Langchain+Ollma 大模型应用开发框架及LLM本地部署与开发

源起

Langchain这个项目，亮相于2022年10月，然后一下子在GitHub上火了起来，没多久就变成了个势头很猛的初创公司。创始人Harrison Chase也顺理成章地成了CEO。虽然一开始LangChain既没有收入，也没啥明确的商业化计划，但它就是厉害，很快就拿到了1000万美元的种子轮融资，后来又搞到了2000多万美元的A轮融资，估值飙到2亿美元。LangChain的快速崛起和资本支持，都说明AI领域特别需要这种创新的工具和平台，大家也都特别认可这种能推动AI技术应用和开发的东西。

创始人

Harrison Chase在2017年的时候还在哈佛学习统计和计算机科学，毕业后，他成了一名机器学习工程师，2022年秋天，Harrison本来准备离开之前的公司，但因为还没想好下一步干啥。于是他就到处参加黑客马拉松、聚会，跟一堆研究LLM的大佬们聊。聊着聊着，他发现好多共同的抽象概念，就搞了个Python项目当副业玩玩。结果这项目火得一塌糊涂，特别是在Chat GPT发布一个月后，LangChain的发展速度把Harrison Chase自己都惊到了。

Langchain 的 Logo

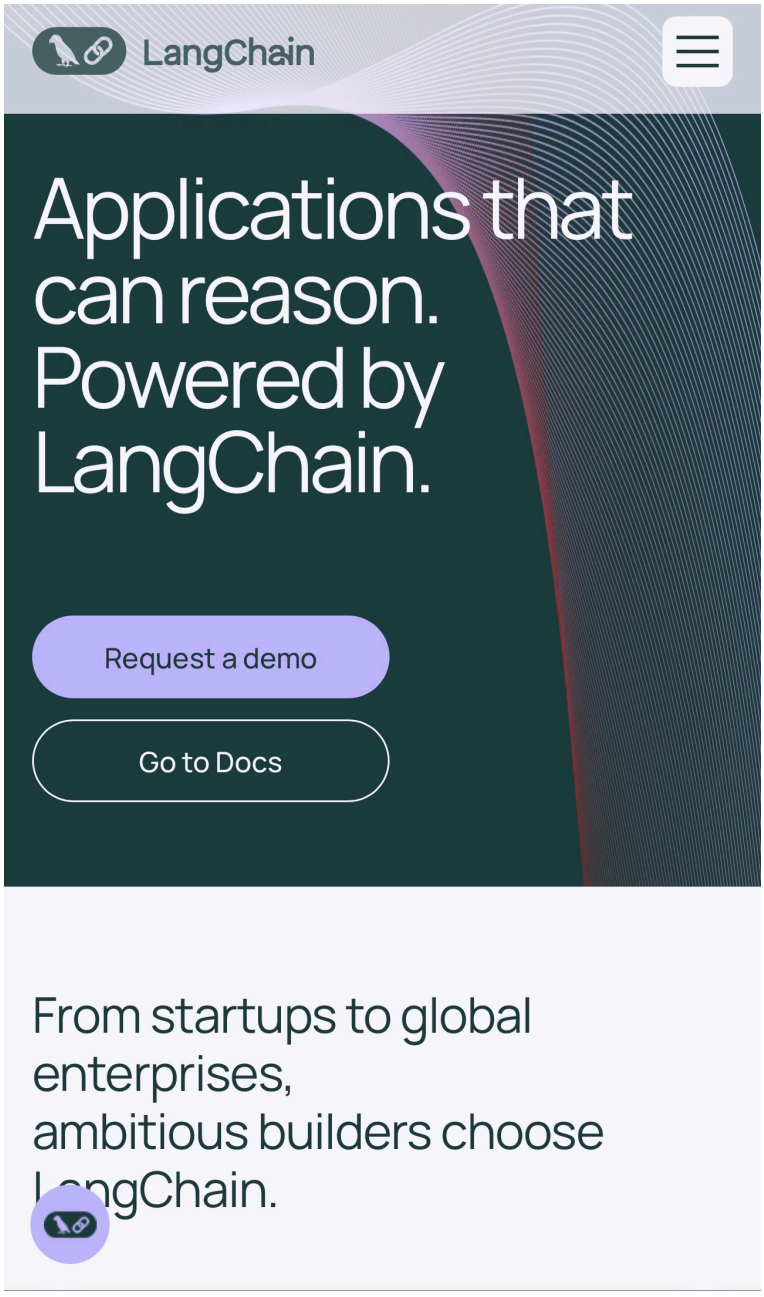
Langchain的logo由一只鸟和链条共同组成。这Logo的背后到底有啥深意呢？在人工智能这行里，大型语言模型（LLM）经常被形象地叫做“随机鹦鹉”，这个称呼挺有意思，意思是说

一只鸟能模仿人类输出文本，但其实并不真正“懂”自己在说啥。而Langchain的库呢，就是想办法把这些“鹦鹉”通过“链条”的方式连在一起，让它们能输出更有意义和实用的东西。加几句题外话，Onebird是我用了很久的一个id，源于我刚进小学那会儿，经常在考卷上把名字错写成“一鸟”。而作为一个域名注册爱好者，我早也就把 onebird.ai 域名给注册了。

产品价值

关于大语言模型（LLM），业界有一个这样的共识，大模型的业务可以被分为三层：最底层是基础设施层，负责搭架子、处理数据并进行存储，这一层同时还会提供MaaS(模型即服务）API；中间层是垂直领域层，可以用特定的数据微调模型，让模型在特定领域里表现更好；而最上层则是应用层，负责提供面向用户的产品和服务，比如聊天机器人、内容生成工具啥的。应用层的重点在用户体验和接口设计上。

LangChain的价值，说白了就是为了让大模型应用的开发和部署变得更简单，更高效。一方面它们提供了很多开箱即用的组件、模板和接口，让开发者从概念验证到生产部署都能变得更快。一方面，他解决了AI应用开发中很多相似的难题。也降低了技术门槛。李彦宏说大模型要多做应用创新，这点我完全赞同。



网站与社区
点击 https://www.langchain.com 可进入Langchain官网。

LangChain主要推出了三款产品，
1. LangChain的开源工具包 – 不仅提供了丰富的模块化组件，还实现了与数百个第三方提供商的集成，为用户提供了更广泛的选择和更便捷的操作体验。
2. LangSmith – 专注于测试、评估和监控等功能，旨在帮助开发者更全面地了解产品的性能和表现。
3. LangGraph – 帮助获得精确控制，从而构建能够可靠处理复杂任务的智能代理。

结合这些产品，总的来说，Langchain可以提供三方面的技术（methods）
– RAG – 搜索增强。用你的数据构建LLM的外部大脑。
– Agent – 加入人为监督，并使用AI代理创建有状态且可扩展的工作流程。
– Evaluation – 评估框架。不要凭感觉发布产品。要在整个生命周期内持续测试，衡量应用的性能（效果）。

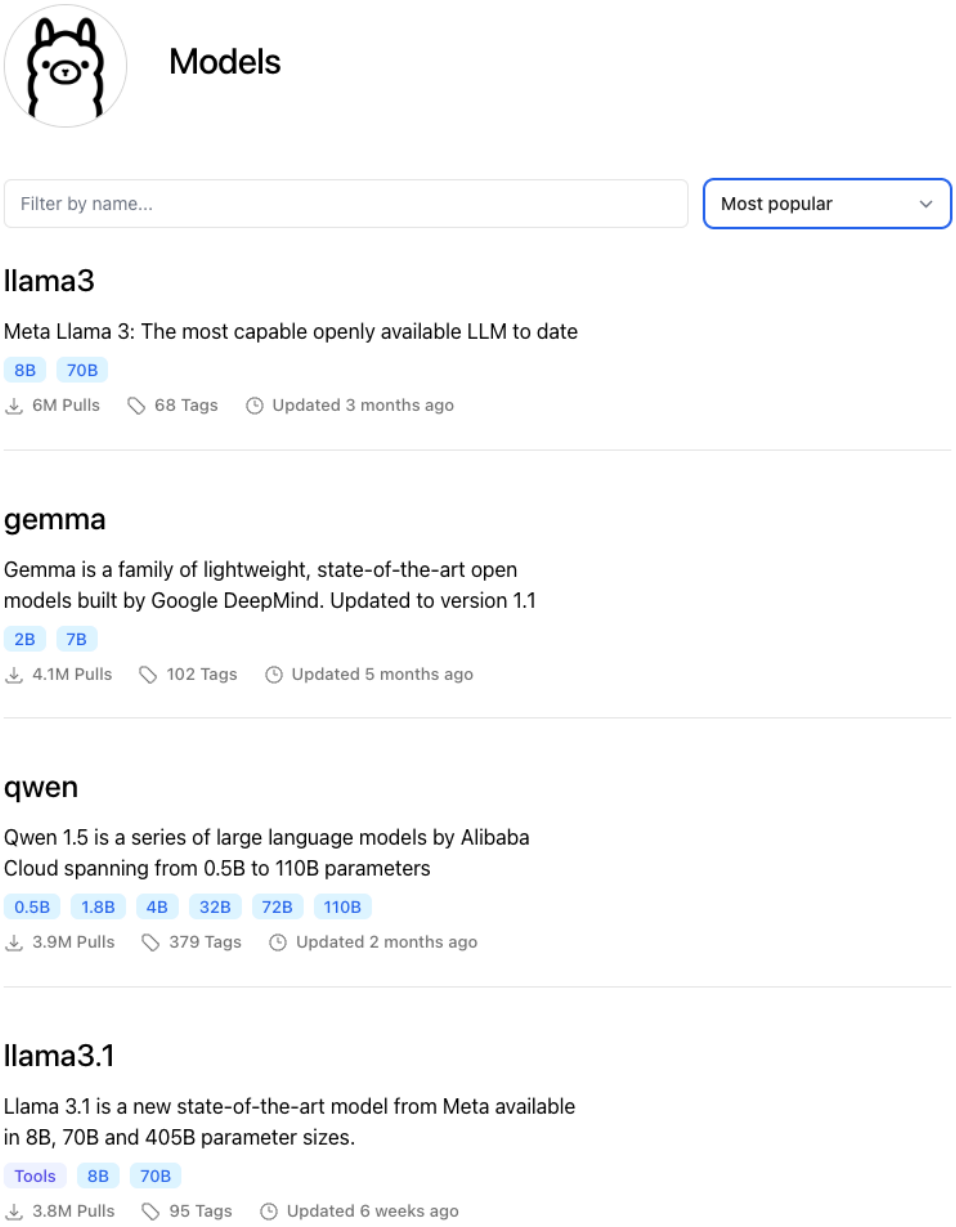
Langchain 开源工具包的GitHub地址是：https://github.com/langchain-ai/langchain
目前，该仓库在GitHub上已累计获得了91.9K的Star和14.6K的Fork，同时拥有超过3111名活跃的贡献者。

你可以通过Python和JavaScript(TypeScript)使用Langchain的开源工具包。在这篇文章的后面，我也提供了一个我制作的使用Cursor AI代码编辑器开发基于Langchain的RAG应用。

而Langchian也提供了相关的文档和例子，你可以去网站上自行探索。

使用 Ollama 在本地运行大模型
Langchain和很多大模型厂商都做了集成，你只需要提供自己的API key，就能去使用这些部署在云端的AI服务。但是，这需要你支付一定的开销。结合一些众所周知的原因，国内访问OpenAI也并不容易。所以，从开发角度，我们需要能够使用本地部署的方式运行LLM。

Ollama 就是这样一个专为本地运行的大语言模型 (LLM) 而设计的平台或工具。Ollama可以运行在多种平台。通过 https://ollama.com/download 这个地址，你就可以下载MAC/Win/Linux的安装。
而通过访问 https://ollama.com/library，可以查看ollama 所支持的模型。我在这里按照流行度做了一个排序。可以看到，排名靠前的是一些小模型。例如 llama 3 支持 8B和70B，而Gemma支持的是2B和7B。毕竟大多数人本地算力都不算宽裕。



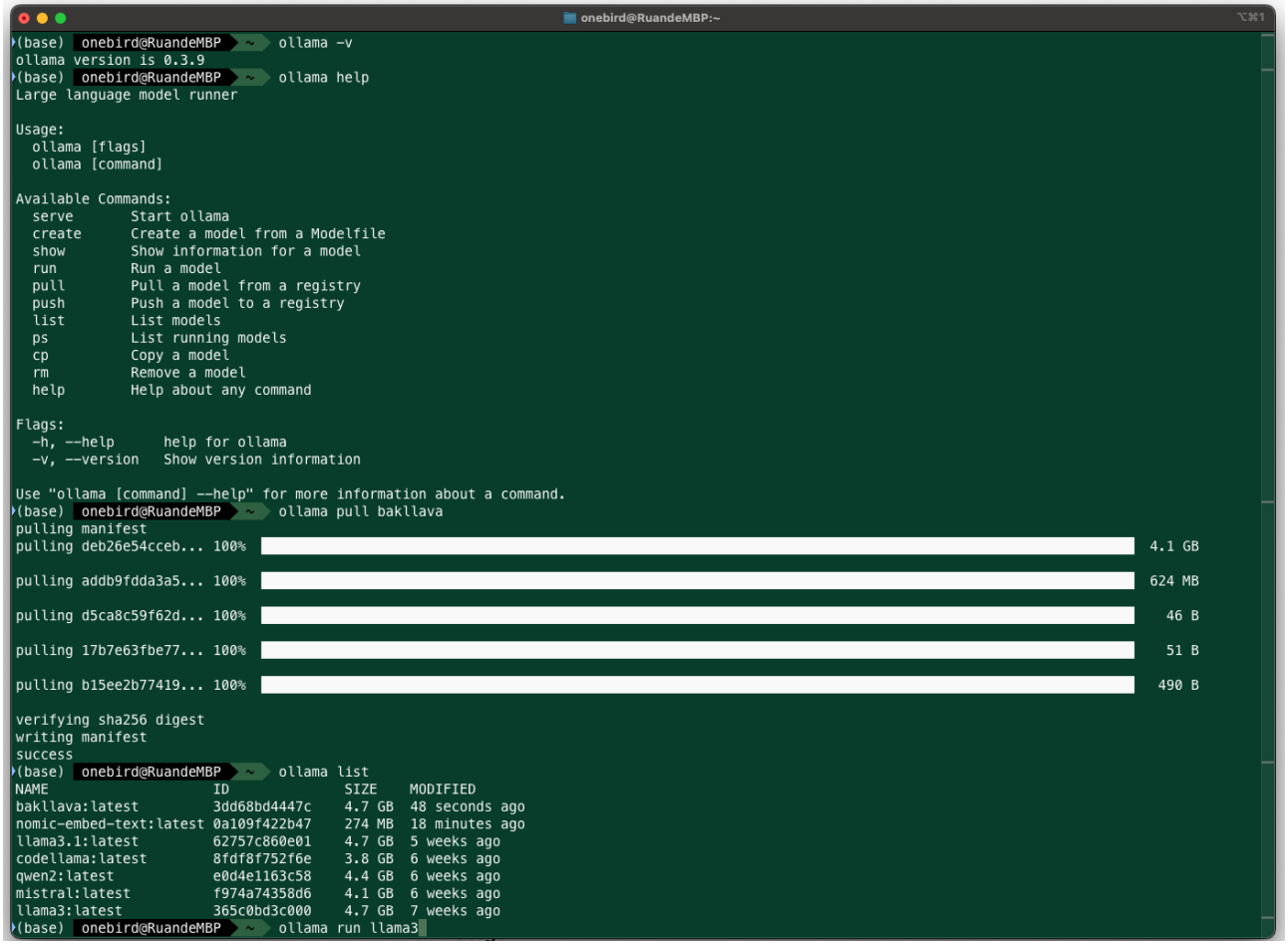
通过截图，你可以看到，在llama3.1的下面，有一个 “Tool”的标签，这代表该模型支持Tool calling的。Tool Calling的能力是在ollama0.3.0以后支持的。而在这之前，你如果需要支持Tool calling，只能使用 llama3-groq-tool-use 这样的模型。

所以，如果想在本地使用Tool Calling，请记得安装或者更新至于的ollama的最新版本。

除了文字到文字的模型。在OIlma上还可以找到一些从图片到文字的多模态模型，例如 bakllava 就是一个由 Mistral 7B 基础模型与 LLaVA 架构组合而成**多模态模型**，这类模型通常都会带上 “Vision”的标签。
而带有“Embedding”便签的模型，望文生义，就知道是用来做embedding的，比较知名的有 nomic-embed-text

Ollama大部分的模型的API都和openAI的借口兼容。ollama也提供了相应的python支持程序中调用。

下面的截图，我列出了Ollama的一些命令供你参考。



设置好环境后，就可以开始Lainchain的开发AI的旅程了。

作者
Onebird
连续创业者/大厂码农
阿里云MVP /数字游民/ 技术信徒
爱阅读，爱音乐，爱骑行，也爱电子产品
我在这里分享我的学习/思考/实践 大家一起进步