Adversarial Databases Improve Success in Retrieval-based Large Language Models

Sean Wu
Department of Computer Science
Pepperdine University
sean.wu@pepperdine.edu

Michael Koo
Department of Computer Science
Pepperdine University
michael.koo@pepperdine.edu

Li Yo Kao
Division of Nephrology
University of California, Los Angeles
lkao@mednet.ucla.edu

Andy Black
Division of Nephrology
University of California, Los Angeles
aablack@mednet.ucla.edu

Lesley Blum
Division of Nephrology
University of California, Los Angeles
lblum@mednet.ucla.edu

Fabien Scalzo
Division of Nephrology
University of California, Los Angeles
fab@cs.ucla.edu

Ira Kurtz
Division of Nephrology
University of California, Los Angeles
ikurtz@mednet.ucla.edu

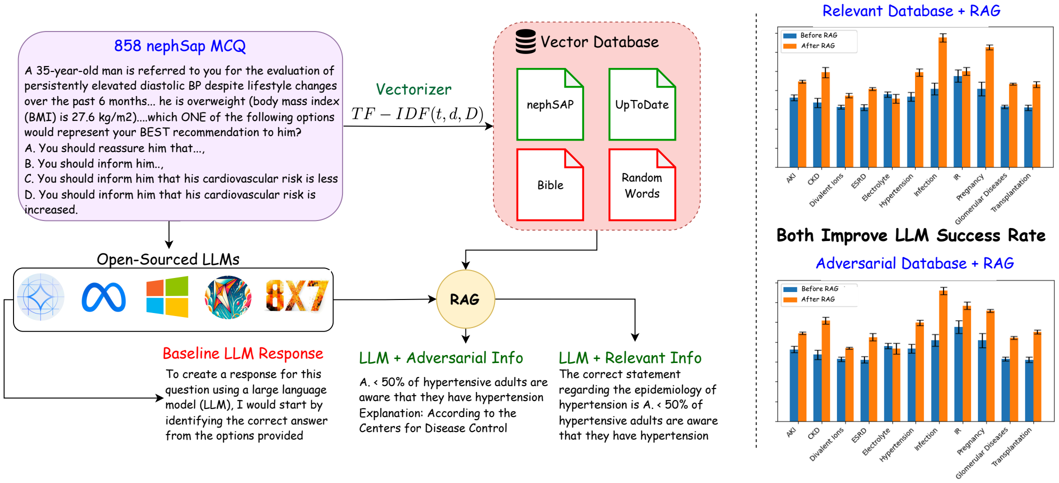
Abstract

Retrieval-Augmented Generation (RAG) is utilized to enhance large language model (LLM) performance by leveraging external knowledge databases. While it is generally believed that adversarial databases should negatively impact RAG's effectiveness, we tested this assumption for the first time in the context of the medical subspecialty field of Nephrology. We used several open-source LLMs, including Llama 3, Phi-3, Mixtral 8x7b, Zephyrβ, and Gemma 7B Instruct in a zero-shot RAG pipeline, incorporating both relevant databases (nephSAP and UpToDate) and adversarial databases (Bible and Random Words). Suprisingly, our results show that adversarial Bible and Random Words databases significantly improved Nephrology multiple choice question (MCQ) test-taking ability of specific LLMs. Utilizing DistilBERT's attention outputs, we provide evidence that adversarial databases can potentially affect LLM performance through changes in attention. Our findings highlight the need for further research into the mechanism(s) and generality of the effect of adversarial databases on LLM performance that we have discovered.

1 Introduction

LLMs have become a leading application in Natural Language Processing (NLP)[5]. Early language models were based on recurrent neural networks (RNNs)[23], which processed sequential data like text by storing information in network nodes. These models were useful for tasks like next-word prediction[9] and language translation[12]. The introduction of the transformer by Vaswani et al.[27]

AdvML-Frontiers'24: The 3nd Workshop on New Frontiers in Adversarial Machine Learning@NeurIPS'24, Vancouver, CA.



A 35-year-old man is referred to you for the evaluation of persistently elevated diastolic BP despite lifestyle changes over the past 6 months... he is overweight (body mass index (BMI) is 27.6 kg/m2)....which ONE of the following options would represent your BEST recommendation to him?
A. You should reassure him that...,
B. You should inform him..,
C. You should inform him that his cardiovascular risk is less
D. You should inform him that his cardiovascular risk is increased.

To create a response for this question using a large language model (LLM), I would start by identifying the correct answer from the options provided

A. < 50% of hypertensive adults are aware that they have hypertension Explanation: According to the Centers for Disease Control

The correct statement regarding the epidemiology of hypertension is A. < 50% of hypertensive adults are aware that they have hypertension

Figure 1: Overall methodology used to demonstrate that in RAG-based settings, adversarial databases counterintuitively can improve the success of correctly answering domain specific MCQ for specific LLMs.

in 2017 revolutionized NLP, with self-attention[10] and cross-attention mechanisms enabling the development of larger, more effective models like OpenAI's GPT series[19]. Recent advancements have led to powerful LLMs such as GPT-4[17, 18], Google's PaLM[6], Meta's Llama[25], and Anthropic's Claude[4], all excelling in benchmarks like ScienceQA[20] and the USMLE[15]. Despite the success of proprietary LLMs, open-source models still lag in certain fields[29]. Primitive algorithms like in-context learning and instruction following allow LLMs to perform tasks without parameter updates by providing examples or instructions in the context window. A more advanced technique is retrieval-augmented generation (RAG)[13], which enhances LLM performance by incorporating external knowledge from vector databases. In RAG, a user query is vectorized, and the closest matching vectors from a database are retrieved to provide context, improving the model's ability to answer queries. RAG has proven effective in domains like legal question answering[28], financial analysis[14], and medicine[30].

1.1 Problem Definition

In this paper, we address the unexplored question of how adversarial information databases used by RAG can affect the success of LLMs. To address this question, we re-utilized the dataset from our previous investigation, which consisted of 858 multiple-choice questions and answers in the subspecialty medical field of Nephrology (Nephrology Self-Assessment Program (nephSAP)). Two databases were created that incorporated relevant Nephrology background information to address the question and answer dataset; the nephSAP syllabus and the UpToDate Nephrology clinical corpus. Furthermore, two additional databases were created for comparison that contained adversarial background information with respect to their not *a priori* being expected to improve the LLM test taking ability; Bible text and a separate Random Words text file. We tested the following open-source LLMs: Mixtral 8x7b, Llama 3, Phi-3, Zephyrβ, and Gemma 7b instruct and compared how the RAG methodology using relevant Nephrology background information compared to adversarial background information in modifying the test-taking success rate of each LLM. A full pipeline of this research is visualized in Figure 1.

2 Methods

This section covers the data sources, criteria for relevant or adversarial corpora, the open-source LLMs used—Mixtral 8x7b, Gemma 7b Instruct, Llama 3, Zephyrβ, Phi-3—and a comprehensive RAG pipeline for vector databases.

2.1 Databases and Definition of Irrelevance

We tested the various open-source LLMs test-taking abilities by utilizing 858 multiple-choice questions and answers in the medical subspecialty field of Nephrology (Nephrology self assessment program (nephSAP))[29]. These patient-oriented questions address various topics in Nephrology. We deployed two relevant databases: nephSAP and UpToDate. The nephSAP syllabus consists of reviews of topics and the latest developments in Nephrology (encompassing information from March 2016 to April 2023). UpToDate is an evidence-based corpus, provides diagnostic and therapeutic information in Nephrology (generated from information available as of March 2023). For non-Nephrology (adversarial) databases, we generated a corpus of Bible text (Latin Vulgate[1]) and in addition a separate random word database using Python's Random Word package. The nephSAP database contains 247,750 lines of text, 1,790,131 words, 60,850 number of unique words, has an average word length of 5.24 characters, and a Flesch-Kincaid grade level of 11.20. UpToDate is a longer corpus with 880,850 lines of text, 7,843,922 words, 47,919 unique words, an average word length of 5.59, and a Flesch-Kincaid grade level of 14.70. The Bible database text has 118,923 lines, 934,970 words, 17,714 unique words, average word length of 3.97, and a lower reading grade level of 8.70. Finally, the Random Words database text contains 1,912,311 words, with an undefined number of sentences, average word length of 9.55 with a reading level score of 745,823.40 that by definition can be ignored. The nephSAP and UpToDate have high attributed reading levels because they are advanced medical corpuses.

Table 1: Comparison of databases used for RAG

Source	**Terminology Matching**		**Embedding**
	Unique Matches	**Overlap (%)**	**GloVe Vector Proximity**
nephSAP	903	33.3	0.80
UpToDate	733	27.1	0.80
Bible	67	2.47	0.59
Random Words	240	8.86	0.06

The nephSAP and UpToDate Nephrology databases were chosen as sources of information that would potentially enable LLMs to more successfully answer the set set of MCQ. The nephSAP database in particular would be predicted to be most informative in this regard because the corpus of information provides a background for the 858 questions. In contrast, the Bible and Random Words databases would not be predicted to provide useful information.

To quantify how relevant the four text databases are to the field of Nephrology, we first curated a Nephrology term dataset using GPT-4o. This dataset consists of medical terms in Nephrology, totaling 2,709 unique words stored in a set. The first demonstration of relevance involved comparing the number of unique matches in each of the databases. The nephSAP database had the highest number of unique matches (903), followed by UpToDate (733), whereas the Bible had 67 unique matches, and Random Words 240. We also examined the percent overlap of each database, where nephSAP had the highest percentage (33.3%), followed by UpToDate (27.1%). Both the Bible and Random Words databases had a minimal overlap (2.47% and 8.86%). Relevance and irrelevance can also be quantified in the embedding space. To do this, we deployed a pre-trained GloVe model. GloVe constructs the word vectors of the Nephrology dataset terms and the four databases by factorizing the word co-occurrence matrix. Specifically, we used the glove-wiki-gigaword-50 model from the gensim package to quantify the embeddings and then calculated the cosine similarity scores. The nephSAP and UpToDate both had a score of 0.80, whereas the Bible and Random Words had values of 0.59 and 0.06 respectively. These results are depicted in Table 1.

2.2 Open-Source Large Language Models

We examined several open-source LLMs, including Llama 3, Phi-3, Mixtral 8x7b, Gemma, and Zephyrβ. Each model was deployed using either the HuggingFace pipeline or Ollama. Llama 3, a foundation model from Meta AI [16], utilizes a decoder-only transformer architecture with over 15 trillion tokens. We used the 8-billion parameter fine-tuned version from HuggingFace [3]. Phi-3, a smaller 3.8 billion parameter model from Microsoft [2], was tested using its extended context version from HuggingFace. Mixtral 8x7b, a "Mixture of Experts" model [11], features eight unique feedforward blocks per layer, each acting as an expert. We used the Mixtral-8x7B-Instruct-v0.1 version. Gemma, from Google DeepMind [24], configured similarly to the Gemini models, was deployed using Ollama with 7 billion parameters and 4-bit quantization for faster inference. Zephyrβ [26], trained via distilled supervised fine-tuning (dSFT) [7], was also run using Ollama with 4-bit quantization.

2.3 Retrieval Augmented Generation

To create a retrieval augmented generation (RAG) system, we first divided our text databases into 1000-word chunks to use as context for the open-source LLMs. Each chunk's embedding was obtained using a TF-IDF vectorizer and stored in a vector database. During inference, we retrieved the most relevant chunks based on cosine similarity between the query and the chunk vectors. The input was formatted as “Context:" “Question:", then “Answer:", following an instruction-based prompting strategy. For each of the 858 MCQs, the context (patient background information) was fed into the RAG pipeline. We chose a TF-IDF vectorizer for its computational efficiency compared to BERT[8]. A more formalized RAG pipeline representation is provided below. The MCQ success rate was measured across four trials for each database, with the means and standard errors analyzed to compare RAG's effect.

1. **Compute TF-IDF**: Compute TF-IDF vectors for the query and each document:

$$\mathbf{v}_Q = \text{TF-IDF}(Q), \quad \mathbf{v}_{D_i} = \text{TF-IDF}(D_i),\ \forall i.$$

2. **Calculate Cosine Similarity**: Calculate the cosine similarity between the query vector and each document vector:

$$\text{cosine\_sim}(\mathbf{v}_Q, \mathbf{v}_{D_i}) = \frac{\mathbf{v}_Q \cdot \mathbf{v}_{D_i}}{\|\mathbf{v}_Q\|\|\mathbf{v}_{D_i}\|}.$$

3. **Retrieve Top 3 Chunks**: Identify the top 3 document indices with the highest cosine similarity scores:

$$\{i_1, i_2, i_3\} = \arg\max_i \text{top 3} \left(\text{cosine\_sim}(\mathbf{v}_Q, \mathbf{v}_{D_i})\right).$$

3 Results

Table 2: Open-source LLM's where RAG on adversarial databases improved question answering

Mixtral 8x7b

Source	Mean (%)	SEM	vs Baseline	vs Bible	vs nephSAP	vs UpToDate	vs Random
Baseline	40.2	0.34	–	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$
nephSAP	59.2	0.50	$p < 0.001$	$p < 0.001$	–	$p < 0.001$	$p < 0.001$
UpToDate	55.3	0.55	$p < 0.001$	NS	$p < 0.001$	–	NS
Bible	54.6	0.51	$p < 0.001$	–	$p < 0.001$	NS	NS
Random Words	54.3	0.22	$p < 0.001$	NS	$p < 0.001$	NS	–

Gemma 7b Instruct

Source	Mean (%)	SEM	vs Baseline	vs Bible	vs nephSAP	vs UpToDate	vs Random
Baseline	36.8	0.27	–	$p < 0.05$	$p < 0.001$	NS	$p < 0.05$
nephSAP	41.1	0.34	$p < 0.001$	$p < 0.001$	–	$p < 0.001$	$p < 0.001$
UpToDate	37.2	0.17	NS	NS	$p < 0.001$	–	NS
Bible	38.1	0.34	$p < 0.05$	–	$p < 0.001$	NS	NS
Random Words	38.1	0.37	$p < 0.05$	NS	$p < 0.001$	NS	–

Zephyrβ 7b

Source	Mean (%)	SEM	vs Baseline	vs Bible	vs nephSAP	vs UpToDate	vs Random
Baseline	29.3	0.01	–	$p < 0.004$	$p < 0.001$	$p < 0.001$	$p < 0.001$
nephSAP	33.4	0.004	$p < 0.001$	NS	–	NS	$p < 0.001$
UpToDate	32.9	0.005	$p < 0.001$	NS	NS	–	NS
Bible	32.3	0.01	$p < 0.004$	–	NS	NS	$p < 0.001$
Random Words	21.2	0.002	$p < 0.001$	$p < 0.001$	$p < 0.001$	NS	–

Table 3: Open-source LLM's where RAG on adversarial databases did not improve question answering

Llama 3 8b

Source	Mean (%)	SEM	vs Baseline	vs Bible	vs nephSAP	vs UpToDate	vs Random
Baseline	53.7	0.17	–	$p < 0.002$	$p < 0.001$	$p < 0.05$	$p < 0.001$
nephSAP	57.0	0.30	$p < 0.001$	$p < 0.001$	–	$p < 0.05$	$p < 0.001$
UpToDate	55.4	0.42	$p < 0.05$	$p < 0.001$	$p < 0.05$	–	$p < 0.001$
Bible	51.7	0.33	$p < 0.002$	–	$p < 0.001$	$p < 0.001$	$p < 0.001$
Random Words	40.4	0.40	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–

Phi-3 128k

Source	Mean (%)	SEM	vs Baseline	vs Bible	vs nephSAP	vs UpToDate	vs Random
Baseline	51.4	0.60	–	$p < 0.003$	NS	$p < 0.005$	$p < 0.001$
nephSAP	51.0	0.23	NS	$p < 0.01$	–	$p < 0.05$	$p < 0.001$
UpToDate	48.4	0.77	$p < 0.005$	NS	$p < 0.05$	–	$p < 0.001$
Bible	48.2	0.50	$p < 0.003$	–	$p < 0.01$	NS	$p < 0.001$
Random Words	42.4	0.42	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–

We analyze scenarios where adversarial information improved the correct question answering on some LLMs, did not make a significant improvement, or even made the question answering worse. We also highlight scenarios where the adversarial information effect was not present. For each LLM, the MCQ were answered in four independent experiments. One-way ANOVA and Dunnett's test were used to compare multiple group means. The results are depicted as mean ± SEM, where $p < 0.05$ was considered significant. (See Appendix for subcategory analysis).

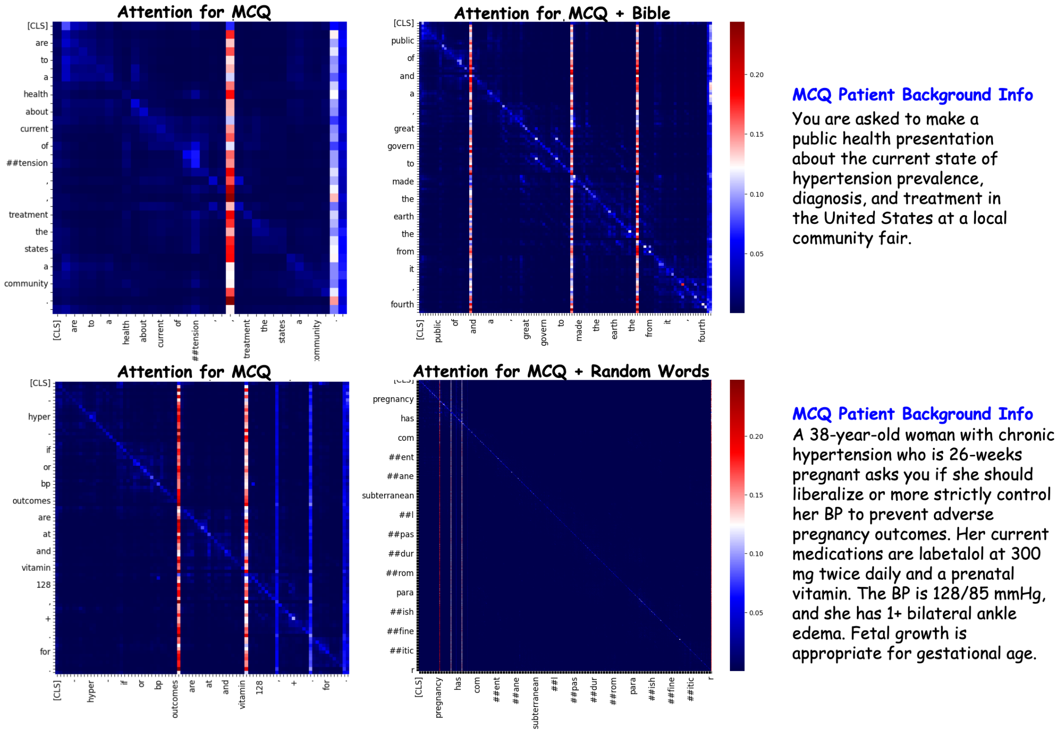
As shown in Table 2 using the nephSAP database, most models significantly improved their test-taking ability except the Phi-3 128k LLM (Table 3) which had no significant change. The improvement was model-dependent and varied from 4.1% (Zephyrβ 7b) to 19% (Mixtral 8x7b). The UpToDate database also improved the test-taking success rate in all LLMs, 0.4% (Gemma 7b Instruct) to 14.1% (Mixtral 8x7b) except for the Phi-3 128k LLM where the percent of correctly answered questions actually decreased significantly from 51.4% to 48.4%. Finally, there was no clear correlation between the intrinsic ability of a given LLM and the magnitude of the RAG-based improvement.

4 Discussion

We have uncovered the novel phenomenon that adversarial information databases are capable of improving RAG-based LLM accuracy, and in some instances are essentially equivalent to relevant databases. To our knowledge this effect has not been previously described. Specifically, we found that adversarial database information was able to significantly improve the the success rate of specific LLMs to answer MCQ accurately in the subspecialty medical field of Nephrology. The finding that various LLMs showed the same phenomenon to various degrees, and that two independent databases (Bible and Random Words) with adversarial information were each effective in certain circumstances, suggest that the phenomenon is not specific to the exact conditions of our experiments. Our findings are potentially generalizable to other domains, where RAG-based approaches are utilized to improve the capability of LLMs.

4.1 Role of Attention Mechanism in RAG Effects

One possible explanation for this phenomenon is the attention mechanism in LLMs. In a non-RAG scenario, the prompt embeddings are passed into a positional encoder, in which each attention score for token i is computed with respect to token j with the query matrix Q, key matrix K, and value matrix V, with also a d_k factor. Accordingly, in a non-RAG scenario where the attention scores of the prompt are computed alone, the attention mechanism is only focused on the prompt. However, when external knowledge is retrieved via RAG, the attention mechanism has more tokens to account for. Therefore, when analyzing the transformer architecture, there is a shift in attention between the tokens in the original prompt even when adversarial information is utilized. When retrieved adversarial information is passed to the multiple-choice question and answer prompt, the latent shift that occurs within the input token representation shifts, which may cause the attention mechanism to emphasize specific parts of the prompt to a greater degree. These findings are consistent with the


Attention for MCQ

Attention for MCQ + Bible

Attention for MCQ

Attention for MCQ + Random Words

MCQ Patient Background Info

You are asked to make a public health presentation about the current state of hypertension prevalence, diagnosis, and treatment in the United States at a local community fair.

MCQ Patient Background Info

A 38-year-old woman with chronic hypertension who is 26-weeks pregnant asks you if she should liberalize or more strictly control her BP to prevent adverse pregnancy outcomes. Her current medications are labetalol at 300 mg twice daily and a prenatal vitamin. The BP is 128/85 mmHg, and she has 1+ bilateral ankle edema. Fetal growth is appropriate for gestational age.

Figure 2: Visualization of DistilBERT attention outputs given both a MCQ prompt and also a Bible or Random Words + MCQ prompt. An evident difference in the weighting matrix is demonstrated.

results of adversarial attacks on LLMs [22], which demonstrate that even a small change in prompt formatting can produce very different LLM outputs.

We demonstrate an example of this attention shift in the multiple choice question and answer prompt and a section from either the Bible or Random Words databases (Figure 2). To visualize this shift, we deployed a DistilBERT[21] model, which is a small scale general pre-trained transformer, and pass both the prompt and the Bible or Random Words + prompt. We then visualized the attention outputs in a heatmap (Figure 3). For simplicity, we extracted the attention outputs from only the final layer. It is evident that, given an excerpt from the Bible database, there is a significant attention shift, which can lead to different and, in this case, improved results.

4.2 Implications on RAG and Future Directions

This work has further implications for future research in retrieval based mechanisms. In some scenarios, the performance of RAG cannot always be attributed to the vector database itself. Importantly, inherent attention mechanisms within the LLM's transformer architecture may come into play. When using RAG, one typically employs databases that contain useful information. Curating these databases can be both time consuming and expensive. Our results suggest that with certain LLMs, it is possible to use non-curated adversarial information to obtain improved LLM results. By simply injecting more tokens into the input stream and shifting the LLM attention span, it might be possible in certain instances improve the accuracy of the the downstream task. This work has additional implications in the area of retrieval based mechanism research, given that the LLM RAG-based performance cannot always be attributed to the vector database itself. We propose a possible explanation for our finding that is based on LLM attention mechanisms. However, further research is needed to determine whether other underlying mechanisms are involved, so that one can predict in a particular scenario exactly when using an adversarial information database with a RAG-based approach, the success rate of a specific LLM will significantly improve or not.

5 Conclusion

In summary, we tested 858 subspecialty MCQ in Nephrology in various retrieval-based scenarios. We experimented with RAG using databases with relevant background information (nephSAP and UpToDate), as well as adversarial databases (Bible text and Random Words). We found that adversarial databases in certain instances improved the test-taking ability of specific open-source LLMs comparable to relevant information databases. We highlight the importance of this previously unrecognized novel effect and provide evidence that one potential mechanism involves the injection of more tokens into the input stream as a basis for shifting LLM attention. All code and data are open-source and available.

Acknowledgments

Ira Kurtz is supported in part by funds from the Smidt Family Foundation, the Factor Family Foundation, the Davita Allen Nissenson Research Fund and the Ralph Block Family Foundation.

References

[1] *The Holy Bible: Translated from the Latin Vulgate, Diligently Compared with the Hebrew, Greek, and Other Editions in Divers Languages*. Douay-rheims version edition, 1609. Originally published in 1582.

[2] M. Abdin, S. A. Jacobs, A. A. Awan, J. Aneja, A. Awadallah, H. Awadalla, N. Bach, A. Bahree, A. Bakhtiari, H. Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

[3] AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

[4] Anthropic. Introducing claude, 2023. URL https://www.anthropic.com/.

[5] Y. Chang, X. Wang, J. Wang, Y. Wu, K. Zhu, H. Chen, and X. Xie. A survey on evaluation of large language models. *arXiv preprint arXiv:2307.03109*, 2023.

[6] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, and N. Fiedel. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[7] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[9] A. F. Ganai and F. Khursheed. Predicting next word using rnn and lstm cells: Stastical language modeling. pages 469–474, 2019.

[10] Y. Hao, L. Dong, F. Wei, and K. Xu. Self-attention attribution: Interpreting information interactions inside transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12963–12971, 2021.

[11] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. d. l. Casas, E. B. Hanna, F. Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[12] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, and W. Zhang. A comparative study on transformer vs rnn in speech applications. pages 449–456, 2019.

[13] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[14] X. Li, Z. Li, C. Shi, Y. Xu, Q. Du, M. Tan, J. Huang, and W. Lin. Alphafin: Benchmarking financial analysis with retrieval-augmented stock-chain framework. *arXiv preprint arXiv:2403.12582*, 2024.

[15] A. B. Mbakwe, I. Lourentzou, L. A. Celi, O. J. Mechanic, and A. Dagan. Chatgpt passing usmle shines a spotlight on the flaws of medical education. *PLOS Digital Health*, 2(2):e0000205, 2023.

[16] Meta AI. Introducing meta llama 3: The most capable openly available llm to date. https://ai.meta.com/blog/meta-llama-3/, April 2024. Accessed: 2024-05-21.

[17] OpenAI. Chatgpt, 2022. URL https://www.openai.com/.

[18] R. OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[19] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. *arXiv preprint arXiv:1806.01261*, 2018.

[20] T. Saikh, T. Ghosal, A. Mittal, A. Ekbal, and P. Bhattacharyya. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3): 289–301, 2022.

[21] V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[22] M. Sclar, Y. Choi, Y. Tsvetkov, and A. Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *arXiv preprint arXiv:2310.11324*, 2023.

[23] A. Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306, 2020.

[24] G. Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivière, M. S. Kale, J. Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[25] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. A. Lachaux, T. Lacroix, and G. Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[26] L. Tunstall, E. Beeching, N. Lambert, N. Rajani, K. Rasul, Y. Belkada, S. Huang, L. von Werra, C. Fourrier, N. Habib, et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, volume 30, 2017.

[28] N. Wiratunga, R. Abeyratne, L. Jayawardena, K. Martin, S. Massie, I. Nkisi-Orji, R. Weerasinghe, A. Liret, and B. Fleisch. Cbr-rag: Case-based reasoning for retrieval augmented generation in llms for legal question answering. *arXiv preprint arXiv:2404.04302*, 2024.

[29] S. Wu, M. Koo, L. Blum, A. Black, L. Kao, F. Scalzo, and I. Kurtz. A comparative study of open-source large language models, gpt-4 and claude 2: Multiple-choice test taking in nephrology. *arXiv preprint arXiv:2308.04709*, 2023.

[30] G. Xiong, Q. Jin, Z. Lu, and A. Zhang. Benchmarking retrieval-augmented generation for medicine. *arXiv preprint arXiv:2402.13178*, 2024.

A Appendix / supplemental material

B Compute

We leveraged Google Colab for cloud CPU and GPU (Nvidia Tesla T4) for preprocessing and running BERT models. Experiments were conducted on a university cluster with eight Nvidia RTX A5000 GPUs, each with 24 GB of memory.

C Analysis on Nephrology Subcategories

Table 4: Mixtral 8x7b RAG percent of MCQ answered correctly by subcategory

Source	**Mean (%)**	**SEM**	**vs Baseline**	**vs nephSAP**	**vs UpToDate**	**vs Bible**	**vs Random**
Hypertension							
Baseline	48.9	1.62	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	70.8	0.79	p < 0.001	–	p < 0.01	p < 0.005	p < 0.001
UpToDate	63.2	0.84	p < 0.001	p < 0.01	–	NS	NS
Bible	62.4	2.45	p < 0.001	p < 0.005	NS	–	NS
Random	60.4	0.28	p < 0.001	p < 0.001	NS	NS	–
Glomerular Diseases							
Baseline	35.2	1.24	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	54.9	1.46	p < 0.001	–	NS	p < 0.05	p < 0.001
UpToDate	50.8	0.69	p < 0.001	NS	–	NS	NS
Bible	50.3	0.73	p < 0.001	p < 0.05	NS	–	NS
Random	46.8	1.07	p < 0.001	p < 0.001	NS	NS	–
AKI							
Baseline	40.2	1.16	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	60.7	0.92	p < 0.001	–	p < 0.001	p < 0.05	NS
UpToDate	51.7	1.59	p < 0.001	p < 0.001	–	NS	NS
Bible	55.3	2.02	p < 0.001	p < 0.05	NS	–	NS
Random	56.2	0.46	p < 0.001	NS	NS	NS	–
Divalent Ions							
Baseline	40.5	1.67	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	51.5	0.98	p < 0.001	–	NS	NS	NS
UpToDate	48.9	0.98	p < 0.001	NS	–	NS	NS
Bible	48.5	1.22	p < 0.001	NS	NS	–	NS
Random	48.5	0.98	p < 0.001	NS	NS	NS	–
Transplant							
Baseline	36.1	1.9	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	60.3	1.89	p < 0.001	–	NS	p < 0.05	NS
UpToDate	56.7	1.43	p < 0.001	NS	–	NS	NS
Bible	51.7	0.72	p < 0.001	p < 0.05	p < 0.005	–	NS
Random	54.4	2.87	p < 0.001	NS	NS	NS	–
CKD							
Baseline	45.8	1.05	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	63.1	2.0	p < 0.001	–	NS	NS	NS
UpToDate	63.3	1.2	p < 0.001	NS	–	NS	NS
Bible	61.1	2.36	p < 0.001	NS	NS	–	NS
Random	58.9	0.79	p < 0.001	NS	NS	NS	–
ESRD							
Baseline	34.2	2.77	–	p < 0.001	p < 0.001	p < 0.001	p < 0.001
nephSAP	59.4	1.16	p < 0.001	–	NS	NS	NS
UpToDate	55.8	2.54	p < 0.001	NS	–	NS	NS

Continued on next page

Table 4 – continued from previous page

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Bible	52.2	1.28	p < 0.001	NS	NS	–	NS
Random	55.3	2.28	p < 0.001	NS	NS	NS	–
				Electrolyte			
Baseline	37.9	2.99	–	p < 0.001	p < 0.05	p < 0.001	p < 0.001
nephSAP	53.0	1.63	p < 0.001	–	NS	NS	NS
UpToDate	46.1	1.29	p < 0.05	NS	–	NS	NS
Bible	51.7	1.22	p < 0.001	NS	NS	–	NS
Random	53.0	1.91	p < 0.001	NS	NS	NS	–
				Pregnancy			
Baseline	42.5	4.17	–	p < 0.05	p < 0.05	p < 0.05	p < 0.01
nephSAP	59.2	3.44	p < 0.05	–	NS	NS	NS
UpToDate	57.5	4.17	p < 0.05	NS	–	NS	NS
Bible	60.0	1.92	p < 0.05	NS	NS	–	NS
Random	61.7	3.97	p < 0.01	NS	NS	NS	–
				IR			
Baseline	46.7	3.04	–	p < 0.05	p < 0.05	p < 0.05	p < 0.01
nephSAP	60.0	2.36	p < 0.05	–	NS	NS	NS
UpToDate	60.0	2.36	p < 0.05	NS	–	NS	NS
Bible	61.7	1.67	p < 0.05	NS	NS	–	NS
Random	62.5	4.59	p < 0.01	NS	NS	NS	–
				Infection			
Baseline	45.8	6.29	–	p < 0.005	p < 0.001	p < 0.05	p < 0.05
nephSAP	67.5	2.85	p < 0.005	–	NS	NS	NS
UpToDate	69.2	2.10	p < 0.001	NS	–	NS	NS
Bible	62.5	1.60	p < 0.05	NS	NS	–	NS
Random	60.0	3.33	p < 0.05	NS	NS	NS	–

Table 5: Gemma RAG percent of MCQ answered correctly by subcategory

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
				Hypertension			
Baseline	38.49	1.68	–	$p < 0.05$	NS	NS	NS
nephSAP	44.38	0.97	$p < 0.05$	–	$p < 0.002$	NS	$p < 0.05$
UpToDate	35.40	1.41	NS	$p < 0.002$	–	NS	NS
Bible	39.61	1.16	NS	NS	NS	–	NS
Random	37.64	1.86	NS	$p < 0.05$	NS	NS	–
				Glomerular Diseases			
Baseline	29.20	0.64	–	$p < 0.001$	$p < 0.001$	$p < 0.002$	NS
nephSAP	35.40	0.42	$p < 0.001$	–	$p < 0.05$	$p < 0.001$	$p < 0.001$
UpToDate	33.06	0.42	$p < 0.001$	$p < 0.05$	–	NS	$p < 0.003$
Bible	32.21	0.48	$p < 0.002$	$p < 0.001$	NS	–	$p < 0.05$
Random	30.20	0.47	NS	$p < 0.001$	$p < 0.003$	$p < 0.05$	–
				AKI			
Baseline	43.26	1.75	–	NS	NS	NS	NS
nephSAP	45.79	0.96	NS	–	NS	NS	NS
UpToDate	45.51	0.97	NS	NS	–	NS	NS
Bible	43.54	0.28	NS	NS	NS	–	NS
Random	45.51	0.33	NS	NS	NS	NS	–

Continued on next page

Table 5 – continued from previous page

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Divalent Ions							
Baseline	37.39	1.67	–	NS	NS	NS	$p < 0.05$
nephSAP	39.60	0.98	NS	–	$p < 0.003$	NS	NS
UpToDate	33.18	0.85	NS	$p < 0.003$	–	NS	$p < 0.001$
Bible	37.39	1.27	NS	NS	NS	–	$p < 0.05$
Random	42.04	0.26	$p < 0.05$	NS	$p < 0.001$	$p < 0.05$	–
Transplant							
Baseline	34.72	0.28	–	$p < 0.05$	NS	NS	NS
nephSAP	39.17	1.14	$p < 0.05$	–	$p < 0.001$	$p < 0.001$	$p < 0.001$
UpToDate	30.83	0.53	NS	$p < 0.001$	–	NS	NS
Bible	32.22	0.00	NS	$p < 0.001$	NS	–	NS
Random	31.11	1.98	NS	$p < 0.001$	NS	NS	–
CKD							
Baseline	42.78	1.40	–	$p < 0.05$	NS	NS	NS
nephSAP	49.44	2.15	$p < 0.05$	–	$p < 0.01$	NS	$p < 0.002$
UpToDate	41.67	1.06	NS	$p < 0.01$	–	NS	NS
Bible	43.89	1.47	NS	NS	NS	–	NS
Random	40.56	0.72	NS	$p < 0.002$	NS	NS	–
ESRD							
Baseline	41.11	1.20	–	NS	$p < 0.05$	NS	NS
nephSAP	40.56	0.72	NS	–	$p < 0.05$	NS	NS
UpToDate	45.28	0.53	$p < 0.05$	$p < 0.05$	–	NS	NS
Bible	44.44	0.46	NS	NS	NS	–	NS
Random	43.06	1.84	NS	NS	NS	NS	–
Electrolyte							
Baseline	32.33	0.83	–	NS	NS	NS	NS
nephSAP	33.62	1.11	NS	–	NS	NS	NS
UpToDate	33.19	0.43	NS	NS	–	NS	NS
Bible	33.19	1.09	NS	NS	NS	–	NS
Random	32.33	0.83	NS	NS	NS	NS	–
Pregnancy							
Baseline	42.50	1.60	–	NS	NS	NS	$p < 0.01$
nephSAP	46.67	0.00	NS	–	$p < 0.003$	$p < 0.05$	NS
UpToDate	39.17	0.83	NS	$p < 0.003$	–	NS	$p < 0.001$
Bible	40.83	1.59	NS	$p < 0.05$	NS	–	$p < 0.001$
Random	49.17	1.59	$p < 0.01$	NS	$p < 0.001$	$p < 0.001$	–
IR							
Baseline	22.50	1.60	–	$p < 0.01$	$p < 0.003$	$p < 0.01$	$p < 0.001$
nephSAP	30.00	0.00	$p < 0.01$	–	NS	NS	NS
UpToDate	30.83	1.59	$p < 0.003$	NS	–	NS	NS
Bible	30.00	1.36	$p < 0.01$	NS	NS	–	NS
Random	31.67	1.67	$p < 0.001$	NS	NS	NS	–
Infection							
Baseline	40.00	0.00	–	$p < 0.001$	$p < 0.05$	$p < 0.05$	$p < 0.001$
nephSAP	54.17	0.83	$p < 0.001$	–	$p < 0.001$	$p < 0.001$	$p < 0.005$
UpToDate	45.84	0.84	$p < 0.05$	$p < 0.001$	–	NS	NS
Bible	45.00	2.15	$p < 0.05$	$p < 0.001$	NS	–	NS
Random	47.50	0.83	$p < 0.001$	$p < 0.005$	NS	NS	–

Table 6: Zephyrβ RAG percent of MCQ answered correctly by subcategory

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Hypertension							
Baseline	31.46	0.65	–	NS	NS	NS	$p < 0.001$
nephSAP	33.15	2.92	NS	–	NS	NS	$p < 0.001$
UpToDate	35.11	2.12	NS	NS	–	NS	$p < 0.001$
Bible	30.90	0.73	NS	NS	NS	–	$p < 0.001$
Random	21.35	0.46	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Glomerular Diseases							
Baseline	25.95	0.22	–	$p < 0.05$	$p < 0.05$	$p < 0.05$	$p < 0.01$
nephSAP	31.71	1.46	$p < 0.05$	–	NS	NS	$p < 0.001$
UpToDate	31.55	1.13	$p < 0.05$	NS	–	NS	$p < 0.001$
Bible	31.21	1.87	$p < 0.05$	NS	NS	–	$p < 0.001$
Random	18.62	0.57	$p < 0.01$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
AKI							
Baseline	35.21	0.75	–	NS	NS	NS	$p < 0.001$
nephSAP	35.12	0.54	NS	–	NS	NS	$p < 0.001$
UpToDate	33.43	0.71	NS	NS	–	NS	$p < 0.001$
Bible	35.11	0.28	NS	NS	NS	–	$p < 0.001$
Random	25.00	1.25	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Divalent Ions							
Baseline	22.42	2.36	–	NS	NS	$p < 0.05$	$p < 0.05$
nephSAP	23.89	1.40	NS	–	NS	NS	$p < 0.01$
UpToDate	24.12	0.84	NS	NS	–	NS	$p < 0.004$
Bible	28.10	0.56	$p < 0.05$	NS	NS	–	$p < 0.001$
Random	17.04	1.22	$p < 0.05$	$p < 0.01$	$p < 0.004$	$p < 0.001$	–
Transplant							
Baseline	25.55	1.70	–	$p < 0.001$	$p < 0.05$	$p < 0.001$	$p < 0.01$
nephSAP	34.44	0.79	$p < 0.001$	–	NS	NS	$p < 0.001$
UpToDate	31.11	0.79	$p < 0.05$	NS	–	NS	$p < 0.001$
Bible	33.33	0.45	$p < 0.001$	NS	NS	–	$p < 0.001$
Random	19.45	1.47	$p < 0.01$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
CKD							
Baseline	30.00	2.31	–	$p < 0.001$	$p < 0.001$	$p < 0.05$	NS
nephSAP	39.17	0.53	$p < 0.001$	–	NS	NS	$p < 0.001$
UpToDate	40.00	0.79	$p < 0.001$	NS	–	$p < 0.05$	$p < 0.001$
Bible	35.28	1.66	$p < 0.05$	NS	$p < 0.05$	–	$p < 0.001$
Random	26.11	0.56	NS	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
ESRD							
Baseline	32.22	1.70	–	$p < 0.05$	NS	NS	$p < 0.001$
nephSAP	38.89	0.79	$p < 0.05$	–	NS	NS	$p < 0.001$
UpToDate	36.95	1.23	NS	NS	–	NS	$p < 0.001$
Bible	34.17	1.15	NS	NS	NS	–	$p < 0.001$
Random	21.95	2.00	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Electrolyte							
Baseline	25.86	4.34	–	NS	NS	NS	NS
nephSAP	17.67	3.62	NS	–	NS	NS	NS
UpToDate	16.81	2.48	NS	NS	–	NS	NS
Bible	18.10	3.19	NS	NS	NS	–	NS
Random	23.28	0.50	NS	NS	NS	NS	–
Pregnancy							
Baseline	40.00	1.92	–	NS	$p < 0.05$	NS	$p < 0.001$

Continued on next page

Table 6 – continued from previous page

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
nephSAP	48.34	3.47	NS	–	NS	NS	$p < 0.001$
UpToDate	50.83	2.85	$p < 0.05$	NS	–	NS	$p < 0.001$
Bible	47.50	2.10	NS	NS	NS	–	$p < 0.001$
Random	19.17	0.83	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
IR							
Baseline	35.56	1.11	–	$p < 0.01$	NS	NS	$p < 0.05$
nephSAP	48.34	0.96	$p < 0.01$	–	NS	NS	$p < 0.001$
UpToDate	44.17	0.84	NS	NS	–	NS	$p < 0.001$
Bible	40.84	2.10	NS	NS	NS	–	$p < 0.001$
Random	23.33	4.08	$p < 0.05$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Infection							
Baseline	37.78	1.11	–	NS	NS	NS	$p < 0.001$
nephSAP	35.84	0.84	NS	–	NS	NS	$p < 0.001$
UpToDate	39.17	1.59	NS	NS	–	NS	$p < 0.001$
Bible	36.67	0.00	NS	NS	NS	–	$p < 0.001$
Random	21.67	2.89	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–

Table 7: Llama 3 RAG percent of MCQ answered correctly by subcategory

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Hypertension							
Baseline	58.43	0.00	–	NS	NS	NS	$p < 0.001$
nephSAP	59.83	0.96	NS	–	NS	NS	$p < 0.001$
UpToDate	57.87	0.73	NS	NS	–	NS	$p < 0.001$
Bible	57.87	0.73	NS	NS	NS	–	$p < 0.001$
Random	48.04	0.84	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Glomerular Diseases							
Baseline	45.31	0.43	–	$p < 0.001$	$p < 0.001$	NS	$p < 0.001$
nephSAP	54.70	0.64	$p < 0.001$	–	$p < 0.002$	$p < 0.001$	$p < 0.001$
UpToDate	50.84	0.64	$p < 0.001$	$p < 0.002$	–	$p < 0.001$	$p < 0.001$
Bible	45.47	0.50	NS	$p < 0.001$	$p < 0.001$	–	$p < 0.001$
Random	33.22	0.80	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
AKI							
Baseline	59.83	0.84	–	NS	NS	NS	$p < 0.001$
nephSAP	63.20	1.33	NS	–	NS	NS	$p < 0.001$
UpToDate	60.68	1.03	NS	NS	–	NS	$p < 0.001$
Bible	59.83	1.25	NS	NS	NS	–	$p < 0.001$
Random	44.10	0.84	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Divalent Ions							
Baseline	48.01	0.56	–	NS	NS	NS	$p < 0.001$
nephSAP	49.56	1.40	NS	–	NS	$p < 0.05$	$p < 0.001$
UpToDate	47.57	0.22	NS	NS	–	NS	$p < 0.001$
Bible	44.69	1.28	NS	$p < 0.05$	NS	–	$p < 0.001$
Random	33.85	0.91	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Transplant							
Baseline	51.67	1.06	–	NS	NS	NS	$p < 0.001$
nephSAP	52.50	0.53	NS	–	NS	NS	$p < 0.001$
UpToDate	50.83	1.23	NS	NS	–	NS	$p < 0.001$
Bible	50.28	1.23	NS	NS	NS	–	$p < 0.001$

Continued on next page

Table 7 – continued from previous page

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Random	40.00	1.20	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
CKD							
Baseline	63.33	1.76	–	NS	$p < 0.05$	NS	$p < 0.001$
nephSAP	66.39	0.83	NS	–	NS	$p < 0.05$	$p < 0.001$
UpToDate	70.83	1.59	$p < 0.05$	NS	–	$p < 0.001$	$p < 0.001$
Bible	59.44	1.84	NS	$p < 0.05$	$p < 0.001$	–	$p < 0.01$
Random	51.67	1.32	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.01$	–
ESRD							
Baseline	58.33	1.32	–	NS	NS	NS	$p < 0.001$
nephSAP	63.06	0.95	NS	–	NS	$p < 0.01$	$p < 0.001$
UpToDate	57.78	2.03	NS	NS	–	NS	$p < 0.001$
Bible	54.72	0.95	NS	$p < 0.01$	NS	–	$p < 0.001$
Random	40.56	2.05	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Electrolyte							
Baseline	47.85	1.47	–	$p < 0.05$	NS	NS	$p < 0.001$
nephSAP	42.24	0.50	$p < 0.05$	–	$p < 0.001$	$p < 0.001$	$p < 0.001$
UpToDate	51.29	0.83	NS	$p < 0.001$	–	NS	$p < 0.001$
Bible	50.86	1.49	NS	$p < 0.001$	NS	–	$p < 0.001$
Random	25.00	1.11	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Pregnancy							
Baseline	65.00	0.96	–	NS	NS	$p < 0.01$	NS
nephSAP	61.67	2.15	NS	–	NS	$p < 0.05$	NS
UpToDate	55.84	0.84	NS	NS	–	NS	NS
Bible	52.50	3.44	$p < 0.01$	$p < 0.05$	NS	–	$p < 0.05$
Random	61.67	3.19	NS	NS	NS	$p < 0.05$	–
IR							
Baseline	49.17	2.10	–	$p < 0.05$	NS	$p < 0.05$	$p < 0.001$
nephSAP	60.83	1.59	$p < 0.05$	–	NS	$p < 0.001$	$p < 0.001$
UpToDate	51.67	3.97	NS	NS	–	$p < 0.05$	$p < 0.001$
Bible	38.33	2.15	$p < 0.05$	$p < 0.001$	$p < 0.05$	–	NS
Random	28.34	3.47	$p < 0.001$	$p < 0.001$	$p < 0.001$	NS	–
Infection							
Baseline	53.34	1.93	–	NS	NS	NS	NS
nephSAP	58.34	0.96	NS	–	NS	$p < 0.05$	NS
UpToDate	55.84	0.84	NS	NS	–	NS	NS
Bible	51.67	0.96	NS	$p < 0.05$	NS	–	NS
Random	55.00	1.67	NS	NS	NS	NS	–

Table 8: Phi-3 RAG percent of MCQ answered correctly by subcategory

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Hypertension							
Baseline	60.40	0.54	–	$p < 0.05$	$p < 0.003$	NS	$p < 0.001$
nephSAP	55.90	0.71	$p < 0.05$	–	NS	NS	NS
UpToDate	54.20	1.48	$p < 0.003$	NS	–	$p < 0.05$	NS
Bible	59.00	0.56	NS	NS	$p < 0.05$	–	$p < 0.004$
Random	53.10	1.48	$p < 0.001$	NS	NS	$p < 0.004$	–
Glomerular Diseases							

Continued on next page

Table 8 – continued from previous page

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Baseline	48.50	1.53	–	NS	NS	NS	$p < 0.001$
nephSAP	50.30	0.55	NS	–	$p < 0.05$	$p < 0.05$	$p < 0.001$
UpToDate	44.50	0.57	NS	$p < 0.05$	–	NS	NS
Bible	44.50	1.21	NS	$p < 0.05$	NS	–	NS
Random	40.30	1.67	$p < 0.001$	$p < 0.001$	NS	NS	–
AKI							
Baseline	57.60	0.84	–	NS	$p < 0.001$	$p < 0.001$	$p < 0.001$
nephSAP	54.80	0.71	NS	–	$p < 0.05$	$p < 0.05$	$p < 0.001$
UpToDate	51.40	1.06	$p < 0.001$	$p < 0.05$	–	NS	$p < 0.001$
Bible	51.10	0.73	$p < 0.001$	$p < 0.05$	NS	–	$p < 0.001$
Random	45.20	0.54	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.001$	–
Divalent Ions							
Baseline	44.50	2.61	–	NS	NS	NS	$p < 0.05$
nephSAP	48.50	0.98	NS	–	NS	NS	$p < 0.002$
UpToDate	45.60	0.85	NS	NS	–	NS	$p < 0.05$
Bible	43.40	2.01	NS	NS	NS	–	NS
Random	37.60	1.79	$p < 0.05$	$p < 0.002$	$p < 0.05$	NS	–
Transplant							
Baseline	46.90	0.95	–	NS	NS	NS	NS
nephSAP	46.70	1.20	NS	–	NS	NS	NS
UpToDate	46.10	1.73	NS	NS	–	NS	NS
Bible	49.40	0.96	NS	NS	NS	–	$p < 0.005$
Random	42.80	0.96	NS	NS	NS	$p < 0.005$	–
CKD							
Baseline	55.80	1.15	–	NS	NS	$p < 0.05$	$p < 0.001$
nephSAP	54.40	0.79	NS	–	NS	NS	$p < 0.001$
UpToDate	52.20	0.91	NS	NS	–	NS	$p < 0.001$
Bible	49.70	2.00	$p < 0.05$	NS	NS	–	$p < 0.005$
Random	41.70	1.95	$p < 0.001$	$p < 0.001$	$p < 0.001$	$p < 0.005$	–
ESRD							
Baseline	48.10	1.66	–	NS	NS	NS	$p < 0.001$
nephSAP	46.40	1.23	NS	–	NS	NS	$p < 0.01$
UpToDate	43.90	0.96	NS	NS	–	NS	NS
Bible	46.10	1.16	NS	NS	NS	–	$p < 0.05$
Random	39.40	1.60	$p < 0.001$	$p < 0.01$	NS	$p < 0.05$	–
Electrolyte							
Baseline	45.70	1.11	–	NS	$p < 0.05$	$p < 0.005$	$p < 0.05$
nephSAP	39.20	3.26	NS	–	NS	NS	NS
UpToDate	37.90	1.22	$p < 0.05$	NS	–	NS	NS
Bible	34.90	1.47	$p < 0.005$	NS	NS	–	NS
Random	37.50	1.91	$p < 0.05$	NS	NS	NS	–
Pregnancy							
Baseline	67.50	2.85	–	NS	NS	NS	$p < 0.004$
nephSAP	56.70	2.36	NS	–	NS	NS	NS
UpToDate	69.20	2.50	NS	NS	–	NS	$p < 0.002$
Bible	60.00	4.08	NS	NS	NS	–	NS
Random	48.30	4.41	$p < 0.004$	NS	$p < 0.002$	NS	–
IR							
Baseline	51.70	2.15	–	$p < 0.01$	$p < 0.003$	NS	NS
nephSAP	64.20	2.85	$p < 0.01$	–	NS	$p < 0.001$	$p < 0.001$
UpToDate	65.80	2.10	$p < 0.003$	NS	–	$p < 0.001$	$p < 0.001$
Bible	47.50	3.44	NS	$p < 0.001$	$p < 0.001$	–	NS

Continued on next page

Table 8 – continued from previous page

Source	Mean (%)	SEM	vs Baseline	vs nephSAP	vs UpToDate	vs Bible	vs Random
Random	47.50	0.83	NS	$p < 0.001$	$p < 0.001$	NS	–
				Infection			
Baseline	50.80	0.83	–	$p < 0.05$	$p < 0.05$	NS	$p < 0.004$
nephSAP	59.20	2.10	$p < 0.05$	–	$p < 0.001$	NS	$p < 0.001$
UpToDate	43.30	1.36	$p < 0.05$	$p < 0.001$	–	$p < 0.001$	NS
Bible	57.50	1.60	NS	NS	$p < 0.001$	–	$p < 0.001$
Random	40.00	3.04	$p < 0.004$	$p < 0.001$	NS	$p < 0.001$	–

D Quantifying LLM Outputs

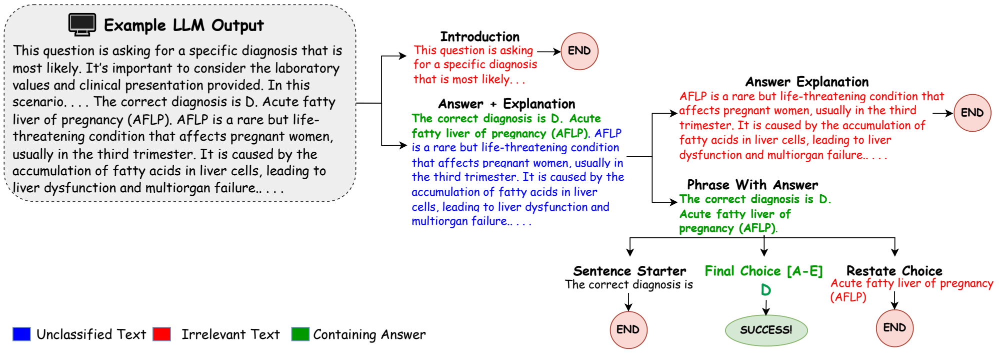

Example LLM Output
This question is asking for a specific diagnosis that is most likely. It's important to consider the laboratory values and clinical presentation provided. In this scenario. . . . The correct diagnosis is D. Acute fatty liver of pregnancy (AFLP). AFLP is a rare but life-threatening condition that affects pregnant women, usually in the third trimester. It is caused by the accumulation of fatty acids in liver cells, leading to liver dysfunction and multiorgan failure.. . . .
Introduction
This question is asking for a specific diagnosis that is most likely. . .
END
Answer + Explanation
The correct diagnosis is D. Acute fatty liver of pregnancy (AFLP). AFLP is a rare but life-threatening condition that affects pregnant women, usually in the third trimester. It is caused by the accumulation of fatty acids in liver cells, leading to liver dysfunction and multiorgan failure.. . . .
Answer Explanation
AFLP is a rare but life-threatening condition that affects pregnant women, usually in the third trimester. It is caused by the accumulation of fatty acids in liver cells, leading to liver dysfunction and multiorgan failure.. . . .
END
Phrase With Answer
The correct diagnosis is D. Acute fatty liver of pregnancy (AFLP).
Sentence Starter
The correct diagnosis is
END
Final Choice [A-E]
D
SUCCESS!
Restate Choice
Acute fatty liver of pregnancy (AFLP)
END
🟦 Unclassified Text 🟥 Irrelevant Text 🟩 Containing Answer

Figure 3: Example of possible parse tree to automatically extract answer choice from the LLM output. After pattern matching of the introductory phrase and explanatory phrase, the automated script can easily output which answer choice is chosen A-E.

To evaluate the performance of the accuracy of each LLM in answering the questions, we utilized regular expressions to match patterns in the generated outputs and extracted the output answer, and then compared that to the correct answers for each question. Regular expressions enable text processing functions such as validating inputs, extracting data, manipulating strings, and searching/replacing content. We utilized regexes because by utilizing special syntax elements, and complex match patterns we were able to define the patterns we were looking for to compare the outputs with the correct answers. This provided more flexibility than literal text matching alone.

D.0.1 Regex Pattern Matching

We used regular expressions to provide a concise and flexible method, while modifying many different variations of similar patterns for pattern matching to ensure the correct validation of the large dataset of questions being evaluated. However, due to the variability of model text generation in answering questions slightly differently for each question, a large amount of regular expressions were used to ensure accuracy. One example of a regular expression or parse tree to extract the answer from the LLM is shown in Figure 3, where the regex performs pattern matching on the model outputs to correctly detect the answer chosen automatically. By benchmarking against regexes for multiple types of expected patterns, we thoroughly evaluated the different LLM performances.