**GPT 활용**

**응급처치 챗봇**

**팀장**

강민호

**팀원**

김서연 김진유 박경희

**목차**

1. **선정배경**
2. **프로젝트 개요**
3. **프로젝트 결과**
4. **결론**

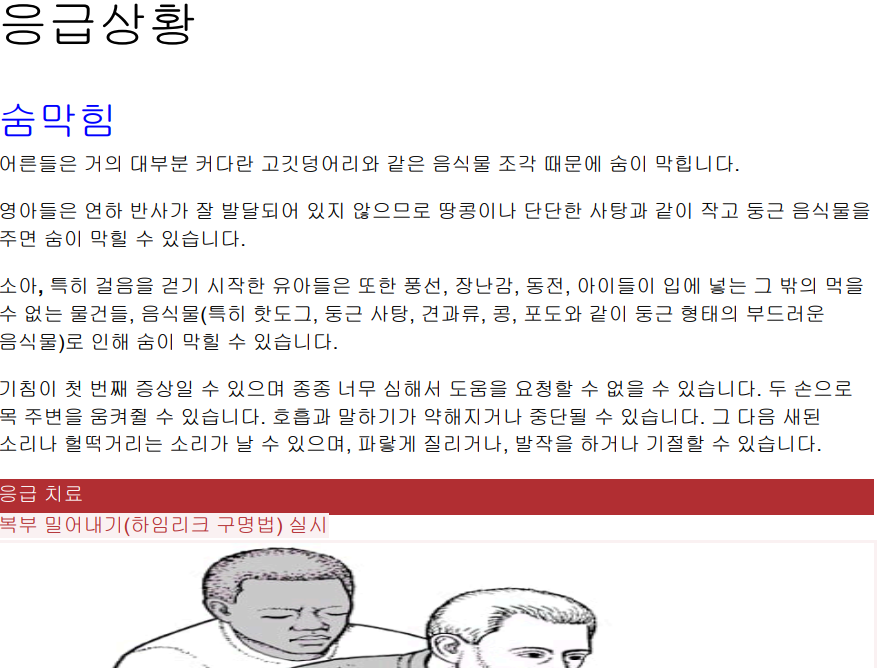
## **1. 선정 배경**

뱀에 물렸을 때의 응급조치 방법을 알고 있는가? 영화나 드라마를 보면 뱀에 물렸을 때 물린 곳 위를 세게 묶거나, 입으로 독을 빨아내는 장면을 볼 수 있다. 허나 이는 물린 곳을 괴사시키거나, 독을 빤 사람 또한 중독되거나, 2차 감염을 일으키는 등 오히려 상태를 악화시킬 수 있다.

위처럼 일상의 응급처치가 필요한 상황에 응급처치 방법을 모를 때, 이에 대한 정보를 쉽게 알 수 있도록, 올바른 응급처치 방법에 대한 답변만 생성하기 위해서 GPT모델에 langchain을 이용해 RAG를 적용하여 응급처치 가이드 챗봇을 구현하려고 한다.

## **2. 프로젝트 개요**

1. **데이터 수집 및 전처리**
   1. MSD 메뉴얼 홈페이지[[1]](#footnote-0)에서 각 응급상황별 조치에 대한 옵션을 선택한 뒤 텍스트를 크롤링하여 PDF로 저장한다. 해당 PDF는 제목, 챕터, 본문으로 구성된 클래스를 생성하여 제작한다. 만약 크롤링이 불가능한 정보가 있다면 url으로부터 PDF파일을 받아와 데이터에 추가한다.
   2. 응급처치 AI 챗봇을 제작하기 위해 저장된 응급처치 PDF 데이터를 불러온 뒤, RecursiveCharacterTextSplitter를 사용하여 텍스트를 청크단위로 분할한다. 이때 chunk\_size=500, chunk\_overlap=50으로 지정하여 청크의 최대 길이 및 텍스트 중복 옵션을 설정한다.
   3. 분할된 텍스트 청크를 Chroma.from\_documents()으로 벡터 데이터베이스에 저장한다. 이 벡터 데이터베이스에서 검색 기능을 제공하는 retriever 객체를 생성한다. 이를 통해 사용자가 입력한 자연어로 된 질문에 대해 관련 텍스트를 찾아 빠르게 답변을 받을 수 있도록 구성하였다.
   4. 타자치기 어려운 상황에서 사용이 용이하도록 음성인식 기능을 추가하였다. pyaudio 객체를 생성한 뒤, ‘whisper-1’모델을 사용하여 음성을 텍스트로 전환한다.
   5. 수집된 데이터



응급 상황명, 원인, 증상, 처치방법이 수집되었다. 이중 동일한 응급상황에 대한 중복데이터가 있는 경우, 챗봇 응답에 필요하지 않은 주석, 링크와 같은 데이터를 제거한다.

1. **모델 설정**
   1. LLM  
      **GPT-4o mini**

이 모델은 대화형 인터페이스에 최적화된 언어 모델이다. 챗봇이라는 특성상   
 사용자와 AI가 대화를 통해 정보를 제공해야 하므로 해당 모델을 채택하였다.  
 GPT-4o mini는 오픈AI가 제공하는 가장 작은 모델인 'GPT-3.5 터보'를 대체한

다.

12만8000개의 토큰으로 구성된 컨텍스트 창을 가지고 있으며 GPT-4o와 공유하는 개선된 토크나이저가 있어 영어가 아닌 텍스트 처리에 효율적이다.

평균 출력 속도는 초당 202토큰으로, 'GPT-4o' 및 GPT-3.5 터보보다 ​​2배 이상 빠르기 때문에 LLM을 사용하는 AI 개발에 적합하다.

**Temperature**

2에 가까워질수록 창의적인 답변을 제공한다. 정확하고 객관적인 답변을 해야  
 하므로 0으로 지정하였다.

* 1. 템플릿

챗봇AI를 제작하기 위해 아래와 같은 템플릿을 제작하였다.

이전 대답을 기억하여 답할 수 있도록 하며, 만일 질문이 명확하지 않다면 더 자세한 정보를 묻도록 하였다. 사용자의 질문 중, 만약 문서에 없는 내용일 경우 답변을 지어내지 않고 알지 못한다는 응답하도록 작성하였다.

| template = """Use the following context to answer the last question.  (다음 장에 이어서)  If the question is ambiguous to answer, request additional information.  If you are responding to a question related to first aid, please provide instructions in a numbered sequence to make it easier to follow.  When analyzing the question, consider who, when, what, how, and why.  If the question refers to content not available in the document, state that you do not know the information.  Always respond in Korean.  절대 한국말로 답변해  {context}  질문: {question}  도움이 되는 답변:""" |
| --- |

1. **함수 분석**

**전처리 부분**

* 1. is\_relevant(query)

주어진 쿼리(문자열)가 특정 키워드 목록 중 하나라도 포함하고 있는지를 확인하여 해당 쿼리가 관련성이 있는지를 판단한다.

* 1. respond\_to\_query(query)

사용자의 질문(query)가 응급 상황과 관련있는지 판단하여 관련있다면 기존 RAG chain이 처리할 수 있도록 None을 반환, 그렇지 않다면 응답하지 않도록 한다.

* 1. voice\_to\_text()

녹음된 사용자 음성파일을 열어 음성인식 모델(whisper)를 사용하여 텍스트로 전환한 뒤 반환한다.

* 1. rag\_chain(query)

질문이 응급처치와 관련이 없는 경우, 미리 지정해둔 응급 상황과 관련된 질문을 해달라는 답변을 반환한다.  
질문이 응급처치와 관련 있는 경우, 관련 문서를 가져와서 Rag-Chain을 통해 응답한다.

* 1. record\_audio()

이 함수는 함수는 사용자의 목소리를 녹음한 후, 해당 음성을 텍스트로 바꾸어 반환한다. 이 함수는 음성 녹음과 오디오 파일 생성, 그리고 녹음된 음성을 텍스트로 변환하는 작업을 수행한다.

* 1. print\_msg(prompt)

사용자 메시지와 그 메시지에 해당하는 AI의 응답을 출력하는 함수이다. 챗봇 특성에 걸맞게 Stream 형식으로 출력이 되도록 하였다.

* 1. streaming\_md()

웹의 main에서 텍스트 답변을 위한 AI 응답 출력 함수이다.

**Streamlit 구현부**

1. main

streamlit.chat\_input() 함수를 사용하여 챗봇 환경을 구축하였다. 사용자가 입력한 내용이 rag\_chain을 거쳐 적합한 응답을 출력한다. 사용자와 AI 아이콘에 구분을 두어 가시성을 높였다.

1. side bar

웹 좌측에 음성 챗봇을 구현하기 위해 녹음 버튼과 안내 문장을 출력하였다. REC 버튼을 누르면 5초간 녹음이 시작되며 만들어진 녹음 파일은 텍스트로 변환되어 rag\_chain을 통해 main 챗봇 부로 응답을 출력한다.

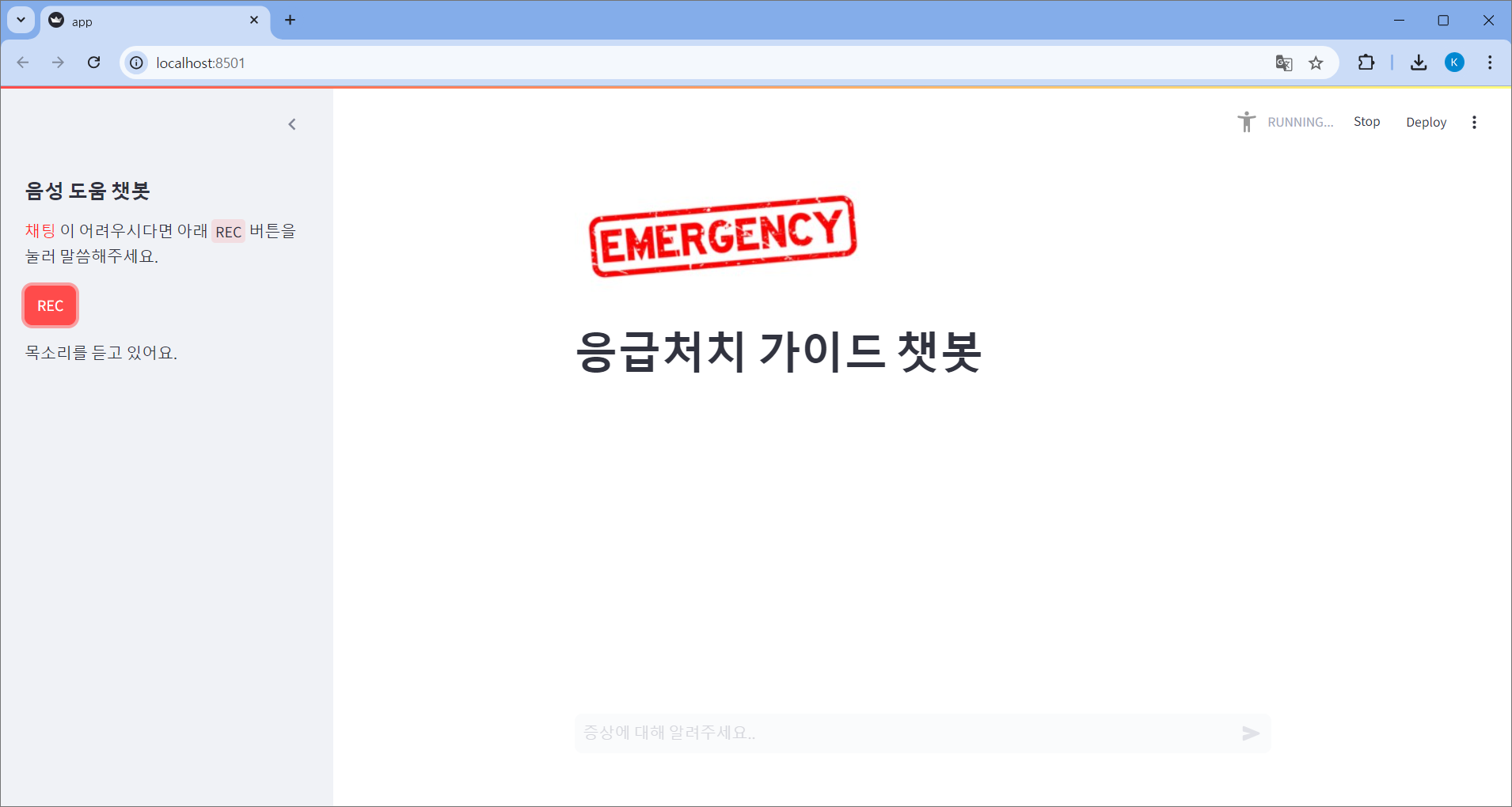
## **3. 프로젝트 결과**



streamlit 실행 시 초기화면

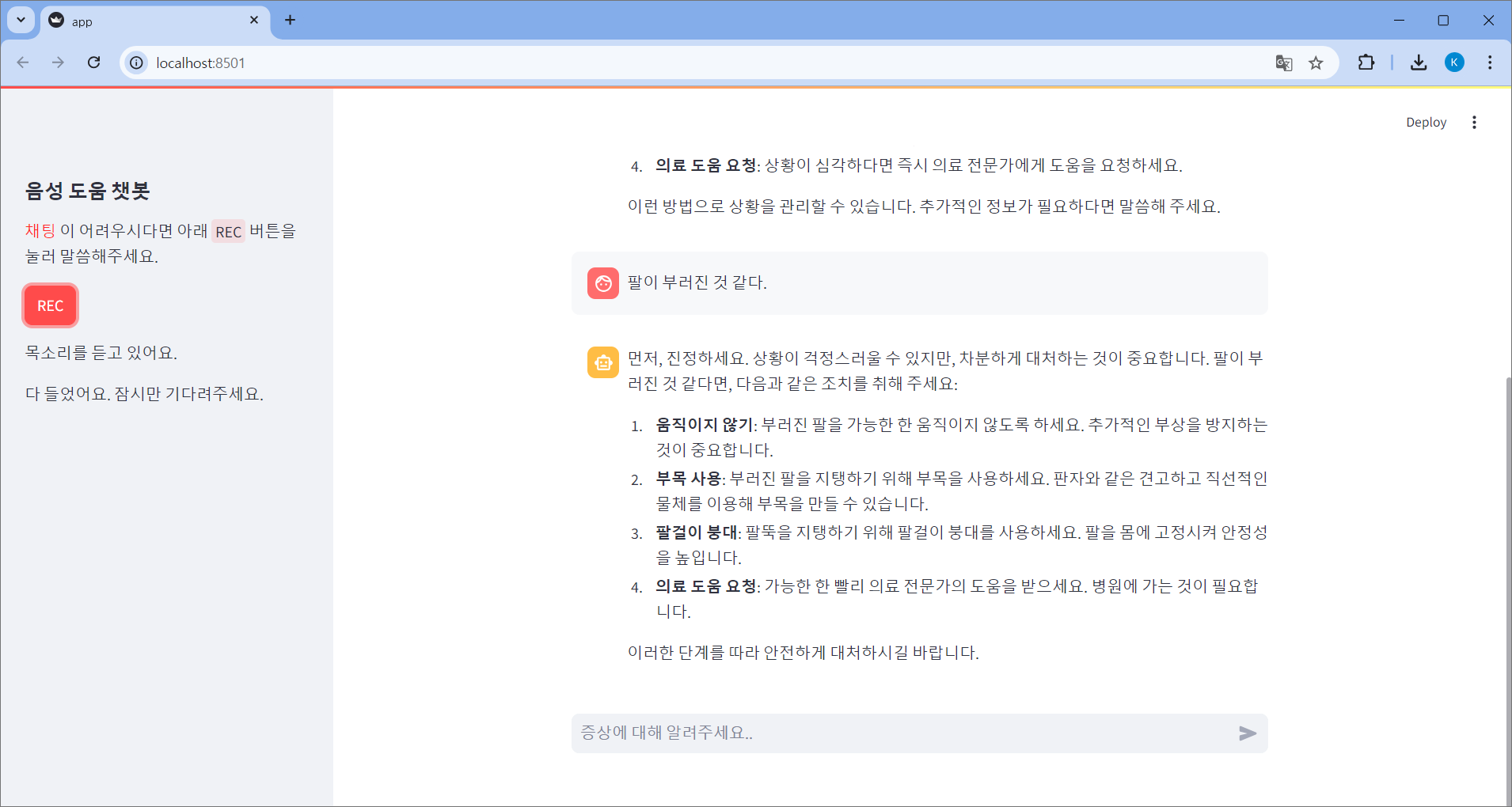


입력에 ‘팔에 금간거같아’라고 입력 후 화면



음성 입력을 위해 [REC] 버튼 입력 후 화면

[REC] 버튼 밑에 ‘목소리를 듣고 있어요.’ 라는 문구가 출력된 것을 확인할 수 있다.



음성 도움 챗봇을 이용하여 ‘팔이 부러진 것 같아’라고 발화 후 화면

## **4. 결론**

1. 구현사항

- MSD 메뉴얼 홈페이지에서 응급상황과 상황별 응급조치에 대해 크롤링에 성공하였다. 일부 크롤링이 되지 않는 정보에 대하여 url에서 pdf 파일로 제작하여 데이터베이스를 완성하였다.

- AI 챗봇 형태로 사용자가 입력한 질문에 대해 응급 상황인지 판단하고 응급조치가 필요한 경우 답변을 하는 기능 제작에 성공하였다.

1. 추가 구현사항

- 초반 app.py에서는 모델이 langchain의 Rag를 사용하여 문서를 참조해 답변을 구하였다. 이 방식은 streamlit으로 구현된 홈페이지가 로드될 때마다 chromadb에 pdf데이터를 매번 로드하고 임베딩이 필요하여 홈페이지 로딩이 오래 걸린다는 문제가 있었다. 이를 해결하기 위하여 chromadb에 임베딩 완료된 데이터를 미리 chromadb폴더에 저장해두었다. 그 결과 홈페이지를 로드할 때 문서를 임베딩하는 시간을 제거하여 속도를 높일 수 있었다.

- 채팅으로만 챗봇을 구성하였지만 급박한 상황에서는 타자치기 어려울 수 있다. 그렇기에 음성 인식 기능도 추가하여 보다 다양한 경우에서 응급 챗봇 애플리케이션을 사용할 수 있도록 하였다.

1. 개선점

- 실제 서비스에 적용 시, 정말 위급한 상황에 119 신고와 연동되도록 한다면 GPS 신호와 증상 설명시간을 단축시킬 수 있어 효과적인 서비스 제공이 가능하다. 혹은 말을 할 수 없는 상황이거나, 청각 장애의 경우에도 사용 및 신고가 가능하게 한다면 접근성을 더욱 높일 수 있을 것이다.

- 초기에 제대로 된 처치만 해주어도 살 확률이 높아지는 심근경색 및 뇌졸중의 경우 전조 증상을 가볍게 넘기기 쉬워 사망률이 상당히 높다. 챗봇의 경우 검색엔진을 이용할 때와 비교하여 내가 원하는 정보를 빠르고 친근하게 얻을 수 있어 증상에 대한 검색을 어려워하는 사람에게도 도움이 된다.

- NPU가 들어가는 최신 스마트폰에서 구동하는 AI 챗봇 애플리케이션을 개발하여 기능을 더욱 개선할 수 있다. 데이터가 원활하지 않은 조난당한 경우에서 사용할 수 있어 응급 상황에 대한 골든 타임을 확보할 수 있을 것이다. 모델을 경량화하여 스마트폰 앱으로 제작이 가능하다는 것이다.

- 향후 사진을 찍어 업로드 했을 때 사진을 분석하여 그에 맞는 응급 조치 정보를 제공하는 기능을 추가할 수 있다. 화상이나 벌레에 물린 경우 사전 정보가 풍부한 사람이 보다 적합한 응급 조치를 시행할 수 있기에 이를 응급 챗봇으로 대체한다면 더 나은 조치를 취할 수 있게 될 것이다.

1. https://www.msdmanuals.com/ko-kr/home/응급상황-및-손상 [↑](#footnote-ref-0)