ARAG: Analysis and Retrieval Augmented Generation for Comprehensive Reasoning over Socioeconomic Data

Yixiong Xiao
Baidu Research
Beijing, China
xiaoyixiong@baidu.com

Jingjia Cao
Baidu Research
Beijing, China
caojingjia@baidu.com

Yangxin Jiang
Baidu Research
Beijing, China
yangxinjiang@baidu.com

Jingbo Zhou*
Baidu Research
Beijing, China
zhoujingbo@baidu.com

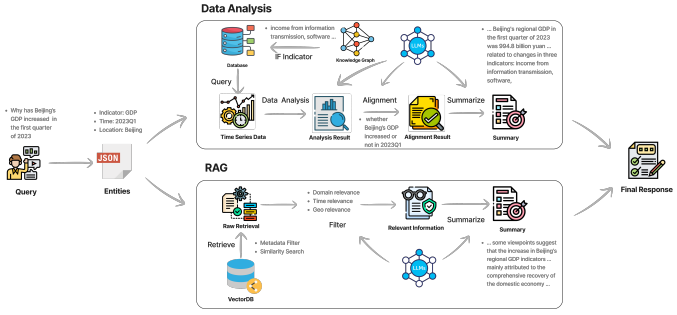

Fig. 1. ARAG System Overview.

***Abstract*—Recent advancements in Large Language Models (LLMs) have significantly impacted the field of question answering systems, particularly with LLM-based data analysis and Retrieval-Augmented Generation (RAG). Yet, applying them independently has limited their effectiveness in scenarios that require a synthesis of both data analysis and contemporary information retrieval. To bridge this gap, we introduce the Analysis and Retrieval Augmented Generation (ARAG) framework, which integrates data analysis with the retrieval of up-to-date information. Based on the framework, we build a system to showcase how ARAG interprets the dynamics of socioeconomic indicators by examining correlated data and retrieving relevant information from news sources. The comparison of ARAG with ChatGPT Search and Perplexity showed that ARAG significantly outperformed them in delivering in-depth analytical insights. Moreover, ARAG is observed to have a stronger ability to verify facts and reject misinformation in users' queries, thus reducing LLM's susceptibility to hallucination.**

***Index Terms*—RAG, data analysis, LLM**

I. Introduction

Recent developments in the field of large language models (LLMs), highlighted by the introduction of models like the GPT series [1], the Erinnie-bot series, and Gemini, have signified a groundbreaking evolution in AI technology. These models demonstrate extraordinary capabilities in engaging in complex dialogues, exhibiting advanced reasoning skills, and producing diverse, creative content. These advancements significantly enhance question-answering applications, especially in data analysis and information retrieval tasks, paving the way for two research directions: LLM-powered data analysis and retrieval-augmented generation (RAG).

Leveraging LLMs can significantly lower the barriers to analyzing data. By transforming natural language into machine-readable representations, such as SQL and Python, LLM-powered data analysis systems can execute commands directly on databases, enhancing the accessibility and efficiency for non-experts. Recent studies have been conducted to employ LLMs in various types of data analysis, including queries [2], visualizations [3], [4], and exploratory insights [5]. However, one limitation of these approaches is that the generated code is mostly applicable to tables or other structured data formats, which may restrict their utility in dealing with unstructured or semi-structured data sources like texts or webpages.

The primary objective of RAG is to reduce the hallucination problem of LLMs. After their training phase, most LLMs' parameters remain static, making them vulnerable to becoming outdated as new information emerges. RAG is particularly beneficial for hallucination reduction because it dynamically retrieve information from external knowledge sources. This information is then incorporated into the context-based learning of LLMs, serving as a reference to generate snippets and citations for responses [6]. However, a limitation of RAG is its primary focus on the retrieval of textual data, while studies highlighting that LLMs often fall short in comprehending and interpreting structured or tabulated data [2].

Despite advancements in LLM-powered data analysis and RAG, these solutions are often implemented separately. This separation limits their ability to effectively handle tasks requiring both textual information retrieval and structured data analysis, such as stock price attribution analysis or socioeconomic indicator interpretation. For example, if we ask why Beijing's GDP grew in the first quarter of 2023 compared to 2022 (i.e., interpreting the dynamics of socioeconomic indicators), the question demands both data analysis and information retrieval. GDP changes are influenced by factors like the ser-

*Jingbo Zhou is the corresponding author. This research was supported in part by the National Natural Science Foundation of China under Grant No.92370204.

vice sector's income, requiring the analysis of corresponding data. Additionally, external factors like public health crises, notably COVID-19, which are primarily documented in news articles or official reports, can also significantly influence GDP. Effectively addressing such queries requires both numerical data analysis and text-based information extraction. Therefore, how to integrate data analysis with RAG method into a unified system is essential.

Here, we propose the Analysis and Retrieval Augmented Generation (ARAG) framework, merging LLM-powered data analysis and RAG into one system (Fig. 1). To showcase the capabilities of the ARAG framework, we developed a system tailored for interpreting the dynamics of socioeconomic data like GDP. The system analyzes related metrics to investigate indicator dynamics, offering data analysis support to explain their variations. Furthermore, it retrieves relevant events and expert commentary from news sources, providing news support for a comprehensive contextual understanding. Our experiments demonstrated that ARAG significantly outperformed Perplexity and the newly released ChatGPT Search in generating high-quality responses and accurately detecting misinformation in user queries. Although our system is primarily designed for socioeconomic analysis, the ARAG framework extends to a wide range of applications. It introduces an innovative strategy for addressing queries by merging insights from both structured and unstructured data sources.

II. System Overview

Fig. 2 presents the process flow of the proposed ARAG system. This system consists of two primary modules: the Data Analysis Module and the RAG Module. Prior to the operation of these two modules, we first employ LLMs to identify vital entities present in the user's query. These entities include geographical locations (e.g., "Beijing"), temporal references (e.g., "2023Q1"), and specific indicators (e.g., "GDP"). This preparatory step ensures the system can more effectively and precisely respond to and process the user's question, enhancing the overall functionality of both the Data Analysis Module and the RAG Module.

A. Data Analysis Module

The Data Analysis module is a crucial component designed to analyze data corresponding to the indicators (i.e., the main indicator) specified in the user's query, as well as influencing indicators associated with each main indicator (i.e., IF indicators). When identifying the main indicators in users' queries, a key challenge is that entities extracted from the queries may not exactly match terms in the database. For example, "GDP" in a query might not match with a table column named "Gross Domestic Product." To solve this, we use vector search to find the top three most similar column names to the queried entity. Subsequently, LLMs are used to pinpoint the most accurate indicator name corresponding to the entity. After identifying the appropriate column name for the indicator, the system proceeds to analyze the changes in the data for the specific time period mentioned in the query (e.g.,

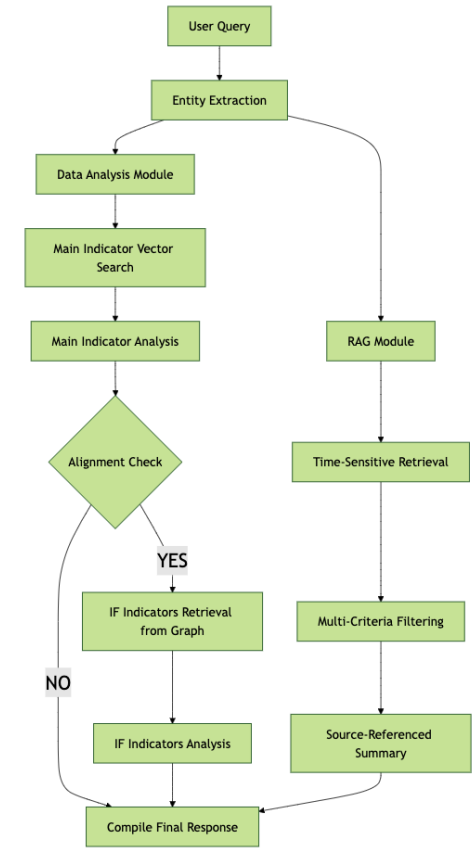

Fig. 2. Flowchart illustrating the process flow of ARAG

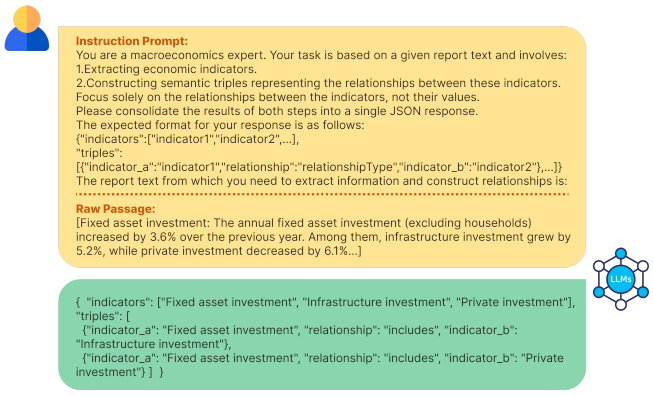

Fig. 3. An example of extracting indicator relationships using LLMs. To enhance clarity, the prompt is divided into two parts: an instruction prompt guiding the LLMs in relationship extraction, and a raw passage sourced from the Internet for processing.

the year-on-year change in GDP for Q1 2023). Furthermore, the system utilizes a pre-defined knowledge graph to identify IF indicators associated with the main indicator and then analyzes the changes in these IF indicators during the same period. The analysis is facilitated by leveraging the capabilities of LLMs to generate Python code. The pre-defined knowledge graph plays a central role in this process. As depicted in Fig. 3, the connections between socioeconomic indicators are extracted from a compilation of government reports and economic news with LLMs. For example, LLMs identify that the indicator "fixed asset investment" is influenced by the indicator "infrastructure investment" based on one passage of government reports shown in Fig. 3.

Before the final summarization of the indicators' data analysis, the data analysis results are processed through a tailored alignment procedure we devised, consisting of two key steps. The first step utilizes LLMs to double-check whether the observed changes in the indicators match what is mentioned in the user's query. For instance, if the user's question concerns why GDP growth in Q1 of 2023 but the system's analysis indicates a decrease, this discrepancy is passed to the summarization step to inform the user accordingly. This process helps prevent the system from "hallucination" due to potentially incorrect assumptions in the user query. The second step of the alignment leverages LLMs to compile the relations of changes among the indicators, aiming to verify the relevance of IF indicator changes to the user's inquiry. This step ensures that the system's responses are precise and closely tailored to the user's needs, offering a comprehensive and targeted analysis rather than merely listing the results of related indicators.

B. RAG Module

The RAG module is designed to retrieve a set of relevant documents that are related to the user's query and facilitate the reasoning over up-to-date information. Inspired by the modular RAG design proposed in recent research, the RAG module of our system breaks down in three steps: retrieval, selection, and summary.

Step 1: Time-Sensitive Retrieval. This initial stage in the RAG process focuses on retrieving relevant data after preprocessing tasks such as chunking and indexing. The user's query is transformed into a vector using the same embedding model employed in the indexing process. The system then calculates the similarity between this query vector and the embeddings of document chunks. Recognizing the time-sensitive nature of many queries, such as those seeking information on a specific quarter's GDP, we implemented a time-sensitive retrieval approach. This method enhances retrieval efficiency by focusing on documents published within a relevant time frame, as including data from other periods could lead to inefficiency and the retrieval of irrelevant chunks. For instance, news about a quarter's GDP is typically published in that quarter or the following one, due to delays in data release. Our system incorporates metadata like publication dates into the chunks, allowing it to first filter for chunks within the targeted time window before performing the retrieval. This approach ensures both efficiency and relevance in the information gathered.

Step 2: Multi-Criteria Filter. This step employs LLMs to rigorously assess each content chunk retrieved in the initial step, ensuring its relevance to the user's query [6]. In particular, LLMs evaluate each content piece based on three key criteria: (1) Relevance to the Query Indicator: The LLM evaluates if the content of the chunk directly pertains to the key topics or indicators mentioned in the user's query. (2) Temporal Alignment: The LLM checks whether the time frame referenced within the chunk corresponds with the time period specified in the user's query. (3) Geographical Relevance: The LLM determines if the locations mentioned in the chunk are consistent with the geographical context of the user's query. Only those chunks that satisfy all three criteria are selected to proceed to the subsequent summarization step. This rigorous filtering process ensures that the final summary is both pertinent and tailored to the specific needs of the user's query.

Step 3: Source-Referenced Summary. The summary step is the generation phase in the RAG process, where the user's query is addressed using information filtered through the preceding retrieval and selection steps, aided by LLMs. To mitigate the risk of generating inaccurate or unrelated content ("hallucination"), the LLMs are required to consider the metadata of each chunk, during the summarization, which includes the title of the source document, the website name, URL, etc. By incorporating this contextual information into each response, the reliability of the summaries is enhanced, further preventing the creation of content that exceeds the scope of the retrieved data.

III. Demonstration Overview

We demonstrate the use of our system in interpreting the dynamics of socioeconomic data like GDP. The statistical data are publicly available from China's National Bureau of Statistics (https://data.stats.gov.cn/https://data.stats.gov.cn/). We use ERNIE-Bot 4.0[1] as our base LLM. For the vector search in our demonstration, we utilized LlamaIndex and Chromadb to manage and query the vector data. The summary of ARAG's response time to 50 queries is presented in Table I). Since both the original data and news corpus are in Chinese, we implemented a translation module based on ERNIE-Bot 4.0 to facilitate accessibility for non-Chinese readers (as shown in the demonstration video). This module translates the user's query from English to Chinese, and subsequently, it translates the system's response from Chinese back to English. Due to the additional interactions with the LLM for translation, the ARAG system's response time in the demonstration video is longer than the response time reported in Table I.

A. Performance Evaluation

To assess the performance of the ARAG system, we conducted a comparative analysis with ChatGPT-4o Search and

[1]https://cloud.baidu.com/doc/WENXINWORKSHOP/s/clntwmv7t

TABLE I
RESPONSE TIME METRICS OF ARAG SYSTEM

Metrics	Min	Max	Average	Median
Statistics	2.7188	4.9169	3.5433	3.4953

Perplexity [2]. Considering that there are no standard answers for tasks akin to data interpretation and research has demonstrated that LLMs are adept at data annotation and evaluation. We adopted a methodology similar to the G-Eval study [7], where an LLM is used to evaluate the responses of ARAG, ChatGPT-4o Search, and Perplexity across four metrics—Domain Relevance, Time Relevance, Information Richness, and Analytical Depth—each rated on a scale of 0 to 5. Domain Relevance measures the alignment of the response with the economic domain or indicators specified in user's query. Time Relevance assesses how well the response corresponds to the specific time period mentioned. Information Richness evaluates the comprehensiveness of the information provided. Analytical Depth evaluates the complexity and depth of the analysis provided in the response. Fig. 4 showcases the evaluation results over 50 questions using ERNIE-Bot 4.0, with each question evaluated five times for robustness. The outcomes illustrate that ARAG consistently scored higher than ChatGPT-4o Search and Perplexity across all dimensions, indicating a more effective performance for combining data analysis and RAG method.

We also conducted an ablation study to assess the impact on system performance following the removal of the RAG component and the Analysis component over the same 50 questions mentioned above. As shown in Fig. 5, this ablation study demonstrates that the removal of RAG and data analysis (i.e., w/o A) components has a detrimental effect on performance across all dimensions, with ARAG scoring highest overall. Without the RAG component, there is a notable decline in all aspects, particularly in Analytical Depth, suggesting its critical role in enhancing the system's ability to provide deep analysis and insights. Furthermore, the absence of the analysis component significantly diminishes the scores in Domain Relevance and Information Richness, confirming its substantial influence in providing more professional and comprehensive insights through the quantitative analysis of changes in related indicators.

TABLE II
HALLUCINATION TEST FOR ARAG, GPT-4O SEARCH, AND PERPLEXITY.

Systems	***ARAG***	***GPT-4o Search***	***Perplexity***
Success Rate	20/20	8/20	0/20

We further conducted adversarial hallucination tests on three systems (ARGA, ChatGPT-4o Search, and Perplexity) by posing 20 factually incorrect queries, such as asking why Beijing's GDP declined in the first quarter of 2023 (Beijing's GDP actually increased in 2023 Q1), to assess their ability to detect misinformation. As shown in Table II, ARAG demonstrated superior accuracy by detecting all inaccuracies (20/20) using rigorously analyzed data. ChatGPT-4o Search identified errors in 8 out of 20 queries, whereas Perplexity failed to detect any inaccuracies. When ChatGPT-4o Search and Perplexity generated incorrect responses, they either directly accepted and responded to the flawed query or referenced data from an inaccurate time period or geographic region.

[2]Full comparison and evaluation methods are available at https://github.com/Ethan-yxx/ARAG_Eval

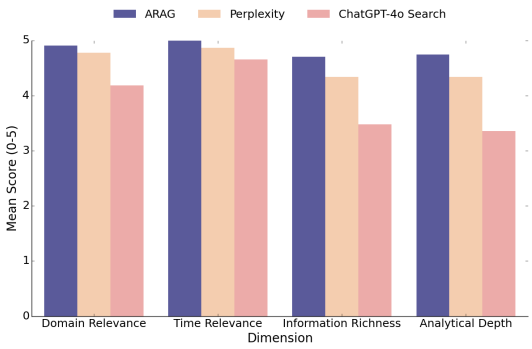

Fig. 4. Comparison of ARAG results with ChatGPT-4o Search and Perplexity.

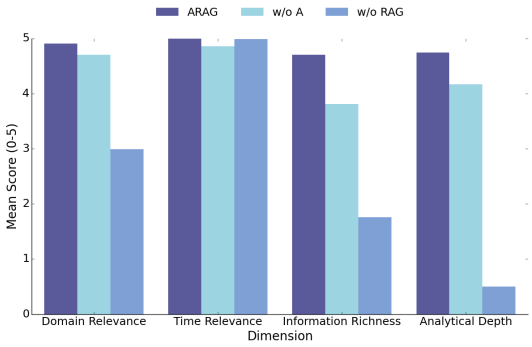

Fig. 5. Ablation study of ARAG.

REFERENCES

[1] P. P. Ray, "ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope," *Internet of Things and Cyber-Physical Systems*, vol. 3, pp. 121–154, 2023.

[2] W. Zhang, Y. Shen, W. Lu, and Y. Zhuang, "Data-copilot: Bridging billions of data and humans with autonomous workflow," in *ICLR 2024 Workshop on Large Language Model (LLM) Agents*.

[3] V. Dibia, "Lida: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models," in *ACL (System Demonstrations)*, 2023, pp. 113–126.

[4] X. Li, J. Zhou, W. Chen, D. Xu, T. Xu, and E. Chen, "Visualization recommendation with prompt-based reprogramming of large language models," in *ACL (Volume 1: Long Papers)*, 2024, pp. 13 250–13 262.

[5] P. Ma, R. Ding, S. Wang, S. Han, and D. Zhang, "InsightPilot: An LLM-Empowered Automated Data Exploration System," in *EMNLP*, 2023, pp. 346–352.

[6] Y. Xia, J. Zhou, Z. Shi, J. Chen, and H. Huang, "Improving retrieval augmented language model with self-reasoning," in *AAAI*, 2025.

[7] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, and C. Zhu, "G-Eval: NLG Evaluation using Gpt-4 with Better Human Alignment," in *EMNLP*, 2023, pp. 2511–2522.