

Using Pretrained Large Language Model with Prompt Engineering to Answer Biomedical Questions

DS@GT CLEF2024 BioASQ Task 12b and Synergy Task Working Note

Wenxin Zhou¹, Thuy Hang Ngo^{2,*}

¹Georgia Institute of Technology, North Ave NW, Atlanta, GA 30332, United States

Abstract

Our team participated in the BioASQ 2024 Task12b and Synergy tasks to build a system that can answer biomedical questions by retrieving relevant articles and snippets from the PubMed database and generating exact and ideal answers. We propose a two-level information retrieval and question-answering system based on pre-trained large language models (LLM), focused on LLM prompt engineering and response post-processing. We construct prompts with in-context few-shot examples and utilize post-processing techniques like resampling and malformed response detection. We compare the performance of various pre-trained LLM models on this challenge, including Mixtral, OpenAI GPT and Llama2. Our best-performing system achieved 0.14 MAP score on document retrieval, 0.05 MAP score on snippet retrieval, 0.96 F1 score for yes/no questions, 0.38 MRR score for factoid questions and 0.50 F1 score for list questions in Task 12b.

Keywords

large language model, prompt engineering, biomedical information retrieval, biomedical question answering

1. Introduction

BioASQ is a challenge for large-scale biomedical semantic indexing and question answering hosted by CLEF. The BioASQ12b and the Synergy tasks [1] are part of the CLEF 2024 BioASQ lab[2], which focuses on biomedical question answering and information retrieval. The challenge consists of four types of questions: yes/no, factoid, list and summary. The participating systems need to perform two subtasks.

The first subtask is to retrieve 10 relevant documents and snippets from the PubMed database that can answer the question. PubMed [3] is a search engine for biomedical literature, which contains millions of abstracts of biomedical articles. The system is evaluated by the relevance of the retrieved documents and snippets using the mean average precision (MAP) metric.

The second subtask is to generate an exact answer and an ideal answer for each question. The exact answer is a short answer that directly answers the question. For yes/no questions, this is a single word "yes" or "no". For list and factoid questions, the short answer is a list of entities. The ideal answer is a long answer that provides more context and details. The system is evaluated based on the quality and accuracy of the generated answers. The evaluation metric is F1 score for yes/no questions, mean reciprocal rank (MRR) for factoid questions and F1 score for list questions. The ideal answer is scored manually based on the readability, recall, precision and repetition of the answers.

An example of the input and output format is shown in Figure 1. The organizers provide BioASQ-QA dataset[4], which contains around 4721 questions from the past BioASQ challenges where 27% are yes/no questions, 29% factoid, 24% summary and 20% list.

We build a system based on pre-trained large language models for document retrieval and question-answering. Although some solutions of previous years used large language models, they only experimented with OpenAI GPT models and basic prompt engineering. In this year's challenge, we experiment with various well-known large language models and use prompt engineering and response

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09-12, 2024, Grenoble, France

*Corresponding author.

✉ wzhou77@gatech.edu (W. Zhou); tngo62@gatech.edu (T. H. Ngo)

ORCID 0009-0002-3325-3357 (W. Zhou)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Question: List five applications of machine learning algorithms in medical diagnosis.

Documents: 24286700, 24263362

Snippets:

1. The algorithm provides excellent discrimination of PD patients from PSP patients at an individual level, thus encouraging the application of computer-based diagnosis in clinical practice.
2. The objective of this project was the development and validation of a multiparameter machine learning algorithm and system capable of predicting the need for life-saving interventions (LSIs) in trauma patients.

Ideal answer:

Machine learning technology is well suited for the induction of diagnostic and prognostic rules and solving of small and specialized diagnostic and prognostic problems. There many several application of machine learning algorithms in medical diagnosis such as Brain glioma progression, Microarray classification, Mass spectral proteomics, Lymph disease classification and Parkinson's disease.

Exact answer: ["Brain glioma progression", "Microarray classification", "Mass spectral proteomics", "Lymph disease classification", "Parkinson's disease"]

Machine learning on brain MRI data for differential diagnosis of Parkinson's disease and Progressive Supranuclear Palsy

C Salvatore ¹, A Cerasa ², I Castiglioni ³, F Gallivanone ⁴, A Augimeri ⁵, M Lopez ⁶, G Arabia ⁷, M Morelli ⁷, M C Gilardi ⁴, A Quattrone ⁸

Affiliations + expand

PMID: 24286700 DOI: 10.1016/j.jneumeth.2013.11.016

Abstract

Background: Supervised machine learning has been proposed as a revolutionary approach for identifying sensitive medical image biomarkers (or combination of them) allowing for automatic diagnosis of individual subjects. The aim of this work was to assess the feasibility of a supervised machine learning algorithm for the assisted diagnosis of patients with clinically diagnosed Parkinson's disease (PD) and Progressive Supranuclear Palsy (PSP).

Method: Morphological T1-weighted Magnetic Resonance Images (MRIs) of PD patients (28), PSP patients (28) and healthy control subjects (28) were used by a supervised machine learning algorithm based on the combination of Principal Components Analysis as feature extraction technique and on Support Vector Machines as classification algorithm. The algorithm was able to obtain voxel-based morphological biomarkers of PD and PSP.

Results: The algorithm allowed individual diagnosis of PD versus controls, PSP versus controls and PSP versus PD with an Accuracy, Specificity and Sensitivity-90%. Voxels influencing classification between PD and PSP patients involved midbrain, pons, corpus callosum and thalamus, four critical regions known to be strongly involved in the pathophysiological mechanisms of PSP.

Comparison with existing methods: Classification accuracy of individual PSP patients was consistent with previous manual morphological metrics and with other supervised machine learning application to MRI data, whereas accuracy in the detection of individual PD patients was significantly higher with our classification method.

Conclusions: The algorithm provides excellent discrimination of PD patients from PSP patients at an individual level, thus encouraging the application of computer-based diagnosis in clinical practice.

Keywords: Machine learning; Magnetic resonance imaging (MRI); Parkinson's disease (PD); Progressive Supranuclear Palsy (PSP); Support Vector Machine (SVM).

Figure 1: BioASQ Data Format

post-processing techniques to improve the performance of the system. At high level, we use LLM model to extract keywords from the question and compose PubMed query to retrieve documents from PubMed database, then use sentence embeddings to find the relevant snippets from the documents. For question answering, we use the snippets as context and construct few-shot examples prompts to guide the LLM to generate the answers in the desired format. In this paper, we will discuss the modeling pipeline, prompt engineering strategies as well as the experiment results with various LLM models on the Synergy and Task12b tasks. Our implementation can be found on Github¹.

2. Related Work

Large language models (LLM) have shown great success recently in various natural language processing tasks, including text generation in the biomedical domain. Chen et al. [5] measured the performance of LLM on Biomedical Language Understanding and Reasoning Benchmark (BLURB), demonstrating the potential of LLM in understanding and reasoning in the biomedical domain. Prompt engineering [6] is a technique that improves the performance of LLM for domain specific tasks. In-context few-shot examples in the prompt can help LLM to generate more accurate answers, without the need for fine-tuning the model. Some well-known LLM models include OpenAI GPT [7], Meta Llama2 [8], and Mistral AI's Mixtral [9] models.

2.1. Information Retrieval Approaches in BioASQ 2023

In the eleventh BioASQ challenge, two predominant methodologies were employed for the Information Retrieval (IR) task, typically segmented into a two-stage pipeline: retrieval and reranking.

For the retrieval stage, the majority of the systems (7 out of 8) involved in Task 11B phase A adopted a BM25 model for the initial document retrieval [10]. BM25 models rely on indexing the entire corpus of documents which is computationally expensive since more than a million biomedical papers are added to the PubMed database each year [11]. The advantage of this method is the comprehensive list of documents that could be retrieved.

¹<https://github.com/dsgt-kaggle-clef/bioasq-2024/>

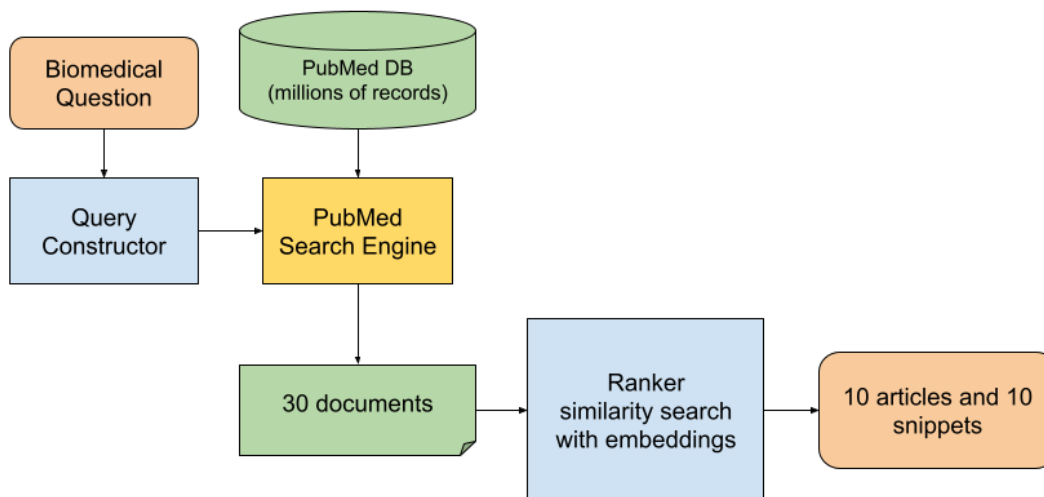


Figure 2: Information Retrieval System Design

In contrast, Ateia and Kruschwitz [12] utilized LLMs for retrieval via zero-shot learning by mirroring the expert’s workflow in curating the BioASQ QA dataset. This method does not require additional computation to index the document corpus, but may not be as comprehensive as the BM25 method. The system did not achieve the highest IR outcomes in last year’s competition.

Most participating systems generated language model embeddings for reranking articles. This process was handled by a dedicated reranking model or through cosine similarity measures [10]. One system employed zero-shot prompting with an LLM to rank the documents, but was resource-intensive and limited by the maximum context length of the model.

2.2. Question Answering Approaches in BioASQ 2023

For the Question Answering (QA) or Phase B task, several systems exploited the capabilities of LLMs by prompting it with essential snippets of information. A comparative analysis revealed that GPT-4 outperformed an ensemble of fine-tuned BERT models, indicating that GPT-4 is more effective at navigating the complexities of biomedical question answering [13].

3. Methodology

3.1. Information Retrieval

We propose a two-stage IR system for the BioASQ task (in Figure 2). The first stage retrieves a set of relevant documents from the PubMed database using PubMed search query [14]. The second stage ranks documents by cosine similarity of sentence embeddings to find the most relevant sentences.

3.1.1. Query Constructor

The query constructor creates the query for PubMed search [3]. To match the PubMed version defined by the organizer, we set the *maxdate* parameter in the *esearch* API to be the date defined by the specific batch of the Synergy task. Task12b requires PubMed 2024 baseline, so we set the *maxdate* parameter to 2024-01-01 as an approximation. The query constructor uses two approaches.

Approach 1: Keyword extraction. We use LLMs or language models finetuned for biomedical terminology to extract keywords (such as biomedical entities) from the question. Then we concatenate those keywords with "AND" to form a PubMed query. For LLMs, we send a few-shot example prompt, shown in Table 1 to generate the keywords. For the biomedical language model, we use *en_ner_bc5cdr_md*, a

spaCy biomedical NER named entity recognition language model trained on BC5CDR corpus [15] to extract the keywords from the question sentence.

Table 1

Query Constructor Prompt, Approach 1

```
Q: What is the mode of action of Molnupiravir?  
Keywords: Molnupiravir, action  
###  
Q: Is dapagliflozin effective for COVID-19?  
Keywords: dapagliflozin, COVID-19  
###  
Q: {question body}  
Keywords:
```

Approach 2: Direct query generation. We directly generate a query from the question using the large language model, which is inspired by Ateia and Kruschwitz [12]. The prompt template is composed of an instruction and two examples, as shown in Table 2.

Table 2

Query Constructor Prompt, Approach 2

```
Given a question, expand into a search query for PubMed by incorporating synonyms and additional terms that would yield relevant search results from PubMed to the provided question while not being too restrictive. Assume that phrases are not stemmed; therefore, generate useful variations. Return only the query that can directly be used without any explanation text.  
  
Question: What is the mode of action of Molnupiravir?  
Query: Molnupiravir AND ("mode of action" OR mechanism)  
###  
Question: Is dapagliflozin effective for COVID-19?  
Query: dapagliflozin AND (COVID-19 OR SARS-CoV-2 OR coronavirus) AND (efficacy OR effective OR treatment)  
###  
Question: Name monoclonal antibody against SLAMF7.  
Query: "SLAMF7" AND ("monoclonal antibody" OR "monoclonal antibodies")  
###  
Question: {body}  
Query:
```

3.1.2. Reranker

The reranker ranks documents by calculating the relevance between documents and questions. We use a sentence transformer, specifically *all-MiniLM-L6-v2* [16] to generate embeddings for the documents and the questions. When the document length is larger than the maximum input length of the sentence transformer, the document is truncated to fit the input length. We calculate the cosine similarity between the embeddings of the question and the document and use the descending score to rank the documents.

3.1.3. Snippet Extraction

After identifying the top 10 documents, we break the documents into sentences and rank the sentences based on the similarity score using the same sentence transformer and similarity calculation method as the re-ranker. We then select the best sentence of each document as the prediction snippets.

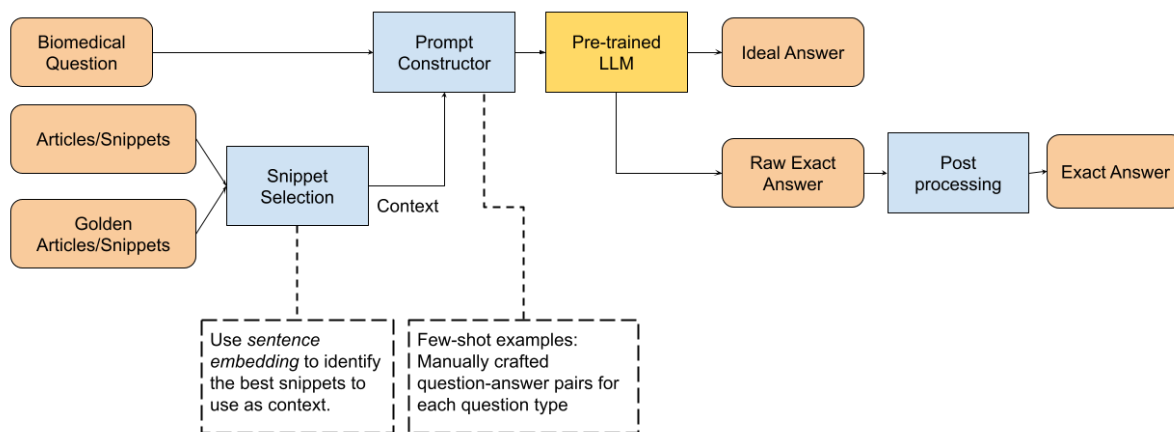


Figure 3: Question Answering System Design

3.2. Question Answering

We use pre-trained large language models (LLM) to generate answers for biomedical questions. In this project, instead of fine-tuning the LLM models, we use prompt engineering and response post-processing to build the system, as illustrated in Figure 3. The two key components of the prompt are the context and the few-shot examples.

We use the first 1000 words of the top 10 snippets for the question, where each snippet consists of one or more sentences of a PubMed abstract, as the context for the question answering. The snippets are generated from the information retrieval system or are the golden snippets provided by the organizers. The reasoning behind using the first 1000 words of snippets is that the higher rank snippets contain the more relevant information to the question. The context is crucial for generating high-quality answers and reducing model hallucinations. Then we construct few-shot examples from the training dataset. The few-shot examples help LLM to generate answers in the desired format. The template prompt used for yes/no questions is shown in Table 3. The maximum input token size of most LLM models we experimented with is larger than 4096, so a prompt consisting of few-shot examples, 1000 words (which is roughly equivalent to 1350 tokens) context as well as the question body is within the LLM input token limit.

Table 3

Prompt Template for Question Answering, Yes/no Question

<p>Context: Papilins are homologous, secreted extracellular matrix proteins which share a common order of protein domains.</p> <p>Question: Is the protein Papilin secreted?</p> <p>Ideal answer: Yes, papilin is a secreted protein</p> <p>Exact answer: yes</p> <p>###</p> <p>Context: Most lncRNAs are under lower sequence constraints than protein-coding genes and lack conserved secondary structures, making it hard to predict them computationally.</p> <p>Question: Are long non coding RNAs as conserved in sequence as protein coding genes?</p> <p>Ideal answer: No. Most long non coding RNAs are under lower sequence constraints than protein-coding genes.</p> <p>Exact answer: no</p> <p>###</p> <p>Context: {context}</p> <p>Question: {question_body}</p> <p>Ideal answer:</p>
--

The prompt templates used for factoid, list and summary questions are similar to the yes/no question

template with some modifications to the ideal and exact answer formatting. Those templates can be found in the appendix A.

"###" is used as the separator for examples in the prompt, which is also used as the "stop" string of the LLM completion. The prompt is intentionally designed to end with "Ideal answer:" to guide the LLM to generate the ideal answer. We expect the "exact answer" line to be generated after the ideal answer in the LLM response as illustrated in appendix B. In terms of the other parameters (such as temperature and top_k) for the LLM completion, we use the default values defined by the TextSynth service [17], except that the *max_tokens* parameter, which controls maximum number of tokens in the LLM output is set to 200. We did not experiment with using different roles or system and user prompts since not all models we experimented with are fined-tuned for chat, so we only relied on the basic completion functionality of LLM to generate the answers.

For list questions, we experiment with synonym grouping. The idea is to group the synonyms among all the LLM responses to reduce the repetition of the answers. This is similar to having LLM perform a second-stage reasoning. The synonym grouping is accomplished by sending a prompt (shown in Table 4) to LLM to group the synonyms. The prompt contains a list of entities, which aggregates all entities returned from multiple responses from LLM for the list question with different prompt contexts.

Table 4
Prompt Template for Synonym Grouping

```
Group the phrases with the same meaning in the ENTITY list into separate lines as follows.  
[ENTITY]: MOG-IgG; AQP4; MOG-IgG; serum neurofilament light chain; NfL; aquaporin-4 (AQP4)-  
immunoglobulin G (IgG)  
[GROUP1]: aquaporin-4 (AQP4)-immunoglobulin G (IgG); AQP4; MOG-IgG  
[GROUP2]: serum neurofilament light chain; NfL  
###  
[ENTITY]: {entity_list}  
[GROUP1]:
```

3.2.1. Context formation

We started with the first 1000 words of the top 10 snippets as the prompt context. The system we submit to Synergy uses this basic version. We experiment with different contexts for the QA system by changing the number and variety of snippets used. Our final context setup used for batch 2 and 3 in Task 12b is as follows:

1. For yes/no questions, we create three prompts with different contexts. The context of the first prompt is the first golden snippet, the context of the second prompt is the second golden snippet, and the context of the third prompt is the third golden snippet. We send all three prompts separately to the LLM and take the majority vote of the answers as the final answer.
2. For factoid and summary questions, we use one prompt with the first 1000 words of the golden snippets as the context.
3. For list questions, we use one prompt with the first 1000 words of the golden snippets as the context. In the synonym grouping setting, we compose five prompts. The context of each prompt is a golden snippet. The first prompt context is the first golden snippet, the second prompt context is the second golden snippet, and so on.

3.2.2. Response Post-processing

Since the prompt we send to LLM has examples with the desired answer format, the answers generated by LLM are usually in the form of a long answer, followed by an exact answer in the second line. We extract answers by parsing the response. For list and factoid questions we give examples where entities

are separated by semicolons, therefore we extract resulting entities by splitting the exact answer string by semicolons. Examples of prompt and response can be found in appendix B. When the answer does not follow the expected format, we resample the LLM output. Some checks we use to detect malformed answers detection are:

- There is no "exact answer" string in the response for yes/no, factoid and list questions.
- The exact answer is not "yes" or "no" for yes/no questions.
- The exact answer for list or factor questions separates entities by commas or newlines instead of semicolons.

4. Results

In this section, we present and discuss the results of our systems for the BioASQ Task 12b and Synergy tasks. The results are based on the official evaluation scores provided by the organizers in the competition leaderboard for BioASQ2024.

4.1. Synergy Task

We submitted five systems for the Synergy task to measure the performance of different pre-trained large language models and strategies. The system configurations are outlined in Table 5.

Table 5
Synergy Task System Configuration

System	Leaderboard Name	IR algorithm	QA Model	QA has context
system1	Gatech competi-tion	spaCy biomedical NER model	Mistral 7B	No
system2	GTBioASQsys2	LLM (Mistral 7B) + few-shot prompt	Mistral 7B	Yes
system3	GTBioASQsys3	LLM (Mixtral 47B) + few-shot prompt	Mixtral 47B Instruct	Yes
system4	GTBioASQsys4	LLM (Llama2 70B) + few-shot prompt	Llama2 70B	Yes
system5	GTBioASQsys5	LLM (GPT-J 6B) + few-shot prompt	GPT-J 6B	Yes

For the information retrieval (IR) part, system 1 uses *en_ner_bc5cdr_md* to extract the keywords from the questions. The rest of the systems use large language generative models (LLM) to extract the keywords. The language models used for systems 2,3,4,5 are Mistral 7B, Mixtral 47B (i.e, Mixtral 8x7B model) [9], llama2 [8], GPT-J [18] respectively.

For the question-answering part, the prompt of system 1 contains no context, whereas the rest of the systems use snippets as the context. The LLM models used for systems 1,2,3,4,5 are Mistral 7B, Mistral 7B, Mixtral 47B, llama2 and GPT-J respectively.

Table 6 shows the information retrieval results for Synergy task round 4. System 3 has the best performance with 0.0434 mean-average precision (MAP) score for document retrieval and 0.031 for snippet retrieval. The performance of systems 2,4 and 5 is similar in MAP score, in the range between 0.02 and 0.03, whereas system 1 is the worst with MAP score of 0.0003. The systems used for the Synergy task only perform the basic first-level retrieval by fetching 10 records from PubMed using a query concatenated by keywords. We can see that Mixtral47B outperforms other systems for the question keyword extraction task. The spaCy language model *en_ner_bc5cdr_md* performs the worst. The reason is that the *en_ner_bc5cdr_md* model is often unable to detect any keywords in the question body since it is limited to detecting only the disease and chemical entities in the sentence.

Table 6

Synergy Round 4 Information Retrieval Results

System	Document Macro-F1	Document MAP	Snippet Macro-F1	Snippet MAP
top competitor	0.1459	0.1308	0.0982	0.1003
system1	0.001	0.0003	0.0014	0.0003
system2	0.0283	0.027	0.0146	0.0215
system3	0.0578	0.0434	0.034	0.0317
system4	0.0355	0.0258	0.0225	0.0249
system5	0.0353	0.0267	0.0273	0.0237

Table 7

Synergy Round 4 Question Answering Results

System	Yes/no F1	Factoid MRR	List F1	Ideal Answer Manual Score (total out of 20)
top competitor	1.0	0.43	0.27	17.41
system1	0.33	0.1818	0.0806	13.56
system2	0.4	0.3182	0.2009	14.91
system3	0.625	0.2727	0.2256	16.4
system4	0.4857	0.2727	0.1919	11.55
system5	0.4857	0.1364	0.1467	9.05

Table 7 shows the question-answering results for Synergy task round 4. Our best-performing system achieved 0.62 F1 score for yes/no questions, 0.3182 on mean reciprocal rank (MRR) metric for factoid questions and 0.22 F1 score for list questions. The Mixtral47B model gives the best performance on all question types except for factoid questions. For factoid questions, system 2 with Mistral 7B model performs the best. Even though the Llama2 model is much larger in model size, it does not perform as well as the Mixtral/Mistral models. By comparing the results of system1 and system2, we can see that adding context to the prompt greatly improves the QA answering accuracy for all types of questions for the same LLM model engine.

The ideal answer is evaluated by manual scores according to the readability, recall, precision and repetition of the answers. Mixtral 47B performs the best in terms of generating long answers. The Llama2 model performs the worst in generating long answers, even though it is on par with other models in exact answer generation. It seems to indicate that the Llama2 model is not good at generating long answers, which could be due to the model architecture differences. It is also possible that the Llama2 model was trained to give vague answers to biomedical questions such that it does not mislead the users on health issues.

4.2. BioASQ Task 12B

For Task 12B, we added the direct query generation method to our experiment. We updated the re-ranker component to get the top 10 documents among the top 30 documents retrieved from the first stage via the PubMed Query for system 1 in batch3. In addition, we enhanced the system by response resampling and adding a fallback to use a query with the original question if LLM keyword extraction fails to generate any keyword or the query generated by GPT-4 [7] returns no results.

4.2.1. Task 12B Phase A

For the three systems we submit to the PhaseA of task 12B, system 1 uses the direct query generation method with GPT-3.5 for batch 1 and GPT-4 for batch 2 and 3. Systems 2 and 3 continue to use the keyword extraction method with Mistral 7B and Mixtral 47B as before. The system configurations are outlined in Table 8.

Table 8

12B System Configurations

System	Leader-board Name	Phase A IR method	Phase A+ QA context	Phase B QA strategy
system1	Gatech competition	Query generation with GPT3.5 or GPT-4	IR snippet	Golden snippet (add synonym grouping for list question in batch 2 and 3)
system2	GTBioASQsys2	Keyword extraction with Mistral 7B	No snippet	Golden snippet
system3	GTBioASQsys3	Keyword extraction with Mixtral 47B	IR snippet	-

In Table 9, all three systems in batch 2 have similar performance in terms of MAP score at around 0.081 for document retrieval, with system 3 having the best performance. For snippet retrieval, system 3 has the best performance with MAP score of 0.0271, followed by system 2 and system 1. The system 1 performance also improved from 0.0497 in batch 1 to 0.081 in batch 2, after switching from using GPT-3.5 to GPT-4 for query generation.

In batch 3, system 1 had a significant improvement of MAP to 0.1385 thanks to increasing the number of articles retrieved from PubMed in the initial retrieval stage from 10 to 30. This allows for more articles to be processed in the reranking stage and results in higher recall overall. In past competitions, solutions that use BM25 models for retrieval fetch hundreds of documents in the initial stage of retrieval [19], these systems also tend to have the best score for IR task. We hypothesize that our systems, which use LLM for the retrieval stage, would have even better performance should the number of articles retrieved initially be increased further to 100. However, due to the time required to fetch the articles from PubMed API, extract the snippets, and score the articles for similarity with the query, increasing the number of retrieved document results in a long wait time.

Table 9

12B Phase A Information Retrieval Results

Batch	System	Document MAP	Snippet MAP
1	top competitor	0.2067	0.1149
	system1	0.0497	0.0150
	system2	0.0599	0.0050
	system3	0.0717	0.0204
2	top competitor	0.2293	0.1539
	system1	0.081	0.0151
	system2	0.0826	0.0261
	system3	0.0836	0.0271
3	top competitor	0.2549	0.2224
	system1	0.1385	0.0505
	system2	0.0883	0.0312
	system3	0.0929	0.0271

4.2.2. Task 12B Phase A+ and Phase B

For the question answering (QA) part, we enhanced our system by adding resampling if the exact answer does not satisfy the requirements. For example, neither "yes" nor "no" is in the answer for yes/no question. We also experimented with setting up different contexts for the QA system, by changing the number of snippets and the variety of the snippets used.

We submit three systems to PhaseA+. Phase A+ system1 uses the snippets generated by system1 in PhaseA as the context for the QA prompt. Phase A+ system 2 does not use any snippet as the context for

the QA prompt. Phase A+ system 3 uses the snippets generated by system 3 in PhaseA as the context. In PhaseB, we use the golden snippet provided by the organizer as the context of the QA prompt for PhaseB system1 and system2. The difference is that PhaseB system1 performs synonym grouping for list questions, whereas PhaseB system2 does not.

Table 10 shows the results of all the five systems in PhaseA+ and PhaseB. Take batch 2 as an example, the system without context (Phase A+ system2) only achieved 0.69 F1 score for yes/no questions. Adding context improves the F1 score to 0.80 for yes/no questions and adding golden snippets as the context further improves the F1 score to 0.96 for yes/no questions. For factoid questions, adding non-golden snippets as context does not improve the MRR score, but adding golden snippets as context improves the MRR score from 0.21 to 0.36. For list questions, adding non-golden snippets as context improves the F1 scores slightly, and adding golden snippets as context further improves the F1 score from 0.21 to 0.50. Batch 1 and 3 results also follow the same pattern.

Table 10
12B Question Answering Results

Batch	System	Yes/no F1	Factoid MRR	List F1
1	Phase A+ top competitor	0.9167	0.3611	0.4808
	Phase A+ system1	0.8397	0.1429	0.3661
	Phase A+ system2	0.7987	0.0952	0.2350
	Phase A+ system3	0.8397	0.1429	0.1866
	Phase B top competitor	0.9589	0.4286	0.5843
	Phase B system1	0.8792	0.1905	0.3975
2	Phase A+ top competitor	0.9601	0.3947	0.4088
	Phase A+ system1	0.8051	0.2105	0.2133
	Phase A+ system2	0.6905	0.2105	0.1268
	Phase A+ system3	0.8003	0.2105	0.1364
	Phase B top competitor	0.9601	0.6842	0.5708
	Phase B system1	0.9601	0.3684	0.2936
3	Phase B system2	0.8802	0.358	0.4964
	Phase A+ top competitor	0.9143	0.3301	0.3708
	Phase A+ system1	0.7822	0.2308	0.1774
	Phase A+ system2	0.6643	0.0769	0.2098
	Phase A+ system3	0.7483	0.1538	0.1860
	Phase B top competitor	1.0	0.5	0.6484
Phase B system1	0.8748	0.3077	0.3399	
Phase B system2	0.9577	0.3846	0.4763	

5. Discussion

The Task12B results show that our systems with golden snippets as the context can achieve an F1 score of 0.87-0.96 for yes/no questions. We improved post-processing steps for factoid and list questions after batch1, by removing duplicate answers and detecting malformed answers. Therefore the MRR score of our final system for factoid question is in the range of 0.3-0.4. The F1 score for list question is 0.45-0.5.

By comparing the list F1 score of PhaseB system1 and system2 in batch 2 and 3, we can see that synonym grouping performs worse than not using synonym grouping. To understand the reasons, we looked at some of the synonym grouping responses from LLM and found that LLM often groups entities that should be in different categories together. Table 11 shows an example prompt and response pair. We can see that "fibromyalgia" and "chronic fatigue syndrome" are grouped as synonyms, and "depression" and "hypermobility spectrum disorders" are grouped as synonyms, whereas they should all be separate entities. As a result, the synonym grouping does not help the list question performance. This also demonstrated that adding second-stage reasoning using LLM does not always give better results for complex problems.

Our key takeaways from the experiments are:

Table 11

Example Response from LLM to Synonym Grouping Prompt

Grouping Prompt:

Group the phrases with the same meaning in the ENTITY list into separate lines as follows.
(examples)
[ENTITY]: fibromyalgia; chronic fatigue syndrome; autosomal dominant polycystic kidney disease; Marfan syndrome; Cutis laxa syndromes; depression; osteogenesis Imperfecta Type 1; Loey-Dietz syndrome; hypermobility spectrum disorders
[GROUP1]:

Grouping Response:

fibromyalgia; chronic fatigue syndrome
[GROUP2]: autosomal dominant polycystic kidney disease; Marfan syndrome; osteogenesis Imperfecta Type 1; Loey-Dietz syndrome
[GROUP3]: Cutis laxa syndromes
[GROUP4]: depression; hypermobility spectrum disorders

1. For IR part, Mixtral 47B is the best-performing model for question keyword extraction among the models we have tested. Retrieving more documents in the initial retrieval stage can improve the performance of the system.
2. For QA part, adding context, especially using "correct" snippets as the context, to the prompt can greatly improve the QA answering accuracy for all types of questions.
3. The improvements of the QA scores in batch 2 and 3 in Task 12B demonstrate that resampling LLM response is a great technique to improve accuracy. Simple response post-processing steps to validate the output format can also improve the performance.
4. By comparing the results of Llama2 and other models in the Synergy task, we found that Llama2 model is not good at generating long answers, even though it is on par with other models in exact answer generation.
5. Two-stage LLM reasoning does not always give better results for complex problems, as shown by the synonym grouping experiment.

6. Future Work

Here are some ideas for future work to improve the performance of our systems.

For the IR part, currently, we only fetch a small amount of documents from PubMed and use embeddings to rank the documents. We found that increasing the number of documents fetched in the initial retrieval stage improves the recall and overall MAP score but leads to long processing time. Calculating embeddings on the fly is especially time-consuming. In the future, we want to embed all the PubMed documents in advance and store the embeddings in a vector database. In this way, we can fetch more documents in the first stage retrieval for second stage reranking as we would be able to look up the embeddings for a specific document quickly. We can also use similarity search on the vector database to directly fetch relevant documents for the question.

When calculating the similarity between the question and the document, we only use the first part of the document, which fits the embedding model input token size. We want to investigate if splitting the documents into multiple parts and calculating the similarity for each part can improve the performance. We can also explore the performance of using different sentence embedding models.

For the QA part, our current system is based on the few-shot examples to guide the LLM to generate answers. We only used a few training examples in the BioASQ dataset and have not utilized the potential

of the BioASQ dataset. The next step would be to fine-tune the pretrained LLM model (specifically Mixtral47B) on the BioASQ dataset. The experience of crafting examples for prompt engineering can help us prepare training data for fine-tuning LLM. We will consider using Low Rank Adaptation (LoRA) [20] as a cost-effective method for finetuning a model with a large number of parameters.

7. Conclusions

We implemented information retrieval and question-answering systems for the BioASQ Task 12b and Synergy tasks. The information retrieval system uses pretrained LLM and prompt engineering to search documents and uses sentence embeddings to rank documents. The question answering system uses in-context few-shot examples to guide the LLM to generate answers while passing article snippets as context. Our final system incorporates several useful techniques such as resampling and response post-processing for LLM interaction. We experimented with various state-of-the-art LLM models, compared their performance and found that Mixtral 47B is overall the best-performing model. Our best-performing system achieved 0.14 MAP score on document retrieval, 0.05 MAP score on snippet retrieval, 0.96 F1 score for yes/no questions, 0.38 MRR score for factoid questions, and 0.50 F1 score for list questions in Task 12b. We hope this work can provide insights for future research in building biomedical question answering systems using large language models.

Acknowledgements

Thank you to the Data Science @ Georgia Tech (DS@GT CLEF) team and Anthony Miyaguchi for their support. We acknowledge the use of Grammarly [21] to proofread this paper.

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] 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. Goeriot, 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.
- [3] Pubmed overview (2023). URL: <https://pubmed.ncbi.nlm.nih.gov/about/>.
- [4] A. Krithara, A. Nentidis, K. Bougiatiotis, G. Paliouras, BioASQ-QA: A manually curated corpus for Biomedical Question Answering, *Scientific Data* 10 (2023) 170.
- [5] Q. Chen, H. Sun, H. Liu, Y. Jiang, T. Ran, X. Jin, X. Xiao, Z. Lin, H. Chen, Z. Niu, An extensive benchmark study on biomedical text generation and mining with ChatGPT, *Bioinformatics* 39 (2023) btad557. URL: <https://doi.org/10.1093/bioinformatics/btad557>. doi:10.1093/bioinformatics/btad557.
- [6] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, D. Schmidt, A prompt pattern catalog to enhance prompt engineering with chatgpt, 2023. doi:10.48550/arXiv.2302.11382.
- [7] OpenAI, Gpt-4 a large-scale transformer-based language model, OpenAI (2023). URL: <https://chat.openai.com>.
- [8] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, Llama open and efficient foundation language models, *arXiv preprint arXiv:2302.13971* (2023). URL: <https://arxiv.org/abs/2302.13971>.

- [9] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, Mixtral of experts, arXiv preprint arXiv:2401.04088 (2024). URL: <https://arxiv.org/abs/2401.04088>.
- [10] A. Nentidis, G. Katsimpras, A. Krithara, S. L. López, E. Farré-Maduell, L. Gasco, M. Krallinger, G. Paliouras, Overview of bioasq 2023: The eleventh bioasq challenge on large-scale biomedical semantic indexing and question answering (2023). URL: <https://arxiv.org/abs/2307.05131>.
- [11] E. Landhuis, Scientific literature: Information overload, *Nature* 535 (2016) 457–458. URL: <https://www.nature.com/articles/nj7612-457a>. doi:<https://doi.org/10.1038/nj7612-457a>, published online 20 July 2016.
- [12] S. Ateia, U. Kruschwitz, Is chatgpt a biomedical expert? exploring the zero-shot performance of current gpt models in biomedical tasks, *CEUR Workshop Proceedings* 3497 (2023). URL: <https://ceur-ws.org/Vol-3497/paper-006.pdf>.
- [13] H. Kim, H. Hwang, C. Lee, W. Y. Minju Seo, J. Kang, Exploring approaches to answer biomedical questions: From pre-processing to gpt-4 (2023). URL: <https://ceur-ws.org/Vol-3497/paper-011.pdf>.
- [14] E. Sayers, A general introduction to the e-utilities, National Center for Biotechnology Information (US) (2009). URL: <https://www.ncbi.nlm.nih.gov/books/NBK25497/>.
- [15] M. Neumann, D. King, I. Beltagy, W. Ammar, ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing (2019) 319–327. URL: <https://www.aclweb.org/anthology/W19-5034>. doi:10.18653/v1/W19-5034. arXiv:arXiv:1902.07669.
- [16] H. Face, sentence-transformers/all-minilm-l6-v2, Hugging Face Community week (2021). URL: <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>.
- [17] Textsynth documentation (2024). URL: <https://textsynth.com/documentation.html>.
- [18] B. Wang, A. Komatsuzaki, GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model (2021).
- [19] G. H. Maël Lesavourey, Bioasq 11b: Integrating domain specific vocabulary bert-based model for biomedical document reranking, *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023)* (2023). URL: <https://ceur-ws.org/Vol-3497/>.
- [20] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-rank adaptation of large language models (2022). URL: <https://openreview.net/forum?id=nZeVKeeFYf9>.
- [21] Grammarly, Grammarly handbook (2024). URL: <https://www.grammarly.com/handbook/>.

A. Prompt Templates

Table 12

Prompt Template for List Questions

<p>Context: The FGFR3 P250R mutation was the single largest contributor (24%) to the genetic group; Syndromic craniosynostosis due to complex chromosome 5 rearrangement and MSX2 gene triplication</p> <p>Question: Which human genes are more commonly related to craniosynostosis?</p> <p>Ideal answer: The genes that are most commonly linked to craniosynostoses are the members of the Fibroblast Growth Factor Receptor family FGFR3 and to a lesser extent FGFR1 and FGFR2. Some variants of the disease have been associated with the triplication of the MSX2 gene and mutations in NELL-1. NELL-1 is being regulated by RUNX2, which has also been associated to cases of craniosynostosis. Other genes reported to have a role in the development of the disease are RECQL4, TWIST, SOX6 and GNAS.</p> <p>Exact answer: FGFR3;FGFR2;FGFR1;MSX2;NELL1;RUNX2;RECQL4;TWIST;SOX6;GNAS</p> <p>###</p> <p>Context: The current article presents a concise review of network theory and its application to the characterization of AED use in children with refractory epilepsy;</p> <p>Recent results suggest that LCM has a dual mode of action underlying its anticonvulsant and analgesic activity.</p> <p>Question: What are the main indications of lacosamide?</p> <p>Ideal answer: Lacosamide is an anti-epileptic drug, licensed for refractory partial-onset seizures. In addition to this, it has demonstrated analgesic activity in various animal models. Apart from this, LCM has demonstrated potent effects in animal models for a variety of CNS disorders like schizophrenia and stress induced anxiety.</p> <p>Exact answer: refractory epilepsy;analgesic;CNS disorders</p> <p>###</p> <p>Context: {context}</p> <p>Question: {body}</p> <p>Ideal answer:</p>
--

Table 13

Prompt Template for Factoid Questions

<p>Context: Ewing sarcoma is the second most common bone malignancy in children and young adults. It is driven by oncogenic fusion proteins (i.e. EWS/FLI1) acting as aberrant transcription factors that upregulate and downregulate target genes, leading to cellular transformation; Ewing sarcoma/primitive neuroectodermal tumors (EWS/PNET) are characterized by specific chromosomal translocations most often generating a chimeric EWS/FLI-1 gene</p> <p>Question: Which fusion protein is involved in the development of Ewing sarcoma?</p> <p>Ideal answer: Ewing sarcoma is the second most common bone malignancy in children and young adults. In almost 95% of the cases, it is driven by oncogenic fusion protein EWS/FLI1, which acts as an aberrant transcription factor, that upregulates or downregulates target genes, leading to cellular transformation.</p> <p>Exact answer: EWS;FLI1</p> <p>###</p> <p>Context: Acrokeratosis paraneoplastica of Bazex is a rare but important paraneoplastic dermatosis, usually manifesting as psoriasiform rashes over the acral sites Bazex syndrome (acrokeratosis paraneoplastica): persistence of cutaneous lesions after successful treatment of an associated oropharyngeal neoplasm.</p> <p>Question: Name synonym of Acrokeratosis paraneoplastica.</p> <p>Ideal answer: Acrokeratosis paraneoplastic (Bazex syndrome) is a rare, but distinctive paraneoplastic dermatosis characterized by erythematous squamous lesions located at the acral sites and is most commonly associated with carcinomas of the upper aerodigestive tract.</p> <p>Exact answer: Bazex syndrome</p> <p>###</p> <p>Context: {context}</p> <p>Question: {body}</p> <p>Ideal answer:</p>
--

Table 14

Prompt Template for Summary Question

<p>Context: Hirschsprung disease (HSCR) is a multifactorial, non-mendelian disorder in which rare high-penetrance coding sequence mutations in the receptor tyrosine kinase RET contribute to risk in combination with mutations at other genes.</p> <p>Question: Is Hirschsprung disease a mendelian or a multifactorial disorder?</p> <p>Answer: Coding sequence mutations in RET, GDNF, EDNRB, EDN3, and SOX10 are involved in the development of Hirschsprung disease. The majority of these genes was shown to be related to Mendelian syndromic forms of Hirschsprung's disease, whereas the non-Mendelian inheritance of sporadic non-syndromic Hirschsprung disease proved to be complex; involvement of multiple loci was demonstrated in a multiplicative model.</p> <p>###</p> <p>Context: Significant decreases and progressively higher reductions of systolic and diastolic blood pressure were observed after RSD. The RSD presents itself as an effective and safe approach to resistant hypertension.</p> <p>Question: What are the outcomes of Renal sympathetic denervation?</p> <p>Answer: Renal sympathetic denervation reduces heart rate, systolic and diastolic blood pressure.</p> <p>###</p> <p>Context: {context}</p> <p>Question: {body}</p> <p>Answer</p>

B. Example LLM Prompt and Response

Table 15

Yes/No Question - Example Prompt and Response

Prompt

(examples)

Context: No significant differences were observed, but FOLFOXIRI-bevacizumab achieved numerically superior survival outcomes versus FOLFOX-panitumumab.

These observations suggest that although doublet chemotherapy plus anti-EGFRs remains the preferred treatment in patients with left-sided RAS/BRAF wild-type mCRC, FOLFOXIRI-bevacizumab is a valuable option able to provide similar, if not better, outcomes at the price of a moderate increase in toxicity and may be adopted based on patients' preference and potential impact on quality of life

Question: Are FOLFIRINOV plus Bevacizuma the preferred therapy for left-sided RAS wild type advanced colorectal cancer with aggressive clinical presentation?

Ideal answer:

Response

Yes, based on the observed survival outcomes, doublet chemotherapy plus anti-EGFRs remains the preferred treatment in patients with left-sided RAS/BRAF wild-type mCRC, but FOLFIRINOV plus Bevacizumab can be considered based on patients' preference and potential impact on quality of life.

Exact answer: Yes

Table 16

List Question - Prompt and Sample response

Prompt

(examples)

Context: Recent studies have indicated associations between aberrant behavior of amyloid-beta and tau proteins and various neurological diseases, such as Alzheimer's disease, Parkinson's disease, and amyotrophic lateral sclerosis, as well as retinal neurodegenerative diseases like Glaucoma and age-related macular degeneration. Additionally, these proteins have been linked to cardiovascular disease, cancer, traumatic brain injury, and diabetes.

Question: Amyloid- β is associated with what diseases?

Ideal answer:

Response

Amyloid- β is associated with Alzheimer's disease, Parkinson's disease, amyotrophic lateral sclerosis, Glaucoma, age-related macular degeneration, cardiovascular disease, cancer, traumatic brain injury, and diabetes.

Exact answer: Alzheimer's disease; Parkinson's disease; amyotrophic lateral sclerosis; Glaucoma; age-related macular degeneration; cardiovascular disease; cancer; traumatic brain injury; diabetes