DOI: https://doi.org/10.13161/kibim.2024.14.3.022

LLM과 RAG 기반 BIM 지식 전문가 에이전트 연구

BIM Knowledge Expert Agent Research Based on LLM and RAG

강태욱[1], 박승화[2]
Kang, Tae-Wook[1] · Park, Seung-Hwa[2]

Received August 22, 2024; Received September 7, 2024 / Accepted September 12, 2024

ABSTRACT: Recently, LLM (Large Language Model), a rapidly developing generative AI technology, is receiving much attention in the smart construction field. This study proposes a methodology for implementing an knowledge expert system by linking BIM (Building Information Modeling), which supports data hub functions in the smart construction domain with LLM. In order to effectively utilize LLM in a BIM expert system, excessive model learning costs, BIM big data processing, and hallucination problems must be solved. This study proposes an LLM-based BIM expert system architecture that considers these problems. This study focuses on the RAG (Retrieval-Augmented Generation) document generation method and search algorithm for effective BIM data retrieval, with the goal of implementing an LLM-based BIM expert system within a small GPU resource. For performance comparison and analysis, a prototype of the designed system is developed, and implications to be considered when developing an LLM-based BIM expert system are derived.

KEYWORDS: BIM, IFC, AI, LLM, Knowledge, Expert, RAG, Process, Retriever

키워드: BIM, IFC, AI, 대형언어모델, 지식, 전문가, 검색증강생성, 프로세스, 검색자

1. 서 론

1.1 연구의 배경 및 목적

스마트 건설 기술의 발전과 함께, BIM은 건설 프로젝트의 계획, 설계, 시공, 유지 보수에 필수적인 역할을 하고 있다. BIM은 스마트 건설에서 데이터 허브 역할을 기대받고 있으나, 아직 BIM 데이터를 효과적으로 검색하고 사용자에게 필요한 정보를 생성하는 방법은 부족하다. 이를 개선하기 위해, IFC (Industry Foundation Classes) 표준이 개발되었다. IFC는 일종의 정보 모델을 표현하는 언어 모델로써 객체지향적인 그래프 형식 스키마로 정의된다. 다만, IFC는 매우 일반적이고 방대한 그래프 구조로 인해, 필요한 정보에 비해 과도한 컴퓨팅 자원이 필요하다.

기존에는 컨텐츠에 대한 정보 검색 시스템을 개발하기 위해서는 컴퓨터가 이해 가능한 스키마 형식의 정보 모델을 이용해 데이터베이스로 저장하는 과정이 일반적인 방식이었다. 데이터베이스 종류는 시스템 개발 전에 모델 스키마 형식 및 응용 목적에 따라 객체지향형, 관계지향성, NoSQL과 같이 결정되어야 한다.

IFC 기반 BIM 전문가 시스템의 경우에도 동일한 프로세스로 개발된다. IFC를 검색하기 위해 BIM 서버를 구현하고, IFC 데이터를 해석하는 별도의 파서(parser)를 이용해, 데이터베이스화하는 과정이 수행된다. BIM을 데이터베이스로 변환한 후 사용자는 쿼리(query) 기능을 이용해 원하는 정보를 검색할 수 있다. 일반적으로 이런 절차를 실행하기 위해서는 많은 개발 비용이 소모된다.

LLM은 정보 검색 및 생성 방식이 이와는 다르게 처리된다. 딥러닝 모델 구조 중 하나인 트랜스포머(Transformers) 구조를 이용해 대량의 데이터를 학습한 후, 이 LLM 모델을 통해 질의에 대한 답변을 자동 생성한다. 2023년 출시된 ChatGPT는 각 산업 분야에서 LLM의 잠재력을 확인하는 계기가 된다. 이후, 불과 2년도 안 된 시점에서 LLM 기술은 많은 발전을 하였고, PDF파일을 이용해 전문가 GPT 앱 서비스를 손쉽게 개발하거나, 인터넷 연결이 없는 상태에서 실행될 수 있는 수준까지 발전했다. 하지만, LLM을 BIM과 같은 도메인 전문분야에서 적용하기 위해서는 과도한 LLM 모델 학습 비용, BIM 빅데이터 처리 방법, LLM 환각 문제 등을 해결해야 한다.

본 연구는 이런 점을 고려해 LLM과 RAG기술을 적용하여 BIM 데이터의 정보를 검색하고, 이를 바탕으로 BIM 지식 전문가 에이전트를 개발하는 효과적인 방법을 제안한다. 이런 에이

[1]정회원, 한국건설기술연구원 연구위원, 공학박사 (laputa99999@gmail.com) (교신저자)
[2]정회원, 한국건설기술연구원 수석연구원, 공학박사 (spark@kict.re.kr)

전트는 자연어 기반의 사용자 쿼리를 통해 필요한 정보를 신속하고 정확하게 제공하며, 건설 프로젝트의 전반적인 효율성을 향상시킬 수 있다. 또한, 본 연구는 실제 스마트 건설 프로젝트에 제안된 에이전트를 적용함으로써 그 실용성을 평가하고, BIM 데이터의 활용 범위를 확대할 수 있는 가능성을 탐구한다.

본 연구는 LLM 기반의 접근 방식을 통해 BIM 데이터를 보다 효과적으로 활용할 수 있는 방법을 제시함으로써, LLM 활용을 위한 BIM 데이터 증강 처리 기술 연구에 기여하고자 한다.

1.2 연구의 범위 및 방법

본 연구는 LLM을 활용하여 BIM 데이터를 탐색하고 활용할 수 있는 BIM 지식 전문가 에이전트를 효과적으로 개발하는 방법론을 제안한다.

첫째, 기존 연구를 분석하여 에이전트의 요구사항을 정의하고, LLM 기반 기술의 가능성을 탐색한다. 둘째, LLM 기반 BIM 전문가 에이전트 구현을 위한 프로세스를 정의한다. 셋째, 다양한 BIM 데이터를 수집하여 LLM 학습에 적합한 형태로 전처리하는 방법을 디자인한다. 넷째, RAG 모델을 설계하여 자연어 쿼리를 통해 BIM 데이터에서 정보를 추출하는 시스템을 구축한다. 다섯째, 프로토타입 시스템을 개발하여 실제 BIM 프로젝트에 적용, 성능을 평가한다. 여섯째, 프로토타입에 구현된 알고리즘 간 성능 비교를 통해 에이전트의 활용 가능성을 검증한다.

마지막으로, 연구 결과를 종합하여 장점과 개선 사항을 논의하고, 스마트 건설 기술 발전에 기여할 연구 방향을 제시한다.

Figure 1은 본 연구 흐름을 보여준다.

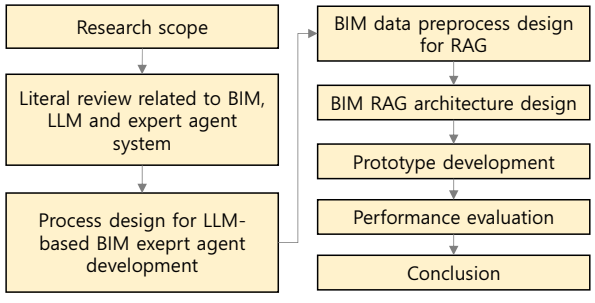

Figure 1. Research process

본 연구는 적은 GPU 리소스 내에서 구현이 가능한 LLM 기반 BIM 지식 전문가 에이전트의 효과적인 구현 방법을 조사하기 위해, LLM 기반 BIM 전문가 에이전트 구현 방법에 대한 프로세스 정의, BIM 데이터의 효과적인 검색을 위한 RAG 문서 생성 방법, 검색 알고리즘 연구에 초점을 맞춘다.

1.3 문헌조사

최근 BIM은 건설 산업에서 정보 관리와 협업의 핵심 도구로 자리 잡고 있으며, 이를 효과적으로 활용하기 위한 다양한 연구가 활발히 진행되고 있다. 특히, BIM 데이터의 상호운용성과 표준화는 다양한 소프트웨어 간의 호환성을 보장하기 위해 중요하다. IFC는 이기종 시스템간 BIM 데이터를 교환하고, 필요한 정보를 검색할 수 있도록 객체지향적인 그래프 스키마 형식으로 표준화된 파일 포맷이다. 다만, IFC는 범용적이고 복잡해 다루기 쉽지 않고, 검색 시스템 구현에 많은 리소스가 필요하다. IFC에서 필요한 정보를 생성하는 전문가 에이전트 구현 비용이 크므로, 현재까지 연구된 시스템은 대부분 특정 분야에 국한된 범위에서 실행된다.

본 연구와 관련된 논문 및 기술을 다음 키워드를 이용해 문헌조사를 하였다.

"BIM", "지식", "전문가", "LLM", "RAG", "프로세스"

BIM 정보 검색을 위한 서버 아키텍처를 연구한 사례가 있다(Kang and Hong, 2014). 이 연구는 IFC 파일을 데이터베이스화하여, 오픈소스를 이용해 데이터를 검색하는 방법을 제안했다.

1.1.1 온톨로지 기반 지속모델 연구

온톨로지 모델 서비스 프레임웍을 연구한 사례가 있다(Kang, 2022). 이 연구는 IFC를 온톨로지 모델로 변환하여, 지식 검색을 용이하게 하는 방법을 제안한다. 설계지식을 검색, 활용하기 위한 프레임웍 연구가 있었다(Woo et al., 2016). 이 연구는 BIM 데이터를 활용해 설계지식을 검색하기 위해 BIM템플릿과 애드인 기술을 활용하였다.

1.1.2 BIM 정보검색기술 기반 연구

BIM 설계 공간용도에 대한 정보를 검색하는 전문가 시스템을 연구한 사례가 있다(Kwon and Cho, 2019). 이 연구는 공간 배치의 품질을 측정하는 전문적 지식을 BIM 데이터로부터 얻는 방법을 제시한다. 이를 위해, 의사결정트리를 사용하여, 전문가 규칙을 추출하는 방식으로 공간 배치의 품질을 계산하여, 지식 검색에 반영한다. BIM 설계품질을 자동검토하는 방법을 제시한 연구가 었었다(Lee et al., 2015). 이 연구는 건축인허가를 위한 설계품질 검토오류를 줄이기 위해, 품질검토 지식을 논리규칙으로 구축하는 방법을 제안한다.

1.1.3 범용 LLM 모델 적용 가능성 연구

GPT 기반 모델을 이용해 안전 관리 지식에 대한 정보를 검

색, 생성하는 방법을 연구한 사례가 있다(Uhm et al., 2024). 이 연구는 건설 안전 지식 제공의 목적으로 관련 텍스트를 수집 및 처리한 후, ChatGPT 3.5, 4.0, RAG 기법을 이용해 지식을 생성하는 성능을 검토하였다.

BIM 지식 기반 건물 분류 방법에 대한 조사연구가 있었다(Woo et al., 2024). 이 연구는 BIM이 건물 스케일 디자인 분류에 도움을 줄 수 있는 방안과 프레임웍을 제안한다.

전통 건축물 리노베이션 프로세스를 가속하기 위해 GPT를 이용한 추론 기능 연구가 있었다(Zhang et al., 2024). 이 연구는 전통 건축물 보존 및 복원과 관련된 지식 답변 만족도를 측정한다.

동적 프롬프트 기반 BIM 정보 검색 가상 비서 기술에 관한 연구가 있었다(Zheng and Fischer, 2023). 이 연구는 BIM 데이터를 MongoDB로 데이터베이스에 저장한 후, ChatGPT를 사용해 사용자 질문에 대한 답변을 생성할 수 있는 코드를 얻고, 이를 실행함으로써 BIM 데이터베이스에 쿼리할 수 있는 프로세스를 제안했다.

앞서 조사한 바와 같이 대부분의 BIM 지식 에이전트 연구 유형은 A) 온톨로지 모델 구축, B) 검색 로직 구현, C) ChatGPT 등 범용LLM 기능 및 생성 코드를 이용한 검색 등이 대부분이다. 기타, LLM 기술 리뷰 및 검토 연구가 있었다. A), B) 유형은 지식 검색 에이전트 구현 시 대상 유스케이스가 결정되어 유연하지 못한 문제가 있다. C)는 GPT같은 범용 LLM 모델의 활용 가능성을 시험하거나, LLM의 코드 자동 생성 능력을 이용한다. 다만, 생성 코드는 LLM이 사전 학습한 IfcOpenShell 같은 별도 라이브러리와 소스코드를 사용하므로 LLM 성능은 해당 라이브러리 기능에 제한된다. 예를 들어, 복잡한 다중 질의는 올바른 코드를 생성하지 못할 수 있다.

본 연구는 유연한 BIM 기반 전문가 에이전트 잠재력을 확인하기 위해 LLM RAG 기반 에이전트 구현 프로세스를 탐색한다. 또한, 제한된 GPU 리소스 내에서 구현이 가능한 LLM 기반 BIM 전문가 에이전트의 효과적인 구현 방법과 프로세스 정의, BIM 데이터의 효과적인 검색을 위한 RAG 문서 생성 방법, 검색 알고리즘 연구에 초점을 맞춘다.

2. LLM 기반 BIM 전문가 에이전트 구현 프로세스

2.1 LLM 기반 전문가 에이전트 개발 방향

현재 대중적인 목적으로 개발된 LLM 기술인 ChatGPT(Open AI), Gemini(Google), Llama(Meta), Phi(Microsoft)는 BIM의 일반적인 지식, 예를 들어, BIM 관련 웹사이트에서 공개된 일반적인 개념 설명, PDF에 포함된 텍스트를 학습한 모델을 제공하고 있다. 이런 이유로, 일반적인 BIM 질문에 대한 답변은 잘 생성해 준다. 다만, 이들 LLM 도구는 BIM 모델링 정보를 담고 있는 IFC와 같은 특수한 데이터셋 파일들은 인식하지 않는다.

현재는 PDF같은 일반적인 파일 형식만 검색증강생성을 지원하는 RAG기술을 이용해, 도메인에 특화된 지식 생성을 지원한다. 이는 특정 도메인 지식을 훈련하기 위해 필요한 비용이 너무 과대하며, 도메인 지식을 모델 학습에 맞게 데이터베이스화하는 방법도 쉽지 않기 때문이다. 예를 들어, ChatGPT-4 모델을 훈련할 때 필요한 GPU 수는 NVIDIA A100 x 25,000개로 알려져 있으며, 학습에 100일이 걸렸다. A100 가격이 수천만원 수준인 것을 감안하면, 사용된 GPU 비용만 천문학적인 금액이 소모된 것을 알 수 있다.

이런 이유로, LLM 모델을 전체 학습하지 않고, 모델 중 작은 일부 가중치만 갱신하는 파인튜닝(fine-tuning), 범용 LLM는 운영체제처럼 사용하여 정보 생성에 필요한 내용을 미리 검색한 후 컨텍스트 프롬프트 정보로써 LLM에 입력해 정보를 생성하는 검색증강생성 기술인 RAG가 주목을 받고 있다.

RAG는 Figure 2와 같은 순서로 사용자 질문에 대한 답변을 생성한다. RAG는 LLM에 입력하는 템플릿에 답변과 관련된 참고 컨텐츠를 프롬프트에 추가하여 원하는 답을 생성하는 기술이다. 이런 이유로, 답변에 포함된 컨텐츠를 처리하고, 검색하는 것이 매우 중요하다. 만약, LLM 입력 프롬프트에 참고할 컨텐츠를 추가하지 못하면, 환각 문제가 발생되는 단점이 있다.

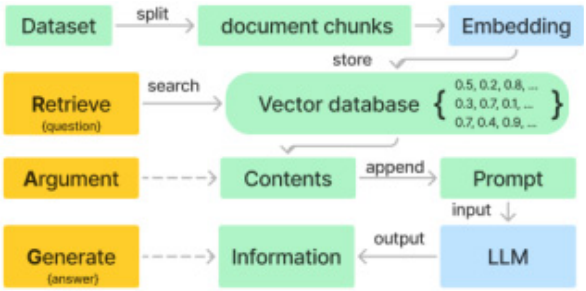

Figure 2. BIM RAG process based on LLM

각 RAG 단계는 검색이 가능하도록 데이터셋을 청크(chunk) 단위로 분할(split)하고, 데이터는 임베딩(embedding)을 통해 검색 연산이 가능한 벡터형식으로 변환된다. 이 벡터들은 저장 및 검색기능을 가진 벡터 데이터베이스(vector database)에 저장된다. 사용자의 질문은 검색 알고리즘을 통해 벡터 데이터베이스에서 가장 근사한 정보를 포함하는 컨텐츠를 얻고, 프롬프트에 추가된 후 LLM에 입력된다. 그 결과, LLM은 원하는 답변을 출력한다. 이를 통해, 학습하지 않은 전문분야의 토큰을 인식하지 못하는 LLM이 원하는 결과를 생성할 수 있도록 한다.

RAG는 LLM 모델 전체 학습이나 파인튜닝에 비해 매우 저렴하고 빠른 성능으로 특정 분야 수많은 데이터셋을 사전 학습된

LLM 모델처럼 활용할 수 있는 장점이 있다. RAG 또한 LLM 성능, 학습된 데이터셋과 토큰 수, 임베딩 모델 성능 등에 큰 영향을 받지만, 해당 기술은 큰 GPU 비용을 감당할 수 있는 몇몇 연구소에서 적용할 수 있다. 이런 이유로, 본 연구는 LLM 모델 학습과 관련된 기법은 연구 범위에 포함하지 않으며, BIM 데이터셋 특징을 고려한 LLM RAG 기반 에이전트 개발 프로세스를 제시한다.

2.2 BIM 데이터셋 특징

앞서 살펴본 바와 같이 RAG 성능은 입력되는 데이터셋의 특징과 검색 알고리즘에 큰 영향을 받는다. 그러므로, 개방형 BIM 데이터 형식으로 사용되는 IFC의 특징을 분석하여, BIM RAG를 위한 데이터 처리 시 이를 고려한다.

IFC 파일 구조는 STEP (ISO 10303), XML 스키마 형식을 준용한다. IFC는 객체지향 모델링과 그래프 모델 구조의 영향을 많이 받았다. 확장성을 고려해, BIM을 구성하고 있는 건축 객체의 부재들, 관계, 속성집합에 Instance ID 및 GUID (Globally Unique IDentifier)와 같은 해쉬값(hash)(Figure 3)을 할당하고, 이들 간의 관계를 해쉬번호로 참조하여, 거대한 온톨로지 그래프 구조를 정의한다(Figure 4).

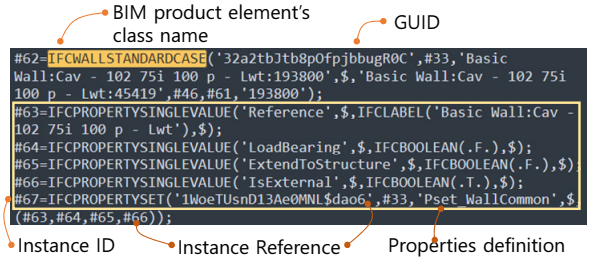

Figure 3. IFC file structure (Wall instance example)

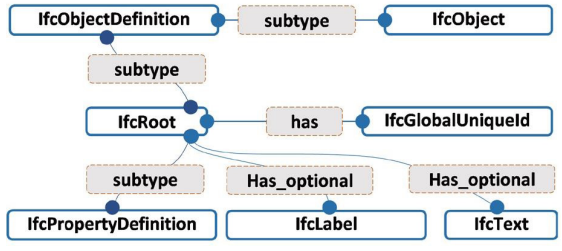

Figure 4. IFC graph structure model (Kang, 2022)

이와 같이 IFC는 일반적으로 특정 단어 토큰(token) 근처에 정보가 밀집된 PDF와 같은 텍스트 형식과는 다른 구조를 가진다. 이는 데이터셋에서 토큰과 관련된 정보를 수집하기 위해 검색해야 하는 비용에 영향을 줄 수 있다는 것을 의미한다.

Table 1은 특정 토큰과 관련된 정보 검색 비용 관점에서 파일 구조에 따른 차이점을 보여준다. 여기서 DPCR (Data Processing Cost for Retrieval)은 RAG에서 특정 주제 관련 정보 검색을 위한 데이터 처리 비용이다.

Table 1. Data Processing Cost for Retrieval (DPCR) between specific token in relation to each other tokens based on file type (DPCR=high,middle,low)

Data file type	Structure feature description	DPCR
1. Text (e.g. TXT, PDF, Table)	Text files have a chapter structure, and related information about topics is likely to be organized around paragraphs or tables. In this case, related information is grouped around tokens of a specific topic, so the cost of information retrieval is low.	Low
2. Scala (e.g. Time-series sensor values)	Time series data sets such as sensor data may show repetitive patterns over time units. In this case, the measurement values in the data pattern affect the pattern type together with the surrounding measurement values. In order to search for patterns, the measurement values must be normalized, and the window size for calculating the pattern and the model for learning the pattern must be determined.	Middle
3. Image (e.g. GIF, JPG)	The definition and meaning of image information vary depending on the purpose, such as classification and object recognition. Images generally have a two-dimensional structure. The values that make up the image are RGB values that express colors, and the relationship between nearby RGB values and changes can be expressed in the form of a convolution pattern. The method of obtaining information from images requires bidirectional learning between text and images. This was developed through CLIP (Contrastive Language-Image Pretraining. Radford et al., 2021) technology.	High
4. Graph (e.g. IFC, STEP, OWL)	Files in the ontology graph format generally require following reference information through links to retrieve tokens related to a specific topic. The graph structure is very flexible and focuses on data extensibility. This structure is significantly different from the SQL-compatible database structure that allows easy querying of information. Since the graph model is likely to have no related information near the tokens of a specific topic, a graph database that supports special indexing and a search query language such as SPARQL are required for information retrieval.	High

BIM 데이터는 Table 1에서 유형 4에 해당하며, RAG를 위한 데이터 처리 비용이 높다. BIM 데이터는 연결을 통해 정보가

검색되므로, 일반적인 RAG 알고리즘으로는 특정 주제와 관련된 정보를 얻기가 어려울 수 있다는 것을 예상할 수 있다. 예를 들어, Figure 3의 IFC구조에서 보여지는 벽체와 관련된 속성값은 해당 객체 인스턴스와 바로 가까이에 있을 수도 있지만, 매우 먼 부분에 존재할 수 있다. 주제에 관련된 모든 정보를 얻기 위해서는 해쉬값으로 표현된 참조를 방문해 관련 데이터를 수집해야 한다. 이런 특성은 PDF와 같은 데이터를 RAG에 사용하는 LLM의 성능에 나쁜 영향을 미칠 수 있다.

2.3 BIM 데이터셋 처리를 위한 RAG 프로세스 정의

BIM과 같은 그래프 구조에서 RAG를 사용할 경우 크게 두 가지를 설계에 고려해야 한다.

A) BIM 그래프 형식 데이터 처리
B) LLM 입력에 필요한 BIM 컨텐츠 검색 방법

A)의 경우, BIM 데이터는 주제를 중심으로 청크 단위로 처리할 방법을 구현해야 한다. 데이터를 검색 가능한 청크 단위로 처리하는 것을 청킹(chunking)이라 한다. 일반적인 청킹은 입력 데이터가 텍스트 구조라 가정하고 처리되므로, 그래프 구조에서는 주제 관련 토큰이 포함되지 않을 가능성이 크다. 그러므로, 주제 중심으로 BIM데이터를 전처리하는 과정이 필수적이다. 이 연구에서는 이 단계를 BIM 청킹이라 한다. 그러므로, BIM 청킹은 LLM에 요청하는 주제와 목적 중심에 따라 데이터가 수집하도록 처리되도록 한다. 식 1은 BIM 청킹의 주제와 연관된 데이터집합을 청킹해야 함을 의미한다.

$$
\begin{aligned}
&BIM_{chunk} = \{x \mid E_{set}, R_{set}, P_{set} \in T \mid x \in T\} \\
&T = \{topic\ *\} \\
&E_{set} = \{element\ *\} \\
&R_{set} = \{relationship\ *\} \\
&P_{set} = \{property\ *\}
\end{aligned} \tag{1}
$$

B)는 청크에서 질문과 관련된 데이터를 검색하는 알고리즘과 관계된다. 각 토큰을 임베딩 모델에 입력해 얻은 벡터값은 수학적 연산에 사용될 수 있다. 임베딩 모델은 토큰 간에 의미 유사 정도를 학습 시 고려했다. 이를 이용하면, 토큰 임베딩 벡터 간 거리를 계산할 수 있어, 이를 통해 관련성있는 데이터를 임베딩 벡터 공간에서 검색할 수 있다.

RAG 검색에 사용되는 코사인(cosine) 유사도는 벡터의 내적 공식을 이용해 벡터 간 거리를 계산한다(2). 그러므로, 식 2로 계산된 값이 크면 거리가 가깝고 비교 토큰 간 의미가 유사한 것이다.

$$
\text{similarity}_{\cos} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} \tag{2}
$$

코사인 유사도는 다수 문서들에 중요 정보가 분산되어 있고, 한 문서에 비슷한 토큰이 많이 반복된 경우, 전체 문서에서 원하는 정보를 검색하기 어려울 수 있다.

$$
\begin{aligned}
&MMR = \lambda \cdot \text{sim}(\text{d}, \text{Q}) - (1-\lambda) \cdot max_{d' \in D'}\, sim(d, d') \\
&d = document \\
&Q = query\ token \\
&sim = similarity_{cos} \\
&D = selected\ document\ set \\
&max = solution\ maximun\ function \\
&\lambda = Similarity\ and\ Distinction\ Adjustment\ Parameters
\end{aligned} \tag{3}
$$

MMR (Maximum Marginal Relavance)는 각 문서의 유사성 점수와 이미 선택된 문서들 간의 차별성 점수를 조합해 검색에 필요한 최종 점수를 계산한다(식 3).

BIM RAG에서 검색은 데이터 특성을 고려해 유사성과 차별성 검색을 수행한다.

Figure 5는 앞서 기술한 BIM 청킹과 검색 방식을 고려한 BIM RAG 처리 파이프라인을 보여준다.

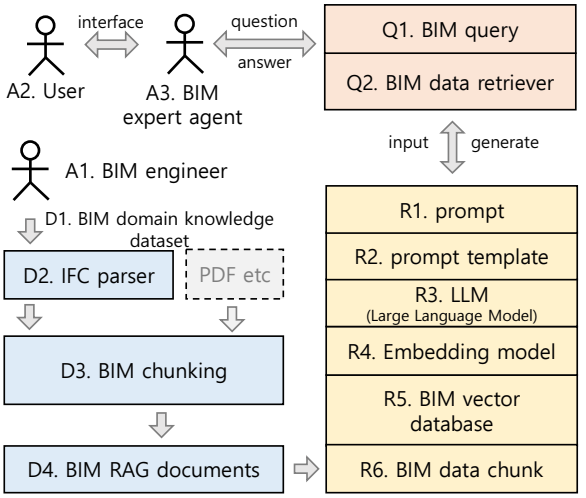

Figure 5. LLM based BIM RAG process pipeline

파이프라인을 구성하는 컴포넌트는 에이전트를 활용하는 액터(actor), LLM을 위한 BIM 데이터 처리, BIM RAG 처리, 정보 질의 및 검색에 따라 구분하였으며, 각 역할은 Table 2와 같이 정의한다.

Table 2. LLM based BIM RAG components description

Name	Role
A1. BIM engineer	Engineer developing BIM datasets. e.g. BIM models, IFC file development, COBie maintenance dataset development, 4D/5D simulation results, design and construction guideline PDF files, etc.
A2. User	User who calls and utilizes a BIM expert agent.
A3. BIM expert agent	Agent uses BIM RAG pipeline components to generate information that users need.
D1. BIM domain knowledge dataset	Domain knowledge dataset required for the BIM expert agent. The dataset is a raw dataset before being input into BIM RAG, and is in the form of IFC, PDF, and TEXT files, for example.
D2. IFC parser	Its role is to analyze IFC files and extract data.
D3. BIM chunking	Convert BIM data into a format that can be processed by RAG so that the BIM agent can perform well.
D4. BIM RAG documents	Converted BIM RAG documents. The documents contain multiple chunks.
R1. Prompt	Prompt generated by a prompt template that includes user questions and augmented search content.
R2. Prompt template	Prompt template suitable for the purpose of BIM agent service. Since this study uses RAG, the template consists of instruct, question, and content sections.
R3. LLM	LLM is used as a foundation model. LLM should be selected according to the agent purpose. LLM supports open source models such as Llama, Gemma, and Phi, and commercial models such as ChatGPT. Note that each LLM has differences in document summarization, code generation, and knowledge type support.
R4. Embedding model	Convert chunks into searchable embedding vectors from a vector database.
R5. BIM RAG vector database	Vector database that stores and manages values converted from sequence tokens included in chunks into vectors.
R6. BIM data chunk	Data set divided into a format convenient for RAG processing of BIM data.
Q1. BIM query	BIM knowledge-related question, expressed in natural language.
Q2. BIM data retriever	Provides the ability to search BIM knowledge in a vector database. There are various search methods, such as the similarity calculation mentioned above. The searcher should be selected considering the agent and data type.

2.4 BIM RAG 생성 정보의 품질 측정 방법 정의

질문에 대해 LLM에서 생성된 정보는 다양할 수 있다. 이런 이유로, 토큰의 일치 여부만으로 결과물의 품질을 판단하기 어렵다. 이런 이유로, BIM RAG 결과물에 대한 적절한 품질 측정 방법이 필요하다.

이 연구에서는 기계번역 분야에서 모델을 평가하는 데 사용하던 BLEU (Bilingual Evaluation Understudy)와 ROUGE (Recall-Oriented Understudy for Gisting Evaluation)를 사용한다.

BLEU (식 4)는 번역된 텍스트의 품질을 평가하는 데 사용되었다. 이 지표는 텍스트에서 n개의 단어로 구성된 연속된 시퀀스인 n-gram을 구성한 후, 참값을 가진 참조 번역 텍스트와 번역된 텍스트 간의 유사도를 계산한다. 예를 들어, n-gram에서 n이 하나일 경우에는 BLEU는 단일 단어만을 서로 비교하게 된다. BLEU는 참조 번역보다 짧게 번역된 텍스트는 패널티를 부여한다.

$$
\begin{aligned}
&BLEU = BP \cdot exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \\
&BP = \begin{cases} 1 & if\ c > r \\ e^{(1-\frac{r}{c})} & if\ c \le r \end{cases} \\
&N = n\ gram\ number \\
&w_n = 1/N \\
&r = length\ of\ reference\ text \\
&c = total\ length\ of\ translation\ text \\
&p_n = geometric\ average\ of\ n\ gram\ precisions
\end{aligned}
\tag{4}
$$

ROUGE 지표는 생성된 텍스트 요약과 참값을 가진 참조 요약 간의 품질을 평가할 수 있다. ROUGE는 단어 시퀀스에서 겹치는 n-gram을 사용하여 상호 중첩된 텍스트 간 유사성을 고려한다. 이를 위해 n-gram 재현율(recall)을 계산한다. ROUGE는 n-gram 중첩도를 얻기 위해 참조 요약과 생성된 텍스트 시퀀스 S간의 전체 개수에 대한 일치 여부를 계산한다(식 5).

$$
ROUGE - N = \frac{\sum_{S \in Reference\ Texts} \sum_{n\ gram \in S} Match(n\ gram)}{\sum_{S \in Reference\ Texts} \sum_{n\ gram \in S} Count(n\ gram)} \tag{5}
$$

BLUE와 ROUGE 지표는 참조와 생성 데이터에 대한 부합성 계산을 할 수 있는 모델이지만, 길고 다양한 시퀀스를 가진 데이터들에 대한 의미론적 일관성을 측정하기 어려울 수 있다. 이런 이유로, 많은 고품질 데이터셋으로 훈련된 LLM을 통해 답변의 정확도 지표를 계산하는 방법을 사용한다. 본 연구에서는 이 모델을 의미론적 유사도(semantic similarity) 평가지표로 사용한다.

3. 프로토타입 구현 및 성능 분석

3.1 프로토타입 구현

제안한 방법의 성능을 분석하기 위해, 앞서 정의된 BIM RAG 파이프라인(Figure 5)을 포함한, RAG 데이터처리, 검색 및 증강 생성 모듈을 구현하고 이를 테스트해 보도록 한다. 이를 위해, IFC에서 정보를 검색할 수 있는 BIM 전문가 에이전트 프로토타입을 개발하였다.

제안한 BIM RAG 청킹 구현에 필요한 데이터 처리는 파이썬 코드로 개발되었다. 파이프라인의 컴포넌트 입출력 로직은 파이썬으로 개발되었다. 각 파이프라인의 컴포넌트를 상호 호출해주는 기능은 LangChain을 사용하였다. Table 3은 이를 기술한 것이며, 프로토타입 개발 결과는 Figure 6과 같다.

Table 3. LLM based BIM RAG component implementation

Component	Implementation
D1. BIM domain knowledge dataset	Develop IFC files using modelers such as Revit.
D2. IFC parser	Extract data by parsing IFC using IfcOpenShell.
D3. BIM chunking	Parsed BIM data is saved in JSON format and then chunked in a searchable manner.
R2. Prompt template	A template is defined in the following format: You are an assistant for question-answering tasks. Use the following pieces of retrieved context to answer the question. Please answer the question concisely. If you don't know the answer, just say that you don't know. #Question: {input} #Context: {context}
R3. LLM	The LLM used in this study is LLama3, an open source model. ollama and LangChain were used.
R4. Embedding model	The embedding model used FastEmbed. This model is a fast and lightweight embedding vector generation model developed by Qdrant.
R5. BIM RAG vector database	The vector database uses chroma. Chroma provides convenient vector storage, management, and search functions.
Q2. BIM data retriever	The BIM chunk search model applies Equations 2 and 3. Equations 2, 4, and 5 are used to analyze the performance of each search model.

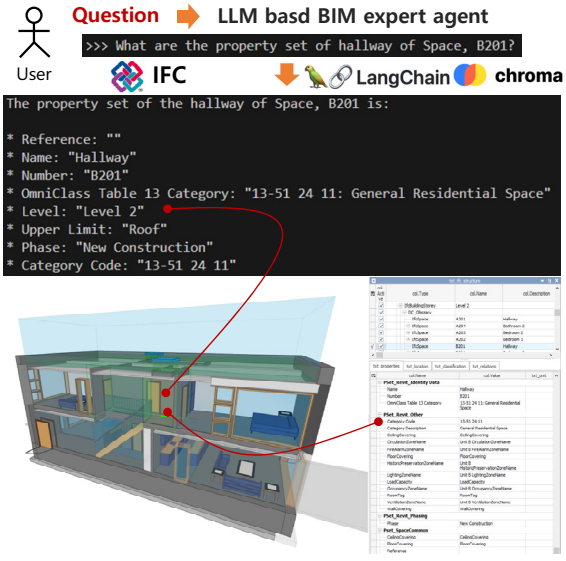

Figure 6. LLM based BIM expert agent prototype

3.2 성능 분석

성능 분석을 위한 방법은 다음과 같다.

1) 입력 IFC를 청킹해 벡터 데이터베이스에 저장.
2) 평가할 질문을 준비(Table 4).
3) 이에 대한 정답인 참조 답변을 준비.
4) 유사도(R1), 허용치 유사도(R2), MMR (R3) 검색함수 준비.
5) 질문을 전문가 에이전트 프로토타입에 입력.
6) BLEU (E1), ROGUE (E2), 의미 유사도(E3) 평가모델을 준비.
7) 생성된 답변과 참조 답변을 각 모델별로 비교.

Table 4. Evaluation questions and reference answers

Question	Reference answer
Q1. What is the classification description of basic wall:interior - furring (38 mm Stud) of wall?	The classification description of basic wall: interior – furring (38 mm Stud) is metal-framed gypsum board fixed partitions
Q2. How many are living room of space?	Number of living room in space is 2
Q3. What is the circulation zone name of living room which number is A102?	The circulation zone name of living room which number is unit a CirculationZone Name
Q4. What are the property set of hallway of space?	The property set of the hallway in space are 'Name': 'Hallway', 'Number': 'B201', 'OmniClass Table 13 Category': '13-51 24 11: General Residential Space', 'Level': 'Level 2', 'Upper Limit': 'Roof', 'Phase': 'New Construction', 'Category

	Code': '13–51 24 11', 'Category Description': 'General Residential Space', 'CirculationZoneName': 'Unit B CirculationZoneName', 'FireAlarmZoneName': 'Unit B FireAlarmZoneName', 'HistoricPreservationZoneName': 'Unit B HistoricPreservationZoneName', 'LightingZoneName': 'Unit B LightingZoneName', 'OccupancyZoneName': 'Unit B OccupancyZoneName', 'VentilationZoneName': 'Unit B VentilationZoneName'
Q5. What is the Onmiclass code of foyer space?	The OmniClass code of foyer space is 13–51 24 11: general residential space

본 연구에서 사용한 의미 유사도 평가모델은 문장 트랜스포머 모델 중 하나인 허깅페이스(huggingface)의 all–MiniLM–L12–v2이다. 이 모델은 문장 클러스터링 등에 사용된다. 생성된 답변과 참조 답변을 이 모델에 입력하여 cosine 유사도 계산해 품질지표로 사용한다.

테스트를 위해 BIM 전문가 에이전트 실행 시 사용된 GPU 메모리는 5.3GB 였다(NVIDIA GPU 3090, 인텔 i9 CPU, 메모리 16GB 환경). 생성된 답변은 Figure 7과 같이 분석을 위해서 RAG 파라메터 옵션, 질문, 정답, 생성된 답변으로 구분해 JSON 파일로 저장하였다.


```
{
    "time": "20240815 20:13",
    "rag_options": "mmr",
    "question": "How many are Living Room of Space?",
    "pred_answer": "There are 2 Living Room of Space.",
    "true_answer": "Living Room of Space is 2",
```


Figure 7. Generated information from question (example)

각 질물 별로 생성된 답변들을 각 평가지 표별로 계산하여 결과를 Table 5, Figure 8과 같이 정리해 보았다.

Table 5. Test score (average)

RAG retriever	Question ID	Score		
		E1. BLEU	E2. ROUGE	E3. Semantic similarity
R1. Similarity	Q1	0.935	1.000	0.990
	Q2	0.000	0.171	0.703
	Q3	0.547	0.813	0.920
	Q4	0.129	0.901	0.941
	Q5	0.000	0.000	0.064
R2. Similarity_score_threshold	Q1	0.657	0.878	0.974
	Q2	0.000	0.000	0.076
	Q3	0.634	0.929	0.899
	Q4	0.108	0.814	0.946
	Q5	0.000	0.000	0.064
R3. MMR	Q1	0.935	1.000	0.990
	Q2	0.000	0.615	0.804
	Q3	0.459	0.897	0.901
	Q4	0.070	0.628	0.913
	Q5	0.652	0.880	0.959

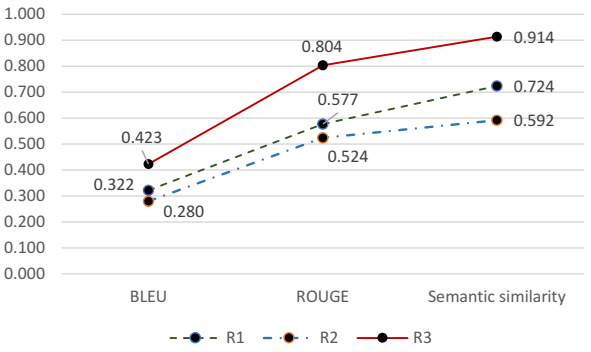

Figure 8. Output quality performance (average) in LLM–based BIM expert agent

Figure 8과 같이, 에이전트의 정보생성 품질은 MMR 검색이 높았고(Semantic similarity=0.914, ROUGE=0.804, BLEU=0.423), 단일 유사도 기반 검색이 제일 낮다는 것을 알 수 있다. MMR은 토큰 유사도 뿐 아니라 문서 간 차별성을 고려하므로 좀 더 정확한 컨텐츠를 검색할 수 있었고, 이를 바탕으로 LLM이 정답에 근사한 정보를 생성할 수 있었던 것으로 판단된다.

단, 각 질문에 대한 LLM 생성 결과들의 문장 구조는 다소 가변적으로 생성된다. 각 질문 중 Q2와 같이 특정 숫자를 요구하는 답변은 제대로 된 검색을 하지 못해 원하는 결과를 얻지 못하는 경우도 있었다. 이는 RAG에서 발생될 수 있는 문제로, BIM 지식 검색기로서 LLM을 이용하기 위해서는 좀 더 연구가 필요하다는 것을 알 수 있다.

3.3 고려사항 도출

성능 분석을 통해 BIM과 같은 데이터구조에서 높은 LLM 생성정보 품질을 기대하려면 다음을 고려할 필요가 있다는 것을 알 수 있다.

첫째, IFC와 같이 그래프 링크 구조로 정보가 문서 내에 펼쳐져 있는 데이터셋은 미리 목적에 맞도록 주제 중심으로 데이터를 가공해 LLM에 증강되는 컨텐츠로 입력될 수 있어야 높은 생성 품질을 기대할 수 있다.

둘째 IFC는 일반적인 텍스트 정보처럼 정보가 주제 중심으로 밀집되어 있지 않으로, MMR과 같은 검색 방식이 좀 더 유리하다.

셋째, 불필요한 데이터가 청크에 포함되지 않도록 노이즈 등은 사전 필터링될 필요가 있다.

넷째, 검색 토큰 주변의 정보를 생성에 참고할 컨텐츠에 포함

되도록 오버랩(overlap), 검색 근처 벡터수 k와 같은 최적 검색 파라메터를 도출해 설정해야 한다.

다섯째, LLM이 질문의 정보를 추론하는 메커니즘을 고려해, 파인튜닝 기법을 사용해 의미론적인 단위로 구분된 토큰 단위 재학습을 고려한다.

이외에, 본 연구 범위에서 다루지 않았던 LLM 모델, 벡터 데이터베이스 유형, 임베딩 벡터 모델 등의 하이퍼파라메터 조합을 통해 BIM RAG 파이프라인을 튜닝할 필요가 있다.

4. 결 론

본 연구는 유연한 BIM 기반 전문가 에이전트 잠재력을 확인하기 위해 LLM RAG 기반 에이전트 구현 프로세스를 탐색하였다. 제한된 GPU 리소스 내에서 구현이 가능한 LLM 기반 BIM 전문가 에이전트의 RAG 파이프라인를 분석하고 정의하였다. 아울러, BIM 데이터 특징을 고려한 효과적인 RAG 검색 문서 처리 방법, 검색 알고리즘 연구를 수행하였다.

제안된 방법은 BIM 정보를 생성한 품질 관점에서 MMR 0.914, 허용치 유사도 검색 0.724, 단일 유사도 검색 0.592 성능을 보여준다.

이 연구를 통해, LLM 기반 정보 생성을 위한 RAG 컨텐츠를 고려해야 할 경우, 입력 데이터 특징에 따른 적절한 전처리와 검색 방법이 필요하다는 것을 확인할 수 있다. IFC파일을 PDF와 같은 텍스트 파일처럼 물리적으로 문서를 분할하여, LLM에 사용한다면, 제대로된 결과를 얻기 어려울 수 있다. BIM과 같은 그레프 모델 형식을 RAG에 활용하기 위해서는 전처리 청킹 과정을 통해, 질문에 관련 정보 검색이 쉽도록 객체 중심으로 모아 둘 필요가 있다.

BIM 분야는 건설 정보 기술 언어를 이용한 모델링 방법을 발전시켜, 상호운용성을 지원함으로서 건설의 정보 재활용성과 생산성을 개선하고자 했다. LLM은 BIM 기술이 사용하는 언어를 충분히 이해할 수 있다면 더욱 발전할 가능성이 있다.

향후, 에이전트의 품질을 개선할 수 있도록 좀 더 다양한 하이퍼파리메터를 검토하여 튜닝할 계획이며, 직접 입력 파일에서 필요한 정보를 추출하는 코드 생성 에이전트와 연계 실행 등을 고려한 BIM 전문가 멀티 에이전트 처리 방법을 연구할 계획이다.

감사의 글

이 연구는 한국건설기술연구원의 “생성AI기반의 건설정보모델 관리 및 인허가 검토 자동화 시스템 개발 (1/1)” 과제의 지원으로 수행되었다.

References

Jeong, J., Gil, D., Kim, D. Jeong, J. (2024). Current Research and Future Directions for Off-Site Construction through LangChain with a Large Language Model. Buildings, 14(8), pp. 23-27.

Kang, T. (2022). Ontology BIM-based Knowledge Service Framework Architecture Development, Journal of KIBIM, 12(4), pp. 52-60.

Kang, T., Hong, C.-H. (2014). GIS-based BIM Object Visualization System Architecture Design using Open source BIM Server Cost-Effectively, Spatial Information Research, 22 (1), pp. 45-33, https://doi.org/10.12672/ksis.2014.22.1.045.

Kwon, O., Cho, J. (2019). Space Usage Knowledge Extraction from BIM Data by Decision Tree and Expert System, Transactions of the Society of CAD/CAM Engineers, 24(2), pp. 126-134, http://dx.doi.org/10.7315/CDE.2019.126.

Lee, H., Park, S., Kim, I., Lee, J. (2015). A Logical Rule-based Approach to the Korea Architecture Code Sentences for BIM-enabled Design Assessment Systems, Journal of Korea Design Knowledge, 34, pp. 101-110, https://doi.org/10.1016/j.jcde.2018.08.002.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G. (2021). Learning Transferable Visual Models from Natural Language Supervision. In International Conference on Machine Learning, Proceedings of Machine Learning Research, pp. 8748-8763.

Uhm, M., Kim, J., Ahn, S., Jeong, H., Kim, H. (2024). Efficacy of Retrieval Augmented Generation-Based Large Language Models for Generating Construction Safety Information, Social Science Research Network, pp. 1-12.

Woo, C.-M., Kim, M.-K., Jun, H.-J. (2016). Design Knowledge Classification based BIM Design System Framework – Representation of Building Scale Estimation Design Knowledge, Architectural Institute of Korea, 32(7), pp. 21-28, http://dx.doi.org/10.5659/JAIK_PD.2016.32.7.21.

Zhang, J., Xiang, R., Kuang, Z., Wang, B. and Li, Y. (2024). ArchGPT: Harnessing Large Language Models for Supporting Renovation and Conservation of Traditional Architectural Heritage. Heritage Science, 12(1), pp. 220-229.

Zheng, J., Fischer, M. (2023). Dynamic Prompt-based Virtual Assistant Framework for BIM Information Search, Automation in Construction, 155, pp. 10-15.