Multi-Modal RAG-Based QA System
Technical Report

November 26, 2025

1 Architecture Summary

The Multi-Modal RAG-Based QA System is a Retrieval-Augmented Generation system for question-answering over PDFs containing text, tables, images, and charts. The pipeline processes documents through ingestion (PyMuPDF, Camelot, Tesseract OCR), chunking (600-character chunks with 100-character overlap), and embedding generation (TF-IDF, Word2Vec, SBERT). The hybrid retrieval system combines three parallel retrieval methods using Reciprocal Rank Fusion (RRF) or weighted sum fusion to produce top-20 candidates. Cross-encoder reranking refines these to top-5 passages, which are filtered by relevance threshold (0.30) and passed to an LLM for answer generation. The system also supports summarization and uses a centralized `RetrievalArtifacts` dataclass containing all embeddings, indices, and models. All artifacts are cached to disk, reducing build time from minutes to seconds on subsequent runs.

2 Design Choices

The three-method hybrid retrieval (TF-IDF, Word2Vec, SBERT) was selected based on evaluation studies comparing 13 embedding approaches. While TF-IDF + Word2Vec achieved best individual performance (MRR: 0.502, NDCG: 0.454, F1: 0.232), all three methods are combined for complementary strengths: TF-IDF for exact keyword matching, Word2Vec for local semantics, and SBERT for contextual understanding. The system uses Reciprocal Rank Fusion (RRF) with $k = 60$ as default, or weighted sum (TF-IDF 35%, Word2Vec 30%, SBERT 15%, cross-encoder 20%). A cross-encoder reranker (`cross-encoder/ms-marco-MiniLM-L-6-v2`) is applied to top-20 candidates for precision improvement and hallucination prevention via relevance threshold filtering. The 600-character chunk size with 100-character overlap balances context preservation, retrieval granularity, and LLM context limits. Multi-modal extraction (text via PyMuPDF, tables via Camelot, images via Tesseract OCR) unifies all content into a common chunk structure for seamless retrieval.

3 Benchmarks

Evaluation of 13 retrieval methods showed TF-IDF + Word2Vec achieving best performance (MRR: 0.502, NDCG: 0.454, F1: 0.232), outperforming dense models like SBERT (MRR: 0.399) and MPNet (MRR: 0.439). Sparse methods excel due to high lexical repetition, technical terminology, OCR robustness, and page-based structural relevance rather than semantic similarity. Performance metrics include hybrid retrieval (50-200ms), reranking (100-300ms), and LLM generation (1-3s for 0.5B models, 5-15s for 7B models), with total latency ranging 2-20 seconds. The system evaluates using MRR, NDCG, and Precision/Recall/F1 with page-based relevance where same-page chunks are considered relevant.

4 Key Observations

Contrary to common assumptions, sparse methods (TF-IDF, Word2Vec) outperformed dense transformers due to semantic similarity collapse in dense space, OCR noise amplification, and structural relevance (page-based) rather than semantic. The hybrid approach provides robustness by combining methods to reduce failure modes. Design trade-offs include speed vs. accuracy (reranking only top candidates), memory vs. performance (FAISS for SBERT, sparse TF-IDF), and caching vs. freshness (fast loads require explicit rebuilds). Practical considerations: OCR quality impacts performance, 600-character chunks balance granularity and context, relevance threshold 0.30 prevents hallucinations while maintaining coverage, and 0.5B models balance speed and quality. Key lessons: empirical evaluation is essential, dataset characteristics determine optimal methods, hybrid approaches provide robustness, and modularity enables experimentation. Future improvements include adaptive chunking, query expansion, relevance feedback, multi-stage reranking, and domain-specific embedding fine-tuning.