

大模型时代，程序员如何实现自我成长？——一名普通开发者的ChatGPT 一周年记

ChatGPT 发布一周年了，切实改变了我的工作方式和职业路径，趁着周末写下这篇文章，我希望以一名普通程序员的视角，带大家回顾一下过去一年大模型领域的发展情况，以及个人的所思所想。文章会分为四个部分，从初次接触ChatGPT沉迷追AI新闻，到开始亲身实践，利用LLM进行一些有价值的工作，然后以开发者视角总结一年来大模型各个层面的发展，标志性的开源项目、基础模型服务商、中间层、以及体验不错的LLM产品，最后还想再浅谈一下对AI未来的一些展望！全文共计7500+字，大约需要20分钟。

欢迎关注公众号，获得 LLM 应用开发相关实时信息！



微信搜一搜



LLM应用全栈开发

初次接触ChatGPT

2022年11月30日，OpenAI 正式发布 ChatGPT，在短短一年时间里，ChatGPT 不仅成为了生成式 AI 领域的热门话题，更是开启了新一轮技术浪潮，每当OpenAI有新动作，就可以占据国内外各大科技媒体头条。从最初的 GPT-3.5 模型，到如今的 GPT-4.0 Turbo 模型，OpenAI 的每一次更新都不断拓宽着我们对于人工智能可能性的想象，最开始，ChatGPT 只是通过文字聊天进行互动，而现在，已经能够借助GPT-4V 解说足球视频。

文字是思想的载体，第一次看到 ChatGPT 的演示效果，我就被震撼到了，看着对话框中的文字逐个跳现，流畅的内容表达、尽情展现想象力（虽然后面了解到实质是概率模型），这与以往任何智能对话机器人截然不同，真正体现了智能的魅力。随后，我便开始搜寻一切关于ChatGPT的信息，频繁刷新reddit上的ChatGPT话题讨论，检索X平台的ChatGPT关键词，查看科技媒体是否有相关报道.....

在接下来的一个月里，我既兴奋又焦虑，兴奋源于每当有空闲时间，我就能与ChatGPT这款“真正的人工智能”进行对话；而焦虑则缘于工作时总忍不住去刷新新闻，生怕错过其他用户分享的新玩法展示和有趣的提示词，当然彼时在国内没引起太大的关注，微信指数、百度指数以及新浪指数也只有浅浅的波动。



随着年假的结束，各类生成式AI产品不断走红，包括AI生成图像和编写代码等，在此期间，一位朋友分享给我一篇文章，发表在红杉资本美国官方网站上，标题为《[生成式 AI：一个创造性的新世界](#)》（英文 Generative AI: A Creative New World）的文章。我既非投资者，也不负责公司战略，仅是一名普通程序员，对我而言，这些由底层逻辑推导出的宏观趋势不过是“正确的废话”，并不能指导我的实际行动，然而，作为一个契机，却激发了我深层的思考：如何借助这些能力做个人层面实实在在的实践，为自己打补丁，构建对别人有益的产品，最大化的传递这其中的价值，这才是最重要的。

	PRE - 2020	2020	2022	2025?	2030?	2050?
TEXT	Spam detection Translation Basic Q&A	Basic copy writing First drafts	Longer form Second drafts	Vertical fine tuning gets good (scientific papers, etc)	Final drafts better than the human average	Final drafts better than professional writers
CODE	1-line auto-complete	Multi-line generation	Longer form Better accuracy	More languages More verticals	Text to product (draft)	Text to product (final), better than full-time developers
IMAGES			Art Logos Photography	Mock-ups (product design, architecture, etc.)	Final drafts (product design, architecture, etc.)	Final drafts better than professional artists, designers, photographers
VIDEO / 3D / GAMING				Basic / first draft videos and 3D files	Second drafts	AI Roblox Video games and movies are personalized dreams

Large model availability: ● First attempts ● Almost there ● Ready for prime time

最后补充下，即使到当下，还有很多人通过追AI新闻的方式，关注AI领域的最新动态，在这里我给大家推荐几个优质的资讯来源，所有这些都支持通过RSS进行订阅，部分可以邮件订阅，希望能解决你的信息焦虑，节省你的时间，更专注于手头的工作。

信息源推荐

1. 吴恩达老师的来信: <https://rsshub.app/zhihu/people/activities/wu-en-da-89>
2. deeplearning.ai 官网上的 *The Batch* 栏目，提供近一周的AI相关消息和观点: <https://www.deeplearning.ai/the-batch/>
3. Hugging Face 的 Daily Papers: <https://huggingface.co/papers>
4. 在 BriefGPT 上按照自己关注的细分领域订阅最新论文: <https://briefgpt.xyz>
5. 论文让人头疼，但你还是想看AI新闻怎么办，那就最节省时间的方式，每天花10分钟扫一遍奇绩论坛的大模型日报: <https://news.miracleplus.com/feeds>

亲自去实践

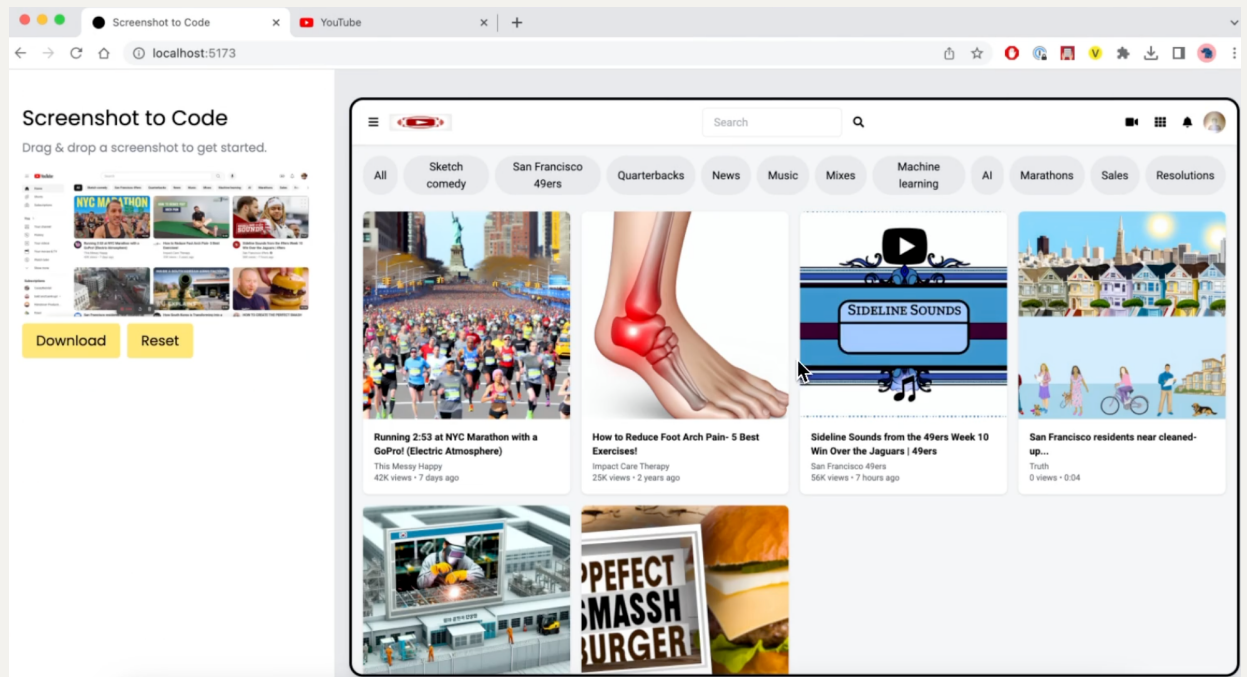
啰哩啰嗦的部分结束咯，终于来到我写文章的重点。对于程序员个人发展来说，目前最理想的选择是加入大厂模型训练或推理平台的基础设施团队，其次，可以考虑前往大模型初创公司的非核心部门，如果这两者不可行，那么可以考虑加入中间层和应用层的创业公司，投身赛道，参与其中才能获得切实的成长和收获。当然，也可以像我一样选择出来单干（大不了最后发现自己太菜😂，继续回厂打工：），但机会依旧有限，坑也就那么多，对大多数程序员来说，比较现实的是在当下，在自己的岗位上用好大模型的能力，我想这大概可以分为四个阶段：

1. 学习提示词，了解LLM的能力，改善自己和朋友的日常生活体验。举一个例子，我做了一个用于口语化记账的 iOS 快捷指令，分享给身边人用，由于大模型有强大的意图识别能力，记账不再通过在 APP 中手动去记录，而是直接唤起 Siri，口语化表达，然后提炼为结构化的内容，存储在备忘录或者具备API 接口的记录软件中。我给自己做了一个[游戏博客](#)，将油管上的游戏类热门视频内容生成文章总结，然后支持RSS订阅主题，这样我不用去刷，但是也不怕漏掉感兴趣的游戏视频。



像上面这种方式，自定义提示词做个简单应用，用于简历优化、文本校对、生成论文大纲等等，当下这些东西方案成熟，有开源参考，先把AI能力用起来。

2. 借助LLM扩展个人能力边界：大模型时代，人人都是能实现 MVP 的产品经理。做一个自己的小产品门槛更低了，你说自己不会设计，[trace](#)可以帮你生成SwiftUI；你说自己不会前端，[Screenshot to Code](#)借助GPT-4 Vision可将任何网站的屏幕截图/网址转换为代码 (HTML/Tailwind CSS、React、Vue 或 Bootstrap)，实现精准的网站实时克隆。关键是看你是否颠覆常规的创意，以及是否能够发掘细分市场的需求，AI提供了更多根据自己的想法创造的时间和机会。



- 产品研发层面，在产品生命周期的各个阶段让LLM更多的介入进来帮助自己。程序员都有造轮子的爱好，为的就是提高效率，节省时间，那么个人的工作流程能不能优化下，让 AI 参与进来。设计技术文档，和技术沟通方案时，使用时序图和UML类图辅助说明十分常见，使用用户旅程图和产品经理交流，使用甘特图 and 项目经理同步进度，这些内容其实都可以借助大模型了，只要你将自己的想法理清清楚表述出来，做图部分交给AI生成就可以，比如我这个项目[智能阅读助手](#)[ReaderGPT 开发记录](#)中所有的设计图都是AI做的。从系统设计到代码编写、测试和维护，我已经将之前每个项目都要经历的标准产出物都交给 AI生成了，自己只做不同项目实施时的具有特殊性的部分进行确认完善，当前 AI 已经分担了我以往30%的日常工作量。
- 在团队中推广LLM知识和技术，提升工作效率，并探索其在组织中的应用：积极地将这些技术回馈到工作中，在可能获得额外回报的情况下帮助同事，改善团队流程，并寻求更多公司层面的机会。例如，可以尝试将团队积累的文档资产转化为知识库问答应用，或者开发一个产品QA机器人，先过滤90%的用户问题，少点烦恼不好吗，这个过程你会认识到，token成本的控制，RAG技术有一堆dirty work要做，还有提示词安全也需要考虑。做完这些你就是团队最懂AI的那个人，此外，还可以设计开发一个与团队技术栈相匹配的类LangChain内部框架，以支持基于LLM的应用规模化开发。



莫尔索

02/27

花两天时间验证了一个自己一直想要的小工具，失败告终😞

预期目标：任意文字搜索播客内容，可以定位到相应内容标记点进行收听

验证过程：

1. 以 ByteTalk 第一期为例，将一个小时的音频文件分成30s的片段
2. 利用whisper 语音识别，得到时间戳和相应文字内容的映射
3. 用sbert-base-chinese-nli （基于bert，针对中文有优化的llm）做了embeddings
4. 结果保存在在线向量数据库 milvus
5. 搜索时对任意搜索文字进行embeddings，然后去milvus匹配相应的 top k结果

主要失败点：

1. 第一步的识别阶段问题很大，因为播客没有像视频一样的标准srt字幕，预处理阶段需要人工介入，无法自动化跑通😞
2. 资源消耗感人（下图中是自己在colab上跑识别模型的资源消耗情况），可以预见采用语音识别 API 服务成本兜不住

5. **LLM**作为一个新兴领域，大家都站在挖掘其潜力的同一起点上，掌握得越多，就越能显现出你的专业度：虽然我还没到这个阶段就自己出来单干了，但是上个阶段的经验已经足够有价值，只有好好钻研设计过提示词你才知道如何更好的控制模型输出稳定性，你也会有一套包含调试提示词、提示词版本管理、设计A/B测试方案、提示词效果跟踪、评估的整体方案，来自用户的交互反馈，收集的高质量数据资产微调一个服务于产品的模型，尽管并不是每个开发者都需要掌握模型训练和微调的能力，但这是随着能力提升也需要扩展的技能方向。

当然，现在开始学习也不迟，按照下面的路径扎实的学习一遍，就能比99%的人更深入地理解大型模型的能力边界，从而更好地驾驭这些大模型。

LLM 开发入门课程

- 《ChatGPT Prompt Engineering for Developers》：面向入门 LLM 的开发者，深入浅出地介绍了对于开发者，如何构造 Prompt 并基于 OpenAI 提供的 API 实现包括总结、推断、转换等多种常用功能，是入门 LLM 开发的经典教程。
- 《Building Systems with the ChatGPT API》：面向想要基于 LLM 开发应用的开发者，简洁有效而又系统全面地介绍了如何基于 ChatGPT API 打造完整的对话系统。
- 《LangChain for LLM Application Development》：介绍了如何基于大模型应用开发框架 LangChain 开发具备实用功能、能力全面的应用。
- 《LangChain Chat With Your Data》：在《LangChain for LLM Application Development》基础上进一步介绍了如何使用 LangChain 架构结合个人私有数据开发个性化大模型应用。
- 《Building Generative AI Applications with Gradio》、《Evaluating and

Debugging Generative AI》教程分别介绍了两个实用工具 Gradio 与 W&B，指导开发者如何结合这两个工具来快速构建 MVP、评估 AI 应用。

LLM 开发进阶课程

1. Google 的《Generative AI learning path》

- 课程地址: <https://cloudskillsboost.google/journeys/118>
- B站播放: <https://space.bilibili.com/589397373/channel/collectiondetail?sid=1468916>

2. 《Full Stack LLM Bootcamp》

- 课程地址: <https://fullstackdeeplearning.com/llm-bootcamp/spring-2023/prompt-engineering/>
- 我的note: <https://zhuanlan.zhihu.com/p/629589593>

3. AWS 的《Generative AI with Large Language Models》

- 课程地址: <https://www.bilibili.com/video/BV12s4y1r7jf/>
- 相关的 Notebook: <https://zhuanlan.zhihu.com/p/642560031>

一些优质的博客

1. LangChain 官方博客: <https://blog.langchain.dev/>
2. LlamaIndex 官方博客: <https://blog.llamaindex.ai/>
3. OpenAI 开发者论坛: <https://community.openai.com/>
4. Pinecone官方博客: <https://www.pinecone.io/blog/>
5. W&B官方博客: <https://wandb.ai/fully-connected>
6. Azure 的AI主题博客: <https://azure.microsoft.com/en-us/blog/product/azure-ai/>
7. Cloudflare: <https://blog.cloudflare.com/tag/developers/>

百花齐放的生态

不得不说，大模型赛道生态太繁荣了，源源不断有新公司冲进来，大厂的新业务也在往这个方向靠，为了话题不要跑太远，我还是以一个技术人视角，从最经典的基础模型层，中间层以及应用层的技术图谱角度来聊聊。

基础模型层

从我自己做应用的角度，将基础模型层的能力我分为三类，直接以API方式提供模型能力的，提供算力和配套工具能够支持模型训练的，集前面两种能力的云厂商，下面都是我体验过的服务。

国外

OpenAI 一骑绝尘，释放的 API 能力越来越强，价格也越来越便宜，Function Call，Plugin，还有后来的 GPTs，可以看出想在提供基础能力之外做的更多。微软押对了筹码，坐上顺风车，也及时上架了搭载OpenAI 模型的云服务，同时 bing搜索，office套件，Windows 系统都集成接入AI能力，开源方面，Meta的Llama和斯坦福的羊驼推动了AI技术平权，基本是个人自托管大模型的首选。Bard 插件系统，可以整合谷歌旗下产品，从 Google Workspace（Doc、Gmail、Drive和 Maps等）到 Youtube、Google Flights 等，Amazon 推出的大模型托管服务 [Bedrock](#)，支持 Amazon 自家和AI21 Labs、Anthropic、Cohere 和 Stability AI几家模型，并且与AWS Lambda 函数服务进行了整合，这里需要提下 Anthropic，Claude 2的能力仅次于GPT-4。

最后不得不提，对开发者比较重要的几家服务，Cloudflare 满足AI时代应用开发范式，[顺势推出AI 推理即服务平台 Workers AI](#)，[矢量数据库产品Vectorize](#)，[用于增强AI 应用可观察性、可靠性和可扩展性的 AI Gateway](#)；Replicate，连续创业者与人工智能专家联手打造的AI模型托管云平台，在费用计算粒度和成本控制及扩展性方面表现出色，让普通开发者也能玩转大模型；最后就是AI领域的Github—Hugging Face，提供了众多好用的预训练模型和数据集，还有限量免费的AI推理能力，是个人寻找灵感、验证点子的不二之选。

国内

国内做大模型的，百度得益于以往在AI领域的积累，作为国内对标OpenAI的厂商，最早推出文心一言，而后紧跟发布插件商店，也是国内最早面向开发者提供API的，而智谱由于早期开源大模型的ChatGLM2-6B，则在开源社区积累了众多用户，百川智能作为新创公司迅速崛起，成立两个月便开源了Baichuan-7B，16 天之后发布了免费可商用的Baichuan-13B，值得关注，以上三家都是个人使用过其API服务的，其他家的暂未体验。下面其他内容来自道听途说，大厂由于人才密集，加资金雄厚，全产品线推进AI能力，比如阿里的通义开始对菜鸟、淘宝、高德等业务进行赋能；大模型创业公司minimax，可能是国内第一家同时拥有3个模态大模型能力的公司，其大模型驱动的toC产品Glow用户数逼近500万（目前已下架），还有团队比较年轻的先锋公司，Moonshot（Kimi Chat主打超长上下文）、深言科技（WantWords 反向词典、深言达意都很好用）和面壁智能。

前段时间还看到统计说国内大模型已经有180家了，虽然可以在风口上糊弄投资人，拉拉股价，但实际体验骗不了做应用的开发者，更骗不了亲自体验的用户，时间会告诉答案。选择哪家模型，可以根据这篇文章提出的一些建议作为参考 [首批国产AI大模型已开放？开发者不容忽视的 5 个问题！](#)

API 申请入口

智谱AI开发者申请入口：<https://open.bigmodel.cn/>

百川智能开发者注册入口：<https://platform.baichuan-ai.com/docs/api>

文心一言API接入指南：<https://cloud.baidu.com/article/1089328>

中间层

狭义上的 LLMOps

说到中间层，不得不提一个新名词 LLMOps，这个没有准确的定义，但是很多做中间层的技术厂商都声称自己是 LLMOps 平台提供商，我个人认为应该分为狭义上和广义上的，你需要什么能力再去补足就行，狭义上的 **LLMOps** 本质上是一套新的工具和最佳实践，用于管理由 **LLM** 驱动的应用的生命周期，包括开发、部署和运营，我认为这是一些关键能力：

- 提示管理，包括提示工程、审核、跟踪、A/B测试、提示链接、调试提示、评估等，包括跨多个大模型提供商进行提示链接；
- 无代码或低代码形式的微调和嵌入管理，包括在特定数据集上重新训练通用模型的工具；
- 代理集成，基于行动的LLM决策，执行行动，目标规划，与外部世界接口等；
- 可观测性，分析成本、延迟、速率限制管理、可解释性等

下面是一些符合上面要求的一些 LLMOps 平台：

Relevance AI集成了OpenAI、Cohere、Anthropic等多家模型厂商，内置模型输出质量控制、结果缓存等，服务了包括联合利华在内的 20 多家企业级客户，是我最钟意的产品之一。

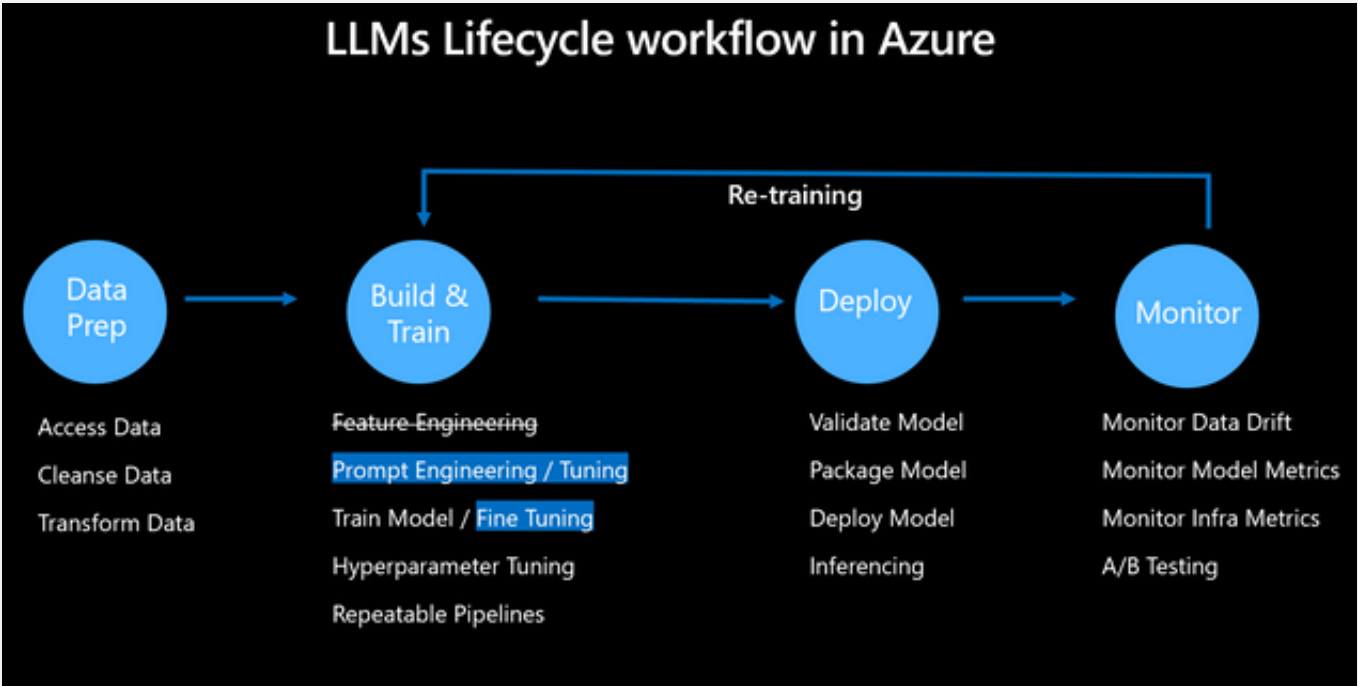
[HoneyHive](#)内置提示词版本控制和日志记录，使用自然语言处理（NLP）指标对模型性能进行评估，执行单元测试以确保模型质量，并结合人工反馈进行优化；收集高质量数据集，用于进一步的微调和蒸馏各种主要模型提供商的自定义模型，此外还可以根据需要定制指标进行可视化，比较不同数据切片的性能，识别异常情况。

[Stack AI](#)一种无代码工具，使用大模型设计并测试AI 工作流后，可以一键将其部署为 API，此外还可以优化提示、收集数据并微调 LLM，已经有付费企业用户在使用了，当然还有国内开源的Dify、FastGPT、Bisheng，做的很薄，缺乏利用收集的高质量用户数据进一步微调模型的环节。

除了平台型产品，还有一些聚焦细分领域的产品尝试，[Rebuff AI](#)是防御提示词攻击的AI应用安全产品，[Humanloop](#)专注于AI应用可观测性，还有我之前介绍过的[如何控制 LLM 应用的使用成本—可观测性平台 Helicone](#) 介绍，专注提示工程工具的[PromptLayer](#)和agenta，我也之前有提到 [使用了这款提示词管理工具后，LLM 应用的稳定性和准确性提高至99%](#)。

广义上的 LLMOps

而广义上还涵盖了大模型本身的开发、部署、维护和优化的一整套实践和流程，是在传统 MLOps基础上做了一点变化（图源来自 Azure 博客）



这部分几大云厂商有完整的产品覆盖，不过也有创业公司来吃细分领域，Weights & Biases 和 Neptune 是用于记录模型训练实验数据的工具，BentoML 是用于机器学习（ML）模型服务和部署的开源平台，OctoML 和 MosaicML 也都是帮助企业优化和部署机器学习模型的平台，这部分我了解不多，训练大模型的实践经验有限，只能谈谈以上都是我简单使用过的产品。

向量数据库

最后，向量数据库需要单独说说，虽然之前的文章[万字长文带你深入了解向量数据库（附大量图表，建议收藏）](#)谈及过向量、以及向量数据库选型的话题，但是我个人比较认同的观点是：专用的向量数据库是会慢慢褪去热度，而加向量扩展的经典数据库会成为主流，独立的向量数据库会带来额外的开销和运维复杂性，在经典数据库和向量数据库之间跨地区传递数十亿个嵌入，这部分成本也非常高。比如 BaaS 服务提供商 **Supabase**（主打 PostgreSQL 数据库服务，并且集成了用户登录验证）使用 `pgvector` 进行向量存储和 `embeddings` 支持后，开发者使用 **Pinecone** 的热度没那么高了，MongoDB 也发布了 **Atlas Vector Search**，Elasticsearch 本身提供了向量搜索功能，作为本就精力有限的开发者，有什么理由给自己多加一个数据库，徒增烦恼呢。

一些链接

1. 向量数据库不是一个单独的数据库分类：<https://nextword.substack.com/p/vector-database-is-not-a-separate>
2. Weights&Biases's 的 LLM Ops 介绍：<https://wandb.ai/iamleonie/Articles/reports/Understanding-LLMOps-Large-Language-Model-Operations--Vmlldzo0MDgyMDc2>
3. Supabase 发布 AI 和 Vector 向量数据库的工具箱：<https://supabase.com/vector>
4. Azure 的 LLM Ops 介绍：<https://techcommunity.microsoft.com/t5/ai-machine-learning-blog/an-introduction-to-llmops-operationalizing-and-managing-large/ba-p/3910996>
5. 上面提到的公司技术博客。

应用层

终于来到应用层，我对好应用标准是，在当前大模型技术还不够成熟的阶段，没有AI能保证基本能力达标，加上AI体验能拉到90分，应用层的过去一年的变化总体可以分为四个阶段，基本都是由开源项目引爆的。

1. 入门阶段：构建以单一提示词为中心的应用程序。
2. 进阶阶段：通过组合提示词+外部知识库创建更复杂的应用。
3. 发展阶段：开发由大模型驱动的智能代理（Agent）应用。
4. 探索阶段：实现多个智能代理协同工作，以应对高度复杂的应用场景。

入门阶段

提示词工程进入更多人的视野，这个阶段最大化的利用了 GPT 文本生成能力，各种内置提示词的套壳应用层出不穷，角色扮演、工具类应用，套壳应用集合，套壳应用发布平台，当然只有一些对细分领域的业务和标准流程有极致理解，并将其提炼为提示词，这类有价值的应用最后脱颖而出，即使到今天也拥有很高的用户留存率。不过应用层的繁荣，必然也伴随着安全风险，提示词攻击的话题也开始被人关注到，文章指路[Claude 2已被越狱？一文带你了解提示攻击！](#)

进阶阶段

由于大模型的知识范围有限，急需引入外部数据提升模型的问答效果，RAG 话题讨论热度一度达到最高，这种技术基于提示，最早由 Facebook AI 研究机构（FAIR）和合作者于 2021 年发布的论文《[检索增强生成用于知识密集型自然语言处理任务](#)》中提出。当然这类技术的快速普及跟开源社区的繁荣发展密不可分，LlamaIndex（GPTIndex，一个用于LLM应用的数据框架，[LlamaIndex原理与应用简介](#)）、LangChain（一个利用LLM构建应用的框架，[从零开始开始学LangChain系列](#)）、Semantic Kernel（一个轻量级的 SDK，可以让传统编程语言与 LLM 集成，[大模型时代软件开发：吴恩达提到的Semantic Kernel是什么？](#)）是其中关注度比较高的项目，不过这一些阶段的能力构建是建立在上一阶段的提示词运用水平上。



莫尔索

04/18

蹭下网红AutoGPT流量，解决个issues再说😁

Resolving Unicode encoding issues

hbeales approved these changes 4 days ago

richbeales merged commit 2c279eb into Significant-Gravitas:master 4 days ago
checks passed

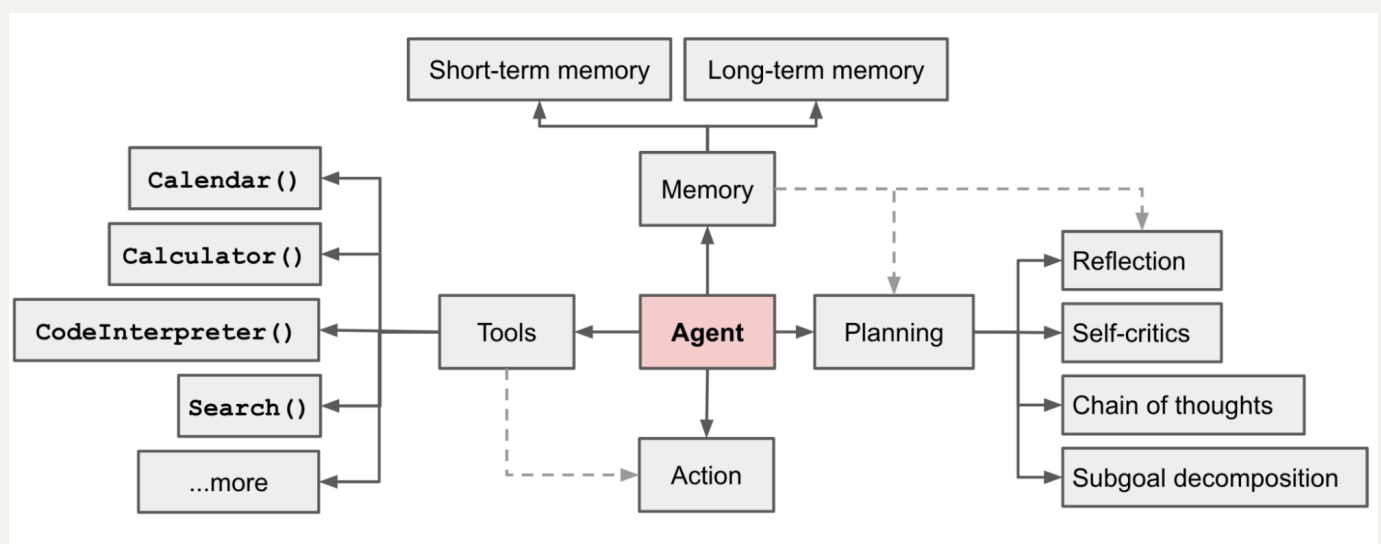
nponeccop mentioned this pull request 3 days ago

Batch 5 (WIP) #1162

1 task

发展阶段

AutoGPT 的炫酷展示，引爆了 Agent（智能代理）话题，当然日常能看到谈论最多也是 LLM 支持的自主代理，具备反省能力，搭配记忆管理，任务规划和工具使用，而不是传统人工智能领域的智能体概念，紧随其后的微软JARVIS，作为一个协作系统，由LLM作为控制器和Hugging Face 上众多的专家模型作为协作执行者。当然这一阶段需要对上一阶段的 RAG技术的理解要更深一层，回溯哪部分记忆，什么时候调用API，如结合LLM的知识编排你的流程等等，如果说AutoGPT 只是概念验证，那 SuperAGI（面向开发者的开源Agent开发框架）和 MindOS（心识宇宙团队）就是完成度和用户体验都极佳的产品化实践。



探索阶段

多 Agent 协作探索是这一阶段的主要关注点，前一阶段单个 Agent 调教能力扎实，才具备构建这一层的可能性。斯坦福小镇项目是这个阶段开始的标志，创造了25个AI NPC，每个NPC都有不同的身份和行动决策，让它们在一个沙盒环境中共存，模拟真实的人类生活。随后开源的 MetaGPT 项目专注于软件开发，从需求分析到代码编写等实现了全流程的覆盖，发布一个月后在GitHub上达到1万star，同时登顶趋势榜，可见热度之高；而 AutoGen 综合上面优势，创造性提出用于构建多智能体合作的可编程元框架，灵活性极高，可以根据自己的需求定义任意的Agent，然后让它们一起工作，这个阶段各种概念验证项目层出不穷，产品化道阻且长，但未来让人充满期待。

应用层总结

Inflection AI 的 Pi 是目前使用体感最好，如果说我将 ChatGPT 当作工具人，那 Pi 就是一个善解人意的小伙伴，优先于人的设计理念，相比其他服务于生产力、搜索或解答问题的 AI，可以放心将Pi视作一个具有创造力的教练、朋友或者是一个“发泄情绪”的对象，当然有类似定位的还有致力于让每个人都可定制自己的个性化AI的Character.ai，不过靠提示词调教出来的虚拟角色稳定性堪忧，但是以这个项目作为入口，收集海量高质量交互数据，具备构建出体验超过Pi的大模型潜力。

国内这边，使用过个性化英语教育（依据你的行业和工作岗位，AI生成个性化的教材，着重于所在行业的特定语言概念和行业术语，然后对接东南亚性价比较高的外教老师视频 1V1 教学）和跨境电商营销工具类产品，还有儿童绘本定制。

还有接到过的咨询，国企是想在数智化建设中把AI融入做舆情监测，偏ToG是在进境物品管理方面在传统模式识别基础上增加AI能力做准确率提升。

AI的未来

AI的发展必须着眼于解决我们日常生活中的实际问题，而不应仅仅局限于追求大模型技术本身，正如我们在使用手机时，很少会意识到背后的通信和计算机技术一样，当AI技术广泛应用时，也不会特别强调其背后的复杂大模型，真正成熟的技术应当是润物无声，无形中融入我们生活的，悄无声息地提升我们的生活质量，此外，AI的发展应当是帮助人们更好地思考自我价值和启发创造力，以及对个人学习潜力的激发，实现人本身的更全面解放。