Langchain

Generative AI ⟶ LLM

Many providers:

1. OpenAI
2. Llama
3. Mistral

so on -----

LLM

Problem:

1. Limited context window
2. outdated knowledge base
3. Data privacy
4. cost
5. No connection with 3rd party tools

so on ----

Gen AI app:

Data in

- .txt
- .PDF
- .CSV
- .SQL
- .XLS
- .html
- .png

so on ----

Action

RAG
Agents
APIs
format
manipulate
Transformation
memory
3rd party tool
so on --

output

anything

Langchain

Orchestration framework

① Langsmith → LLOps tool → monitor
② Langserve → LLMops tool → serve
③ Langgraph → AgenAI tool

Qwen 32 → LLM

① OpenAI
⑤ Llama
⑥ Mistral
⑦ Falcon
som - -

LLM

limited context window
#

Llama Index

Haystack

① powerful open-source LLM framework by deepset

② Primarily used for building RAG

③ Supports QA system, semantic search, chatbot and document search

Why ?

① Combine retrievers and readers in modular pipelines.

② Plug and play support for HuggingFace, FAISS, OpenAI, Elasticsearch, Pinecone, Milvus.

③ Designed for production Ready, Scalable Gen AI apps.

Core components.

1. Document stores
2. Retriever
3. Reader
4. pipelines

#Langchain

① Primary purpose:
Building LLM application with tools / agent chaining

② Core strength: Agents, memory, tools, chain of reasoning

③ Ideal use case: multi tool agents, LLM apps, custom workflow

#LlamaIndex:

① Primary purpose: For building indexing, structuring and querying data with LLMs

② core strength: Data connectors, indexing, Query engines.

③ Ideal use case: connecting private data to LLMs (docs, API, SQL - -)

#Haystack

① Primary purpose: End to End search/QA (RAG, QA)

② core strength: modular pipeline for Retrieval Reader system

③ Ideal use case: Document search. QA system search RAG.

1, 2, 3, 4

$\frac{1+2+3+4}{4}$ → mean

male, female

↓

$\frac{\text{male} + \text{female}}{2}$ → male

10 + (mid)

→ 5 − fend