

# The DataHour on

## Demystifying HuggingFace, LangChain, and LlamaIndex

- by Tarek A. Atwan

# Table of contents

- What is Hugging Face
  - Demo: Exploring Hugging Face
  - Demo: Hugging Face API
- What is LangChain
  - Demo: LangChain reading a PDF file
  - Demo: LangChain to Query a Database/Pandas
  - Demo: LangFlow
- What is LlamaIndex
  - Demo: LlamaIndex to Query a Database/Pandas
- Q&A

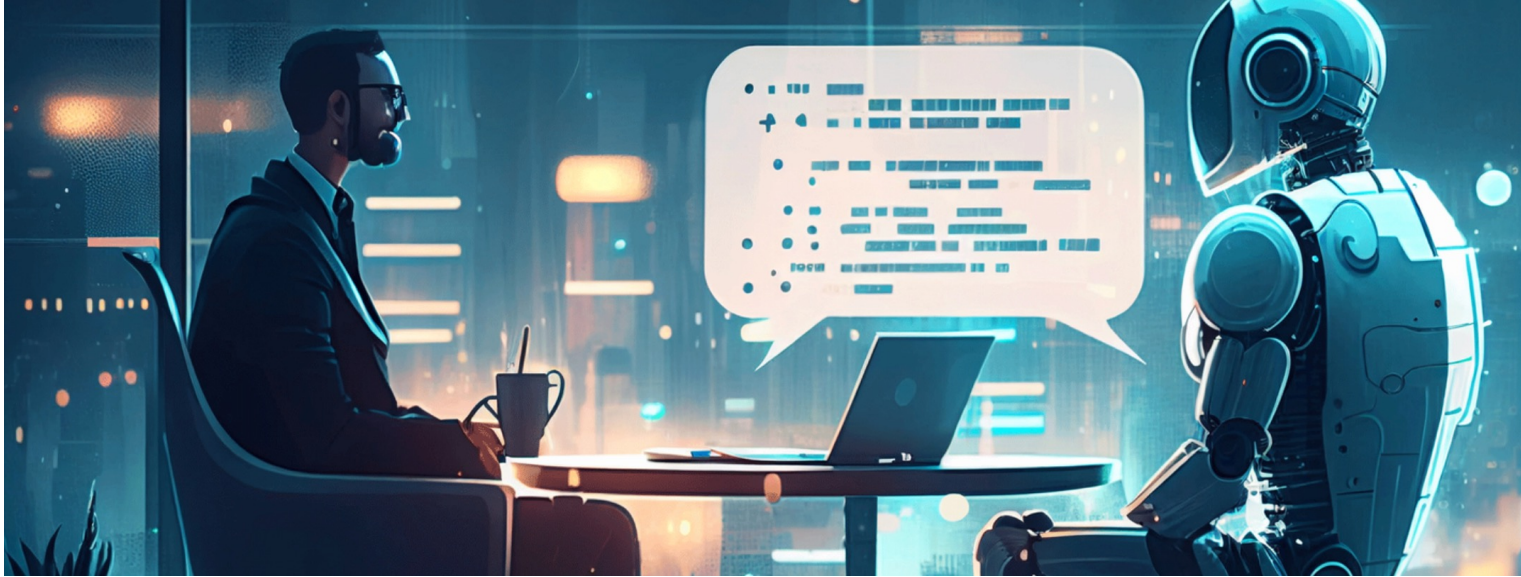
# Brief about me..



- My name is Tarek Atwan
- AI/ML Consultant
- ~17 years of work experience in Analytics (NAM & EMEA)
- ~5 years of teaching & mentoring (2U, Pluralsight, Thinkful)
- Book Author, "Time Series Analysis with Python Cookbook" by Packt Publishing



# The era of ChatGPT



# Hugging Face

# What is Hugging Face

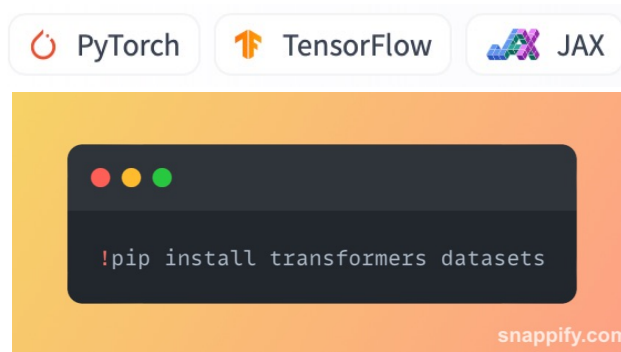
- Think of it as **GitHub** but for AI/ML folks
- Committed to the Open Source Community
- Democratize AI/ML to the masses
- Started off with its Transformers library
- Now, Hugging Face consists of many components:
  - Transformers Library
  - Tokenizers Library
  - HuggingChat
  - Hugging Face Hub (Git Repositories)
    - Datasets
    - Models
    - Spaces (ML apps made by the community)
  - Learn (Courses on NLP and Deep RL)



**Hugging Face**

# Hugging Face Getting Started

- Supports:
- Easy install:



- Easy to use:



# Hugging Face Task

Task	Description	Modality	Pipeline identifier
Text classification	assign a label to a given sequence of text	NLP	pipeline(task="sentiment-analysis")
Text generation	generate text given a prompt	NLP	pipeline(task="text-generation")
Summarization	generate a summary of a sequence of text or document	NLP	pipeline(task="summarization")
Image classification	assign a label to an image	Computer vision	pipeline(task="image-classification")
Image segmentation	assign a label to each individual pixel of an image (supports semantic, panoptic, and instance segmentation)	Computer vision	pipeline(task="image-segmentation")
Object detection	predict the bounding boxes and classes of objects in an image	Computer vision	pipeline(task="object-detection")
Audio classification	assign a label to some audio data	Audio	pipeline(task="audio-classification")
Automatic speech recognition	transcribe speech into text	Audio	pipeline(task="automatic-speech-recognition")
Visual question answering	answer a question about the image, given an image and a question	Multimodal	pipeline(task="vqa")
Document question answering	answer a question about the document, given a document and a question	Multimodal	pipeline(task="document-question-answering")
Image captioning	generate a caption for a given image	Multimodal	pipeline(task="image-to-text")



## Hugging Face



# Exploring Hugging Face & Coding Demo

# LangChain

# What is LangChain

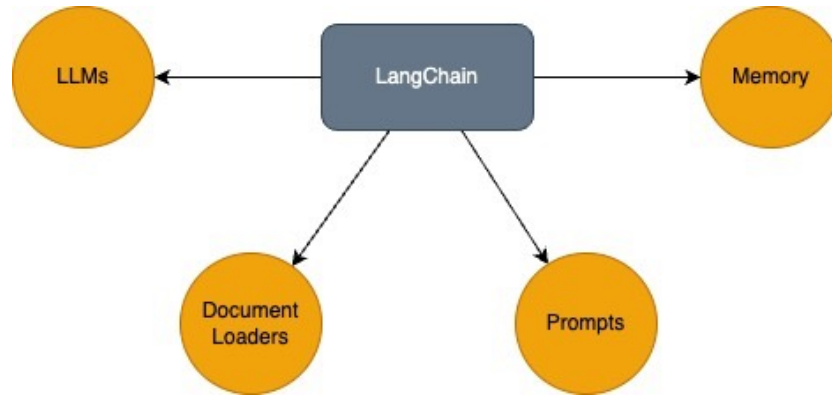


- It is a **framework**
- LangChain wants to simplify the development of applications powered by language models
- Supports Prompt Templates
- Makes it easy to swap different models (LLMs)
- Provides chains for creating a pipeline
- Makes it easy to query datasets with natural language in variety of formats: CSV files, SQL Databases, or pandas DataFrames

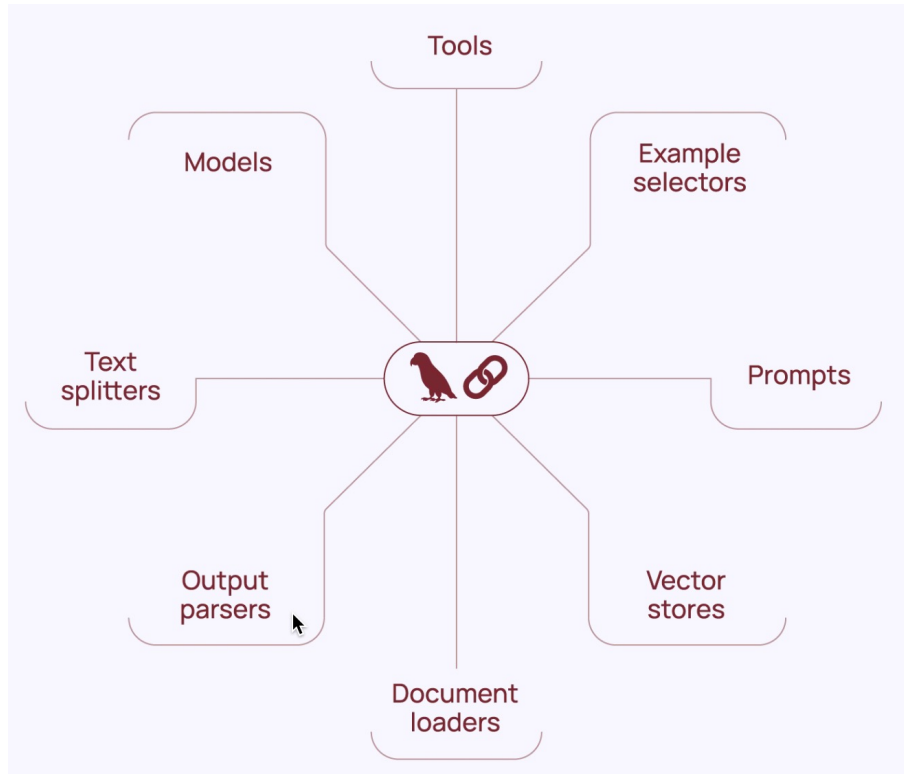
# What is LangChain



# What is LangChain



# What is LangChain



# What is RAG

“Retrieval-Augmented Generation (RAG) is the process of optimizing the output of a large language model, so it references an authoritative knowledge base outside of its training data sources before generating a response”

“ RAG extends the already powerful capabilities of LLMs to specific domains or an organization's internal knowledge base, all without the need to retrain the model. It is a cost-effective approach to improving LLM output so it remains relevant, accurate, and useful in various context”

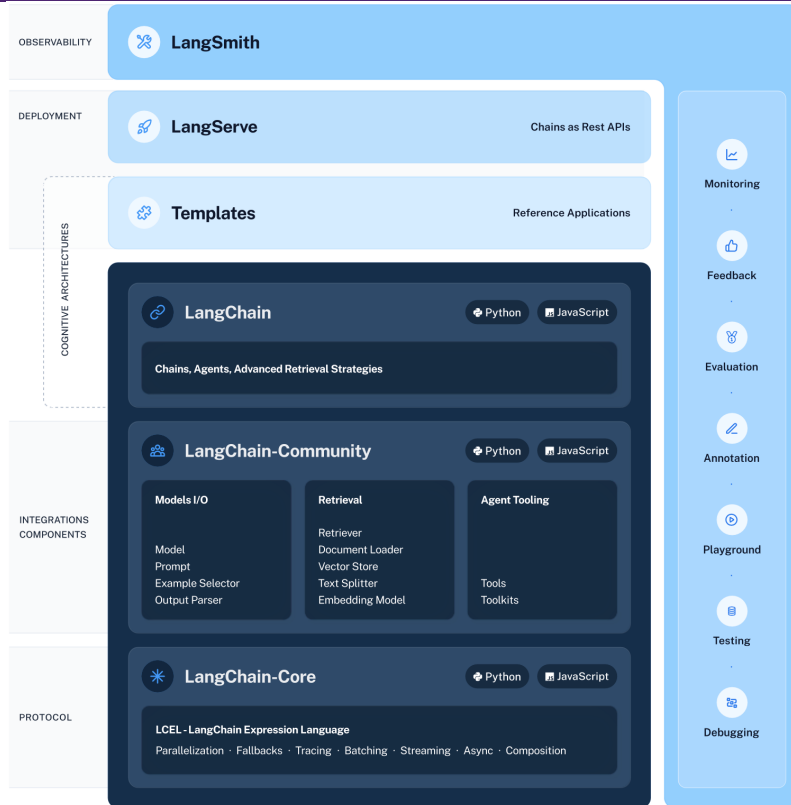
Source



# LangChain Components



- LangChain Libraries
  - LangChain
  - LangChain-Core
  - LangChain-Community
- LangChain Templates
- LangServe
- LangSmith
- LCEL (LangChain Expression Language)





# LangChain Getting Started



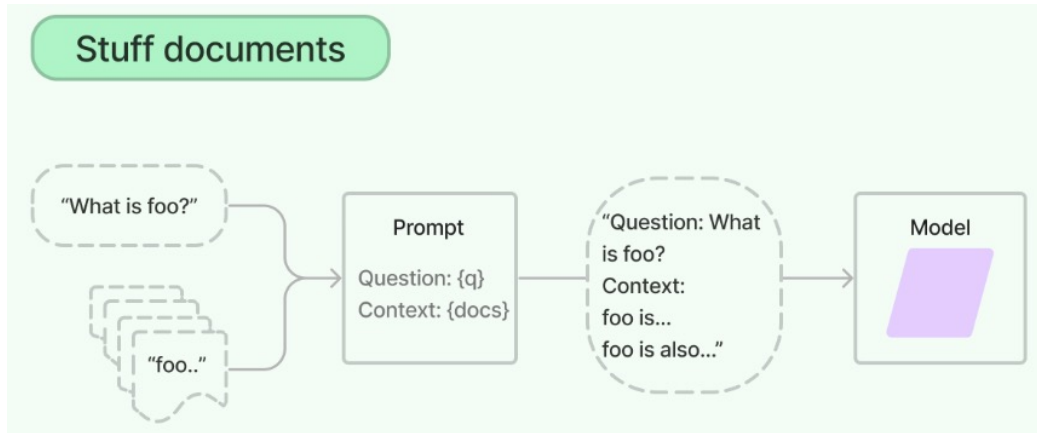
- Easy install:

```
!pip install langchain
!pip install langchain-community
!pip install langchain-experimental
```

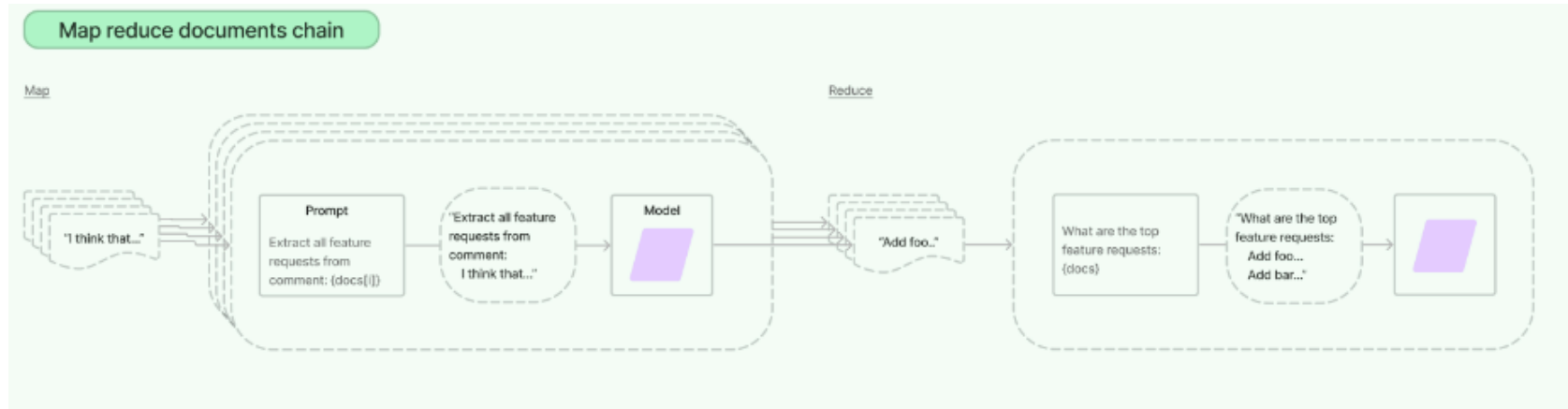
- Document Loaders

```
from langchain_community.document_loaders.csv_loader import CSVLoader
from langchain_community.document_loaders import UnstructuredPDFLoader, PyPDFLoader
```

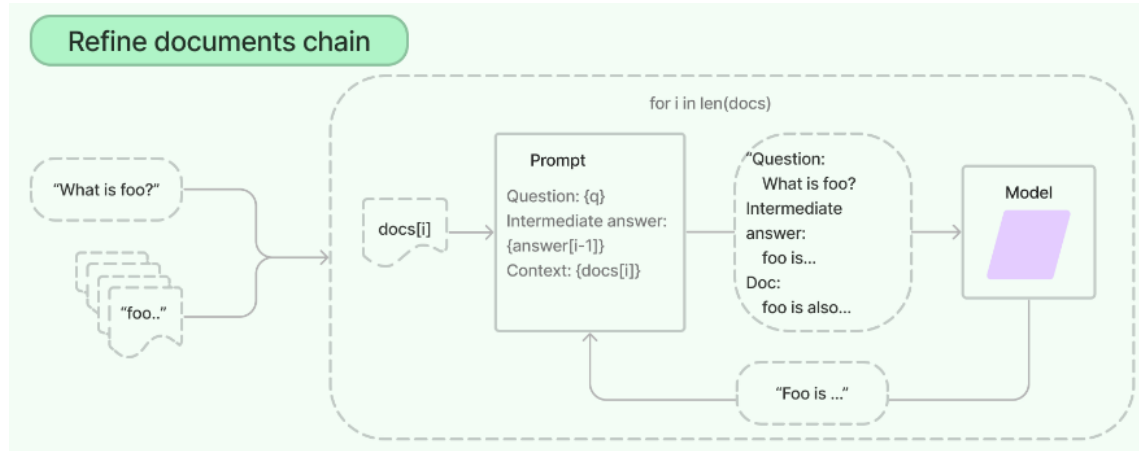
# Document Chains - Stuff



# Document Chains - MapReduce



# Document Chains - Refine



# LangChain Coding Demo

# LlamaIndex

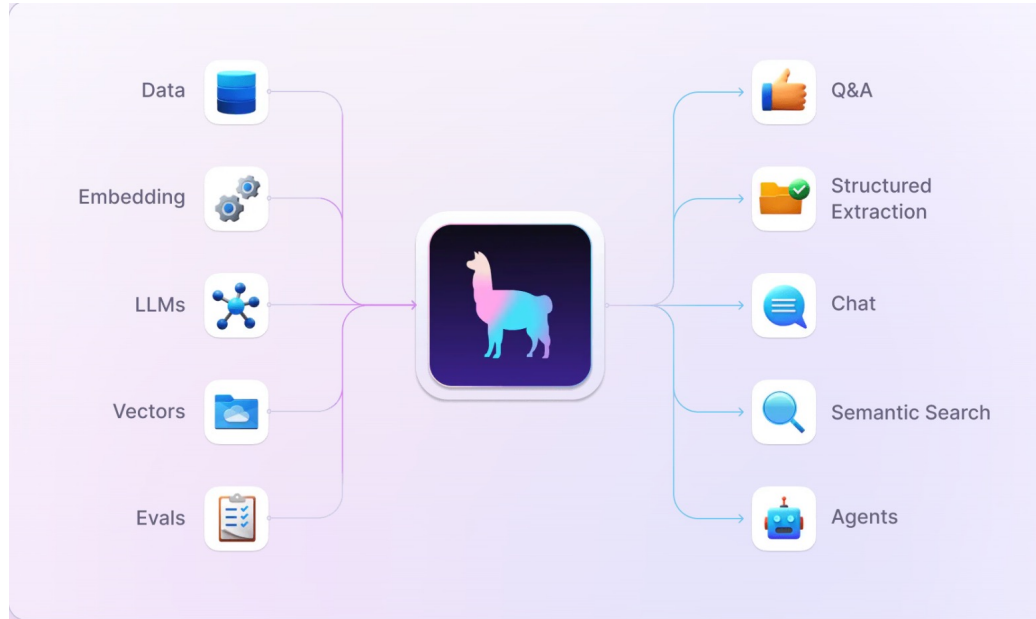
# What is Hugging Face

- It is a **framework**
- LlamaIndex wants to simplify the development of applications powered by language models
- Focuses on RAG use cases
- Excels in search, data indexing and retrieval



LlamaIndex

# What is Hugging Face





# LangChain Getting Started



LlamaIndex

- Easy install:

```
%pip install llama-index
```

- Pandas Query Engine

```
from llama_index.core.query_engine import PandasQueryEngine
```

# LLamaIndex Coding Demo

# DataHour : Q&A

# Thank you!