

LLM-BASED SMART BUILDING STATUTE CONFLICT REVENTION TOOL

전희권 · 정상길 · 박재형 | 중꺅마 팀

ChatGPT와 이를 보완해 주는 LangChain 기술을 융합한 건설 플랫폼 제작

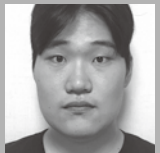
건축물 사용승인을 받기 위해서는 건축물의 계획, 설계, 허가 및 신고, 착공 및 시공 등 많은 단계를 걸쳐야 한다. 이런 건축물의 모든 단계에서 작성해야 하는 문서들이 다양하고 각각 해당하는 법령들이 다르기에 건축주, 설계자, 감리자 등 직접 법령을 찾아가면서 수동적으로 해야 하는 방식이다. 이때, 법령의 기준을 못 미치는 상황이나 누락될 수 있는 상황 등 휴먼에러가 발생한다. 따라서 인공지능 AI를 사용하여 기존의 휴먼에러를 보완해주는 플랫폼 개발을 진행하고자 한다.

시공

CONSTRUCTION &
MANAGEMENT



전희권 Jeon Huik Won
학번 : 2001689
e-mail : huikwon08256@naver.com



정상길 Jeong Sang Gil
학번 : 2001691
e-mail : wjdtkdrf123@naver.com



박재형 Park Jae Hyeong
학번 : 2001657
e-mail : bruce2011@naver.com

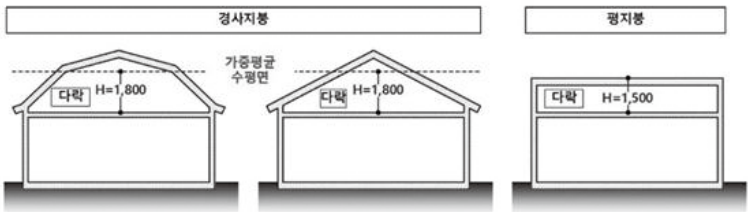


Opening hazard automatic detection system

ChatGPT와 이를 보완해 주는 LangChain 기술을 융합한 건설 플랫폼 제작

건축물 사용승인을 받기 위해서는 건축물의 계획, 설계, 허가 및 신고, 착공 및 시공 등 많은 단계를 걸쳐야 한다. 이런 건축물의 모든 단계에서 작성해야 하는 문서들이 다양하고 각각 해당하는 법령들이 다르기에 건축주, 설계자, 감리자 등 직접 법령을 찾아가면서 수동적으로 해야 하는 방식이다. 이때, 법령의 기준을 못 미치는 상황이나 누락될 수 있는 상황 등 휴먼에러가 발생한다. 따라서 인공지능 AI를 사용하여 기존의 휴먼에러를 보완해주는 플랫폼 개발을 진행하고자 한다.

법령 충돌 관련 사례



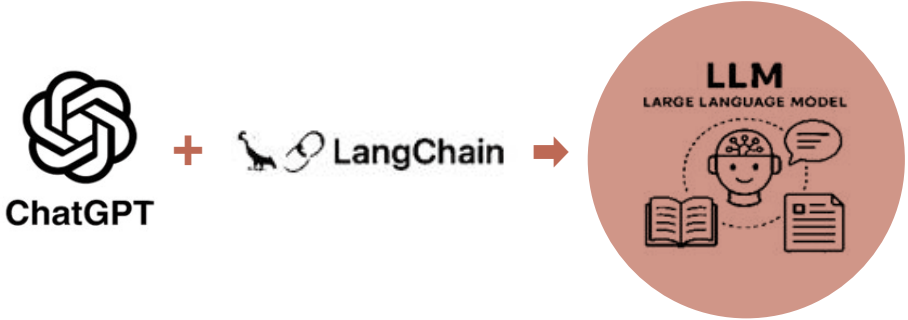
■ 문제 상황

건축주는 경기도 XX시 XX구에 단독주택 건축을 위해
층고가 1.8 이하인 다락이 포함된 지하 1층, 지상 2층의 건축물을 설계

■ 충돌사항

- ➡ 건축법 시행령에 따르면 다락은 층수에 들어가지 않음
- ➡ 지구단위계획 지침에 따르면 해당 다락은 층수에 포함되므로 지상 3층으로 봐야 하여 건축허가 거절함

기술 개요



ChatGPT와 Langchain을 융합한 새로운 LLM 개발
법령, 조례 문서를 자동으로 로딩 후, 사용자 문서 입력 시, 충돌을 자동 탐지

■ LLM(Large Language Model)이란?

방대한 양의 텍스트 데이터를 학습하여 인간의 언어를 이해하고 생성하는 인공지능 모델을 말함.

■ GPT의 한계를 Langchain과 융합한 LLM으로 해결하고자 한 사항

- 데이터 로딩
기존에 학습한 내용으로만 결과를 도출하는 GPT와 달리, 다양한 형식의 문서를 불러와 LLM이 활용할 수 있게 처리하여 기존 학습 범위 밖인 최신 법령·지침까지 반영 가능
- 문서의 분할 처리 가능
토큰값의 제한으로 인해 일정 문서의 양을 넘어가면 해석에 어려움을 겪지 않고, 긴 법령 문서들을 일정 단위로 분할하여 특정 질문에 맞는 조항을 신속히 검색해 맥락 확인 가능

■ 프로그램의 논리적 흐름도



작동 원리

- 제목, 문서 내 Keyword로 해당 문서의 종류 인식

```
def guess_doc_type_from_content(text):
    candidates = {
        "건축허가신청서": ["건축허가신청서", "건축허가", "허가번호", "신청인"],
        "건축계획서": ["건축계획서", "건축개요", "건축면적", "건폐율", "용적률"],
        "주차장설치계획서": ["주차장 설치", "주차대수", "장애인 주차구역"],
        "지구단위계획서": ["지구단위계획", "용도지역", "건축제한"],
        "건축심의자료": ["건축심의", "경관 심의", "조화", "심의결과"]
    }
    for doc_type, keywords in candidates.items():
        if any(keyword in text for keyword in keywords):
            return doc_type
    return None
```

문서 입력 시, 문서의 제목이나 문서 내에 있는 중요 단어들로
해당 문서의 종류를 파악

- 문서가 적용되는 지역과 Mapping

```
@app.post("/analyze")
async def analyze(file: UploadFile = File(...)):
    os.makedirs("user_docs", exist_ok=True)
    filepath = os.path.join("user_docs", file.filename)
    with open(filepath, "wb") as f:
        f.write(await file.read())

    user_docs = load_user_document(filepath)
    user_text = "\n".join([doc.page_content for doc in user_docs])

    filename = file.filename.replace("서울", "서울특별시") \
        .replace("부산", "부산광역시") \
        .replace("경기", "경기도") \
        .replace("인천", "인천광역시") \
        .replace("대전", "대전광역시") \
        .replace("대구", "대구광역시") \
        .replace("광주", "광주광역시") \
```

문서에 적용해야하는 지자체 조례를
파악하기 위해 해당 지역의 조례와 Mapping

- 문서와 그에 맞는 법령들과 Mapping

```
doc_type_law_map = {
    "건축허가신청서": [
        "건축법", "소방시설 설치 및 관리에 관한 법률", "국토의 계획 및 이용에 관한 법률",
        "소방기본법", "화재의 예방 및 안전관리에 관한 법률", "하수도법", "수도법"
    ],
    "건축계획서": [
        "건축법", "장애인·노인·임산부 등의 편의증진 보장에 관한 법률",
        "에너지법", "소방시설 및 관리에 관한 법률", "소방기본법", "화재의 예방 및 안전관리에 관한 법
    ],
    "주차장설치계획서": [
        "건축법", "교통약자의 이동편의 증진법", "교통영향평가 지침"
    ],
    "지구단위계획서": [
        "국토의 계획 및 이용에 관한 법률", "도시개발법", "환경영향평가법", "문화유산의 보존 및 활용에
    ],
    "건축심의자료": [
        "건축법", "경관법", "문화유산의 보존 및 활용에 관한 법률", "도로법"
    ]
}
```

문서마다 적용되는 법을 인식하고, 이에 따라 적용

- 대답 형식 조정(프롬프트 지정)

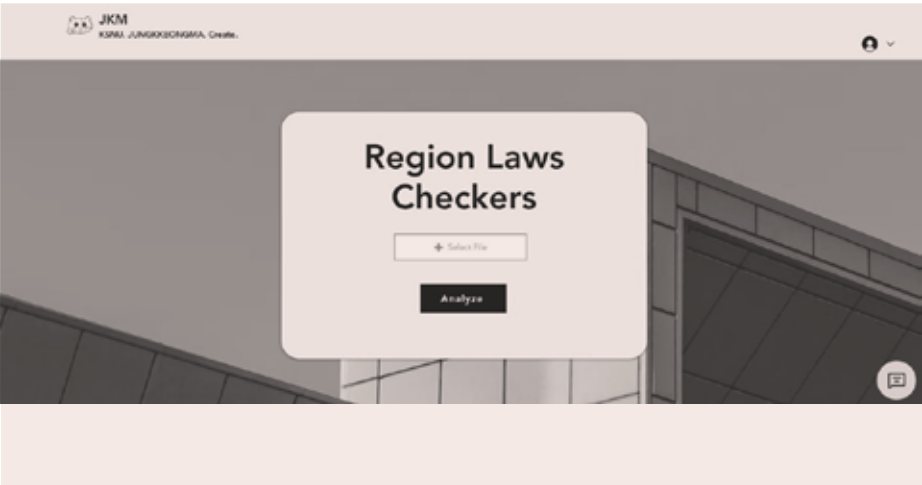
```
1. 문서 내용이 아래 법령 또는 지역 조례와 **충돌되는 항목**을 조항 번호와 함께 지
적해 주세요.
2. **충돌이 발생한 경우, 어떤 법령(혹은 조례)이 우선 적용되는지** 판단해 주세요.
3. **우선순위 판단 기준은 아래와 같습니다.**
   - (1) 법률 > 시행령 > 시행규칙 > 고시 > 지침 > 조례 > 시행서 > 민간표준 순으
로 상위법 우선
   - (2) 조례가 법령보다 **강화된 경우** 유효, **완화하거나 위반 시** 무효
   - (3) 특별법이 일반법보다 우선됨
4. 판단 이유를 간단히 설명해 주세요.
5. 충돌이 없다면 '충돌한 내용이 없습니다.'라고만 출력하세요.
"""
result = chain.invoke({"query": query})

full_result = f"""
■ 문서 유형: {doc_type}
적용 지역: {region_name if region_name else '지역 정보 없음'}
적용 법령: {'', '.join(doc_type_law_map[doc_type])}
```

법령의 우선순위를 적용하여 답변 도출 유도
법령이 충돌할 시, 충돌되는 항목을 조항번호와 함께 자세히 설명



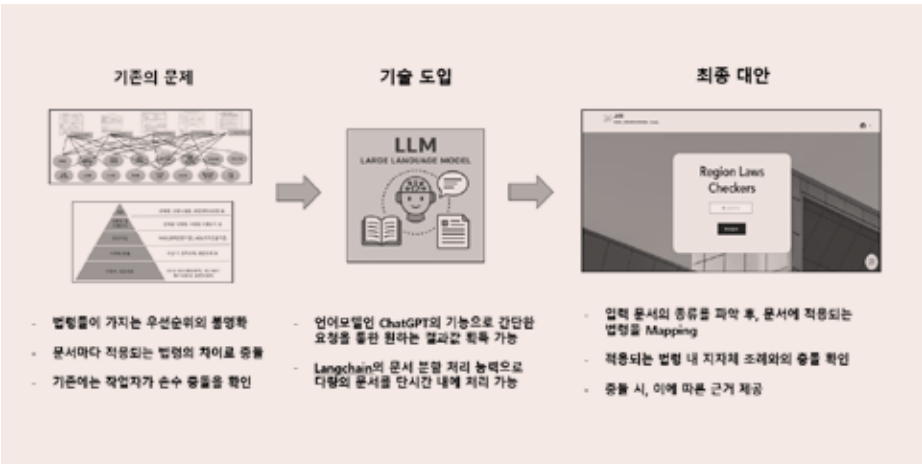
기능 구현
예시



■ 실행 방법

- 1) 분석하고자하는 파일 선택
- 2) 우측의 검토 요청 클릭
- 3) 건축 법령과 지자체 조례의 충돌에 관한 답변 획득 가능

■ LLM 기반 스마트 건축 법령 충돌 방지 도구



기대효과

- **다량의 법령으로 인한 복잡성 해결**
문서마다 적용되는 법령들이 서로 각기 다르고, 법령 간 우선순위도 존재한다. 따라서 수작업으로 검토할 시, 휴먼에러나 법령의 중복 발생 가능성을 시가반인 LLM을 통해 보다 정확하게 검토하여 해결할 수 있다.
- **지연, 사업비 증가로 법적 분쟁 가능성 완화**
법령 충돌로 인해 공사가 중단되는 경우 지연으로 인한 사업비의 증가나, 이에 따른 법적 분쟁 가능성을 사전에 차단하여 예방할 수 있다.
- **행정적 효율성 향상**
수작업으로 검토를 진행하는 것과 달리, AI기반 컴퓨터로 작업을 진행하므로 소요 시간이 감소 되어 업무의 질적 향상 효과를 기대할 수 있다.
- **투명성 강화**
충돌 여부뿐만 아니라, 충돌이 발생한 조항을 언급해주므로, 충돌 내용에 대해 빠른 수정이 가능하다.

개선 사항

- **지자체 조례에 관한 코드의 부재**
건축 관련 법령들은 '법령정보센터 API'를 통해 실시간으로 최신의 법령을 프로그래밍 코드 형식으로 발급받을 수 있다. 그러나 지자체 조례를 묶어 배포하는 코드가 아직 존재하지 않으므로, 해당 코드의 개발 향후 계획을 고려해볼 필요가 있다.
- **GPT 5.0의 등장**
현재의 LLM은 GPT 4.0 기준으로 개발된 프로그램이며, 2025년 8월 8일에 개발된 GPT 5.0을 사용한다면 보다 더 나은 답변을 얻을 확률이 올라갈 것으로 예상된다.