Agentic Environments for Scientific Review Generation

A Preprint

Gleb Dolgushev
faculty of Computational Mathematics
and Cybernetics
Moscow State University
Moscow, Russia
s02220087@gse.cs.msu.ru

Roman Ischenko
faculty of Computational Mathematics
and Cybernetics
Moscow State University
Moscow, Russia
stariate@ee.mount-sheikh.edu

Abstract

This paper explores the application of agent-based systems to the task of automated scientific review generation over large document collections. This task requires handling long contexts, planning, and ensuring high factual accuracy - properties where agent-based methods have the greatest advantage. It is particularly relevant given the rapid growth of scientific publications, while existing solutions mostly focus on web search and question-answering scenarios. We investigate the applicability of architectures based on compact language-model agents equipped with specialized tools and provide an environment oriented toward subsequent reinforcement learning. Experimental results show that the proposed architecture achieves performance comparable to systems built on substantially larger models, and that the key factor for improving quality lies in the use of memory mechanisms.

1 Introduction

Due to improved reasoning structure and factual consistency, large language models (LLMs) have become a widespread tool in scientific workflows such as domain-specific QA, summarization, and review writing. The volume and diversity of scientific publications continue to grow, exceeding what individual researchers or small groups can regularly analyze, motivating the development of automated solutions. Producing high-quality reviews remains technically challenging for existing approaches: it requires effective information extraction, identification of key methods and results, citation-based attribution, and stable operation under extremely long contexts. This work investigates the effectiveness of agent-based approaches under these conditions.

Early approaches to multi-document summarization relied on generative transformer models; however, as corpus size and context length increased, they evolved toward RAG architectures, where retrieval of relevant fragments grounds generation on fixed sources and improves factual accuracy (Lewis et al. [2020]). In practice, RAG pipelines often involve complex multi-stage processes manually designed by researchers, heavily dependent on expert intuition to define retrieval, selection, and generation steps (Zhang et al. [2024]). In parallel, reinforcement learning (RL) has been applied to review generation, with varying designs of quality signals and preference aggregation (including training from human feedback and textual edits). RLHF has shown the best results, while RLAIF serves as a scalable alternative in low-annotation settings (Ouyang et al. [2022], Lee et al. [2023]). The ideas of step-by-step reasoning further evolved into ReAct and multi-agent pipelines with cooperating writer and critic models for long-context tasks (Yao et al. [2022], Shinn et al. [2023]). Compared to rigid RAG scripts, such systems enable more flexible role distribution and coordination of intermediate reasoning. Moreover, RL-trained agent systems have proven effective in web-search tasks, suggesting the potential of similar strategies for review generation (Qi et al. [2024]).

Despite significant progress, current solutions remain limited. RAG pipelines impose fixed sequential stages and poorly model inter-article relations, yielding fragmented and weakly structured reviews as the corpus grows. Agent-based methods have mostly targeted narrow domains (e.g., tables, long narratives) and were not explored as RL-trainable systems (although they naturally support policy learning for tool and memory interaction). In long-context tasks, effective transfer and compression of working memory between iterations is critical, yet the most successful short-term memory methods have not been applied to review writing. RL approaches to summarization typically optimize proxy metrics and rarely integrate multi-agent coordination, limiting their applicability to scientific reviews. Finally, most existing systems focus on retrieval, while the main bottlenecks lie in information aggregation and verifiable attribution.

We developed an agent system tailored for scientific review generation. Its core consists of two interacting agents—a writer and a critic-coordinating text generation and verification. The writer drafts the review using specialized tools: semantic search over a fixed corpus, citation extraction, fragment rephrasing, and working memory management, which aggregates intermediate insights and maintains coherence across sections. The critic analyzes the writer's statements, verifies them against source documents, and identifies ungrounded claims using both working and episodic memory. We implemented an environment for further reinforcement learning that includes: tools for corpus interaction; a memory subsystem; a step-by-step interaction protocol with traceable actions and memory snapshots for reproducibility; multiple reward functions (evaluating coverage, factuality, and organizational coherence); and evaluation scenarios.

We introduce a new formulation of the review generation task that excludes the retrieval component, enabling isolated analysis of the agent's reasoning and aggregation abilities. We demonstrate that compact models operating in an agentic mode achieve quality comparable to much larger models on the same review tasks. The analysis of memory modules shows that their use is critical for this problem, significantly improving factual accuracy and textual coherence. Finally, we present a framework implementing the agent environment for review generation on fixed document corpora, supporting reproducible experimentation and further research.

2 Related Work

From Retrieval-Augmented Generation to Agent Systems Early works addressing similar problems used RAG (Lewis et al. [2020]), suffering from efficient aggregation mechanism. To cope with this disadvantage, solutions with modified pipelines were built (Zhang et al. [2024]). But they were mostly based on researcher intuition, static, lacking adaptive, iterative control. Agent-based approaches enable more flexible alternation. ReAct demonstrated that interleaving reasoning and action reduces hallucinations and improves QA performance (Yao et al. [2022]). Reflexion extends this idea: agents store prior decisions and verbal experiences, analyze mistakes, and adjust future actions (Shinn et al. [2023]). For long-context tasks, LongAgent and MA-RAG employ multiple worker agents and managers reading text in parts and synthesizing answers (Zhao et al. [2024], Nguyen et al. [2025]), scaling LLMs to longer inputs. However, the success of agent systems is highly known in QA tasks, not in overview generation.

Memory Mechanisms in Agent Systems Simplest agent architectures pass all information between iterations directly through the model context, leading to input growth and poor scalability in deep reasoning tasks. Scientific review writing requires many iterations of information gathering and processing under large contexts. A straightforward solution is adding a separate aggregation model (Shinn et al. [2023], Yu et al. [2025], Li et al. [2025]). More advanced solutions employ specialized memory architectures that index accumulated insights, context, and interaction history, accessible later via dedicated tools (Packer et al. [2023], Xu et al. [2025]). Recent studies introduce graph-based modifications that reduce token processing costs and improve generalization in long dialogues (Chhikara et al. [2025], Zhou et al. [2025]). In our task, interest in memory mechanisms stems from their absence in prior work on automated review generation.

Evaluation of Scientific Review Quality Assessing review quality remains difficult due to the scarcity of datasets containing reference reviews, as producing them requires substantial human effort. Until recently, the only relevant dataset was Multi-XScience, which included only Related Work sections or abstracts (Lu et al. [2020]). Only in the past year several works have addressed this via automatic and manual filtering of the S2ORC dataset (Lo et al. [2020]) (Bao et al. [2025], Su et al. [2025]). Many open challenges concern abstractive summarization evaluation. Classical metrics such as ROUGE (Lin [2004]) fail to capture semantic similarity, leading to LLM-based alternatives (Zhang et al. [2019]). For evaluating individual aspects of generated reviews without references, metrics such as FineSure (Song et al. [2024]) and G-eval (Liu et al. [2023]) were introduced, both showing high correlation with human judgments.

3 Problem statement

Data Let the dataset be $\{(\mathcal{D}_i, q_i, G_i)_{i=1}^N\}$, where $(\mathcal{D}_i = d_{i1}, \ldots, d_{im_i})$ is a closed set of source documents, G_i is written by experts gold overview and q_i is query that helps to clarify overview purpose. By construction, G_i cites only items from $\mathcal{D}_i$, no external data is used, G_i can contain claims based on info in $\mathcal{D}_i$ but not directly mentioned in it.

Task mapping The model implements a mapping

$$f_\theta : \mathcal{P}(\mathcal{D}_i, q_i) \longrightarrow \hat{S}_i \in \Sigma^*,$$

producing a single sequence $\hat{S}_i$ (the review) that contains in-text citations referring to $\mathcal{D}_i$ and is relevant to q_i. We do not hard-constrain attribution.

Agent formulation We formalize the generator as an **MDP** $(\mathcal{S}, \mathcal{A}, T, R, \gamma)$. A state $s_i \in \mathcal{S}$ contains the partial draft, tool outputs (closed-corpus retrieval over chunks of $\mathcal{D}_i$), and an agent memory m_i. The transition T advances the draft and memory given tool results. In this work, we do **zero-shot** generation and evaluate only terminal outputs (no policy learning), while the same MDP supports future RL by defining episodic rewards from the external metrics below. Also, we can easily formalize all RAG-like pipelines as similar iterative interactions with the environment.

3.1 External metrics

For the most comprehensive and comprehensive assessment, the following metrics were selected.

BERTScore Determine semantic similarity to G_i. With contextual token embeddings, recall and precision are

$$R_{\text{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} x_i^\top \hat{x}_j, \quad P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} x_i^\top \hat{x}_j, \quad F_{\text{BERT}} = \frac{2 P_{\text{BERT}} R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}$$

FineSurE FineSurE is a fine-grained LLM-based evaluator for summarization along faithfulness, completeness, and conciseness. Given source documents D and a generated summary $S = s_1, \ldots, s_N$, sentence-level fact checking yields $S_{\text{fact}} \subseteq S$ (sentences with no factual error), and

$$\text{Faithfulness}(D, S) = \frac{|S_{\text{fact}}|}{|S|}.$$

With a key-fact list $K = k_1, \ldots, k_M$ build a bipartite alignment $E = (k, s) : k \to s$ between key-facts and summary sentences; then

$$\text{Completeness}(K, S) = \frac{|\{k : (k, s) \in E\}|}{|K|}, \qquad \text{Conciseness}(K, S) = \frac{|\{s : (k, s) \in E\}|}{|S|}.$$

For sentence-level meta-evaluation it reports balanced accuracy

$$bACC = \tfrac{1}{2}(TPR + TNR).$$

G-Eval G-Eval is a rubric-based, reference-free evaluator that uses an LLM with chain-of-thought and a form-filling template. A prompt provides task introduction and evaluation criteria; the LLM generates evaluation steps and returns a discrete rating $s_i \in S$. The final continuous score is the probability-weighted expectation:

$$\text{Score} = \sum_{i=1}^{n} p(s_i), s_i.$$

Where $p(s_i)$ is probability of s_i token calculated in the last layer of the generative model.

4 Proposed solution

We model our system as a triad of large-language-model policies $\mathcal{M} = (M_W, M_C, M_M)$. The **writer** M_W is a generative policy that drafts review text, the **critic** M_C is an evaluator that judges draft segments against evidence and produces textual feedback, the **memory manager** M_M maintains persistent memory. This architecture is a combination of the Reflection ideas and the MemGPT, created for working with long contexts.

4.1 Agent roles

Writer Conditioned on the current draft and memory, M_W alternates between reasoning and action. At each step, it either issues a **retrieve call** $T_{\text{DB}}(q, k)$ to collect the top-k document chunks most similar to a query q (using a dense embedding e and cosine similarity), or it calls **rewrite** current review $S_i \to S_{i+1}$. This ReAct-style interleaving of thought and retrieval enables the writer to plan and ground its output in evidence.

Critic After each segment, M_C queries the same evidence via T_{DB} and the memory manager T_{MEM} to verify the factuality of claims. It produces a natural-language critique, highlighting unsupported statements or missing sources. Unlike a scalar reward, this verbal feedback is appended to memory and used to bias subsequent actions, similar to the self-reflection model in Reflexion.

Memory manager We adopt and simplify MemGPT's hierarchical memory architecture. Memory is partitioned into a **main context** that fits into the LLM's window, and an **external context** that stores compressed long-term experiences. The memory manager provides two operations: $\textsc{MemWrite}(s)$ adds summaries or critiques to the archive; $\textsc{MemRead}(\ell)$ returns the ℓ most relevant past entries.

4.2 Interaction protocol

The calling of agents occurs iteratively. On turn i, M_W forms a query, calls T_{DB} to fetch evidence and then writes overview changes S_{i+1}. Its output is assessed by M_C with respect to stored memory and database info; it then generates feedback pointing out errors, which is which is combined with the output of M_W and processed by M_M to expand external memory and form new input for M_W. This loop continues until a coherent, well-supported review is produced or maximum amount of steps achieved.

5 Experiments

5.1 Dataset

We evaluate our system on the SciReviewGen dataset, a large-scale benchmark for automated literature review generation built on the Semantic Scholar Open Research Corpus (S2ORC). It contains over 10,000 human-written literature reviews and roughly 690,000 cited papers across diverse scientific domains. For each review, SciReviewGen provides the abstracts of all cited papers, along with the review title and chapter titles, forming a query-focused multi-document summarization task. Also, due to chapter titles are included into model input, this task doesn't require sufficient planning. We use bibliographic metadata to build our database, define. Compared with earlier datasets such as Multi-XScience, SciReviewGen features much longer inputs and outputs and covers a broader range of topics, making it a more challenging benchmark.

5.2 Baselines

Direct prompting. A single-pass generation baseline: the model directly produces the chapter from the provided inputs with no intermediate planning, external retrieval, or revision loop.

RAG Retrieval-Augmented Generation combines a neural retriever with a sequence-to-sequence generator to ground outputs in external documents. The retriever fetches the top-k passages relevant to a query, and the generator produces a summary conditioned on these retrieved passages. This architecture improves factual accuracy and enables explicit source citation, and it has been widely adopted across open-domain QA and summarization tasks. However, vanilla RAG employs a static single retrieval step and cannot iteratively refine retrieval; as a result, it may miss key documents or propagate hallucinations when the initial retrieval is imperfect.

Query-weighted Fusion-in-Decoder QFiD extends the Fusion-in-Decoder framework by weighting each cited paper according to its relevance to the query (the review title or chapter title). It encodes each paper with a BART encoder, computes a similarity weight between the paper's hidden state and the query using inner products, and then concatenates the weighted hidden states before feeding them to a BART decoder. This weighting mechanism allows the model to focus more on papers closely related to the chapter topic. We compare our solution with QFiD because it is designed for query-focused summarization and represent strong RAG-like baseline; but its static weighting cannot dynamically adjust when new information surfaces.

ChatCite It is an LLM-based solution designed for literature summarization with human workflow guidance. It first employs a Key Element Extractor to pull out research questions, methodologies, results, and other core elements from each reference paper. A Reflective Incremental Generator then iteratively produces comparative summaries, while a reflective evaluator votes on candidate summaries to select the best result. ChatCite emphasizes comparative analysis and structural coherence, but it requires complex prompt engineering and multiple iterations; we include it as a strong baseline, which is a transition from RAG-like systems to agents.

5.3 Hypotheses

Our experiments test several hypotheses:

- **H1** (Metric sensitivity): Our multi-agent pipeline will achieve higher scores on FineSurE and G-Eval (reflecting factuality, coherence and integrity) but may underperform on BERTScore. If confirmed, this supports the argument that reference-based metrics alone (such as BERTScore) are insufficient for assessing scientific literature reviews.
- **H2** (Model efficiency): Smaller base models, when combined with our agentic pipeline and memory, can match or surpass larger RAG-based baselines in FineSurE and G-Eval, indicating that usage of tools, specialized on working with long contexts and modified feedback loops compensate for reduced model size.
- **H3** (Role of the critic): Introducing a critic module will reduce hallucinations and unsupported claims compared with single-agent baselines, leading to higher faithfulness scores in FineSurE.
- **H4** (Memory utility): Leveraging MemGPT's memory manager will improve integrity and comparative analisis by enabling long-range context tracking and reducing repetition, especially on longer reviews.

6 Results

The main baseline for our study was the QFid model, provided directly by the authors of the SciReviewGen benchmark. To verify the comparison, we also provide ROUGE-L values for all measured setups.

6.1 Comparison of prompt-based solution with baselines

Due to GPU memory and throughput constraints, we were not able to run locally our full agent architecture reliably with open-weight $\leq$10B models. To ensure a stable evaluation protocol, we therefore conduct all end-to-end experiments using *API-hosted* instruction-tuned LLMs.

To isolate the impact of model scale while minimizing confounds due to training recipes, we select two models from the *same* family (Meta Llama 3.1 Instruct):

- **API-S:** `meta-llama/llama-3.1-8b-instruct` (8B parameters; 131k context).[1]
- **API-L:** `meta-llama/llama-3.1-70b-instruct` (70B parameters; 131k context).[2]

We evaluate on the SciReviewGen test set under the same chapter-generation setting as (Kasanishi et al., 2023): input consists of the review title, a chapter title, and the texts of the cited papers. We report ROUGE and BERTScore as reference-based measures, and FineSurE and G-Eval as reference-free and LLM-based measures of fine-grained quality. Main results presented in Table 1

System	R-L	BERTScore	Faith.	Compl.	Conc.	G-Eval
QFiD (trained) [14B]	**16.52**	**0.864**	0.86	0.52	0.61	3.0
Direct (API-S) [**8B**]	14.92	0.854	0.74	0.49	0.55	3.1
RAG (API-S) [**8B**]	15.3	0.856	0.80	0.46	0.57	2.9
Agent+Mem+Critic (API-S) [**8B**]	15.17	0.855	0.84	0.55	0.61	3.6
Direct (API-L) [70B]	14.4	0.859	0.82	0.56	0.58	3.7
Agent+Mem+Critic (API-L) [70B]	15.2	0.861	**0.90**	**0.60**	**0.64**	**4.0**

Table 1: Comparision of proposed method with baselines on SciReviewGen. Columns Faith./Compl./Conc. denote FineSurE faithfulness/completeness/conciseness Song et al., 2024. G-Eval is reported on a 1–5 scale and corresponds to coherence aspect from paper (Liu et al., 2023).

Across reference-based metrics, the supervised QFiD baseline remains strongest (ROUGE-L $= 16.52$, BERTScore $= 0.864$), indicating that training on SciReviewGen optimizes for the particular phrasing and structure of the gold

[1] `https://openrouter.ai/meta-llama/llama-3.1-8b-instruct`
[2] `https://openrouter.ai/meta-llama/llama-3.1-70b-instruct`

chapters. Direct prompting with API models yields lower ROUGE-L while remaining competitive in semantic similarity (e.g., BERTScore $= 0.859$ for Direct API-L), suggesting that instruction-tuned LLMs produce plausible paraphrases that are weakly captured by lexical overlap. Adding naive RAG improves faithfulness for API-S (0.74→0.80) yet reduces completeness and overall judged quality (G-Eval 3.1→2.9), consistent with retrieval filtering that can drop evidence needed for broad coverage. Our Agent+Mem+Critic pipeline consistently improves reference-free dimensions—especially faithfulness, completeness, and conciseness—with the largest gains at 70B (FineSurE faithfulness 0.82→0.90; G-Eval 3.7→4.0). Overall, these trends support H3–H4 (critic and memory) and partially support H1 (metric sensitivity): the agentic pipeline yields substantial improvements under FineSurE/G-Eval while the same improvements are only weakly reflected by ROUGE-L.

7 Discussion: Which evaluation metrics are suitable for abstractive overviews?

Since we use a set of sophisticated evaluation metrics, we provide additional explanation about its meaning in Table 2.

We found conflicting results, as for all reference-free metrics, using our pipeline leads to significant improvements, while those based on comparison with a benchmark review degrade. In this section, we try to provide explanation of this observation.

Table 1 shows that system rankings depend strongly on the chosen metric, reflecting the fact that literature-review chapters admit multiple valid realizations. We therefore interpret reference-based similarity as a comparability tool and emphasize faithfulness/coverage/coherence diagnostics for scientific usefulness.

ROUGE and BERTScore. ROUGE-L and BERTScore rank the supervised QFiD baseline highest, suggesting that these metrics primarily reward alignment to the single gold chapter's phrasing and structure. Consequently, they are useful for continuity with prior SciReviewGen work but can understate improvements from iterative refinement when the output reorganizes or paraphrases content.

FineSurE. FineSurE separates factual grounding from content adequacy, and our agentic pipeline achieves the best faithfulness and completeness, especially at 70B (0.90/0.60 vs. 0.82/0.56 for Direct).

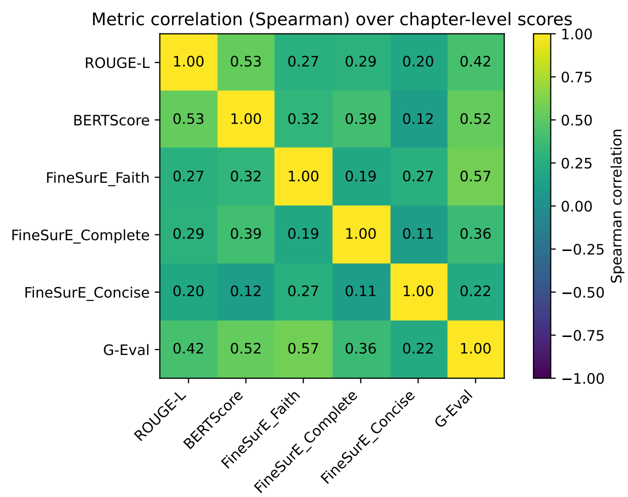

Figure 1: Metric correlation (Spearman) computed over chapter-level scores: ROUGE/BERTScore typically cluster together, while FineSurE faithfulness forms a distinct axis capturing factual grounding.

This pattern is consistent with memory enabling cross-paper aggregation and the critic suppressing unsupported claims and redundancy.

G-Eval. G-Eval assigns higher scores to the API models and favors Agent+Mem+Critic (4.0), reflecting gains in coherence and readability that are not captured by ROUGE-L. However, as an LLM-judge it may amplify stylistic preferences of the judge model, so we treat it as complementary rather than a stand-alone factuality measure.

8 Conclusion

We studied agent-based literature review generation on SciReviewGen, focusing on how to evaluate *abstractive* overview chapters in a way that reflects scientific utility rather than surface overlap. We proposed an iterative pipeline with three roles: a writer, a critic, and an explicit memory manager, designed to aggregate evidence across cited texts and revise drafts toward fewer unsupported statements and less redundancy. Using two API-hosted models from the same family (Llama 3.1 Instruct 8B and 70B), we find that the agentic pipeline yields the strongest reference-free performance: it improves FineSurE faithfulness and completeness and increases G-Eval coherence, with the largest gains at 70B. In contrast, reference-based metrics (ROUGE-L and BERTScore) still prefer the supervised QFiD baseline, highlighting

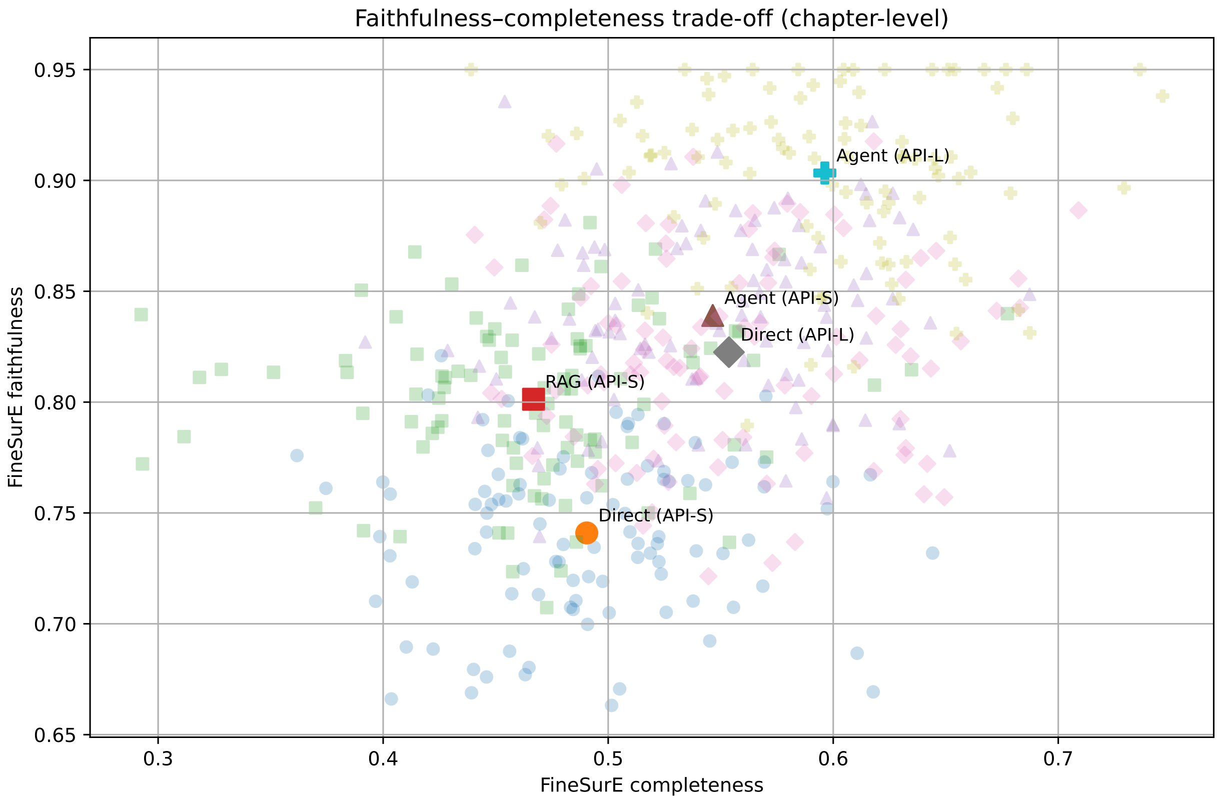

Figure 2: Faithfulness–completeness trade-off scatter plot. Systems with retrieval-only behavior often increase faithfulness at the cost of completeness; memory+critic shift the Pareto frontier outward.

a mismatch between single-reference similarity and improvements in grounding and organization. Taken together, our results show importance of multi-dimensional evaluation protocols for review generation, where faithfulness and content adequacy (e.g., FineSurE) are treated as primary signals and ROUGE/BERTScore as comparability measures. These findings motivate future fully open-weight implementations to test whether memory- and critic-driven gains persist under strict compute constraints and to improve quality with RL.

9 Limitations

- **API dependency.** Our strongest systems rely on black-box API models, which limits exact reproducibility and don't allow improving quality with any training methods.
- **Evaluation cost and variance.** FineSurE and G-Eval require LLM calls and careful prompting, introducing cost and potential variance. It's necessary to improve both reliability of such metrics and thier efficiency.
- **Citation accuracy is only partially evaluated.** While our formulation restricts generation to a closed cited set, we do not fully evaluate citation placement correctness or reference-list completeness, which is emphasized in survey-generation benchmarks.
- **Abstract-only inputs.** SciReviewGen provides abstracts of cited papers; this inherently limits fine-grained detail and can cap informativeness even for strong generators.
- **Only partial review generation.** SciReviewGen chooses only chapters of complete review as a target due to limitations of existing models. This approach simplifies the task and may hinder the realization of the potential of agent systems.

10 Future Work

- **Multi-agent Reinforcement learning** Replace or improve hand-designed critique heuristics with RL/RLAIF-style optimization over multi-dimensional rewards (faithfulness, completeness, conciseness, and citation correctness), building on evidence that RL-based alignment can improve long-form generation.
- **Explicit citation metrics.** It's worth considering a modification of the problem that allows for a complete search across a large database rather than constructing a review based on a strictly defined set of documents. This approach allows for a broader range of problems to be solved, but it does not allow for a separate examination of the models' ability to construct complete, structured, and coherent reviews.
- **Multi-reference evaluation.** Extend beyond single-reference ROUGE/BERTScore by using multiple references (when available).
- **Human studies at scale.** Conduct expert evaluations on informativeness and factuality to validate that improvements in FineSurE faithfulness translate into real reviewer preference in scientific settings.

References

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

Yusen Zhang, Ruoxi Sun, Yanfei Chen, Tomas Pfister, Rui Zhang, and Sercan Arik. Chain of agents: Large language models collaborating on long-context tasks. *Advances in Neural Information Processing Systems*, 37:132208–132237, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023.

Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Wenyi Zhao, Yu Yang, Xinyue Yang, Jiadai Sun, Shuntian Yao, et al. Webrl: Training llm web agents via self-evolving online curriculum reinforcement learning. *arXiv preprint arXiv:2411.02337*, 2024.

Jun Zhao, Can Zu, Hao Xu, Yi Lu, Wei He, Yiwen Ding, Tao Gui, Qi Zhang, and Xuanjing Huang. Longagent: scaling language models to 128k context through multi-agent collaboration. *arXiv preprint arXiv:2402.11550*, 2024.

Thang Nguyen, Peter Chin, and Yu-Wing Tai. Ma-rag: Multi-agent retrieval-augmented generation via collaborative chain-of-thought reasoning. *arXiv preprint arXiv:2505.20096*, 2025.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, et al. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*, 2025.

Tongyi DeepResearch Team Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, Kuan Li, Liangcai Su, Litu Ou, Liwen Zhang, Pengjun Xie, Rui Ye, Wenbiao Yin, Xinmiao Yu, Xinyu Wang, Xixi Wu, Xuanzhong Chen, Yida Zhao, Zhen Zhang, Zhengwei Tao, Zhongwang Zhang, Zile Qiao, Chenxi Wang, Donglei Yu, Gang Fu, Haiyang Shen, Jiayi Yang, Jun Lin, Junkai Zhang, Kuijie Zeng, Li Yang, Hailong Yin, Maojia Song, Ming Yan, Peng Xia, Qian Xiao, Rui Min, Rui Ding, Runnan Fang, Shaowei Chen, Shen Huang, Shihang Wang, Shihao Cai, Weizhou Shen, Xiaobin Wang, Xin Guan, Xinyu Geng, Yi Shi, Yuning Wu, Zhuo Chen, Zijian Li, and Yong Jiang. Tongyi deepresearch technical report. 2025. URL `https://api.semanticscholar.org/CorpusID:282400924`.

Charles Packer, Vivian Fang, Shishir_G Patil, Kevin Lin, Sarah Wooders, and Joseph_E Gonzalez. Memgpt: Towards llms as operating systems. 2023.

Wujiang Xu, Kai Mei, Hang Gao, Juntao Tan, Zujie Liang, and Yongfeng Zhang. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*, 2025.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*, 2025.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Jinhua Zhao, Bryan Kian Hsiang Low, and Paul Pu Liang. Mem1: Learning to synergize memory and reasoning for efficient long-horizon agents. *arXiv preprint arXiv:2506.15841*, 2025.

Yao Lu, Yue Dong, and Laurent Charlin. Multi-XScience: A large-scale dataset for extreme multi-document summarization of scientific articles. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8068–8074, Online, November 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.648. URL `https://aclanthology.org/2020.emnlp-main.648/`.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. S2ORC: The semantic scholar open research corpus. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online, July 2020. Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.447. URL `https://aclanthology.org/2020.acl-main.447/`.

Tong Bao, Mir Tafseer Nayeem, Davood Rafiei, and Chengzhi Zhang. Surveygen: Quality-aware scientific survey generation with large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Suzhou, China, 2025.

Weihang Su, Anzhe Xie, Qingyao Ai, Jianming Long, Jiaxin Mao, Ziyi Ye, and Yiqun Liu. Surge: A benchmark and evaluation framework for scientific survey generation. *arXiv preprint arXiv:2508.15658*, 2025.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-1013/`.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

Hwanjun Song, Hang Su, Igor Shalyminov, Jason Cai, and Saab Mansour. Finesure: Fine-grained summarization evaluation using llms. *arXiv preprint arXiv:2407.00908*, 2024.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using gpt-4 with better human alignment. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore, December 2023. Association for Computational Linguistics. doi:10.18653/v1/2023.emnlp-main.153. URL `https://aclanthology.org/2023.emnlp-main.153/`.

Tetsu Kasanishi, Masaru Isonuma, Junichiro Mori, and Ichiro Sakata. SciReviewGen: A large-scale dataset for automatic literature review generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6695–6715, Toronto, Canada, July 2023. Association for Computational Linguistics. doi:10.18653/v1/2023.findings-acl.418. URL `https://aclanthology.org/2023.findings-acl.418/`.

A Metrics explanation

Metric	Range	Practical interpretation on abstractive overviews
ROUGE-1	0–100	Captures lexical overlap; useful mainly for compatibility with prior work, but penalizes legitimate re-structuring and paraphrase.
ROUGE-2	0–100	Sensitive to short phrase copying; can be inflated by semi-extractive generation, even when reviews lack insight.
ROUGE-L	0–100	Tracks longest common subsequence; mildly reflects discourse similarity but remains reference-style dependent.
BERTScore (F1)	0.–1.	Measures semantic similarity to a single reference.
FineSurE Faithfulness	0.–1.	Primary indicator of factual grounding; values ≥ 0.85 usually correspond to low hallucination rates in our manual spot checks.
FineSurE Completeness	0.–1.	Proxy for coverage of key facts.
FineSurE Conciseness	0.–1.	Penalizes redundancy and generic filler.
G-Eval	1–5	Measure general text quality (structure and coherence), but can be biased toward the judge model's preferences.

Table 2: Illustrative metric ranges and interpretation on SciReviewGen-like abstractive overview generation. FineSurE is designed to correlate with human judgments on faithfulness and completeness/conciseness.