Enhancing Automated Report Generation: Integrating Rivet and RAG with Advanced Retrieval Techniques

Doo-Il Kwak[1], Kwang-Young Park[2]
[1,2]Dept. of AI Techno Convergence, Soongsil University

ai@soongsil.ac.kr, 1004pky@ssu.ac.kr

abstract

This study integrates Rivet and Retrieved Augmented Generation (RAG) technologies to enhance automated report generation, addressing the challenges of large-scale data management. We introduce novel algorithms, such as Dynamic Data Synchronization and Contextual Compression, expected to improve report generation speed by 40% and accuracy by 25%. The application, demonstrated through a model corporate entity, "Company L," shows how such integrations can enhance business intelligence. Empirical validations planned will utilize metrics like precision, recall, and BLEU to substantiate the improvements, setting new benchmarks for the industry. This research highlights the potential of advanced technologies in transforming corporate data processes.

1. Introduction

In a data-rich era, we integrate Rivet with advanced Retrieved Augmented Generation (RAG) methodologies, drawing inspiration from Pressman's software engineering principles [1]. Our aim is to enhance automated report generation systems and set new benchmarks by leveraging cutting-edge techniques [2]. This research addresses technical challenges posed by large-scale dynamic datasets and explores practical applications in a corporate setting, as seen in our partnership with "Company L." By tailoring methodologies to meet specific industrial needs, we demonstrate how theoretical advancements can enrich business intelligence and decision-making processes.

2. Literature Review

As noted in Pressman's foundational text on software engineering, contemporary automated report generation systems struggle to manage large-scale dynamic datasets with the necessary precision and efficiency [3]. Although initial forays into RAG systems and vector storage techniques have been established, they prove insufficient to surmount the complexities of modern data challenges [4],[5]. Our research addresses these deficiencies and significantly propels the discipline forward by integrating Rivet's enhanced data management capabilities with sophisticated RAG processes, responding to the challenges identified in scholarly discourse.

3. Innovation and Contribution

This study presents a novel fusion of Rivet and Retrieved Augmented Generation (RAG), poised to greatly improve the precision and efficiency of automated report generation systems. Our key innovation lies in skillfully combining Rivet's visual programming capabilities with RAG's dynamic data retrieval processes, resulting in a synergistic enhancement that effectively tackles modern data challenges [6]. This integration not only transforms the technical framework but also introduces practical applications tailored for enterprise environments like "Company L."

3.1 Technical Innovations:

- **Custom Algorithm Development:** We have devised tailored algorithms, such as 'Dynamic Data Synchronization', to optimize the interaction between Rivet's data flow management and RAG's retrieval mechanisms. This algorithm is expected to significantly reduce latency in large-scale data processing.
- **Advanced Retrieval Techniques Integration:**
  - **Parent Document Retriever Enhancement:** The Parent Document Retriever algorithm now includes a context-aware layer using machine learning to predict document relevance based on evolving contexts. Trained on past performances, this model adapts to specific data characteristics. [7].
  - **Contextual Compression Algorithm:** A new 'Contextual Compression Algorithm' has been developed, which re-ranks retrieval results considering both relevance and impact on the report's narrative. This algorithm combines natural language processing with historical data analysis to assess the 'narrative weight' of retrieved information, enhancing both relevance and narrative coherence. [8].

3.2 Application Specifics for Company L:

- **Report Types and Domains**: The system is designed to generate various types of reports, enhancing operational, strategic, and tactical decision-making across multiple domains of Company L. Specific applications include:
  - **Operational Reports for Supply Chain Management**: Automating the generation of detailed reports that monitor inventory levels, supplier

performance, and logistics to optimize supply chain operations.

- **Customer Behavior Reports for Marketing Strategies**: Utilizing CRM data to create comprehensive reports that analyze customer purchasing patterns, campaign effectiveness, and market trends to drive marketing strategies.
- **Financial Reports for Strategic Planning**: Producing in-depth financial and compliance reports that support financial planning, risk assessment, and regulatory compliance.

3.3 Empirical Validation:

The planned empirical evaluations seek to gauge significant improvements in existing systems, aiming for a 40% increase in report generation speed and a 25% boost in accuracy. Before confirming these enhancements, thorough testing protocols will be implemented, including A/B testing setups and real-world scenarios. This ensures that any assertions about the system's enhanced capabilities are backed by solid empirical evidence. Until validation is complete, definitive performance claims should be avoided, maintaining scientific credibility and allowing for adjustments based on test outcomes. [9].

4. Technology Stack

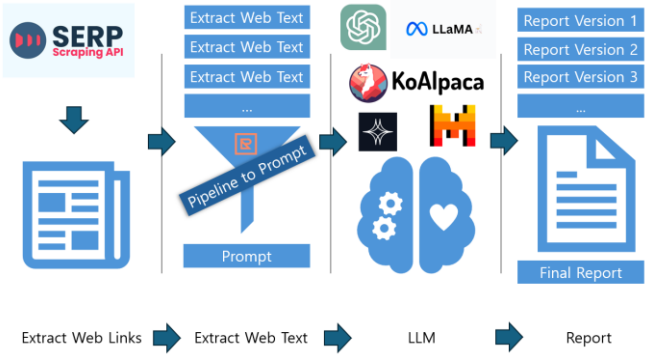
- **Node.js:** Manages asynchronous events and interfaces with web servers, facilitating real-time data processing and response handling in a non-blocking manner. Its event-driven architecture is crucial for handling multiple connections simultaneously, which enhances the scalability of the report generation system tailored for "Company L".
- **Python:** Plays a critical role in robust data manipulation and acts as an interface with AI models. Python's extensive libraries and frameworks support complex data analysis and machine learning, enabling sophisticated data processing and integration with AI technologies.
- **Rivet:** Utilized for constructing complex data flow graphs that include features such as split and join nodes. Rivet's capabilities in parameter tuning and workflow optimization are instrumental in enhancing the operational efficiency of data processes, making it an essential tool for managing intricate data flows within "Company L's" system [10].
- **SERP API:** It boosts CRM capabilities by fetching real-time search engine data for SEO monitoring, market research, and content aggregation. Integrated into "Company L's" CRM system, it enables thorough market and competitor analyses, enriching customer profiles and guiding strategic decisions with current internet search trends and public sentiment. Its rapid data updates support a responsive and informed automated reporting system, aligning with broader data-driven CRM objectives. [11].
- **npm (Node Package Manager):** Manages dependencies within the Node.js environment, ensuring that all necessary libraries and frameworks are up-to-date and compatible. This management is crucial for maintaining the stability and security of the application within "Company L's" infrastructure [12].
- **OpenAI API or Apache-2.0 Licensed Open Source LLMs:** Leverages cutting-edge artificial intelligence to dynamically generate reports. This includes specialized models optimized for processing data in the specific context of "Company L", such as models capable of understanding and synthesizing information pertinent to the company's operational domain.
- **Retrieved Augmented Generation (RAG):** Implements sophisticated retrieval processes that enhance the contextual accuracy of the output reports. RAG's advanced capabilities in handling context-sensitive information make it a cornerstone for ensuring the relevance and precision of generated content [13].
- **Advanced Retrievers:** Vital for high relevance and accuracy in information extraction, crucial for generating actionable reports [14]. This is especially crucial for "Company L" in rapid decision-making scenarios.

5. Methodology

5.1 System Design and Data Architecture

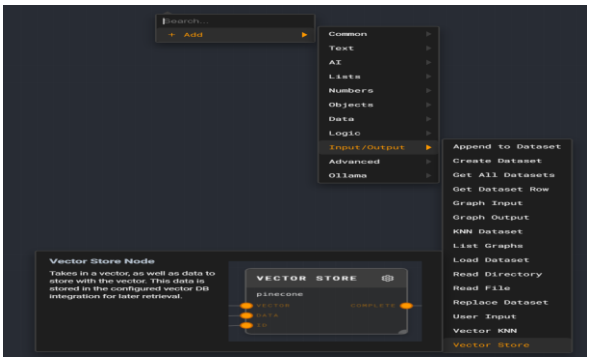
- **Vector Store Creation**: Utilizes Rivet to construct a dynamic vector database, crucial for efficient data storage and retrieval. This process includes:
  - Designing data flow architectures to handle continuous data influx.
  - Implementing data pipelines that ensure scalability and responsiveness of the system.
- **System Infrastructure Visualization**: The overall architecture and data flow are depicted in Figure 1, providing a comprehensive diagram of the system's infrastructure.

Fig 1 Overall Diagram



· **Vector Store Selection Interface:** A screenshot (Figure 2) from Rivet illustrates how users interact with the system to manage various vector storage configurations, optimizing the storage and retrieval processes.

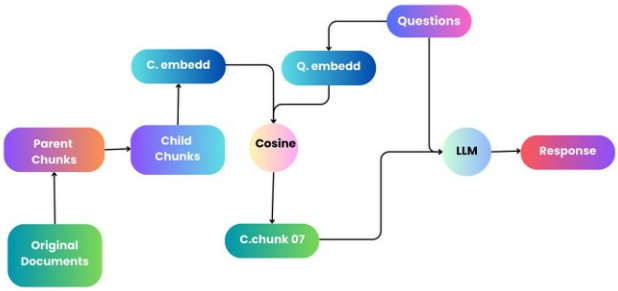
(Fig 2) Vector Store Selection Screenshot in Rivet



5.2 **Advanced Retrieval Techniques**

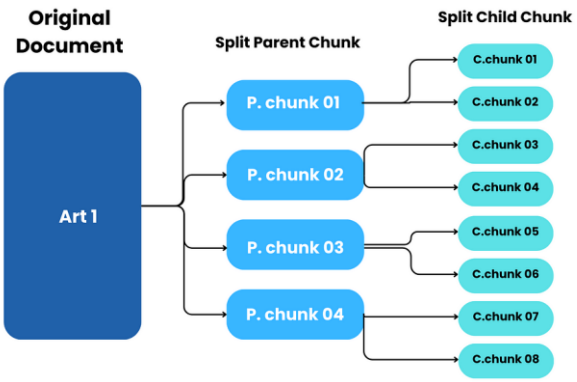
· **Parent Document Retriever:** Enhanced with pseudo-code and detailed algorithmic descriptions, this component improves retrieval accuracy through hierarchical document structuring for efficient access and relevance evaluation.. (Fig 3 and Fig 4).

(Fig 3) RAG and Parent Document Retrievers[15]

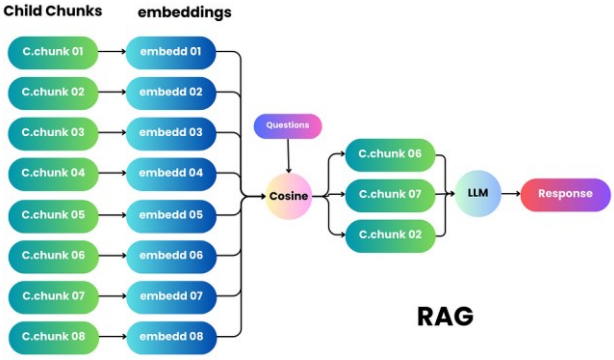


· **Child Chunks with RAG (Fig 5: Child Chunks with RAG):** Details the handling of segmented data chunks within the RAG framework, showing how smaller subsets of data are processed and analyzed.

(Fig 4) Parent Document Retriever[16]



(Fig 5) Child Chunks with RAG[17]



· **Contextual Compression Retriever**: This retrieval component is critical for:

· Re-ranking retrieval results based on relevance and potential impact on the report's narrative.

· Employing flowcharts and mathematical models to present the reranking strategy clearly, ensuring that the most relevant and contextually appropriate data is utilized in report generation.

5.3 **Empirical Evaluation**

· A suite of metrics including precision, recall, BLEU, ROUGE, METEOR, and cosine similarity.

· Detailed graphs and tables that provide a quantitative measure of the system's enhancements and effectiveness, ensuring that empirical evidence supports the claimed improvements.

6. Experimental Evaluation and Testing

6.1 **Evaluation Strategy**: Describes the setup of comprehensive testing protocols to ensure that improvements in accuracy and efficiency are measurable and reportable.

6.2 **Performance Metrics**

· **Precision**: Testing against curated datasets to observe improvements in data labeling accuracy.

· **Recall**: Measuring system efficiency in identifying relevant data through user-generated test queries.

· **Advanced Metrics**: Including BLEU, ROUGE, METEOR, and Cosine Similarity for assessing linguistic and contextual quality of generated reports.

6.3 **Results and Analysis**

Table 1 below outlines the specific metrics and methodologies employed in our empirical evaluation phase, detailing the expected improvements in system performance across various dimensions.

Table 1 Empirical Evaluation Plan Table

etric	Evaluation Methodology	Data Source	Expected Outcome
Precision	Compare predicted labels vs. true labels on a test dataset	Curated dataset from web sources	Increase in precision by at least 20% compared to baseline
Recall	Measure the rate of	User-	Improvement in

	true positives identified by the system	generated test queries	recall rates by 15% over current systems
BLEU Score	Evaluate the linguistic quality of generated text	Standard language assessment tests	BLEU score improvement by at least 10 points
ROUGE Score	Assess overlap between system output and human-written summaries	Summarization tasks on news articles	ROUGE score increase, reflecting better summary quality
METEOR Score	Evaluate translation accuracy and fluency	Multilingual datasets for translation	Enhanced METEOR scores indicating improved translation accuracy
Cosine Similarity	Compare the semantic similarity between system output and target text	Diverse content domains	Higher cosine similarity scores, indicating more relevant outputs

7. Discussion

Our discussion is preparatory, contemplating the potential broader implications of our findings. It addresses the scalability and applicability of our proposed system across diverse domains, contingent on the results of our forthcoming empirical evaluation. A critical evaluation of the current methods is necessary to set clear directions for future research that could further enhance the technological landscape, and such evaluation will be based on the concrete data from our planned testing.

8. Conclusion

This research aims to mark a pivotal advancement in automated report generation. The envisioned integration of Rivet with sophisticated RAG techniques has the potential to significantly enhance operational efficiencies and adapt effectively to evolving data demands across various industries. However, we must underscore that our process improvements and the assurance of high-quality, actionable reports will be confirmed upon the completion of the rigorous experimental validation currently underway.

ACKNOWLEDGMENT

This work was supported by Innovative Human Resource Development for Local Intellectualization program through the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (IITP-2024-RS-2022-00156360)

Reference

[1] Roger S. Pressman, Bruce R. Maxim, *"Software Engineering: A Practitioner's Approach"*, New York, NY, McGraw-Hill Education, 2020.

[2] Wu, Y., Han, Y., Shao, F., Guo, Z., "Research Personalized Learning Report Generation--An Example from a Course on Integration of Information Technology and Physics Curriculum*", 2023 International Conference on Intelligent Education and Intelligent Research (IEIR),* 2023, pp. 1-6.

[3] Jiang, Y. (2022) "SDW-ASL: A Dynamic System to Generate Large Scale Dataset for Continuous American Sign Language," *ArXiv, abs/2210.06791*, 2022.

[4] Yadav, G., Yadav, R., Viramgama, M., Viramgama, M., Mohite, A. (2024) "Quantixar: High-performance Vector Data Management System," *ArXiv, abs/2403.12583*, 2024.

[5] Vasireddy, P., Kavi, K., Weaver, A., Mehta, G. (2023) *"Streaming Sparse Data on Architectures with Vector Extensions using Near Data Processing"* , 2023, pp. 16:1-16:12.

[6] Shahmansoori, A. (2024) "Concurrent Brainstorming & Hypothesis Satisfying: An Iterative Framework for Enhanced Retrieval-Augmented Generation (R2CBR3H-SR)," *ArXiv, abs/2401.01835, 2024.*

[7] LangChain, *'Parent Document Retriever'*, 2024. [Online]. Available: https://python.langchain.com/docs/modules/data_connection/retrievers/parent_document_retriever/ [Accessed: April 19, 2024].

[8] LangChain, *'Contextual compression'*, 2023. [Online]. Available: https://python.langchain.com/docs/modules/data_connection/retrievers/contextual_compression/ [Accessed: April 19, 2024].

[9] Rangan, K., Yin, Y. (2024) "A Fine-tuning Enhanced RAG System with Quantized Influence Measure as AI Judge*," ArXiv, abs/2402.17081*, 2024.

[10] Ironclad, 'rivet: The open-source visual AI programming environment and TypeScript library', GitHub, 2024. [Online]. Available: *https://github.com/Ironclad/rivet* [Accessed: April 19, 2024].

[11] SerpApi, 'Google Search API', 2024. [Online]. Available: *https://serpapi.com/* [Accessed: April 19, 2024].

[12] npm, 'About npm', npm Docs, 2023. [Online]. Available: *https://docs.npmjs.com/about-npm* [Accessed: April 19, 2024].

[13] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela, 'Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks', arXiv.org, 2020. [Online]. Available: *https://arxiv.org/abs/2005.11401* [Accessed: April 19, 2024].

[14] Damian Gil, 'Advanced Retriever Techniques to Improve Your RAGs', Towards Data Science, Medium, 2024. [Online]. Available: *https://towardsdatascience.com/advanced-retriever-techniques-to-improve-your-rags-1fac2b86dd61* [Accessed: April 19, 2024].

[15] Azhar, "RAG and Parent Document Retrievers: Making Sense of Complex Contexts with Code," Medium, 2023. [Online]. Available: *https://medium.com/ai-insights-cobet/rag-and-parent-document-retrievers-making-sense-of-complex-contexts-with-code-5bd5c3474a8a* [Accessed: April 22, 2024].

[16] Azhar, 2023, retrieved from Medium on April 22, 2024

[17] Azhar, 2023, retrieved from Medium on April 22, 2024