LogicPro: Improving Complex Logical Reasoning via Program-Guided Learning

Jin Jiang[1,3], Yuchen Yan[2,3], Yang Liu[3], Jianing Wang[3], Shuai Peng[1], Mengdi Zhang[3], Xunliang Cai[3], Yixin Cao[4], Liangcai Gao[1]
[1]Peking University, [2]Zhejiang University, [3]Meituan Group, [4]Fudan University

ACL 2025 VIENNA JULY 27 - AUGUST 1

+ Inject code logic into text CoT
+ Synthesize complex reasoning data from LeetCode problems and test cases
+ Use code + test case execution to generate gold answers (outcome signal)
+ Extract intermediate variables to guide step-by-step reasoning (process signal)
+ Improves LLM performance on diverse reasoning benchmarks (BBH, LogicBench, GSM8K)

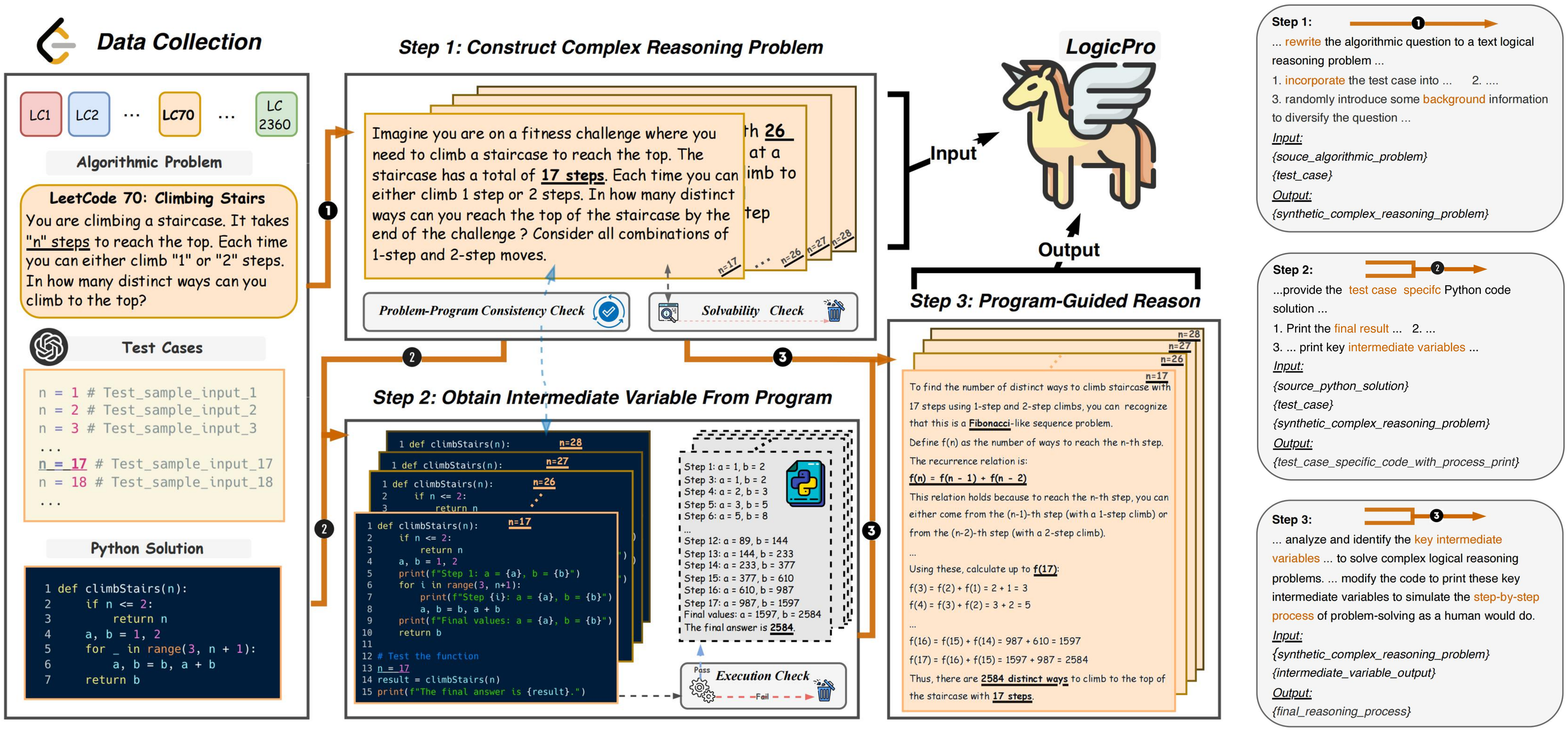
Resources & Contact Info

jiangjin@stu.pku.edu.cn

https://arxiv.org/pdf/2409.12929

https://github.com/jiangjin1999

Synthesize Data with Both Outcome and Process Supervision Signals !!!



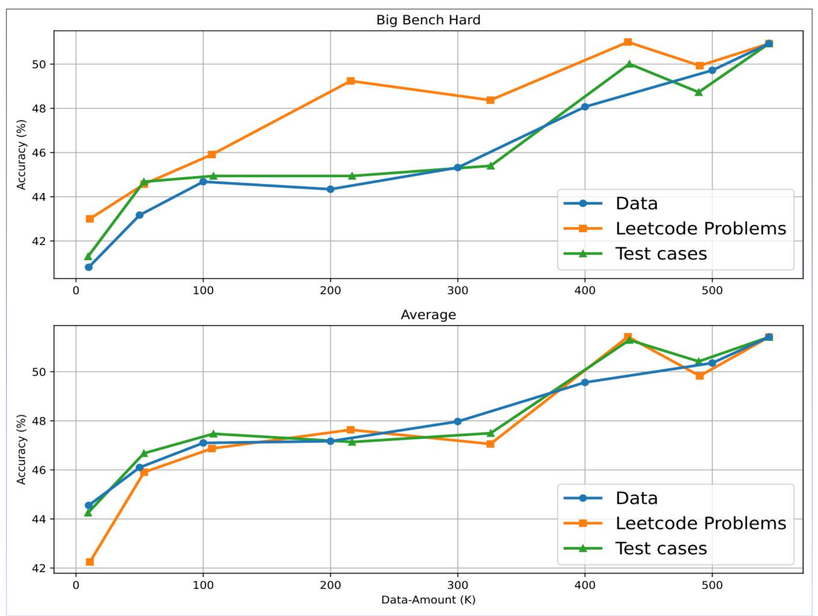
Step 1:
... rewrite the algorithmic question to a text logical reasoning problem ...
1. incorporate the test case into ... 2. ....
3. randomly introduce some background information to diversify the question ...
Input:
{souce_algorithmic_problem}
{test_case}
Output:
{synthetic_complex_reasoning_problem}

Step 2:
...provide the test case specifc Python code solution ...
1. Print the final result ... 2. ...
3. ... print key intermediate variables ...
Input:
{source_python_solution}
{test_case}
{synthetic_complex_reasoning_problem}
Output:
{test_case_specific_code_with_process_print}

Step 3:
... analyze and identify the key intermediate variables ... to solve complex logical reasoning problems. ... modify the code to print these key intermediate variables to simulate the step-by-step process of problem-solving as a human would do.
Input:
{synthetic_complex_reasoning_problem}
{intermediate_variable_output}
Output:
{final_reasoning_process}

Main Results: OOD Gains Across Benchmarks

Model	BBH[27]	LogicBench	DROP	ARLSAT	#Boar
Qwen2-7B-RuleTakers	45.4	59.1	65.7	16.5	42
Qwen2-7B-LogicNLI	43.3	71.3	67.4	17.8	45
Qwen2-7B-ProofWriter	40.8	68.6	64.3	17.0	36
Qwen2-7B-CLUTRR	43.0	72.0	64.0	17.0	**51**
Qwen2-7B-RuleBert	46.2	69.1	67.4	16.5	43
Qwen2-7B-LogicBench	44.7	*95.9	67.4	17.8	41
Qwen2-7B-FLD	42.0	69.5	**68.3**	14.8	34
Qwen2-7B-LogicPro (ours)	**50.9**	**73.5**	**68.3**	**19.1**	48
Llama3-8B-RuleTakers	38.5	59.9	65.9	12.6	**47**
Llama3-8B-LogicNLI	40.4	54.0	65.3	12.6	41
Llama3-8B-ProofWriter	37.2	62.1	66.4	**15.2**	31
Llama3-8B-CLUTRR	40.5	61.1	66.6	10.4	43
Llama3-8B-RuleBert	34.7	48.8	66.5	**15.2**	43
Llama3-8B-LogicBench	41.0	*93.5	66.2	10.9	38
Llama3-8B-FLD	35.7	67.8	61.2	13.5	39
Llama3-8B-LogicPro (ours)	**45.0**	**67.9**	**68.8**	**15.2**	44
Qwen2-72B-RuleTakers	61.3	72.4	76.6	19.6	61
Qwen2-72B-LogicNLI	61.7	80.7	77.0	21.3	60
Qwen2-72B-ProofWriter	61.8	75.5	77.2	16.5	55
Qwen2-72B-CLUTRR	68.1	79.0	78.4	24.4	61



Data Scaling Analysis

Scale on
- Data
- Leetcode Problems
- Test Cases

Ablation Study

- Source LeetCode
- without Inter-Var
- with Inter-Var

Qwen2-7B	BBH[27]	LogicBench	#Avg
Source_LeetCode	41.0	63.4	42.2
LogicPro$_{\text{w/o Inter-Var}}$	48.2$_{+7.2}$	69.6$_{+6.2}$	49.0$_{+6.8}$
LogicPro$_{\text{w. Inter-Var}}$	**50.9**$_{+9.9}$	**73.5**$_{+10.1}$	**51.2**$_{+9.0}$
Llama3-8B	**BBH[27]**	**LogicBench**	**#Avg**
Source_LeetCode	36.6	51.4	38.5
LogicPro$_{\text{w/o Inter-Var}}$	44.0$_{+7.4}$	63.6$_{+12.2}$	45.1$_{+6.6}$
LogicPro$_{\text{w. Inter-Var}}$	**45.0**$_{+8.4}$	**67.9**$_{+16.5}$	**46.2**$_{+7.7}$