MODE: Mixture of Document Experts for RAG

Rahul Anand

rahulanand1103@gmail.com

Abstract

Retrieval-Augmented Generation (RAG) augments language models by integrating retrieval of external knowledge. However, traditional RAG pipelines depend heavily on large vector databases and complex retrieval mechanisms, making them inefficient for small to medium-sized datasets. We introduce MODE (Mixture of Document Experts), a novel framework that organizes documents into semantically coherent clusters and leverages centroid-based retrieval for query processing. By eliminating the reliance on vector databases and re-rankers, MODE provides a scalable, interpretable, and efficient alternative for retrieval and generation, particularly suited for specialized or smaller datasets. Our experiments demonstrate MODE's effectiveness in improving retrieval precision and generation quality compared to standard RAG systems.

Keywords: Retrieval-Augmented Generation, RAG, Document Clustering, Expert Models, Centroid-Based Retrieval, Language Models

1 Introduction

Recent advances in language models have shown that access to external knowledge greatly improves factual accuracy and coherence. Retrieval-Augmented Generation (RAG) frameworks integrate retrieval modules with language models to accomplish this. However, traditional RAG pipelines often rely on large-scale vector databases and computationally expensive retrieval mechanisms, making them suboptimal for smaller datasets. In this work, we introduce MODE (Mixture of Document Experts), a lightweight and efficient RAG alternative designed for scenarios where document corpora are small to medium-sized.

2 Background: Traditional RAG Systems

The standard Retrieval-Augmented Generation (RAG) framework [2] involves:

1. **Text Chunking:** Splitting documents into smaller, semantically meaningful chunks.
2. **Embedding Generation:** Converting text chunks into dense vector representations.
3. **Vector Database Storage:** Storing embeddings in a fast retrieval database.
4. **Retrieval:** Query embeddings retrieve top-k relevant chunks.
5. **Re-ranking:** Cross-encoder models improve retrieval precision.
6. **LLM Generation:** The retrieved passages are passed to an LLM.

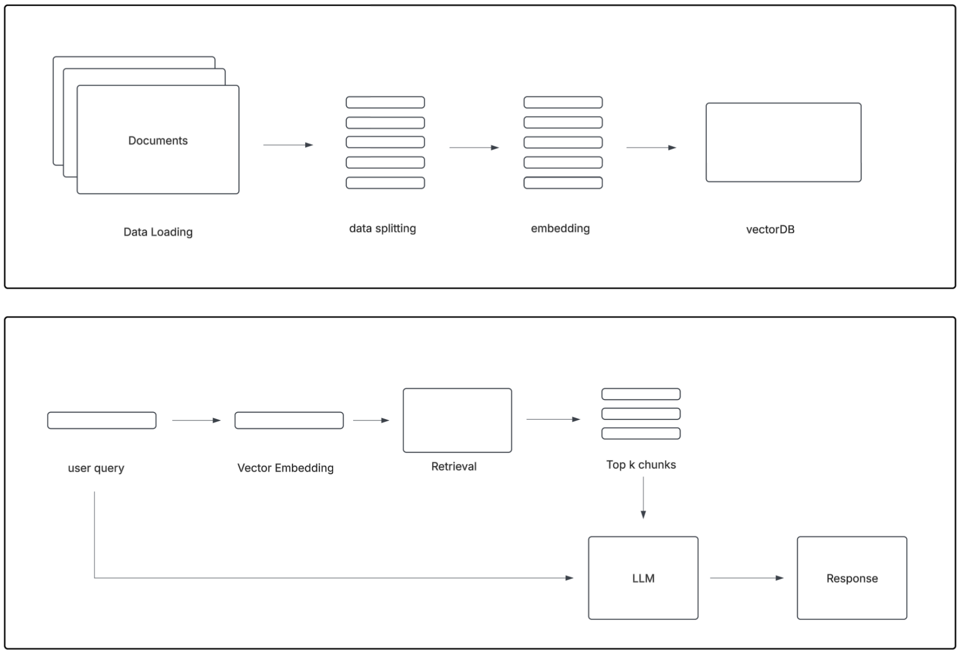

Figure 1: Traditional RAG pipeline.

3 MODE: Mixture of Document Experts

MODE redesigns the RAG pipeline by replacing vector databases and re-rankers with document clustering and centroid-based retrieval.

3.1 Ingestion Phase

- **Text Chunking:** Incoming documents are divided into semantically coherent text chunks.
- **Embedding Generation:** Each chunk is embedded using a model.
- **Hierarchical Clustering:** HDBSCAN groups chunks into clusters [4], optionally refined with KMeans [3] for large clusters.
- **Centroid Computation:** Each cluster's centroid is computed by averaging the embeddings.

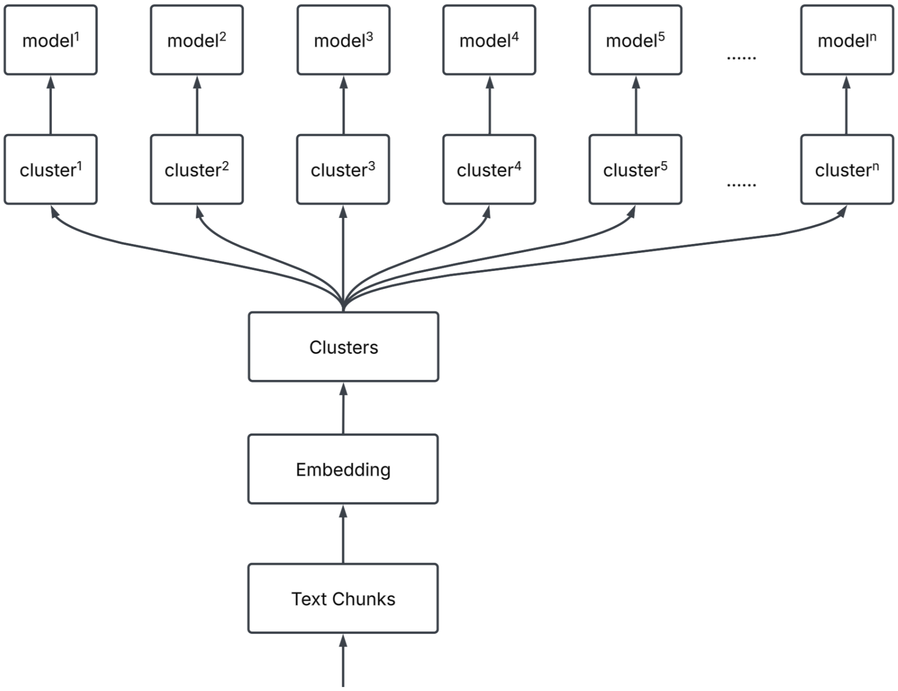

Figure 2: MODE ingestion pipeline.

3.2 Inference Phase

- **Query Embedding:** The user's query is embedded into a dense vector representation.
- **Centroid Matching:** The query embedding is compared against pre-computed cluster centroids to find the most relevant cluster.
- **Expert Selection:** Based on the matched cluster, a subset of specialized expert models is selected.
- **Expert Inference:** Selected expert models independently generate responses conditioned on the query and their respective knowledge.
- **Synthesis:** The outputs from multiple experts are aggregated by a synthesizer module to produce the final answer.

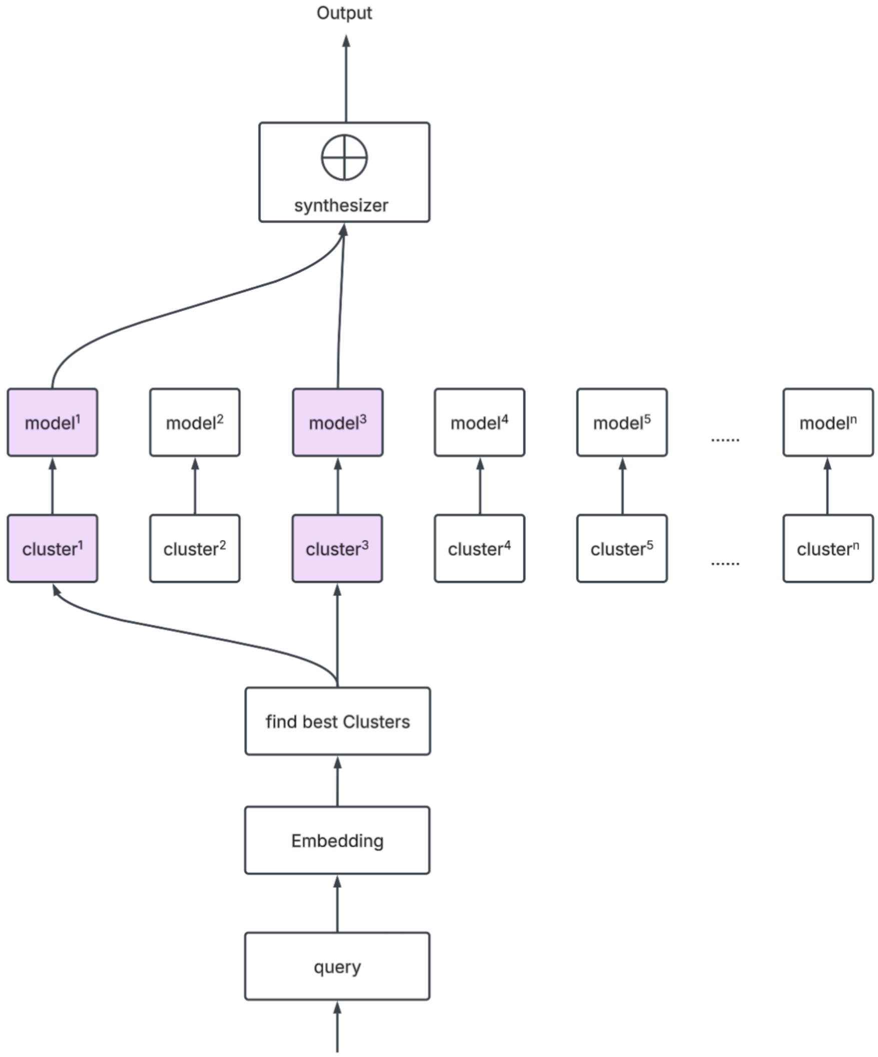

Figure 3: MODE inference pipeline.

4 Experimental Setup

We evaluate MODE against a standard Retrieval-Augmented Generation (RAG) pipeline using two QA datasets: HotpotQA [7] and SQuAD [6]. For both approaches, we measure the following metrics:

- **GPT-based Evaluation:** Using a GPT-4o-based system prompt [5] to check if the generated answer is contextually equivalent to the ground truth, reporting both Accuracy and F1 Score.
- **BERTScore:** Semantic similarity metrics (Precision, Recall, and F1) using a pre-trained DeBERTa model [1].

MODE organizes documents into clusters and retrieves via centroid matching, while Traditional RAG stores embeddings in a vector database with re-ranking. We experiment with different dataset sizes (100, 200, 500 chunks) and test on 100 QA pairs per setup.

5 Results and Analysis

5.1 Evaluation of MODE

Dataset	No Chunk	No Question	Model	GPT Accuracy	GPT F1 Score	BERT Precision	BERT Recall	BERT F1 Score
HotpotQA	100	100	1	0.80	0.8889	0.8059	0.8276	0.8154
HotpotQA	100	100	2	0.70	0.8235	0.7427	0.7612	0.7493
HotpotQA	200	100	1	0.75	0.8571	0.8048	0.7582	0.7745
HotpotQA	200	100	2	0.80	0.8889	0.7746	0.7910	0.7811
HotpotQA	500	100	1	0.7843	0.8791	0.7777	0.7581	0.7613
HotpotQA	500	100	2	0.8039	0.8913	0.7208	0.7507	0.7320
SQuAD	100	100	1	0.78	0.8764	0.7881	0.7939	0.7852
SQuAD	100	100	2	0.89	0.9418	0.7805	0.8241	0.7993
SQuAD	200	100	1	0.72	0.8372	0.7449	0.7380	0.7336
SQuAD	200	100	2	0.78	0.8764	0.7429	0.7828	0.7595
SQuAD	500	100	1	0.71	0.8304	0.7495	0.7473	0.7408
SQuAD	500	100	2	0.82	0.9011	0.7660	0.8047	0.7825

Table 1: Evaluation of MODE across HotpotQA and SQuAD datasets.

5.2 Evaluation of Traditional RAG

Dataset	No. Chunks	GPT Accuracy	GPT F1 Score	BERT Precision	BERT F1 Score
HotpotQA	100	0.70	0.82	0.23	0.29
HotpotQA	200	0.70	0.82	0.37	0.40
HotpotQA	500	0.72	0.84	0.25	0.29
SQuAD	100	0.88	0.94	0.46	0.51
SQuAD	200	0.87	0.93	0.46	0.51
SQuAD	500	0.86	0.92	0.46	0.51

Table 2: Evaluation of Traditional RAG across HotpotQA and SQuAD datasets.

5.3 Comparison

The evaluation results highlight MODE's advantage over traditional RAG systems, especially in handling datasets with complex reasoning requirements like HotpotQA. MODE consistently delivers higher semantic relevance, as evidenced by superior BERTScore metrics across all chunk sizes. While traditional RAG achieves strong GPT-based accuracy on SQuAD, this performance does not translate into high semantic fidelity, suggesting that its retrieved contexts are not always meaningfully aligned with the queries.

In contrast, MODE's cluster-based retrieval mechanism yields more contextually relevant content, contributing to better generation quality. This is particularly evident in HotpotQA, where MODE's expert-driven inference allows it to handle multi-hop reasoning more effectively. Overall, the results demonstrate that MODE is a compelling alternative for applications where retrieval quality and interpretability are critical, especially in resource-constrained or domain-specific settings.

6 Discussion

MODE offers a practical and interpretable solution to the limitations of traditional RAG pipelines. By leveraging document clustering and centroid-based retrieval, it reduces computational overhead and removes the dependency on large-scale vector databases and re-ranking systems. This makes MODE particularly well-suited for small to medium-sized datasets, where traditional retrieval systems may be overkill or inefficient.

An additional benefit of MODE is its modularity—experts can be fine-tuned for specific domains, and clusters can be refined or updated without retraining the entire system. However, scalability remains a consideration. As document corpora grow larger and more diverse, fixed clustering strategies may become insufficient. In such scenarios, MODE could benefit from dynamic clustering, adaptive expert routing, or hybrid approaches that combine lightweight vector retrieval with centroid-guided inference.

7 Conclusion

In this work, we introduced MODE, a Mixture of Document Experts framework that reimagines Retrieval-Augmented Generation by replacing dense retrieval and re-ranking pipelines with document clustering and centroid-based retrieval. Through comprehensive evaluation on HotpotQA and SQuAD datasets, MODE demonstrated superior retrieval precision and generation quality compared to traditional RAG systems, particularly on tasks requiring complex reasoning. By reducing reliance on large vector databases and computationally intensive re-rankers, MODE offers a scalable, interpretable, and efficient alternative for small to medium-sized datasets, paving the way for more lightweight and accessible retrieval-augmented language model systems.

GitHub Repository

The source code for MODE is available at:
`https://github.com/rahulanand1103/mode`

References

[1] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021.

[2] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020.

[3] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[4] Leland McInnes, John Healy, and Steve Astels. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017. doi: 10.21105/joss.00205.

[5] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. URL `https://doi.org/10.48550/arXiv.2303.08774`.

[6] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of EMNLP*, pages 2383–2392, 2016.

[7] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of EMNLP*, pages 2369–2380, 2018.