Chapter 1

论文阅读

1.1 LLM-based autonomous agent construction

最近的研究表明 agent 在解决 question-answering (QA) 问题上有巨大潜力，但是要建立自主代理 QA 远远不够。要弥合传统代理和自主代理之间的差距，一个主要的方面是设计合理的架构，使得 LLM 最大限度发挥能力。

[1] 中提出了一个统一的框架，由 profiling 模块,memory 模块,planning 模块和 action 模块组成，如图 1.1 所示

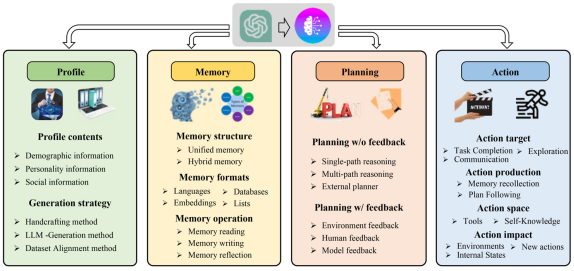

图 1.1: LLM-based autonomous agent construction

1.1.1 profiling

自主代理扮演程序员，老师等特定角色来完成任务。在 profiling 模块中，通过编写特定的 prompts 可以对大模型行为产生影响，如 List1 所示。

1.1.2 memory

1.1.3 planning

1.1.4 action

目前有三种主要的设计方法：第一种是手工制作。如 MetaGPT、ChatDev 和 Self-cooperation 预先定义了软件开发中的各种角色及其相应职责，手动为每个代理分配不同的配置文件以促进协作，这种方法灵活但是需要人力; 第二种是大模型自动生成配置文件。大模型首先指定配置文件的生成规则，如规定组成和属性，然后给定少数作为示例的配置文件，最后通过大模型生成配置文件。

第三种是数据集对齐方法。这种方法中配置文件是从真实世界数据集中获得的，通常情况下将真实数据集转换为自然语言的 prompt。例如根据美国国家选举研究（ANES）参与者的人口统计背景（如种族/民族、性别、年龄和居住州），将角色分配给 GPT-3，随后研究 GPT-3 是否可以产生与真实人类相似的结果。

Listing 1: prompt 示例

1.2 LLM Multi-Agent

Multi-Agent 与 Single-Agent 相比更注重多样化的配置文件，代理之间的交互以及集体决策，可以处理更动态和复杂的问题

[2] 中介绍了 Multi-Agent 框架的四个主要模块：环境接口，代理配置，代理通信，代理能力获取。

1.2.1 环境接口

环境接口指的是代理与环境交互并感受环境的方式，当前分为 sandbox，physical 和 none 三类。sandbox 指人类构建的虚拟环境，如狼人杀，代理可以自由地与环境交互并尝试各种方法和策略。physical 指现实世界的环境，代理与物理实体交互并受物理规则约束，如代理机器人。none 指不与任何环境交互，如考试和辩论，它们基于代理间的通信而不需要与环境交互。

1.2.2 代理配置

代理配置指的是代理的特征，行为和技能是根据特定任务量身定制的。如软件开发任务中有产品经理和工程师等角色，辩论平台中有正方，反方和评委等角色。当下有三种主要方式：预定义，模型生成和数据来源。预定义指人为定义配置文件，模型生成指模型生成代理配置文件，数据来源指根据数据集构建模型代理文件。

1.2.3 代理通信

代理通信是构建 Multi-Agent 的关键基础设施，主要从通信范式，通信结构和通信内容三方面进行分析。通信范式指代理之间沟通的风格和方法，主要采取合作，辩论，竞争三种范式。合作指多个代理为了共同目标交换信息，集体决策。辩论指代理会维护自己的观点和方案，批评其他的方案。竞争则是代理的目标可能与其他代理的目标冲突。

通信结构主要有: 前馈网络，去中心化，中心化，共享信息池四种，如图 1.2 所示:

1.2.4 代理能力获取

代理能力获取是指代理能够学习和进化。其中有两个基本概念：应学习的反馈类型和调整自己得到解决复杂问题的策略。反馈类型主要有四种：从环境中得到的反馈 (如机器人从现实世界获得反馈); 从其他代理获得的反馈，如不同代理之间辩论; 从人类得到的反馈; 没有反馈，因为有些情况以结果分析为主要目标，如模拟真实世界。

代理可以从三个方面调整自己: 记忆，自我进化和动态生成。代理将以前交互和反馈的信息存储在他们的记忆中。在执行操作时，他们可以检索相关的、有价值的备忘录。代理可以通过修改自

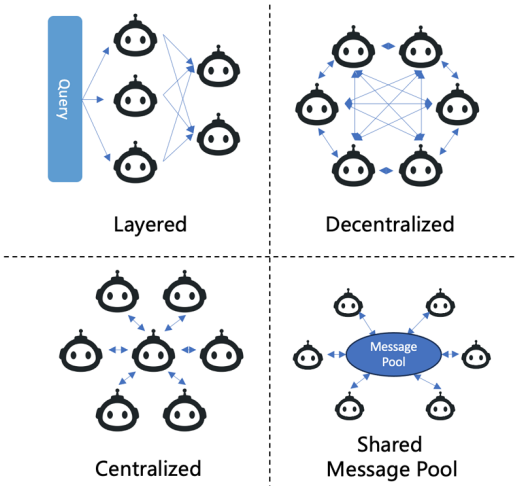

图 1.2: Communication Structure

已来动态自我进化，例如改变其初始目标和规划策略，并根据反馈或通信日志进行自我训练在某些情况下，系统可以在运行过程中动态生成新的代理。此功能使系统能够有效地扩展和适应。

1.3 LLM GAME

大模型可以在游戏中扮演玩家，非玩家（敌人，NPC），玩家的助手，控制游戏流程的主持人，或者游戏机制的管理者。而且可以担任游戏的设计师或玩家数据的分析师，甚至是游戏的评论员(足球游戏中的解说等)。

1.3.1 LLM 扮演 player

LLM 适合扮演玩家的游戏有如下特征：状态和动作可以紧凑地表示为抽象的 token；主要的输入和输出是自然语言；外部程序可以通过 API 控制玩家动作

回合制棋盘游戏

与第一人称射击游戏相比，棋子的位置和移动可以更容易表示为 token，通过对游戏数据库中的动作序列进行标记，动作选择问题可以映射到训练 LLM 的标准自回归学习目标——在给定前一个动作的背景下预测下一个动作。而且这种方法可以适用于更复杂的场景，例如，在 Bateni 和 Whitehead 的作品 [3] 中，LLM 玩 Slay the Spire(杀戮尖塔)（Mega Crit，2017），仅根据卡片的描述理解卡片之间的协同作用，并适应游戏规则的变化.

文字冒险游戏

LLM 在这类文本游戏中的最早应用是 CALM，aGPT-2 系统从各种文本游戏中收集的人类游戏记录。该模型被训练来预测人类玩家在给定先前状态、动作和信息（例如他们的库存）的情况下提供的自然语言字符串。为了实际玩游戏，训练语言模型生成多个候选动作，并使用深度强化学习（RL）来优化从候选动作中选择动作的策略。

有强大 API 的游戏

优秀的 API 使大模型不必直接生成玩家动作，而是生成充当策略的程序。例如，VOYAGER 系统 [4] 利用了 GPT-4 的代码生成能力玩 Minecraft（Mojang Studios，2011）流行的 Mineflayer API。VOYAGER 使用复杂的提示链生成代码块，利用对 API 的调用来执行高级“技能”（例如“攻击最近的僵尸”），这些技能会自动转换为低级游戏输入（例如鼠标移动和按键）。

1.3.2 LLM 扮演 Non-Player Characters

LLM 可以通过对话和行为来控制 NPC。NPC 主要分为三种：前景 NPC，背景 NPC 和讲述者 NPC。

前景 NPC 担任游戏的主叙事或字叙事，它们的对话严重受叙事范围，它们在叙事中的作用以及玩家行为的限制。它们需要考虑游戏背景，玩家行为并且跟进发生的事件。所以人们担忧 LLM 的记忆存储是否足够，而且 LLM 容易出现幻觉（看似合理但是虚假的陈述）。其他研究表明，多个基于 LLM 的代理能够遵循游戏规则并参与游戏，不同的模型在应用于特定角色时始终表现出自己的能力和弱点。这种在约束内进行交互的能力有助于在背景 NPC 中灌输可信的行为，使他们的行为和对话建立在游戏环境规则的基础上。LLMs 表演类似于戏剧即兴表演的东西，而不是作为学习角色的演员。通过这种不受约束的过程，LLM 容易产生不符合预期场景的幻觉，这种波动性可以通过向 LLM 提供对话历史以及环境的当前状态来缓解

这种方法也可以扩展到其他场景，或涵盖将 LLM 用作主动或交互式叙述者。Ubisoft 在他们的 Neo-NPC 演示中展示了基于 LLM 的 NPC [5]，玩家可以在其中与游戏中的角色自由对话。每个 NPC 都被赋予了一个精心制作的角色，所有 NPC 都可以在游戏叙事和规定的性格的约束下做出反应，同时产生现实的反应。玩家能够与这些 NPC 进行游戏特定的活动，例如策划抢劫，甚至尝试完全无关的对话过程。通过迅速定义每个 NPC，我们注意尽量减少 LLM 中固有的毒性和社会偏见，这也影响到后者对任何玩家的攻击性或不守规矩的话语的反应。

背景 NPC 的目的是使环境更可信，并独立于玩家行事。这样的 NPC 的存在纯粹是装饰性的，他们的对话本质上是闲聊。

1.3.3 LLM 扮演 assistant

LLM 还可以担任助手的角色，给玩家一系列提示，指导玩家活动但不与游戏世界进行任何交互。例如，在《模拟人生》(Electronic Arts，2000）中，一个无实体的助手通过对话框提供特定于游戏环境的提示。《文明 VI》（Firaxis Games，2016）使用不同的助手根据其独特的启发式方法建议最佳构建选项，帮助玩家进行决策。由于其表达和对话能力，LLMs 作为玩家助理很有吸引力。一个基于 LLM 的玩家助手可以合理地选择一个动作来向玩家建议，更重要的是，将这个建议的解释形成一个由无实体或有实体的代理人发出的自然语言，甚至可能伴随着相应的情绪。

玩家助手的一个特殊情况来自玩家自己的化身发出的“内心声音”。比如 Guybush Threewood 在《猴岛的秘密》(Lucasfilm games，1990）中拿起物品时评论道：“一只中间有滑轮的橡皮鸡……这有什么用?”。尽管 LLM 的自由度有限，但“内心声音”这一特定概念在 Rist 的工作中被探索为 LLM 的应用 [6]。Rist 使用了一个手工制作的游戏环境，其中包含预先编写的事件和基于位置的触发器，这些事件和触发器是在玩家与世界互动或观察某事时发生的，当遇到某些预定义的游戏事件时，LLM 会被提示以不同的风格（例如中性或讽刺的语气）生成简短的评论。这些评论是基于用户在那一刻看到和可以做什么的文本描述生成的。LLM 仍然需要人工编写的知识（例如评论发生在哪里，提示是什么，以及为什么要达到设计师的目标），因此它不是一个完全自主的玩家助手。尽管

Rist [6] 的工作更侧重于沉浸而非辅助，但目前的研究并没有探索 LLM 驱动的玩家助手的潜力。

LLM 也非常适合作为评论员或复述者。在这里，我们将这些角色视为一个产生和叙述一系列事件的代理。这样的代理可能只考虑游戏中的事件和游戏环境，充当游戏中的实体，如国际足联的体育评论员（EA sports，1993），或者也考虑游戏外的事件和环境，如玩家（他们的行为、策略、动机等）。复述者专门讲述过去的事件，通常分为一个简洁的“块”，如游戏会话（即基于游戏外的上下文）或任务（即仅基于游戏内的上下文）。评论员可能正在讲述尚未结束的当前正在进行的事件，类似于流媒体同时讨论他们当前的行动（包括游戏外的情况），或者在正在进行的体育游戏（如国际足联）中的体育解说员。

1.3.4 LLM 扮演 Game Master

LLMs 作为 GM 也开启了单人游戏的潜力，而 TTRPG 至少需要一名玩家和一名人类 GM。由 GPT-2 的微调版本管理的第一个值得注意的文本冒险之一是 AI Dungeon [7]。这是一个基于在线交互式聊天的讲故事应用程序，玩家仅通过语言输入即可采取行动。这个 LLM 以人类 GM 的方式根据玩家的输入继续故事。游戏自创建以来已经发展到使用最新的 LLM 模型，玩家可以在开始游戏之前从中进行选择，还提供了不同的游戏世界设置，玩家还可以分享他们创建的故事。

1.3.5 LLM 作为 GameMechanic

游戏也可以围绕依赖 LLM 的特定机制构建。在这方面，生成代理项目 [8] 采用 LLM 来填充一个有 25 个角色的虚拟村庄，使他们能够在沙盒环境中进行交流和参与社交行为。玩家可以通过文本与这些代理人互动。每个代理的环境状态和动作以基于语言的格式存储并汇总，以便在提示其动作时保留每个代理的知识。这导致了新出现的可信的社交互动，例如代理人自发地邀请其他代理人参加其中一人正在组织的聚会。同样，GoodAI 正在开发 AI 人视频游戏，该游戏作为沙盒模拟运行，LLM 驱动的 NPC “与彼此及其环境互动，形成关系并展示情感”[9]。玩家可以通过自然语言聊天与代理人互动，引发反应，并可能破坏 NPC 之间的关系。

自然语言交互形成了一个自然的机制池来构建游戏，例如将用户越狱 LLMs 的尝试游戏化 [10]。游戏《1001 夜》通过让 LLM 根据人类提示共同创造一个故事来证明这一点，玩家的目标是试图引导故事包含特定的关键词，以便主角 Scheherazade 将这些转化为有形的物品来帮助她逃跑 [11]。同样，Gandalf7 挑战玩家诱骗 LLM 泄露密码。游戏通过调整提示规范来提高任务的难度，例如迫使 LLM 重新检查其生成的响应，以确保其中不包含密码。

同样，语言合成也被 Infinite Craft8 所利用，这是一款“炼金术”游戏，玩家将元素组合起来产生新的元素。在《无限工艺》中，玩家从一组核心元素（水、火、风和土）开始。虽然前者有一组由设计师手动定义的交互，但 Infinite Craft 提示 Llama 2 想象这些元素组合的产物 [12]。从游戏玩法来看，对于每种不同的组合，Llama 似乎只会被提示产生一次结果，并将产品存储在数据库中以供将来参考。因此，语言模型词汇表中的任何内容都可能从搜索元素的组合中“出现”.

1.4

参考文献

[1] Wang, L., Ma, C., Feng, X. et al. A survey on large language model based autonomous agents. Front. Comput. Sci. 18, 186345 (2024). https://doi.org/10.1007/s11704-024-40231-1

[2] Guo, T., "Large Language Model based Multi-Agents: A Survey of Progress and Challenges" , arXiv e-prints, Art. no. arXiv:2402.01680, 2024. doi:10.48550/arXiv.2402.01680.

[3] B. Bateni and J. Whitehead, "Language-driven play: Large lan guage models as game-playing agents in Slay the Spire," in Proceedings of the International Conference on the Foundations of Digital Games, 2024.

[4] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, and A. Anandkumar, "Voyager: An open-ended embodied agent with large language models," in Proceedings of the NeurIPS Work shop on Foundation Models for Decision Making, 2023.

[5] L. O' Brien, "How Ubisoft' s New Generative AI Prototype Changes the Narrative for NPCs," https://news.ubisoft.com/en us/article/5qXdxhshJBXoanFZApdG3L/how-ubisofts-new generative-ai-prototype-changes-the-narrative-for-npcs, 2024, accessed 12 June 2024.

[6] T. Rist, "Using a large language model to turn explorations of virtual 3d-worlds into interactive narrative experiences," in Proceedings of the IEEE Conference on Games, 2024.

[7] M. Hua and R. Raley, "Playing with unicorns: AI dungeon and citizen NLP," Digital Humanities Quarterly, vol. 14, no. 4, 2020.

[8] J. S. Park, J. O' Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein, "Generative agents: Interactive simulacra of human behavior," in Proceedings of the Annual ACM Symposium on User Interface Software and Technology, 2023.

[9] GoodAI, "Introducing our work on general-purpose LLM agents," https://www.goodai.com/introducing-general purpose-llm-agents/, 2023, accessed 24 Feb 2024.

[10] Y. Liu, G. Deng, Z. Xu, Y. Li, Y. Zheng, Y. Zhang, L. Zhao,T. Zhang, and Y. Liu, "Jailbreaking ChatGPT via prompt engineering: An empirical study," arXiv preprint arXiv:2305.13860,2023.

[11] Y. Sun, Z. Li, K. Fang, C. H. Lee, and A. Asadipour, "Language as reality: A co-creative storytelling game experience in 1001 Nights using generative AI," in Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2023.

[12] T. Litchfield, “This browser-based ‘endless crafting game’ starts you off with fire and water, but it quickly escalates to God, the Big Bang, and ‘Yin-Yoda’,” https://www.pcgamer.com/this-browser-based-endless crafting-game-starts-you-off-with-fire-and-water-but-it quickly-escalates-to-god-the-big-bang-and-yin-yoda/, 2024,accessed 28 Feburary 2024.