Journal of KIIT. Vol. 23, No. 6, pp. 31-40, Jun. 30, 2025. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2025.23.6.31

자동차 법률 문서 해석에 특화된 데이터 전처리 및 프롬프트 엔지니어링 방법론 연구

안성규*, 문벼리**[1], 박상범***, 김세희**[2], 김경외****

A Study on Data Preprocessing and Prompt Engineering Methodologies Specialized for the Interpretation of Automobile Law Documents

Seonggyu Ahn*, Byeori Moon**[1], Sangbeom Park***, Sehee Kim**[2], and Keungoui Kim****

요 약

자동차 설계 및 개발 과정에서 개발자는 법과 규제를 준수하기 위해 담당 법률 전문가와 지속적으로 소통하고 협력해야 한다. 그러나 기업 내 법률 전문가의 수가 제한적이라는 현실에서는 다수의 개발자가 적시에 법률 자문과 검토를 받기 어렵다. 본 연구는 법률 비전문가인 자동차 개발자가 생성형 AI를 활용하여 법률 및 규제를 검토하고 해석할 수 있도록, 자동차 법률 문서에 특화된 데이터 전처리 및 프롬프트 엔지니어링 방법론을 제안한다. 제안된 방법론은 법률 문서의 구조적 정보를 활용하여 검색 효율성을 높이는 Parent-Child Chunking 기법과 답변 정답률을 향상시키기 위한 프롬프트 엔지니어링 기법을 포함한다. 이러한 접근 방식으로 구축된 RAG 서비스는 자동차 개발자가 법규 관련 업무를 보다 효율적이고 자율적으로 수행하는데 기여할 수 있을 것으로 기대된다.

Abstract

In the process of automobile design and development, developers must maintain continuous communication and collaboration with legal experts to ensure compliance with laws and regulations. However, it is often challenging for most developers to receive timely legal advice and reviews from a limited number of in-house legal experts, which poses a risk of legal issues arising during the design phase. To address this challenge, this study proposes a methodology for data preprocessing and prompt engineering tailored to automotive legal documents, enabling non-expert developers to review and interpret laws and regulations using generative AI. The proposed methodology incorporates the Parent-Child Chunking technique, which enhances search efficiency by leveraging the structural information of legal documents, and a prompt engineering to improve the accuracy of responses. he RAG service built on this approach is expected to enable automotive developers to perform regulatory tasks more efficiently.

Keywords

automobile law, braking system, prompt engineering, RAG, parent-child chunking

* 한동대학교 생명과학부 학사과정
- ORCID: https://orcid.org/0009-0003-1780-3870
** 한동대학교 ICT창업학부 학사과정
- ORCID[1]: https://orcid.org/0009-0007-2312-8778
- ORCID[2]: https://orcid.org/0009-0006-3555-0135
***한동대학교 법학부
- ORCID: https://orcid.org/0009-0001-3134-3757
**** 연세대학교 융합인문사회과학부 조교수(교신저자)
- ORCID: https://orcid.org/0000-0002-0839-8813
· Received: Mar. 25, 2025, Revised: Jun. 05, 2025, Accepted: Jun. 08, 2025
· Corresponding Author: Keungoui Kim
Humnaities, Arts, and Social Sciences Division,
Underwood International College, Yonsei University, Seoul, South Korea
Tel.: 82+32-749-3611, Email: awekim@yonsei.ac.kr

Ⅰ. 서 론

자동차 개발자는 자동차 설계 및 개발 과정에서 법률 담당 전문가와 지속적인 소통과 협력을 통해 다양한 법규와 규제를 고려하여 자동차를 설계해야 한다[1]. 만약 법률 개정에 미리 대응하지 못해 중간 단계에서 설계를 변경하거나, 신차 출시 후 법률 위반 사례가 발견될 경우에 이는 엄청난 비용 손실로 이어질 수 있기 때문에 설계 과정 중 법률 검토는 매우 중요하다. 문제는 대부분의 자동차 개발자가 기업 내 소수의 법률 전문가로부터 적시에 법률 자문과 검토를 받기가 쉽지 않다는 것이다. 자동차 관련 법규는 여러 환경적 또는 사회경제적인 이유로 법률 개정이 빈번히 일어나고, 고려해야 하는 법규의 범위와 대상이 다양하기 때문에 개발자들이 관련 내용을 설계 과정 중에 지속적으로 확인하기 위해서는 상당한 업무 외적인 노력을 기울여야 한다.

본 연구에서는 자동차 설계 및 개발 과정에서 발생하는 법률 검토의 비효율성 문제[2]를 해결하기 위해, 개발자가 직접 법률 및 규제 관련 질문을 던지고 이에 대한 답을 얻을 수 있는 생성형 AI 모델 기반의 챗봇 서비스 구조를 제안하고자 한다. 또한, 적용 법률 규제는 선도국의 법률을 주로 참조한다는 특성을 반영하여, 자동차 선진국들이 모두 참여하는 유일한 글로벌 협의체인 UNECE 법률 문서 데이터를 활용하였다[3]. 이를 통해 본 연구는 자동차 개발자의 법률 자문 접근성을 높여 보다 효율적으로 법률 요구사항을 반영할 수 있는 환경을 제공하고자 한다.

Ⅱ. 관련 연구

생성형 AI 모델의 주요 특성 중 하나는, 보유한 정보의 범위와 양이 사전 학습된 데이터에 한정된다는 점이다. 따라서 학습에 포함되지 않았거나 학습 완료 이후에 생성된 데이터와 관련된 질의에는 제한적인 답변만을 제공할 수밖에 없다[4]. 더욱이 확률적 계산을 기반으로 가장 확률값이 높은 응답을 생성해내는 생성형 AI는 그럴듯해 보이지만 잘못된 정보를 제공하는, 이른바 “환각” 현상을 야기하기도 한다[4]. 이러한 구조적 한계를 극복하기 위해 제시된 검색 증강 생성(RAG, Retrieval-Augmented Generation)은 추가적인 학습 없이 데이터베이스로부터 사용자 질의와 관련된 정보를 검색하고 이를 활용해 강화된 답변을 생성하는 방식으로 고안되었다[5].

RAG 시스템을 구성하기 위해서 우선적으로 고려되어야 하는 부분 중 하나는 바로 추가 데이터 검색에 사용될 원시 데이터의 전처리 작업이다. 이는 단순히 원시 데이터를 검색이 가능한 형태로 변환하는 것만이 아니라 그 안에 포함된 표, 그림, 글의 구조 등에 관한 정보 손실 없이 유지해야 하는 매우 중요한 작업이다. 가령 말뭉치 조합이나 청크 및 오버랩 크기 조정은 모델의 표 데이터 인식률을 높일 수 있으며[6], 이는 질문에 대한 올바른 답변 생성에 필수적인 정확한 정보 수집에 직접적으로 기여할 수 있다.

이에 더하여 데이터 모델링에 특화된 검색기 설계를 통해서도 RAG의 성능을 향상시킬 수 있다. RAG에서 사용자 답변에 근거가 되는 문서들은 검색 단위인 청크로 분할되는데 이는 임베딩 벡터의 형태로 저장된다. 임베딩 벡터로 전환된 사용자의 질의와 증강된 데이터 간의 유사도 측정을 통해 RAG는 사용자 질의와 가장 관련성이 높은 문서 근거를 찾아 답변을 생성하게 된다. 이전 연구에 따르면 법률 도메인의 특수성을 반영하기 위해 정관 - 질문 검색기 학습을 수행한 결과 모델의 법률 도메인 이해가 향상되었으며 특히 질문과 연관성 높은 정관 문서를 잘 검색하여 정확한 답변을 생성해 내었음을 알 수 있다[7].

그러나 설령 아무리 좋은 데이터나 좋은 검색기를 사용했더라도 미처 고려되지 못한 예외적인 질문 또는 상호작용으로 인해 생성형 AI 모델이 사용자의 질문을 정확히 이해하지 못하거나 사용자가 원하는 형식의 답변을 생성하지 못하는 문제가 여전히 발생할 수 있다. 프롬프트 엔지니어링은 이러한 예기치 못한 답변의 불확실성을 통제하기 위한 방안으로 제시되었다. 프롬프트 엔지니어링은 생성형 AI의 답변의 형식과 구조를 일관성 있게 통제할 수 있을 뿐만 아니라, 역할 부여 또는 답변의 구체성을 명시하는 방식으로 답변의 질을 향상시킬 수 있다[8].

이에 본 연구에서는 환각 효과 없이 원시 법률 데이터에 특화된 전처리 방법과 법률 도메인에 맞춘 프롬프트 설계 기반의 RAG시스템을 제안하고자 한다.

II. 데이터 수집 및 전처리

2.1 데이터 수집

본 연구에서는 UNECE에서 제공하는 자동차 관련 법률 문서 중 제동 장치 관련 규제 번호 78, 198, 89 문서를 사용하였다. 한국교통안전공단 리콜 통계에 따르면 제동 장치 관련 결함이 주요 5대 리콜 원인 중 하나로 나타났으며[9] 이는 제동 장치 설계에 대한 면밀한 안전 규제 검토의 필요성을 시사한다. 또한, 제동 장치는 자율주행 등 미래형 차량 기술의 핵심적 요소이기에[10] 관련 규제의 지속적인 추적이 요구되는 분야기도 하다. 따라서 설계 단계에서 규제 검토의 종합적인 중요성을 고려하여 제동 장치를 본 연구의 대상 부품으로 선정하였다. 분석을 위해 수집한 PDF 파일 형식으로 구성된 3개의 문서를 하나의 문서로 합쳐 총 80 페이지 분량의 병합본을 만들었으며 해당 병합본 안에는 텍스트 데이터, 이미지 데이터 Latex 형식의 수식 데이터, 테이블 데이터가 포함되어 있었다.

2.2 데이터 전처리

UNECE 문서를 포함한 대다수의 법률 및 규정 문서에는 문자 텍스트뿐만 아니라 이미지, 수식, 표를 포함한 다양한 형태의 정보가 포함된다. 본 연구에서는 이미지를 제외한 모든 형태의 데이터를 수집하였는데, 이는 대부분 텍스트 내에 이미지 자료에 대한 보충 설명이 포함되어 있어 이를 제외하더라도 정보 손실이 일어나지 않기 때문이다. 이와 다르게 수식과 표는 관련 내용을 텍스트에서 설명하지 않은 경우가 많아 이를 제외할 경우 심각한 정보 손실 문제를 초래할 수 있다. 따라서 조문의 정확한 해석을 위해서는 수식과 표에 내포된 정보를 수집 및 전환하는 것이 필수적이나 일반적인 텍스트 데이터의 형식으로는 이를 온전히 표현하기에는 한계가 있다.

이에 본 연구에서는 다음의 방식으로 조문의 전처리를 진행하였다(그림 1). 수식의 경우 문서 내 수식이 포함된 조문을 먼저 확인하고 해당 수식을 LaTeX 포맷으로 변환하여 조문 JSON 객체에 삽입하였다. LaTeX는 TeX 언어를 기반으로 한 문서 포맷으로 수식 작성이 용이하여 수학 및 공학 등 이공계 학술 작업물에 주로 사용된다[11]. 표 역시 마찬가지로 표가 포함된 조문을 확인한 다음 이를 Markdown 포맷으로 전환하여 조문 JSON 객체에 삽입하였다. Markdown은 병합 셀 및 텍스트 정렬 방식 등 표의 세부 구조를 표현하는 문법 체계를 타 포맷에 비해 보다 간결한 형태로 제공한다.

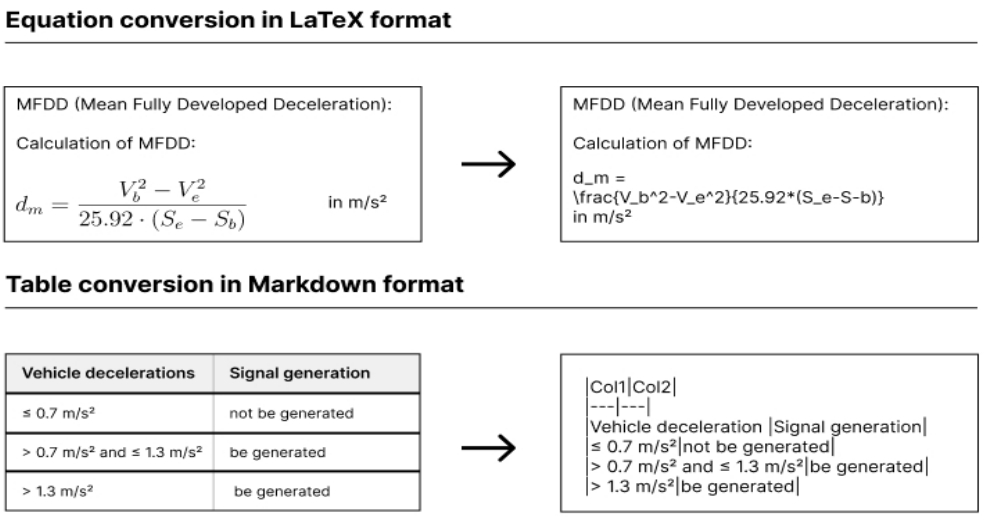

그림 1. 수식, 표 변환 예시
Fig. 1. Examples of equation and table conversion

III. 검색기 개발 방법

3.1 기존 검색기의 문제점

기존의 RAG 방법론에서는 원본 문서에서 추출한 텍스트를 고정 토큰 수로 분할하여 청크를 생성하는 방식을 사용한다[12]. 고정 토큰 수 기반의 청크 생성은 문장의 의미와 구조적 특징을 고려하지 않은 기계적인 분할을 수행하기 때문에 경우에 따라서는 불완전한 문장이 청크에 포함될 수 있다. 더욱이 PDF 문서와 같은 원시 데이터를 이용할 경우 비표준화된 내부 포맷과 인코딩 문제로 인해 텍스트 추출 과정에서 다양한 오류가 발생할 수 있다. 이러한 노이즈는 언어 모델의 성능을 저하시킬 수 있으며, 특히 법률 문서처럼 높은 수준의 정확성이 요구되는 경우 이러한 문제는 더욱 심각해질 수 있다.

UNECE를 비롯한 법률 및 규정 문서는 계층적인 조문 체계로 구성되며, 상위 조문은 하위 조문의 해석에 있어 중요한 의미적 맥락과 전제를 제공한다[13]. 이러한 계층적 구조를 이해하는 것은 법률 문서를 온전히 해석하는 데 필수적이다. 그러나 기존의 고정 토큰 수 기반 청크 생성 방식은 이러한 계층적 구조를 반영하지 못해 조문 해석에 필요한 맥락 정보를 누락할 가능성이 있다. 이에 본 연구에서는 계층적 구조를 보존하고 필요한 정보를 압축적으로 전달하기 위해, 문서 청크 단위 및 데이터 구성 포맷으로 활용할 수 있는 계층적 전처리 방법론을 제안한다.

3.2 계층적 구조를 반영한 chunk 생성

계층적 전처리 방법론을 활용하기 위해 상용 표준 데이터 포맷을 참고하여 구조를 설계하였다. JSON(JavaScript Object Notation)은 JavaScript의 객체 표기법을 따르는 데이터 포맷으로, 객체 내부에 다른 객체를 포함할 수 있어 객체 간 상하 관계를 효과적으로 표현할 수 있다[14]. 이러한 특성을 활용하여, 상위 조문을 JSON 객체로 정의하고 그 내부에 하위 조문 객체를 포함하는 방식으로 법규 및 규정 문서의 계층적 구조를 반영하였다(그림 2).

```
{
 "Chapter": 5

 "Title": "Specifications"

 "Description": [
       "This chapter specifies the requirements that must be satisfied in the brake system."
 ],

 "Article 5.1.": {
       "Description": [
             "Parking brake system ",
             "If a parking brake system is fitted, it shall hold the vehicle stationary",
             "The parking brake system shall: [Item]"
       ]

       "Item": [
             "(a) Have a control which is separate from the service brake system controls;"
             "(b) Be held in the locked position by solely mechanical means."

       ]
 },

 "Article 5.2.": {
       "Description: [.....],

       "Paragraph 1.2.1.": {.....},

       "Paragraph 1.2.2.": {.....}
 }
}
```

그림 2. JSON 구조 예시
Fig. 2. Example of a JSON format

또한, 각 조문 객체에는 해당 조문이 포함된 원본 문서의 정보를 메타데이터로 추가하여, 모델이 다양한 문서에서 생성된 조문들을 구분하고 원문 문맥을 파악할 수 있도록 구성하였다. 계층적 전처리 방법론은 조문 해석에 필요한 모든 정보(예: 수식, 테이블, 주석 등)를 하나의 JSON 조문 객체에 통합함으로써, 모델이 조문의 의미를 종합적으로 이해할 수 있도록 설계되었다. 문서에 포함된 테이블과 주석 같은 보충 정보는 관련 조문 객체에 포함되어, 모델이 해당 정보를 명확히 인식하고 활용할 수 있도록 하였다. 특히, 테이블이나 수식과 같은 데이터는 일반 텍스트로 추출하거나 전달하기 어려운 경우가 많아, 추가적인 변환 과정을 통해 조문 객체에 포함되도록 처리하였다.

3.3 Parent-Child 청킹

계층적 전처리 방법론은 법조문 체계의 계층 구조 반영과 관련 정보의 압축적 전달을 목표로 설계되었다. 그러나 해당 방법론을 통해 생성된 청크들은 다음과 같은 요인으로 의미적 해상도가 저하되는 문제, 즉 개별 청크에 담긴 의미들을 명확히 구분하기에 한계가 나타났다. 첫째, 문맥 정보를 보존하기 위해 Article 단위로 청크를 생성할 경우, 청크 간 토큰 크기의 편차가 심화되었으며 이는 토큰 크기가 매우 크거나 작은 일부 청크들에 대해 검색

정확도를 악화시켰다. 둘째, 동일한 상위 조항에 속한 하위 청크들 간에 내용 중복이 발생하고, 문서 출처와 같은 공통 메타데이터가 청크들에 일괄적으로 삽입되면서 각 청크 간 의미적 변별을 저해시키는 요인으로 작용하였다. 본 연구에서는 이러한 문제를 해결하면서도 계층적 전처리 방법론의 구조적 장점을 유지하기 위해 Parent-Child 청크 방식을 적용하여 데이터 형식을 재구성하였다.

Parent-Child 청크 방식의 핵심 아이디어는 벡터 검색을 위한 Child 청크와 풍부한 문맥 정보를 제공하는 Parent 청크를 분리하는 것이다. Child 청크는 포함된 내용이 적지만, 청크 간 의미적으로 뚜렷하게 구분되는 특징이 있다. Parent 청크는 앞뒤 문맥을 포함한 풍부한 내용을 제공하지만, 청크 간 의미적 구분(해상도)이 상대적으로 낮은 경향을 보인다. 사용자 질의에 대해, 우선 Child 청크를 대상으로 높은 해상도의 벡터 유사도 검색을 수행한 후, 검색된 Child 청크가 속한 Parent 청크를 식별하여 이를 모델에 전달함으로써 풍부한 맥락 정보를 활용할 수 있도록 설계하였다.

본 연구에서는 문서를 문장 단위로 분할하여 Child 청크를 구성하고, 각 문장에 고유 ID를 부여하였다. Parent 청크는 계층적 전처리 방법론을 기반으로 구성되며, 각 Parent 청크에는 포함된 Child 청크의 ID 정보가 저장되어 있다. 사용자 질문 시 벡터 유사도 검색을 통해 적합한 Child 청크를 식별하고, 해당 ID를 기반으로 연관된 Parent 청크를 추출하여 모델에 제공한다.

마지막으로, 본 연구에서는 계층적 전처리 방법론의 효과를 평가하기 위해, 원본 문서를 표 1과 같은 4가지 형식으로 가공하여 기존 방법론과 비교하였다.

표 1. 문서 가공 표
Table 1. Data processing method table

Data	Processing method	Chunking unit
Plain text	None	Tokens*
Markdown	Markdown conversion	Tokens*
JSON	Structure based JSON conversion	Meanings
JSON-PC	Structure based JSON conversion	Meanings**

*Chunk base unit: 500 tokens, overlap unit: 100 tokens
**Includes additional sentence-level chunking

3.4 벡터 임베딩 및 유사도 검색

사용자 질의와 개별 청크 간의 유사도를 효과적으로 비교하기 위해서는 텍스트 데이터를 숫자로 이루어진 벡터로 변환하는 임베딩 과정이 필요하다(그림 3). 본 연구에서는 오픈소스 모델을 통해 임베딩을 수행하였으며, 검색 효율성을 높이기 위해 상용 벡터 스토어 프레임워크를 활용하여 이를 저장하였다.

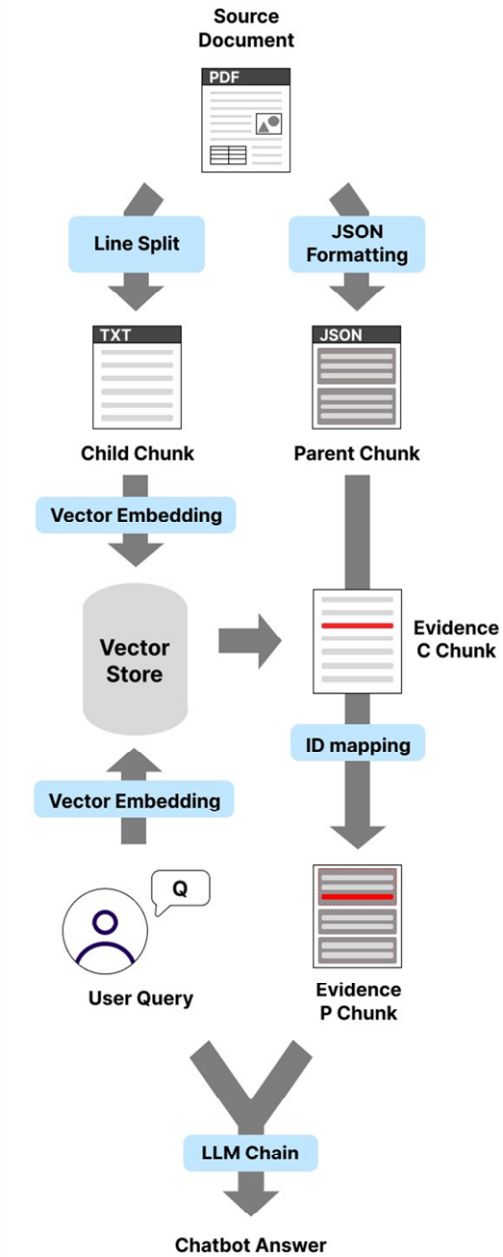

그림 3. Parent-Child 청킹을 적용한 RAG 동작 과정
Fig. 3. Workflow of RAG implementing Parent-Child Chunking

유사도 검색은 코사인 유사도를 기반으로 진행하였는데 이는 개별 청크의 크기와 관계없이 적용할 수 있다는 장점을 가진다. 사용자 질문이 입력되면, 해당 질문을 벡터로 변환한 뒤 벡터 저장소에서 유사도 검색을 수행하여, 유사도가 가장 높은 상위 K개의 문서를 추출한다. 특히 Parent-Child 청크 방식이 적용된 데이터에서는 벡터 임베딩 및 유사도 검색을 통해 사용자 질의와 가장 유사한 K개의 Child 청크를 먼저 식별한다. 이후, 각 Child 청크에 연동된 ID를 기반으로 해당 Child 청크가 포함된 Parent 청크를 확인하고 이를 모델에 전달함으로써 풍부한 문맥 정보를 제공한다.

Ⅳ. 프롬프트 엔지니어링

4.1 프롬프트 엔지니어링 목적

LLM의 추론 성능을 더욱 향상시키기 위해서는 프롬프트 엔지니어링의 적절한 적용이 필수적이다[15]. 그러나 다양한 프롬프트 엔지니어링 기법을 목적에 부합하지 않게 무분별하게 적용할 경우, 오히려 모델 성능에 부정적인 영향을 미칠 수 있다. 본 논문에서 제안하는 프롬프트 엔지니어링은 모델이 사용자 질문에 대해 더 높은 관련성을 가진 답변을 생성하면서 보다 적은 양의 토큰을 사용할 수 있는 프롬프트 설계에 그 목적을 두고 있다.

4.2 프롬프트 엔지니어링 적용 방법

본 연구에서 제안한 프로세스는 사용자의 질문에 대해 연관된 c 청크(Child 청크)를 검색한 후, 해당 c 청크를 p 청크(Parent 청크)와 매핑하여 LLM에 질문과 함께 전달함으로써 답변을 생성하는 방식으로 구성된다. 이 과정에서 중요한 점은, 모델의 성능이나 설계된 프롬프트의 품질이 아무리 뛰어나더라도, 답변의 근거가 되는 p 청크가 적절하지 않으면 올바른 답변을 생성할 수 없다는 것이다. 따라서 사용자 질문에 대해 유사도 기반으로 상위 K개의 근거 청크를 선별하는 과정에서, 선택된 c 청크가 답변 생성에 유의미한 영향을 미친다는 점을 확인하였다. 이를 기반으로, 검색 범위를 확장하는 방향으로 프롬프트 엔지니어링을 설계하였다.

우선, 임베딩 과정에서 검색되는 청크의 수를 늘리는 방법을 도입하였으며, 사용자 질문의 주요 개념과 의도를 반영하여 3개의 추가 질문을 생성하도록 설계하였다. 이 과정은 주요 키워드와 문맥적 유사성을 바탕으로 다양한 형태의 질문을 생성함으로써 검색 결과를 다변화한다. 이를 통해 기존 방식에서는 유사도에 따라 상위 3개의 문서만 검색되던 것을 최대 12개의 추가 청크를 검색할 수 있도록 개선하였다. 그러나 이 방법은 추가적인 청크 검색이 필요하지 않은 질문에도 불필요한 증강 검색을 시도하여 시간 및 비용 효율성을 저하시키는 문제가 발생했다. 이를 해결하기 위해, 검색된 청크가 답변 작성을 위한 충분한 정보를 포함하고 있는지를 먼저 검증하는 단계를 추가하였다. 질문과 검색된 청크를 LLM에 전달하여 유효성을 평가한 뒤, 그 결과에 따라 질문 증강을 진행할지 혹은 바로 답변 생성을 수행할지를 결정하는 구조를 제안하였다.

이 과정은 AI 에이전트 구조를 참고하여 설계되었으며, 전체 프로세스는 세 가지 에이전트로 구성된다[16]. 첫 번째 에이전트는 질문에 대해 검색된 청크의 유효성을 검증하며, 두 번째 에이전트는 필요 시 질문을 증강하여 추가 검색을 수행한다. 마지막으로, 최종 질문과 근거가 되는 p 청크 기반 임베딩으로 답변을 생성하는 에이전트가 이를 처리한다.

위 과정을 종합하여 사용자 질의 입력 시 프롬프트 작동 단계는 다음과 같다(그림 4). 먼저, 코사인 유사도 검색을 기반으로 사용자 쿼리와 유사한 K개의 c 청크을 획득한다. 그럼 다음, 획득한 c 청크와 연결된 p 청크를 LLM에게 전달하여 사용자 쿼리에 대한 답변 가능 여부를 판별한다. 만약 LLM이 답변이 가능하다고 판단한 경우, LLM은 입력된 시스템 프롬프트의 지시를 기반으로 답변을 작성하여 사용자에게 전달한다. 그 반대로 LLM이 주어진 정보로 답변이 불가능하다고 판단한 경우에는 LLM은 필요한 추가 정보를 얻기 위한 질문을 생성한다. 생성된 추가 질문들을 이용하여 첫 단계에의 c 청크 획득을 위한 백터 유사도 검색을 수행하며 이후 과정을 반복한다.

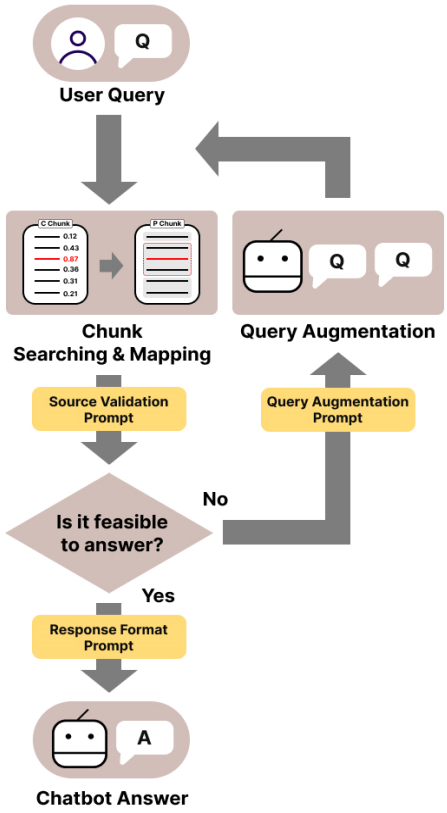

그림 4. 프롬프트 작동 개념도
Fig. 4. Conceptual diagram of prompt operation

V. 평가 및 결과 분석

5.1 평가 질문 제작 방법

본 연구에서는 시스템의 성능을 검증하기 위해 자체적으로 개발한 95개의 질문-답 세트를 활용하여 평가를 진행하였다. 사용자로부터 실제 질문 데이터를 확보하기 어려운 상황을 고려해 UNECE 문서를 참고하여 질문을 설계하였다. 모델이 제공한 답변의 정확성을 객관적으로 평가하기 위해 질문 유형은 참/거짓 문제, 객관식 문제, 단답형 문제로 제한하였다. 참/거짓 문제는 "예" 또는 "아니오"로 답할 수 있는 형식으로 구성되었으며, 객관식 문제는 4~5개의 선지 중에서 올바르거나 올바르지 않은 하나의 선지를 선택하도록 설계되었다. 단답형 문제는 명확한 값, 단어, 또는 짧은 구와 같이 간결한 답변을 요구하는 형식이다. 문항의 내용은 표, 수식, 법률 용어의 정의 및 개념, 그리고 특정 상황에서의 시나리오를 포함하여 다양하게 구성되었다.

5.2 평가 방법

모델이 항상 정확한 답변을 제공하지 않을 가능성을 고려할 때, 문제를 발견하고 이를 해결하는 과정이 매우 중요하다[17]. 이에 따라 본 연구에서는 모델 응답의 정확성을 평가하는 것에 더해, 응답의 근거가 된 청크를 적절히 식별하고 이를 통해 문서의 관련 조항을 정확히 제시할 수 있는지를 함께 검토하였다. 이에 모델의 응답을 정답 여부와 근거 조항 일치라는 두 가지 기준에 따라 표 2와 같이 네 가지 유형(a, b, c, d)으로 분류하였고, 이 중 a 유형 응답만을 정답으로 정의하였다.

표 2. 평가 등급 분류표
Table 2. Criteria for evaluation grade classification

Grade	Answers	Evidences
A	Correct	Correct
B	Correct	Incorrect
C	Incorrect	Correct
D	Incorrect	Incorrect

5.3 평가 결과

만들어진 평가-답 세트와 평가 지표를 활용하여 각 데이터에 대한 결과를 도출하였다. 먼저, Parent-Child 기법을 적용하지 않고 각 데이터를 평가하였다. 이후, 가장 성능이 낮게 나타난 Markdown 데이터를 제외한 상태에서 Parent-Child 기법을 적용하여 다시 평가를 진행하였다. 마지막으로, Parent-Child 기법을 적용한 데이터 중 가장 우수한 성능을 보인 JSON 데이터를 기반으로 프롬프트를 적용하여 추가 평가를 수행하였다. 평가 결과는 다음 표 3과 표 4와 같다.

표 3. Parent-child 청킹을 적용하지 않은 데이터 실험 결과표
Table 3. Experimental results for data without Parent-Child Chunking

Data	A	B	C	D	Accuracy
PDF	60	3	4	28	63%
Markdown	45	3	6	41	47%
JSON	63	6	7	19	66%

표 4. Parent-child 청킹을 적용한 데이터 실험 결과표
Table 4. Experimental results for data with Parent-Child Chunking

Data	A	B	C	D	Accuracy
PDF with parent-child	77	4	2	12	81%
JSON with parent-child	83	2	0	10	87%
JSON with parent-child + Prompts	87	1	0	7	92%

Parent‐Child 기법을 적용하지 않았을 때 정답률은 PDF와 JSON 데이터에서 63%, Markdown 데이터에서 43%로 나타났다. 해당 기법 적용 후에는 PDF는 81%, Markdown은 87%로 정답률이 상승했으며, 특히 JSON 데이터는 92%로 가장 높은 성능을 기록했다.

5.4 결과 분석

앞서 제안한 전처리 방법론을 기반으로 서로 다른 RAG 프로세스를 구축한 뒤, 평가 질문지를 활용하여 데이터 유형별 성능을 평가하였다. PDF와 Markdown 형식의 데이터는 각각 60개와 45개의 정답을 기록했으며, 전처리 방법이 적용된 JSON 형식의 데이터가 가장 높은 정답률을 보였다. 그러나 JSON 데이터는 해상도가 낮다는 문제로 인해 PDF 데이터와 큰 차이를 보이지 않았으며, 문서 기반 답변임에도 정답률이 다소 낮게 나타나는 경향이 있었다. 이러한 문제를 해결하기 위해 Parent-Child 청킹을 적용한 결과, 청킹을 적용하지 않았을 때보다 약 20% 높은 정답률을 기록하였다. 또한 PDF와 JSON 데이터 간의 정답률 차이도 이전보다 더 크게 나타났다. 이는 Parent-Child 청킹이 상하위 구조를 가진 법률 문서를 더 효과적으로 이해하도록 돕는다는 점을 보여준다. 한편, Parent-Child 청킹을 적용한 JSON 데이터에서 발생한 오답은 모두 근거 청크를 제대로 찾지 못한 경우로 확인되었다. 이에 따라 AI 에이전트를 추가적으로 도입하여 근거 청크를 탐색하는 데 집중한 결과, 성능이 더욱 향상된 것으로 나타났다.

Ⅵ. 결론 및 한계점

본 논문에서는 검색 증강 생성을 활용하여 자동차 법률 데이터의 전처리 및 청크 분리 방법을 제안하였으며, 검색 효율을 극대화하기 위한 프롬프트를 개발하였다. UNECE의 제동장치 관련 법률 문서를 활용해 수식과 테이블을 적절히 변환하여 JSON 포맷으로 구조화한 뒤 Parent-Child 청킹을 적용해 사전처리된 데이터를 지식 베이스로 활용한 결과, 상대적으로 높은 정확도의 성능을 달성할 수 있었다.

본 연구에서 제안한 접근 방식은 법률 데이터에 특화된 데이터 전처리 및 검색 방식을 제안했다는 점에서 그 의의가 있다. 뿐만 아니라 그 성능 또한 90% 이상으로기존의 방식 대비 더 높은 수준이나, 이를 실무 현장에 반영하기 위해서는 현장의 높은 기대치를 반영할 수 있을 정도의 성능 개선이 요구된다. 관련하여 오답의 원인 분석을 수행한 결과, 근거 청크 검색 실패가 오답의 주요 원인인 것으로 파악하였다. 가령 문맥에 따라 용어가 지시하는 내용이 달라질 경우, 문장 단위의 청크 안에서는 그 의미를 명확히 특정하기가 어려워진다. 이 외에도 테이블과 같이 도식화된 정보는 자연어 문장과 직접적인 유사도 비교를 통한 검색에 있어 상대적으로 정확도가 떨어지는 경향성을 보였으며, 고정 차원 벡터로 다양한 길이의 문장을 임베딩하는 과정에서 청크 간 검색 공정성이 저하될 가능성이 있음을 확인하였다.

근거 청크 검색의 정확도와 강건성을 향상시키기 위한 방안은 다음과 같다. 첫째, 동일 p 청크 내 문장들을 설정한 길이로 조합하여 확장된 c 청크를 생성한다. 이를 통해 용어의 구체적 의미를 분별하기 위한 c 청크 내 문맥 정보를 보강하는 동시에, 청크 크기 정규화를 통해 의미 비교의 공정성을 확보할 수 있다. 둘째, p 청크의 핵심 내용을 LLM으로 요약하여 추가적인 c 청크로 활용한다. 이로써 개별 문장 단위의 c 청크가 담기 어려운 p 청크의 전체적인 맥락을 검색 과정에서 효과적으로 포함시킬 수 있다. 셋째, Markdown과 같은 특수 포맷의 문법적 표현이 c 청크 검색 과정에서 불이익으로 작용하지 않도록 전처리 과정을 개선하고 핵심 내용을 효과적으로 설명하는 자연어 요약을 청크에

추가함으로써 자연어 쿼리 기반 유사도 검색에 대해 적합도를 향상시킬 수 있다. 마지막으로, 위에서 제시된 여러 청킹 전략으로 생성된 c 청크 후보군에 대해 다중 검색을 수행하고 해당 결과를 재순위화 (re-rank) 하는 방안을 통하여 다양한 질의 상황에서도 최적의 근거 청크를 찾을 가능성을 최대화할 수 있다.

본 연구에서 사용한 단순화된 질문 유형 기반의 평가 체계는 심층적인 질문 유형에 대한 검증에 있어 제한적이다. 또한, 실제 사용자(설계 및 디자인 팀)의 직접적인 평가 없이는 실질적인 유용성을 측정하기에는 한계가 있다. 아울러, 시스템은 유료 모델인 4o-mini를 기반으로 하기에 비용 효율성 측면에서 제약이 있음을 지적할 수 있다. 이러한 한계에도 불구하고, 본 연구에서 제시된 청크 구성 방식과 프롬프트 작동 구조는 자동차의 여러 다른 분야의 규정에도 확장되어 적용될 수 있으며, 이를 통해 전반적인 자동차 개발 과정에서 업무 효율성을 향상시키는 데 기여할 것으로 기대된다.

References

[1] Office of Public Affairs, U.S. Department of Justice, https://www.justice.gov/archives/opa/pr/takata-corporation-pleads-guilty-sentenced-pay-1-billion-criminal-penalties-airbag-scheme. [accessed: Feb. 18, 2025]

[2] Y. Wu, S. Zhou, Y. Liu, W. Lu, X. Liu, Y. Zhang, C. Sun, F. Wu, and K. Kuang, "Precedent-Enhanced Legal Judgment Prediction with LLM and Domain-Model Collaboration", arXiv preprint arXiv:2310.09241, Oct. 2023. https://doi.org/10.48550/arXiv.2310.09241.

[3] H. W. Lee, "A Study on the Improvement of Motor Vehicle Safety Standards", Ajou University, pp. 5-6, Dec. 2010.

[4] G. Yunfan, X. Yun, G. Xinyu, J. Kangxiang, P. Jinliu, B. Yuxi, D. Yi, S. Jiawei, W. Meng, and W. Haofen, "Retrieval-Augmented Generation for Large Language Models: A Survey", arXiv preprint arXiv:2312.10997, Dec. 2023. https://doi.org/10.48550/arXiv.2312.10997.

[5] H. Liu, R. Ning, Z. Teng, J. Liu, Q. Zhou, and Y. Jang, "Evaluating the logical reasoning ability of chatgpt and gpt-4", arXiv preprint arXiv:2304.03439, Apr. 2023. https://doi.org/10.48550/arXiv.2304.03439.

[6] M. S. Jung and J. H. Lee, "Optimal Hyperparameter Combination for Improving Table Data Recognition Rate in Large Language Model based Retrieval-Augmented Generation Systems", Journal of the Korea Institute of Information and Communication Engineering, Vol. 28, No. 11, pp 1282-1290, Dec. 2024. https://doi.org/10.6109/jkiice.2024.28.11.1282.

[7] M. J. Kim, H. I. Jung, S. Y. Lee, S. Y. Kim, P. W. Kang, and M. W. Koo, "Retrieval-augmented LLM based Chain-of-subtasks pipeline for automatic reviewing legal documents", Korea Software Congress 2023, Busan, South Korea, pp. 335-337, Dec. 2023.

[8] H. R. Kang, S. S. Park, and H. S. Kim, "Enhancing Performance of Medical Q&A Systems through Prompt Engineering Research", Korea Computer Congress 2024, Jeju, South Korea, pp. 513-515, Jul. 2024

[9] Korea Transportation Safety Authority - automobile defect recall status, Open Government Data portal, https://www.data.go.kr/data/3048950/fileData.do. [accessed: Mar. 04, 2025]

[10] S. B. Choi and S. K. Jung, "Trends and Market Analysis of Electric Braking Systems", Korea Institute of Science and Technology Information, pp. 60-62, 2017.

[11] V. Lemenkov and P. Lemenkova, "Using TeX markup language for 3D and 2D geological plotting", Foundations of Computing and Decision Sciences, Vol. 46, No. 1, pp. 43-69, Mar. 2021. http://dx.doi.org/10.2478/fcds-2021-0004.

[12] P. Finardi, L. Avila, R. Castaldoni, P. Gengo, C. Larcher, M. Piau, P. Costa, and V. Caridá, "The

Chronicles of RAG: The Retriever, the Chunk and the Generator", arXiv preprint arXiv:2401.07883, Jan. 2024. https://doi.org/10.48550/arXiv.2401.07883.

[13] M. Araszkiewicz, "Conceptual Structures in Statutory Interpretation", In Legal Knowledge and Information Systems, IOS Press, pp. 107-112, 2023. https://doi.org/10.3233/FAIA230952.

[14] J. Xin, C. Afrasiabi, S. Lelong, J. Adesara, G. Tsueng, A. I. Su, and C. Wu, "Cross-linking BioThings APIs through JSON-LD to facilitate knowledge exploration", BMC bioinformatics, Vol. 19, pp 1-7, Feb. 2018. https://doi.org/10.1186/s12859-018-2041-5.

[15] S. E. Park and J. Y. Kang, "Analysis of Prompt Engineering Methodology and Research Trends to Enhance Reasoning Capabilities of ChatGPT and Large Language Models", Journal of Intelligence and Information Systems, Vol. 29, No. 4, pp. 287-308, Dec. 2023. https://doi.org/10.13088/jiis.2023.29.4.287.

[16] LangChain-AI "LangGraph Tutorials": https://langchain-ai.github.io/langgraph/tutorials/. [accessed: Dec. 12, 2024]

[17] H. B. Kim and Y. K. Yoo, "Development of a Regulatory Q&A System for KAERI Utilizing Document Search Algorithms and Large Language Models", Journal of the Korean Society of Industrial Information, Vol. 28, No. 5, pp. 31-39, Oct. 2023. https://doi.org/10.9723/jksiis.2023.28.5.031.

저자소개

안 성 규 (Seonggyu Ahn)

2018년 2월 ~ 현재 : 한동대학교 생명과학부 학사과정
관심분야 : 공중 보건, sLLM, graph RAG

문 벼 리 (Byeori Moon)

2020년 2월 ~ 현재 한동대학교 ICT창업학부 학사과정
관심분야 : LLM, 데이터 분석, 빅데이터 모델링

박 상 범 (Sangbeom Park)

2025년 2월 : 한동대학교 법학부(법학사)
관심분야 : 시스템 보안, 인공지능 보안, 악성코드 분석

김 세 희 (Sehee Kim)

2018년 2월 ~ 현재 : 한동대학교 ICT창업학부 학사과정
관심분야 : 데이터 분석, 사용자 행태 분석, 텍스트 마이닝

김 경 외 (Keungoui Kim)

2019년 2월 : 서울대학교 기술경영경제정책대학원(경제학 박사)
2025년 3월 ~ 현재 : 연세대학교 융합인문사회과학부 조교수
관심분야 : 과학기술정책, 인공지능 응용, 자연어처리