Asylum seeking RAG app

Problem introduction

Building a RAG solution which can make use of 187 pages of legal documents in German in order to accurately answer questions which some employees of the "Hamburger Asylbehörde" might have about the asylum seeking process. Besides accurately answering the inquiries, the model should also give sources of information.

Challenges

- to prevent encountering hallucinations and falsehoods claims; - to present sources for the given information;

Solutions

RAG - Retrieval Augmented Generation

Create a langchain solution which is performing the loading of the data from all the PDFs, then its division into one page documents, followed by the splitting of it with the help of the SpacyTextSplitter with the option for the small spacy pipeline.

These are then fed to a ChromaDB for embeddings and then as an LLM gpt-3.5-turbo is used and the prompt is the default "rlm/rag-prompt" from the langchain hub:

"You are an assistant for question-answering tasks. Use the following pieces of retrieved context to answer the question. If you don't know the answer, just say that you don't know. Use three sentences maximum and keep the answer concise.\nQuestion: {question} \nContext: {context} \nAnswer:" This is then turned into a streamlit app

Addressed questions

- Does the solution have an interface? If not: what should it look like? If yes: how would you improve it? Make a rough sketch.
- Describe how to evaluate the solution.
- How should the solution be deployed?
- Describe the future: What would you do next? What would you do if you had more time? What decisions have to be made?

Does the solution have an interface? If not: what should it look like? If yes: how would you improve it? Make a rough sketch

The solution comes with a streamlit solution.

It could be improved both:

- functionally:
    - non-AI related: by improving visually the frontend of the UI
    - AI related: to make it able to keep the history of the conversation and actually use it as context for future prompts withing the same conversation

- non-functionally: by ensuring scalability, availability and low latency at higher throughputs by doing some form of stress testing

Describe how to evaluate the solution

- the evaluation is of two natures:

  - firstly quantitative, for how many of the relevant questions it produces as an output and for how many it is actually saying something very similar semantically to "I don't know the answer"; - but then qualitative, where a human expert is needed to tell exactly if it's good or bad; - there is some other way to evaluate it with candidate questions, but that's described in the future works

Initial question no. 1

- Wann darf jemand in privaten Wohnraum ziehen?

Jemand darf in privaten Wohnraum ziehen, wenn er/sie erhebliche gesundheitliche Störungen hat, die eine Unterbringung in öffentlichen Unterkünften unmöglich machen, wenn besondere soziale Umstände vorliegen, für die eine Unterbringung in privatem Wohnraum unabdingbar ist, oder wenn mindestens eine Person in der Haushaltsgemeinschaft im Besitz eines Bleiberechts ist.

Initial question no. 2

- Hat man mit einer Duldung nach § 60a AufenthG Anspruch auf Leistungen nach dem AsylbLG?

Ja, Personen mit einer Duldung nach § 60a AufenthG haben Anspruch auf Leistungen nach dem AsylbLG.

Initial question no. 3

- Was bedeutet rechtsmissbräuchliche Beeinflussung der Aufenthaltsdauer i.S.d. § 2 AsylbLG?"

Rechtsmissbräuchliche Beeinflussung der Aufenthaltsdauer im Sinne des § 2 AsylbLG bezieht sich auf bewusstes, sozialwidriges Verhalten, das kausal für die Begründung oder Verlängerung des Aufenthalts ist. Dies kann beispielsweise durch die Zerstörung des Reisepasses oder die unterlassene Mitwirkung bei der Beschaffung des Reisepasses geschehen.

New question no. 1

- "Wann überquerte George Washington den Delaware River?" - totally unrelated to the datasources

Ich weiß es nicht.

New question no. 2

- "Wann wurde die Hamburger Asylbehörde gegründet?" - related, but might not be specified in the datasource

I don't know when the Hamburg Asylum Authority was founded. Small comment:

oddly enough it replied in English to a question posed in German

New question no. 3

- "Was können Sie mir über die Verordnung Nr. 156/2023 sagen, die am 16. Dezember 2023 verabschiedet wurde?" - it’s a trap intended to check whether the model will hallucinate or not

Ich weiß nicht, was die Verordnung Nr. 156/2023 besagt, da keine Informationen darüber in den gegebenen Kontexten enthalten sind.

How should the solution be deployed?

Since we have a streamlit app already, that could then be deployed as a Heroku app with the command streamlit deploy.

Describe the future: What would you do next? What would you do if you had more time? What decisions have to be made?

I would like to invest more into evaluation methods, as well as test out other types of embeddings, possibly play around with the text splitters as well and if going for the CharacterTextSplitter or the RecursiveCharacterTextSplitter to play around with the chunk_size/chunk_overlap parameters.

The decisions would have to be guided to choose a final solution based on the evaluation metrics.

Future improvements

- hyde could be used to generate a document and then embed it and then look for similarities within that generated document and the other datapoints; - we could also use a ContextualCompressionRetriever to extracts from each of the returned documents and select only relevant information to the posed question;
- of course using mistral could be of great help, there was not enough time for it, even though it was planned to use it

Other future improvements ideas

- generate questions from already existing text and then answer them and expect the model to actually answer it and maybe to generate from one page and expect the answer to be asked from that same page and not from some other page;
- take random questions from the internet, possibly questions which are already of potential to cause hallucination, maybe dataset of hallucinative questions, nonsensical questions;
- apply optical character recognition techniques for a document which has pictures with some schemas that describe some legal workflow processes for the asylum seekers;
- probably a bit extra, but possibly we could apply Visual Question Answering on those schemas afterwards;

Code

Instructions for how to run the code are provided in "SETUP.md" file and some details about the problem and use case in the "readme.md" file

Bibliography

https://www.datascienceengineer.com/blog/post-multiple-pdfs-with-gpt

https://www.youtube.com/watch?v=4sRigbRITF0&list=PL8motc6AQftn-X1HkaGG9KjmKtWImCKJS&index=10

https://python.langchain.com/docs/use_cases/question_answering/

https://medium.com/@meta_heuristic/dont-make-this-data-mess-mistakes-with-langchain-and-rag-a07f813c21e9

https://python.langchain.com/docs/modules/data_connection/vectorstores/

https://python.langchain.com/docs/modules/data_connection/

Thank you!

Questions?