LLM based Agent

1. **初始化**：

游戏规则：两名玩家可以选择成为“鹰”（Hawk）或“鸽”（Dove）。如果两只鸟都选择成为鹰，它们会互相争斗并都得到低分。如果一只鸟是鹰，另一只是鸽，鹰得到高分，鸽得到低分。如果两只鸟都是鸽，它们和平分享得分。

1. **模型输入**：

提示：“如果你选择成为鹰，而对方选择成为鸽，你将得到5分，对方得到1分。如果你们俩都选择成为鹰，你们都将得到0分。如果你们俩都选择成为鸽，你们都将得到3分。请进行推理，你会怎么做？”

1. **生成推理**：

LLM-Reasoner可能会推理：“为了最大化我的得分，我应该选择成为鹰，因为这样我要么得到5分，要么得到0分，这取决于对方的选择。”

1. **翻译推理**：

LLM-Translator将上述推理翻译成逻辑查询，例如：“如果我选择鹰，对方的选择是什么？”

1. **逻辑验证**：

Solver评估查询并根据游戏规则提供结果。

1. **错误检测**：

如果Solver发现推理与游戏规则不符，它会记录下来。

1. **反馈循环**：

如果有错误，Solver将反馈给LLM-Reasoner，例如：“你的推理可能有误，因为选择成为鸽可能会得到更稳定的得分。”

1. **自我修正**：

LLM-Reasoner根据反馈重新评估其推理，并可能决定改变策略，选择成为鸽。

1. **输出结果**：

最终推理：“考虑到稳定性，我选择成为鸽，以确保至少得到3分。”