

مبانی بازیابی اطلاعات و جستجوی وب

نیم سال دوم ۱۴۰۲-۱۴۰۳

تمرین ۲

اهداف: با اسناد خود چت کنید 😊

مهلت تحويل^۱:

جمعه ۱۸ خرداد ۱۴۰۳

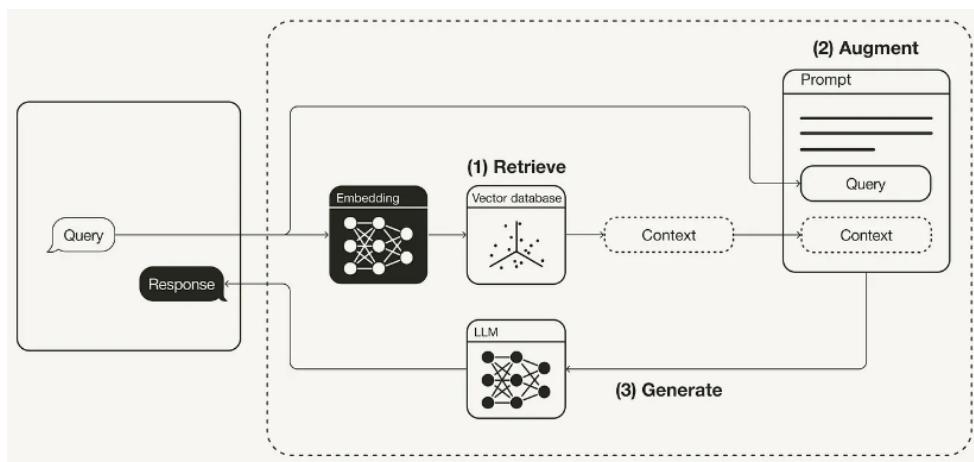
^۱ این تمرین به صورت شفاهی تحول گرفته می‌شود و آپلود کردن آن در LMS بدون تحويل شفاهی فاقد نمره می‌باشد. و زمان ارایه شفاهی بعد از بازه امتحانات خواهد بود.

توضیحات تمرین

الان می‌خوایم با کمک هم یک سیستمی درست کنیم که یه فایل PDF بهش بدی و ازش درباره اون فایل سوال بپرسی و اون بعثت جواب بدی! نکته‌ای که باید بهش توجه بکنیم این هست که باید همه قدم‌ها مون سنجیده باشه، مثلاً اگه ابزار X‌ای رو انتخاب می‌کنیم باید بتونیم از انتخاب‌مون دفاع کنیم و گرنه سرهم کردن چند تا مازول رو هر کسی که برنامه نویسی پیشرفت‌هه پاس کرده باشه بلده؛ لطفاً این موضوع رو تا پایان این تمرین در گوشه‌ای از ذهن‌مون داشته باشیم. کار رو در چند گام انجام می‌دیم:

گام اول

الان می‌خوایم درباره معماری کلی سیستم‌مون حرف بزنیم، این رو از من قبول کنید که توی سال ۲۰۲۴ بهترین روش برای چت کردن با اسناد، RAG هست، همونی که سر کلاس درباره اش صحبت کردیم.



ما می‌توانیم RAG رو با هر زبان برنامه‌نویسی‌ای بسازیم. می‌توانیم از framework های مختلفی هم استفاده کنیم مثل [RAGFlow](#), [LlamaIndex](#), [Langchain](#) با شماست ولی من پیشنهادم Langchain هست چون از بقیه شون محبوب‌تره و جواب نیازهای ما رو هم می‌ده و سرعت کار رو بیشتر می‌کنه.

گام دوم

می‌خوایم یک LLM انتخاب کنیم، خب اولین چیزی که به ذهن میرسه استفاده از ChatGPT هست ولی ازش استفاده نمی‌کنیم، نه که API-Key را نداشته باشیم‌ها، کلا دوست ندارم که ازش استفاده کنید. ☺️ از همین لحظه استفاده از هرگونه سرویسی که میزبانی شده باشه و برای دسترسی بهش به API-Key نیاز باشه رو من ممنوع اعلام می‌کنم. بجاش می‌خوایم از GPU‌ای که گوگل کولب در اختیار همه می‌گذاره استفاده می‌کنیم و روی اون خودمون یک LLM بالا

بیاریم. گوگل یک Nvidia Tesla T4 Tensor Core که 16 گیگ رم داره رو در اختیار همه میگذاره که برای بالا آوردن یک LLM کوچیک (با حدود 7 بیلیون پارامتر) کافیه. مثلًا [اینجا](#) می‌تونید ببینید که چطوری میشه این کار رو کرد.²

توی [اینجا](#) توضیح داده شده که یک LLM‌ای که به صورت لوکال بالا اومده رو چطوری میشه به langchain وصل کرد. اگه براتون سوال شد که کدوم مدل بهتره یه سر به [اینجا](#) بزنید. نکته اینه که یک مدل رو باید انتخاب کنیم که به اندازه کافی خوب باشه و توی 16 گیگ رم جا بشه. توی انتخاب مدل خیلی وسوس است به خرج ندید، سه چهار تا رو پیدا کنید که میشه توی گوگل کولب اجراشون کرد و از بین اونها یکی رو انتخاب کنید.

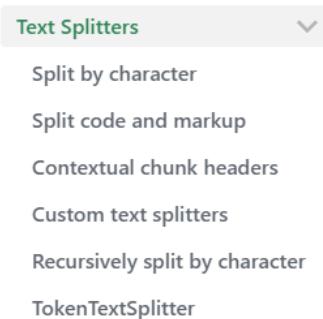
برای انتخاب مدل embedding هم یک سر به [اینجا](#) بزنید و یکی رو انتخاب کنید که هم بشه راحت ازش استفاده کرد و هم به اندازه کافی خوب باشه.

گام سوم

توی این گام باید داده رو برای بازیابی آماده کنیم. یعنی باید متن رو از PDF بخونیم، اون رو به قطعه‌های کوچیک‌تر تقسیم کنیم و اونها قطعه‌ها رو embed کنیم و در نهایت در یک پایگاه داده برداری مثل ChromaDB قرار بدیم. الزامی به استفاده از ChromaDB نیست ولی چون محبوب هست و نیازهای ما رو هم جواب می‌ده انتخابش معقول هست.

برای خوندن متن از PDF می‌تونیم از [Langchain document_loaders](#) استفاده کنیم و یا هر روشی که خودتون راحت‌تر هستید.

به مرحله سخت کار رسیدیم! باید اون PDF بزرگی که خوندیم رو به قطعه‌های کوچیک تقسیم کنیم، سوال اینه که چقدر کوچیک؟! اصلاً چطوری تقسیم کنیم؟! انتظار میره که به داکیومنت‌های Langchain یعنی [اینجا](#) مراجعه کنید و ببینید اصلاً چه کارهایی میشه کرد و یکی از اونها رو انتخاب کنید و انجام بدید. من شما رو به Langchain نمی‌کنم یعنی اگه ایده‌ای به ذهنتون زد که می‌خواستید اون رو اجرا کنید، مجازید. مواردی که توی عکس زیر اومدن رو بررسی کنید حتماً:



در نهایت هم قطعه‌های داده رو embed کرده و در ChromaDB می‌ریزیم که از اون جایی که سر کلاس انجام دادیم نباید مشکلی باشه.

² یه کم گوگل کنید همه چیز دستتون میاد. برای این کار آموزش زیاده.

گام چهارم

توی این مرحله باید RAG مون رو تکمیل کنیم اگه هنوز براتون واضح نیست که باید چیکار کنید پیشنهاد می‌کنم اینجا رو بخونید یا [این](#) رو نگاه کنید. در نهایت یکی از مقالات مورد علاقه‌تون رو به سیستمی که ساختید بدهید و ازش سوال بپرسید و ببینید که چطوری جواب می‌ده. اگه جواب‌هاش خوب نبود می‌توانید prompt‌تون رو بهتر کنید تا نتیجه بهتری بگیرید.

مقاله مورد علاقه من [این](#) هست و یک سوال خوب اینه:

"Can I use large language models for password modeling?"

گام پنجم (نمره مازاد)

این بخش را مطالعه کنید و از بین مواردی که در جدول اومده (به جز مورد اول) یکی رو که فکر می‌کنید باعث بهبود عملکرد RAG مون می‌شه انتخاب و به RAG مون اضافه کنید و بررسی کنید که آیا بهتر شد یا نه.

شیوه تحويل

خروجی موردنظر برای هر قسمت تمرین شامل بخش‌هایی است که توضیحات آن به شرح زیر می‌باشد.

در بخش گزارش هر قسمت، اسامی اعضای گروه و مشارکت هر عضو که عددی بین 1 تا 10 هست را بنویسید.

گزارش

در بخش گزارش کافی است دلایل خود را برای انتخاب موارد زیر شرح دهید:

- LLM •
- Embedding •
- مدل •
- روش تقسیم سند به قطعه‌های کوچک •
- Prompt •
- Retriever •
- (اختیاری - گام پنجم) •

کد

فایلی با فرمت *ipnby*. که شامل کدهای شما و خروجی‌ها هست (همونی که توی گوگل کولب نوشته‌یدش).

در نهایت فایل مربوط به این تمرین را به صورت [که شامل موارد خواسته شده بخش IR_1402-02_HW2_Group Number.zip](#) آپلود کنید.

لطفا همه اعضای تیم، فایل رو در [U7](#) آپلود کنند.

موفق باشید.