

PROBLEM

Plenty of people have a folder of PDFs they'd love to ask questions of, and plenty of them work somewhere that makes cloud AI a non-starter: the documents are confidential, regulated, or not allowed off the premises. Every convenient "chat with your PDF" service is the wrong answer, since the price of convenience is uploading the sensitive thing. The gap is a tool that gives the same experience without the data leaving the room.

APPROACH

A standard RAG pipeline, built entirely from local pieces. PDFs are extracted with PyMuPDF, chunked and embedded with a Sentence-Transformers model into vectors held in NumPy. A question is embedded the same way, the nearest chunks are retrieved by similarity, and those chunks are handed to a quantised GGUF language model that writes an answer grounded in the retrieved context. It's a Flask app reached in a browser tab, frozen with PyInstaller into one executable.

OUTCOME

100%

offline · no API keys

1

.exe, double-click to run

0

bytes of data leave the machine

excelRAG lets a non-technical user on a locked-down machine ask their documents questions and get sourced answers without any of it touching the internet: real AI engineering and real shipping discipline, together in one file.

Python

Flask

Sentence-Transformers

PyMuPDF

ctransformers + GGUF

PyInstaller