

컴퓨터교육학회 논문지 2026년 제29권 제9호
<https://doi.org/10.32431/kace.2026.29.9.002>



RAG 기반 항공기 정비교육 AI 챗봇 시스템 개발 및 적용 방안 연구*

Research on the Development and Application Methods of a RAG-Based Aircraft Maintenance Training AI Chatbot System

김영수[†] 
 Youngsoo Kim[†]

요약

본 연구는 RAG(Retrieval-Augmented Generation) 기반 항공기 정비교육 AI 챗봇 시스템을 개발하고 성능 향상을 통한 적용방안을 도출한다. RAG 기반 챗봇 시스템 구축 시 발생하는 방대한 파이프라인 조합의 복잡성을 해결하고 적용 방안을 강구하기 위해, 오픈소스 활용, 한국어 지원, 상용 라이선스 확보 등을 고려한 국내에 활용 가능한 최적의 기술 스택을 실험적으로 도출하였다. 연구 결과, 제안된 시스템은 기존 단순 챗봇 대비 응답 품질 지수 5점을 기준으로 2.6에서 3.6으로 약 38% 향상시켰으며, 평균 응답 속도는 기존 17분 54.83초에서 5.17초로 대폭 단축하였다. 이를 통해 정비사의 오류를 줄여서 대형 항공기 사고를 예방하는 데 도움을 줄 수 있는 항공기 정비교육 AI 챗봇 시스템의 적용 가능성을 확인할 수 있었다.

주제어 지능형 항공기 정비교육 시스템, AI 챗봇, 검색 증강 생성, 거대언어모델

ABSTRACT

This paper develops a RAG (Retrieval-Augmented Generation)-based AI chatbot system for aircraft maintenance training and derives application methods through performance enhancement. To resolve the complexity of the vast pipeline combinations involved in building a RAG-based chatbot system and to devise application methods, an optimal technology stack applicable in Korea was experimentally derived, considering factors such as the use of open source, Korean language support, and securing commercial licenses. As a result of the study, the proposed system improved the response quality index by approximately 38% from 2.6 to 3.6 (based on a 5-point scale) compared to existing simple chatbots, and significantly reduced the average response speed from the existing 17 minutes and 54.83 seconds to 5.17 seconds. Through this, the applicability of an aircraft maintenance training AI chatbot system was confirmed, which can help prevent major aircraft accidents by reducing technician errors.

Keywords Intelligent Aircraft Maintenance Training System, AI chatbot, Retrieval-Augmented Generation (RAG), Large Language Model (LLM)

†정회원 전주대학교 인공지능학과 조교수(단독
저자)
 논문투고 2026년 04월 10일
 심사완료 2026년 06월 16일
 게재확정 2026년 06월 23일
 발행일자 2026년 09월 03일

* 본 연구는 2026년도 교육부 및 전라북도의 재원으로
 전북 RISE센터의 지원을 받아 수행된 지역혁신중심 대
 학지원체계(RISE)의 결과입니다.(74260012)

1. 서론

항공기 정비는 항공기의 안전한 운항을 위해 고려할 가장 중요한 요소 중 하나이다. 정비사의 작은 실수나 정비 지침의 위반은 커다란 항공사고로 이어져서 막대한 인명과 재산 피해를 발생시킨다. 2005년 8월 14일 발생한 헬리오스 항공 522편 추락으로 인한 탑승자 121명 전원 사망사고는 정비사가 정비 간 기내 압력 조절 스위치를 '수동' 모드로 전환한 후, 정비 완료 후 이를 원위치로 되돌리지 않아 발생한 참사였다[1]. 2012년에는 대한민국 공군의 블랙이글 특수비행팀 소속 T-50B 항공기가 추락하여 조종사가 사망하였는데, 이는 정비사가 항공기 엔진의 연료 공급 장치에 있는 오일 필터를 잘못 설치하여 엔진이 정상적으로 작동하지 않아서 발생한 것으로 드러났다[2]. 2016년에는 (주)한라스카이에어의 정비사가 정비 미흡으로 연료 호스에 이상이 발생함으로써 지속적인 연료 누출로 인한 엔진 실린더 마모로 기체가 추락해 탑승자 2명 전원이 사망하는 사건이 발생했다[3]. 이 외에도 정비사의 작은 실수로 인해 큰 사고로 이어진 사례들을 어렵지 않게 찾을 수 있다.

항공기 정비사의 인적오류를 줄이기 위한 다양한 교육적 시도와 연구가 있었다. 정비사의 교육을 체계적으로 관리하거나 가상현실/혼합현실(VR/XR)과 같은 첨단 정보통신기술(ICT)을 활용하기도 하였다[4-6]. 최근 인공지능(AI)의 거대언어모델(Large Language Model, LLM)이 급격한 발전을 이루면서 교육관련 서비스에도 적용이 확대되고 있다[7]. 하지만 아직 LLM을 항공기 정비분야에 대한 교육 지원 서비스에 적용한 사례 및 연구는 부재하다. 정비사의 정비교육을 위해서는 정확한 정보를 제공해야 하는데 LLM은 아직 잘못된 정보를 생성하는 환각(Hallucination) 현상이나 전문성이 필요한 질문에는 다소 부족한 모습을 보이기 때문이다.

RAG(Retrieval-Augmented Generation)는 사용자의 질의와 유사한 문서를 검색한 후 질의와 문서를 LLM에 전달하여 LLM이 실제 문서에 기반한 근거 있는 답을 생성하는 방식으로 기존 LLM의 환각이나 부정확한 정보의 문제를 크게 개선할 수 있다[8]. 이는 최근 여러 분야의 챗봇 시스템 개발에 RAG가 적용되어 그 성능이 입증되었다. 하지만 RAG를 활용한 챗봇 개발 프로세스는 문서 로더부터 답변 생성에 이르기까지 여러 단계로 구성되어 있고 각 단계별로 적지 않은 후보도들이 상호 복잡적으로 연결되어 수많은 선택조합이 존재한다. 따라서 특정한 서비스에 최적화된 RAG 기반의 챗봇 시스템을 설계하고 개발하는 것은 복잡하고 고려사항이 많다[9].

본 논문에서는 항공기 정비교육에 최적화된 챗봇 서비스를 개발하기 위해 많은 RAG의 각 단계별 선택사항들에 대해 실험적으로 검증하여 선택하고 성능 향상 방안과 활용성을 탐색하고자 한다. 2장에서는 RAG 기반의 AI 챗봇 서비스 개발 관련 연구를 살펴보고, 3장에서는 활용 데이터의 특성과 항공기 정비교육 챗봇 개발을 위한 RAG 단계별 선택

사항에 대한 실험 및 검증 그리고 성능향상 방안을 다룬다. 마지막으로 4장에서는 본 논문을 통해 수행된 연구결과를 정리하고 향후 연구 방향을 제시하고자 한다.

2. 관련 연구

RAG를 활용한 다양한 분야의 LLM 기반 챗봇 서비스는 최근 활발히 연구되고 있는 분야이다. 도메인 전문 지식 데이터베이스를 구축하여 실시간 검색이 가능하고 높은 정확도와 보안을 강화할 수 있으며 별도로 부가비용이 요구되지 않기 때문에 국내외에서 다양한 분야에 빠르게 확산되고 있고 다양한 사례와 전략들이 제안되었다.

Jeong, M. 외 3명[10]은 RAG를 활용하여 건설 안전 관련 질문에 대해 보다 정확하고 관련성 높은 답변을 제공하는 Q&A 시스템 구성을 제안하였다. 이 논문에서는 미국 산업안전보건청(OSHA) 안전 규정 문서와 같은 외부 문서들을 참조하도록 함으로써 LLM이 더욱 정확하고 사실적인 답변을 생성하도록 하였다. 저자들은 본 연구를 통해 기본 RAG(Naive-RAG), 재순위 RAG (Rerank-RAG), 그리고 반복적 탐색 생성(Iterative Retrieval Generation)이라는 세 가지 적용 방식을 비교 실험함으로써 건설 안전 분야에서 RAG 기반 최적화된 Q&A 챗봇 시스템의 실현 가능성을 제시하였다. 하지만, 실제 건설 현장의 데이터가 충분하지 않은 시뮬레이션 환경에서 수행되었고 현장 테스트를 통한 검증이 이뤄지지 않았다.

Arslan, M. 외 2명[11]은 기후 변화, 인구 증가 및 자원 부족으로 인해 도시의 물 관리 시스템이 점점 더 중요해짐에 따라 이와 관련된 정보들을 쉽게 얻을 수 있는 Q&A 챗봇인 SustainWaterBot을 제안하였다. RAG기법을 적용하여 단순한 LLM 기반 답변이 아니라 정확하고 신뢰할 수 있는 데이터 기반 답변을 제공하도록 설계됐으며, Llama 3.1 개방형 LLM 모델을 활용하여 비용 효율적으로 기업들이 쉽게 접근하고 활용할 수 있도록 지원하였다. 또한 SustainWaterBot은 뉴스, 정부 발간물, 산업 보고서, 과학 연구, 소셜 미디어 플랫폼 등 다양한 출처의 데이터를 통합하여 정보 격차를 해소하는 통합 정보 시스템 역할을 한다. 하지만 본 논문에서는 기본 RAG 시스템 수준으로 개발되었으며, 성능 평가도 정밀도(Precision), 재현율(Recall), 그리고 F1-Score에 대해서만 수행하였다. 따라서 최적의 RAG 시스템 구축을 위한 각 단계별 다양한 선택사항에 대한 추가적인 연구가 필요하다.

Liu, F. 외 2명[12]은 자동차 산업 분야 종사자 보조를 위해 Langchain 기반 RAG 시스템을 개선하여 자동차 산업 PDF 문서 안의 기술 사양과 같은 복잡한 자동차 산업 문서에서 필요한 정보를 검색할 수 있는 챗봇을 설계하였다. 특히, 복잡한 자동차 산업의 문서 특성에 맞춰 다중 열 레이아웃과 같은 복잡다단한 PDF 문서들을 효율적으로 분석할 수 있도록 다양한 방식으로 최적화하였다. 또한 로컬 환경에

서 실행 가능한 RAG 시스템 구축을 통해 클라우드 기반이 아닌 온프레미스에서 정보 검색 및 챗봇을 활용할 수 있도록 하였다. 더하여 LLM이 스스로 정보를 검색할지 판단하고, 검색 결과를 반영하여 답변을 개선하는 Self-Reflective RAG 기법을 도입하여 RAG 시스템이 보다 정확하고 신뢰할 수 있는 정보 제공이 가능하도록 개선하였다. 하지만 제안된 시스템은 자동차 산업에 최적화된 모델로 설계되어 다른 산업으로의 확장성이 부족하고, 챗봇의 특성상 산업 환경에서는 반응속도가 중요하기 때문에 처리속도를 단축시키는 연구가 필요하다.

Quidwai, M. A. 외 1명[13]은 의료분야에서 다발성 골수종(Multiple Myeloma, MM) 치료에서 정밀 의학을 지원하기 위해 RAG를 활용하여 MM 연구 논문을 실시간으로 검색하고 분석하여 의료진에게 신뢰성 높은 정보를 제공하는 챗봇 시스템을 제안하였다. 임상자들은 방대한 유전체 데이터와 생물학적 문헌을 활용하는 것이 매우 복잡하고 빠르게 변화하는 MM 연구 및 치료법을 최신 상태로 유지하기 어려운 상황에서 챗봇을 통해 MM 관련 논문 및 임상 데이터를 실시간으로 검색하고 환자 맞춤형 치료법을 추천하는 자동화된 AI 시스템을 구축하였다. 이를 위해 BioMed- RoBERTa-base 모델을 활용한 문서 임베딩 생성과 Mistral-7B 모델을 이용하여 질의답변 시스템을 결합하였으며 Amazon OpenSearch Service와 Amazon Kendra를 활용하여 빠르고 정확한 정보 검색 및 시각화 기능을 개발하였다. 하지만, RAG 시스템 최적화를 위한 단계별 검증이 부족하고 실제 의료 환경에서 챗봇의 응답속도의 적절성에 대해서도 의료진의 피드백을 반영한 추가적인 개선 연구가 필요하다.

본 논문에서는 실제 UH-60 항공기 정비 문서를 바탕으로 항공기 정비교육 챗봇 시스템의 성능향상방안을 분석 및 도출하기 위해 RAG의 각 단계별 선택사항들에 대해 실험을 통해 최적의 방안을 찾고 실제 환경에서의 적용가능성을 입증한다.

3. RAG 기반 AI 챗봇 개발 및 고려사항

3.1 RAG의 개념 및 프로세스

RAG는 단순히 문서를 찾는 것을 넘어, 데이터 전처리부터 검색 후 정제에 이르는 정교한 파이프라인을 필요로 하며 그 과정은 다음과 같다. 먼저, 원시 데이터를 지식으로 변환하는 과정(Chunking), 질문의 의도를 파악하기 위한 사전 검색(Pre-retrieval), 데이터베이스에서 관련 문서를 검색하는 과정(Retrieval), 그리고 최적의 문서를 선택하는 사후 처리(Post-retrieval)이다.

첫 번째 단계인 청킹(Chunking)은 Eq. (1)에 따라 방대한 텍스트(D)를 모델이 처리하기에 적절한 크기의 청크(c_i)로 나누는 것이다. 분할 방법에는 고정 크기 분할(Fixed Chunking), 재귀적 분할(Recursive Chunking), 의미적

분할(Semantic Chunking) 등이 있다. 일반적으로 청크 간 적절한 크기의 중복을 두어서 문맥 손실을 최소화한다.

$$D = c_1, c_1, \dots, c_n \quad (1)$$

사전 검색(Pre-retrieval) 단계에서는 쿼리(q)를 LLM에 입력하여 쿼리 재작성(Query Rewrite), 쿼리 확장(Query Expansion), 또는 쿼리 변환(Query Transformation)을 수행한다. 일반적으로 사람은 구조화된 질의를 하기 어렵기 때문에 이 과정을 수행하여 쿼리를 더욱 정교하게 다듬어 검색 정확도를 향상시키는 것이다. Eq. (2)와 같이 사용자의 질의(q)를 검색에 최적화된 형태의 쿼리(q')로 변환하는 것이다. 여기서 쿼리 q는 LLM으로부터 생성된 결과 y_{hypo}가 병합된다.

$$q' = Embed(q \oplus y_{hypo}) \quad (2)$$

검색(Retrieval) 단계는 변환된 쿼리(q')와 벡터 공간에서 가장 유사한 k 개의 청크를 찾는 과정이며, 이는 Eq. (3)에 나타나 있다. 사전 학습된 검색기는 쿼리 q'가 주어졌을 때 지식 소스 Z에서 가장 관련성이 높은 k 개의 문서 c_{1,...,k}를 선택한다. 보통 코사인 유사도, 유클리디안 거리 또는 내적을 통해 관련성을 측정한다. E_q와 E_z는 각각 쿼리와 문서를 밀집 벡터로 매핑하는 인코더 함수(예: DPR, BERT)이고, sim은 관련성(유사도)을 측정하는 함수이다.

$$c_{1, \dots, k} = \arg \max_{z \in Z} sim(E_q(q'), E_z(z)) \quad (3)$$

사후 검색(Post-retrieval) 단계에서는 검색된 청크가 실제로 생성에 도움이 되는지 검증하거나 순서를 조정한다. 예를 들어, 벡터 유사도만으로는 포착할 수 없는 세밀한 의미적 연관성을 재계산하기 위해 Eq. (4)와 같이 교차 인코더 모델을 사용하여 재순위화(Re-ranking)하는 기법이나, 추출된 문서에서 중복되거나 불필요한 노이즈를 제거하고 LLM의 입력 제약 조건(Context window) 내의 핵심 정보만 제공하는 컨텍스트 압축(Compression) 기법 등이 사용된다.

$$C_{rerank} = f_{model}(q', c_{1, \dots, k}) \quad (4)$$

마지막으로, 정제된 컨텍스트 C_{final}은 쿼리와 결합되어 Eq. (5)와 같이 답변 y를 생성한다.

$$P(y|q', C_{final}) = \prod_{i=1}^T P(y_i | y_{<i}, q', C_{final}) \quad (5)$$

3.2 RAG 단계별 고려 사항

RAG의 각 단계에서는 다양하고 많은 선택사항들이 존재한다. 따라서 특정 도메인에 맞는 최적 조합을 찾기 위해 모든 조합에 대해 실험하는 것은 현실적으로 제한된다. 따라서 RAG 각 단계에 맞는 모듈을 찾기 위해서는 해당 영역의 비즈니스와 관련 문서의 특성들을 복합적으로 고려해야 한다.

본 논문에서는 국내에서 자유롭게 활용 가능한 항공기 정비교육 AI 챗봇을 개발하기 위해 오픈 소스이고 상업적 라이선스가 허용되며 한국어와 영어가 모두 지원되는 선택 사항들만을 대상으로 하여 각 단계별로 후보 모듈들을 선택하였고 실험을 통해 최적의 모듈을 선택하였다. 이렇게 RAG의 시작 단계에서부터 단계적으로 선택되고 검증된 최적의 선택조합들을 연결하여 최종적으로 서비스로 적용 가능한 최적화된 항공기 정비교육 AI 챗봇 시스템을 구성하였다.

3.3 데이터 세트

본 연구에서는 UH-60 다목적 헬리콥터의 영어 정비 문서의 번역본 한글 정비 문서를 활용하였다. UH-60은 전 세계적으로 널리 사용되는 헬리콥터로, 록히드 마틴 사의 회사인 시코르스키 에어크래프트(Sikorsky Aircraft)에서 생산하여 지금도 수색, 구조, 수송 및 공중 지휘통제용 등 민간 및 군사용으로 활발하게 사용되고 있다. 한국에 있는 정비사들을 지원하기 위한 본 프로젝트에서는 UH-60 정비 문서가 기본적으로 영어로 작성되어 있기 때문에 Fig. 1과 2와 같이 구글 번역기를 활용하여 한글로 번역하여 데이터로 활용하였다. UH-60 정비 문서는 총 150개의 문서로 구성되어 있고 내용은 크게 항공기 정비 매뉴얼, 기술 지침서, 부품 목록 및 도해, 고장 이력 및 진단 데이터, 규제 및 안전 지침, 그리고 교육 자료 및 FAQ로 구성되어 있으며 세부 내용 구성은 다음과 같다.

- 항공기 정비 매뉴얼(AMM) : UH-60의 각 부품 및 시스템에 대한 상세한 작동 원리, 점검 방법, 조립 및 분해 절차 등이 기술되어 있다.

- 기술 지침서(SB, TSB) : 제조사에서 제공하는 UH-60 관련 서비스 공지로, 제품 개선 사항, 수정된 정비 절차, 안전 관련 업데이트 등이 포함된다.

- 부품 목록 및 도해(IPC) : UH-60의 부품 일러스트와 함께 부품 번호, 명칭, 사양 등을 제공하여 정확한 부품 식별과 주문을 돕는다.

- 고장 이력 및 진단 데이터 : UH-60의 과거 고장 사례와 진단 결과를 기록한 데이터로, 문제 해결에 유용한 정보를 제공한다.

- 규제 및 안전 지침 : UH-60 운용과 관련된 항공 안전 법규, 규제, 국제 표준 등의 정보를 포함한다.

- 교육 자료 및 FAQ : UH-60 정비사 교육에 사용되는 자료와 자주 묻는 질문과 답변을 포함한다.

번역된 문서들은 여러 가지 측면에서 성능을 저하시키

는 요소들을 포함한다. Fig. 1과 2 같이 이미지, 텍스트, 도표 등 다양한 형식의 입력 데이터들로 구성되어 있어 번역 과정에서 그 형식과 순서가 깨지는 경우가 빈번하였다. 또한 항공기 전문 영역인 UH-60 정비 분야 특성상 전문 용어와 영어로 된 약어가 많이 사용되어 정확한 의미 파악이 어렵고 일부 오역이 발생하기도 하였다. 이를 해결하기 위해 정규표현식(Regex) 기반의 특수문자 및 깨진 텍스트 필터링, 표나 데이터의 순서 깨짐을 최소화하는 규칙기반 파서(Parser)를 적용하여 정형화된 문맥을 유지하였다.

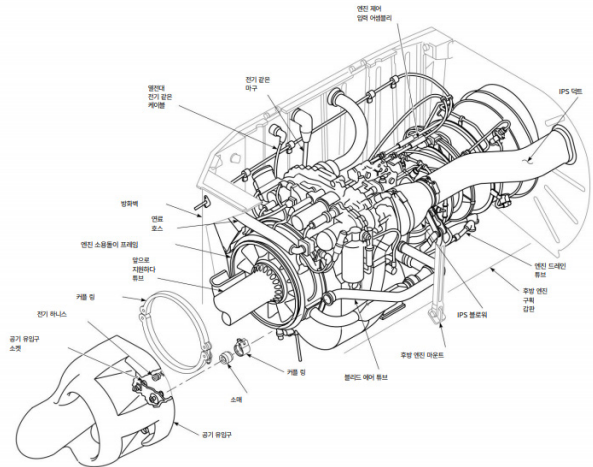


Figure 1. An illustration in the translated UH-60 maintenance document

엔진 제어 시스템

각 엔진의 출력 제어 시스템은 제어 시퀀스에 있는 엔진 출력 제어 레버를 통해 작동합니다. 이 레버를 사용하면 시작 및 시퀀스-중단이 가능합니다. 각 레버는 OFF, IDLE, FLY 및 LOCKOUT의 네 가지 위치가 있습니다. 출력 제어 레버는 무시-물 제어를 통해 엔진 연료 제어 장치의 출력 이동 가능한 스핀들을 연결합니다. 스핀들은 기계적으로 스톱클 연료를 사용하거나 IDLE과 FLY 사이에서 작동하기 위한 허용 가스 발생기(Ng) 속도를 설정할 수 있습니다. 출력 제어 레버를 LOCKOUT으로 이동하면 전기 제어 장치(ECU) 또는 유체 역학 장치에 입력되는 DEC가 기계적으로 잠기고 출력 제어 레버를 스톱클로 사용하여 가스 발생기(Ng) 속도를 수동으로 제어할 수 있습니다.

속도 제어 시스템

엔진 속도 제어 시스템은 모터 구동 전위차계 시스템과 ECU 또는 DEC를 통해 작동합니다. 이 시스템은 엔진의 RPM을 측정하고, 이를 ENG RPM 스위치를 누르면 컴퓨터로 스틱 또는 1-600 HH-60L로 전달합니다. INC 또는 DEC에 대한 전기 신호는 전위차계를 통해 엔진의 ECU 또는 DEC로 전달됩니다. 이 신호는 차례로 각 엔진의 HMM을 전위차계로 조정하여 엔진 출력 터빈 속도 Np(%) RPM 1 또는 2를 증가 또는 감소시켜 엔진 로터 속도 N(%) RPM RTR를 96%에서 100% 사이로 증가시킵니다.

UH60L HH-60L 엔진의 액세서리 모듈에 Nr 센서가 추가되었습니다. Nr 센서는 메인로터 RPM을 감지합니다. 이 신호는 엔진 속도 제어를 위해 두 엔진의 DEC로 전송되고 계기판 디스플레이를 위해 두 SDC로 전송됩니다.

Figure 2. Texts in the translated UH-60 maintenance document

성능 평가 데이터는 항공기 정비 경험이 많은 정비교관과 현장 정비사 3인으로 구성된 전문가 집단의 교차 검증을 통해 모범답안(Ground truth)을 만들었으며, 문서를 요소, 관리 방법, 절차, 용어로 범주화하고 각 범주별 자주 발생할 수 있는 핵심문항으로 25개씩 총 100개의 질의 및 답변(Q&A)을 구성하여 성능을 측정하였다. 또한 생성된 답변은 유창성(Fluency), 일관성(Coherence), 정확성(Accuracy), 완전성(Completeness)의 4가지 지표에 대해 Table 1과 같이 5점 만점 기준으로 GPT-4 기반의 LLM 심판(LLM-as-a-Judge) 방식[14]을 적용하여 평가하였다. 평가 결과의 편향성(Bias)이나 환각(Hallucination) 등의 오류 발생을 방지하고 평가결과의 신뢰도를 향상시키기 위해

전문가그룹에 의한 추가적인 검증(Human-in-the-loop)을 수행하였다.

Table 1. Evaluation criteria for LLM/chatbot responses

구분	점수	핵심기준
매우 우수	5	매우 정확하고 완전한 답변
우수	4	대체로 정확하나 미세한 보완 필요
보통	3	핵심적인 답변을 제공하나 일부 오류 포함
미흡	2	핵심 오류가 있어 활용이 제한
매우 미흡	1	완전히 틀리거나 무관한 정보를 출력

3.4 실험 환경

사용자는 주로 훈련 정비사이기 때문에 시스템 프롬프트는 Table 2와 같이 구성하여 질의 속의 기술 용어와 같은 전문적인 단어들은 유지하고 오타, 문법 오류 또는 불명확하고 관련 없는 질문들은 수정·보완하여 더 명확하고 간결하게 개선하여 검색 효율을 극대화하였다. 또한, 기술 문서 수준의 전문적인 어투를 사용하여 전문적인 대화 톤을 유지하며 훈련과 무관한 대화는 하지 않도록 하였다.

Table 2. System prompt

```
{
  "role": "system",
  "content": ""You are an AI designed to refine and clarify user questions for an aircraft maintenance specialist model. Your task is to:
  1. Correct any typos or grammatical errors while maintaining the technical terminology.
  2. Reorganize the questions to ensure they are concise and contextually clear.
  3. Maintain a professional tone suitable for technical documentation.
  4. Do not provide any answers, solutions, or explanations to the underlying question. Your sole purpose is to produce an improved version of the user's question.
  If the user asks for clarification on the APU, engine operation, or any other technical topic, you must refrain from answering directly and instead return a refined question.
  5. Even if the user asks unrelated questions (e.g., greetings like "Hello" or "What is your role?"), do not provide any information or explanations. Instead, refine and restate the user's question in a clearer and more concise manner without addressing the content. Your sole role is to improve the question itself, not to answer it. ""}
```

시스템 개발 및 성능 평가는 CPU는 Intel(R) Xeon(R) Gold 5218R CPU @ 2.10GHz X 80Cores, GPU는 A100 80GB, RAM 396GB, 그리고 HDD는 1.8TB인 환경에서 진행하였다. 이에 따라 본 논문에서는 RAG의 성능에 가장 중요한 부분인 청킹(Chunking), 검색(Retrieval), 그리고 언어모델(LLM)을 대상으로 가장 성능향상을 가져올 수 있는 방안을 찾기 위한 실험을 수행하였다.

3.5 청킹(Chunking)과 검색(Retrieval)

RAG는 답변의 정확도와 처리속도의 향상을 위해 큰 문서들은 작은 문서단위(Chunk)로 나뉘어서 임베딩을 통해 데이터베이스에 저장해야 한다. 청킹에는 크게 고정

글자 수를 기준으로 자르는 고정크기 청킹(Fixed-sized chunking), 구분자를 사용하여 계층적이고 반복적으로 자르는 재귀적 청킹(Recursive chunking), 그리고 문장 간의 의미적 유사성을 분석하여 자르는 의미적 청킹(Semantic chunking) 등이 있다. 어느 것이 더 효과적인가에 대해서는 문서들의 특성에 따라 좌우된다.

검색에는 크게 키워드 검색(Keyword search), 벡터 검색(Vector search), 그리고 이 둘을 조합한 하이브리드 검색(Hybrid search) 방식이 있다. 키워드 검색은 간단하고 빠르게 문서를 추출할 수 있는 반면에 동의어나 유의어에 의한 문맥을 정확히 이해하는데 제한이 있다. 벡터 검색은 이러한 키워드 검색의 단점을 보완할 수 있으나 계산 비용이 높다. 하이브리드 검색이 키워드 검색과 벡터 검색의 비율을 조절하여 정확도를 향상시킬 수 있으나 개발과 계산 복잡성은 증가한다.

본 연구에서는 키워드 검색과 벡터 검색에서 각각 가장 많이 쓰이는 BM25(Best Matching 25), S-BERT(Sentence-BERT)와 이를 조합한 Hybrid Search를 비교 실험하였다. 이때, Hybrid Search 방식의 비율(키워드 : 벡터)은 각각 3:7, 5:5, 7:3으로 진행했으며, 평가할 청크의 수(top-k)를 5개로 지정하였다. 또한, 검색의 효율성을 높이기 위해 검색 전에는 EuroLLM-9B-Instruct 모델을 통해 질의를 LLM이 보다 잘 이해할 수 있도록 보완하는 질의 재작성(Query Rewrite) 기법을 적용하였고, 검색 후에는 Dongjin-kr/ko-reranker 모델을 통한 재순위(Rerank) 기법을 사용하여 정확도를 높였다. 검증 데이터에 대한 청크의 적정성 평가는 청크가 얼마나 정보를 가졌는지에 대한 가치성을 평가하였으며, 종합적인 결과는 Table 3과 같다.

청킹 방식에서는 재귀적 청킹(평균 3.51)은 가장 낮은 성능을 보였으며, 의미적 청킹(Semantic Chunking)의 일부 설정에서는 상대적으로 우수한 성능을 나타냈다. 특히 Semantic Chunking(percentile, 50)은 평균 3.90으로 가장 높은 점수를 기록하였다. 다만 Fixed Chunking 또한 평균 3.75의 높은 점수를 나타내어 일부 의미적 청킹 기법과 유사한 수준의 성능을 보였다. 이는 항공기 정비 문서가 기술 문서의 특성상 구조화된 것에 기인한 것으로 분석된다. 검색 방식에서는 Hybrid Search(7:3)가 평균 3.70으로 가장 높은 점수를 기록하였으나 BM25(3.69)와의 차이는 매우 미미하였다. 반면 Hybrid Search(3:7) 및 Hybrid Search(5:5)는 각각 3.52와 3.43으로 오히려 성능이 감소하는 경향을 보였다. 이러한 결과는 항공기 정비 문서가 전문 용어와 고유 명칭 중심의 기술 문서로 구성되어 있어 의미적 유사성보다 정확한 용어 일치가 검색 성능에 더 큰 영향을 미치기 때문으로 판단된다. 따라서 본 연구에서는 가장 높은 성능을 나타낸 Semantic Chunking(percentile, 50)과 Hybrid Search(7:3)를 최적 조합으로 선정하여 후속 실험에 적용하였다.

Table 3. Comparison of the Value of Retrieved Chunks according to Chunkers and Retrieval Algorithms

Chunkers	Retrieval Algorithms	BM25	S-BERT	Hybrid (3:7)	Hybrid (5:5)	Hybrid (7:3)	Average Chunking Point
Fixed		3.60	3.50	3.95	3.90	3.80	3.75
Recursive		3.55	3.60	3.70	2.95	3.75	3.51
Semantic(percentile, 50)		4.10	3.85	3.75	3.80	4.00	3.9
Semantic(standard deviation, 50)		3.90	3.55	3.20	3.35	3.65	3.53
Semantic(interquartile, 50)		3.70	3.90	3.00	2.95	3.30	3.37
Semantic(gradient, 50)		4.05	3.70	3.65	3.60	3.80	3.76
Semantic(percentile, 80)		3.85	3.65	3.75	3.60	3.90	3.75
Semantic(standard deviation, 80)		3.25	3.60	3.70	3.55	3.65	3.55
Semantic(interquartile, 80)		3.20	3.45	3.10	3.05	3.60	3.28
Semantic(gradient, 80)		3.70	3.45	3.40	3.55	3.50	3.52
Average Retrieval Point		3.69	3.63	3.52	3.43	3.70	

3.6 거대언어모델(LLM)

LLM은 RAG 기반의 챗봇 시스템에서 사용자의 질의와 벡터 DB에서 검색한 청크들을 가지고 사용자가 이해할 수 있는 문장으로 답변을 생성하는 중요한 역할을 수행한다. 본 논문에서는 지속 가능한 비용 구조와 내부 데이터의 보안을 위해 앞서 서술한 바와 같이 오픈소스이고 상용화가 가능한 라이선스를 갖고 영어와 한국어를 지원하는 조건을 충족하는 LLM들 중에서 실험을 통해 가장 정확도가 높고 빠른 답변을 생성하는 모델을 선택하였다. 본 프로젝트에서 고려 대상이 된 LLM들은 다음과 같다.

OrionStarAI에서 만든 Orion-14B-Chat-Int4는 140억 개의 파라미터를 가진 다중언어모델인 Orion-14B- Base를 int4로 양자화하여 모델 크기를 70% 줄이고 추론 속도를 30% 높인 대화모델로서 한국어를 비롯한 다국어를 지원하고 우수한 성능을 보이는 모델이다. eagle-3b-preview와 egpt-1.3b-preview는 모두 한국전자통신연구원(ETRI)에서 개발한 모델로서 과학기술(STEM)분야에 최적화되어 설계된 언어모델이다. AI 허브, KISTI, 위키피디아 등

대규모의 한국어 데이터셋으로 학습되었으며 성능과 효율성 간의 균형이 유지되도록 만들어졌다. 한성대학교 AI연구팀이 오픈소스 LLama3를 기반으로 Q&A 데이터를 학습시켜 개발한 Hansung-Blossom-8B는 한국어와 영어를 모두 지원하며 추가학습을 통해 한국어와 영어 간의 지식 연결을 강화하고 한국어 어휘를 확장하여 한국의 문화와 표현력을 강화시켰다. 고려대 NLP&AI 연구실에서 개발한 kullm-polyglot-5.8b-v2와 KULLM3는 한국어 특화 모델이다. kullm-polyglot-5.8b-v2는 EleutherAI의 polyglot-ko-5.8b 기반으로 하여 KULLMv2 데이터셋을 활용하여 파인튜닝을 통해 만들어졌다. KULLM3는 Upstage의 SOLAR-10.7B 모델을 기반으로 지시 튜닝(instruction-tuning) 학습을 통해 개발되었으며 영어와 한국어를 지원하고 성능이 우수한 것으로 평가된다.

기존 Naive LLM 모델들에 대한 실험결과를 살펴보면, Table 4의 상단부에서 보는 바와 같이 약간의 변동은 있지만 기본적으로 모델의 크기가 커질 수록 답변의 성능은 좋으나 속도가 급격히 느려지는 경향을 보인다. 답변 품질은

Table 4. Performance evaluation scores by model before and after vLLM application

LLM	Response Quality (1 ~ 5)					Time (sec.)	
	Fluency	Coherence	Accuracy	Completeness	Avg. Quality	Avg.	Total
RAGs							
KULLM3	4.95	4.25	3.5	3.5	4.05	166.72	3,334.38
kullm-polyglot-5.8b-v2	1.00	1.00	1.00	1.00	1.00	46.07	921.49
Hansung-Blossom-8B	1.95	1.85	1.35	1.4	1.64	70.48	1,409.64
eagle-3b-preview	1.25	1.25	1.2	1.2	1.23	65.16	1,303.20
egpt-1.3b-preview	1.3	1.25	1.25	1.25	1.26	29.26	585.21
Orion-14B-Chat-Int4	3.95	3.45	3.15	3.1	3.41	12.80	255.92
RAGs with vLLM							
KULLM3	4.20	3.70	3.05	2.9	3.46	11.32	226.32
kullm-polyglot-5.8b-v2	1.05	1.05	1.00	1.00	1.02	18.12	362.47
Hansung-Blossom-8B	2.30	1.60	1.10	1.15	1.54	20.80	415.96
eagle-3b-preview	1.00	1.00	1.00	1.00	1.00	40.04	800.71
egpt-1.3b-preview	1.25	1.05	1.05	1.05	1.10	19.50	391.00
Orion-14B-Chat-Int4	3.95	3.50	2.70	2.75	3.23	3.70	73.91

KULLM3가 평균 4.05로 가장 우수하나 답변속도는 166.72초로 매우 느리게 나왔다. Orion-14B-Chat-Int4는 답변품질이 3.41이고, 양자화를 통해 평균 답변속도도 12.8초로 준수하게 나왔다. 결론적으로, 답변 품질 측면에서는 3.0 이상이면 기본적으로 사용자의 질의에 대부분 의미있는 답변을 생성하기에 답변속도 개선이 필수적이라 할 수 있다.

3.7 LLM의 속도 개선 방안 분석

vLLM(Virtual LLM)은 운영체제의 페이지링과 가상 메모리 관리 기법에서 아이디어를 가져온 방식으로 GPU 메모리를 더욱 효율적으로 활용하여 LLM의 답변 생성 속도를 가속화하는 기법이다[15]. vLLM은 단기 기억장치인 KV(Key Value) 캐시 메모리에 LLM의 생성된 답변으로부터 키 값과 이에 대응하는 값을 저장한다. vLLM은 PagedAttention 기법을 활용한 KV 캐시를 페이지 단위로 동적 할당하여 GPU 메모리 낭비를 줄이고, 연속 배치(Continuous Batching)을 통해 LLM의 추론 처리량(Throughput)을 극대화하여 서비스 성능을 획기적으로 개선한다.

기존 RAG에 vLLM을 적용한 실험 결과는 Table 3의 하단부와 같다. 답변 품질이 가장 우수한 KULLM3에 vLLM을 적용한 결과, 답변의 품질은 3.46으로 다소 떨어졌으나 평균 답변 생성 시간은 166.72초에서 11.32초로 15배 이상 크게 단축되었다. Orion-14B-Chat-Int4 모델은 답변 품질은 약간 떨어져서 3.23을 나타냈지만, 답변 생성 속도는 기존 12.8초에서 3.7초로 사용자가 지루함을 느끼지 않을 만큼 빠른 답변을 생성하였다. 따라서 본 연구에서 활용된 번역문서의 개선의 여지가 많아 추후 답변 품질을 충분히 향상시킬 수 있는 것을 고려하면, Orion-14B-Chat-Int4 모델은 현재 수준의 운용환경에서도 충분히 활용 가능할 것으로 판단된다.

3.8 성능 종합 평가

항공기 정비교육 AI 챗봇 시스템에서 가장 중요한 요소는 사용자의 질의에 정확한 답을 적절한 시간 내에서 제공하는 것이다. 따라서 본 논문에서는 답변 품질도 양호하면서 속도가 빠른 OrionStarAI/Orion-14B-Chat-Int4 모델을 기준으로 항공기 정비 데이터의 특성에 맞게 RAG 각 단계별로 개선한 Improved RAG를 정비문서의 내용 범주별로 Naive RAG와 비교 실험하였다. 여기서 Naive RAG는 일반적으로 RAG 기반 AI 챗봇 시스템을 구축할 때, 많이 사용하는 선택옵션인 재귀적 청킹과 LLM만을 사용하는 경우를 의미하며 Improved RAG와의 차이점은 Table 5와 같다.

Table 5. Comparison of Naive RAG and Improved RAG

비교항목	Naive RAG	Improved RAG
분할(청킹) 방식	고정 크기 분할	의미적 분할
검색 방식 (Retrieval)	단일 벡터 검색	하이브리드 검색
질의 재작성 (Query Rewrite)	미적용	적용 (질의의 모호성 제거 및 키워드 확장)
검색결과 재순위화 (Reranker)	미적용	적용 (문맥 유사도 재순위화)
LLM 종류	기본 LLM 모델	도메인 특화 경량화/양자화 모델
vLLM	미적용	적용

실험 결과, Table 6에서 보는 바와 같이 전반적으로 Improved RAG가 Naive RAG에 비해 답변의 품질과 속도에서 모두 더 좋은 성능을 나타냈다. 답변 품질측면에서 모든 범주별로 Improved RAG가 성능이 좋은데, 그중에서도 무엇을 점검하고, 어떤 자원을 투입해야 하는지를 기술하는 요소 범주에서 상대적으로 더 좋은 성능을 보였다. 이는 의미적 청킹과 검색 전후의 질의 재작성과 재순위 기법의 영향에 따른 것으로 분석된다. 속도 측면에서도 Improved RAG가 Naive RAG보다 훨씬 좋은 성능의 서비스로 적용가능한 수준의 속도를 보임으로써, vLLM의 적용 효과를 확인할 수 있었다.

Table 6. Average Response Quality and Generation Time for Evaluation Queries

Category	Elements	Mgt. Methods	Procedures	Terminology
Average Response Quality				
Naive RAG	1.8	2.6	2.2	3.8
Improved RAG	4.2	3.0	3.4	3.8
Average Response Time (sec)				
Naive RAG	413.04	906.15	2,745.78	234.35
Improved RAG	5.50	5.72	5.76	3.70

정비문서의 각 범주별로 Naive RAG와 Improved RAG의 실제 답변을 생성하여 품질을 비교·분석해 보았다. Table 7에서 보는 바와 같이, 용어(Terminology)와 같은 단편적인 정보와 관련된 질문 범주에서는 양쪽 모두 큰 차이 없는 답변을 생성하였다. 하지만 관리 방법이나 절차 범주에서와 같이 세부적으로 문맥이 연결되는 정보가 요구되는 범주에 대해서는 Table 8과 같이 큰 차이가 발생하였다. 이는 Naive RAG에서는 전달된 청크에 해당 정보가 충분히 포함되지 않고 의미적인 문맥 연결의 부족에서 비롯된 것으로 분석된다.

Table 7. Responses of Naive RAG and Improved RAG in the Terminology Category

질문: TGT가 무엇인가?	
Naive RAG	Improved RAG
"TGT"가 특정 기술적 용어나 제품명일 가능성이 높습니다. 그러나 제공된 정보만으로는 "TGT"의 정확한 의미나 맥락을 파악하기 어렵습니다. "TGT"가 특정 기술적 장비나 시스템의 일부분일 수 있으며, 이는 기술적 문제나 상태를 나타내는 데 사용될 수 있습니다. 그러나 더 구체적인 정보나 문맥이 제공되지 않는 한, "TGT"의 정확한 의미를 파악하는 것은 어렵습니다.	TGT는 "Turbine Gas Temperature"의 약자로, 터보샤프트 엔진에서 터빈 가스의 온도를 의미합니다. 이 온도는 엔진의 성능과 안전 운영에 중요한 지표 중 하나로, 엔진의 효율성과 안전성을 보장하기 위해 철저히 모니터링되고 관리됩니다. TGT는 엔진의 온도 관리 시스템에서 중요한 역할을 하며, 엔진의 과열을 방지하고 최적의 작동 온도를 유지하는 데 필수적입니다.

Table 8. Responses of Naive RAG and Improved RAG in the Procedure Category

질문: 엔진 카울링 래치 제거 절차를 알려줘	
Naive RAG	Improved RAG
엔진 카울링과 래치를 닫습니다. 작업 종료 패키지	엔진 카울링 래치를 제거하는 과정은 다음과 같습니다. 먼저, 엔진 카울링을 열고, 엔진 기체와 연결된 래치 핸들 어셈블리를 분리합니다. 이러한 과정에서 나사와 와셔 너트를 사용하여 래치 핸들 어셈블리와 핸들 릴리스를 제거합니다. 또한, 엔진 카울링에 연결된 잠금 핀 튜브를 제거하고, 엔진 카울링에 연결된 래치 핸들 릴리스를 제거합니다.

4. 결론 및 향후 연구

본 논문에서는 항공기 정비사들이 정비 수행 간 실시간으로 정비교육을 받을 수 있는 RAG 기반의 AI 챗봇 시스템을 개발하고 성능 향상을 통해 적용가능성을 검증하였다.

AI 챗봇 시스템은 각 단계별로 실험을 통해 항공기 정비분야에서 가장 우수한 성능을 나타내는 모듈을 식별 및 통합함으로써 개발되었다. RAG의 의미적 청킹을 사용하고 검색에는 키워드와 벡터를 7 대 3으로 하는 하이브리드 검색 방식을 적용하며, LLM은 Orion- 14B-Chat-Int4를 기반으로 vLLM 적용하였을 때, 5점 만점에 평균 3.6의 양호한 답변 품질과 평균 5.17 초의 빠른 답변 속도를 기록하였다. 이를 통해 RAG 기반의 항공기 정비교육 AI 챗봇 시스템은 서비스로도 충분히 적용가능함을 확인하였다. 따라서 본 연구를 통해 RAG 기반 항공기 정비교육 AI 챗봇 시스템이 정비사의 교육에 기여하여 정비 오류를 줄이고 항공기 사고를 예방하는 데 충분히 기여할 수 있을 것으로 판단된다.

향후 연구과제로서, 다양한 기종의 많은 정비 문서를 추가하고 영어 원문서로부터 번역의 품질을 높이는 것이 필요하다. 특히, 항공기 정비문서들은 대부분 도면과 표 등이 들어가 있기 때문에 이미지를 정교하게 가독할 수 있는 방안이 강구되어야 한다. 또한, 더 많은 정비 전문가들의 피

드백을 반영하여 답변의 품질과 응답속도의 개선에 대한 연구가 이뤄져야 한다.

참고문헌

- [1] AAIASB. (2005). *American Airlines DC-10-10 Aircraft Accident Report, Helios Airways Flight HCY522 Boeing 737-31S at Grammatiko*.
- [2] Korea Air Force Accident Investigation Team. (2012). *Black Eagles Accident Investigation Report*.
- [3] Korea Ministry of Land, Infrastructure and Transport. (2016). *Aircraft accident investigation report*.
- [4] Nam, M. & Park, S. (2014). Evaluation Criteria Development of Courses Based on CIPP Evaluation Model Dedicated to Approval of Training Organizations. *Journal of the Korean Society of Aviation Operations*, 22(4), 8–14. <https://doi.org/10.12985/ksaa.2014.22.4.008>
- [5] Kim, S. (2021). Development of Aircraft Powerplant Maintenance Training Course Based on Virtual Reality. *Journal of the Korea Academia-Industrial Cooperation Society*, 22(7), 594–600. <https://doi.org/10.5762/KAIS.2021.22.7.594>
- [6] Park, S. & Bae, J. (2021). A Study on the Preventive Maintenance Daily (PMD) Training Method of XR-based Rotary Wing Aircraft. *Journal of the Digital Content Society*, 22(7), 1025–1030. <https://doi.org/10.9728/dcs.2021.22.7.1025>
- [7] Yoon, Y. & Kim, S. (2025). Trends and prospects of Retrieval-Augmented Generation (RAG) technology for generative AI. *Journal of the Korean Society for Computer Education*, 28(2), 69–80. <https://doi.org/10.32431/kace.2025.28.2.007>
- [8] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474. <https://doi.org/10.48550/arXiv.2005.11401>
- [9] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint*. <https://doi.org/10.48550/arXiv.2312.10997>
- [10] Jeong, M., Kim, T., Kim, S. & Kim, H. (2024). Retrieval-augmented generation-based question answering technology for construction safety. In *International Conference on Construction Engineering and Project Management*. Korea Institute of Construction Engineering and Management. 439–446. <https://doi.org/10.6106/ICCEPM.2024.0439>
- [11] Arslan, M., Munawar, S., & Riaz, Z. (2024). Sustainable urban water decisions using generative artificial intelligence. In *International Conference on Decision Aid Sciences and Applications (DASA)*. 1–5. <https://doi.org/10.48550/arXiv.2312.10997>

org/10.1109/DASA63652.2024.10836374

- [12] Liu, F., Kang, Z., & Han, X. (2024). Optimizing RAG techniques for automotive industry PDF chatbots: A case study with locally deployed Ollama models. In *Proceedings of the 3rd International Conference on Artificial Intelligence and Intelligent Information Processing*. 152–159. <https://doi.org/10.48550/arXiv.2408.05933>
- [13] Quidwai, M. & Lagana, A. (2024). A rag chatbot for precision medicine of multiple myeloma. *MedRxiv*, 2024-03. <https://doi.org/10.1101/2024.03.14.24304293>
- [14] Zheng, L., Chiang, W., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., Zhang, H., Gonzalez, J., & Stoica, I. (2023). Judging LLM-as-a-judge with MT-bench and chatbot arena. *Advances in neural information processing systems (NeurIPS)*, 36, 46595–46623. <https://doi.org/10.48550/arXiv.2306.05685>
- [15] Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C., Gonzalez, J., Zhang, H. & Stoica, I., (2023). Efficient memory management for large language model serving with pagedattention. *Proceedings of the 29th Symposium on Operating Systems Principles*, 611–626. <https://doi.org/10.48550/arXiv.2309.06180>



김영수

· 1994년 공군사관학교 전산과학과(이학사)
· 2001년 서강대학교 컴퓨터공학과(공학석사)
· 2009년 한국과학기술원 전산학과(공학박사)
· 2021년~현재 전주대학교 인공지능학과 조교수

✚ 관심분야 : 인공지능, 지능형 시스템, 임베디드 시스템

✉ ineland@jj.ac.kr