Optimizing Vector Search Performance

- Issue: Queries take around 5 minutes due to metadata API delays.
- Proposed Fixes:
    - Preload metadata into a structured SQL/NoSQL database instead of calling the API on every query.
    - Parallelize metadata retrieval using asynchronous calls.
    - Implement multi-vector search (title, date, abstract in separate indexes).

Preloading Metadata for Faster Search

- Store metadata locally to eliminate repeated API calls.
- Use SQLite or PostgreSQL to store metadata with quick lookups.
- Sync metadata periodically to ensure freshness.
- Modify retrieval function to fetch metadata from local storage instead of API calls.
- Blocker: Do not know if our local computers can handle this, as it is already very intensive to run the task

Alternative Vector Stores to Pinecone

- Problem: Pinecone's cloud storage incurs high costs and slows down with large datasets.
- Alternatives:
    - FAISS: Local vector search with HNSW indexing (faster & cheaper).

Performance Criteria

- Reduce query latency from current times of around 5 minutes.
- Improve ranking for title & date-based queries (Mean Reciprocal Rank ≥ 0.7).
- Lower Pinecone costs by 50% using FAISS or hybrid retrieval.

Next Steps

- Test out implementation with metadata preloading in local storage.
- Test FAISS as an alternative to Pinecone for vector search.
- Optimize retrieval pipeline with hybrid search.