Comparative Analysis of Retrieval-Augmented Generation Performance in a Clinical Nutrition Question-Answering System

Muhammad Yasin Hakim[1], Tirana Noor Fatyanosa[1]*, Mahendra Data[1], and Ummu Ditya Erliana[2]

[1] Faculty of Computer Science, Universitas Brawijaya, Malang 65145, Indonesia
yasinh20@student.ub.ac.id, fatyanosa@ub.ac.id, mahendra.data@ub.ac.id
[2] Faculty of Health Sciences, Universitas Brawijaya, Malang 65145, Indonesia
Ummuditya.fk@ub.ac.id

Abstract. The rapidly expanding body of biomedical information poses a challenge for healthcare professionals in accessing accurate answers, particularly given the limited availability of Indonesian-language literature. The use of Large Language Models (LLMs) for biomedical question-answering systems also faces the issue of hallucination, which poses serious risks in clinical settings. To address this, the present study designs, implements, and evaluates a Retrieval-Augmented Generation (RAG) pipeline for a question-answering system focused on clinical nutrition, using an Indonesian-language dataset. The pipeline consists of five main components: a query embedder, a document retriever, a prompt builder, a generator based on Llama 3.2 3B and GPT-OSS, and an answer builder. Evaluation was conducted using BERTScore and RAGAS across four experiments comparing the two models with and without RAG. The results show that RAG consistently improves answer quality. Llama 3.2 3B with RAG achieved the highest BERT F1 score (0.5703), with a 5.8% improvement, while GPT-OSS with RAG reached 0.5658 with a 10.3% improvement. The RAGAS evaluation revealed a trade-off between faithfulness and answer relevance: GPT-OSS excelled in faithfulness (0.6881) but scored low on answer relevance (0.1595), while Llama 3.2 3B demonstrated a better balance. These findings confirm that small-scale open-source models with RAG can serve as an efficient alternative for Indonesian-language clinical question-answering systems.

Keywords: Retrieval-Augmented Generation · Question and Answer System · Clinical Nutrition · BERTScore · RAGAS · Large Language Model · Indonesian Language

1 Introduction

Data and literature in the biomedical field have expanded rapidly in recent decades. However, this growth poses challenges for healthcare professionals—particularly physicians, nutritionists, and health researchers—in quickly and

* Corresponding author.

© The Author(s) 2026
M. Murtadha et al. (eds.), *Proceedings of the 2026 11th International Conference on Sustainable Information Engineering and Technology (SIET 2026)*, Atlantis Highlights in Computer Sciences 26,
https://doi.org/10.2991/978-94-6239-828-3_11

accurately accessing valid and relevant information. The increasing volume of information makes it more difficult to separate irrelevant or invalid information from genuinely needed knowledge, thereby raising the risk of errors in clinical decision-making [1]. This challenge is further exacerbated by the limited availability of Indonesian-language literature in the biomedical field, especially clinical nutrition, which makes it difficult for local healthcare professionals to obtain references that align with their practice context.

The use of Large Language Models (LLMs) in building question-answering (QA) systems in the biomedical domain has become one effort to address this challenge. However, despite producing convincing-sounding answers, LLMs often experience *hallucination*—generating non-factual information. In a clinical context, even minor hallucinations can have serious implications for patient safety. One documented case involved the Whisper transcription system used to record patient complaints in hospitals, which produced inaccurate information [2]. This issue demonstrates that using LLMs without verification mechanisms against trusted knowledge sources is insufficient for medical question-answering systems.

Retrieval-Augmented Generation (RAG) has emerged as a promising approach to address the limitations of standalone LLMs. RAG combines a retrieval component that extracts relevant documents from external knowledge bases with an LLM-based generation component, resulting in more factual and accountable answers [3]. Several studies have demonstrated the effectiveness of RAG in the biomedical domain. Stuhlmann et al. [4] developed a RAG system on 24 million PubMed documents with a combination of BM25 and MedCPT reranking, achieving accuracy and recall of approximately 0.90. Sohn et al. [5] introduced Rationale-Guided RAG, which improved accuracy by 6.1% and outperformed other RAG baselines by up to 5.6%. Furthermore, Woo et al. [6] demonstrated that applying RAG to LLMs can improve accuracy by up to 39.7% compared to non-customized LLMs. However, most of these studies were conducted on English-language datasets, while exploration on Indonesian-language datasets remains very limited.

Based on these issues, this study aims to design, implement, and evaluate a RAG pipeline for a question-answering system in the field of clinical nutrition using an Indonesian-language dataset. The pipeline is constructed from five main components: a query embedder, a document retriever based on cosine similarity, a prompt builder, a generator based on Llama 3.2 3B and GPT-OSS models, and an answer builder. Performance evaluation is conducted using BERTScore to measure semantic similarity between generated answers and ground truth, as well as RAGAS (Retrieval Augmented Generation Assessment) to assess retrieval and generation quality comprehensively through four dimensions: faithfulness, answer relevancy, context precision, and context recall. Four experimental configurations are conducted to compare both models with and without RAG to examine the extent to which RAG can improve accuracy and reliability of answers in the Indonesian-language clinical nutrition domain.

2 Theoretical Background

2.1 Question-Answering Systems and Large Language Models

A question-answering (QA) system is one of the primary tasks in natural language processing that aims to provide specific answers to natural language questions based on relevant documents or knowledge bases [7]. Unlike traditional search, which returns a list of documents, QA systems provide direct answers.

Large Language Models (LLMs) are transformer-based language models with tens to hundreds of billions of parameters trained on very large-scale text data, such as PaLM, LLaMA, and GPT-4 [8]. Although LLMs are capable of performing various NLP tasks, their ability to access and process factual information accurately remains limited, particularly for tasks requiring specific and up-to-date knowledge.

2.2 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) is an approach that combines the parametric memory of LLMs with non-parametric memory in the form of external knowledge sources to produce more factual answers [3]. The RAG architecture consists of three main components: *Retrieval*, which searches for relevant documents from external sources using dense vector search; *Augmentation*, which processes and extracts the most relevant information to be combined with the query; and *Generation*, which combines this information with the pre-trained LLM knowledge to produce coherent responses [9]. Compared to standalone LLMs, RAG has been proven to significantly improve accuracy, as demonstrated by Woo et al. [6] with improvements of up to 39.7%.

2.3 Cosine Similarity

Cosine Similarity is a metric that measures the degree of similarity between two vectors in a multidimensional space by calculating the cosine of the angle between them [10]. Its value ranges from -1 to 1, where 1 indicates identical direction, 0 indicates no similarity, and -1 indicates opposite direction. In the context of RAG, cosine similarity is used during the retrieval stage to match query embedding vectors with document vectors so that the most semantically relevant documents can be retrieved. The cosine similarity equation is defined in Eq. (1).

$$\cos\theta = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \cdot \|\mathbf{B}\|} \tag{1}$$

2.4 Evaluation Methods

The evaluation in this study uses two main metrics. First, BERTScore [11], which assesses semantic similarity between the generated text and the reference text

Table 1. Distribution of Question-Answer Pairs by Category.

Question Category	Count
Nutritional status screening (*Skrining Gizi*)	81
Nutritional status (*Status Gizi*)	96
Abnormal biochemical data (*Data Biokimia*)	77
Nutritional diagnosis in PES (*Diagnosis Gizi*)	78
Diet objectives (*Tujuan Diet*)	78
Diet principles (*Prinsip Diet*)	77
Diet requirements (*Syarat Diet*)	77
Dietary guidelines (*Pedoman Diet*)	8
Total	**572**

using contextual embeddings from the BERT model, producing precision, recall, and F1 scores. Second, RAGAS [12], which provides four metrics: *faithfulness* for factual consistency with the context, *answer relevancy* for answer relevance to the question, *context precision* for the proportion of relevant context, and *context recall* for the completeness of context in covering the required information.

3 Research Methodology

3.1 Dataset

The dataset used in this study is a secondary Indonesian-language dataset obtained from the research of Ummu Ditya Erliana, S.Gz, MS, Ph.D, RDN, CLC from the Faculty of Health Sciences, Universitas Brawijaya, collected between 2021 and 2023. The dataset contains 81 pediatric patient cases encompassing report titles, anthropometric assessments (body weight, height, age, gender), medical diagnoses, and eight types of clinical questions with their answers for each patient. These questions cover nutritional status screening methods, patient nutritional status, abnormal biochemical data, nutritional diagnosis in PES format, diet objectives, diet principles, diet requirements, and dietary guidelines. Since not all patients have complete answers for every question category, the dataset yields a total of 572 valid question-answer pairs rather than the theoretical maximum of 648 (81×8). Table 1 presents the distribution across question categories.

The distribution is relatively balanced across the first seven categories (77–96 pairs each), while dietary guidelines is underrepresented with only 8 pairs, as most patients did not have this question answered in the original clinical records. No train–test split was applied, as no model fine-tuning was performed; the entire dataset was used exclusively for evaluation. Although the dataset size is modest compared to large-scale English-language biomedical benchmarks, the 572 QA pairs spanning 81 distinct clinical cases provide sufficient coverage to observe consistent performance patterns across all experimental configurations and to draw meaningful comparative conclusions between the RAG and non-RAG conditions.

Before being used as input, the dataset was converted into CSV format with three main columns. The *context* column contains a combination of anthropometric assessments, medical diagnoses, and all question-answer pairs for each data row. The *question* column contains individual questions, and the *answer* column contains specific answers serving as ground truth for evaluation.

3.2 RAG Pipeline Design

The design of the RAG-based question-answering system in this study consists of four main phases as illustrated in Fig. 1. The first phase is *preprocessing*, where the CSV dataset is converted into PDF documents based on grouping by unique context. Each context group produces a single PDF document containing all clinical information for that patient.

The second phase is *indexing*. The PDF documents are loaded and split into chunks using recursive chunking with parameters of chunk size 500 characters and overlap 50 characters. The chunk size of 500 was chosen to balance between preserving sufficient clinical context within a single chunk (e.g., a complete anthropometric assessment or a single question-answer pair) and maintaining granularity for precise retrieval; larger chunks risk diluting relevance, while smaller chunks risk fragmenting clinical information across multiple segments. The overlap of 50 characters (10% of the chunk size) ensures continuity at chunk boundaries. This process produced approximately 800 chunks across all 81 documents, which were stored in Qdrant as 800 collection points. Each chunk is converted into a 768-dimensional embedding vector using the embeddinggemma model[3] through Ollama, and stored in the Qdrant vector store with Hierarchical Navigable Small World (HNSW) indexing and cosine similarity, along with its text payload using upsert operations with Universally Unique Identifier (UUID) identifiers.

The third phase is *query*. The user's query is embedded using the same model as the indexing process, then the query vector is compared with all vectors in Qdrant using cosine similarity to retrieve the top-k most relevant chunks ($k = 3$). The value of $k = 3$ was selected to provide sufficient context diversity while staying within the context window constraints of the smaller models tested; higher values of k would increase the risk of exceeding the 800-character context limit and introducing irrelevant information that could degrade generation quality. The retrieved chunks are combined into a single context, then assembled together with the question into an Indonesian-language prompt template, and sent to the LLM to generate an answer.

The fourth phase is *evaluation*. The system is tested through four experimental configurations: Llama 3.2 3B with RAG, Llama 3.2 3B without RAG, GPT-OSS with RAG, and GPT-OSS without RAG. Each experiment is evaluated using BERTScore to measure semantic similarity of answers with references, as well as RAGAS to comprehensively assess retrieval and generation quality.

[3] https://ollama.com/library/embeddinggemma

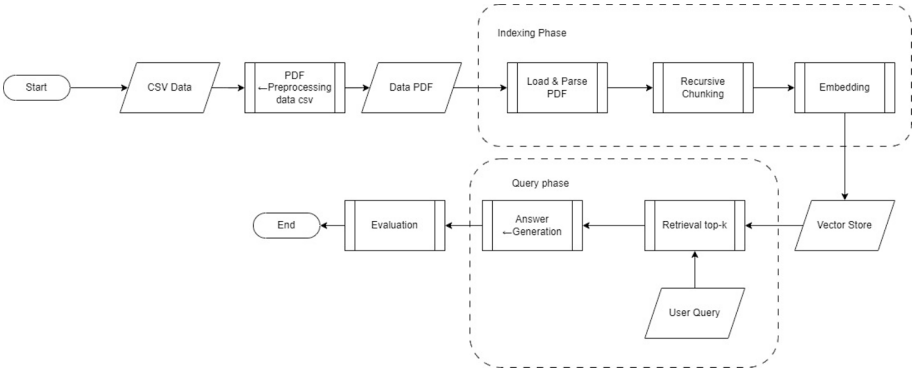


Fig. 1. Algorithm Flow Diagram of the RAG QA System.

3.3 Pipeline Components

The designed RAG pipeline consists of five main components. The *Query Embedder* uses the embeddinggemma model to transform questions into numerical vector representations. The *Document Retriever* searches for the most relevant documents based on cosine similarity calculations between the query vector and document vectors stored in Qdrant. The *Prompt Builder* combines the retrieval context with the query into a prompt template that instructs the model to answer strictly based on the provided context. The *Generator* sends the prompt to the LLM through the Ollama API to generate answers, with two models being tested: Llama 3.2 3B and GPT-OSS. Finally, the *Answer Builder* integrates the output from the retriever and generator into a final answer object used for evaluation.

3.4 Experimental Environment

The experiments were conducted at the Intelligent Systems Laboratory, Faculty of Computer Science, Universitas Brawijaya, using an NVIDIA L40 GPU (1×48 GB). All models were run locally through Ollama: Llama 3.2 3B was quantized using the Q4_K_M scheme, while GPT-OSS (∼20.9B parameters) used MXFP4 quantization. Qdrant was used as the vector store in persistent local storage mode. The retrieval parameters used include top-$k = 3$ with a maximum limit of 800 characters per context to avoid too long contexts. The RAGAS evaluation used Llama 3.1 8B as the internal LLM judge to compute faithfulness, answer relevancy, context precision, and context recall.

Table 2. Average BERTScore Results.

Experiment	BERT Prec.	BERT Rec.	BERT F1
Llama + RAG	0.5540	**0.6094**	**0.5703**
Llama without RAG	0.4898	**0.6094**	0.5369
GPT-OSS + RAG	**0.6226**	0.5298	0.5658
GPT-OSS without RAG	0.4324	0.5995	0.4973

4 Results and Discussion

4.1 Qualitative Analysis of Model Answers

Qualitative analysis was conducted on two categories of questions that represent the distinctive characteristics of each experiment. For questions with specific patient-based answers, such as determining nutritional status based on Z-score values, GPT-OSS + RAG demonstrated superiority by directly quoting Z-score values from the retrieved context, achieving a BERT F1 of 0.9052. Conversely, Llama 3.2 3B + RAG, despite successfully finding a relevant context, misinterpreted the Z-score value of +2.7 SD by concluding "underweight nutritional status," when in fact this value indicates overweight risk. Both models without RAG produced independent calculations or generative analyses without patient-specific values.

For questions with single document-based answers, such as identifying abnormal biochemical parameters, GPT-OSS + RAG again excelled with concise and accurate answers, achieving a BERT F1 score of 0.8872, while Llama 3.2 3B + RAG responded with "Not found in context" despite the relevant context being available. These findings indicate that Llama 3.2 3B's instruction-following capability in reading numerical clinical data still has limitations, even when the correct context has been retrieved.

Analysis of the experimental results reveals three distinct failure categories. The first category is *reasoning errors*, where the model retrieves the correct context but misinterprets numerical values, as exemplified by Llama 3.2 3B's misclassification of a Z-score of +2.7 SD as "underweight" when it actually indicates overweight risk. The second category is *retrieval failures*, where the per-patient document segmentation strategy may cause retrieved chunks to contain information from the correct patient but not the specific clinical data needed for the question, leading to incomplete or irrelevant context. The third category is *generation refusals*, where the model responds with "Not found in context" despite relevant information being present in the retrieved chunks, indicating an overly conservative interpretation of the prompt instruction to answer strictly based on context.

4.2 Answer Quality Evaluation with BERTScore

BERTScore evaluation was conducted across all four experiments to measure semantic similarity between the generated answers and the reference answers. Table 2 summarizes the average BERTScore for each experiment.

Table 3. Average RAGAS Results.

Experiment	Ctx Prec.	Ctx Rec.	Faithful.	Ans. Rel.
Llama + RAG	0.5552	0.6293	0.5926	**0.4749**
GPT-OSS + RAG	**0.5667**	**0.6378**	**0.6881**	0.1595

Based on Table 2, Llama 3.2 3B + RAG obtained the highest BERT F1 with a value of 0.5703, followed by GPT-OSS + RAG (0.5658). This pattern indicates that RAG consistently improves the performance of both models. GPT-OSS + RAG excels in precision (0.6226), meaning the tokens in its answers have high semantic alignment with the reference. Conversely, experiments without RAG tend to produce higher recall than precision, indicating longer but semantically less precise answers.

An interesting finding is the identical recall value for Llama 3.2 3B between RAG and non-RAG conditions (0.6094). This suggests that RAG does not change the semantic coverage of Llama's answers but rather significantly improves precision, making the answers more focused and precise.

4.3 Retrieval and Generation Quality Evaluation with RAGAS

The RAGAS evaluation was conducted specifically on the two experiments that use RAG. Table 3 presents the average results for all four RAGAS metrics.

In terms of context quality, both experiments show relatively comparable values for context precision and context recall, indicating that the retrieval component has consistent performance. The performance difference between experiments is more influenced by the generation capability of each model.

On the faithfulness dimension, GPT-OSS + RAG obtained a higher value (0.6881) compared to Llama 3.2 3B + RAG (0.5926), indicating that GPT-OSS is more consistent in restricting its answers to information from the context. However, the most striking finding is the large gap in answer relevancy: GPT-OSS + RAG achieved only 0.1595 compared to Llama 3.2 3B + RAG which reached 0.4749. Further investigation revealed that 37 out of 572 samples (6.5%) in the GPT-OSS + RAG experiment produced very short answers (4–21 characters), causing the question regeneration process by RAGAS for the answer relevancy metric evaluation to fail and produce NaN values. The RAGAS answer relevancy metric works by generating synthetic questions from the model's answer and measuring their cosine similarity with the original question; when the answer is too short, the internal question generation by the RAGAS LLM judge (Llama 3.1 8B) fails entirely, yielding NaN rather than a low score. This represents a known limitation of the RAGAS evaluation framework when applied to models that tend to produce terse, extractive answers—a behavior characteristic of GPT-OSS's high faithfulness but low verbosity response pattern. The low answer relevancy value for GPT-OSS should therefore be interpreted as partially an artifact of the metric's sensitivity to answer length rather than a purely qualitative deficiency in relevance.

These findings have practical implications for the deployment of such systems in real clinical environments. The trade-off between faithfulness and answer relevancy suggests that model selection should be guided by the specific clinical use case: GPT-OSS's high faithfulness makes it suitable for scenarios requiring strict factual adherence (e.g., retrieving specific lab values), while Llama 3.2 3B's balanced performance may be preferable for generating comprehensive dietary recommendations that require elaboration beyond the retrieved context.

4.4 Comprehensive Comparison and Implications

In general, the application of RAG consistently improves the answer quality for both models. The improvement in BERT F1 is more significant for GPT-OSS (from 0.4973 to 0.5658, an increase of 10.3%) compared to Llama 3.2 3B (from 0.5369 to 0.5703, an increase of 5.8%). These findings confirm the effectiveness of RAG with augmentation strategies in clinical domains that require high factual accuracy.

The comparison between models shows that Llama 3.2 3B + RAG slightly outperforms GPT-OSS + RAG on BERT F1, despite GPT-OSS having advantages in precision and faithfulness. This has practical implications: small-scale open-source models such as Llama 3.2 3B with RAG can serve as a computationally efficient alternative without significantly sacrificing answer quality. However, the RAGAS results remind us of the trade-off between faithfulness and answer relevancy.

5 Conclusions and Future Work

Based on the research results, several conclusions can be drawn. First, the RAG pipeline for a question-answering system in the field of clinical nutrition was successfully designed and implemented using five main components: a query embedder (embeddinggemma), a document retriever based on cosine similarity with Qdrant, a prompt builder, a generator based on Llama 3.2 3B and GPT-OSS through Ollama, and an answer builder. All components are integrated in a RAG pipeline designed and running on an Indonesian-language clinical nutrition dataset.

Second, evaluation using BERTScore shows that RAG consistently improves answer quality. Llama 3.2 3B + RAG achieved the highest BERT F1 of 0.5703 (5.8% improvement), while GPT-OSS + RAG reached 0.5658 (10.3% improvement). GPT-OSS + RAG excels in the precision metric (0.6226), while Llama 3.2 3B + RAG demonstrates a better balance between precision and recall.

Third, RAGAS evaluation revealed different characteristics of the two models. GPT-OSS + RAG excels in faithfulness (0.6881), context precision (0.5667), and context recall (0.6378), but has low answer relevancy (0.1595) due to its tendency to produce very short answers. Llama 3.2 3B + RAG shows a better balance across all metrics with answer relevancy of 0.4749. This finding is attributed to the trade-off between context adherence and answer relevance to the user's question.

5.1 Limitations

This study has several limitations that should be acknowledged. First, the dataset is limited to 81 pediatric patient cases (572 QA pairs) from a single institution, which may not fully capture the diversity of clinical nutrition scenarios encountered in broader practice. The underrepresentation of the dietary guidelines category (only 8 pairs) further limits the evaluation coverage for that question type. Second, the per-patient document segmentation strategy groups all clinical information for a single patient into one PDF document; this design choice may cause retrieved chunks to contain adjacent but irrelevant clinical data, potentially degrading retrieval precision for questions targeting specific data points. Third, the evaluation is limited to two LLMs, both running under quantization constraints (Q4_K_M for Llama 3.2 3B and MXFP4 for GPT-OSS) due to available VRAM, and does not include alternative retrieval methods such as BM25 or hybrid sparse-dense retrieval. Fourth, the reported BERTScore improvements (5.8% and 10.3%) are not accompanied by statistical significance testing, and the modest magnitude of these improvements warrants cautious interpretation. Fifth, the RAGAS evaluation framework exhibits sensitivity to very short generated answers, as demonstrated by the 37 NaN values in GPT-OSS's answer relevancy scores, which limits the reliability of cross-model comparisons on this metric. Finally, the source code for this study is not publicly released due to dataset restrictions, which limits external reproducibility.

5.2 Future Work

Future research directions are proposed in direct response to the identified limitations. To address the document segmentation limitation, future work should explore alternative document granularity strategies, including per-topic segmentation and more adaptive chunking techniques such as semantic or agentic chunking, which may improve the precision of retrieved context. To mitigate the retrieval method constraint, incorporating sparse retrieval baselines (e.g., BM25), hybrid sparse-dense retrieval, or reranking techniques would provide a more comprehensive assessment of the pipeline. The embedding limitation could be addressed by using embedding models specifically trained on Indonesian biomedical data to improve semantic relevance during the retrieval stage. To strengthen the statistical validity of the findings, future experiments should include confidence intervals or significance tests to confirm that the observed improvements are robust. The dataset scope limitation can be addressed through expansion with more clinical cases from multiple institutions and additional question variations, including adult patient cases, to improve system generalization. Finally, testing with larger LLM models or models fine-tuned on the biomedical domain would help determine whether the observed failure patterns (reasoning errors, generation refusals) are attributable to model capacity or to pipeline design choices.

References

1. Jin, Q., Yuan, Z., Xiong, G., Yu, Q., Ying, H., Tan, C., Chen, M., Huang, S., Liu, X., Yu, S.: Biomedical question answering: A survey of approaches and challenges. ACM Computing Surveys **55**(2), 1–36 (2023). https://doi.org/10.1145/3490238
2. Koenecke, A., Choi, A.S.G., Mei, K.X., Schellmann, H., Sloane, M.: Careless Whisper: Speech-to-text hallucination harms. In: The 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT '24), pp. 1672–1681. ACM, Rio de Janeiro (2024). https://doi.org/10.1145/3630106.3658996
3. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Advances in Neural Information Processing Systems, vol. 33, pp. 9459–9474 (2020)
4. Stuhlmann, L., Saxer, M.A., Fürst, J.: Efficient and reproducible biomedical question answering using retrieval augmented generation. arXiv preprint arXiv:2505.07917 (2025). https://doi.org/10.48550/arXiv.2505.07917
5. Sohn, J., Park, Y., Yoon, C., Park, S., Hwang, H., Sung, M., Kim, H., Kang, J.: Rationale-guided retrieval augmented generation for medical question answering. arXiv preprint arXiv:2411.00300 (2024). https://doi.org/10.48550/arXiv.2411.00300
6. Woo, J.J., Yang, A.J., Olsen, R.J., Hasan, S.S., Nawabi, D.H., Nwachukwu, B.U., Williams, R.J., Ramkumar, P.N.: Custom large language models improve accuracy: Comparing retrieval augmented generation and artificial intelligence agents to noncustom models for evidence-based medicine. Arthroscopy **41**(3), 565–573.e6 (2025). https://doi.org/10.1016/j.arthro.2024.10.042
7. Caballero, M.: A brief survey of question answering systems. International Journal of Artificial Intelligence & Applications **12**(5), 01–07 (2021). https://doi.org/10.5121/ijaia.2021.12501
8. Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., Gao, J.: Large language models: A survey. arXiv preprint arXiv:2402.06196 (2025). https://doi.org/10.48550/arXiv.2402.06196
9. Singh, A., Ehtesham, A., Kumar, S., Khoei, T.T.: Agentic retrieval-augmented generation: A survey on agentic RAG. arXiv preprint arXiv:2501.09136 (2025). https://doi.org/10.48550/arXiv.2501.09136
10. Han, J., Pei, J., Tong, H.: Data Mining: Concepts and Techniques. 4th edn. Morgan Kaufmann, Cambridge (2023). https://doi.org/10.1016/C2013-0-18660-6
11. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: BERTScore: Evaluating text generation with BERT. arXiv preprint arXiv:1904.09675 (2020). https://doi.org/10.48550/arXiv.1904.09675
12. Es, S., James, J., Espinosa-Anke, L., Schockaert, S.: RAGAS: Automated evaluation of retrieval augmented generation. arXiv preprint arXiv:2309.15217 (2025). https://doi.org/10.48550/arXiv.2309.15217

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

