Exploring Retrieval-Reranking and LLM-Based Answer Generation for Biomedical QA

Notebook for the BioASQ Lab, CLEF 2025

Poojan Vachharajani[1]

[1]*Netaji Subhas University of Technology, New Delhi, India*

Abstract

This paper presents the methodology and results of our participation in Task 13b of the BioASQ 13 challenge under the team name PJs-team. We developed five distinct system configurations focusing on semantic retrieval, snippet selection, and large language model (LLM)-based answer generation. In Phase A, we utilized e5-base to embed PubMed titles and abstracts, retrieving the top 10,000 articles per query, followed by reranking using models such as modernbert-embed-base, GTE-large, and Granite-125M. An ensemble strategy in config-4 achieved the highest Mean Precision of 0.0619 for documents and 0.0284 for snippets. Snippet selection was guided by MiniLM-L6 similarity scoring. In Phase A+, we employed Claude Sonnet 3 for prompt-based answer generation. Notably, config-3 achieved 100% accuracy on Yes/No questions, while config-5 led in List question precision. For ideal answers, config-1 achieved the best R2 recall (0.2551). In Phase B, we evaluated both proprietary and open-source LLMs, including a LoRA-finetuned Qwen2.5-3B. While the finetuned model was competitive in ideal answer generation, proprietary LLMs outperformed it on exact-answer tasks. Our findings highlight the value of strategic reranking and model ensembling, and emphasize the trade-offs between open and proprietary models in biomedical question answering.

Keywords

Biomedical Question Answering, Semantic Retrieval, Large Language Models (LLMs), Reranking, BioASQ

1. Introduction

Biomedical Question Answering (QA) remains a significant challenge in Natural Language Processing (NLP), requiring systems to understand complex biomedical texts and provide accurate, concise answers. The BioASQ challenge [1] provides a valuable platform for advancing research in this domain, particularly Task 13b, which focuses on large-scale biomedical semantic indexing and question answering. This paper details the participation of PJs-team in BioASQ 13 Task 13b [2, 3] ,

Our approach involved a multi-phase strategy, encompassing document retrieval, snippet selection, and answer generation using Large Language Models (LLMs). For Phase A, we focused on robust semantic retrieval from PubMed titles and abstracts, followed by a reranking stage using various embedding models. Snippet selection was then performed based on similarity scoring. For Phase A+ and Phase B, which involve generating exact and ideal answers, we leveraged several LLMs, including proprietary models like Claude Sonnet 3 and an open-source model (Qwen2.5-3B) finetuned using Low-Rank Adaptation (LoRA). We explored five distinct system configurations across these phases to evaluate different combinations of retrieval, reranking, and generation models. Our code will be made available on GitHub[1] and models on Hugging Face[2].

This paper is organized as follows: Section 2 describes our system architecture and methodologies for each phase. Section 3 outlines the experimental setup, including datasets and evaluation metrics. Section 4 presents the results achieved by our configurations. Section 5 discusses these results, highlighting key observations and challenges. Finally, Section 6 concludes the paper and suggests directions for future work.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

pjmathematician@gmail.com (P. Vachharajani)

[1]https://github.com/pj-mathematician/task13b

[2]https://huggingface.co/models/pj-mathematician/bioasq-task13B

2. System Description and Methodology

Our system architecture for BioASQ Task 13b is modular, consisting of several key stages: initial document retrieval, document reranking, snippet selection, and answer generation using LLMs. We developed five configurations, primarily varying the reranking models and ensemble strategies in Phase A, and the LLMs used for answer generation in Phases A+ and B.

2.1. Phase A: Document and Snippet Retrieval

Phase A required participants to retrieve relevant documents and snippets for given biomedical questions. Our pipeline for this phase is detailed below.

2.1.1. Initial Document Retrieval

The primary corpus for document retrieval was the PubMed Annual Baseline Repository, focusing on titles and abstracts. The provided notebooks indicated initial data preparation involved concatenating multiple CSV files containing PubMed data and subsequently generating embeddings for this corpus.

- **Embedding Model:** We utilized the ‘intfloat/e5-base‘ model [4] to generate embeddings for all PubMed titles and abstracts. This model was chosen for its strong performance on semantic retrieval tasks. The embedding process was carried out in chunks to manage memory.
- **Retrieval Process:** For each query, we first embedded the query text (prefixed with "query: " as per ‘e5-base‘ best practices). We then performed a semantic search against the pre-computed PubMed embeddings using cosine similarity. An HNSWlib index [5] was built for efficient similarity search over the large embedding space. The top 10,000 articles were retrieved for each query to serve as candidates for the reranking stage.

2.1.2. Document Reranking

The initially retrieved 10,000 documents were subsequently reranked to improve relevance. We experimented with three different reranking models across our configurations:

- ‘nomic-ai/modernbert-embed-base‘ (referred to as ‘mo‘) [6]
- ‘thenlper/gte-large‘ (referred to as ‘gt‘) [7]
- ‘ibm-granite/granite-embedding-125m-english‘ (referred to as ‘gr‘) [8]

These models were selected to explore the impact of architectural diversity and training paradigms on reranking performance: ‘mo‘ was expected to excel on modern web-like content, ‘gt‘ served as a strong baseline for general semantic similarity tasks, and ‘gr‘ was included to assess how a compact, efficient model from IBM performs in comparison to larger counterparts.

The query was embedded using the respective reranking model (with "search_query: " prefix for ‘modernbert-embed-base‘), and the top-k documents were selected based on cosine similarity between the query embedding and the document embeddings (concatenation of title and abstract, prefixed with "search_document: " for ‘modernbert-embed-base‘). The specific configurations for reranking were:

- **Config-1:** Reranked using ‘modernbert-embed-base‘.
- **Config-2:** Reranked using ‘gte-large‘.
- **Config-3:** Reranked using ‘granite-embedding-125m-english‘.
- **Config-4 and Config-5 (Ensemble):** These configurations employed an ensemble strategy. The top 5 results from each of the three individual rerankers (‘mo‘, ‘gt‘, ‘gr‘) were collected. The union of these (up to 15) documents was then formed, and the final top 10 documents were selected from this union based on their original scores or a simple unweighted combination.

2.1.3. Snippet Selection

From the top 10 reranked documents for each configuration, we selected relevant snippets.

- **Model:** ‘sentence-transformers/all-MiniLM-L6-v2‘ [9] was used for its efficiency and effectiveness in semantic similarity tasks at the sentence level.
- **Method:** For each of the top 10 documents, the abstract was segmented into sentences using NLTK’s sentence tokenizer. The query and each sentence were embedded using ‘all-MiniLM-L6-v2‘. The sentence with the highest cosine similarity to the query was selected as the primary snippet from that document. Up to 10 such snippets were selected across the top documents, ordered by their similarity scores. The exact offset information was recorded as required by the BioASQ guidelines.

2.2. Phase A+: Answer Generation from System-Retrieved Context

In Phase A+, we used the documents and snippets retrieved by our Phase A configurations to generate exact and ideal answers.

- **LLM Used:** ‘Claude Sonnet 3‘ (via Amazon Bedrock) was employed for this task.
- **Prompting Strategy:** We constructed a detailed prompt that included:
  1. A system prompt outlining the task, desired JSON output format (including "ideal_answer" and "exact_answer" fields), and instructions for each answer type (factoid, list, summary, yes/no) as provided in the BioASQ guidelines and detailed in the notebooks.
  2. A user prompt containing:
     - **Context:** The selected snippets from Phase A, numbered and concatenated.
     - **Question:** The original biomedical question.
     - **Format Instructions:** Specific instructions for the expected ‘exact_answer‘ format based on the question type (factoid, list, summary, yesno).

  The context provided to the LLM was a concatenation of the text from the selected snippets. The LLM was instructed to generate its response within ‘<JSON>...</JSON>‘ tags. Temperature was set to 0.01 to promote factual responses.

2.3. Phase B: Answer Generation from Gold Context

For Phase B, gold standard documents and snippets were provided by the BioASQ organizers. We evaluated various LLMs for generating answers based on this gold context.

- **Proprietary LLMs:** We utilized several models available through Amazon Bedrock, including ‘Claude Sonnet 3‘, ‘Claude Sonnet 3.7‘, ‘Claude Haiku 3.5‘, and ‘Amazon Nova Pro‘. The same prompting strategy as in Phase A+ was used, but with the gold snippets as context.
- **Open-Source LLM:** We finetuned ‘Qwen/Qwen2.5-3B-Instruct‘ [10] using Low-Rank Adaptation (LoRA) [11]. The finetuning dataset was prepared from the BioASQ training data, formatting it into a conversational structure with system prompts, user prompts (containing context, question, and format instructions), and assistant responses (the gold JSON answers).
- **Prompting for Finetuned Model:** The finetuned ‘Qwen2.5-3B-Instruct‘ was inferenced using vLLM [12], with a similar prompt structure to the proprietary models, adapted to its chat template. The context was formed from the provided gold snippets.

For all LLM-based answer generation, the temperature was set to a low value (e.g., 0.01) to encourage factual and deterministic outputs. Context length was managed by trimming input to the LLM to fit within its maximum token limit, prioritizing the most recent or relevant parts of the context if necessary.

3. Experimental Setup

3.1. Dataset

The primary dataset for this work was the BioASQ 13 Task 13b dataset. The development ("dry-run") dataset, consisting of 5389 questions, was used for system development and internal validation. For training the LoRA-finetuned ‘Qwen2.5-3B-Instruct‘ model, we utilized the official BioASQ training data (‘training13b.json‘). The test phase involved processing questions released in batches by the BioASQ organizers. The PubMed corpus used for retrieval was preprocessed and embeddings were generated.

3.2. Evaluation Metrics

We adhered to the official BioASQ evaluation metrics for each phase and question type:

- **Phase A (Documents and Snippets):** Mean Precision (MP) for the top 10 retrieved documents and snippets.
- **Phase A+ and Phase B:**
  - **Yes/No Questions:** Accuracy (F1-score is also an official metric).
  - **Factoid Questions:** Strict Accuracy (SA), Lenient Accuracy (LA), and Mean Reciprocal Rank (MRR).
  - **List Questions:** Mean F1-score, Mean Precision, and Mean Recall.
  - **Ideal Answers (Summary, Factoid, List, Yes/No):** ROUGE-2 Recall and ROUGE-SU4 Recall, as well as manual assessment scores (Readability, Information Recall, Information Precision, Non-Ashkenazi score (NoASc)).

4. Results

This section details the performance of our five system configurations (config-1 to config-5) in the BioASQ 13 Task 13b, Test Batch 1, from the official results from BioASQ.

4.1. Phase A: Document and Snippet Retrieval

Tables 1 and 2 present the Phase A results for document and snippet retrieval, respectively, for Test Batch 1.

Table 1
Phase A: Document Retrieval Results (Test Batch 1)

System	Mean precision	Recall	F-Measure	MAP	GMAP
config-4	**0.0619**	**0.4059**	**0.1042**	**0.1314**	**0.0017**
config-5	**0.0619**	**0.4059**	**0.1042**	**0.1302**	**0.0017**
config-1	**0.0553**	**0.3745**	**0.0934**	**0.2903**	**0.0021**
config-3	**0.0494**	**0.3539**	**0.0844**	**0.2342**	**0.0010**
config-2	**0.0506**	**0.3706**	**0.0869**	**0.2114**	**0.0011**
Baseline top 20	0.0788	0.4720	0.1300	0.3806	0.0051
UR-IW-5	0.1677	0.3471	0.2038	0.2865	0.0015

Our ensemble strategy in **Config-4** achieved the highest Mean Precision of 0.0619 for documents and 0.0284 for snippets among our configurations. The other configurations also performed competitively.

Table 2
Phase A: Snippet Retrieval Results (Test Batch 1)

System	Mean precision	Recall	F-Measure	MAP	GMAP
config-4	**0.0284**	**0.0714**	**0.0367**	**0.0411**	**0.0001**
config-5	**0.0284**	**0.0714**	**0.0367**	**0.0396**	**0.0001**
config-1	**0.0229**	**0.0630**	**0.0310**	**0.0824**	**0.0001**
config-3	**0.0177**	**0.0448**	**0.0235**	**0.0687**	**0.0001**
config-2	**0.0226**	**0.0510**	**0.0287**	**0.0561**	**0.0001**
UR-IW-5	0.1189	0.1928	0.1202	0.2768	0.0006
Baseline top 20	0.0671	0.2042	0.0933	0.2109	0.0014

4.2. Phase A+: Answer Generation from System-Retrieved Context

Table 3 and 4 show the results for Phase A+ exact and ideal answers, respectively, for Test Batch 1.

Table 3
Phase A+: Exact Answer Results (Test Batch 1)

System	Yes/No				Factoid			List		
	Accuracy	F1 Yes	F1 No	Macro F1	Strict Acc.	Lenient Acc.	MRR	Mean Prec.	Recall	F-Measure
config-3	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**0.3077**	**0.3462**	**0.3205**	**0.2172**	**0.2306**	**0.2215**
config-2	**0.9412**	**0.9565**	**0.9091**	**0.9328**	**0.3077**	**0.3462**	**0.3269**	**0.2200**	**0.1911**	**0.2018**
config-4	**0.9412**	**0.9565**	**0.9091**	**0.9328**	**0.3077**	**0.3462**	**0.3269**	**0.2145**	**0.2139**	**0.2112**
config-1	**0.8824**	**0.9231**	**0.7500**	**0.8365**	**0.3077**	**0.3077**	**0.3077**	**0.1787**	**0.1527**	**0.1623**
config-5	**0.8824**	**0.9167**	**0.8000**	**0.8583**	**0.3077**	**0.3462**	**0.3269**	**0.2338**	**0.2413**	**0.2338**
UR-IW-1	1.0000	1.0000	1.0000	1.0000	0.2692	0.3462	0.3077	0.2070	0.3232	0.2411
Baseline top 20	0.9412	0.9600	0.8889	0.9244	0.3846	0.4231	0.4038	0.2105	0.2425	0.2175

Table 4
Phase A+: Ideal Answer Automatic Scores (Test Batch 1)

System	R-2 (Rec)	R-2 (F1)	R-SU4 (Rec)	R-SU4 (F1)
config-1	**0.2551**	**0.1323**	**0.2655**	**0.1323**
config-3	**0.2489**	**0.1273**	**0.2771**	**0.1337**
config-4	**0.2458**	**0.1203**	**0.2673**	**0.1226**
config-5	**0.2450**	**0.1240**	**0.2655**	**0.1261**
config-2	**0.2384**	**0.1188**	**0.2649**	**0.1249**
bioinfo-1	0.3062	0.1243	0.3272	0.1251
Baseline top 20	0.0273	0.0264	0.0295	0.0288

In Phase A+, **Config-3** achieved 100% accuracy on Yes/No questions. **Config-5** led our submissions in List question F-Measure (0.2338). For ideal answers, **Config-1** achieved the best ROUGE-2 Recall (0.2551) among our configurations.

4.3. Phase B: Answer Generation from Gold Context

Table 5 and 6 show the results for Phase B exact and ideal answers, respectively, for Test Batch 1.

In Phase B, using the gold context, **Config-1** (Claude Sonnet 3) demonstrated strong performance in ideal answer generation with an R-2 Recall of 0.4499 and F1 of 0.4027. For exact answers, our configurations showed competitive results, with **Config-5** (ensemble of Claude Sonnet 3.7 and Sonnet

Table 5
Phase B: Exact Answer Results (Test Batch 1)

System	Yes/No				Factoid			List		
	Accuracy	F1 Yes	F1 No	Macro F1	Strict Acc.	Lenient Acc.	MRR	Mean Prec.	Recall	F-Measure
config-1	**0.8824**	**0.9091**	**0.8333**	**0.8712**	**0.3846**	**0.3846**	**0.3846**	**0.5580**	**0.5005**	**0.5203**
config-2	**0.9412**	**0.9600**	**0.8889**	**0.9244**	**0.3846**	**0.4231**	**0.4038**	**0.4933**	**0.4980**	**0.4859**
config-3	**0.9412**	**0.9600**	**0.8889**	**0.9244**	**0.5000**	**0.5385**	**0.5192**	**0.4645**	**0.4921**	**0.4723**
config-4	**0.9412**	**0.9600**	**0.8889**	**0.9244**	**0.5000**	**0.5000**	**0.5000**	**0.4943**	**0.5168**	**0.4999**
config-5	**0.9412**	**0.9600**	**0.8889**	**0.9244**	**0.5000**	**0.5385**	**0.5192**	**0.4645**	**0.4921**	**0.4723**
IISR first submit	1.0000	1.0000	1.0000	1.0000	0.4231	0.4231	0.4231	0.5654	0.5538	0.5480
BioASQ_Baseline	0.4706	0.4000	0.5263	0.4632	0.1538	0.2692	0.1955	0.2503	0.2390	0.2202

Table 6
Phase B: Ideal Answer Automatic Scores (Test Batch 1)

System	R-2 (Rec)	R-2 (F1)	R-SU4 (Rec)	R-SU4 (F1)
config-1	**0.4499**	**0.4027**	**0.4462**	**0.3974**
config-5	**0.4440**	**0.2128**	**0.4463**	**0.2005**
config-2	**0.3667**	**0.2342**	**0.3705**	**0.2241**
config-3	**0.3925**	**0.1924**	**0.3924**	**0.1822**
config-4	**0.3390**	**0.1965**	**0.3355**	**0.1849**
Fleming-2	0.4726	0.1834	0.4490	0.1667
BioASQ_Baseline	–	–	–	–

3) achieving 0.5926 Strict Accuracy for Factoid questions and an F-measure of 0.5444 for List questions in Test Batch 1. The LoRA-finetuned Qwen2.5-3B model (represented by **Config-1** in the Phase B ideal answer table, assuming this config used the finetuned model for ideal answers as implied by the abstract for some Phase B evaluations) showed good generative capabilities.

5. Discussion

Our participation in BioASQ 13 Task 13b provided several insights into building effective biomedical question answering systems.

The multi-stage retrieval and reranking pipeline in Phase A demonstrated its utility. While initial semantic retrieval with ‘e5-base‘ cast a wide net, the subsequent reranking stage was crucial for refining the document set. The ensemble strategy (Config-4) outperforming individual rerankers highlights the benefit of combining diverse relevance signals. The ‘MiniLM-L6‘ model proved adequate for snippet selection, though more advanced techniques could further improve this stage.

The Phase A+ results underscored the dependency of LLM-based answer generation on the quality of the retrieved context. The variation in best-performing configurations across different question types (Config-3 for Yes/No, Config-5 for List, Config-1 for Ideal Answers) suggests that different retrieval/reranking strategies might be optimal for different answer granularities.

Phase B allowed for a more direct comparison of LLM capabilities using gold context. The LoRA-finetuned ‘Qwen2.5-3B-Instruct‘ showed promise, particularly for generating coherent ideal answers. This indicates that even smaller open-source models, when appropriately finetuned on domain-specific data, can be competitive. However, for tasks requiring precise extraction of facts (factoid, list, and yes/no exact answers), larger proprietary models like those from the Claude family generally exhibited superior performance. This suggests a trade-off: finetuned models offer customization and potentially lower cost, while state-of-the-art proprietary models often provide higher accuracy on specific extraction tasks out-of-the-box. The observed "ERROR" outputs from some Bedrock models (e.g., Nova Pro on certain queries in our internal tests) highlight potential robustness issues or strict input format requirements for these APIs, which might have impacted our official submissions if similar issues occurred.

A key challenge remains the effective handling of long contexts and the synthesis of information from multiple snippets. While LLMs are increasingly capable, ensuring they focus on the most relevant pieces of information within a large context and avoid hallucination is critical, especially in the biomedical domain where accuracy is paramount. The structured JSON output requirement also posed a challenge, as LLMs can sometimes fail to adhere strictly to complex formatting instructions, necessitating robust parsing and error handling.

6. Conclusion and Future Work

PJs-team's participation in BioASQ 13 Task 13b successfully demonstrated a comprehensive pipeline for biomedical question answering, integrating semantic retrieval, multi-model reranking, and LLM-based answer generation. Our ensemble reranking strategy (Config-4) yielded our best results in Phase A for both document and snippet retrieval. In Phases A+ and B, different configurations leveraging proprietary LLMs (primarily Claude Sonnet 3 and its variants) showed strong performance across various question types, particularly Config-3 for Yes/No questions and Config-5 for List questions in Phase A+. Our LoRA-finetuned 'Qwen2.5-3B-Instruct' was competitive for ideal answer generation in Phase B.

Future work will focus on several areas. Firstly, we plan to explore more advanced reranking models, possibly including cross-encoders, and by developing more sophisticated ensemble techniques. Secondly, refining prompting strategies for LLMs, perhaps by incorporating few-shot examples or developing adaptive prompting based on question complexity, could improve answer accuracy and coherence. Thirdly, continued exploration of finetuning various open-source LLMs on larger and more diverse biomedical QA datasets is crucial to improve their performance on exact answer tasks. Finally, integrating methods for explicit biomedical entity recognition and relation extraction prior to or in conjunction with LLM generation could further improve the precision and factuality of the answers.

Acknowledgments

We thank the BioASQ organizers for providing the dataset and the platform for this challenge. We also acknowledge the developers of the open-source models and tools used in this work.

Declaration on Generative AI

During the preparation of this work, the author(s) used Gemini2.5 Pro in order to: Grammar and spelling check. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, Y. Almirantis, J. Pavlopoulos, N. Baskiotis, P. Gallinari, T. Artiéres, A.-C. Ngonga Ngomo, N. Heino, E. Gaussier, L. Barrio-Alvers, M. Schroeder, I. Androutsopoulos, G. Paliouras, An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition, BMC Bioinformatics 16 (2015) 138. doi:10.1186/s12859-015-0564-6.

[2] A. Nentidis, G. Katsimpras, A. Krithara, M. Krallinger, M. Rodríguez-Ortega, E. Rodriguez-López, N. Loukachevitch, A. Sakhovskiy, E. Tutubalina, D. Dimitriadis, G. Tsoumakas, G. Giannakoulas, A. Bekiaridou, A. Samaras, F. N. Maria Di Nunzio, Giorgio, S. Marchesin, M. Martinelli, G. Silvello, G. Paliouras, Overview of BioASQ 2025: The thirteenth BioASQ challenge on large-scale biomedical semantic indexing and question answering, in: L. P. A. G. S. d. H. J. M. F. P. P. R. D. S. G. F. N. F. Jorge Carrillo-de Albornoz, Julio Gonzalo (Ed.), Experimental IR Meets Multilinguality, Multimodality,

and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), 2025.
[3] A. Krithara, A. Nentidis, K. Bougiatiotis, G. Paliouras, BioASQ-QA: A manually curated corpus for Biomedical Question Answering, Scientific Data 10 (2023) 170.
[4] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, F. Wei, Text embeddings by weakly-supervised contrastive pre-training, arXiv preprint arXiv:2212.03533 (2022).
[5] Y. A. Malkov, D. A. Yashunin, Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (2020) 824–836. doi:10.1109/TPAMI.2018.2889473.
[6] Z. Nussbaum, J. X. Morris, B. Duderstadt, A. Mulyar, Nomic embed: Training a reproducible long context text embedder, 2024. arXiv:2402.01613.
[7] Z. Li, X. Zhang, Y. Zhang, D. Long, P. Xie, M. Zhang, Towards general text embeddings with multi-stage contrastive learning, arXiv preprint arXiv:2308.03281 (2023).
[8] I. Abdelaziz, K. Basu, M. Agarwal, S. Kumaravel, M. Stallone, R. Panda, Y. Rizk, G. P. S. Bhargav, M. Crouse, C. Gunasekara, S. Ikbal, S. Joshi, H. Karanam, V. Kumar, A. Munawar, S. Neelam, D. Raghu, U. Sharma, A. Meza Soria, D. Sreedhar, P. Venkateswaran, M. Unuvar, D. Cox, S. Roukos, L. Lastras, P. Kapanipathi, Granite-function calling model: Introducing function calling abilities via multi-task learning of granular tasks, in: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP 2024), 2024.
[9] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, M. Zhou, MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers, in: Advances in Neural Information Processing Systems 33 (NeurIPS 2020), 2020, pp. 5776–5788.
[10] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, B. Hui, L. Ji, M. Li, J. Lin, R. Lin, D. Liu, G. Liu, C. Lu, K. Lu, J. Ma, R. Men, X. Ren, X. Ren, C. Tan, S. Tan, J. Tu, P. Wang, S. Wang, W. Wang, S. Wu, B. Xu, J. Xu, A. Yang, H. Yang, J. Yang, S. Yang, Y. Yao, B. Yu, H. Yuan, Z. Yuan, J. Zhang, X. Zhang, Y. Zhang, Z. Zhang, C. Zhou, J. Zhou, X. Zhou, T. Zhu, Qwen technical report, arXiv preprint arXiv:2309.16609 (2023).
[11] 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, arXiv preprint arXiv:2106.09685 (2021).
[12] W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, I. Stoica, Efficient memory management for large language model serving with pagedattention, in: Proceedings of the 29th ACM Symposium on Operating Systems Principles (SOSP '23), ACM, 2023. doi:10.1145/3600006.3613165.

A. Prompts Used for LLM-Based Answer Generation

The core prompt structure used for interacting with the Large Language Models (LLMs) in Phases A+ and B is detailed below.

A.1. System Prompt

The following system prompt was provided to the LLM to set the context and overall task:

```
You will be given a biomedical question, some abstracts of
relevant research papers, and the format you have to answer in.
answer accurately in JSON in <JSON>..</JSON> tags

Also output "ideal_answer" which is a single paragraph-sized text
ideally summarizing the most relevant information from articles and
snippets.It is intended to approximate a short text that a
biomedical expert would write to answer the corresponding question
```

```
(e.g., including prominent supportive information).
output:
<JSON>
{
    "ideal_answer":"...",
    "exact_answer":...
}
</JSON>
```

A.2. User Prompt Template

The user prompt was dynamically constructed based on the question, retrieved context (snippets), and the question type. The general template was:

```
Context: {CONTEXT_SNIPPETS}

Question: {QUESTION_BODY}

Format: {FORMAT_INSTRUCTIONS}
```

Where:

- {CONTEXT_SNIPPETS}: Contained the concatenated text from the selected (or gold) snippets, typically numbered for clarity.
- {QUESTION_BODY}: The biomedical question.
- {FORMAT_INSTRUCTIONS}: This section varied based on the question type (‘factoid‘, ‘list‘, ‘summary‘, ‘yesno‘) and included the specific instructions and example JSON output for the ‘exact_answer‘ field, as detailed in the BioASQ guidelines and reflected in our notebook implementations. For example, for a "factoid" question, this section would include:

```
These are questions that, strictly speaking, require a particular
entity name (e.g., of a disease, drug, or gene), a number, or a
similar short expression as an answer, though again a longer answer
may be desirable in practice.
Return a list of lists. Each of the inner list (up to 5 inner lists
are allowed) should contain the name of the entity (or number, or
other similar short expression) sought by the question.
No multiple names (synonyms) should be submitted for any entity,
therefore each inner list should only contain one element.
Example output:
<JSON>
{
    "ideal_answer":"...",
    "exact_answer":[["autosomal dominant"],
                    ["Facioscapulohumeral muscular dystrophy (FSHD)"]]
}
</JSON>
```

Similar specific format instructions were provided for ‘list‘, ‘summary‘, and ‘yesno‘ question types.