÷

NCU-IISR: Enhancing Biomedical Question Answering with GPT-4 and Retrieval Augmented Generation in BioASQ 12b Phase B

Bing-Chen Chih[1], Jen-Chieh Han[1] and Richard Tzong-Han Tsai[1,2,*]

[1]Department of Computer Science and Information Engineering, National Central University, Taiwan
[2]Research Center for Humanities and Social Sciences, Academia Sinica, Taiwan

Abstract

In this paper, we introduce our system and submissions in BioASQ 12b phase b [1], highlighting a significant improvement with GPT-4 and the integration of Retrieval Augmented Generation (RAG) techniques. We describe our prompt engineering methods and the experimental procedures followed. Because GPT-4 has proven effectiveness in generating answers and its ability in the biological domain, our system utilizes GPT-4 to address biomedical question-answering (QA). Leveraging OpenAI's ChatCompletions API, we refined previous prompt engineering approaches [2] for BioASQ 11b phase b. This year, the addition of RAG techniques significantly improved the information retrieval capabilities of our system. Consequently, our latest submission employed what we experimented to be the most effective prompts and techniques, achieving excellent performance across multiple metrics in the fourth batch.

Keywords

Biomedical Question Answer, Large Language Models (LLMs), Generative Pre-trained Transformer, Retrieval Augmented Generation

1. Introduction

BioASQ [3] has been at the forefront of advancing biomedical semantic indexing and question-answering through its annual challenges since 2013. The 12th time of BioASQ, specifically Task 12b Phase B[1], tasks participants with generating exact or ideal answers to biomedical questions using provided text snippets. This year's training dataset comprises 5,046 questions, which includes the previous year's test set annotations with gold answers, along with 340 new test questions for evaluation. These questions are organized into four batches, each containing 85 questions, meticulously crafted by a team of biomedical experts. The questions in Task 12b Phase B are categorized into four types: yes/no, factoid, list, and summary. Among these, the yes/no, factoid, and list questions require exact answers, while all types require an ideal answer. Participants can submit up to five results per batch, encouraging continuous optimization of their models and techniques. By structuring these rigorous challenges, BioASQ aims to drive innovation and enhance the capabilities of information retrieval systems in the biomedical domain.

Table 1 illustrates examples across four categories in the BioASQ dataset. Each instance contains a question along with several snippets, and answers are categorized into "ideal answer" and "exact answer." Notably, in the "summary" category, there is no requirement for an "exact answer." Last year, we leveraged the understanding capabilities of GPT-4 combined with prompt engineering techniques, achieving great results. This year, we continue to utilize GPT-4's comprehension abilities while incorporating Retrieval-Augmented Generation (RAG) techniques. By harnessing RAG's retrieval capabilities, we enhance the model's domain knowledge, thereby improving output performance. Additionally, we conducted a deeper analysis of the dataset and refined the answer generation approach. Furthermore,

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France
*Corresponding author.
charlie963258@gmail.com (B. Chih); joyhan@cc.ncu.edu.tw (J. Han); thtsai@csie.ncu.edu.tw (R. T. Tsai)

we continued to implement prompt engineering techniques, ensuring a robust and effective approach to improving model responses.

Yes/No	
Question	Can modulation of KCNQ1 splicing prevent arrhythmias?
Exact Answer	yes
Ideal Answer	Amiloride reduces arrhythmogenicity through the modulation of KCNQ1 splicing. Therefore, the modulation of KCNQ1 splicing may help prevent arrhythmias.
List	
Question	Which drugs are included in the AZD7442?
Exact Answer	[tixagevimab, cilgavimab]
Ideal Answer	AZD7442 is a combination of two long-acting monoclonal antibodies tixagevimab and cilgavimab. It has been authorized for the prevention and treatment of coronavirus disease 2019 (COVID-19).
Factoid	
Question	Olokizumab is tested for which disease?
Exact Answer	[rheumatoid arthritis]
Ideal Answer	Olokizumab, a monoclonal antibody against interleukin 6, improves outcomes of rheumatoid arthritis.
Summary	
Question	What is the definition of dermatillomania?
Ideal Answer	Dermatillomania is a condition that leads to repetitive picking of their skin ending up in skin and soft tissue damage. It is a chronic, recurrent, and treatment resistant neuropsychiatric disorder with an underestimated prevalence that has a concerning negative impact on an individual's health and quality of life.

Table 1
Examples across four categories in the BioASQ dataset

2. Related Work

The biomedical domain is characterized by extensive specialized knowledge and complex terminology, making the process of acquiring and applying this information both intricate and time-consuming. Traditional methods often involve reading a substantial number of academic papers, which requires not only significant professional expertise but also considerable effort and time. This approach is inefficient, failing to quickly meet the needs of both professionals and the general public.

Natural Language Processing (NLP) based Question-Answering (QA) systems provide a promising solution to these challenges. By leveraging advanced language models, these systems can interpret, retrieve, and generate responses from medical texts, significantly enhancing the efficiency of QA tasks. Consequently, QA systems streamline the process of accessing biomedical information, making it faster and more efficient for both experts and the general public. With the continuous advancements in deep learning technologies, QA models based on these techniques are progressively bridging the gap between complex biomedical data and practical usability, facilitating broader knowledge dissemination and application.

Prompt Engineering Prompt engineering has emerged as a critical technique in the field of natural

language processing (NLP) and machine learning, particularly in the utilization of large language models like GPT-3 and GPT-4 [4]. This technique involves crafting specific prompts or input queries that guide the language model to produce desired outputs. Various studies have highlighted the effectiveness of prompt engineering in improving the performance of language models across different tasks. For instance, Brown et al. [5] demonstrated that by carefully designing prompts, the accuracy of few-shot learning in GPT-3 significantly increased, enabling the model to perform complex tasks with minimal examples. This approach has been widely adopted in various applications, including question-answering, text summarization, and language translation.

Retrieval Augmented Generation Retrieval Augmented Generation (RAG) is a technique that combines retrieval-based methods with generative models to enhance the relevance and accuracy of generated text. Initially introduced by Lewis et al. [6], RAG has demonstrated significant improvements in open-domain question answering by retrieving relevant documents and using them to inform the generation process.

In the RAG framework, the process typically involves two main components: the retriever and the generator. The retriever is responsible for fetching relevant documents or snippets from a large corpus based on the input query. This is usually achieved using a dense passage retrieval (DPR) model, which encodes both the query and the documents into dense vectors and retrieves the most similar documents. The generator then takes these retrieved documents, concatenates them with the query, and generates a response using a generative model such as GPT-4. This combination allows the generative model to produce more accurate and contextually relevant answers by leveraging the additional context provided by the retrieved documents.

In the biomedical domain, RAG has been particularly beneficial due to the complexity and specificity of the information. Systems utilizing RAG can retrieve pertinent biomedical literature, thus improving the contextual relevance and accuracy of the generated answers. This approach has shown promising results in challenges such as BioASQ.

Our work leverages RAG to enhance our GPT-4-based system for the BioASQ 12b phase B challenge. By integrating RAG, we aim to improve the retrieval and utilization of relevant biomedical documents, ensuring that generated answers are well-supported by accurate and relevant information. This integration enhances the generative capabilities of GPT-4, providing more reliable answers in the biomedical context.

3. Method

3.1. Dataset

The BioASQ Task 12b Phase B dataset [7] provided 5,049 training data samples, comprising 1,210 summary type samples, 1,515 factoid type samples, 967 list type samples, and 1,357 yes/no type samples. Each sample included multiple snippets along with their source documents. Last year, due to the token limit of OpenAI's API, we summarized the snippets and selected the top five snippets. This year, we resolved this practice and found that the performance did not drop, while allowing us to access more comprehensive information.

3.2. Prompting

Snippets: For each question, we incorporate all available snippets as references. Although last year we observed that the top five snippets could cover most of the necessary information, we found that considering all snippets results in more accurate model outputs. Therefore, we provide the model with all snippets listed before the question and prompt for reference. To use Retrieval-Augmented Generation (RAG), we compile all snippets into a database for the model to retrieve relevant information effectively.

Questions: When directly using GPT-4 to generate both the ideal answer and the exact answer simultaneously, we retained most of the prompts from last year. We instructed the model to generate responses in JSON format and to keep the replies as concise as possible. In cases where the ideal answer

and exact answer are generated separately, we first focused on generating a high-quality ideal answer. This is based on our observation that the entities in the exact answer typically appear in the ideal answer. Therefore, the model is first tasked with generating a well-crafted ideal answer, and then it generates the exact answer based on this ideal answer. In both stages, snippets are provided, and during the exact answer generation, few-shot examples are included to ensure accuracy.Please refer to Table 2 for the details of the relevant prompts for generating answer separately, and the other prompts are mostly the same with the prompts we used last year.

In Batch-4, we observed that the ideal answers from previous tests often contained complete segments from the snippets. Therefore, in this batch, we instructed the model to duplicate snippet fragments into the ideal answer during generation. This approach resulted in significantly improved outcomes.

Table 2
The prompts using for generating ideal answer and exact answer separately

Tasks	Prompts
Ideal Answer	Reply to the answer clearly and easily in less than 3 sentences. You should read the chat history's content before answer the question. You can directly copy part of the above snippets as part of your answer. The question is: {QUESTION_BODY}
Yes/No	Please answer me only yes or no. You should read the ideal_answer and snippets before answer the question.
List	Please answer me and follow the following rules: 1. Give me a list of precise key entities to answer the question, as clear and concise as possible. 2. You should read the ideal_answer and snippets before answer the question.
Factoid	Please answer me and follow the following rules: 1. Give me a list of precise key entities to answer the question, as clear and concise as possible. 2. The list should contain at least 1 and up to 5 entity names, ordered by decreasing confidence. 3. You should read the ideal_answer and snippets before answer the question.

3.3. Strategy

Our approach primarily consists of two strategies for generating answers: direct generation of both ideal and exact answers, and sequential generation of these answers. This idea draws from the chain-of-thought methodology [**?**]. In cases where we observed that generating exact answers directly often resulted in responses that were either imprecise or overly verbose, we provided few-shot examples to refine the accuracy of the answers. However, our experiments showed that separate generation does not consistently outperform direct generation; in most scenarios, direct generation proved sufficiently effective.

To enhance the model's understanding of the provided snippets or documents, we adopted the Retrieval-Augmented Generation (RAG) [6] technique. In our implementation, we used the OpenAI "text-embedding-ada-002" embedding model for embedding both the query and the snippets, which provided high-quality dense representations and improved retrieval accuracy. The retrieval process involved encoding the input query and the snippets using the OpenAI embedding model, retrieving the top-k(k = 4) most relevant snippets via computing cosine similarity between the query and documents, and then directly concatenating these snippets with the query to form the prompt for GPT-4.

Although there is a risk of retrieving incorrect fragments, our experiments indicate that this risk has minimal impact on the task's overall performance. In our detailed analysis of the dataset, we observed that standard ideal answers from previous years often contained segments identical to those in the snippets. As a result, in Batch-4, we modified our prompts to allow the model to appropriately duplicate snippet fragments into the answers. This adjustment led to improvements in automated evaluations.

Furthermore, we experimented with adjusting the model's temperature settings. We found that setting the temperature to 0 often produced the highest quality outputs, although this setting was not consistently stable and sometimes resulted in suboptimal performance in certain cases.Overall,

our integration of RAG significantly enhanced the performance of our system by providing more contextually relevant and accurate information, thus improving the quality of the generated answers in the BioASQ 12b challenge.

3.4. Systems

We use different systems in different batches. The detailed configuration of each system can be seen in Table 3. Please note that Batch-1 has no recorded configuration due to some errors.

Table 3
All submitted systems' settings. In the generation process, one approach directly outputs both the ideal and exact answers and the other approach involves initially generating the ideal answer, followed by deriving the exact answer based on it. Additionally, in certain configurations, parts of the snippet fragments are incorporated into the prompt to enhance the output.

Batch	System Name	Generation Strategy	RAG	Duplicate
Batch-2, 3	IISR-1	Split	Yes	No
	IISR-2	Direct	No	No
	IISR-3	Split	No	No
	IISR-4	Direct	Yes	No
	IISR-5	Split	Both RAG	No
Batch-4	IISR-1	Split	Yes	Yes
	IISR-2	Direct	No	Yes
	IISR-3	Split	No	Yes
	IISR-4	Direct	Yes	Yes
	IISR-5	Split	Both RAG	Yes

Table 4
The Exact Answers test results on BioASQ. We define FIN scores as the average of Accuracy in Yes/No, MRR in Factoid, and F-Measure in List.

Batch	System	Yes/No		Factoid			List		
		Acc	maF1	SAcc	LAcc	MRR	Precision	Recall	F1
Batch-1	IISR-1	0.8800	0.8792	0.2381	0.2857	0.2540	0.6317	0.5426	0.5692
	IISR-2	0.9200	0.9188	0.2381	0.2381	0.2381	0.5813	0.5066	0.5244
	IISR-3	0.8800	0.8768	0.2381	0.2381	0.2381	0.5417	0.5300	0.5218
	IISR-4	0.9600	0.9589	0.1905	0.1905	0.1905	0.5449	0.5186	0.5208
	IISR-5	0.9200	0.9188	0.2381	0.2381	0.2381	0.5466	0.5309	0.5301
Batch-2	IISR-1	0.9231	0.9150	0.3684	0.3684	0.3684	0.6166	0.4915	0.5261
	IISR-2	0.9231	0.9150	0.5263	0.5263	0.5263	0.5784	0.5247	0.5456
	IISR-3	0.7692	0.7451	0.2632	0.2632	0.2632	0.4981	0.4443	0.4610
	IISR-4	0.9615	0.9585	0.4211	0.4211	0.4211	0.5436	0.4515	0.4840
	IISR-5	0.8846	0.8689	0.4211	0.4211	0.4211	0.5595	0.4947	0.5023
Batch-3	IISR-1	0.9167	0.9111	0.3077	0.3077	0.3077	0.3476	0.6010	0.4118
	IISR-2	0.9583	0.9564	0.4231	0.4231	0.4231	0.5536	0.5483	0.5475
	IISR-3	0.9583	0.9564	0.3077	0.3077	0.3077	0.3316	0.2670	0.2900
	IISR-4	1.0000	1.0000	0.4231	0.4231	0.4231	0.5452	0.5187	0.5247
	IISR-5	0.9583	0.9564	0.3846	0.3846	0.3846	0.4768	0.4664	0.4677
Batch-4	IISR-1	0.8889	0.8782	0.4211	0.4211	0.5132	0.6230	0.6097	0.6103
	IISR-2	0.9630	0.9571	0.5789	0.6316	0.5965	0.6226	0.5594	0.5828
	IISR-3	0.9630	0.9571	0.5263	0.6316	0.5702	0.5558	0.4791	0.5069
	IISR-4	0.9259	0.9112	0.5789	0.6316	0.5965	0.6735	0.6393	0.6460
	IISR-5	0.9259	0.9112	0.4211	0.5789	0.4649	0.5839	0.5207	0.5442

4. Result

Our results are presented separately for Exact Answer (Table 4) and Ideal Answer (Table 5). We observed significant improvements in Batch-4, where prompt modifications and strategy adjustments were implemented. These changes led to noticeable performance improvement. Although the Manual Score has not yet been released, we achieved competitive rankings across various metrics in the automated evaluations. Our experiments demonstrated that employing Retrieval-Augmented Generation (RAG) to enhance domain knowledge comprehension significantly benefited our Question Answering system, proving to be a crucial component worth considering.

Table 5
The Ideal Answers test results on BioASQ.

Batch	System	R-2 (Rec)	R-2 (F1)	R-SU4 (Rec)	R-SU4 (F1)
Batch-1	IISR-1	0.2796	0.2339	0.2826	0.2292
	IISR-2	0.2625	0.1980	0.2660	0.1907
	IISR-3	0.3379	0.1366	0.3400	0.1270
	IISR-4	0.3675	0.1289	0.3774	0.1197
	IISR-5	0.3750	0.1442	0.3924	0.1369
Batch-2	IISR-1	0.3439	0.1801	0.3657	0.1725
	IISR-2	0.2365	0.1840	0.2457	0.1786
	IISR-3	0.2338	0.1421	0.2577	0.1448
	IISR-4	0.2840	0.2124	0.2962	0.2065
	IISR-5	0.3382	0.1818	0.3548	0.1731
Batch-3	IISR-1	0.4157	0.2480	0.4154	0.2343
	IISR-2	0.3321	0.2890	0.3284	0.2775
	IISR-3	0.3192	0.2225	0.3221	0.2139
	IISR-4	0.3818	0.3170	0.3753	0.3016
	IISR-5	0.4187	0.2493	0.4175	0.2340
Batch-4	IISR-1	0.4141	0.3451	0.4045	0.3250
	IISR-2	0.4120	0.3475	0.4008	0.3286
	IISR-3	0.4161	0.2998	0.4025	0.2794
	IISR-4	0.4398	0.3697	0.4208	0.3458
	IISR-5	0.4188	0.3505	0.4115	0.3310

5. Discussion and Conclusions

In this task, we conducted extensive testing on various techniques. Initially, we considered that focusing the model on generating only one type of answer might yield better performance. However, our experiments revealed that the model's ability to simultaneously generate both ideal and exact answers was equally effective. By utilizing the Retrieval-Augmented Generation (RAG) technique, we calculated the similarity between queries and medical texts, extracting text directly relevant to the questions. This allowed the model to focus on high-quality data when formulating answers. Our experiments confirmed that this approach improved the model's performance. Future work could explore strategies for segmenting snippet documents, which may further enhance effectiveness. Additionally, continuous refinement of prompt engineering and RAG integration could lead to even more significant improvements in answer accuracy and relevance.

References

[1] A. Nentidis, G. Katsimpras, A. Krithara, G. Paliouras, Overview of BioASQ Tasks 12b and Synergy12 in CLEF2024, in: G. Faggioli, N. Ferro, P. Galuščáková, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, 2024.

[2] Y.-W. L. J.-C. H. W. M. R. T.-H. T. Chun-Yu Hsueh, Yu Zhang, Ncu-iisr: Prompt engineering on gpt-4 to stove biological problems in bioasq 11b phase b, CEUR Workshop Proceedings 3497 (2023).

[3] A. Nentidis, G. Katsimpras, A. Krithara, S. Lima-López, E. Farré-Maduell, M. Krallinger, N. Loukachevitch, V. Davydova, E. Tutubalina, G. Paliouras, Overview of BioASQ 2024: The twelfth BioASQ challenge on Large-Scale Biomedical Semantic Indexing and Question Answering, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. Maria Di Nunzio, P. Galuščáková, A. García Seco de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), 2024.

[4] OpenAI, Gpt-4 technical report, 2023. arXiv:2303.08774.

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, arXiv preprint arXiv:2005.14165 (2020).

[6] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive nlp tasks, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 9459–9474. URL: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

[7] A. Krithara, A. Nentidis, K. Bougiatiotis, G. Paliouras, BioASQ-QA: A manually curated corpus for Biomedical Question Answering, Scientific Data 10 (2023) 170.