A Data Solutions Internship: Enhancing Document Retrieval Systems

CS4991 Capstone Report, 2025

Sarah Michelitch
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
unt3qq@virginia.edu

ABSTRACT

In large-scale development settings, debugging underperforming client solutions often requires extensive data analysis and iterative testing. During an internship at Deloitte, I investigated and enhanced a document retrieval system by developing comprehensive analysis and testing frameworks. Though the specifics of the client's project are classified, I may disclose high-level ideas. The solution involved creating Python-based visualization tools utilizing Principal Component Analysis (PCA) and Uniform Manifold Approximation and Projection (UMAP) to analyze embeddings and query-to-document proximities. I also evaluated tens of thousands of documents to identify potential factors affecting embedding performance, including document length, language variations, the presence of images, and character set differences. Implementation included Python-based pipelines for visualizations, targeted queries, and comprehensive testing using Postman collections and Pytest scripts. My work improved the understanding of embedding behavior across different document types and enhanced system validation capabilities. Future work includes expanding the analysis to additional document characteristics, developing adaptive embedding strategies for varying document lengths, and refining embedding models.

1. INTRODUCTION

Consider a document retrieval system designed to match user queries with relevant documents; a task that appears straightforward in theory but often unravels in practice. The system that I worked on during my internship excelled with some queries and inexplicably failed with others, returning documents with no relevance. This inconsistency not only frustrated the development team but also raised concerns of the need to completely rewrite the code. How could a system leveraging some of the most advanced large language models (LLMs) produce such unreliable results?

The challenge lay in diagnosing the root causes of the unpredictability. Unlike my traditional academic experiences, where homeworks and projects come with predefined problems and answer keys, working with real-world systems meant tackling open-ended challenges with no guaranteed solutions. Problems may require hours, days, weeks, or even months to resolve. My task was to devise the web of potential variables affecting the system's performance and identify the factors most significantly impacting its accuracy. Addressing these issues required creativity to generate hypotheses about potential causes and systematic testing to isolate and evaluate each variable's effects.

2. RELATED WORKS

The development of document retrieval systems has seen significant advancements in recent years, driven by innovations in semantic embeddings, query optimization, and retrieval-augmented generation. While these sources were not directly referenced during my internship, they represent seminal developments in the field that align with the challenges and goals of my project.

Li, et al. (2025) systematically investigates the design and optimization of Retrieval-Augmented Generation (RAG) systems, focusing on query expansion techniques, retrieval strategies, multilingual knowledge bases, and a novel Contrastive In-Context Learning RAG. They found that query expansion significantly improved retrieval accuracy by enriching queries with additional contextual information. Techniques like pseudo-relevance feedback were particularly effective in capturing latent information and broadening the scope of retrieval. Another finding was that dense retrieval methods (comparing meanings) outperformed sparse methods (exact keyword matches) in capturing semantic nuances, while hybrid approaches and multi-vector search operations further enhanced retrieval precision and efficiency. Lastly, the introduction of contrastive learning, a new method that teaches the system to focus on the most relevant parts of documents by contrasting relevant and irrelevant documents, refined retrieval precision leading to more accurate and contextually relevant responses.

In another study, VectorSearch proposes a novel approach to document retrieval by leveraging optimized algorithms, semantic embeddings, and multi-vector indexing (Monir, et al., 2024). By representing documents with multiple embeddings, the system was able to capture different aspects of their context, enabling more nuanced and precise retrieval. Additionally, the integration of state-of-the-art language models, like BERT and RoBERTa, for generating embeddings and efficient indexing methods like Hierarchical Navigable Small World (HNSW) graphs, ensured fast retrieval for both large and small datasets. These innovations collectively reduced semantic gaps and improved the system's ability to match queries with the most relevant documents.

Complementary research introduced a fine-tuned BERT model for automatic query expansion (AQE) to improve document retrieval systems (Deepak & Kumar, 2024). The researchers generated contextual embeddings for both queries and documents, then used co-occurrence statistical information to identify and add relevant keywords to the original query. This approach allowed the system to better capture the semantic intent of the query, even when the initial terms were ambiguous or incomplete. The findings demonstrated that this method significantly improved retrieval accuracy, with higher precision and recall, for complex and poorly defined queries and reduced the likelihood of irrelevant document retrievals.

These works collectively provide valuable insights into improving document retrieval systems, aligning with my project's goals of enhancing retrieval accuracy and understanding embedding behavior. Techniques like query expansion, multi-vector search, and contrastive learning offer innovative approaches to address the query-document mismatches.

3. PROJECT DESIGN

This section outlines the technical structure, requirements, and limitations of the document retrieval system, along with the strategic approach taken to diagnose and improve its performance.

3.1 System Architecture

The document retrieval system consisted of a user-facing web application designed to accept user queries and return relevant documents from a large database. The backend relied on Python-based pipelines integrated with advanced LLMs to generate semantic embeddings for both queries and documents. These embeddings aimed to capture semantic meaning rather than relying on exact keyword matches, theoretically ensuring accurate and contextually relevant retrieval.

3.2 Requirements

The client's primary requirement was to develop a user-friendly interface that allowed customers to efficiently access relevant documents stored in their database. The system aimed to eliminate the extensive manual labor previously required to filter and identify documents based on user requests. Key objectives included improving retrieval accuracy, understanding embedding behaviors, and developing testing frameworks to validate system performance.

3.3 Limitations

Initial testing revealed several limitations, including inconsistent retrieval accuracy. Specific issues included: irrelevant documents being returned for certain queries, overly broad document matches, and correct behavior limited to specific query types without consistent predictability. These inconsistencies highlighted the need for a deeper investigation into the factors affecting embedding performance and retrieval accuracy.

3.4 Key Components

The following components represent the core tools and methods I developed to investigate embedding inconsistencies.

3.4.1 Visualization Tools

To better understand embedding behavior, I developed Python-based interactive visualization tools using Principal Component Analysis (PCA) and Uniform Manifold Approximation and Projection (UMAP). These tools allowed for the analysis of embedding clusters and query-to-document proximities, both of which helped identify potential issues within the embeddings.

3.4.2 Automated Testing Frameworks

I implemented automated testing frameworks using Pytest scripts to systematically evaluate embedding accuracy. These tests were designed to identify cases where queries failed/succeeded to retrieve intended documents. Additionally, Postman collections were created to streamline the execution and monitoring of these tests.

3.4.3 Targeted Query Analysis

Customized queries were crafted to precisely match specific documents, facilitating clear diagnostics for evaluating system performance. This approach helped isolate problematic characteristics among documents, such as encoding discrepancies (e.g., UTF-8 vs. ASCII), formatting inconsistencies, and the presence of hidden or invisible characters.

3.4.4 Challenges

The primary challenge was the inconsistent behavior of embeddings generated by the LLMs. Additional complications included variations in document length (ranging from hundreds of pages to less than a single page), multilingual documents, images, and formatting issues. Each of these challenges disrupted the semantic representation of documents, negatively impacting the embeddings.

4 RESULTS

Due to the limited duration of the internship (seven weeks), fully resolving all performance issues within the retrieval system was not possible. However, significant progress was achieved in identifying potential causes of inaccuracies. The newly developed visualization tools improved the understanding of embedding behaviors across various document types. Last, the implementation of comprehensive testing frameworks ensure consistent and repeatable validation of system performance. These outcomes, while not completely solving the challenges, provided my team with a clear roadmap for future improvements.

5 CONCLUSION

This project demonstrated the complexity and importance of fine-tuning systems in real-world environments, where unpredictable system behavior can undermine even the most sophisticated architectures. By developing interactive visualization tools, automated testing pipelines, and targeted diagnostics, I helped create a deeper understanding of embedding inconsistencies and laid the groundwork for more robust system validation. Beyond the technical work, this internship taught me how to navigate ambiguity, collaborate effectively within a team, and approach open-ended problems with a strategic, investigative mindset skills that will continue to shape my future in data science, computer science, and engineering.

6 FUTURE WORK

Given that the system was still under active development at the conclusion of my internship, next steps should focus on addressing the remaining inconsistencies in retrieval performance. I would recommend implementing a contrastive learning approach, as discussed in the work by Li et al. (2025) to help the system better distinguish between relevant and irrelevant documents. This method could sharpen the semantic precision of query-document matching and enhance retrieval accuracy. Additionally, future enhancements could include adaptive embedding strategies for handling variable-length documents, further language normalization and expanding testing frameworks to include performance benchmarking across multilingual corpora.

REFERENCES

Li, S., Stenzel, L., Eickhoff, C., & Bahrainian, S. (2025). Enhancing retrieval-augmented ceneration: A study of best practices. ACL Anthology, 6705–6717. aclanthology.org/2025.coling-main.449

Monir, S., Lau, I., Yang, S., & Zhao, D. (2024). VectorSearch: Enhancing document retrieval with semantic embeddings and optimized search. ArXiv.org. arxiv.org/abs/2409.17383

Vishwakarma, D., & Kumar, S. (2024). Fine-tuned BERT algorithm-based automatic query expansion for enhancing document retrieval system. Cognitive Computation, 17(1). doi.org/10.1007/s12559-024-10354-5