

人工智能与社会课程

大作业选题意向书

题 目： 基于 Casevo 框架的智能体决策能力优化研究

小组成员： 王宇东（组长）、陈文远

日 期： 2025 年 10 月 16 日

摘要

随着人工智能技术的快速发展，基于大语言模型的多智能体社会模拟系统成为研究社会演化、集体决策和复杂社会现象的重要工具。本研究以开源框架 Casevo 为基础，聚焦于智能体决策能力的优化问题。通过引入多层次推理机制、优化记忆检索策略、改进反思算法以及增强协同决策能力，我们期望显著提升智能体在复杂社会场景中的决策质量和适应性。本意向书详细阐述了研究背景、技术路线、实验设计和预期成果，为后续研究工作奠定基础。

1 拟定题目

基于 Casevo 框架的智能体决策能力优化研究

2 研究背景与意义

2.1 多智能体社会模拟的研究现状

社会模拟作为计算社会科学的重要分支，长期以来致力于通过构建计算模型来理解和预测复杂的社会现象。传统的基于规则的智能体（Agent-based Model, ABM）虽然在特定场景下表现出色，但其行为逻辑高度依赖于预设规则，难以捕捉人类决策的复杂性和不确定性。近年来，大语言模型（Large Language Models, LLMs）的突破性进展为社会模拟带来了新的可能性。通过将 LLM 与多智能体系统相结合，研究者能够构建具有类人认知能力的虚拟智能体，这些智能体不仅能够理解自然语言，还能进行推理、记忆和反思，从而更真实地模拟人类在社会环境中的行为。

Casevo（Cognitive Agents and Social Evolution Simulator）正是在这一背景下应运而生的开源框架。该框架由 Jiang 等人于 2024 年提出，专门用于构建基于复杂网络的社会模拟多智能体实验。Casevo 在 Mesa 框架的基础上进行了扩展，集成了大语言模型接口、思维链（Chain of Thought）推理机制、检索增强生成（RAG）的记忆系统以及可定制的反思机制。这些特性使得 Casevo 在模拟选举投票、舆论演化、信息传播等社会现象方面展现出显著优势。

2.2 当前面临的挑战

尽管 Casevo 框架已经展现出强大的能力，但在应对更加复杂和动态的社会场景时，其智能体的决策机制仍存在一些不足之处。具体而言，现有的基于线性思维链的决策方式在处理多约束、多目标的复杂决策问题时，往往表现出较大的波动性，难以保证决策的稳定性和一致性。此外，当智能体需要处理大量历史信息时，现有的记忆检索机制在效率和准确性方面还有待提升。在多智能体协同场景中，智能体之间的信息交换和决策协商机制也相对简单，限制了群体智能的发挥。

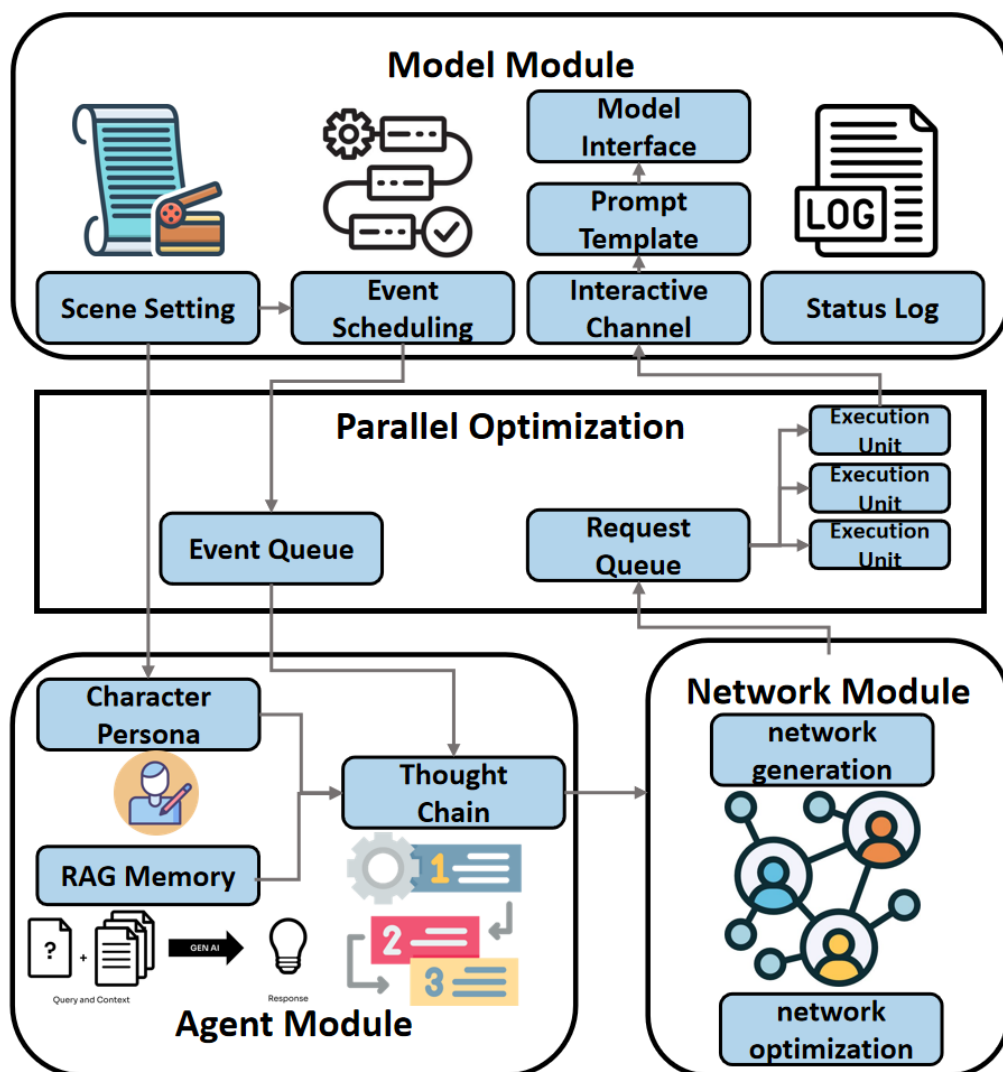


图 1: Casevo 系统架构图 (引自原论文 [1])。该架构包含四个核心模块: Model Module (模型模块) 负责场景设置和事件调度, Agent Module (智能体模块) 管理智能体行为和记忆, Parallel Optimization Module (并行优化模块) 提高计算效率, Network Module (网络模块) 构建和管理社会网络结构。

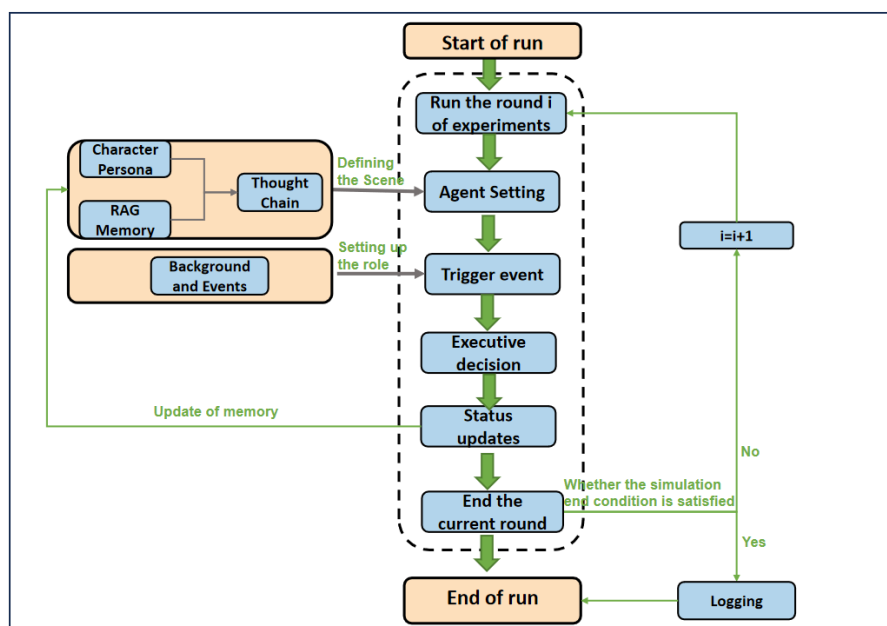


图 2: Casevo 轮次执行机制示意图（引自原论文 [1]）。系统采用基于轮次的离散事件模拟，每轮包含公共事件、智能体交互和状态更新等阶段，确保行为同步和事件有序调度。

2.3 研究意义

本研究针对上述挑战，提出对 Casevo 框架中智能体决策能力的系统性优化方案。研究的理论意义在于探索大语言模型在社会模拟中的深层应用，推动认知智能体理论的发展；实践意义在于提供一套可落地的优化方法和工具，为社会科学研究者提供更强大的模拟平台，帮助他们更准确地预测和理解复杂社会现象。

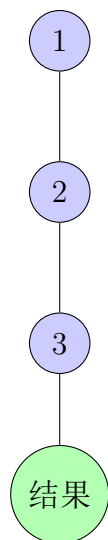
3 研究内容

本研究将从理论研究、系统实现和实验验证三个层面展开，形成一个完整的研究闭环。

3.1 多层次推理机制研究

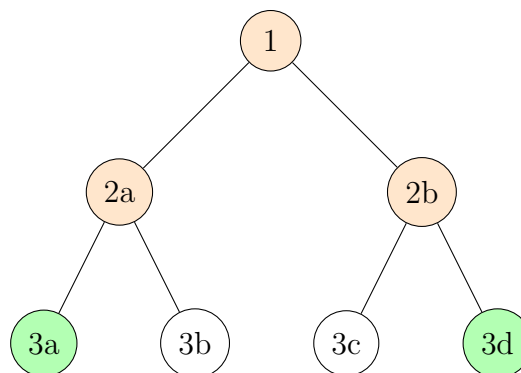
传统的思维链（Chain of Thought）方法采用线性的推理结构，智能体按照固定的顺序执行一系列推理步骤。这种方法在处理简单问题时效率较高，但面对复杂的决策场景时，线性结构的局限性就显现出来了。为了克服这一问题，我们计划引入树状思维（Tree of Thought, ToT）机制。与线性思维链不同，树状思维允许智能体在每个决策节点上探索多个可能的推理路径，通过对比不同路径的结果来选择最优方案。

Chain of Thought



线性推理

Tree of Thought



多路径探索

图 3: 思维链与树状思维的对比

具体而言，我们将设计一个分层决策架构，在该架构中，智能体首先对问题进行初步分析，识别出关键的决策维度，然后针对每个维度生成多个候选方案。通过对这些候选方案进行评估和筛选，智能体最终选择综合评分最高的方案作为决策结果。这种方法不仅能够提高决策的质量，还能增强决策的鲁棒性，因为即使某一路径上出现错误，其他路径仍可能导向正确的结果。

此外，我们还将研究推理深度对决策质量的影响。过浅的推理可能导致决策考虑不周，而过深的推理则会增加计算成本。因此，我们需要找到一个平衡点，使得智能体能够在有限的计算资源下做出高质量的决策。

3.2 记忆检索与利用策略优化

记忆系统是认知智能体的核心组件之一。Casevo 当前使用基于 ChromaDB 的向量数据库来存储和检索智能体的记忆。虽然向量检索在相似度匹配方面表现出色，但在处理大规模记忆数据时，检索效率和准确性仍有提升空间。我们计划从以下几个方面对记忆系统进行优化。

首先，我们将引入上下文感知的记忆筛选机制。传统的向量检索主要基于语义相似度，但在实际应用中，记忆的相关性不仅取决于内容的相似性，还取决于时间、情境等因素。例如，在选举投票场景中，最近发生的辩论事件通常比几周前的事件更具参考价值。因此，我们将在相似度计算中引入时间衰减因子和情境匹配度，使得检索结果更加符合当前决策的需要。

其次，我们将研究短期记忆与长期记忆的协同利用策略。短期记忆存储最近的交互信息，具有较高的时效性；长期记忆则存储经过反思和总结的稳定观点，具有较强的一致性。如何在决策过程中合理平衡两者的权重，是一个值得深入探讨的问题。我们计划设计一个动态权重分配机制，根据决策任务的性质自动调整短期记忆和长期记忆的贡献比例。

最后，为了应对大规模记忆数据带来的存储压力，我们将实现智能记忆压缩与遗忘机制。该机制能够识别出低重要性的记忆，并将其从数据库中删除或归档，从而保持记忆系统的精简和高效。

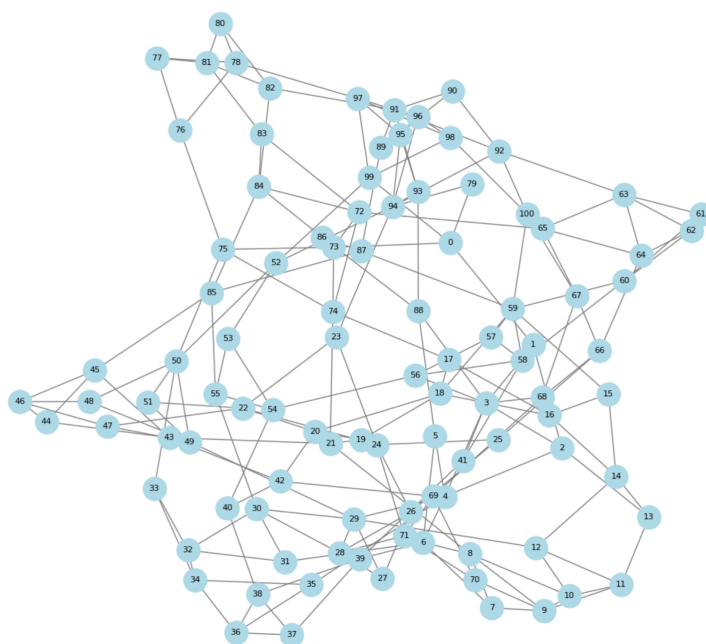


图 4: 选民社会网络结构示意图（引自原论文 [1]）。该网络采用小世界网络拓扑，包含 101 个节点，每个节点代表一个选民智能体，边表示选民之间的社交关系。这种网络结构能够真实模拟现实社会中的信息传播和意见交流模式。

3.3 反思机制优化

反思（Reflection）是智能体自我改进的重要手段。通过定期回顾历史经验，智能体能够提取出一般性的规律和见解，从而在未来的决策中做出更明智的选择。Casevo 已经集成了基于 Chain of Thought 的反思机制，但该机制的触发条件和反思深度都是固定的，缺乏灵活性。

我们计划引入元认知机制，使智能体能够评估自身决策的可信度。具体而言，智能体在做出决策后，不仅输出决策结果，还输出一个置信度分数。当置信度较低时，智能体会自动触发反思过程，重新审视自己的推理逻辑和所依据的记忆信息，并尝试修正可能的错误。这种动态反思策略能够使智能体在保证决策效率的同时，避免因信息不足或推理错误导致的严重失误。

此外，我们还将设计多层次的反思机制。浅层反思关注具体决策的对错，而深层反思则关注决策背后的价值观和信念。通过结合这两个层次的反思，智能体能够形成更加稳定和一致的长期观点。

3.4 多智能体协同决策机制

在社会模拟中，智能体之间的交互和协作是不可或缺的环节。当前 Casevo 支持智能体之间的对话和信息交换，但缺乏系统化的协同决策机制。我们计划设计一套完整的协同决策框架，包括信息交换协议、决策协商机制和结果聚合算法。

在信息交换方面，我们将定义标准化的消息格式，使得智能体能够清晰地表达自己的观点、依据和置信度。在决策协商方面，我们将借鉴共识算法的思想，设计一个迭代式的协商过程。在每一轮协商中，智能体交换彼此的观点，并根据其他智能体的意见调整自己的立场。经过多轮协商后，群体逐渐形成共识，或者识别出无法调和的分歧。

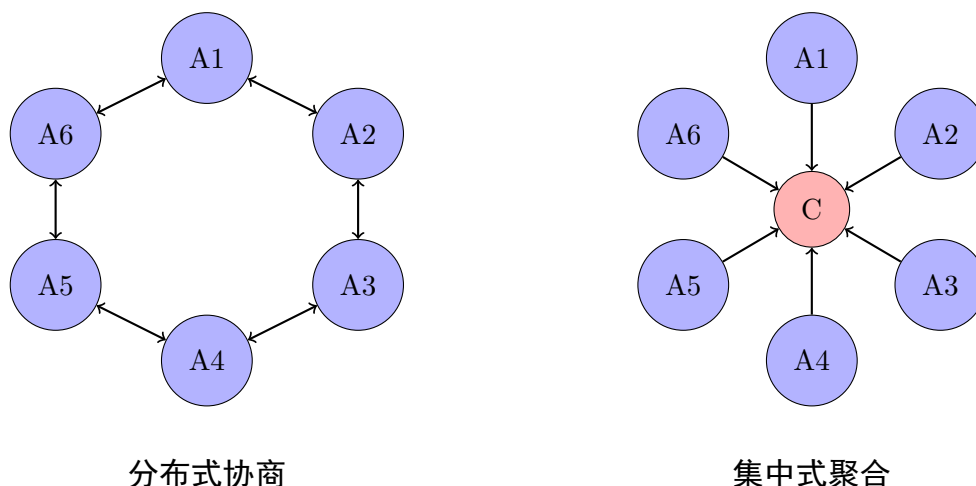


图 5: 多智能体协同决策模式对比

在结果聚合方面，我们将探索分布式决策与集中式决策的权衡。分布式决策允许每个智能体保持独立性，适合处理观点多元的场景；集中式决策则通过一个中心节点来整合所有智能体的意见，适合需要快速达成一致的场景。我们将根据不同任务的特点，设计自适应的决策模式选择策略。

3.5 系统实现

理论研究需要通过系统实现来验证其可行性和有效性。我们将基于 Casevo 框架进行扩展开发，实现上述各项优化方案。具体而言，我们将开发以下几个核心模块：增强型思维链模块（`enhanced_chain.py`），支持树状推理和多路径探索；优化的记忆管理模块（`advanced_memory.py`），集成上下文感知检索和智能遗忘机制；协同决策

模块 (`collaborative_decision.py`), 提供信息交换和协商框架; 决策质量评估模块 (`decision_evaluator.py`), 用于量化评估决策的质量和可信度。

所有代码将遵循 Casevo 项目的编码规范, 确保与现有框架 (v0.3.19) 完全兼容。同时, 我们将提供详细的 API 文档和使用示例, 方便其他研究者使用和扩展我们的工作。

3.6 实验验证

为了全面评估优化方案的效果, 我们将设计一系列严谨的对比实验。实验方案遵循科学实验的基本原则, 采用控制变量法, 确保结果的可靠性和可重复性。

实验场景设计。我们将构建三个具有代表性的社会模拟场景, 每个场景针对不同的决策挑战。第一个是选举投票场景, 该场景基于 2020 年美国总统大选的辩论内容, 包含六轮辩论的完整文本资料。我们将配置 101 个具有不同政治倾向的选民智能体, 按照 Pew Research Center 的政治类型学分为九个类别。智能体在小世界网络中相互连接, 网络平均度数为 6, 聚类系数约为 0.3, 模拟真实的社交关系。每轮辩论后, 智能体将观看辩论内容、与邻居讨论、进行反思, 最终投票。我们将记录每轮的投票变化、意见演化轨迹以及网络中的信息传播路径。

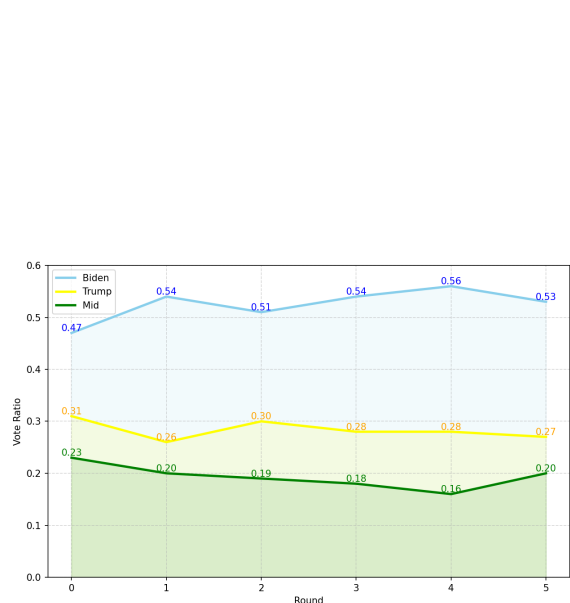
第二个是资源分配场景, 模拟 50 个智能体在资源受限情况下的协商过程。总资源量固定为 1000 单位, 每个智能体有不同的资源需求 (范围 15-30 单位) 和优先级权重。智能体需要通过多轮协商来达成分配方案, 既要满足自身需求, 也要考虑整体公平性。我们将评估协商轮次、达成共识的速度以及最终分配方案的公平性指标 (基尼系数、方差等)。

第三个是信息传播场景, 研究虚假信息在社交网络中的扩散动力学。我们构建一个包含 200 个节点的无标度网络, 少数节点 (10%) 初始接收到虚假信息。智能体需要根据信息来源的可信度、内容的逻辑一致性以及与自身认知的匹配度来判断信息真伪。我们将追踪虚假信息的传播范围、传播速度以及不同智能体决策机制对抑制虚假信息的有效性。

实验参数配置。对于每个场景, 我们将设置详细的实验参数。在选举场景中, LLM 温度参数设为 0.7 (平衡创造性和稳定性), 记忆检索返回前 5 条最相关记忆, 反思触发阈值为置信度低于 0.6。在资源分配场景中, 协商最大轮次设为 10 轮, 每轮允许智能体修改自己的需求提议, 收敛判定标准为连续两轮所有智能体的提议变化小于 5%。在信息传播场景中, 信息传播概率与边权重成正比 (范围 0.3-0.7), 智能体判断阈值根据个体特征有所差异。

基线方法与对比实验。我们设置三组对照实验: 基线组使用原始 Casevo 框架的 Chain of Thought 决策机制; 优化组 A 仅应用 Tree of Thought 多层次推理; 优化组 B 综合应用所有优化方案 (ToT + 增强记忆 + 动态反思 + 协同决策)。每组实验独立运行 10 次, 使用不同的随机种子, 以排除随机因素的影响。我们将采集每次运行的完整

日志，包括决策轨迹、记忆检索记录、推理过程以及最终结果。



(a) 选举投票结果变化趋势



(b) 选民意见词云分析

图 6: Casevo 框架选举模拟实验结果（引自原论文 [1]）。左图展示了六轮辩论中选民对两位候选人支持度的动态变化，可以观察到 Biden 的支持率稳步上升，Trump 的支持率有所波动，中立选民比例逐渐减少。右图通过词云展示选民对候选人的意见分布，Biden 相关的词汇以“支持”、“赞同”等正面词汇为主，而 Trump 相关的词汇则呈现正负两极分化的特点。

对比实验将采用控制变量法，基线方法使用原始 Casevo 框架的决策机制，优化方法则集成我们提出的各项改进。评估指标涵盖四个维度：决策质量（准确率、一致性、合理性）、推理能力（深度、多样性、连贯性）、计算效率（响应时间、内存占用、可扩展性）以及社会效应（群体共识度、意见分化程度、社会稳定性）。通过多维度的量化评估，我们能够全面了解优化方案的优势和局限，为后续改进提供依据。

4 预期成果

本研究预期在以下三个方面产出高质量的成果。

优化后的 Casevo 框架代码。我们将开发一套完整的决策能力优化模块，包括增强型思维链、优化的记忆管理、协同决策和决策质量评估等核心组件。所有代码将开源发布，并确保与现有 Casevo 框架完全兼容。我们还将提供详细的 API 文档和使用教程，帮助研究者快速上手。代码的开发将严格遵循软件工程规范，包括单元测试、代码审查和版本控制，以保证代码的质量和可维护性。

详细实验报告。实验报告将包含完整的实验设计说明、数据采集过程、对比实验结

果以及深入的案例分析。我们将使用丰富的图表来可视化实验数据，使得结果更加直观易懂。报告还将讨论实验过程中遇到的问题和挑战，以及我们是如何解决这些问题的。这些经验和教训对于后续研究具有重要的参考价值。

性能对比分析。我们将提供全面的性能评估报告，从决策质量、推理能力、计算效率和社会效应四个维度对优化方案进行量化分析。通过与基线方法的详细对比，我们将明确指出各项优化的具体贡献度，以及在不同场景下的适用性。此外，我们还将进行敏感性分析，探讨关键参数（如推理深度、记忆容量、协商轮次等）对系统性能的影响，为参数调优提供指导。

5 小组成员及分工

本研究由两名成员组成，分工明确、协作紧密。

姓名	角色	主要职责
王宇东	组长	负责项目整体规划与协调，深入开展理论研究，设计优化算法，撰写研究报告和学术论文，确保研究的学术质量和创新性
陈文远	组员	负责代码实现与系统集成，设计和执行实验，进行数据分析与可视化，协助报告撰写，保证研究成果的技术质量和可复现性

两位成员将采用敏捷开发的协作模式，定期召开小组会议讨论研究进展和遇到的技术难题。所有代码通过 Git 进行版本管理，确保开发过程的可追溯性。在研究过程中，双方将相互审阅对方的工作成果，通过同行评审机制提高研究质量。对于重要的技术决策和理论创新，两位成员将共同讨论并达成一致意见后再推进实施。

6 时间规划

本研究计划在 10 周内完成，具体安排如下。

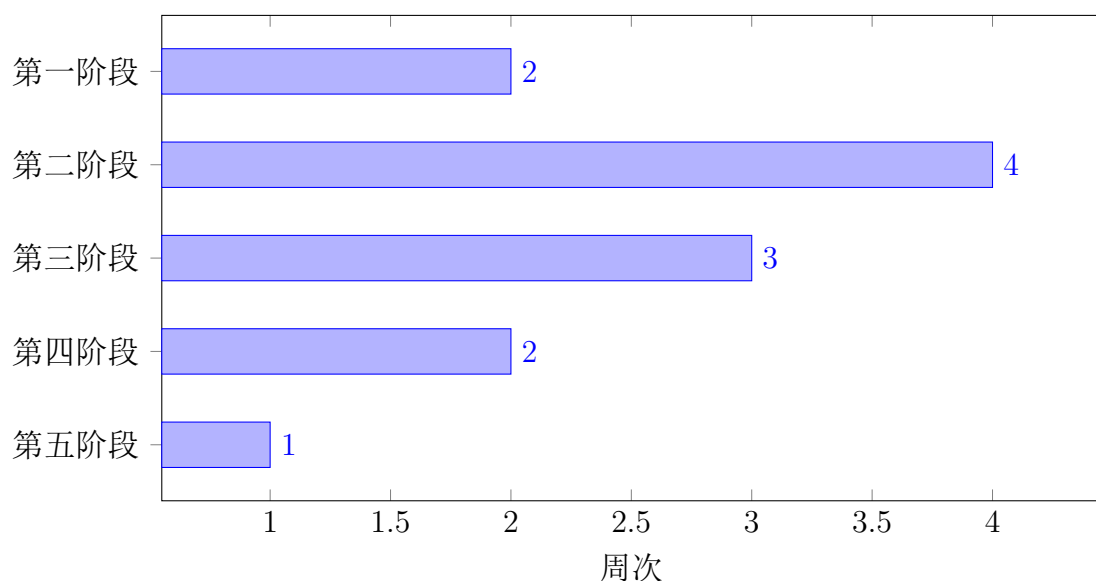


图 7: 研究时间安排甘特图

阶段	周期	主要任务
第一阶段	第6-7周	文献调研与理论研究，深入阅读相关论文，分析现有方法的优缺点；完成技术方案设计，确定具体的优化策略和实现路径
第二阶段	第8-11周	代码实现与模块开发，完成增强型思维链、优化记忆管理等核心模块的编码工作；进行单元测试，确保各模块功能正确
第三阶段	第12-14周	实验设计与数据采集，构建实验场景，配置参数，运行基线实验和优化实验；进行初步数据分析，验证方案可行性
第四阶段	第15-16周	对比实验与性能评估，完成所有计划实验，收集完整数据；进行深入的结果分析，总结优化方案的优势和不足
第五阶段	第17周	报告撰写与成果整理，完成实验报告和代码文档的编写；准备成果展示材料，包括幻灯片和演示视频

整个研究过程将采用迭代式推进的方式，每个阶段结束时进行阶段性总结和评审。如果某一阶段的工作未能达到预期目标，我们将及时调整计划，确保最终成果的质量。

7 参考文献

参考文献

- [1] Jiang Z, Shi Y, Li M, et al. Casevo: A Cognitive Agents and Social Evolution Simulator[J]. arXiv preprint arXiv:2412.19498, 2024.
- [2] Wei J, Wang X, Schuurmans D, et al. Chain-of-thought prompting elicits reasoning in large language models[J]. Advances in Neural Information Processing Systems, 2022, 35: 24824-24837.
- [3] Yao S, Yu D, Zhao J, et al. Tree of thoughts: Deliberate problem solving with large language models[J]. Advances in Neural Information Processing Systems, 2024, 36.
- [4] Park J S, O'Brien J, Cai C J, et al. Generative agents: Interactive simulacra of human behavior[C]//Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology. 2023: 1-22.
- [5] Wilensky U, Rand W. An introduction to agent-based modeling: modeling natural, social, and engineered complex systems with NetLogo[M]. MIT Press, 2015.
- [6] Axelrod R. The complexity of cooperation: Agent-based models of competition and collaboration[M]. Princeton University Press, 1997.
- [7] Bonabeau E. Agent-based modeling: Methods and techniques for simulating human systems[J]. Proceedings of the National Academy of Sciences, 2002, 99(suppl_3): 7280-7287.
- [8] Mesa: Agent-based modeling in Python 3+. <https://github.com/projectmesa/mesa>
- [9] ChromaDB: The AI-native open-source embedding database. <https://www.trychroma.com/>