APOLLO: A SELF-GUIDED MULTI-AGENT SYSTEM FOR SCIENTIFIC ARTICLE GENERATION INSPIRED BY HUMAN THINKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Automatic generation of Wikipedia-like articles through Retrieval-Augmented Generation (RAG) has recently gained increasing attention. While recent advances in Large Language Models (LLMs) show considerable promise for synthesizing complex information, current RAG-based systems suffer from two fundamental limitations: they often rely on shallow retrieval strategies, leading to redundant content, and they lack effective mechanisms for factual verification and content organization. To address these challenges, we present APOLLO, a multi-agent framework specifically designed to generate high-quality, comprehensive articles with citations to the given sources. APOLLO simulates the iterative research and editorial process of human contributors through a set of specialized agents that collaboratively retrieve, fact-check, and structure information. To evaluate our method, we introduce SciWiki-2k, a dataset comprising 2,000 high-quality Wikipedia articles spanning 20 scientific domains. Compared to baseline methods, APOLLO produces articles with significantly improved structural coherence, content diversity, and factual accuracy. Human evaluations further establish the practical value of our approach for generating trustworthy, comprehensive articles.

Code – https://github.com/frosty-compiler/apollo

INTRODUCTION

Large Language Models (LLMs) have demonstrated remarkable capabilities in text generation. However, producing comprehensive, well-structured, and factually accurate articles remains a significant challenge Yang et al. (2023); Liang et al. (2023). Current approaches struggle to maintain coherence across extended content, synthesizing diverse information sources, and ensuring factual grounding throughout the generation process. While Retrieval Augmented Generation (RAG) has emerged as a promising solution to enhance LLM capabilities with external knowledge Lewis et al. (2020); Gao et al. (2024), most existing systems rely on static retrieval (orange dotted line Figure 1). Although recent variants like oRAG Shao et al. (2024) have attempted to improve upon this approach, these systems still lack reflective mechanisms which are fundamental in human research when exploring and synthesizing information.

Recent work has begun to address these limitations by introducing more structured and agent-based frameworks. For instance, STORM Shao et al. (2024) and OmniThink Xi et al. (2025) employ multi-agent frameworks to collect information from diverse perspectives or to simulate tree-based mind maps. These methods enhance topic coverage by collecting information from multiple perspectives or simulating reflective exploration (blue dotted line Figure 1). However, while these methods enhance coverage through dynamic retrieval, they struggle to represent relationships among the retrieved information and to organize it coherently Han et al. (2025). This is important because building a coherent view of a topic requires not only collecting isolated facts, but also understanding how the different concepts relate to each other Booth et al. (2003).

Furthermore, human research is inherently iterative and reflective, often involving repeated cycles of exploration, synthesis, and re-evaluation Doyle (1994); Kuhlthau (2004). For instance, when investigating the topic of Ensemble Learning (EL), a researcher might start with a broad overview of the concept and, as their understanding deepens, formulate more focused queries such as "bagging in ensemble methods" or "applications of ensemble learning". This evolving inquiry gradually

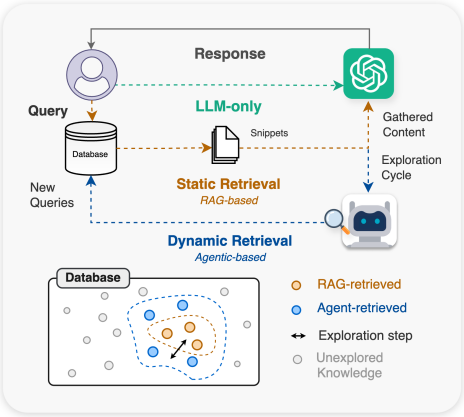

Figure 1: **Overview of retrieval strategies**. Static retrieval systems uses a single query to retrieve relevant content. In dynamic retrieval, gathered information is further analysed to issue new queries and discover new content.

builds what cognitive scientists refer to as "knowledge structure"; a mental framework in which new information is systematically integrated with prior knowledge Ausubel (1968). Such structures help researchers uncover conceptual relationships and identify information gaps that shape future searches Novak (1998); Chi et al. (2014).

Motivated by these observations, we introduce APOLLO, a multi-agent framework designed to emulate the iterative and reflective patterns of human research and structured writing. APOLLO begins by gathering information about a given topic through iterative proactive retrieval, organizing the retrieved evidence into a Knowledge Graph (KG) that captures entities, relationships, and topical hierarchies. This KG serves as both a record of discoveries and a scaffold for further investigation. Specialized agents analyze the evolving KG to identify missing links and underexplored subtopics, then generate targeted search queries to fill these gaps. This process repeats iteratively, with each cycle enriching the KG with additional relevant information.

After constructing the comprehensive KG, APOLLO transitions to article generation by extracting a hierarchical outline that represents the main concepts and their relationships. For each section, relevant content is retrieved from the information gathered during the knowledge curation step. A specialized writer agent synthesizes this material into an organized, well-supported text, while a reviewer agent systematically examines each draft section, verifies claims against referenced sources, and provides actionable feedback for refinement. This review-and-revision loop continues until all sections are complete and properly cited, mirroring best practices in collaborative academic writing Viégas et al. (2007).

To evaluate the effectiveness of our method, we introduce SciWiki-2k, a comprehensive benchmark for Scientific Article Generation (SAG). We evaluate APOLLO across multiple aspects, including knowledge curation, outline generation, and article generation, using automatic metrics, LLM-based qualitative assessments, and human evaluation. Inspired by the fact-checking literature Pang et al. (2023); Min et al. (2023); Thorne et al. (2018), we introduce two novel metrics to measure hallucination and content coverage. We conduct extensive experiments and compare APOLLO with state-of-the-art (SOTA) baselines. The results demonstrate substantial improvements across multiple key evaluation metrics Specifically, APOLLO increases information diversity by 9.2% over OmniThink, achieves a 7.1-point higher coverage rate than STORM on SciWiki-100, and reduces hallucination rate by 18% compared to the next-best baseline. Finally, our human evaluation study confirms that APOLLO outperforms competitive baselines in both overall article quality and factual accuracy, further validating its effectiveness in generating high-quality scientific content.

The main contributions of this work are as follows.

- We present APOLLO, a multi-agent framework that automates long-form, structured article generation through iterative KG construction, reflective gap detection, and agent-based fact verification
- We release the SciWiki-2k, a large benchmark dataset designed for assessing article generation models, and introduce two novel evaluation metrics called Hallucination and Coverage Rate to assess the factuality of the generated text.

Table 1: Capability matrix for automated article generation across different multi-agent systems.

Framework	oRAG	STORM	OmniThink	APOLLO
Dynamic Retrieval	×	✓	✓	✓
Structured Memory	×	✓	✓	✓
Reflective Thinking	×	×	✓	✓
Fact Verification	×	×	×	✓
Total Capabilities	**0/4**	**2/4**	**3/4**	**4/4**

Table 2: Quantitative analysis of the content found in the SciWiki-2k benchmark dataset.

Attribute	Value
Total Articles	2000
Scientific Domains	20
Avg. Number of Sections	7.8
Avg. Number of All-level Headings	19.9
Avg. Length of a Section (words)	483.3
Avg. Length of Article (words)	3672.7
Avg. Number of References	71.5

- We provide extensive experiments and a human evaluation study to demonstrate that APOLLO outperforms existing baselines in terms of coverage, diversity, and factual reliability metrics.

PRELIMINARY

PROBLEM DEFINITION

We define the task of SAG as follows. Given a topic T, representing a scientific concept (e.g., "Ensemble Learning"), the goal is to produce a comprehensive, factually grounded article $\mathcal{A}$ that explains the concept, outlines its key components, and organizes relevant subtopics and relationships in a coherent structure. We break the task of SAG through a three-stage process: (i) Knowledge Curation, retrieving and organizing relevant information $\mathcal{I} = \text{Retrieve}(T, \mathcal{C})$ from information sources $\mathcal{C}$, (ii) Outline Generation, constructing a structured outline $\mathcal{O} = \text{Construct}(\mathcal{I}, T)$ based on the retrieved information, and (iii) Article Generation, synthesizing the final article $\mathcal{A} = \text{Write}(\mathcal{O}, \mathcal{I})$ using the outline and retrieved information. The challenge lies in ensuring comprehensive coverage, factual accuracy, and coherent organization while avoiding redundancy and hallucinations. As shown in Table 1, none of the existing methods can fully support this task.

SCIWIKI-2K BENCHMARK

To address the lack of comprehensive benchmarks for SAG, we introduce SciWiki-2k, a curated dataset of 2,000 high-quality Wikipedia articles spanning 20 scientific domains. Unlike existing benchmarks Shao et al. (2024); Jiang et al. (2024c); Liu et al. (2018); Fan & Gardent (2022) that focus on general topics, SciWiki-2k specifically targets scientific concepts, providing high-quality Wikipedia articles as ground truth references for evaluating how well multi-agent systems can generate comparable scientific content.

The construction of our dataset follows a rigorous process. We begin by selecting a diverse set of topics representing key trends and core concepts from a broad range of scientific domains. For each topic, we identify and extract its corresponding Wikipedia article [1]. We then apply a quality filtering using the ORES API[2], retaining only articles rated as "B-Class", "Good Article", or "Featured Article" according to the Wikipedia grading scheme Wikipedia contributors (2025), thereby excluding low-quality, ambiguous, or insufficiently developed pages.

After quality filtering, we extract only the main text and section headings from these articles, omitting non-textual elements to standardize the dataset for text-based evaluation. A subsequent manual review is done to ensure the content of each article closely matches the intended scientific topic and domain; articles with misaligned or overly broad coverage are removed. For reproducibility, we release the full pipeline used to extract these pages, alongside SciWiki-2k[3]. Table 2 shows the composition of our dataset.

METHODOLOGY

We present APOLLO, a multi-agent framework that automates the generation of comprehensive, factually grounded articles. APOLLO 's pipeline consists of three main stages: (i) knowledge curation,

[1] All Wikipedia articles used in this dataset were retrieved between February and March 2025

[2] https://www.mediawiki.org/wiki/ORES

[3] https://huggingface.co/SciWiki

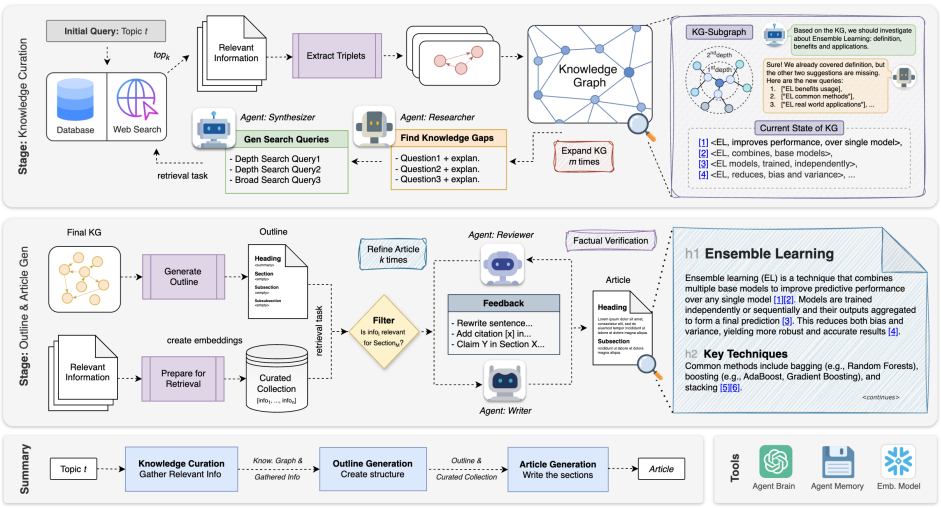

Figure 2: **Overview of APOLLO 's multi-agent framework for SAG**. The pipeline consists of three stages: (i) Knowledge Curation, where information is iteratively gathered and structured as a KG; (ii) Outline Generation, which derives a hierarchical article structure from the graph; and (iii) Article Generation, where writer and reviewer agents collaborate to produce and fact-check each section, resulting in comprehensive, well-cited articles.

(ii) outline generation, and (iii) article synthesis. Figure 2 provides an overview of our proposed framework.

KNOWLEDGE CURATION

APOLLO begins article generation by proactively gathering and organizing relevant information through an iterative process that constructs KGs to identify coverage gaps and guide targeted exploration. This approach addresses the limitation of simple retrieval strategies that miss valuable related information discoverable through more exploratory search processes Marchionini (2006); Savolainen (2018).

Initialization Stage. Given topic T, APOLLO supports two retrieval modes: (i) domain-constrained search from a curated corpus $\mathcal{C}_D$ and (ii) open-domain web search. For domain-constrained retrieval, we perform retrieval from the domain-specific collection as follows.

$$\mathcal{I}_0 = \text{Retrieve}(T, \mathcal{C}_D), \tag{1}$$

where $\mathcal{I}_0 = \{s_{i_1}, s_{i_2}, \ldots, s_{i_k}\} \subseteq \mathcal{C}_D$ represents the top-k most relevant snippets based on cosine similarity. Each snippet s_i contains raw text, embedding vectors, and metadata including source URLs. For web search, we perform an analogous retrieval using search engines.

Knowledge Graph Construction. For each retrieved snippet $s_i \in \mathcal{I}_0$, we apply an extraction operator as follows.

$$\Phi : s_i \longmapsto \big\{ (h, r, t) \;\big|\; h, t \in \mathcal{E},\, r \in \mathcal{R} \big\}, \tag{2}$$

where an LLM extracts a set of triplets in the form of (h, r, t) comprising of a head entity (h), a relation label (r), and a tail entity (t). The extracted triplets define a snippet-level sub-graph $G_i = (V_i, E_i)$ where $V_i \subseteq \mathcal{E}$ represents entities found in s_i and $E_i = \{(h, r, t) \mid h, t \in V_i,\ r \in \mathcal{R}\}$ contains edges linking those entities.

Graph Aggregation and Normalization. We construct the initial KG named $\mathcal{G}_0$ by aggregating all sub-graphs, as shown in the following.

$$\mathcal{G}_0 = \bigcup_{i=1}^{k} G_i = \left(\bigcup_{i=1}^{k} V_i, \bigcup_{i=1}^{k} E_i \right), \tag{3}$$

Since aggregation may result in duplicate entities, we apply a normalization function η as follows.

$$\mathcal{G}_0^* = \eta(\mathcal{G}_0) \tag{4}$$

where η is an LLM-based normalizer[4] that merges semantically equivalent entities and their associated edges.

Expansion Stage.

The APOLLO framework employs two collaborative agents that proactively expand the knowledge representation in multiple iterations through targeted information gathering, emulating human research patterns Pirolli & Card (1999).

- **Agent 1: Research Question Generator.** The first agent analyses the current KG, $\mathcal{G}_m^*$ at iteration m and produces focused research questions $\mathbb{Q}_m$ as follows.

$$\mathbb{Q}_m \;=\; \Psi\big(\mathcal{G}_m^*,\, \mathcal{M}_Q\big) \;=\; \{\,(q_j,\; \rho_j)\}_{j=1}^{n}, \tag{5}$$

  where Ψ denotes the system prompt which guides the agent to analyse $\mathcal{G}_m^*$, identify underexplored entities or relations, and generate $n = 10$ research questions including $n_d = 5$ "in-depth" questions targeting specific concepts and $n_b = 5$ "breadth" questions that branch into adjacent areas, (see agent interaction, top right Figure 2). Each question q_j includes a rationale ρ_j justifying its importance. To avoid repetition, the agent maintains a memory set $\mathcal{M}_Q$ that tracks all previously generated questions.
- **Agent 2: Query Synthesizer.** The second agent synthesizes focused search queries from the research questions:

$$\mathbb{L}_m \;=\; \Lambda\big(\mathbb{Q}_m,\, \mathcal{M}_L\big) \;=\; \{\ell_1, \ell_2, \ldots, \ell_t\}, \tag{6}$$

  where Λ is the system prompt guiding the agent to (i) decompose each question into salient entities and relations, (ii) paraphrase them into concrete search queries, and (iii) filter out terms already present in its memory $\mathcal{M}_L$. We set $t \leq 10$ to balance the exploration between depth and breadth.

Retrieval & Graph Update.

At each iteration, newly generated queries retrieve additional snippets (Eq. 1). These are used to create sub-graphs (Eq. 2), which are merged into the main KG (Eq. 3) after filtering previously seen content (Eq. 4). The agents update their memory sets to track queries and research questions already explored. The expansion continues for m iterations until a maximum depth is reached. The resulting curated collection is:

$$\mathcal{K} \;=\; \bigcup_{j=0}^{m} \mathcal{I}_j, \tag{7}$$

which contains all the collected snippets during the iterative process. This curated collection forms the foundation for constructing the final article.

Outline Generation

Given the final KG $\mathcal{G}_m^*$ and topic T, the framework generates a hierarchical article outline using an LLM-based function called Ω. This function analyses the structure of the KG and produces a set of headings and subheadings.

$$\mathcal{O} = \Omega(\mathcal{G}_m^*, T) = \{h_1, h_2, \ldots, h_p\}, \tag{8}$$

where each h_i is a main section or subsection, reflecting the entities and relations captured in $\mathcal{G}_m^*$. This step ensures that the outline matches the conceptual organization found during the knowledge curation step.

Article Generation

In this phase, the outline $\mathcal{O}$ is expanded into the final article $\mathcal{A}$ by gathering section-specific content and refining it through multiple revision cycles.

Section-Specific Retrieval

To support each section h_i in the outline, we first retrieve a set of candidate snippets from the collection $\mathcal{K}$ as follows.

$$\mathcal{R}_i = \text{Retrieve}(h_i, \mathcal{K}), \tag{9}$$

where $\mathcal{R}_i$ denotes the top-k snippets most similar to the section heading h_i. To ensure relevant information was gathered, we utilize an LLM-based filter and select a subset of the more relevant snippets as follows.

$$\mathcal{S}_i = \{s \in \mathcal{R}_i \mid \Theta(s, h_i) = \text{relevant}\}, \tag{10}$$

[4] Detailed specifications of all LLM-based functions and prompts are available in our open-source repository.

Table 3: **Outline & Article Quality Evaluation** Comparison across lexical and LLM-as-judge metrics between Apollo and baseline methods. † denotes significant differences ($p < 0.05$) compared to the best baseline.

Method	Outline Quality Metrics				
	Soft Recall	Entity Recall	Content Guidance	Hierarchical Clarity	Logical Coherence
Scientific Corpus					
oRAG	86.42	37.44	3.22	3.97	3.86
STORM	87.63	37.10	3.39	3.95	3.87
OmniThink	88.31	37.74	4.03	3.99	3.98
APOLLO	**91.82**†	**38.52**	**4.16**†	**4.00**	**4.01**†
w/o Reflection	80.75	36.14	3.36	3.93	3.82
Web Search					
oRAG	87.18	38.30	4.12	4.05	4.27
STORM	87.89	38.47	4.37	4.08	4.36
OmniThink	88.45	38.41	4.44	4.04	4.51
APOLLO	**92.25**†	**42.44**†	**4.63**†	**4.10**	**4.65**†
w/o Reflection	88.92	40.92	4.25	4.02	4.33

Method	Article Quality Metrics						
	ROUGE R1	ROUGE RL	Entity Recall	Interest Level	Coherence Organization	Relevance Focus	Depth Exploration
Scientific Corpus							
oRAG	41.84	14.03	5.92	2.34	4.32	3.92	3.88
STORM	42.11	14.44	6.51	1.61	4.85	4.10	4.54
OmniThink	41.76	13.94	5.53	1.37	4.28	4.12	4.27
APOLLO	**52.10**†	**15.81**†	**9.17**†	**3.29**†	**4.92**†	**4.90**†	**4.94**†
w/o Filter	49.17	15.51	7.35	1.99	4.74	4.57	4.77
Web Search							
oRAG	39.95	13.65	5.07	2.22	4.57	3.88	4.05
STORM	42.32	14.60	5.64	2.27	4.69	4.11	4.35
OmniThink	32.07	12.58	3.57	1.85	4.68	3.45	3.61
APOLLO	**52.01**†	**16.88**†	**9.44**†	**2.92**†	**4.84**†	**4.87**†	**4.46**†
w/o Filter	50.28	15.92	8.21	2.02	4.83	4.79	3.89

where Θ determines whether snippet s provides valuable information for writing section h_i. The resulting set $\mathcal{S}_i$ serves as the supporting material for generating the content of section h_i.

Iterative Content Generation After collecting the relevant information for a section, two agents collaborate to produce the content. For each section h_i, let $a_i^{(r)}$ denote the content generated in revision r.

- **Agent 3: Writer Agent.** The writer first generates a draft for section h_i using the supporting content found earlier:

$$a_i^{(0)} = \Gamma(h_i, \mathcal{S}_i), \tag{11}$$

  where Γ is the system prompt that instructs the agent to transform $\mathcal{S}_i$ into a well-organized and factual text which includes in-line citations to the given snippets. After the initial draft, the writer updates the section iteratively based on the feedback of the reviewer agent as follows.

$$a_i^{(r+1)} = \Gamma^{\text{revise}}(a_i^{(r)}, \mathbb{F}_i^{(r)}, \mathcal{S}_i), \tag{12}$$

  where $\mathbb{F}_i^{(r)}$ contains a list of bullet points that the writer agent follows to refine the content of the section at revision r. An example of this process is shown in Figure 2 (e.g., *Rewrite sentence...*).
- **Agent 4: Reviewer Agent.** The reviewer evaluates the generated content $a_i^{(r)}$ and maintains feedback memory $\mathcal{M}_F$:

$$\mathbb{F}_i^{(r)} = \pi(a_i^{(r)}, \mathcal{S}_i, \mathcal{M}_F), \tag{13}$$

  where π is the system prompt that instructs the agent to (i) assess whether cited snippets support claims, (ii) identify inconsistencies, and (iii) produce a structured feedback list $\mathbb{F}_i^{(r)} = \{f_1, f_2, \ldots, f_q\}$ with actionable revision items for the writer agent.

This collaborative process continues until either all feedback items are addressed or a maximum number of revisions $r_{\max}$ is reached.

Article Assembly. The final article is constructed by combining all refined sections while preserving the hierarchical structure from the outline. This produces a comprehensive article $\mathcal{A}$ where all claims are supported by evidence from the curated collection $\mathcal{K}$.

EXPERIMENTS

Baselines. We compare articles generated by our method with those generated by three other baselines, including oRAG Shao et al. (2024), STORM Shao et al. (2024), and OmniThink Xi et al. (2025). oRAG is a two-stage RAG baseline that generates an outline first, then processes each section independently using section-specific retrieval. STORM is a multi-agent system that simulates conversations between perspective-guided agents to gather diverse information before generating articles. OmniThink leverages a hierarchical tree representation to organize and synthesize information for article generation.

Hyper-parameters. We implement all the agents using Chain-of-Thought (CoT) Wei et al. (2022) and Zero-Shot (ZS) prompting the `gpt-4o-mini-2024-02-15` model. For reproducibility, we

set the *temperature* to 1.0 and *top-p* to 0.9. For web retrieval, we use Brave's API[5] with each query returning up to 3 web pages. For retrieval using the scientific corpus, we use Qdrant[6] with Snowflake embeddings[7]. During the knowledge curation step, we set the maximum depth of exploration to $m = 3$. For article generation, we allow up to $r_{\max} = 3$ writer–reviewer revision cycles per section. All experiments are conducted on a single AWS g5.2xlarge instance (24GiB GPU, 8 vCPUs). To ensure a fair comparison, we allow a maximum of 135 search queries for all baselines. For STORM, we use default values and set the limit of perspectives to 3; for Omnithink, we set the depth of the tree expansion to 3. To ensure robust evaluation, we conduct five independent runs for APOLLO and the baselines under two retrieval settings, including (i) web search using the Brave API and (ii) domain-constrained search using a curated scientific corpus. To build this dataset we use a set of review articles published across 2,700 journals, as well as the content of 43,000 books published in different science domains. For segmenting books and articles into passages, we considered each (sub-)section as a passage[8].

Dataset. Following prior work Shao et al. (2024); Jiang et al. (2024a), we evaluate APOLLO using SciWiki-100, a subset of SciWiki-2k dataset constructed by randomly selecting 5 topics from each of 20 scientific domains in this dataset. We generate articles using APOLLO and each baseline for the topics SciWiki-100 dataset. To evaluate whether the extracted KGs effectively capture the information from retrieved snippets, we employ the Measure of Information in Nodes and Edges (MINE) benchmark[9], a dataset designed to evaluate the completeness and factual consistency of KGs extracted from scientific text Mo et al. (2025).

Evaluation Setup

In the following, we explain the metrics used for evaluating the performance of each stage of our framework.

Knowledge Curation Quality. We assess the effectiveness of our knowledge curation module by measuring the number of unique sources retrieved and measure information diversity defined by Jiang et al. (2024b) as: $\text{Div}(\mathcal{I}) = 1 - \frac{1}{n(n-1)} \sum_{i \neq j} \cos(\mathbf{e}_i, \mathbf{e}_j)$. **Outline Quality.** We compare generated section headings against SciWiki-100 reference articles using soft recall and entity recall (named entity overlap via FLAIR NER Akbik et al. (2019)). LLM-as-judge assessments are done using M-Prometheus-7B Kim et al. (2024) to evaluate *Content Guidance*, *Hierarchical Clarity*, and *Logical Coherence* on a 5-point scale. Moreover, we perform an AB preference test comparing APOLLO's generated outlines against the best baseline (i.e., Omnithink) judged by three LLM evaluators (Claude-3.7-Sonnet Anthropic (2024), Llama-3.3-70B-Instruct Touvron et al. (2024), GPT-4o-mini)[10].

Article Quality. We assess the quality of the generated content for each section by using Recall, ROUGE-1, and ROUGE-L metrics, considering the content of articles from the SciWiki-100 dataset as gold data. Also, we conduct LLM-as-judge assessments on four different metrics, namely *Interest*, *Organization*, *Relevance*, and *Depth*.

Citation Quality. We introduce two novel automatic metrics for evaluating citation quality in generated scientific articles: hallucination rate and coverage. Hallucination rate ($1 - \frac{|C_v|}{|C|}$) quantifies the proportion of claims not supported by any evidence linked through in-line citations Min et al. (2023), and coverage ($\frac{|S_v|}{|S|}$) measures the proportion of article sections with at least one claim verifiably grounded in cited retrieved snippets Samarinas et al. (2025). LLM-based entailment is used for automated claim verification.

Human Evaluation. We select one topic at random from each of the 20 scientific domains and generate articles using the scientific corpus for both APOLLO and the best baseline (STORM) according to article generation metrics. This results in a total of 40 articles which is scored by Subject Matter

[5] https://brave.com/search/api/

[6] https://qdrant.tech/

[7] https://huggingface.co/Snowflake/snowflake-arctic-embed-m-v2

[8] The details of the corpus will be added upon publication.

[9] https://github.com/stair-lab/kg-gen

[10] We access this models via Amazon Bedrock: claude-3-7-sonnet-20250219-v1 and llama3-3-70b-instruct-v1

Table 4: Results of KG construction quality of our proposed model and baseline model using different LLM backbones.

LLM Backbone	Method	MINE Score	
		Normalized	Non-Normalized
Claude-3.7-Sonnet	Ours	0.714	0.701
	KG-Gen	**0.725**	0.680
	LightRAG	0.709	**0.705**
Llama-3.3-70B	Ours	**0.620**	**0.610**
	KG-Gen	0.580	0.550
	LightRAG	0.535	0.542
GPT-4o-mini	Ours	**0.501**	**0.486**
	KG-Gen	0.392	0.388
	LightRAG	0.432	0.428

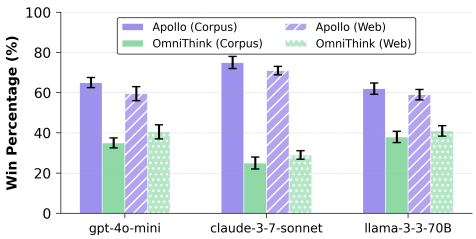

Figure 3: Win rate results from A/B preference tests comparing APOLLO's generated outlines against the best performing baseline across different LLMs evaluators. Claude-3-7-Sonnet displays the highest preference for APOLLO (79.5%). Error bars show standard deviation across 5 runs.

Table 5: Average number of unique URLs retrieved by each method.

Feature	APOLLO	OmniThink	STORM	oRAG
Scientific Corpus				
Num Unique URLs ↑	105.71	83.27	60.12	45.45
Info Diversity (%) ↑	60.81	54.74	42.23	33.02
Web Search				
Num Unique URLs ↑	88.60	63.22	59.82	19.49
Info Diversity (%) ↑	66.02	61.64	45.13	34.92

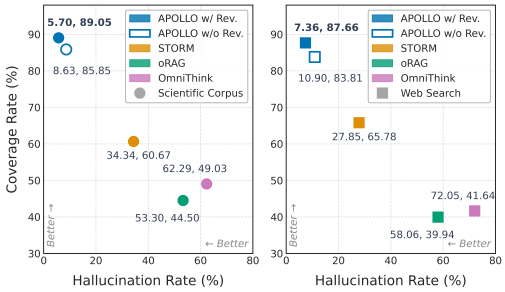

Figure 4: Scatter plot showing the trade-off between coverage rate and hallucination rate for APOLLO and baseline models.

Experts (SMEs) using the same 5-point rubric used by our LLM-as-judge for both outline and article quality. SMEs are scientific domain experts with advanced training relevant to the evaluation topics.

RESULTS

Starting with the knowledge curation phase we analyse whether our constructed KGs can capture meaningful entities and relationships to further guide the research stage of our framework. To this end, we measure the quality of our KGs construction using the MINE Score. We report the performance of APOLLO using different LLM backbones and two KG construction baseline methods, including KG-Gen Mo et al. (2025) and LightRAG Guo et al. (2024), in Table 4. The results show that our knowledge construction agent outperforms the baselines when using Llama-3.3-70B and GPT-4o-mini LLMs.

Building on these results, we measure how much unique retrieved information APOLLO can discover in the knowledge curation phase. As shown in Table 5, our proposed method consistently retrieves more unique URLs and achieves greater information diversity across both scientific corpus and web search settings. In particular, APOLLO outperforms the next-best method (i.e., OmniThink) by a wide margin in information diversity, confirming the effectiveness of our proactive retrieval agents.

Following our evaluation setup, we assess how well APOLLO constructs article outlines and organizes retrieved information into coherent sections. Looking at the left side of Table 3, we observe that outlines generated by APOLLO outperform baseline methods using both retrieval settings over all metrics. Higher value of Entity recall and Soft recall for APOLLO compared to the baselines, shows that the outlines generated by APOLLO are more similar to the outlines of the gold standard SciWiki-100 dataset. Moreover, considering the three metrics judged by LLM, we observe a significant improvement in *Content* and *Coherence* over the best baseline method (i.e., OmniThink). Additionally, we use an AB Preference test to validate the superior performance of APOLLO in generating outlines using SOTA LLMs and report the results in Figure 3. As can be seen, using different LLMs, APOLLO constantly wins the OmniThink baseline.

Based on the results presented in Table 3, we can see that APOLLO outperforms the baselines in terms of article-level evaluation metrics. Notably, we observe a significant improvement in *Depth*,

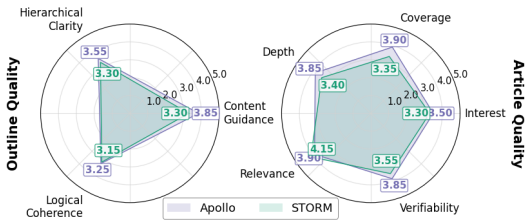

Figure 5: Result of human evaluation study comparing APOLLO and STORM article quality across eight metrics.

Relevant, and *Interest* metrics compared to the best performing baseline (i.e., STORM). Moreover, the results of the ablation study show that disabling the reflecting agent and our LLM-based filter function decreases the quality of our generated outlines and articles, respectively. The results of factual accuracy evaluation from Figure 4 demonstrate the importance of our review-and-revision iterative process to avoid the generation of hallucinated content and increase coverage rate. Particularly, we observe that when disabling the Reviewer agent, the articles generated by our method tend to include more unsupported claims and show a marked increase in hallucination rates.

Finally, based on the results of the human evaluation study depicted in Figure 5, we observe that in 7 out of 8 metrics APOLLO is consistently rated higher than STORM (best baseline for article creation). This result further validates the reliability of our automated evaluation metrics.

Related Work

Retrieval-Augmented Generation. RAG enhances LLMs with external knowledge to improve factuality and relevance Karpukhin et al. (2020); Guu et al. (2020). Early work focused on static retrieval pipelines for tasks such as QA Izacard & Grave (2021), summarization Menick et al. (2022), and citation generation Ram et al. (2023). Recent studies explore dynamic retrieval strategies that trigger queries adaptively during generation Jiang et al. (2023); Yao et al. (2023a). However, most RAG systems still operate as flat, single-step pipelines, lacking iterative reflection of retrieved knowledge Guo et al. (2024).

Knowledge Graphs for Generation. KGs have shown strong potential in improving the factuality, structure, and completeness of LLM outputs Dai et al. (2024); Markowitz et al. (2025). Methods like GraphRAG Yao et al. (2023b) and HopRAG Liu et al. (2025) explicitly leverage graph-based representations for multi-hop question answering and evidence tracing. KGs have also been used to structure retrieved evidence, support outline generation, and mitigate hallucinations Zhu et al. (2025); Cao et al. (2024).

Factual Grounding and Verification. Factual accuracy is a key challenge in knowledge-intensive generation tasks Zhang et al. (2023); Thorne et al. (2018). While some systems apply post-hoc verification Huang et al. (2023), recent work explores integrating self-reflection and iterative feedback into the generation process Madaan et al. (2024); Ye et al. (2023). However, maintaining factual consistency across multi-stage or multi-agent pipelines remains difficult, as agents can introduce unsupported claims or drift from retrieved evidence Nie et al. (2023); Liang et al. (2024).

Conclusion

We introduced APOLLO, a multi-agent framework for generating comprehensive, factually grounded scientific articles. By combining iterative KG construction, agent-based fact verification, and reflective writer–reviewer interactions, APOLLO produces content with high coverage, diversity, and factual reliability. To support rigorous evaluation, we also curated SciWiki-2k for the evaluation of the content quality, and propose two novel factuality metrics: Hallucination Rate and Coverage Rate. Extensive experiments and human evaluations confirm APOLLO 's superiority over existing baselines.

REFERENCES

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pp. 54–59. Association for Computational Linguistics, 2019.

Anthropic. Claude 3 models. https://www.anthropic.com/news/claude-3-family, 2024. Accessed: 2024-07-28.

David Paul Ausubel. *Educational psychology: A cognitive view*. Holt, Rinehart and Winston, 1968.

Wayne C Booth, Gregory G Colomb, and Joseph M Williams. Asking effective questions in information acquisition. *The craft of research*, pp. 40–57, 2003.

Lang Cao, Jimeng Sun, and Adam Cross. AutoRD: An Automatic and End-to-End System for Rare Disease Knowledge Graph Construction Based on Ontologies-enhanced Large Language Models, October 2024. URL http://arxiv.org/abs/2403.00953. arXiv:2403.00953 [cs].

Michelene TH Chi, James D Slotta, and Nicholas De Leeuw. Conceptual change in physics: The role of ontological categories. *Learning and instruction*, 4(1):27–43, 2014.

Xinbang Dai, Yuncheng Hua, Tongtong Wu, Yang Sheng, Qiu Ji, and Guilin Qi. Large Language Models Can Better Understand Knowledge Graphs Than We Thought, June 2024. URL http://arxiv.org/abs/2402.11541. arXiv:2402.11541 [cs] version: 3.

Christina S Doyle. *Information literacy in an information society: A concept for the information age*. ERIC Clearinghouse on Information and Technology, 1994.

Angela Fan and Claire Gardent. Generating biographies on wikipedia: The impact of gender bias on the retrieval-based generation of women biographies. In *ACL*, 2022.

Yunfan Gao et al. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2024.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. Lightrag: Simple and fast retrieval-augmented generation. In *arXiv preprint arXiv:2410.05779v1*, 2024. Highlights limitations of flat data representations and inadequate contextual awareness in existing RAG systems.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training. In *ICML*, 2020.

Haoyu Han, Harry Shomer, Yu Wang, et al. Rag vs. graphrag: A systematic evaluation and key insights. *arXiv preprint arXiv:2502.11371*, 2025.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaotian Feng, Bing Qin, and Ting Liu. A survey on hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*, 2023.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 874–880, 2021.

Yucheng Jiang, Yijia Shao, Dekun Ma, Sina Semnani, and Monica Lam. Into the unknown unknowns: Engaged human learning through participation in language model agent conversations. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9917–9955, Miami, Florida, USA, November 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.554. URL https://aclanthology.org/2024.emnlp-main.554/.

Yucheng Jiang, Yijia Shao, Dekun Ma, Sina J. Semnani, and Monica S. Lam. Co-STORM - Into the Unknown Unknowns: Engaged Human Learning through Participation in Language Model Agent Conversations, October 2024b. URL http://arxiv.org/abs/2408.15232. arXiv:2408.15232 [cs].

Yucheng Jiang, Yijia Shao, Peter Xu, Theodore A Kanell, Omar Khattab, and Monica S Lam. Assisting in writing wikipedia-like articles from scratch with large language models. *arXiv preprint arXiv:2402.14808*, 2024c.

Zhengbao Jiang, Rahul Gupta, Caiming Xiong, Chitta Baral, and Jiachang Liu. Self-rag: Learning to retrieve, generate, and reason over multiple passes. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP*, 2020.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. Prometheus: Inducing Fine-grained Evaluation Capability in Language Models, March 2024. URL http://arxiv.org/abs/2310.08491. arXiv:2310.08491 [cs].

Carol C. Kuhlthau. *Seeking meaning: A process approach to library and information services*. Libraries Unlimited, 2004.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

Percy Liang et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2023.

Xinyi Liang, Kangyu Chen, Yao Wang, Jintian Chen, Jiahao Zhang, Nan Xu, et al. A survey on llm-based multi-agent systems: workflow, infrastructure and challenges. *Frontiers of Computer Science*, 2024.

Hao Liu, Zhengren Wang, Xi Chen, Zhiyu Li, Feiyu Xiong, Qinhan Yu, and Wentao Zhang. HopRAG: Multi-Hop Reasoning for Logic-Aware Retrieval-Augmented Generation, February 2025. URL http://arxiv.org/abs/2502.12442. arXiv:2502.12442 [cs].

Peter J. Liu et al. Generating wikipedia by summarizing long sequences. In *ICLR*, 2018.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36, 2024.

Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006.

Elan Markowitz, Krupa Galiya, Greg Ver Steeg, and Aram Galstyan. KG-LLM-Bench: A Scalable Benchmark for Evaluating LLM Reasoning on Textualized Knowledge Graphs, April 2025. URL http://arxiv.org/abs/2504.07087. arXiv:2504.07087 [cs] version: 1.

Jacob Menick et al. Teaching language models to support answers with verified quotes. In *Transactions of the Association for Computational Linguistics*, 2022.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*, 2023.

Belinda Mo, Kyssen Yu, Joshua Kazdan, Proud Mpala, Lisa Yu, Chris Cundy, Charilaos Kanatsoulis, and Sanmi Koyejo. KGGen: Extracting Knowledge Graphs from Plain Text with Language Models, February 2025. URL http://arxiv.org/abs/2502.09956. arXiv:2502.09956 [cs].

Yixin Nie, Timo Schick, Nora Kassner, Hinrich Schütze, and Dan Roth. Faithful language models: A survey of existing methods and challenges. *arXiv preprint arXiv:2302.13629*, 2023.

Joseph D Novak. Learning, creating, and using knowledge: Concept maps as facilitative tools in schools and corporations. 1998.

Liang Pang, Qing Lin, Jiacheng Zhang, Zhiyuan Liu, and Maosong Sun. Text generation with fact-checking. In *ACL*, 2023.

Peter Pirolli and Stuart Card. Information foraging. *Psychological review*, 106(4):643–675, 1999.

Ankur Ram et al. Context-aware citation generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, 2023.

Chris Samarinas, Alexander Krubner, Alireza Salemi, Youngwoo Kim, and Hamed Zamani. Beyond factual accuracy: Evaluating coverage of diverse factual information in long-form text generation. *arXiv preprint arXiv:2501.03545*, 2025.

Reijo Savolainen. Berrypicking and information foraging: Comparison of two theoretical frameworks for studying exploratory search. *Journal of Information Science*, 44(5):580–593, 2018.

Yijia Shao, Yucheng Jiang, Theodore A. Kanell, Peter Xu, Omar Khattab, and Monica S. Lam. STORM - Assisting in Writing Wikipedia-like Articles From Scratch with Large Language Models, April 2024. URL http://arxiv.org/abs/2402.14207. arXiv:2402.14207 [cs].

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*, 2018.

Hugo Touvron et al. Llama 3: Open foundation and instruction models. https://ai.meta.com/llama/, 2024. Meta AI.

Fernanda B Viégas, Martin Wattenberg, and Matthew M McKeon. The hidden order of wikipedia. *International conference on online communities and social computing*, pp. 445–454, 2007.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Wikipedia contributors. Wikipedia: Content assessment, February 2025. URL https://en.wikipedia.org/w/index.php?title=Wikipedia:Content_assessment&oldid=1276507661. Page Version ID: 1276507661.

Zekun Xi, Wenbiao Yin, Jizhan Fang, Jialong Wu, Runnan Fang, Ningyu Zhang, Jiang Yong, Pengjun Xie, Fei Huang, and Huajun Chen. OmniThink: Expanding Knowledge Boundaries in Machine Writing through Thinking, February 2025. URL http://arxiv.org/abs/2501.09751. arXiv:2501.09751 [cs].

Jiannan Yang et al. Holoeval: Leveraging large language models for holistic text generation. *arXiv preprint arXiv:2310.14746*, 2023.

Shinn Yao, Jeffrey Zhao, Dian Yu, Karthik Narasimhan, Shixiang Cao, Xiaoxiao Chen, Jungo Kasai, et al. React: Synergizing reasoning and acting in language models. In *Advances in Neural Information Processing Systems*, 2023a.

Yao Yao, Xiaobin Cheng, Zhiyuan Liu, Maosong Sun, et al. Kg-augmented language models: Recent advances and prospects. *arXiv preprint arXiv:2311.08530*, 2023b.

Seonghyeon Ye, Doyoung Kim, Sungdong Hwang, Hyeonbin Yoo, and Minjoon Lee. Flask: Fine-grained language model evaluation based on alignment skill sets. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5769–5793. Association for Computational Linguistics, 2023.

Rui Zhang, Yuwei Chen, Yichong Lin, Xiang Ren, and Dong Yu. Faithfulqa: A benchmark for faithful open-domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Xiangrong Zhu, Yuexiang Xie, Yi Liu, Yaliang Li, and Wei Hu. Knowledge Graph-Guided Retrieval Augmented Generation, February 2025. URL http://arxiv.org/abs/2502.06864. arXiv:2502.06864 [cs] version: 1.