When in Doubt, Ask First: A Unified Retrieval Agent-Based System for Ambiguous and Unanswerable Question Answering

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have shown strong capabilities in Question Answering (QA), but their effectiveness in high-stakes, closed-domain settings is often constrained by hallucinations and limited handling of vague or underspecified queries. These challenges are especially pronounced in Vietnamese, a low-resource language with complex syntax and strong contextual dependence, where user questions are often short, informal, and ambiguous. We introduce the Unified Retrieval Agent-Based System (URASys), a QA framework that combines agent-based reasoning with dual retrieval under the Just Enough principle to address standard, ambiguous, and unanswerable questions in a unified manner. URASys performs lightweight query decomposition and integrates document retrieval with a question–answer layer via a two-phase indexing pipeline, engaging in interactive clarification when intent is uncertain and explicitly signaling unanswerable cases to avoid hallucination. We evaluate URASys on Vietnamese and English QA benchmarks spanning single-hop, multi-hop, and real-world academic advising tasks, and release new dual-language ambiguous subsets for benchmarking interactive clarification. Results show that URASys outperforms strong retrieval-based baselines in factual accuracy, improves unanswerable handling, and achieves statistically significant gains in human evaluations for clarity and trustworthiness.

1 Introduction

Large Language Models (LLMs) have become a cornerstone of modern *Question Answering* (QA) systems (Rasool et al., 2024), demonstrating strong fluency across diverse tasks. As their capabilities expand, QA systems powered by LLMs are increasingly deployed in high-stakes, closed-domain environments such as academic advising, where answers must remain precise and context-aware (Raiaan et al., 2024). Questions on tuition fees, course prerequisites, or institutional policies often carry significant consequences: errors can delay graduation, incur financial penalties, and mislead enrollment decisions, making factual grounding and contextual sensitivity essential (Nguyen and Quan, 2025). This challenge is amplified in low-resource, nuance-rich languages such as Vietnamese, where syntactic variation and strong context dependence complicate interpretation and motivate creating dedicated datasets and benchmarks (Pham et al., 2024). Queries are frequently short, informal, and underspecified, reflecting natural advising conversations and exposing limitations of conventional QA pipelines without interactive clarification.

Although LLMs provide strong generative capabilities, they are inherently probabilistic and prone to *hallucination* when facing vague or incomplete questions (Li et al., 2024). *Retrieval-Augmented Generation* (RAG) (Lewis et al., 2020) mitigates this by grounding responses in external evidence, yet most RAG pipelines assume queries are fully specified and answerable. They often retrieve loosely related passages and attempt to answer despite insufficient evidence, rarely engaging in targeted clarification (Fan et al., 2024). Such behavior is especially problematic in advising contexts where users expect not only answers but reliable guidance. Recent advances such as IRCoT (Trivedi et al., 2023), MiniRAG (Fan et al., 2025), LightRAG (Guo et al., 2024), NodeRAG (Xu et al., 2025), and HippoRAG (Liu et al., 2025) improve evidence synthesis and multi-hop retrieval through interleaved reasoning and graph-based routing, but they still treat queries as static inputs and lack mechanisms for decomposition-driven understanding, interactive clarification, and explicit *unanswerable handling*. These gaps highlight the need for a reasoning-driven QA framework that can jointly decide when to answer, when to clarify, and when to explicitly signal no-answer without over-relying on retrieval confidence.

Table 1: Illustrative examples showing *User* (U) and *URASys* (S) handling different query types in educational QA.

Scenario	Example
Normal: direct, well-specified question	**U:** What are the prerequisites for Machine Learning 101? **S:** Introduction to Programming and Basic Statistics.
Ambiguous: questions with multiple interpretations, insufficient context, or overly broad scope (Wang et al., 2023)	**U:** How much is the fee for this program? **S:** The program has a tuition fee, a lab fee, and a registration fee. Could you clarify which one? **U:** Tuition. **S:** Tuition fee is 2 000 USD per semester.
Unanswerable: specific query with no matching information in the database	**U:** What is the course instructor's office phone number? **S:** Sorry, our advising database stores only email addresses and no phone numbers.

To address these gaps, we introduce the ***Unified Retrieval Agent-Based System*** (URASys), a QA framework for closed-domain settings with underspecified or critical queries. Unlike prior RAG pipelines, URASys leverages agent-based reasoning and a dual retrieval architecture under a *Just Enough* paradigm. It prioritizes understanding before answering, engages in clarification when user intent is ambiguous, and explicitly signals unanswerable when evidence is insufficient rather than hallucinating. A central *Manager Agent* performs query decomposition to infer intent and split complex questions into sub-queries for better evidence aggregation. It coordinates two specialized retrieval agents: (i) a *Document Retrieval Agent* over a hybrid *chunk-and-title* corpus index for lexical and semantic search, and (ii) a *FAQ Retrieval Agent* querying an automatically generated *Frequently Asked Questions* (FAQs) repository created via an *ask-and-augment* procedure. This two-phase indexing pipeline transforms raw documents into both evidence chunks and a standardized question–answer layer, enabling URASys to combine fast FAQ-style lookup with grounded document reasoning in a unified architecture.

URASys jointly addresses three critical QA scenarios: (1) resolving ambiguous queries via interactive clarification, (2) handling unanswerable cases through cross-source evidence synthesis and reasoning-driven decisions, and (3) answering standard queries with grounded single-hop or multi-hop reasoning, as illustrated in Table 1. The clarification loop mirrors human advisory behavior, while the dual retrieval pipeline reflects the natural workflow of consulting FAQs and policy documents. Our contributions are summarized as follows.

- We introduce URASys, an agent-based QA framework that integrates query decomposition, dual retrieval, and interactive clarification under a Just Enough principle. This paradigm prioritizes understanding over generation, enabling unified handling of standard, ambiguous, and unanswerable queries while improving accuracy and robustness in closed-domain QA. We further propose a two-phase indexing pipeline for dual retrieval, effective in low-resource languages without requiring complex graph infrastructure.
- We comprehensively evaluate URASys on Vietnamese and English QA benchmarks covering single-hop and multi-hop closed-domain tasks and a real-world academic advising dataset, including unanswerable subsets. URASys outperforms both traditional and advanced RAG baselines in factual accuracy.
- We release new ambiguous subsets targeting interactive clarification, enabling systematic evaluation of underspecified queries in both English and Vietnamese and establishing a benchmark for this underexplored setting.
- We conduct real-world human evaluations with end users interacting with the deployed system, demonstrating practical effectiveness in live advising workflows and statistically significant gains in user satisfaction. These results highlight the broader applicability of URASys to other high-stakes, closed-domain QA scenarios beyond academic advising.

All code and datasets will be released to ensure full reproducibility and to support future research.

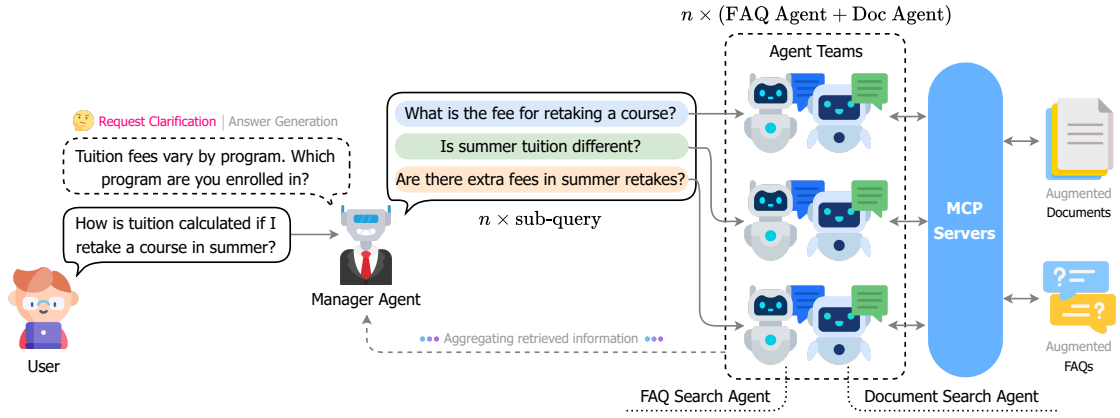

Figure 1: Overview of the URASys framework. Each user query is decomposed into n sub-queries, handled by parallel agent teams consisting of FAQ and document agents. Retrieved evidence is aggregated and used to generate a final answer, with optional clarification or explicit no-answer signaling if needed.

2 Related Works

2.1 RAG and Recent Advances

RAG has emerged as a promising approach for improving factual accuracy in QA systems by grounding LLM outputs in retrieved evidence (Lewis et al., 2020). Standard pipelines often combine dense retrievers, token-based methods such as BM25, or hybrid approaches with generative models to produce context-aware responses (Fan et al., 2024), but typically assume well-formed inputs and underperform when queries are vague or underspecified. Recent studies propose advanced retrieval-reasoning architectures: IRCoT (Trivedi et al., 2023) alternates retrieval and reasoning in multi-step QA; LightRAG (Guo et al., 2024) and MiniRAG (Fan et al., 2025) enhance indexing with graph structures and semantic-aware topologies; NodeRAG (Xu et al., 2025) integrates heterogeneous graphs for structured evidence; HippoRAG (Gutierrez et al., 2024) employs hierarchical memory to capture long-range dependencies. While these models excel on benchmarks, they rely on high-quality inputs and complex infrastructure, posing challenges in resource-constrained, high-stakes domains such as educational QA. Critically, they lack mechanisms for interactive clarification and explicit unanswerable handling, motivating architectures combining modular retrieval with clarification-first interaction for ambiguous and underspecified queries.

2.2 Interactive Clarification, Unanswerable Handling, and Multi-Agent QA

Most QA systems treat user queries as fully specified and directly answerable. Interactive clarification challenges this assumption by posing follow-up questions to resolve vagueness (Guo et al., 2021), showing promise in task-oriented settings but remaining underexplored in academic advising, where precision is critical (Deng et al., 2024). Surveys on Asking Clarification Questions datasets highlight the lack of standardized resources for training systems to handle ambiguity (Rahmani et al., 2023). In parallel, multi-agent QA decomposes tasks into retrieval, reasoning, and planning roles (Viswanathan et al., 2022; Elizabeth et al., 2025; Deng et al., 2025), but few integrate lightweight retrieval with clarification into deployable pipelines for informal, context-dependent queries. Work on unanswerable QA has focused mainly on extractive benchmarks like SQuAD 2.0 (Rajpurkar et al., 2018), with limited evaluation in LLM-based RAG and rare integration of cross-source synthesis with explicit no-answer signaling. These gaps motivate URASys, which combines clarification, agent-based reasoning, and dual-agent retrieval to decide when a query is answerable, when it requires interactive clarification, or when it should be explicitly marked as unanswerable. They further underscore the need for Ambiguous QA datasets in both English and low-resource, nuance-rich languages such as Vietnamese, with a particular emphasis on educational QA contexts.

2.3 System Overview

Figure 1 illustrates the overall architecture of URASys. When a user submits a query q, the *Manager Agent* first analyzes its structure and semantics, then decomposes it into a set of sub-queries

$\{q_1, q_2, \ldots, q_n\}$. Each sub-query q_i is assigned to a dedicated agent team $\mathcal{A}_i$, which comprises two specialized components: a *FAQ Search Agent* and a *Document Search Agent*. These teams operate concurrently, retrieving relevant evidence from both an augmented FAQ repository $\mathcal{F}$ and a structured document corpus $\mathcal{D}$, yielding evidence sets $E_i = \mathcal{A}_i(q_i, \mathcal{F}, \mathcal{D})$. Once retrieval is complete, the Manager Agent aggregates the results into a unified evidence pool $E = \bigcup_{i=1}^{n} E_i$, which serves as the basis for generating the final answer. If the evidence is insufficient or contradictory, the system proactively engages the user in a clarification round before finalizing the response, and in rare cases where both retrieval streams provide no supporting signals, URASys gracefully returns an explicit no-answer response. This architecture enables URASys to handle ambiguous, incomplete, or context-dependent queries with high precision and modularity, particularly in educational domains.

2.4 Modular Agent Design

URASys follows a modular multi-agent architecture inspired by how human advisors handle complex or underspecified queries. In real-world educational settings, effective advising typically involves two key behaviors: (i) seeking clarification when a user's intent is unclear, and (ii) consulting multiple sources to ensure accurate and comprehensive responses. These practices motivate the separation of responsibilities in URASys, enabling each agent to specialize while maintaining coherent coordination through a central controller.

Manager Agent The *Manager Agent* is the system's central reasoning component. It orchestrates the workflow by decomposing the user query into sub-queries, delegating them to retrieval agents, evaluating the evidence, and deciding whether the system is ready to respond confidently. Its decision-making follows the *Just Enough* principle: an answer is generated only when the evidence is both sufficient and internally consistent. To implement this, the agent adopts two complementary prompting strategies: *Tree-of-Thought* (Yao et al., 2023), which explores multiple reasoning paths in parallel, and *Chain-of-Thought* (Wei et al., 2022), which enforces coherent, step-by-step logic. If the aggregated evidence is inconclusive or contradictory, the Manager Agent applies *ask-before-answer*: it refrains from speculation and initiates clarification to refine the user query. If clarification fails or key information is missing, it explicitly reports that no answer can be provided. This iterative reasoning workflow is formalized in Algorithm 1.

Algorithm 1: Manager Agent Reasoning

Input: Query q, LLM p_θ, FAQ corpus $\mathcal{F}$, document corpus $\mathcal{D}$, max attempts T
Output: Answer a or clarification q_c

$t \leftarrow 0$;
while $t < T$ **do**
  $S \leftarrow \texttt{Decompose}(q)$;
  $E \leftarrow \emptyset$;
  foreach $q_i \in S$ **do**
    $E \leftarrow E \cup \texttt{FAQSearch}(q_i, \mathcal{F}) \cup \texttt{DocSearch}(q_i, \mathcal{D})$;
  if $\texttt{IsSpecific}(q)$ ***and*** $\texttt{HasDirectAnswer}(E)$ **then**
    $a \leftarrow \texttt{GenerateAnswer}(p_\theta, E)$;
    return a ;
  else if $\texttt{IsBroad}(q)$ ***and*** $\texttt{RevealCategories}(E)$ **then**
    $q_c \leftarrow \texttt{ExtractCategories}(E)$;
    return $\texttt{AskClarification}(q_c)$;
  else if $\texttt{IsVague}(q)$ ***or*** $\texttt{Insufficient}(E)$ **then**
    if $t + 1 < T$ **then**
      $q \leftarrow \texttt{Refine}(q, E)$;
    else
      return $\texttt{NoInformationFound}()$;
  $t \leftarrow t + 1$;

Retrieval Sub-Agents To retrieve information from distinct sources, the system instantiates two specialized LLM-based retrieval agents: one for FAQs and one for official documents. Each sub-query q_i is processed **concurrently** by a dedicated agent pair A_i, which is instantiated dynamically and invoked through a unified tool interface, implemented as a function named `search_information` and called by the Manager Agent.

- The *FAQ Search Agent* is optimized for high-precision lookup over a curated FAQ repository $\mathcal{F}$. It performs lightweight iterative search, with at most a few reformulation attempts based on result adequacy.
- The *Document Search Agent* performs semantic retrieval over a structured corpus of academic and administrative documents $\mathcal{D}$. This agent follows a more elaborate prompting

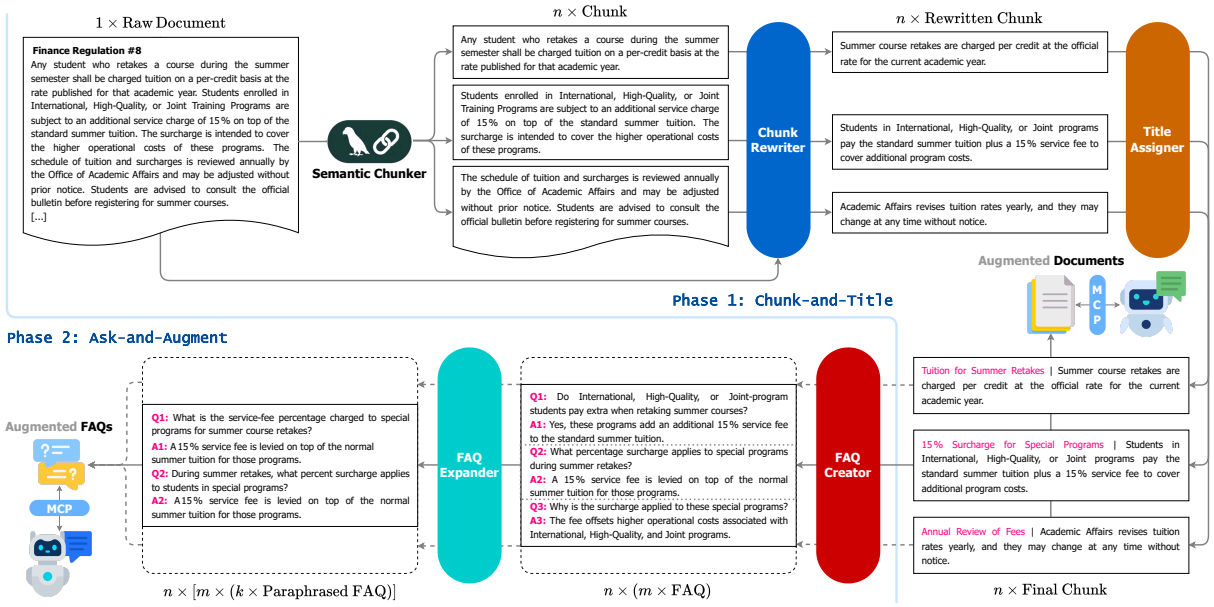

Figure 2: Two-phase indexing pipeline in URASys. Phase 1 (Chunk-and-Title) processes raw documents into coherent text blocks composed of a concise title and a rewritten chunk, enabling effective hybrid document retrieval. Phase 2 (Ask-and-Augment) generates and paraphrases question-answer pairs from each document block, forming a high-coverage and query-resilient FAQ corpus.

loop, allowing multiple reformulations when necessary. At each step, it analyzes the query, refines the search expression, and evaluates results to decide whether to continue or stop.

Both agents are strictly grounded: their final responses must be based solely on content returned by their respective retrieval tools, with no speculative or hallucinated generation. Each agent communicates with its backend service using the *Model Context Protocol* (MCP) over *Server-Sent Events* (SSE), allowing low-latency streaming of search results. This behavior implements a constrained form of CoT reasoning applied externally via tool outputs rather than internal deliberation alone.

2.5 Hybrid Retrieval Technique

Each retrieval sub-agent in URASys employs a hybrid search strategy that combines lexical and semantic signals via BM25 and dense vector retrieval (Fan et al., 2024). Rather than aggregating scores via a weighted linear combination (e.g., $\alpha \cdot \text{Dense} + (1-\alpha) \cdot \text{BM25}$), which requires manual tuning of α, we adopt *Reciprocal Rank Fusion* (RRF) (Cormack et al., 2009), a simple yet effective rank-based method that merges results without assuming score normalization or compatibility. Given two ranked lists $\mathcal{L}_1$ and $\mathcal{L}_2$ from BM25 and dense retrieval respectively, the fused score for a document d is computed as shown in Equation 1.

$$s(d) = \sum_{i=1}^{2} \frac{1}{k + \text{rank}_{\mathcal{L}_i}(d)} \tag{1}$$

Here, $\text{rank}_{\mathcal{L}_i}(d)$ denotes the position of d in list $\mathcal{L}_i$, and k is a smoothing constant. This formulation ensures that documents ranked highly in either modality receive strong fused scores, enhancing both robustness and interpretability.

2.6 Proposed Indexing Strategy

To support high-quality retrieval for the two specialized agents in URASys, namely the Document Search Agent and the FAQ Search Agent, we design a two-phase indexing pipeline as illustrated in Figure 2. Each phase constructs a distinct type of retrieval unit tailored to the specific needs of its corresponding agent. This pipeline is tightly coupled with a suite of LLM-based modules, including the *Chunk Rewriter*, *Title Assigner*, *FAQ Creator*, and *FAQ Expander*, all implemented through prompt-based techniques (Kamath et al., 2024). These components enable flexible adaptation to new domains and play a central role in generating semantically rich and query-resilient retrieval units.

Phase 1: Chunk-and-Title Given a raw document d, we apply a semantic chunking module to segment it into a set of discourse-aligned fragments

$\texttt{SemanticChunker}(d) \rightarrow \{d_1, d_2, \ldots, d_n\}$, where each d_i is a semantically coherent span. Because these initial fragments may include mid-sentence boundaries or depend on broader context, each d_i is rewritten with document-level context using a context-aware module $\texttt{ChunkRewriter}(d_i, d) \rightarrow c_i$ to produce a self-contained and fluent chunk. Each rewritten chunk c_i is then passed through a title assignment function $\texttt{TitleAssigner}(c_i) \rightarrow t_i$, which generates a concise and descriptive title summarizing the core content. Finally, the title and chunk are concatenated into a single final chunk $x_i = \texttt{Concat}(t_i, c_i)$, forming an augmented document corpus

$$\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$$

where each x_i is a unified retrieval unit that combines a semantic anchor with its associated content. This corpus serves as the retrieval basis for the Document Search Agent, which performs hybrid retrieval over each x_i using both lexical and semantic signals. The inclusion of titles enhances retrieval effectiveness by injecting salient keywords that benefit sparse retrieval, while preserving the semantic richness of the underlying chunk.

Phase 2: Ask-and-Augment Each final chunk $x_i \in \mathcal{D}$, which contains both the generated title and rewritten chunk, is passed to a FAQ generation module $\texttt{FAQCreator}(x_i)$ to synthesize m canonical FAQ pairs $\{(q_{i,1}, a_{i,1}), \ldots, (q_{i,m}, a_{i,m})\}$. These questions are designed to reflect plausible user intents, guided by the semantic scope introduced by the title and grounded in the content of the chunk. To improve robustness against surface variation in user phrasing, each question $q_{i,j}$ is paraphrased into k diverse variants via $\texttt{FAQExpander}(q_{i,j}) \rightarrow \{q_{i,j}^{(1)}, \ldots, q_{i,j}^{(k)}\}$, all sharing the same answer $a_{i,j}$. The result is a richly paraphrased FAQ corpus

$$\mathcal{F} = \{(q_{i,j}^{(l)}, a_{i,j}) \mid i \in [1, n],\ j \in [1, m],\ l \in [1, k]\}$$

which serves as the retrieval basis for the FAQ Search Agent. The inclusion of multiple paraphrases for each intent improves coverage and increases robustness to syntactic and stylistic variation, which is especially important for Vietnamese, where the same meaning can be expressed in many different ways.

3 Experimentations

We conduct two experiments to evaluate URASys in terms of retrieval performance and real-world usability. The first benchmarks our system against a range of *state-of-the-art* (SOTA) and classical RAG baselines across multiple public QA datasets, covering diverse reasoning types and domain settings. The second is a user study with real end-users to assess practical effectiveness and user trust under different interaction scenarios.

3.1 Datasets

We evaluate URASys on five datasets spanning three QA settings: single-hop, multi-hop, and domain-specific queries. For each public dataset, we sample 1,000 representative questions. Several include **unanswerable cases**, making them well-suited for testing the system's ability to handle uncertainty and trigger clarification.

Single-hop QA SQuAD 2.0 (Rajpurkar et al., 2018) has English questions from Wikipedia, including adversarially unanswerable. UIT-ViQuAD 2.0 (Nguyen et al., 2022) is its Vietnamese counterpart with similar design.

Multi-hop QA HotpotQA (Yang et al., 2018) and VIMQA (Le et al., 2022) require reasoning over multiple documents. VIMQA adapts this to Vietnamese, making it suitable for low-resource multi-hop evaluation.

Domain-specific QA UniQA is a custom dataset of real-world Vietnamese student queries on university admissions. Each question links to official academic documents, reflecting URASys's target deployment scenario.

We also build **ambiguous subsets** from SQuAD 2.0, UIT-ViQuAD 2.0, and UniQA. These include underspecified questions requiring clarification, paired with ground-truth paraphrases of clarified queries, providing a dedicated benchmark for interactive clarification in both English and Vietnamese.

3.2 Evaluation Metrics

Standard metrics like *Exact Match* (EM) and token-level F1 often overlook reasoning quality, especially in multi-hop or underspecified scenarios (Schuff et al., 2020). We instead use the *LLM-as-a-Judge* protocol (Gu et al., 2024), where an external model scores answer correctness and explanation quality. For unanswerable subsets, models must indicate insufficient information, while for ambiguous ones they should request necessary clarifications. We also conduct a *human evaluation*, with participants rating outputs on seven dimensions (e.g., factuality, trust) using a 5-point *Likert*

Table 2: Answer correctness percentages (↑) across five QA benchmarks. For datasets with unanswerable and ambiguous questions (SQuAD 2.0, UIT-ViQuAD 2.0, UniQA), results are split into Overall, Unanswerable (Unans.), and Ambiguous (Ambig.) subsets. Best scores are in **bold**; second-best are underlined.

Method	SQuAD 2.0		UIT-ViQuAD 2.0		HotpotQA	VIMQA	UniQA		Ambiguous Subsets		
	Overall	Unans.	Overall	Unans.			Overall	Unans.	SQuAD	UIT-ViQuAD	UniQA
Naive RAG (Dense)	30.3	7.2	18.2	2.9	11.3	41.2	40.1	9.2	14.8	11.6	13.8
Naive RAG (BM25)	30.1	13.6	18.7	3.4	11.8	40.0	39.8	9.3	12.9	31.3	8.6
Naive RAG (Hybrid)	30.3	8.3	18.3	3.7	11.6	42.4	41.3	9.03	10.6	21.3	3.2
IRCoT	45.3	33.4	46.9	28.0	43.2	20.7	56.7	9.4	13.7	31.7	67.5
LightRAG	43.4	36.2	44.3	49.0	59.0	76.0	51.8	51.0	18.2	45.6	68.2
MiniRAG	49.0	10.6	55.0	11.3	21.7	54.5	52.6	49.0	17.5	70.5	36.9
NodeRAG	44.1	23.0	48.1	8.1	56.7	45.3	67.7	14.0	69.3	68.3	32.5
HippoRAG 2	67.3	56.2	78.8	54.0	60.3	75.0	50.7	13.0	67.4	62.7	49.7
URASys (Ours)	**75.0**	**83.7**	**80.0**	**86.5**	**90.0**	**83.2**	**85.0**	**82.6**	**71.9**	**73.9**	**81.2**

scale (Batterton and Hale, 2017), complementing automatic metrics with user-centered feedback.

3.3 Baselines

We evaluate URASys by comparing it against the following baselines.

Naive RAG A basic RAG pipeline using BM25, dense retrieval, and hybrid retrieval with score interpolation (Fan et al., 2024).

IRCoT + SOTA LLM Multi-step QA approach interleaving retrieval and CoT reasoning using a SOTA LLM (Trivedi et al., 2023).

LightRAG Incorporates graph structures into text indexing and retrieval (Guo et al., 2024).

MiniRAG Lightweight system with small LLM and heterogeneous graph index for efficient structured retrieval (Fan et al., 2025).

NodeRAG Graph-based framework integrating structured evidence for improved multi-hop retrieval (Xu et al., 2025).

HippoRAG 2 Retrieval system inspired by hippocampal theory for better long-term knowledge integration (Gutiérrez et al., 2025).

3.4 Implementation Details

To ensure consistency and fairness, we adopt Google's `gemini-2.0-flash`[1] as the LLM backbone for all baselines. For embeddings, we use OpenAI's `text-embedding-3-large`[2], a high-performance multilingual model suited to our diverse datasets. All advanced baselines run with default hyperparameters to reflect typical out-of-the-box performance. For LLM-as-a-Judge evaluation, we use the GPT-4o API[3]. For our custom ambiguous subsets, prompts are designed to enable the LLM to ask users for clarification when additional information can improve answer accuracy.

3.5 Results and Analysis

Table 2 reports answer correctness percentages across five QA benchmarks, divided into Overall, Unanswerable, and Ambiguous subsets. URASys consistently outperforms all baselines across all subsets, with especially large gains on unanswerable (83.7% on SQuAD 2.0, 86.5% on UIT-ViQuAD 2.0) and ambiguous questions (up to 81.2% on UniQA), surpassing the second-best HippoRAG 2 by significant margins. While models like NodeRAG and HippoRAG 2 perform well on English data, their scores drop notably on Vietnamese ambiguous and unanswerable subsets, exposing cross-lingual challenges. Lightweight models such as MiniRAG show competitive results on some Vietnamese ambiguous data but lag behind URASys's robust agent-based dual reasoning and interactive clarification. Traditional RAG baselines with simpler retrieval struggle on complex queries, underscoring the advantage of URASys's advanced architecture. Overall, URASys demonstrates strong generalization across languages, domains, and question complexities, effectively addressing real-world QA challenges, especially in low-resource, multilingual contexts.

Table 3 reports results from a real-world human evaluation of URASys. We deployed the system to two groups of prospective university applicants:

[1] https://ai.google.dev/gemini-api/docs/models/#gemini-2.0-flash

[2] https://platform.openai.com/docs/models/text-embedding-3-large

[3] https://platform.openai.com/docs/models/gpt-4o

Table 3: Human evaluation scores from end-user deployment. *Accuracy* is binary; *Number of Thoughts* (NoT) counts reasoning steps per answer; other metrics are rated on a 5-point Likert scale (↑).

Group	Accuracy	NoT	Explanation Quality	Trust
G1	0.80	1.95	4.51	4.26
G2	0.88	2.24	4.76	4.37

Table 4: Ablation study results across five QA datasets under the LLM-as-a-Judge protocol.

System Variant	SQuAD 2.0	UIT-ViQuAD 2.0	HotpotQA	VIMQA	UniQA
URASys	**0.75**	**0.80**	**0.90**	**0.83**	**0.85**
w/o FAQ Search Agent	0.73	0.26	0.43	0.14	0.48
w/o Document Search Agent	0.59	0.16	0.86	0.15	0.62
w/o Manager Decomposition	0.72	0.22	0.87	0.21	0.63
w/o Proposed Indexing	0.71	0.29	0.85	0.29	0.64

10 first-year students (G1) and *10 high school seniors* (G2), each tasked with 20 randomly sampled queries about common university-related topics. Across both groups, URASys achieved high accuracy (>0.80) and answered with an average of only two reasoning steps. Participants rated the system highly in both explanation quality and trustworthiness (>4.2), highlighting its potential for real-world deployment in educational settings.

3.6 Ablation Study

We conduct an ablation study by removing individual components of URASys and evaluating their impact on answer correctness across five benchmark datasets, as summarized in Table 4.

The full URASys system consistently achieves the highest scores, confirming that its effectiveness stems from the synergy between its modules. On SQuAD 2.0, an English single-hop dataset, removing any single component has limited effect (drop below 0.05), indicating that no individual module dominates in well-specified English queries. In contrast, performance drops sharply on ViQuAD 2.0, a Vietnamese single-hop dataset, especially when either the sub-agents, particularly the Document Search Agent, or the query decomposition module is removed, with accuracy falling by more than 60%. Similar degradations occur when replacing the Chunk-and-Title indexing with a standard chunking baseline, highlighting the indexing strategy's importance for handling ambiguous Vietnamese inputs. For multi-hop datasets, the absence of the FAQ Search Agent causes a 52% drop on HotpotQA and a 69% drop on VIMQA, showing that decomposing queries and retrieving supporting FAQs is critical for multi-step reasoning, particularly in low-resource languages. On UniQA, a real-world educational dataset, removing the FAQ Search Agent leads to the steepest decline (from 0.85 to 0.48), while other ablations reduce accuracy by 20–25%. These results underscore the central role of FAQ retrieval in educational domains, where queries are often vague or fragmented.

4 Conclusion

We present URASys, a modular and interaction-aware QA system tailored for educational scenarios. Evaluated on five benchmarks, including multi-hop and real-world academic datasets, URASys consistently outperforms retrieval-based baselines in factual accuracy and user trust. Its effectiveness stems from integration of query decomposition, dual-agent retrieval, and structure-aware indexing, as shown by our ablation study. While designed for education, the system architecture generalizes well to domains where queries tend to be vague, underspecified, or context-dependent, such as technical support or legal consultation. In such cases, URASys can identify missing information and opt not to answer until clarification is obtained, preserving factual integrity. Future directions include improving intent alignment in open-ended queries, seamless updates to evolving knowledge sources, and gradually replacing commercial LLM APIs with in-house lightweight models.

Limitations

While URASys demonstrates strong performance and broad adaptability across QA scenarios, several practical limitations remain. First, the system relies on a prompting strategy that may require careful tuning and ongoing maintenance, particularly in dynamic or evolving domains. Second, although overall computation is lightweight and stable, the use of multiple LLM calls across agents can incur notable monetary cost when relying on commercial APIs. Third, while URASys is designed for responsiveness and clarification, latency may increase for complex queries that involve deep decomposition or iterative refinement. These trade-offs between transparency, flexibility, and cost highlight directions for future work, including more streamlined agent orchestration and the adoption of lightweight, self-hosted LLMs.

Supplementary Materials Availability Statements

All datasets used in our experiments are publicly available or accessible under minimal conditions. Specifically, SQuAD 2.0, UIT-ViQuAD 2.0 (via the VLSP 2021 - ViMRC Challenge), and HotpotQA are freely accessible online. Access to VIMQA requires signing a user agreement and contacting the dataset maintainers. Code for baseline systems, including NodeRAG (Xu et al., 2025), HippoRAG2(Gutiérrez et al., 2025), and MiniRAG (Fan et al., 2025), was obtained from their official repositories using the latest versions available as of July 1, 2025. The UniQA dataset, its accompanying academic document collection, and the full implementation of URASys are released at `https://anonymous.4open.science/r/URASys/`, **including the ambiguous subsets** used in evaluation.

References

Katherine A. Batterton and Kimberly N. Hale. 2017. The Likert Scale What It Is and How To Use It. *Phalanx*, 50(2):32–39.

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, page 758–759, New York, NY, USA. Association for Computing Machinery.

Yang Deng, Lizi Liao, Wenqiang Lei, Grace Hui Yang, Wai Lam, and Tat-Seng Chua. 2025. Proactive Conversational AI: A Comprehensive Survey of Advancements and Opportunities. *ACM Trans. Inf. Syst.*, 43(3).

Yang Deng, Lizi Liao, Zhonghua Zheng, Grace Hui Yang, and Tat-Seng Chua. 2024. Towards Human-centered Proactive Conversational Agents. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 807–818, New York, NY, USA. Association for Computing Machinery.

Michelle Elizabeth, Morgan Veyret, Miguel Couceiro, Ondrej Dusek, and Lina M. Rojas Barahona. 2025. Exploring ReAct Prompting for Task-Oriented Dialogue: Insights and Shortcomings. In *Proceedings of the 15th International Workshop on Spoken Dialogue Systems Technology*, pages 143–153, Bilbao, Spain. Association for Computational Linguistics.

Tianyu Fan, Jingyuan Wang, Xubin Ren, and Chao Huang. 2025. MiniRAG: Towards Extremely Simple Retrieval-Augmented Generation.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 6491–6501, New York, NY, USA. Association for Computing Machinery.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. 2024. A survey on llm-as-a-judge.

Meiqi Guo, Mingda Zhang, Siva Reddy, and Malihe Alikhani. 2021. Abg-CoQA: Clarifying Ambiguity in Conversational Question Answering. In *3rd Conference on Automated Knowledge Base Construction*.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. 2024. LightRAG: Simple and Fast Retrieval-Augmented Generation.

Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From RAG to Memory: Non-Parametric Continual Learning for Large Language Models.

Uday Kamath, Kevin Keenan, Garrett Somers, and Sarah Sorenson. 2024. *Prompt-based Learning*, pages 83–133. Springer Nature Switzerland, Cham.

Khang Le, Hien Nguyen, Tung Le Thanh, and Minh Nguyen. 2022. VIMQA: A Vietnamese Dataset for Advanced Reasoning and Explainable Multi-hop Question Answering. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6521–6529, Marseille, France. European Language Resources Association.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Junyi Li, Jie Chen, Ruiyang Ren, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024. The Dawn After the Dark: An Empirical Study on Factuality Hallucination in Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10879–10899, Bangkok, Thailand. Association for Computational Linguistics.

Hao Liu, Zhengren Wang, Xi Chen, Zhiyu Li, Feiyu Xiong, Qinhan Yu, and Wentao Zhang. 2025. HopRAG: Multi-Hop Reasoning for Logic-Aware Retrieval-Augmented Generation.

Kiet Nguyen, Son Quoc Tran, Luan Thanh Nguyen, Tin Van Huynh, Son Thanh Luu, and Ngan Luu-Thuy Nguyen. 2022. VLSP 2021 - ViMRC Challenge: Vietnamese Machine Reading Comprehension. *VNU Journal of Science: Computer Science and Communication Engineering*, 38(2).

Long S. T. Nguyen and Tho T. Quan. 2025. URAG: Implementing a Unified Hybrid RAG for Precise Answers in University Admission Chatbots – A Case Study at HCMUT. In *Information and Communication Technology*, pages 82–93, Singapore. Springer Nature Singapore.

Quoc-Hung Pham, Huu-Loi Le, Minh Dang Nhat, Khang Tran T., Manh Tran-Tien, Viet-Hung Dang, Huy-The Vu, Minh-Tien Nguyen, and Xuan-Hieu Phan. 2024. Towards Vietnamese Question and Answer Generation: An Empirical Study. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 23(9).

Hossein A. Rahmani, Xi Wang, Yue Feng, Qiang Zhang, Emine Yilmaz, and Aldo Lipani. 2023. A Survey on Asking Clarification Questions Datasets in Conversational Systems. In *The 61st Annual Meeting of the Association for Computational Linguistics*.

Mohaimenul Azam Khan Raiaan, Md. Saddam Hossain Mukta, Kaniz Fatema, Nur Mohammad Fahad, Sadman Sakib, Most Marufatul Jannat Mim, Jubaer Ahmad, Mohammed Eunus Ali, and Sami Azam. 2024. A Review on Large Language Models: Architectures, Applications, Taxonomies, Open Issues and Challenges. *IEEE Access*, 12:26839–26874.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Zafaryab Rasool, Stefanus Kurniawan, Sherwin Balugo, Scott Barnett, Rajesh Vasa, Courtney Chesser, Benjamin M. Hampstead, Sylvie Belleville, Kon Mouzakis, and Alex Bahar-Fuchs. 2024. Evaluating LLMs on document-based QA: Exact answer selection and numerical extraction using CogTale dataset. *Natural Language Processing Journal*, 8:100083.

Hendrik Schuff, Heike Adel, and Ngoc Thang Vu. 2020. F1 is Not Enough! Models and Evaluation Towards User-Centered Explainable Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7076–7095, Online. Association for Computational Linguistics.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Nethra Viswanathan, Sofia Meacham, and Festus Fatai Adedoyin. 2022. Enhancement of online education system by using a multi-agent approach. *Computers and Education: Artificial Intelligence*, 3:100057.

Bing Wang, Yan Gao, Zhoujun Li, and Jian-Guang Lou. 2023. Know What I don't Know: Handling Ambiguous and Unknown Questions for Text-to-SQL. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5701–5714, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*.

Tianyang Xu, Haojie Zheng, Chengze Li, Haoxiang Chen, Yixin Liu, Ruoxi Chen, and Lichao Sun. 2025. NodeRAG: Structuring Graph-based RAG with Heterogeneous Nodes.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

A Dataset Statistics

To better characterize the datasets used in our experiments, we report descriptive statistics on context documents and evaluation queries. Table 5 presents word-level statistics, including the number of context documents, the maximum, minimum, and average context length, the number of evaluation queries, and the number of unanswerable queries where applicable. The datasets vary widely in both context structure and query types. UniQA, derived from real-world educational content, contains significantly longer passages, with an average length over 1,200 words and a maximum exceeding 5,000. URASys maintains strong performance under these conditions, suggesting that it handles long-form, high-complexity contexts effectively.

B Multi-hop Performance Breakdown

To evaluate performance under varying reasoning demands, we conduct a stratified analysis on HotpotQA and VIMQA by grouping questions according to reasoning hops. Table 6 presents accuracy across these levels.

URASys performs strongly across all categories, particularly on 2-hop and 3-hop questions where it leads all baselines. For example, it achieves 0.89 and 0.94 on 2-hop and 3-hop HotpotQA respectively, compared to 0.86 and 0.82 from the next best. This aligns with Table 3, where URASys averages two thoughts per answer, indicating structured reasoning supported by query decomposition.

Traditional RAGs also perform reasonably well on higher-hop queries. Dense RAG reaches 0.88 on 4-hop HotpotQA, BM25 RAG reaches 0.90 on 5+ hop VIMQA, suggesting that strong retrieval via dense embeddings or lexical overlap can sometimes resolve complex questions without explicit reasoning. However, these methods often struggle when deeper inference or cross-document synthesis is required. NodeRAG and HippoRAG 2 show competitive results at selected depths. For instance, NodeRAG reaches 0.89 on both 2-hop and 4-hop VIMQA, while HippoRAG 2 peaks at 0.86 on 5+ hop HotpotQA. These results highlight the potential of graph-based indexing and memory-augmented retrieval, though their performance remains inconsistent across datasets. In contrast, URASys maintains robustness across depths by combining retrieval specialization with adaptive reasoning, and applies ask-before-answer to avoid speculation when evidence is incomplete.

C Details of Human Evaluation

C.1 Procedure and Evaluation Criteria

To complement the LLM-as-a-Judge protocol, we conducted a human evaluation involving 20 target end-users, including *10 university freshmen* (G1) and *10 high school seniors* (G2). These participants were randomly recruited from admission-related events and information sessions hosted by our university. Each participant received a brief introduction to URASys and was invited to interact freely with the system by asking 20 questions of personal interest, as long as the topics fell within the scope of the system's indexed data. Each system response was evaluated along nine dimensions, as follows.

- **Number of Thoughts (NoT):** The number of intermediate reasoning steps generated before reaching a final answer. Higher values often indicate more structured or multi-step reasoning.
- **Accuracy:** Whether the final answer is factually correct, assessed using a binary label indicating *Correct* or *Incorrect*.
- **User Experience (UX):** Overall satisfaction with the system's interface, clarity of output, and ease of interaction.
- **Explanation Quality:** The clarity, coherence, and usefulness of the accompanying explanation, especially in helping users understand the rationale behind the answer.
- **Factuality:** The extent to which the explanation contains accurate and verifiable information supported by retrieved evidence.
- **Completeness:** Whether the system's output fully addresses the user's question without omitting important aspects.
- **Fluency:** The grammatical correctness and naturalness of the language used.
- **Relevance:** How directly the answer and explanation pertain to the original question, avoiding irrelevant or off-topic content.

Table 5: Word-level context statistics and evaluation query composition across QA datasets.

Metric	SQuAD 2.0	UIT-ViQuAD 2.0	HotpotQA	VIMQA	UniQA
Context document count	46	7	996	1000	42
Maximum context length	259	613	2075	676	5153
Minimum context length	27	99	103	8	298
Mean context length	91.1	182.1	972.4	264.7	1266.2
Unanswerable queries	492	200	–	–	38
Evaluation query count	1000	1000	1000	1000	574

Table 6: Answer accuracy scores on HotpotQA and VIMQA, stratified by reasoning depth. Best scores are in **bold**; second-best are underlined.

Method	HotpotQA				VIMQA			
	2-hop	3-hop	4-hop	5+ hop	2-hop	3-hop	4-hop	5+ hop
Sample count	550	300	121	29	500	300	150	50
() Naive RAG (Dense)	0.84	0.82	**0.88**	0.79	0.80	**0.90**	0.68	0.88
Naive RAG (BM25)	0.84	0.81	**0.88**	0.79	0.83	0.85	0.82	**0.90**
Naive RAG (Hybrid)	0.81	0.78	0.85	0.76	0.88	0.84	0.86	0.88
IRCot	0.86	0.80	0.81	0.76	0.89	0.79	**0.89**	0.60
LightRAG	0.86	0.80	0.81	0.76	0.89	0.79	**0.89**	0.60
MiniRAG	0.86	0.80	0.81	0.76	0.89	0.79	**0.89**	0.60
NodeRAG	0.86	0.80	0.81	0.76	0.89	0.79	**0.89**	0.60
HippoRAG 2	0.83	0.79	0.82	**0.86**	0.71	0.70	0.63	0.40
MiniRAG	0.69	0.47	0.40	0.50	0.88	0.45	0.47	0.60
URASys (Ours)	**0.89**	**0.94**	0.80	0.75	**0.91**	0.64	0.60	0.80

- **Trust:** The participant's confidence in the system's response, influenced by tone, coherence, and evidential grounding.

C.2 Summary of Results

In addition to Table 3, which reports scores on Accuracy, NoT, Explanation Quality, and Trust, we present the remaining human evaluation results in Table 7. These scores reflect average ratings from each user group on a 5-point Likert scale (↑).

G1 reported higher satisfaction in terms of user experience (4.61) compared to G2 (3.73), likely because university freshmen are more accustomed to navigating institutional systems, whereas high school seniors may be more cautious due to the importance of the information for their university decisions or their familiarity with traditional advising. Both groups gave strong ratings for fluency and factuality, indicating that URASys responses were generally clear, coherent, and grounded in reliable evidence. However, completeness received lower scores (3.45 and 3.55), possibly because the system prompts for clarification when facing vague queries instead of guessing, which, while improving factual accuracy, can make the final answer feel unresolved. This trade-off may also affect perceived relevance, reflected in G1's lower relevance score (4.03) compared to G2 (4.75). Still, trust and accuracy remain high across both groups, confirming that URASys meets its core objective of delivering reliable, well-supported answers in educational settings.

The results suggest that different user groups may prioritize different aspects of system behavior, such as interaction flow versus completeness, depending on their background and expectations.

Table 7: Average human evaluation scores by user group on five additional dimensions.

Group	UX	Factuality	Completeness	Fluency	Relevance
G1	4.61	4.34	3.45	4.96	4.03
G2	3.73	4.47	3.55	4.86	4.75

Table 8: Comparison of state-of-the-art (SOTA) methods on the UniQA dataset for university admission counseling. The results are reported on three subsets: overall accuracy, unanswerable questions (Unans.), and ambious questions (Amans.).

Method	Overall	Unans.	Amans.
GPT-4o	49.3	37.1	18.2
Gemini-2.5-pro	24.05	25.8	12.1
Claude-3.7-Sonnet	37.8	16.9	15.4
URASys (Ours)	**84.1**	**82.6**	**81.2**