Can Multi-turn Self-refined Single Agent LMs with Retrieval Solve Hard Coding Problems?

Md Tanzib Hosain[1,3,*], Md Kishor Morol[2,3]

[1]American International University-Bangladesh
[2]Cornell University
[3]EliteLab.AI
[*]Work done while working as a remote RA at QCRI.

Problem & Motivation

Challenge

- Existing coding benchmarks (HumanEval, MBPP) are **saturated** with >90% solve rates
- Need more challenging benchmarks for algorithmic reasoning
- Competitive programming requires sophisticated thinking

Current State

- Even **o1** achieves only **19.1%** pass@1 on ICPC problems
- Lack of comprehensive evaluation frameworks
- Missing official analysis and quality test suites

Research Question: Can we develop inference techniques that significantly improve LM performance on competitive programming while maintaining practical applicability?

ICPC Benchmark

📋 **Hard Coding Dataset: 254** expert-written competitive programming problems

Problem Sources

Category	Problems#
WF & CF	167
Regional	87
Total	**254**

Rich Resources

- Official human-written analysis
- Reference C++ solutions
- Sample & synthesized unit tests
- Hidden test cases

Multi-turn Self-judge Framework

🔄 Architecture Overview: Problem → LLM → Knowledge Retrieval → Self-judge → Feedback Loop

Problem Sources

- **Episodic Retrieval**: Similar problems with solutions
- **Self-reflection**: Learning from execution feedback
- **Multi-turn Iteration**: Iterative refinement
- **Self-judge**: Unit test validation

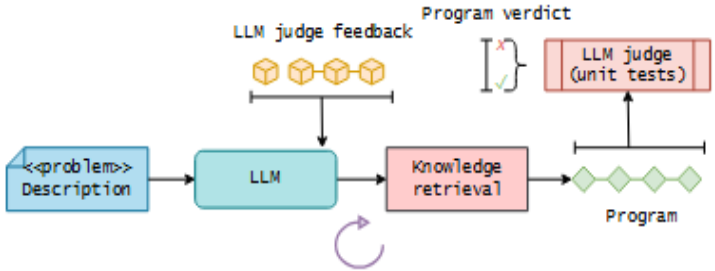
Inference Techniques

- Zero-shot Chain-of-Thought
- Few-shot prompting
- Semantic & Episodic retrieval
- Self-reflection
- Combined approaches

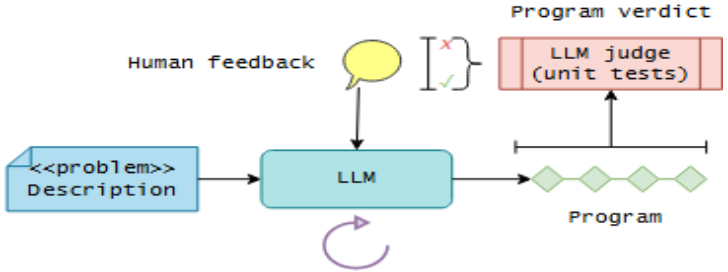
Multi-turn Self-judge Framework

🔄 Architecture Overview: Problem → LLM → Knowledge Retrieval → Self-judge → Feedback Loop

Knowledge Retrieval with Self-Reflection



Human Agent Interaction



Baseline Performance

Zero-shot Performance Across Models

Model	Pass@1
gpt-4	7.3
claude-3.5-sonnet	14.1
gpt-4o	14.2
qwen2.5-coder	14.8
athene-v2-chat	16.4
deepSeek-v3-chat	17.6
gemini-exp	18.3
o1	**19.1**

Key Observation: Even state-of-the-art models struggle with competitive programming, highlighting the need for advanced inference techniques.

Results

Inference Techniques

Inference technique	Model		
	gpt-4	gpt-4o	o1
zero_shot	7.3	14.2	19.1
brainstorm_then_select	8.6	16.9	21.7
few_shot	10.1	19.4	24.2
self_reflection	11.3	20.6	25.4
semantic_retrieval	12.4	22.1	27.3
semantic_retrieval + self_reflection	12.8	22.5	28.1
episodic_retrieval	13.2	23.3	29.0
semantic_retrieval + episodic_retrieval	14.5	24.4	29.8
semantic_retrieval + episodic_retrieval + self_reflection	16.4	27.1	33.2
episodic_retrieval + self_reflection	**24.3**	**38.4**	**42.2**

🎉 **Major Achievement: 120.94%** improvement over o1's baseline performance!

Human Agent Interaction Study

🤝 **Interactive Tutoring Setup:** Humans provide targeted guidance without revealing solutions

Interaction Rules

- **Allowed:** General concepts, sample walkthrough, high-level directions
- **Forbidden:** Exact algorithms, specific code fixes, detailed explanations

Results on 18 Not Solved Problems

Model	Final solve rate
gpt-4	0
gpt-4o	0
o1	0
o1 + interact	**94.4**

Insight: o1 solved 17/18 previously unsolvable problems with minimal human guidance, suggesting potential for automated feedback generation.

Error Analysis

Episodic Retrieval with Self-reflection's Error Distribution

Model	Wrong Ans.	TLE	MLE	Runtime	Syntax + Other
gpt-4	58.81	5.33	0	10.16	1.38
gpt-4o	28.95	25.06	0	6.83	0.77
o1	27.87	23.56	0	5.78	0.59

Key Observations

- Advanced models trade speed for accuracy
- Compilation errors are not the main issue
- Time limit exceeded indicates complex reasoning

Implications

- Models generate syntactically correct code
- Challenges lie in algorithmic understanding
- Need for efficiency optimization

Ablation Studies

Retrieval Query Optimization

Query	Pass@1
problem_description	28.5
problem_description + proposed_code_solution	**29.0**
problem_description + proposed_solution + code_solution	29.8

Problems# Retrieved Hyperparameter Tuning

Problems	Pass@1
$p = 1$	28.1
$p = 2$	**29.0**
$p = 3$	28.4

Iteration Analysis

Iterations	Pass@1
$i = 0$	21.3
$i = 1$	23.8
$i = 2$	**25.6**
$i = 3$	25.4

Optimal Configuration: p=2 problems, i=2 iterations for best performance

Key Insights & Analysis

🧠 Emergence of Self-reflection

Stronger models (o1, GPT-4o) show emergent ability to effectively use self-reflection, while weaker models benefit more from retrieval augmentation.

🔍 Retrieval vs. Memorization

Models actually use retrieved reasoning rather than memorizing - removing key solution components drops performance to 2.3%.

💭 Human Feedback Potential

o1's exceptional performance with minimal human guidance (94.4%) suggests automated feedback generation could unlock similar improvements.

⚖️ Evaluation Beyond Pass@n

Traditional execution success metrics may not fully capture model capabilities - need for more nuanced evaluation approaches (e.g.; Refine@n).

Future Work & Limitations

Current Limitations

- Focus on competitive programming only
- Not evaluated on software engineering tasks (SWE-bench)
- Accuracy-focused, not cost-optimized

Future Directions

- Automated human-level feedback generation (e.g.; Xolver)
- Extension to broader coding domains
- Cost-efficiency optimization
- Advanced evaluation metrics
- Integration with model internals

Research Opportunities

- Developing multi-agent techniques to generate corrective feedback automatically
- Mitigating computational cost while maintaining accuracy
- Creating evaluation frameworks beyond execution success
- Scaling to larger, more diverse problem sets

Conclusions

🏆 Major Contributions

- **ICPC Benchmark**: 254 comprehensive competitive programming problems with rich resources
- **Novel Framework**: Multi-turn self-judge with retrieval achieving 120.94% improvement
- **Human-AI Insights**: Demonstrating potential of guided interaction (94.4% solve rate)

🎯 Key Takeaways

- Episodic Retrieval with Self-Reflection is highly effective for complex reasoning
- Different models benefit from different techniques (Retrieval vs. Reflection)
- Human guidance unlocks significant potential in advanced models

🌟 Impact

This work provides a **foundation for advancing language models toward grounded, imaginative, and algorithmic thinking**, opening new research directions at the intersection of NLP and advanced problem-solving.

Thanks!

Do you have any questions?

20-42737-1@student.aiub.edu, mmorol@cornell.edu