Intelligent Campus Information Hub

Final Year Project Presentation

Raz Muhammad (135) • Abdur Rauf Shah (124) • Imran Ullah (107)

BS (CS), Session: 2021-2025

Institute of Computer Science and Information Technology

The University of Agriculture, Peshawar, Pakistan

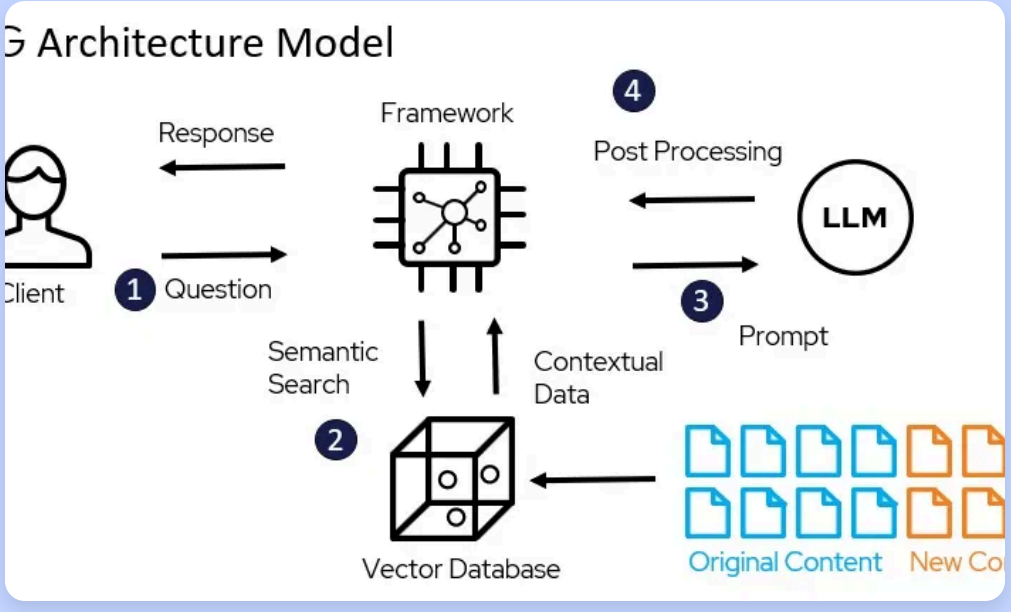
Supervised by: Mr. Imran Uddin

AUP
2025

Abstract

An Agentic Retrieval-Augmented Generation (RAG) system tailored as a centralized information platform for the Agricultural University Peshawar.

- Leverages textual data from university website, processed and stored in Pinecone vector database
- Innovative agentic framework for intelligent query analysis and routing to specialized tools
- Methodology: data scraping, text segmentation, embedding generation, system integration
- Utilizes APIs (Groq, Tavily, Pinecone) and LangChain framework
- Versatile and precise system for diverse query types, enhancing information accessibility



Project Objectives

Centralized Information Access

Create a unified platform for accessing all AUP information resources efficiently and intuitively.

Intelligent Query Handling

Develop a system that can interpret user queries and route them to suitable response mechanisms (university-specific, web search, general knowledge).

Demonstration of RAG Potential

Showcase the effectiveness of Retrieval-Augmented Generation in an educational setting, with open-source code on GitHub.

Methodology: Data Scraping

Data Scraping Process

Website Crawling

Started at AUP homepage, recursively exploring linked pages

Tracking mechanism to prevent revisiting URLs

200-page cap with 2-second delay between requests

Content Extraction

Extracted title and body text as primary content

HTML parsing focused on the <body> section

Data Cleaning

Consolidated whitespace and line breaks

Removed irrelevant details (contact info, etc.)

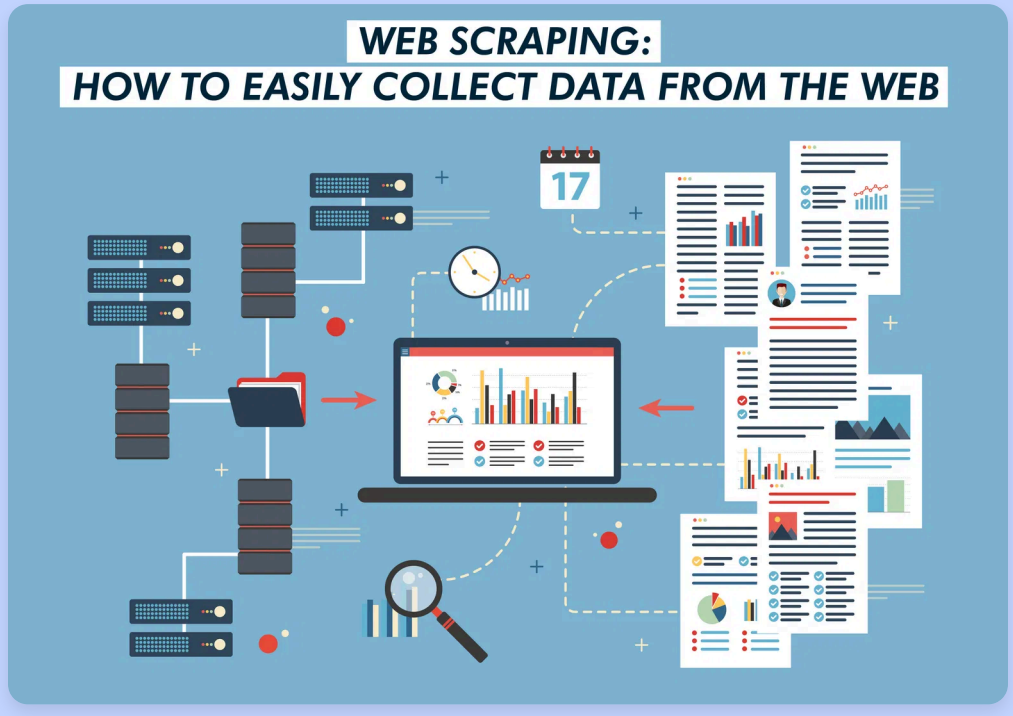
Data Consolidation

Unified format (JSON/structured text)

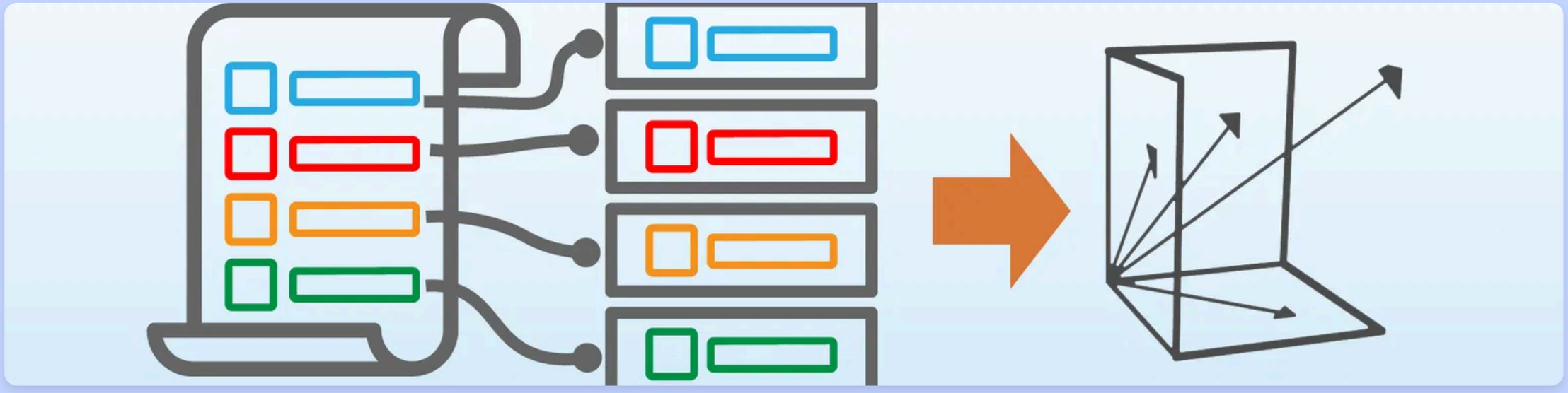
Prepared for embedding and storage components

Tools & Technologies

BeautifulSoup | Requests | Python | Selenium



Methodology: Text Processing



Text Chunking

- Broke down documents into smaller segments for relevance and context
- Used recursive character text splitter
- Optimal chunk size: ~1000 characters
- Overlap: 100 characters for context preservation

Embedding Generation

- Converted text chunks into numerical vector representations
- Used state-of-the-art embedding models
- Captured semantic meaning in high-dimensional space
- Similar text = closer vectors

Vector Storage

- Stored embeddings in Pinecone vector database
- Enabled efficient similarity search
- Indexed with metadata (URL, section title)
- Fast approximate nearest neighbor (ANN) searches

Key Challenge

Determining optimal chunk size and overlap to balance context preservation with search precision

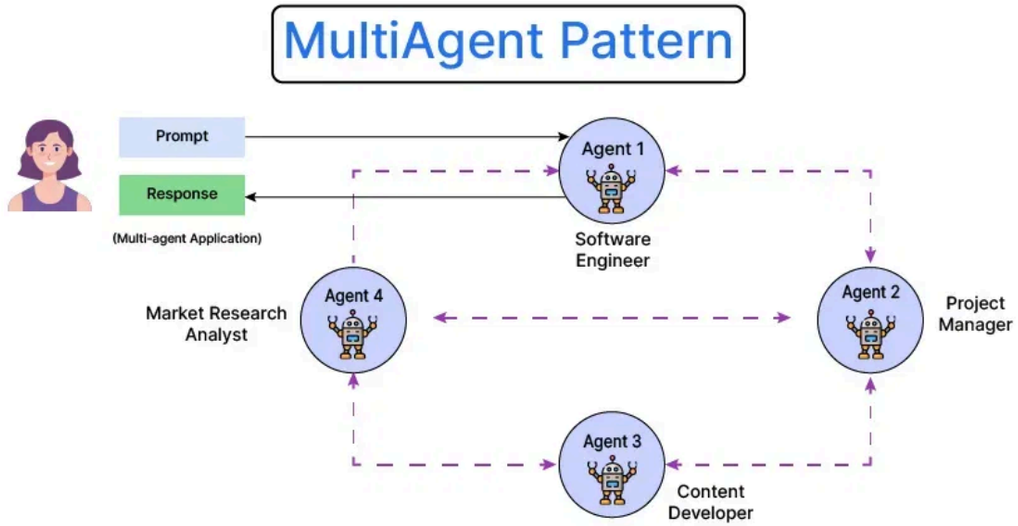
Resolution

Extensive experimentation with various sizes (500-1500 chars) and overlaps (10-20%)

Tools Used

LangChain for text splitting, OpenAI embeddings, Pinecone for vector storage

Methodology: Agentic RAG System



Operational Workflow

User Query → Query Classification → Agent Selection → Information Retrieval → Response Generation → Response Delivery

Specialized Agents

AUP-Specific Agent

Handles university-related queries by retrieving information from the Pinecone vector database containing AUP website content.

Web Search Agent

Retrieves external or real-time information from the web when queries require data beyond the university's scope.

General Query Handler

Utilizes the language model's inherent knowledge for general questions that don't require specific university data or web search.

Tools & Technologies

Intelligent Campus Information Hub

AI & Language Models

Groq
High-performance inference API for language model integration

LangChain
Framework for developing applications powered by language models

OpenAI Embeddings
Text embedding models for semantic understanding

Data Storage & Retrieval

Pinecone
Vector database for efficient similarity search and retrieval

JSON
Lightweight data interchange format for structured data

Web Scraping & Search

BeautifulSoup
Library for parsing HTML and XML documents

Requests
HTTP library for making web requests

Tavily
Search API for retrieving external information

Development Tools

Python
Primary programming language for system development

Selenium
Browser automation for dynamic content scraping

Source Code: https://github.com/RazMuhammad/UniAgent

Limitations & Future Work

Current Limitations

Data Freshness

Reliance on scraped data that may not always be up-to-date with the latest university information.

Ambiguity in Queries

Less precise responses for vague or ambiguous user queries requiring additional clarification.

Cost of Operations

Language models and vector databases incur significant computational and financial costs for maintenance.

Modality Limitations

Primarily text-based system that cannot interpret or process images, videos, or audio content.

Future Work

Real-time Data Integration

Direct synchronization with university website backend for always up-to-date information.

Multi-modal RAG

Incorporate image and video analysis capabilities to process visual university content.

Proactive Information Delivery

Anticipate user needs and deliver relevant information like personalized alerts for students.

Enhanced Personalization

Implement user profiles and preferences to tailor responses and information delivery.

Conclusion

The Intelligent Campus Information Hub, powered by an Agentic RAG system, revolutionizes information access in educational institutions by combining robust data scraping, advanced text processing, and intelligent query routing to offer a versatile and precise solution for diverse information needs.

Centralized Knowledge Repository

Intelligent Agentic Framework

Enhanced Information Retrieval

Open-Source Implementation

Thank You!

GitHub: https://github.com/RazMuhammad/UniAgent

AUP
2025