УДК 681.12

Pobidash V. V.

ENHANCING LANGUAGE MODELS WITH RETRIEVAL-AUGMENTED GENERATION

Vinnytsia National Technical University

Анотація

У цій статті досліджуються основні принципи генерації з доповненням через пошук, її переваги, такі як: підвищення фактичної точності і зменшення кількості галюцинацій, а також недоліки, зокрема обчислювальні накладні витрати і проблеми з надійністю даних. Крім того, розглядаються реальні застосування та нещодавні дослідження в галузі систем на основі RAG.

Ключові слова: генерація з доповненням через пошук, RAG, мовні моделі, штучний інтелект, пошук інформації, точність ШІ, контекстна релевантність.

Abstracts

This paper explores the core principles of RAG, its benefits — such as improved factual accuracy and reduced hallucinations — and its challenges, including computational overhead and data reliability concerns. Additionally, real-world applications and recent research in RAG-based systems are reviewed.

Keywords: Retrieval-Augmented Generation, RAG, language models, artificial intelligence, information retrieval, AI accuracy, contextual relevance.

Introduction

The rapid evolution of artificial intelligence (AI) has led to the development of sophisticated models capable of understanding and generating human-like text. However, these models often face challenges related to the accuracy and relevance of their outputs, especially when relying solely on pre-existing training data. Retrieval-augmented generation (RAG) systems enhance LLMs' inherent capabilities by pulling information that is not in the initial training dataset from external databases or documents before generating responses. This approach improves the factual accuracy and contextual relevance of outputs.

The phrase "Retrieval Augmented Generation" accurately captures the core concept involved:

- **Retrieval** refers to the process of searching for and obtaining relevant information. The component responsible for this search and retrieval is known as the retriever.
- **Retrieval Augmented** means enhancing a user's query with the relevant information that has been found.
- **Retrieval Augmented Generation** involves creating a response for the user that incorporates this additional relevant information.

RAG systems offer several benefits. They extend the model's knowledge beyond its initial training data, allowing for more up-to-date and relevant responses. They also help reduce inaccuracies and "hallucinations" in generated content. Additionally, RAG systems can incorporate company- or project-specific proprietary information into their responses.

However, there are also some disadvantages. The retrieval process may introduce delays, affecting real-time user interactions. The system's reliability depends on the quality and trustworthiness of the data sources it uses. Moreover, RAG systems require higher computational resources compared to traditional models, as they involve both generative and retrieval processing.

Here are a few recent works that showcase the potential for innovation and progress in this exciting field:

- *"Harnessing Retrieval-Augmented Generation (RAG) for Uncovering Knowledge Gaps"* [2]. This study presents a methodology for using RAG to identify and address knowledge gaps in information retrieval systems. The approach demonstrates a consistent accuracy of 93% in generating relevant suggestions, highlighting RAG's potential in various fields, including scientific discovery and educational enhancement.
- The Wall Street Journal's insightful piece titled “AI Doesn’t Know Much About Golf. Or Farming. Or Mortgages. Or …” [3], that dives into the complexities and costs of RAG, fine-tuning, and

custom-built AI models, as well as the hurdles companies face in managing data integrity to fully optimize these models.

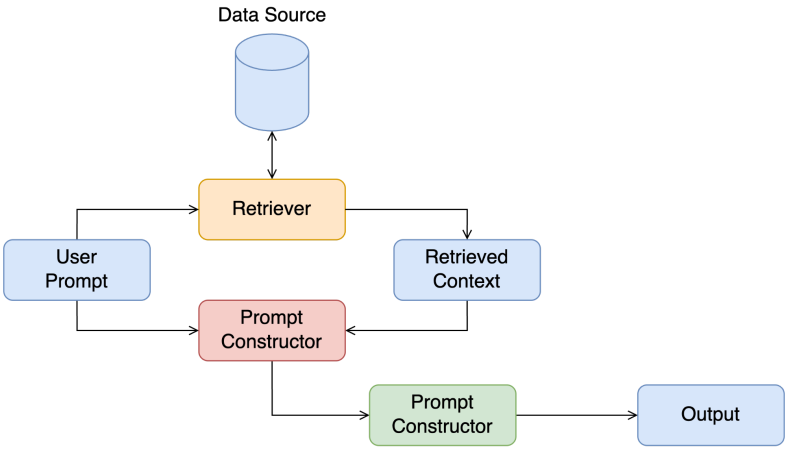
- Another publication at WSJ discusses how companies are leveraging specialized internal AI tools like RAG to connect large language models with their data, enhancing productivity and protecting against AI hallucinations. For example, firms such as Shorenstein Properties use RAG to automate file tagging and organize data more efficiently, indicating broader applicability beyond familiar AI tasks [4].

RAG under the hood

RAG systems always include two key components:

- **Retriever:** This component functions like a search engine, scanning a data source to find and retrieve the most relevant documents or pieces of information that can help answer a user request. It may employ AI models designed for specific tasks (such as BERT or SBERT for generating embeddings), search methods and algorithms, plugins, vector databases, and other programming mechanisms.
- **Generator:** This is a core LLM (such as GPT-4) that generates the final output based on the user input and the retrieved context.

Picture 1 shows a simplified diagram of how the RAG system works.



Picture 1 - Simplified scheme of how the RAG system works

The data source includes raw data such as attached documents, server collections, or internet content like Wikipedia articles. Not all extracted information is sent to the LLM input due to the limited context window. The prompt constructor prepares the input by combining the user prompt, selected context, and additional instructions or metadata to help the model generate a better response.

One major issue is fuzzy search—simply matching a user's query with exact text from a knowledge base is ineffective. The challenge lies in selecting the right algorithm to retrieve only the most relevant pieces of text while filtering out noise.

Another key challenge is determining the optimal size of text chunks provided to the LLM. If the segments are too large, the model might receive unnecessary information; if they are too small, it may lack context to generate a meaningful response.

Additionally, when working with multiple or lengthy knowledge base articles, it is crucial to decide how to trim, combine, or compress them efficiently so that the model receives the most relevant data without losing essential details.

Examples

Here are a couple of specific use cases of RAG systems.

E-Commerce Customer Service: Imagine a customer receives a damaged product and asks a chatbot on the store's website, "How can I return my order? The product inside the package was damaged.".

Without RAG a basic chatbot might give a vague or outdated response, such as: "Please check our returns information on the website.".

With RAG the chatbot can pull the latest return policy from the store's database and provide a detailed, helpful answer: "You can return damaged items within 30 days of receipt. Please fill out the return form on our website [link], and we'll send you a prepaid shipping label via email. For more details, click [here].".

Academic Research Assistance: Imagine a graduate student working on a thesis about climate change asks a chatbot, "What are the latest findings on sea level rise in coastal cities?"

Without RAG a regular chatbot might only offer general information based on outdated training data: "Sea levels are rising due to climate change, which poses risks to coastal cities. You should check recent academic journals for specific data.".

With RAG the chatbot can retrieve the latest research and provide a more precise answer: "A 2023 study by [Authors] in [Source] found that sea levels are rising 3.7mm per year, which is 30% faster than previous estimates. Cities like Miami and New Orleans could see up to 1 meter of sea level rise by 2100 if current trends continue. For more details, here are three recent studies with summaries: [list of papers]. Would you like more details on specific cities or mitigation strategies?"

Conclusion

AI tools are constantly evolving, and Retrieval-Augmented Generation (RAG) systems, along with Large Language Model (LLM) plugins, significantly enhance the capabilities of language models. However, implementing these systems effectively comes with challenges that must be addressed.

RAG systems can fully leverage external data sources to improve accuracy and user experience. These advancements not only refine how AI retrieves and processes information but also expand the range of complex tasks that language models can handle, paving the way for more innovative applications in the future.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Hurtado, J. F. (2023). Harnessing Retrieval-Augmented Generation (RAG) for Uncovering Knowledge Gaps. DOI: https://doi.org/10.48550/arXiv.2312.07796
2. Bousquette I. (2024). AI Doesn't Know Much About Golf. Or Farming. Or Mortgages. Or … [Електронний ресурс] - Режим доступу до ресурсу: https://www.wsj.com/articles/ai-doesnt-know-much-about-golf-or-farming-or-mortgages-or-ae296774
3. Rosenbush S. (2024). Companies Look Past Chatbots for AI Payoff. [Електронний ресурс] - Режим доступу до ресурсу: https://www.wsj.com/articles/companies-look-past-chatbots-for-ai-payoff-c63f5301

Побідаш Владислав Віталійович — студент групи 2ICT-24м, факультет комп'ютерних систем та автоматики, Вінницький національний технічний університет, Вінниця, e-mail: vladpobidash@gmail.com

Науковий керівник: ***Мокін Віталій Борисович*** – д-р. техн. наук, проф., завідувач кафедри системного аналізу та інформаційних технологій, Вінницький національний технічний університет, Вінниця, e-mail: vbmokin@vntu.edu.ua

Pobidash Vladyslav V. — student of group 2IST-24m, faculty of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: vladpobidash@gmail.com

Scientific adviser: Mokin Vitalii B. – Dr. tech. Sciences, Prof., Head of the Department of System Analysis and Information Technologies, Vinnytsia National Technical University, Vinnytsia, e-mail: vbmokin@vntu.edu.ua