Andy Banks – Lead Data Scientist
Colin Daglish – Senior Data Scientist
Iva Spakulova – Senior Data Scientist
Martin Wood – Data Scientist
Edward Jackson – Data Scientist
Pragya Paudyal – Data Scientist
Kate Milligan – Delivery Manager

Building an LLM powered search and summary engine using open-source tools

(and other bits)

26th July 2023

Data Science Campus

Agenda

- Approach – open-source tools
- Our experimental search engine
- Looking at the code
- Evaluating performance of an ML system
- Future possibilities
- Total tangent – my pet project

Why are we experimenting with open-source?

- The search part is a solved problem, **embedding models and semantic search**
- And we can plug paid-for services like OpenAI's offerings in later anyway
- Makes getting started very cheap!

Some + points for gov and science work

- Options for transferring a model to a secure environment
- We are given information about what data went into training
- The models are at fixed checkpoints for reproducibility
- Processing large amounts of data efficiently in back-end applications
- Can scale to appropriately larger or smaller models, saves compute/power/bills!

Hugging Face

(and here's the Hugging Face course on YouTube)

If you’re reading this slide, you’ve missed a demo

Approach

- Haystack (AI NLP framework) wrapped in Flask (web application framework)
- Have found LangChain equally capable

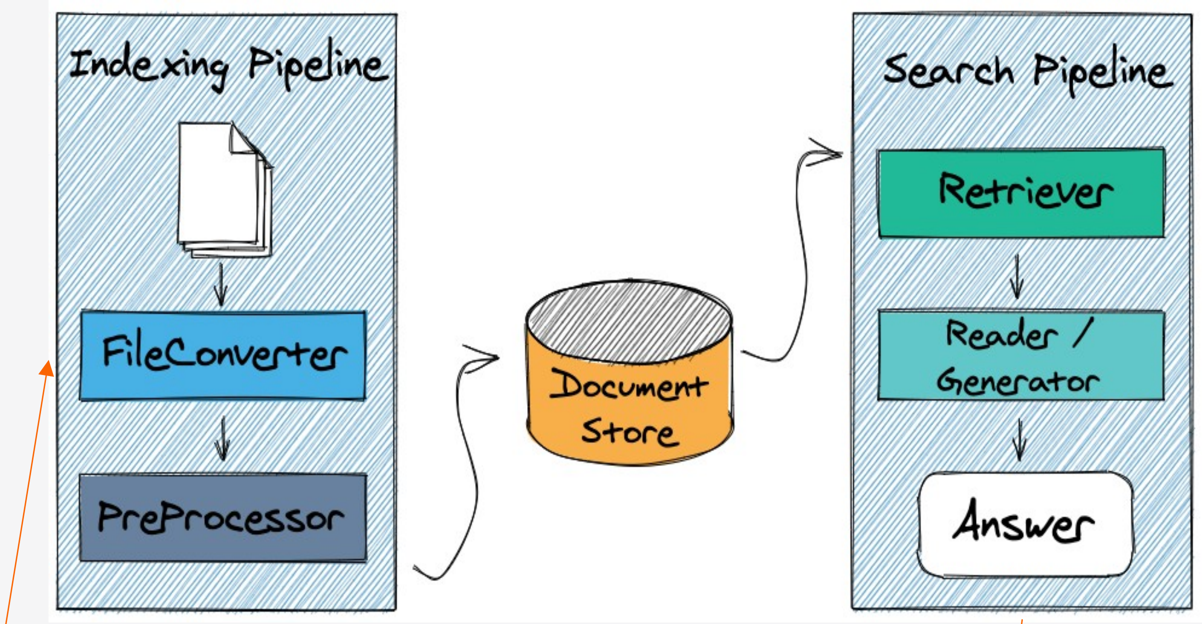
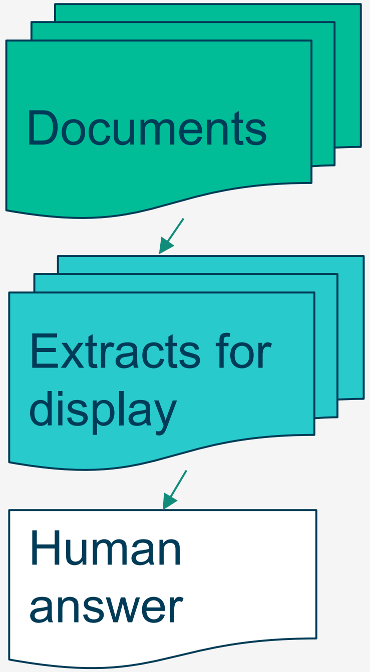


Image source: https://haystack.deepset.ai/overview/intro

BM25 + Embedding

RoBERTa

google/flan-T5-large (experimental)



Documents

Extracts for display

Human answer

Business checks/QA

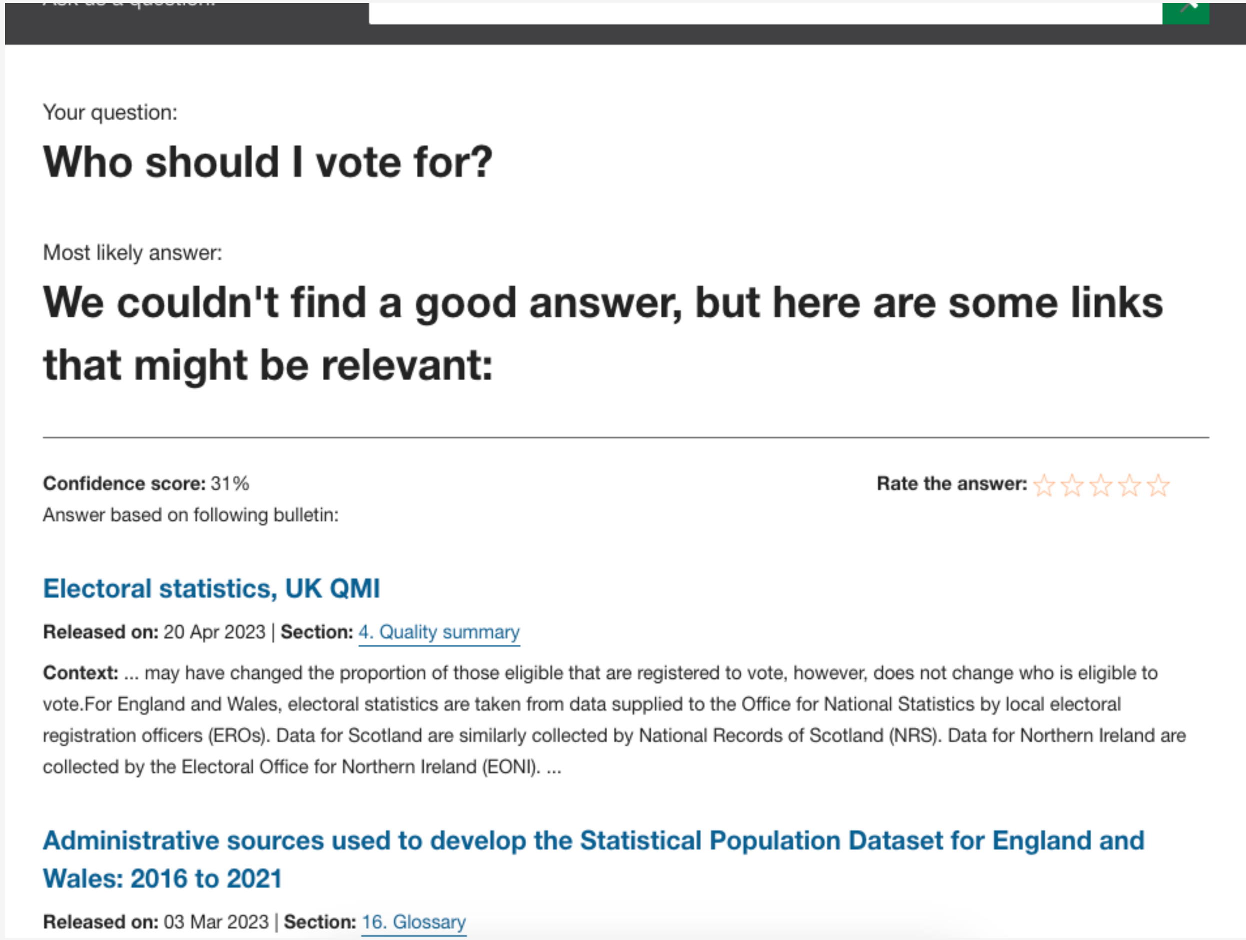
Gloss over that…

(and here's Haystack's intro tutorial)

Don’t forget to look at the actual code, Martin

Future possibilities

- Use a BIG model – falcon-40b-instruct
- Adapt the search engine for internal use, just a different corpora
- Expand from simple search to a chatbot with recommendation functionality and conversation memory, an AI assistant
- Other – integrate into the mining of our text data assets, digesting and understanding Ofsted reports



Your question:

Who should I vote for?

Most likely answer:

We couldn't find a good answer, but here are some links that might be relevant:

Confidence score: 31%

Rate the answer: ☆☆☆☆☆

Answer based on following bulletin:

Electoral statistics, UK QMI

Released on: 20 Apr 2023 | **Section:** 4. Quality summary

Context: ... may have changed the proportion of those eligible that are registered to vote, however, does not change who is eligible to vote.For England and Wales, electoral statistics are taken from data supplied to the Office for National Statistics by local electoral registration officers (EROs). Data for Scotland are similarly collected by National Records of Scotland (NRS). Data for Northern Ireland are collected by the Electoral Office for Northern Ireland (EONI). ...

Administrative sources used to develop the Statistical Population Dataset for England and Wales: 2016 to 2021

Released on: 03 Mar 2023 | **Section:** 16. Glossary

They can’t escape, time for your pet project

In case useful

- https://haystack.deepset.ai/
- https://flask.palletsprojects.com/en/2.3.x/
- https://python.langchain.com/docs/get_started/introduction.html
- https://huggingface.co/docs/transformers/index
- https://huggingface.co/tiiuae/falcon-40b-instruct
- https://huggingface.co/tiiuae/falcon-7b-instruct <- little version
- https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/prompt-engineering
- https://www.youtube.com/watch?v=00GKzGyWFEs&list=PLo2EIpI_JMQvWfQndUesu0nPBAtZ9gP1o <- Hugging Face Transformers course / lecture series
- https://towardsdatascience.com/testing-large-language-models-like-we-test-software-92745d28a359 ← blog on testing LLM outputs programatically, kudos to Sam Hollings of NHS England for finding it

Thank you!

```
# As a reward for listening, here’s some code for creating ML-generated images:


!pip install pytorch
!pip install diffusers

from diffusers import StableDiffusionPipeline


pipe = StableDiffusionPipeline.from_pretrained("runwayml/stable-diffusion-v1-5")
pipe = pipe.to("mps")  # Remove this line if NOT on an M1/M2 MacBook

# Recommended if your computer has < 64 GB of RAM (?!)
pipe.enable_attention_slicing()

prompt = "A robot head eating paper documents, against a background of more paper documents and computer wires."
image = pipe(prompt).images[0]

image.save("sd1.5_image.png")
```