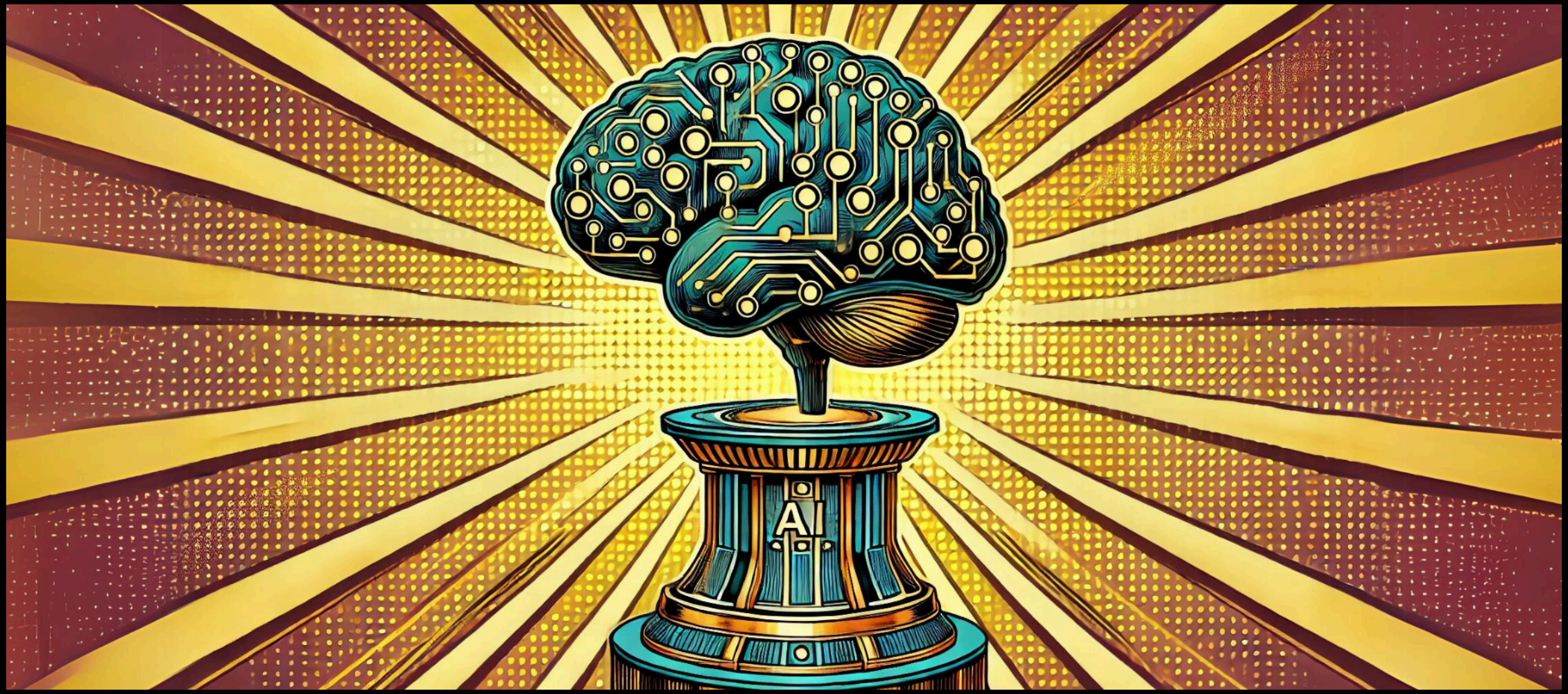
LLM Engineering

MASTER AI & LARGE LANGUAGE MODELS



Week 5 Day 5

Completing your RAG training

What you can now do

- Generate text and code with Frontier Models including AI Assistants with Tools, and with open-source models with HuggingFace transformers
- Confidently choose the right LLM for your project, backed by metrics
- Create a RAG Knowledge Worker using LangChain and Chroma

By end of today you will also be able to:

- Be familiar with LangChain's declarative language LCEL
- Understand how LangChain works behind the scenes
- Debug and fix common issues with RAG
- BUT Wait... there's more!

LangChain Expression Language

LCEL is a declarative language that can be used as an alternative to the code approach

Describe what you want to achieve in a YAML file

Arguably not much easier than coding directly

```yaml
variables:
  - name: MODEL
  - name: TEMPERATURE
    default: 0.7
  - name: PERSIST_DIRECTORY
    default: 'vector_db'

components:
  - name: OpenAI_LLM
    type: ChatOpenAI
    parameters:
      temperature: ${TEMPERATURE}
      model_name: ${MODEL}

  - name: ConversationMemory
    type: ConversationBufferMemory
    parameters:
      memory_key: chat_history
      return_messages: true

  - name: OpenAIEmbeddings
    type: OpenAIEmbeddings

  - name: ChromaVectorStore
    type: Chroma
    parameters:
      documents: ${chunks}
      embedding: ${OpenAIEmbeddings}
      persist_directory: ${PERSIST_DIRECTORY}

  - name: VectorStoreRetriever
    type: VectorStoreRetriever
    parameters:
      vectorstore: ${ChromaVectorStore}
      search_kwargs:
        k: 20

  - name: ConversationalChain
    type: ConversationalRetrievalChain
    parameters:
      llm: ${OpenAI_LLM}
      retriever: ${VectorStoreRetriever}
      memory: ${ConversationMemory}

output:
  - name: conversation_chain
    from: ${ConversationalChain}
```

Behind the curtain

Understanding how LangChain works, and identifying & fixing common problems

Using Callbacks

To output prompt details

Diagnosing a common problem

The right chunks are not being provided

Fixing the problem

Chunking differently; providing more chunks

Demystifying LangChain

It's actually not hard to build RAG directly

WEEK 4 CHALLENGES

Major challenge for you - your own private Knowledge Worker

Create a Knowledge Worker on your information to boost productivity

- Assemble all your files in 1 place; your personal Knowledge Base
- Vectorize everything in Chroma - your vector datastore
- Build a Conversational AI and ask questions!

Advanced ideas to take it to the next level

- If you use Google Workspace, use Google's API to read your own docs
- If you use MS Office, use libraries to read Office docs
- Harder - use libraries to connect to your email inbox, and Slack, and more!

WEEK 5 COMPLETE!

62.5% To LLM Mastery

What you can now do

- Generate text and code with Frontier Models including AI Assistants with Tools, and with open-source models with HuggingFace transformers
- Confidently choose the right LLM for your project, backed by metrics
- Create advanced RAG solutions with LangChain

Next week we introduce the major commercial project! You'll be able to:

- Download a Dataset from the HuggingFace hub
- Examine and curate a dataset
- Identify evaluation criteria for judging success