Agentic Knowledge Graphs for Research Progression

Jason Cusati
Virginia Tech
Blacksburg, VA
djjay@vt.edu

***Abstract*—The accelerating growth of scientific literature has outpaced researchers' ability to not only identify promising directions, but also to *progress* research from problem statements to validated outcomes. While large language models (LLMs) improve access to information, today's tools rarely provide (i) structured representations of research problems and their operational context (assumptions, constraints, datasets, metrics), (ii) provenance for trustworthy reuse, and (iii) agentic workflows that support continuation, evaluation, and iterative refinement. We propose a system and research agenda for *agentic knowledge graphs for research progression*: an architecture that (1) constructs a provenance-grounded research knowledge graph from papers using automated extraction pipelines (leveraging reusable automation components where possible), and (2) orchestrates specialized agents over the graph to prioritize problems, propose next-step experiments or proofs, execute reproducible evaluations, and write results back as new structured artifacts. The central hypothesis is that combining symbolic structure (explicit problem objects and relations such as extends/contradicts/depends-on) with semantic retrieval and agentic control policies enables more reliable, interpretable, and actionable research workflows than text- or citation-based systems alone. We outline research questions and an evaluation plan measuring extraction reliability, linkage quality, retrieval utility, and human trust, with an emphasis on transparent provenance and human oversight.**

***Index Terms*—Agentic AI, Knowledge Graphs, Research Progression, Scientific Workflows, Provenance, Trustworthy AI, Interpretability, LLM-based Information Extraction**

I. Introduction / Background and Significance

The volume and velocity of scientific publication have reached a point where progress in many fields is constrained not by a lack of ideas, but by the difficulty of systematically advancing existing ones. Researchers must identify open problems, understand the assumptions and constraints under which they arise, evaluate their tractability, and design follow-on studies that meaningfully extend prior work. While digital libraries, citation indices, and large language models (LLMs) have substantially improved access to scientific information, they remain poorly suited for supporting this full process of *research progression*.

Most existing tools emphasize retrieval and summarization. Keyword search and citation-based navigation surface relevant papers, but leave the burden of synthesis, comparison, and continuation to the researcher. LLM-based assistants further reduce friction by summarizing content or suggesting ideas, yet typically operate over unstructured text and provide limited provenance, making it difficult to verify, reproduce, or operationalize their outputs. As a result, research workflows remain largely manual, fragmented, and difficult to scale—particularly for early-career researchers and interdisciplinary teams.

A growing body of work on scholarly knowledge graphs (SKGs) has demonstrated the value of structured representations of scientific knowledge. Systems such as the Open Research Knowledge Graph (ORKG) and large-scale automatically generated graphs like AI-KG and CS-KG show that research artifacts can be represented as machine-navigable entities with explicit semantic relations [1], [2]. These efforts improve comparison, querying, and analysis across papers, but they primarily treat papers or claims as atomic units of knowledge. They do not explicitly model open research problems, nor do they support the active continuation of research through evaluation, experimentation, or proof.

In parallel, recent work has explored the extraction of research questions, gaps, and hypotheses from scientific text using information extraction pipelines and LLMs [**?**], [3], [4]. While these approaches demonstrate that open problems can be identified automatically, they typically stop at extraction or ranking. They do not provide a framework for linking problems across papers, reasoning over their operational context, or advancing them toward resolution.

This paper argues that supporting *research progression* requires a tighter integration of three capabilities: (1) structured representations of research problems and their operational context (assumptions, constraints, datasets, metrics), (2) explicit relational structure with provenance to support trust and interpretability, and (3) agentic workflows that can reason over these structures to prioritize, continue, and validate research. We propose *agentic knowledge graphs for research progression* as a unifying architecture that addresses these needs.

In the proposed system, research papers are ingested and transformed into a knowledge graph where problems are first-class entities connected by semantic relations such as *extends*, *contradicts*, and *depends-on*. Automated extraction pipelines populate the graph with structured representations grounded in quoted evidence and metadata, enabling transparent reuse. On top of this graph, specialized agents are orchestrated using modern agent frameworks to perform tasks such as ranking problems by tractability, proposing follow-on experiments or proofs, executing reproducible workflows, and writing results back into the graph as new structured artifacts.

This framing shifts the role of AI from a passive assistant that surfaces information to an active participant in research workflows—one that operates over explicit structure, exposes its reasoning, and remains subject to human oversight. By focusing on progression rather than discovery alone, the proposed approach aims to lower barriers to contribution, accelerate cumulative science, and support more reliable and interpretable AI-assisted research practices.

II. Related Work

This work builds on and connects several lines of research, including scholarly knowledge graphs, scientific information extraction, semantic retrieval, and agent-based systems for scientific reasoning. Our contribution differs from prior work in its explicit focus on *research progression*—supporting not only identification of problems, but also structured continuation, evaluation, and validation.

A. Scholarly Knowledge Graphs

Scholarly knowledge graphs (SKGs) aim to represent scientific knowledge in structured, machine-navigable form. The Open Research Knowledge Graph (ORKG) pioneered the manual and semi-automated representation of research contributions such as research questions, methods, and results, enabling semantic comparison across papers [1]. While ORKG emphasizes human curation and FAIR principles, its reliance on manual input limits scalability.

Automated approaches such as AI-KG and CS-KG demonstrated that large-scale extraction of scientific entities and relations is feasible using natural language processing and machine learning [**?**], [2]. These systems enable trend analysis, semantic search, and bibliometric studies across millions of papers. However, they primarily treat papers or claims as the central unit of representation and do not explicitly model open research problems or their operational context. In contrast, our work elevates research problems to first-class entities and connects them directly to assumptions, constraints, datasets, and metrics, enabling progression beyond comparison and retrieval.

B. Extraction of Research Problems and Gaps

A parallel body of work focuses on extracting research questions, gaps, and hypotheses directly from scientific text. Early efforts relied on heuristics targeting high-yield sections such as *Future Work* or *Limitations*. More recent approaches combine information extraction pipelines with supervised models and large language models (LLMs). SciREX introduced document-level annotations for tasks, datasets, metrics, and relations, demonstrating the importance of structured context beyond sentence-level extraction [3].

Recent work has explored extracting or generating research questions and hypotheses using LLMs [**?**], [4]. While these approaches show that open problems can be identified at scale, they typically stop at extraction or ranking. They do not provide a framework for linking problems across papers, reasoning over constraints, or advancing problems toward resolution through structured continuation and evaluation.

C. Semantic Retrieval and Knowledge Discovery

Embedding-based retrieval methods such as SPECTER and hybrid symbolic–semantic search have improved research discovery by capturing conceptual similarity beyond keywords. Graph-based retrieval further leverages relationships between entities such as authors, methods, and datasets to support discovery and recommendation. Systems such as ResearchLink integrate graph structure with embeddings to recommend hypotheses and connections [5].

Despite these advances, most retrieval systems remain oriented toward finding relevant papers or claims rather than supporting actionable next steps. Our work complements these approaches by combining semantic retrieval with explicit problem representations and agentic workflows that support continuation and evaluation.

D. Agent-Based Scientific Systems

Agent-based systems have a long history in scientific automation, including early robot scientists that performed closed-loop hypothesis generation and experimentation [**?**], [**?**]. More recently, LLM-based agents have been proposed for autonomous experimentation and scientific reasoning in domains such as chemistry and materials science [**?**], [6].

While these systems demonstrate the feasibility of agent-driven science, they often operate over unstructured text or domain-specific protocols and provide limited transparency into decision making. In contrast, our approach grounds agent behavior in a structured, provenance-aware knowledge graph, enabling interpretability, auditability, and human oversight. This positions agentic behavior as a mechanism for progressing structured research problems rather than replacing human judgment.

III. Research Questions and Hypotheses

This research investigates whether agentic knowledge graphs can support research progression more effectively than existing discovery- or retrieval-oriented systems. The study is guided by the following research questions and hypotheses.

A. RQ1: Reliability of Structured Extraction

RQ1: To what extent can LLM-based extraction pipelines reliably identify and structure research problems, assumptions, constraints, datasets, and metrics from heterogeneous scientific papers?

H1: With structured prompting, schema validation, and lightweight human-in-the-loop review, LLM-based extraction will achieve precision, recall, and F1 scores within 10% of inter-annotator agreement among human experts across representative paper sections.

B. RQ2: Benefits of Graph-Based Representation

RQ2: Does representing extracted research problems as a knowledge graph with explicit relations and embeddings improve retrieval and linkage quality compared to text- or citation-based approaches?

H2: A hybrid symbolic–semantic representation will outperform keyword and citation-based baselines in retrieval tasks, yielding higher mean reciprocal rank (MRR), normalized discounted cumulative gain (nDCG), and more accurate identification of *extends*, *contradicts*, and *depends-on* relations.

C. RQ3: Measuring Research Progression

RQ3: Do agentic workflows operating over structured knowledge graphs improve researchers' ability to progress from literature review to concrete, evaluable research actions?

H3: Researchers using the proposed system will identify actionable continuations more quickly, propose more feasible next steps, and report higher trust and perceived usefulness than when using baseline discovery tools.

D. RQ4: Trust and Interpretability

RQ4: How do provenance, explicit structure, and human-in-the-loop controls affect user trust in agent-generated research recommendations?

H4: Explicit provenance, structured representations, and inspectable agent decisions will significantly increase user-reported trust and confidence compared to opaque, text-only AI assistants.

IV. Methodology

A. System Overview

The proposed system supports research progression by transforming unstructured scientific papers into a structured knowledge substrate and enabling agentic workflows over that structure. At a high level, the system ingests research papers, extracts structured representations of research problems and their operational context, organizes them into a knowledge graph with explicit relations and provenance, and orchestrates agents that operate over this graph to prioritize, continue, and evaluate research.

The system is designed as a closed-loop workflow. Structured problems extracted from the literature serve as inputs to agentic processes that propose follow-on studies or experiments, execute reproducible evaluations, and write results back into the knowledge graph as new structured artifacts. Human oversight is integrated throughout the loop to ensure correctness, trust, and interpretability.

V. System Architecture

The proposed architecture is designed to support *research progression* by integrating structured knowledge representation, automated extraction, and agentic reasoning into a single, closed-loop system. Rather than treating research papers as static documents, the system models research problems and their operational context as first-class entities that can be reasoned over, extended, and evaluated. Figure 1 (Appendix A) illustrates the high-level architecture, while Figure 2 shows the extended agentic configuration envisioned in future phases.

A. Architectural Overview

The system is organized into three conceptual layers:

1) **Knowledge Representation Layer**: a research knowledge graph that encodes problems, assumptions, constraints, datasets, metrics, baselines, and their relations with explicit provenance.
2) **Automation and Extraction Layer**: reusable document processing and information extraction components that populate and update the graph from research papers.
3) **Agentic Orchestration Layer**: specialized agents that operate over the graph to prioritize problems, propose continuations, execute evaluations, and write results back into the graph.

This layered design separates concerns between representation, automation, and reasoning, allowing each component to be developed, evaluated, and extended independently while supporting an end-to-end progression workflow.

B. Knowledge Representation Layer

At the core of the architecture is a research knowledge graph in which *problems* are modeled as first-class entities. Each problem node is associated with structured attributes including assumptions, constraints, datasets, metrics, and baselines, as well as evidence spans anchored to the source paper via DOIs, section identifiers, and quoted text. Problems are linked to one another through explicit semantic relations such as *extends*, *contradicts*, *reframes*, and *depends-on*.

This representation enables queries and analyses that are difficult or infeasible using paper-centric or citation-only models, such as identifying problems with shared constraints, ranking open challenges by tractability, or tracing how a problem evolves across multiple studies. By encoding provenance and confidence scores directly into the graph, the system supports trustworthy reuse and inspection of extracted knowledge.

The graph is stored in a property graph or RDF-compatible backend, paired with a vector index for semantic retrieval. This hybrid symbolic–semantic design allows structured filtering over explicit fields (e.g., domain, metric availability) while supporting approximate similarity search over problem statements and contextual embeddings.

C. Automation and Extraction Layer

Populating and maintaining the knowledge graph requires scalable and reliable automation. The architecture therefore leverages reusable document understanding and workflow components—such as those provided by Denario—either directly or through targeted extensions. These components support tasks including document ingestion, section segmentation, structured information extraction, schema validation, and versioned updates.

Research papers are ingested from open-access sources (e.g., arXiv, OpenAlex) and segmented into logical sections such as introduction, methods, limitations, and future work. Structured extraction pipelines, combining heuristics and LLM-based prompts, identify candidate problem statements and associated attributes, emitting normalized JSON objects that conform to a predefined schema. Validation and deduplication steps ensure consistency, while all extracted content is linked to source evidence to preserve provenance.

By relying on reusable automation components rather than bespoke pipelines, the system reduces engineering overhead and improves reproducibility, while retaining flexibility to adapt extraction strategies as models and domains evolve.

D. Agentic Orchestration Layer

On top of the populated knowledge graph, the system introduces an agentic orchestration layer that supports active research progression. Agents are specialized, goal-directed components that operate over the structured graph using tool interfaces and control policies. Modern agent orchestration frameworks, such as LangChain and related graph-based control abstractions, provide a practical foundation for implementing these workflows [**?**].

Examples of agent roles include:

- **Ranking agents**, which prioritize problems based on freshness, tractability, dataset availability, and cross-domain impact.
- **Continuation agents**, which propose follow-on experiments, proofs, or algorithmic extensions grounded in existing problem context.
- **Evaluation agents**, which execute reproducible workflows using specified datasets, metrics, and baselines.
- **Synthesis agents**, which summarize outcomes and write results back into the graph as new structured artifacts.

Crucially, agents do not operate over raw text alone. Instead, they consume and produce structured graph objects, enabling transparent reasoning, explicit dependency tracking, and iterative feedback. Human oversight is integrated at key decision points, allowing researchers to approve, modify, or reject agent-generated proposals and results.

E. Closed-Loop Research Progression

The interaction between the knowledge graph and the agentic layer enables a closed-loop research cycle. Extracted problems are ranked and selected; agents propose and execute continuations; results are validated and written back into the graph; and updated graph state informs subsequent iterations. This feedback loop transforms the system from a passive repository into an active substrate for research progression.

While the initial prototype focuses on ranking and structured continuation, the architecture is designed to support progressively richer forms of agentic behavior, including autonomous experimentation and mechanistic interpretability of agent decisions. These extensions are discussed in Section 7.

F. Extraction and Graph Construction

The extraction and graph construction pipeline is responsible for transforming unstructured research papers into structured, provenance-grounded knowledge graph entities. The design prioritizes reliability, reproducibility, and traceability, recognizing that downstream agentic reasoning depends critically on the quality of extracted representations.

1) Document Ingestion and Segmentation: Research papers are ingested from open-access repositories such as arXiv and OpenAlex. Each document is converted into a structured intermediate representation that preserves section boundaries, figures, tables, and metadata. Logical segmentation into sections (e.g., introduction, methods, limitations, future work) is performed using a combination of layout-aware parsing and heuristic rules, enabling downstream extractors to leverage section priors when identifying research problems and contextual attributes.

Reusable document processing components, such as those provided by Denario, are employed to manage ingestion, segmentation, and workflow orchestration. By decoupling document understanding from problem extraction logic, the system remains adaptable to evolving formats and corpora.

2) Structured Problem Extraction: Candidate research problems are extracted using a hybrid approach that combines pattern-based heuristics with LLM-driven structured extraction. High-yield sections such as *Limitations* and *Future Work* are prioritized, while contextual cues in introductions and discussions are also considered. LLMs are prompted to emit normalized JSON objects that conform to a predefined schema, including fields for problem statements, assumptions, constraints, datasets, metrics, and baselines.

All extracted elements are linked to explicit evidence spans in the source document, including section identifiers and character offsets, ensuring full provenance. When uncertainty is high, fields may be left empty or marked with lower confidence scores, avoiding hallucinated structure.

3) Normalization, Validation, and Deduplication: Extracted objects undergo schema validation and normalization prior to graph insertion. Controlled vocabularies and ontology mappings are applied where available (e.g., for domains or metrics), while free-text fields are preserved when standardization is infeasible. Duplicate or overlapping problem statements are identified using a combination of lexical similarity, embedding-based clustering, and human review.

Relations between problems, such as *extends* or *contradicts*, are inferred using pairwise comparison prompts and contextual similarity signals. Candidate links are assigned confidence scores and can be approved, edited, or rejected through lightweight human-in-the-loop interfaces.

4) Graph Population and Versioning: Validated problem objects and their associated attributes are inserted into the research knowledge graph as versioned entities. Each node and edge records provenance metadata, including source identifiers, extraction timestamps, and extractor versions, enabling auditability and rollback. Updates to existing problems (e.g., refined assumptions or new evidence) create new versions rather than overwriting prior state.

In parallel, vector embeddings are computed over selected textual fields (e.g., problem statements and assumptions) and stored in an associated vector index. This supports hybrid retrieval that combines structured graph queries with semantic similarity search.

5) Human-in-the-Loop Oversight: While automation is central to scalability, human oversight remains essential for maintaining quality and trust. The system therefore incorporates review checkpoints at key stages of the pipeline, including problem approval, relation validation, and conflict resolution. Feedback from human reviewers is logged and can be used to refine extraction prompts, heuristics, and confidence calibration over time.

G. Agentic Research Progression

The agentic layer operationalizes the research knowledge graph as an active substrate for *research progression*. Rather than treating extracted knowledge as a static repository, the system introduces specialized agents that reason over structured graph entities to prioritize, continue, and validate research in an iterative, human-supervised workflow.

1) Agent Roles and Responsibilities: Agents are designed as task-specific components that operate over the knowledge graph using explicit tool interfaces. Each agent consumes and produces structured graph objects, enabling transparent reasoning and dependency tracking. Core agent roles include:

- **Ranking agents**, which prioritize open research problems using graph-native features such as problem age, availability of datasets and metrics, strength of supporting evidence, and cross-domain connectivity.
- **Continuation agents**, which propose concrete next steps for a given problem, including follow-on experiments, algorithmic extensions, or formal analyses. Proposals are grounded in the problem's assumptions, constraints, and prior relations.
- **Evaluation agents**, which execute reproducible workflows using specified datasets, metrics, and baselines, producing structured results suitable for comparison and replication.
- **Synthesis agents**, which summarize outcomes, identify unresolved issues, and write results back into the knowledge graph as new problem versions, refinements, or resolved nodes.

This explicit decomposition limits agent scope, reduces error propagation, and enables targeted evaluation of each capability.

2) Agent Orchestration and Control: Agent workflows are orchestrated using modern agent frameworks that support tool calling, state management, and conditional execution (e.g., LangChain-style abstractions). Control policies govern when agents may act autonomously and when human approval is required, allowing the system to balance efficiency with oversight.

Orchestration is graph-aware: agents query the knowledge graph to retrieve structured context and update graph state upon task completion. For example, a continuation agent may retrieve a ranked problem and its dependencies, propose an experiment, and defer execution until approval is granted. This design enables reproducible, auditable workflows that can be paused, resumed, or replayed.

3) From Discovery to Validation: A central goal of the agentic layer is to move beyond problem discovery toward validation and progress. Agents do not merely suggest ideas; they operate on explicit representations of datasets, metrics, and baselines to support execution and evaluation. When experiments are run, results are captured in standardized schemas and linked back to the originating problem, preserving the full lineage of decisions and outcomes.

By grounding agent actions in structured graph context rather than free-form text alone, the system reduces ambiguity and improves interpretability. Researchers can inspect which assumptions, constraints, or prior results influenced a given agent decision, supporting trust and debugging.

4) Human-in-the-Loop Governance: Human oversight is integrated throughout the agentic workflow. Researchers may approve ranked problem selections, edit or reject continuation proposals, and validate experimental results before they are committed to the graph. This governance model ensures that agents augment rather than replace human judgment, and that responsibility for research direction remains explicit.

Feedback from human interactions is logged and associated with agent actions, enabling analysis of failure modes and iterative refinement of agent policies. This design supports gradual increases in agent autonomy while maintaining accountability.

5) Closed-Loop Research Progression: The interaction between the knowledge graph and the agentic layer forms a closed-loop progression cycle: extracted problems are prioritized, extended, evaluated, and updated in the graph; updated graph state informs subsequent agent actions. Over time, this loop enables cumulative advancement of research problems toward resolution or formal closure, transforming the system from a discovery aid into a progression-oriented research infrastructure.

H. Evaluation Plan

The evaluation strategy is designed to assess whether agentic knowledge graphs meaningfully support *research progression*, rather than discovery or retrieval alone. Evaluation is structured around the three research questions defined in Section 3, combining quantitative metrics with human-centered assessment.

1) RQ1: Extraction Reliability: To evaluate the reliability of structured problem extraction, a representative subset of papers from the target domain will be manually annotated by expert reviewers. Annotations will identify research problem statements, assumptions, constraints, datasets, metrics, and baselines.

LLM-based extraction outputs will be compared against human annotations using standard information extraction metrics, including precision, recall, and F1-score. Performance will be reported both overall and stratified by section type (e.g., introduction, limitations, future work) to assess where extraction is most reliable. Inter-annotator agreement will be measured to contextualize model performance relative to human variability.

Error analysis will further categorize failure modes, such as missed implicit assumptions, misclassification of future work, or incorrect evidence linkage. These results directly test Hypothesis 1, assessing whether structured prompting and validation can achieve extraction quality within an acceptable margin of human agreement.

2) RQ2: Graph-Based Retrieval and Linkage Quality: To evaluate the benefits of graph-based representation, retrieval and linkage tasks will be compared against baseline systems, including keyword search and citation-based navigation. Queries will be constructed to reflect realistic research tasks, such as identifying open problems with specific constraints or tracing extensions of a given technique.

Retrieval effectiveness will be measured using mean reciprocal rank (MRR) and normalized discounted cumulative gain (nDCG). Linkage quality for relations such as *extends* and *contradicts* will be evaluated by comparing predicted links against human judgments, reporting accuracy and confidence calibration.

This evaluation tests Hypothesis 2 by isolating the contribution of explicit structure and hybrid symbolic–semantic retrieval over text-only approaches.

3) RQ3: Measuring Research Progression Utility: Evaluating research progression requires assessing not only what information is surfaced, but how effectively it supports next-step action. To this end, controlled user studies will be conducted with graduate students and researchers in the target domain.

Participants will be asked to complete progression-oriented tasks, such as:

- Identifying a high-priority open problem suitable for follow-on work.
- Proposing a concrete continuation (e.g., experiment or extension) grounded in prior assumptions and constraints.
- Assessing the feasibility and expected impact of the proposed continuation.

Participants will perform tasks using the proposed system and baseline tools (e.g., Google Scholar, Semantic Scholar). Evaluation metrics will include task completion time, number of actionable continuations proposed, and user-rated usefulness and trust (Likert-scale). Qualitative feedback will be collected to assess perceived transparency, confidence in provenance, and ease of reasoning about next steps.

This evaluation directly tests Hypothesis 3 by measuring whether the system enables faster, more confident progression from literature to validated research actions.

4) Ablation and Sensitivity Analysis: To isolate the contributions of individual system components, ablation studies will be performed. These include removing graph structure (text-only retrieval), disabling agentic orchestration (retrieval-only workflows), and limiting provenance visibility. Changes in retrieval metrics, task performance, and user trust will be analyzed to assess sensitivity.

5) Reproducibility and Reporting: All evaluation datasets, prompts, schemas, and experimental configurations will be documented and released where possible. Results will be reported with confidence intervals and accompanied by qualitative examples illustrating successful and failed progression cases.

VI. Limitations

While the proposed architecture demonstrates a promising direction for supporting research progression, several limitations must be acknowledged.

A. Extraction Accuracy and Ambiguity

The reliability of downstream reasoning depends on the accuracy of structured extraction from unstructured text. Despite advances in LLM-based information extraction, research papers often express assumptions, limitations, and future work implicitly or ambiguously. Even with structured prompting, schema validation, and human-in-the-loop review, extraction errors such as missed assumptions or misclassified problem statements may persist [3], [4]. These errors can propagate into the knowledge graph and affect agent behavior if not carefully monitored.

B. Domain Scope and Generalizability

The initial evaluation focuses on a limited set of domains and venues to enable controlled study. While this scope is sufficient for proof-of-concept, results may not generalize directly to domains with different writing conventions, experimental practices, or publication norms. Scaling to broader scientific areas will require domain-adaptive extraction strategies and additional ontology alignment, as observed in prior large-scale scholarly knowledge graph efforts [2].

C. Agent Reliability and Oversight

Although agents operate over structured representations rather than free-form text, they remain susceptible to reasoning errors, incomplete context, or overconfidence in weak evidence. The system therefore requires careful governance to prevent inappropriate prioritization or misleading continuation proposals. Human oversight mitigates these risks but also limits full automation and introduces additional coordination costs.

D. Evaluation Constraints

User studies assessing research progression utility are necessarily small-scale and task-specific. Metrics such as task completion time and perceived usefulness may not fully capture long-term research impact or creativity. As with other AI-assisted discovery systems, evaluating true scientific value remains inherently challenging [**?**].

E. Resource and Infrastructure Costs

The system relies on LLM inference, graph storage, and vector indexing, which impose nontrivial computational and financial costs. These constraints may limit corpus size, update frequency, or interactive use, particularly for under-resourced institutions. Efficient batching, caching, and incremental updates are necessary but not fully explored in this work.

VII. Future Work

This work establishes a foundation for agentic knowledge graphs as an infrastructure for research progression. Several extensions represent promising directions for future research.

A. Richer Agent Specialization and Coordination

Future work will explore more specialized agents, including hypothesis critics, replication monitors, and review assistants, operating in coordinated workflows. Multi-agent interaction and role separation may improve robustness and reduce error propagation, as suggested by prior work on agent-based scientific systems [**?**].

B. Autonomous Experimentation and Verification

Beyond proposing continuations, agents may be extended to execute computational experiments or formal analyses using standardized workflows. Results would be logged back into the knowledge graph with full provenance, enabling comparison, replication, and incremental refinement. Recent advances in autonomous experimentation in chemistry and materials science suggest the feasibility of such closed-loop workflows [**?**], [6].

C. Mechanistic Interpretability of Agent Decisions

As agents play a more active role in research progression, understanding *why* they prioritize certain problems or continuations becomes critical. Future work will apply mechanistic interpretability techniques to analyze internal representations used by ranking and continuation agents, building on circuit-level analysis methods [**?**], [**?**]. Such analysis may support transparency, debugging, and responsible deployment.

D. Cross-Domain and Longitudinal Studies

Scaling the system across domains and over longer time horizons would enable analysis of how research problems evolve, converge, or are resolved. Longitudinal studies could assess whether agentic knowledge graphs meaningfully accelerate cumulative scientific progress or improve reproducibility across communities.

E. Community-Governed Research Progression

Finally, future work may explore community-driven governance models in which researchers contribute feedback, annotations, and corrections directly to the knowledge graph. Such participation could complement automated extraction and help balance agent-driven prioritization with diverse human values, aligning with the goals of open and responsible science.

VIII. Conclusion

This paper introduced the concept of *agentic knowledge graphs for research progression*, a system architecture that moves beyond research discovery toward structured continuation, evaluation, and validation of scientific work. By representing research problems as first-class entities with explicit assumptions, constraints, datasets, and provenance, the proposed approach enables transparent reasoning and actionable next-step decisions.

We showed how automated extraction pipelines can populate such graphs at scale, and how specialized agents operating over structured representations can prioritize problems, propose continuations, and support reproducible evaluation under human oversight. Unlike text- or citation-centric tools, this architecture integrates symbolic structure, semantic retrieval, and agentic control to support iterative research workflows.

From a broader perspective, agentic knowledge graphs offer a pathway toward more cumulative, interpretable, and inclusive scientific progress. By lowering barriers to understanding and extending prior work, the approach has the potential to accelerate research while preserving accountability and trust. At the same time, careful evaluation, governance, and interpretability remain essential to ensure responsible deployment. We view this work as a step toward AI-augmented science in which humans and agents collaborate to advance knowledge in a transparent and principled manner.

References

[1] M. Y. Jaradeh *et al.*, "Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge," in *Proceedings of the 10th International Conference on Knowledge Capture*, 2019.

[2] D. Dessi *et al.*, "Title placeholder," *Journal Placeholder*, 2022.

[3] S. Jain *et al.*, "Scirex: A challenge dataset for document-level information extraction," *arXiv preprint arXiv:2005.00512*, 2020.

[4] A. Taslimi *et al.*, "Extracting implicit scientific claims with large language models," *arXiv preprint arXiv:2501.00001*, 2025.

[5] F. Borrego and Others, "Researchlink: Integrating graph structure with embeddings for scientific discovery," *Journal of Example*, 2025, to appear.

[6] A. Bran, "Title placeholder," *Journal Placeholder*, 2024.