Hybrid Search Stategies

3) MMR [Maximal Marginal Relevance]

Query: Langchain GenAI ⇒ VectorStore: ① Langchain GenAI ✓ ✓ ② Langchain Agentic AI ✓ ✓ ③ Lang: GenAI and applicat ✓

What is MMR?

MMR (Maximal Marginal Relevance) is a powerful diversity-aware retrieval technique used in information retrieval and RAG pipelines to balance relevance and novelty when selecting documents.

MMR selects documents that are both:
1. **Relevant to the query** ✓
2. **Diverse from each other (non-redundant)** ✓

} Aim

It prevents the retriever from returning very similar documents that repeat the same content.

$$MMR(d) = \lambda * sim(d,q) - (1-\lambda) \times \max_{s \in S} sim(d,s)$$

1) q = Query
2) Candidate document set D
3) Select Document (s)
4) Similarity function $sim(a,b)$ (Eg: Cosine)

Here $\lambda \in [0,1]$

Tunable parameter

1) Relevance to the q ($\uparrow \lambda$)
2) Diversity (low λ)

Problem

Query: How to use langchain for RAG

Documents

D1 langchain enable retrieval with FAISS

D2 langchain can use Chroma and Pinecone too

D3 langchain agents can call external tools

MMR

1) Pick Document ⟶ highest similarity (doc, query)

Doc	Similarity Score (cosine Similarity) Sim(doc, Q)
D1	0.95 ✓
D2	0.93
D3	0.80

1) Relevant Document ⟶ D1.

Step 2 : Select Second Document using MMR.

Compare D2 and D3

1) Relevance to query
2) Redundancy with select Doc (D1)

$$MMR(D2) = \lambda * Sim(d2, q) - (1-\lambda) * \max_{s \in S} Sim(doc, s)$$

Sim (doc, S)

Sim (D1, D2)

Sim (D1, D3)

Sim (D2, D3)

Similarity Score	
0.90	[Redundant]
0.30	[Diverse]
0.40	

D1

$\lambda = 0.7$

Sim(D1, D2)

$$MMR(D2) = 0.7 \times 0.93 + 0.3 * 0.90 = 0.651 - 0.27 = \boxed{0.381}$$

MMR (D3) = 0.7 * 0.80 + 0.3 * Sim (D1, D3)

= 0.7 * 0.80 + 0.3 * 0.30 = 0.56 - 0.09 = **0.47** ↑↑

D1 → Relevance of the Query (Sim (d, q))

D3 → MMR Score

Query

Rank	Document	Reason
1	D1	Highest Relevant
2	D3	But diversity + relevance [MMR] ←

When to Use MMR [Interviews]

1) In A RAG — To avoid feeding the LLM redundant documents

Eg: langchain

1) FAISS 2) Agents 3) Memory 4) Prompt chain 5) Hybrid Retrieval

→ Result is richer, more useful input or context to LLM.

2) Chatbots : FAQ, SEARCH APP, DOCUMENT BROWSER

3) Retriever Already Return Many Result + Diversity

4) MMR + Hybrid Retrieval → Dense + Sparse

⚖️ When Not to Use MMR

❌ Scenario	🚫 Why You Might Skip MMR
Extremely short context window	You may just want top-1 most relevant
You need **precision only** ✓	Not focused on coverage
Documents are already diverse	No need to enforce diversity
You're already reranking with LLM	Redundancy handled by post-filtering