MATPO: Multi-Agent Tool-Integrated Policy Optimization

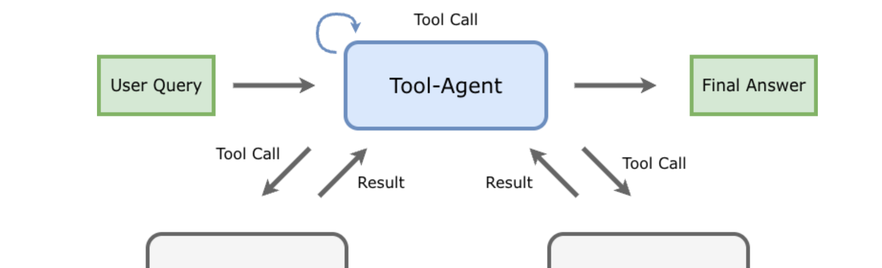
Zhanfeng Mo, Xingxuan Li, Yuntao Chen, and Lidong Bing

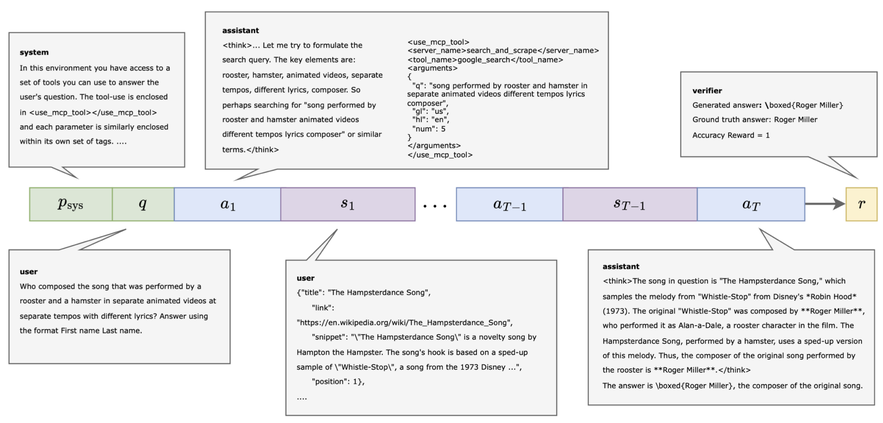
Shanda Group

Problem Setup

Advanced AI agent systems require **l**arge **l**anguage **m**odels (**LLMs**) to accomplish complicate tasks via sophisticated multi-turn **t**ool-**i**ntegrated **p**lanning (**TIP**).

A common approach is to train an LLM using reinforcement learning (RL) within a **single-agent** TIP framework, where a single LLM iteratively performs planning, tool-calling, and reasoning based on tool responses to solve complex tasks.





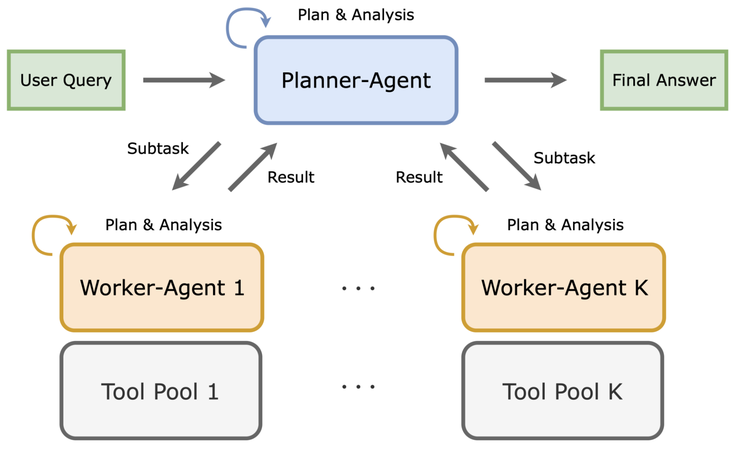
Motivation

Limitations of single-agent framework:

High token cost: tool-responses consume excessive context tokens, making long-horizon multi-turn TIP infeasible.

Noisy tool-response: tool-response are often noisy, degrading the planning and reasoning quality of subsequent actions.

Multi-agent framework addresses these issues by introducing a planner-agent and work-agents: **planner agent** coordinates the workflow and decomposes tasks into subtasks; **worker agents** execute subtasks and return compact TIP summaries.



Contributions

Q1: How to perform **RL training** within **multi-agent** TIP system?

Q2: Can **a single LLM** be used to perform multiple agent roles?

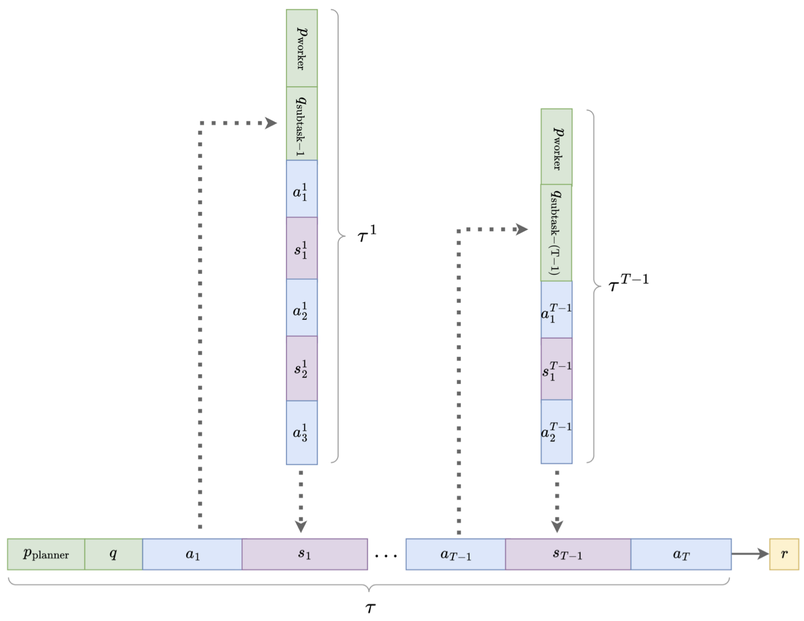
We establish **M**ulti-**A**gent **T**ool-Integrated **P**olicy **O**ptimization (**MATPO**), a principled algorithm to perform end-to-end **multi-agent-in-one-LLM** RL in multi-agent TIP systems.

$$J_{\text{MATPO}}(\pi_\theta) \triangleq \mathbb{E}_{\{\tau_i\}\sim(\pi_{\theta_{\text{old}}},\text{Tool})}^{q\sim\mathcal{D}} \left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{\sum_{t=0}^{T_i}|\tau_i^t|}\sum_{t=0}^{T_i}R_i^{\text{clip}}\right]$$

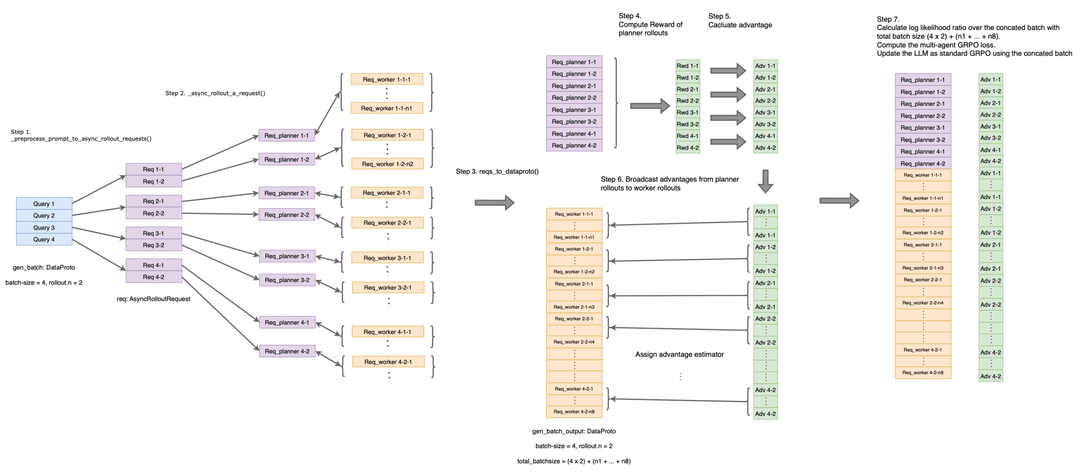
$$R_i^{\text{clip}} \triangleq \min(R_{i,t}\hat{A}_{i,t}, \text{clip}(R_{i,t}, 1-\varepsilon, 1+\varepsilon)\hat{A}_{i,t})$$

$$\hat{A}_{i,t} \triangleq (r(\tau_i) - \text{mean}(\{r(\tau_i)\}_{i=1}^{G}))/\text{std}(\{r(\tau_i)\}_{i=1}^{G})$$

$$R_{i,t} \triangleq \begin{cases} \sum_{j=1}^{T_i} \dfrac{\pi_{\theta_{\text{old}}}(a_j^t|[p_{\text{planner}}, q, a_1, s_1, .., s_{j-1}])}{\pi_\theta(a_j^t|[p_{\text{planner}}, q, a_1, s_1, .., s_{j-1}])}, \\ \sum_{j=1}^{T_{i,t}} \dfrac{\pi_{\theta_{\text{old}}}(a_j^t|[p_{\text{worker}}, q_{\text{subtask}-t}, a_1^t, s_1^t, .., s_{j-1}^t])}{\pi_\theta(a_j^t|[p_{\text{worker}}, q_{\text{subtask}-t}, a_1^t, s_1^t, .., s_{j-1}^t])}, \end{cases}$$



Our proposed MATPO is readily deployable within existing single-agent RL frameworks (e.g., VeRL) and incurs no additional LLM hosting overhead.



In the deep-research setting, our MATPO consistently outperforms single-agent baselines across three datasets.