Evaluation of Open-Source Locally Runnable RAG Frameworks

Objective

As a developer, the goal is to assess and compare three popular open-source, locally runnable Retrieval-Augmented Generation (RAG) frameworks that support embedding-based document querying. This will help identify the best approach for integration in local LLM workflows.

Frameworks Evaluated

1. LangChain
2. Haystack
3. LlamaIndex (formerly GPT Index)

📌 Framework 1: LangChain

Overview
LangChain is a modular framework that integrates LLMs with external data sources like documents, vector stores, tools, etc.

Pros

- Highly flexible and composable
- Native support for local embedding models and vector stores like FAISS
- Active community and wide HuggingFace integration
- Rich abstractions for RAG (chains, agents, tools)

Cons

- Complexity increases with scale/custom use cases
- Slightly heavier learning curve for beginners
- Documentation can be fragmented due to rapid growth

📌 Framework 2: Haystack

Overview
Haystack by deepset is an enterprise-grade framework for building NLP pipelines, with robust RAG features and production-readiness.

Pros

- Modular pipeline design with local or cloud deployment
- Comes with UI, REST API, and eval capabilities out-of-the-box
- Fast inference using DocumentStore + Retriever + Generator architecture

Cons

- Larger footprint and setup overhead
- Less community-driven than LangChain
- May be overkill for small/academic projects

📌 Framework 3: LlamaIndex

Overview
LlamaIndex (formerly GPT Index) focuses on efficient retrieval by building indexes over structured and unstructured data for LLMs.

Pros

- Lightweight and easy to use
- Excellent for custom index building
- Integrates well with LangChain and OpenAI/HF models

Cons

- Less suited for complex, large-scale retrieval pipelines
- Documentation is improving but still maturing
- Limited to fewer backend integrations natively