Non-Archival Abstract

Designing Reliable RAG Systems for Large-Scale Engineering Environments

Ekaterina Voronina
Institut für Maschinelle Sprachverarbeitung, University of Stuttgart
European XFEL, Hamburg
st194336@stud.uni-stuttgart.de

Documents that describe hardware requirements in large-scale engineering facilities are dense with the very details that constitute their substance: component names, model numbers, and numeric specifications scattered across free text, tables, and schemas. Hybrid lexical-semantic retrieval, which is a classic RAG-pipeline component that fuses sparse matching such as BM25 with dense retrieval, and has been shown to outperform either method alone (Luan et al., 2021), still ranks these details as ordinary passages by relevance. Such a mismatch of hybrid retrieval degrades exactly the exact-lookup, comparison, and range queries engineers actually ask.

Recent surveys name entity-aware pre-filtering as a promising but largely unimplemented remedy (Wang et al., 2024; Huang and Huang, 2026). We argue that a structured and typed entity index resolves this cheaply within an Agentic RAG system. Unlike a fixed retrieval pipeline, an agent behind Agentic RAG decides at each step which tool to call and when to stop searching, and when facts are extracted once at indexing time, the agent can resolve a query by direct lookup instead of repeated LLM-mediated reasoning over retrieved passages. This results in reducing both latency and token cost.

This agentic flexibility introduces a second, understudied variable (Sabherwal, 2026): the natural-language instruction layer (e.g., an AGENTS.md-style system-prompt section). This type of files are meant to govern tool priority, whose effect on actual tool selection and retrieval recall is not yet established.

We present an Agentic RAG system (see Figure 1) deployed for hardware documentation at a large-scale scientific facility and isolate such variables as 1) entity-index presence and 2) instruction-layer presence via a 2×2 factorial ablation with the generator, corpus, and judge model as constants.

We measure retrieval coverage and answer quality with deterministic set metrics and DeepEval's (Ip and Vongthongsri, 2026) contextual and faithfulness metrics, and operational cost (latency, tool calls, tokens) via trace logging, over a benchmark of expert-validated questions sourced from expert sessions, LLM generation, and production logs.

We expect the results to reveal one of two patterns. The entity layer either enables queries hybrid search cannot express, or it improves cost and reliability on queries hybrid search already handles.

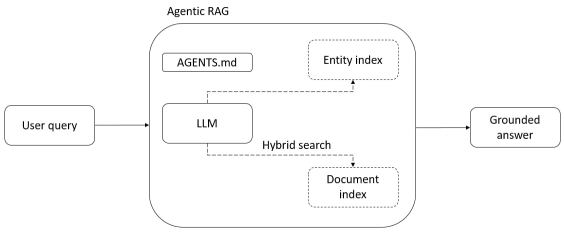

Figure 1: Proposed Agentic RAG system. At query time, the agent selects between two tool servers: one performs SQL-based filtering over the structured entity index, the other performs hybrid lexical-semantic search. Both indexes are built offline from the source documents.

References

Yizheng Huang and Jimmy Xiangji Huang. 2026. A survey on retrieval-augmented text generation for large language models. *ACM Computing Surveys*, 58(12).

Jeffrey Ip and Kritin Vongthongsri. 2026. DeepEval: The open-source LLM evaluation framework. https://github.com/confident-ai/deepeval. Version 4.1.5.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.

Arpit Sabherwal. 2026. The instruction layer: Markdown files as architectural artifacts in AI-assisted

software development. *International Research Journal of Modernization in Engineering Technology and Science*, 08(05).

Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, Ruicheng Yin, Changze Lv, Xiaoqing Zheng, and Xuanjing Huang. 2024. Searching for best practices in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17716–17736.