9/14~

최종방향성

1.질문핵심파악

2.답변중에 어투가 이상한것들에대한 대체 질문(추천)

3.답변이 잘 나왔는데, 이와 관련된 질문을 물어봄

[것을 어떻게 설계하고 모델링할것인가..

bertModel을 구성하는 모듈이 하는역할]

일단이건 fix

최종해결목표 -> 챗봇의 질문의 답변이 hallucination 인지 아닌지 + 추가적으로 학습된 내부 DB를 참조했다면, 어디를 참조했는가 에 대하여

그렇다면 NLP 적 관점에서 보았을 때, hallucination이 무엇인지, 이게 어떤 이유에서 발생하는지, 이것과 정상적인 답변을 구분하는 해결점은 무엇인지

오픈소스 파인튜닝 끝~? [큰 방향성]

1. 임베딩에서 hallucination 판단열쇠가 있나?

-논문읽고 이해 / 정리 할 내용있으면 정리

-임베딩을 해서 고차원벡터로 변환되었을 때,

-일단 QA챗봇이므로 웬만하면 DB(매뉴얼 문서)를 참조해서 나오는 형식이면 좋은거니까, 그렇게 할 수 있는 방법과

-

1. Kochat 들여다보고 시스템 이해해보기(이거는 좀 별로인듯)

<https://github.com/hyunwoongko/kochat/blob/master/docs/02_about_chatbot.md>

1. Langchain 강의 및 티스토리 참고해서 코드 테스트 해보며, 원리 파악(db에서 끌어와서 쓰는[\*\*이걸 뭐라하지?->DB참조?] (수요일 수행예저)

-코드는 짜놓음, 이 세부적인 테스크를 이해해보려고 함

3.softGpt가 모델을 파인튜닝하는 식이 아니기 때문에 프롬프트나 그외의 것을 어떻게 잘 변경 시켜야 hallucination이 안일어날수있을까?

4.Hallucination이 발생하는 근본적 원인에대해서 깊은 탐구(논문읽기)

5.학습된 DB를 참조할수있으면, 우선적으로 참조하는 방법(이건 문서를 참조하는 QA봇 코드를 공부해보면서 어떻게 하면 우선참조 가능할까 연구[Langchain])

9/18~

[월요일]<vicuna, korani써서 프롬프트, input 파인튜닝해서 deploy해보기>

\*\*프롬프트 구조:

PROMPT = """우리는 아래와 같은 정보를 갖고 있습니다.

---------------------

{context}

---------------------

### 주어진 정보에 따라, 질문에 답해주세요.: '{question}'

### Assistant:"""

\*\* 프롬프트 테스트 + Testcase짜기

[화요일]

\*\*실패

\*\*Hugg ace api다운로드(모델 크기가 너무커서 api형식으로는 불가능하였음)

\*\*어떻게 이걸테스트 하는지 잘 모르겠음 자꾸 런타임 에러 뜨고, 테스트,deploy어떻게 해야할지모르겠음(실패)

\*\*논문읽기전 reasoning, hallucination, Interactivity 에 대한 search

[수요일,목,(금)]

\*\*\*\*논문정리 (이해안되서 다시 기본개념부터 좀보기) [목요링1번]



\*\* [롬프트 엔지니어링]

------------------

\*\* 문서를 기반으로 대답하는 챗봇 유투브,자료 참조해서 주석달고 깃헙 올리기

\*\* sentence\_embedding 코드 보고 이해

\*\* gpu 사용방식과 cuda 및 gpu를 쓰면 뭐가좋은지 개념혼동 다시찾아보기

\*\*비쿠냐 gpu들어오면 돌려보기 (협력)

\*\* 감정기반 분류로 kobert 어떻게 쓰고, 임베딩 어떻게 되는지 알아보기 코드테스트(해보고싶었다)

\*\* pre\_trained모델을 어떻게 사용하는지, 상세 방법 tokenizer, (encoding)코드화

\*\* 프롬프트에 ‘이렇게이렇게 나오게 해줘’ 라고 해서 다 그대로 나오는게 아니다. 그것을 정교화 할 수 있는 것을 찾아내야한다?

-금요일

\*\*LLM hallucination관련 논문읽기(위의 정답에 답이 될수있들만한 것 체크하기)

<\*\*\*\* 그렇게 하기위해 문서기반 챗봇을 일단 만들어보고 그 과정에서 해답을 얻어보자>

[사용기술]예상

-langchain의 openai 임베딩, 리트리버 QA, openai llm(chatgpt 3.5 turbo), chromaD

갤러리 참조

문서를 불러오기(Data loader)

헷 api 에 입력토큰수 맞추기

임베딩 방식과 문서를 벡터DB에 저장(chroma DB)

이 벡터 DB를 기반으로 검색해주는 retriever설정 vectordb.as\_retriver

체인만들기(검색+답변생성)

설정된 파라미터들

<Llm, chaintype, 리트리버, source\_document->출처표시 true>

+일단 영어로 질답이 되어있는데 한글로 되는지, txt파일인데 pdf도 되는지 추가적으로 테스트 해보기 (될거같음)

어떻게 문서기반 QA챗봇이 이루어지는지 과정을 알긴알았고, 여러가지 메소드,모듈라이브러리 등을 알았다.

\*\*메소드들 어떻게 구성되어있는지 알아보기 langchain(이게 결론적 해결에 의미가 있을지는 싶다)

\*\*여기서 hallucination test -> 문서에 없는 질문이나, 대답할 수 없는 형식으로 질문을 해보면 얘는 어떻게 반응할까

\*\* 이를 해결할 수 있는 방법은 뭘까

\*\*

근데 이 임베딩과 리트리버 모듈의 내부를 볼수없으니…

이 과정속에서 이루어지는 hallucination을 어떻게 방지하지? ,출처 표시는 vectordb를 통해서 해해결할수있을 것 같다. -> 논문을 읽자 +

++문장과 문장간의 유사도를 판별하는 실습을 어느 방법이라든지 해보자

++문장에서 핵심 context를 파악하는 실습을 해보자

\*\* 주는 일 ++

사용자 질문임베딩

영어 트랜스래이션

모델갈아끼워보고, 잘안되면 케이스

Use question encoder

필터링

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

사용자 질문에서 어떻게 답을 찾는가

qa

\*\*월 문서기반 답변 QA 챗봇 결과가 어떻게 나왔으며

검색기를 사용해서 llm\_response가 나왔는데, 결과를 일단 잘 해석해보고 이렇게 답은 잘나온다 식으로

이런걸하는이유, 질답과정에서 hallucination이 안생기고, 출처를 분명히 할수있게 하는게 최종목표인데, 이걸 구현하기 위해서는 그이전의 과정들도 필요하기 때문에 QA질답의 전체적인 과정을 이해해보려고 계속 llama,langchain,openai같은 오픈소스에서 코드를 찾아서 테스트를 해보고있는중이다. 그리고 이런 일련의 과정말고, llm(kobert,vicuna,)를 이용한 테스트도 해볼예정이다.

\*\* 문서데이터만 넣은 것 말고, QA데이터셋을 추가로 학습시킨 것을 합쳐서 나오게

\*\* 논문제대로 파악해서 위에서 계속 질문해오던것들을 답할수있게

\*\* 프롬프트입력- > 출력 잘보기 hallucination없나 있나 그 기준이 뭔가[앞으로의 위에서 제시한 해답이나, 방향성 찾아보기]

Test\_case

gp t를 통해 생성된 정보를 바탕으로한 정보를 통해 나오는 답변이 정확하게 prompt기반된 답변인지(G P T 3.5) [vicuna 33b 테스트기로 프롬프트 테스트]

--수욜 9/27

\*\*Hallucination 논문 의문점 및 정리

해소되지않은 질문: chatgpt같은 경우 입력창에 context가 아닌 질문만 입력하면, 답변이 형식과 질문의 의도에 맞게 잘나오는데, 이런경우 질문이 어떤 질문이 나올지 모르는데, context를 그에 따라서 어떻게 설정하는지, 아니면 질문 답변 쌍을 훈련을 시켜서 질문만 넣으면 답이 나오는건가? // 그러니까 context가 미리 결정되어있지 않고 질문만 넣으면 context를 설정해서 답변이 잘나오게 하는 방법,

이논문을 우리gpt에는 어떻게 적용할수있을까 내가 생각

우선 컨택스트가 참조하는 문서(manual DB,질문 입력->컨텍스트 결정(리트리버 사용?(이 질문에 맞는 컨텍스트를 매뉴얼 DB나 질답 데이터 셋에서 검색할때, 코사인 유사도나, 군집거리 등으로 제일 질문을 잘 참조할수있는 것을 찾음, DB 나 컨텍스트에 태그를 부여해서()context에 해당하는 부분에 태그)->프롬프트 완성->질답 데이터셋과 매뉴얼 DB가 pretrained 된 언어모델(Bert,gpt,vicuna)에 입력-> 추론->디코딩 및 답변출력 이런 형식으로 나오는게 맞는지 ,

\*\*Sql generation이나 다른업무를 수행하기 위해 인코더를 만들고, 임베딩을 어떻게 할지 아직 결정이 안난 상태에서 코드를 구현해야 하는 목표가 있는 것 같다. 따라서 여러가지 NLP 모델 테스크 들을 해보면서 감각을 익힐예정 –Bert/KoBert

-KoBert -> 한국어 위키백과 문장 5백만개, 뉴스기사 2천만개 학습 LLM 모델

-자연어 문장의 감정 분류

-

\*\* 논문에서 언급된 태그 방식을 사용하는 방법을 하려면 질문이 참조하는 컨택스트가 참조하는 문서(매뉴얼 DB, QA데이터셋

\*\*10/4 ~

1순위: 주시는 업무

-코드 메소드 상세분석

<langchain 사용>

{\*\*문서에 있는 article내용을 context로써 사용하기 위해, LLM 모델의 context length 에 초과되지 않게 순서대로 context text를 분류 해서 documents변수에 리스트 형식으로 넣어준다

\*\*vector DB 사용(from langchain.vectorstores import Chroma) <https://velog.io/@tura/vector-databases> ->sqlite파일로 vector화된 데이터가 DB로 저장되는데 이 DB안에 있는 내용을 확인하여서, 어떤데이터가 들어갔는지, 어떻게 임베딩ㅇ됬는지 확인

\*\*Chroma 에서 .as\_retreive , as\_relevant\_document 등을 사용해 vectorDB내 주어진 질문에 대답할수있는 답변검색

}

\*\*답변이 여러 개의 문서를 참조해서 잘 나오는지, 원리가 뭔지 어떤 프롬프트 형식을 쓰면 이렇게 되는지 (langchain 라이브러리 )

\*\*문서가 참조된 부분이 참조된 방식(어떤식으로 참조됬는지, 참조 정도는 어느정도인지, -> 직접 예시를 들어서

+RAG 와 retreiver에 대해서 좀 research + transformer에대한 상세한 이해 해보기 -> context의 선택 방식에 Rag 나 retreiver의 역할이 크기 때문

+이게 어느 알고리즘을 사용되면 이렇게 될까 (코사인 유사도, 맨해튼 거리 유클리드 거리,

+직접 코드 좀 짜보기 retreive코드참조

\*\*답변 분석

답변:”what is generative AI?”

참조 context :서로다른 3개의 문서

->어떻게 참조하는지 모르겠어서 문장 문단, or 문장 문장의 유사도를 구하는 것을 코드로 구현해보고자 함

-임베딩

10/5

답변이 잘 나왔다고 가정하고, 이와 관련된 질문을 물어봄 ->->어떻게 실행시킬지 방향성

-방법론적 방향-

기본 토대: 질문 질문 유사도 konlpy , okt 로 한글 형태소 단위로 토크나이징을 한다

Try A. 원래의 질문 셋을 vectorDB(weviate, ChromaDB)등에 임베딩후 집어넣고, 사용자가 질문을 했을 때, retriever query 에 그 질문을 집어넣고 검색하면, 벡터 거리 기반으로 유사한 질문이 나올것같다.

Try B. word2vec, Tf-Idf 등의 벡터화 방식 사용해서 유사도 거리 구하기, 거리가 가까운순 sorting 하는 방식으로

Try C. LLM의 토크나이저,임베딩 사용

-유사 케이스 조사(은행 챗봇 위주)-

기업이나 은행 챗봇의 경우 질문 종류가 자신이 물어보려는 키워드 중심의 질문이 많았고, 추천질문 역시 물어보는 목적어가 되는 키워드를 포함하는 질문이 많아서, 문장 벡터화와 유사도 측정 중점을 핵심 단어를 파악하는 것이 중요, 추천질문중 겹치는 단어가 있다고 해서, 그둘이 유사한가는 생각해봐야할듯

1. Langchain 코드 따라가 보면서 프롬프트가 어떻게 구성되어있는지, response가 어떤방식으로 나오는 지 따라가보기[너무 얽혀있고, 프롬프트가 제대로 안나와있는것같아서 이건 일단 보류하고 -> (llm\_response from multic docs) 이거에 대해 search

<https://medium.com/@tushitdavergtu/multiple-document-summary-and-llm-powered-qa-system-9b4df618c22>

<https://betterprogramming.pub/building-a-multi-document-reader-and-chatbot-with-langchain-and-chatgpt-d1864d47e339>

-langchain이 쿼리(+프롬프트) 가 구성되는 그 전까지의 과정

제일중요해보이는 QA chain이 형성된 근간이 되는 vectordb를 만든 Chroma 객체 ->langchain.vectorstores 🡪 vectorDB생성 = 사용자가 쿼리(질의)를 하면 벡터 DB에 분포값과 가장 가까운 벡터거리 순으로 질의와 가장 가까운값을 return해주는 형식

from langchain.docstore.document import Document

from langchain.schema.embeddings import Embeddings

from langchain.schema.vectorstore import VectorStore

from langchain.utils import xor\_args

from langchain.vectorstores.utils import maximal\_marginal\_relevance

1. 한글-영어 번역 모델 deploy

-vicuna korani move +프롬프트 엔지니어링 <git>

10/6

4한글-영어 모델 deploy

1sql모델 사용해보기 (아침에 말하기)

0 여러 개의 추천질문 뽑는 것(tf-idf방식 , 그리고 이 방식에 대한 이해) , 매뉴얼 QA csv좀 자세히 봐보기

Try A. 원래의 질문 셋을 vectorDB(weviate, ChromaDB)등에 임베딩후 집어넣고, 사용자가 질문을 했을 때, retriever query 에 그 질문을 집어넣고 검색하면, 벡터 거리 기반으로 유사한 질문이 나올것같다.

Try B. word2vec, Tf-Idf 등의 벡터화 방식 사용해서 유사도 거리 구하기, 거리가 가까운순 sorting 하는 방식으로

Try C. LLM의 토크나이저,임베딩 사용

1다른방식의 임베딩, 유사도 구하는 방법알아보기,

2추천시스템 쪽 으로 공부해보기