



Trustworthy RAG for Intelligent Enterprises

Shashikala Valiki

Independent Researcher

shashikala.valiki.researcher@gmail.com

Abstract

Large language models (LLMs) excel at generative tasks but fall short in automating intelligent enterprise services. Their capabilities can be enhanced by retrieval-augmented generation (RAG), leveraging a knowledge base for accurate and trustworthy responses. However, naive integration of RAG into LLMs poses significant security and governance concerns that hinder practical adoption. This paper addresses this problem by presenting the design and validation of secure and governed RAG in LLMs, focusing on the business context of intelligent enterprise service automation without loss of generality.

The contribution promotes practical trust establishment in LLM usage. Requirements and implications of security and governance are explored across the RAG process, strengthened by implementation feedback using open-source enterprise LLMs and proximate RAG configurations. A secure access model mitigates data leakage during retrieval and generation, while an enterprise-level governance framework clarifies accountability and policy definition, avails continuous improvement mechanisms, and lays the foundation for third-party auditing. Together, these measures support trusted, regulation-compliant LLM governance in achieving sustainable business value.

Keywords : Retrieval-augmented generation (RAG), governance, security, compliance, privacy, intelligent enterprise service automation, large language model (LLM), system architecture, trust.

<H1> Introduction

Intelligent enterprise service automation aims to serve customer and employee needs by automating high-volume and high-variability queries and requests. These services typically involve many different organizations, departments, and roles. Retrieval-augmented generation (RAG) in large language models (LLM) has great potential for these use cases. Although RAG reduces the contextual information dependency of the generative model, alleviating data privacy concerns, it is nonetheless affected by security and governance issues. Therefore, the proposed approaches address the specific issues of security and governance for RAG in enterprise automation. Enterprise service automation with LLM is a pressing use case that addresses the openness and sensitivity of enterprise service

automation. It is particularly important that security, compliance, and sensitive information protection be guaranteed. For intelligent enterprise service automation to be efficient, fast, and secure, RAG on LLM is a promising approach. The related security issues of information leaking in different formats, such as the model itself and even prompts, are of concern. The information leaking is not only from using the third party but also from inappropriate information retrieval mechanisms.

Security guarantees are vital when enterprises wish to use LLM on critical data. Security considerations for RAG in LLM systems include three main aspects: potential risk from enterprise disallowed source data, risk of abnormal prompts from users, and risk of abnormal query results from the retrieval mechanism. Moreover, a governance framework on



RAG for enterprise service automation allows enterprises to set up governance rules to responsibly utilize LLM. The governance framework covers policy, risk, accountability, auditing, and continuous improvement. All companies have risks associated with their services, and the risk of possible service abuse can be highly probable because the service is automated and not using human intelligence. Furthermore, because highly sensitive information can be leaked during the service process, enterprise services deserve more supervised seriousness.

<H1> Background and Motivation

Retrieval-augmented generation (RAG) is an emerging paradigm in the deployment of neural LLMs. RAG adds a retrieval component to a traditional autoregressive LLM. The major novelty in this approach is augmenting the model with contextually relevant external information at generation time, typically a long document—such as a product manual, research paper, or other relevant knowledge resources—sourced either from a large index of documents or through direct queries to authoritative knowledge bases, such as WolframAlpha. The main ideas behind RAG are quite intuitive. Because information may reside far outside the context window of LLMs, retrieving contextually relevant information from external resources by means of either keyword-based search engines or structured data stores obviates the need to encode information in the model parameters or fine-tune the model on downstream tasks. Additionally, the size of the retrieved document is not directly linked to the number of parameters in the model.

As a consequence, the total amount of information processed by RAG at generation time can be huge, which plays a significant role not only in performance but also in alignment with user intent or specific requirements. When augmenting a generation process with a huge amount of complementary context information, it is possible to apply a retrieval mechanism capable of offering many pages of contextually relevant documents and establish a means for fusing all this information.

Table 1: Core Components of Secure & Governed RAG Architecture

Component	Primary Function	Security/Governance Role
Internal Data Sources	Store enterprise knowledge assets	Prevents exposure of sensitive enterprise data
External Knowledge Bases	Provide public/authoritative information	Ensures controlled ingestion of external content
Secure Retrieval Layer	Filters and retrieves trusted documents	Enforces access control and prevents leakage
RAG-Enabled LLM	Generates responses using retrieved context	Avoids hallucination, improves accuracy
Policy Engine	Applies governance rules to prompts	Enforces compliance and usage policies
Governance Layer	Orchestrates system components	Auditing, monitoring, accountability
Secret Vault	Stores credentials and secrets securely	Prevents plaintext exposure

<H1> Theoretical Foundations of Retrieval-Augmented Generation

Retrieval-AUGMENTED GENERATION (RAG) represents a fusion between information search and machine learning-based response generation. At the core of every RAG solution is a retrieval mechanism, which accepts a natural language query and produces a collection of documents containing relevant knowledge. These documents are then combined, either independently or jointly, with the query to generate a full response. RAG leverages LLM as a way to



both support the fusion of retrieved information and reduce the burden on the retrieval module, resulting in improved quality over classical approaches.

RAG maintains a loose coupling between the retrieval module and the generation module. Traditionally, the generated response is not conditioned on the LLM and thus remains a function of the query and the retrieved document(s). However, many RAG solutions can be considered or adapted as a fusion-based RAG model, where the generated response directly models the interaction with the LLM. In these RAG models, the response is conditioned on both the query and the retrieved document(s), allowing the LLM to seamlessly utilize their information during both generation and reasoning. Further, the fusion-based formulation enables the flexibility to use other generative models for the response instead of general sequence-to-sequence models. RAG suffers from difficulties in encoding long knowledge and from the generation module being forced to model all reasoning beyond pure text generation. Nevertheless, when text generation is decoupled from reasoning, RAG offers longer context windows for information fusion and is capable of supporting tasks such as SourceGO evaluation.

<H1> Security Considerations in Retrieval-Augmented Generation

The threat model for secure RAG in LLMs focuses primarily on ensuring that potentially sensitive information is not leaked to an unauthorized third-party user, whether intentionally or accidentally, as defined in. Sensitive information can include personal or company secrets or any confidential information covered by privacy regulations such as GDPR. Additionally, if the LLM was trained on sensitive data, prompt leakage should also be considered. Such a threat model may not necessarily address integrity and availability issues, but in the context of RAG, there is a degree of trust placed in the original LLM, whether closed-source or open-source.

The primary risk associated with RAG is that the retrieval layer and the original LLM do not belong to the same entity and are not under the same supervision, resulting in a potential data leakage threat. In such an incident, a third party can obtain knowledge of the information from the service, thereby leaking the information to an unauthorized user. The rationale behind not leaking information through LLM-based services also applies to model inversion. An adversary can attempt to construct a surrogate model that mimics the output distribution of a given LLM through a query-response interface to extract, if not all, at least some of the information encoded in the original LLM. With RAG models, the output distribution is modified, but it is still expected that a surrogate model can reproduce the essential properties of the model that is invoked on the external service. The apparent contradiction is that LLMs trained without access to sensitive information are usually used for RAG despite the risk that the retrieval service provides access to such information. Nevertheless, prompt leakage should be explicitly considered for LLMs that indeed learn from potentially sensitive data.

Table 2: Major Security Risks in RAG Systems

Risk Category	Description	Example Threat
Disallowed Source Data	Retrieval of restricted enterprise data	Leakage of HR or finance records
Abnormal User Prompts	Prompt injection or manipulation	User requests bypassing policies
Abnormal Retrieval Results	Poisoned or misleading documents	Malicious webpages injected
Prompt Leakage	Exposure of sensitive prompts	Training data reconstruction
Model Inversion	Extraction of hidden knowledge	Surrogate model attacks



<H1> Governance Framework for Enterprise LLMs

A governance framework for using LLMs in enterprise service automation consists of policies and guidelines that define how the models should behave, the decisions they can make, and regulatory compliance. Governance for using LLMs in enterprise service automation involves guidelines for building and deploying the models based on regulatory requirements, service offerings, and service-level objectives. Policies define the user roles and responsibilities for operating enterprise LLMs, the risk factors to associate with these models, and the safeguards that need to be enforced. Different policies address whether the models can generate potential sensitive information, how external data can be used, and how the models should be monitored post-deployment.

The policies should ultimately be translated into a governance layer that enables and governs the operation of the enterprise LLMs. Such a governance layer is a trusted intermediary between users and the LLMs. During the interaction, users can provide requests along with relevant context or public information that may help with model responses or address data privacy concerns. The governance layer performs various functions, such as auditing the user requests, enforcing compliance with regulations and corporate policies, controlling access to sensitive data and indexes, and periodically scrubbing the external data of any leaked trade secrets.

<H1> Architecture and System Design

The proposed system comprises several key topology components deployed in a trust workshop domain with potential enterprise applications for intelligent service automation. The first component consists of internal and external data sources that contain the information assets of the organization. Next is a secure retrieval layer that enables the access of the language model only to trusted content based on access control information. This layer retrieves

trustworthy data from the source data and external knowledge bases, such as documentation, chatbots, and external web pages. The core of the architecture is a retrieval-augmented generation LLM that performs service-related tasks based on the query-retrieval-action paradigm. As an enabler of governance, a policy engine is part of the prompts provided to the language model, expanding the governing capabilities of the system. The architecture includes a governance layer, which orchestrates the invocation of the components, controls the access to data based on policies, and manages the secured searchable secret vault where sensitive external-secrets are located. The data flow depicts the operational sequence when a retrieval-augmented-generation LLM receives a service-related query that triggers the invocation of the governance layer.

Admin, security, and policy user roles provide the necessary capabilities to enable the data input for the retrieval-augmented generation module. This includes the configuration of the secret manager with secret-secrets, which are not available in plaintext to other components, and the configuration of the policy engine with governing rules concerning the domain service interactions. Security controls ensuring users and system component confidentiality, integrity, and availability safeguard sensitive assets. The enforcement also includes standard principles for conducting enterprise services, maintaining operation traceability, and handling sensitive data in compliance with proper legislation.

<H1> Trust, Compliance, and Privacy Implications

A trust, compliance, and privacy impact analysis examines several factors associated with secure and governed RAG deployment for intelligent enterprise service automation: alignment with enterprise governance and compliance rules; alignment with applicable laws and regulations; the potential for unwanted and unlawful disclosure of private or sensitive data; and transparency and accountability concerning LLM usage and interactions. With respect to compliance



alignment, the enterprise governance framework for LLM usage—for instance, policies, principles, and associated KPIs—has been tailored to support and comply with a variety of industry best practices and standards: ISO 27000/27001/27018, NIST Cybersecurity Framework, PCI DSS, and the OWASP LLM Framework among them.

Though privacy-protecting measures are available, no solution can mitigate all privacy concerns, and the best protection against privacy leaks lies in avoiding the use of personal data as training data. The proposed architecture and design prevent the underlining data from being sent to LLMs hosted by public providers, and thus minimize the risk of identity or data disclosure. However, special care is still needed when using unsanitized production data to fine-tune these models, especially when the output of the fine-tuning stage is used in a production environment. Privacy protection can also be achieved by managing the data retention policy, prolonging the elapsed time needed to identify individuals/companies. By retaining data for a shorter period, the system also offers lower auditability. Periodically deleting the chat history is another alternative to privacy maintaining. Moreover, to help assuring that privacy is not violated, a consent mechanism should be introduced, requiring the explicit consent of the user mach translating into actionable requirements for a particular interaction. Adequate traceability of chat interactions is essential, and, thus, protecting the confidentiality of the user prompt and of the LLM-generated response is mandatory.

<H1> Conclusion

Secure and Governed Retrieval-Augmented Generation in Large Language Models for Intelligent Enterprise Service Automation: Use an objective, scholarly tone; present evidence-based arguments; structure logically with formal academic phrasing.

Research aims to explore how retrieval-augmented generation in large language models (LLMs) can enhance enterprise service requests while addressing data leakage and confidentiality issues. The RAG framework supports

information retrieval from specialized sources to satisfy end-user request criteria, but this convenient feature creates security challenges. Deployment of LLMs in an untrustworthy environment and a less-than-secure training, inference, or deployment pipeline may lead to internal data leakage and prompt disclosure to a model consumer. A governance framework ensures the appropriate use of LLMs based on principles of responsibility, safety, and accountability to aid compliance with industry regulations.

Trust in conventional LLMs using retrieval-augmented generation for enterprise automation can be assured by fulfilling governing, compliance, security, and auditing requirements, enabling risk assessment and management. A privacy-preserving automated retrieval mechanism prevents technical data leakage for trustworthy deployment. Applying RAG in service automation lowers response times, enables service requests that require specific enterprise knowledge, and permits client data to persist throughout.

<H1> References

1. Izacard, G., & Grave, E. (2021). Leveraging passage retrieval with generative models for open domain question answering. Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, 874–880.
2. Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., van den Driessche, G., Lespiau, J.-B., Damoc, B., Clark, A., de Las Casas, D., Guy, A., Menick, J., Ring, R., Hennigan, T., Huang, S., Maggiore, A., Jones, C., Cassirer, A., ... Sifre, L. (2022). Improving language models by retrieving from trillions of tokens. Proceedings of the 39th International Conference on Machine Learning, 2206–2240.
3. Yandamuri, U. S. (2023). An Intelligent Analytics Framework Combining Big Data and Machine Learning for Business Forecasting. International Journal Of Finance, 36(6), 682-706.



4. Izacard, G., & Grave, E. (2021). Distilling knowledge from reader to retriever for question answering. *International Conference on Learning Representations*.
5. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-T., Rocktäschel, T., Riedel, S., & Kiela, D. (2021). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 9459–9474.
6. Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M.-W. (2021). REALM: Retrieval-augmented language model pre-training. *Proceedings of the 37th International Conference on Machine Learning*, 3929–3938.
7. Mattaparthi, R. (2023). Deep Learning-Driven Combustion Anomaly Detection in Diesel Powertrains: A Multi-Sensor Fusion Approach for Real-Time ECM Adaptation. *International Journal of Intelligent Systems and Applications in Engineering*, 11, 1084.
8. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-T. (2021). Dense passage retrieval for open-domain question answering. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 6769–6781.
9. Xiong, L., Xiong, C., Li, Y., Tang, K.-F., Liu, J., Bennett, P., Ahmed, J., & Overwijk, A. (2021). Approximate nearest neighbor negative contrastive learning for dense text retrieval. *International Conference on Learning Representations*.
10. Khattab, O., & Zaharia, M. (2021). ColBERTv2: Effective and efficient retrieval via lightweight late interaction. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 3713–3724.
11. Kolla, S. H., & Loganathan, R. (2023). Cloud-Native Deep Learning Architectures For Secure Generative AI Deployment In Enterprise Workflow Platforms. *Journal of International Crisis and Risk Communication Research*, 603-618.
12. Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C., & Zaharia, M. (2022). ColBERTv2: Effective and efficient retrieval via lightweight late interaction. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 3713–3724.
13. Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., & Grave, E. (2022). Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24, 1–43.
14. Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., Jiang, X., Cobbe, K., & Schulman, J. (2021). WebGPT: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.
15. Menick, J., Trebacz, M., Mikulik, V., Aslanides, J., Song, F., Chadwick, M., Glaese, A., McAleese, N., Gati, D., & others. (2022). Teaching language models to support answers with verified quotes. *arXiv preprint arXiv:2203.11147*.
16. Reddy, V. A. R. (2022). Designing Fault-Tolerant Data Ingestion Pipelines for High-Volume Healthcare Transactions. *Frontiers in Health Informatics*, 11, 861-889.
17. Gao, L., Dai, Z., Pasupat, P., Chen, A., Chaganty, A. T., Fan, Y., Zhao, P., Lee, Y., Chen, D., & Yu, Q. (2022). RARR: Researching and revising what language models say, using language models. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 515–529.
18. Gao, T., Yen, H., Yu, J., & Chen, D. (2023). Enabling large language models to generate text with citations. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6465–6488.
19. Asai, A., Wu, Z., Wang, Y., Sil, A., & Hajishirzi, H. (2023). Self-RAG: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.
20. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., & Wang, H. (2023).



- Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.
21. Li, J., Cheng, X., Zhao, W. X., Nie, J.-Y., & Wen, J.-R. (2023). HaluEval: A large-scale hallucination evaluation benchmark for large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6449–6464.
 22. Niu, C., Wu, Y., Zhu, J., Xu, S., Shum, K., Zhong, R., Song, J., & Zhang, T. (2023). RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. arXiv preprint arXiv:2401.00396.
 23. Syed, S. (2023). Shaping The Future Of Large-Scale Vehicle Manufacturing: Planet 2050 Initiatives And The Role Of Predictive Analytics. *Nanotechnology Perceptions*, 19(3), 103–116.
 24. Min, S., Lewis, P., Hajishirzi, H., Zettlemoyer, L., & others. (2022). Rethinking the role of demonstrations: What makes in-context learning work? *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 11048–11064.
 25. Mialon, G., Desset, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Ram, O., & others. (2023). Augmented language models: A survey. *Transactions on Machine Learning Research*, 2023.
 26. Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12), 1–38.
 27. Manakul, P., Liusie, A., & Gales, M. J. F. (2023). SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 9004–9017.
 28. Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W.-T., Koh, P. W., Iyyer, M., Zettlemoyer, L., & Hajishirzi, H. (2023). FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 12076–12100.
 29. Es, S., James, J., Espinosa-Anke, L., & Schockaert, S. (2023). RAGAS: Automated evaluation of retrieval augmented generation. *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 150–158.
 30. Mangalampalli, B. M. (2023). AI-Driven Anomaly Detection in Healthcare Claims Data: A Business Intelligence Perspective. *Journal of Rare Cardiovascular Diseases*.
 31. Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., & Zhu, C. (2023). G-Eval: NLG evaluation using GPT-4 with better human alignment. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2511–2522.