Document Retrieval

Hok Wai Chan
hokwai@hawaii.edu
University of Hawaii at Manoa
Honolulu, Hawaii, USA

ABSTRACT

In this project, we investigate the evolution and performance of document retrieval methods through a systematic code review and empirical experimentation. Our study covers three main categories: sparse retrieval using TF-IDF and BM25, dense retrieval using state-of-the-art sentence embeddings and FAISS-based nearest neighbor search, and hybrid retrieval approaches combining sparse and dense approaches. We implement these methods from scratch, evaluate them on the MS MARCO dataset, and measure performance using Mean Reciprocal Rank (MRR) and retrieval time. Beyond simple implementation, we explore key factors affecting retrieval quality and efficiency, such as different embedding models, data scales, and combination strategies. Through our findings, we reveal the strengths and limitations of each approach, discuss emerging trends in retrieval technologies, and propose future research directions aiming to further bridge the gap between sparse and dense methods.

KEYWORDS

Document Retrieval, Sparse retrieval, Dense retrieval, TF-IDF, BM25, FAISS, Mean Reciprocal Rank, Retrieval time, KNRM, Sentence-BERT, ColBERT, Similarity metrics

ACM Reference format:

Hok Wai Chan, 2025. Document Retrieval. In *Proceedings of ACM Woodstock conference (WOODSTOCK'18). ACM, New York, NY, USA, 2 pages.* https://doi.org/10.1145/1234567890

1 INTRODUCTIONS

Document retrieval is a fundamental component of information retrieval systems, essential for a wide range of applications such as web search engines, recommendation systems, and digital libraries. The primary goal of document retrieval is to identify and rank documents from a large corpus that are most relevant to a given user query. Over the decades, retrieval methods have advanced significantly, evolving from early models based on exact term matching to more advanced neural network-based representations that capture deeper semantic relationships between queries and documents. Classical sparse retrieval methods, such as Term Frequency-Inverse Document Frequency (TF-IDF) and BM25, represent documents and queries as sparse vectors over a predefined vocabulary. These approaches rely heavily on matching terms between queries and documents, offering efficiency and interpretability but often struggling with vocabulary mismatch and synonymy. In contrast, dense retrieval techniques leverage advances in deep learning to embed queries and documents into dense, continuous vector spaces. Models like Sentence-BERT and ColBERT aim to capture semantic meaning beyond exact token matches, allowing for improved retrieval performance, especially in scenarios involving paraphrasing or complex information needs. However, dense retrieval methods bring new challenges related to computational cost, storage overhead, and retrieval latency, particularly for large-scale corpora. In this project, we aim to systematically explore the evolution and performance trade-offs between sparse, dense, and hybrid retrieval strategies. Our study consists of two main components: (1) a code review and reimplementation of representative retrieval methods, and (2) a set of empirical experiments evaluating these methods on a subset of the MS MARCO dataset, a widely used benchmark for passage retrieval. For evaluation, we adopt Mean Reciprocal Rank (MRR) to measure retrieval quality and retrieval time to assess efficiency. In addition to basic performance comparisons, we investigate several key factors that influence retrieval outcomes, including corpus size and combination strategies for hybrid approaches. Through experimentation and analysis, we aim to uncover the strengths and weaknesses of different retrieval paradigms, explore emerging trends in document retrieval research, and propose potential directions for future development. By providing a comprehensive and practical perspective on sparse, dense, and hybrid retrieval methods, this project aims not only to deepen our understanding of retrieval system design but also to inform future research on bridging the gap between traditional and neural representations for document retrieval.

2 RELATED WORK

2.1 Boolean Retrieval

The earliest methods of document retrieval were based on Boolean retrieval models, where documents were retrieved if they satisfied logical combinations of query terms using operators like AND, OR, and NOT. This approach was straightforward: a document either matched a query exactly or it did not, without any ranking of results. Boolean retrieval systems assumed that documents and user queries could be accurately represented by sets of index terms. In practice, this was unrealistic because of the uncertainty in how users formulate queries and how documents cover topics. Furthermore, Boolean retrieval treated relevance as a binary condition; documents were either relevant or not, without recognizing varying degrees of relevance. To address the inflexibility and limitations of pure Boolean models, extended Boolean approaches were proposed, such as the p-norm model, which introduced the concept of partial matching and ranking documents based on the degree to which they satisfied the query. However, despite these theoretical advances, many retrieval systems continued to rely on the traditional Boolean framework for years, primarily due to financial constraints and a lack of awareness of practical research advancements (Radecki, 1988).

2.2 Vector Space Model

Over time, the shortcomings of Boolean retrieval, especially its inability to rank documents by relevance or handle partial matches, became more apparent. This led to the development of ranked retrieval models, such as the vector space model (VSM), which offered a major improvement by introducing the concept of graded relevance and ranked results. In this model, documents and queries are represented as sparse vectors in a high-dimensional space, where each dimension corresponds to a term from the vocabulary. Most vector entries are zero, indicating the absence of certain terms in a document or query, which is characteristic of sparse retrieval. The relevance of a document to a query is determined by calculating the similarity between their vectors, most commonly using cosine similarity, which measures the angle between two vectors rather than their distance. This approach allowed documents to be ranked according to their degree of match to a query, rather than returning an unranked set. Unlike Boolean retrieval, VSM could recognize partial matches and varying degrees of relevance, making it significantly more flexible and effective, particularly for long or complex queries. In addition, term weighting schemes such as TF-IDF (term frequency–inverse document frequency) were developed to give more importance to distinctive and informative terms. TF-IDF increases the weight of terms that are frequent in a document but rare across the corpus, thereby improving the ability of vector-based retrieval systems to differentiate between highly relevant and less relevant documents (Salton, Wong, & Yang, 1975).

2.3 Probabilistic Models

The next significant step in the evolution of information retrieval models was the development of probabilistic models, which provided a more advanced approach to ranking documents based on relevance. These models, including the widely used BM25 ranking function, relied on probabilistic inference, a technique that models the likelihood that a document is relevant to a given query based on statistical principles. Unlike the Vector Space Model (VSM), which depends on deterministic term matching to measure document similarity, probabilistic models aim to consider the uncertainty in determining a document's relevance. The main idea of probabilistic models is to estimate the probability of relevance of a document for a query, evaluating elements such as term frequency (how often a term appears in a document), document frequency (how common or rare a term is across the collection), and document length (which influences term distribution within a document). One well-known example of a probabilistic model is BM25, which builds on the basic probabilistic framework by adjusting the weight of terms based on their frequency in a document and their occurrence across the entire document collection. BM25 uses a limiting function for term frequency, meaning the contribution of a term to the relevance score increases with frequency, but at a decreasing rate as the term becomes more frequent. It also normalizes for document length to prevent longer documents from being unfairly ranked higher (Hiemstra, 1998). By considering these factors, BM25 provides a more advanced ranking of documents that often outperforms traditional vector space models, especially when working with large datasets. The flexibility and robustness of probabilistic models, particularly in large and varied datasets, make them more effective for many retrieval tasks. This shift to probabilistic models also laid the foundation for more complex methods, such as Latent Dirichlet Allocation (LDA).

2.4 Semantic Analysis

Building on the advancements made by probabilistic models, the focus of further improving document retrieval shifted towards semantic analysis, where the goal was to capture deeper relationships between words and their meanings. One of the first significant steps in this direction was Latent Semantic Indexing (LSI). LSI utilized singular value decomposition (SVD) to reduce the dimensionality of

the term-document matrix, revealing hidden structures and relationships between terms and documents. By mapping both documents and queries into a latent semantic space, LSI was able to capture synonymy (different words with similar meanings) and polysemy (words with multiple meanings) more effectively than traditional term-based models (Deerwester et al., 1990). However, despite its strengths, LSI was computationally intensive and struggled with scalability for very large collections. As the field advanced, Latent Dirichlet Allocation (LDA) emerged as a more scalable and theoretically robust approach to topic modeling (Blei, Ng, & Jordan, 2003). Unlike LSI, which focused on dimensionality reduction, LDA treated documents as mixtures of topics, with each topic being characterized by a distribution over words. This probabilistic framework allowed for a more insightful understanding of a document's content, enabling retrieval based on inferred semantic topics rather than simple surface-level word matching. The shift towards topic modeling with LDA significantly improved retrieval quality, especially for complex or large-scale corpora, marking a major advancement in the evolution of retrieval models. While semantic models like LSI and LDA brought substantial improvements in capturing the underlying meaning of documents, they still had limitations when it came to optimizing ranking based on specific user intent and contextual factors. As the field of information retrieval continued to advance, the focus turned towards integrating machine learning techniques. This evolution led to the development of Learning to Rank (LTR) models, which moved the emphasis from semantic analysis to learning optimal ranking strategies directly from data.

2.5 Learning to Rank (LTR)

With the rise of machine learning, document retrieval shifted towards models that could learn optimal retrieval strategies from data. Traditional IR models relied on manually tuned parameters and heuristics, but learning to rank (LTR) approaches allowed systems to automatically learn ranking functions by using labeled training data. Features such as term frequency, document length, query document similarity, and metadata could be integrated into supervised learning models like Support Vector Machines (SVM) or gradient-boosted decision trees to predict document relevance. LTR frameworks typically involve three paradigms: pointwise, pairwise, and listwise approaches, each offering different ways to model the ranking problem (Cao et al., 2007). These machine learning techniques enabled significant improvements over traditional retrieval models by adapting to specific user behaviors, tasks, and domains. While LTR models helped improve document retrieval by automatically learning how to rank documents based on various features, they still depended on manually selected features. These models worked well for ranking but struggled with understanding the deeper meaning behind words and how they relate to each other in a more natural way (Liu, 2009).

2.6 Neural Approaches

The limitations of Learning to Rank (LTR) in capturing deeper semantic relationships between words prompted the development of neural approaches that could better understand the context and meaning of language. While LTR models relied on manually selected features, neural approaches leverage deep learning models to automatically learn these relationships from data. The development of word embeddings enabled words to be represented as dense vectors in a continuous semantic space, allowing retrieval systems to capture deeper semantic relationships between terms based on their usage patterns and contexts. This advancement led to the development of dense retrieval methods, in which both queries and documents are embedded into a shared vector space (Gao & Callan, 2021). This approach enables retrieval based on semantic similarity rather than exact term matching. Dense retrieval systems, such as Dense Passage Retrieval (DPR) and ColBERT, utilize deep learning models like BERT to generate contextual embeddings, where the meaning of a word varies depending on its surrounding words (Devlin et al., 2019). This ability to model words dynamically greatly improves the system's capacity to understand user meaning and the complex interpretations of queries and documents.

2.7 Hybrid Retrieval Models

Despite the advancements brought by dense retrieval methods, such as the ability to understand semantic relationships and context, there remain cases where traditional sparse retrieval methods, like BM25, still offer distinct advantages. Sparse methods are particularly effective in precise term matching, especially for rare terms, and they are computationally efficient for large-scale datasets. On the other hand, dense retrieval methods are particularly strong in capturing the deeper, semantic meaning of words and handling paraphrases. Recognizing that sparse and dense methods each had unique strengths, hybrid retrieval systems were developed to combine the advantages of both. These systems combined traditional sparse retrieval (e.g., BM25) with dense neural retrieval. By combining both approaches, hybrid systems can retrieve a broader range of relevant documents that might be missed if only one method is used (Chen et al., 2022).

2.8 Retrieval Augmented Generation (RAG)

Following the development of hybrid retrieval systems, research expanded toward using retrieved documents not just to rank results but to actively support the generation of new content. Retrieval Augmented Generation (RAG) models combine retrieval and generation into a single unified process. Instead of just identifying and presenting relevant documents, RAG models feed retrieved documents directly into a generative language model to help craft more accurate and contextually grounded responses. This architecture is well-suited for tasks such as open domain question answering and knowledge-intensive applications, where relying only on a model's internal knowledge may be insufficient. A typical RAG system consists of two components, a retriever (such as DPR or a hybrid retriever) and a generator (such as T5, BERT, or GPT-based models). The retriever first identifies the top-k most relevant documents, and the generator then conditions on both the original query and the retrieved documents to produce an accurate and contextually grounded response. By directly coupling retrieval with generation, RAG systems can reduce hallucination and better handle the need for up-to-date information (Lewis et al., 2020).

3 Technical details

In the experiment, we implemented and evaluated a range of document retrieval methods, specifically sparse retrieval (TF-IDF and BM25), dense retrieval (Sentence-BERT with FAISS), and hybrid approaches that combine both sparse and dense methods. The primary dataset used for evaluation was the MS MARCO passage ranking dataset (version 2.1), which is a well-established benchmark in information retrieval tasks. All retrieval methods were tested on 50, 4988, 49896, and 997,459 document subsets of MS MARCO to ensure correctness and evaluate performance at scale. Queries, passages, and relevance labels were parsed and stored in dictionaries for efficient lookup. Each query is associated with a unique set of relevant passages, which enables precise evaluation of retrieval performance using metrics such as Mean Reciprocal Rank (MRR).

3.1 Sparse Retrieval

For sparse retrieval, two classical methods were implemented: TF-IDF and BM25. The TF-IDF approach utilized TfidfVectorizer from scikit-learn to construct a term-document matrix for the corpus. Cosine similarity between the query and all passages was computed to retrieve the top-k documents. For BM25, the rank_bm25 library was used. The corpus was first tokenized into word-level units, and then BM25 scoring was performed against the tokenized query. Both methods produced ranked document lists based on their respective scoring mechanisms.

3.2 Dense Retrieval

Dense retrieval was implemented using the Sentence-BERT model multi-qa-distilbert-cos-v1 from the sentence-transformers library. Both corpus and queries were embedded into fixed-length vector representations using the model and normalized to unit vectors to enable cosine similarity search. To perform efficient nearest neighbor search in large embedding spaces, the FAISS library was used to index the corpus embeddings. An IndexFlatIP index (for inner product similarity) was created. Each query embedding was searched against this index to retrieve top-k passages based on dense similarity scores.

3.3 Hybrid Retrieval (Sparse + Dense)

To combine the strengths of both sparse and dense retrieval methods, a hybrid retrieval mechanism was implemented. The system allows for merging the results through different strategies, including intersection, union, and score-based ranking. In the intersection strategy, only documents appearing in both the sparse and dense top-k results are retained, while the union strategy combines all unique documents retrieved by either method. The framework supports flexible switching between sparse methods (TF-IDF or BM25) and dense retrieval, enabling experiments with various integration approaches.

4 Evaluation & Discussion

The results clearly show that dense retrieval and BM25 consistently outperform TF-IDF across all tested document subset sizes. While TF-IDF achieves a Mean Reciprocal Rank (MRR) of 0.5, both BM25 and dense retrieval reach perfect MRR scores of 1.0, maintaining this performance even as the dataset scales from 50 to 5000 queries and thousands of documents. This indicates that BM25 and dense retrieval are highly effective at capturing semantic relevance and ranking relevant passages at the top. Dense retrieval, utilizing semantic embeddings, is also slightly faster than BM25 due to efficient vector search enabled by FAISS. Hybrid retrieval strategies further emphasize these strengths. Combining TF-IDF with dense retrieval does not significantly improve MRR, maintaining a score of 0.5, indicating limited complementarity between the two methods. However, combining BM25 with dense retrieval (both intersection and union strategies) preserves the strong performance (MRR = 1.0), as BM25 retrieves highly relevant documents based on term frequency and

normalization, while dense retrieval captures semantic relationships not represented in sparse methods. The minimal overhead in retrieval time for hybrid approaches supports their practical viability. As the number of queries and documents increases, the relative performance trends remain consistent, but the scalability and robustness of each method become more apparent. Dense retrieval continues to maintain perfect MRR even at larger scales, showcasing its ability to generalize semantic relevance across a broader corpus. BM25 also remains highly effective, proving that traditional lexical methods remain competitive when properly tuned. In contrast, TF-IDF's performance stagnates at an MRR of 0.5, indicating its limitations in modeling complex contextual relationships. Dense retrieval's scalability is further supported by its minimal increase in retrieval time due to efficient vector indexing (e.g., via FAISS), which becomes crucial as the dataset grows. These results provide evidence of the robustness of dense retrieval and BM25 methods while exposing the limitations of simpler techniques like TF-IDF in more demanding real-world scenarios.

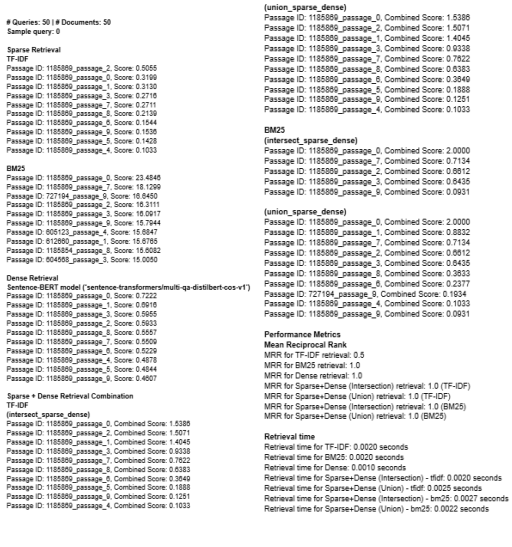

Figure 1: Results of sample size 50

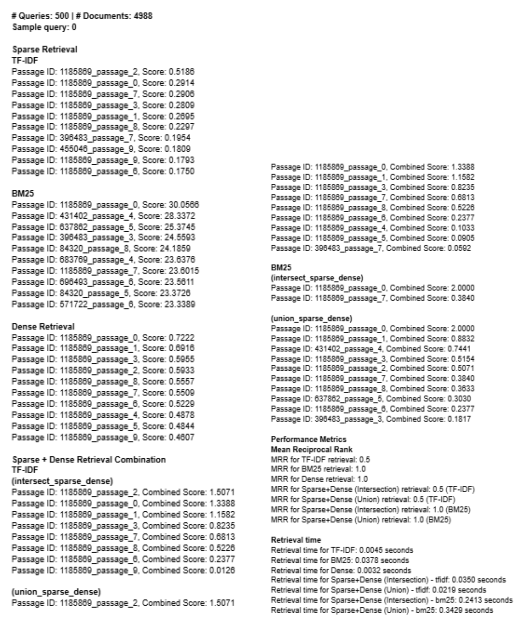

Figure 2: Results of sample size 4988

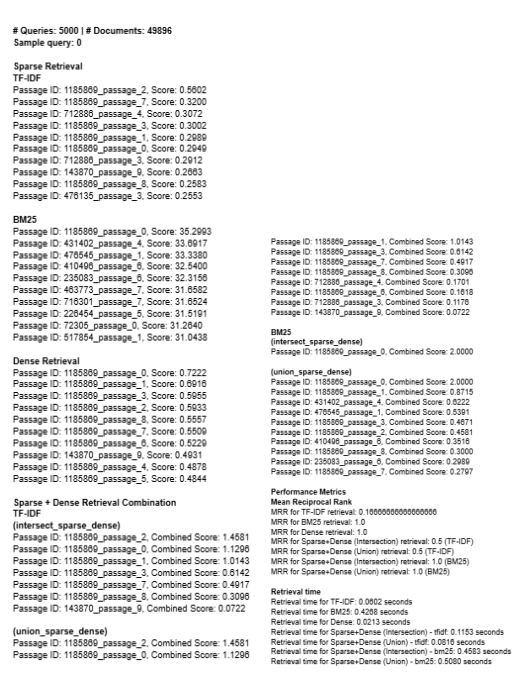

Figure 3: Results of sample size 49896

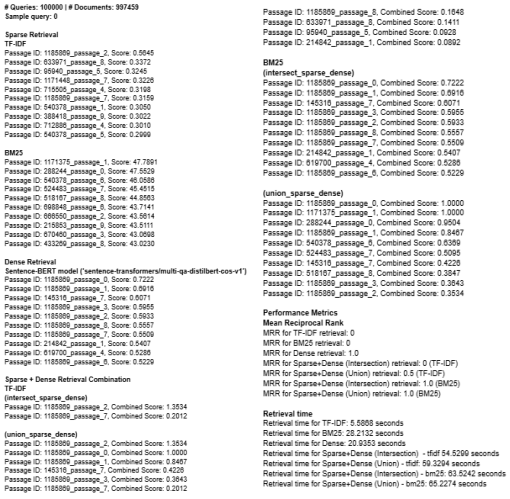

Figure 4: Results of sample size 997459

5 Limitations

Although the results demonstrate potential, this study has several limitations that should be acknowledged. Primarily, all experiments were conducted on a single dataset, which may limit the generalizability of the findings. Different datasets, especially those with varying domain vocabularies, passage lengths, or noise levels, could produce different performance outcomes across retrieval methods. Additionally, the evaluation was primarily based on Mean Reciprocal Rank (MRR) and retrieval scores, without deeper analysis of downstream task performance, such as question answering accuracy or user satisfaction. The scalability of Dense Retrieval was only partially explored; while runtime was measured, indexing and memory usage were not evaluated in depth, which could be critical when deploying these methods in real-world systems.

6 Proposal for future directions

Building on the current findings, several important research directions can be pursued to deepen and broaden the understanding of retrieval systems. First, expanding the evaluation across diverse datasets, such as biomedical literature, legal documents, or multi-lingual corpora, would help assess how retrieval methods generalize across domains with different linguistic and semantic characteristics. Second, incorporating domain-adaptive fine-tuning for Dense Retrieval models could significantly enhance their semantic matching capabilities, particularly in specialized contexts where vocabulary and usage differ from pretraining data. Another direction is the exploration of hybrid architectures that dynamically weight sparse and dense signals, rather than relying on static intersection or union strategies. This could allow retrieval models to adaptively emphasize the most relevant features based on the query context. Additionally, instead of always combining sparse and dense signals, training a small rule-based or AI system to choose the best signal for each query could improve retrieval efficiency. In addition, teaching the system to detect query styles, such as formal language, slang, or question formats, and adapting the retrieval strategy accordingly would further refine the process. Integrating user feedback loops, such as click-through rates or explicit relevance judgments, could lead to personalized or adaptive retrieval systems that evolve with user behavior.

REFERENCES

[1] Radecki, T. (1988). Trends in research on information retrieval—the potential for improvements in conventional boolean retrieval systems. Information Processing & Management, 24(3), 219-227.

[2] Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. Communications of the ACM, 18(11), 613-620.

[3] Hiemstra, D. (1998). A linguistically motivated probabilistic model of information retrieval. In Research and Advanced Technology for Digital Libraries: Second European Conference, ECDL'98 Heraklion, Crete, Greece September 21–23, 1998 Proceedings 2 (pp. 569-584). Springer Berlin Heidelberg.

[4] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), 993-1022.

[5] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. Journal of the American society for information science, 41(6), 391-407.

[6] Liu, T. Y. (2009). Learning to rank for information retrieval. Foundations and Trends® in Information Retrieval, 3(3), 225-331.

[7] Cao, Z., Qin, T., Liu, T. Y., Tsai, M. F., & Li, H. (2007, June). Learning to rank: from pairwise approach to listwise approach. In Proceedings of the 24th international conference on Machine learning (pp. 129-136).

[8] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers) (pp. 4171-4186).

[9] Gao, L., & Callan, J. (2021). Unsupervised corpus aware language model pre-training for dense passage retrieval. arXiv preprint arXiv:2108.05540.

[10] Chen, T., Zhang, M., Lu, J., Bendersky, M., & Najork, M. (2022, April). Out-of-domain semantics to the rescue! zero-shot hybrid retrieval models. In European Conference on Information Retrieval (pp. 95-110). Cham: Springer International Publishing.

[11] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in neural information processing systems, 33, 9459-9474.