사물인터넷융복합논문지 Vol. 12, No. 4, pp. 103-109, 2026 https://doi.org/10.20465/KIOTS.2026.12.4.103

LLM 기반 RAG 시스템을 활용한 중소제조기업 사내 문서 업무 자동화에 관한 연구

이근호*
백석대학교 컴퓨터공학부 교수

A Study on Enterprise Document Task Automation for Small and Medium-sized Manufacturing Enterprises Using an LLM-based RAG System

Keun-ho Lee*
Professor, Division of Computer Engineering, BaekSeok University

요 약 최근 생성형 인공지능의 발전으로 다수의 기업이 사내 업무에 거대언어모델을 도입하고 있으나, 클라우드 기반 LLM은 정보 유출 위험과 환각현상으로 인해 보안이 중요한 제조 현장에 곧바로 적용하기 어렵다는 한계가 있다. 본 연구는 사내지식의 분산, 반복적인 문서 작성 업무, 개인화된 업무 노하우로 인한 신입 직원의 온보딩 지연 문제를 해결하기 위하여, D사와의 산학연계 프로젝트를 통해 온프레미스(On-premise) 환경에서 동작하는 LLM 검색 증강 생성 기반 사내 문서 업무 자동화 시스템을 설계하고 구현하였다. 제안 시스템은 질의 확장, Parent-Child 청킹, 2단계 재순위화로 구성된 RAG 파이프라인을 통해 검색 정확도를 개선하였으며, document_type 컬럼을 활용한 확장 가능한 데이터베이스 스키마를 설계하여 견적서·거래명세서·주간보고서 등 서로 다른 문서 서식을 시스템 구조 변경 없이 수용할 수 있도록 하였다. 4명의 구현 담당자를 대상으로 한 검색 성능 평가에서 최고 성능 담당자는Recall@3 96.0%, MRR@3 92.7%를 달성하였으며, 최소 4GB(GTX 1650)에서 다중 사용자 환경 기준12GB(RTX 5070) 이상의GPU로 로컬LLM·임베딩·재순위화 모델을 함께 운용할 수 있음을 실측을 통해 확인하였다. 본 연구는 온프레미스RAG 시스템이 정보 유출 위험 없이 중소제조기업의 사내 지식 검색과 반복 문서 업무를 자동화할 수 있음을 실증한 사례연구로서 의의를 가진다.

주제어 : 검색 증강 생성, 거대언어모델, 사내 문서 자동화, 하이브리드 검색, 온프레미스

Abstract With the rapid advancement of Generative AI, many enterprises are adopting Large Language Models(LLMs) for internal operations; however, cloud-based LLMs are difficult to deploy directly in security-sensitive manufacturing environments due to the risk of data leakage and the hallucination problem. To address the fragmentation of in-house knowledge, repetitive document-drafting tasks, and delayed onboarding of new employees caused by personalized tacit knowledge, this study designed and implemented an on-premise LLM Retrieval-Augmented Generation(RAG) system for enterprise document task automation through an industry-academia project conducted with Dujeong Tech. The proposed system improves retrieval accuracy through a RAG pipeline composed of query expansion, parent-child chunking, and two-stage reranking, and adopts an extensible database schema based on a document_type column so that heterogeneous document formats-such as quotations, transaction statements, and weekly reports-can be accommodated without structural changes to the system. In a retrieval performance evaluation conducted by four implementers, the best-performing implementer achieved a Recall@3 of 96.0% and an MRR@3 of 92.7%, and measurements confirmed that the local LLM, embedding, and reranking models can be operated together on GPUs ranging from a minimum of 4GB (GTX 1650) to 12GB or higher (RTX 5070) for multi-user environments. This study presents an empirical case demonstrating that an on-premise RAG system can automate internal knowledge retrieval and repetitive document tasks for small and medium-sized manufacturing enterprises without the risk of data exposure.

Key Words : Retrieval-Augmented Generation, Large Language Model, Enterprise Document Automation, Hybrid Search, On-premise

이 연구는 2026학년도 백석대학교 대학연구비 지원에 의하여 이루어졌음.
*교신저자 : 이근호(leekeunho1004@gmail.com)
접수일 2026년 08월 04일 수정일 2026년 08월 14일 심사완료일 2026년 08월 21일

1. 서론

생성형 인공지능(Generative AI) 기술의 급속한 발전에 힘입어 대형 언어 모델(LLM)은 문서 요약, 정보 검색, 코드 생성 등 다양한 업무 영역에서 활용되고 있다. 그러나 LLM은 사실과 다른 정보를 그럴듯하게 생성하는 환각(Hallucination) 현상에 취약하며, 이는 파라미터에 내재된 지식만으로 응답을 생성하는 구조적 한계에서 비롯된다[1]. 특히 사내 매뉴얼, 거래 문서, 업무 보고서와 같이 기업 고유의 비정형 지식을 다루어야 하는 환경에서는 환각으로 인한 오답이 실제 업무 오류로 직결될 수 있어 신뢰성 확보가 핵심 과제가 된다. 이러한 한계를 보완하기 위한 대표적인 접근이 검색 증강 생성(Retrieval-Augmented Generation, RAG)이다. RAG는 질의와 관련된 외부 문서를 실시간으로 검색하여 LLM의 입력 컨텍스트에 포함시킴으로써, 모델의 파라미터를 재학습하지 않고도 최신·도메인 특화 지식을 반영한 응답을 생성할 수 있도록 한다[2]. 국내에서도 LangChain과 같은 프레임워크를 활용해 사내 데이터 기반 생성형AI 서비스를 구현하는 사례가 보고되고 있으며[3], RAG의 구조적 요소와 발전 동향을 정리한 연구도 이어지고 있다[4]. 그러나 다수의 선행 사례는 오픈 도메인 질의응답이나 공개 클라우드LLM 연동에 초점을 맞추고 있어, 정보 유출을 허용할 수 없는 중소제조기업의 폐쇄망 환경에 곧바로 적용하기 어렵다는 실무적 공백이 남아 있다.

국내 중소제조기업은 디지털 전환에 대한 기술적 중요도는 높게 인식하면서도 실제 도입 수준은 낮아 이상과 현실의 괴리가 크다는 점이 실증적으로 보고된 바 있다[5]. 이는 사내 지식이 소수 숙련자에게 편중되어 축적되고, 반복적인 보고서·전표 작성 업무에 많은 시간이 소요되며, 보안 정책상 GPT·Claude와 같은 외부 클라우드 AI를 사용할 수 없는 현실적 제약과 맞물려 있다. 본 연구는 동일한 AX 방법론을 생산 현장이 아닌 사무·행정 업무 영역으로 확장 적용하여, AX 접근이 제조업 전반의 서로 다른 업무 영역에 일관되게 적용될 수 있음을 보이고자 한다.

이에 본 연구는 D사와의 산학연계 프로젝트를 통해, 온프레미스 환경에서 동작하는LLM RAG 기반 사내 문서 업무 자동화 시스템을 설계·구현한 실증 사례를 제시한다. 구체적인 연구 목적은 다음과 같다. 첫째, 사내에 축적된 업무 매뉴얼과 문서를 검색 가능한 지식 자산으로 전환하고, 주간보고서 작성과 자재입출고 처리 등 반복적인 문서 업무를 자동화하는 RAG 시스템을 설계한다. 둘째, 질의 확장, Parent-Child 청킹, 2단계 재순위화로 구성된 파이프라인을 통해 검색 정확도를 개선하고 그 효과를 정량적으로 검증한다. 셋째, 서로 다른 문서 서식을 시스템 구조 변경 없이 수용할 수 있는 확장 가능한 데이터베이스 스키마를 설계한다. 넷째, 온프레미스 환경에서 시스템을 안정적으로 운용하기 위한 최소·권장 하드웨어 사양을 실측을 통해 제시한다.

본 논문은 총 5장으로 구성된다. 2장에서는 RAG 및 정보검색 기술, LLM 기반 답변 생성과 검색 정확도 개선 기법, 국내 중소제조기업의 AX·DX 현황에 관한 선행연구를 고찰한다. 3장에서는 제안 시스템의 요구사항, 아키텍처, RAG 파이프라인, 데이터베이스 스키마를 기술한다. 4장에서는 검색 성능 평가 결과와 시스템 리소스 요구사항을 제시한다. 5장에서는 연구 결과를 종합하고 한계 및 향후 연구 방향을 논의한다.

2. 관련연구

2.1 RAG 및 정보검색 기술 동향

RAG는 사전학습된 파라메트릭 생성 모델과 비파라메트릭 검색 모델을 결합하는 미세조정 접근으로 제안되었으며[2], 이후 정보검색 단계의 성능이 전체 RAG 파이프라인의 응답 품질을 좌우하는 핵심 요소로 지목되어 왔다. 검색 방식은 크게 키워드 매칭에 기반한 희소 검색(Sparse Retrieval)과 임베딩 유사도에 기반한 밀집 검색(Dense Retrieval)으로 구분된다. Karpukhin 등은 질의와 문서를 동일한 임베딩 공간에 사상하는 이중 인코더 구조의 밀집 검색기(Dense Passage Retriever, DPR)를 제안하여, 기존 BM25 기반 희소 검색 대비 높은 검색 정확도를 보였다[7]. BM25는 용어 빈도와 역문서 빈도를 확률적으로 결합한 순위화 함수로, 여전히 많은 산업 현장의 검색 시스템에서 밀집 검색을 보완하는 기준선으로 활용되고 있다[8]. 대규모 임베딩 벡터에 대한 근사 최근접 이웃 탐색을 위해서는 계층적 탐색 가능 소규모 세계 그래프(Hierarchical Navigable Small World, HNSW) 알고리즘이 널리 사용되며[9], 본 연구에서 채택한 벡터 데이터베이스Qdrant 또한 HNSW 기반 인덱싱을 지원한다. 임베딩 모델로는 다국어·다중 세분화 검색을 지원하는 BGE 계열 모델이 대표적이며, C-Pack 프레임워크는 이러한 범용 텍스트 임베딩 모델의 학습 데이터·평가·모델을 통합적으로 제공한다[10].

2.2 LLM 기반 답변 생성과 온프레미스 배포

RAG의 생성 단계를 담당하는 LLM은 대부분Transformer 아키텍처를 기반으로 하며, 셀프 어텐션(Self-Attention) 메커니즘을 통해 입력 시퀀스 내 장거리 의존성을 효율적으로 학습한다[11]. 클라우드 API 기반 LLM은 높은 생성 품질을 제공하지만, 사내 문서를 외부로 전송해야 한다는 점에서 보안이 중요한 산업 현장에는 적합하지 않다. 이러한 문제를 해결하기 위한 대안으로 오픈소스 LLM을 사내 서버에 직접 배포하는 온프레미스 방식이 주목받고 있으며, LLaMA와 같은 공개 가중치 모델은 상대적으로 적은 연산 자원으로도 경쟁력 있는 생성 성능을 보여 로컬 추론 프레임워크(예: Ollama)를 통한 사내 배포의 기반이 되고 있다[12]. 다만 온프레미스 환경은 클라우드 대비 가용 연산 자원이 제한적이므로, 검색 품질을 높여 생성 단계에 전달되는 컨텍스트의 정확도를 최대화하는 것이 환각[1] 완화와 응답 신뢰성 확보에 더욱 중요해진다.

2.3 검색 정확도 개선 기법

검색 증강 생성의 정확도를 높이기 위한 대표적인 기법으로 질의 확장, 청킹 전략, 재순위화가 있다. Gao 등이 제안한 HyDE(Hypothetical Document Embeddings)는 질의에 대해 가상의 답변 문서를 먼저 생성하고 이를 임베딩하여 검색에 활용함으로써, 질의와 문서 간 표현 격차를 줄이는 질의 확장 기법이다[13]. 본 연구에서는 이와 유사한 원리로 하나의 질문을 여러 표현으로 재구성하여 검색 누락을 방지하는 질의 확장을 적용하였다. 청킹 전략과 관련하여 Liu 등은 LLM이 긴 컨텍스트의 중간에 위치한 정보를 상대적으로 잘 활용하지 못하는 현상을 실증하였으며[14], 이는 검색된 문서를 지나치게 길게 결합하기보다 정밀하게 선별된 짧은 단위로 제공하되 필요한 맥락은 함께 제공하는 Parent-Child 청킹 전략의 필요성을 뒷받침한다. 마지막으로 재순위화는 1차 검색으로 얻어진 다수의 후보 문서를 질의와의 관련성 기준으로 재정렬하는 단계로, Nogueira와 Cho는 BERT 기반 재순위화 모델이 MS MARCO 등 대규모 벤치마크에서 기존 순위화 기법 대비 큰 폭의 성능 향상을 보임을 확인하였다[15]. 본 연구는 이러한 질의 확장[13], Parent-Child 청킹[14], 재순위화[15] 기법과 희소·밀집 검색을 결합한 하이브리드 검색[7,8]을 통합하여 RAG 파이프라인을 구성하였다.

2.4 국내 중소제조기업의 AX·DX 및 사내 지식관리 현황

국내 중소제조기업을 대상으로 한 실태 조사에 따르면, 재직자들은 디지털 전환 기술의 중요성을 높게 인식하고 있음에도 실제 도입 수준에 대한 인식은 낮았으며, 도입이 필요하다고 응답한 기술 역시 고도화된AI 기술보다는 전통적 제조 시스템 관련 기술에 편중되어 있었다[5]. 이는 생성형AI 활용에 관한 연구가 활발해지는 것과 별개로, 현장 적용을 위한 구체적인 구현 사례와 보안 요건을 고려한 설계 지침이 여전히 부족함을 시사한다. 국내 LangChain·RAG 활용 사례[3,4]는 주로 공개API 기반 챗봇 구현에 집중되어 있어, 문서 서식이 다양하고 사내지식이 개인에게 편중된 중소제조기업의 실제 업무 맥락을 반영한 온프레미스 구현 사례는 상대적으로 드물다. 이에 본 연구는 D사의 실제 업무 데이터(자재관리 양식, 거래처 주문서, 주간보고서, 업무 매뉴얼)를 기반으로 RAG 시스템을 설계·구현함으로써 이러한 공백을 보완하고자 하였다.

3. 시스템 설계

3.1 요구사항 분석 및 페르소나

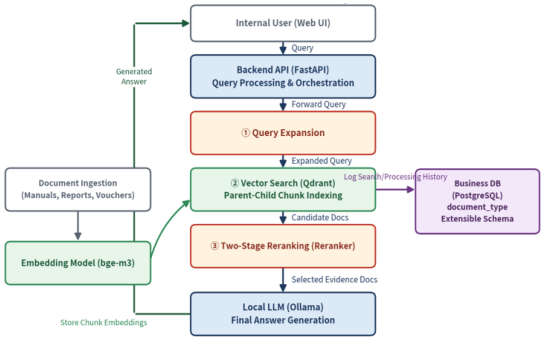
시스템 요구사항을 구체화하기 위하여 D사의 실제 업무 환경을 바탕으로 페르소나를 설정하였다. 총무부에 배치된 26세 신입사원을 대상으로 한 페르소나 분석 결과, 주요 불편 사항(Pain Point)은 다음과 같이 도출되었다. 첫째, 업무에 필요한 문서가 어디에 있는지 파악하기 어렵고 탐색에 많은 시간이 소요된다. 둘째, 보고서·집계 파일 등 정형화된 문서 작업에 과도한 시간이 소요된다. 셋째, 업무 방법을 알지 못할 때 선배 직원에게 개별적으로 문의하거나 문서를 직접 탐색해야 하므로 학습 곡선이 길다. 이러한 불편 사항으로부터 다음과 같은 요구(Needs)가 도출되었다: 보고서 작성 시간과 실수를 줄이고, 업무 매뉴얼을 신속하게 습득할 수 있어야 한다는 것이다.

현행(As-Is) 업무 방식과 목표(To-Be) 상태를 비교하면 다음과 같다. 현재는 매뉴얼 탐색과 데이터 취합에 많은 시간이 소요되며, 신입 직원의 업무 적응에도 긴 시간이 필요하다. 또한 반복적인 작성 업무로 인한 비효율과 휴먼에러가 발생하고 있으며, 보안 정책상 클라우드AI를 사용할 수 없다는 제약이 있다. 이에 대응하는 목표 상태

는 사내 매뉴얼 데이터를 체계적으로 수집·자산화하고, 업무 매뉴얼 접근성을 높여 신입 온보딩 기간을 단축하며, 표준화된 보고서 작성을 통해 업무 품질을 높이고, 데이터 노출 위험이 없는 사내 전용 온프레미스AI를 도입하는 것이다.

3.2 시스템 아키텍처

제안 시스템의 전체 아키텍처는 〈그림1〉과 같다. 사용자는 웹 기반 프론트엔드를 통해 질의를 입력하며, 이는 질의 처리와 전체 흐름을 오케스트레이션하는FastAPI 기반 백엔드로 전달된다. 백엔드는 질의를 확장한 뒤 벡터 검색 데이터베이스인 Qdrant에 질의하여Parent-Child 구조로 색인된 문서 청크 후보를 검색하고, 검색된 후보는 2단계 재순위화를 거쳐 관련성이 높은 문서만 선별된다. 선별된 근거 문서는 사내 서버에 배포된 로컬 LLM(Ollama 기반)에 전달되어 최종 답변이 생성되고, 생성된 답변은 다시 프론트엔드로 반환된다. 한편 매뉴얼·보고서·전표 등 원본 문서는 사전에 수집·적재되어 임베딩 모델(bge-m3)을 통해 청크 단위로 벡터화된 뒤 Qdrant에 저장되며, 검색·처리 이력은 PostgreSQL 기반 업무 데이터베이스에 기록되어 추후 조회와 감사에 활용된다.

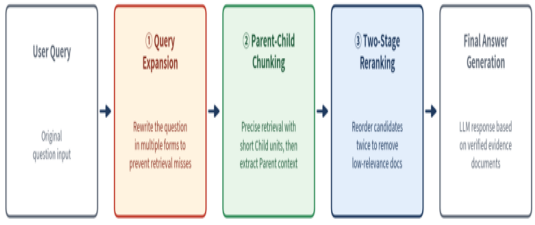


[Fig. 1] Architecture of the LLM RAG-based Enterprise Document Task Automation System

본 시스템은 주요 기능으로 (1) 사내에 축적된 업무 매뉴얼 검색, (2) 주간보고서 자동 작성, (3) 자재입출고 내역의 자동 반영을 지원한다. 활용 데이터는 자재관리 엑셀 양식과 거래처 주문서, 1월부터 6월까지의 가상 주간보고서 문서, 그리고 각 기능에 대한 사용 매뉴얼로 구성된다. 이 구조에서 AI는 검색된 사내 업무 매뉴얼을 답변 형식으로 가공하여 제공하고, 문서 내 항목을 사전 정의된 보고서 양식에 맞추어 자동으로 매칭하는 역할을 수행한다. 본 시스템의 적용 범위는 반복적인 문서 업무와 사내 지식 관리 문제를 동시에 겪고 있는 제조업 중소기업이다.

3.3 RAG 파이프라인 설계

검색 정확도를 개선하기 위하여 본 연구는 〈그림2〉와 같이 질의 확장, Parent-Child 청킹, 2단계 재순위화로 구성된3단계RAG 파이프라인을 설계하였다.



[Fig. 2] RAG Pipeline for Improving Answer Accuracy

① 질의 확장(Query Expansion) 단계에서는 사용자의 원 질문을 여러 표현으로 재작성하여, 원 질문과 표현이 다르지만 의미상 동일한 문서가 검색에서 누락되지 않도록 한다[13].

② Parent-Child 청킹 단계에서는 문서를 짧은 Child 단위로 분할하여 정밀한 유사도 검색을 수행한 뒤, 검색된 Child 청크에 대응하는 Parent 단위의 주변 맥락을 함께 추출함으로써 지나치게 단편적인 정보로 인한 오답을 방지한다. 이는 LLM이 과도하게 긴 컨텍스트의 중간 정보를 놓치기 쉽다는 선행 연구 결과[14]와도 부합하는 설계로, 검색 단위는 정밀하게 유지하되 응답 생성에 필요한 맥락은 놓치지 않도록 균형을 맞춘 것이다.

③ 2단계 재순위화(Reranking) 단계에서는 검색된 문서 후보를 두 차례에 걸쳐 재정렬하여 관련성이 낮은 문서를 제거하고 최종적으로 가장 관련성이 높은 문서만 답변 생성에 사용한다[15]. 아울러 각 단계의 후보 생성 과정에는 희소 검색과 밀집 검색을 함께 사용하는 하이브리드 검색을 적용하여 [7,8], 정확한 키워드가 포함된 문서와 의미적으로 유사한 문서를 모두 놓치지 않도록 하였다.

3.4 확장 가능한 데이터베이스 스키마 설계

D사와 같은 제조 중소기업은 견적서, 거래명세서, 주간업무보고, 회의록 등 서로 다른 형식의 문서를 취급한

〈Table 1〉 Extensible Document Schema Based on the document_type Column

Document_type	Document Type(English)	Representative Fields
weekly_report	Weekly Work Report	Department, Period, Work Performance, Work Plan
transaction_statement	Transaction Statement	Company Name, Item, Quantity, Supply Amount + Tax
quotation	Quotation	Client, Item, Unit Price, Quotation Amount, Validity Period
meeting_minutes	Meeting Minutes/Others	Meeting Date, Attendees, Agenda, Action Items

다. 신규 문서 서식이 추가될 때마다 테이블 구조를 변경해야 한다면 시스템의 유지보수 비용이 크게 증가한다. 이에 본 연구는〈표1〉과 같이documents 테이블에 document_type 컬럼을 두어 문서 서식을 구분하고, 서식별로 달라지는 세부 항목은 동일한 컬럼 구조 안에서 키-값 형태로 저장하는 방식을 채택하였다. 예를 들어 거래명세서(transaction_statement)는 상호, 품명, 수량, 공급가액 등의 항목을, 주간업무보고(weekly_report)는 기간, 업무 실적, 업무 계획 등의 항목을 포함하지만, 두 문서 모두documents 테이블의 동일한 스키마 안에 저장된다. 이러한 설계를 통해 새로운 문서 서식이 추가되더라도document_type 값 하나를 늘리는 것만으로 대응이 가능하며, 벡터 검색 파이프라인과 백엔드API 구조는 변경할 필요가 없다.

4. 실험 및 결과

4.1 평가 방법

제안 RAG 파이프라인의 검색 성능을 정량적으로 평가하기 위하여 구현 담당자4명이 각자50개씩 질의-정답 문서 쌍을 구성하고, 상위3개 검색 결과를 기준으로 Recall@3와MRR@3(Mean Reciprocal Rank at 3)를 측정하였다. Recall@3는 상위3개 검색 결과 안에 정답 근거 문서가 포함될 확률을, MRR@3는 정답 근거 문서가 상위3개 결과 중 얼마나 높은 순위에 위치하는지를 나타내는 지표로, 값이1에 가까울수록 정답 문서가1순위에 근접하게 검색되었음을 의미한다.

4.2 정량적 결과

담당자별 검색 성능 평가 결과는 〈표2〉에 제시하였다. 4명의 담당자 중 3명은 Recall@3 기준 82% 이상의 안정적인 검색 성능을 보였으며, 이 중 최고 성능을 보인 담당자는 Top-3 성공 48/50건, Recall@3 96.0%, MRR@3 92.7%를 기록하였다. 담당자 간 성능 편차는 질의-정답 쌍 구성 방식과 질의 확장에 사용한 프롬프트 설계, 그리고 청킹 단위 설정 차이에서 주로 기인한 것으로 분석된다. 상대적으로 낮은 성능을 보인 담당자의 경우 질의 표현이 정답 문서의 어휘와 크게 달라 확장 질의로도 의미적 격차가 충분히 해소되지 못한 사례가 다수 확인되었으며, 이는 질의 확장 프롬프트와 청킹 경계 설정을 표준화할 필요성을 시사한다.

〈Table 2〉 Retrieval Performance by Team Member

Member	Top-3 Success	Recal@3	MRR@3
Hwang ○○	48 / 50	96.0%	92.7%
Son ○○	41 / 50	82.0%	80.0%
Lee ○○	41 / 50	82.0%	78.0%
Lee ○○	12 / 50	24.0%	21.0%

4.3 시스템 리소스 요구사항

온프레미스 환경에서 로컬LLM, 임베딩 모델(bge-m3), 재순위화 모델(bge-reranker-v2-m3)을 함께 운용하기 위한 최소 사양을 실제 운영PC와 로그 기준으로 실측하

〈Table 3〉 Minimum System Requirements and Initial Disk Footprint

GPU VRAM	4GB (GTX 1650)	8GB (RTX 5060)	12GB+ (RTX 5070)
Typical Use	Tight margin; large uploads fall back to CPU embedding	Stable single-user operation	Concurrent multi-user operation
Disk (Docker: Qdrant+PostgreSQL+Ollama)	~9GB	~9GB	~9GB
Disk (Embedding + Reranker Models)	6.5GB	6.5GB	6.5GB
Disk (LLM Weights, model-dependent)	~5-10GB	~5-10GB	~5-10GB
Total Initial Disk	~20-25GB	~20-25GB	~20-25GB

였다. 결과는 〈표3〉과 같다. GPU VRAM은 단일 사용자 기준 최소 4GB(GTX 1650)에서 동작 가능하나 대량 문서 업로드 시 임베딩 연산이CPU로 전환될 만큼 여유가 빠듯하며, 안정적인 단일 사용자 운영을 위해서는 8GB(RTX 5060) 이상을, 다중 사용자 동시 운영을 위해서는 12GB(RTX 5070) 이상을 권장한다. 초기 설치 디스크 용량은Qdrant·PostgreSQL·Ollama로 구성된 Docker 이미지가 약 9GB, 임베딩·재순위화 모델이 약 6.5GB, LLM 가중치가 모델 크기에 따라 5~10GB로, 총 20~25GB 내외의 디스크 공간이 필요하다.

5. 결론

본 연구는 D사와의 산학연계 프로젝트를 통해 온프레미스 환경에서 동작하는 LLM RAG 기반 사내 문서 업무 자동화 시스템을 설계하고 구현한 실증 사례를 제시하였다. 질의 확장, Parent-Child 청킹, 2단계 재순위화로 구성된 RAG 파이프라인과 하이브리드 검색을 결합함으로써 담당자에 따라 최대Recall@3 96.0%, MRR@3 92.7%의 검색 성능을 확보하였으며, document_type 컬럼을 활용한 확장 가능한 데이터베이스 스키마를 통해 시스템 구조 변경 없이 다양한 문서 서식을 수용할 수 있음을 보였다. 아울러 실제 운영 환경 실측을 통해 최소 4GB에서 다중 사용자 기준12GB 이상의 GPU로 시스템을 운용할 수 있는 사양 기준을 제시하였다. 다만 본 연구는 몇 가지 한계를 가진다. 첫째, 검색 성능 평가는 4명의 구현 담당자가 자체 구성한 질의-정답 쌍을 기준으로 수행되어 평가 데이터의 규모와 다양성이 제한적이며, 담당자 간 성능 편차가 크게 나타난 원인에 대한 정량적 분석이 충분히 이루어지지 못하였다. 둘째, 가상의 주간보고서 데이터를 활용하였기 때문에 실제 운영 데이터를 반영한 장기간 현장 검증이 추가로 필요하다. 향후 연구에서는 질의 확장 프롬프트와 청킹 경계 설정의 표준화를 통해 담당자 간 성능 편차를 줄이고, 실제 제조 현장의 다개월 운영 데이터를 기반으로 한 종단적 성능 검증을 수행할 계획이다. 또한 저자가 선행 연구에서 수행한 중소기업 정보보호 컨설팅 및 보안 취약점 대응모델 설계 연구[6]의 문제의식을 확장하여, 본 연구의 온프레미스 RAG 시스템에 대한 보안 취약점 진단과 접근통제·감사로그 체계를 정교화함으로써 중소제조기업이 안심하고 도입할 수 있는 전사적AX 보안 체계로 발전시키는 것을 향후 연구 과제로 삼고자 한다.

REFERENCES

[1] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of Hallucination in Natural Language Generation," ACM Computing Surveys, Vol.55, No.12, pp.1-38, 2023.

[2] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Advances in Neural Information Processing Systems, Vol.33, pp.9459-9474, 2020.

[3] C. Jeong, "Generative AI Service Implementation Using LLM Application Architecture: Based on RAG Model and LangChain Framework," Journal of Intelligence and Information Systems, Vol.29, No.4, pp.129-164, 2023.

[4] 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.69-80, 2025.

[5] S.W.Oh, Y.G.Ji, S.C.Lee, and N.Y.Kim, "Digital Transformation of Small and Medium Manufacturers: Ideal and Reality," The Journal of Society for e-Business Studies, Vol.27, No.4, pp.135-151, 2022.

[6] K.H.Lee, "Security Vulnerability Analysis and Response Model Design through Information Security Consulting for Small and Medium-sized Enterprises," Journal of Internet of Things and Convergence, Vol.11, No.1, pp.9-15, 2025.

[7] V. Karpukhin, B. Oğuz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W. Yih, "Dense Passage Retrieval for Open-Domain Question Answering," Proc. of EMNLP, pp.6769-6781, 2020.

[8] S. Robertson and H. Zaragoza, "The Probabilistic Relevance Framework: BM25 and Beyond," Foundations and Trends in Information Retrieval, Vol.3, No.4, pp.333-389, 2009.

[9] Y. A. Malkov and D. A. Yashunin, "Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.42, No.4, pp.824-836, 2018.

[10] S. Xiao, Z. Liu, P. Zhang, N. Muennighoff, D. Lian, and J. Nie, "C-Pack: Packed Resources for General Chinese Embeddings," Proc. of ACM SIGIR, pp.641-649, 2024.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," Advances in Neural Information Processing Systems, Vol.30, 2017.

[12] H. Touvron et al., "LLaMA: Open and Efficient Foundation Language Models," arXiv:2302.13971, 2023.

[13] L. Gao, X. Ma, J. Lin, and J. Callan, "Precise Zero-Shot Dense Retrieval without Relevance Labels," Proc. of ACL, pp.1762-1777, 2023.

[14] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, "Lost in the Middle: How Language Models Use Long Contexts," arXiv:2307.03172, 2023.

[15] R. Nogueira and K. Cho, "Passage Re-ranking with BERT," arXiv:1901.04085, 2019.

이 근 호(Keun Ho Lee) [종신회원]

- 2006년 8월 : 고려대학교 컴퓨터학과(이학박사)
- 2006년 9월 ~ 2010년 2월 : 삼성전자 DMC연구소 책임연구원
- 2010년 3월 ~ 현재 : 백석대학교 컴퓨터공학부 교수

〈관심분야〉
AX, AI보안, 침해사고대응, 융합보안, 개인정보보호, 블록체인, 산업보안, 취약점분석, 모의해킹 등