MAR: Multi-Agent Reflexion Improves Reasoning Abilities in LLMs

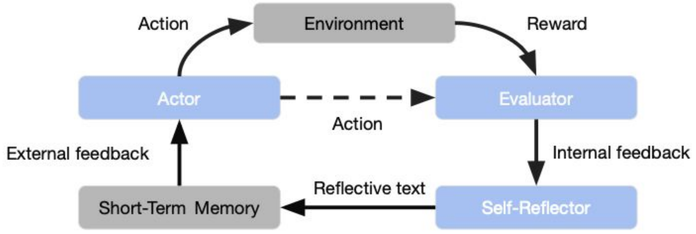
MICHIGAN ENGINEERING
UNIVERSITY OF MICHIGAN

Vivi De La Rue, Onat Ozer, Daniel Dosti, Yuchen Wang, Honghao Zhang, Grace Wu

Introduction

- ***Reflexion*** (Shinn et al., 2023) enables language models to revise their reasoning through feedback.
- However, ***Reflexion*** often fails due to self-reinforcing errors and limited reasoning diversity.
- We propose ***Multi-Agent Reflexion (MAR)*** which improves upon standard Reflexion, achieving an **3%** gain on HotPotQA and a **6.2%** improvement on HumanEval.

Reflexion: Overview



Reflexion: Limitations

Reflexion improves reasoning through self-reflection, but has:

- Confirmation bias
- Repeated failed reasoning
- ***Degeneration-of-Thought*** (Liang et al., 2025)

Task:
```
def iscube(a: int) -> bool:
    '''returns True if this integer is a cube.'''
```

❌ **Output 1:**
```
return (a >= 0) and (round(a ** (1/3)) ** 3 == a)
```

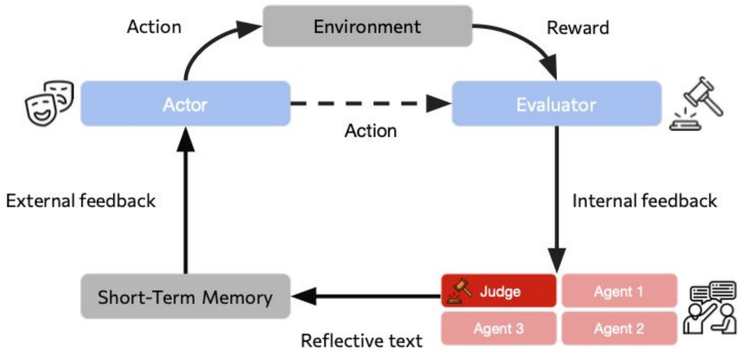
💭 **Reflection:** The condition `(a >= 0)` allows negative numbers to pass through.

❌ **Output 2:**
```
if a < 0:
    return False
return round(a ** (1/3)) ** 3 == a
```

⌛ TIMEOUT!

Incorrect! *Input (-1) is still a perfect cube*

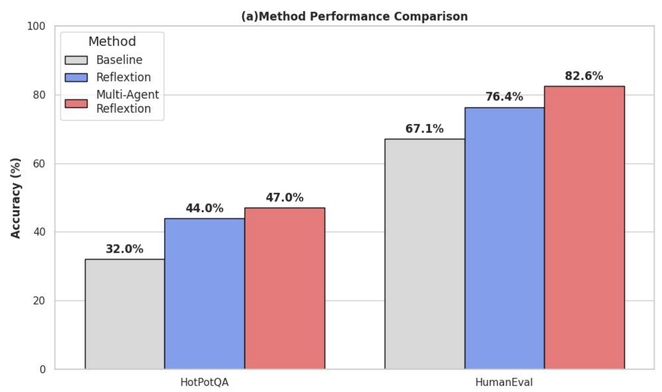
Multi-Agent Reflexion (MAR)



Persona Design

- Build distinct personas that *disagree by design,* encouraging diverse reasoning and avoiding stagnation of debates
  - Final Judge persona to guide, assess, and finalize multi-agent debates
- Encourage *"tit for tat"* disagreement rather than extreme adversarial behavior better reasoning

MAR Results



- *MAR improves HumanEval by **+6** points over Reflexion and **15.5** over baseline GPT-3.5.*
- *MAR improves with ReAct on HotPotQA by **+1** points over Reflexion and **+15** over baseline with GPT-3.5.*

Task:
```
def iscube(a: int) -> bool:
    '''returns True if this integer is a cube.'''
```

❌ **Output 1:**
```
return (a >= 0) and (round(a ** (1/3)) ** 3 == a)
```

Debater 1: (correct direction)
"Check if cube root is `int`."

Debater 2: (important correction)
"Account for negative roots."

Debater 3: (synthesizing correctly)
"Add this code to the return condition: `round(abs(a) ** (1/3)) ** 3 == abs(a)`*"*

Debate Judge:
*"The **bug** was misclassifying numbers as perfect cubes. The **fix** was to verify the cube of the number's absolute value."*

✅ **Output:**
```
def iscube(a: int) -> bool:
    if a == 0:
        return True
    return round(abs(a) ** (1/3)) ** 3 == abs(a)
```

Takeaways

Why MAR Outperforms Baselines and Reflexion

- **Breaks Single-Agent Blind Spots:** Multiple agents bring diverse reasoning styles, reducing repeated mistakes.
- **Catches More Errors:** Different personas (e.g., QA, senior dev, reviewer) surface different failure modes in the code.
- **Judge-Based Selection:** A final judge aggregates all the debaters into the best reflection, preventing convergence on a bad fix.
- **Explores Broader Solution Space:** MAR increases the chance that at least one agent finds the correct logic

References

1. Shinn, N.; Cassano, F.; Berman, E.; Gopinath, A.; Narasimhan, K.; and Yao, S. 2023. Reflexion: Language.Agents with Verbal Reinforcement Learning. arXiv preprint. arXiv:2303.11366.

2. Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J. B.; and Mordatch, I. 2023. Improving Factuality and Reasoning in Language Models through Multiagent Debate. arXiv preprint. arXiv:2305.14325.