Summary of Generative Agent

沙之洲 2020012408

这篇工作提出了一种基于 LLM 控制 agent 在模拟世界中自由交互的任务，并且构造了一种 memory 机制，让 agent 能够在一定程度上拥有对经历过事情的记忆。本文在这个世界中尝试举办 party，并且分析了这个消息在社会中的传播速度。

优点在于，首次尝试了以 LLM 作为基座的多智能体任务，以往的多智能体往往都是基于强化学习 agent 的。同时，构建了 memory buffer，在 agent 面对事情的时候，能够借助 retrieval 的方法从 memory buffer 里边获取相关的信息；还构建了 reflection环节，能够借助 LLM 对 memory buffer 中的内容进行定期整理；以及 planning 环节，能够帮助 agent 完成多部的任务。

这篇文章的局限之处在于，由于 gpt 的闭源导致无法对于模型参数的进一步调整。同时，由于 agent 本质是基于 LLM 的，每一次的推理都过于昂贵，导致这种方法的 scalability 不强。最后，这篇文章中模拟的都是一些简单的日常生活，不涉及到 agent 之间的复杂合作，所以很多人性的本质在这篇工作中无法体现。

这篇工作一个可能的发展方向是探索如何更合理地保存 memory。我认为这篇工作中以自然语言形式保存 memory 的行为并不会成为最终的答案。虽然这篇文章中有 reflection 机制能够保证自然语言以一定的周期进行压缩，但是这种压缩必然会导致一些细节但是重要的信息损失。反观人类，我们虽然会随着时间推移对一些事情的记忆逐渐淡忘，但是当我们遇到对应的事情的时候，以往重要的细节仍然会浮现出来。举一个例子，我们在小时候读童话故事的时候会看到反派想要利用毒蘑菇去毒杀主人公，而蘑菇有毒这个细节并不是童话故事的重点。如果采用本文中 reflection 的方法，蘑菇有毒这个细节肯定会被优化掉。而对于人类而言，虽然蘑菇有毒这个事情不是童话故事的重点，但是若干年后，当我们真正看到一个蘑菇的时候，我们会因为读到过这个细节而变得谨慎。

因此，我们不可能通过简单的 memory buffer 加上 reflection的方法来模拟人类的记忆。一种更好的解决方案是让 model 自己完成记忆的选择存储和更新的事情，这就意味着下一步需要解决的一个关键问题是如何让 model 具备 online learning 或者说 life-long learning 的能力。

最后，关于 scalability 的问题。一种可能的解决方案是将 LLM 作为 agent 的知识库，而非去指导 agent 的每一个动作。具体来说，可以考虑将求助于 LLM 作为 RL agent 的一种 action。通过训练这个 agent，使得只有当 agent 觉得需要求助于 LLM 的时候再进行 LLM 的 inference。由于运行一个 RL agent 的成本远远小于 LLM，这种方法或许能够让 generative agent scale 到一个非常大的规模上。