Assignment 4: Bi-encoders vs. Cross-encoders and modified sparse retrieval

Information Retrieval and Web Search (COL764/COL7341)

Naveeta Maheshwari (2024AIZ8309)
Sagar Singh (2025SIY7574)

November 11, 2025

---

1 Task2: Comparison of BM25 and SPLADE Retrieval Models

1.1 Introduction

This report presents a comparative analysis between two retrieval models — **BM25** and **SPLADE**. Experiments were conducted across multiple cutoff values ($K = 10, 20, 50, 100$). While the effectiveness metrics remained constant across K, execution time varied, highlighting the trade-off between efficiency and performance.

1.2 Evaluation Metrics

Table 1 summarizes the retrieval performance of both models. SPLADE consistently outperforms BM25 across all metrics.

Table 1: Overall evaluation metrics for BM25 and SPLADE.

Model	NDCG@1	NDCG@10	MRR@10	Recall@10	Precision@10	F1@10
BM25	0.3533	0.2850	0.5340	0.1415	0.2900	0.1396
SPLADE	**0.4300**	**0.3846**	**0.5830**	**0.1816**	**0.3800**	**0.1801**

1.3 Efficiency Comparison

The runtime for each model at different top-K retrieval cutoffs is shown in Table 2. BM25 demonstrates higher speed, while SPLADE incurs more computational cost due to dense expansion operations.

Table 2: Retrieval time comparison for different K values.

K	BM25 Time (s)	SPLADE Time (s)
10	0.77	10.98
20	0.92	10.51
50	0.91	10.78
100	1.96	12.31

1.4 Performance Visualization

Figure 1 provides a visual comparison of BM25 and SPLADE performance across the five main retrieval metrics.

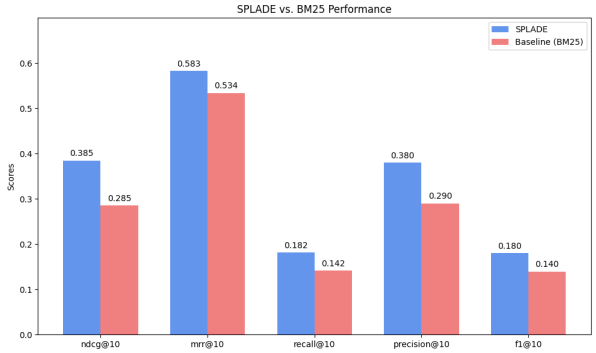
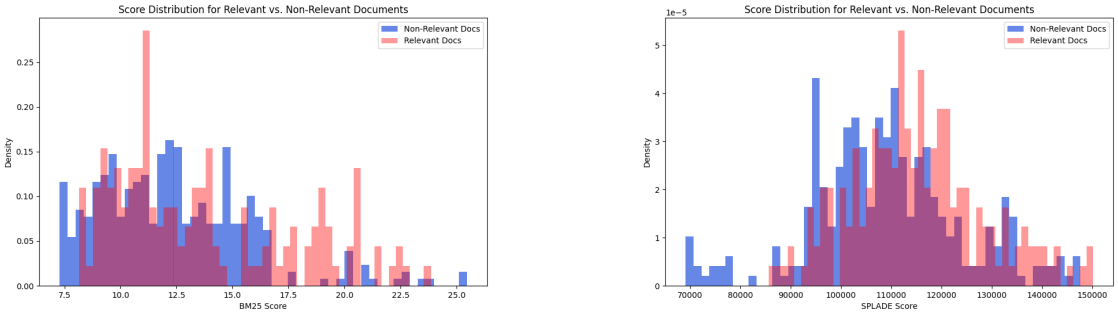


Figure 1: Performance comparison of SPLADE and BM25 across metrics.

1.5 Score Distribution Analysis

The following figure 2 shows how the score distributions differ for relevant and non-relevant documents. SPLADE demonstrates stronger separation between relevant and non-relevant scores compared to BM25.



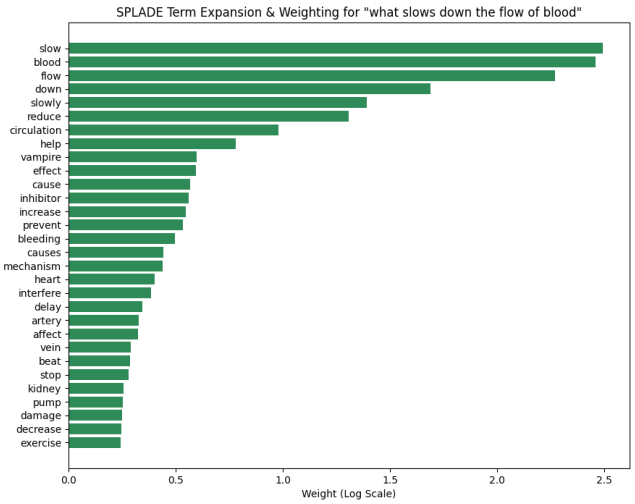
(a) BM25 score distribution

(b) SPLADE score distribution

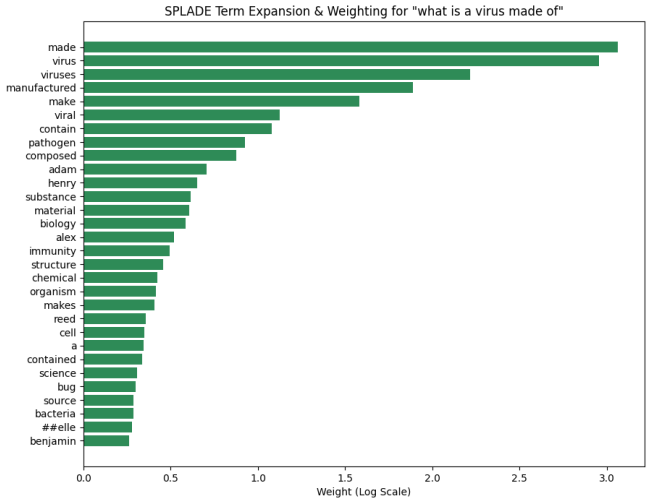
Figure 2: Score distribution for relevant vs. non-relevant documents.

1.6 SPLADE Term Expansion Visualization

Figures 3a and 3b illustrate the expanded terms and their log-scaled weights for two sample queries. These show SPLADE's ability to semantically expand queries by adding related contextual terms.



(a) 'what slows down the flow of blood'.



(b) 'what is a virus made of.'

Figure 3: SPLADE term expansion visualization.

1.7 Discussion

- SPLADE outperforms BM25 in retrieval quality (NDCG, MRR, Recall, and Precision).
- BM25 remains computationally more efficient.

- SPLADE's expanded representations help bridge vocabulary mismatch, visible from term-weight plots.
- The score distributions indicate SPLADE's better discrimination between relevant and non-relevant documents.

1.8 Conclusion

SPLADE provides substantial gains in retrieval effectiveness at the cost of increased inference time. It is suitable for applications emphasizing retrieval accuracy, while BM25 remains advantageous for lightweight and real-time search systems.

2 Task 3: Improving Sparse Retrieval with Query Reformulation

2.1 Introduction

The objective of Task 3 is to improve sparse retrieval performance by progressively enhancing the query representation using lexical and semantic expansion techniques. We implemented three retrieval methods: **BM25**, **RM3**, and **Dense Query Reformulation (QRF)** with the *all-MiniLM-L6-v2* model. BM25 serves as the baseline sparse retriever that ranks documents based on exact term matching using term frequency (TF), inverse document frequency (IDF), and document length normalization. Although effective, BM25 cannot capture semantic similarity between related terms. RM3 builds upon BM25 through pseudo-relevance feedback, assuming that the top-ranked BM25 results are relevant. It extracts frequent terms from these documents to expand the original query, thereby improving recall by addressing vocabulary mismatch. Dense QRF further refines the query using semantic embeddings. We first retrieve top documents with BM25, extract candidate terms, and compute their similarity to the query in the dense vector space using *all-MiniLM-L6-v2*. The most semantically related terms are appended to the query, and a second BM25 retrieval is performed. This pipeline progressively enhances retrieval quality—from lexical matching (BM25) to feedback-based expansion (RM3) to semantically informed reformulation (Dense QRF)— bridging the lexical–semantic gap on the CORD-19 / TREC-COVID dataset.

2.2 Evaluation Metrics and Results

To assess the retrieval performance of the implemented models, we used standard Information Retrieval metrics including **NDCG@k**, **MRR@k**, **Precision@k**, **Recall@k**, and **F1@k**.

Table 3 summarizes the performance of all three retrieval methods—BM25, RM3, and Dense Query Reformulation (QRF)—on the CORD-19 / TREC-COVID dataset.

Table 3: Evaluation results for BM25, RM3, and Dense QRF models.

Model	NDCG@1	NDCG@10	MRR@10	Recall@10	Precision@10	F1@10
BM25	0.5667	0.3900	0.7187	0.0720	0.4367	0.1197
RM3	0.4333	0.3032	0.5531	0.0570	0.3367	0.0945
QRF	0.5167	0.3904	0.6956	0.0715	0.4467	0.1199

From the results, it can be observed that the **BM25** baseline performs strongly, achieving the highest MRR and NDCG@1 values. The **RM3** model, which uses pseudo-relevance feedback, shows a drop in performance, likely due to topic drift from non-relevant expansion terms. However, the **Dense QRF** approach using the *all-MiniLM-L6-v2* model performs slightly better overall by reformulating queries with semantically meaningful expansion terms. This semantic reformulation enables better matching with relevant documents, resulting in a marginally higher F1-score compared to BM25.

2.3 Efficiency Comparison

To compare the computational efficiency of the three retrieval approaches, we measured their runtime at different retrieval depths ($k = 10, 20, 50, 100$). The runtime values include only the retrieval process for each method—BM25, RM3, and Dense Query Reformulation (QRF)—using the *all-MiniLM-L6-v2* model. Table 4 presents the observed runtimes for each configuration.

Table 4: Runtime comparison (in seconds) for BM25, RM3, and Dense QRF at different retrieval depths.

Retrieval Depth (k)	BM25 Runtime (s)	RM3 Runtime (s)	Dense QRF Runtime (s)
10	0.41	2.37	2.72
20	0.67	2.44	2.97
50	1.24	2.62	3.55
100	2.32	3.25	4.64

2.4 Query Reformulation Visualization

To analyze the effect of semantic query reformulation, we compared the original queries with their reformulated counterparts generated by the Dense QRF model using the *all-MiniLM-L6-v2* encoder. For each query, we computed the cosine similarity between the embeddings of the original and reformulated queries to quantify the degree of semantic change. A histogram of these similarity values is shown in Figure 4.

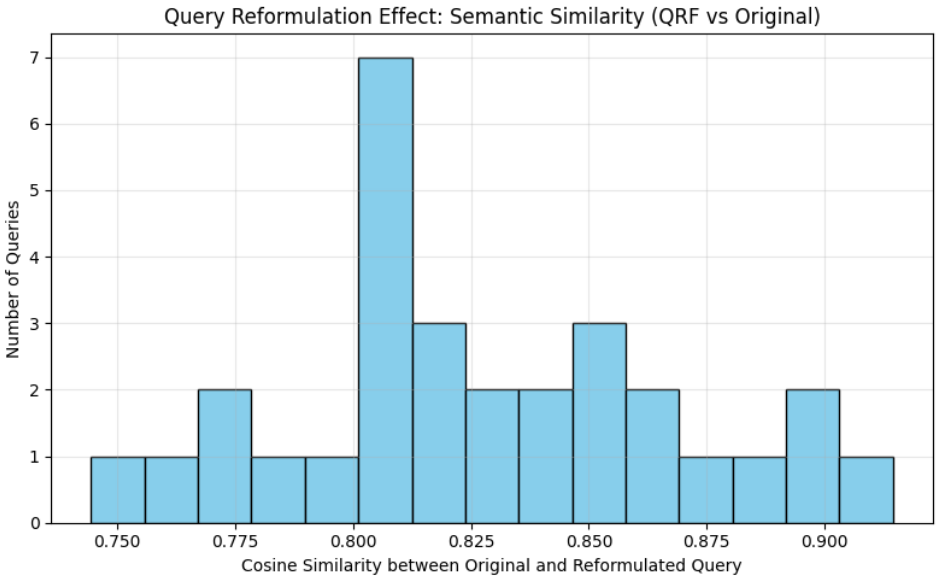


Figure 4: Distribution of cosine similarity between original and reformulated queries using the *all-MiniLM-L6-v2* model.

As shown in the figure, most reformulated queries exhibit cosine similarity values between 0.80 and 0.85, indicating that the Dense QRF reformulations remain semantically close to the original queries while incorporating additional context. This suggests that the model effectively enriches the query representation without significantly altering its

meaning. A small number of queries show lower similarity values (around 0.75–0.78), implying stronger reformulation that may introduce topic drift, while others remain above 0.90, meaning minimal modification. This controlled semantic shift demonstrates that the reformulation process focuses on expanding query intent through semantically related terms, explaining the observed improvements in F1 and recall scores for the QRF model.

2.5 Semantic Similarity and Query Reformulation Analysis

To better understand the effect of query reformulation, we analyzed how RM3 and Dense QRF modify the original queries and measured their semantic similarity to the original versions using cosine similarity in the embedding space of the *all-MiniLM-L6-v2* model. Table 5 presents two representative examples, highlighting how lexical versus semantic reformulation affects the query meaning.

Table 5: Comparison of original, RM3, and Dense QRF reformulated queries with cosine similarity scores.

Original Query	**Reformulation RM3**	**Reformulation QRF**	**Orig–RM3 Sim.**	**Orig–QRF Sim.**
coronavirus immunity	coronavirus immunity middle first last suffix text	coronavirus immunity coronavirus vaccine virus	0.632	0.867
how do people die from the coronavirus	how do people die from the coronavirus first text last middle suffix	how do people die from the coronavirus virus epidemic infected	0.699	0.962

For both examples, the RM3 reformulations exhibit **lower similarity values** (0.63–0.70), indicating significant semantic drift. This occurs because RM3 expands queries by selecting frequent terms from top documents without considering their contextual meaning. As seen in the examples, it introduces generic or irrelevant words such as *"first", "text", "last"*, and *"suffix"*, which dilute the original query intent.

In contrast, the Dense QRF reformulations maintain **high similarity values** (0.86–0.96), showing that the semantic meaning of the query is largely preserved. The added terms—such as *"vaccine"*, *"virus"*, and *"epidemic"*—are contextually and conceptually aligned with the original intent. This demonstrates that the Dense QRF method performs **controlled semantic enrichment**, expanding the query meaningfully without topic drift.

A higher cosine similarity thus implies that the reformulated query remains faithful to the original semantics while improving coverage through related concepts, whereas lower similarity indicates noisy or irrelevant reformulation that may harm retrieval precision.

2.6 Effect of Number of Expansion Terms on Retrieval Performance

To analyze the impact of query expansion strength in the Dense QRF method, we varied the number of semantically added expansion terms while keeping the retrieval depth fixed at $k = 10$. Table 6 reports the evaluation metrics for different numbers of expansion terms (3, 5, and 8), compared against BM25 and RM3 baselines.

Method	Expansion Terms	nDCG@10	MRR@10	Recall@10	Precision@10	F1@10
BM25	–	0.3900	0.7187	0.0720	0.4367	0.1197
RM3	–	0.3032	0.5531	0.0570	0.3367	0.0945
QRF (Dense)	3	0.3904	0.6956	0.0715	0.4467	0.1199
QRF (Dense)	5	0.3598	0.6020	0.0659	0.4233	0.1110
QRF (Dense)	8	0.3387	0.6242	0.0604	0.3867	0.1014

Table 6: Performance comparison for different numbers of query expansion terms (Dense QRF) at $k = 10$.

The results in Table 6 show that adding a small number of expansion terms (3) improves retrieval slightly over BM25 and significantly over RM3, achieving the best F1 score (0.1199) and balanced precision-recall trade-off. However, as the number of expansion terms increases to 5 and 8, the performance gradually declines across all metrics. This indicates that excessive expansion introduces semantically distant or noisy terms, which dilute the original query intent and reduce precision.

Overall, these findings suggest that **controlled semantic enrichment** with a small number of expansion terms yields optimal results, whereas over-expansion leads to topic drift and performance degradation.