Advances in Computer, Signals and Systems (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2024.080608
ISSN 2371-8838 Vol. 8 Num. 6

Application of RAG Model Based on Retrieval Enhanced Generation Technique in Complex Query Processing

Xiangcan Li

Bytedance Inc, San Jose, 95131, USA

Keywords: RAG model, retrieval enhancement generation, complex query processing, natural language processing, deep learning

Abstract: In the field of complex query processing, traditional natural language processing models are often difficult to effectively deal with the diversity and complexity of query contents. The Retrieval Augmented Generation (RAG) model demonstrates unique advantages in processing complex queries by combining the two processes of retrieval and generation. This paper provides an in-depth discussion on the working principle of the RAG model and applies it to complex query processing scenarios. By analyzing real cases and validating experimental results, we demonstrate the significant advantages of the RAG model in enhancing query processing results. Although the RAG model shows good performance in processing complex queries, its application still faces some challenges and limitations. This paper concludes with an outlook on the future development of the RAG model, exploring possible optimization directions and application prospects.

1. Introduction

With the increasing complexity of information needs, traditional Natural Language Processing (NLP) techniques show limitations when facing complex queries[1]. Complex queries usually involve multi-level information association and deeper semantic understanding, requiring models that can not only generate relevant content, but also retrieve and integrate information from large-scale data[2]. To cope with these challenges, models that combine retrieval and generation capabilities are gradually becoming a hot research topic.

The purpose of this paper is to explore the application of models based on Retrieval-Augmented Generation (RAG) technology in complex query processing[3].RAG models provide an effective way to better cope with complex queries by combining the generation task with the retrieval process[4]. In this paper, we will analyze the structure of the RAG model and its working mechanism in detail, and evaluate its performance in handling complex query tasks.

In recent years, many studies have been devoted to improving the comprehension and generation capabilities of natural language processing models, but most of the models are still unable to effectively handle the demands of complex queries[5]. Traditional generative models often lack in-depth understanding of contextual information, while retrieval models, while excelling in acquiring relevant information, still have limitations in integrating and generating natural language responses. The RAG model overcomes the shortcomings of a single model by combining the two, demonstrating potential in several domains. This paper first introduces the basic concepts and

principles of the RAG model and its retrieval enhancement generation technique[6]. Then, through specific case studies and experiments, it demonstrates the effectiveness of the RAG model in complex query processing and analyzes it in comparison with other methods. Finally, this paper discusses the limitations and challenges of the RAG model in its application, and provides an outlook on the future development direction.

2. Overview of the RAG model and its retrieval enhancement generation technique

The RAG model, known as Retrieval-Augmented Generation (RAG), is a natural language processing framework that combines information retrieval with generative modeling[7].The core idea of the RAG model is to retrieve document fragments relevant to the query from a large-scale document repository before generating a natural language response, and these fragments are fed into the generative model as contextual information to make the generated response more relevant and accurate[8]. These snippets are fed into the generation model as contextual information to make the generated response more relevant and accurate. By combining retrieval and generation in this way, the RAG model is able to provide more accurate answers in complex query scenarios， showed in Figure 1:

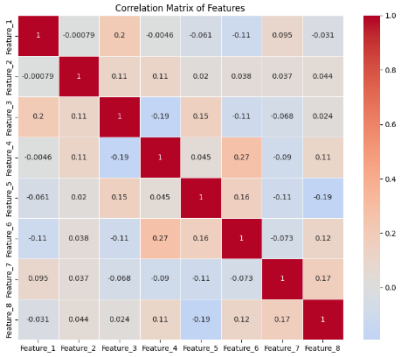

Figure 1: Correlation Matrix of Features

Retrieval Augmented Generation (RAG) is a technique that deeply integrates traditional information retrieval methods with generative modeling. In the RAG model, the retriever is responsible for filtering out the most relevant documents or information fragments from a large-scale dataset, while the generator generates natural language responses based on these retrieved contents [9]. The advantage of this approach is that it can utilize the rich information in the external knowledge base to compensate for the knowledge blindness of the generative model due to insufficient training data when facing complex queries, thus improving the quality and accuracy of the response [10]. The Information Retrieval Step:

$$\text{Retrieval Score}(q, d) = \sum_{i=1}^{n} \text{Embedding}(q_i) \cdot \text{Embedding}(d_i) \quad (1)$$

The workflow of a RAG model is usually divided into two main steps: retrieval and generation. In the retrieval phase, the model selects the document fragments with the highest relevance to the query from a pre-constructed document library based on the input query. These snippets are passed to the model in the generation phase as additional contextual information. The model in the generation phase then uses these information fragments to generate the final natural language response. Through this two-stage processing, the RAG model is able to better understand the deeper semantics behind complex queries and generate answers that match the user's needs. Response Generation Step:

$$P(r \mid q, D) = \prod_{t=1}^{T} P(r_t \mid r_{<t}, q, D) \quad (2)$$

The advantages of RAG models in complex query processing are mainly reflected in their efficient utilization of external knowledge and high relevance of generated content. Compared with traditional generative models, RAG models not only rely on pre-trained language models, but also dynamically acquire the most relevant knowledge to the query, which gives them an obvious advantage when facing diverse and high-complexity queries. Retrieval-Augmented Generation (RAG) models are employed in various applications, including question-answering systems, dialogue systems, information retrieval, and knowledge graph construction. They are particularly effective for tasks that require the integration of extensive background information to produce accurate responses.

3. Application of RAG model in complex query processing

In complex query processing, the RAG model demonstrates its unique advantages to effectively respond to the need for multi-level semantic understanding and information association in queries. By combining retrieval and generation, the RAG model not only improves the accuracy and relevance of responses, but also provides more in-depth analysis of answers in complex contexts. The following three sections will explore in detail the specific applications of the RAG model in complex query processing, including its process for responding to complex queries, actual case studies and experimental results, and the comparative advantages with other methods.

3.1 Definitions and Challenges of Complex Queries

Complex queries typically refer to information retrieval and processing tasks that require deep semantic understanding and span multiple topics or domains. Such queries often contain multiple sub-questions, contextual dependencies, and complex logical relationships, requiring the system to be able to integrate multiple sources of information, understand the implied semantic intent, and generate precise, coherent answers when processing them. For example, complex queries may involve detailed analysis of a specific topic, the combination of cross-domain knowledge, or multi-step reasoning in a dynamic context，showed in Figure 2 :

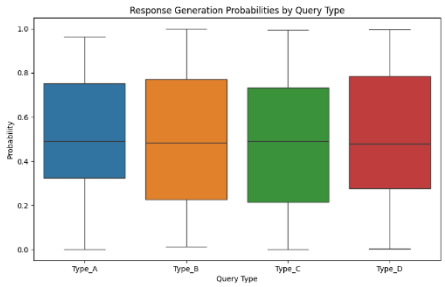

Figure 2: Response Generation Probabilities by Query Type

The diversity of complex queries is reflected in the complexity of their content, structure and semantics. A query may cover multiple topics, requiring the system to extract and integrate information from knowledge bases in different domains; it may also contain implicit premises or assumptions, requiring the system to have the reasoning capability to recognize and supplement such implicit information. In addition, complex queries may also manifest as coherent multi-round dialogs, requiring the system to understand the context and track every change in intent during the dialog. Coping with this diversity requires models with strong semantic understanding and information integration capabilities.

Processing complex queries faces multiple challenges. First is the complexity of semantic understanding, where the system needs to accurately understand the deeper intent behind the query, rather than just matching based on superficial keywords. Second, the correlation and integration of information is also extremely challenging, as the system has to extract relevant information from huge datasets or knowledge bases and integrate them into coherent answers. In addition, complex queries often involve multiple domains or topics, and systems need to have extensive background knowledge and cross-domain processing capabilities to generate accurate and meaningful responses.

A variety of natural language processing models have been used for complex query processing, but most approaches exhibit significant limitations when faced with highly complex queries. Traditional generative models are prone to generating irrelevant or incomplete answers to queries, while pure retrieval models may not be able to handle queries that require deep understanding and semantic reasoning. Models relying solely on generation or retrieval can hardly satisfy the requirements of accuracy and coherence at the same time, which is the background for the proposal and development of the RAG model, aiming at overcoming these limitations and improving the effectiveness of complex query processing by combining the dual advantages of retrieval and generation.

3.2 Flow of RAG model for processing complex queries

The first step for RAG models to process complex queries is to receive and parse the user input query. Since complex queries usually have multi-level semantics and multi-dimensional information requirements, the model first needs to semantically parse the query through a Natural Language Understanding (NLU) module to extract the key elements and identify the main topics and potential sub-questions in the query. The purpose of the initial processing is to structure the query so that it is suitable for subsequent retrieval and generation steps.

After completing the initial processing, the RAG model enters the information retrieval phase. In this phase, the model filters the most relevant document fragments to the query from a large-scale document repository or knowledge base by means of a retriever (usually a retrieval model based on vector representation, such as DPR, etc.). Since complex queries may involve cross-domain information, the retriever needs to search for the most relevant content in multiple knowledge bases or domains. These retrieved document fragments will be fed into the generative model as contextual information, which is used to enhance the accuracy and relevance of the generation phase.

In the generation phase, the RAG model utilizes the retrieved relevant document fragments as well as the initially processed query information to generate natural language responses that are coherent and consistent with the user's needs. Generation models are usually based on pre-trained language models (e.g., BERT, GPT, etc.), which are able to understand the complex semantics behind the query more deeply by fusing the retrieved contextual information and generating accurate and detailed responses. The core of the generation phase is to effectively integrate the retrieved information with the query content to ensure that the generated response is not only accurate, but also coherent in answering all the user's needs.

After the generation phase is completed, the model also needs to integrate the generated multi-segment answers to ensure that the output answers are logically consistent and comprehensive. For complex queries, the generation model may generate multiple sub-answers, which need to be organized and integrated in a post-processing step to form a coherent and complete final answer. Finally, the integrated answer is returned to the user through the output module of the system.This integration process of the RAG model ensures that the information obtained by the user in a complex query is accurate, comprehensive, and organized, which effectively enhances the user

experience of query processing.

3.3 Comparative analysis and advantages

Traditional generative models, such as neural network-based sequence-to-sequence models, perform well when dealing with simple queries, but often fail when faced with complex queries. These models mainly rely on the information in the training data and are prone to produce empty or inaccurate answers when encountering questions that are beyond the scope of the training corpus. In contrast, RAG models greatly expand the scope of available knowledge by introducing relevant documents retrieved in real-time during the generation process, providing accurate and rich answers even when faced with complex queries beyond the scope of the training data. This combination of retrieval and generation allows the RAG model to significantly outperform traditional generative models in terms of knowledge breadth and answer accuracy.

Traditional retrieval models mainly extract relevant information from pre-indexed document repositories through keyword matching and simple text similarity calculation. However, these models usually only provide direct matching results and lack in-depth understanding of query semantics, making it difficult to provide comprehensive and coherent responses in complex query scenarios. RAG models, based on the relevant information obtained in the retrieval phase, further process and integrate the retrieval results through generative modeling, generating more semantically comprehensible and coherent responses. This capability enables the RAG model to go beyond simple information extraction to provide deeper and more insightful responses, making up for the shortcomings of traditional retrieval models in terms of semantic understanding and answer generation.

In recent years, some models try to combine the retrieval and generation processes, but most of them are deficient in fusion depth and efficiency. Some models only combine the retrieval results at a shallow level, failing to make full use of external knowledge, or are inefficient in the fusion process, making it difficult to be applied to real-world scenarios.The RAG model realizes the deep fusion of retrieval and generation through clever architectural design, which can make full use of the rich information retrieved while maintaining efficient processing performance. In addition, the RAG model adopts an end-to-end training approach, which enables the retriever and generator to be optimized collaboratively, and further improves the overall performance and answer quality of the model. This makes the RAG model outperform other similar fusion models in terms of fusion depth, efficiency and effectiveness.

The RAG model demonstrates multiple significant advantages when dealing with complex queries. First, by combining the retrieval and generation capabilities, the RAG model is able to provide rich, semantically accurate and coherent answers to meet the diverse information needs of users. Secondly, the model has strong generalization ability and can cope with unknown questions beyond the scope of training data. In addition, the end-to-end training and efficient architectural design of the RAG model ensures its feasibility and efficiency in real-world applications for complex query processing tasks of various scales and domains. These advantages make the RAG model a powerful tool for solving complex query challenges, with broad application prospects and development potential.

4. Limitations and future developments in the application of RAG modeling

Although RAG models can significantly extend their knowledge coverage through retrieval-enhanced generation techniques, their performance is still dependent on the quality and breadth of the document repository used. If the information in the retrieval repository is not comprehensive or biased, the model may generate inaccurate or one-sided responses. In addition,

RAG models may still face knowledge blindness when dealing with complex queries involving highly specialized domains or up-to-date knowledge, leading to limitations in the generated content. The performance of the model depends heavily on the frequency of updates and the quality of content in the retrieval repository, which makes the application of the model limited in some scenarios.

The two-phase processing of the RAG model includes both the screening of relevant documents in the retrieval phase and the generation of answers in the generation phase, a process that has a higher computational overhead compared to a single generation or retrieval model. Especially in real-time applications, processing complex queries may lead to delays and affect the user experience. In addition, as the query complexity increases, the required computational resources and processing time also increase significantly. This poses a challenge for resource-constrained scenarios in real-world applications, and how to reduce computational costs while ensuring model performance is a problem to be solved in the future.

While RAG models are able to augment the generated results with external documents when processing complex queries, they may also introduce erroneous or unreliable information as a result, especially if the document repository contains erroneous data or untrustworthy content. How the model screens and filters such information during the generation process to ensure the reliability and security of the output results is an issue that has not yet been fully addressed. In addition, the generation process of RAG models may still be affected by adversarial inputs, and how to improve the robustness of the models so that they can still generate accurate and secure answers in the face of malicious queries is an important direction for future development.

The development of RAG models is likely to focus on several key directions. One is to improve the model's knowledge coverage and updating capabilities, and to enhance the model's performance in the face of up-to-date and specialized queries by dynamically updating the document repository and smarter retrieval algorithms. The second is to optimize the computational efficiency of the model by developing more efficient retrieval and generation algorithms to reduce the time and resource consumption for processing complex queries. To enhance the robustness and security of the model, more effective content filtering and validation mechanisms should be studied to ensure the credibility of the generated results. In addition, further applications of the RAG model may be extended to more complex scenarios, such as multi-round dialog systems, cross-domain knowledge integration, etc., to give full play to its potential in complex query processing. Through continuous optimization in these directions, the RAG model is expected to show its unique value in a wider and deeper range of applications in the future.

5. Conclusion

In today's era of information explosion, processing complex queries has become a key technical challenge. The RAG model provides an innovative solution to this challenge by combining retrieval and generation. This paper explores in detail the application of RAG models in complex query processing, analyzing their processes, advantages, limitations, and future directions. Although RAG models show significant advantages in improving query accuracy and information integration, there are still some limitations in their application, such as data dependency, computational cost, and robustness issues.

With the continuous advancement of technology and the continuous optimization of models, RAG models are expected to play an important role in more fields. By enhancing knowledge coverage, improving computational efficiency, and strengthening robustness, RAG models will be able to better cope with various complex queries and provide more accurate and comprehensive answers. On this basis, the research and application of RAG models will not only promote the

development of natural language processing technology, but also bring new possibilities to the field of intelligent search and information services.

References

[1] Cambazoglu B B, Aykanat C. Performance of query processing implementations in ranking-based text retrieval systems using inverted indices [J]. Information Processing & Management, 2006, 42(4):875-898. DOI: 10. 1016/j. ipm. 2005. 06. 004.

[2] Tamine L, Chrisment C, Boughanem M. Multiple query evaluation based on an enhanced genetic algorithm [J]. Information Processing & Management, 2003, 39(2):215-231. DOI:10. 1016/S0306-4573 (02)00048-1.

[3] Doller M, Tous R, Gruhne M, et al. The MPEG Query Format: Unifying Access to Multimedia Retrieval Systems [J]. IEEE Multimedia, 2008, 15(4):82-95. DOI:10. 1109/MMUL. 2008. 96.

[4] Gao Y, Zheng B, Chen G, et al. Continuous visible nearest neighbor query processing in spatial databases[J]. The VLDB Journal, 2011, 20(3):p. 371-396. DOI:10. 1007/s00778-010-0200-z.

[5] Candan K S, Li W S, Priya M L. Similarity-based ranking and query processing in multimedia databases[J]. Data & Knowledge Engineering, 2000, 35(3):259-298. DOI:10. 1016/S0169-023X (00)00025-2.

[6] Aldous K J. A system for the automatic retrieval of information from a specialist database [J]. Information Processing & Management, 1996, 32(2):139-154. DOI:10. 1016/s0306-4573 (96)85002-3.

[7] Wu J K, Narasimhalu A D. Fuzzy content-based retrieval in image databases [J]. Information Processing & Management, 1998, 34(5):513-534. DOI:10. 1016/S0306-4573(98)00017-X.

[8] Komai Y, Sasaki Y, Hara T, et al. A kNN Query Processing Method in Mobile Ad Hoc Networks[J]. IEEE Transactions on Mobile Computing, 2011, 1(13):287-288. DOI: 10. 1109/MDM. 2011. 39.

[9] Fuhr, Norbert. A decision-theoretic approach to database selection in networked IR[J]. Acm Transactions on Information Systems, 1999, 17(3):229-249. DOI:10. 1145/314516. 314517.

[10] Mamouras K, Raghothaman M, Alur R, et al. Stream QRE: modular specification and efficient evaluation of quantitative queries over streaming data[J]. Acm Sigplan Notices, 2017, 52(6):693-708. DOI: 10. 1145/3140587. 3062369.