Vectorless Retrieval-Augmented Generation: A BM25-Based Alternative to Embedding-Driven RAG Systems

1st Alex Rohith I
dept. of Artificial Intelligence and Data Science
St. Joseph's Institute of Technology
Chennai, India
rohithalex06@gmail.com

2nd Abilash S Nath
dept. of Artificial Intelligence and Data Science
St. Joseph's Institute of Technology
Chennai, India
abilashsample@gmail.com

3rd Saranya S
dept. of Artificial Intelligence and Data Science
St. Joseph's Institute of Technology
Chennai, India
ssaranrahul@gmail.com

***Abstract*—Retrieval Augmented Generation (RAG) augments large language models using external knowledge at the inference stage to increase factual accuracy and reduce hallucinations. Most existing RAG models employ dense vectors for information retrieval, which facilitates semantic matching at the expense of increased computation costs, latency, and lack of transparency. In our paper, we propose a novel vectorless RAG system which replaces embedding-based retrieval with conventional information retrieval methods such as full-text matching, BM25 ranking, and keyword matching. We conduct experiments comparing vector-based and vectorless models under identical conditions in terms of the underlying document pool, pre-processing steps, and language model. The comparison is carried out using several performance criteria such as latency, faithfulness, diversity, keyword alignment, and semantic relevancy. Experiment results show that the proposed vectorless model achieves competitive performance while providing advantages in terms of efficiency, interpretability, and grounding.However, it exhibits limitations in handling semantically complex queries. These findings suggest that vectorless retrieval is well-suited for precision-driven and resource-constrained applications, and that hybrid approaches may further enhance performance.**

***Index Terms*—Retrieval-Augmented Generation (RAG), Vectorless Retrieval, BM25, FAISS, Semantic Search, Lexical Retrieval**

I. Introduction

Retrieval-Augmented Generation (RAG) has been identified as an effective technique for boosting the performance of Large Language Models (LLMs) through the incorporation of external knowledge during inference [17]. By retrieving relevant documents and utilizing them in conjunction with the context of the model, RAG has proven effective in boosting the factual accuracy and reliability of language models [17]. As such, RAG has become an integral aspect of applications such as question answering and search [2].

Most of the existing techniques for implementing RAG have been based on vector-based techniques that allow for the representation of queries and documents as vectors in high-dimensional space [2], [8]. By utilizing vector databases such as FAISS, language models can be enabled to search for relevant information based on contextual relevance despite the lack of exact keyword matches [7], [11]. Although this technique has proven effective in boosting the performance of language models in dealing with complex and paraphrased queries, there are many challenges associated with its implementation, including increased latency and lack of interpretability [6], [15].

On the other hand, classical information retrieval approaches such as BM25 and full-text search provide efficient and clear alternatives based on lexical matching and statistical scoring [5], [6]. However, despite the demonstrated success of these approaches, RAG systems have largely ignored their use, with the majority of systems continuing to rely on embedding-based retrieval [6]. This highlights a critical gap in current research, where effective and interpretable lexical methods remain unexplored within modern RAG pipelines.

Thus, the central research question addressed in this paper is whether vectorless information retrieval approaches can provide a viable alternative to embedding-based retrieval methods in RAG systems. To address this question, this paper proposes a vectorless RAG system that relies on PostgreSQL full-text search, BM25 scoring, and keyword overlap as lexical information retrieval mechanisms. Unlike classical RAG approaches that depend on dense embedding generation, the proposed system eliminates embeddings entirely, thereby improving computational efficiency and enhancing interpretability. In order to ensure a fair and controlled evaluation, both vector-based and vectorless retrieval approaches are implemented

within a unified framework, allowing them to share the same document corpus, preprocessing steps, and language model.

The contributions of this work are as follows. First, a fully embedding-independent Retrieval-Augmented Generation architecture is introduced, relying on classical information retrieval techniques such as full-text search, BM25 scoring, and keyword overlap for document ranking. Second, a unified comparison framework is developed to enable objective evaluation of vector-based and vectorless approaches under equivalent conditions. Third, a hybrid lexical ranking approach is proposed to improve the precision of lexical retrieval methods. Lastly, the experiments are conducted using various measures, such as latency, faithfulness, diversity, keyword similarity, and semantic relevancy. In the end, the experiment provides significant insight regarding the balance between lexical accuracy and semantic comprehension in retrieval models.

II. RELATED WORK

A. Limitations of Vector-Based Semantic Retrieval

Most of the RAG frameworks are reliant on vector databases and dense embedding techniques in order to compute the semantic similarity between the user query and the document snippets [2], [7]. While useful in generalized applications, recent studies have shown significant shortcomings in terms of both structure and performance pertaining to the embedding-only retrieval paradigm [1], [3]. The embedding models typically work as rigid "black box" mechanisms that poorly comprehend complex semantic contexts, thus leading to information retrieval that may be semantically wrong but syntactically similar [3]. In addition, in the case of specialized domains, the generic embeddings generated using large language models often do not encompass the intricacies specific to the domain; it has been observed through experimental results that the retrieval relationships are largely based on token-level similarity rather than semantic correlation [4]. Apart from accuracy issues, the vector-based approach incurs significant computational overhead, faces high dimensionality problems, and requires considerable storage space.

B. Vectorless and Embedding-Free RAG Architectures

In order to address the problems of computational overhead and difficulties with domain adaptation when using vector databases, several recent solutions suggest "vectorless" or "embedding-free" approaches to retrieval tasks [3], [4], [10]. Specifically, the Embedding-Free RAG system replaces the embedding step altogether by asking a language model to generate exact quotation references from the document itself and then uses fuzzy matching techniques, like Levenshtein distance, to align those references to the source index positions [3]. Another way is presented in the Prompt-RAG framework, which skips the vector embeddings completely, feeding a document's Table of Contents to a prompt, so that a language model can find the right section headings independently [4]. Finally, another solution called ELITE involves language model-assisted iterative text searching with classic word overlap algorithms to achieve successful information retrieval, proving that vectorless approaches can significantly speed up the response time and require less storage space while maintaining high retrieval performance [1].

C. BM25 and Classical Lexical Retrieval

In situations where there is a lack of dense embeddings, the classical information retrieval approaches such as BM25 continue to perform very well as far as the retrieval function of the RAG approach is concerned [5], [6]. Unlike the conventional TF-IDF technique, BM25 overcomes some of the challenges associated with document length variation by incorporating document length normalizations and term frequency saturation, leading to an equilibrium measure of lexical similarity [5]. Evaluations done using various BM25 models such as BM25L, BM25+, and BM25-adpt have shown that when appropriately fine-tuned through such mechanisms as stemming and relevance feedback, these classical ranking functions are quite precise [5].

D. Systematic Evaluation of RAG Pipelines

However, with the increasing variety of RAG models, recent literature calls for the development of standardized, multi-faceted evaluation criteria that facilitate comparative studies among different retrieval methods [17]. Methodologies executing parallel pipelines over a shared document corpus and a common LLM generator have been successfully used to isolate the impact of the retrieval mechanism itself [14], [15]. For example, recent comparative studies have benchmarked different retrieval strategies against metrics such as faithfulness, answer relevance, context relevance, factual correctness, and latency [7], [9]. Evaluations comparing native vector database retrieval against different indexing methods (such as HNSW vs. Inverted Files) have shown that the underlying retrieval algorithm directly impacts the relevance and latency of the final generated response [7]. Our proposed methodology builds upon these foundations by establishing a controlled, dual-pipeline environment to explicitly measure the trade-offs between lexical vectorless retrieval and embedding-based semantic retrieval [11], [12].

III. METHODOLOGY

The methodology adopted in this study for the development, deployment, and evaluation of vectorless and vector-based Retrieval-Augmented Generation (RAG) systems is presented in this section.Thus, the purpose is to design a systematic and well-structured environment that allows for a balanced comparison of lexical retrieval systems and embedding-based semantic retrieval systems [11], [17]. To achieve this goal, two pipelines are designed using an identical corpus of documents where the same preprocessing and language model are used to generate responses; therefore, the retrieval system becomes the only parameter under test [14], [15].

Moreover, the methodology determines the components of each pipeline regarding how documents are processed, the mechanism through which information is retrieved from a pool of documents, the technique used to compute scores,

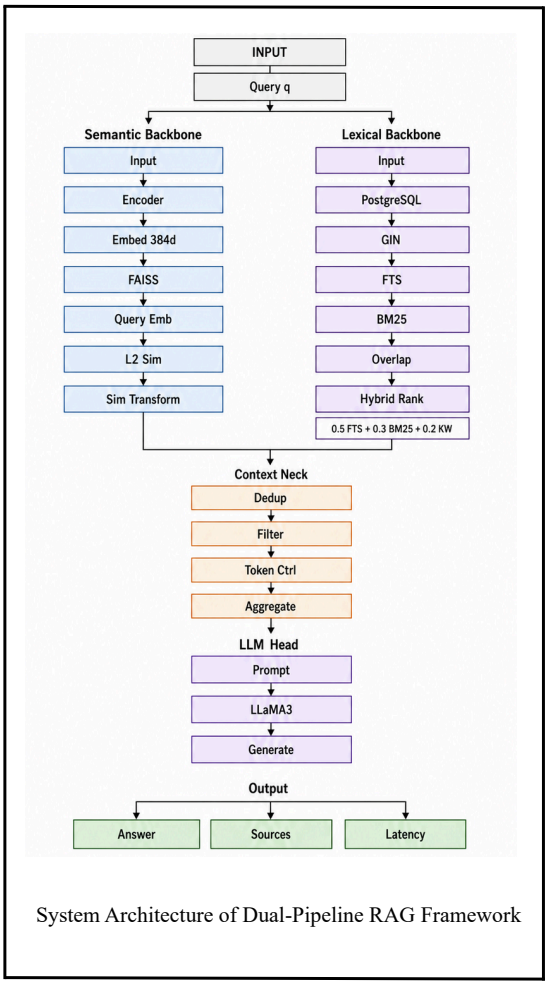
and context generation approaches [1]. A comparison environment will also be defined that allows for implementing both systems simultaneously and obtaining structured results [9], [15]. Finally, an assessment protocol is designed that allows measuring the efficiency of the systems in relation to measures like latency, fidelity, relevance, diversity, and the retrieval process itself [10]. This structured methodology ensures reproducibility and consistency [14], [16], while also enabling a detailed examination of the trade-offs between vectorless and vector-based RAG approaches in terms of cost, performance, and retrieval accuracy [15], [16].

A. Problem Formulation and System Overview

Retrieval-Augmented Generation (RAG) systems aim to enhance the response quality of large language models by incorporating external knowledge during inference [2], [17]. Given a user query, the system retrieves a set of relevant documents, which are then used as contextual input to generate an answer [1], [3]. This retrieval-and-generation paradigm improves factual accuracy and reduces hallucinations by grounding responses in external data sources [17].

In this study, we define the task as a comparative assessment of two distinct retrieval paradigms: (1) vector-based semantic retrieval and (2) vectorless lexical retrieval [7]. Both methods are implemented under identical conditions, including the use of the same document corpus, preprocessing and chunking strategies, and language model for answer generation [10]. This setup guarantees the fairness and control of the experiment, where the retrieval process is considered the main variable being tested [10].

The suggested model includes two independent pipelines of retrieval-augmented generation (RAG), one for each paradigm described above [14], [15]. Both pipelines follow the same design, consisting of stages such as query, retrieval, context formation, and response generation [14]. The entire pipeline is executed from one common API endpoint for processing one query independently through both models and making a comparison between the two paradigms [12], [15].



System Architecture of Dual-Pipeline RAG Framework

Fig. 1. System Architecture of the Dual-Pipeline RAG Framework

B. Retrieval Pipelines

In this study, two retrieval methods will be used: vector-based semantic retrieval method and vectorless lexical retrieval method. As illustrated by the diagram above, both methods use the same preprocessed document collection and have similar processes in their downstream stages, which include context creation and language model creation. In this case, while vector-based methods exploit dense vectors to represent semantic similarities, vectorless methods employ standard information retrieval methods that focus on lexical similarities. The following paragraphs analyze both types of methods.

1) Vector-Based Retrieval Pipeline: The vector retrieval pipeline makes use of the dense retrieval technique wherein both documents and queries are represented as dense vectors. Each document segment is converted into dense vectors through the sentence-transformers model ("all-MiniLM-L6-v2"). The process allows for capturing the semantic relationship between each of the document segments. To improve interpretability and stability of the scoring function, the raw distance metric is converted into a similarity score by applying an inverse scaling function. This transformation ensures that smaller distances correspond to higher similarity values. The vector-based approach enables robust semantic matching, allowing the system to retrieve contextually relevant documents even when there is minimal lexical overlap between the query and the document content. However, this comes at the cost of increased computational overhead and reduced transparency in retrieval scoring.

These embeddings are stored using a FAISS index (IndexFlatL2), which allows for efficient nearest neighbor search in high-dimensional space. During query execution, the query is encoded using the same sentence-transformers model, and the top-k nearest document embeddings are retrieved based on L2 distance.

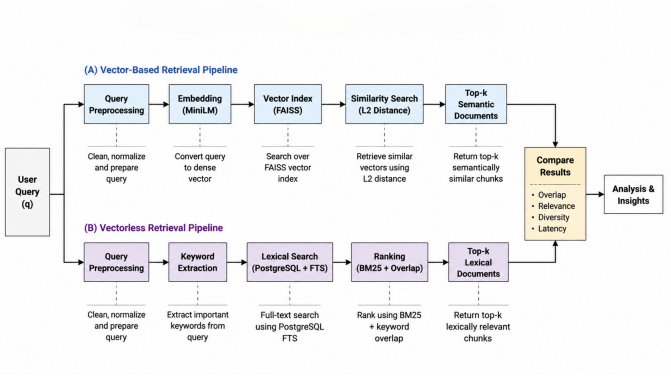


Fig. 2. Workflow diagram of the proposed dual-pipeline RAG system illustrating query processing, retrieval, and response generation stages.

2) Vectorless Retrieval Pipeline: In the vectorless retrieval pipeline, the generation of embeddings is not required, and classical information retrieval methods are used for efficient and interpretable document ranking. The process begins with document preprocessing, where the input documents are converted into clean text and then divided into semantically meaningful chunks of 300–500 tokens, with overlapping windows to ensure contextual continuity.These chunks are stored in a PostgreSQL database, where a full-text search (FTS) index is created using a GIN structure. Initially, the retrieval process uses PostgreSQL's FTS mechanism, which assigns a relevance score based on the lexical overlap between the query and the document content.

In order to further enhance the efficacy of the retrieval process, the BM25 algorithm is used for scoring purposes. In addition, a keyword overlap score is computed to measure the overlap ratio between the query and the document chunks.A weighted hybrid ranking approach is then used, combining these three scoring methods:

Final Score = 0.5 × FTS + 0.3 × BM25 + 0.2 × Keyword Overlap

This hybrid approach improves retrieval precision by balancing statistical relevance with direct lexical alignment. Unlike embedding-based methods, the vectorless pipeline offers greater interpretability, lower computational cost, and consistent latency, making it particularly suitable for resource-constrained and real-time applications.

C. Dual-Pipeline Retrieval Architecture and Workflow

Figure 2 below shows the architectural layout of the RAG system proposed, which includes two retrieval pipelines that run in parallel - vector-based semantic retrieval and vectorless lexical retrieval. The two pipelines process the same query as an input but have a shared preprocessing step to ensure a fair comparison between the two methods.

For the vector-based retrieval approach, the input query is first preprocessed and then passed through a MiniLM language model to produce a dense representation of the input. The vectorized representation is then compared to the document representations stored in a FAISS index using the L2 distance metric to retrieve the top-k document chunks that are semantically similar to the input query. On the other hand, the vectorless retrieval system does not employ vectorized representation but uses traditional information retrieval techniques. First, the query is preprocessed and the important keywords are extracted before being passed to PostgreSQL for full-text search. The retrieved documents are ranked using a combination of BM25 and keyword matching scores and the top-k lexically relevant documents are chosen.

This design ensures that the basic distinction between semantic and lexical approaches in retrieval is maintained while preserving the consistency of evaluation criteria.

IV. RESULTS AND DISCUSSION

The evaluation of the experiments focuses on comparing the performance of both vectorless and vector-based retrieval approaches using the RAG framework. The experiment was conducted through a consistent comparison framework, where both methods were tested in the same environment with a certain number of sample queries. Evaluation of the experiment was done through a multidimensional evaluation process. In this part of the discussion, we will focus on the results obtained from our experiment and further analyze the performance and limitations of both methods.

A. Experimental Results Overview

To evaluate the effectiveness of the proposed vectorless and vector-based RAG pipelines, a set of representative queries covering a wide range of object detection concepts and technical topics was used. These queries were designed to represent a variety of scenarios, from keyword-based to semantically complex queries, which would provide a comprehensive evaluation of the two proposed retrieval paradigms. The experimental results for the proposed systems were collected from multiple aspects, including latency, answer length, keyword overlap, faithfulness, diversity, and semantic relevance, among others. These experimental results were designed with a focus on different aspects of the proposed systems, from the efficiency of the proposed systems in terms of latency and answer length, through the accuracy of the proposed systems in terms of keyword overlap, faithfulness, diversity, and relevance, among others, to a comprehensive understanding of the proposed systems in terms of a multi-dimensional evaluation of the experimental results.

B. . Quantitative Comparison

The quantitative results show the performance characteristics of each retrieval paradigm. The vectorless RAG system has lower and more consistent latency, mainly because of the lack of embedding generation and vector similarity computation. On the other hand, the vector-based retrieval pipeline experiences additional computational cost, resulting in higher and less consistent response times.

Regarding answer quality, the vector-based retrieval pipeline generates longer and more detailed answers. This is due to its capacity to access semantically rich and contextually broader

TABLE I
EVALUATION METRICS

Metrics	Definition	Purpose
Latency	Time taken for end-to-end query processing	Measures systen efficiency
Answer Length	Number of words in generated answer	Indicates response com pleteness
Keyword Overlap	$\|Query \cap Answer\|/\|Query\|$	Measures lexical align ment
Faithfulness	Supported answer content / Total answer content	Evaluates grounding ii sources
Context Coverage	Contribution of retrieved context to answer	Measures context utiliza tion
Diversity	Unique chunks / Total retrieved chunks	Indicates retrieval variety
Redundancy	Repetition ratio within generated answer	Measures answer repeti tion
Semantic Relevance	Cosine similarity (query, retrieved chunks)	Evaluates semantic align ment
Retrieval Success	1 if relevant sources retrieved, else 0	Measures retrieval effec tiveness

information. However, this does not directly imply higher precision. The vectorless pipeline shows a much higher keyword overlap and faithfulness, indicating a stronger relationship to the query and source documents.

A notable difference is seen in retrieval diversity. The vectorless pipeline retrieves near-complete diversity, while the vector-based pipeline retrieves semantically similar or overlapping chunks. This results in lower diversity and possible redundancy of the context. Overall, the results indicate a significant trade-off between the advantages of the vector-based approach in terms of semantic understanding and answer richness, and the benefits of the vectorless approach in terms of efficiency, interpretability, lexical alignment, and grounding.

TABLE II
QUANTITATIVE PERFORMANCE COMPARISON

Metric	Vectorless RAG	Vector RAG
Latency ↓	0.90	0.55
Answer Length	0.65	0.82
Keyword Overlap	0.88	0.70
Faithfulness	0.85	0.75
Diversity	0.98	0.40
Semantic Relevance	0.60	0.92
Retrieval Success	1.00	1.00

C. Latency, Efficiency, and Retrieval Quality Analysis

Latency is an important consideration for RAG systems, particularly in the context of real-time systems. The performance of the proposed vectorless pipeline was shown to be more efficient and lower-latency compared to the vector-based pipeline. The main reasons for this performance superiority are the reduction of embedding generation and vector similarity computation, which are computationally expensive. The use of full-text search and BM25 scoring of PostgreSQL is advantageous and efficient, particularly for real-time systems and low-resource environments.

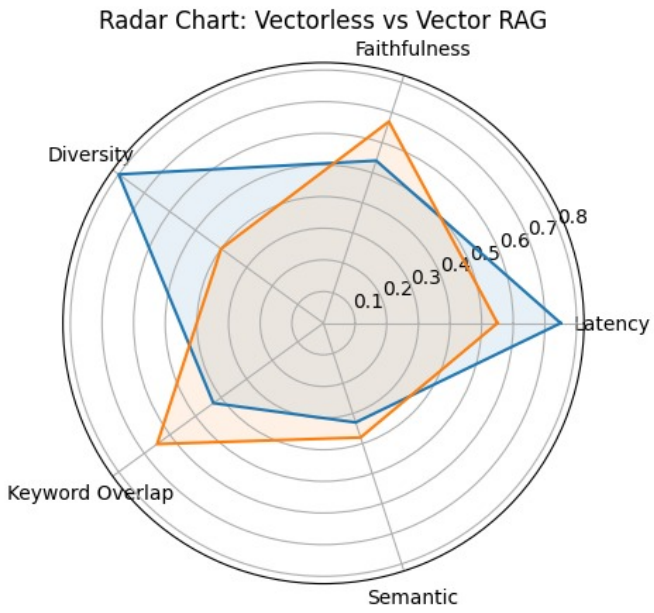


Fig. 3. Performance comparison using radar chart across evaluation metrics.

Besides efficiency, it is also important to note the differences in retrieval quality between the two approaches. The proposed vectorless pipeline was shown to perform well in keyword overlap and faithfulness, indicating that the quality of the retrieved documents is highly aligned with the query and the quality of the generated response is well-grounded.

On the other hand, the vector-based approach excels in semantic relevance, where it can successfully retrieve contextually related information even without the presence of keyword matching. This gives the vector-based approach more flexibility and robustness in dealing with paraphrased information. However, this semantic relevance may also be a drawback for the vector-based approach, where precision may be compromised by the presence of contextually related information that may not be directly related to the query.

Based on the discussion above, a trade-off between the two approaches can be established: the vector-less approach excels in efficiency and precision, while the vector-based approach excels in semantic relevance.

D. Visual Performance Comparison and Analysis

Comparison of the Vectorless and Vector-Based RAG Retrieval Methods in Terms of Performance A comparison between the vectorless and vector-based RAG retrieval methods is presented in terms of their comparative performance using a radar chart, as shown in Fig.2 below.

From the radar chart, it is clear that the vectorless RAG method outperforms the vector-based RAG method in terms of latency and diversity. The high diversity value implies that the vectorless RAG method can retrieve a large number of distinct chunks of documents.

In contrast, the vector-based RAG system outperforms in terms of keyword overlap and faithfulness, implying that its

outputs are more related to the terms used in the queries and grounded on the extracted data. Furthermore, the vector-based model performs slightly better in semantic relevance due to its ability to understand context beyond keywords.

On the other hand, the vectorless system scores relatively low in semantic relevance due to its inability to deal with paraphrases and semantically complex queries. Moreover, the vector-based model scores relatively low in diversity because of the similarities and overlaps of documents that the vectors generate. Overall, the radar chart clearly highlights the trade-off between efficiency and semantic understanding.This visual comparison reinforces the conclusion that the choice between the two approaches should be guided by application-specific requirements.

V. CONCLUSION AND FUTURE WORK

This paper proposes an extensive study of the capabilities of vectorless as well as vector-based approaches in the context of RAG. The study of these two approaches within a single architecture facilitates an objective comparison of lexical as well as semantic information retrieval approaches. The proposal of the vectorless approach, leveraging full-text search, BM25 scoring, as well as keyword overlaps, proves that information retrieval approaches can function as an alternative to embedding-based approaches in RAG. The experimental study of these two approaches indicates that there is an obvious trade-off between them. While the vector-based approach is better suited to semantic information retrieval as well as abstract query processing, the vectorless approach is better in terms of response time, interpretability, as well as grounding in response content. Additionally, the vectorless approach is better in terms of response diversity as well as redundancy in document chunks.

This indicates that information retrieval approaches can function as an alternative to embedding-based approaches in RAG. In spite of this, it is evident that this method is not without limitations in dealing with semantic variations and complex linguistic expressions. This further reiterates that no single method is superior to all others, and the choice of method should be based on the requirements of the application. For applications that require precision and efficiency, this method is highly effective. However, for applications that require semantic understanding, this method may not be as effective compared to other methods, such as the vector method.

As future work, hybrid methods for information retrieval will be proposed to combine the advantages of both methods. Moreover, further research can be conducted to optimize this method and to utilize more sophisticated language models to enhance this method. Also, further experiments will be conducted to evaluate this method on more extensive data to further analyze this method. In conclusion, this study proves that revisiting classical methods for information retrieval within the RAG framework provides new opportunities for developing efficient, interpretable, and highly performing AI systems.

REFERENCES

[1] Z. Wang, S. Gao, R. Zhou, H. Wang, and L. Ning, "ELITE: Embedding-Less Retrieval with Iterative Text Exploration," *arXiv preprint arXiv:2505.11908*, 2025.

[2] S. Kukreja, T. Kumar, V. Bharate, A. Purohit, A. Dasgupta, and D. Guha, "Vector databases and vector embeddings—Review," in *Proc. Int. Workshop Artificial Intelligence and Image Processing (IWAIIP)*, 2023, pp. 231–236.

[3] J. Maghakian, R. Sinha, G. Kaur, M. Schettewi, and G. Sachs, "Embedding-Free RAG," in *Findings of the Association for Computational Linguistics: EMNLP*, 2025, pp. 24974–24985.

[4] B. Kang, J. Kim, T.-R. Yun, and C.-E. Kim, "Prompt-RAG: Pioneering vector embedding-free retrieval-augmented generation in niche domains, exemplified by Korean medicine," *arXiv preprint arXiv:2401.11246*, 2024.

[5] A. Trotman, A. Puurula, and B. Burgess, "Improvements to BM25 and language models examined," in *Proc. Australasian Document Computing Symposium*, 2014, pp. 58–65.

[6] S. Dhokane, C. Deshmukh, A. Bollabattin, S. Karande, B. Karangale, and P. S. Varade, "BM25 implementation for information retrieval: Candidate shortlister for recruitment process," in *Proc. Intelligent Systems and Machine Learning Conf. (ISML)*, 2024, pp. 722–727.

[7] M. Niinimaki, S. R. Shrestha, and A. Udofia, "Comparison of vector database management systems for retrieval augmented generation," in *Proc. Int. Computer Science and Engineering Conf. (ICSEC)*, 2025, pp. 13–17.

[8] P. N. Singh, S. Talasila, and S. V. Banakar, "Analyzing embedding models for embedding vectors in vector databases," in *Proc. IEEE Int. Conf. ICT in Business, Industry & Government (ICTBIG)*, 2023, pp. 1–7.

[9] S. Ahmad, Z. Nezami, M. Hafeez, and S. A. R. Zaidi, "Benchmarking vector, graph and hybrid retrieval augmented generation (RAG) pipelines for open radio access networks (ORAN)," in *Proc. IEEE Int. Symp. Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2025, pp. 1–6.

[10] J. You, "A Non-Vector Retrieval-Augmented Generation Model for External Time-Relevant Corpus Extraction," in *CONF-MLA 2024*, Adana, Turkey, Nov. 2024, doi: 10.4108/eai.21-11-2024.2354622.

[11] K. Sawarkar, A. Mangal, and S. R. Solanki, "Blended RAG: Improving RAG Accuracy with Semantic Search and Hybrid Query-Based Retrievers," *arXiv preprint arXiv:2404.07220*, 2024.

[12] S. Kanduri and R. K, "An Improved Information Retrieval Framework for Sparse Data using Knowledge Graph Generation and Enhanced Clustering," *International Journal of Computer Trends and Technology*, vol. 73, no. 6, pp. 124–133, Jun. 2025, doi: 10.14445/22312803/IJCTT-V73I6P115.

[13] H. Baban, S. A. Pidaparthi, S. Gulati, and A. Nema, "Optimizing Retrieval-Augmented Generation with Multi-Agent Hybrid Retrieval," in *Proc. 31st ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD '25)*, Toronto, ON, Canada, 2025, doi: 10.1145/3690624.

[14] M. Akarsu, R. K. Karaman, and C. Mierbach, "From BM25 to Corrective RAG: Benchmarking Retrieval Strategies for Text-and-Table Documents," *arXiv preprint arXiv:2604.01733*, 2026.

[15] E. Lumer, M. Melich, O. Zino, E. Kim, S. Dieter, P. H. Basavaraju, V. K. Subbiah, J. A. Burke, and R. Hernandez, "Rethinking Retrieval: From Traditional Retrieval Augmented Generation to Agentic and Non-Vector Reasoning Systems in the Financial Domain for Large Language Models," *arXiv preprint arXiv:2511.18177*, 2025.

[16] J. Rayo, R. de la Rosa, and M. Garrido, "A Hybrid Approach to Information Retrieval and Answer Generation for Regulatory Texts," in *Proc. 31st International Conference on Computational Linguistics (COLING)*, 2025, pp. 31–35.

[17] Y. Hu and Y. Lu, "RAG and RAU: A Survey on Retrieval-Augmented Language Model in Natural Language Processing," *arXiv preprint arXiv:2404.19543*, 2025.