**Build a RAG-based Chatbot for AMC Fund Factsheet.**

**High-Level Architecture**

┌─────────────────────────────────────────────────────────────────┐

│ USER INTERFACE │

│ (Gradio Web Interface) │

└─────────────────────────────────────────────────────────────────┘

│

▼

┌─────────────────────────────────────────────────────────────────┐

│ FASTAPI SERVER (Modal Labs) │

│ L40S GPU Instance │

└─────────────────────────────────────────────────────────────────┘

│

┌─────────────────────┼─────────────────────┐

▼ ▼ ▼

┌──────────────────┐ ┌──────────────┐ ┌──────────────────┐

│ PDF PROCESSING │ │ RETRIEVAL │ │ GENERATION │

│ │ │ │ │ │

│ LlamaParse API │ │ Qdrant │ │ Llama 3.1-8B │

│ ↓ │ │ Vector DB │ │ (4-bit Quant) │

│ Financial │ │ ↓ │ │ │

│ Keyword Filter │ │ BGE-small │ │ │

│ ↓ │ │ Embeddings │ │ │

│ Table-Aware │ │ ↓ │ │ │

│ Chunking │ │ Hybrid Search │ │ │

│ ↓ │ │ Top-K=15 │ │ │

│ BGE Embeddings │ │ │ │ │

│ ↓ │ │ │ │ │

│ Store in Qdrant │ │ │ │ │

└──────────────────┘ └──────────────┘ └──────────────────┘

│

▼

┌──────────────────┐

│ Redis Cache │

│ (Optional Fast │

│ Responses) │

└──────────────────┘

**Processing Pipeline**

**1. Document Indexing Flow**

PDF Upload

│

▼

LlamaParse API (Cloud)

│ (extracts tables and text)

▼

Financial Keyword Detection

│ (filters relevant financial content)

▼

Table-Aware Chunking

│ (preserves table integrity, 400-7200 chars/chunk)

▼

BGE-small-en-v1.5 Embedding Generation

│ (768-dim vectors)

▼

Qdrant Vector Storage

│

▼

Ready for Queries

**Output:** 140+ tables extracted → 838 searchable chunks

**2. Query Processing Flow**

User Query

│

▼

Check Redis Cache

│

├─ Cache Hit → Return Cached Response (80ms)

│

└─ Cache Miss

│

▼

Generate Query Embedding (BGE-small)

│

▼

Qdrant Hybrid Search (Top-K=5)

│

▼

Retrieve Relevant Chunks

│

▼

Assemble Context + System Prompt

│

▼

Llama 3.1-8B Generation

│

▼

Store in Redis Cache

│

▼

Return Response (1.8-2.2s total)

**Component Details**

**PDF Processing Layer**

* **LlamaParse API**: Specialized financial document parser
* **Financial Keywords**: Filters chunks containing fund-specific terms
* **Table-Aware Chunking**: Preserves complete tables in single chunks (avg 2000 chars)

**Embedding & Retrieval Layer**

* **Model**: BAAI/bge-small-en-v1.5
* **Dimensions**: 768
* **Vector DB**: Qdrant with HNSW indexing
* **Search Strategy**: Hybrid (semantic + keyword matching)
* **Top-K**: 5 chunks retrieved per query

**Generation Layer**

* **Model**: Llama 3.1-8B (4-bit quantized)
* **Context Window**: 8K tokens
* **Quantization**: 4-bit for memory efficiency (16GB → 4.5GB)
* **GPU**: L40S on Modal Labs serverless infrastructure

**Infrastructure**

* **Deployment**: Modal Labs (serverless GPU)
* **Caching**: Redis for repeated queries
* **API**: FastAPI for request handling
* **Frontend**: Gradio for user interface

**Data Flow Summary**

1. **Indexing**: PDF → LlamaParse → Chunks → Embeddings → Qdrant (one-time)
2. **Query**: User Input → Embedding → Search → Context → LLM → Response (per query)
3. **Optimization**: Redis caching reduces repeat query latency by 95%

**System Metrics**

| **Metric** | **Value** |
| --- | --- |
| Tables Extracted per PDF | 140+ |
| Total Chunks Generated | 838 |
| Average Chunk Size | 2000 characters |
| Embedding Dimensions | 768 |
| Vector Retrieval Time | <50ms |
| LLM Generation Time | 1.2-1.5s |
| Total Query Time (uncached) | 1.8-2.2s |
| Total Query Time (cached) | 80-120ms |
| GPU Memory Usage | 6.5GB / 48GB |

**Tech Stack**

| **Layer** | **Technology** |
| --- | --- |
| **PDF Parsing** | LlamaParse API |
| **Chunking** | Custom table-aware splitter |
| **Embeddings** | BGE-small-en-v1.5 |
| **Vector Database** | Qdrant |
| **LLM** | Llama 3.1-8B (4-bit) |
| **API Server** | FastAPI |
| **Deployment** | Modal Labs |
| **Caching** | Redis |
| **Frontend** | Gradio |