RAGTrace: Understanding and Refining Retrieval-Generation Dynamics in Retrieval-Augmented Generation

Sizhe Cheng*
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, Guangdong, China
chengsz2021@mail.sustech.edu.cn

Jiaping Li*
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, Guangdong, China
lijp2024@mail.sustech.edu.cn

Huanchen Wang
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, Guangdong, China
Department of Computer Science
City University of Hong Kong
Hong Kong, China
wanghc2022@mail.sustech.edu.cn

Yuxin Ma†
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, Guangdong, China
mayx@sustech.edu.cn

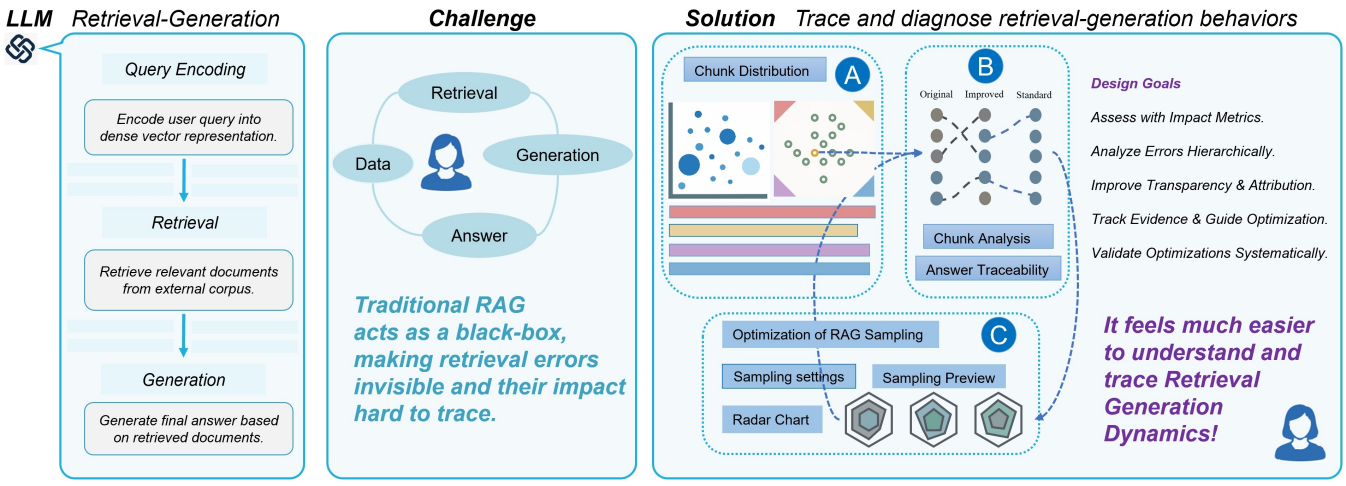

Figure 1: RAGTrace supports interactive refinement of retrieval-augmented generation (RAG) workflows. Users can move beyond passively receiving answers to actively analyzing retrieval-generation dynamics, diagnosing retrieval errors, and steering retrieval strategies for better performance.

Abstract

Retrieval-Augmented Generation (RAG) systems have emerged as a promising solution to enhance large language models (LLMs) by integrating external knowledge retrieval with generative capabilities. While significant advancements have been made in improving retrieval accuracy and response quality, a critical challenge remains that the internal knowledge integration and retrieval-generation interactions in RAG workflows are largely opaque. This paper introduces RAGTrace, an interactive evaluation system designed to analyze retrieval and generation dynamics in RAG-based workflows. Informed by a comprehensive literature review and expert interviews, the system supports a multi-level analysis approach, ranging from high-level performance evaluation to fine-grained examination of retrieval relevance, generation fidelity, and cross-component interactions. Unlike conventional evaluation practices that focus on isolated retrieval or generation quality assessments, RAGTrace enables an integrated exploration of retrieval-generation relationships, allowing users to trace knowledge sources and identify potential failure cases. The system's workflow allows users to build, evaluate, and iterate on retrieval processes tailored to their specific domains of interest. The effectiveness of the system is demonstrated through case studies and expert evaluations on real-world RAG applications.

Keywords

Retrieval-Augmented Generation, Knowledge Tracking, Evaluation

*Equal contribution.
†Corresponding author.

1 Introduction

As data-driven applications proliferate across AI, scientific computing, and business analytics, the rapid advancement of large language models (LLMs) further amplifies the need for robust evaluation mechanisms to ensure the effectiveness and reliability of visual representations in increasingly complex and AI-driven workflows [32, 35]. Despite these impressive achievements, LLMs still face notable challenges in complex, knowledge-intensive tasks that require real-time access to domain-specific information [21, 58]. Due to their reliance on static training data, LLMs often struggle with retrieving up-to-date or specialized knowledge [45], leading to outdated or inaccurate responses. To address this limitation, the Retrieval-Augmented Generation (RAG) paradigm has been introduced, integrating external knowledge sources with generative processes to enable the dynamic retrieval of relevant information before generating model responses [39, 57, 67]. Such hybrid approach enhances the factual accuracy, relevance, and overall utility of the generated content, making it particularly valuable for applications in scientific research, business intelligence, and legal documentation [51].

Despite the significant advantages of RAG over conventional LLMs, a critical gap remains in the systematic evaluation of RAG-based systems [61]. Existing research mainly focuses on optimizing retrieval mechanisms and model architectures, yet comprehensive evaluation methodologies remain underdeveloped [42, 49]. Specifically, current approaches often assess retrieval precision and generation quality separately [52], lacking an integrated system to holistically analyze RAG workflows. Although existing automated evaluation methods primarily emphasize performance metrics and output quality, they often do not explain why RAG behave in certain ways or how to diagnose underlying issues in the retrieval-generation process [20, 55]. Meanwhile, recent advances in the visualization community, aimed at improving model interpretability and shedding light on AI behavior through visual analytics approaches [41, 61, 64], underscore the effectiveness of visualization-driven techniques for model evaluation and debugging. Furthermore, the interaction between RAG systems and visualization platforms remains underexplored, despite their crucial role in interpreting AI-generated knowledge and enhancing model transparency for understanding model behavior [64]. Visualization is important in interpreting complex data and supporting decision-making [41], yet the absence of a structured evaluation paradigm hinders researchers and practitioners from systematically assessing how well RAG systems retrieve, integrate, and present knowledge. Without a unified evaluation scheme, it remains challenging to refine RAG models for real-world applications and to build user trust through interactive visualization.

The last decade has seen the community successfully develop evaluation systems to assess AI-driven decision-making systems, ensuring their reliability and interpretability. We believe that the same paradigm is equally relevant to evaluating RAG workflows. Accordingly, this paper introduces an interactive visual evaluation system, RAGTrace, designed to facilitate the comprehensive evaluation of RAG workflows by analyzing retrieval quality and generative coherence. Drawing from an extensive review of relevant literature and collaboration with domain experts in RAG, various analytical tasks have been derived to inform the visualization and interaction design in a hierarchical manner. By employing the "overview+detail" scheme, RAGTrace utilizes a multi-level design that exposes the performance of RAG workflows from three perspectives: the analysis of retrieval relevance and redundancy, evaluation of the quality and consistency of generated outputs, and in-depth tracking of error patterns and propagation across retrieval and generation steps. The effectiveness of RAGTrace is demonstrated through case studies on real-world datasets and expert interviews, showcasing its capability to enhance interpretability, diagnose retrieval-generation inconsistencies, and support the optimization of RAG workflows.

In summary, our contributions include:

(1) **A formative study (N=12)** that captures, through conversational insights, the practices, challenges, and expectations of users in diagnosing, and optimizing RAG outputs.

(2) **A novel diagnostic methodology** comprising six quantitative and qualitative metrics organized into two categories that systematically evaluate retrieval precision, knowledge transfer efficiency, and generation quality in RAG workflows.

(3) **RAGTrace**, a comprehensive interactive evaluation system integrating the metrics to analyze, evaluate, and optimize retrieval-generation interactions, enabling iterative refinement of RAG workflows.

(4) **Empirical validation through user studies and expert interviews**, demonstrating how RAGTrace helps diagnose retrieval inefficiencies, debug generation errors, and refine retrieval strategies for improved performance.

2 Background and Related Work

This section reviews existing literature on evaluation methods for LLM and RAG, highlighting current advancements in evaluation analytics for retrieval processes, generation explainability, and model debugging.

2.1 Core Process of RAG

The RAG architecture builds upon a structured pipeline of indexing, retrieval, and generation, which together facilitate the efficient transformation of raw data into contextually grounded outputs [6, 22, 67].

- **Indexing.** Obtain data from dataset and create an index. Specifically, the construction of the data index includes the following steps: 1) Data Preprocessing: Converting raw datasets into a structured text format suitable for retrieval; 2) Chunking: Dividing text into smaller segments [14] to accommodate the context length limitations of LLMs; 3) Embedding and Indexing: Encoding text chunks into vector representations using an embedding model. The resulting vectors, along with their corresponding text, are stored in an index for fast similarity-based retrieval.
- **Retrieval.** Given a user query, the system encodes it into a vector and computes similarity scores against indexed document embeddings. The top-K most relevant chunks are selected as context for the response generation phase.
- **Generation.** The retrieved context is concatenated with the query and passed to an LLM. Depending on task requirements,

the model may either rely solely on retrieved information or incorporate prior knowledge when formulating its response.

While these three stages form the foundation of RAG systems, evaluating their effectiveness presents unique challenges, especially in assessing retrieval quality and the impact of retrieved context on the generated response.

2.2 Evaluation for Retrieval-Augmented Generation

The widespread adoption of LLMs across various fields has made the evaluation and analysis of their performance increasingly crucial [10, 62, 71]. Traditional evaluation methods for LLMs primarily rely on automated metrics, such as BLEU [43], ROUGE [1], and human annotations, including expert subjective ratings of generated text [8, 60]. While these approaches assess output quality, they fail to provide deep insights into the underlying mechanisms and decision-making processes of LLMs [**?**]. Moreover, due to the inherent "black-box" nature of these models [26], their decision paths remain opaque, making it challenging to explain their behaviors comprehensively.

These challenges become even more pronounced in RAG systems, which couple retrieval and generation modules and thus introduce new layers of complexity to evaluation. While visualization techniques such as causal analysis [50] and uncertainty heatmaps [9] can trace error propagation, most tools focus on specific errors rather than providing a comprehensive evaluation environment. Recent automated evaluation frameworks, including RAGAS [20], ARES [48], and RAGProbe [55], have streamlined RAG workflows assessment through multi-metric integration. However, these frameworks primarily emphasize metric aggregation, lacking actionable insights for system improvement and interactive visualization for human-in-the-loop RAG optimization. Additionally, most current approaches evaluate retrieval and generation separately [5, 37], overlooking their interdependencies. These limitations highlight the need for evaluation tools that not only provide accurate performance metrics but also support transparent, interpretable, and interactive diagnostics across the entire RAG workflow.

2.3 Interactive Visualization for Evaluating Text Retrieval and Generation

Interactive visualization has become a vital tool in LLM research to enhance model interpretability, error analysis, and fine-tuning [3, 34], which has significantly improved the usability and transparency of LLMs [30, 53]. Several interactive systems aid in understanding LLM behaviors from different perspectives. RELIC [13] evaluates LLM response reliability through self-consistency analysis, allowing users to generate multiple responses and visually identify inconsistencies, particularly hallucinations. The iScore system [15] helps interpret automated scoring of summaries in education, offering parallel visualizations of student responses and model-assigned scores. These interactive methods have made significant contributions to improving LLM interpretability. However, they are typically tailored for specific tasks (e.g., summarization, consistency checking) or particular models (e.g., ChatGPT self-consistency), which limits their applicability across different contexts. More comprehensive tools such as LLM Comparator [29, 30], HaLLMark [27], and ChainForge [4] offer broader evaluation capabilities by comparing outputs from multiple LLMs, while systems like Aletheia [23] and WaitGPT [63] focus on explaining the generation process itself. Despite these advancements, most existing interactive methods analyze individual models in isolation and lack the ability to provide systematic interactive visualizations for complex architectures like RAG, which combine both retrieval and generation processes.

Traditional information retrieval research has established foundational methodologies for understanding user needs and system performance. Ellis et al. [19] proposed a behavioral approach to information retrieval system design, emphasizing the importance of understanding user search patterns and information-seeking behaviors. Building on this behavioral foundation, Chaudhuri et al. [11] introduced hypothetical analysis techniques that enable quantitative impact assessment of system modifications. As interactive requirements grew to better adapt retrieval systems to user needs [28], Belkin et al. [7] designed experimental interfaces supporting user interaction by integrating cognitive models with retrieval strategies, while Ahn et al. [2] proposed adaptive visualization techniques to enhance exploratory information retrieval. These foundational approaches remain relevant in contemporary retrieval-augmented systems, where integrating retrieval with generative models introduces challenges that traditional evaluation methods cannot fully address [72]. Issues like query-document mismatches, vocabulary gaps, and semantic ambiguity [16] often lead to error propagation in RAG systems, necessitating advanced analysis tools.

Interactive visualization systems help tackle key challenges at different stages of the retrieval-generation pipeline, including retrieval inaccuracies and the opacity of black-box models. For query formulation, LinkQ [36] provides visual tools to refine queries for knowledge graph question answering. In retrieval evaluation, Angler [46] uses coordinated visualizations to detect errors and guide iterative improvements. DeepLens [56] identifies out-of-distribution retrievals that may lead to unreliable generation, while BERT-based similarity mappings [18] compare retrieved texts and generated outputs to highlight semantic misalignments. VEQA [54] visually explains BERT-based retrieval systems, aiding model interpretability.

Overall, RAGTrace bridges this gap by integrating visualizations of retrieval distributions, semantic similarity mappings, and content attribution. Unlike specialized tools, RAGTrace enables cross-stage error detection, improving transparency and robustness in retrieval-augmented systems.

3 FORMATIVE INTERVIEWS

We conducted formative interviews with domain experts and LLM researchers, aiming to understand the black-box nature of RAG systems and uncover unmet evaluation needs.

3.1 Participants and Procedure

We recruited 12 experts through forum posts and word-of-mouth recommendations, consisting of researchers focused on LLMs and RAG (n=4) and domain experts from various fields who frequently use large models (n=8). The participant pool included 8 males and 4 females. 7 participants had over 2 years of experience with LLM,

while the remaining five had at least 1 year of experience. Each interview lasted between 30 minutes and 1 hour, and participants were compensated with local currency equivalent to $15 for their time. Detailed demographic information and background of all participants are provided in Appendix A.

Our participants, labeled E1 through E12, represented diverse expertise backgrounds. E1 to E4 were researchers specializing in LLM with deep understanding of RAG workflows and extensive experience in evaluating the performance of both LLMs and RAG applications. E5 through E12 were domain experts from various fields, each with significant experience using LLMs to address specialized knowledge requirements in their respective domains.

The interviews were designed to uncover practical pain points and research gaps. We began by asking participants about their experiences using and evaluating LLM and RAG workflows, including the strategies they employ to assess system outputs. Next, we explored the limitations of existing evaluation methodologies and the specific challenges they face when interpreting or improving RAG-generated responses. Finally, we discussed their expectations for an ideal evaluation system, particularly in terms of automation, interpretability, and reliability.

Our interviews followed a semi-structured format divided into three main sections. In the first section, we collected demographic information (gender, occupation) and background data on participants' experience with language models, including preferred models and usage patterns. We then explored their specific use cases, asking questions like "In which scenarios do you primarily use large language models?" and "How frequently do you interact with these systems?" The second section focused on core research questions regarding RAG systems, including participants' current understanding of RAG technology, their evaluation practices, and their most pressing optimization needs, particularly the improvement of overall performance in users' specific domains. We inquired about their preferences between holistic versus component-based evaluation approaches and their requirements for interactive evaluation tools. In the final section, participants discussed their expectations for error tracing, automatic optimization across different usage scenarios, and long-term improvement strategies for RAG systems.

During the interviews, we employed contextual inquiry techniques by asking participants to demonstrate their typical workflows when using or evaluating RAG systems. These demonstrations provided valuable insights into practical challenges and workarounds that might not emerge through verbal descriptions alone. All sessions were recorded with permission, transcribed, and subsequently analyzed through qualitative coding. We independently coded the transcripts using an open coding approach, then collaboratively developed a codebook through multiple discussion rounds. The resulting themes were organized into the system presented in our findings section.

3.2 Findings

Based on the interviews, we identified several recurring themes that highlight key challenges in RAG evaluation. These findings are categorized into the following dimensions:

3.2.1 The Critical Need for Domain-Grounded RAG Solutions. Domain expertise integration emerged as a critical requirement, with general-purpose RAG systems failing to meet specialized knowledge needs. Notably, after we provided detailed explanations about RAG to our domain experts, 7 out of 8 expressed strong interest in deploying RAG to enhance knowledge reliability in their respective fields.

- **Knowledge Foundations as a Countermeasure to Model Hallucination.** Hallucination emerged as a pervasive concern among experts working with language models, particularly when applying these systems to specialized domains. The integration of reliable knowledge retrieval serves as a critical anchor to constrain model outputs within factual boundaries. As E3 noted, "the biggest problem I usually encounter with RAG algorithms is hallucination, sometimes with repetitive answers." This phenomenon creates particular challenges in professional contexts where accurate information is paramount. Experts consistently emphasized how retrieval mechanisms must not only fetch information but verify its accuracy before incorporation into responses. Interestingly, even with retrieval-augmentation, models sometimes misinterpret or ignore retrieved information, suggesting that effective hallucination mitigation requires both improved retrieval precision and enhanced reasoning about retrieved content. The interviews revealed a nuanced understanding that hallucinations manifest differently across knowledge domains, requiring tailored detection and prevention strategies rather than one-size-fits-all approaches.
- **Specialized Knowledge Ecosystems Demand Bespoke Retrieval Systems.** Our interviews revealed a fundamental tension between general-purpose RAG systems and the specialized knowledge requirements of domain experts. Complex fields such as healthcare, engineering, and scientific research operate with distinct epistemological systems that standard retrieval approaches often fail to adequately capture. E5, working in nuclear engineering, articulated how "different domains have different focuses" necessitating "diversified output methods to address different emphases." For instance, nuclear engineering demands high precision in technical documents and zero tolerance for factual errors which could lead to safety risks, while healthcare prioritizes robust prompt fragility and evidence-supported generation anomaly detection. This insight extends beyond simple keyword matching to encompass domain-specific credibility assessment, contextual relevance, and hierarchical knowledge organization. Particularly revealing was E4's observation regarding the importance of "credibility/importance of search materials," highlighting how domain expertise inherently involves judgments about informational authority that must be encoded within RAG systems. These findings suggest that truly effective domain-specific RAG systems require not merely access to specialized content, but architectures that embody the evaluation standards and knowledge structures unique to each field of expertise.

3.2.2 Transparency and Evaluation Challenges in RAG Systems. The "black box" nature of current RAG systems creates substantial barriers to effective assessment and improvement.

- **Information Pathway Visibility: The Black Box Problem in RAG Processes.** A fundamental challenge identified across expert interviews concerns the opacity of information flow between

retrieval and generation components. Unlike traditional search systems that present results separately from summaries, RAG systems obscure the relationship between retrieved information and generated responses, creating what several experts described as a "black box" problem. E1 highlighted how "large models often cannot judge the quality of retrieval results," suggesting that failures may cascade undetected through the system. This opacity creates significant diagnostic challenges - when faced with inaccurate outputs, experts struggle to determine whether the retrieval component failed to access relevant information, or if the generation component misinterpreted correctly retrieved content. Our interviews revealed that 6 of 7 domain experts lacked a clear understanding of how RAG retrieves knowledge, and all these experts believed that understanding this mechanism would be meaningful for increasing their trust in model outputs. This lack of transparency fundamentally undermines trust and adoption, especially in high-stakes domains.

- **Granular Assessment Infrastructure: Beyond Binary Evaluation Paradigms.** Our interviews uncovered a critical gap in evaluation methodology - the lack of nuanced, multi-dimensional assessment systems tailored to the complex nature of RAG systems. Current evaluation approaches tend toward oversimplified metrics that fail to capture the intricate interplay between retrieval accuracy, information relevance, response coherence, and factual correctness. E5's experience of having "no relatively quantified feedback" highlights the reliance on rudimentary methods that are "very inefficient and cannot provide quantitative results." This limitation forces experts to develop ad-hoc evaluation scripts, as noted by E1, or resort to crude comparative techniques. All participants reported experiencing hallucinations in LLMs, with these fabrications potentially undermining trust in model outputs. E3 specifically mentioned encountering "numerical issues, such as when data mentions hazardous area classifications, the model responds with strange numbers." Notably, 10 of 12 participants (E1, E4-E12) stated that hallucination problems lack effective workflows for improvement, making evaluation and enhancement challenging. E5 explained, "I usually just manually read or check, or use different models to repeat the same work to check result similarity. Both methods are very inefficient and cannot provide quantitative results." This sentiment was echoed by E11, highlighting how current systems lack efficient workflows for identifying and addressing output issues. These insights suggest that advancing RAG systems requires not just improved algorithms but fundamentally new evaluation infrastructures that can decompose performance across multiple dimensions while adapting to the specific requirements of diverse application contexts.

3.2.3 Iterative Development and Collaborative Refinement. Participants identified unstructured development processes as a key barrier to effective RAG system improvement.

- **Systematic Performance Evolution: Mapping the Trajectory of System Improvements.** The development of effective RAG systems emerges from our interviews as fundamentally iterative, yet current practices lack structured methodologies for tracking performance evolution across system modifications. Experts described ad-hoc improvement processes characterized by reactive problem-solving rather than systematic optimization. E1's admission of not having "a systematic iteration method" where they "discover a problem and solve it" reflects a broader pattern of unstructured development processes. This approach creates significant inefficiencies, as improvements in one aspect often produce regressions in others without clear visibility into these trade-offs. E7 noted that "the biggest challenge is likely the inability to pinpoint problems. Different users need models to solve different problems, so it may be very difficult to develop a systematic iteration method." E7 further emphasized the need for "automatic analysis of problem patterns" and "specific case analysis," explaining that "this would be more practical than generalized metrics because it could help developers identify systemic issues rather than addressing symptoms individually." These findings suggest that advancing RAG systems requires not just better components but more sophisticated development methodologies that can track multidimensional performance metrics across iterations, enabling developers to understand the holistic impact of incremental changes rather than focusing on isolated improvements.
- **Collaborative Visual Analysis: Enabling Multi-stakeholder System Refinement.** Our interviews revealed strong consensus around the need for interactive visualization tools that transform RAG development from an opaque technical process to a collaborative analytical activity involving diverse stakeholders. Experts consistently expressed enthusiasm for interfaces that would allow real-time parameter adjustment and immediate performance feedback, with E5 noting how such tools "would significantly improve my efficiency" for "rapid iteration and optimization." Beyond efficiency gains, these visualization needs reflect a deeper requirement for shared understanding between technical developers and domain experts who may lack programming expertise but possess crucial knowledge about information quality and relevance. E2's anticipation that such tools would be "very meaningful" highlights the current absence of accessible interfaces for RAG optimization. E11, from a domain expert perspective, specifically requested "one-click parameter adjustment functionality for specific optimizations (such as balancing false positives and false negatives)," emphasizing how domain-specific RAG evaluation requires tailored workflow aligned with specialized knowledge requirements. E9 underscored this point by stating that "interactive models significantly improve efficiency for trial-and-error and iterative adjustments," suggesting that even for users with strong technical backgrounds, the ability to rapidly test hypotheses and observe outcomes remains a critical productivity enhancement. The interviews suggest that effective visualization must go beyond simplistic dashboards to reveal the relationship between system parameters, retrieval patterns, and generation outcomes, enabling intuitive exploration of the complex trade-offs inherent in RAG system design while accommodating stakeholders with varying technical backgrounds.

4 Design Goals

Based on the themes extracted from our formative interviews and literature review, we derived five design goals to support interpretable

and iterative evaluation of RAG workflows. The five design goals are as follows:

- **DG1: Leverage Impact Metrics for Initial Assessment.** The system should adopt high-level impact metrics that quantify the influence of individual information fragments on the generated outputs. These metrics can serve as early indicators of potential biases and guide further in-depth analysis.
- **DG2: Integrate Multi-Level Error Analysis.** To comprehensively diagnose failures, the system must combine global, local, detailed, and relational error analysis. Such an integrated approach will better identify key factors affecting the reliability of the outputs and support targeted refinements.
- **DG3: Enhance Transparency and Attribution.** In line with the need for traceability, the system should facilitate the inspection of the generation process by clearly attributing output components to specific retrieved documents or reasoning steps. This goal echoes the importance of explaining and justifying evaluations to support user trust and iterative improvements.
- **DG4: Support Evidence Tracking and Optimization.** The system should allow users to track evidence across different contexts, thereby linking core optimization factors with the underlying retrieval and generation processes. By relating these factors, designers can make more precise adjustments and improve overall result quality.
- **DG5: Enable Systematic Validation of Optimizations.** It is crucial to not only identify influencing factors but also to validate and quantify the improvements brought by different optimization strategies. A systematic set of performance metrics will ensure that adjustments are both applicable and generalizable.

These design goals collectively form the foundation for an evaluation system that is both scalable and interpretable, addressing the key challenges identified in our formative interviews.

5 User Interface

Based on these design goals, we propose RAGTrace, an interactive evaluation system for tracking and evaluating retrieval-generation dynamics in RAG workflows. By following DG2, the entire system employs multi-level error analysis by dividing the interface into three complementary components: (A) visual exploration for global trend analysis and local discrepancy detection, (B) retrieval performance analysis for evidence traceability, and (C) comparative optimization for systematic refinement across different RAG workflow stages.

5.1 Visual Exploration of Transparency and Attribution

In this component shown in Fig. 2 (A), a heatmap and Force-directed Graph metrics, along with corresponding visualization modules, are utilized to provide a detailed analysis of chunk distribution and question classification. These modules also offer comprehensive data regarding the questions to support in-depth understanding of answer generation after retrieval, thereby enhancing the transparency and traceability of the RAG workflow. Additionally, manual search capabilities for specific questions are provided to further explore the details associated with each question.

5.1.1 Heatmap for Chunks Distribution Analysis. Drawing from established visualization techniques for spatial data analysis, the chunk distribution is designed to show to key factors: chunk density and uniformity. In this module, the chunks from the knowledge base are encoded as small white dots positioned based on their semantic embeddings using dimensionality reduction (detailed in Section 6.2). Users can perceive the concentration of retrieved chunks through the background intensity in the heatmap; darker backgrounds indicate higher chunk density while uniform background color intensity suggests evenly distributed chunks. White points on the heatmap represent individual chunks, with a global limit of 20,000 chunks to prevent performance issues. When users hover over individual question or chunk, relevant document metadata is immediately displayed.

As illustrated in Fig. 2 (A1), the chunk distribution is displayed in a heatmap. The intensity of the background color denotes chunk density, with darker regions reflecting higher chunk concentrations. For questions, the yellow color highlights mark areas of interest, while blue represents less relevant or unimportant chunks. An interactive zooming feature enables detailed examination of specific chunk regions, with synchronized zooming across all visualizations to facilitate direct comparison.

The heatmap is divided into a grid of adjustable-sized square cells to facilitate granularity control based on their analytical needs. When a question is selected, the system automatically highlights the corresponding cell and performs a named entity extraction and thematic analysis on all chunks in the cell. This generates a concise set of topic keywords displayed in a dedicated summary view in the lower right corner.

5.1.2 Force-directed Graph Metrics for Question Analysis. Aligned with DG1, the Force-directed Graph utilizes composite performance metrics for node positioning and color coding, enabling rapid identification of high-impact failure patterns. Understanding the failure patterns in RAG is crucial for diagnosing retrieval effectiveness and enhancing generation quality. To achieve this, we utilize a Force-directed Graph visualization, Fig. 2 (A2), that maps questions based on their distance from an ideal retrieval and generation outcome.

The nodes in the Force-directed Graph represent four different question types, with each vertex applying an attractive force to question nodes that exhibit similar characteristics. The question nodes represent test questions filtered and selected through the search interface, Fig. 2 (A3), displaying only questions of interest to the user. Question nodes simultaneously repel each other to prevent overcrowding and maintain visual clarity. Users can hover over any node to display detailed evaluation information, including specific metrics and contextual details. The identified failure types are categorized as follows:

- **Retrieval Failure**: Highlighted in red, these problems arise when critical chunks are not retrieved, often due to excessive dispersion in retrieval results. High retrieval dispersion is visualized through cosine similarity distributions, where a wider spread indicates a lack of focus.
- **Prompt Vulnerability**: Highlighted in yellow, this category captures errors stemming from ambiguous prompts that lead to multiple interpretations. The distance between different possible

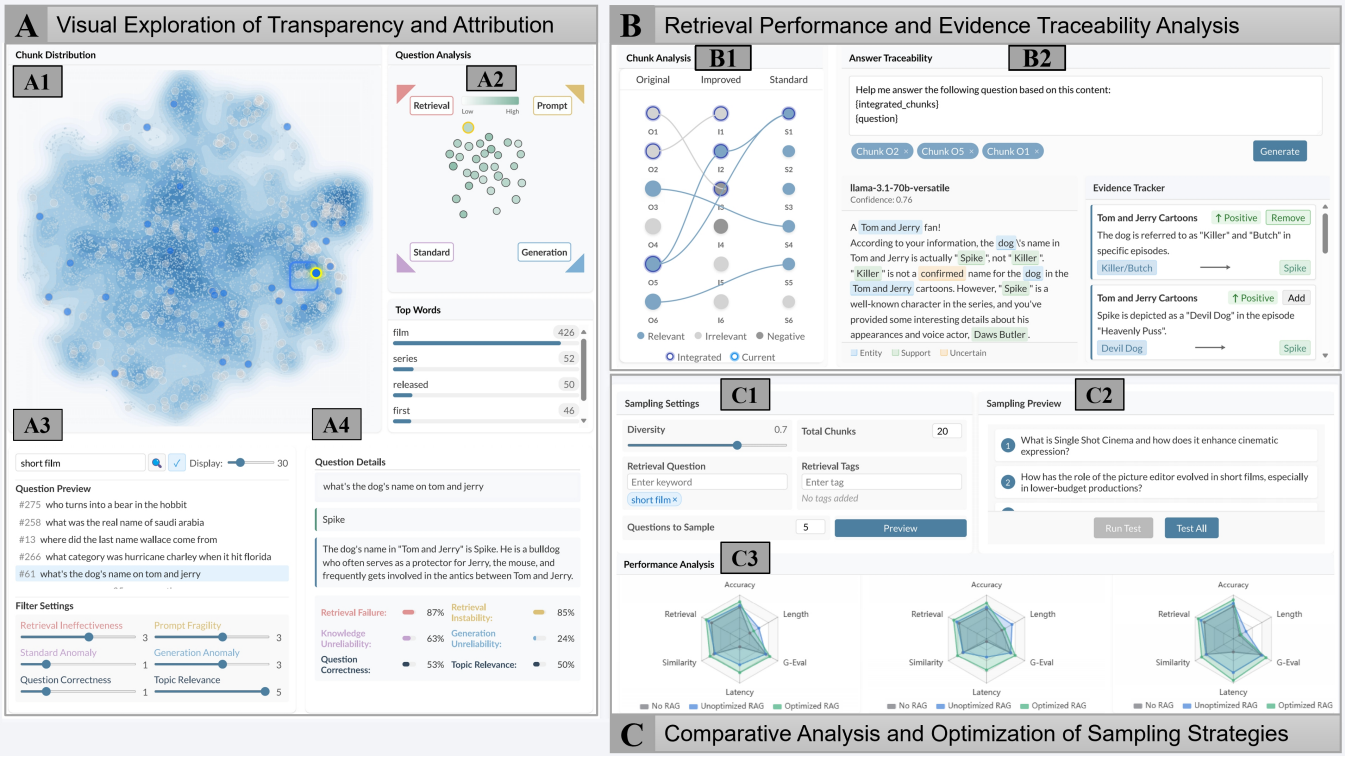

Figure 2: RAGTrace aims to support users refine RAG workflows by interactively analyzing retrieval-generation dynamics, comparing retrieval behaviors, evaluating answer reliability, and optimizing retrieval strategies. In RAGTrace, users examine retrieval behavior through visual analysis and performance metrics (A), evaluate retrieval quality and answer reliability by comparing different retrieval strategies (B), and optimize retrieval configurations by analyzing sampling distributions and performance changes (C).

responses serves as a measure of ambiguity, with larger distances indicating greater uncertainty.

- **Generation Anomalies**: Highlighted in blue, these errors occur during response generation and are characterized by deviations in content attribution. Metrics such as hallucination rates and reference errors (incorrectly citing retrieved chunks) are used to quantify these anomalies.
- **Standard Inconsistencies**: Highlighted in purple, these problems emerge from outdated or incorrect information sources, insufficient knowledge base coverage, or errors in the evaluation dataset. Outliers in this category include ambiguous questions or flawed reference answers.

The magnitude of each problem directly influences the strength of attraction exerted on corresponding question nodes, where more significant problems create stronger attractive forces and cause question nodes to cluster around the most problematic nodes. The repulsive forces between question nodes promote optimal spacing for interactive exploration, minimize visual clutter, and preserve the visibility of relationships.

5.1.3 Interest-driven Question Search. Providing manual search for related questions is essential for refining query exploration and improving retrieval accuracy. This module, Fig. 2 (A3) provides an interactive search interface for locating and filtering questions based on specific criteria. The interface includes:

- A search bar for direct query input;
- A suggested question list displaying example queries;
- A result ranking mechanism that sorts retrieved questions by similarity scores and hallucination values.

Our system supports four Granular Diagnostic Metrics (**Retrieval Failure Value**, **Prompt Fragility Value**, **Standard Hallucination Value**, **Generation Anomaly Value**) and two Composite Performance Metrics (**Question Correctness** based on BLEU/ROUGE, and **Topic Relevance** based on embeddings). Users can select preset weighting configurations (e.g., prioritizing high hallucination, similarity matching, or improvement potential) and adjust question display proportions via an integrated slider. The question preview pane is bidirectionally bound with nodes in the heatmap and the Force-directed Graph visualization, enabling synchronized highlighting and consistent information display when selecting elements. All questions found in the search interface are encoded as highlighted dots in the heatmap and as question nodes in the Force-directed Graph visualization, ensuring coordinated illustration across all three visualization modules.

5.1.4 Evaluation Indicators for Question Details. To provide an intuitive representation of question evaluation metrics, we employ a progress bar visualization, Fig. 2 (A4), to depict **Granular Diagnostic Metrics** used during the problem localization phase. These indicators are visualized as progress bars, each corresponding to one metric. The length of each bar represents the magnitude of the respective measure, while distinct colors help differentiate between them. This view displays the model-generated answer and the ground truth answer (i.e., the standard reference from the test dataset) side by side for direct comparison. It also provides key metadata, including Question ID, Question Type, and the number of Related Chunks retrieved, to support a comprehensive overview.

The module is dynamically linked with the other views. When a different question is selected in the heatmap or the Force-directed Graph view, the detailed view automatically updates its content to reflect the evaluation metrics, answers, and metadata for that newly selected question.

5.2 Retrieval Performance and Evidence Traceability Analysis

A comprehensive evaluation of retrieval quality and evidence traceability is crucial for assessing the reliability of generated responses, going beyond mere parameter metrics and statistical data. This component fulfills the requirements in DG3 by implementing comprehensive transparency and attribution mechanisms through both retrieval flow visualization and evidence traceability analysis, enabling clear mapping of outputs to their retrieval sources and systematic tracking of information provenance throughout the generation process. The features are fulfilled by adopting efficient comparative analysis of original, optimized, and reference retrievals.

5.2.1 Chunk-Relink Graph for Retrieval Flow Analysis. A comprehensive examination of retrieved document chunks is essential for understanding retrieval effectiveness and relevance distribution. This module enables comparative analysis across different retrieval strategies, i.e., original, optimized, and reference retrievals, which allows users to assess ranking shifts, content variations, and retrieval consistency within the main workspace, Fig. 2 (B1). By visualizing retrieval flow using the Chunk-Relink Graph diagram, how retrieved chunks evolve across different retrieval configurations can be highlighted.

- **Visual Encoding.** To directly convey retrieval relevance and facilitate comparative analysis, a categorical color scheme is employed to distinguish chunk significance and retrieval status:
  - Blue – Relevant chunks, strongly contributing to response generation;
  - Gray – Irrelevant chunks, providing little contextual information;
  - Dark Gray – Negative information chunks, which may introduce noise or inconsistencies;
  - Dark Blue rings – Integrated chunks that have been merged or aggregated from multiple retrieval sources;
  - Blue rings – Currently selected chunks under analysis or user interaction.

The size of each chunk node encodes its semantic similarity with the query, where larger nodes represent higher cosine similarity in the embedding space between the chunk and the query statement used for retrieval. Guided by DG4, the system implements a synchronized selection mechanism and maintains visual consistency across components. This enables evidence tracking and optimization through coordinated interactions. This visual cue provides an immediate indication of retrieval relevance independent of the color-based categorization. When identical chunks are retrieved through multiple retrieval methodologies, connecting lines are drawn between these nodes. In addition, selecting or interacting with any chunk automatically highlights all identical chunks across different retrieval strategies.

5.2.2 Generation Confidence and Evidence Traceability. Ensuring the reliability and traceability of generated answers is crucial for evaluating RAG models, as illustrated in Fig. 2 (B2). We employ a structured visualization approach that highlights confidence levels and evidence sources within responses.

- **Confidence-Based Annotation.** We use a categorical highlight scheme to indicate content reliability: blue for named entities, green for well-supported information, and orange for uncertain or weakly backed content. This visual differentiation helps users quickly assess response robustness.
- **Interactive Evidence Traceability.** Users can modify evidence sources in real time, triggering dynamic updates to highlights and enabling direct comparison of different evidence chunk sets used to generate the target answer. A directed graph visualization links named entities to supporting evidence chunks, offering an intuitive representation of information provenance.

5.3 Comparative Analysis and Optimization of RAG Sampling Strategies

Following DG5, we design a comprehensive resampling evaluation component that enables systematic validation through comparative analysis of different RAG configurations. Building upon the transparency analysis from previous components, such optimization scheme provides structured interfaces for parameter adjustment, real-time preview, and performance tracking across multiple retrieval strategies, Fig. 2 (C).

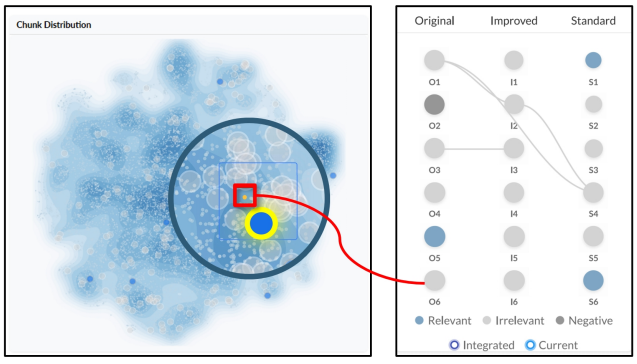

Figure 3: The small dots on the left correspond to the circular nodes in the Chunk-Relink Graph, both representing a chunk in the RAG knowledge base.

5.3.1 Sampling Settings. This module provides a structured interface for configuring key sampling parameters that directly influence retrieval diversity, response consistency, and overall generation quality (C1). The configurable parameters include: *Diversity* (controlling generation randomness to balance creativity and reliability), *Number of Chunks* (determining the context volume with trade-offs between information completeness and redundancy), *Retrieval Keywords and Tags* (refining retrieval focus through term prioritization and metadata filtering), and *Number of Questions to Sample* (defining evaluation scope with considerations for statistical robustness versus computational efficiency). As depicted in Fig. 2 (C1), these parameters are presented within an interactive configuration panel with real-time effects visualization and parameter sensitivity analysis for optimizing RAG strategies.

5.3.2 Sampling Preview. Building on established methods for evaluating RAG strategies, this module offers an direct preview of sampling outcomes, Fig. 2 (C2).

5.3.3 Radar Chart Visualization. As shown in Fig. 2 (C3), this module visualizes the performance of different configurations using Radar Charts, where each chart represents a single question. The charts are designed to simultaneously display the performance of the original configuration, RAG before optimization, and RAG after optimization.

- *Color-coded Stages:* In each Radar Chart, the performance metrics for the original configuration are depicted in gray, those for RAG before optimization in blue, and those for RAG after optimization in green, as a superposed manner.
- *Interactive Highlighting:* Users can click on any segment of a Radar Chart to view detailed numerical performance values, facilitating an in-depth analysis of specific metrics.

6 Implementation

In this section, we introduce the technical implementation details of RAGTrace, including our evaluation metrics and the data processing pipeline that powers our visualization and analysis capabilities.

6.1 Evaluation Metrics

To comprehensively assess RAG systems, we categorize the evaluation metrics into two complementary classes: **Composite Performance Metrics** for end-to-end performance measurement, and **Granular Diagnostic Metrics** for modular component analysis. While composite metrics like BLEU, ROUGE and GPTScore provide efficient, standardized evaluation of overall output quality, their reliance on system-level assessment often masks underlying failures in specific components of the RAG workflow [74].

To investigate the scientific validity and necessity of granular metrics, we conducted an extensive literature review examining failure modes in RAG systems. Recent literature has identified several key failure patterns that require targeted evaluation techniques:

- **Retrieval Failure** occurs when the retrieval module does not capture key evidence or exhibits poor result distribution. Ru et al. [47] introduce fact-level metrics like "Claim Recall" to measure factual coverage, while Saad-Falcon et al. [48] propose ARES to assess context relevance. Chen et al. [12] benchmark LLMs in RAG settings, highlighting retriever performance as a critical factor.

  To quantify retrieval failures, we define the Retrieval Failure Value ($\mathcal{R}_{fail}$), which evaluates both the hit rate on key document chunks and the entropy of the retrieval distribution:

$$\mathcal{R}_{fail} = \alpha \frac{\sum_{c_k \in C_{\text{gold}}} \mathcal{T}_\theta(\text{sim}(c_k, C_{\text{ret}}))}{|C_{\text{gold}}|} + \beta \frac{1}{n} \sum_{i=1}^{n} \text{Entropy}\big(\text{sim}(c_i, C_{\text{ret}})\big) \tag{1}$$

  where $\text{sim}(c_i, C_{\text{ret}})$ is the cosine similarity between chunk c_i and retrieved chunks, and C_{gold} is the set of ground truth chunks. The threshold function $\mathcal{T}_\theta(x)$ indicates retrieval success when similarity exceeds θ. Entropy is computed as $\text{Entropy}(X) = -\sum_{x \in X} p(x) \log p(x)$, capturing retrieval concentration.
- **Prompt Fragility** occurs when ambiguous prompts lead to inconsistent outputs. Zhang et al. [70] and Zhuo et al. [73] propose benchmarks and metrics to assess model robustness, showing that larger models perform better but remain sensitive to prompt phrasing. To measure fragility, we define the Prompt Fragility Value, which captures retrieval divergence across m prompts, each retrieving n chunks:

$$C_{\text{sem}} = \frac{1}{n(n-1)m} \sum_{i=1}^{n} \sum_{j \neq i}^{n} \sum_{k=1}^{m} \max_{l \in [1,m]} \big(\text{sim}(d_{i,k}, d_{j,l})\big) \tag{2}$$

  where d_{ik} is the i-th chunk from the k-th prompt, and $\text{sim}(d_{ik}, d_{jl})$ measures semantic similarity between chunks retrieved by different prompt variations.
- **Generation Anomaly** occurs when generated answers contain factual errors or inappropriate content. Yue et al. [68] propose methods for evaluating attribution correctness, while Datta et al. [17] introduce the TruLens system to detect hallucinations. Recent approaches focus on automatic fact-checking and citation evaluation.

  To measure generation anomalies, we define the Generation Anomaly Value, which combines two components: the model's self-reported confidence and the proportion of erroneous citations:

$$\mathcal{A}_{gen} = \alpha \cdot (\text{Mean Confidence}) + \beta \cdot (\text{Error Chunk Ratio}) \tag{3}$$

  where α and β are adjustable parameters, with default values of 0.5, reflecting the balance between confidence and error chunk ratio. Mean Confidence is extracted from the model's internal confidence scores, while Error Chunk Ratio represents the proportion of citations that reference incorrect or irrelevant chunks.
- **Standard Anomaly** refers to deficiencies in reference answers or evaluation standards. Kamalloo et al [31] find that discrepancies between system outputs and references often arise from incomplete or incorrect gold standards, with over 50% of lexical matching failures due to semantically equivalent answers. Researchers advocate for more sophisticated methods to evaluate model generalization and accuracy.

  The Standard Anomaly Value combines two components: GPTCheck for uncertainty detection and FactScore for verifying facts against a knowledge base:

$$R_{\text{hall}} = \alpha \cdot \text{GPTCheck}(\text{output}) + \beta \cdot \text{FactScore}(\text{output}) \tag{4}$$

This metric captures issues like outdated information, erroneous sources, and evaluation set errors. Default values for α and β are 0.4 and 0.6 respectively, emphasizing factual accuracy over uncertainty detection.

RAGTrace supports extensible evaluation metrics, allowing users to incorporate additional metrics tailored to domain-specific requirements.

6.2 Data Processing Pipeline

The data processing pipeline in RAGTrace handles both the initial corpus and runtime queries through a structured workflow. For embedding generation, we utilize OpenAI's text-embedding-3-large model to create high-dimensional vector representations of all text chunks and questions. These embeddings capture semantic relationships between chunks and queries, serving as the foundation for both retrieval and visualization components.

For dimensionality reduction and visualization, we employ openTSNE [44], a modular Python library that implements the t-SNE algorithm. The initial corpus of chunks is projected into a two-dimensional space while preserving local neighborhood relationships:

$$X_{2D} = \text{openTSNE}(X_{emb}, \text{perplexity}, \text{iterations}) \tag{5}$$

where X_{emb} represents the high-dimensional embeddings and X_{2D} is the resulting two-dimensional projection. This 2D space is then partitioned into a 200×200 grid, with cell density calculated as:

$$\text{Density}(c_{i,j}) = \frac{|\{x \in X_{2D} | x \text{ falls in cell } c_{i,j}\}|}{|X_{2D}|} \tag{6}$$

These density values directly inform the heatmap visualization, with higher density regions rendered with greater intensity.

For thematic analysis and topic extraction within grid cells, we implement a weighted entity-based approach:

$$T(c_i, t) = \sum_{e \in E(c_i)} \text{sim}(e, t) \cdot w(e) \tag{7}$$

Where $T(c_i, t)$ represents the relevance between topic t and cell c_i. $E(c_i)$ is the set of named entities extracted from chunks in cell c_i, $\text{sim}(e, t)$ measures the semantic similarity between entity e and topic t, and $w(e)$ is the weighted importance, calculated by TF-IDF value, of entity e based on its frequency.

For new user queries at runtime, we implement an incremental fitting approach where new embeddings are projected onto the existing t-SNE space. This approach allows new questions to be positioned appropriately relative to the existing knowledge base without recomputing the entire projection. The iteration count (default: 50) can be adjusted by users based on their acceptable latency-quality tradeoff.

The data flow for each evaluation metric follows a carefully orchestrated process. For Retrieval Failure assessment, the system compares retrieved chunks against ground truth annotations provided by domain experts. Prompt Fragility evaluation involves generating multiple query variations through controlled paraphrasing, collecting top-ranked chunks for each variation, and performing pairwise comparisons to assess retrieval consistency. Generation Anomaly detection processes model outputs to extract confidence signals and identify citation claims, subsequently verifying each citation against referenced chunk content to calculate error ratios. Standard Anomaly evaluation employs dual verification through GPTCheck for uncertainty detection and FactScore for fact verification against curated knowledge bases.

To provide the most plausible standard chunks in Chunk-Relink Graph, we concatenate each question with its corresponding ground truth in the dataset to form a retrieval prompt that is indexed in the vector database. If no ground truth exists, users can be guided to define custom standards.

7 Usage Scenario

This section walks through RAGTrace using hypothetical use cases, demonstrating its capacity to enhance transparency and control in RAG-based systems.

Scenario Setting. Alex, a researcher exploring the performance of RAG systems in handling factual questions, turns to RAGTrace to investigate retrieval inconsistencies. He is particularly interested in identifying issues where entity fragmentation leads to inaccurate or incomplete responses. To explore this, Alex examines the question: "What's the dog's name in Tom and Jerry?" - a seemingly simple fact-based inquiry that poses challenges for retrieval and generation consistency.

Visual Exploration of Transparency and Attribution. Upon launching RAGTrace, Alex engages in a structured exploration of

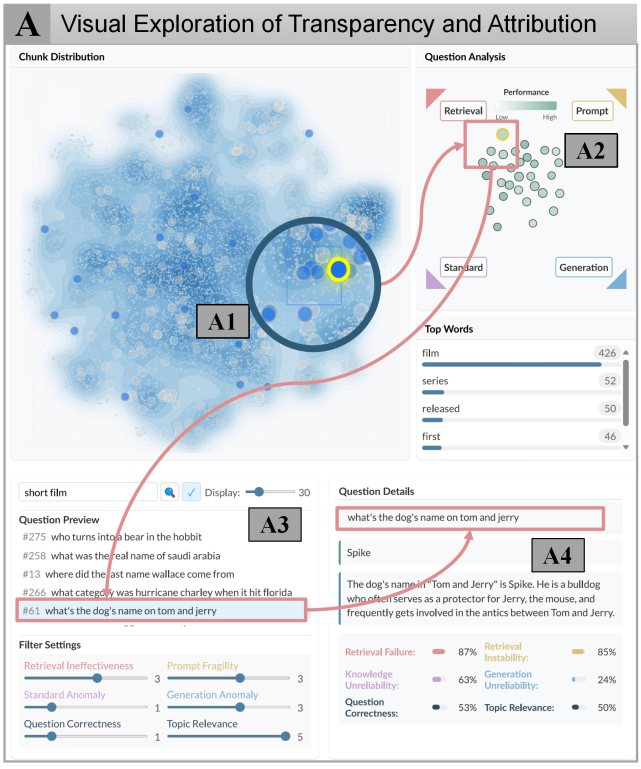

Figure 4: In Heatmap (A1), Force-directed Graph (A2), Question Search (A3) and Question Details (A4), the same question is represented using different visual encodings.

transparency and attribution using the visual diagnostics interface in Fig. 4 (A). He begins with the chunk-level Heatmap view (A1), which presents an overview of retrieval activity across documents. By entering "short film" in the Question Search (A3) to perform an embedding-level search, Alex observes a densely populated cluster of retrieved chunks, suggesting concentrated retrieval behavior in a specific domain. To investigate further, he transitions to the Force-directed Graph view (A2), which encodes similarity relationships among queries. Here, he identifies anomalous patterns in a subset of queries that diverge from canonical retrieval paths. Notably, the query "What's the dog's name in Tom and Jerry?" resides at the intersection of two distinct failure clusters (**Retrieval Failure** and **Prompt Fragility**), suggesting it embodies characteristics of multiple retrieval error modes. Continuing the analysis, Alex uses the entity-level diagnostic view (A4) and uncovers a recurring issue of entity fragmentation across semantically related queries. Together, these coordinated views (A1-A4) offer a comprehensive understanding of retrieval behaviors and failure patterns. As illustrated in Fig. 4, this workflow demonstrates how RAGTrace enables users to systematically trace and interpret retrieval dynamics through integrated visual analysis.

Retrieval Performance and Evidence Traceability Analysis. Intrigued, Alex moves to the Retrieval Performance and Evidence Traceability Analysis in Fig. 5 (B) to inspect the document ranking visualization. After automatic cross-component synchronization, he quickly noticed a blue chunk in the third row of the "standard" field, suggesting a close match to his desired answer. Using RAGTrace's Chunk-Relink Graph, Fig. 5 (B1.1), Alex observes that the

Figure 5: Changes in the Chunk-Relink Graph, corresponding evidence chains, and model-generated answers before and after user tuning.

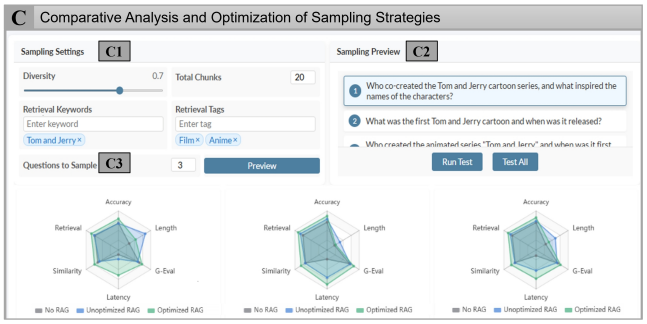

Figure 6: The new questions generated by users regarding the "Tom and Jerry" question, as well as their performance variations before and after optimization.

system's retrieval step did not prioritize this chunk effectively. Correlating this with the previously identified retrieval anomalies, he cross-references the Force-directed Graph and confirms that the retrieval ranking inconsistencies align with the observed entity fragmentation patterns. The Evidence Tracker and the model's answer reveal that information related to "Spike" is scattered across multiple retrieved documents, with its information density diluted by irrelevant documents, Fig. 5 (B1.2).

Alex narrows the broad issue of indicator anomalies down to retrieval anomalies and overly scattered retrieval content, possibly due to the lack of relevance labeling in the data source. So, he tries to recall algorithms that address such fine-grained problems. He applies the HyDE [24] (Hypothetical Document Embeddings) method, which generates a hypothetical answer document based on the question and uses its embedding for retrieval. As shown in Fig. 5 (B2.1), switching to the multi-round retrieval visualization, he tracks how chunk rankings evolve. He observes that after applying HyDE, the system gradually prioritizes previously overlooked "Spike" chunk, which actually contained complete information on the topic he wanted, Fig. 5 (B2.2), rather than fragmented information.

Comparative Analysis and Optimization of RAG Sampling Strategies. To evaluate the effectiveness of this optimization, Alex transitions to the Comparative Analysis and RAG Sampling Optimization in Fig. 6 (C). Using the Sampling Settings panel, Fig. 6 (C1), he configures parameters to identify similar entity-centric questions with potential retrieval inconsistencies. He selects queries related to entertainment media and character identification, which might suffer from similar fragmentation challenges Fig. 6 (C2).

The system presents these sampled questions alongside a Radar Chart visualization, Fig. 6 (C3), enabling Alex to compare performance metrics before and after optimization. The visualization highlights a consistent improvement in factual correctness and BLEU scores, particularly for queries that previously exhibited entity fragmentation. Further comparative analysis using multiple RAG strategies confirms that improved chunk consolidation and advanced retrieval techniques significantly enhance response accuracy, as illustrated in Fig. 6.

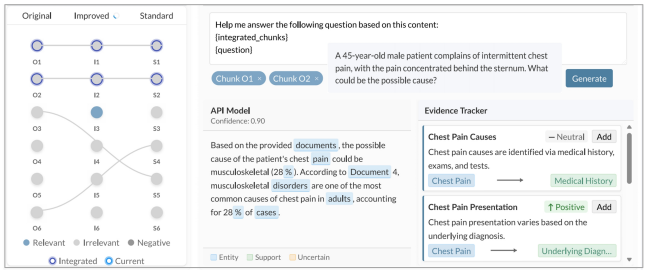

Figure 7: Robin's case: Initial retrieval results (left) versus results after prompt refinement (right), showing improved but still suboptimal answer generation.

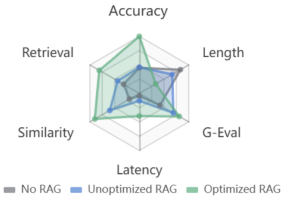

Figure 8: Robin uses the Radar Chart to validate improved accuracy and relevance metrics.

Domain Expert Workflow for RAG Optimization. To illustrate the system's utility for users without deep RAG algorithm knowledge, consider Robin, a medical researcher unfamiliar with RAG optimization intricacies. Robin uses RAGTrace to investigate potential causes for a patient's symptoms by identifying the question: "A 45-year-old male patient complains of intermittent chest pain, with the pain concentrated behind the sternum. What could be the possible cause?".

Transitioning to Fig. 7, Robin examines the initial retrieval results in the Chunk-Relink Graph, Fig. 7 (B1). He finds the top retrieved chunks unsatisfactory, offering only generic information about "chest pain" without specific diagnostic pointers. Guided by the system, Robin refines the retrieval prompt in Fig. 7 (B2), incorporating his domain expertise. He adds phrases like "possible differential diagnoses for substernal chest pain" and "cardiac conditions presenting with atypical symptoms". The system then retrieves slightly more relevant information, and the generated answer improves, but still lacks the required clinical depth and precision.

Recognizing that the general RAG data source might lack specialized medical knowledge, Robin decides to augment the knowledge base. He incorporates a more domain-specific dataset, such as the PubMed Central Open Access dataset[1]. After integrating this new data source, RAGTrace's retrieval mechanism accesses more precise information. The Chunk-Relink Graph now highlights documents discussing conditions like microvascular angina and anxiety-related chest pain, leading to a significantly more accurate and clinically relevant answer.

[1]https://pmc.ncbi.nlm.nih.gov/tools/openftlist/

Similar to Alex's approach, Robin transitions to the Comparative Analysis and Optimization component to evaluate his optimizations. He configures the Sampling Settings panel to test similar medical diagnostic questions. The Radar Chart visualization reveals substantial improvements in clinical accuracy and relevance scores across the sampled questions, Fig. 8. Encouraged by these results, Robin can iteratively return to the two previous components to further refine his approach, creating a continuous improvement cycle.

Through RAGTrace, Alex and Robin effectively diagnose and mitigate entity fragmentation and document missing issues in RAG-based queries, leading to more reliable and precise information retrieval. This workflow exemplifies how interactive visual analysis can empower users to refine retrieval strategies and enhance system transparency.

8 User Study

To understand how RAGTrace supports users in diagnosing and refining retrieval-generation dynamics in RAG systems, we conducted a user study where participants used RAGTrace to analyze and debug RAG systems. Our goal was to explore how the interactive visualization tools in RAGTrace assist users in identifying issues, interpreting model behavior, and iterating retrieval strategies.

In this study, we aimed to answer the following research questions:

- **RQ1.** How does RAGTrace help users identify attribution failures, entity fragmentation, or retrieval inconsistencies in RAG systems?
- **RQ2.** In what ways does RAGTrace influence users' strategies for debugging and optimizing RAG workflows, particularly in terms of chunk selection, retrieval refinement, and prompt adaptation?
- **RQ3.** How do users interpret and make use of the visual explanations provided by RAGTrace, and to what extent do they trust the insights generated across different views (e.g., retrieval ranking, chunk relinking, generation anomaly graphs)?

8.1 Study Design

8.1.1 Participants. We recruited 11 participants (8 males, 3 females) through surveys and referrals. All participants had prior experience working with LLMs, particularly in contexts involving RAG. Four participants (P1-P4) are RAG researchers familiar with prompt engineering and information retrieval, while seven of them (P5-P11) are experts in various research fields. Participants were compensated with local currency equivalent to $15 for their participation.

8.1.2 Procedure. Participants first signed an informed consent form before beginning the study. Each session began with a 5-minute introduction to the RAGTrace system, providing an overview of its key features and functionality. Participants were then asked to select three questions of interest from the Natural Questions (NQ)[2] [33] dataset to evaluate the performance of the Llama3-70B model with a RAG system. The study utilized a randomly sampled subset of 300 questions from NQ test set. The RAG system was configured with a Wikipedia-based knowledge

[2]The Natural Questions (NQ) corpus is a real-user open-domain QA dataset where systems must comprehend entire Wikipedia articles to answer questions, designed for realistic and challenging evaluation.

base containing approximately 20 million semantically segmented chunks, ensuring comprehensive coverage of factual information across multiple domains.

For each selected question, participants were instructed to assess the model's performance, identify potential issues in cases where the model performed poorly, and propose insights regarding possible strategies to enhance the RAG workflow. After evaluating the questions, participants were asked to improve the model's generation quality by adjusting retrieval parameters, refining generation prompts, or incorporating external or custom data sources. They then used RAGTrace's iterative system evaluation to assess the actual performance improvements resulting from these modifications. When participants encountered retrieval or generation failures, they were encouraged to analyze the underlying causes and suggest actionable refinements to improve system performance.

After completing the assigned tasks, participants were given time to freely explore the RAGTrace tools, investigating additional questions or features according to their interests. The sessions concluded with a semi-structured interview where participants shared their comprehensive impressions of the tool and evaluated its effectiveness in diagnosing and refining RAG systems. Study sessions ranged from 45 to 90 minutes in duration, depending on the depth of participants' exploration and discussion.

8.1.3 Measures. For qualitative data, we transcribed participants' responses from the semi-structured interviews and applied thematic analysis to extract key themes related to their reasoning strategies, trust in RAGTrace, and refinement approaches. Two researchers independently coded the transcripts and resolved discrepancies through discussion.

For quantitative data, we used six items from the NASA-TLX questionnaire (including Physical Demand). Notably, we reversed the scale for the "Performance" dimension (related to "How successful were you in accomplishing what you were asked to do?") to align with users' intuitive expectations, so that higher values indicate greater perceived success, enhancing interpretability of the results. Additionally, we employed the Post-Study System Usability Questionnaire (PSSUQ) to evaluate participants' subjective satisfaction with RAGTrace's usability, interface design, and overall functionality. The PSSUQ provided valuable insights into users' perceived effectiveness and satisfaction with the system across multiple dimensions. Full survey items are listed in Fig. 10. Likert-scale responses were analyzed using the Wilcoxon signed-rank test due to the ordinal nature of the data.

We did not conduct external evaluations of participants' refinements, as debugging RAG workflows is highly contextual. Participants set their own debugging goals, and external criteria could lead to misaligned assessments. Their explanations also involved subjective interpretations of relevance and hallucination, making external judgment unreliable.

8.2 Results

Our quantitative analysis revealed high user satisfaction with RAGTrace's usability and effectiveness in supporting RAG system diagnosis and refinement. The PSSUQ results indicated strong overall satisfaction (mean = 6.18, SD = 0.60 on a 7-point scale), addressing **RQ3** by confirming users' positive reception of the system's visual explanations. Participants particularly valued the system's ability to effectively help them complete tasks (mean = 6.36, SD = 0.81) and its pleasant interface (mean = 6.27, SD = 0.79). The NASA-TLX results showed moderate mental demand (mean = 10.36, SD = 4.30 on a 20-point scale), low physical demand (mean = 4.09, SD = 3.73), and notably high self-reported task success (mean = 16.73, SD = 3.64), the latter supporting **RQ2** by demonstrating that users could effectively implement optimization strategies using the system. Participants also reported low stress levels (mean = 4.00, SD = 4.22) despite engaging with complex analytical tasks, suggesting that RAGTrace successfully reduced cognitive burden when analyzing RAG workflows. The combination of high usability ratings with moderate workload metrics indicates that RAGTrace effectively balances analytical power with accessibility, allowing both RAG experts and domain specialists to effectively leverage the system's capabilities.

Based on the semi-structured interviews, the feedback revealed several key benefits and areas for improvement regarding RAGTrace. Participants consistently reported that the system enhanced their ability to understand, debug, and refine RAG workflows.

8.2.1 Enhanced Understanding and Debugging Capabilities. In response to **RQ1**, participants consistently highlighted RAGTrace's effectiveness in demystifying the complex retrieval-generation process. In particular, the evidence chain visualization was frequently praised for its utility in tracing the origin of generated answers and verifying their correctness. As P7 articulated, it helped determine "if the answer is genuinely correct or just superficially so," providing crucial insights into the "source of the retrieved answer and its validity." P5 also found this feature "very practical for clearly understanding the pipeline." The system was noted for surfacing issues that might otherwise go unnoticed (P4). For instance, P9 utilized the evidence traceability analysis component to understand how query modifications (like adding quotes or themes) influenced retrieval outcomes, finding it particularly valuable when "embedding models are unstable," enabling them to "retrieve useful knowledge through experimentation." P9 further appreciated the system's ability to "supplement missing information, like the subject," in fragmented text outputs from the LLM. This deeper understanding directly facilitated more effective debugging strategies. P10 mentioned using the tool to diagnose "whether the knowledge base was lacking content" or assess "where RAG is useful," informing subsequent prompt improvements. Participants felt the system provided a much clearer grasp of the overall RAG workflow (P3, P9), with P9 contrasting it favorably against previous experiences of manually feeding knowledge to models.

System Intuitiveness and Responsiveness. Participants commended the system's intuitive design and responsive interface. P5 described the system as having "very good responsiveness, with operations feeling smooth and natural," noting that "the interface design is clear with well-defined functional modules, making the entire system easy to comprehend." P4 emphasized how the structured layout effectively linked inputs to analytical outputs, significantly reducing debugging time. The system was generally characterized as "clear and responsive," with P5 adding that it became "quite clear after familiarization." The color design was particularly praised for its clarity and visual satisfaction. P3 appreciated how the system

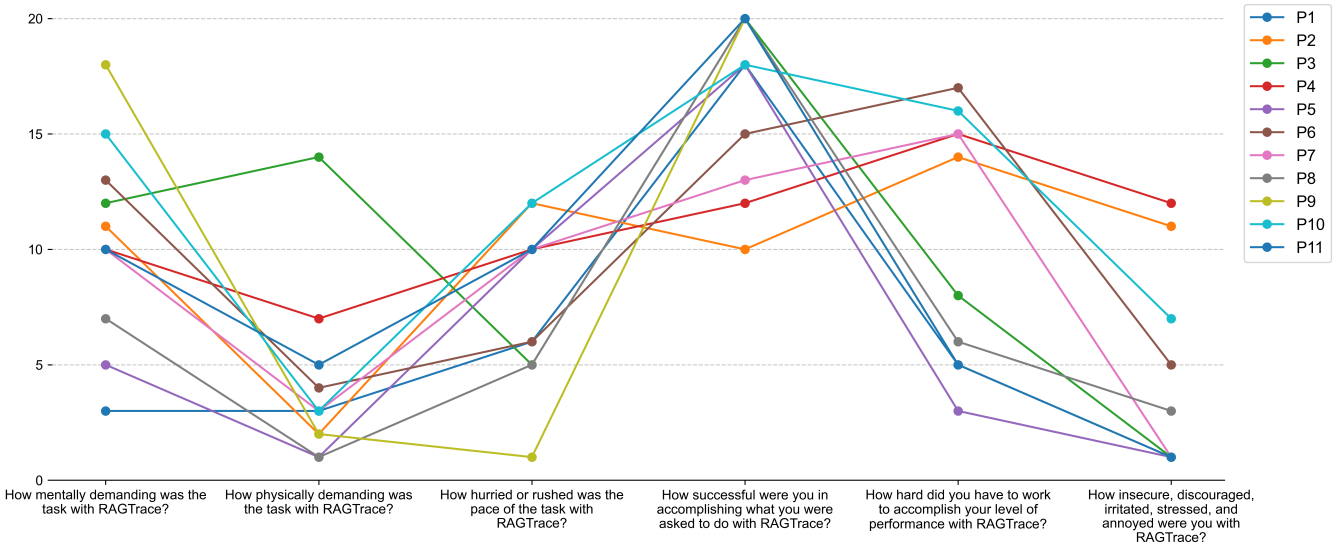

Figure 9: Participant Responses to the NASA-TLX Cognitive Load Questionnaire

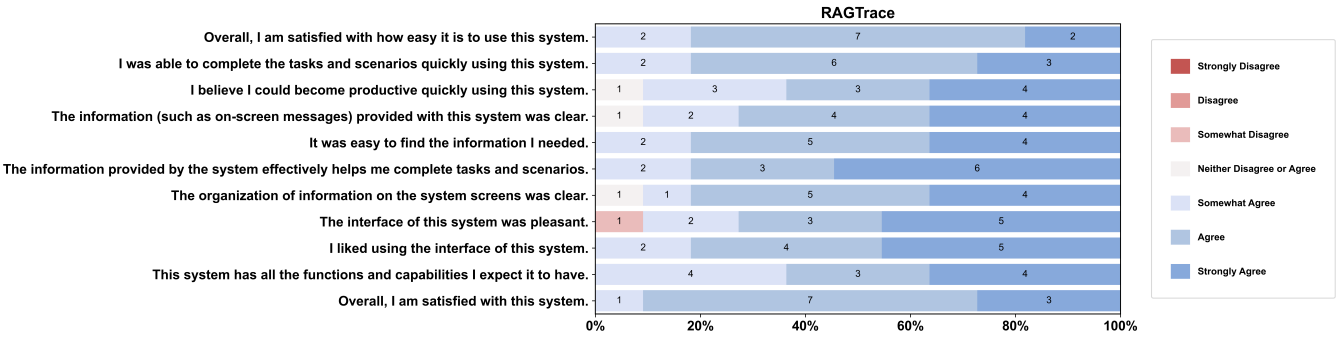

Figure 10: Participant Responses to Post-Study System Usability Questionnaire (PSSUQ)

provided a clearer understanding of "the overall workflow," while P6 highlighted the logical coherence of the system's process flow. P1 valued how the system "clearly separates various functional modules, allowing users to utilize them as needed." Overall, participants found that the visualization components significantly enhanced their understanding of the complex RAG processes.

8.2.2 Improved Efficiency and Workflow Integration. In response to **RQ2**, participants commonly reported substantial improvements in their debugging strategies and optimization workflows, particularly in terms of chunk selection, retrieval refinement, and prompt adaptation, when using RAGTrace. P7 estimated an efficiency improvement of retrieving document "at least 80%" compared to their previous methods without the tool. P4 emphasized that the structured layout, linking inputs to analytical outputs, "effectively reduces time" spent debugging. P9 reflected that the tool significantly enhanced their understanding of RAG, appreciating how it enables leveraging external knowledge to "improve the model's results when its own knowledge is limited." Furthermore, RAGTrace was seen as a valuable asset for demonstrating RAG system behavior and effectiveness (P2). P6 noted that the comprehensive functionality covering "retrieval, fine-tuning, and analysis" made the system applicable across multiple scenarios, from academic research to industrial optimization.

Opinions varied regarding the tool's integration into existing workflows. While industry practitioners (P6, P2) were enthusiastic about its potential for immediate integration into production environments, academic researchers (P3, P9) expressed more measured views about its applicability across different research contexts. P3 commented that "while extremely valuable for targeted diagnostics, the system might require customization for novel RAG architectures that deviate from standard retrieval patterns." This distinction highlights the different requirements between standardized production environments and more experimental research settings.

8.2.3 User Exploration Strategies and Insights. In response to **RQ3**, participants' interaction feedback revealed distinct exploration and sensemaking patterns, with users leveraging different components

of RAGTrace to interpret retrieval-generation dynamics and develop trust in the system's multi-faceted visual explanations.

Interactive Question Discovery and Location. Participants valued RAGTrace's capabilities for question exploration and rapid location. The force-directed graph was described by P4 as effectively facilitating *"navigation and locating desired questions,"* while P2 appreciated how it helped *"position questions to find the most relevant ones."* P6 considered the search questions feature a *"core function"*, and P7 emphasized the system's ability to *"generate new questions and add custom queries beyond the static dataset,"* enabling *"dynamic assessment of system performance."* When combined with the heatmap, these tools provided powerful problem identification capabilities, with P4 noting that this combination could effectively reveal *"issues that might otherwise go unnoticed."* P5 stated that the heatmap provided a *"clear cognitive map"* for identifying problematic terms to avoid, particularly how it visualized *"white spots that could be used to avoid certain specific keywords."* These exploration patterns demonstrate RAGTrace's multi-layered support for question discovery and location.

Query Reformulation and Retrieval Strategy Experimentation. Multiple participants engaged in systematic experimentation with query formulations to understand retrieval behavior. P9 discovered meaningful patterns through the system: *"After using this tool, I found that adding quotes around key terms in prompts produces different results, and introducing thematic elements seems to change outcomes as well."* Through knowledge base evaluation, P9 also revealed insights about embedding model instability: *"I've realized that embedding models can be unstable—trying different chunks a few times might eventually find the correct answer."* P10 and P1 employed the system to identify and remove redundant or misleading information from the knowledge base. P10 noted that *"cleaning up noisy documents improved retrieval precision more than adding additional context in several test cases,"* representing an alternative optimization strategy that some participants discovered through system exploration. However, P3 disagreed with this approach, arguing that *"maintaining comprehensive coverage is often more important than removing noise, since the ranking algorithm should handle relevance sorting appropriately."* This difference in perspective reflected participants' diverse backgrounds and the varied requirements of their use cases, while demonstrating the system's flexibility in supporting different retrieval strategy experiments.

Attribution and Evidence Chain Analysis. Participants found the system effective in revealing answer sources and verifying correctness. P2 particularly valued the Chunk-Relink Graph's comparison between standard and retrieved chunks, noting it was *"most useful"* because it visually revealed *"which standard answers were not utilized,"* thus identifying *"what information was lost"* and providing clear *"directions for improvement."* P1 also praised the Chunk-Relink Graph for *"clarifying several types of problems in one view,"* while P6 highlighted that the Chunk-Relink Graph helped *"quickly locate specific chunks when diagnosing retrieval-based errors."* P7 found the evidence chain visualization effective for tracing the origin of generated answers and verifying their correctness, helping determine *"if the answer is genuinely correct or just superficially so,"* providing crucial insights into the *"source of the retrieved answer and its validity."* P5 also found this feature *"very practical for clearly understanding the pipeline."* These responses demonstrate RAGTrace's value in providing detailed diagnostic capabilities and transparency.

Iterative Performance Analysis and Optimization. Participants evaluated and optimized system performance through complementary visualization components. The Radar Chart was utilized by P10 to effectively *"compare performance before and after improving prompts,"* while P8 appreciated its ability to facilitate *"associations between related questions,"* and P11 observed that *"when indicator values change, the Radar Chart might provide hints about problem locations."* P11 particularly appreciated how the system provided *"direct descriptive terminology that predefined good performance,"* reducing cognitive load by *"helping users determine what constitutes good performance without extensive analysis,"* while still allowing them to *"investigate how conclusions were reached."* P3 valued the ability to quickly grasp *"the overall workflow more clearly,"* demonstrating RAGTrace's value in providing high-level performance perspectives while supporting iterative system improvements. P9 contrasted this exploration pattern favorably against previous experiences: *"Before using this, I never worked with RAG systems—I always manually fed knowledge to models. Now I understand how to leverage systems to process knowledge that may not be initially verifiable."* This exploration pattern shows RAGTrace's capacity to expand users' conceptual understanding of RAG while supporting practical comparative analysis and iterative optimization.

8.2.4 System Limitations. Participants provided constructive feedback for future enhancements. Increased interactivity was a common theme, with suggestions to add previews of text content on hover/click within visualizations (P6) and functionality to highlight corresponding elements (e.g., selected nodes, chunks) across different views (P2). Customization options were also desired, including the ability to define "custom metrics" (P3), apply "directional filtering" to searches or visualizations (P1), and modify how metrics are displayed on the Radar Chart (P2). P11 suggested incorporating "pre-defined interpretations" or hints, possibly on the Radar Chart, to "reduce cognitive load" by offering initial assessments of performance changes, guiding further investigation. Lastly, enhancing text exploration by highlighting searched terms within retrieved chunks (P2) and making chunks easily clickable to view original context (P6) were also proposed.

Participants' improvement suggestions sometimes reflected competing priorities. For instance, while some participants (P11, P3) advocated for more automated guidance and interpretations to reduce cognitive load, others (P10, P7) emphasized the importance of maintaining user control and avoiding excessive automation that might obscure important nuances. P10 specifically cautioned against "over-abstracting the underlying retrieval mechanics," noting that "sometimes the 'messiness' of the raw retrieval results contains important diagnostic information." These contrasting perspectives highlight the challenge of balancing automation and transparency in analytical systems designed for diverse user groups with varying levels of technical expertise.

9 Discussion and Conclusion

In this paper, we present RAGTrace, an interactive evaluation system to support diagnosis and optimization of RAG workflows. Based

on a formative study (N=12) with RAG users, we distill key practices, challenges, and expectations in understanding retrieval relevance, tracing knowledge propagation, and resolving generation inconsistencies. Building on these insights, we develop a diagnostic methodology and an interactive analysis system that enable users to assess retrieval quality, identify generation errors, and refine retrieval strategies for improved RAG performance. A user study (N=11) and expert interviews demonstrated that RAGTrace could effectively facilitate troubleshooting, enhance users' understanding of retrieval-generation interactions, and foster more reliable and controllable RAG workflows.

9.1 Comparison with Conventional Evaluation Approaches

Traditional evaluation practices for RAG systems typically focus on isolated metrics for retrieval precision or generation quality [67, 69]. In contrast, RAGTrace provides an integrated analytical environment that enables users to explore the full spectrum of retrieval-generation relationships. By visualizing high-level performance metrics alongside fine-grained analyses of retrieval relevance, generation fidelity, and cross-component interactions, our system facilitates a deeper understanding of the influence that external knowledge sources exert on generated outputs. This multi-level approach addresses the research gaps identified in previous studies and extends the capabilities of conventional toolkits by emphasizing transparency in internal knowledge integration.

9.2 Design Implications

9.2.1 Understanding RAG Interactions Via Transparent Visualization. RAGTrace provides real-time visualization of the retrieval-generation dynamics, enabling users to clearly trace how retrieved chunks influence the generated content across iterations. This design echoes the "visible hands" approach proposed in prior work (e.g., WaitGPT [63]), where abstracted operations and dynamic visual feedback help externalize the behavior of LLM agents for better user understanding. In line with research advocating for interpretable and user-steerable AI systems, RAGTrace demonstrates how visualizing intermediate retrieval results and generation paths can foster user trust and facilitate more informed interactions.

9.2.2 Scrollytelling for Chunk Iteration. Building on scrollytelling techniques, which dynamically reveal content through user scrolling to enhance engagement with LLM-generated outputs [63], RAGTrace adapts this approach to visually trace the iterative refinement of retrieved chunks during RAG processes. This progressive and contextual representation allows users to better understand how the system refines its responses over time, aligning with design guidelines for incremental information disclosure in interactive systems [40]. Such design implications can inspire future RAG interfaces to support transparent and user-friendly exploration of retrieval and generation dynamics.

9.3 Scalability and Adaptability

RAGTrace is designed to operate effectively across a range of system scales—from standard laboratory experiments to real-world deployments with large-scale document repositories. While current implementations manage typical retrieval volumes and generation tasks efficiently, challenges such as increasing log data and the risk of visual clutter in dense scatterplot visualizations remain. Future enhancements will explore advanced sampling and visualization simplification techniques to ensure that our system remains robust and responsive even under high-demand conditions.

9.4 Generalizability and Extensibility

The data abstraction and multi-level analytical system underlying RAGTrace is inherently adaptable. Although we have demonstrated its effectiveness using several real-world RAG applications, the system's modular design allows it to be extended to various domains and customized for different retrieval and generation strategies. For example, while our current implementation effectively tracks token-level interactions and generation fidelity, adapting the system to support specialized retrieval strategies (e.g., domain-specific re-ranking) will further enhance its utility. This flexibility paves the way for broader application across multiple criteria decision-making environments, where relative comparisons between algorithm runs are essential even in the absence of a ground truth.

9.5 Limitations and Future Directions

The system currently relies on pre-defined metrics and static visualization techniques, which may not capture all nuances of dynamic retrieval-generation interactions in rapidly evolving systems. Future work will focus on incorporating adaptive, plugin-based modules that allow for easy integration of emerging evolutionary operators and retrieval strategies. Furthermore, extensive user studies—spanning diverse expertise levels and application contexts—are necessary to validate and refine the system's design. By enhancing both the transparency of internal RAG operations and the interpretability of generated outputs, RAGTrace aims to provide a critical tool for both researchers and practitioners seeking to optimize the performance and trustworthiness of RAG-based systems.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No. 62202217), Guangdong Basic and Applied Basic Research Foundation (No. 2023A1515012889), and Guangdong Key Program (No. 2021QN02X794). An implementation of RAGTrace is available at https://github.com/VIS-SUSTech/RAGTrace.

References

[1] Hasan Abu-Rasheed, Christian Weber, and Madjid Fathi. 2024. Knowledge Graphs as Context Sources for LLM-Based Explanations of Learning Recommendations. In *Proceedings of IEEE Global Engineering Education Conference*. IEEE, Kos Island, Greece, 1–5. doi:10.1109/EDUCON60312.2024.10578654

[2] Jae-wook Ahn and Peter Brusilovsky. 2013. Adaptive visualization for exploratory information retrieval. *Information Processing & Management* 49, 5 (2013), 1139–1164.

[3] Gulsum Alicioglu and Bo Sun. 2022. A survey of visual analytics for Explainable Artificial Intelligence methods. *Computers & Graphics* 102 (2022), 502–520. doi:10.1016/j.cag.2021.09.002

[4] Ian Arawjo, Chelse Swoopes, Priyan Vaithilingam, Martin Wattenberg, and Elena L. Glassman. 2024. ChainForge: A Visual Toolkit for Prompt Engineering and LLM Hypothesis Testing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3613904.3642016

[5] Lukas Bahr, Christoph Wehner, Judith Wewerka, José Bittencourt, Ute Schmid, and Rüdiger Daub. 2025. Knowledge graph enhanced retrieval-augmented generation for failure mode and effects analysis. *Journal of Industrial Information Integration* 45 (2025), 100807. doi:10.1016/j.jii.2025.100807

[6] Scott Barnett, Stefanus Kurniawan, Srikanth Thudumu, Zach Brannelly, and Mohamed Abdelrazek. 2024. Seven Failure Points When Engineering a Retrieval Augmented Generation System. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI*. Association for Computing Machinery, New York, NY, USA, 194–199. doi:10.1145/3644815.3644945

[7] Nicholas J. Belkin, Pier Giorgio Marchetti, and Colleen Cool. 1993. BRAQUE: Design of an interface to support user interaction in information retrieval. *Information processing & management* 29, 3 (1993), 325–344.

[8] Marialena Bevilacqua, Kezia Oketch, Ruiyang Qin, Will Stamey, Xinyuan Zhang, Yi Gan, Kai Yang, and Ahmed Abbasi. 2025. When Automated Assessment Meets Automated Content Generation: Examining Text Quality in the Era of GPTs. *ACM Transactions on Information Systems* 43, 2 (2025), 1–36. doi:10.1145/3702639

[9] Adrian M.P. Brasoveanu, Arno Scharl, Lyndon J.B. Nixon, and Răzvan Andonie. 2024. Visualizing Large Language Models: A Brief Survey. In *Proceedings of the International Conference Information Visualisation*. IEEE, Coimbra, Portugal, 236–245. doi:10.1109/IV64223.2024.00049

[10] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A Survey on Evaluation of Large Language Models. *ACM Transactions on Intelligent Systems and Technology* 15, 3 (2024), 45 pages. doi:10.1145/3641289

[11] Surajit Chaudhuri and Vivek Narasayya. 1998. AutoAdmin "what-if" index analysis utility. *ACM SIGMOD Record* 27, 2 (1998), 367–378.

[12] Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Press, Vancouver, Canada, 17754–17762. doi:10.1609/aaai.v38i16.29728

[13] Furui Cheng, Vilém Zouhar, Simran Arora, Mrinmaya Sachan, Hendrik Strobelt, and Mennatallah El-Assady. 2024. RELIC: Investigating Large Language Model Responses using Self-Consistency. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3613904.3641904

[14] Adam Coscia and Alex Endert. 2024. KnowledgeVIS: Interpreting Language Models by Comparing Fill-in-the-Blank Prompts. *IEEE Transactions on Visualization and Computer Graphics* 30, 9 (2024), 6520–6532. doi:10.1109/TVCG.2023.3346713

[15] Adam Coscia, Langdon Holmes, Wesley Morris, Joon Suh Choi, Scott Crossley, and Alex Endert. 2024. iScore: Visual Analytics for Interpreting How Language Models Automatically Score Summaries. In *Proceedings of the International Conference on Intelligent User Interfaces*. Association for Computing Machinery, New York, NY, USA, 787–802. doi:10.1145/3640543.3645142

[16] Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The Power of Noise: Redefining Retrieval for RAG Systems. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, New York, NY, USA, 719–729. doi:10.1145/3626772.3657834

[17] Anupam Datta, Matt Fredrikson, Klas Leino, Kaiji Lu, Shayak Sen, Ricardo Shih, and Zifan Wang. 2022. Exploring Conceptual Soundness with TruLens. In *Proceedings of the NeurIPS Competitions and Demonstrations Track*, Vol. 176. PMLR, 302–307.

[18] Julien Pierre Edmond Ghali, Kosuke Shima, Koichi Moriyama, Atsuko Mutoh, and Nobuhiro Inuzuka. 2024. Enhancing Retrieval Processes for Language Generation with Augmented Queries to Provide Factual Information on Schizophrenia. *Procedia Computer Science* 246 (2024), 443–452. doi:10.1016/j.procs.2024.09.424

[19] David Ellis. 1989. A behavioural approach to information retrieval system design. *Journal of documentation* 45, 3 (1989), 171–212.

[20] Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAs: Automated Evaluation of Retrieval Augmented Generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, St. Julians, Malta, 150–158.

[21] 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 ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, USA, 6491–6501. doi:10.1145/3637528.3671470

[22] 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*. Association for Computing Machinery, New York, NY, USA, 6491–6501. doi:10.1145/3637528.3671470

[23] Yu Fu, Shunan Guo, Jane Hoffswell, Victor S. Bursztyn, Ryan Rossi, and John Stasko. 2024. "The Data Says Otherwise" — Towards Automated Fact-Checking and Communication of Data Claims. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, New York, NY, USA, 1–20. doi:10.1145/3654777.3676359

[24] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise Zero-Shot Dense Retrieval without Relevance Labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Toronto, Canada, 1762–1777. doi:v1/2023.acl-long.99

[25] Binglan Han, Teo Susnjak, and Anuradha Mathrani. 2024. Automating Systematic Literature Reviews with Retrieval-Augmented Generation: A Comprehensive Overview. *Applied Sciences* 14, 19 (2024), 17 pages. doi:10.3390/app14199103

[26] Vikas Hassija, Vinay Chamola, Atmesh Mahapatra, Abhinandan Singal, Divyansh Goel, Kaizhu Huang, Simone Scardapane, Indro Spinelli, Mufti Mahmud, and Amir Hussain. 2024. Interpreting Black-Box Models: A Review on Explainable Artificial Intelligence. *Cognitive Computation* 16, 1 (2024), 45–74. doi:10.1007/s12559-023-10179-8

[27] Md Naimul Hoque, Tasfia Mashiat, Bhavya Ghai, Cecilia D. Shelton, Fanny Chevalier, Kari Kraus, and Niklas Elmqvist. 2024. The HaLLMark Effect: Supporting Provenance and Transparent Use of Large Language Models in Writing with Interactive Visualization. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–15. doi:10.1145/3613904.3641895

[28] Peter Ingwersen. 1992. *Information retrieval interaction*. Vol. 246. Taylor Graham Publishing, GBR.

[29] Minsuk Kahng, Ian Tenney, Mahima Pushkarna, Michael Xieyang Liu, James Wexler, Emily Reif, Krystal Kallarackal, Minsuk Chang, Michael Terry, and Lucas Dixon. 2024. LLM Comparator: Visual Analytics for Side-by-Side Evaluation of Large Language Models. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–7. doi:10.1145/3613905.3650755

[30] Minsuk Kahng, Ian Tenney, Mahima Pushkarna, Michael Xieyang Liu, James Wexler, Emily Reif, Krystal Kallarackal, Minsuk Chang, Michael Terry, and Lucas Dixon. 2025. LLM Comparator: Interactive Analysis of Side-by-Side Evaluation of Large Language Models. *IEEE Transactions on Visualization and Computer Graphics* 31, 1 (2025), 503–513. doi:10.1109/TVCG.2024.3456354

[31] Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. 2023. Evaluating Open-Domain Question Answering in the Era of Large Language Models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Toronto, Canada, 5591–5606. doi:10.18653/v1/2023.acl-long.307

[32] Tae Soo Kim, Yoonjoo Lee, Jamin Shin, Young-Ho Kim, and Juho Kim. 2024. EvalLM: Interactive Evaluation of Large Language Model Prompts on User-Defined Criteria. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–21. doi:10.1145/3613904.3642216

[33] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a Benchmark for Question Answering Research. *Transactions of the Association of Computational Linguistics* 7 (2019), 452–466. doi:10.1162/tacl_a_00276

[34] B. La Rosa, G. Blasilli, R. Bourqui, D. Auber, G. Santucci, R. Capobianco, E. Bertini, R. Giot, and M. Angelini. 2023. State of the Art of Visual Analytics for eXplainable Deep Learning. *Computer Graphics Forum* 42, 1 (2023), 319–355. arXiv:https://doi.org/10.1111/cgf.14733

[35] Sam Yu-Te Lee, Aryaman Bahukhandi, Dongyu Liu, and Kwan-Liu Ma. 2025. Towards Dataset-Scale and Feature-Oriented Evaluation of Text Summarization in Large Language Model Prompts. *IEEE Transactions on Visualization and Computer Graphics* 31, 1 (2025), 481–491. doi:10.1109/TVCG.2024.3456398

[36] Harry Li, Gabriel Appleby, and Ashley Suh. 2024. LinkQ: An LLM-Assisted Visual Interface for Knowledge Graph Question-Answering. In *Proceedings of IEEE Visualization and Visual Analytics*. IEEE, St. Pete Beach, FL, USA, 116–120. doi:10.1109/VIS55277.2024.00031

[37] Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2025. From Matching to Generation: A Survey on Generative Information Retrieval. *ACM Transactions on Information Systems* (2025). doi:10.1145/3722552

[38] Yuanjie Lyu, Zhiyu Li, Simin Niu, Feiyu Xiong, Bo Tang, Wenjin Wang, Hao Wu, Huanyong Liu, Tong Xu, and Enhong Chen. 2025. CRUD-RAG: A Comprehensive Chinese Benchmark for Retrieval-Augmented Generation of Large Language Models. *ACM Transactions on Information Systems* 43, 2 (2025), 359–369. doi:10.1145/3701228

[39] Aigerim Mansurova, Aiganym Mansurova, and Aliya Nugumanova. 2024. QA-RAG: Exploring LLM Reliance on External Knowledge. *Big Data and Cognitive Computing* 8, 9 (2024), 15 pages. doi:10.3390/bdcc8090115

[40] John M. O'Hara and S. Fleger. 2020. *Human-System Interface Design Review Guidelines*. Technical Report. Brookhaven National Lab. (BNL), Upton, NY (United States). doi:1644018

[41] Emre Oral, Ria Chawla, Michel Wijkstra, Narges Mahyar, and Evanthia Dimara. 2024. From Information to Choice: A Critical Inquiry Into Visualization Tools for Decision Making. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (2024), 359–369. doi:10.1109/TVCG.2023.3326593

[42] Sarah Packowski, Inge Halilovic, Jenifer Schlotfeldt, and Trish Smith. 2025. Optimizing and Evaluating Enterprise Retrieval-Augmented Generation (RAG): A Content Design Perspective. In *Proceedings of the International Conference on Advances in Artificial Intelligence*. Association for Computing Machinery, New York, NY, USA, 162–167. doi:10.1145/3704137.3704181

[43] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, USA, 311–318. doi:10.3115/1073083.1073135

[44] Pavlin G. Poličar, Martin Stražar, and Blaž Zupan. 2024. openTSNE: A Modular Python Library for t-SNE Dimensionality Reduction and Embedding. *Journal of Statistical Software* 109, 3 (2024), 1–30. doi:10.18637/jss.v109.i03

[45] 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 (2024), 26839–26874. doi:10.1109/ACCESS.2024.3365742

[46] Samantha Robertson, Zijie J. Wang, Dominik Moritz, Mary Beth Kery, and Fred Hohman. 2023. Angler: Helping Machine Translation Practitioners Prioritize Model Improvements. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–20. doi:10.1145/3544548.3580790

[47] Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, et al. 2024. Ragchecker: A fine-grained framework for diagnosing retrieval-augmented generation. *Advances in Neural Information Processing Systems* 37 (2024), 21999–22027.

[48] Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2024. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics, Mexico City, Mexico, 338–354. doi:10.18653/v1/2024.naacl-long.20

[49] Alireza Salemi and Hamed Zamani. 2024. Evaluating Retrieval Quality in Retrieval-Augmented Generation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, New York, NY, USA, 2395–2400. doi:10.1145/3626772.3657957

[50] Chamod Samarajeewa, Daswin De Silva, Evgeny Osipov, Damminda Alahakoon, and Milos Manic. 2024. Causal Reasoning in Large Language Models using Causal Graph Retrieval Augmented Generation. In *Proceedings of the International Conference on Human System Interaction*. IEEE, Paris, France, 1–6. doi:10.1109/HSI61632.2024.10613566

[51] Bhaskarjit Sarmah, Dhagash Mehta, Benika Hall, Rohan Rao, Sunil Patel, and Stefano Pasquali. 2024. HybridRAG: Integrating Knowledge Graphs and Vector Retrieval Augmented Generation for Efficient Information Extraction. In *Proceedings of the ACM International Conference on AI in Finance*. Association for Computing Machinery, New York, NY, USA, 608–616. doi:10.1145/3677052.3698671

[52] Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. 2024. Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. In *Proceedings of IEEE International Conference on Multimedia Information Processing and Retrieval*. IEEE, San Jose, CA, USA, 155–161. doi:10.1109/MIPR62202.2024.00031

[53] JooYoung Seo, Sanchita S. Kamath, Aziz Zeidieh, Saairam Venkatesh, and Sean McCurry. 2024. MAIDR Meets AI: Exploring Multimodal LLM-Based Data Visualization Interpretation by and with Blind and Low-Vision Users. In *Proceedings of the International ACM SIGACCESS Conference on Computers and Accessibility*. Association for Computing Machinery, New York, NY, USA, 1–31. doi:10.1145/3663548.3675660

[54] Zekai Shao, Shuran Sun, Yuheng Zhao, Siyuan Wang, Zhongyu Wei, Tao Gui, Cagatay Turkay, and Siming Chen. 2024. Visual Explanation for Open-Domain Question Answering With BERT. *IEEE Transactions on Visualization and Computer Graphics* 30, 7 (2024), 3779–3797. doi:10.1109/TVCG.2023.3243676

[55] Shangeetha Sivasothy, Scott Barnett, Stefanus Kurniawan, Zafaryab Rasool, and Rajesh Vasa. 2024. RAGProbe: An Automated Approach for Evaluating RAG Applications. arXiv:2409.19019 [cs.CL] https://arxiv.org/abs/2409.19019

[56] Da Song, Zhijie Wang, Yuheng Huang, Lei Ma, and Tianyi Zhang. 2023. DeepLens: Interactive Out-of-distribution Data Detection in NLP Models. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–17. doi:10.1145/3544548.3580741

[57] Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. 2024. Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*. Association for Computing Machinery, New York, NY, USA, 12—-22. doi:10.1145/3673791.3698415

[58] Teo Susnjak, Peter Hwang, Napoleon Reyes, Andre L. C. Barczak, Timothy McIntosh, and Surangika Ranathunga. 2025. Automating Research Synthesis with Domain-Specific Large Language Model Fine-Tuning. *ACM Transactions on Knowledge Discovery from Data* 19, 3 (2025), 1––39. doi:10.1145/3715964

[59] Prashant D. Tailor, Lauren A. Dalvin, John J. Chen, Raymond Iezzi, Timothy W. Olsen, Brittni A. Scruggs, Andrew J. Barkmeier, Sophie J. Bakri, Edwin H. Ryan, Peter H. Tang, D. Wilkin. Parke, Peter J. Belin, Jayanth Sridhar, David Xu, Ajay E. Kuriyan, Yoshihiro Yonekawa, and Matthew R. Starr. 2024. A Comparative Study of Responses to Retina Questions from Either Experts, Expert-Edited Large Language Models, or Expert-Edited Large Language Models Alone. *Ophthalmology Science* 4, 4 (2024), 100485. doi:10.1016/j.xops.2024.100485

[60] Tempest A. van Schaik and Brittany Pugh. 2024. A Field Guide to Automatic Evaluation of LLM-Generated Summaries. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, New York, NY, USA, 2832–2836. doi:10.1145/3626772.3661346

[61] Xingbo Wang, Renfei Huang, Zhihua Jin, Tianqing Fang, and Huamin Qu. 2024. CommonsenseVIS: Visualizing and Understanding Commonsense Reasoning Capabilities of Natural Language Models. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (2024), 273–283. doi:10.1109/TVCG.2023.3327153

[62] L. Wu, Z. Zheng, Z. Qiu, et al. 2024. A survey on large language models for recommendation. *World Wide Web* 27 (2024). doi:10.1007/s11280-024-01291-2

[63] Liwenhan Xie, Chengbo Zheng, Haijun Xia, Huamin Qu, and Chen Zhu-Tian. 2024. WaitGPT: Monitoring and Steering Conversational LLM Agent in Data Analysis with On-the-Fly Code Visualization. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, New York, NY, USA, 1–14. doi:10.1145/3654777.3676374

[64] Youfu Yan, Yu Hou, Yongkang Xiao, Rui Zhang, and Qianwen Wang. 2025. KNowNEt:Guided Health Information Seeking from LLMs via Knowledge Graph Integration. *IEEE Transactions on Visualization and Computer Graphics* 31, 1 (2025), 547–557. doi:10.1109/TVCG.2024.3456364

[65] Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. 2024. Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond. *ACM Transactions on Knowledge Discovery from Data* 18, 6 (2024), 1–32. doi:10.1145/3649506

[66] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Wen-tau Yih, and Xin Luna Dong. 2024. CRAG - Comprehensive RAG Benchmark. In *Proceedings of the Conference on Neural Information Processing Systems, Track on Datasets and Benchmarks*. 10470–10490. doi:10.48550/arXiv.2406.04744

[67] Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. 2024. Evaluation of retrieval-augmented generation: A survey. In *Proceedings of CCF Conference on Big Data*. Springer, Singapore, 102–120. doi:10.1007/978-981-96-1024-2_8

[68] Xiang Yue, Boshi Wang, Ziru Chen, Kai Zhang, Yu Su, and Huan Sun. 2023. Automatic Evaluation of Attribution by Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP*. Association for Computational Linguistics, Singapore, 4615–4635. doi:10.18653/v1/2023.findings-emnlp.307

[69] Hamed Zamani, Fernando Diaz, Mostafa Dehghani, Donald Metzler, and Michael Bendersky. 2022. Retrieval-Enhanced Machine Learning. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, New York, NY, USA, 2875–2886. doi:10.1145/3477495.3531722

[70] Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. 2024. CLAMBER: A Benchmark of Identifying and Clarifying Ambiguous Information Needs in Large Language Models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Bangkok, Thailand, 10746–10766. doi:10.18653/v1/2024.acl-long.578

[71] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for Large Language Models: A Survey. *ACM Transactions on Intelligent Systems and Technology* 15, 2 (2024), 1–38. doi:10.1145/3639372

[72] Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2024. Dense Text Retrieval Based on Pretrained Language Models: A Survey. *ACM Transactions on Information Systems* 42, 4 (2024), 1–60. doi:10.1145/3637870

[73] Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and Understanding the Prompt Sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP*. Association for Computational Linguistics, Miami, Florida, USA, 1950–1976. doi:10.18653/v1/2024.findings-emnlp.108

[74] T. Şakar and H. Emekci. 2025. Maximizing RAG Efficiency: A Comparative Analysis of RAG Methods. *Natural Language Processing* 31, 1 (2025), 1–25. doi:10.1017/nlp.2024.53

A Research Methods

A.1 Formative Study Participants

In our formative study, we recruited 12 participants (E1-E12) with varying degrees of experience with large language models and from diverse research backgrounds. Table 1 presents the demographic information and expertise details of these participants.

Table 1: Demographic Information and Background of Formative Study Participants

ID	Gender	LLM Experience	Research Background
E1	Male	2 years	RAG and LLM Agents
E2	Male	3 years	RAG Researcher
E3	Male	2 years	RAG
E4	Male	3 years	RAG Specialist
E5	Male	3 years	Nuclear Physics
E6	Female	1 year	Public Health
E7	Female	1 year	Business Analysis
E8	Male	2 years	Backend Development
E9	Male	1 year	Quantum Computing
E10	Male	2 years	Medical Science
E11	Male	3 years	Computer Science
E12	Female	1 year	Artificial Intelligence

The participants were recruited through university mailing lists and professional networks. They all had prior experience with large language models, ranging from 1 to 3 years. Their diverse research backgrounds allowed us to gather insights from multiple perspectives, enhancing the comprehensiveness of our formative study.

A.2 User Study Participants

For our user study, we recruited 11 participants (P1-P11) to evaluate the effectiveness and usability of RAGTrace. Table 2 presents the demographic information and background details of these participants.

The user study participants were selected to represent a range of expertise levels in large language models and diverse academic backgrounds. Notably, some participants (P1/E2, P2/E3, P4/E1, P6/E11, P8/E6, P9/E9, P10/E5) also participated in our formative study, providing valuable continuity in our research process. This overlap allowed these participants to compare their experiences with and without the RAGTrace system.

B Technical Implementation Details

This appendix details the technical infrastructure and methodologies supporting our RAGTrace system.

Model Infrastructure. Our system employs Llama3-70B-4bit as the primary language model, deployed locally across four NVIDIA GeForce RTX 3090 GPUs in a distributed inference configuration.

Table 2: Demographic Information and Background of User Study Participants

ID	Gender	LLM Experience	Research Background
P1	Male	3 years	RAG Researcher
P2	Male	2 years	RAG
P3	Male	2 years	RAG
P4	Male	2 years	LLM Agents
P5	Male	2 years	HCI
P6	Male	3 years	Computer Science
P7	Female	2 years	HCI
P8	Female	1 year	Public Health
P9	Male	1 year	Quantum Computing
P10	Male	3 years	Nuclear Physics
P11	Female	2 years	HCI

Confidence Score Acquisition. Confidence scores are obtained through two distinct pathways: (1) For open-source models, confidence values are extracted directly through local model interfaces that provide access to internal probability distributions; (2) For proprietary models, confidence scores are retrieved via OpenAI API endpoints that expose model uncertainty metrics.

Relevance Node Classification. The determination of node relevance employs a multi-stage classification pipeline. Candidate nodes are evaluated against ground truth annotations using semantic similarity measures. Nodes deemed relevant are highlighted in blue or grey within the visualization interface. We implements a multi-step pipeline as follows:

```
Algorithm: Named Entity Processing Pipeline
1. EXTRACT entities from ground truth using NLTK
2. FOR each entity:
   a. CHECK cache for existing synonyms/antonyms
   b. IF cached: RETURN cached results
   c. ELSE:
      - GENERATE contextual synonyms/antonyms via LLM
      - CACHE results in database
3. COMPARE entity sets between ground truth and model response
4. COMPUTE semantic similarity scores
5. RETURN evaluation metrics (precision, recall, F1)
to determine color
```

The pipeline first extracts named entities from ground truth documents using NLTK's pre-trained named entity recognition model. These entities undergo contextual filtering through a language model (DeepSeek-V3 in our experiments) that evaluates contextual relevance and generates semantically related terms, including synonyms and antonyms. The expanded entity vocabulary is then systematically compared against model-generated responses to identify semantic alignments and discrepancies.