specify the User Query*. Following Deng et al. [24] who proposed proactive prompting, i.e. making LLMs decide actions to take instead of simply responding to instructions, we propose an LLM action-based taxonomy that encompasses three dimensions, each corresponding to a clarification pattern discovered in previous work [15, 31, 59]:

- *Semantic*: accounts for ambiguity in query interpretations.
- *Generalize*: addresses ambiguity in information needs when users seek relevant yet more general information. It occurs when user queries do not precisely describe real user intents.
- *Specify*: addresses ambiguity in information needs when users seek more specific information. It occurs when user queries lack details and may correspond to a too large search scope. Most ATs in existing taxonomies can be categorized

Table 1: Proposed action-based ambiguity type.

Ambiguity Type	Definition	Related ATs from previous work
Semantic	The query is semantically ambiguous for several common reasons: it may include homonyms; a word in the query may refer to a specific entity while also functioning as a common word; or an entity mentioned in the query could refer to multiple distinct entities.	*Question Focus* [35] *Linguistic Ambiguity* [61]
Generalize	The query focuses on specific information; however, a broader, closely related query might better capture the user's true information needs.	*Generalization* [8, 31]
Specify	The query has a clear focus but may encompass too broad a research scope. It is possible to further narrow down this scope by providing more specific information related to the query.	*Faceted* [15] *Time Dependency* [43] *Underspecified References* [3]

under this category (e.g. *Needs Elaboration*, *Underspecified References* [3]).

Table 1 presents detailed explanations of our AT taxonomy. Compared to previous taxonomies, the strength of our taxonomy lies in its dual function: each AT in our taxonomy not only helps LLMs understand underlying ambiguities, but can also be easily interpreted as an action for LLMs to take.

3.2 Prompting Formulation for Clarification Generation

This section aims to respond to RQ2, i.e. how to integrate ambiguities, abstracted by the ambiguity type taxonomy in Section 3.1, into reasoning for LLM prompting methods. We aim to achieve this by constraining the reasoning of CoT prompting. Intuitively, we seek to require LLMs to predict ATs in our taxonomy to integrate ambiguities into reasoning, which endows LLMs the capability to reason in a way that we expect and take explainable actions to clarify ambiguous queries. We hypothesize that ambiguity-oriented reasoning is better than freely generated LLM reasoning. Therefore, we propose Ambiguity Type-Chain of Thought (AT-CoT) that extends CoT prompting. To effectively access the impact of integrating ambiguities into LLM reasoning, we use another two prompting schemes as baselines: standard prompting, which simply requires LLMs to generate clarifications without any intermediate steps; AT-standard prompting, for which we add definitions of ATs in our taxonomy into prompt instructions. We use AT-standard to validate the impact of simply informing LLMs of possible ATs without asking LLMs to generate reasoning. Table 2 shows detailed system instructions for different prompting schemes. Mathematically, each prompting method can be formulated as follows:

• **standard** [24]: Standard prompting relies only on inherent knowledge of LLMs to generate clarifications without intermediate steps. The objective of standard prompting is to maximize:

$$p(c|\mathcal{D}, C, q) \tag{1}$$

where c denotes the generated clarification, q denotes an ambiguous query, C denotes the conversation history, and $\mathcal{D}$ denotes the task description.

• **AT-standard**: The only difference between AT-standard and standard prompting is that AT definitions are included in the prompt:

$$p(c|\mathcal{D}, \mathcal{A}, C, q) \tag{2}$$

where $\mathcal{A}$ refers to the AT definitions from Table 1.

• **CoT (Chain of Thought)** [57]: CoT prompting requires LLMs to generate texts of reasoning before making clarifications (reasoning without constraints):

$$p(a, c|\mathcal{D}, C, q) \tag{3}$$

where a refers to the generated textual reasoning.

• **AT-CoT**: AT-CoT requires LLMs to first predict ATs from our taxonomy, then generate clarifications correspondingly. The objective of AT-CoT prompting is to maximize:

$$p(a, c|\mathcal{D}, \mathcal{A}, C, q) \tag{4}$$

4 Experimental Setup

To evaluate the effectiveness of AT-CoT, we conduct experiments on three types of tasks: (1) Clarification generation (CG), for which we use datasets containing human-annotated clarifying questions (CQs) and we evaluate by computing the semantic similarity between generated CQs and human-annotated ones. (2) Information retrieval (IR), for which we simulate multi-turn conversations and transform conversations into reformulated queries to retrieve documents. (3) CG+IR, for which we align CG performance and IR performance to investigate the correlation between the performance of CG and IR, i.e., if better clarifications could improve IR performance.

4.1 Datasets

We present datasets that are used in our experiments in this section. Table 3 summarizes the statistics of different datasets.

4.1.1 CG Datasets.

• Qulac [2]: Qulac uses queries from TREC web track 2009-2012. Annotators are asked to first figure out facets related to given queries by scanning snippets of web searching results using a search engine, then generate CQs to address the facets.

• ClariQ [1]: Similarly to Qulac, ClariQ is crowdsourced by annotating CQs for provided queries. Ambiguity level labels ranging from 1-4 are provided in ClariQ, with 4 representing extreme ambiguous queries.

• RaoCQ [51]: A domain-specific dataset containing clarifying

Table 2: Prompts of four prompting schemes: standard, AT-standard, CoT and AT-CoT. <AT definitions> is a placeholder for AT definitions in Table 1.

Prompt Type	System Instruction
standard	`Given a query in an information-seeking system, generate a clarifying question that you think is most appropriate to gain a better understanding of the user's intent.` `<query>`

AT-standard	`Given a query in an information-seeking system, generate a clarifying question that you think is most appropriate to gain a better understanding of the user's intent. The ambiguity of a query can be multifaceted, and there are multiple possible ambiguity types:`
	`<AT definitions>`
	`Consider the above ambiguity types when generating.`
	`<query>`

CoT	`Given a query in an information-seeking system, generate a clarifying question that you think is most appropriate to gain a better understanding of the user's intent.`
	`Before generating the clarifying question, provide a textual explanation of your reasoning about why the original query is ambiguous and how you plan to clarify it.`
	`<query>`

AT-CoT	`Given a query in an information-seeking system, generate a clarifying question that you think is most appropriate to gain a better understanding of the user's intent. The ambiguity of a query can be multifaceted, and there are multiple possible ambiguity types:`
	`<AT definitions>`
	`Before generating the clarifying question, provide a textual explanation of your reasoning about which types of ambiguity apply to the given query. Based on these ambiguity types, describe how you plan to clarify the original query.`
	`<query>`

question annotations. Annotators are asked to identify relevant CQs given (question, follow-up questions) pairs, where each question refers to an original question from a post on StackExchange, and follow-up questions are sampled from comments of the same post. Similarly to [51], in this work, we evaluate our methods on the subset with human annotations.

4.1.2 IR Datasets.

• TREC Web track 2009-2012 [15–18]: An IR dataset that focus on web search queries. We use ClueWeb09[1] Category B as the document collection which contains 50 million English web pages. Since facet-specific document relevance judgments are provided in TREC web track diversity tasks, we use facets as user intents in user simulation.

• TREC Web track 2013-2014 [19, 20]: As TREC Web track 2009-2012, TREC Web track 2013-2014 contains multifaceted web search queries, while including more focused topics to present more challenging queries. ClueWeb12[2] is used as the document collection.

• TREC DL Hard [40]: A benchmark containing queries from TREC DL 2019 & 2020 [21, 22], which we believe may require multi-turn clarification to resolve implied ambiguities. The queries in TREC DL Hard are sampled from MS Marco [10]. We use the MS Marco passage corpus as the document collection.

[0]https://lucene.apache.org/

[1]https://lemurproject.org/clueweb09.php/

[2]https://lemurproject.org/clueweb12/

Table 3: Statistics of datasets used in our experiments for three tasks: CG, IR, CG+IR.

Dataset	# queries	# CQs	# intents
Task 1: CG			
Qulac	198	2575	-
ClariQ	298	3991	-
RaoCQ	500	2248	-
Task 2: IR			
TREC Web Track 2009-2012	198	-	717
TREC Web Track 2013-2014	100	-	315
TREC DL Hard	50	-	350
Task 3: CG+IR			
Qulac-TREC Web Track 2009-2012	198	2575	717

4.1.3 CG+IR Dataset. Since Qulac is based on queries from TREC Web Track 2009-2012, we align Qulac queries to document relevance judgments provided by TREC Web Track 2009-2012. We refer to this dataset by Qulac-TREC Web Track 2009-2012, which contains human-annotated CQs from Qulac and document relevance judgments from TREC Web Track 2009-2012.

4.2 Evaluation Protocol

Clarification Generation. For each query, we generate multiple CQs to fairly evaluate the performance of different prompting methods on the clarification generation (CG) task. Several factors drive

this decision: 1) In CG datasets, each query corresponds to numerous human-annotated CQs, covering different clarification possibilities. However, human-annotated CQs cannot contain all possible CQs. Since automatic metrics such as BERTScore [60] capture the semantic similarity between generated CQs and reference CQs, it is likely that a high-quality generated CQ gets a low BERTScore. To mitigate this issue, we seek to generate multiple diverse CQs, therefore reducing the probability that none of the generated CQs is semantically similar to any of the human-annotated ones. 2) For AT-CoT, since multiple ambiguity types may exist for a query, it is natural to generate multiple CQs to account for different ATs. For other prompting methods that do not predict ATs, generating multiple CQs is also helpful for fair comparison. As evidenced in Wang et al. [54], the voting strategy can increase the performance of LLM prompting methods. Generating multiple CQs and comparing the best-performing CQ can be regarded as a type of voting, by which we reduce the variance of prompting performances on CG.

User Simulation for IR. We also evaluate the impact of our methodology on IR via user simulation. Two clarification interaction scenarios are tested (Figure 3):

• *select*: In each turn, 5 RQs are generated. We adopted a moderate temperature of 0.6 to balance the diversity and consistency of the generated RQs, ensuring that they are varied while avoiding excessive creativity. The user agent selects the RQ that best corresponds to their intent, and the conversation continues based on the selected RQ.

• *respond*: In each turn, a CQ is generated. The user agent responds to it based on the provided user intent.

We choose the two interaction scenarios for the following reasons: 1) *select* is widely studied in previous studies [11, 23], and applied in certain real-world scenarios such as search engine suggestions through query reformulation. 2) *respond* corresponds to the more naturalistic settings for conversational search modeling interactions in natural language [2, 37, 52]. A baseline w/o clarification is also considered, which uses original queries without clarification for document retrieval.

To simulate multi-turn conversations, three prompts are chained: *generation*, *response*, *reformulation*. The *generation* prompt has four variants, each representing a prompting methods as described in Section 3.2. We use this prompt to generate RQs (scenario *select*) or CQs (scenario *respond*). The *response* prompt generates simulated user responses and has two variants, each corresponding to an interaction scenario. The clarifications generated by *generation* and paragraphs describing user intents are used as input for the *response* prompt. Since facet-specific document relevance judgments are provided in TREC Web track 2009-2014, we directly use facets as user intents for *response* and use gold document relevance labels. For TREC DL Hard, we use each relevant document as user intent for *response*, then use the same document as the gold label in IR evaluation. For each simulated conversation, the *reformulation* prompt uses the simulated conversation as input and summarizes the conversation into a reformulated query. The objective of using *reformulation* is to facilitate the evaluation of IR tasks, since most existing IR models retrieve documents by queries rather than conversations. Examples of the *response* and *reformulation* prompts can be found in Table 4. As a result, we have eight types of simulated conversations, each corresponding to a unique (clarification generation prompt, interaction scenario) combination. To simplify the simulation, we assume that users are always cooperative and that no intent shifts occur during the conversation. Each conversation is initialized by a user query and simulated to three turns with no stopping rules. User agents are always provided with complete descriptions of user intents.

4.3 Evaluation Metric

Following [61], we use BERTScore [60] for CG tasks, since metrics based on N-gram matching like BLEU or ROUGE cannot measure clarification abilities [30]. As mentioned in Section 4.2, we ask LLMs to generate multiple CQs. For each query, we compute a score for generated CQs as follows: suppose that for each query q, we generate a list of CQs $(gcq_1, ..., gcq_M)$, with a list of annotated CQs $(acq_1, ..., acq_N)$ as gold standards. We first compute a query-specific score matrix S:

$$S_{i,j} = BERTScore(gcq_i, acq_j) \tag{5}$$

where $i = 1, ..., M$, $j = 1, ..., N$. The score on q is computed by: $score_q = max(S_{[:,:]})$, which is the maximum value of S. We take the BERTScore of the best-performing generated CQ to access the overall CG performance on q for the following reason: a CG method is good if it is able to generate a CQ that is highly similar to one of the reference CQs.

For the IR task, we use different standard metrics following previous work [15, 62]: We use nDCG@10 (Normalized Discounted Cumulative Gain [55]) for TREC web track 2009-2014. For TREC DL Hard, since we use each relevant document as user intent for simulation then verify whether the target document is ranked higher through clarification, we use MRR@10 (Mean Reciprocal Rank [48]) as the evaluation metric.

4.4 Implementation Details

Prompting Scheme. Following previous work [24], we adopt few-shot settings for all prompting schemes. We have two reasons to do so: 1) results of preliminary experiments demonstrate that few-shot prompting always significantly outperform zero-shot, regardless of the prompting scheme; 2) zero-shot prompting is likely to generate over lengthy analysis, causing incomplete generation due to maximum output token limitation or slowing down inference. In this work, we assume without further notice that all prompting methods are under few-shot settings.

LLM. We use Llama-3-8B [25] as our base model and load pretrained weights from Huggingface. LLM hyperparameters are fixed with: $k = 10$ for top-k sampling; temperature $t = 0.6$. Due to the extensive prompting inference involved in our experiments, we quantize Llama-3 to NF4 (4-bit NormalFloat) and conduct our experiments on a single 12G TITAN Xp GPU.

Parsing LLM Outputs. We ask LLMs to give JSON-style structured outputs through format instructions and few-shot examples containing reference formatted outputs. LLM outputs are parsed using the Pydantic parser from LangChain[3]. In case of parsing errors,

[3]https://python.langchain.com/v0.1/docs/modules/model_io/output_parsers/types/pydantic/

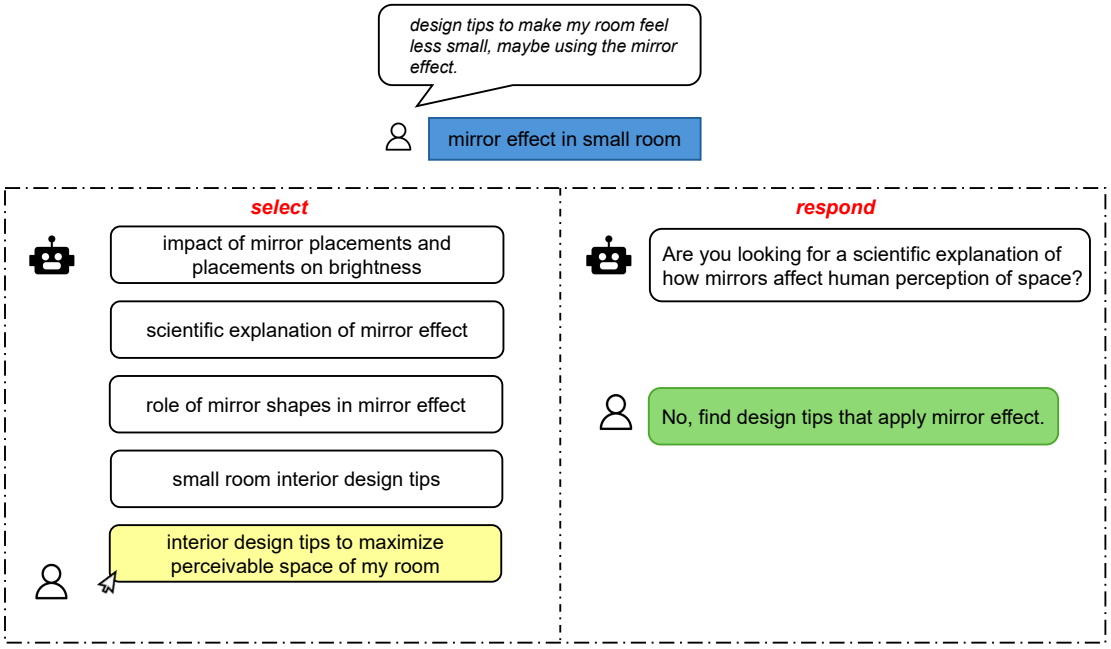

Figure 3: Illustration of the two clarification interaction scenarios in our user simulation: *select* and *respond*.

Table 4: Prompts of *respond* and *reformulation*. There is no *reformulation* prompt for the mode *select*.

Prompt type	System instruction
response (scenario *select*)	`Imagine that you are a user seeking information with the help of a conversational assistant. At each turn of the conversation, the assistant provides you several reformulated queries to better understand your intent. Given a conversation history and a paragraph describing the user intent, choose the reformulated query that most accurately reflects the provided user intent.` `<chat history>` `<user intent>`
response (scenario *respond*)	`Imagine that you are a user seeking information with the help of a conversational assistant. At each turn of the conversation, the assistant asks a clarification question to better understand your intent. Given a conversation history and a paragraph describing the user intent, respond to the clarification question based on the provided user intent.` `<chat history>` `<user intent>`
reformulation (scenario *respond*)	`Given a conversation history, summarize the conversation as a reformulated query. The conversation history includes the initial query and several clarification turns between the user and a virtual assistant.` `<chat history>`

we ask LLMs to regenerate the outputs with a maximum number of retry attempts set to 10. In rare cases, we manually parse the LLM outputs to address persistent parsing errors.

BERTScore. We use the third-ranked pre-trained model of BERTScore[4] based on their experimental results [5] on the WMT16 machine translation task [7].

IR Pipeline. Following [32, 45, 46], we adopt a two-stage retriever-rerank pipeline for IR tasks. Top-k relevant documents are first retrieved from a large-scale document collection using BM25 and then reranked using MonoT5. For the retriever, we use a no-tuning *pyserini*[6] Lucene implementation with k fixed to 100. For the reranker, we use a pre-trained MonoT5 [46].

[4]https://huggingface.co/microsoft/deberta-large-mnli
[5]https://github.com/Tiiiger/bert_score

[6]https://github.com/castorini/pyserini

IR Datasets. We use *ir_datasets* [39], a commonly used Python package in IR community to manage IR datasets. The Python implementation of *pyserini* is used to build BM25 indexes of Clueweb09 and Clueweb12.

5 Task 1: Clarification Generation (CG)

This section aims to evaluate the impact of integrating ambiguities into LLM reasoning on the performance of the clarification generation (CG) task. Table 5 depicts the overall comparison between different prompting methods using various datasets and the BERTScore metric. Results show that AT-CoT consistently outperforms the three baselines with significant margins across all datasets. For instance, AT-CoT reaches a BERTScore of 82 vs. scores ranging from 78.8 to 80 for other baselines on ClariQ. This suggests that ambiguity-oriented reasoning (AT-COT) helps generate better clarifying questions. This improvement is consistent on both specific-domain datasets (RaoCQ) and open-domain datasets (Qulac, ClariQ), showing that our method generalizes to different types of queries. Besides, through the comparison between AT-standard and standard prompting, we find that only informing LLMs of existing ambiguity types is not helpful, and even degrades the CG performance in some cases (e.g. 77 vs 77.9 for resp. AT-standard vs. standard on Qulac). This demonstrates that integrating ambiguity types is only helpful when integrated into LLM reasoning. Our observation on CoT prompting is coherent with previous work [24, 61]: CoT is more effective than standard prompting in terms of generating clarifying questions. We explore even further by claiming that ambiguity-oriented reasoning is more helpful.

Stratification by Ambiguity-level. We further evaluate the performance of CG tasks across different ambiguity levels. We use labels provided in ClariQ for this analysis, and present results in Table 6. Generally, both CoT and AT-CoT outperform standard and AT-standard prompting on the first three ambiguity levels, demonstrating the usefulness of both freely generated LLM reasoning and ambiguity-oriented reasoning for CG when queries are not extremely ambiguous. However, in cases of extreme ambiguity (level-4), the performance of CoT falls below standard prompting (BERTScore of 78 vs. 78.5 and 79.7 for standard and AT-standard resp.), meanwhile, the improvement of AT-CoT is still consistent (BERTScore=82.4). This suggests that ambiguity types could be particularly useful for handling ambiguous queries.

Distribution of Ambiguity Types. To provide more insight into AT-CoT, we analyze the distribution of ambiguity types predicted by AT-CoT. We first investigate the frequency of predicted ATs

Table 5: Overall evaluation on CG datasets. *, †, $^\Delta$ marks statistically significant improvements over standard, AT-standard, CoT respectively with $p < 0.01$ under a t-test.

Prompt	Qulac	ClariQ	RaoCQ
standard	77.9	79.3	60.0
AT-standard	77.0	78.8	59.9
CoT	79.2*	80.0	60.5
AT-CoT	**80.6**$^{*\dagger\Delta}$	**82.0**$^{*\dagger\Delta}$	**62.4**$^{*\dagger\Delta}$

Table 6: CG results on ClariQ stratified by ambiguity levels. *, †, $^\Delta$ marks statistically significant improvements over standard, AT-standard, CoT, respectively.

	level-1	level-2	level-3	level-4
standard	78.7	80.0	78.9	78.5
AT-standard	77.6	79.2	78.4	79.7
CoT	78.6	80.5	80.7†	78.0
AT-CoT	**80.9***†	**82.0**$^{*\dagger\Delta}$	**82.1**$^{*\dagger\Delta}$	**82.4**$^{\dagger\Delta}$

Table 7: Distribution of ATs predicted by AT-CoT. In parentheses, we show corresponding CG performance differences between AT-CoT and CoT regarding the BERTScore.

	Qulac	ClariQ	RaoCQ
Semantic	44.6 (↑ 1.3)	45.9 (↑ 1.8)	42.4 (↑ 2.0)
Generalize	1.7 (↓ 0.6)	1.9 (↑ 1.4)	12.3 (↑ 2.0)
Specify	53.7 (↑ 1.4)	52.2 (↑ 2.0)	45.3 (↑ 1.9)

(namely *Semantic*, *Generalize* and *Specify*), and then focus on the impact of predicting different ATs on the performance of the CG task. Predicted ATs are extracted from the reasoning generated by AT-CoT. Table 7 shows statistics about each group on all CG datasets: the frequency of queries identified as a specific AT and the performance difference in terms of BERTScore between AT-CoT and CoT (in parentheses). Our main conclusions are the following: 1) *Semantic* and *Specify* are the most frequent types for all CG datasets, with *Specify* being slightly more common (i.e. at most 45.9% vs. at most 53.7% for *Specify*). This observation aligns with the fact that most ATs in existing taxonomies can be categorized as *Semantic* or *Specify*. However, though less frequent, the importance of *Generalize* cannot be overlooked. 2) The *Generalize* type is more marginal but the fact that 12% of queries in RaoCQ are predicted to be generalizable justifies our decision to include *Generalize* in our taxonomy. 3) The observation that queries in RaoCQ more often require generalization suggests that the AT predictions of AT-CoT effectively capture the clarification needs of queries and are less likely to be random. Since RaoCQ queries are extracted from user posts on StackExchange, they are generally longer compared to queries in Qulac and ClariQ. It is therefore very likely that a query in RaoCQ does not precisely describe user intents and requires generalization. Differently, queries in Qulac are often short, used for web search, making them less possibly to require generalization. This gap between the frequency of *Generalize* being predicted and improvements caused by predicting *Generalize* reflects that AT-CoT adapts well to datasets with different characteristics.

6 Task 2: Information Retrieval (IR)

This section aims to investigate the impact of integrating ambiguity into LLM reasoning on IR performance. Table 8 shows IR results of the two different interaction scenarios (*select & respond*) and the baseline without clarification. We detail the result alongside three successive turns. Generally, we observe that AT-CoT > CoT > AT-standard ≈ standard for most of the interaction modes

Table 8: Results on IR datasets based on user simulation. Scores are in nDCG@10 (%) for Trec Web Track 2009-2012 and TREC Web Track 2013-2014; MRR@10 (%) for TREC DL Hard. *, †, $^{\Delta}$, indicates statistically significant improvements over standard, AT-standard, CoT respectively with $p < 0.01$ under a t-test.

	TREC Web Track 09-12		*TREC Web Track 13-14*		*TREC DL Hard*	
	select	*respond*	*select*	*respond*	*select*	*respond*
w/o clarification	0.123	0.123	0.277	0.277	0.084	0.084
Turn-1						
standard	0.161	0.232	0.336	0.387	0.060	0.120
AT-standard	0.165	0.230	0.337	0.383	0.066	0.113
CoT	0.174*†	0.238	0.341	0.392†	0.063	0.123†
AT-CoT	**0.188**$^{*\dagger\Delta}$	**0.244***†	**0.347***†	**0.397**†	**0.074**$^{*\Delta}$	**0.125**†
Turn-2						
standard	0.152	0.223	0.307	0.379	0.054	0.127
AT-standard	0.149	0.228	0.291	0.376	0.052	0.151*
CoT	0.160*†	0.226	0.310†	0.384	0.062†	0.174*†
AT-CoT	**0.176**$^{*\dagger\Delta}$	**0.233**	**0.320**$^{*\dagger\Delta}$	**0.391***†	**0.071**$^{*\dagger\Delta}$	**0.184**$^{*\dagger\Delta}$
Turn-3						
standard	0.141	0.212	0.295	0.371	**0.056**	0.141
AT-standard	0.149	0.213	0.276	0.367	0.051	0.154
CoT	0.148	**0.216**	0.300†	0.373	0.054	0.184*†
AT-CoT	**0.152**	0.213	**0.305**†	**0.381**	0.052	**0.188***†

and turns. For instance, clarifications obtained with the method AT-COT allow to reach the best IR metrics values for the TREC Web Track 2013-2014 dataset over all turns (0.397, 0.391, and 0.381 for each turn respectively vs 0.392, 0.384, and 0.373 at most for the baselines). We also note that IR performance is always better for the *respond* interaction mode corresponding to the generation of clarifying questions (in contrast to the *select* mode based on query reformulation. Altogether, these results highlight two main conclusions: 1) it aligns with our remarks on the CG performance, demonstrating the benefits of introducing ambiguity-oriented LLM reasoning for clarification, both intrinsically and extrinsically. And 2) this reinforces our hypothesis based on the need for clarification interactions based on ambiguity and reasoning in IR. Our findings also demonstrate the robustness of our methodology in interaction scenarios. For both interaction scenarios *select* and *respond*, AT-CoT consistently provides the best IR performance, implying that our method can adapt to various real-world scenarios such as query suggestion-based scenarios (e.g. search suggestions) or chat scenarios (e.g. chatbot).

Per-turn IR performance. We observe the same pattern of performance changing across multiple conversation turns for all prompting schemes. For example, under *select*, the IR performance reaches the highest value in the first turn, then monotonically decreases; for Trec DL Hard under *respond*, the IR performance steadily increases as the conversation continues. This IR performance changing pattern is coherent to query difficulties. As a collection that contains complex queries from Trec DL 2019/2020 datasets [21, 22], queries in Trec DL Hard are relatively longer, more challenging in terms of resolving ambiguities. Therefore, Trec DL Hard may necessitate multi-turn conversations to fully clarify ambiguities, which is reflected in the increasing scores across conversation turns. Similarly, for Trec Web Track datasets, the peak IR performance appearing at the first turn is reasonable, since queries in these datasets are not highly ambiguous. Nevertheless, in terms of turn-specific IR performances, AT-CoT still outperforms other prompting schemes, demonstrating that there is no need to increase conversation turns to reflect the improvements of AT-CoT. Regardless of how many turns of conversation a user intends to have, AT-CoT is able to provide better clarifications compared to other prompting schemes.

7 Task 3: Alignment between Clarification Generation & Information Retrieval

To mitigate potential bias introduced in user simulation, we further align the performance of CG and IR by using Qulac-Trec Web Track 2009-2012 as mentioned in Section 4.1.3. Since reference CQs in Qulac are only provided for initial queries, we use the CG and IR results from the first turn under *respond*. We compute the Pearson correlation coefficient to measure the strength of the linear relationship between the CG and IR results, and obtain $r = 0.92$, $p = 0.08$, which shows a strong positive correlation. We hypothesize that the insignificance of this correlation may due to the complexity of the document collection, which is insufficient to differentiate the quality of clarifications. A query may be refined by high-quality CQs through user simulation, but there lack of relevant documents to account for this refinement and reflect it by IR performance. However, given that we obtain a correlation coefficient greater than 0.9, it does not undermine our observation that IR performance is correlated to CG, i.e. IR performance improvements brought by AT-CoT are due to better clarifications.

8 Conclusion

In this work, we investigate the integration of ambiguities and reasoning into LLM prompting methods for clarification, proposing a new action-based ambiguity type taxonomy and a new prompting scheme, AT-CoT. Experiments on clarification generation and information retrieval datasets demonstrate the effectiveness of our methodology. Besides, in-depth analyses show that our method is robust in different clarification interaction scenarios and can capture the clarification needs of datasets with different characteristics.

However, our work is not without limitations. First, we establish an ambiguity type taxonomy containing three general ATs for integration with LLM reasoning. We do not experimentally study the impact of AT granularity, particularly investigating whether reasoning over a structured ambiguity taxonomy would be beneficial. Second, we only use Llama-3-8B without testing LLMs of different scales. It would be interesting to study the reasoning capability of different model scales. Nevertheless, we believe that our work acts as a foundation to better understand the role of ambiguity types in LLM prompting methods for clarification and may provide useful insights for future work.

Acknowledgments

This work benefited from support from the French National Research Agency (Project GUIDANCE, ANR-23-IAS1-0003) and SCAI (Sorbonne Center for Artificial Intelligence).

References

[1] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2021. Building and Evaluating Open-Domain Dialogue Corpora with Clarifying Questions. In *EMNLP*.

[2] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking Clarifying Questions in Open-Domain Information-Seeking Conversations. In *International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)* (Paris, France) *(SIGIR '19)*.

[3] Reinald Kim Amplayo, Kellie Webster, Michael Collins, Dipanjan Das, and Shashi Narayan. 2023. Query Refinement Prompts for Closed-Book Long-Form QA. In *Annual Meeting of the Association for Computational Linguistics*. https://api.semanticscholar.org/CorpusID:259370749

[4] Jaime Arguello, Adam Ferguson, Emery Fine, Bhaskar Mitra, Hamed Zamani, and Fernando Diaz. 2021. Tip of the tongue known-item retrieval: A case study in movie identification. In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*. 5–14.

[5] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR* abs/1409.0473 (2014). https://api.semanticscholar.org/CorpusID:11212020

[6] N. J. Belkin, D. Kelly, G. Kim, J.-Y. Kim, H.-J. Lee, G. Muresan, M.-C. Tang, X.-J. Yuan, and C. Cool. 2003. Query length in interactive information retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval* (Toronto, Canada) *(SIGIR '03)*. Association for Computing Machinery, New York, NY, USA, 205–212. https://doi.org/10.1145/860435.860474

[7] Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. 2016. Findings of the 2016 conference on machine translation (wmt16). In *First conference on machine translation*. Association for Computational Linguistics, 131–198.

[8] Paolo Boldi, Francesco Bonchi, Carlos Castillo, and Sebastiano Vigna. 2011. Query reformulation mining: models, patterns, and applications. *Inf. Retr.* 14, 3 (June 2011), 257–289. https://doi.org/10.1007/s10791-010-9155-3

[9] Arthur Câmara, David Maxwell, and Claudia Hauff. 2022. Searching, Learning, and Subtopic Ordering: A Simulation-Based Analysis. In *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I* (Stavanger, Norway). Springer-Verlag, Berlin, Heidelberg, 142–156. https://doi.org/10.1007/978-3-030-99736-6_10

[10] Daniel Fernando Campos, Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, Li Deng, and Bhaskar Mitra. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *ArXiv* abs/1611.09268 (2016). https://api.semanticscholar.org/CorpusID:1289517

[11] Huanhuan Cao, Daxin Jiang, Jian Pei, Qi He, Zhen Liao, Enhong Chen, and Hang Li. 2008. Context-aware query suggestion by mining click-through and session data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Las Vegas, Nevada, USA) *(KDD '08)*. Association for Computing Machinery, New York, NY, USA, 875–883. https://doi.org/10.1145/1401890.1401995

[12] Claudio Carpineto and Giovanni Romano. 2012. A Survey of Automatic Query Expansion in Information Retrieval. *ACM Comput. Surv.* 44, 1, Article 1 (jan 2012), 50 pages. https://doi.org/10.1145/2071389.2071390

[13] Paul Alexandru Chirita, Claudiu S. Firan, and Wolfgang Nejdl. 2007. Personalized query expansion for the web *(SIGIR '07)*. Association for Computing Machinery, New York, NY, USA, 7–14. https://doi.org/10.1145/1277741.1277746

[14] Aleksandr Chuklin, Pavel Serdyukov, and Maarten de Rijke. 2013. Modeling clicks beyond the first result page. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 1217–1220.

[15] Charles LA Clarke, Nick Craswell, and Ian Soboroff. 2009. Overview of the TREC 2009 Web Track.. In *Trec*, Vol. 9. 20–29.

[16] Charles LA Clarke, Nick Craswell, Ian Soboroff, and Ellen M Voorhees. 2011. Overview of the TREC 2011 Web Track.. In *TREC*.

[17] Charles L. A. Clarke, Nick Craswell, Ian Soboroff, and Gordon V. Cormack. 2010. Overview of the TREC 2010 Web Track. In *Text Retrieval Conference*. https://api.semanticscholar.org/CorpusID:16213318

[18] Charles L. A. Clarke, Nick Craswell, and Ellen M. Voorhees. 2012. Overview of the TREC 2012 Web Track. In *Text Retrieval Conference*. https://api.semanticscholar.org/CorpusID:11517775

[19] Kevyn Collins-Thompson, Paul N. Bennett, Fernando Diaz, Charles L. A. Clarke, and Ellen M. Voorhees. 2013. TREC 2013 Web Track Overview. In *Text Retrieval Conference*. https://api.semanticscholar.org/CorpusID:2180933

[20] Kevyn Collins-Thompson, Craig Macdonald, Paul N Bennett, Fernando Diaz, and Ellen M Voorhees. 2014. TREC 2014 Web Track Overview.. In *TREC*, Vol. 13. 1–15.

[21] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2020. Overview of the TREC 2020 deep learning track. In *TREC*.

[22] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen Voorhees. 2019. Overview of the TREC 2019 deep learning track. In *TREC 2019*.

[23] Silviu Cucerzan and Ryen W White. 2007. Query suggestion based on user landing pages. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 875–876.

[24] Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. 2023. Prompting and Evaluating Large Language Models for Proactive Dialogues: Clarification, Target-guided, and Non-collaboration. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 10602–10621. https://doi.org/10.18653/v1/2023.findings-emnlp.711

[25] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

[26] Georges E Dupret and Benjamin Piwowarski. 2008. A user browsing model to predict search engine click data from past observations.. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. 331–338.

[27] Wieland Eckert, Esther Levin, and Roberto Pieraccini. 1997. User modeling for spoken dialogue system evaluation. In *1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings*. IEEE, 80–87.

[28] Pierre Erbacher, Ludovic Denoyer, and Laure Soulier. 2022. Interactive query clarification and refinement via user simulation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2420–2425.

[29] Pierre ERBACHER, Jian-Yun Nie, Philippe Preux, and Laure Soulier. 2024. Augmenting Ad-Hoc IR Dataset for Interactive Conversational Search. *Transactions on Machine Learning Research* (2024). https://openreview.net/forum?id=z8d7nT1HWw

[30] Meiqi Guo, Mingda Zhang, Siva Reddy, and Malihe Alikhani. 2021. Abg-CoQA: Clarifying Ambiguity in Conversational Question Answering. In *3rd Conference on Automated Knowledge Base Construction*. https://openreview.net/forum?id=SlDZ1o8FsJU

[31] Bernard J. Jansen, Danielle L. Booth, and Amanda Spink. 2009. Patterns of query reformulation during Web searching. *J. Am. Soc. Inf. Sci. Technol.* 60, 7 (July 2009), 1358–1371.

[32] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 6769–6781. https://doi.org/10.18653/v1/2020.emnlp-main.550

[33] Robert Krovetz. 1997. Homonymy and polysemy in information retrieval. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*. 72–79.

[34] Saar Kuzi, Anna Shtok, and Oren Kurland. 2016. Query Expansion Using Word Embeddings. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (Indianapolis, Indiana, USA) *(CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 1929–1932. https://doi.org/10.1145/2983323.2983876

[35] Shalom Lappin (Ed.). 1996. *The Handbook of Contemporary Semantic Theory*. Blackwell Reference, Cambridge, Mass., USA. 329–423 pages.

[36] Tessa Lau and Eric Horvitz. 1999. Patterns of search: analyzing and modeling Web query refinement. In *Proceedings of the Seventh International Conference on User Modeling* (Banff, Canada) *(UM '99)*. Springer-Verlag, Berlin, Heidelberg, 119–128.

[37] Dongryeol Lee, Segwang Kim, Minwoo Lee, Hwanhee Lee, Joonsuk Park, Sang-Woo Lee, and Kyomin Jung. 2023. Asking Clarification Questions to Handle Ambiguity in Open-Domain QA. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 11526–11544. https://doi.org/10.18653/v1/2023.findings-emnlp.772

[38] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

[39] Sean MacAvaney, Andrew Yates, Sergey Feldman, Doug Downey, Arman Cohan, and Nazli Goharian. 2021. Simplified Data Wrangling with ir_datasets. In *SIGIR*.

[40] Iain Mackie, Jeffrey Dalton, and Andrew Yates. 2021. How deep is your learning: The DL-HARD annotated deep learning dataset. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2335–2341.

[41] David Maxwell, Leif Azzopardi, Kalervo Järvelin, and Heikki Keskustalo. 2015. Searching and stopping: An analysis of stopping rules and strategies. In *Proceedings of the 24th ACM international on conference on information and knowledge management*. 313–322.

[42] Qiaozhu Mei, Dengyong Zhou, and Kenneth Church. 2008. Query suggestion using hitting time. In *Proceedings of the 17th ACM Conference on Information and*

Knowledge Management (Napa Valley, California, USA) *(CIKM '08)*. Association for Computing Machinery, New York, NY, USA, 469–478. https://doi.org/10.1145/1458082.1458145

[43] Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering Ambiguous Open-domain Questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 5783–5797. https://doi.org/10.18653/v1/2020.emnlp-main.466

[44] Dan Morris, Meredith Ringel Morris, and Gina Venolia. 2008. SearchBar: a search-centric web history for task resumption and information re-finding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 1207–1216. https://doi.org/10.1145/1357054.1357242

[45] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *arXiv preprint arXiv:1901.04085* (2019).

[46] 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*, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 708–718. https://doi.org/10.18653/v1/2020.findings-emnlp.63

[47] Paul Owoicho, Ivan Sekulic, 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* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 632–642. https://doi.org/10.1145/3539618.3591683

[48] Dragomir R. Radev, Hong Qi, Harris Wu, and Weiguo Fan. 2002. Evaluating Web-based Question Answering Systems. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Manuel González Rodríguez and Carmen Paz Suarez Araujo (Eds.). European Language Resources Association (ELRA), Las Palmas, Canary Islands - Spain. http://www.lrec-conf.org/proceedings/lrec2002/pdf/301.pdf

[49] Filip Radlinski and Nick Craswell. 2017. A Theoretical Framework for Conversational Search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval* (Oslo, Norway) *(CHIIR '17)*. Association for Computing Machinery, New York, NY, USA, 117–126. https://doi.org/10.1145/3020165.3020183

[50] Hossein A. Rahmani, Xi Wang, Yue Feng, Qiang Zhang, Emine Yilmaz, and Aldo Lipani. 2023. A Survey on Asking Clarification Questions Datasets in Conversational Systems. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 2698–2716. https://doi.org/10.18653/v1/2023.acl-long.152

[51] Sudha Rao and Hal Daumé III. 2018. 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)*, Iryna Gurevych and Yusuke Miyao (Eds.). Association for Computational Linguistics, Melbourne, Australia, 2737–2746. https://doi.org/10.18653/v1/P18-1255

[52] 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 International Conference on Theory of Information Retrieval* (Virtual Event, Canada) *(ICTIR '21)*. Association for Computing Machinery, New York, NY, USA, 167–175. https://doi.org/10.1145/3471158.3472257

[53] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'14)*. MIT Press, Cambridge, MA, USA, 3104–3112.

[54] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Huai hsin Chi, and Denny Zhou. 2022. Self-Consistency Improves Chain of Thought Reasoning in Language Models. *ArXiv* abs/2203.11171 (2022). https://api.semanticscholar.org/CorpusID:247595263

[55] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, and Tie-Yan Liu. 2013. A Theoretical Analysis of NDCG Type Ranking Measures. In *Annual Conference Computational Learning Theory*. https://api.semanticscholar.org/CorpusID:3804244

[56] Zhenduo Wang, Yuancheng Tu, Corby Rosset, Nick Craswell, Ming Wu, and Qingyao Ai. 2023. Zero-shot Clarifying Question Generation for Conversational Search. In *Proceedings of the ACM Web Conference 2023* (Austin, TX, USA) *(WWW '23)*. Association for Computing Machinery, New York, NY, USA, 3288–3298. https://doi.org/10.1145/3543507.3583420

[57] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 24824–24837. https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf

[58] Jingjing Xu, Yuechen Wang, Duyu Tang, Nan Duan, Pengcheng Yang, Qi Zeng, Ming Zhou, and Xu Sun. 2019. Asking Clarification Questions in Knowledge-Based Question Answering. 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)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 1618–1629. https://doi.org/10.18653/v1/D19-1172

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

[60] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SkeHuCVFDr

[61] Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. 2024. CLAMBER: A Benchmark of Identifying and Clarifying Ambiguous Information Needs in Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 10746–10766. https://doi.org/10.18653/v1/2024.acl-long.578

[62] Yujia Zhou, Jing Yao, Zhicheng Dou, Yiteng Tu, Ledell Wu, Tat-Seng Chua, and Ji-Rong Wen. 2024. ROGER: Ranking-oriented Generative Retrieval. *ACM Trans. Inf. Syst.* (June 2024). https://doi.org/10.1145/3603167 Just Accepted.

[63] Jie Zou, Mohammad Aliannejadi, Evangelos Kanoulas, Maria Soledad Pera, and Yiqun Liu. 2023. Users Meet Clarifying Questions: Toward a Better Understanding of User Interactions for Search Clarification. *ACM Trans. Inf. Syst.* 41, 1, Article 16 (jan 2023), 25 pages. https://doi.org/10.1145/3524110