# 构建基于 LLM 的五子棋 AI 系统：策略选择、位置评分与强化学习探索

**成员分工:**

王辉 2211110796

*本组只有我一个人，负责 PPT、课堂汇报、全部代码，测试实验、撰写报告等所有相关的全部内容

**摘 要** 近年来，大型语言模型（LLM）在自然语言处理（NLP）领域取得了显著进展，展现出强大的生成、理解和推理能力，并在教育、智能决策、游戏等多领域有广泛应用。然而，在策略性较强的五子棋游戏中，如何有效利用 LLM 进行合理策略规划和决策仍面临挑战。本研究旨在开发一种基于 LLM 的五子棋智能系统，模拟人类学习下棋过程，使其能理解和运用五子棋策略与逻辑，做出合理有效决策。研究方法包括让模型“读懂棋盘”、读懂规则”、“策略选择”、“位置评分”并通过自我对弈和强化学习提升下棋水平。实验结果表明，该方法在五子棋落子位置选择上成效显著，解决了语言模型输出非法位置问题，并通过并行位置评分技术大幅缩短流程时间，经大规模自我对弈训练后，模型下五子棋能力大幅提升。

## 1. 研究背景

近年来，大型语言模型在自然语言处理领域取得了显著进展，其强大的生成、理解和推理能力使其在多个任务中表现出色。例如，ChatGPT 等模型通过预训练和微调，能够处理复杂的语言任务，并在少样本甚至零样本的情况下展现出卓越的性能。此外，LLM 还被广泛应用于教育、智能决策、游戏等领域，显示出其广泛的应用潜力。在教育领域，LLM 可以作为教育智能体，支持个性化学习，帮助学生提高阅读、写作等多方面的知识技能。在智能决策方面，LLM 能够通过分析大量数据，为决策者提供有价值的建议和方案。在游戏领域，LLM 也展现出其独特的魅力，能够生成有趣的游戏内容和互动体验。

然而，尽管 LLM 在许多任务中表现出色，其在特定复杂任务中的应用仍面临挑战。例如，在五子棋这类策略性较强的游戏中，如何有效利用 LLM 进行合理的策略规划和决策是一个值得探索的问题。五子棋作为一种经典的策略性棋类游戏，因其简洁的规则和深奥的策略深度而深受玩家喜爱。传统的搜索算法如穷举法在有限计算资源下表现不佳，而基于机器学习的方法虽然强大但训练和预测效率低。因此，如何将 LLM 的优势与深度学习、强化学习等技术相结合，设计出高效且准确的五子棋 AI，是当前研究的一个重要方向。

## 2. 研究目的

本研究旨在开发一种基于大型语言模型（LLM）的五子棋智能系统，使其能够像人类棋手理解和运用五子棋的策略与逻辑，从而在对局中做出合理而有效的决策。研究的核心在于模拟人类学习下棋的过程，关键是“如何根据当前棋局选择合适的下棋策略和分析逻辑”。

## 3. 研究方法

### 3.1 整体思路：

首先让模型“读懂棋盘”，通过输入棋盘信息，使其能够准确识别出当前棋盘的状态，包括己方和对方的棋子位置、棋盘的边界等关键信息，为后续的决策提供基础；接着让模型“读懂规则”，输入五子棋的基本规则，如棋子的摆放方式、获胜条件等，确保模型在对局中遵循规则进行操作；然后让模型“了解下棋策略”，输入多种不同的下棋策略，如先手布局、防守反击、连珠进攻等，使模型能够根据不同的对局情况灵活运用相应的策略；再让模型“分析棋局”，输入各种分析逻辑，如对局势的判断、对对手意图的推测、对下一步棋的预判等，培养模型的棋局分析能力；之后让模型通过“尝试下棋”进行自我对弈（self-play），在不断的对局实践中积累经验，优化决策；最后通过“在实战中提升”，运用强化学习的方法，让模型在与人类棋手或其他智能系统的对局中不断学习和进步，逐步提升其下棋水平和策略运用能力，最终达到高水平的五子棋对弈能力。

### 3.2 代码编写逻辑

整体代码编写的逻辑主要分为 5 个方面，包括 Prompt 设计、策略与分析逻辑选择、局部位置评分、自我对弈、奖励模型与强化学习，本文就逐个介绍各部分的具体内容。

#### 3.2.1 Prompt 设计

为了使大型语言模型能够更准确地模拟人类棋手在五子棋中的决策过程，我们设计了一个通用的提示模板。该模板旨在通过输入当前棋盘的状态信息、五子棋的基本规则、下棋策略以及分析逻辑等关键要素，来还原人类棋手在对局中的思考过程，具体的 Prompt 模板如图 1 所示。通过这种方式，我们希望语言模型能够更好地理解和模拟人类棋手的决策逻辑，从而在五子棋对局中做出更为合理和高效的棋步选择。

**Prompt-1**

"You are an outstanding Gomoku player. These are the basic rules of Gomoku: {rule}. Later, I will need your help to play Gomoku.

I will be playing on a board of size {self.board_size}*{self.board_size}, and I will provide an array representing the state of the board from left to right, line by line, where {self.player_id} indicates my pieces, {-self.player_id} indicates my opponent's pieces, and 0 indicates empty spaces.

Please use the {think} strategy to analyze the behavior of both me and my opponent, and use the {relationship} to understand the current situation. Please ensure not to place a piece on an already occupied position. Remember, only choose the one you consider the best from the {zero_position} positions!"

图 1　用于进行落子位置思考的 prompt 模板

### 3.2.2 策略与分析逻辑选择

在五子棋对弈中，人类棋手常借助五子棋棋谱与策略来选取更具胜算的落子点，并且需挑选合适的分析方式以确定如何运用特定策略。本研究受此启发，旨在模拟人类棋手的思考流程，收集并学习常见的五子棋策略及分析逻辑，以便于在面对棋局时能精准地挑选出适宜的策略与逻辑进行思考。具体而言，本研究共收集了 52 种常见的下棋策略，涵盖基本战术、防守策略、进攻策略以及开局方法这四个关键部分；同时，还收集了 9 种分析逻辑，诸如因果关系、条件关系、比较关系等。在思考过程中，大语言模型会从所收集的策略与逻辑中分别选取 1 种进行深入思考，。为了进一步提升大语言模型分析的精准度，本研究还会将五子棋规则以及当前棋局的具体信息一并融入思考过程的 Prompt 中并输出认为最合理的下棋位置，其具体流程如图 2 所示。

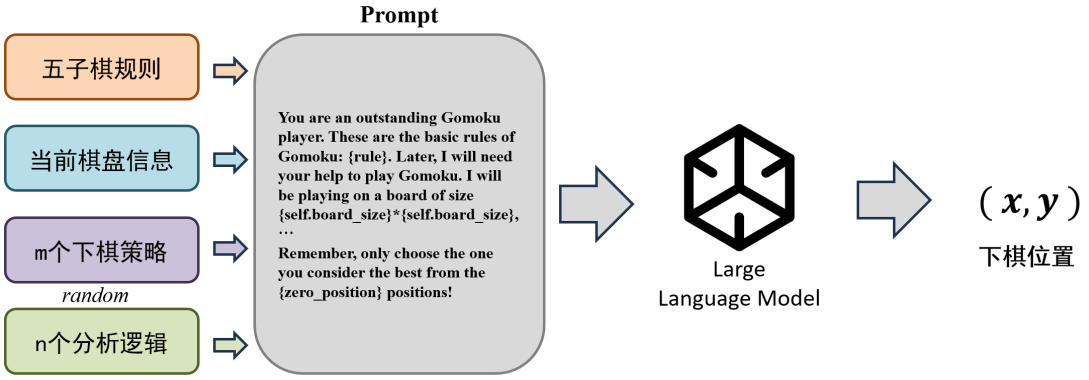


图 2　策略与分析逻辑选择融入思考过程并输出下棋位置的整体流程图

### 3.2.3 局部位置评分

在五子棋对弈过程中，大语言模型在选择落子位置时，常会遇到选择非法位置的问题，尤其是在已有棋子的位置落子。为了解决这一问题，常见的方法包括循环思考直至找到合法位置，或者提前将非法位置信息纳入 Prompt 中。实验结果表明，虽然这两种方法能在棋局早期有效缓解选择非法位置的问题，但随着棋盘上棋子数量的

增加，它们仍无法彻底克服该问题，导致思考过程陷入无限循环，程序出现卡死现象。

针对这一难题，本研究精心设计了一种局部位置评分方法。该方法通过对待选落子位置及其局部邻居所有合法位置逐一进行评分，从而精准选出评分最高的合法位置作为最终的落子点。这种方法不仅能够充分利用思考过程，还能有效确保最终落子位置的合法性。具体实施时，我们将待选落子位置及其周围的一阶邻居视为待评分位置，首先判断这些位置是否合法（若均不合法，则扩展至二阶邻居，依此类推），然后再次借助大语言模型对各合法位置进行评分。最终，得分最高的合法位置将被确定为最终的落子位置。该方法的详细流程如图 3 所示。

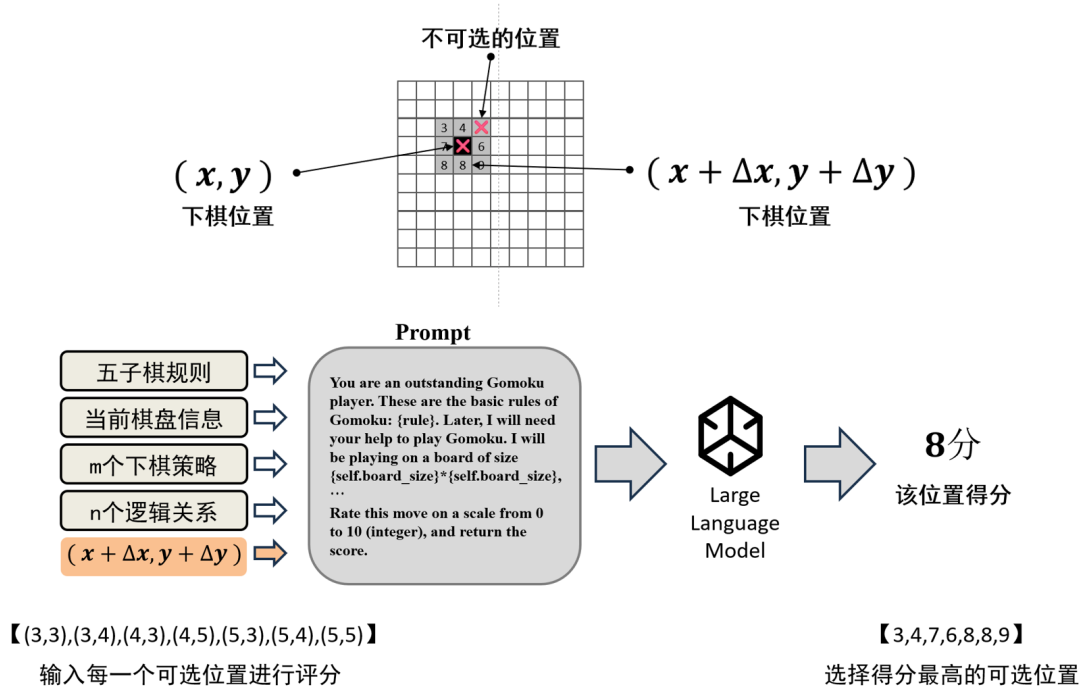


图 3 局部位置评分的整体流程图

### 3.2.4 自我对弈

尽管在 2.2 节中，本研究广泛收集了众多五子棋策略与分析逻辑，但需注意的是，这些策略和分析逻辑之间存在相互矛盾的情况，或者部分策略仅适用于特定棋局。因此，如何依据当前棋局精准地挑选出最为合适的策略与分析逻辑，成为了一个至关重要的问题。以 alpha-go 或 alpha-zero 等机器学习模型下棋方法为例，它们通常借助 Self-Play 方法，即模型自身与另一个自身不断进行棋局对弈，并利用最终棋局结果来提升模型的自我下棋水平，其中胜者的奖励设定为 100，负者的奖励为-100，平局的奖励为 0。本研究汲取了这一思路，采用自我对弈的方法，在持续的实战过程中锻炼并提升策略与分析逻辑的选择能力，以实现为当前棋局精准选择最优策略及对应分析逻辑的目标。具体实施时，本研究设置了两个棋手 Agent，分别代表黑棋和白棋，两者均运用同一策略选择模型来为各自所处的棋局挑选策略和分析逻辑，并以交替进行的方式持续

开展对弈，直至棋局结束。

### 3.2.5 奖励模型与强化学习

与常规的强化学习训练模型不同，使用语言模型进行对弈时，完成一个完整的棋局（从开始到棋局结束）需要大量时间，这严重减慢了下棋能力提升的效率。为此，本研究希望在棋局对弈的中间过程加入逐轮的奖励。具体来说，设计了一个专用于评价棋局的 Agent，该 Agent 通过对当前棋局两位选手进行评分，给出每一位棋手的胜率，并以此作为奖励来更新策略选择模型。

本研究使用深度 Q 网络（DQN）在不断对弈的过程中进行强化学习训练，以得到每个棋局对应的动作值，进而选择最优的策略和分析逻辑。具体来说，表示下棋前的棋盘状态，表示 agent 选择的策略动作，表示该步棋得到的奖励，表示对应的动作值函数，表示网络的参数，其具体流程如图 4 所示。具体的网络结构为三层 MLP。输入尺寸为 15*15（棋盘的所有位置），输出尺寸为 52*9（策略与分析逻辑的排列组合）。

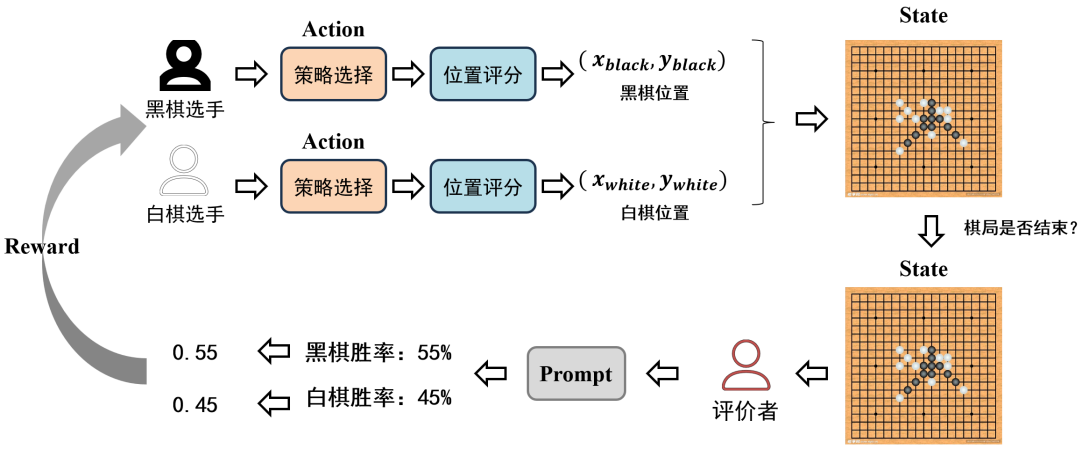


图 4 强化学习自我对弈训练的整体流程图

## 3.3 算法优化

### 3.3.1 并行位置评分

在 2.3 节中，局部位置评分需对待选落子位置及其局部邻居的所有合法位置逐一进行评分，此过程通常极为耗时，平均速度仅为“150 秒/每步棋”，致使强化学习过程进展缓慢。为攻克这一难题，本研究精心设计了一个并行框架，旨在实现所有局部区域中所有合法位置的并行评分。具体实施时，借助 Ray 构建并行框架，为每一个位置配备一个单独的大语言模型进行位置评分，各模型相互独立且同步运行。最终，将各模型输出的结果进行汇总排序，从中筛选出评分最高的位置作为最终的落子位置。

### 3.3.2 “状态-动作-奖励”库

在借助语言模型开展五子棋对弈研究的过程中，为了通过多次自我对弈来提升模型的下棋水平，需要频繁且并行地调用 API。然而，在实际操作中，由于 API 调用存在稳定性不足的问题，时常会出现程序在中途意外停止的情况，这不仅打断了对弈进程，还导致之前已完成的 rollout 过程中的所有宝贵信息瞬间丢失，进而使得研究者不得不重新启动程序，从头开始对弈，极大地降低了研究效率，也增加了研究成本。

为了解决这一棘手问题，避免因中途停止而导致的信息丢失，本研究精心设计并实现了一个创新性的架构。该架构的核心功能是能够实时、准确地保存与加载所有已进行的 rollout 过程中所产生的“状态-动作-奖励”对，将其系统地存储在一个专门构建的数据库中。这个数据库不仅作为一个信息存储库，完整地记录了每一次对弈过程中的关键数据，而且具有高度的灵活性和实用性，可以单独用于模型的重新训练。通过这种方式，即便在 API 调用过程中出现意外中断，研究者也能够迅速地从数据库中恢复之前的状态，无缝衔接地继续进行对弈和模型训练工作，从而极大地提高了研究的连续性和效率，确保了研究工作的稳定推进。

### 3.3.3 可视化

为了使语言模型在五子棋对弈中的决策过程以及与人类玩家的交互能够以更加直观、清晰且易于理解的方式呈现，本研究设计并增加了一个专门的可视化模块。该模块采用了与经典五子棋游戏（Gomuku）相似的展示形式，能够生动、形象地将对弈过程以及人机交互的每一个细节进行可视化呈现。通过这一模块，观察者可以清晰地看到每一步棋的落子位置、策略选择以及局势的动态变化，从而更加深入地理解语言模型的决策逻辑和对弈策略。具体的可视化结果如图 5 所示。为了提供更为全面和深入的参考，本研究还将整局对弈过程的可视化结果完整地整理并收录于补充材料中。

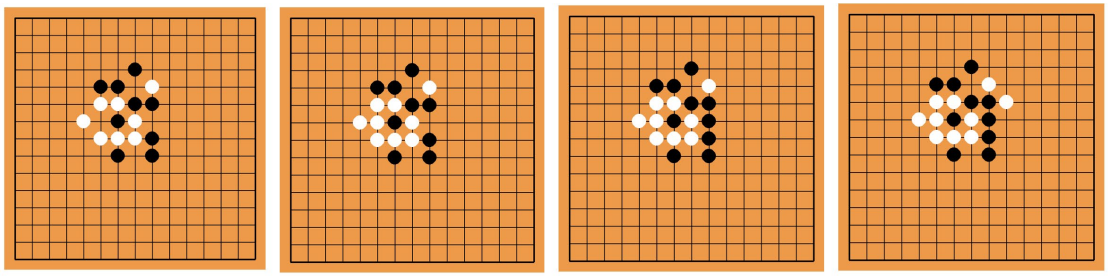


图 5 对弈过程的可视化效果

## 4. 实验结果

通过精策略与分析逻辑选择机制，语言模型在五子棋落子位置的选择上取得了显著成效，相较于 Zero-shot，Few-shot 或直接 Chain of Thought 方法，其表现有了明显

的提升。此外，与传统方法相比，本研究引入的局部位置评分模块从根本上解决了语言模型在下棋过程中频繁输出非法位置，导致棋局无法正常进行的问题，使得语言模型能够顺畅地开展自我对弈。同时，采用并行位置评分技术，将整体流程的平均速度从“150秒/每步棋”大幅缩短至“28秒/每步棋”，在确保性能稳定不变的基础上，实现了速度约5倍的飞跃式提升。在实验部分，本研究充分利用了24次CPU核的强大计算能力，通过自我对弈的方式进行了1046次对弈，以此来训练深度Q网络。经过如此大规模的训练，模型的下五子棋能力相较于未训练时有了质的飞跃，其决策更加精准，对弈水平显著提高，充分展现了本研究方法的有效性和实用性。

## 5. 性能评估

为了合理评估本研究中语言模型在五子棋对弈中的性能，我们采用了两种评估方法：人类定性评估和与AlphaZero对弈的存活步数评估。人类定性评估主要依赖于人类玩家对模型下棋策略、决策合理性和整体表现的主观评价，这种方法能够从人类玩家的角度直观地反映模型的对弈水平。定量评估则通过让模型与Alpha-zero（根据已有研究得到的模型）进行对弈，记录模型在每局对弈中的存活步数来衡量其性能，共进行了8局对弈以确保评估结果的可靠性。这种定量评估方法能够客观地反映模型在实际对弈中的持久性和竞争力。详细的评估结果如表1所示，详细结果如表1所示。

表 1 不同方法的定性与定量评估结果

| | 是否能流畅的完成棋局 | 由人类评估的下棋水平 | 与alpha-zero对弈的平均存活步数 |
|---|---|---|---|
| Zero-shot | 否 | 很差 | - |
| Few-shot | 否 | 很差 | 5 |
| Chain-of-thought | 否 | 差 | 6 |
| 随机策略选择 | 否 | 差 | 7 |
| 随机策略选择+局部位置评分 | 是 | 差 | 7 |
| 训练100次对弈 | 是 | 一般 | 9 |
| 训练500次对弈 | 是 | 一般 | 11 |
| 训练1000次对弈 | 是 | 中等 | 12 |

## 6. 未来工作

尽管本研究的方法成功实现了使用语言模型下五子棋，并通过强化学习训练出了一个具有一定水平的五子棋棋手，但该方法仍存在一些亟待解决的问题。首先，自我对弈过程耗时过长，这使得模型难以快速掌握一些基本的下棋法则，需要通过大量的对弈才能逐步提升下棋能力。其次，在策略与分析逻辑的选择上，为了简化推理过程，本研究每次仅选择一个策略和一个分析逻辑进行思考，这种做法在一定程度上限制了模型分析棋局的全面性和深度。

在未来的研究中，我们计划采用多组“策略+分析逻辑”结合的方式，以更全面地评估和选择最佳的下棋策略。此外，我们将探索使用深度确定性策略梯度（DDPG）等先进的深度强化学习模型，以进一步提升模型在复杂棋局中的思考能力。同时，我们还将尝试利用 AlphaZero 的结果，引导语言模型朝着最正确的落子位置方向思考，从而加快模型能力提升的速度。

## 7. 结论

本研究成功开发了一种基于大型语言模型的五子棋智能系统，通过让模型读懂棋盘、规则，了解下棋策略，分析棋局，策略选择、位置评分以及采用自我对弈和强化学习等方法，使模型在五子棋对弈中能够像人类棋手一样进行合理而有效的决策。实验结果表明，该系统在落子位置选择上取得了显著成效，有效解决了语言模型在下棋过程中输出非法位置的问题，并大幅提升了决策速度和对弈水平。本研究不仅为利用大型语言模型解决复杂策略游戏问题提供了新的思路和方法，也为推动 LLM 在游戏等领域的应用拓展了可能性，具有重要的理论和实践意义。未来，我们将在多策略组合、先进深度强化学习模型应用以及利用 AlphaZero 结果引导等方面进一步优化和提升系统性能，以期实现更高水平的五子棋对弈能力。