AGENT-CQ: Automatic Generation and Evaluation of Clarifying Questions for Conversational Search with Large Language Models

CLEMENCIA SIRO, University of Amsterdam, Amsterdam, The Netherlands and Centrum Wiskunde en Informatica, Amsterdam, The Netherlands
YIFEI YUAN, ETH Zürich, Zürich, Switzerland and University of Copenhagen, Copenhagen, Denmark
MOHAMMAD ALIANNEJADI and MAARTEN DE RIJKE, University of Amsterdam, Amsterdam, The Netherlands

Clarifying questions enable **Conversational Search (CS)** systems to resolve underspecified queries by eliciting missing information from users. However, how prompting strategies shape the quality of clarifying questions and how such questions should be evaluated at scale remains understudied. We present **Automatic GENeration and evaluaTion of Clarifying Questions (AGENT-CQ)**, a framework for systematically generating and evaluating clarifying questions and simulated user responses using **Large Language Models (LLMs)**. To support scalable and multi-perspective evaluation, we introduce *CrowdLLM*, an LLM-based evaluation paradigm that simulates diverse annotator judgments through distinct evaluator personas. Our experiments span both open-domain CS and a regulatory question-answering setting, allowing us to examine the extent to which clarification strategies generalize across domains with different interaction constraints. Across settings, temperature-variation prompting leads to higher quality clarifying questions than baseline prompting and human-authored questions on several dimensions of the task. In addition, LLM-generated clarifying questions lead to improved downstream retrieval performance than human-authored questions in open-domain search. Together, AGENT-CQ and CrowdLLM provide a practical framework for studying and improving clarification strategies in conversational IR systems.

CCS Concepts: • **Information systems** → **Retrieval models and ranking**; **Language models**;

Additional Key Words and Phrases: Conversational search, clarifying questions, LLMs, evaluation

ACM Reference format:
Clemencia Siro, Yifei Yuan, Mohammad Aliannejadi, and Maarten de Rijke. 2026. AGENT-CQ: Automatic Generation and Evaluation of Clarifying Questions for Conversational Search with Large Language Models. *ACM Trans. Inf. Syst.* 44, 5, Article 110 (June 2026), 39 pages.
https://doi.org/10.1145/3809182

This research was (partially) supported by the Dutch Research Council (NWO), under project numbers 024.004.022, NWA.1389.20.183, and KICH3.LTP.20.006, and the European Union's Horizon Europe program under grant agreement No. 101070212 (FINDHR) and No. 101201510 (UNITE). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.
Authors' Contact Information: Clemencia Siro (corresponding author), University of Amsterdam, Amsterdam, The Netherlands and Centrum Wiskunde en Informatica, Amsterdam, The Netherlands; e-mail: c.n.siro@cwi.nl; Yifei Yuan, ETH Zürich, Zürich, Switzerland and University of Copenhagen, Copenhagen, Denmark; e-mail: yuanyif@ethz.ch; Mohammad Aliannejadi, University of Amsterdam, Amsterdam, The Netherlands; e-mail: m.aliannejadi@uva.nl; Maarten de Rijke, University of Amsterdam, Amsterdam, The Netherlands; e-mail: m.derijke@uva.nl.

ACM 1558-2868/2026/6-ART110
https://doi.org/10.1145/3809182

1 Introduction

Conversational search (CS) systems aim to support multi-turn interactions between users and retrieval systems, enabling users to express information needs more naturally and incrementally [5, 35, 65]. A recurring challenge in CS is handling under-specified or ambiguous **User Queries (UQ)**. Without sufficient context, retrieval systems risk returning results that are either irrelevant or incomplete. **Clarifying questions (CQ)** offer a natural mechanism for disambiguation: by prompting users for additional information, systems can better understand user intent and improve the quality of retrieved documents [64]. Despite their importance, generating and evaluating effective CQ at scale remains a significant challenge.

1.1 Research Gap

Prior work on CQ has primarily focused on either generation or evaluation, but not both in an integrated manner. Early methods for generating CQ in CS systems relied on manual curation by experts and template-based approaches [3, 63]: human experts craft CQ, relying on their ability to understand complex user intents and contextual nuances intuitively. This approach yields high-quality questions in controlled settings but lacks scalability and adaptability to the breadth and variability of user intents in open-domain conversational systems [13]. Additionally, human curators may lack in-depth knowledge of all conversation topics, which can limit their ability to craft effective CQ, especially in complex or domain-specific conversations. In contrast, template-based methods improve scalability and efficiency by using pre-defined question formats, yet they tend to lack flexibility. This results in generic or less diverse questions that could hurt the overall user interaction experience [60].

To address these limitations, another line of research focuses on generating CQ based on learning-based methods. These approaches include training models to generate CQ in a sequence-to-sequence manner [40, 58], selecting CQ from a candidate set with retrieval models [19], or augmenting neural generation with template-based constraints. While these methods offer greater flexibility than rule-based systems, they rely on large annotated datasets and are typically evaluated in isolation using human judgments or static metrics.

Recent years have seen a surge in the use of **large language models (LLMs)** to overcome the scalability limits of traditional template-based and supervised methods. For instance, Wang et al. [59] introduced zero-shot generation using query facets to overcome data scarcity, while Zhao et al. [67] proposed intent-aware frameworks that optimize for broad coverage of potential user needs. Furthermore, researchers have begun exploring specialized domains, such as Liu et al. [29] work on legal case retrieval and Ramezan et al. [38] extension into multimodal conversational settings.

However, despite these advancements, existing research typically treats generation and evaluation as isolated components. There remains a critical need for an integrated framework that can simulate the entire interaction loop to measure the downstream impact of clarifications on retrieval performance. This disconnect hinders the ability to systematically assess the impact of prompt design on retrieval effectiveness and prevents the consistent filtering of low-quality clarifications. We thus address this gap by asking:

> *Can large language models (LLMs) be used within an integrated framework to generate and evaluate clarifying questions that capture user intent and enhance retrieval in conversational search?*

1.2 Approach

To advance research in CQ generation and evaluation for CS, we introduce **Automatic GENeration and evaluaTion of Clarifying Questions (AGENT-CQ)**, a unified framework that integrates

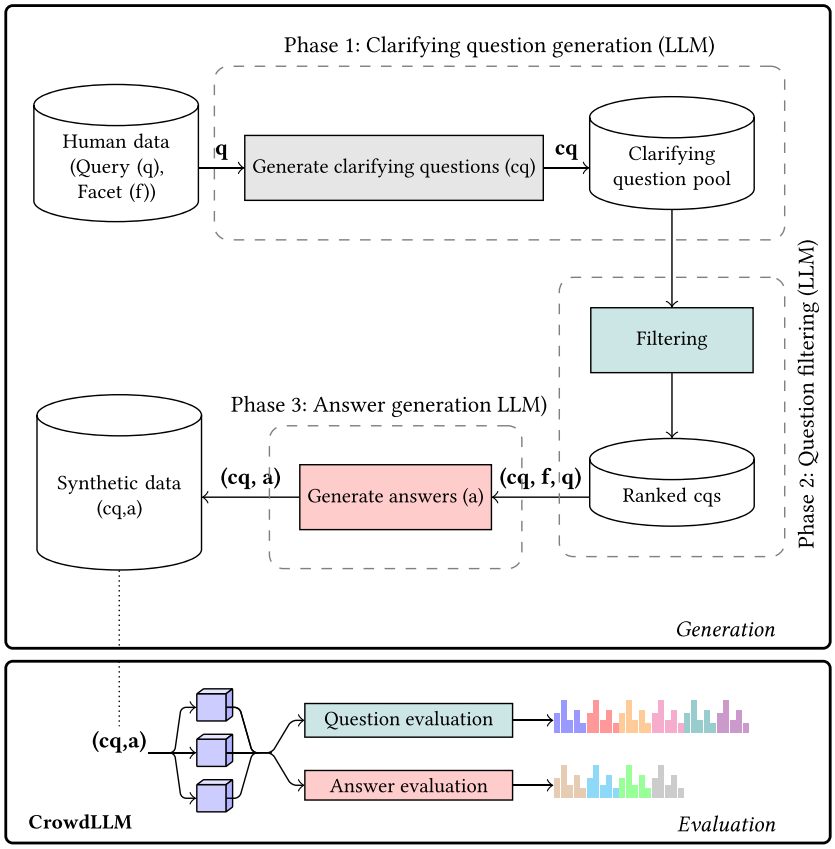

Fig. 1. Overview of the AGENT-CQ framework: the upper component focuses on the generation of clarifying questions and simulation of corresponding answers using an LLM, while the lower component, CrowdLLM, is responsible for the evaluation of the generated questions and answers.

LLM-based prompting, user simulation, and multi-dimensional evaluation. As illustrated in Figure 1, AGENT-CQ consists of two main components: a *generation component* (top) and an *evaluation component* (bottom). The generation component comprises three sequential phases: in Phase 1, we generate multiple candidate CQ using various prompting strategies such as temperature-controlled decoding and facet-based prompting; in Phase 2, we filter low-quality questions using LLM-based scoring for relevance and clarification potential; and in Phase 3, we simulate user responses by prompting an LLM to act as a cooperative search user. The evaluation component of the framework, *CrowdLLM*, is an LLM-based evaluator that approximates the diversity of human judgments by prompting multiple LLM models with varied decoding temperatures and prompt templates. CrowdLLM conducts a multi-dimensional evaluation of both questions and answers, enabling scalable and consistent assessment across quality metrics. AGENT-CQ addresses key limitations in prior work, where generation, simulation, and evaluation are often handled in isolation or constrained to small-scale setups. By integrating these components into a unified framework, AGENT-CQ supports systematic, scalable comparisons across prompting strategies, response simulation methods, and evaluation dimensions.

1.3 Research Questions (RQ)

Using AGENT-CQ, we conduct comprehensive experiments on the ClariQ dataset [4] to address the following RQs:

(*RQ1*) *Can LLM-based evaluation (i.e, CrowdLLM) provide consistent, multi-dimensional quality assessments that align with human judgments?*

To answer this, we compare CrowdLLM scores against crowdworker annotations using agreement metrics and preference judgments, and analyze which evaluation dimensions are most predictive of overall quality. CrowdLLM exhibits strong inter-rater agreement and high correlation with human ratings, validating its effectiveness as a substitute for manual annotation.

Next, we investigate

(*RQ2*) *How do different prompting strategies, such as temperature-controlled decoding and facet-based prompting, affect the quality of clarifying questions generated by LLMs?*

We perform structural and linguistic analyses of the generated questions (e.g., length, complexity, readability), categorize their types, and assess simulated user responses in terms of length, informativeness, and naturalness. Using CrowdLLM, we conduct multi-dimensional quality comparisons across all systems. We observe that temperature variation generates clearer and more useful questions, while GPT-Facet yields more specific but complex CQs.

Finally, we answer

(*RQ3*) *Do CQ generated using AGENT-CQ improve downstream retrieval performance in conversational search?*

We evaluate four input configurations: baseline prompting, AGENT-CQ-generated questions, **Human-generated questions (H-Gen)**, and answer sources from both human and simulated responses. Retrieval is performed using lexical, discriminative, and generative ranking models.

Our results show that CQs generated by temperature variation consistently yield higher retrieval effectiveness across both sparse and dense retrieval models, outperforming those generated by GPT-Baseline and H-Gen. These findings demonstrate AGENT-CQ's ability to generate clarification strategies that not only improve response quality but also enhance retrieval outcomes.

In addition, we examine the generalizability of AGENT-CQ beyond ClariQ. We evaluate the CQ generation component of AGENT-CQ on the ShARC dataset [45], which consists of regulatory and eligibility-driven conversational queries. By adapting ShARC to a single-turn setting and isolating generation behavior, we observe that prompting strategies learned in open-domain settings transfer to domains with different interaction structures and linguistic constraints.

1.4 Contributions

Our *main contributions* in this article are:

- (C1) We introduce *AGENT-CQ*, a unified framework for controlled CQ generation, simulation, and evaluation in CS, enabling systematic analysis across datasets and interaction settings.
- (C2) We propose *CrowdLLM*, an LLM-based multi-perspective evaluation component within AGENT-CQ, and show that its evaluation is aligned with human judgment.
- (C3) We identify systematic effects of prompt design on CQ quality, showing that temperature-based prompting consistently yields more effective clarifications, leading to substantial improvements in document retrieval performance.

2 Related Work

2.1 CQ in CS

With the rapid development of IR techniques, CS represents a shift from traditional single-turn IR interactions toward multi-turn conversations, where users iteratively refine or clarify their information needs based on system feedback [35, 65]. A primary challenge in such multi-turn scenarios is handling incomplete, vague, or ambiguous UQ, which can negatively impact retrieval effectiveness [5, 63]. CQs have been recognized as an effective tool for resolving query ambiguity and refining the user's intent, ultimately enabling the system to provide more relevant responses in subsequent retrieval turns [6, 62]. The use of CQs has been extensively studied in interactive IR settings, including community question–answering platforms [6], CS systems [41], and conversational question answering scenarios [58], and so on.

Early research emphasized the importance of proactive clarification to avoid conversational breakdowns, reduce cognitive load, and avoid costly reformulations by the user [12, 22, 35]. Subsequent studies demonstrated that effective CQ can significantly improve document ranking and user satisfaction [5, 19]. The introduction of the ClariQ benchmark [4] further solidified CQ as a core evaluation focus for CS systems.

Nevertheless, generating and evaluating CQs at scale remains a major research challenge due to the diversity of user intents and the subjectivity of conversational quality. This is where we contribute with this article.

2.2 Methods for Generating CQ

CQ generation has been addressed through manually curated question sets, template-based strategies, retrieval methods, and learning-based models. Early work relied on expert-authored CQ or hand-crafted templates [5, 63], which ensured quality but lacked scalability and diversity. Template-based approaches partially improve scalability [60], although they often produce repetitive or overly generic outputs. Retrieval-based systems, which select CQ from pre-existing corpora, offer fluent, domain-consistent language but are constrained by the diversity of available candidates [15, 19, 44].

Early learning-based approaches to CQ generation relied on supervised sequence-to-sequence models trained on annotated question–answer pairs [40, 58], which enabled flexible, context-aware generation but required large-scale labeled data and often generalized poorly to unseen intents or domains. More recent work has shifted toward reducing supervision by leveraging LLMs in zero- or few-shot settings via prompting [20, 26, 33, 59], as well as toward incorporating additional structure through zero-shot facet-driven generation [59] and intent-aware frameworks [67]. Furthermore, the scope of clarification has expanded into specialized modalities and high-stakes domains, such as multi-modal interaction [38] and context-aware legal case retrieval without external knowledge [29].

While these studies advance CQ generation along complementary dimensions, they typically consider generation, evaluation, or domain adaptation in isolation. AGENT-CQ adopts an integrated perspective by leveraging LLMs across generation, filtering, and user simulation to study the downstream impact of clarification strategies on conversational retrieval performance.

2.3 User Simulation in CS

User simulation has been an important component in the evaluation of conversational systems, particularly for task-oriented dialogue and CS [27, 46]. In the context of CQ, simulating plausible user answers is essential to assess whether generated clarifications lead to effective disambiguation of information needs. Early user simulators were rule-based or template-driven [48], offering

interpretability but limited diversity. Neural user simulators subsequently introduced more dynamic, data-driven response generation [25, 46], although they require substantial training resources and may not generalize well across domains.

Prompt-based user simulation with LLMs has recently been explored as a scalable alternative [42, 49, 51, 52], enabling diverse and context-sensitive responses without fine-tuning. However, these efforts primarily focus on system responses, task success, or high-level user behavior, with only limited simulation of user answers to CQ. In this work, we integrate LLM-based user simulation into a closed-loop generation pipeline, enabling systematic evaluation of clarification effectiveness through plausible user responses.

2.4 Evaluation of CQ

Evaluating CQ in CS requires more than assessing topical relevance. Effective clarification depends on multiple factors, including clarity, specificity, usefulness, and conversational appropriateness [4, 58]. Early evaluation efforts relied primarily on expert annotation or crowdsourced judgments [23, 50], which provide valuable insights but are costly, time-consuming, and difficult to scale. Automated proxies, such as retrieval effectiveness metrics or semantic similarity scores, have been explored [19, 21], but these approaches only capture limited aspects of clarification quality.

The use of LLMs as evaluators has recently gained attention across various natural language generation tasks, a trend referred to as "LLM-as-a-judge"[18, 28, 30, 68]. In machine translation [24], summarization [53], and dialogue evaluation [28], LLMs prompted with carefully designed instructions have demonstrated competitive or superior agreement with human annotators [8, 17]. However, existing LLM-based evaluation methods typically rely on a single model and a single prompt template, which may introduce systematic biases and limit judgment diversity. Furthermore, while multi-dimensional evaluation frameworks are emerging [28], they have not been systematically applied to CQ evaluation in CS contexts.

To address these limitations, we introduce *CrowdLLM*, a scalable evaluation framework that simulates a diverse set of annotators by employing multiple independently prompted LLM models. CrowdLLM aggregates judgments from varied prompts and sampling configurations to better approximate the diversity observed in human crowdsourcing [9]. It enables multi-dimensional evaluation of CQ across several conversational quality dimensions, supporting more robust and scalable benchmarking than traditional single-model approaches.

In contrast to prior research that addresses generation, simulation, and evaluation in isolation [4, 6, 19, 21, 23, 50, 63], the proposed *AGENT-CQ* framework offers a unified pipeline for CQ generation, user answer simulation, and multi-dimensional evaluation. While recent multi-task frameworks [44, 54, 58] explore related tasks such as query rewriting or suggestion generation, they do not focus on clarification in CS. AGENT-CQ integrates prompt-based generation strategies, scalable user simulation with LLMs, and a novel LLM-based evaluation framework (CrowdLLM), enabling systematic comparison of clarification strategies and supporting robust experimentation in CS.

3 AGENT-CQ: A Framework for Clarification Generation and Evaluation

In this section, we describe the architecture and components of AGENT-CQ, our end-to-end framework for generating, simulating, and evaluating CQs in CS. As illustrated in Figure 1, AGENT-CQ consists of two main components: (1) *generation*, comprising three phases: CQ generation, filtering, and user response simulation; and (2) *evaluation*, which assesses the quality of generated content using our LLM-based evaluator, named *CrowdLLM*. We first introduce the key variables and formal structure of the framework, followed by a detailed description of each phase.

3.1 Preliminaries

We define the core elements of AGENT-CQ in the context of conversational information retrieval. Each search interaction begins with a user-issued query q, which represents an initial, possibly ambiguous, information request. Underlying each query is a *facet* f, that captures the user's actual need. These facets are used for answer simulation and evaluation, but are not available to the system at generation time.

The goal of AGENT-CQ is to generate a set of CQ C that help reveal the user's intent, simulate realistic user responses a, and evaluate the effectiveness of both. In other words, CQ are generated directly from the initial query, while answers are generated conditioned on the query, the CQ, and the facet.

Consequently, AGENT-CQ consists of three sequential processes: (i) generation of CQ c from q, (ii) simulation of user answers based on (q, c, f), and (iii) evaluation of these outputs. Evaluation is performed independently for questions and answers, but follows a unified structure. We define a general evaluation function:

$$E(x, \text{context}) \rightarrow \mathbf{v}, \tag{1}$$

where x is the item to be evaluated (CQ or answer), context is the relevant conditioning input (e.g., q or (q, c)), and $\mathbf{v}$ is a multi-dimensional quality vector. All evaluations are conducted using CrowdLLM, which simulates a diverse set of human-like judgments by prompting multiple LLM models with varied configurations.

3.2 LLM-Based CQ Generation (Component 1)

AGENT-CQ's generation component generates and scores CQs using state-of-the-art LLMs in an end-to-end manner. The framework has three phases; see Figure 1 (top).

3.2.1 CQ Generation (Phase 1). Let $Q = \{q_1, q_2, \ldots, q_n\}$ be a set of n initial UQ. For each $q_i \in Q$, we aim to generate a set of CQs $C_i = \{c_{i1}, c_{i2}, \ldots, c_{im}\}$, where m is the number of CQs per set. We define a question generation function $G(\cdot)$:

$$C_i^1, C_i^2, \ldots, C_i^k = G(q_i) \tag{2}$$

that generates k sets of CQ for query q_i. In this study, we explore two prompt-based approaches: facet-based prompting and temperature-controlled prompting.

Facet-Based Approach. To generate CQ that reflect multiple plausible interpretations of an ambiguous query, we adopt a facet-based strategy inspired by prior work on query disambiguation [4, 47, 63]. The core idea is to decompose a query into distinct topical aspects, referred to as facets, which are then used to guide the CQ generation process. For example, given a query "teddy bear," the query can be decomposed to several facets such as "teddy bear color," "teddy bear size," and so on.

In this method, an LLM is first prompted to generate a set of facets for a given query, each capturing a specific possible intent or information need. For every query–facet pair, the model generates a CQ that aims to reduce ambiguity for that specific facet. Algorithm 1 details the implementation of this approach, where the function ϕ generates facets from the query q_i, and G_F uses the query and a selected facet f_{ij} to generate a corresponding clarifying question; Appendix A lists the prompts used.

Temperature-Controlled Decoding Approach. We adopt a decoding-based strategy that systematically varies the underlying LLM sampling temperature to promote diversity in generated CQs. Rather than explicitly modeling query facets, this approach relies on the stochastic nature of higher temperatures to implicitly explore multiple plausible interpretations of the user's query. As the temperature

Algorithm 1: Facet-Based CQ Generation

Require: Query q_i
Ensure: Set of clarifying questions C_i

$F_i \leftarrow \phi(q_i)$ ▷ Generate facets from the query
$C_i \leftarrow \{\}$ ▷ Initialize empty set of questions
for each facet $f_{ij} \in F_i$ **do**
  $c_{ij} \leftarrow G_F(q_i, f_{ij})$ ▷ Generate clarifying question from query and facet
  $C_i \leftarrow C_i \cup \{c_{ij}\}$
end for
return C_i

Algorithm 2: Temperature-Controlled CQ Generation

Require: Query q_i, number of temperature settings k
Ensure: Set of clarifying questions C_i

$C_i \leftarrow \{\}$ ▷ Initialize empty set of questions
for $j \leftarrow 1$ **to** k **do**
  $\tau \leftarrow \min(0.9, 0.5 + (j-1) \times 0.1)$ ▷ Select decoding temperature
  $c_{ij} \leftarrow G_T(q_i, \tau)$ ▷ Generates questions at temperature τ
  $C_i \leftarrow C_i \cup \{c_{ij}\}$
end for
return C_i

increases, the model is encouraged to produce more varied and exploratory outputs, potentially surfacing alternative clarification directions that a lower-temperature setting might not capture.

We generate a set of candidate questions for each query by prompting the LLM multiple times with the same input but incrementally increasing the decoding temperature. The method offers a mechanism for inducing diversity without requiring external knowledge or facet annotations. Algorithm 2 outlines the generation procedure, where $G_T(q_i, \tau)$ generates a set of CQs given query q_i and decoding temperature τ. The prompts used are listed in Appendix A.

3.2.2 Question Filtering (Phase 2). Following the initial generation, not all candidate outputs meet the criteria for high-quality CQ. Our preliminary analysis revealed that LLMs occasionally generate off-topic or non-clarifying outputs. To address this, we implement a scoring-based filtering method that selects the most effective CQs based on two key properties: *topical relevance* and *clarification potential*.

We define a scoring function $S(q_i, C_i^j)$ that assigns a weighted score to each candidate question C_i^j generated for query q_i:

$$S(q_i, C_i^j) = \alpha \cdot R(q_i, C_i^j) + (1 - \alpha) \cdot L(C_i^j), \tag{3}$$

where $R(q_i, C_i^j)$ denotes the relevance of the candidate question to the original query, and $L(C_i^j)$ quantifies its potential to clarify user intent. This strategy follows the weighted evaluation paradigm proposed by Rao and Daumé III [39]. Both components are scored on a scale of 1–10 by prompting an LLM. Relevance ensures that the question remains on-topic, while the clarification potential prediction score favors the questions likely to disambiguate the user's intent. We rank all candidate questions using this combined score and retain only the top-k (where $k = 10$ in our experiments) per query. Overall, this filtering step enhances the quality of questions forwarded to the user simulation and evaluation parts of AGENT-CQ.

Algorithm 3: Parameterized User Response Generation

Require: Query q_i, Facet f_i, Set of clarifying question C_i, User characteristics U
Ensure: Set of parameterized responses A_i

$A_i \leftarrow$
for each $c_{ij} \in C_i$ **do**
    $\leftarrow$ ConstructParameterizedPrompt(q_i, f_i, c_{ij}, U)
    $a_{ij} \leftarrow \psi(q_i, f_i, c_{ij}, U)$ ▷ Generate response
    $A_i \leftarrow A_i \cup a_{ij}$
end for
return A_i

3.2.3 User Response Simulation (Phase 3). In Phase 3 of AGENT-CQ's generation component, we simulate user responses to the ranked CQ from Phase 2. Recent work has demonstrated the efficacy of simulated users as cost-effective proxies for real users in conversational systems [1, 61, 66]. Therefore, in this phase, we use *LLM-as-a-simulator* to generate diverse answers to the system-generated CQ. We introduce a *parameterized-user simulation* approach to simulate user response. This method incorporates user characteristics (U) in the simulation to generate diverse and realistic answers (Algorithm 3). Our parameterized function is defined as:

$$a_{ij} = \psi(q_i, f_i, c_{ij}, U), \tag{4}$$

where q_i is the original query, f_i is the user information need, c_{ij} is the CQ, and U is the set of user characteristics. ψ extends the basic non-parameterized method by incorporating user characteristics U, primarily verbosity, which controls the response length detail, and revealment probability used to determine the likelihood of disclosing the true user information need.

As shown in Algorithm 3, the `ConstructParameterizedPrompt` function generates a structured prompt by incorporating the original query q_i, user information need f_i, and CQ c_{ij}, along with verbosity level and reveal probability randomly selected from the user characteristics set U defined in Section A.1 in Appendix A. This approach generates a wider range of responses, better reflecting real-world user behavior diversity. It also provides a richer dataset for training and evaluating CS systems, enabling a systematic study of the impact of user characteristics on system performance.

3.3 CrowdLLM: LLM-Based Automatic Clarification Evaluation (Component 2)

After generating CQs, we assess their quality across multiple dimensions, including clarity, usefulness, and specificity. Evaluating both the questions and their corresponding answers ensures a comprehensive understanding of their effectiveness. Traditional evaluation methods rely on human annotators, which can be time-consuming, expensive, and inconsistent. To overcome these limitations, we propose *CrowdLLM*, a scalable and robust automatic evaluation module that simulates human crowd judgments using multiple LLM models with varied configurations.

3.3.1 LLM-as-a-Crowd. Inspired by recent work on using LLMs as evaluators [8, 28, 68], CrowdLLM extends the "LLM-as-a-judge" paradigm by simulating a crowd of annotators rather than relying on a single model. For each question–answer pair, we prompt three independently initialized GPT-4 models, each with a different decoding temperature and slight prompt variation. This setup introduces both prompt- and sampling-level diversity, intended to reflect the range of judgment styles found in real-world crowdsourced evaluation. Specifically, we define three *simulated judge personas*: a *strict judge* (temperature 0.2), a *typical judge* (temperature 0.5), and a *lenient judge* (temperature 0.7), corresponding to common crowdworker profiles varying in strictness and

interpretive flexibility. This approach enables us to approximate inter-annotator variability without the need for human raters.

3.3.2 Multi-Dimensional Scoring. Evaluation in CrowdLLM is based on distinct sets of metrics for CQs and simulated answers, drawn from prior work on conversational information seeking and general conversational systems [4, 37, 39, 55, 56, 63]. For CQ, we assess seven dimensions: *clarity*, *usefulness*, *specificity*, *clarification*, *on-topicness*, *complexity*, and *overall quality*. For answers, we evaluate *relevance*, *usefulness*, *naturalness*, and *overall quality*. All scores for CQ are rated on a 1–10 scale, and the final score per dimension is computed by averaging across the three LLM judges. The simulated answers are evaluated using a pairwise comparison approach between the LLM-simulated and human responses.

Question Quality Metrics. We evaluate the quality of CQs along the following dimensions, each capturing a distinct aspect of question effectiveness:

(1) *Clarification*: Assesses whether the question seeks missing or uncertain information that is necessary to resolve ambiguity in the user's original query.
(2) *On-topic*: Measures the extent to which the question remains topically aligned with the subject matter of the original query. This dimension focuses on topical relevance and does not consider whether the question meaningfully advances clarification.
(3) *Specificity*: Evaluates how focused the question is on particular aspects of the user's query, as opposed to being overly broad or generic.
(4) *Usefulness*: Assesses the expected utility of the question in improving the system's ability to provide an appropriate response to the original query. A question may be on-topic and clear but still score low on usefulness if answering it is unlikely to substantially improve the final response.
(5) *Clarity*: Evaluates the linguistic clarity and interpretability of the question from the user's perspective, including whether it is unambiguous, well-formed, and easy to understand. This dimension focuses on formulation quality rather than intent or relevance.
(6) *Question complexity*: Examines whether the question introduces unnecessary technical terms, assumptions, or domain-specific knowledge that were not present in the original query, potentially increasing cognitive burden for the user.
(7) *Overall quality*: Provides an overall assessment of the CQ, considering all of the above dimensions jointly.

Answer Quality Metrics. Answers were also evaluated from a multidimensional perspective on the following three metrics and overall quality.

(1) *Relevance*: The extent to which the answer directly addresses the system's CQ.
(2) *Usefulness*: The degree to which the answer helps clarify the user's original information need and reduces ambiguity.
(3) *Naturalness*: How human-like, fluent, and conversational the answer appears.
(4) *Overall quality*: A holistic assessment of how effectively the answer supports system understanding, considering all dimensions.

To validate CrowdLLM's effectiveness, we compare its outputs against human annotations on a stratified sample of question–answer pairs.

3.4 Human Evaluation of CrowdLLM

To evaluate the reliability of CrowdLLM, we conducted a human study assessing the quality of both CQ and simulated user responses. Human judgments serve as a benchmark to validate whether CrowdLLM aligns with the preferences and perceptions of real users.

Participant Details. We recruited experienced crowdworkers from the United States via Amazon Mechanical Turk. All participants were designated Master Workers with an approval rate exceeding 95% and at least 10,000 completed HITs. Workers were compensated at a rate of $8.50/hour. A total of 30 workers (18 male, 12 female) contributed to the question ranking task, while 18 workers (7 male, 11 female) participated in the answer evaluation task. Each HIT was independently completed by three workers.

CQ Evaluation. Unlike CrowdLLM, which performs multi-dimensional scoring, human evaluation uses a preference-based ranking approach to reduce annotation time and cognitive load. In each HIT, workers were presented with a UQ and five CQ generated by different systems: Llama 3.1, GPT-Facet, GPT-Temp, GPT-Baseline, and H-Gen. Using a drag-and-drop interface, workers ranked the questions from most helpful (Rank 1) to least helpful (Rank 5). To mitigate position bias, the question order was uniformly randomized so that each system's question appeared in the top position in approximately 20% of the HITs. In total, 1,000 questions were assessed (200 per system), with each model ranked by three independent workers.

Simulated Answer Evaluation. To assess the quality of simulated responses, we adopted a pairwise comparison setup. For each HIT, workers were shown (i) the user's information need (facet), (ii) the initial query, (iii) a human-generated CQ, and (iv) two answers from two sources: one from a human and one generated by an LLM. Workers selected which answer was better for each dimension or marked them as equal if no difference was perceived. The answer order was randomized across HITs to prevent position bias. We used human CQ from the ClariQ dataset, which already include corresponding human answers, allowing direct comparison with LLM-generated responses. In total, 100 pairs (200 answers) were evaluated.

This human evaluation enables a direct comparison with CrowdLLM outputs and provides a reference point for validating the consistency and interpretability of our automatic judgments. Detailed results are presented and analyzed in Section 4.

3.5 Experimental Details

We describe the experimental setup used to evaluate AGENT-CQ, including the dataset, model configurations, and evaluation metrics.

3.5.1 Dataset and Sampling Strategy. We use the ClariQ dataset [4], built from TREC Web Track (2009–2012) queries [10]. Each topic consists of an initial query, user intent facets, human-written CQ, simulated user answers, and ground-truth relevance judgments for document retrieval.

In our framework, we retain the original queries but generate new CQ and simulated answers via LLM prompting. For GPT-Temp and GPT-Baseline, we use the original ClariQ topic–facet pairs to maintain compatibility with relevance annotations. For GPT-Facet and Llama, we prompt GPT-3.5 to generate new facets based solely on the query. These facets support open-ended CQ generation but are not linked to existing relevance labels.

3.5.2 Models and Implementation Details. We outline the specific models and parameter configurations used across the three core components of our framework: CQ generation, user simulation, and LLM-based evaluation.

CQ Generation. We use GPT-4 [69] for temperature-controlled generation (GPT-Temp), varying the decoding temperature $\tau \in \{0.5, 0.7, 0.9\}$ to promote diversity. For GPT-Facet and Llama, GPT-3.5 [70] first generates facets from the query, and CQ are then generated using either GPT-4 (for GPT-Facet) or Llama-3.1-8B [71]. The same model is used for filtering and scoring candidate questions based on relevance and clarification potential.

Table 1. CrowdLLM ICC and Weighted κ (W-κ) Agreement Scores for Different Prompting Strategies across Models, Including Human-Generated CQ (H-Gen)

Aspects	GPT-Baseline		GPT-Facet		GPT-Temp		Llama 3.1		H-Gen	
	ICC	W-κ	ICC	W-κ	ICC	W-κ	ICC	W-κ	ICC	W-κ
Clarification	0.96	0.87	0.95	0.85	0.85	0.72	0.97	0.89	0.97	0.89
Clarity	0.90	0.79	0.80	0.67	0.81	0.77	0.94	0.84	0.95	0.87
On-topic	0.93	0.81	0.87	0.78	0.87	0.82	0.93	0.86	0.96	0.88
Question-C	0.86	0.76	0.93	0.84	0.87	0.80	0.94	0.85	0.78	0.73
Specificity	0.92	0.80	0.83	0.66	0.80	0.66	0.92	0.79	0.96	0.87
Usefulness	0.94	0.84	0.93	0.82	0.92	0.78	0.97	0.89	0.97	0.90
Overall-quality	0.92	0.81	0.88	0.82	0.75	0.68	0.94	0.85	0.95	0.88

Question-C denotes question complexity.

User Simulation. Simulated answers are generated using GPT-3.5. Given a query and a CQ, the model is prompted to respond naturally as a search user. We use a temperature of 0.7 to ensure responses are coherent but diverse.

CrowdLLM Evaluation. For multi-dimensional quality evaluation, we use GPT-4o in the CrowdLLM framework. Each question–answer pair is evaluated by three independently prompted GPT-4o models, each with a distinct decoding temperature ($\tau = 0.3, 0.7, 0.9$), prompt, and user characteristics. Final scores are computed by averaging across models. Full prompt templates and evaluator personas are provided in Appendix A.

4 Evaluating the Reliability of CrowdLLM

In this section, we answer *RQ1* by evaluating whether LLM-based assessment via CrowdLLM provides consistent and human-aligned judgments of CQs and simulated answers. We focus on two aspects: the degree of agreement between CrowdLLM and human annotators, and the relative importance of individual evaluation dimensions such as clarity, specificity, and usefulness.

4.1 Agreement on CQ Evaluation

In Table 1, we report the inter-annotator agreement among the LLM models (i.e., GPT-4o) using **Intraclass Correlation Coefficient (ICC)** and weighted κ [11]. To evaluate the quality of the model's assessments, we additionally measure the level of agreement between CrowdLLM and **Human Evaluation (H-Eval)**. Because of the large number of generated questions by each model and the associated costs, H-Eval assess a sample of 200 questions from each model and rank them based on their preference. For each user information need and initial request, human evaluators are presented with CQs generated by each system. They rank these questions from most helpful (Rank 1) to least helpful (Rank 5), and justify their choice of the least helpful question. For CrowdLLM evaluations, we rank the questions based on their overall quality scores and conduct pairwise comparisons using Tukey's **Honestly Significant Difference (HSD)** *post hoc* test [2].

4.1.1 Inter-Rater Agreement among CrowdLLM Judges. CrowdLLM demonstrates consistent performance across most aspects of question quality. As shown in Table 1, we observe strong agreement across the three GPT-4 evaluator personas, particularly for clarification, usefulness, and on-topicness, where ICC values exceed 0.90 across most prompting strategies. Nonetheless, agreement varies by system and aspect. GPT-Temp exhibits the lowest consistency, with lower agreement for overall quality ($ICC = 0.75$, $W\text{-}\kappa = 0.68$) and specificity ($ICC = 0.80$, $W\text{-}\kappa = 0.66$), indicating that these outputs elicit more divergent assessments across simulated judges. In contrast,

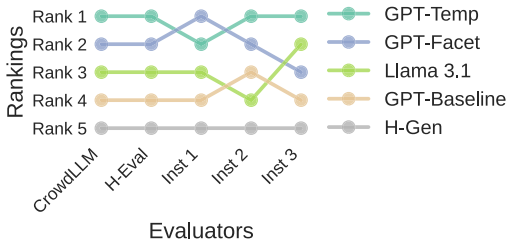

Fig. 2. Rankings of the question sets from different systems by different evaluators. Inst 1, Strict Judge; Inst 2, Typical Judge; Inst 3, Lenient Judge.

Llama 3.1 and H-Gen yield the most stable evaluations, with consistently high agreement across all dimensions. Despite these variations, agreement remains high for most aspects, reinforcing that CrowdLLM offers a reliable and flexible evaluation framework. These findings confirm that CrowdLLM effectively captures judgment variance and highlights which system outputs are more sensitive to evaluator bias.

4.1.2 Agreement between CrowdLLM and H-Eval. CrowdLLM evaluations strongly align with human evaluation results (H-Eval), confirming its effectiveness (Figure 2). Both approaches consistently rank the GPT-Temp question set as the most helpful and H-Gen as the least helpful, with H-Gen receiving an average rank of 4 out of 5 in human evaluation (lower is better). CrowdLLM models show ranking variations: Inst 1 (strict judge), prioritizes GPT-Facet > GPT-Temp > Llama 3.1; Inst 2 (typical judge), mirrors the aggregate rankings with GPT-Baseline > Llama 3.1; Inst 3 (lenient judge), ranks Llama 3.1 second, above GPT-Facet and GPT-Baseline. Despite the mid-rank differences, all three judges consistently assign top ranks to GPT-Temp or GPT-Facet and bottom ranks to H-Gen. These variations highlight the value of aggregating judgments from multiple simulated evaluators and demonstrate that CrowdLLM effectively captures nuanced perspectives while maintaining strong alignment with human assessments.

4.2 Agreement on Simulated Answer Evaluation

We also assess CrowdLLM's reliability in evaluating generated answers and comparing them with H-Eval. For each presented answer pair, evaluators judge which answer is better with respect to naturalness, relevance, usefulness, and overall quality, or indicate if the two are of equal quality.

4.2.1 Inter-Rater Agreement among CrowdLLM Judges. CrowdLLM shows high internal consistency. Table 2 reports inter-rater reliability among the three CrowdLLM judges, measured using Fleiss' κ for simulated answer evaluation. Agreement is highest for naturalness ($\kappa = 0.81$), indicating strong consistency in judgments of fluency and linguistic quality. Moderate agreement is observed for relevance ($\kappa = 0.68$), overall quality ($\kappa = 0.71$), and usefulness ($\kappa = 0.62$), suggesting that while the CrowdLLM judges apply similar evaluation criteria, there is some variation in how they interpret task-centric dimensions.

4.2.2 Agreement between CrowdLLM and H-Eval. Table 2 reports agreement among human annotators (% Agr.), and between CrowdLLM and human judgments (% H-Agr.) for simulated answers. Naturalness shows the highest internal agreement among both CrowdLLM ($\kappa = 0.81$) and human annotators (89%), but results in the lowest human–CrowdLLM agreement (53%). This indicates that while judgments within each group are consistent, their evaluation criteria for fluency may differ. For relevance, usefulness, and overall quality, both CrowdLLM and human annotators

Table 2. Inter-Rater Reliability Measures for CrowdLLM and H-Eval in Terms of Fleiss' κ, Annotator Agreement Percentage (% Agr.), and Human-CrowdLLM Agreement (H-Agr.) for Simulated Answers

	Fleiss' κ	**% Agr.**	**% H-Agr.**
Naturalness	0.81	89%	53%
Relevance	0.68	86%	73%
Usefulness	0.62	73%	68%
Overall-quality	0.71	79%	75%

Table 3. Kendall's τ and Spearman's ρ Correlations of CrowdLLM Question Evaluation Aspects with Overall Quality

Aspect	**Kendall's τ**	**Spearman's ρ**
Clarification	0.76	0.87
Clarity	0.75	0.85
On-topic	0.71	0.81
Question-C	0.07	0.08
Specificity	0.63	0.73
Usefulness	0.80	0.90

Question-C denotes question complexity.

exhibit moderate internal agreement ($\kappa = 0.62$–0.71; 73–86% Agr.), with higher alignment between groups (68–75% H-Agr.).

These results suggest that dimensions involving subjective interpretation, such as naturalness, may lead to consistent but divergent judgments across evaluator types. Meanwhile, task-centric aspects such as relevance and overall quality are assessed more similarly by both CrowdLLM and human annotators. CrowdLLM reliably reflects human preferences in task-centric dimensions while also identifying dimensions where human-model misalignment is likely.

4.3 Significance of Evaluation Aspects in CrowdLLM Judgments

We analyze which evaluation aspects are most predictive of overall quality for both CQ and clarifying answers in CrowdLLM. Table 3 presents Kendall's τ and Spearman's ρ correlations between question-level evaluation aspects and overall quality, as rated by CrowdLLM. Usefulness shows the strongest correlation with overall quality ($\tau = 0.80$, $\rho = 0.90$), followed by clarification ($\tau = 0.76$, $\rho = 0.87$) and clarity ($\tau = 0.75$, $\rho = 0.85$), indicating that these task-centric dimensions are central to the perception of question quality. Specificity and on-topicness show moderate correlations, while question complexity exhibits a very weak correlation ($\tau = 0.07$, $\rho = 0.08$), suggesting limited influence on overall assessments.

For answer evaluation, Figure 3 shows Spearman's ρ correlations between individual aspects and overall quality. Usefulness ($\rho = 0.76$) and relevance ($\rho = 0.72$) show the strongest correlation with overall quality, indicating their critical role in perceived answer quality. Naturalness shows moderate correlation ($\rho = 0.50$), suggesting less impact. Strong correlations between relevance and usefulness ($\rho = 0.70$) highlight their interconnectedness in high-quality answers. The relatively

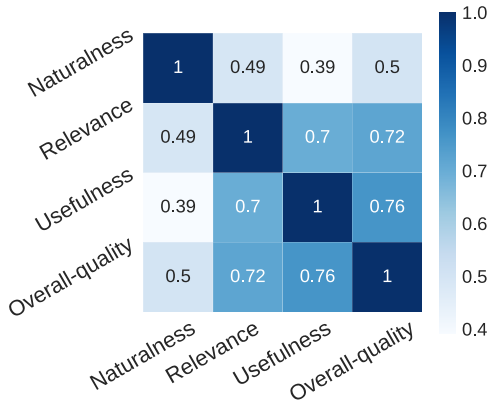

Fig. 3. Spearman's ρ correlations between answer evaluation aspects.

lower correlations of naturalness with relevance ($\rho = 0.49$) and usefulness ($\rho = 0.39$) indicate that it reflects a distinct aspect of answer quality.

Summary. To address *RQ1*, we examined whether CrowdLLM yields consistent and human-aligned evaluations of CQ and simulated answers. Results demonstrate high inter-rater agreement among the simulated judges across most evaluation dimensions, especially task-focused aspects such as clarification, usefulness, and relevance. These aspects also exhibit the strongest correlations with overall quality, reinforcing their significance in the evaluation of conversational responses. In addition, they align well with human judgments, demonstrating CrowdLLM's effectiveness in capturing quality signals that are critical for assessing system outputs in CS. While naturalness shows strong internal agreement, the lower alignment between CrowdLLM and human annotators suggests it captures a more subjective quality dimension, leading to variability in interpretation across evaluator types.

Overall, these findings validate CrowdLLM as a scalable and multi-perspective evaluation framework for CS. It effectively captures dimensions aligned with human judgments while also revealing areas of potential interpretive variance. Having established its reliability, we now use CrowdLLM to systematically evaluate CQ and simulated answers generated under different prompting strategies in AGENT-CQ.

5 Analysis of CQ and Simulated Answers in AGENT-CQ

To answer *RQ2*, we examine the quality of CQ and simulated user answers generated by different prompting strategies under AGENT-CQ, including GPT-Temp, GPT-Facet, GPT-Baseline, and H-Gen. This analysis aims to understand how prompting methods influence the structure, intent, and overall effectiveness of system responses in a clarification pipeline. We first analyze the linguistic patterns, intent categories, and expected response types of the generated CQ. We then use CrowdLLM to evaluate the quality of CQ and simulated answers across multiple dimensions.

5.1 Characterization of CQ

To analyze how different prompting strategies, influence the structure and intent of generated CQ, we perform a detailed characterization across different methods. This analysis focuses on three aspects: (i) recurring linguistic patterns [6], (ii) identifying question intent categories [6, 63], (iii) classifying the expected response types, and (iv) analyzing features such as question length and readability.

Table 4. Question Length Statistics and Readability Scores

Source	Mean length	Std. dev.	Flesch ease	FK grade
Llama3.1	19.02	9.26	53.77	9th
GPT-Baseline	10.72	2.15	61.10	7th
GPT-Facet	23.53	4.98	35.58	14th
GPT-Temp	15.68	3.33	52.74	9th
Human	9.71	2.48	75.99	5th

Length differences are statistically significant, $p < 0.05$.

We measure average question length and apply standard readability metrics, including the Flesch-Kincaid readability score, to compare the linguistic accessibility of outputs across models. To categorize question intent, we use a prompting-based LLM classification approach to distinguish between types such as disambiguation, information seeking, and topical refinement. We identify common structural patterns using a hierarchical phrase-matching system that extracts frequently occurring syntactic forms and key phrases. Expected response types are labeled using rule-based heuristics, grouping questions into Yes/No, Multiple choice, Open-ended, or Factual.

5.1.1 Question Length and Readability Analysis. Table 4 shows that human questions are concise (9.71 words) and simple (5th-grade level). LLM outputs vary: GPT-Facet generates complex, lengthy questions (college-level, 23.53 words), Llama 3.1 generates variable-length high school-level questions, and GPT-baseline closely matches human question length but with higher complexity. There is a consistent gap in LLMs' ability to replicate the brevity and simplicity of human-written questions. These readability differences suggest that longer and more complex LLM-generated questions may increase the effort required to interpret CQ, whereas the brevity and simplicity of human-written questions may lead to faster comprehension and response in conversational settings.

5.1.2 Question Categories. We develop a classification framework for CQs based on [6, 63]. Table 5 presents each question type with descriptions and examples, including UQ and corresponding CQs to illustrate their application in real conversations. This taxonomy provides a robust framework for comparing clarification strategies across various LLMs in conversational information seeking. Initial rule-based classification approaches did not effectively distinguish between different types of question intent. For instance, "Did you mean the book or the movie?" could be categorized as disambiguation or information gathering, depending on context. To address this, we employed GPT-3.5 for categorization, using its context awareness to select the most appropriate category. This approach enabled more accurate classification, particularly for questions that were ambiguous or could belong to more than one intent category.

Table 6 shows that all models except Llama 3.1 favor *preference identification* questions, with GPT-Facet leading at 74.00%. Llama 3.1 has a more balanced distribution between *preference identification* (47.20%) and *information gathering* (41.00%), and the highest disambiguation rate (10.20%). H-Gen have the highest confirmation rate (17.91%). Comparison questions are consistently low (<1.61%) across all approaches.

5.1.3 Question Patterns and Response Types. We developed a systematic approach to identify and classify question patterns using a hierarchical matching system. This process analyzes the linguistic structure and key phrases, starting with primary question words (e.g., "What," "How," "Are you") and then examining subsequent words for more specific patterns. For example, "What specific" and "What kind of" are categorized differently from general "What" questions. "How"

Table 5. CQ Categories and Examples

Category	Description	Example
Disambiguation [63]	Addresses queries that are ambiguous and could refer to different concepts or entities.	UQ: I'm looking for information on Java CQ: Are you referring to Java the programming language, Java the island, or Java coffee?
Preference identification [63]	Clarifies the user's specific preferences, including personal, spatial, temporal, or purpose-related information.	UQ: I want to buy a new laptop CQ: What will be the primary use of this laptop? Gaming, work, or general use?
Information gathering [6, 63]	Seeks additional details, verifications, or narrows down broad topics.	UQ: Tell me about artificial intelligence CQ: Which aspect of artificial intelligence are you most interested in learning about: machine learning, neural networks, or natural language processing?
Comparison [6, 63]	Involves comparing entities or options to aid decision-making.	UQ: I'm researching electric cars CQ: Would you like to compare the range, performance, or price of different electric car models?
Confirmation [6, 63]	Questions that seek to verify or confirm previously provided information or assumptions.	UQ: I need a new phone CQ: Are you specifically looking for a smartphone, or would you consider other types of mobile phones?
General [6]	Broad questions that prompt for additional details or elaboration on a topic.	UQ: I want to start a business CQ: Can you provide more details about your business idea and what stage of planning you're in?

UQ stands for user query, which represents an example of a typical user question. CQ stands for CQ, which shows how a system might respond to the UQ by asking for more specific or relevant information.

Table 6. Percentage of Question Categories for Different Models.

Model	Pref.	Info.	Disamb.	Conf.	Comp.
Llama 3.1	47.20	41.00	10.20	0.60	1.00
GPT-Baseline	73.64	15.29	2.41	7.04	1.61
GPT-Facet	74.00	18.00	6.20	0.60	1.20
GPT-Temp	66.80	14.40	16.40	1.60	0.80
Human	64.39	10.66	6.84	17.91	0.20

Conf., Confirmation; Comp., Comparison; Disamb., Disambiguation; Info., Information Seeking; Pref., Preference Identification.

questions are differentiated based on inquiries about methods, duration, or extent. We also consider compound structures like "Are you looking for" or "Do you need," which are common in CQ. The implementation uses a combination of regular expressions and string-matching algorithms, balancing flexibility in pattern recognition with consistency in categorization. This approach enables a nuanced analysis of how different prompting strategies formulate questions.

Table 7. Results Showing the Percentage Distribution of Question Patterns Generated by Various Models, Including Human Questions (H-Gen)

Pattern	Llama3.1 (%)	GPT-Base. (%)	GPT-Facet (%)	GPT-Temp (%)	H-Gen (%)
Other	29.00	15.90	1.40	11.60	29.44
Are you X	22.60	37.63	75.80	50.80	29.64
What specific	20.60	2.62	20.40	5.00	0.00
Do you need/want/have	6.80	20.32	0.00	22.80	17.74
Would you like	4.60	18.11	0.00	4.80	21.17
How X	2.00	0.60	1.60	0.00	0.20
Are you looking for X	1.40	0.00	0.00	0.20	0.00
Which specific	1.20	0.00	0.80	0.00	0.00
Is there	0.40	2.82	0.00	2.80	0.81

"GPT-Base." is short for "GPT-Baseline."

Table 8. Results Showing the Percentage Distribution of Response Types Elicited by the Generated Questions from Various Models, Including Human Questions (H-Gen)

Response type	Llama3.1 (%)	GPT-Base. (%)	GPT-Facet (%)	GPT-Temp (%)	H-Gen (%)
Multiple choice	41.40	22.33	73.20	73.00	10.46
Open-ended	37.20	6.84	8.80	3.40	4.02
Yes/No	16.00	70.22	17.80	22.60	80.68
Factual	5.40	0.60	0.20	1.00	4.83

"GPT-Base." is short for "GPT-Baseline."

Tables 7 and 8 reveal distinct question patterns and response types across human and LLM outputs. Humans show greater pattern diversity, favoring "Would you like" (21.17%) and "Do you need/want/have" (17.74%), with a strong preference for Yes/No responses (80.68%). In contrast, "Are you X" dominates GPT-Facet (75.80%) and GPT-Temp (50.80%), aligning with their preference for Multiple Choice responses ($\approx$ 73%). Llama 3.1 most closely mirrors human diversity. GPT-Baseline shows unique tendencies, preferring "Do you need/want/have" (20.32%) and Yes/No responses (70.22%). Some patterns (e.g., "What specific") are almost exclusively LLM-generated, highlighting significant differences in question formulation between humans and LLMs. To complement the quantitative analysis, Appendix C presents representative examples of CQ generated by different strategies for the same UQ, and their user responses.

Generally, *LLMs exhibit model-specific tendencies in generating CQ, often diverging from human patterns w.r.t. question structure, expected response type, category focus, length, and complexity, highlighting the challenges in replicating natural human question-asking behavior.*

5.2 CrowdLLM Evaluation of CQ

Next, we assess the quality of the CQ on seven aspects using CrowdLLM: *clarification*, *on-topic*, *specificity*, *usefulness*, *clarity*, *query complexity*, and *overall quality*. Figure 4 shows mean scores across all aspects per model. We use one-way ANOVA with *post hoc* Tukey's HSD for statistical analysis ($p < 0.05$).

To provide a detailed view of model performance, we analyzed mean ratings across individual evaluation aspects. Figure 4 presents a comparative analysis of mean ratings across individual evaluation aspects for CQ generated by different models and prompting strategies. GPT-Temp

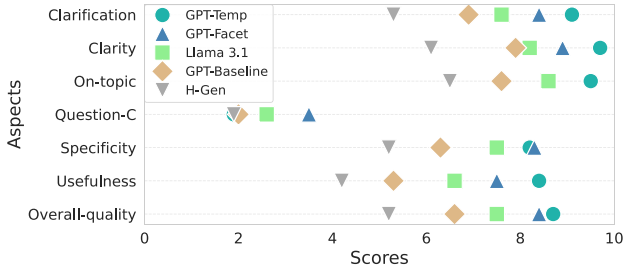

Fig. 4. Mean question quality scores evaluated by CrowdLLM across all aspects for different models. H-Gen, Human-Generated Questions.

consistently outperforms other approaches across most aspects, surpassing GPT-Baseline in usefulness (mean difference = 3.781, $p < 0.001$). Facet-based models (GPT-Facet and Llama 3.1) show improvements over the baseline, with GPT-Facet often ranking second. GPT-Facet excels in specificity, significantly outperforming GPT-Baseline, as it generates specific facets before producing targeted CQ. Thus, facet-based approaches enhance specificity but lead to more complex questions (GPT-Facet: 3.5) than GPT-Temp and H-Gen (both 1.9), aligning with our Fleisch readability and Kincaid analysis.

H-Gen score the lowest across most aspects, except for complexity. GPT-Temp significantly outperforms human questions in usefulness (8.4 vs. 4.2, $p < 0.001$), challenging assumptions about human expertise in question formulation. LLMs' superior performance can be attributed to their knowledge and consistent optimization for specific criteria. While recent work suggests that LLMs may favor their own outputs [31], our H-Eval independently rate human questions as least helpful, corroborating the rankings produced by CrowdLLM. This agreement between human and LLM evaluations strengthens our findings and suggests that the evaluation is not biased in favor of LLM-generated questions.

In summary, *LLM-generated CQ, particularly from GPT-Temp, outperform human-generated ones across most quality aspects. GPT-Temp's strong performance and low complexity make it ideal for general-purpose clarification tasks. Facet-based approaches enhance specificity but increase complexity; they best fit specialized domains requiring detailed clarifications.*

5.3 CrowdLLM Evaluation of Simulated Answers

We conduct pairwise comparisons of 200 answer pairs across four aspects: *relevance, usefulness, naturalness,* and *overall quality*. Each pair was evaluated by three human workers and three LLM models of CrowdLLM. We define a win for a model when at least two out of three (human or LLM) evaluators agree that the model's answer is superior; if the majority rates the answers as equal, the result is set as a tie.

LLM responses are longer (mean 13.21 vs. 8.19 words) and more variable (std dev. 8.06 vs. 4.36) than human-generated ones. Table 9 shows that LLM answers perform comparably to human answers in relevance and usefulness, demonstrating our approach's success in generating contextually appropriate and valuable responses. Naturalness assessments reveal a consistent internal preference within both evaluator groups, but divergence between them: human annotators slightly favor LLM answers (34% vs. 32%), while CrowdLLM shows a stronger preference (55.16% vs. 37.74%). This pattern reflects the distinct evaluative tendencies observed in our agreement analysis, where both humans and CrowdLLM were internally consistent but differed in how naturalness was judged. The results suggest that each group applies different, yet systematic, criteria when evaluating

Table 9. Percentage of Pairwise Comparisons Won by Each Model and Ties, as Evaluated by CrowdLLM and H-Eval

Aspects	CrowdLLM			H-Eval		
	Human	LLM-answer	Tie	Human	LLM-answer	Tie
Relevance	37.34	37.15	19.71	32.7	36.6	30.7
Usefulness	38.26	41.86	14.47	36.6	38.6	24.8
Naturalness	37.74	55.16*	6.82	32.0	34.0	34.0
Overall-quality	45.52	53.17*	1.82	39.9	41.8*	18.3

*Statistical significance (trinomial test, $p < 0.05$).

fluency. Overall quality marginally favors LLM answers with statistical significance, contrasting with previous work where H-Eval consistently preferred human answers [49].

Overall, *our answer simulation approach generates LLM-simulated answers that closely match or slightly outperform human answers across key aspects, contrasting with previous work and demonstrating successful capture of real user response diversity* [16].

Summary. To answer *RQ2*, we analyzed CQ and simulated answers generated under different prompting strategies in the AGENT-CQ framework. Our analysis reveals that prompting strategies significantly influence the quality, intent, and structure of clarification in CS. GPT-Temp consistently generates concise, high-clarity, and useful questions, while facet-grounded approaches like GPT-Facet improve specificity at the cost of increased complexity. CrowdLLM evaluation confirms that human questions, though linguistically simpler, are rated lower in usefulness and clarity. Distinct model tendencies emerge in question categories, patterns, and expected response types, highlighting systematic variation in generation behavior. Furthermore, our simulated answers, evaluated by both human and CrowdLLM judges, perform comparably or better than human answers on task-focused aspects, with divergence observed in subjective dimensions like naturalness reflecting distinct evaluative norms across annotator types.

Overall, these findings demonstrate AGENT-CQ's utility as a reproducible framework for analyzing how prompting strategies influence the structure, intent, and effectiveness of CQ in CS. In the next section, we examine how these prompting strategies and generated clarifications influence downstream retrieval performance.

6 Impact of the Generated CQ and Simulated Answers on Retrieval Performance

In this section, we focus on evaluating the end-to-end effectiveness of AGENT-CQ in improving document retrieval performance within a CS setting to answer *RQ3*. Building on the setup in [4], we simulate CS interactions where the user initiates a query, prompting the system to pose a CQ aimed at resolving ambiguity or underspecification. The user then responds with additional information, which the retrieval system incorporates to refine its understanding of the user's intent and retrieve a more relevant set of documents. This process captures the dynamic and iterative nature of CS.

We hypothesize that *CQ with higher quality, particularly those that reduce ambiguity, together with informative user responses, improve post-question answering retrieval performance.* These inputs provide critical context that allows the system to better align its output with the user's underlying information need. We assess this hypothesis by comparing retrieval performance across different prompting strategies and by contrasting AGENT-CQ's output with H-Gen. In addition, we examine the effect of using simulated answers versus human-written answers to determine the extent to which user simulation impacts retrieval effectiveness.

6.1 Retrieval Implementation

Our retrieval experiments focus on GPT-Temp and GPT-Baseline, as both models generate CQ using the original ClariQ topic–facet pairs. This ensures alignment with the ground-truth relevance labels, which are constructed based on these predefined facets. GPT-Facet and Llama 3.1, in contrast, rely on GPT-3.5-generated facets to guide their question generation and are therefore excluded from retrieval evaluation to maintain consistency with the annotated ground truth.

We follow the original ClariQ train/dev/test splits for the retrieval training data construction. We run all experiments using PyTorch on two NVIDIA A100 GPUs (80 GB). We use default optimizer settings and apply early stopping based on validation performance. Random seeds are fixed across all experiments to support reproducibility. All models are evaluated using the same preprocessing pipeline to ensure consistency across experiments. The evaluation metrics we use for retrieval include: (i) **Mean Reciprocal Rank (MRR)**: Average inverse rank of the first relevant document. (ii) *Precision@k* (P@1, P@5, P@10): Proportion of relevant results in the top-k. (iii) *nDCG@k*: Normalized Discounted Cumulative Gain at rank k, rewarding early placement of relevant documents.

We compare multiple input configurations to evaluate the effect of clarification quality and answer type on retrieval performance. These include: (i) *GPT-Temp+LLM*, where CQs are generated using temperature-controlled decoding and paired with LLM-generated answers. We also include (ii) *GPT-Baseline+LLM*, which uses a fixed GPT prompt (direct prompting) without decoding variation or facet modeling for generating the questions; (iii) *Human+Human*, which uses human-written CQs and corresponding human answers from the ClariQ test set; and (iv) *Human+LLM*, where the same human-written questions are paired with LLM-simulated answers.

In all cases, the retrieval input is constructed by concatenating the initial query, the CQ, and the answer (human or simulated). This setup enables a direct comparison between human-curated and automatically generated clarifications and the impact of responses on retrieval.

6.2 Retrieval Models

To assess the effect of CQs on retrieval, we compare four document ranking models covering lexical, discriminative, and generative approaches:

- *BM25* [43]: A lexical retrieval baseline using the initial query concatenated with the simulated answer. Implemented with Anserini.
- *BERT-Ranker* [14, 32]: We use a discriminative BERT-based reranker trained on labeled query–document pairs to re-score candidate documents retrieved in the first stage. The reranker models query–document relevance using contextualized representations and are trained with supervised relevance judgments. We adopt the CEDR implementation for reranking [32]. We follow the original setup and keep all hyperparameters unchanged, using VanillaBERT as the base model, a learning rate of 0.001, a maximum of 100 training epochs, and a batch size of 32.
- *T5 Retriever* [36]: A generative retrieval model that retrieves relevant documents by generating keywords or document identifiers conditioned on the input query. The model is trained to produce sequences of keywords corresponding to relevant documents. During inference, it generates document keywords using keyword-conditioned beam search over candidate documents. We follow the experimental setups and training procedures described in prior work [7, 34]. Specifically, we use a batch size of 32, a learning rate of 1e-4, and train the model for up to 30 epochs.

Table 10. Retrieval Performance with Different CQ Paired with LLM Answers under the Agent-CQ Framework

Method	Question src.	MRR	P			nDCG		
			@1	@5	@10	@1	@5	@10
BM25	GPT-Baseline	0.141	0.350	0.339	0.328	0.198	0.237	0.276
BM25	GPT-Temp	0.199*	0.377*	0.352*	0.334*	0.243*	0.265*	0.298*
BERT	GPT-Baseline	0.183	0.467	0.422	0.396	0.321	0.384	0.462
BERT	GPT-Temp	0.214*	0.470	0.435*	0.415*	0.381*	0.396*	0.498*
T5	GPT-Baseline	0.204	**0.618**	0.525	0.472	**0.480**	**0.600**	0.701
T5	GPT-Temp	**0.245***	0.569	**0.534***	**0.477***	0.407	0.581	0.701
Llama-Ranker	GPT-Baseline	0.194	0.521	0.489	0.452	0.408	0.421	0.693
Llama-Ranker	GPT-Temp	0.216*	0.543*	0.521*	0.468*	0.414	0.415	**0.704***

*Statistically significant improvements over the GPT-Baseline model in the same setup based on the paired randomization test ($p < 0.05$).
Number in bold shows the best score.

—*Llama-Ranker* [57]: A generative retriever initialized from Llama-3.1-8B, trained with the same objective as T5 for re-ranking. The model is trained with a batch size of 32, a learning rate of 1e-5, and for a maximum of 20 epochs.

6.3 Effect of Prompting Strategy

Table 10 presents the retrieval performance of several retrieval models augmented with different sets of CQ and their corresponding LLM-generated answers. Several key observations emerge from the results. In general, CQ generated by GPT-Temp lead to improvements in retrieval effectiveness. For example, under BM25, BERT, and Llama-Ranker, GPT-Temp consistently outperforms GPT-Baseline across various metrics under the same setup. Although GPT-Temp does not surpass GPT-Baseline under T5 for some metrics, possibly because T5's ranking model is less sensitive to the diversity or phrasing of CQ—the results still demonstrate its ability to elicit user responses that could enhance the quality of retrieved results. This result is consistent with our earlier findings, where both evaluators identified GPT-Temp questions as the most helpful. Its ability to generate precise and contextually relevant CQ appears to directly contribute to the observed improvements in retrieval effectiveness.

6.4 Impact of LLM-Simulated vs. Human Answers

Table 11 further examines the impact of different answer sources on retrieval performance. The results show that human-generated answers generally outperform LLM-generated answers. For example, across BM25, BERT, and Llama-Ranker, the results follow the same trend, with human answers outperforming LLM-generated answers on all metrics. Under T5, human answers also surpass LLM answers on six out of seven metrics. This effectiveness is likely due to two main factors: humans tend to use terms that overlap more with the original query, improving lexical matching, and the higher overall quality of human answers, which contributes to better ranking in term-based retrieval. However, when human questions are paired with LLM-generated answers, performance declines across both retrieval models. This result contrasts with our finding that LLM-simulated answers are often indistinguishable from human answers in quality assessments. Suggesting that while our parametric approach successfully mimics human-like responses, it may

Table 11. Performance Comparison of Different Retrieval Methods Using H-Gen

			P			nDCG		
Method	**Answer src.**	**MRR**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**
BM25	Human	0.172*	0.387*	0.365*	0.348*	0.229*	0.263*	0.296*
BM25	LLM	0.157	0.367	0.346	0.337	0.213	0.243	0.280
BERT	Human	**0.266***	0.186*	0.129*	0.110*	0.392*	0.387*	0.412*
BERT	LLM	0.230	0.149	0.109	0.099	0.350	0.354	0.385
T5	Human	0.191*	**0.507***	0.457	**0.416***	0.401*	**0.528***	**0.638***
T5	LLM	0.186	0.491	**0.458**	0.412	0.376	0.518	0.629
Llama-Ranker	Human	0.225*	0.488*	0.423	0.396*	**0.403***	0.524*	0.629*
Llama-Ranker	LLM	0.214	0.456	0.417	0.388	0.396	0.509	0.622

The answers are sourced from either humans or LLMs.
*Statistically significant improvements of human answers over LLM answers in the same setup based on the paired randomization test ($p < 0.05$).
Number in bold shows the best score.

not fully capture the nuanced interactions between CQ and clarifying answers crucial for retrieval tasks.

6.5 Comparison with Human-Generated CQ

We also compare the impact of human-generated versus LLM-generated CQ (i.e., GPT-Baseline and GPT-Temp) using Table 10 and the rows with LLM-generated answers in Table 11 for fair comparison. Interestingly, LLM-generated questions yield better retrieval performance than human-generated ones. For example, the Llama-Ranker achieves an NDCG@10 of 0.693 with GPT-Baseline questions and LLM answers, compared to 0.622 with human questions and the same answers. This suggests that LLMs can generate more retrieval-effective CQ, possibly due to their ability to generate more diverse and informative questions that are better aligned with user intent.

Summary. This section addressed *RQ3* by evaluating the impact of different prompting strategies and answer sources on retrieval performance in CS. Our findings show that high-quality CQ, particularly those generated by GPT-Temp, significantly improve downstream retrieval outcomes across lexical, discriminative, and generative models. These improvements reflect the ability of well-formed questions to elicit informative user responses that clarify intent and enhance ranking precision. Although LLM-simulated answers were previously rated as comparable to human answers in quality assessments, this similarity did not fully translate into retrieval gains. Human answers consistently yielded better performance, likely due to stronger lexical alignment and richer contextual signals. Notably, LLM-generated CQ surpassed human-written ones in retrieval performance, suggesting their potential for generating contextually rich, intent-aligned clarifications.

Overall, these findings reinforce the importance of prompt design and answer quality in optimizing retrieval and further validate AGENT-CQ as an effective framework for analyzing clarification strategies in end-to-end retrieval.

7 Generalizability of AGENT-CQ Beyond ClariQ

To examine whether AGENT-CQ generalizes beyond the ClariQ benchmark, we evaluate its CQ generation component on the ShARC dataset [45], which represents a qualitatively different conversational setting. ShARC focuses on regulatory and eligibility-based queries, in which user questions are often underspecified with respect to formal conditions encoded in natural-language rules. In

Table 12. Mean and Standard Deviation of CQ Length (in Words) and Readability Scores across Sources on ShARC

Source	Mean length	Std. dev.	Flesch Ease	FK Grade
Human	8.98	4.39	71.46	5.50
Temp (GPT-4o)	13.29	3.45	52.73	9.18
Temp (GPT-5)	21.18	4.69	46.83	11.44
GPT-Baseline	12.93	3.37	44.33	10.28
Llama-3.3	12.76	3.38	53.45	8.95
Qwen-3	27.32	21.39	65.78	6.49

this setting, clarification serves a different role than in ClariQ: instead of eliciting preferences or topical intent, CQ must identify missing logical conditions required to determine rule applicability.

Unlike ClariQ, which focuses on open-domain ambiguous queries, ShARC is a conversational question-answering dataset with underspecified UQ. This contrast allows us to test whether AGENT-CQ's generation transfers to domains with different linguistic structures and interaction demands.

Our analysis is restricted to the generation of CQ. We do not consider response generation or retrieval effectiveness, since responses in ShARC are binary, and in addition, the dataset has no retrieval corpus. To align ShARC with the ClariQ setting, we adapt it to a single-turn formulation: each model receives only the original user question as input. We omit dialog history, gold answers, evidence passages, and scenarios, and we exclude non-self-contained inputs (e.g., unresolved co-references) that are unsuitable for standalone clarification.

Models and Strategies. We evaluate prompt-based generation methods under identical input and output constraints. These include GPT-Temp and GPT-Baseline, as used in our ClariQ experiments, and additional LLMs (GPT-5.1 [69], Qwen-3 (32B) [72], and Llama-3.3 (70B) [71]) to test robustness across model families and scales. We exclude facet-based generation methods, as they rely on facet annotations available in ClariQ but not directly transferable to ShARC without additional supervision.

7.1 Characterization of ShARC's CQ

Similar to ClariQ, we conduct an analysis to gauge the quality of CQ generated by different models and strategies.

7.1.1 Question Length and Readability Analysis. Table 12 summarizes differences in the length and readability of CQ across sources on ShARC. Human questions are consistently short and simple, averaging 8.98 words and corresponding to approximately a fifth-grade reading level. In contrast, LLM-generated CQ are generally longer and more complex, with systematic variation across models. Questions generated by GPT-5.1 tend to be longer on average (21.18 words), a pattern supported by significant differences in length distributions across sources (Kruskal-Wallis, $H = 1,239.38$, $p \ll 0.001$) and large effect sizes in *post hoc* analysis (Cliff's $\delta > 0.8$) for several contrasts, including comparisons with human-authored questions. Other models generate questions closer in length to human-authored questions but remain more complex in terms of readability, while baseline models with similar generation strategies show negligible differences in length ($\delta \approx 0.04$). Qwen-3 exhibits higher variance in question length, indicating varying clarification strategies.

7.1.2 Question Intent and Response Types. To analyze the intent of CQ in ShARC, we categorize questions according to the type of missing information they seek to resolve. Although prior

Table 13. Distribution of CQ Intents on ShARC

System	Elig.	Context	Intent	Disamb.	Process	Entropy
Human	87.35	10.67	1.98	–	–	0.63
Temp (GPT-4o)	52.17	25.30	16.80	4.94	0.79	1.69
Temp (GPT-5)	26.48	46.05	19.96	7.31	0.20	1.78
GPT-Baseline	38.54	6.52	45.85	1.98	6.52	1.72
Llama-3.3	59.88	11.26	23.32	3.56	1.78	1.58
Qwen-3	66.80	14.82	16.80	0.99	0.59	1.34

Percentages are computed over 506 questions per system.

Table 14. Distribution of Expected Response Types for CQ on ShARC

System	Yes/No	MC	Factual	Open	Descr.
Human	99.21	0.79	–	–	–
Temp (GPT-4o)	50.00	12.45	27.27	9.49	0.79
Temp (GPT-5)	12.25	45.65	37.35	4.55	0.20
GPT-Baseline	83.60	5.73	2.37	7.31	0.99
Llama-3.3	60.28	12.25	11.07	15.22	1.19
Qwen-3	70.36	13.83	10.28	5.14	0.40

Percentages are computed over 506 questions per system. Readability is reported using Flesch reading ease and Flesch-Kincaid grade level.

taxonomies of CQ were developed primarily for open-domain CS [6, 63], ShARC exhibits a more constrained interaction structure. In this setting, ambiguity typically arises from underspecified eligibility conditions, missing contextual variables, or unclear procedural scope. As a result, several categories are rare or absent and not considered in ShARC. In ClariQ, on the other hand, these categories are highly used (e.g., comparison and preference identification).

We define a compact set of six intent categories tailored to ShARC-style interactions. *Eligibility Check* questions verify whether a user satisfies specific rule conditions (e.g., age, employment status, residency). *Context Clarification* questions request missing situational information required to interpret the query. *Intent/Scope Clarification* questions refine what the user is asking about, such as narrowing the applicable benefit or policy variant. *Disambiguation* questions resolve lexical or conceptual ambiguity, while *Process Inquiry* questions focus on procedural aspects such as application steps or documentation requirements. *General Clarification* captures broad follow-up questions that do not clearly fall into the previous categories. We assign categories using a cue-based, rule-driven approach grounded in question structure (e.g., yes/no eligibility formulations and procedural cues), which is well-suited to ShARC's explicit, condition-focused clarifications.

Table 13 shows that human-authored CQ are dominated by *Eligibility Checks* (87%), reflecting the checklist-style logic of regulatory dialogs. In Table 14, nearly all human questions expect Yes/No responses (99%), resulting in very low entropy across both intent and response type distributions.

In contrast, GPT-Temp systems exhibit substantially more diverse clarification behavior. For GPT-5, only 26% of questions clarify direct eligibility checks, while a large fraction focuses on *Context Clarification* (46%) and *Intent/Scope Clarification* (20%). This shift is reflected in the expected response types, with multiple-choice (46%) and factual (37%) responses dominating, and Yes/No questions accounting for only 12%. GPT-4o shows a similar but less pronounced pattern, combining eligibility checks (52%), context clarification (25%), and intent refinement (17%).

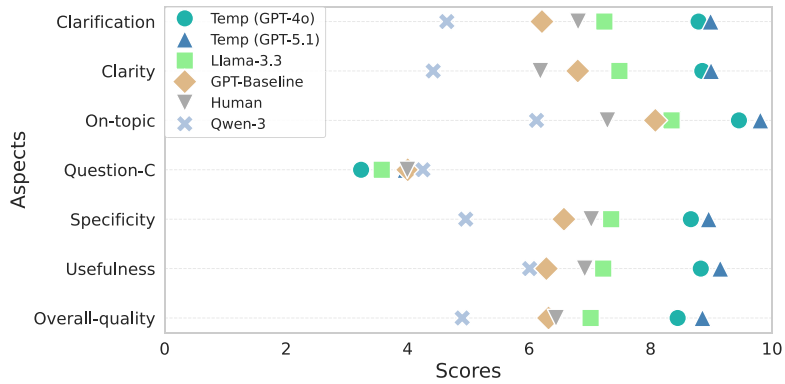

Fig. 5. Mean CrowdLLM ratings for CQ on the ShARC dataset, reported per model across seven evaluation aspects: clarification, on-topic, specificity, usefulness, clarity, query complexity (lower is better), and overall quality.

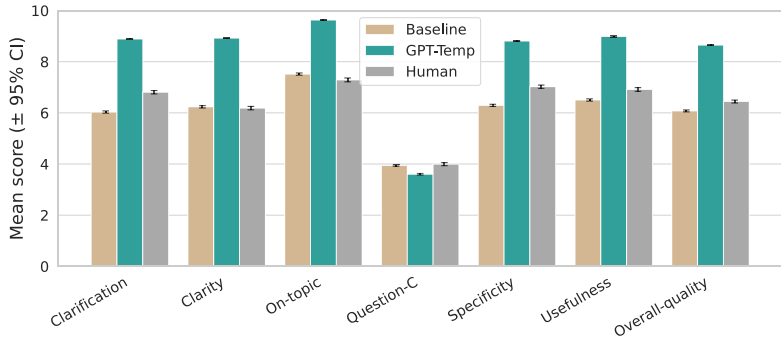

Fig. 6. Mean CrowdLLM ratings for CQ on the ShARC dataset, aggregated by generation strategy (Human, Baseline, GPT-Temp) across all evaluation aspects. Error bars indicate 95% confidence intervals.

Baseline models more closely resemble the human distribution. GPT-Baseline questions are split primarily between eligibility checks (39%) and intent or scope clarification (46%), while Llama-3.3 and Qwen-3 retain eligibility checks as the dominant category (60–67%) and primarily elicit Yes/No responses. Across all LLMs, entropy values are consistently higher than for human-authored questions, indicating broader clarification strategies and more varied expected response formats.

7.2 CrowdLLM Evaluation of CQ (ShARC)

We evaluate the quality of CQ on the ShARC dataset using CrowdLLM across seven aspects: *clarification*, *on-topic*, *specificity*, *usefulness*, *clarity*, *query complexity* (lower is better), and *overall quality*. Figure 5 reports mean scores per model, while Figure 6 aggregates results by generation strategy. One-way ANOVA shows statistically significant differences across sources for all aspects ($p \ll 0.001$). Effect sizes are large for clarification, clarity, and overall quality ($\eta^2 = 0.42$–0.47), and smaller but non-negligible for query complexity ($\eta^2 = 0.05$).

At the model level (Figure 5), *GPT-Temp consistently achieves the highest scores across all quality-oriented aspects*. Temp (GPT-5) obtains the highest mean ratings for clarification (8.99), on-topic relevance (9.81), specificity (8.96), usefulness (9.15), clarity (9.00), and overall quality (8.85). Temp (GPT-4o) follows closely, with similarly strong performance (e.g., clarification 8.79, usefulness 8.83,

overall quality 8.45). Both temperature-controlled models maintain relatively low query complexity (3.23 for GPT-4o; 3.97 for GPT-5), indicating that higher-quality clarification is not achieved by substantially increasing question complexity.

GPT-Baseline and Llama-3.3 achieve reasonable scores for on-topic relevance (8.08 and 8.34) and specificity (6.57 and 7.35), but lag behind GPT-Temp on clarity (6.80 and 7.49) and overall quality (6.32 and 7.01). Qwen-3 performs worst across all aspects, with notably low scores for clarification (4.64), clarity (4.42), and overall quality (4.90), and the highest variance among models. Human-authored CQ show lower query complexity (4.00) but also lower scores for clarification (6.81) and usefulness (6.92) compared to GPT-Temp.

Beyond individual systems, we examine whether clarification strategies differ at a higher level by grouping models into Baseline, GPT-Temp, and Human categories. Strategy-level analysis using one-way ANOVA reveals significant differences across all evaluation aspects ($p \ll 0.001$), with large effect sizes for clarity, usefulness, and overall quality. Temperature variation models consistently outperform baseline strategies on most aspects, while human-authored questions remain distinct in terms of brevity. These results suggest that differences observed at the model level reflect broader strategy choices rather than isolated model behaviors.

8 Discussion

8.1 Generalization of CQ Strategies

Our analysis across ClariQ and ShARC provides insight into which aspects of CQ generation generalize across CS settings, and which remain domain-dependent. Although the two datasets differ substantially in interaction structure, with ClariQ focusing on open-domain preference and intent ambiguity and ShARC centering on eligibility- and rule-based reasoning, we observe consistent differences between generation strategies that persist across both settings.

First, results from ShARC confirm that differences observed between generation strategies in ClariQ are not limited to open-domain search. In both datasets, temperature-controlled generation (GPT-Temp) consistently produces CQ that receive higher quality ratings than baseline prompting and human-authored questions. This holds across multiple evaluation aspects, including clarification effectiveness, usefulness, clarity, and overall quality. Importantly, these differences remain statistically significant under a substantially different task formulation, suggesting that the advantages of GPT-Temp reflect general properties of the generation strategy rather than dataset-specific artifacts.

At the same time, the role that clarification plays differs substantially across datasets. In ClariQ, CQ often elicit preferences or refine topical intent. In ShARC, clarification primarily targets missing rule conditions required to determine eligibility. This shift is reflected in both intent distributions and expected response types. Human-authored CQ in ShARC focus mostly on binary eligibility checks, resulting in highly concentrated intent and response distributions. In contrast, GPT-Temp systems distribute clarification more broadly across eligibility, contextual refinement, and scope clarification, and frequently elicit multiple-choice or factual responses rather than simple Yes/No answers. These differences indicate that generalization does not occur at the level of specific intent categories or surface question forms, but rather in how models balance different clarification objectives given the constraints of the task.

The CrowdLLM evaluation further suggests that higher clarification quality is not solely attributable to increased question complexity. While GPT-Temp models generate longer questions than humans on average, their query complexity scores remain comparable or lower, indicating that improved quality does not arise from unnecessarily complex formulations. Instead, GPT-Temp appears to more consistently align CQ with the type of information required to resolve ambiguity

in the current context, whether that ambiguity concerns preferences in ClariQ or rule applicability in ShARC.

We observe systematic differences in how CQ strategies manifest across datasets. On ClariQ, baseline prompting often yields generic or underspecified questions that insufficiently narrow user intent. On ShARC, the same strategies tend to over-rely on binary eligibility checks, producing predominantly Yes/No questions even when multiple conditions remain unresolved. GPT-Temp exhibits a different pattern. While the dominant intent categories shift between datasets, the model adapts its response structure to the task. For example, multiple-choice questions support preference narrowing in ClariQ, but enumerate alternative rule conditions or scope constraints in ShARC. This behavior suggests that GPT-Temp generalizes by adjusting clarification structure rather than by transferring fixed question types across domains.

Taken together, these findings suggest that AGENT-CQ generalizes at the level of clarification strategy rather than through direct transfer of dataset-specific question types. Strategies that encourage diversified clarification—balancing eligibility verification with contextual and scope refinement—remain effective across domains, even as the dominant sources of ambiguity change. At the same time, the substantial differences in intent and response structure across datasets highlight that CQ generation remains sensitive to domain-specific interaction constraints, underscoring the importance of evaluating clarification methods beyond a single benchmark.

8.2 Implications and Limitations

The experiments in this work are conducted in controlled benchmark settings and are not intended to demonstrate end-to-end deployment in real-world conversational systems. Instead, our findings provide insights into the design of clarification components in conversational IR pipelines, particularly in scenarios where user intent is underspecified.

Across both ClariQ and ShARC, we observe that generating multiple candidate CQ and explicitly selecting among them based on clarification-oriented criteria leads to higher-quality clarification than single-shot generation. In ClariQ, this improvement is reflected in downstream retrieval gains, while in ShARC it manifests as higher ratings for clarification effectiveness, usefulness, and clarity. Taken together, these results suggest that clarification benefits from being treated as a decision problem, where the choice of which question to ask plays a central role.

From a system design perspective, these findings support modeling clarification as a modular component that precedes retrieval or response generation. Selecting questions that target decision-critical missing information, rather than merely remaining on topic, can reduce ineffective clarification turns and improve interaction efficiency. Importantly, the ShARC analysis indicates that effective clarification does not require fixed intent templates. Instead, strategies that balance eligibility verification with contextual and scope refinement adapt more robustly across domains with different interaction structures.

Limitations. Our ShARC analysis evaluates CQ in isolation and does not measure downstream answer accuracy or dialog success, which limits direct comparison with the end-to-end retrieval experiments conducted on ClariQ. In addition, both benchmarks rely on curated or expert-authored queries rather than live user interactions. Future work should examine how clarification strategies influence user behavior, task completion, and satisfaction in interactive, multi-turn conversational settings.

9 Conclusion

In this study, we have introduced AGENT-CQ, a framework for systematically analyzing how prompting strategies shape the generation of CQ and simulated answers in CS. We have proposed

CrowdLLM, a multi-perspective evaluation paradigm that simulates diverse annotator personas using LLMs, enabling scalable and reliable assessment of answer and question quality.

Our experiments show that prompting strategies strongly influence the structure, intent, and effectiveness of CQ. Across settings, usefulness, relevance, and clarity are the most informative aspects for predicting overall clarification quality, providing practical guidance for system design and evaluation. In retrieval-based experiments on ClariQ, LLM-generated CQ consistently improve downstream retrieval effectiveness relative to human-authored questions. Analysis across ClariQ and ShARC further suggests that these findings are not limited to a single benchmark. While the nature of ambiguity differs substantially between open-domain and rule-based settings, temperature-controlled generation strategies consistently yield higher-quality clarifications. At the same time, clarification intent and response structure adapt to domain-specific interaction constraints, indicating that generalization occurs at the level of clarification strategy rather than surface question forms. Finally, CrowdLLM aligns closely with human judgments on task-centric aspects while exposing systematic disagreement on more subjective dimensions such as naturalness, making it useful both for scalable evaluation and diagnostic analysis.

Overall, AGENT-CQ provides a modular framework for studying and improving clarification strategies in CS. While our experiments focus on controlled benchmarks, the results offer guidance for designing clarification components that operate prior to retrieval or response generation. Future work includes optimizing clarification objectives for retrieval effectiveness, incorporating explicit user feedback, and extending the framework to multimodal and hybrid human–LLM evaluation settings.

Resources

We release the full set of generated CQ, evaluation prompts, and simulation data used in our experiments. This includes:

- —CQ generated using multiple prompting strategies (facet-first, temperature-based, and baseline) across different LLMs.
- —Prompt templates for facet generation, question generation, user simulation, and metric-based evaluation (CrowdLLM).
- —Evaluation scores and user simulation responses aligned with the ClariQ dataset.

All resources are available at: https://github.com/Clemenciah/AGENT-CQ-Data.

References

[1] Zahra Abbasiantaeb, Yifei Yuan, Evangelos Kanoulas, and Mohammad Aliannejadi. 2024. Let the LLMs talk: Simulating human-to-human conversational QA via zero-shot LLM-to-LLM interactions. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM '24)*. Luz Angelica Caudillo-Mata, Silvio Lattanzi, Andrés Muñoz Medina, Leman Akoglu, Aristides Gionis, and Sergei Vassilvitskii (Eds.), ACM, New York, NY, 8–17. DOI: https://doi.org/10.1145/3616855.3635856

[2] Hervé Abdi and Lynne J. Williams. 2010. *Tukey's Honestly Significant Difference (HSD) Test*. Technical Report. University of Texas at Dallas.

[3] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2020. ConvAI3: Generating clarifying questions for open-domain dialogue systems (ClariQ). arXiv:2009.11352. Retrieved from https://arxiv.org/abs/2009.11352

[4] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail 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 '21)*. 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 *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19)*. Benjamin Piwowarski, Max Chevalier, Éric Gaussier,

Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.), ACM, New York, NY, 475–484. DOI: https://doi.org/10.1145/3331184.3331265

[6] 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 (CHIIR '17)*. Ragnar Nordlie, Nils Pharo, Luanne Freund, Birger Larsen, and Dan Russel (Eds.), ACM, New York, NY, 345–348. DOI: https://doi.org/10.1145/3020165.3022149

[7] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Yiqun Liu, Yixin Fan, and Xueqi Cheng. 2022. CorpusBrain: Pre-train a generative retrieval model for knowledge-intensive language tasks. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*. Mohammad Al Hasan and Li Xiong (Eds.), ACM, New York, NY, 191–200. DOI: https://doi.org/10.1145/3511808.3557271

[8] Yi Chen, Rui Wang, Haiyun Jiang, Shuming Shi, and Ruifeng Xu. 2023. Exploring the use of large language models for reference-free text quality evaluation: An empirical study. arXiv:2304.00723. Retrieved from https://arxiv.org/abs/2304.00723

[9] Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. 2015. *Click Models for Web Search*. Synthesis Lectures on Information Concepts, Retrieval, and Services, Vol. 7, Morgan & Claypool Publishers.

[10] Charles L. A. Clarke, Nick Craswell, and Ian Soboroff. 2009. Overview of the TREC 2009 web track. In *Proceedings of the 18th Text REtrieval Conference (TREC '09)*. Ellen M. Voorhees and Lori P. Buckland (Eds.), National Institute of Standards and Technology (NIST). Retrieved from http://trec.nist.gov/pubs/trec18/papers/WEB09.OVERVIEW.pdf

[11] Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin* 70, 4 (1968), 213–220.

[12] 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 *Proceedings of the Findings of the Association for Computational Linguistics (EMNLP '23)*. Houda Bouamor, Juan Pino, and Kalika Bali (Eds.), Association for Computational Linguistics, 10602–10621. DOI: https://doi.org/10.18653/v1/2023.findings-emnlp.711

[13] Jan Deriu, Álvaro 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. DOI: https://doi.org/10.1007/S10462-020-09866-X

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.

[15] Kaustubh D. Dhole. 2020. Resolving intent ambiguities by retrieving discriminative clarifying questions. arXiv:200807559. Retrieved from https://arxiv.org/abs/2008.07559

[16] Yao Dou, Maxwell Forbes, Rik Koncel-Kedziorski, Noah A. Smith, and Yejin Choi. 2022. Is GPT-3 text indistinguishable from human text? Scarecrow: A framework for scrutinizing machine text. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL '22)*. Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.), Association for Computational Linguistics, 7250–7274. DOI: https://doi.org/10.18653/V1/2022.ACL-LONG.501

[17] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences* 120, 30 (2023), e2305016120.

[18] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. 2026. A survey on LLM-as-a-judge. *The Innovation* 7, 6 (2026), 101253. DOI: https://doi.org/10.1016/j.xinn.2025.101253

[19] Helia Hashemi, Hamed Zamani, and W. Bruce Croft. 2020. Guided transformer: Leveraging multiple external sources for representation learning in conversational search. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. ACM, New York, NY, 1131–1140. DOI: https://doi.org/10.1145/3397271.3401061

[20] Vitor Jeronymo, Luiz Henrique Bonifacio, Hugo Queiroz Abonizio, Marzieh Fadaee, Roberto de Alencar Lotufo, Jakub Zavrel, and Rodrigo Frassetto Nogueira. 2023. InPars-v2: Large language models as efficient dataset generators for information retrieval. arXiv:2301.01820. Retrieved from https://arxiv.org/abs/2301.01820

[21] Kimiya Keyvan and Jimmy Xiangji Huang. 2022. How to approach ambiguous queries in conversational search: A survey of techniques, approaches, tools, and challenges. *ACM Computing Surveys* 55, 6 (2022), 1–40.

[22] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. 2018. Toward voice query clarification. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18)*. Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz (Eds.), ACM, New York, NY, 1257–1260.

[23] Julia Kiseleva, Kyle Williams, Jiepu Jiang, Ahmed Hassan Awadallah, Aidan C. Crook, Imed Zitouni, and Tasos Anastasakos. 2016. Understanding user satisfaction with intelligent assistants. In *Proceedings of the 2016 ACM*

Conference on Human Information Interaction and Retrieval (CHIIR '16). Diane Kelly, Robert Capra, Nicholas J. Belkin, Jaime Teevan, and Pertti Vakkari (Eds.), ACM, New York, NY, 121–130. DOI: https://doi.org/10.1145/2854946.2854961

[24] Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*. Mary Nurminen, Judith Brenner, Maarit Koponen, Sirkku Latomaa, Mikhail Mikhailov, Frederike Schierl, Tharindu Ranasinghe, Eva Vanmassenhove, Sergi Alvarez Vidal, Nora Aranberri, et al. (Eds.), European Association for Machine Translation, 193–203. Retrieved from https://aclanthology.org/2023.eamt-1.19/

[25] Florian Kreyssig, Iñigo Casanueva, Paweł Budzianowski, and Milica Gašić. 2018. Neural user simulation for corpus-based policy optimisation of spoken dialogue systems. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*. Kazunori Komatani, Diane Litman, Kai Yu, Alex Papangelis, Lawrence Cavedon, and Mikio Nakano (Eds.), Association for Computational Linguistics, 60–69. DOI: https://doi.org/10.18653/v1/W18-5007

[26] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2022. Clam: Selective clarification for ambiguous questions with generative language models. arXiv:2212.07769. Retrieved from https://arxiv.org/abs/2212.07769

[27] Xiujun Li, Zachary C. Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. 2016. A user simulator for task-completion dialogues. arXiv:1612.05688. Retrieved from https://arxiv.org/abs/1612.05688

[28] Yen-Ting Lin and Yun-Nung Chen. 2023. LLM-Eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. arXiv:2305.13711. Retrieved from https://arxiv.org/abs/2305.13711

[29] Bulou Liu, Yiran Hu, Qingyao Ai, Yiqun Liu, Yueyue Wu, Chenliang Li, and Weixing Shen. 2025. Generating clarifying questions for conversational legal case retrieval without external knowledge. *ACM Transactions on Information Systems* 43, 4 (2025), 1–26. DOI: https://doi.org/10.1145/3736161

[30] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP '23)*. Houda Bouamor, Juan Pino, and Kalika Bali (Eds.), Association for Computational Linguistics, 2511–2522. DOI: https://doi.org/10.18653/V1/2023.EMNLP-MAIN.153

[31] Yiqi Liu, Nafise Sadat Moosavi, and Chenghua Lin. 2024. LLMs as narcissistic evaluators: When ego inflates evaluation scores. In *Proceedings of the Findings of the Association for Computational Linguistics (ACL '24)*. Association for Computational Linguistics, 12688–12701.

[32] Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. 2019. CEDR: Contextualized embeddings for document ranking. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19)*. Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.), ACM, New York, NY, 1101–1104. DOI: https://doi.org/10.1145/3331184.3331317

[33] Fangwen Mu, Lin Shi, Song Wang, Zhuohao Yu, Binquan Zhang, Chenxue Wang, Shichao Liu, and Qing Wang. 2023. ClarifyGPT: Empowering LLM-based code generation with intention clarification. arXiv:2310.10996. Retrieved from https://arxiv.org/abs/2310.10996

[34] Thong Nguyen and Andrew Yates. 2023. Generative retrieval as dense retrieval. arXiv:2306.11397. Retrieved from https://arxiv.org/abs/2306.11397

[35] 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 (CHIIR '17)*. Ragnar Nordlie, Nils Pharo, Luanne Freund, Birger Larsen, and Dan Russel (Eds.), ACM, New York, NY, 117–126. DOI: https://doi.org/10.1145/3020165.3020183

[36] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67.

[37] Hossein A. Rahmani, Xi Wang, Mohammad Aliannejadi, Mohammadmehdi Naghiaei, and Emine Yilmaz. 2024. Clarifying the path to user satisfaction: An investigation into clarification usefulness. In *Proceedings of the Findings of the Association for Computational Linguistics (EACL '24)*. Yvette Graham and Matthew Purver (Eds.), Association for Computational Linguistics, 1266–1277. Retrieved from https://aclanthology.org/2024.findings-eacl.84/

[38] Kimia Ramezan, Alireza Amiri Bavandpour, Yifei Yuan, Clemencia Siro, and Mohammad Aliannejadi. 2025. Multi-turn multi-modal question clarification for enhanced conversational understanding. arXiv:2502.11442. Retrieved from https://arxiv.org/abs/2502.11442

[39] 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 (ACL '18)*. Iryna Gurevych and Yusuke Miyao (Eds.), *Long Papers*, Vol. 1, Association for Computational Linguistics, 2737–2746. DOI: https://doi.org/10.18653/V1/P18-1255

[40] Sudha Rao and Hal Daumé, III. 2019. Answer-based adversarial training for generating clarification questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (NAACL-HLT '19)*. Jill Burstein, Christy Doran, and Thamar Solorio (Eds.), Association for Computational Linguistics, 143–155. DOI: https://doi.org/10.18653/V1/N19-1013

[41] Pengjie Ren, Zhumin Chen, Zhaochun Ren, Evangelos Kanoulas, Christof Monz, and Maarten de Rijke. 2021. Conversations with search engines: SERP-based conversational response generation. *ACM Transactions on Information Systems* 39, 4 (2021), 1–29. DOI: https://doi.org/10.1145/3432726

[42] Ruiyang Ren, Peng Qiu, Yingqi Qu, Jing Liu, Xin Zhao, Hua Wu, Ji-Rong Wen, and Haifeng Wang. 2024. BASES: Large-scale web search user simulation with large language model based agents. In *Proceedings of the Findings of the Association for Computational Linguistics (EMNLP '24)*. Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.), Association for Computational Linguistics, 902–917. DOI: https://doi.org/10.18653/v1/2024.findings-emnlp.50

[43] Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval* 3, 4 (2009), 333–389.

[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 *Proceedings of the Web Conference 2020 (WWW '20)*. ACM, New York, NY. DOI: https://doi.org/10.1145/3366423.3380193

[45] Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.), Association for Computational Linguistics, 2087–2097. DOI: https://doi.org/10.18653/v1/D18-1233

[46] Jost Schatzmann, Karl Weilhammer, Matt Stuttle, and Steve Young. 2006. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. 21, 2 (2006), 97–126. DOI: https://doi.org/10.1017/S0269888906000944

[47] Ivan Sekulic, 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 the Theory of Information Retrieval (ICTIR '21)*. Faegheh Hasibi, Yi Fang, and Akiko Aizawa (Eds.), ACM, New York, NY, 167–175. DOI: https://doi.org/10.1145/3471158.3472257

[48] Ivan Sekulic, Mohammad Aliannejadi, and Fabio Crestani. 2022. Evaluating mixed-initiative conversational search systems via user simulation. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining (WSDM '22)*. K. Selcuk Candan, Huan Liu, Leman Akoglu, Xin Luna Dong, and Jiliang Tang (Eds.), ACM, New York, NY, 888–896. DOI: https://doi.org/10.1145/3488560.3498440

[49] Ivan Sekulic, Mohammad Aliannejadi, and Fabio Crestani. 2024. Analysing utterances in LLM-based user simulation for conversational search. *ACM Transactions on Intelligent Systems and Technology* 15, 3 (2024), 1–22. DOI: https://doi.org/10.1145/3650041

[50] Ivan Sekulic, Weronika Lajewska, Krisztian Balog, and Fabio Crestani. 2024. Estimating the usefulness of clarifying questions and answers for conversational search. In *Proceedings of the 46th European Conference on Information Retrieval (ECIR '24)*. Nazli Goharian, Nicola Tonellotto, Yulan He, Aldo Lipani, Graham McDonald, Craig Macdonald, and Iadh Ounis (Eds.), Lecture Notes in Computer Science, Vol. 14610, Springer, 384–392. DOI: https://doi.org/10.1007/978-3-031-56063-7_30

[51] Ivan Sekulić, Lili Lu, Navdeep Singh Bedi, and Fabio Crestani. 2024. Simulating conversational search users with parameterized behavior. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '24)*. ACM, New York, NY. DOI: https://doi.org/10.1145/3673791.3698425

[52] Ivan Sekulic, Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, Andre Ferreira Manso, and Roland Mathis. 2024. Reliable LLM-based user simulator for task-oriented dialogue systems. In *Proceedings of the 1st Workshop on Simulating Conversational Intelligence in Chat (SCI-CHAT '24)*. Yvette Graham, Qun Liu, Gerasimos Lampouras, Ignacio Iacobacci, Sinead Madden, Haider Khalid, and Rameez Qureshi (Eds.), Association for Computational Linguistics, 19–35. Retrieved from https://aclanthology.org/2024.scichat-1.3/

[53] Chenhui Shen, Liying Cheng, Xuan-Phi Nguyen, Yang You, and Lidong Bing. 2023. Large language models are not yet human-level evaluators for abstractive summarization. In *Proceedings of the Findings of the Association for Computational Linguistics (EMNLP '23)*. Houda Bouamor, Juan Pino, and Kalika Bali (Eds.), Association for Computational Linguistics, 4215–4233. DOI: https://doi.org/10.18653/v1/2023.findings-emnlp.278

[54] Tao Shen, Xiubo Geng, Tao Qin, Daya Guo, Duyu Tang, Nan Duan, Guodong Long, and Daxin Jiang. 2019. Multi-task learning for conversational question answering over a large-scale knowledge base. 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, 2442–2451. DOI: https://doi.org/10.18653/v1/D19-1248

[55] Clemencia Siro, Mohammad Aliannejadi, and Maarten de Rijke. 2023. Understanding and predicting user satisfaction with conversational recommender systems. *ACM Transactions on Information Systems* 42, 2, Article 55 (Sep. 2023), 1–37. DOI: https://doi.org/10.1145/3624989

[56] Clemencia Siro, Mohammad Aliannejadi, and Maarten de Rijke. 2024. Context does matter: Implications for crowdsourced evaluation labels in task-oriented dialogue systems. In *Proceedings of the Findings of the Association for Computational Linguistics (NAACL '24)*. Kevin Duh, Helena Gomez, and Steven Bethard (Eds.), Association for Computational Linguistics, 1258–1273. DOI: https://doi.org/10.18653/v1/2024.findings-naacl.80

[57] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. LLaMA: Open and efficient foundation language models. arXiv:2302.13971. Retrieved from https://arxiv.org/abs/2302.13971

[58] Svitlana Vakulenko, Shayne Longpre, Zhucheng Tu, and Raviteja Anantha. 2021. Question rewriting for conversational question answering. In *Proceedings of the 14th International Conference on Web Search and Data Mining (WSDM '21)*. ACM, New York, NY, 355–363. DOI: https://doi.org/10.1145/3437963.3441748

[59] 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 (WWW '23)*. Ying Ding, Jie Tang, Juan F. Sequeda, Lora Aroyo, Carlos Castillo, and Geert-Jan Houben (Eds.), ACM, New York, NY, 3288–3298. DOI: https://doi.org/10.1145/3543507.3583420

[60] Xuchen Yao, Gosse Bouma, and Yi Zhang. 2012. Semantics-based question generation and implementation. *Dialogue & Discourse* 3, 2 (2012), 11–42.

[61] Se-Eun Yoon, Zhankui He, Jessica Echterhoff, and Julian McAuley. 2024. Evaluating large language models as generative user simulators for conversational recommendation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Kevin Duh, Helena Gomez, and Steven Bethard (Eds.), Association for Computational Linguistics, 1490–1504. DOI: https://doi.org/10.18653/v1/2024.naacl-long.83

[62] Hamed Zamani and Nick Craswell. 2020. Macaw: An extensible conversational information seeking platform. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. ACM, New York, NY. DOI: https://doi.org/10.1145/3397271.3401415

[63] Hamed Zamani, Susan T. Dumais, Nick Craswell, Paul N. Bennett, and Gord Lueck. 2020. Generating clarifying questions for information retrieval. In *Proceedings of the Web Conference 2020 (WWW '20)*. Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.), ACM/IW3C2, 418–428. DOI: https://doi.org/10.1145/3366423.3380126

[64] 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. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.), ACM, New York, NY, 1181–1190. DOI: https://doi.org/10.1145/3397271.3401160

[65] Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. 2023. Conversational information seeking. *Foundations and Trends® in Information Retrieval* 17, 3–4 (2023), 244–456. DOI: https://doi.org/10.1561/1500000081

[66] Yuyue Zhao, Jiancan Wu, Xiang Wang, Wei Tang, Dingxian Wang, and Maarten de Rijke. 2024. Let me do it for you: Towards LLM empowered recommendation via tool learning. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, New York, NY, 1796–1806.

[67] Ziliang Zhao, Zhicheng Dou, and Yujia Zhou. 2024. Generating intent-aware clarifying questions in conversational information retrieval systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24)*. Edoardo Serra and Francesca Spezzano (Eds.), ACM, New York, NY, 3384–3394. DOI: https://doi.org/10.1145/3627673.3679851

[68] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, et al. 2023. Judging LLM-as-a-judge with MT-bench and ChatBot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS '23)*. Curran Associates Inc., Article 2020.

[69] OpenAI. (2023). GPT-4 Technical Report. arXiv preprint. arXiv:2303.08774. Retrieved from https://arxiv.org/abs/2303.08774

[70] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems* 35 (2022), 27730–27744. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html

[71] 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. Retrieved from https://api.semanticscholar.org/CorpusID:271571434

[72] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. 2025. Qwen3 Technical Report. arXiv:2505.09388. Retrieved from https://arxiv.org/abs/2505.09388

Appendices

A Additional Methodology Details

In this section, we give additional details on the implementation of AGENT-CQ.

A.1 Hyperparameters

Our framework employs various hyperparameters, carefully chosen to balance performance and diversity.

Question generation:

- Temperature variation: We use temperatures ranging from 0.5 to 0.9, incrementing by 0.1. We set $n\_sets = 3$.
- Facet-based approach: Temperature is set to 0.7, top_p = 0.95, for Llama: top_k = 50 and max_length = 1,024.
- Baseline: A fixed temperature of 0.7 is used to generate 10 questions for each query.

Question filtering:

- We set $\alpha = 0.4$ in the filtering stage to balance relevance and clarification potential of the selected questions.
- Temperature is set to 0.7.

User simulation:

- Verbosity: 10–60 tokens.
- Cooperativeness: reveal probabilities 0.0–0.9.
- Answer generation: temperature = 0.7, top_p = 0.98, frequency_penalty = 0.5, presence_penalty = 0.2.

These parameters simulate diverse user behaviors while maintaining coherent responses.

CrowdLLM Evaluation. We use three GPT-4 instances to simulate diverse human judgments. The personas are defined as follows:

(1) *Strict judge* (temperature = 0.2) follows instructions strictly, prioritizes correctness, and is conservative in assigning high scores.
(2) *Typical judge* (temperature = 0.5) reflects average crowdworker behavior, balancing precision and flexibility.
(3) *Lenient judge* (temperature = 0.7) interprets tasks more liberally, rewards intent and creativity, and is more lenient overall.

The selection of these hyperparameters was based on:

- Extensive experimentation with various setups to optimize performance.
- Analysis of output quality and diversity across different parameter combinations.
- Alignment with observed patterns in human evaluation behaviors from prior crowdsourcing studies.

A.2 Prompts

This section lists the prompts used in different prompting strategies and stages of AGENT-CQ.

Facet-Based Prompt.

Facet generation

```
For the user query: '{query}'
Generate a list of 40 diverse facets that this query might be addressing.
This query represents multiple user information needs. Generate diverse facets to
capture these varied needs.
Ensure each facet is unique and explores different aspects or interpretations of the
query. Avoid repetition and strive for a wide range of perspectives in your facets.
```

Generating CQ

```
For the user query: '{query}'
And considering this specific facet: '{facet}'
Generate a clarifying question that addresses this facet and helps to better
understand the user's specific information need.
Use diverse language and question structure to formulate the questions.
```

Temperature-Variation Prompt.

```
for i in range(n_sets):
   For the user query: '{query}'

   Generate a set of 10 clarifying questions. The goal is to better understand the
   user's specific information need.

   This query represents multiple user information needs. Generate diverse clarifying
   questions to capture these varied needs.
   Ensure each question is unique and explores different aspects or interpretations of
   the query. Avoid repetition and strive for a wide range of perspectives in your
   questions.

   IMPORTANT GUIDELINES:
   1. Each question should aim to clarify a different aspect of the user's intent or
   information need.
   2. Ensure all questions are unique. Do not repeat questions.
   3. Focus on questions that will help narrow down or specify the user's request.
   4. Consider potential ambiguities or multiple interpretations of the query.
```

Scoring and Filtering Prompt.

```
Evaluate the following question for the user query: '{query}'
Question: "{question}"
Consider these aspects:
   1. Clarification: How well does this question help to better understand the user's
   original query?
   2. On Topic: To what degree does this question directly relate to the subject
   matter of the user's original query?

Provide a score (0-10) for each aspect and a brief explanation.
```

User Response Simulation Prompt.

```
You are a user who initially made this request: '{query}'.
Your actual information need is: '{facet}'.
Respond to the clarifying question based on this information need.

Your verbosity level is {verbosity_level}.
Your reveal probability is {reveal_probability:.2f}.
Keep your response short, ideally under {verbosity["max_tokens"]} tokens.

Remember: Your answer should not include any additional information that is not part
of your actual information need ('{facet}').
```

A.3 CrowdLLM Prompts

Below is an example of the CrowdLLM prompt for question complexity. Other metrics follow the same prompt except for the definition of the metric. Each metric is evaluated independently to avoid bias from previous metric ratings. Overall quality followed a slightly different approach, apart from having access to the query and system clarifying question, it also included the ratings from the other six metrics in order to ground the overall quality on these metrics.

Question Complexity Evaluation Prompt.

```
As a user, you are evaluating the complexity of the system's clarifying question in
relation to your original query.

Definition:
    - Question Complexity: The degree to which the clarifying question introduces
    technical terms, specialized concepts, or requires domain-specific knowledge not
    present in the original query.

Scale:
    1-10, where 1 is very simple (uses only general terms and concepts) and 10 is
    highly complex (introduces specialized terminology or concepts).

Your original query: "{original_query}"
System's clarifying question: "{system_question}"

Evaluate the complexity of the system's question compared to your original query.
Consider:
    1. Does it introduce technical terms or jargon not present in the original query?
    2. Does it require specialized knowledge that might not be evident from the
    original query?
```

Overall Quality Evaluation Prompt.

```
As a user, you are providing an overall evaluation of the system's clarifying
question, taking into account your ratings from other aspects.

Definition:
    - Overall Quality: Your comprehensive assessment of how well the system's
    clarifying question helps you get a better response to your original query,
    considering clarity, relevance, specificity, and usefulness.

Scale:
    1-10, where 1 is the lowest quality and 10 is the highest quality.

Your original query: "{original_query}"
System's clarifying question: "{system_question}"

Your rating from the other metrics: {other_ratings}

Consider these ratings and provide an overall evaluation of the system's clarifying
question quality. Explain your reasoning, referencing your other metric ratings.
```

B Elicited Response Analysis

We classified expected response types of CQ into four categories: Yes/No, Multiple-choice, Open-ended, and Factual. Yes/No questions are identified by auxiliary verb initiation (e.g., "Are," "Is," "Do"). Multiple-choice questions contain explicit options or suggest selection from a limited set. Factual questions use specific question words (e.g., "When," "Where," "Who") seeking concise information. Open-ended questions, not fitting other categories, typically invite elaboration. We implemented this classification using regular expressions and conditional logic.

To ensure accuracy, particularly for edge cases, we followed the automated classification with a manual review process. This combined approach allowed us to systematically analyze large volumes of CQ while maintaining high classification accuracy. By examining patterns, categories, and response types, we gained insights into how different models and prompting strategies influence the structure and intent of CQ generated by language models in conversational information-seeking contexts.

C Data Statistics and Sample Generated Data by Various Models

In Table C1, we report the number of questions generated by different systems and the overall filtered questions. Table C2 shows sample CQ generated by various models for each UQ. Table C3 shows a sample of user responses to the human-generated CQ by humans and an LLM.

Given that GPT-Temp demonstrated superior performance in question quality, we use it to generate CQ for all queries in the ClariQ collection. To maintain consistency with the original dataset, we match the number of CQ per facet as present in the original data. This approach ensures a fair retrieval comparison while using the enhanced question-generation capabilities of GPT-Temp. For comparative analysis, we also generate a dataset using a GPT-Baseline model. This allows us to assess the relative performance gains achieved by GPT-Temp and provides insights into the effectiveness of different prompting strategies in generating CQ for CS systems.

Table C1. Number of Queries Used in the Experiments and Total Questions Generated and Filtered for Each Model

Model	Queries	Generated	Filtered
Llama3.1	50	2,000	500
GPT-Baseline	50	500	500
GPT-Facet	50	2,000	500
GPT-Temp	50	1,500	500
H-Gen	50	1,200	500
Total	50	7,200	2,500

Table C2. CQ Generated by Different Models for Various Queries

Query	Llama3.1	GPT-Baseline	GPT-Facet	GPT-Temp	H-Gen
Tell me about atypical squamous cells	What specific information about atypical squamous cells are you looking for?	Is there a specific type of testing or screening you're curious about?	What specific cytopathological techniques are you interested in regarding the diagnosis of atypical squamous cells?	Are you interested in the treatment options available if atypical squamous cells are detected?	Would you like to know if there is a cure?
Tell me information about computer programming	What specific aspect of the history of computer programming are you interested in?	Do you want to know about programming tools and software?	What specific problem-solving strategies or techniques in programming are you interested in learning more about?	Are you looking for beginner resources, advanced topics, or something in between?	Are you interested in a coding bootcamp?
Tell me about defender	Are you interested in learning more about the features and capabilities of defender as a cybersecurity tool?	Are you referring to a specific type of defender, like in sports?	Are you looking for an explanation of defensive coding practices in general, or are you interested in specific techniques or examples of how they are implemented in programming?	Are you referring to a specific type of defender, such as a legal defender, a sports defender, or a cybersecurity defender?	Are you interested in the television series defender?
How to cure angular cheilitis	What specific over-the-counter treatments are you looking for to cure angular cheilitis?	What symptoms are you experiencing with angular cheilitis?	Are you looking for home remedies or professional medical treatments to manage angular cheilitis in children?	Would you prefer natural remedies, over-the-counter options, or prescription medications?	Are you looking for the definition of angular cheilitis?

Table C3. Comparison of Human and LLM Answers to Human-Generated CQ

Query	**User-information-need**	**Clarifying-question**	**Human-answer**	**LLM-answer**
Tell me about atypical squamous cells	What do atypical squamous cells mean on a pap smear test?	Would you like to know if there is a cure?	No I want to know what it means when atypical squamous cells appear on a pap smear test	Not specifically. Just curious about their meaning.
Tell me information about computer programming	What type of careers are there for computer programmers?	Are you interested in a coding bootcamp?	No I want to know what career options programmers have	I'm curious about the various job roles within computer programming.
Tell me about defender	Find information on the Land Rover Defender sport-utility vehicle.	Are you interested in the television series defender 3?	No I am looking for information about the land rover defender	Not exactly, I'm interested in a specific vehicle model.
How to cure angular cheilitis	What home remedies are there for angular cheilitis?	Are you looking for the definition of angular cheilitis?	No home remedies	I'm more interested in finding home remedies for angular cheilitis.

Received 5 June 2025; revised 27 January 2026; accepted 30 March 2026