



Red Bull | GenAI Challenge.

Owen Sims
20 March 2024

Red Bull Tasks.

01 | LLM Recommendations as a Corporate.

02 | HR Knowledge Chatbot Prototype.

03 | Technology & Innovation Landscape.

0A | Appendix.

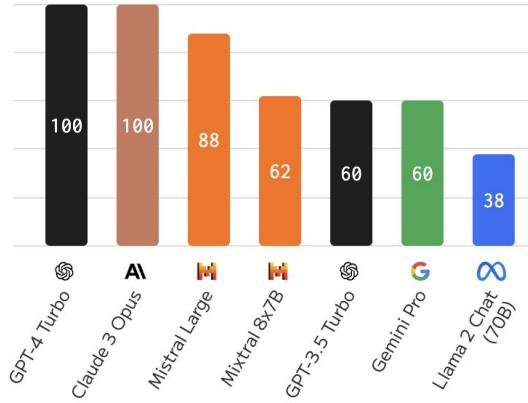


01 | LLM Recommendations as a Corporate.

Not All LLMs Are Created Equal...

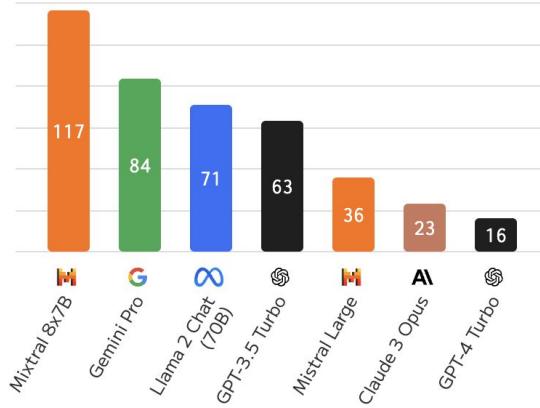
Accuracy.

Quality Index; Higher is better



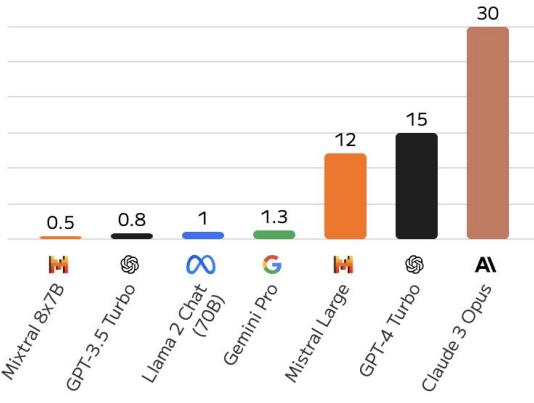
Performance.

Throughput in Tokens per Second

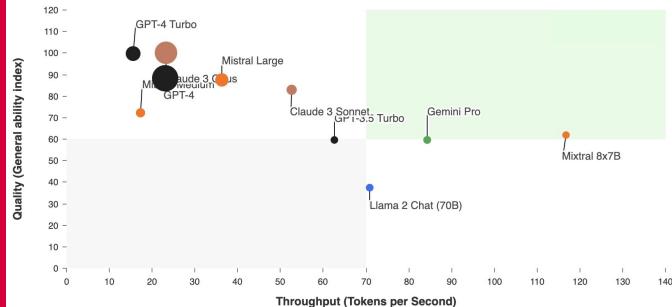


Cost.

USD per 1M Tokens



Accuracy vs Throughput



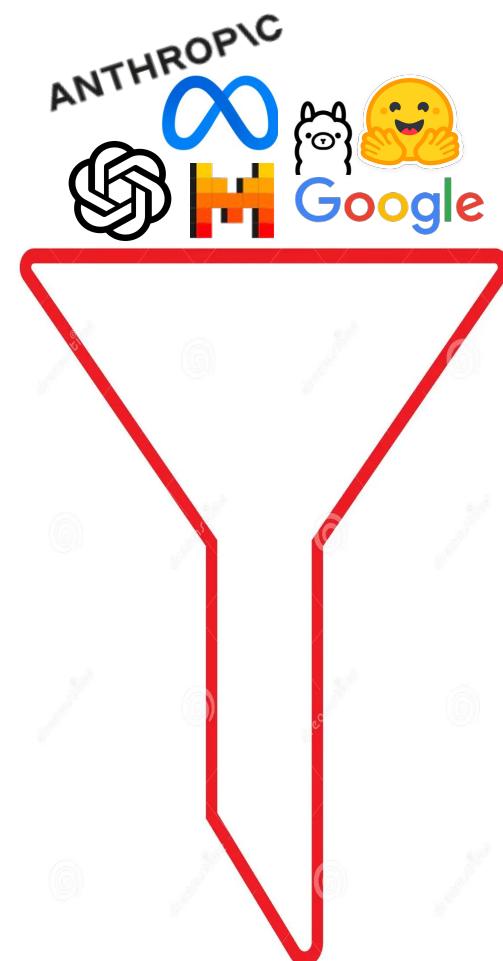
Quality vs Price



Filtering Factors When Selecting an LLM

Best way to rationalise LLMs is to think of the use case(s).

- **Commercial vs Non-Commercial Use.** The commercial use of LLMs automatically restricts the use of many Open LLMs.
- **Broad functionalities to perform.** Summarisation; Code Generation; Draft Documentation; Knowledge Management; Insights & Analytics.
- **Who are the users / audience?** Will define the “Evaluation Criteria” and inform the weighting that a decision-maker would place on the LLM trade-offs (speed, security, accuracy, etc.).
- **What do the trends look like?** Bigger is not always better.
- **Corporate Partnerships.** Depending on that partnerships that the Corporate has, it may be more efficient to
- **Cost.** Is cost a significant factor? Does it outweigh the potential efficiency gain?
- **Customisation.** Does the model require fine-tuning or customisation?
- **Development Community & Support.** Needs a good development community.
- **Where is the LLM application going to be hosted?** Data security can be vital in some use cases.



Ratifying LLMs

Stepwise Selection Framework to Pair LLM and Use Case.

Step	Example
1	Identify use case(s).
2	Construct a short-list of foundation LLMs available to potentially achieve the use case(s).
3	What size(s) of foundation LLMs are available?
4	Understand the Evaluation Criteria. The broad dimensions are important to the use case(s): Accuracy; Performance; Cost; or Reliability.
5	Test the models in a pairwise LLM Tournament.
6	Choose the option that provides the most value.
	Text Generation; Summarisation; Code Generation; etc.
	Depending on the use case and corporate partnerships, it may be that only some commercial models are available.
	Important to pick the right size model for the specific use case. Assess the Model Cards to understand if any models have been trained for the specific use case. Zero-shot.
	If customer-facing, performance / speed may be vital. If internal, accuracy and reliability may be more important.
	Compare the models output against the Evaluation Criteria. Potential to use a third LLM to score the others.
	Consider further aspects, such as Hosting, Ecosystem, Development Community...

Conclusion

The more diverse the use cases, it may be appropriate to use multiple LLMs.

- **Numerous Diverse Use Cases.** The larger the Corporate, the more processes and tasks in which to apply GenAI and LLMs.
- **Language & Localisation Needs.** Some LLMs are better with
- **Performing Comparative Analysis, LLM Competitions, and A/B Testing.** Having multiple LLMs allows organisations to perform well-defined comparisons.
- **Customisation and Fine-tuning Needs.** Some use cases require significant customisation and provide specific responses.
- **Cost Optimisation.** Ability to use smaller, more specialised models for certain tasks.
- **Innovation & Experimentation.** Comparing and testing between multiple LLMs.
- **Avoiding Vendor Lock-in.** With the LLM Landscape changing so fast, it's important to keep options open.

Recommendations...



OpenAI's GPT-4 Turbo. Large & general enough to perform well at most broad use cases.



Mixtral 8x 7B. Shows the most future potential by combining a Mixture of Experts (combination of NNs).



Llama 2. Open Source alternative for larger configuration and local / on prem hosting.

02 | HR Knowledge Chatbot Prototype.

The Problem Statement

You convert this epoch timestamp to a human-readable date and time: 1710794949

ChatGPT
The epoch timestamp 1710794949 corresponds to "Friday, November 10, 2023 02:35:49 AM"
(UTC).

Patrick Lucas Austin
@patbits · Follow
Can't wait til @LEVIS starts using AI models so I can just never buy anything from them again.

FEBRUARY 7, 2024 | 9 MIN READ

Even ChatGPT Says ChatGPT Is Racially Biased

When asked, ChatGPT declared that its training material—the language we humans use every day—was to blame for stories it generated

BY CRAIG PIERS

Bloomberg

Technology | AI

Samsung Bans Staff's AI Use After Spotting ChatGPT Data Leak

- Employees accidentally leaked sensitive data via ChatGPT
- Company preparing own internal artificial intelligence tools

Issues with Out-of-the-Box LLMs.

GPT's are pre-trained in a self-supervised / unsupervised way on a massive corpus of data.

This can lead to issues:

- Presenting **false information** when it does not have the answer.
- Presenting **out-of-date or generic information** when the user expects a specific, current response.
- Potential for **biases and hate speech**.
- Creating a response from **non-authoritative sources** (Doesn't know when to say "I don't know").
- Can **think too fast / too quick to provide responses**.
- Excessive use of **tokens**: Token limitations can restrict inputs and responses.
- Data leaks**.

An HR chatbot will have to deal with:

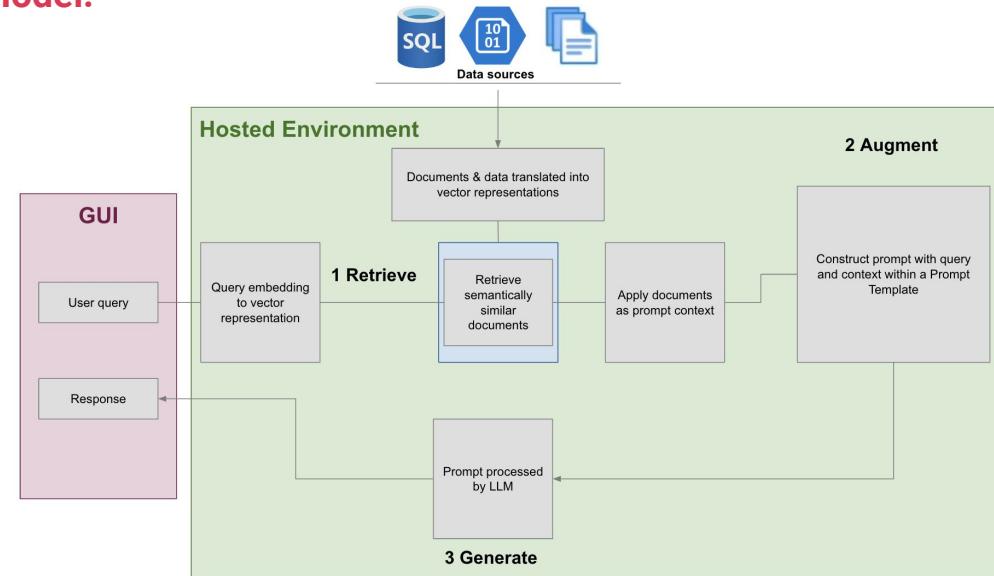
- Firm and location **specific information**.
- Sensitive private employee information**.

How Would You Approach The Topic?

The Methodology

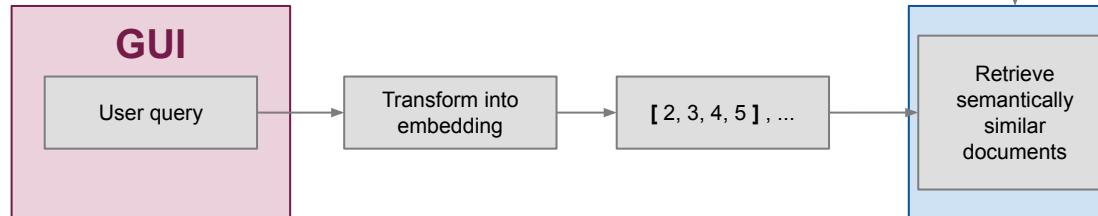
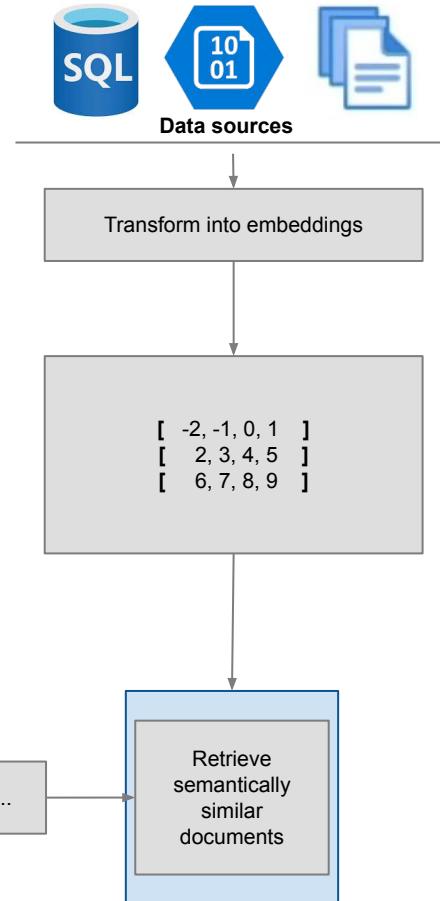
Retrieval Augmented Generation (i.e., RAG) Model.

- An architecture that augments the capabilities of an LLM, like GPT-4, by **adding an information retrieval system** that provides grounding data.
- Adding an information retrieval system **gives control over grounding data** used by an LLM when it formulates a response.
- In a RAG pattern, **queries and responses are coordinated between the search engine and the LLM**.
- A user's question or query is forwarded to both the search engine (**1 Retrieve**) and to the LLM as a prompt (**2 Augment**). The search results come back from the search engine and are redirected to an LLM (**3 Generate**).



Step 1: Retrieve (with Vectors!)

- In the style of a typical encoder model, **data sources are transformed from multimodal media (such as text, audio, video) into vector representations**.
- Vector representations highlight the **embedding** of each token of the vector which illustrates their **positional and relational aspects** with regards the whole data source.
- This encoding is stored in a vector database and allowed to be queried upon using algorithms, such as Approximate Nearest Neighbour (or, specifically, HNSW), that treat the data in the form of a **distance matrix**.
- The **users query is also vectorised**, embedded, and used as input to search the vector database.
- The closest results of the embedded documents are extracted from the vector database and passed on.



Example 1: Vectorising Prompt

```
import requests

from langchain.embeddings import OpenAIEmbeddings

OPENAI_API_CONFIG = {
    "api_secret": "123",
    "api_url": "https://u2zqklhkasoi01.openai.azure.com/",
    "api_url_version": "/openai/deployments?api-version=2022-12-01",
    "model_name": "gpt-35-turbo",
    "openai_api_version": "2023-05-15",
    "openai_api_type": "azure",
    "embedding_model_name": "text-embedding-ada-002",
}

def get_specific_model(deployments, model):
    for val in deployments['data']:
        if(val['model'] == model):
            return val['id']

deployment_response = requests.get(
    url=f'{OPENAI_API_CONFIG["api_url"]}{OPENAI_API_CONFIG["api_url_version"]}',
    headers={"api-key": OPENAI_API_CONFIG['api_secret']})

embeddings = OpenAIEmbeddings()
deployment=get_specific_model(
    deployments=deployment_response.json(),
    model=OPENAI_API_CONFIG['embedding_model_name']),
model=OPENAI_API_CONFIG['embedding_model_name'],
openai_api_base=OPENAI_API_CONFIG['api_url'],
openai_api_key=OPENAI_API_CONFIG['api_secret'],
openai_api_type="azure",
chunk_size=1)

prompt = input("Provide a prompt to vectorise: ")

print(""""
Query embedding:
{}
""".format(embeddings.embed_query(prompt)))
```

```
root@60ea9ac54e23:/app/backend/research# python 01_vector_embedding.py
Provide a prompt to vectorise: Hello!
('n'
'Query embedding: \n'
'\n'
'[0.0017646646274985171, -0.030391215730791038, 0.007377998770530714,
0.009331364016578756, -0.00095698323007284, -0.00621261483046525,
0.014706229526215349, 0.051459363666348595, 0.015801109089023053,
0.001340605992896588, -0.03372562243406023, -0.008875164043521786,
0.020885666456218678, -0.01507118969025437, -0.014034371570341415,
0.011263993365876397, -0.026061463631761185, -0.0043131638472908105,
-0.018612961880956548, 0.016531029418443227, 0.011977324046573692,
-0.0026044873051757, -0.00572326491285081, 0.01891156438209768,
0.018364125532016384, 0.010874150125065827, 0.004450024025472413,
0.013204917073874197, -0.016381729099195223, -0.00806644544380497,
0.020205514141644582, -0.01901109854914472, 0.009206945842108674,
0.013420575056691161, -0.0015562642085441925, 0.02199713659907182,
-0.028898197637150045, 0.00904934930151539, -0.0005197051702542308,
-0.018380714249416705, 0.02387170264350066, -0.020736364274325556,
-0.015212197514118397, -0.004690565550051139, -0.0022623373253788477,
-0.009522138923295238, -0.012201277319413376, 0.0075355948454627186,
0.006764202256880462, 0.013992898845518053, -0.006863736889588783,
0.014183673752234536, 0.021715120952685896, 0.059654372601328616,
0.0020725996791611636, 0.006478040362467016, 0.026957274860474802,
0.009306480940478274, 0.004124462995910646, -0.011977324046573692,
0.000666155700682894, 0.014125611378688297, 0.00569835294952332,
-0.001474355553734991, -0.009364542382701957, -0.003284640085406949,
0.019392648362577685, 0.006055018289871967, -0.01659738615068963,
0.013080498899404115, -0.005673469407761559, 0.02398782739059314,
-0.00029082752938994606, 0.0060011040269983655, -0.006693698345283981,
0.01916400731037842, -0.010384771785885656, -0.02372400461607538,
-0.00811621345865105, 0.0015884055469992332, -0.02151605261859181,
-0.020023034524950818, -0.01199391369529657, -0.00048445333087131015,
-0.012740422742117067, -0.016589091791989467, -0.016531029418443227,
0.004037370366913844, 0.012773601108240266, 0.005648585865999798,
-0.005690058590823159, 0.013287862523520919, 0.0017739960138668373,
-0.01711994192467044, -0.006768349436230542, -0.02340720924306609,
```

Example 2: Document Embedding into a VectorDB

```
texts = text_splitter.split_documents(documents)

deployment_response = get_deployment_response()

embeddings = OpenAIEmbeddings(
    deployment=get_specific_model(
        deployments=deployment_response.json(),
        model=OPENAI_API_CONFIG['embedding_model_name'],
        model=OPENAI_API_CONFIG['embedding_model_name'],
        openai_api_base=OPENAI_API_CONFIG['api_url'],
        openai_api_key=OPENAI_API_CONFIG['api_secret'],
        openai_api_type="azure",
        chunk_size=1)

if os.path.isdir(persist_directory):
    vectordb = Chroma(
        persist_directory=persist_directory,
        embedding_function=embeddings)
else:
    vectordb = Chroma.from_documents(
        texts, embeddings,
        persist_directory=persist_directory,
        collection_metadata={
            "hnsw:space": "cosine"
        })
e = vectordb._collection.get(
    include=['embeddings'])

pprint(e)
```



```
-0.02132401056587696,
0.007120268885046244,
0.009834066964685917,
-0.022372731938958168,
0.022041555494070053,
-0.010560812428593636,
-0.00641652150079608,
0.006756896153092384,
0.007865413092076778,
-0.007893010973930359,
0.02577647753059864,
-0.007423846982419491,
0.0012488068314269185,
0.0087894446247816,
0.0016593262553215027,
-0.008642755448818207,
-0.012777847237884998,
0.007640039700713396,
0.03028414398431778,
0.011572737246751785,
-0.008527764119207859,
-0.004282280802726746,
-0.02892264351248741,
0.004627254791557789,
-0.030652116984128952,
-0.016466772183775902,
0.00478242746144533,
0.011287557892501354,
0.010570012032985687,
0.025702882558107376],
'ids': ['5f3613b8-e5fa-11ee-8df9-0242ac130005',
'5f36161a-e5fa-11ee-8df9-0242ac130005',
'5f36169c-e5fa-11ee-8df9-0242ac130005',
'5f36170a-e5fa-11ee-8df9-0242ac130005',
'5f36176e-e5fa-11ee-8df9-0242ac130005',
'5f36176e-e5fa-11ee-8df9-0242ac130005',
'5f3617c8-e5fa-11ee-8df9-0242ac130005',
'5f36182c-e5fa-11ee-8df9-0242ac130005',
'5f36187c-e5fa-11ee-8df9-0242ac130005',
'5f3618d6-e5fa-11ee-8df9-0242ac130005',
'5f3618d8-e5fa-11ee-8df9-0242ac130005']
```

Example 3: Retrieving from a VectorDB

```
loader = TextLoader('./document/path')
documents = loader.load()

text_splitter = RecursiveCharacterTextSplitter(
    chunk_size=1000,
    chunk_overlap=0)
texts = text_splitter.split_documents(documents)

deployment_response = get_deployment_response()

embeddings = OpenAIEMBEDDINGS(
    deployment=get_specific_model(
        deployments=deployment_response.json(),
        model=OPENAI_API_CONFIG['embedding_model_name']),
    model=OPENAI_API_CONFIG['embedding_model_name'],
    openai_api_base=OPENAI_API_CONFIG['api_url'],
    openai_api_key=OPENAI_API_CONFIG['api_secret'],
    openai_api_type="azure",
    chunk_size=1)

if os.path.isdir(persist_directory):
    vectordb = Chroma(
        persist_directory=persist_directory,
        embedding_function=embeddings)
else:
    vectordb = Chroma.from_documents(
        texts, embeddings,
        persist_directory=persist_directory,
        collection_metadata={
            "hnsw:space": "cosine"
        })

query = input("Prompt: ")

docs = vectordb.similarity_search_with_score(query)
pprint(docs)
```

```
root@60ea9ac54e23:/app/backend/research# python 02_vectorstore_qa.py
Prompt: When should I submit my International Remote Working request?
[Document(page_content="If you already hold the required right to work for the destination please refer to Categories 1 and 2 under Where can I work?) or meet the immigration 'easement' route, you will need to submit your application for international remote working permission (i.e. a visa), you should submit a request at least 4 months in advance of the visa application process to complete. In both cases, please ensure that you indicate you will require to obtain business approvals before submitting your IRW application. Follow the correct approval process to utilise our international remote working policy, or do so would be subject to our standard disciplinary procedures (for example - working permission).", metadata={'source': './international_remote_working.txt'}),
  0.11261559752200256),
(Document(page_content="I have had to travel at a short notice for an emergency. Can the IRW requests be reviewed on a first come, first served basis? The IRW team aims to issue authorisations in time to match the request. Where an authorisation has not been issued by the request, you will be required to book paid or unpaid leave until such time as an approval is issued. The IRW team will state the period for which the international remote working authorisation is valid. Does my IRW request need to be as one block of three weeks, or can it be spread over several weeks, all of which make up a single trip? It is expected that an international remote working period will be for a continuous period of up to three whole weeks per firm's holiday year.", metadata={'source': './international_remote_working.txt'}),
  0.12153132919158605),
(Document(page_content="All other employees/partners planning to work in countries where the immigration support should submit a request at least 4 months in advance to allow right to work processing. Please see the 'Where can I work?' section. The UK government issues a visa/right to work permission as this will depend on several factors outlined. You are advised not to make any travel arrangements until an international travel document has been issued by the IRW team. Can I make a backdated request? An approval to work from overseas will only be valid once it has been granted by the IRW team.", metadata={'source': './international_remote_working.txt'}),
  0.12171278999830593),
```

Step 2: Augment

- When data is retrieved from the data store, it is **decoded from a vector into text** and applied to a **Prompt Template**.
- Prompt Engineering** is a methodology that may be applied to construct the Prompt Template.
- An example Prompt Template may look something like 
- The Template will **combine both the query that the user has injected and the output that has been extracted from the data store** along with the context that has been engineered.
- This forms the basis of the **Prompt**.

Example Prompt Template

"""

You are a Human Resources (HR) assistant that helps company employees with questions about the employee handbook, which can cover topics such as International Remote Working (IRW), Maternity and Paternity Policy, and Holiday Entitlement. Be brief in your answers.

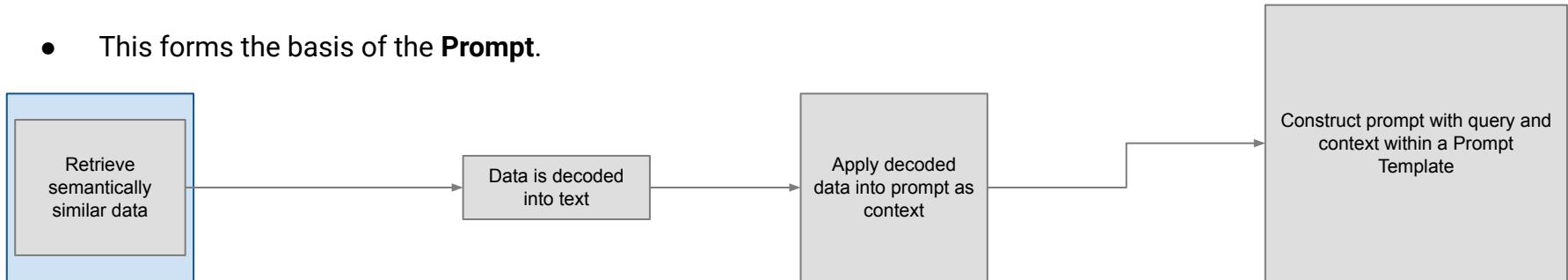
Answer ONLY with the facts listed in the list of sources below. If there isn't enough information below, say you don't know. Do not generate answers that don't use the sources below. If asking a clarifying question to the user would help, ask the question.

If the question is not in English, answer in the language used in the question.

Each source has a name followed by colon and the actual information, always include the source name for each fact you use in the response.

```
{follow_up_questions_prompt}  
(injected_prompt)
```

"""



Example 4: Augmenting a Prompt Template

```
from langchain import PromptTemplate
from langchain.vectorstores import Chroma
from langchain.embeddings import OpenAIEmbeddings
from langchain.text_splitter import RecursiveCharacterTextSplitter
from langchain.chat_models import AzureChatOpenAI
from langchain.chains import ConversationalRetrievalChain
from langchain.document_loaders import TextLoader

PROMPT_TEMPLATE = """
You are a Human Resources (HR) assistant that helps company employees with questions about the employee handbook, which can cover topics such as International Remote Working (IRW), Maternity and Paternity Policy, and Holiday Entitlement. Be brief in your answers.
Answer ONLY with the facts listed in the context below. If there isn't enough information below, say you don't know. Do not generate answers that don't use the sources below. If asking a clarifying question to the user would help, ask the question.
If the question is not in English, answer in the language used in the question.
Each source has a name followed by colon and the actual information, always include the source name for each fact you use in the response.

Question: {injected_prompt}

"""

condense_question_prompt = PromptTemplate.from_template(
    template=PROMPT_TEMPLATE)

chain = ConversationalRetrievalChain.from_llm(
    llm=llm,
    retriever=vectorstore.as_retriever(),
    condense_question_prompt=condense_question_prompt,
    return_source_documents=True,
    verbose=True)

query = input("Prompt: ")

llm_result = chain({
    "question": "{}".format(
        query),
    "chat_history": []})

print("""
""".format(llm_result["answer"]))
```



```
Prompt: When Should I submit my International Remote Working request?

> Entering new StuffDocumentsChain chain...

> Entering new LLMChain chain...
Prompt after formatting:
System: Use the following pieces of context to answer the users question.
If you don't know the answer, just say that you don't know, don't try to make up an answer.
_____
If you already hold the required right to work for the destination country (for clarification please refer to Categories 1 and 2 under Where can I work?) or meet the immigration criteria under the 'easement' route, you will need to submit your application for international remote working a minimum of 3 weeks before you intend to travel overseas. If you require the firm's support with obtaining a relevant right to work permission (i.e. a visa), you should submit a request at least 4 months in advance to allow for the necessary visa application process to complete. In both cases, please ensure that you incorporate additional time you will require to obtain business approvals before submitting your IRW application.

It is critical to follow the correct approval process to utilise our international remote working policy, anyone found failing to do so would be subject to our standard disciplinary procedures (for example - working outside the UK without permission).

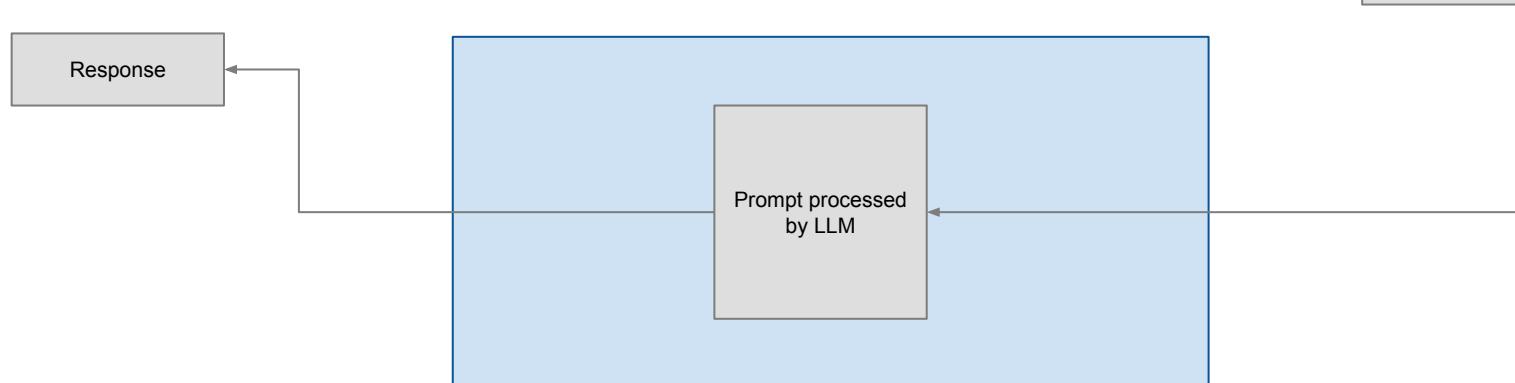
All other employees/partners planning to work in countries where the firm can provide the immigration support should submit a request at least 4 months in advance to allow for the necessary visa/right to work processing. Please see the 'Where can I work?' section.

The UK firm cannot guarantee an issuance of a visa/right to work permission as this will depend on several factors outside of the firm's control.
```

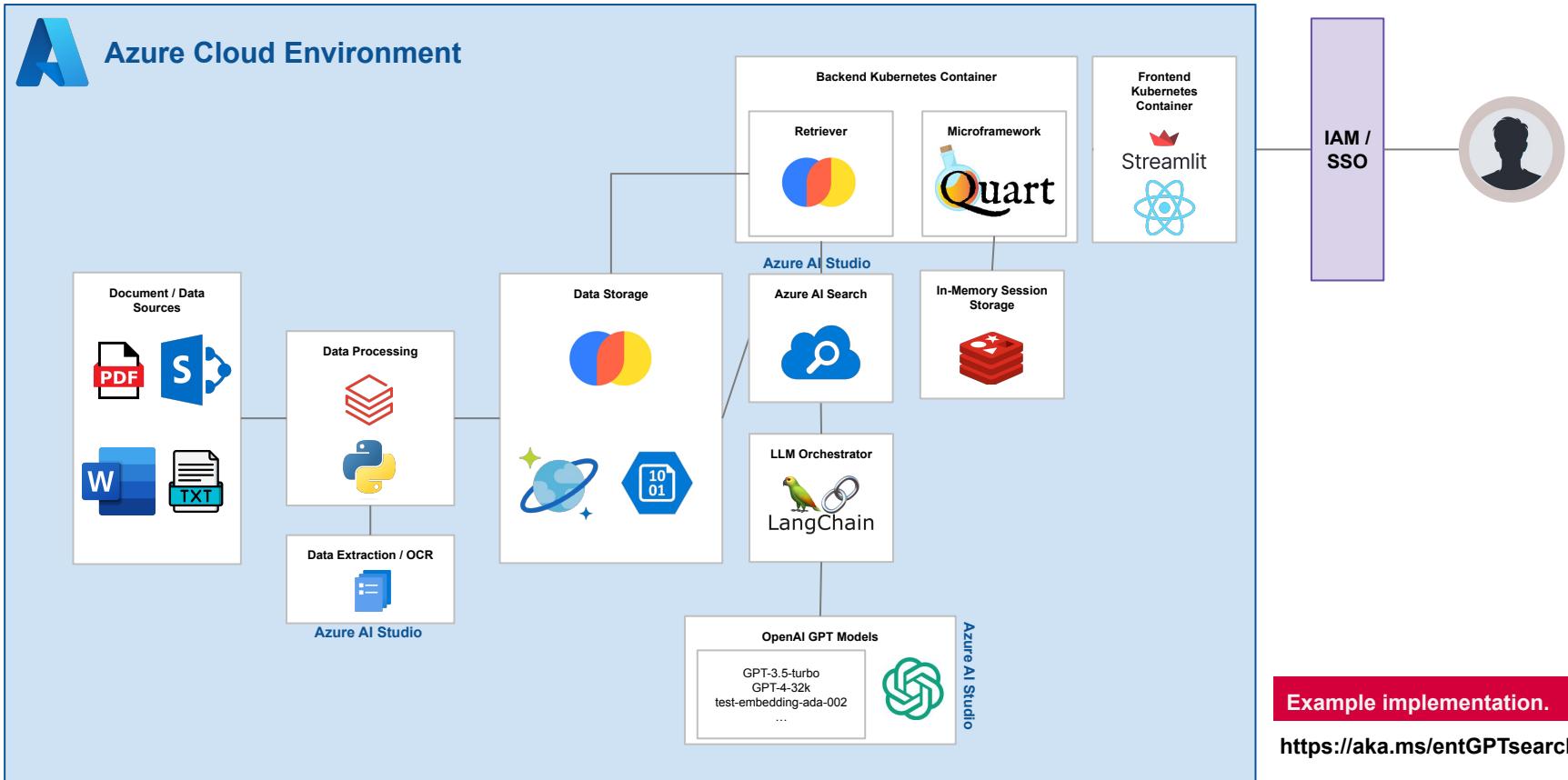
Step 3: Generate

- When the Prompt is constructed it is sent to the LLM (e.g., GPT-4).
- As part of this process, the **Prompt is encoded and tokenised**.
- The resulting token vector is analysed in a bidirectional manner that utilises a **self-attention mechanism** to extract the Input **Embedding and Positional Encoding**.
- This is then passed on to a **feedforward network**...
- The data is decoded and a **Response** is generated.

Construct prompt with query and context within a Prompt Template



Proposed Application Architecture



Putting it all together...



| Generic GPT Chatbot Architecture

You
Hey Harry! What can you tell me about maternity leave?

Harry
 Employees are entitled to up to 52 weeks of maternity leave, comprising of 26 weeks of ordinary maternity leave and 26 weeks of additional maternity leave. Statutory maternity pay (SMP) is available to employees who have been employed by Globex UK for at least 26 weeks by the Qualifying Week (15th week before the baby is due) and meet certain criteria. Globex maternity pay is also available to eligible employees. Employees are entitled to take reasonable time off work to attend antenatal appointments and may be eligible for additional paid leave if their baby is born prematurely and requires hospital stay. Keeping in touch (krt) days are available for employees to work during their maternity leave without bringing their leave to an end. If a Career Coach/Talent Coach/Relationship Leader is going on maternity leave, outstanding tasks should be completed and a change request form should be submitted to request a change in employee.

Citation
[Maternity Policy](#)

You
what about how long I get for paternity leave?

Harry
 For babies due, born placed 1 July 2023, paternity leave can be taken in a block of four whole weeks or one block of either one or two weeks' leave. Paternity leave must be taken within eight weeks of birth or adoption and must be taken in whole weeks – it can't be taken as one-off days or taken in one block.

Citation
[Paternity Policy](#)

Ask Harry a question here...  

Example implementation.

<http://localhost:9091/home/>

Overview of the RAG Approach

Benefits.

Primary

- **Cost-effective** implementation
- **Current** information
- Reliability / Enhanced user **trust**
- More **developer control**
- Knows when to say “I don’t know!”

Secondary

- Data Access Rights & Policies
- Citations and references
- Checking for hallucinations
- Removal of stale data / adding new data
- Forgetting Data / Privacy / Copyright issues
- Audit Trails
- Versioning / testing & deploying AI systems

Drawbacks.

- Firm data needs to be collected and stored.
- Needs to be kept up-to-date.
- Can be costly to host and maintain.

What are your findings?

On the technology side.

- For internal processes; the Accuracy and Reliability of response from the HR Chatbot is more important than speed.
 - Ensure that there is no scope for hallucination through hyper-parameterisation;
 - Storing an audit trail is important;
 - Measuring Prompt Drift - How effectively is the chatbot being used;
 - What
 - Issues raised with false information
- The LLM needs to be hosted.
- The accessibility of data.

On the human side.

- **Trust** is difficult to build and can be eroded quickly:
 - Providing citations to relevant documents associated with the query and response is good practice in order to build trust;
- Queries are typically specific and can be personal:
 - **Multi-tenancy** is key so that chat histories and information provided by the user are not leaked to other users/

03 | Technology & Innovation Landscape.

1 Quantum (Machine Learning)

Particularly applied to feature extraction, optimisation and classification problems.

Optimisation of Supply Chain & Distribution Operations.

- **Distribution & Route Optimisation.** Efficiently compute optimal routes for transportation vehicles, considering factors such as traffic patterns, delivery schedules, vehicle capacities, and fuel consumption. By minimizing travel distances and maximizing resource utilization, quantum computing can reduce transportation costs and improve delivery efficiency.
- **Environmental Sustainability.** Optimise environmental factors such as carbon emissions, energy consumption, and waste generation. By incorporating sustainability objectives into decision-making processes, quantum algorithms can identify opportunities to minimise environmental impact while maintaining supply chain efficiency.

Medical Imaging and Diagnostics.

- **Accuracy and speed of medical imaging techniques.** Quantum algorithms can enhance the use of MRI, CT, and PET scans. Quantum computers can process and analyse imaging data more efficiently, leading to faster diagnoses and improved treatment planning.

2 Wearable Technology

Particularly applied to healthcare and wellness.

Utilisation of AR & VR.

Treating debilitating conditions, such as Alzheimer's.

Wellness, Sports & Lifestyle.

A | Appendix.



Hosted Environment

GUI

User query

Response

1 Retrieve

Documents & data translated into vector representations

Query embedding to vector representation

Retrieve semantically similar documents

2 Augment

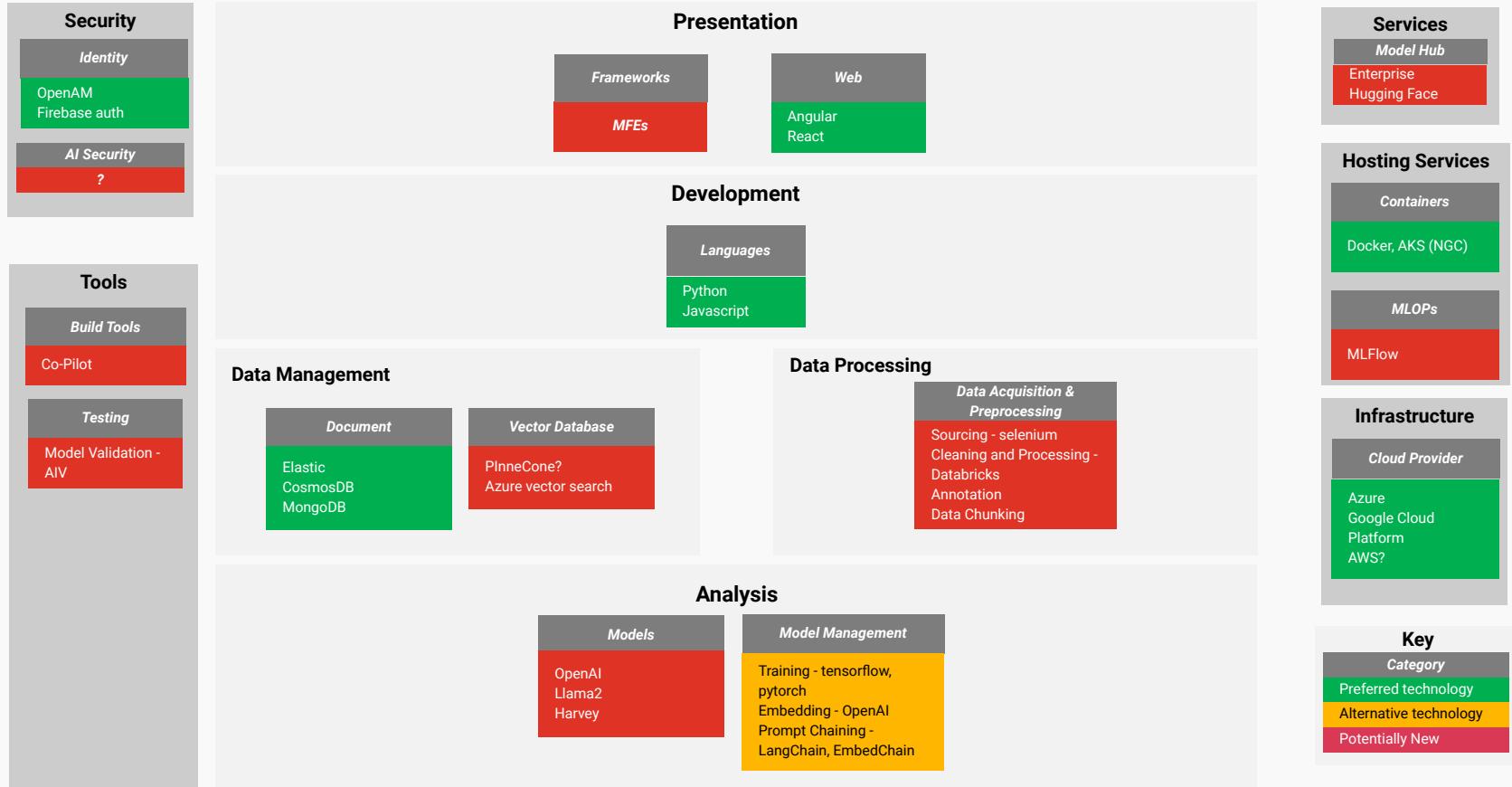
Apply documents as prompt context

Construct prompt with query and context within a Prompt Template

Prompt processed by LLM

3 Generate

Technologies | Evolving GenAI Golden Stack



Technologies | Generic Application Architecture

