# Multi-Agent AI System

# Dynamic Decision Making Architecture

## Technical Implementation Report

## Table of Contents

## 1. Introduction

This report details the implementation of a multi-agent AI system with dynamic decision making capabilities. The system is designed to intelligently route user queries to specialized agents based on the content and context of the query. The architecture includes a controller agent that makes routing decisions, and specialized agents for PDF document processing, web search, and academic paper retrieval.

## 2. System Architecture

The system follows a modular architecture with clearly defined components:

| Component | Description |
|---|---|
| Controller Agent | Orchestrates the system and routes queries to appropriate agents |
| PDF RAG Agent | Processes PDF documents and enables semantic search capabilities |
| Web Search Agent | Performs real-time web searches for current information |
| ArXiv Agent | Retrieves and summarizes recent academic papers |
| Frontend | Provides user interface for queries and PDF uploads |
| Backend API | FastAPI-based REST API serving all endpoints |
| Logging System | Tracks all interactions and decisions for traceability |

# 3. Agent Implementation

### *PDF RAG Agent*

The PDF RAG (Retrieval-Augmented Generation) agent processes uploaded PDF documents using the following workflow: 1. Text Extraction: Uses PyMuPDF (fitz) library to extract text from PDF files 2. Text Chunking: Splits documents into 500-token chunks with 50-token overlap 3. Embedding Generation: Uses SentenceTransformer 'all-MiniLM-L6-v2' to create 384-dimensional embeddings 4. Vector Storage: Stores embeddings in FAISS vector store for efficient similarity search 5. Retrieval: Performs nearest neighbor search to find relevant document chunks

### *Web Search Agent*

The Web Search agent provides real-time information retrieval using: 1. DuckDuckGo Instant Answer API for primary search functionality 2. Result parsing and filtering to extract relevant information 3. Summarization of search results for concise responses

### *ArXiv Agent*

The ArXiv agent specializes in academic paper retrieval: 1. Queries the official ArXiv API for recent papers 2. Extracts paper metadata including titles, abstracts, and authors 3. Provides concise summaries of relevant papers

# 4. Controller Decision Logic

The Controller agent uses a hybrid approach combining rule-based logic and LLM-based decision making: Rule-Based Routing: - Queries containing "pdf" or "document" → PDF RAG Agent - Queries containing "recent papers", "arxiv", or "paper" → ArXiv Agent - Queries containing "latest news" or "recent developments" → Web Search Agent - Default routing → Web Search Agent The controller logs all decisions including: - Input query - Routing decision with rationale - Agents called - Documents retrieved - Final synthesized response

# 5. Security and Privacy

The system implements several security and privacy measures: 1. File Upload Security: - Maximum file size limited to 10MB - File type validation (PDF only) - Temporary storage with automatic cleanup 2. Data Privacy: - No PII storage in logs - Encrypted data transmission (HTTPS) - GDPR-compliant data handling 3. API Security: - Rate limiting for all endpoints - Input validation and sanitization

# 6. Deployment

The system is designed for deployment on Hugging Face Spaces: 1. Containerization using Docker with the provided Dockerfile 2. Environment variable configuration for API keys 3. Automatic processing of sample PDFs on startup 4. Health checks and monitoring capabilities

# 7. Limitations and Future Work

Current Limitations: - Rule-based routing could be enhanced with more sophisticated LLM-based decision making - PDF processing is limited to text extraction (no image or table processing) - Web search results depend on the quality of the search API Future Enhancements: - Integration with additional LLM APIs (Groq, Google AI Studio) - Advanced PDF processing including image and table extraction - User feedback mechanisms to improve routing decisions - Multi-language support