

## ● 思考 Agent

OpenAI 发布了 ChatGPT 这个具有划时代意义的产品，有人说它堪比“iphone”，但 OpenAI 并不满足于此。他们希望成为 AI 时代的苹果公司。之前的 ChatGPT 插件引起了广泛关注，被称为 ChatGPT 的 App Store 时刻。然而，插件的影响力相对有限，无法与 ChatGPT 相提并论。相比之下，Agent 能带来更大的影响力，真正重塑现有的应用场景。它集成了 LLM（大型语言模型）、记忆、规划技能和工具使用，展现了更广阔丰富的想象空间。OpenAI 现在将重点放在 Agent 上也就不难理解了。



### 挑战

目前的 Agent 还存在一些问题，包括以下几个方面：

1. 有限的上下文长度：Agent 的设计必须适应有限的通信带宽，这限制了历史信息、详细说明、API 调用上下文和响应的有效性。虽然使用向量存储和检索可以提供对更大知识库的访问，但它们的表示能力不如 LLM 强大。
2. 长期规划和任务分解的挑战：在长对话上进行规划并有效地探索解决方案空间仍然具有挑战性。当出现意外错误时，LLM 很难调整计划，相比不断试错学习的人类，它们的稳健性较低。

3. 自然语言接口的可靠性：目前的代理系统依赖于自然语言作为LLM和记忆、工具等外部组件之间的接口。然而，模型输出的可靠性是有问题的，因为LLM可能会出现格式错误，并且偶尔出现不遵循指令的行为。因此，在许多代理演示代码中，重点放在解析模型输出上。

这些问题需要不断的研究和改进，以使Agent在交互方式、自然语言准确性和注意力窗口等方面更加稳定和高效。

## 个人思考

Agent的理念并非昙花一现，它们是首批由通用AI驱动、能解决任务的实体。随着时间的推移，随着更强大的模型和工具的支持，它们将变得越来越复杂。

例如，你可以想象一个简单的客户服务Agent，它可以接手某人的问题，迭代地将其分解、解决，并验证答案。为了实现这个目标，需要几个关键条件：

1. 更强大的模型：GPT-4工作得很好，但放在 Agent 的使用场景仍然有限。
2. 更好的工具系统：对于真正的生产使用场景来说，现在的API库还有所欠缺。
3. 不同的架构：随着模型的演化，将目标分解为子任务可能不再是正确的设计决策，有许多像从最终状态开始并向后工作的方法可能同样有效。

在我的研究中，我发现 AutoGPT 在处理一些简单且明确定义的知识任务时表现良好，但在更困难的任务上会出现不可靠的情况。这种不可靠性主要是由GPT-4的固有限制引起的。我认为这些问题无法仅通过更复杂的提示技巧来根本解决，而可能需要更多的微调来改进。