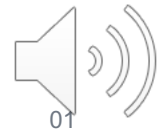
Evolvent AI

The Scaling Laws of Skills in LLM Agent Systems

Qiguang Chen, Qiming Yu, Yuhang Gu, Zhuoye Huang, Hanjing Li, Hongyu Liu, Simin Liu, Jinhao Liu, Dengyun Peng, Jiangyi Wang, Zheng Yan, Fanqing Meng, Ethan Qin, Carl Che, Mengkang Hu

Evolvent AI Team



A library is a search space

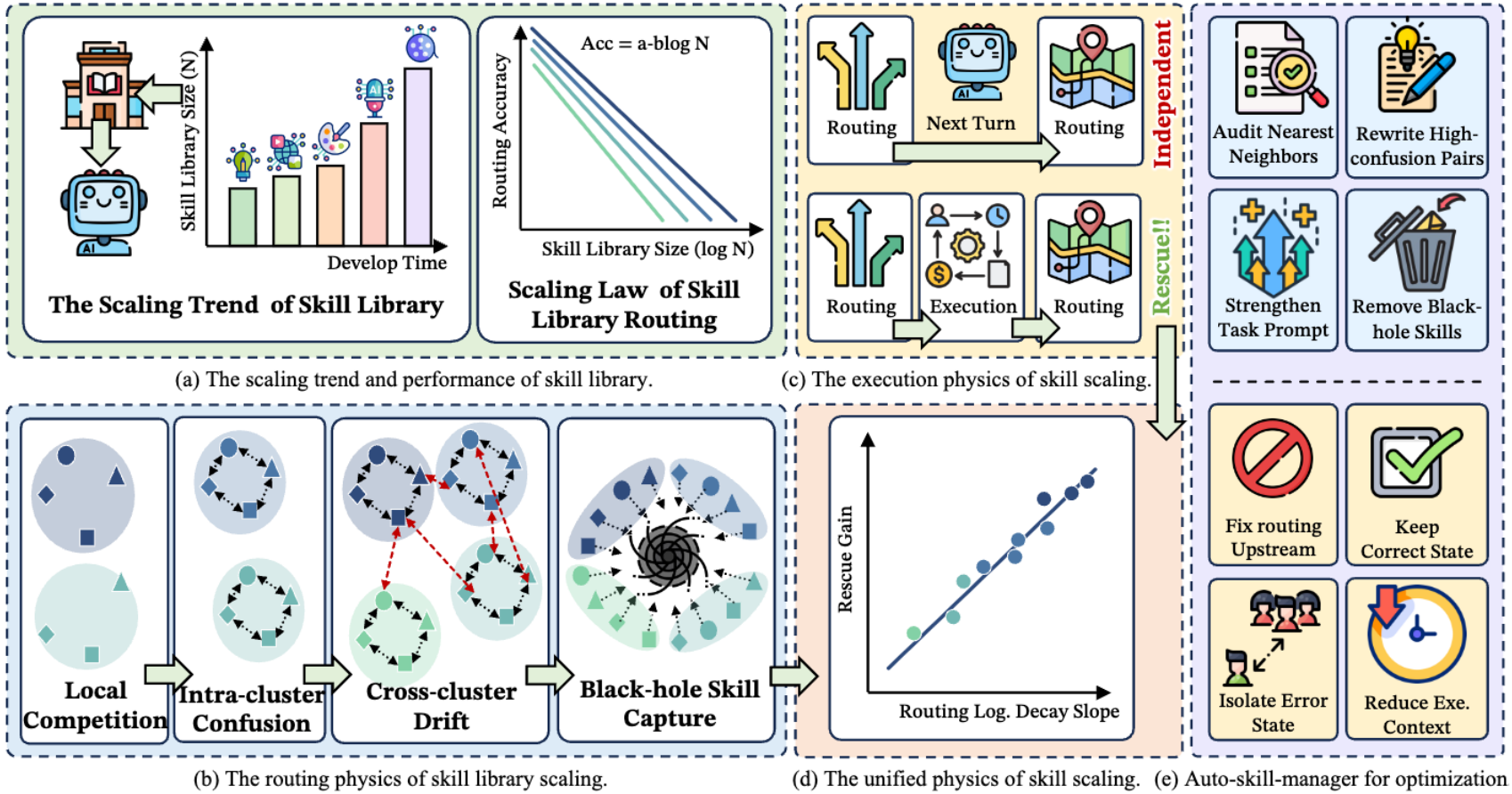
💡 **A larger skill library doesn't necessarily make an agent more capable.**

We found that large skill libraries don't fail at random. Instead, they follow a repeatable, analyzable progression: from local semantic congestion, to confusion within skill families, to cross-cluster drift, and finally to capture by "black-hole skills."



Three questions, one variable

The same library geometry governs routing, execution state, and their coupling



Q1: How fast does the router lose the right skill as the library grows?

Q2: When does one action help or hurt the next decision?

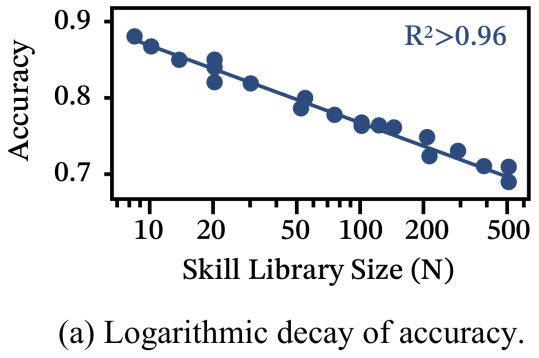
Q3: Can one measurable state variable explain both?

arXiv:2605.16508v1

What exactly is the mechanism of Skill routing?

✅ **As the skill library grows, accuracy declines logarithmically.**

❓ Once multiple skills must be planned in parallel and routed, things change: accuracy drops far below what independent probabilities would predict.



(a) Logarithmic decay of accuracy.

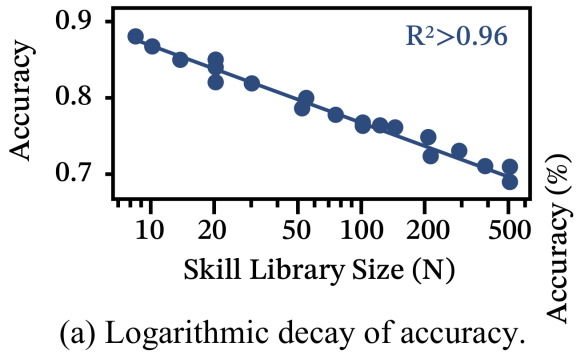
(b) Beyond independent logarithmic decay of 3 step parallel routing accuracy.

What exactly is the mechanism of Skill routing?

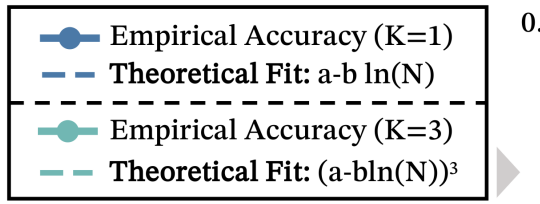
🎯 **Under the condition of parallel planning + routing of multiple skills:**

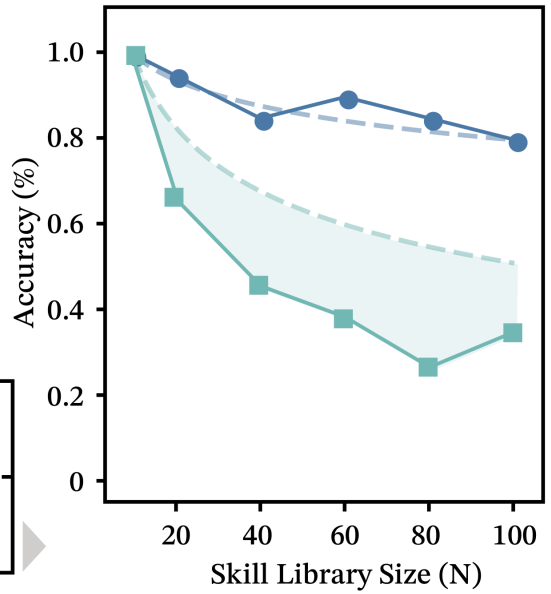
✅ Multi-step tasks further amplify this loss through an exponential factor:

$$Acc(N) = (a - b \ln N)^{\gamma K}$$

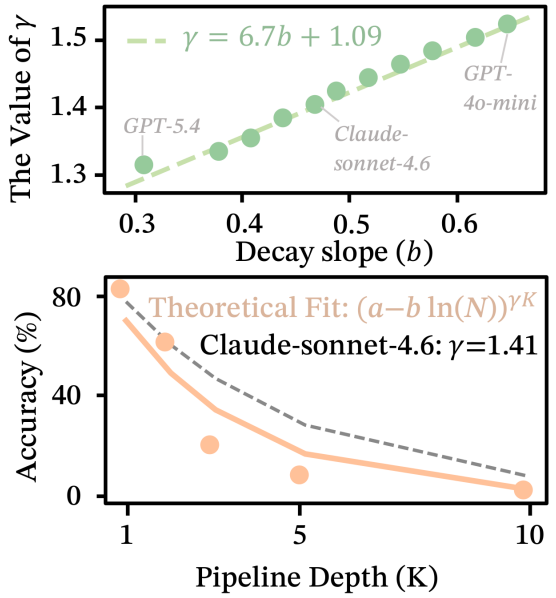


(a) Logarithmic decay of accuracy.

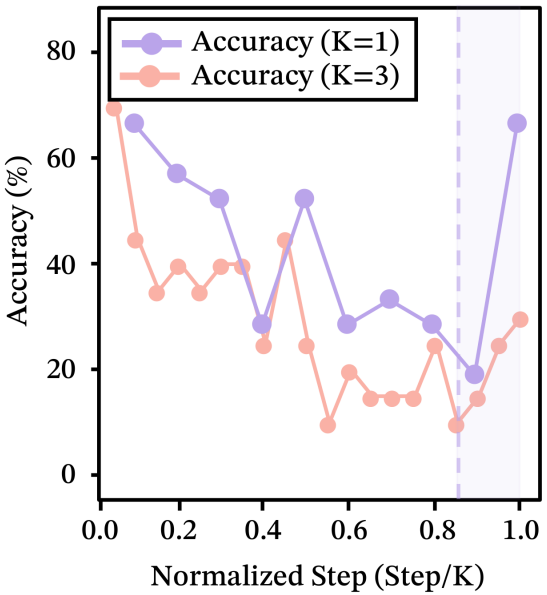




(b) Beyond independent logarithmic decay of 3 step parallel routing accuracy.



(c) Pipeline compounding law of accuracy strongly associated with decay slope (b) .



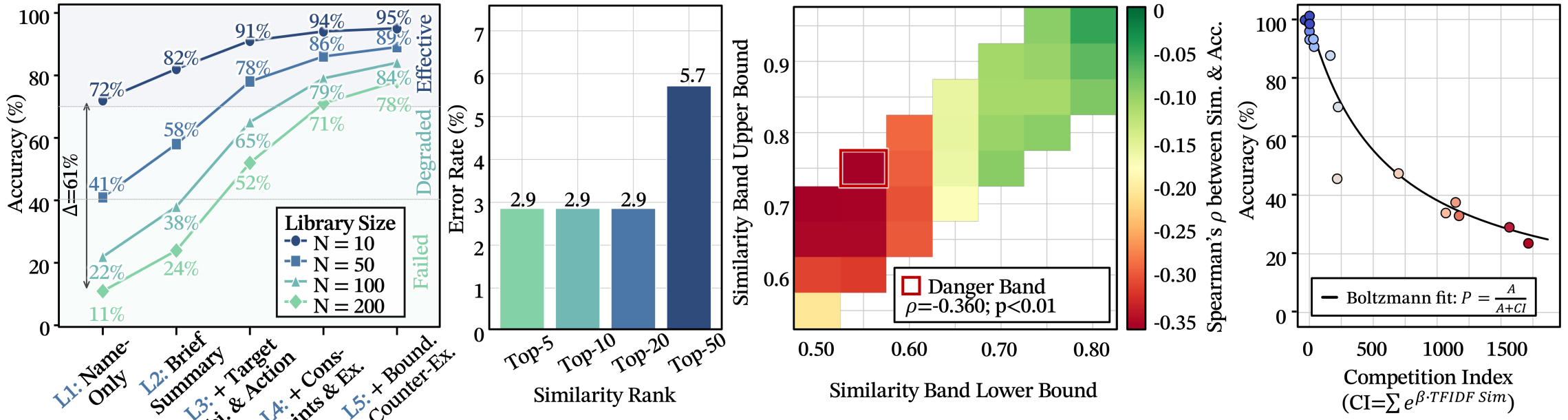
(d) The U-shape rebound of per-step accuracy in pipeline.

What exactly is the mechanism of Skill routing?

✅ **Semantic competition:** There is a dangerous, deceptively plausible skill region that models can easily be misled.

✅ **Pattern of semantic competition:** Within this region, competition level and accuracy follow a Boltzmann function.

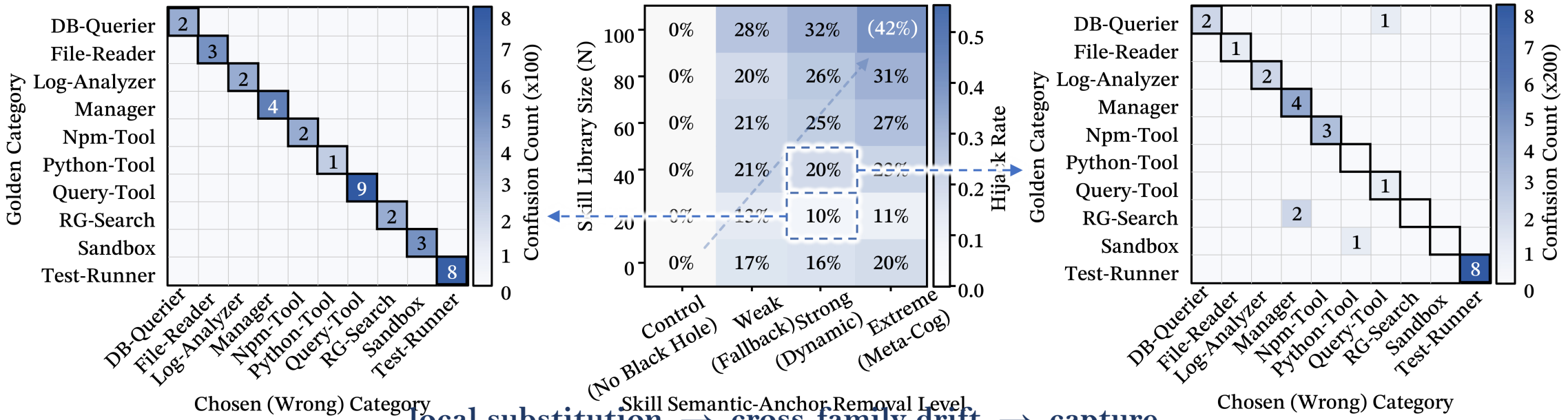
💡 **Essentially, the skill and the request match need to be sufficiently distinguishable.**



What exactly is the mechanism of Skill routing?

✅ We found that errors almost never cross functional clusters.

✅ They tend to occur in the parts that seem plausible but aren't quite right.



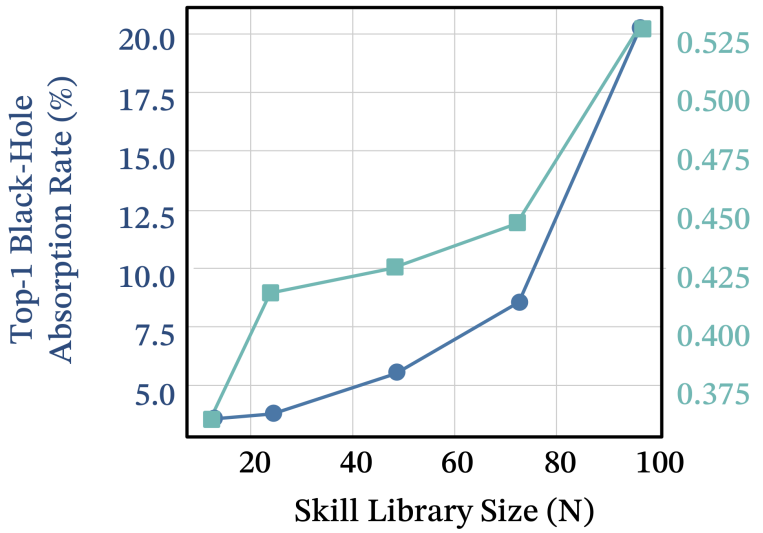
local substitution → cross-family drift → capture

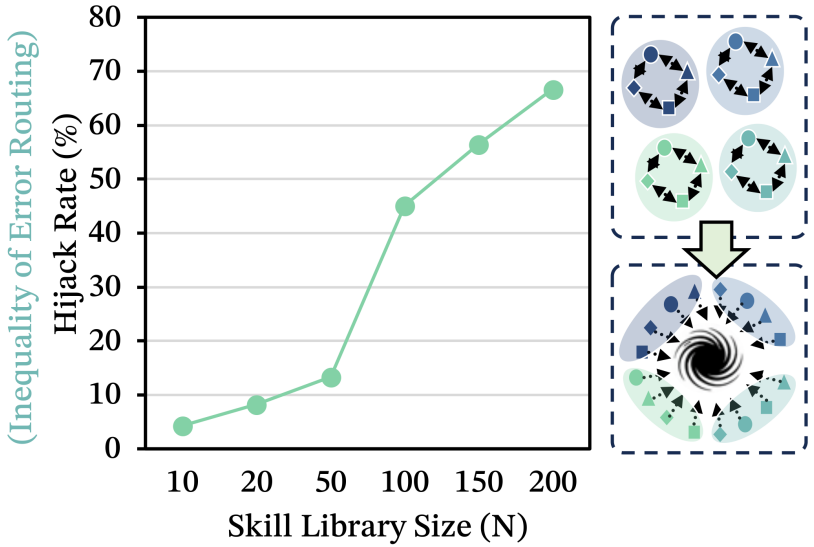
What exactly is the mechanism of Skill routing?

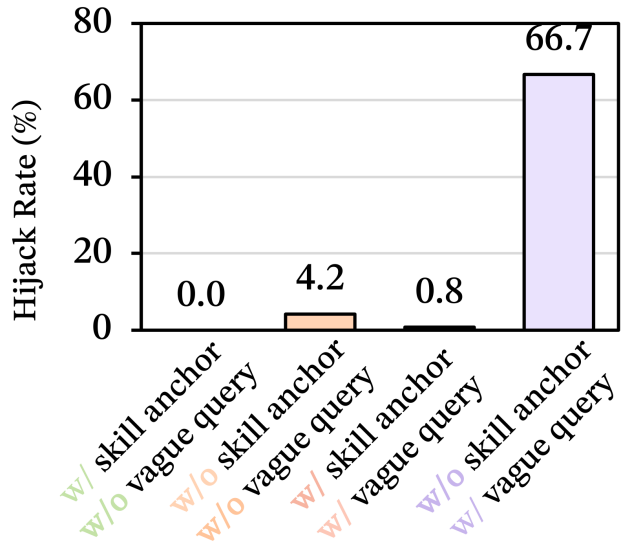
The Black-Hole Skill Effect

Is there a "black-hole skill" that attracts most of the routing traffic?

Yes. As the skill library grows, certain skills attract a disproportionate share of routes, and Gini coefficient rises sharply.





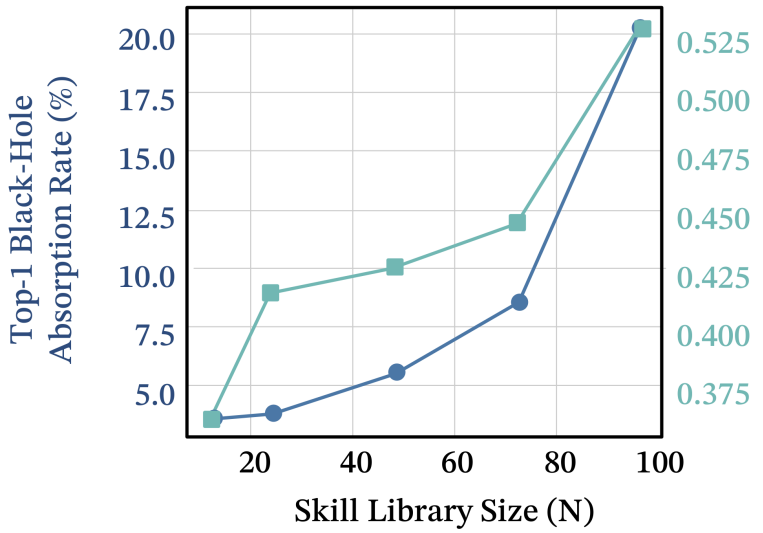


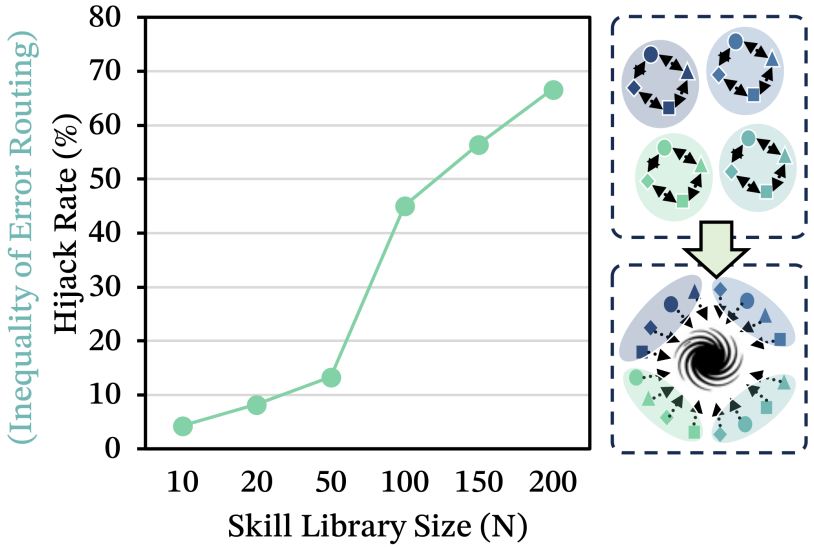
What exactly is the mechanism of Skill routing?

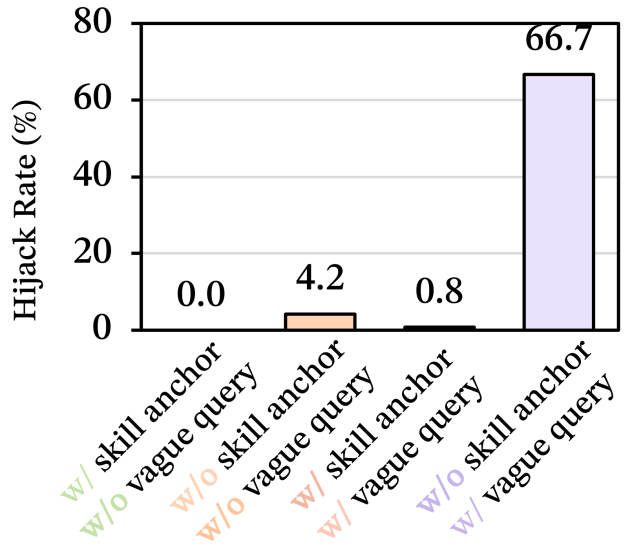
💡 The Black-Hole Skill Effect

❓ When does a “black-hole skill” emerge?

✅ When skill descriptions are vague and queries don’t clearly specify what skills they need. In those conditions, black-hole skills emerge and route requests to the wrong skill in ways users don’t want.



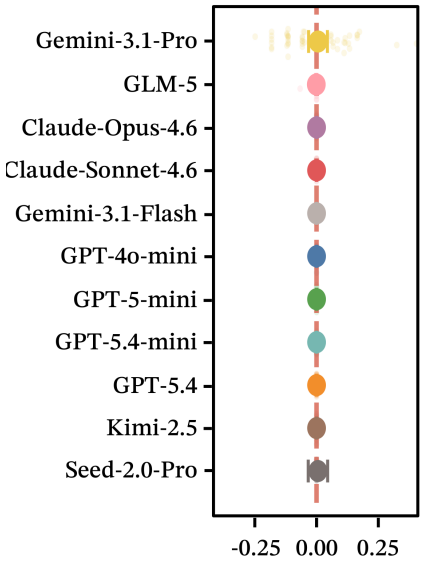




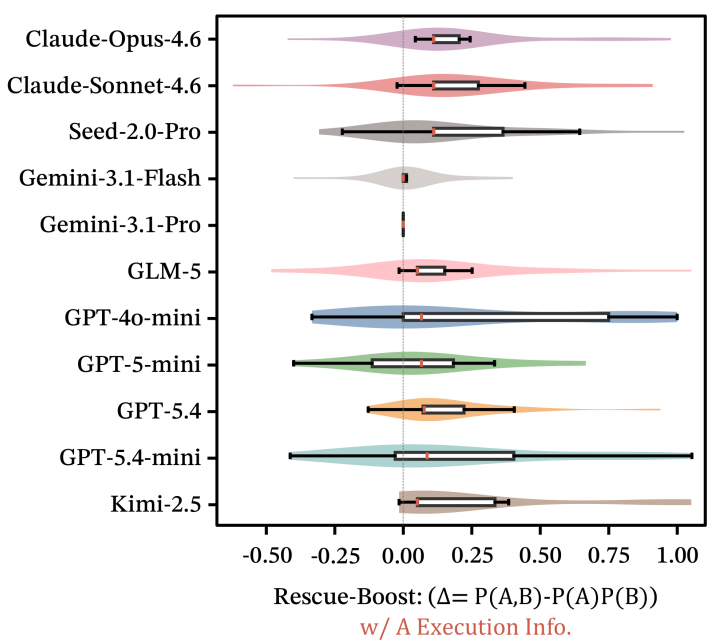
What exactly is the mechanism of Skill executing?

💡 **Sequential Routing Across Multiple Skills**

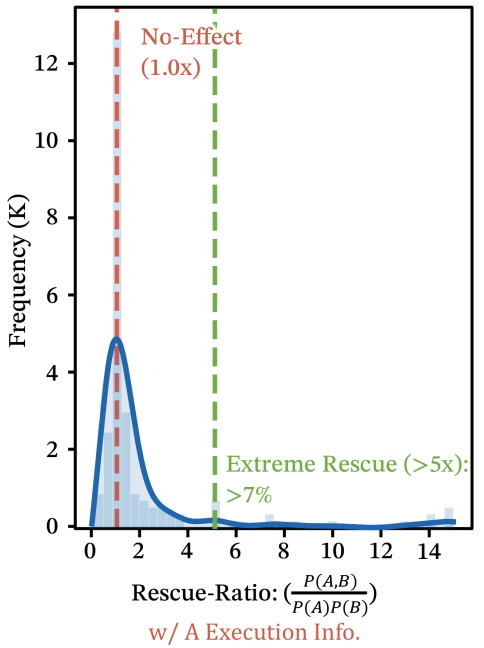
✅ When there's no execution-result information: only a record of which skills were called, the model routes to each skill ***independently***: calling Skill A has no effect on its routing to Skill B.



(a) The pipeline routing independence without execution information.



(b) The pipeline routing with rescue-boost with real execution information.

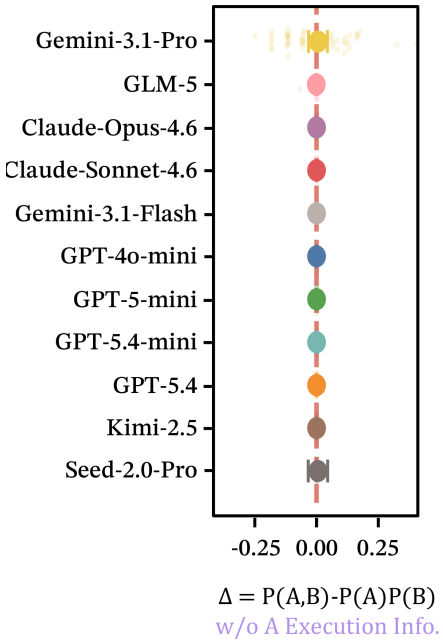


(c) The distribution of rescue-ratio with real execution information.

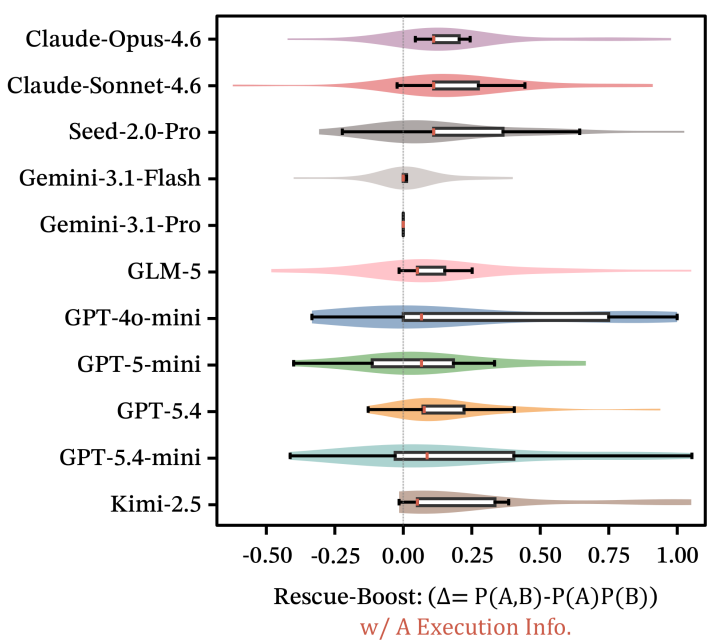
What exactly is the mechanism of Skill executing?

💡 Sequential Routing Across Multiple Skills

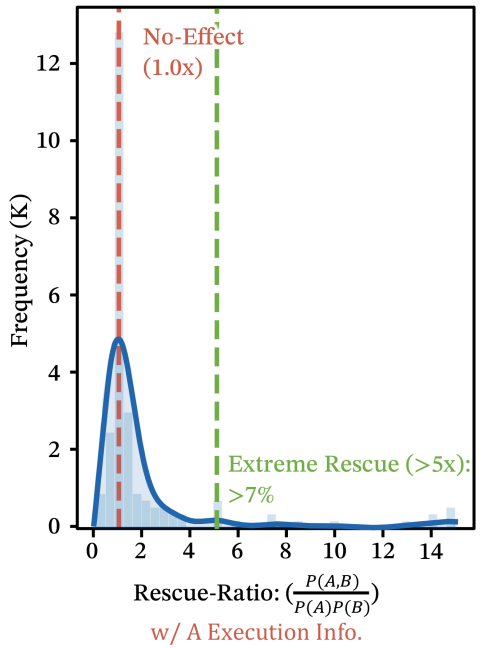
✅ When execution results are available alongside call information, routing shows a ***healing effect***: successful execution can rescue subsequent steps that would otherwise fail.



(a) The pipeline routing independence without execution information.



(b) The pipeline routing with rescue-boost with real execution information.



(c) The distribution of rescue-ratio with real execution information.

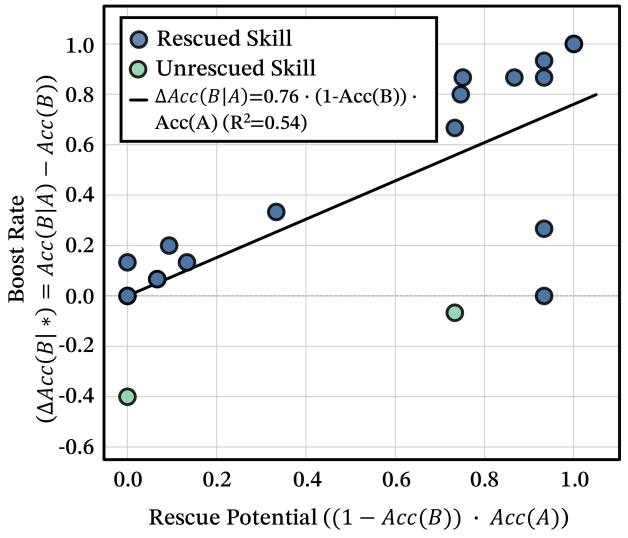
What exactly is the mechanism of Skill executing?

🎯 The Drag Effect of Execution Errors

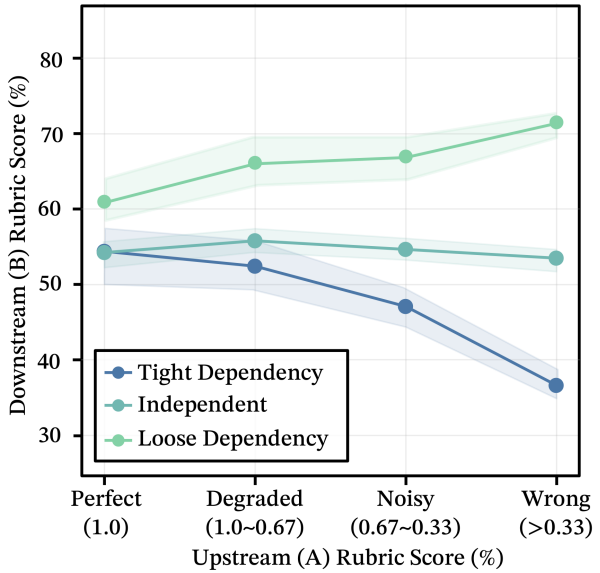
✅ Tightly dependent skills can have a major impact on the overall pipeline.

✅ Independent skills have little impact.

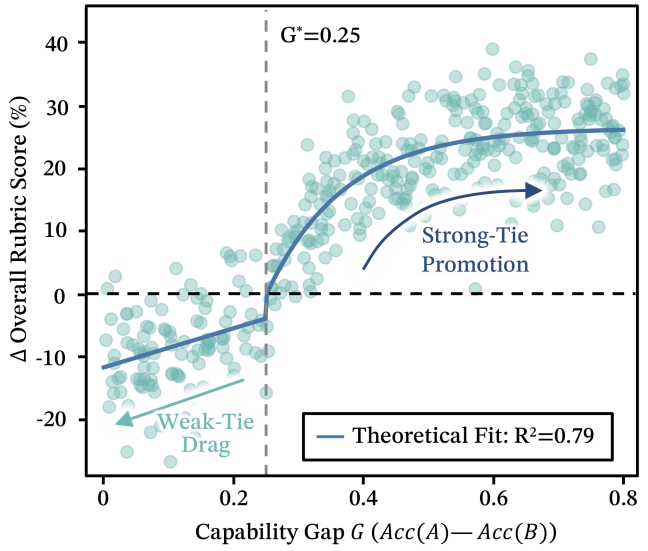
✅ Loosely coupled skills can actually help the overall pipeline recover.



(a) The validation of rescue law: $\Delta Acc(B|A)$=0.76 · (1-Acc(B)) · Acc(A) (R2=0.54)



(b) Rubric scores for two-step task completion under varying degrees of semantic coupling between skills.



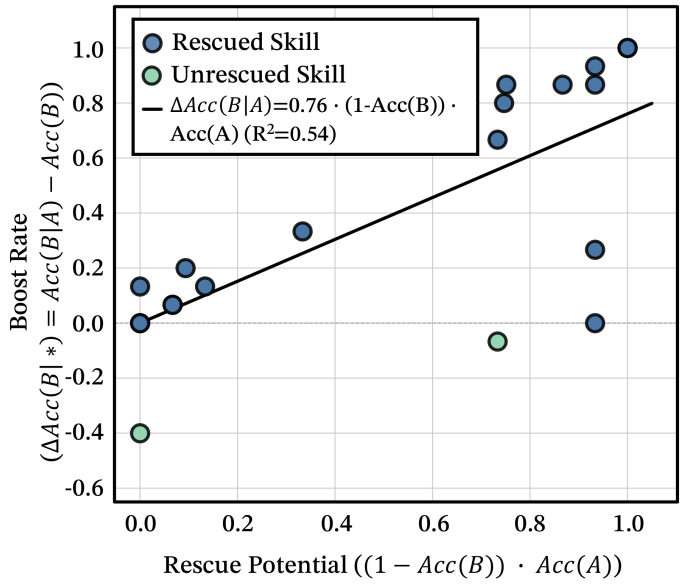
(c) Strong-Tie Promotion and Weak-Tie Drag under Crowding Risk.

What exactly is the mechanism of Skill executing?

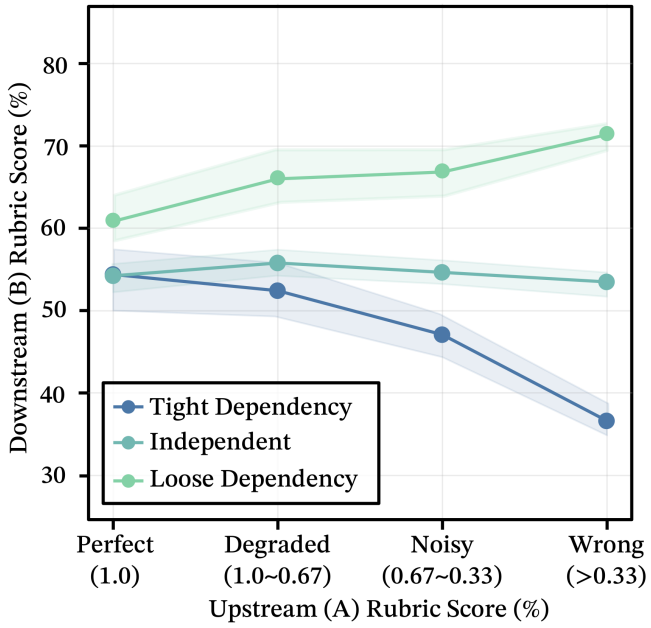
💡 **Strong-to-Weak Lift, Weak-to-Strong Drag**

✅ If the skill for the first task is more capable than skill for the second task, it lifts the Acc. of subsequent tasks ***nonlinearly***.

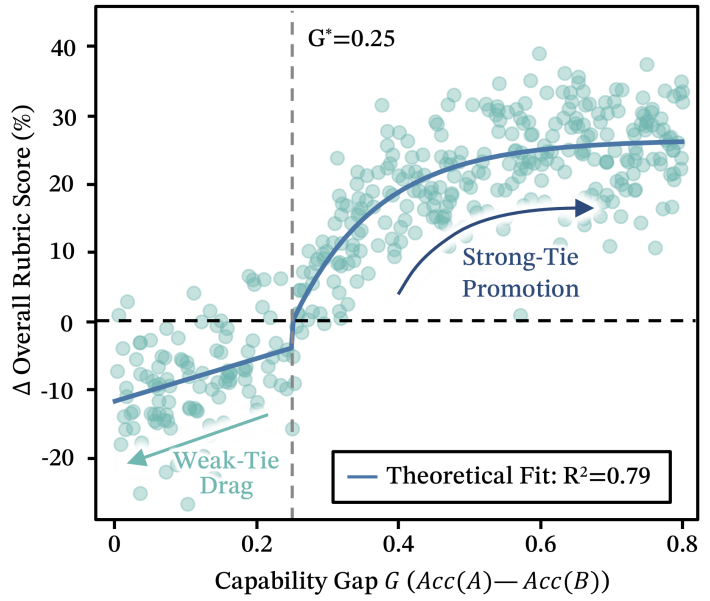
✅ If the first task's skill is less capable than the second task's, it drags down subsequent tasks ***linearly***.



(a) The validation of rescue law: $\Delta Acc(B|A)$=0.76 · (1-Acc(B)) · Acc(A) (R2=0.54)



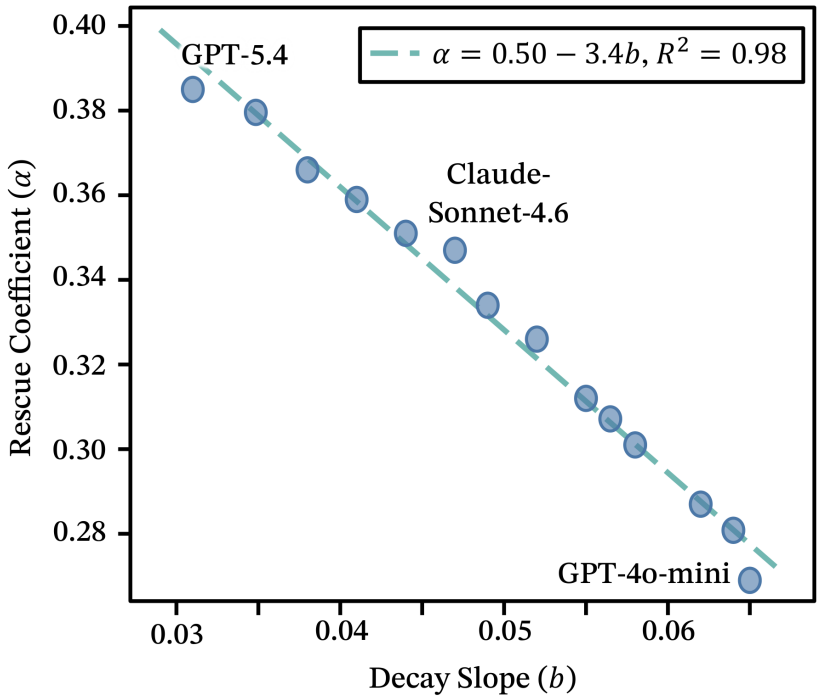
(b) Rubric scores for two-step task completion under varying degrees of semantic coupling between skills.



(c) Strong-Tie Promotion and Weak-Tie Drag under Crowding Risk.

Unified Routing & Executing Law

The routing slope b is the shared state variable



$$Acc_{res}(B|N) = Acc(B|N) + 2\alpha\left(1 - Acc(B|N)\right)(\alpha - b \ln N)$$

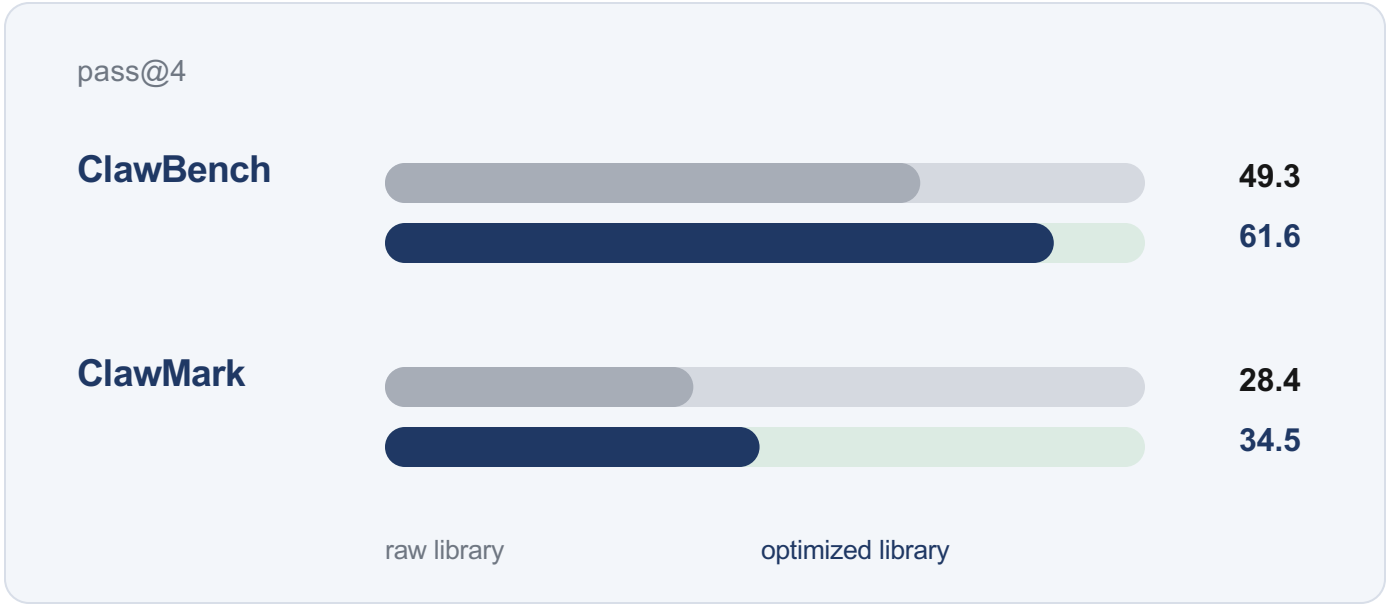
No new parameter is introduced. The same b that costs routing accuracy also determines how much correct state survives for downstream use.

Lowering b is the design objective

$R^2 = 0.98 \cdot \rho = 0.74$ on independent rescue measurements

The gain transfers across agents

The optimized library transfers across agents and domains



30 / 31

domains improve

ClawBench

49.3 → 61.6

ClawMark

28.4 → 34.5

Design the library, not just the model

Large skill libraries should be structured systems, not pools of tools.

PAPER

CODE

BLOG