European Journal of Information Technologies and Computer Science
Vol 5 | Issue 1 | February 2025
ISSN 2736-5492

RESEARCH ARTICLE

Exploring RAG Solutions for a Specific Language: Albanian

Leotrim Ramadani* and Fisnik Doko

ABSTRACT

The primary goal of this project is to develop a powerful information retrieval and question-answering system specifically tailored for Albanian-speaking users, bridging the gap between traditional document search methods and modern, context-aware responses. This solution aims to address the unique linguistic and document-processing challenges present in Albanian-language data by combining state-of-the-art Retrieval-Augmented Generation (RAG) techniques with advanced natural language processing (NLP) capabilities. Through the implementation of this RAG solution, we aim to empower organizations, educational institutions, and users in Albanian-speaking regions with fast, accurate, and contextually relevant access to information within their documents. By leveraging vector-based search, large language models, and optimized document processing adapted to the nuances of the Albanian language, this system will simplify information access, reduce reliance on manual searches, and enhance decision-making processes. Retrieval-augmented generation (RAG) is a technique for increasing the accuracy and reliability of generative models of Artificial Intelligence with facts obtained from various external resources. This technique or solution fills a gap in the way LLM works. In other words, LLMs are like neural networks of the brain, usually measured by the number of parameters they contain in the current digital era, organizations and institutions in Albanian-speaking regions face significant challenges in processing, analyzing, and efficiently retrieving information from their documents. Traditional search methods often fail to understand the contextual nuances of the Albanian language, leading to inefficient information retrieval and suboptimal user experiences. Also, the lack of specialized "Natural language processing" or NLP (natural language processing) tools for the Albanian language creates barriers in the effective implementation of document management and question-answering systems.

Keywords: Albanian, LangChain, Retrieval-Augmented Generation, Similarity Search.

Submitted: December 13, 2024

Published: February 12, 2025

10.24018/ejcompute.2025.5.1.148

Faculty of Informatic Sciences, Mother Teresa University Skopje, North Macedonia.

****Corresponding Author:***
e-mail:
Leotrim.Ramadani@students.unt.edu.mk

1. Introduction

The main goal of this project is to develop a powerful information retrieval and query answering system tailored specifically for Albanian-speaking users, bridging the gap between traditional document search and modern, context-aware answers. This solution aims to address the unique language and document handling challenges present in Albanian-language data by combining the latest augmented generation (RAG) techniques with advanced natural language processing (NLP) capabilities.

Through the implementation of this RAG solution, we aim to empower organizations, educational institutions, and users in Albanian-speaking regions with fast, accurate and appropriate access to the context of information within their documents. Utilizing vector-based search, large language models, and optimized document processing adapted to the nuances of the Albanian language, this system will simplify access to information, reduce reliance on manual search, and improve decision-making processes.

2. Methodology

During the drafting of this research titled: "Exploring RAG Solutions for a specific language–Albanian" various scientific methods will be employed to elaborate on the topic and achieve its objectives. The scientific methods used in this research are as follows:

- *Collection and analysis of professional and scientific information:* This involves exploring contemporary literature and other relevant materials in this field to gather and analyze data related to the study's theme.
    - *Label:* Literature Review and Data Analysis
- *Data chunk generation:* Using OpenAI embedding models, we created vector representations of document segments, capturing the semantic nuances of Albanian text.
    - *Label:* Data Preparation and Representation
- *Processing RAG solutions:* The primary process for developing the RAG solution began by connecting the vector database with the LLM. For each user query, the system first retrieves the most relevant document segments based on semantic similarity and then uses these retrieved segments to generate a well-informed response.
    - *Label:* Solution Development and Query Processing

3. Similar Examples like Our Research

During our research about RAG solutions and because of our idea that was to only focus on one language, we found some research papers that were like ours, and that helped us understand and find more interesting things as well.

We first found this paper [1], which dives deeper into understanding the RAG solutions and does a calculation to see which one of the AI platforms will perform better for this specific language.

We also have found this paper [2], which compares the results for two specific languages. It calculates the results and gives out the accuracy of each AI platform and in some way or another tells the readers which platform is better for what purposes.

There is another paper that we found, and which was interesting to read and to compare with our idea [3].

This article is important for us since it exactly points out the type of questions we can do to our system and what we can expect from those questions.

4. RAG solutions

Retrieval-augmented generation (RAG) is an advanced method for enhancing the output of generative AI models by incorporating factual information from external resources [4]. This approach addresses a key limitation of large language models (LLMs): their inability to access real-time or domain-specific knowledge without additional training.

LLMs, much like neural networks in the human brain, function based on their parameters, which represent patterns in language usage derived from extensive training datasets. These parameters allow LLMs to generate coherent responses quickly and efficiently, making them well-suited for general-purpose queries. However, they often fall short when tasked with providing detailed or up-to-date information on specialized topics. For example, while an LLM can summarize general knowledge, it struggles to answer context-specific questions about a newly provided document.

RAG solves this problem by enabling LLMs to reference external sources dynamically during query processing. Much like footnotes in a research paper, these sources can be cited, ensuring transparency and trust. By offering verifiable facts and reducing errors—often called "hallucinations" in AI—the RAG approach improves the reliability and accuracy of responses.

One notable advantage of RAG is its efficiency. Instead of retraining a model with additional datasets, RAG allows developers to connect external resources to LLMs seamlessly. This lightweight implementation is not only faster but also more cost-effective, often requiring minimal code to integrate new data sources.

RAG also expands the potential of AI applications across industries. For instance, integrating a medical database with an LLM could create a valuable assistant for healthcare professionals, while linking financial data could support analysts in making informed decisions. Similarly, businesses can enhance customer support, training, and productivity by converting internal documents, logs, or multimedia content into accessible knowledge bases. The concept of retrieval-augmented techniques can be traced [5] (Fig. 1).

5. Technologies Utilized in Our Research

Throughout the course of this research, we employed various technologies to present the data we collected in diverse formats. Some of these technologies were familiar to us prior to the study, while others were new and required us to acquire the necessary skills to effectively utilize them. The technologies utilized in this research include: LangChain, Vector Databases, and Qdrant.

5.1. LangChain

LangChain [6] is an open-source framework designed to build applications that leverage large language models (LLMs). LLMs are pre-trained deep learning models capable of generating text-based outputs such as answering queries or creating images from prompts. LangChain enhances LLMs by providing tools for better customization, improving the relevance and precision of model-generated content. Developers can use LangChain's components to create custom prompt chains or adapt existing templates for specific needs.

Additionally, LangChain enables LLMs to interact with new data sets in real-time, eliminating the need for retraining. This flexibility allows organizations to apply LLMs to specialized domains, using internal documents or proprietary data to refine responses. A practical use case is building applications that can access and summarize company data, offering users context-specific answers.

One of the key workflows LangChain supports is Retrieval-Augmented Generation (RAG), which allows

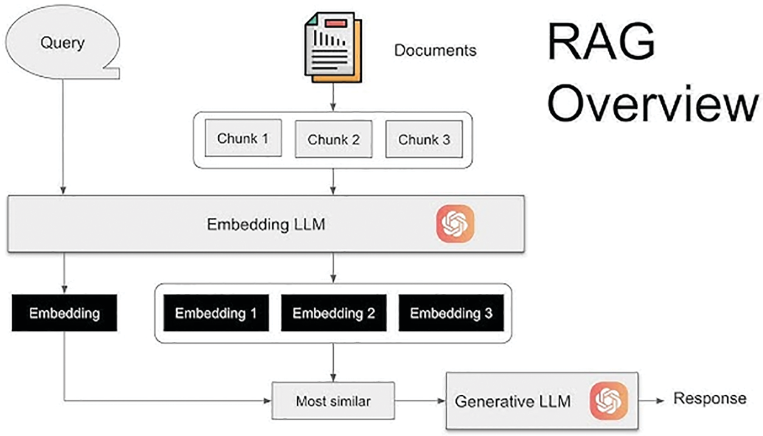

Fig. 1. RAG schema overview.

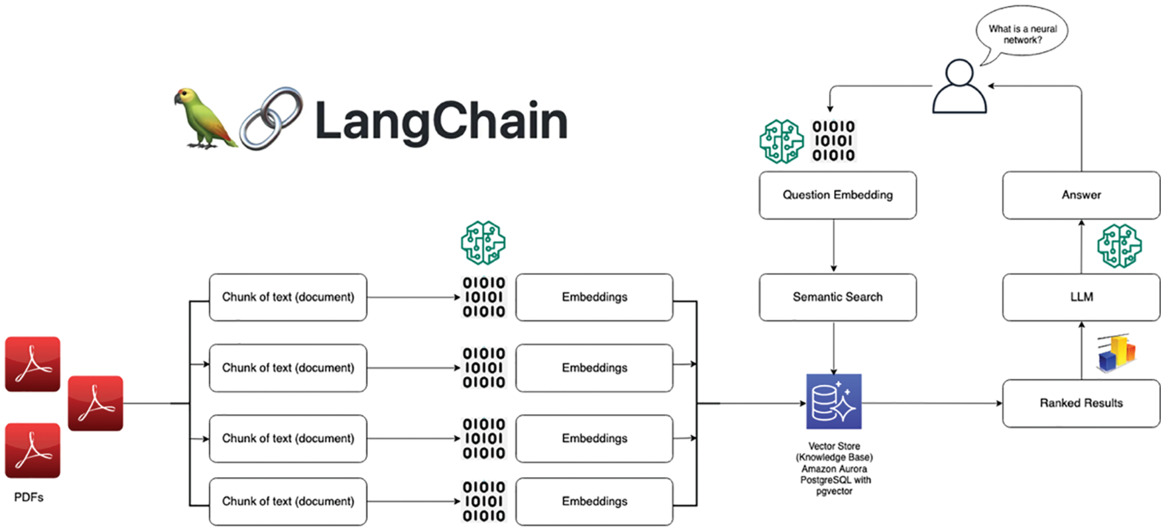

Fig. 2. LangChain schema, structuring, and analysis.

a model to incorporate external information during its response generation. This method helps reduce the likelihood of inaccurate or fabricated responses, improving the overall quality and reliability of the model (Fig. 2).

5.2. *Vector Databases*

A vector database is a collection of data stored as mathematical representations. Vector databases [7] (Fig. 1) make it easier for machine learning models to remember previous inputs, allowing machine learning to be used to power search, recommendations, and text generation use cases. Data can be identified based on similarity metrics instead of exact matches, making it possible for a computer model to understand data contextually.

Data comes in both structured and unstructured forms. With advancements in artificial intelligence, embedding models have been developed to encode various types of information—such as text, images, or audio—into vectors. These vectors preserve the underlying meaning and context of the data, enabling efficient search and retrieval of related items. For example, by using vector search techniques, we can find similar images from a photograph taken on a smartphone. One such vector database used in these applications is Qdrant.

5.3. *Qdrant*

Qdrant is a powerful vector database and similarity search engine designed for efficient storage, management, and retrieval of vector data (Fig. 4). It offers a production-grade API that supports advanced filtering capabilities, making it ideal for applications such as semantic search, faceted exploration, and neural-network-based matching [8]. By enabling seamless integration of vector embeddings, Qdrant allows developers to create sophisticated systems for recommendations, similarity searches, and more.

Built with Rust [9], Qdrant delivers exceptional performance and reliability, even under heavy workloads. This makes it a robust choice for handling the demands of large-scale data processing and real-time query execution. Whether you're working with neural network encoders or embeddings, Qdrant provides the tools to transform raw data into comprehensive applications tailored to diverse use cases (Fig. 3).

6. Fundamental Issues that We Want to Address

1. *Challenges of a Specific Language:*
   - Limited availability of NLP tools and resources for the Albanian language.

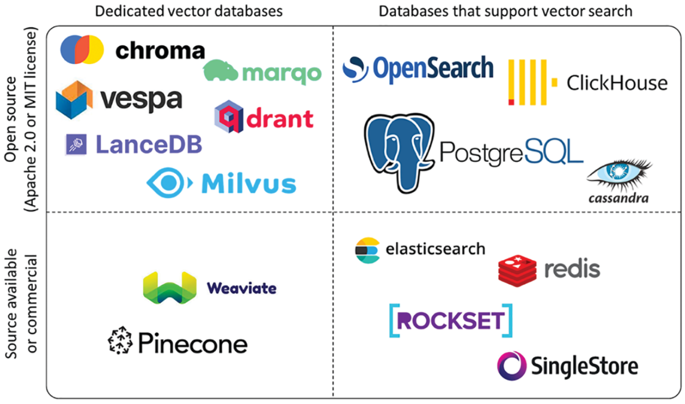
Dedicated vector databases
Databases that support vector search
Open source (Apache 2.0 or MIT license)
Source available or commercial
chroma
marqo
vespa
qdrant
LanceDB
Milvus
OpenSearch
ClickHouse
PostgreSQL
cassandra
Weaviate
Pinecone
elasticsearch
redis
ROCKSET
SingleStore

Fig. 3. Dedicated vector databases, tools, and databases that support vector search.

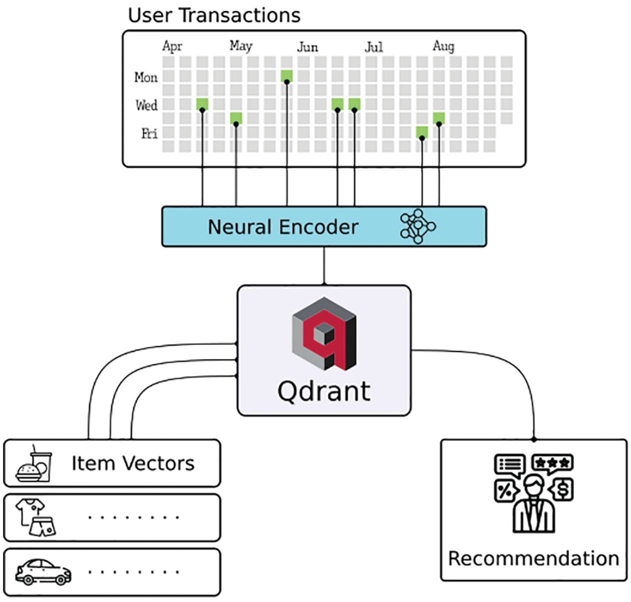

Fig. 4. Qdrant presentation on performance and scalability.

- Complexity in handling the unique characters and grammatical structures of Albanian.
- The need for accurate semantic understanding of text in Albanian.

2. *Document Processing Issues:*
   - Inefficient manual search through large volumes of documents in Albanian.
   - Difficulty in maintaining context when processing large documents.
   - Challenges in extracting relevant information from various document formats.

3. *Information Retrieval Limitations:*
   - Poor accuracy in traditional keyword-based search methods.
   - Lack of systems capable of returning responses in Albanian content.
   - Inability to effectively handle natural language questions.

4. *User Interaction Barriers:*
   - The need for intuitive interfaces that support the Albanian language.
   - Demand for accurate and relevant responses.
   - Requirement for real-time processing and response generation.

7. Proposed Solution

To address these challenges, we propose the development of a specialized response generation system, or an RAG system, which:

1. *Optimized Document Processing:*
   - Implements advanced document processing capabilities specifically tailored for the Albanian language.

2. *Semantic Understanding in Albanian:*
   - Utilizes modern techniques to divide data into smaller chunks and accurately capture semantic meaning in Albanian.

3. *Enhanced Information Retrieval:*
   - Employs vector-based search methods for improved information retrieval.

4. *Accurate Response Generation:*
   - Integrates large language models (LLMs) to generate precise and contextually relevant answers.

5. *User-Friendly Interface:*
   - Offers an intuitive and user-friendly interface designed for the Albanian language, enhancing accessibility and usability.

8. Expected Impact

1. *Reduce Time Spent on Document Search:*
   - Significantly decrease the time required to search through documents and retrieve information.

2. *Improve Accuracy:*
    - Enhance the precision of responses to questions related to document content.
3. *Enhance User Experience:*
    - Improve the user experience through natural language interaction.
4. *Lay a Foundation for Future Applications:*
    - Establish a basis for future applications in Albanian language processing.

9. Results and Discussion

After the research that we did and the courses we followed, we managed to create an RAG solution that will help us in our use case.

We used Python as a primary programming language, Qdrant as our host for our vector database, OpenAI's pre-trained models, and LangChain for document processing, managing vector store operations, and creating QA chains.

In the beginning, it was a bit challenging to understand how all the technologies work separately and then combining all of them into one single project. The first results that we had were not good enough for our case.

Initially, OpenAI's API returned answers in English, even when both the questions and the document were in Albanian. After conducting research and implementing technical improvements, we resolved this issue.

However, we then noticed that the system would answer questions unrelated to the uploaded document. For instance, after uploading a document about the Albanian constitution and its laws, we asked questions about programming or even general queries like "Who is the founder of Apple?" The system provided correct answers to all these questions, regardless of the document's context.

Our goal was to ensure that the answers were strictly based on the uploaded document and to ignore any unrelated queries. To achieve this, we implemented a limitation that restricts the system to only return answers derived from the document. This adjustment successfully addressed the issue, ensuring focused and contextually relevant responses.

We can now upload any document and use our RAG solution to ask questions specifically about its content. The system provides answers in Albanian and is strictly based on the uploaded document. If a question is unrelated to the document, the service responds by indicating that the question is out of context and that no information is available regarding it.

10. Summary of the Study

In Albanian-speaking regions, accessing information from extensive document collections has been challenging due to the specific complexities of the language and the limited availability of natural language processing (NLP) tools tailored for Albanians. Traditional search methods often fail to understand the unique grammatical structures, characters, and context required for effective retrieval, leading to inefficient information processing and poor user experiences.

To address these challenges, we developed a Retrieval-Augmented Generation (RAG) system specifically designed for Albanian-language content. By leveraging LangChain, Qdrant as a vector database, and OpenAI's API for response generation, we built a system capable of processing Albanian text, generating embeddings for semantic understanding, and retrieving relevant document segments in response to user queries. This RAG solution integrates state-of-the-art NLP and machine learning methods to deliver context-aware responses based on accurate, relevant Albanian-language data.

The implementation of this RAG solution has demonstrated significant improvements in both retrieval speed and accuracy for Albanian content. The system enables users to pose natural language queries and receive specific, relevant answers from a wide range of documents, reducing reliance on inefficient keyword-based searches. This project not only enhances access to information but also establishes a foundational approach for future NLP applications in Albanian, opening opportunities for expanded digital experiences in sectors such as education, administration, and customer support.

11. Conclusion

The development of this RAG solution marks a significant step forward in the field of natural language processing for the Albanian language. Through the successful integration of LangChain, OpenAI API [10], Streamlit [11], [12], Python, and vector databases in Qdrant, we have demonstrated the potential of RAG systems to revolutionize how information is accessed and utilized.

This project achieved its primary goal of enabling users to upload documents and receive accurate, context-specific answers based solely on the content of those documents. By ensuring that out-of-context questions are flagged and not answered, the system provides a reliable and focused interaction with the uploaded material. This feature addresses critical challenges in traditional information retrieval, such as irrelevant or ambiguous results, and underscores the precision and adaptability of RAG solutions.

The implications of this work extend beyond just answering questions. RAG solutions can be applied in various fields, including education, legal research, healthcare, and business, where accessing accurate information from large document repositories is vital. Their ability to handle complex, context-aware queries makes them particularly helpful in scenarios where traditional keyword-based search methods fall short.

By focusing on a less-represented language like Albanian, this project also emphasizes the importance of democratizing access to advanced NLP technologies for all languages, regardless of their global prominence. This approach not only supports language preservation but also opens doors for further innovations in digital tools tailored to specific linguistic and cultural contexts.

Looking forward, this work provides a solid foundation for expanding RAG solutions. Future developments could include support for multilingual document processing, enhanced semantic understanding, and integration with additional data sources. Ultimately, the adaptability and efficiency of RAG systems ensure their growing relevance and utility in an increasingly data-driven world.

Conflict of Interest

The authors declare that they do not have any conflict of interest.

References

[1] El-Beltagy SR. ScienceDirect. [Online]. 2024. Available from: https://www.sciencedirect.com/science/article/pii/S1877050924030047.

[2] Leka E. ResearchGate. [Online]. 2024. Available from: https://www.researchgate.net/publication/384927770_Large_Language_Models_LLMs_Output_Quality_Comparison_Between_English_and_Albanian.

[3] Trendafili E. ProQuest. [Online]. 2023. Available from: https://www.proquest.com/openview/5782eeaaa5c38e9cc8a9728612b63dd1/1?pq-origsite=gscholar&cbl=5444811.

[4] Amazon. aws.amazon. [Online]. 2024. Available from: https://aws.amazon.com/what-is/retrieval-augmented-generation/.

[5] Ramadani L, TN, RS. The prediction of growth of the GDP of north macedonia for 2024 using logistic and linear regression model. *Asian J Econ, Bus Account*. 2024;24(3):221–8. doi: 10.9734/ajeba/2024/v24i31255.

[6] LangChain. LangChain. [Online]. 2024. Available from: https://www.langchain.com/.

[7] Schwaber-Cohen R. PineCone. [Online]. 2023. Available from: https://www.pinecone.io/learn/vector-database/.

[8] More Qda. Github. Available from: https://github.com/qdrant/qdrant.

[9] Qdrant-Github. Github. [Online]. 2025. Available from: https://github.com/qdrant/qdrant.

[10] Api O. Platform OpenAi. [Online]. 2024. Available from: https://platform.openai.com/docs/overview.

[11] Mhadhbi N. Datacamp. [Online]. 2024. Available from: https://www.datacamp.com/tutorial/streamlit.

[12] Guide S. GeeksForGeeks. [Online]. 2024. Available from: https://www.geeksforgeeks.org/a-beginners-guide-to-streamlit/.