Are Smaller Open-Weight LLMs Closing the Gap to Proprietary Models for Biomedical Question Answering?

Notebook for the BioASQ Lab at CLEF 2025

Damian **Stachura**[1], Joanna **Konieczna**[1] and Artur **Nowak**[1]

[1]*Evidence Prime, Krakow, Poland*

Abstract

Open-weight versions of large language models (LLMs) are rapidly advancing, with state-of-the-art models like DeepSeek-V3 now performing comparably to proprietary LLMs. This progression raises the question of whether small open-weight LLMs are capable of effectively replacing larger closed-source models. We are particularly interested in the context of biomedical question-answering, a domain we explored by participating in Task 13B Phase B of the BioASQ challenge. In this work, we compare several open-weight models against top-performing systems such as GPT-4o, GPT-4.1, Claude 3.5 Sonnet, and Claude 3.7 Sonnet. To enhance question answering capabilities, we use various techniques including retrieving the most relevant snippets based on embedding distance, in-context learning, and structured outputs. For certain submissions, we utilize ensemble approaches to leverage the diverse outputs generated by different models for exact-answer questions. Our results demonstrate that open-weight LLMs are comparable to proprietary ones. In some instances, open-weight LLMs even surpassed their closed counterparts, particularly when ensembling strategies were applied. All code is publicly available at https://github.com/evidenceprime/BioASQ-13b.

Keywords

Biomedical Question Answering, Large Language Models, Zero-Shot Prompting, Few-Shot Prompting, In-Context Learning, GPT-4, Claude, Open-Weight LLM, Ensembling

1. Introduction

In question answering tasks, access to domain-specific knowledge can significantly enhance response quality, particularly when answers need to be grounded in provided supplementary materials.

The BioASQ Challenge [1] exemplifies such a task in biomedical domain. In this challenge, participating systems are provided with relevant biomedical papers from the PubMed database. These materials can then be leveraged to generate high-quality responses to the posed questions. The questions themselves span four distinct types: yes/no, factoid, list, and summary questions.

Over the years, various approaches have been employed for question answering tasks in the BioASQ Challenge. In its earliest editions, classic methods were applied, such as BM25, which ranked retrieved documents based on their relevance to a question. Additionally, researchers applied similarity algorithms using vector embeddings, algorithms based on linguistic annotations of texts, and early deep learning models. For an extended period, BERT-based solutions, notably BioBERT [2] and PubMedBERT [3], dominated these question answering tasks. Sequence-to-sequence models like T5 [4] also proved to be useful.

However, since the global emergence of ChatGPT [5, 6] in 2022, large language models have significantly reshaped the competitive landscape of question answering. Proprietary models, including OpenAI's offerings, Gemini [7], and Claude [8], initially dominated this field, fostering the belief that creating robust LLMs requires a massive financial investment.

A significant shift began in 2024 and 2025, with a growing number of organizations publicly releasing models featuring open weights and permissive licenses. Today, large open-weight models like Llama 3-405B [9], DeepSeek-V3 [10], and Qwen3-235B-A22B [11] are proving capable of challenging even the

best proprietary models, as evidenced by platforms like the LM Arena [12]. Perhaps even more impactful are the smaller open-weight models, which can run on consumer-grade machines and demonstrate impressive competitiveness in tasks requiring access to domain-specific knowledge, especially within a retrieval-augmented generation (RAG) setup.

This year marks the 13th edition of the BioASQ Challenge, and we participated in Task 13B, Phase B [13]. Our primary objective was to investigate whether relatively small LLMs, primarily those up to 14 billion parameters, could effectively compete with more powerful proprietary models in this biomedical question-answering context. To achieve this, we explored multiple strategies for enhancing our results. We utilized in-context learning by leveraging the provided database of questions from previous BioASQ challenge editions. Additionally, we used similarity algorithms using vector embeddings to select a pertinent subset of snippets from the provided PubMed articles for each question. This paper details the results from four question batches of Task 13B Phase B and discusses our conclusions.

2. Methodology

We experimented with numerous techniques to optimize performance in biomedical question answering. The successful approaches implemented in our solutions are outlined below.

2.1. Best Snippets Selection

For our submissions, we selected the 10 best-matching snippets from the provided PubMedQA articles. Our team experimented with varying snippet counts, validating them against datasets from previous BioASQ challenge versions. This process led us to select 10 snippets as the optimal number, as it consistently produced the most robust results. We utilized the sentence-transformers library [14] with the `nomic-embed-text-v1` [15] model. Our approach involved computing embeddings for all snippets and the question. Subsequently, we calculated the cosine similarity between each (snippet, question) pair to identify options with the highest similarity. Finally, these selected snippets were provided to the model, ordered from most to least similar.

2.2. In-Context Learning

Research has demonstrated that in-context learning enhances the performance of language models in diverse applications [5, 16]. We investigated how different models performed with in-context learning, sourcing examples from previous BioASQ challenge editions. We used Qdrant [17], a vector database, into which we inserted computed embeddings for all previous questions combined with their 10 best snippets. Subsequently, for each new question, we queried the database for the most similar elements, following the approach presented in [18]. Experimentally, we determined that 3 examples were optimal for factoid and list questions, while a zero-shot approach was used for yes/no and summary type questions.

2.3. Prompts

For all question types, we utilized hand-crafted prompts. As noted previously, a zero-shot prompting approach was employed for yes/no and summary questions, given empirical observations that few-shot prompting detrimentally affected performance for these specific categories. Conversely, for factoid and list questions, few-shot prompting demonstrated clear benefits. Accordingly, we implemented a 3-shot prompting strategy for these questions, based on insights gained through comprehensive experimentation. The system prompts are detailed in Table 1, and the actual prompts guiding the models to generate answers for all question types are presented in Table 2. We also briefly experimented with DSPy [19] for the automated generation of prompts based on predefined input and output schemas in batch 2. However, responses achieved using DSPy were slightly worse than those achieved with our

Table 1
System prompts for exact answers and ideal answers questions

Question type	System prompt
Exact answer	You are a biomedical AI expert specializing in question answering, research, and entity extraction.
Ideal answer	You are an expert in the medical texts summarization. Answer the given question with a single paragraph text and your answer should be based on the provided context snippets. You should generate your response in at most 2-3 sentences (30-50 words).

Table 2
Prompts for all types of questions in the Task 13B Phase B

| Question type	Prompt
Yes/No	Given only the following SNIPPETS and QUESTION, answer the QUESTION only with 'Yes' or 'No'.
Factoid	Extract key biomedical entities **strictly using the provided SNIPPETS** to answer the QUESTION. List **1 to 5** of the most relevant entities, ranked by confidence. **Never exceed 5 entities.** If more exist, return only the top 5. Prefer concise entities and **remove redundant or longer variants** of the same term. If no relevant entities exist, return 'None.'.
List	Extract key biomedical entities **strictly using the provided SNIPPETS** to answer the QUESTION. List **1 to 5** of the most relevant entities. Prefer concise entities and **remove redundant or longer variants** of the same term. If no relevant entities exist, return 'None'.
Summary	Answer the QUESTION by returning a single paragraph sized text (use max 50 words) ideally summarizing only the most relevant information in the SNIPPETS.

hand-crafted prompts. In the future, we plan to investigate whether automatic prompt optimization can help by creating model-specific prompts.

2.4. Structured Outputs

We opted to use structured outputs to facilitate the extraction of LLM results in a predefined format. We defined a JSON schema for response formatting, and subsequently followed a context-free grammar (CFG) approach for it. We used CFG implementations introduced by model providers or accessed via external libraries like Outlines [20] to guide the token sampling process. This methodology ensures that the generated tokens adhere strictly to the schema, eliminating the need for complex regex-based extraction from the model response.

2.5. Models

In our study, we utilized various LLMs, drawing from both open-weight and closed options. Our primary focus was on relatively smaller open-weight models, specifically those with up to 14 billion parameters, such as Phi-4 [21], Gemma-3-12B [22], Qwen2.5 14B [23], and Meditron Phi-4 14B [24]. For a third batch of experiments, we expanded our testing to include quantized versions of Gemma3-27B [22] and Mistral3-24B [25]. Although we briefly attempted to use HuatuoGPT-o1 [26], our limited exploration of reasoning models meant we did not achieve strong results with it. We also incorporated several of
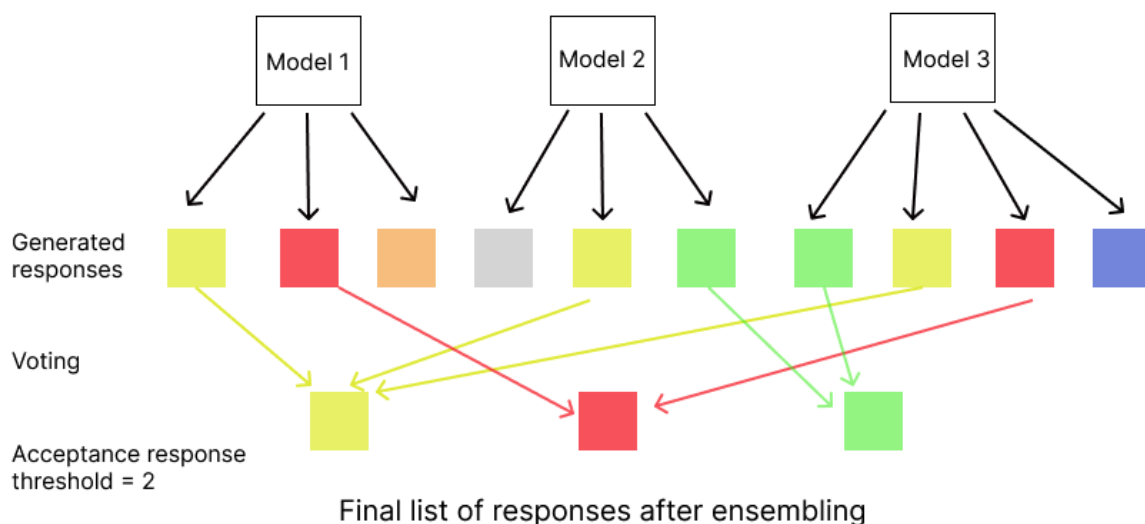
Figure 1: Ensembling strategy for factoid and list type of questions

the newest closed-source models, including recent generations of GPT (GPT-4o [27], GPT-4.1 [28]) and Claude (Claude Sonnet 3.5 [29], Claude Sonnet 3.7 [30]).

2.5.1. Quantized Models

For batches 1-3, our experiments used open-weight models quantized to 4-bit. However, in the final batch, we proposed solutions based on the full, unquantized versions of these models. Interestingly, in both setups, the open-weight models proved to be competitive with the closed alternatives.

2.6. Ensembling Method

Ensembling methods are widely recognized as beneficial when combining responses from multiple weaker models to achieve a single, stronger result. Various techniques exist for this purpose, including majority voting, confidence scoring, and aggregation.

For yes/no questions, we applied a straightforward majority voting approach, where the final answer was determined by the option chosen by the most models. For factoid and list questions, we developed a more sophisticated aggregation method. This involved collecting responses from all models and calculating the frequency of each distinct response. The most frequent response was then selected, provided its number of appearances exceeded a predefined threshold. The process is visualized on Figure 1. For factoid questions, we limited the output list to a maximum of five best responses, as specified in the rules for this question type.

For later batches, we incorporated different classes of LLMs into our ensembling strategy. This approach leveraged the observed benefits of integrating the diverse characteristics of various LLM families such as Phi, Qwen, Mistral, and Gemma. As emphasized by Jiang et al. [31], different LLMs, trained on varied data and architectures, inherently exhibit unique strengths that can be synergistic in an ensemble.

3. Results

We participated solely in Task 13B, Phase B, of the BioASQ challenge [1][13]. This task encompasses four types of questions: yes/no, factoid, and list questions, which are evaluated by matching exact answers provided by the challenge organizers. In addition, there are summary questions that require an ideal and free-form summary as a response, scored through automatic metrics and manual reviews. Our

approach involved testing multiple techniques and models across published question batches, leading to distinct strategies for each.

3.1. System Definitions

For each submission, the rules detailed in 2 were applied. The models used to generate responses in each submission, grouped by system name from 1 to 5, are presented below. The results for these systems in all batches are summarized in Table 3, Table 4, Table 5, and Table 6. A detailed specification of each submitted system follows:

Batch 1:

- EP-1: Phi-4
- EP-2: HuatuoGPT-o1 8B
- EP-3: Qwen2.5-14B
- EP-4: GPT-4o
- EP-5: Claude 3.5 Sonnet

Batch 2:

- EP-1: Ensemble - Gemma-3-12B, Qwen2.5-14B, Phi-4, GPT-4o, Claude 3.5 Sonnet
- EP-2: Claude 3.5 Sonnet
- EP-3: Phi-4
- EP-4: Phi-4 + DSPy prompt (only for factoid questions, without prompt optimization)
- EP-5: Qwen2.5-14B + DSPy prompt (only for factoid questions, without prompt optimization)

Batch 3:

- EP-1: Ensemble - Mistral-Small-3.1-24B, Gemma-3-12b, Gemma-3-27b, Qwen2.5-14B, Phi-4
- EP-2: Ensemble - GPT-4o, GPT-4.1, Claude 3.5 Sonnet
- EP-3: GPT-4.1
- EP-4: Phi-4

Batch 4:

- EP-1: Ensemble - GPT-4.1, GPT-4o, Claude 3.5 Sonnet, Claude 3.7 Sonnet
- EP-2: Ensemble - Gemma-3-12B, Qwen2.5-14B, Meditron3-Phi4-14B, Phi-4
- EP-3: Ensemble - Qwen2.5-14B, Meditron3-Phi4-14B, Phi-4, GPT-4.1, GPT-4o, Claude 3.5 Sonnet, Claude 3.7 Sonnet
- EP-4: Ensemble - Qwen2.5-14B, Phi-4, GPT-4.1, GPT-4o, Claude 3.7 Sonnet
- EP-5: GPT-4.1

3.2. Exact Answers

We used distinct answering strategies for each batch, as detailed in 3.1. For batch 1, our primary focus was on assessing the performance of individual models within each system. In batches 2 and 3, we also incorporated ensembling techniques, specifically involving combinations of open-weight and selected closed models. It is important to note that for these batches, we only used quantized versions of open models. Finally, for the last batch, we conducted a comparative analysis between full open-weight models and proprietary models.

The most advantageous approach for yes/no questions was difficult to determine. Proprietary models achieved the best results in batches 1 and 4, while open-weight models dominated in the remaining. It is shown in Table 3. The quality of the provided context appears to be a critical factor, with both model types demonstrating sufficient capability to extract key information pertinent to the question.

Table 3

Task 13B, Phase B, Yes/No questions, evaluated by exact answer matches

Batch Nr	Position	System	Accuracy	F1 Yes	F1 No	Macro F1
	1 of 72	EP-4	1.0	1.0	1.0	**1.0**
	23 of 72	EP-1	0.9412	0.9565	0.9091	**0.9328**
Batch 1		Best result	1.0	1.0	1.0	**1.0**
	1 of 72	EP-4	1.0	1.0	1.0	**1.0**
	22 of 72	EP-5	0.9412	0.9524	0.9231	**0.9377**
Batch 2		Best result	1.0	1.0	1.0	**1.0**
	1 of 66	EP-1	0.9545	0.9697	0.9091	**0.9394**
	30 of 66	EP-2	0.9091	0.9412	0.8000	**0.8706**
Batch 3		Best result	0.9545	0.9697	0.9091	**0.9394**
	1 of 79	EP-1	1.0	1.0	1.0	**1.0**
	1 of 79	EP-3	1.0	1.0	1.0	**1.0**
Batch 4	3 of 79	EP-4	0.9615	0.9730	0.9333	**0.9532**
	3 of 79	EP-5	0.9615	0.9730	0.9333	**0.9532**
		Best result	1.0	1.0	1.0	**1.0**

List-based questions demonstrably pose a greater challenge for LLMs. Despite this, open-weight models performed competitively to proprietary ones. Furthermore, we found that ensembling more diverse models leads to improved scores.

In more detail, ensembling a mixture of open and closed models proved to be beneficial for factoid questions. In batch 2, single proprietary models were outperformed by such ensembling mixture. This solution also represented the best approach in batch 4, surpassing individual closed models and ensembles composed solely of open-weight or closed models. In batch 3, multiple solution types exhibited competitive performance. Table 4 provides a summary of the results for factoid questions across all batches.

These insights are further corroborated by the findings from list questions, presented in Table 5. Ensembling solely open-weight models or a mixture of both model types consistently yielded the best approaches in batches 2, 3, and 4. For batch 1, the results between both model types were notably similar. These observations strongly suggest that ensembles of open-weight models can address more challenging tasks at a comparable, or even superior, level to closed models.

3.3. Ideal Answers

For the summary questions, we directly generated each summary using a chosen LLM and the prompt detailed in Table 2, without employing ensembling techniques. For systems involving multiple LLMs, we generated candidate summaries from all participating models for each question. The best summary was then selected using a cross-encoder reranking approach. This method involved calculating the similarity score between each generated summary and its corresponding question. The summary with the highest score was subsequently selected. For this purpose, we used the BiomedBERT model [3] to compute these similarity measures.

The recall scores for ROUGE metrics [32] achieved by our method for batches 2 and 4 were comparable to those of the top-performing solutions. However, F1 scores for ROUGE metrics in these batches were significantly lower. The results for summary questions can be seen in Table 6.

As shown in Table 6, Phi-4 exhibited the strongest performance among the evaluated LLMs, suggesting that open-weight models can also be competitive for summary-based questions. However, a full analysis of the responses to these questions requires manual scores that have not yet been published.

Table 4
Task 13B, Phase B, Factoid questions, evaluated by exact answer matches

Batch Nr	Position	System	Strict Acc.	Lenient Acc.	MRR
Batch 1	18 of 67	EP-1	0.4231	0.5385	**0.4808**
	18 of 67	EP-2	0.4231	0.5385	**0.4808**
	25 of 67	EP-3	0.4231	0.5000	**0.4615**
		Best result	0.5385	0.6538	**0.5962**
Batch 2	10 of 66	EP-1	0.5185	0.7407	**0.6031**
	29 of 66	EP-2	0.5185	0.5556	**0.5370**
		Best result	0.7037	0.7778	**0.7037**
Batch 3	1 of 59	EP-3	0.4000	0.6500	**0.5100**
	5 of 59	EP-1	0.4000	0.5500	**0.4625**
	7 of 59	EP-2	0.3500	0.6000	**0.4542**
	14 of 59	EP-4	0.4000	0.4500	**0.4250**
		Best result	0.4500	0.6500	**0.5100**
Batch 4	7 of 73	EP-3	0.5455	0.5909	**0.5682**
	18 of 73	EP-4	0.5000	0.5909	**0.5455**
	18 of 73	EP-1	0.5455	0.5455	**0.5455**
		Best result	0.6364	0.6818	**0.6364**

Table 5
Task 13B, Phase B, List questions, evaluated by exact answer matches

Batch Nr	Position	System	Precision	Recall	F-Measure
Batch 1	24 of 66	EP-5	0.5583	0.5019	**0.5174**
	24 of 66	EP-3	0.5583	0.5019	**0.5174**
		Best result	0.6226	0.6483	**0.5327**
Batch 2	2 of 66	EP-1	0.5685	0.7083	**0.6027**
	6 of 66	EP-2	0.5877	0.5889	**0.5721**
	10 of 66	EP-5	0.6044	0.5415	**0.5538**
		Best result	0.6360	0.7132	**0.6152**
Batch 3	2 of 59	EP-1	0.6659	0.6530	**0.6331**
	6 of 59	EP-3	0.5969	0.6787	**0.6148**
	10 of 59	EP-2	0.6364	0.6075	**0.6075**
	15 of 59	EP-4	0.5956	0.6141	**0.5964**
		Best result	0.6659	0.6787	**0.6337**
Batch 4	12 of 73	EP-2	0.5709	0.6268	**0.5896**
	13 of 73	EP-4	0.5650	0.6284	**0.5859**
	17 of 73	EP-3	0.5355	0.6589	**0.5768**
		Best result	0.7491	0.6791	**0.6492**

4. Conclusion

Our primary goal was to evaluate the competitive performance of open-weight LLMs against state-of-the-art proprietary LLMs for biomedical question answering. To this end, we rigorously tested numerous configurations, including both smaller open-weight models and various closed models. Our results consistently demonstrated the competitiveness of ensembles of open models for BioASQ 13B Phase B questions.

For yes/no questions, this thesis is supported by results from batches 2 and 3 (Table 3), where open-weight models outperformed closed-weight models. This trend also holds for list-based questions, with batches 2 through 4 demonstrating strong performance by open models on both factoid and list-type questions (Tables 4 and 5, respectively). For summary questions, the open-weight model Phi-4 exhibited promising performance in terms of ROUGE metrics in Batches 1 through 3, as detailed in Table 6.

This conclusion holds significant implications. The ability to use open-weight models negates the need for proprietary solutions in every application. This is particularly relevant for applications involving

Table 6
Task 13B, Phase B, Summary questions requiring free-form ideal answers

Batch Nr	System	R-2 (Rec)	R-2 (F1)	R-SU4 (Rec)	R-SU4 (F1)
	EP-1	0.3594	0.2021	0.3585	0.1905
	EP-4	0.3488	0.2021	0.3409	0.2309
Batch 1	Best result	0.4726	0.4122	0.4490	0.4008
	EP-4	0.4213	0.2494	0.4165	0.2326
	EP-3	0.3816	0.2990	0.3613	0.2725
Batch 2	Best result	0.4838	0.4417	0.4652	0.4287
	EP-4	0.3550	0.2077	0.3545	0.1946
	EP-3	0.3340	0.2308	0.3339	0.2175
Batch 3	Best result	0.4309	0.3520	0.4357	0.3439
	EP-4	0.2947	0.2151	0.2973	0.2074
	EP-3	0.2931	0.2158	0.3030	0.2092
Batch 4	Best result	0.4139	0.3604	0.3963	0.3515

highly restricted data that require on-premise deployment, a common scenario with medical data. In such contexts, smaller self-deployable models offer a compelling and practical alternative.

Acknowledgments

Declaration on Generative AI

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

References

[1] 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, M. G. Di Nunzio, N. Ferro, 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: J. Carrillo-de Albornoz, J. Gonzalo, L. Plaza, A. García Seco de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), 2025.

[2] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2019) 1234–1240. URL: http://dx.doi.org/10.1093/bioinformatics/btz682. doi:10.1093/bioinformatics/btz682.

[3] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, 2020. arXiv:arXiv:2007.15779.

[4] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL: https://arxiv.org/abs/1910.10683. arXiv:1910.10683.

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh,

D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, 2020. URL: https://arxiv.org/abs/2005.14165. arXiv:2005.14165.

[6] OpenAI, Chatgpt (nov 2022 version), 2022. URL: https://chat.openai.com.

[7] G. Team, Gemini: A family of highly capable multimodal models, 2025. URL: https://arxiv.org/abs/2312.11805. arXiv:2312.11805.

[8] Anthropic, Introducing claude, 2023. URL: https://www.anthropic.com/index/introducing-claude, accessed: 2025-05-28.

[9] A. . M. Llama Team, The llama 3 herd of models, 2024. URL: https://arxiv.org/abs/2407.21783. arXiv:2407.21783.

[10] DeepSeek-AI, Deepseek-v3 technical report, 2025. URL: https://arxiv.org/abs/2412.19437. arXiv:2412.19437.

[11] Q. Team, Qwen3 technical report, 2025. URL: https://arxiv.org/abs/2505.09388. arXiv:2505.09388.

[12] W.-L. Chiang, L. Zheng, Y. Sheng, A. N. Angelopoulos, T. Li, D. Li, H. Zhang, B. Zhu, M. Jordan, J. E. Gonzalez, I. Stoica, Chatbot arena: An open platform for evaluating llms by human preference, 2024. URL: https://arxiv.org/abs/2403.04132. arXiv:2403.04132.

[13] A. Nentidis, G. Katsimpras, A. Krithara, G. Paliouras, Overview of BioASQ Tasks 13b and Synergy13 in CLEF2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), CLEF 2025 Working Notes, 2025.

[14] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: https://arxiv.org/abs/1908.10084.

[15] Z. Nussbaum, J. X. Morris, B. Duderstadt, A. Mulyar, Nomic embed: Training a reproducible long context text embedder, 2024. arXiv:2402.01613.

[16] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, L. Zettlemoyer, Rethinking the role of demonstrations: What makes in-context learning work?, in: Y. Goldberg, Z. Kozareva, Y. Zhang (Eds.), Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 11048–11064. URL: https://aclanthology.org/2022.emnlp-main.759/. doi:10.18653/v1/2022.emnlp-main.759.

[17] Q. Team, Qdrant documentation, 2021. URL: https://qdrant.tech/documentation/.

[18] O. Rubin, J. Herzig, J. Berant, Learning to retrieve prompts for in-context learning, ArXiv abs/2112.08633 (2021). URL: https://api.semanticscholar.org/CorpusID:245218561.

[19] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, M. Zaharia, C. Potts, Dspy: Compiling declarative language model calls into self-improving pipelines, 2024.

[20] B. T. Willard, R. Louf, Efficient guided generation for llms, arXiv preprint arXiv:2307.09702 (2023).

[21] M. Abdin, J. Aneja, H. Behl, S. Bubeck, R. Eldan, S. Gunasekar, M. Harrison, R. J. Hewett, M. Java-heripi, P. Kauffmann, J. R. Lee, Y. T. Lee, Y. Li, W. Liu, C. C. T. Mendes, A. Nguyen, E. Price, G. de Rosa, O. Saarikivi, A. Salim, S. Shah, X. Wang, R. Ward, Y. Wu, D. Yu, C. Zhang, Y. Zhang, Phi-4 technical report, 2024. URL: https://arxiv.org/abs/2412.08905. arXiv:2412.08905.

[22] G. Team, Gemma 3 (2025). URL: https://goo.gle/Gemma3Report.

[23] Q. Team, Qwen2.5 technical report, 2025. URL: https://arxiv.org/abs/2412.15115. arXiv:2412.15115.

[24] O. M. Team, Model card: Phi4 meditron-3[14b], 2025. URL: https://huggingface.co/OpenMeditron/Meditron3-Phi4-14B.

[25] M. Team, Mistral small 3.1, 2025. URL: https://mistral.ai/news/mistral-small-3-1.

[26] J. Chen, Z. Cai, K. Ji, X. Wang, W. Liu, R. Wang, J. Hou, B. Wang, Huatuogpt-o1, towards medical complex reasoning with llms, 2024. URL: https://arxiv.org/abs/2412.18925. arXiv:2412.18925.

[27] OpenAI, Gpt-4o system card, 2024. URL: https://arxiv.org/abs/2410.21276. arXiv:2410.21276.

[28] OpenAI, Introducing gpt-4.1 in the api, 2025. URL: https://openai.com/index/gpt-4-1/.

[29] Anthropic, Claude 3.5 sonnet, 2024. URL: https://www.anthropic.com/news/claude-3-5-sonnet.

[30] Anthropic, Claude 3.7 sonnet system card, 2025. URL: https://assets.anthropic.com/m/785e231869ea8b3b/original/claude-3-7-sonnet-system-card.pdf.

[31] D. Jiang, X. Ren, B. Y. Lin, Llm-blender: Ensembling large language models with pairwise ranking and generative fusion, 2023. URL: https://arxiv.org/abs/2306.02561. arXiv:2306.02561.

[32] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81. URL: https://aclanthology.org/W04-1013/.