Insurance Claim Timeline Retrieval System

System Overview: Multi-agent RAG system using hierarchical indexing (3-level chunking: 2048/512/128 tokens), MapReduce summarization, and MCP tool integration for analyzing insurance claim timelines with high precision and contextual understanding. Architecture diagram available in README.md.

Core Components

- **Manager Agent (gpt-4o):** Routes queries to specialized experts with claim validation
- **Summary Expert (gpt-4o-mini):** High-level overviews using SummaryIndex with tree_summarize
- **Needle Expert (gpt-4o-mini):** Precise fact retrieval using AutoMergingRetriever
- **MCP Tools:** Policy limit validation and date calculations

Data Indexing

- **Dataset:** 3 synthetic insurance claims (auto collision, water damage, theft) totaling ~15 pages
- **Hierarchical Structure:** Root (2048), Intermediate (512), Leaf (128 tokens)
- **Storage:** ChromaDB with metadata filtering by claim_id
- **Auto-Merging:** 40% sibling threshold for dynamic context expansion

Evaluation Results (LLM-as-a-Judge with gpt-4o)

Category	Tests	Correctness	Relevancy	Faithfulness
Needle Questions	16	4.73/5.0	0.97/1.0	0.98/1.0
Summary Questions	8	4.81/5.0	0.96/1.0	0.99/1.0
MCP Tool Usage	5	4.90/5.0	1.0/1.0	1.0/1.0
Overall Average	**29**	**4.78/5.0**	**0.98/1.0**	**0.99/1.0**

MCP Integration

Tool: validate_policy_limit(claimed_amount, policy_limit)
Purpose: Validates claims against coverage limits with risk assessment
Integration: FastMCP server wrapped as FunctionTool for Manager Agent

Example Query: "Is $9,766.90 within a $100,000 limit?"
Tool Response: Within limits (9.8% used), $90,233 remaining, Low risk
Success Rate: 100% (5/5 MCP tests passed with perfect scores)

Key Design Decisions

Hierarchical Chunking: Three-level structure (128/512/2048 tokens) enables both precision retrieval and contextual understanding. AutoMergingRetriever dynamically expands from leaf nodes to parent nodes when 40% of siblings are retrieved, providing context on-demand.

MapReduce Summarization: tree_summarize mode implements map-reduce pattern for efficient high-level queries without long-context processing. Each chunk summarized independently then combined into final response.

Metadata Filtering: LLM-extracted claim_id enables targeted retrieval from specific claims, improving precision and reducing noise. Smart claim resolution system handles ambiguous queries.

Complete documentation and architecture diagram available in README.md