JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 30 No. 7, pp. 79-86, July 2025
https://doi.org/10.9708/jksci.2025.30.07.079

Design of a RAG-based Q&A System for University Academic Administration Assistants

Jae-Kwang Oh*, Soo Kyun Kim**

*Student, Dept. of Computer Engineering, Jeju National University, Jeju, Korea
**Professor, Dept. of Computer Engineering, Jeju National University, Jeju, Korea

[Abstract]

This paper aims to implement a Retrieval-Augmented Generation based question-answering system to efficiently process and utilize various unstructured academic administrative documents, such as university regulations, course catalogs, and admissions guidelines. To achieve this, a document parsing and preprocessing pipeline was designed to accurately interpret the structure of PDF and HWP formats, enabling generative AI models to comprehend context and generate natural responses. To overcome the limitations of large language models (LLMs), such as token constraints and vector-based similarity search inefficiencies, optimal chunking and embedding strategies were developed. Furthermore, a hybrid retrieval method was integrated using the LangChain framework to improve response accuracy and reliability. The proposed system provides university members with fast and precise access to institutional information, enhancing administrative efficiency and offering a practical reference for future AI-driven knowledge retrieval systems in higher education and public institutions.

▸ **Key words**: RAG, LangChain, LangGraph, Document Preprocessing Pipeline, Ranking Mechanism, Question Answering Pipeline

[요 약]

본 연구는 대학의 학사규정, 수강편람, 입시요강 등 다양한 비정형 학사 행정 문서를 효율적으로 처리하고 활용하기 위한 검색-증강-생성(Retrieval-Augmented Generation, RAG) 기반 질의응답 시스템을 구현하는 것을 목표로 한다. 이를 위해 PDF 및 HWP 형식 문서의 구조를 정확히 파악하는 파싱 및 전처리 절차를 설계하였으며, 생성형 AI가 문맥을 이해하고 자연스러운 응답을 생성할 수 있도록 시스템을 구성하였다. LLM의 토큰 수 제한 및 벡터 기반 유사도 검색의 한계를 극복하고자, 문서 청킹 전략과 임베딩 방식을 최적화하고, 랭체인(LangChain)을 활용한 다중 검색 기법을 통합하여 응답의 정확도와 신뢰도를 높였다. 제안된 시스템은 대학 구성원에게 필요한 정보를 신속하고 정확하게 제공함으로써 행정 업무의 효율성을 제고하며, 향후 국내 대학 및 공공기관의 AI 기반 지식 검색 시스템 구축에 실질적인 참고가 될 수 있다.

▸ **주제어**: RAG, 랭체인(LangChain), 랭그래프(LangGraph), 문서전처리 파이프라인, 순위결정 메카니즘, 질의응답 파이프라인

• First Author: Jae-Kwang Oh, Corresponding Author: Soo Kyun Kim
*Jae-Kwang Oh (ojk0533@jejunu.ac.kr), Dept. of Computer Engineering, Jeju National University
**Soo Kyun Kim (kimsk@jejunu.ac.kr), Dept. of Computer Engineering, Jeju National University
• Received: 2025. 06. 23, Revised: 2025. 07. 17, Accepted: 2025. 07. 21.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

1.1 Research Background

최근 인공지능(AI) 기술의 발전과 함께 AI 에이전트 시대가 도래하면서, 대학 업무 및 학생 지원을 위한 AI 활용 방안이 다각도로 모색되고 실제 적용 사례들이 증가하고 있다. 블룸버그 인텔리전스(Bloomberg Intelligence)에 따르면 생성형 AI 시장은 2032년까지 1.3조 달러 규모로 성장할 것으로 예측되는 등 [1] AI 기술은 전 산업 분야에 걸쳐 혁신을 주도하고 있으며, 이는 교육 분야에도 예외 없이 적용되고 있다. 국내 여러 대학들은 이러한 변화에 발맞춰 AI 기반 시스템을 적극적으로 도입하고 있으며, 맞춤형 강의 추천 시스템, 정보 알림 챗봇, 학사 공지 챗봇, 적응형 학습 시스템 등 다양한 형태로 구축되고 있다. 이는 대학가의 전방위적인 AI 혁명을 시사한다. 구체적인 국내 대학 AI 도입 사례를 살펴보면 Table 1과 같다. 고려대학교는 2020년 8월경 AI 기반 맞춤형 강의 추천 시스템 ‘A.I. 선배’를 교내 포털에 조기 적용하고, 2021년 8월에는 베타버전을 출시하여 현재까지 운영 중이다[2]. 부산대학교는 2023년 4월 경 AI 챗봇 ‘산지니’를 구축하여 학생들에게 편의를 제공하고 있으며 [3], 충남대학교는 2025년 4월 경 마이크로소프트 애저(Azure) 클라우드 기반의 AI 챗봇 'AI 차차'를 도입하여 24시간 질의응답 서비스를 제공하고 있다[4]. 울산대학교는 2025년 2월 경 AI 학사상담 시스템 ‘U-MATE’ 개발하였다고 발표하였다. 핵심 기술로는 RAG 기술이 적용되었고 답변할 수 없는 질문을 ‘답변 불가’로 응답해 잘못된 정보를 생성하는 ‘환각 현상(Hallucination)’ 문제를 최소화 하였고, 사전 테스트 결과 92% 정확도를 보였다고 한다 [5].

Table 1. AI System Adoption in South Korean Universities

University	Type of AI System	Adoption	Environment
Korea University	Personalized Lecture Recommendation System	2020	Integrated with campus Portal System
Pusan National University	AI Chatbot 'Sanjini'	2023	Unknown
Chungnam National University	AI Chatbot 'Chacha'	2025	Microsoft Azure Cloud
University of Ulsan	Ai Academic Advising System 'U-MATE'	2025	RAG

이러한 AI 시스템의 도입은 대학의 디지털 전환을 가속화하며 업무 효율성을 높이는 데 기여하고 있으나, 실제 사용자들의 상세한 후기나 에브리타임과 같은 플랫폼에서의 심층적인 운영 평가는 공개적으로 찾기 어려운 실정이다. 일반적으로 AI 질의응답 시스템의 성공적인 운영을 위해서는 답변의 정확도, 관리자와 사용자의 편리한 이용 환경, 그리고 정보 보안의 안정성과 신뢰성이 핵심 요인으로 간주된다 [6].

국내 대부분의 공공기관과 대학에서는 다양한 행정 문서를 아래아한글(HWP) 형식으로 작성·배포하고 있다. 그러나 ChosunBiz [7]에 따르면, HWP는 국제 표준을 따르지 않아 챗GPT 등 대규모언어모델(LLM)이 이를 데이터화하기 어렵고, 과거 문서는 변환 작업이 필요하다고 지적된다. 대학 내 RAG(Retrieval-Augmented Generation) 시스템을 적용하기 위해선 다양한 형태의 HWP로 작성된 학사 행정 문서를 파싱 기술이 반드시 요구된다. 문서 파싱은 비정형 문서에서 정보를 자동 추출하고 구조화하여 기계가 이해할 수 있도록 하는 기술로, RAG 기반 질의응답의 핵심 전처리 과정이다.

HWP는 바이너리 포맷 특성상 구조 해석이 어렵고, 스캔된 PDF는 OCR(Optical Character Recognition) 없이는 처리할 수 없기 때문에, 고성능 파서의 활용이 필수적이다. 업스테이지 도큐먼트 파서(Upstage Document Parse)는 이미지 기반 문서에서 텍스트를 정밀 추출하고 문서 구조를 분석해 LLM이 활용 가능한 형태로 변환한다. 또한 다양한 질의응답 시나리오 구성과 LLM 모델 연동을 위해 RAG 체인을 효과적으로 설계할 수 있는 오픈소스 도구가 필요하다. 랭체인(LangChain)은 벡터 검색, 문서 인코딩, 프롬프트 템플릿 등을 모듈화하여 제공하며, 랭그래프(LangGraph)는 노드-에지 기반 흐름 제어를 통해 복잡한 질의응답 흐름 설계를 지원한다.

본 연구는 단순한 AI 응답 생성이 아닌, 실제 대학 행정 문서의 복잡한 구조와 다양한 형식(HWP, PDF 등)을 효과적으로 처리하고, 사용자 질의에 대해 정확하고 신뢰도 높은 답변을 생성할 수 있는 RAG기반 질의응답 시스템을 구현하고자 하였다. 기존의 키워드 기반 챗봇 시스템은 문맥 파악의 한계와 정보 누락, 환각 현상(Hallucination) 등의 문제를 갖고 있어 실무 적용에 어려움이 있었다. 이에 본 연구에서는 문서 자동 파싱 , 다중 검색 기법, 랭그래프(LangGraph) 기반 응답 평가 및 재시도 구조를 포함한 실제 학사 행정 환경에 적용 가능한 고정확도 RAG 시스템을 구현하였다. 이 시스템은 특히 표 기반 데이터 처리, 답변 품질 반복 개선 등의 기능을 통해 기존 시스템 대비 실무 활용성과 학술적 완성도를 동시에 제고하고자 한다. 이후 장에서는 본 시스템의 주요 구성 요소와 기술적 특징, 그리고 성능 평가 결과를 구체적으로 제시한다.

II. Preliminaries

2.1 Related Works

2.1.1 Retrieval-Augmented Generation

RAG는 대규모 언어 모델(LLM)의 사실성 부족과 최신 정보 반영 한계를 극복하기 위해 고안된 기술로, 외부 지식 검색(Retrieval)과 텍스트 생성(Generation)을 결합한 구조를 갖는다 [8].

RAG는 ① 정보 검색 단계, ② 검색 문서를 바탕으로 문맥을 구성하는 Augmented 컨텍스트 단계, ③ 이를 바탕으로 응답을 생성하는 Generation 단계로 구성된다.

이는 지식 데이터베이스만 갱신함으로써 최신 정보를 빠르게 반영할 수 있게 하며, 특히 기업이나 기관의 특화된 도메인 지식에 기반한 질의응답 시스템을 구축하는 데 있어 효율적이다 [9]. 이러한 특성으로 인해 RAG는 기업, 공공기관, 교육기관 등에서 LLM 기반 자동화 및 고신뢰 질의응답 시스템 구현에 핵심 기술로 부상하고 있다.

2.1.2 LangChain

랭체인(LangChain)은 대규모 언어 모델(LLM)의 활용을 보다 체계적이고 유연하게 구성할 수 있도록 지원하는 오픈소스 프레임워크로, 다양한 컴포넌트(프롬프트 템플릿, 체인, 에이전트, 벡터 검색 등)를 모듈화하여 제공한다. 랭체인(LangChain)은 오픈AI(OpenAI), 앤트로픽(Anthropic), 허깅페이스(Hugging Face) 등 다양한 모델 공급자와 통합되며, 공급자 간 API 구조를 일관되게 유지할 수 있는 장점을 갖는다 [10][11].

삼성 SDS는 2024년 11월, 자사 클라우드 환경에서 Kubernetes 장애를 자동 진단하고 해결 방안을 제시하는 시스템(SKE-GPT)을 구축하며, 랭체인(LangChain)을 기반으로 문서 검색과 응답 생성을 처리하는 RAG 체인을 적용한다 [12].

2.1.3 LangGraph

랭그래프(LangGraph)는 랭체인(LangChain) 기반의 상태 기반 오케스트레이션 프레임워크로, 복잡한 에이전트 워크플로우를 그래프 구조로 설계할 수 있도록 한다. 노드(node)와 엣지(edge)를 통해 작업 흐름을 구성하며, 반복 실행, 분기 조건, 에러 핸들링, 인간 개입(Human-in-the-loop) 등 고급 제어가 가능하다 [13][14].

2.1.4 Upstage Document Parse

업스테이지 도큐먼트 파서(Upstage Document Parse)는 다양한 문서 형식을 처리할 수 있는 고성능 문서 파싱 시스템으로, 한글에 특화된 구문 분석 능력과 OCR 기반 시각 정보 인식 기능을 결합해 문서를 LLM 응답에 최적화된 입력 포맷으로 변환한다 [15].

지원 형식은 JPEG, PNG, BMP, PDF, TIFF, HEIC, DOCX, PPTX, XLSX, HWP, HWPX이며, 특히 표 레이아웃 인식과 LaTeX 기반 수식 처리에 강점을 가지며, 출력 결과는 HTML 또는 Markdown 형식으로 제공된다.

III. The Proposed Scheme

3.1 System Architecture

본 시스템의 질의응답 테스트 대상 문서는 제주대학교에서 발행한 『2025학년도 신입생 수강편람.pdf』(총 172페이지)이며, 이미지, 병합된 표, 다층 구조의 표(Table-within-Table), 다면적 표 구조 등 다양한 문서 구조를 포함하고 있어 질의응답의 정확성을 실험하는 데 적합하였다. 본 시스템의 워크플로우는 크게 두 단계로 구분된다.

3.1.1 Document Preprocessing Pipeline

첫 번째 단계는 문서를 업스테이지 도큐먼트 파서(Upstage Document Parse)에 업로드하고 전처리하여 Vector Database에 저장하는 과정으로 Fig 1과 같다.

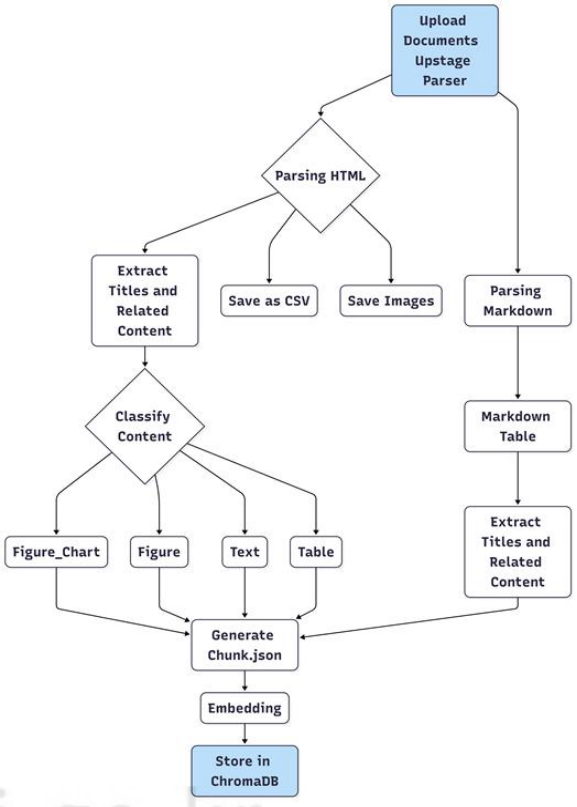

Fig. 1. Document Preprocessing Pipeline

3.1.1.1 Document Upload, and Vector DB Storage

업스테이지 도큐먼트 파서(Upstage Document Parse)는 HTML, Markdown, Text의 세 가지 형식으로 파싱할 수 있으며, 이 중 HTML 파싱은 문서의 요소(Element) – 예: 표(Table), 이미지(Figure), 차트(Figure_Chart), 제목, 주석 등 – 를 시각적 읽기 흐름에 따라 태깅하여 구조화된 결과를 제공한다. 위 파이프라인은 문서를 우선 HTML 형식으로 파싱하고 그 결과값을 Text, Table, Figure, Figure_Chart 유형별로 분류하여 저장한다. 저장 시에는 의미 있는 문장 단위로 문서를 청크(Chunk)화 하여 저장하는데, 일반적인 Text 기반 문서에서는 적절한 Chunk Size 및 Overlap을 설정함으로써 청크 간 의미 단절을 방지할 수 있다. 그러나 표나 이미지가 포함된 문서에서는 단순 청킹 방식으로 처리할 경우 표 구조가 절단되거나 이미지 내용이 누락되는 문제가 발생한다. 이를 방지하기 위해 Table 및 Figure 요소는 하나의 청크 단위로 묶어 분리되지 않도록 설계한다.

특히 대학 행정 문서는 다면적이고 장문의 표(Table)가 다수를 차지하며, 이 경우 표 전체를 청크 하나로 처리하면 LLM의 토큰 수 제한과 검색 성능 저하 문제가 발생하게 된다. 본 연구 과정 중에는 이러한 문제를 해결하기 위해 다양한 파일 형식(CSV, HTML, Markdown 등)으로 파싱하여 실험하였고, Markdown 기반의 Table 파싱 결과가 가장 높은 질의응답 정확도를 보였다.

3.1.1.2 Handling Merged Cells and Internal Table Structure

표 처리에서 중요한 과제 중 하나는 헤더(Header) 검출의 정확성이다. 예를 들어 '연도 | 학기 | 교과목코드 | 학점'과 같은 헤더 구성에서 하나의 열이 누락되면 LLM은 숫자 '3'을 학점인지 교과목 코드인지 판단하지 못하여 오류가 발생한다. 이는 병합된 셀(Merged cells)이나 표 내부에 또 다른 표가 있는 복잡한 구조로 인해 발생하며, 단순한 HTML-to-Markdown 변환이나 일반 파서 도구만으로는 정밀한 처리가 어렵다.

업스테이지 도큐먼트 파서(Upstage Document Parse)로 문서를 Markdown 형식으로 파싱할 경우 이러한 병합 구조를 일정 수준까지 보존할 수 있으며, LLM에 구조화된 정보를 보다 명확히 제공할 수 있다. 그러나 Markdown 파싱 결과는 HTML과 달리 Element 태그 정보가 포함되지 않아 Table 제목이나 구조 인식에 어려움이 따른다. 이를 해결하기 위해 특정 패턴 기반 규칙을 활용하여 Table 제목과 Header를 정확히 추출하고, 연결된 표에서 페이지 전환 시 분리되는 표 뒷부분에 제목 누락을 방지하기 위해 직전 추출된 제목을 상속하는 로직을 설계하였다. 또한, 표 뒤에 이어지는 관련 설명을 5줄 정도 추출하여 메타데이터로 저장하는 기능도 구현한다.

최종적으로 HTML 과 Markdown 2가지 형식으로 Table 이 각각 저장되도록 하였다.

3.1.1.3 Image Processing and Chunk File Generation

이미지는 업스테이지 도큐먼트 파서(Upstage Document Parse)의 Base64_Encoding 기능을 이용해 인코딩/디코딩하여 저장하고, HTML 파싱에서 추출한 Figure, Figure_Chart 청크와 매핑하여 파일명과 경로를 메타데이터에 저장하였다. 추후 LLM 질의응답 시 관련 이미지를 파이썬의 대표적인 시각화 라이브러리인 매트플롯리브(Matplotlib)로 답변과 함께 제공한다.

HTML형식으로 파싱하여 분류한 Text, Table, Figure, Figure_Chart 청크 목록에 Markdown 형식으로 파싱한 Table을 추가하면 최종적으로 생성된 청크 문서 유형은 Table 2와 같이 분류된다.

Table 2. Types of Chunked Documents

Category	Main included content
Text	Document_Number : int Content : HTML Metadata : Title
Table (HTML)	Document_Number : int Content : html_original Metadata : {Title, CSV_Filename, Related_Texts}
Table (Markdown)	Document_Number : int Content : Markdown Metadata : {Title, Header, Related_Texts}
Figure	Document_Number : int Content : HTML Metadata :{ Title, Related_Texts, Image_Filename, Image_Path }
Figure_Chart	Document_Number : int Content : HTML Metadata :{ Title, Related_Texts, Image_Filename, Image_Path }

이러한 청크들은 오픈AI(OpenAI)의 임베딩모델(Text-Embedding-3-Large)을 통해 임베딩되어, 크로마 DB(Chroma DB)에 저장된다.

3.1.2 Question Answering Pipeline (RAG Chain)

두 번째 워크플로우는 질의응답의 정확성과 효율성을 극대화하기 위한 RAG 체인을 구성하는 과정이다. 본 시스템은 오픈AI(OpenAI)의 LLM 모델(GPT-4o-mini)을 기반으로 답변을 생성하며, 특히 복잡한 구조의 문서에서 정확한 정보를 찾아내기 위해 고안된 다각적인 검색 및 평가 메커니즘을 포함한다.

Vector DB에 저장된 임베딩 청크를 검색하는 방식은 키워

드 기반 검색(BM25)과 유사도 검색(Similarity Search)을 결합한 하이브리드 방식으로 설계한다.

순위 결정 메카니즘으로 문서 우선순위를 정하고 LLM이 스스로 답변을 평가하고 새답변을 생성하는 노드를 추가한다.

질의 응답 파이프라인 구성은 Fig 2와 같다.

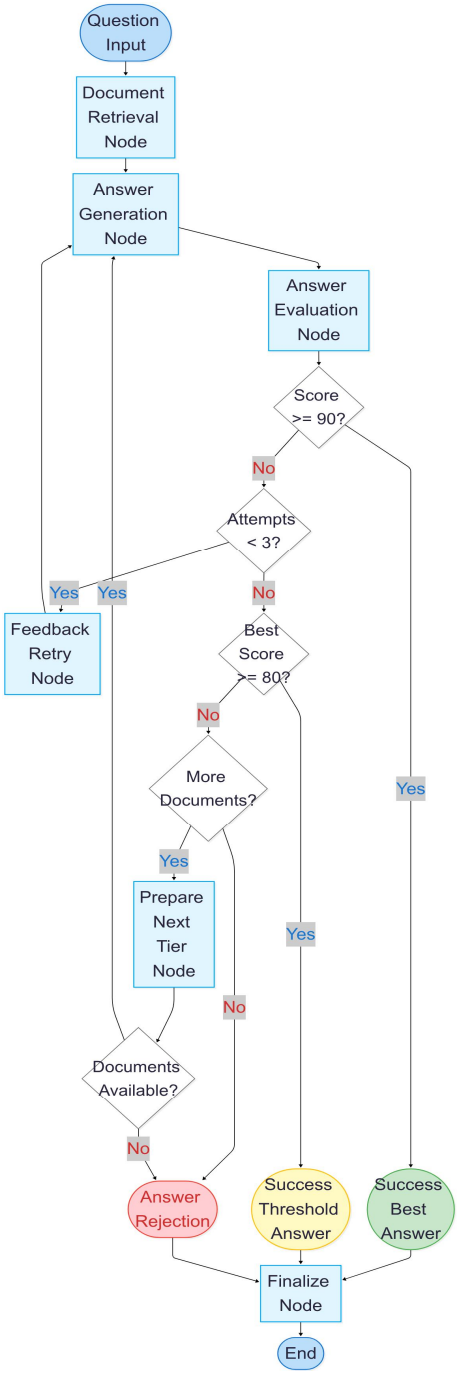

Fig 2. Question Answering Pipeline

3.1.2.1 Design of Ensemble Retriever

단일 검색 방식의 한계를 극복하기 위해, 본 시스템은 두 가지 검색 전략을 동시에 활용하는 앙상블 리트리버(Ensemble Retriever)를 구현한다. 첫 번째 전략은 키워드 기반 검색 (BM25)이다. 실험 결과, 표(Table) 형식으로 구조화된 청크에 포함된 정보는 사용자의 질의에 포함된 특정 키워드와 직접적으로 일치하는 경우가 많았다. 예를 들어 '2025년 1학기 학사일정'과 같은 질의는 '학사일정', '1학기' 등의 키워드로 정확한 표를 찾아낼 수 있다. 이를 위해 보편적으로 널리 사용되는 키워드 검색 알고리즘인 BM25를 도입하였다. 이는 실험적으로 표 구조의 청킹 문서에서 정답이 포함된 문서를 효과적으로 검색하는 것을 확인하였기 때문이다.

두 번째는 다중 쿼리 생성기(Multi-Query Generator)를 활용한 유사도 검색이다. 유사도 검색의 경우, 사용자의 단일 질문만으로는 질문의 본질적인 의도를 포괄적으로 파악하기 어려울 수 있다. 이러한 의미적 검색의 범위를 확장하기 위해, LLM을 활용하여 원본 질문과 유사한 질문 3개를 추가로 생성하는 다중 쿼리 생성기(Multi-Query Generator)를 도입하였다. 예를 들어, 사용자가 "등록금 납부 기간이 언제인가요?"라고 질문하면, LLM은 "등록금은 언제까지 내야 하나요?", "2025학년도 등록금 납부 일정을 알려주세요." 등과 같이 다양한 형태의 질문을 생성한다. 이 생성된 질문들을 바탕으로 여러 관점에서 동시 다발적인 유사도 검색을 수행함으로써, 관련 문서를 검색할 확률을 높였다.

3.1.2.2 Ranking Mechanism and Final Document Selection

앙상블 리트리버는 BM25 검색 결과와 다중 쿼리 기반 유사도 검색 결과를 결합하여 최종 순위를 결정한다. 두 검색 방식은 점수 체계가 다르기 때문에, 가중치 기반 정규화와 함께 LRF(Linear Rank Fusion) 방식으로 통합하였다. 각각의 검색 방식에 별도의 가중치를 부여하여 특정 검색 결과의 중요도를 조절할 수 있도록 설계하였으며, 두 결과를 LRF 알고리즘으로 융합한 후 가장 높은 점수를 받은 상위 6개의 문서를 최종적으로 선택하여 LLM에 전달한다.

Table 3. Ranking Mechanism

Category	Formula and Explanation
Keyword Search (BM25)	BTF-IDF: Term Frequency × Inverse Document Frequency → Top 6 docs ranked → Score = (6-rank)/6 × 0.3
Multi-Query Similarity	4 queries (1 original + 3 LLM-generated) × 6 docs each = 24 docs → Deduplication → Final Score = Avg Similarity + (Frequency-1)×0.05 + Original Query Bonus(0.12) → Top 6 selected → Score = (6-rank)/6 × 0.7
Ensemble Combination	BM25 normalized scores + Similarity normalized scores for each document
Final Selection	Compare all documents by combined scores → Select top 6 documents
Final Ensemble Score	BM25 Score + Similarity Score (MAX: 1.0)

Table 3은 순위 결정 메카니즘에 의해 문서의 우선순위를 판단하는 과정이다.

i. 키워드 검색(BM25)은 문서 내 키워드의 빈도와 희귀도를 반영하며, 정규화된 점수로 상위 6개 문서를 평가한다. 최대 점수는 0.3(가중치0.3)이다.

ii. 다중 쿼리 기반 유사도 검색은 원본 쿼리와 LLM이 생성한 3개의 보조 쿼리를 활용해 총 24개의 문서를 검색하고 중복을 제거한 뒤, 빈도수 가중치(0.05)와 원본질문 가중치(0.12)를 포함해 최종 점수를 계산한다. 정규화된 점수는 최대 0.7(가중치0.7)이다.

iii. 최종적으로 두 검색 결과를 합산하여 최고 1.0점까지 부여할 수 있으며, 이 방식은 정확한 키워드 일치성과 의미 기반 유사도를 균형 있게 반영한다.

iv. 최종 선택 단계(Final Selection)에서는 두 방식에서 나온 정규화 점수를 합산하여 문서별 최종 점수를 계산하고, 이를 기준으로 상위 6개 문서를 선택한다.

3.1.2.3 LangGraph-Based Answer Generation and Self-Evaluation

본 연구에서는 답변 생성과 검증 과정에 랭그래프(LangGraph) 기반 상태 워크플로우를 설계하였다. 이 워크플로우는 문서 검색(Document Retrieval), 답변 생성(Answer Generation), 답변 평가(Answer Evaluation), 피드백 재시도(Feedback Retry), 차순위 문서 준비(Prepare Next Tier), 최종화(Finalize)의 6개 핵심 노드로 구성된다. 답변 평가 노드에서는 질문 직접 답변성(25점), 문서 정보 정확 활용(25점), 명확성 및 이해성(20점), 사실적 오류 부재(15점), 적절한 문서 검색(15점)의 5개 기준으로 총 100점 만점 평가를 수행한다. 90점 이상 달성 시 즉시 최적 답변으로 채택하며, 90점 미만일 경우 최대 3회까지 동일 문서 집합으로 피드백 기반 재생성을 실시한다. 90점 기준은 본 연구에서 수행한 50문항 평가 결과, 정답일 경우 자체 평가 점수가 대부분 90점 이상으로 나타났으며, 실제로 정답으로 판단된 47개 문항의 평균 점수는 98점에 달해 신뢰도 있는 기준으로 판단하였다. 3회 시도 완료 후에도 90점에 미달할 경우, 최고 점수가 80점 이상이면 임계값 답변으로 채택하고, 80점 미만이면 후순위 문서를 활용한 차순위 검색을 수행한다. 차순위 검색에서는 잔여 문서 풀을 재순위화하여 새로운 6개 문서를 선택하고 시도 횟수를 초기화한다. 모든 문서가 소진되거나 최종적으로 80점에 미달하는 경우 환각 현상 방지를 위해 답변을 거부한다. 이러한 계층적 문서 활용과 점진적 품질 향상 메커니즘을 통해 답변 품질의 일관성을 확보하면서도 시스템 자원의 효율적 활용과 무한 루프 방지를 달성하였다.

IV. Implementation

4.1 Experimental environment

본 시스템은 Anaconda 가상환경 내 Jupyter Notebook 환경에서 랭체인(LangChain) 라이브러리를 활용하여 구축되었다. 업스테이지 도큐먼트 파서(Upstage Document Parse)를 활용하여 문서를 파싱하고 오픈AI(OpenAI)의 임베딩모델(Text-Embedding-3-Large)로 임베딩하여 크로마 DB(Chroma DB)에 저장하였고, 랭체인(LangChain)라이브러리인 랭그래프(LangGraph)를 활용하여 질의응답 노드를 구성하였다. Table 4와 5는 실험 환경을 나타낸다.

Table 4. Hardware Environment

Item	Value
CPU	AMD Ryzen™ 5 5625U
RAM	8GB
OS	Windows 11

Table 5. Software Environment

Item	Value	Version
Virtual Environment	Anaconda	24.11.3v
Language	Python	3.12.7v
Vector DB	Chroma DB	0.6.3v
IDE	Jupyter Notebook	7.2.2v

V. Result

5.1 System Evaluation

본 연구는 170페이지 분량의 「2025학년도 신입생 수강편람」이라는 복잡한 표 구조를 포함한 문서를 대상으로 질의응답 성능을 평가하였다. 초기에는 오픈소스 파서인 Win32 Hwp Loader를 사용하였으나, 병합 셀, 표 내 표 등 고난이도 구조 처리에 한계가 있었고, OCR 기능이 없어 시각 정보 활용에도 제약이 있었다. 이에 따라 업스테이지 도큐먼트 파서(Upstage Document Parse) 기반 파이프라인을 도입하였고, 병합 셀 분석, 표 제목 추출, 이미지 OCR 등 정교한 전처리가 가능해졌다. 표 기반(29문항), 텍스트 기반(16문항), 이미지 기반(2문항), 환각 현상 검증용(3문항) 등 총 50개의 질의를 바탕으로, 상이한 4가지 시스템 제원에 따른 성능 평가를 수행하였다. 각 시스템의 구성은 Table 6에 정리하였다.

Table 6. Test System Specifications

No.	Upload Parser	Chunking Method	Retrieval Strategy
1	Win32 Hwp Loader	- Type: Text, Table - File Format: Plain Text, HTML-to-Markdown	Single Similarity Search (k=3)
2			Multi-query similarity search + keyword search
3	Upstage Document Parse	- Type: Text, Table, Figure, Figure_Chart - File Format: Plain Text, HTML, Markdown	Single Similarity Search (k=3)
4			Multi-query similarity search + keyword search

특히 랭그래프(LangGraph) 기반 구성에서는 "표 내 표"에 대한 질문과 OCR 이미지에 대한 질문에도 정확성이 우수 하였고 그 결과 94% 의 우수한 답변율을 보였다.

자체 평가 기준점수를 90점으로 설정하고 환각(Hallucination) 현상에 의한 답변이 생성되지 않도록 하였으며, 3가지 함정 문항에서도 답변을 거부하는 결과를 도출하였다.

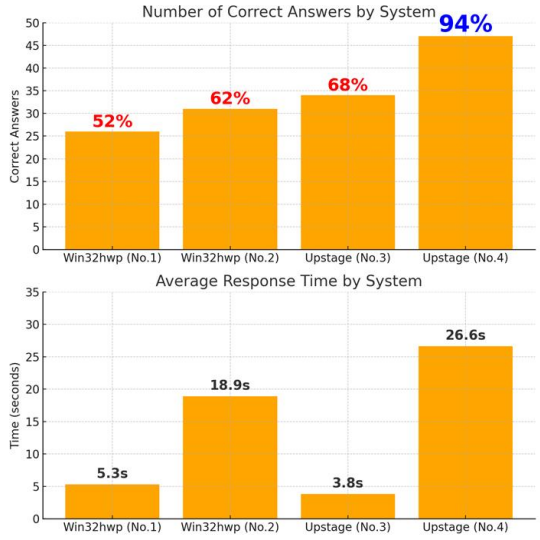

Fig. 3. Accuracy and Response Time of Each System

Fig 3은 시스템의 초기 모델과 최종모델의 평가결과를 비교한 그래프로 정답률 개선 결과와 평균 응답 속도를 보여준다.

Fig 4는 실제 시스템에서 검색된 문서의 우선순위가 부여되고 답변을 생성한 결과를 보여준다.

5.2 Contribution of the Proposed System

본 시스템은 기존 RAG 기반 질의응답 시스템이 취약한 표 기반 문서 처리에 특화되어 있다. 특히 대학 행정 문서에서 자주 사용되는 복잡한 표 구조, 병합 셀, 이미지 기반 정보를 효과적으로 처리하기 위한 다음과 같은 전략을 도입하였다.

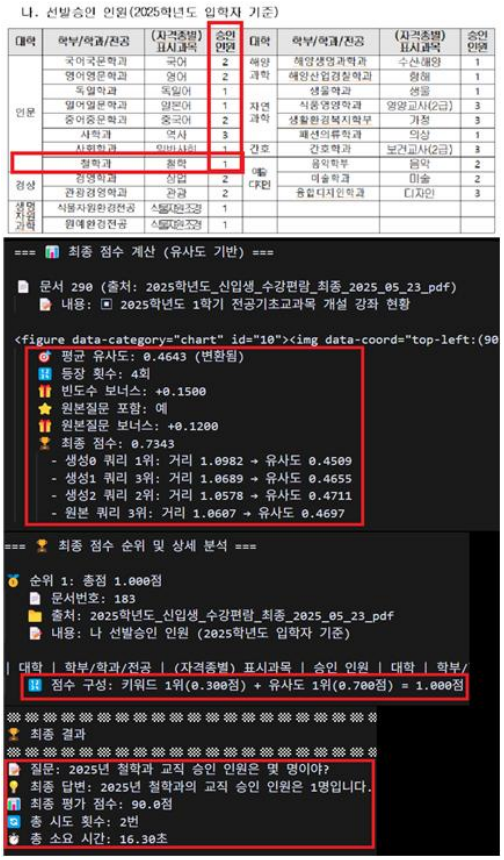

Fig. 4. Answer Results and Scoring Log

i. 표 구조 중심 청크화: 표를 원형 그대로 하나의 청크로 구성하고, 표 제목·헤더·설명문을 함께 포함하여 LLM이 표의 의미를 온전히 이해하도록 설계하였다. 페이지 분리로 인한 제목 누락은 제목 상속 방식으로 해결하였다.

ii. 병합 셀 표구조 파악: 병합 셀 표구조는 OCR 기반 Markdown 형식의 파싱결과로 병합되기 전 각 셀에 동일 정보를 삽입함으로써 열 누락 없이 정확한 표 구조를 전달하였다.

iii. 이미지 정보 연동: 이미지 OCR 결과를 텍스트로 추출하고, 메타데이터와 함께 저장하여 검색된 답변에 시각 자료를 연계할 수 있도록 하였다.

이러한 개선은 단순 파이프라인 구성 이상의 효과를 가져왔으며, 특히 표 기반 문서에서 응답 정확도를 결정짓는 핵심 요인이 문서 파싱 후 전처리 과정임을 실험을 통해 입증하였다.

VI. Conclusions

본 연구에서는 대학의 학사규정, 수강편람, 입시요강 등 다양한 비정형 학사 행정 문서를 효과적으로 처리하고 활용하기 위해 RAG 기반 질의응답 시스템을 설계하였다. 제안된 시스템은 학사 행정 관련 질의에 대해 빠르고 신뢰성 있는 응답을 제공함으로써, 대학 구성원들의 정보 접근성을 크게 향상시키는 효과를 보였다. 특히 표가 포함된 복합 문서에서도 일정 수준 이상의 구조 이해 능력을 보였다는 점에서 의미 있는 성과로 평가된다. 그러나 일부 복잡한 문서 구조나 긴 문맥이 포함된 질의에 대해서는 여전히 환각(Hallucination) 현상이 관찰되었으며, 이로 인해 부정확한 정보가 제공될 가능성도 확인되었다.

향후 연구에서는 생성된 응답의 사실성뿐 아니라 그 근거 문서의 정확성과 최신성을 동시에 검증할 수 있는 체계적인 검증 메커니즘의 도입이 필요함을 확인하였다.

REFERENCES

[1] Bloomberg Intelligence, “Generative AI to Become a $1.3 Trillion Market by 2032,” Bloomberg, https://www.bloomberg.com/company/press/generative-ai-to-become-a-1-3-trillion-market-by-2032-research-finds/.

[2] Korea University Office of Digital Information, “The Beginning of a New General Education Recommendation Service: A.I. Sunbae,” https://ic.korea.ac.kr/ic/newsletter/newsletter14.do.

[3] Pusan National University, “Pusan National University Chatbot 'Sanji-ni' Service Open Announcement,” https://his.pusan.ac.kr/nursing/14523/subview.do?enc=Zm5jdDF8QEB8JTJGYmJzJTJGbnVyc2luZyUyRjI1ODQlMkYxMTcwOTk4JTJGYXJ0Y2xWaWV3LmRvJTNG.

[4] News TNT, “Chungnam National University Introduces 'AI ChaCha' AI Chatbot, Smallest Among Regional National Universities,” https://www.newstnt.com/news/articleView.html?idxno=479002.

[5] Munhwa Ilbo, “The University of Ulsan has become the first university in South Korea to implement 'U-MATE', an AI-powered academic advising system,” https://www.munhwa.com/article/11485910.

[6] J. S. Lee, M. H. Park, S. Y. Kim, and H. J. Choi, “Development and Application of an AI-Powered Adaptive Course Recommender System in Higher Education: An Example from K University,” Journal of Educational Technology, vol. 37, no. 2, pp. 267-307, 2021. DOI: 10.17232/KSET.37.2.267.

[7] ChosunBiz, “The South Korean Government's Persistent Reliance on HWP in the Age of AI: Hindering Public Data Utilization and Falling Behind Global Trends,” https://biz.chosun.com/it-science/ict/2024/02/14/R76VBXNVVZEI5EFJECIEJUYL4E/.

[8] Y. Yoon and S. Kim, “Trends and Prospects of Retrieval-Augmented Generation (RAG) for Generative AI,” The Journal of Korean Association of Computer Education, vol. 28, no. 2, pp. 69-76, Feb. 2025. DOI: 10.32431/kace.2025.28.2.007.

[9] G.-W. Yi and S. K. Kim, “Design of a Question-Answering System Based on RAG Model for Domestic Companies,” Journal of The Korea Society of Computer and Information, vol. 29, no. 7, pp. 81-88, Jul. 2024. DOI: 10.9708/jksci.2024.29.07.081.

[10] LangChain, “LangChain Documentation: Introduction,” https://python.langchain.com/docs/introduction/.

[11] LangChain, “LangChain Homepage,” https://langchain.com/.

[12] S. Gang, “Case Study: Samsung SDS Uses Retrieval Augmented Generation for Kubernetes Troubleshooting,” Samsung SDS Blog, Nov. 18, 2024. https://tech.samsungsds.com/insights/SKE-GPT-Kubernetes-RAG.

[13] LangChain, “LangGraph,” https://www.langchain.com/langgraph.

[14] LangChain, “Learn LangGraph basics,” https://langchain-ai.github.io/langgraph/.

[15] Upstage, “Document Parse: Convert complex documents into LLM-readable formats,” https://www.upstage.ai/products/document-parse.

Authors

Jae-Kwang Oh is currently a student in the Department of Computer Engineering, Jeju National University. He has also been working in the Office of Digital Information at Jeju National University since 2022. His research interests include artificial intelligence, natural language processing, and AI agents.

Soo Kyun Kim received Ph.D. in Computer Science & Engineering Department of Korea University, Seoul, Korea. He is now a professor at the Department of Computer Engineering at Jeju National University, Dr. Kim has published many research papers in international journals and conferences. His research interests include multimedia, pattern recognition, image processing, mobile graphics, geometric modeling, and interactive computer graphics. He is a member of ACM, IEEE, IEEE CS, KACE, KMMS, KKITS, and KIIT.