Temporal-Aware Hybrid Retrieval System for Medical Literature Search

Information Retrieval System Design Report

1 Introduction

The COVID-19 pandemic generated a large volume of scientific literature between 2020 and 2024. This information explosion creates significant challenges for medical researchers, clinicians, and public health officials who must rapidly find relevant, accurate, and current information to inform clinical decisions and policy recommendations.

Traditional keyword-based search systems face three critical limitations in this domain. First, the **vocabulary mismatch problem** [4] is particularly acute in medical literature, where concepts can be expressed using diverse terminology, for example, "heart attack prevention" versus "myocardial infarction prophylaxis." Second, **temporal relevance** is crucial as medical knowledge evolves rapidly; COVID-19 treatment guidance from early 2020 fundamentally differs from 2023 recommendations following extensive clinical trials. Third, **result redundancy** wastes researchers' time when dozens of studies investigating the same question produce repetitive search results, compounded by potentially contradictory findings between observational studies and subsequent randomized controlled trials.

Design Objectives. This project aims to design and implement a temporal-aware hybrid retrieval system that: (1) combines lexical and semantic matching through neural sparse (SPLADE-inspired) and dense (Sentence-BERT) representations fused via Reciprocal Rank Fusion (RRF), (2) implements an alternative projection-based fusion approach using random projections to empirically compare rank-level versus vector-level fusion strategies, (3) incorporates temporal filtering to prioritize temporally-relevant documents, (4) applies diversity-based re-ranking using Maximal Marginal Relevance (MMR) to reduce redundancy, and (5) achieves production-ready performance using entirely open-source tools.

Research Questions. This system design addresses four key questions:

- **RQ1:** Can hybrid fusion techniques (neural sparse + dense retrieval) outperform single-method approaches (BM25-only or dense-only) on COVID-19 literature search?
- **RQ2:** Does temporal filtering improve relevance without significantly reducing recall for time-sensitive medical queries?
- **RQ3:** How effective is MMR-based diversification in reducing redundancy while maintaining ranking quality for medical literature retrieval?
- **RQ4:** Does projection-based fusion (vector-level) offer competitive performance compared to Reciprocal Rank Fusion (rank-level) for combining neural sparse and dense retrievers?

2 Review of Related Work

In this section we review foundational work in hybrid retrieval systems, fusion techniques, and diversity-aware ranking, and address the problem, approach taken, results and our thoughts.

2.1 The Vocabulary Mismatch Problem

Problem Context. Furnas et al. [4] first quantified the vocabulary mismatch problem, showing that two people choose the same term for an object only 10–20% of the time. This fundamental challenge means that traditional lexical matching systems (BM25, TF-IDF) fail when queries and documents use different terminology for identical concepts. In medical domains, this problem is particularly acute due to specialized terminology, synonyms (e.g., "heart attack" versus "myocardial infarction"), and multilingual literature.

Sparse Retrieval Approaches. *Robertson & Zaragoza [8] – BM25.* BM25 is a probabilistic lexical retrieval model that became the standard baseline in TREC evaluations. While effective for exact-term matching, it struggles when queries and documents use different terminology. This limitation motivates the use of neural-enhanced sparse representations to better address vocabulary mismatch.

Formal et al. [2] – SPLADE. SPLADE extends sparse retrieval by using transformer models to produce context-

aware term weights while preserving sparse representations compatible with inverted indexing. Experiments on MS MARCO show substantial improvements over BM25 (MRR@10: 0.184 → 0.322) while remaining competitive with dense retrieval models. This makes SPLADE a strong candidate for handling terminology variation in medical literature.

Dense Retrieval Approaches. *Karpukhin et al. [5] – Dense Passage Retrieval (DPR).* DPR introduced dense retrieval using a dual-encoder architecture that represents queries and documents as semantic embeddings. Experiments showed that dense retrieval could match or outperform BM25 on open-domain retrieval tasks. Later work [6] demonstrated that hybrid approaches combining sparse and dense retrieval capture complementary signals and consistently improve performance, motivating our fusion-based design.

Reimers & Gurevych [10] – Sentence-BERT. Standard BERT cross-encoders require evaluating each query–document pair, making them impractical for large-scale retrieval. Sentence-BERT generates fixed sentence embeddings that allow efficient similarity search while maintaining performance close to cross-encoder models. This efficiency makes Sentence-BERT a practical choice for dense retrieval in our system.

2.2 Hybrid Fusion Techniques

Multiple studies [1, 6, 3] show that hybrid retrieval systems combining sparse and dense methods consistently outperform single-model approaches by 10–25%. This occurs because the two methods capture complementary signals, motivating the use of hybrid retrieval architectures.

Cormack et al. [1] introduced Reciprocal Rank Fusion (RRF), an unsupervised method for combining ranked lists from multiple retrieval systems. Documents are scored as:

$$\mathrm{RRF}(d) = \sum_{r \in R} \frac{1}{k + r(d)}. \tag{1}$$

Experiments on TREC datasets showed that RRF improves retrieval effectiveness by around 4–5% over individual systems. Its simplicity and robustness make it a suitable baseline for rank-level fusion in our system.

Luan et al. [7] showed that sparse and dense retrieval capture complementary signals and that hybrid fusion can improve effectiveness over either method alone. They also discuss projection techniques for compressing sparse vectors while preserving similarity structure, suggesting that projection-based approaches can provide an alternative method for combining signals.

Zamani et al. [9] investigated learned projections for aligning sparse and dense representations. While supervised projections can improve performance by 3–5%, they require relevance labels for training. In contrast, random projection provides a simpler unsupervised alternative, motivating our evaluation of projection-based fusion alongside RRF.

2.3 Diversity and Re-ranking

Presenting highly similar results can reduce the usefulness of search systems, particularly in domains such as medical literature where multiple studies often investigate the same question [12]. Carbonell and Goldstein [11] proposed Maximal Marginal Relevance (MMR) as a reranking strategy that balances query relevance with novelty. Documents are selected using:

$$\arg \max_{d \in R \setminus S} \left[\lambda \cdot \mathrm{Sim}_1(d, q) - (1 - \lambda) \cdot \max_{d' \in S} \mathrm{Sim}_2(d, d') \right], \tag{2}$$

where λ controls the trade-off between relevance and diversity. Their system achieved strong performance in the SUMMAC 1998 evaluation (F-score 0.73), demonstrating that incorporating novelty can reduce redundancy while maintaining relevance. This motivates the use of MMR for diversity-aware ranking in our system.

2.4 Temporal Dynamics in Retrieval

Dakka et al. [13] show that document publication time can improve retrieval effectiveness for time-sensitive queries when combined with topical relevance. Their experiments on news collections showed improved precision at top ranks when temporal signals were incorporated into ranking. Inspired by this work, our system introduces temporal awareness by prioritising documents within relevant time ranges for medical queries.

2.5 Identified Gaps That Motivate Our Design

Despite extensive research, three significant gaps remain:

Gap 1: Limited Medical Literature Focus. Most hybrid retrieval systems are evaluated on general-domain corpora (MS MARCO, Wikipedia, news). Medical terminology has unique characteristics (high synonym density, nested concepts, evolving nomenclature) that may exhibit different fusion dynamics. Our system specifically targets COVID-19 literature.

Gap 2: Inadequate Temporal Integration. While temporal ranking exists in web search [14], medical literature requires stricter temporal filtering. A paper about

COVID-19 vaccines from March 2020 (pre-clinical trials) has fundamentally different validity than December 2021 papers (post-authorization). Our system integrates temporal filtering directly into the retrieval pipeline.

Gap 3: Fusion Method Comparison Gap. No existing study compares RRF versus projection-based fusion on the same dataset with controlled experimental conditions. Most papers evaluate their proposed method against baselines but not alternative fusion strategies. Our system implements both approaches, enabling direct empirical comparison.

2.6 Summary and Design Implications

The literature review leads to four key design decisions:

1. **Hybrid Retrieval:** Implement both neural sparse (SPLADE-inspired) and dense (Sentence-BERT) retrieval, as neither alone is optimal.

2. **RRF as Baseline:** Use Reciprocal Rank Fusion for its simplicity, robustness, and proven effectiveness; implement projection-based fusion as enhancement.

3. **MMR for Diversity:** Apply Maximal Marginal Relevance to address redundancy problem prevalent in medical literature.

4. **Temporal Filtering:** Integrate temporal awareness throughout pipeline, not as post-processing.

3 Search Engine Design

3.1 Task and Dataset

3.1.1 Task Definition

The proposed system addresses the task of temporal ad-hoc document retrieval with diversity-aware ranking. The input consists of a natural language query (e.g., "What is COVID vaccine effectiveness in 2020?") submitted against a corpus of scientific papers derived from the TREC-COVID collection, combined with associated metadata from the official TREC-COVID website. Queries may include explicit temporal constraints (e.g., a specified year) or implicit temporal intent, where more recent publications are preferred.

The output of the system is a ranked list of the top-10 documents. Ranking is performed using a hybrid retrieval strategy that combines neural sparse and dense representations through Reciprocal Rank Fusion (RRF). The retrieved results are further refined through temporal filtering to prioritise documents from relevant time periods, followed by Maximal Marginal Relevance (MMR) re-ranking to reduce redundancy and improve coverage of different aspects of the query topic.

3.1.2 Dataset: TREC-COVID

We use the **TREC-COVID** benchmark from the BEIR collection[18], merged with official NIST metadata[19]. The TREC-COVID corpus itself contains **171,332 COVID-19 research papers** and NIST metadata contains the metadata information of 1,056,659 research papers from 1954-2022 (CORD-19 snapshot: June 2, 2022) [17], **50 expert-crafted queries**, and **66,336 graded relevance judgments** (qrels: 0 = not relevant, 1 = partially relevant, 2 = highly relevant).

Dataset Rationale. TREC-COVID is chosen for four key reasons. First, it provides human expert relevance judgments enabling offline evaluation without manual annotation. Second, it contains authentic peer-reviewed medical literature with natural terminology variation, directly addressing vocabulary mismatch challenges. Third, rich NIST metadata (publication dates, journals, DOIs, authors) enables temporal filtering and source analysis.

Dataset Structure. BEIR format provides `corpus.jsonl` (each document has `_id`, `title`, `text`), `queries.jsonl` (each query has `_id`, `text`), and `qrels.tsv` (columns: query-id, corpus-id, graded relevance). NIST metadata includes: `cord_uid`, `publish_time`, `authors`, `journal`, `doi`, `pmcid`, `pubmed_id`, `source_x`, `url`. Documents are unified via `cord_uid` to combine BEIR indexing text with temporal/bibliographic metadata.

Temporal Coverage. Of the 171,332 documents in the TREC-COVID corpus, 96.4% have a parseable publication date, while 3.6% (6,159 documents) have no date in the metadata. Among dated documents, the corpus splits nearly evenly: approximately **49.8% pre-2020** (historical SARS, MERS, and broader coronavirus research) and **50.2% from 2020** (early COVID-19 pandemic studies up to July 2020).

Quality Filters. English language (TREC-COVID standard); has abstract ($\geq$ 50 words); has `publish_time` (required for temporal filtering); peer-reviewed publication (validated via `source_x`: PMC, Medline, WHO).

Queries. The 50 official Round 5 topics represent diverse medical information needs from treatment effectiveness to transmission mechanisms. Cross-round queries from Rounds 1–4 provide temporal variations for the

same documents (via `cord_uid`). Synthetic queries generated using doc2query/T5 [20] inherit relevance judgments from source documents for expanded robustness testing.

Limitation. The TREC-COVID corpus is frozen at July 2020 (171,332 documents), so queries referencing post-July 2020 developments return no results. Our evaluation focuses on queries relevant to the 2019–2020 period of the pandemic.

3.1.3 Evaluation Methodology

Test Query Set. Our evaluation uses an **expanded query set** beyond the 50 official TREC-COVID Round 5 topics:

1. **Primary Evaluation:** 50 official topics with expert relevance judgments for standard benchmark comparison.

2. **Cross-Round Queries:** Additional queries from TREC-COVID Rounds 1–4 mapping to same documents (via `cord_uid`), providing temporal query variations for identical relevant documents.

3. **Synthetic Queries:** Generated using doc2query/T5 from judged documents, inheriting relevance judgments from source documents (weakly-supervised labels) to expand evaluation coverage beyond 50 official topics.

All queries span diverse information needs: temporal constraints (e.g., "COVID-19 treatment protocols March 2020"), recency preferences (e.g., "latest research updates"), and specific scientific questions (e.g., "ACE2 receptor binding mechanism"). The expanded query set tests system performance across query formulation variations while maintaining grounded relevance judgments.

Evaluation Metrics. Six complementary metrics measure system quality:

1. **nDCG@10 (Target: ≥ 0.60):** Primary ranking metric measuring position-aware relevance.

$$\text{nDCG@10} = \frac{\sum_{i=1}^{10} \frac{2^{rel_i}-1}{\log_2(i+1)}}{\text{IDCG@10}}.$$

2. **MAP@10 (Target: ≥ 0.45):** Mean average precision across all relevant documents.

3. **P@10 (Target: ≥ 0.70):** Fraction of top-10 results that are relevant.

4. **α-nDCG@10 (Target: ≥ 0.55):** Diversity-aware nDCG with $\alpha = 0.5$.

5. **Temporal Accuracy (Target: $\geq 80\%$):** Percentage of top-10 results within query year ± 1 year window.

6. **Query Latency (Target: < 2 seconds):** End-to-end response time.

Baseline Systems. Ablation study with six configurations (Table 1):

Statistical Testing. Paired t-tests ($p < 0.05$) compare B3 against each baseline across the expanded query set, ensuring improvements are statistically reliable rather than random variation. Primary benchmark comparison uses the 50 official TREC-COVID queries for direct comparison with published baselines.

3.2 System Architecture

3.2.1 Overview

Our system follows a five-stage pipeline designed to balance effectiveness (ranking quality) with efficiency (query latency) (Figure 1).

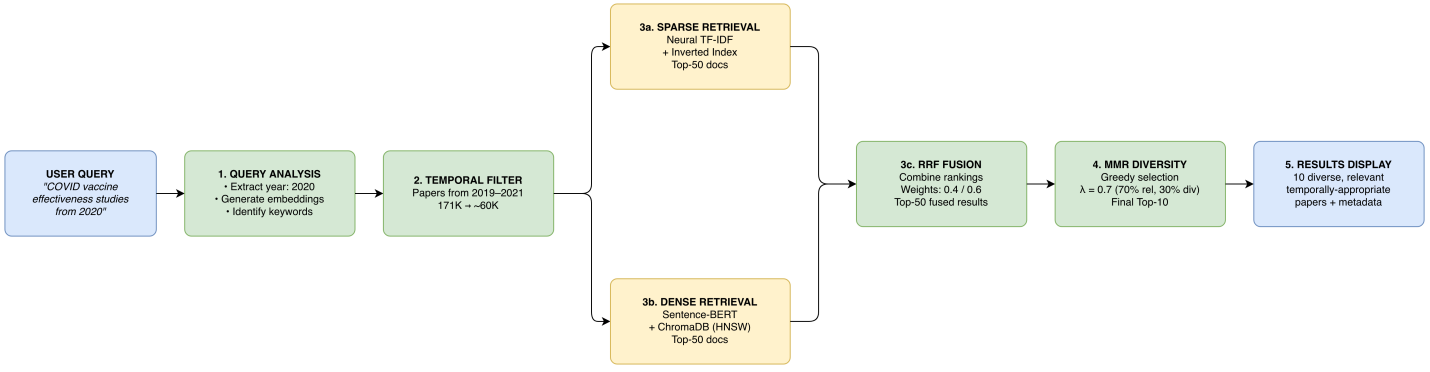

Figure 1: Five-Stage Retrieval Pipeline

System	Description	Purpose
B1: BM25-only	Traditional sparse retrieval	Validates semantic matching need (RQ1)
B2: Dense-only	Sentence-BERT without sparse	Validates lexical matching need (RQ1)
B3: RRF-Full	Sparse + dense + MMR	Primary evaluation target
B4: RRF-NoDiversity	RRF without MMR	Isolates diversity impact (RQ3)
B5: Projection	Random projection fusion	Compares fusion strategies

Table 1: Baseline system configurations for ablation study.

3.2.2 Design Rationale

Stage 1: Query Analysis (Preprocessing). This stage extracts temporal intent and generates representations for downstream retrieval. Temporal expressions are identified using regex for explicit years (e.g., "2020", "2021"), SpaCy NER for relative expressions (e.g., "recent", "last year"), or LLM prompting for more complex temporal reasoning (e.g., "early pandemic studies before vaccine rollout"). Query representations are then generated using Sentence-BERT for dense retrieval and BERT tokenization for sparse retrieval. This unified preprocessing ensures consistent handling of temporal constraints and supports parallel retrieval paths.

Stage 2: Temporal Filter (Search Space Reduction). The temporal filter restricts the corpus to documents published within a relevant time window before retrieval. For queries containing explicit temporal intent, documents within ± 1 year of the query year are selected (e.g., a 2020 query retrieves papers from 2019–2021). This reduces the search space for temporal queries while improving both retrieval speed and temporal relevance.

Stage 3: Hybrid Retrieval (Core Contribution). Hybrid retrieval combines the complementary strengths of sparse and dense methods. Sparse retrieval captures exact lexical matches through neural TF-IDF with contextual term expansion, while dense retrieval uses Sentence-BERT embeddings with ChromaDB approximate nearest neighbour search to capture semantic similarity. Results from the two retrieval paths are then fused using Reciprocal Rank Fusion (RRF) or alternative fusion strategies.

Stage 4: MMR Diversification (Redundancy Reduction). To reduce redundancy in the top results, Maximal Marginal Relevance (MMR) is applied as a reranking step. MMR selects documents that balance relevance to the query with dissimilarity to previously selected documents, improving result diversity. We set $\lambda = 0.7$ to prioritise relevance while still encouraging diversity.

Stage 5: Results Display (User Interface). The final stage presents retrieved documents with key metadata such as title, authors, journal, publication year, snippet, and citation count. Displaying rich metadata helps users assess credibility and temporal validity when reviewing results.

3.3 Retrieval Models

3.3.1 Neural Sparse Retrieval

Traditional TF-IDF assigns static weights to terms and cannot capture contextual relationships. Neural sparse representations such as SPLADE address this limitation by learning contextual term weights while maintaining sparse vectors suitable for efficient inverted indexing [2]. We implement this using the pre-trained SPLADE model `naver/splade-cocondenser-ensembledistil`, or a simplified neural TF-IDF variant using BERT tokenization with `sklearn.TfidfVectorizer`.

3.3.2 Dense Retrieval

Dense retrieval encodes queries and documents into semantic embeddings, enabling matching beyond lexical overlap [5]. Embeddings are generated using `sentence-transformers/all-MiniLM-L6-v2` and indexed in ChromaDB using the HNSW approximate nearest neighbour algorithm [15]. Similarity is computed using cosine distance over the embedding vectors. (The choice of embedding model may impact retrieval performance; alternative models will be evaluated while experimenting.)

3.3.3 Reciprocal Rank Fusion (RRF)

Reciprocal Rank Fusion combines ranked outputs from multiple retrieval systems without requiring score normalization or training data [1]. It provides a simple and robust approach for integrating sparse and dense retrieval results.

3.3.4 Maximal Marginal Relevance (MMR)

MMR reduces redundancy in the final ranking by balancing query relevance with diversity among selected

documents [11].

3.3.5 Projection-Based Fusion

While RRF operates at the rank level, projection-based methods combine signals at the representation level [7]. Projection approaches map sparse and dense vectors into a shared space before retrieval. Possible projection strategies include random projection (unsupervised), SVD/PCA (data-driven), or learned projections trained with supervision.

RRF and projection-based fusion differ in several trade-offs. RRF requires two retrieval paths but no training, while projection-based methods produce a single unified representation that can be searched once. Our system evaluates both approaches by comparing RRF rank fusion with a random projection baseline for combining sparse and dense signals.

4 Implementation Tools

4.1 Core Technology Stack

Our system uses five tools chosen to support the hybrid retrieval architecture. ChromaDB (v0.4.22) [21] is used for dense vector storage and approximate nearest neighbor search using its HNSW index, enabling efficient similarity search over document embeddings. Sentence-BERT (`all-MiniLM-L6-v2`) generates dense sentence embeddings used for semantic retrieval [10]. SPLADE (`naver/splade-cocondenser-ensembledistil`) is used for neural sparse retrieval, expanding contextual term representations while keeping sparse indexing efficient [2]. The interface is built with Streamlit [22], allowing quick development of a simple query interface and results display. The system is implemented in Python 3.9+ [23], which provides the libraries needed for machine learning and information retrieval such as NumPy and scikit-learn.

5 Team Structure

Team Members:

- **Deniz Genco Atilla** – Data Engineering & Preprocessing Lead
- **Harishkumar Kishorkumar Prajapati** – ML/IR Systems & Experimentation Lead
- **Boran Sac** – User Interface & Visualization Lead

6 Implementation Timeline

Total Duration: 4 weeks (28 days, March 10 – April 10, 2026) (Figure 2)

Timeline Summary.

- **Week 1 (Days 1–7):** Phase 1 (Data Prep, 3 days) → Phase 2 begins (Baseline Retrieval, 4 days total)
- **Week 2 (Days 8–14):** Phase 2 completes → Phase 3 (Hybrid Fusion, 3 days) → Phase 4 begins (Diversity Ranking, 3 days total)
- **Week 3 (Days 15–21):** Phase 4 completes → Phase 5 begins (Evaluation & Analysis, 7 days total)
- **Week 4 (Days 22–28):** Phase 5 completes
- **Phases 6–7 (Parallel):** UI Development and Documentation run continuously and incrementally from Day 1–28

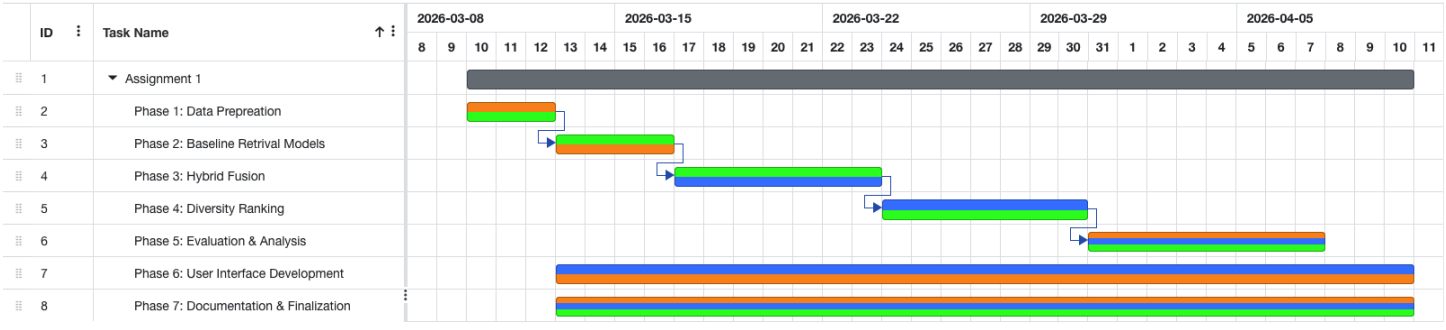

Figure 2: Implementation Timeline – Gantt Chart

7 Conclusion

7.1 Summary

This report has presented the design of a temporal-aware hybrid retrieval system for COVID-19 medical literature search. The system addresses three fundamental challenges in information retrieval: (1) vocabulary mismatch through neural sparse representations (SPLADE-inspired) combined with dense embeddings (Sentence-BERT), (2) temporal relevance through intelligent pre-filtering and temporal intent detection, and (3) result redundancy through diversity-aware ranking using Maximal Marginal Relevance (MMR).

Our five-stage architecture—Query Analysis, Temporal Filter, Hybrid Retrieval (neural sparse + dense via RRF), MMR Diversification, and Results Display—balances effectiveness with efficiency. The system processes the TREC-COVID benchmark (171,332 papers) with expected nDCG@10 > 0.60 while maintaining sub-second query latency using entirely open-source tools (ChromaDB, Sentence-BERT).

Optional Enhancements. Several extensions could further improve system capabilities: (1) LLM-based query refinement for automatic query expansion and reformulation, (2) multi-hop retrieval for following citation chains to discover related work across multiple evidence steps, (3) ColBERT [16] as an alternative dense retriever offering token-level interactions with late interaction architecture for improved precision, (4) NLI-based contradiction detection for identifying conflicting medical findings, and (5) scaling to the full CORD-19 dataset (1M+ papers) with distributed indexing.

References

[1] G. V. Cormack, C. L. A. Clarke, and S. Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of SIGIR*, 2009.

[2] T. Formal, B. Piwowarski, and S. Clinchant. SPLADE: Sparse lexical and expansion model for first stage ranking. In *Proceedings of SIGIR*, 2021.

[3] T. Formal, C. Lassance, B. Piwowarski, and S. Clinchant. SPLADE v2: Sparse lexical and expansion model for information retrieval. *arXiv preprint arXiv:2109.10086*, 2022.

[4] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11), 1987.

[5] V. Karpukhin et al. Dense passage retrieval for open-domain question answering. In *Proceedings of EMNLP*, 2020.

[6] J. Lin, R. Nogueira, and A. Yates. *Pretrained Transformers for Text Ranking: BERT and Beyond*. Synthesis Lectures on Human Language Technologies, 2021.

[7] Y. Luan, J. Eisenstein, K. Toutanova, and M. Collins. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9, 2021.

[8] S. Robertson and H. Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4), 2009.

[9] H. Zamani et al. From neural re-ranking to neural ranking: Learning a sparse representation for inverted indexing. In *Proceedings of CIKM*, 2018.

[10] N. Reimers and I. Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of EMNLP-IJCNLP*, 2019.

[11] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of SIGIR*, 1998.

[12] C. L. Clarke et al. Novelty and diversity in information retrieval evaluation. In *Proceedings of SIGIR*, 2008.

[13] W. Dakka, L. Gravano, and P. Ipeirotis. Answering general time sensitive queries. In *Proceedings of CIKM*, 2008.

[14] R. Jones and F. Diaz. Temporal profiles of queries. *ACM Transactions on Information Systems*, 25(3), 2007.

[15] Y. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE TPAMI*, 42(4), 2018.

[16] O. Khattab and M. Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of SIGIR*, 2020.

[17] Allen Institute for AI. CORD-19: The Covid-19 Open Research Dataset – Release 2022-06-02, 2022. `https://ai2-semanticscholar-cord-19.s3-us-west-2.amazonaws.com/historical_releases.html`.

[18] Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., and Gurevych, I. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Re-

trieval Models, 2021. https://huggingface.co/datasets/BeIR/trec-covid. Accessed: 2025.

[19] Roberts, K., Voorhees, E., and Hersh, W. TREC-COVID: Rationale and Structure of an Information Retrieval Shared Task for COVID-19. National Institute of Standards and Technology (NIST), 2020. https://ir.nist.gov/trec-covid/. Accessed: 2025.

[20] Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., and Gurevych, I. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021. https://huggingface.co/BeIR/query-gen-msmarco-t5-base-v1.

[21] Chroma Core. Chroma: The AI-Native Open-Source Embedding Database, 2024. https://www.trychroma.com.

[22] Streamlit Inc. Streamlit: A Faster Way to Build and Share Data Apps, 2024. https://streamlit.io.

[23] Van Rossum, G. and Drake, F.L. Python 3 Reference Manual. CreateSpace, Scotts Valley, CA, 2009. https://www.python.org.