NANJING UNIVERSITY

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

Neuro-Symbolic Agents: Why Symbols Still Matter in the Foundation-Model Era

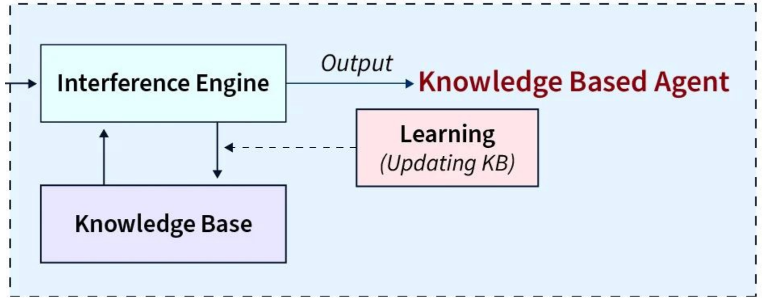
Jie-Jing Shao

Nanjing University, LAMDA Group

Open Tasks, Unresolved Reliability

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

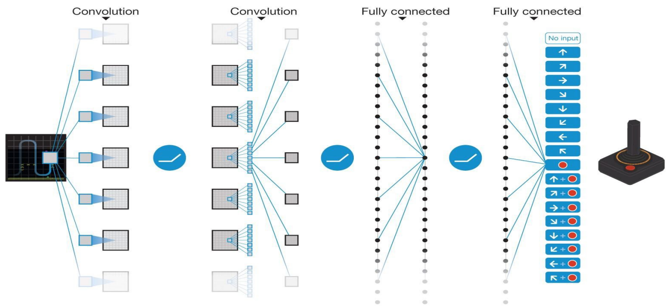
Symbolic Agent



Explicit rules
+ search / planning

Logic Theorist, 1956; STRIPS, 1971

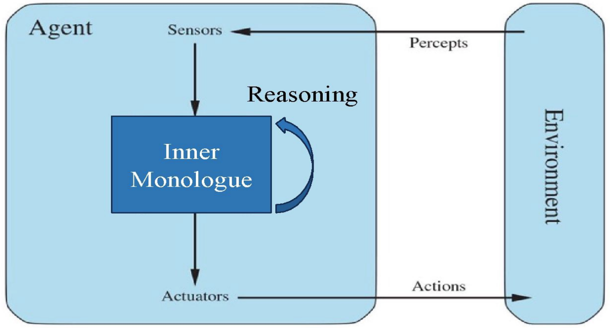
Neural & RL Agent



Neural perception
+ learned policies

DQN, 2015; AlphaGo 2016

LLM Agent



Open-language understanding + reasoning

React, 2023; SWE-Agent 2024

Goal space	Fixed symbols	Task rewards	Open language
Perception	Needs grounding	Learned perception	Multimodal inputs
Transfer	Within symbols	Data-hungry	Model priors + examples

Can LLM agents generalize reliably to new goals and environments?

Challenge 1: Unseen Goal Compositions

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

RL Agent View

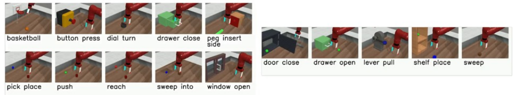
State s — Agent $\pi(a|s)$ — Action a

Reward r — Environment $P(s'|s,a)$

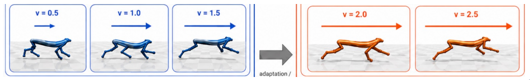
Generalization = unseen states, rewards, or environment dynamics.

Train | Test

Discrete goals



Continuous goals



Language Defines the Goal Space

Language Agent View

Development: seen compositions

Visit museums in Beijing, prefer local cuisine, spend ≤ CNY 2,000, avoid flights, and finish in 3 days.

Development

Deployment

Daily meal budget ≤ CNY 200

Return to Shanghai on Day 2 afternoon

Same concepts, unseen composition.

Composition

Cost | Cuisine | Attraction
Transport | Duration | ...

Infinite goal compositions from finite concepts.

Travel-planning benchmark

ChinaTravel

Real-world planning with compositional constraints:

Cost | Cuisine | Attraction | Transportation | Duration | ...

T2I benchmark

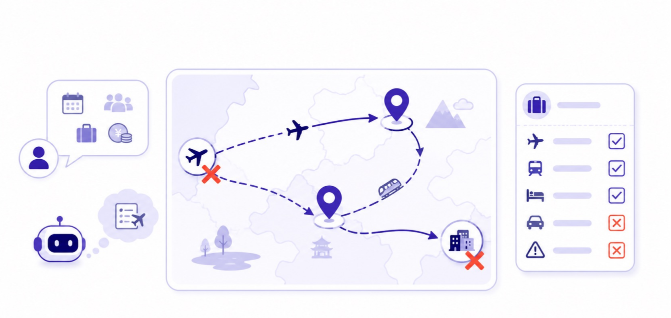
FormalIMG

Formalized image-generation instructions with compositional s… …ures

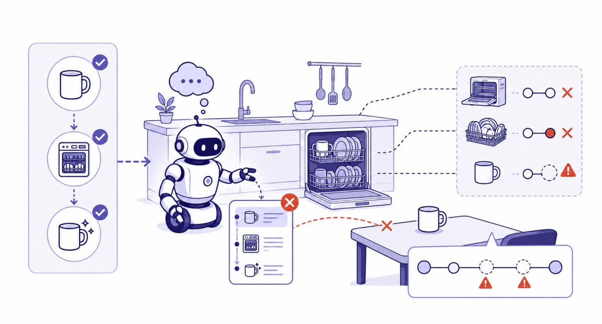
Objects | Relations | Styles | Rules | ...

Challenge 2: Reliable Grounding

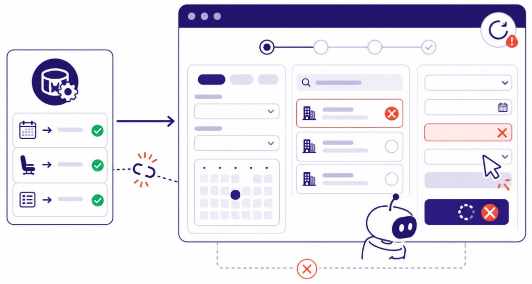
LLMs lack environment-specific feedback for grounding language into action.



Travel: unseen constraints

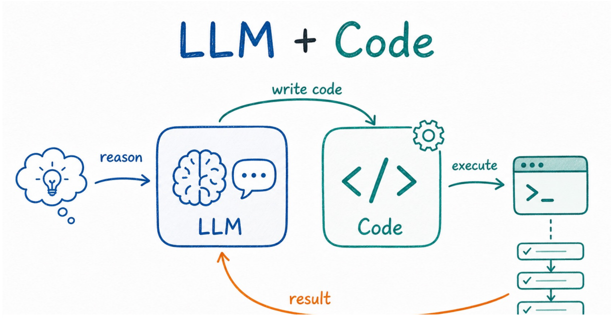


Embodied: state mismatch

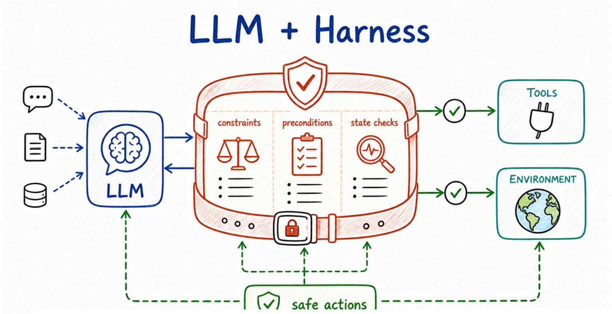


Web/tool: experience reuse

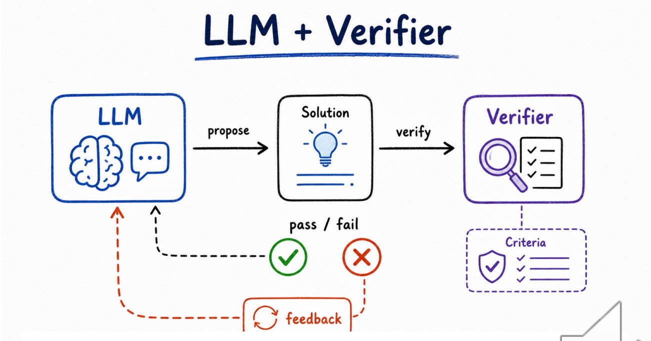
A neuro-symbolic layer makes goals explicit, actions checkable, and experience reusable.



Representation: formalize goals



Execution: induce state-aware skills



Validation: execute checks

Three Studies: Specify, Diagnose, Execute

The Point: Symbols make agent behavior explicit, testable, and reusable.

❖Part 1. Make open-ended goals testable: ChinaTravel + FormalImG

- ChinaTravel: executable constraints for language agents (ICLR 2026)
- FormalImG: structural generalization for T2I models (ICML 2026 Workshop)

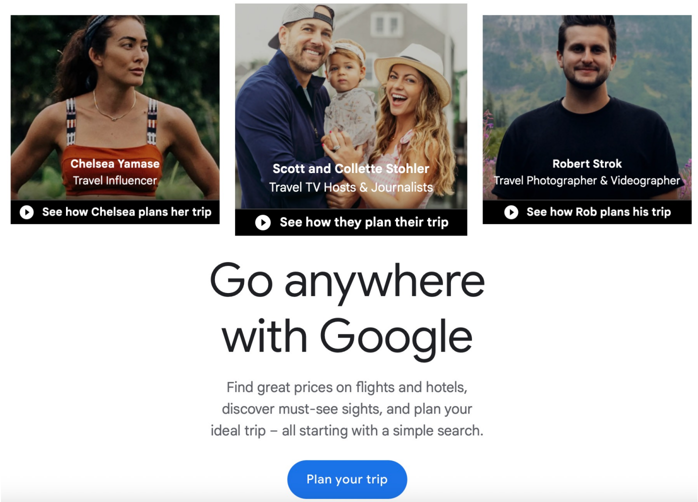
❖Part 2. Turn experience into executable skills: Lifting Traces to Logic

- Programmatic skill induction with neuro-symbolic learning (ICML 2026)
- Evidence across ALFWorld, WebShop, and TextCraft

ChinaTravel

❖Travel planning is a real-world stress test for long-horizon constraint satisfaction.

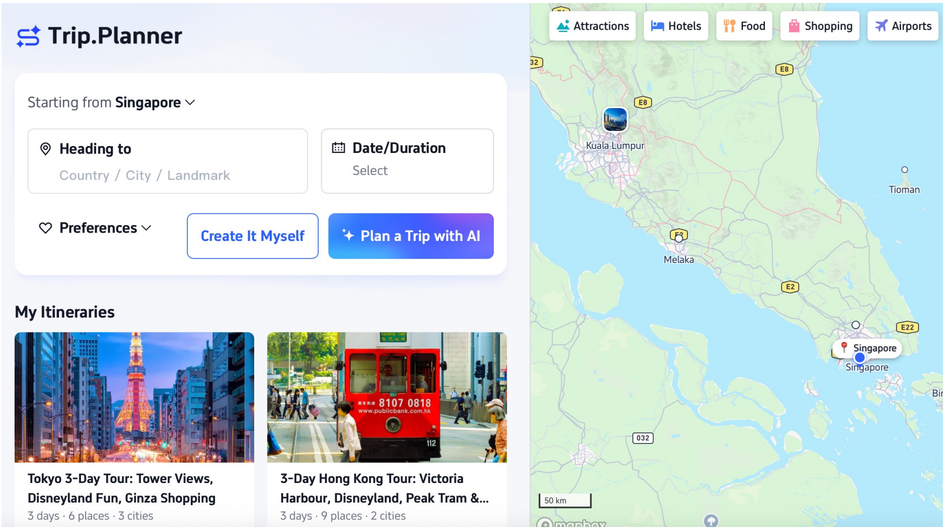
Google



Tongyi DeepResearch

Detailed and realistic travel itineraries

Trip.com

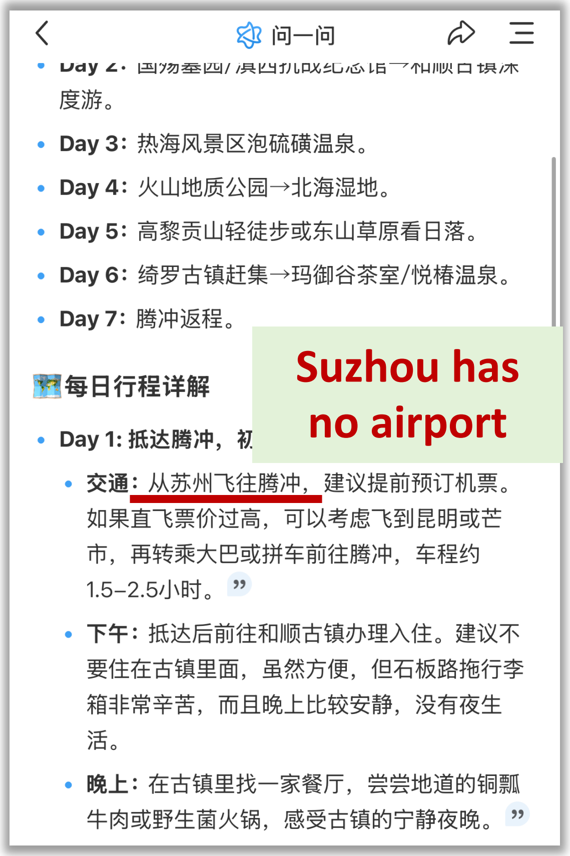


Plausible Plans Can Still Be Infeasible

LAMDA Learning And Mining from DatA
https://www.lamda.nju.edu.cn

User: Plan a five-day trip from Suzhou to Tengchong and Mangshi.

Incorrect facts and underspecified details



Suzhou has no airport

No concrete hotel recommendation

Infeasible or internally inconsistent plan

Return route incorrectly ends in Nanjing

The flight arrives in Mangshi, but the itinerary visits Tengchong first.

Deployment Creates Unseen Goals

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

Known constraints recombine beyond the finite training set.

Training: finite combinations

Goal mix	T	$	H	F	M
1	✓	✓	✓	—	—
2	—	✓	✓	✓	—
3	✓	—	✓	—	✓
4	—	✓	—	✓	✓

Travel for 5 days; spend ≤ CNY 3,000; book a queen room

Spend ≤ CNY 5,000; book a queen room; try West Lake vinegar fish

- Only a finite subset is observed

Deployment: unseen combinations

User request	Constraint space	
Non-flight spend ≤ CNY 3,000	M x $ →	Cost of non-flight
High-speed rail; leave Friday after work	T x M →	Depart time
West Lake before Lingyin Temple	T x A →	Visit route

- Known constraints
- Free recombination
- Mostly unseen goals

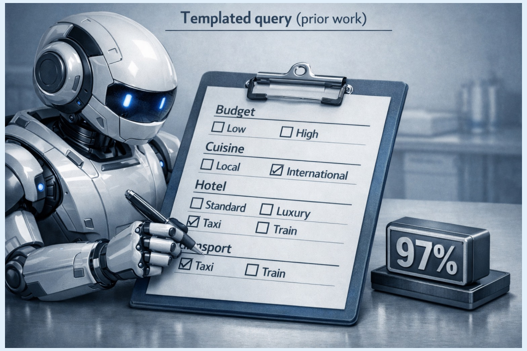
Fixed Schemas Hide Generalization Gaps

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

Predefined slots constrain the goal space by design.

Existing benchmarks

TravelPlanner, DeepPlanning, etc.



- ☐ Fixed fields
- ☐ Predefined templates
- ☐ In-schema evaluation

What They Miss

- ☐ Implicit constraints
- ☐ Cross-domain compositions
- ☐ Held-out instructions

ChinaTravel

Constraints

Cost | Cuisine | Attraction | Transportation | Duration

Open Instructions

The budget of dining is ¥1000.

Visit the Great Wall at the second day

The budget excluding flights is ¥3000.

- ☐ Open-language requests
- ☐ Compositional splits
- ☐ Executable validators

Measures generalization beyond predefined templates

Executable Validation by Composition

➢ Reusable DSL programs replace one validator per template.

Prior benchmarks: fixed constraint templates

Cost	Time	M	Pref.
cost()	time()	m()	p()

- A new template for every composition is expensive and does not scale.

Core requirements

1. **Compose** primitives
2. **Run** validator

Verifiable · Scalable

ChinaTravel: DSL validator

User goal — DSL — Validator

Primitives + syntax

T | $ | M | F

Example: dining cost < CNY 1,000

```
# Dining expenses <= 1000 CNY.
dining_cost = 0
for act_i in allactivities(plan):
  typ = activity_type(act_i)
  if typ=="breakfast" or typ=="lunch"
or typ=="dinner": dining_cost =
dining_cost + activity_cost(act_i)
return dining_cost <= 1000
```

✓valid
× invalid

✓ **Executable and compositional: check feasibility; reuse primitives.**

Sandbox: Retrieve, Plan, Validate

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

Agents retrieve, plan, and receive executable feedback.

Multi-source travel data

- **10** cities
- **720** air
- **3,413** POIs
- **4,655** food
- **5,770** rail
- **4,124** hotels

Transport / POIs / services

Interactive travel environment

- **Goal** time /budget
- **Agent** act
- **Tool APIs**: Rail, POI, Food, Rule
- **Check** revise
- **Plans** candidates
- **Database** records

Environment coverage

- **Records** Travel data
- **Unseen goals** Human query
- **Validator** Program

Closed loop: tool retrieval → itinerary generation → constraint validation.

Two Data Regimes for Generalization

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

Synthetic coverage + human requests

1. Data construction

Templates + LLM

Easy 1 | Medium 2–5

4 domains

Human requests

Human-Val 154 | Human-Test 1000

Open-ended requests

Align

2. Semantic annotation

DSL

1. **Draft** — GPT-4o→DSL
2. **Revise** — 5 annotators
3. **Verify** — 3 reviewers

3. Executable evaluation

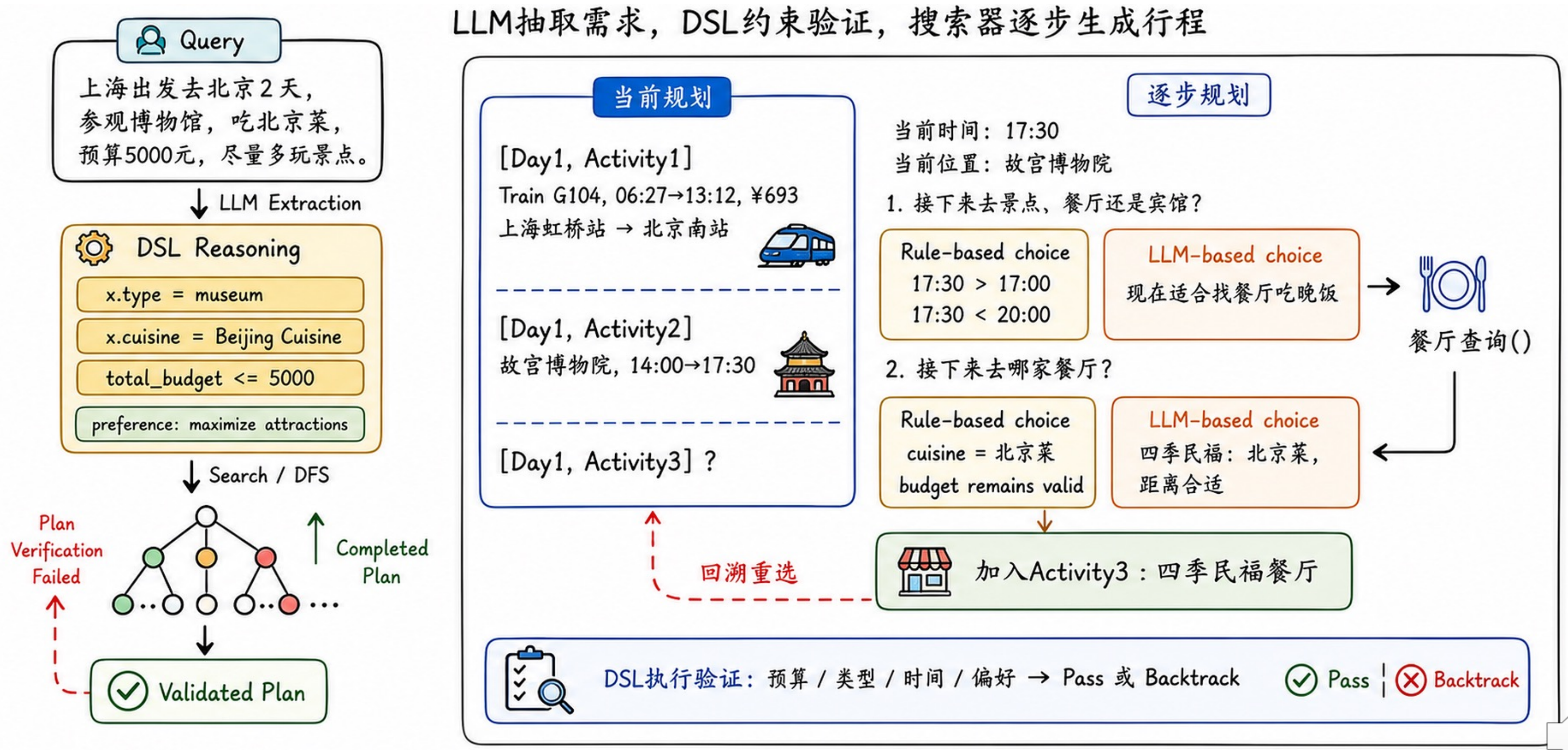
Executable evaluation data

- **15 → 100 types** — Program composition
- **85 new types** — Beyond seeds
- **Executable** — Per instance

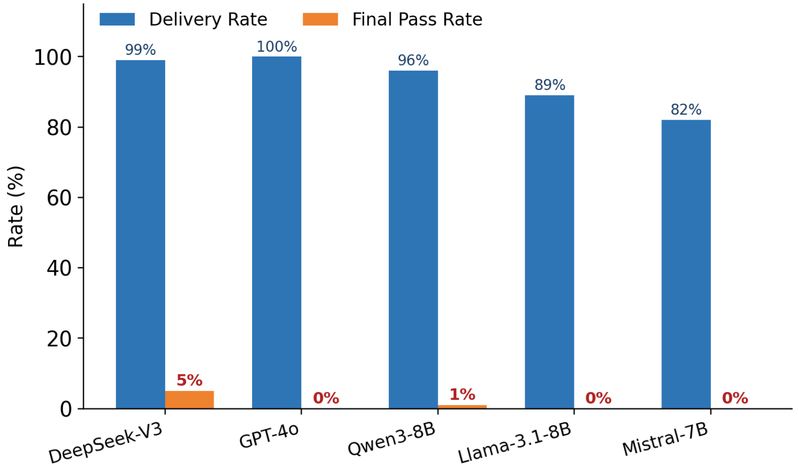
Real requests + extensibility + verifiability

Neuro-Symbolic Planning

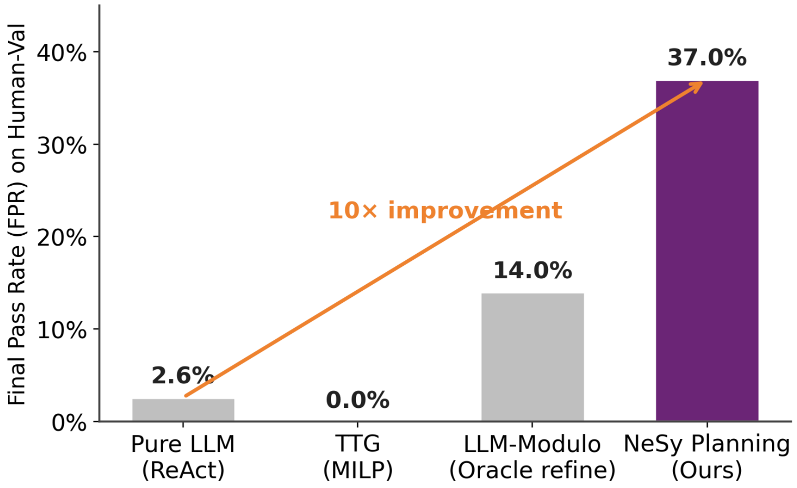
LAMDA
Learning And Mining from DatA
https://www.lamda.nju.edu.cn



Neuro-Symbolic Planning: Results



- Structured output ≠ a valid plan: high delivery rate can hide low final-pass rate.
- One-shot ReAct on Human-Val: 2.59% FPR



2.59% → 37.0%

One-shot ReAct → Neuro-symbolic planning

14.3× higher final-pass rate

Observation

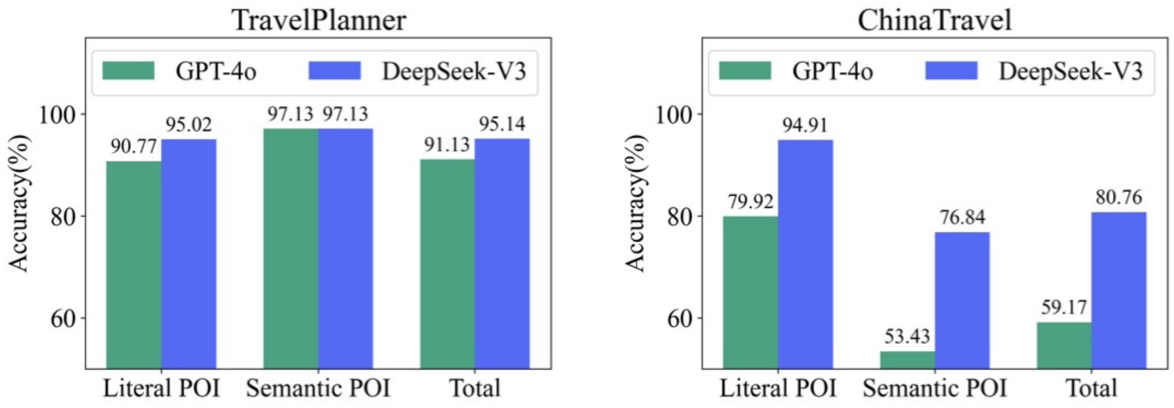
Fluent itineraries still violate cross-day, budget, route, and timing constraints.

Result

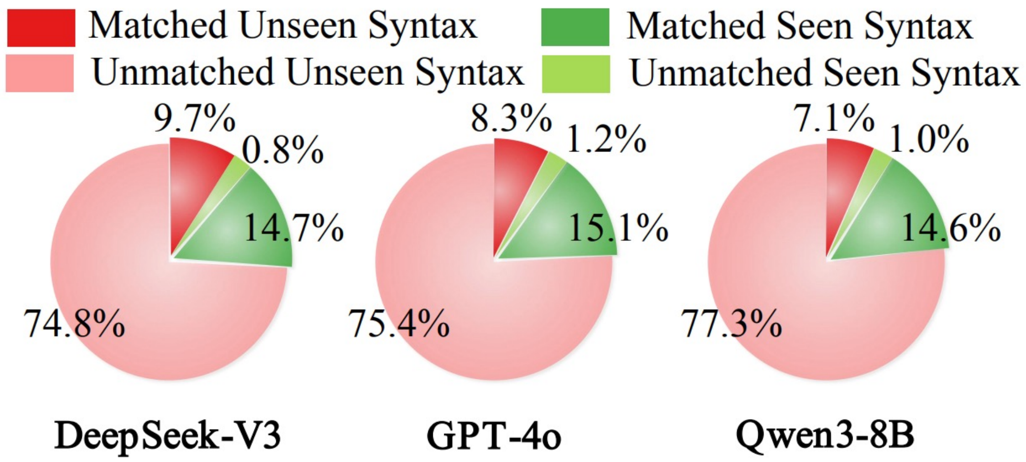
Use the LLM for intent and proposals; use search and validators for long-horizon constraint satisfaction.

Open Problem: Semantic Grounding

Many intents are implicit in local context or commonsense; mapping them to the correct POIs and predicates remains brittle.



Users recombine time, budget, type, and preference concepts into constraints never seen during development.



ChinaTravel: Evidence and Remaining Gap

Learning And Mining from DatA
https://www.lamda.nju.edu.cn

- Open-language agents face a combinatorial goal space, not a fixed menu of task slots.
- ChinaTravel represents requirements as composable, executable constraints.
- Validators support evaluation and search without using model self-judgment as ground truth.

- Application: Huawei's Business travel scheduling.
- Impact: Over 20K downloads on Hugging Face; adopted as the benchmark for the IJCAI 2025 and 2026 Travel Planning Competitions; used by Tencent's Hunyuan team to evaluate LLM planning capabilities.

Selected references:

1. **Jie-Jing Shao**, Bo-Wen Zhang, Xiao-Wen Yang, Bai-Zhi Chen, Si-Yu Han, Jing-Hao Pang, Wen-Da Wei, Guohao Cai, Zhenhua Dong, Lan-Zhe Guo, Yu-Feng Li. ChinaTravel: An Open-Ended Travel Planning Benchmark with Compositional Constraint Validation for Language Agents. **ICLR 2026**.
2. **Jie-Jing Shao**, Hong-Jie You, Guohao Cai, Quanyu Dai, Zhenhua Dong, Lan-Zhe Guo. Breaking the Self-Evaluation Barrier: Reinforced Neuro-Symbolic Planning with Large Language Models. **IJCAI 2025**.
3. Bo-Wen Zhang, Jin Ye, **Jie-Jing Shao**, Yu-Feng Li, Lan-Zhe Guo. **Mind the Gap to Trustworthy LLM Agents: A Systematic Evaluation on Constraint Satisfaction for Real-World Travel Planning**. **AAAI 2026 Trust Agent Workshop Best Student Paper Award**.
4. Bo-Wen Zhang, Jin Ye, **Jie-Jing Shao**, Xiao-Wen Yang, Kun-Yang Yu, Zhi Zhou, Yu-Feng Li, Lan-Zhe Guo. **A Review of the IJCAI 2025 Travel Planning Challenge: Benchmarking LLM Agents with Compositional Constraints**. **Frontiers of Computer Science**.

FormalImG Tests Structural Composition

Concepts

Object Texture Spatial

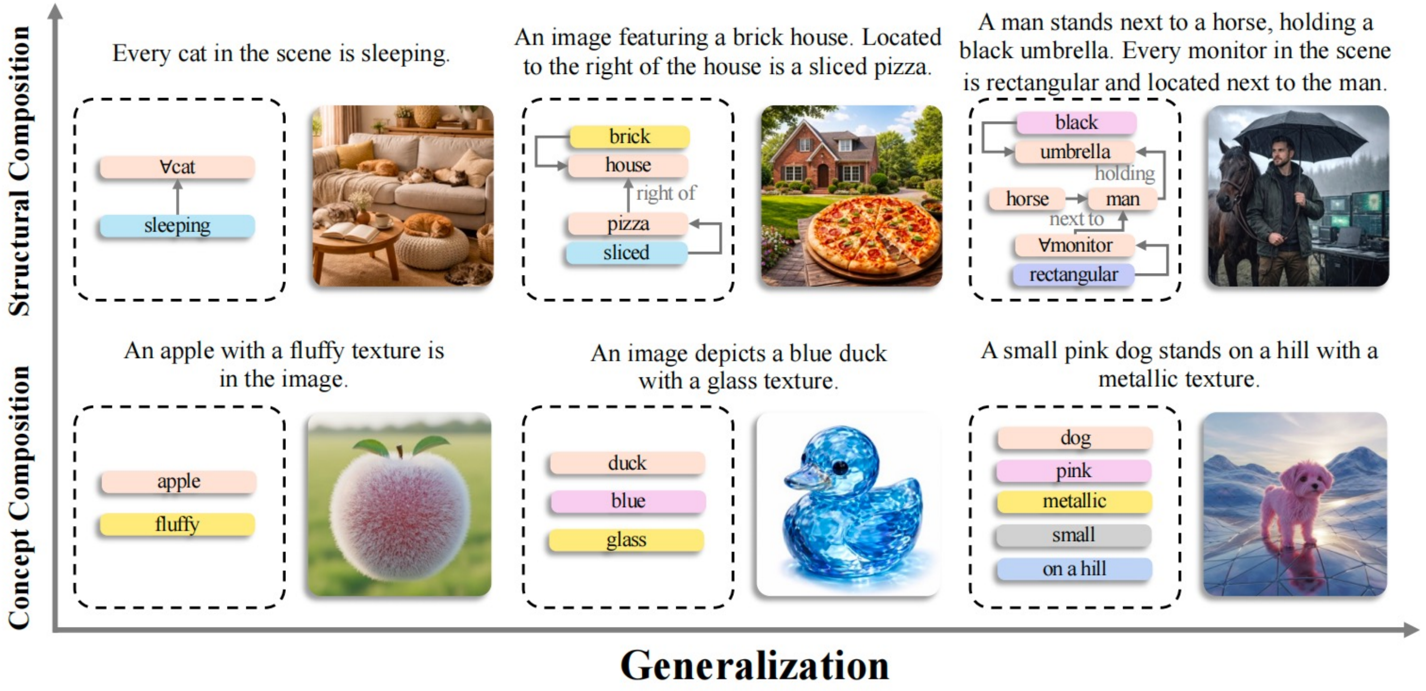
Shape Style

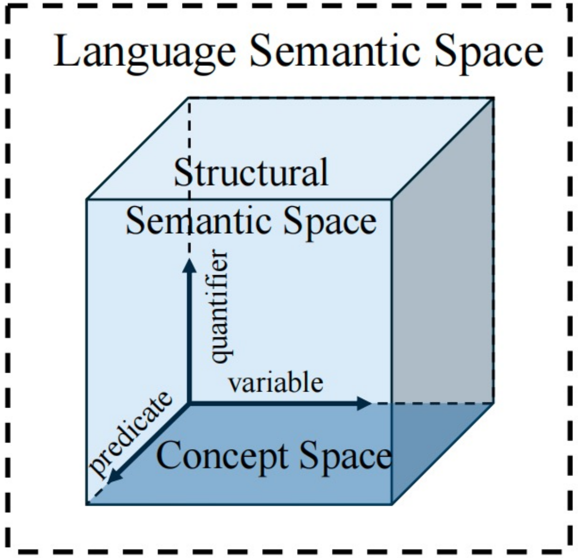
Classic text2img prompt

Pixel style, cow, fluffy texture

Language instruction

A pixel art image depicts a cow with a fluffy texture, shaped like a heart.



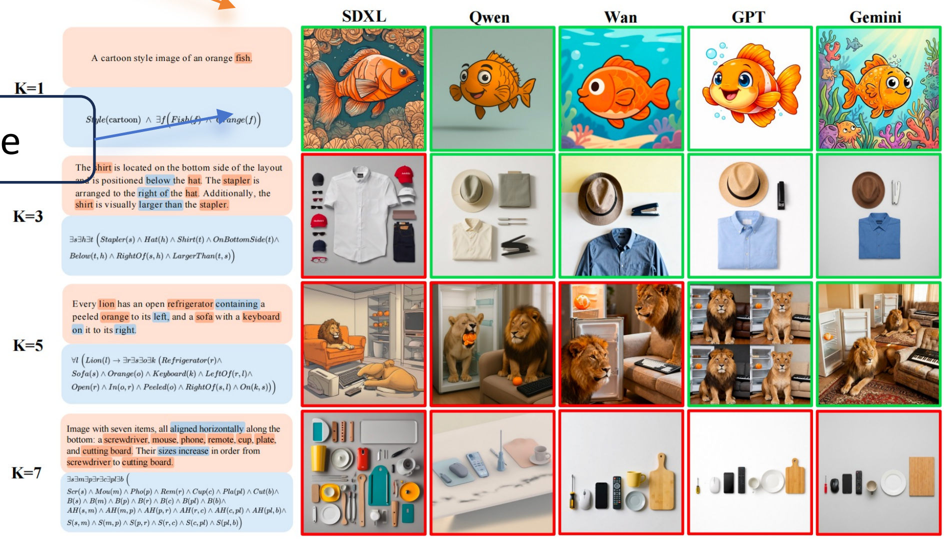


Hong-Jie You, **Jie-Jing Shao**, Xiao-Wen Yang, Zhi-Fan Wu, Lin-Han Jia, Lan-Zhe Guo, and Yu-Feng Li.
FormalImG: Evaluating Structural Compositional Generalization for T2I Models.
In *ICML 2026 Workshop on Combining Theory and Benchmarks*.

FormalImG: Structural Scaling Gap

Instruction

DSL / judge



Model	K									
	1	2	3	4	5	6	7	8	9	10
Natural										
Gemini-3-Pro-Image (Google, 2025b)	92.5	78.5	**76.0**	**79.5**	**78.5**	**74.0**	**68.0**	**76.0**	**71.5**	**65.5**
Gemini-2.5-Flash-Image (Google, 2025a)	90.0	63.5	54.0	54.5	61.5	49.0	38.0	39.5	42.0	28.5
GPT-Image-1 (OpenAI, 2025a)	95.0	81.0	71.0	76.5	57.0	59.0	61.5	54.5	51.0	47.5
GPT-Image-1.5 (OpenAI, 2025b)	**96.0**	**84.0**	73.5	73.5	63.5	63.5	62.0	59.5	52.5	53.0
Seedream-4.5 (Chen et al., 2025)	95.5	83.5	55.0	67.0	58.0	48.5	41.5	44.5	35.5	30.0
Wan-2.6-T2I (Wang et al., 2025)	92.5	72.0	58.5	54.0	45.0	45.0	36.5	41.0	30.0	34.0
Knolling										
Gemini-3-Pro-Image (Google, 2025b)	93.0	**90.0**	**80.0**	**70.0**	**66.5**	**63.5**	**58.5**	**49.0**	**53.0**	**47.0**
Gemini-2.5-Flash-Image (Google, 2025a)	**94.5**	73.0	57.5	53.5	36.5	35.0	23.5	18.5	18.5	12.5
GPT-Image-1 (OpenAI, 2025a)	94.0	71.0	59.5	49.5	45.5	37.5	36.5	24.0	20.5	16.0
GPT-Image-1.5 (OpenAI, 2025b)	92.0	75.5	62.5	52.0	51.5	48.0	42.0	30.0	28.0	20.0
Seedream-4.5 (Chen et al., 2025)	85.0	65.0	48.5	34.0	33.5	26.5	21.0	17.0	12.0	12.5
Wan-2.6-T2I (Wang et al., 2025)	63.0	53.0	39.5	30.0	30.0	28.0	24.0	14.5	18.0	12.0

Obs. Performance drops rapidly as structural complexity K grows.

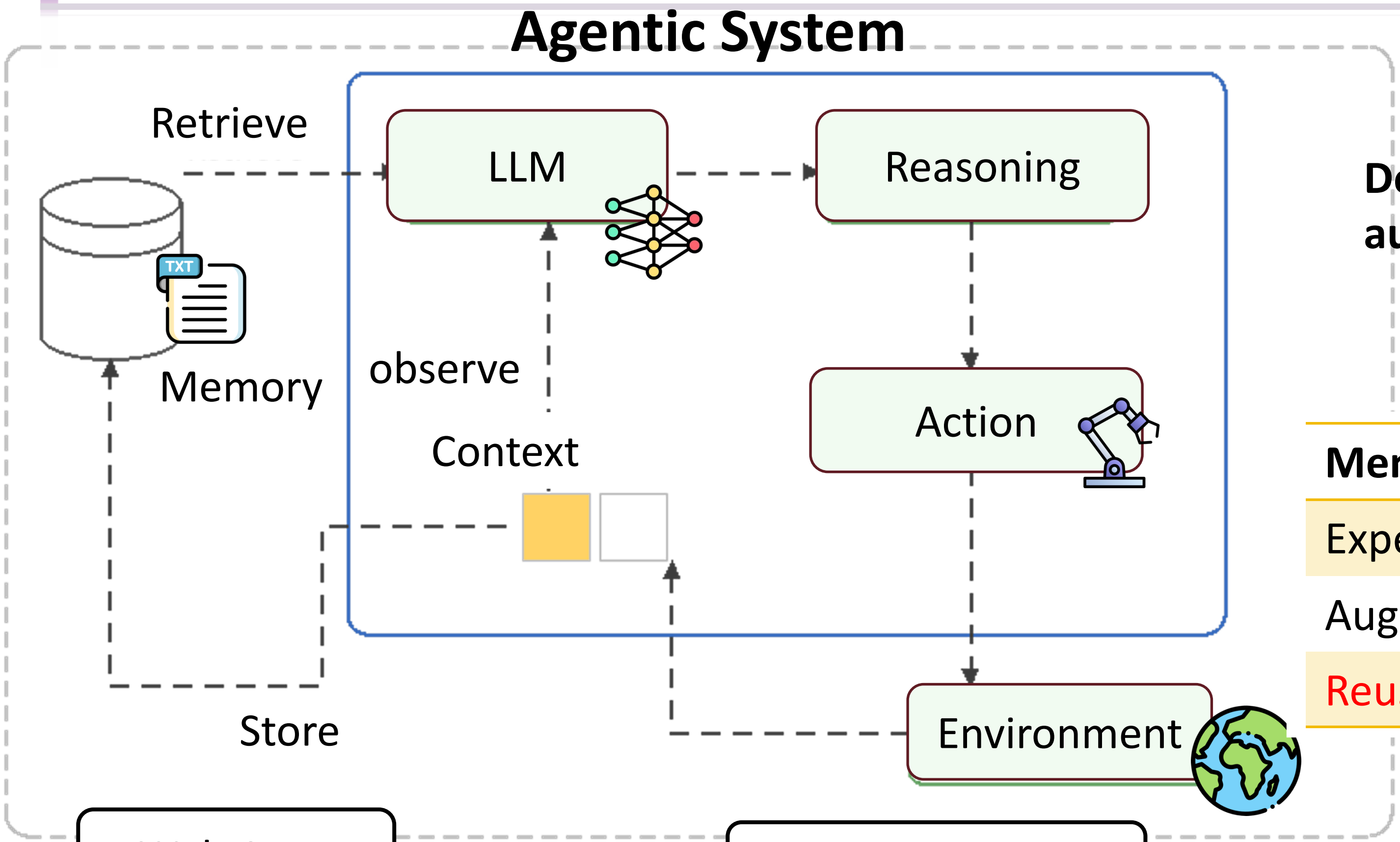
Imp.

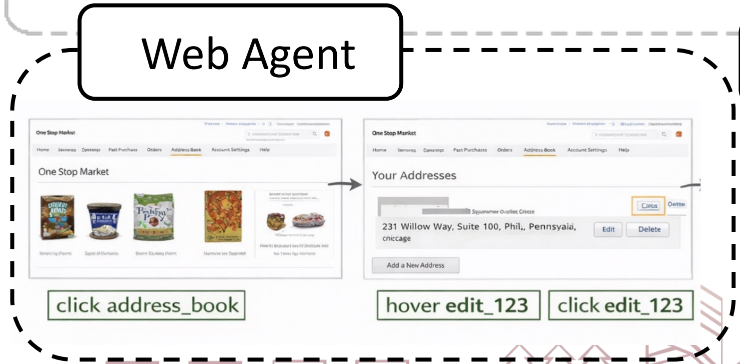
- Models can generate individual concepts; preserving the overall structure remains difficult.
- Structural complexity offers a new scaling axis for evaluating generalization.

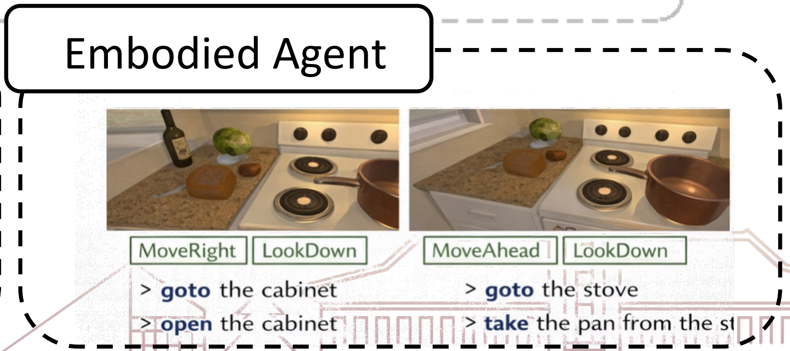
From Structure Tests to Executable Skills

- **Benchmarks expose unseen combinations and dependency failures**
  - ChinaTravel: constraint compositions
  - FormalImG: variable-connected structure
- **The next problem: preserve state, branches, and preconditions in reusable skills**
  - Lifting Traces to Logic (ICML 2026)
  - Goal: reusable execution, not memorized action sequences

Experience Needs Executable Logic







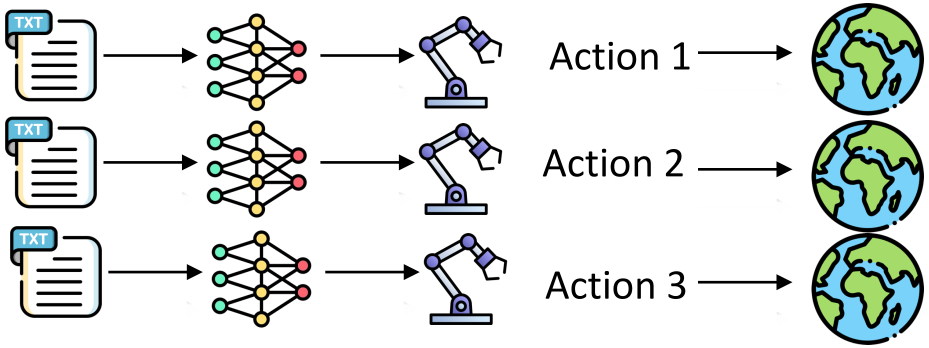
Deployment creates traces, but raw traces do not transfer automatically.

Memory	Skill
Experience → memory	Experience → executable workflow
Augments memory	Augments action space
Reuse without replaying every step	

Reusable Skills Need State Conditions

Memory-Based Adaptation

Text retrieval + reasoning each time

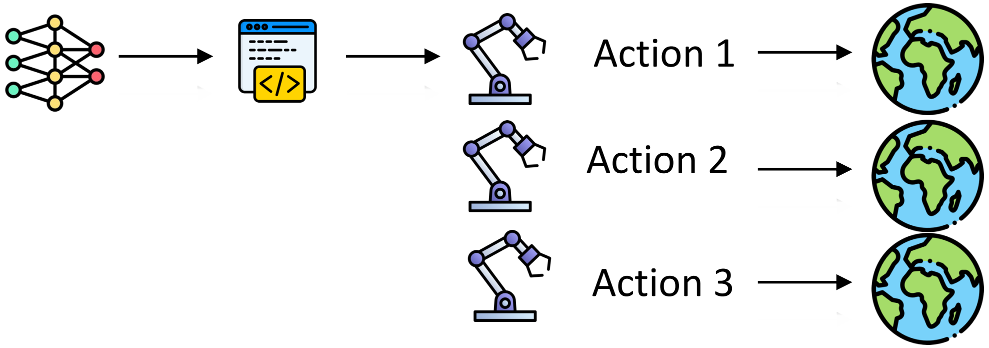


Reflexion (NeurIPS‘23), AWM (ICML’25)

RAG-like reuse: retrieve text, then reconstruct the logic on every invocation.

Skill-Based Adaptation (Programmatic Skill)

Distill experience into executable programs



ASI (COLM‘25), SkillWeaver (2025)

Programmatic reuse is efficient and executable, but only if the program checks the current state.

(System-2 Thinking) → (System-2 → System-1 Transition)

Agentic Skill Induction （COLM'25）

Key Idea: Turn successful experience into reusable programs

Example: "Find # of satisfied reviews for Product 37"

Without skills — 5 primitive actions:

click("37") → click("All Reviews")→ fill(box, "satisfied")
→ click("Search")→ count

<u>Each action needs a separate LLM call</u>

With ASI — 2 skill calls:

open_product_reviews("37")
search_reviews("satisfied")

Skill: open_product_review

```
def open_product_reviews(product_id):
    click(product_id)
    click("All Reviews")

```

Skill: search_reviews

```
def search_reviews(box, search_text)
    fill(box, search_text)
    click("Search")
```

Without Skills
5 actions
× 5 LLM calls

→

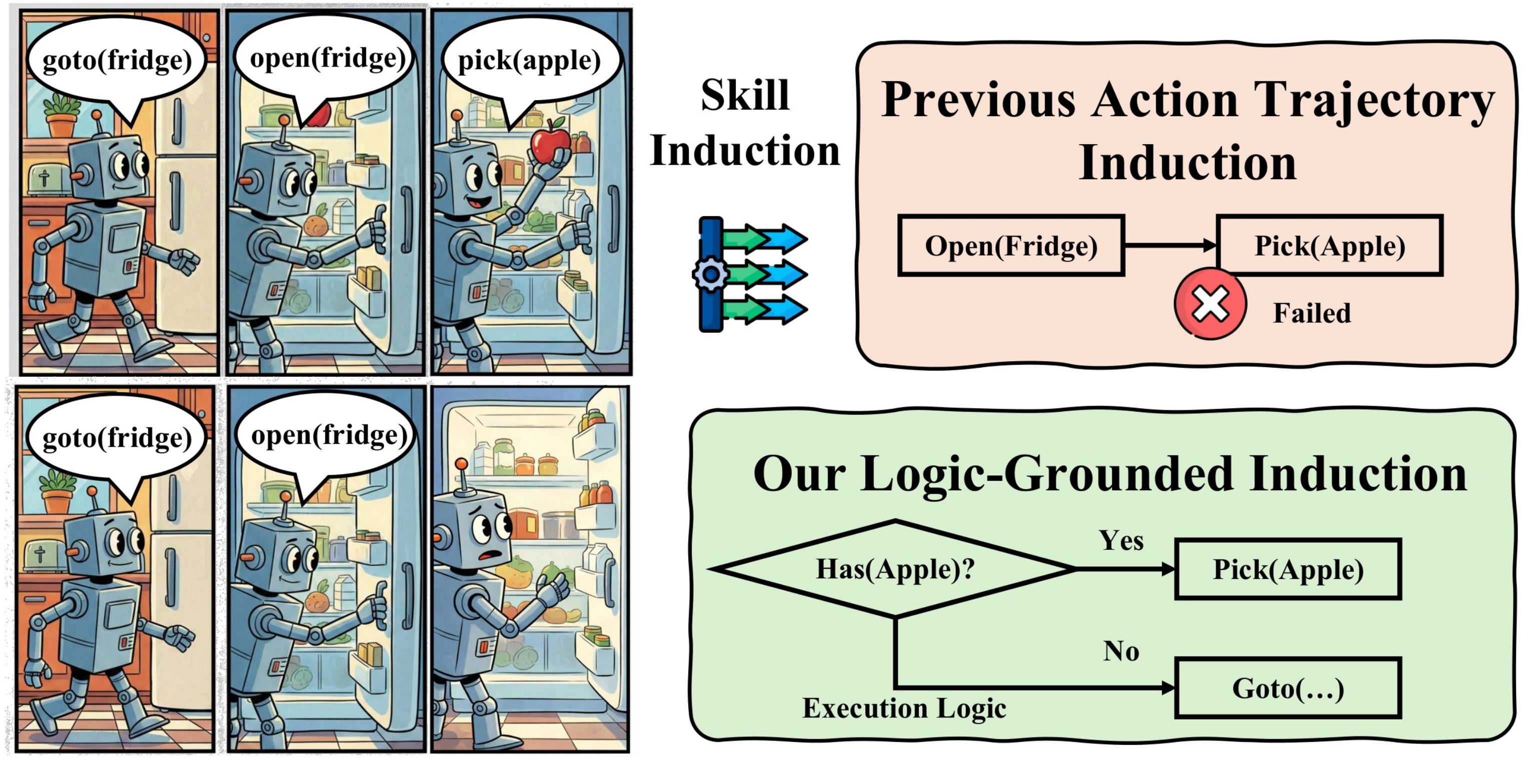
With ASI Skills
5 actions -> 2 skills
× 3 LLM calls

State-Blind Scripts Fail in New States

State-blind script: no state checks.

Seen state: Open → Pick succeeds.

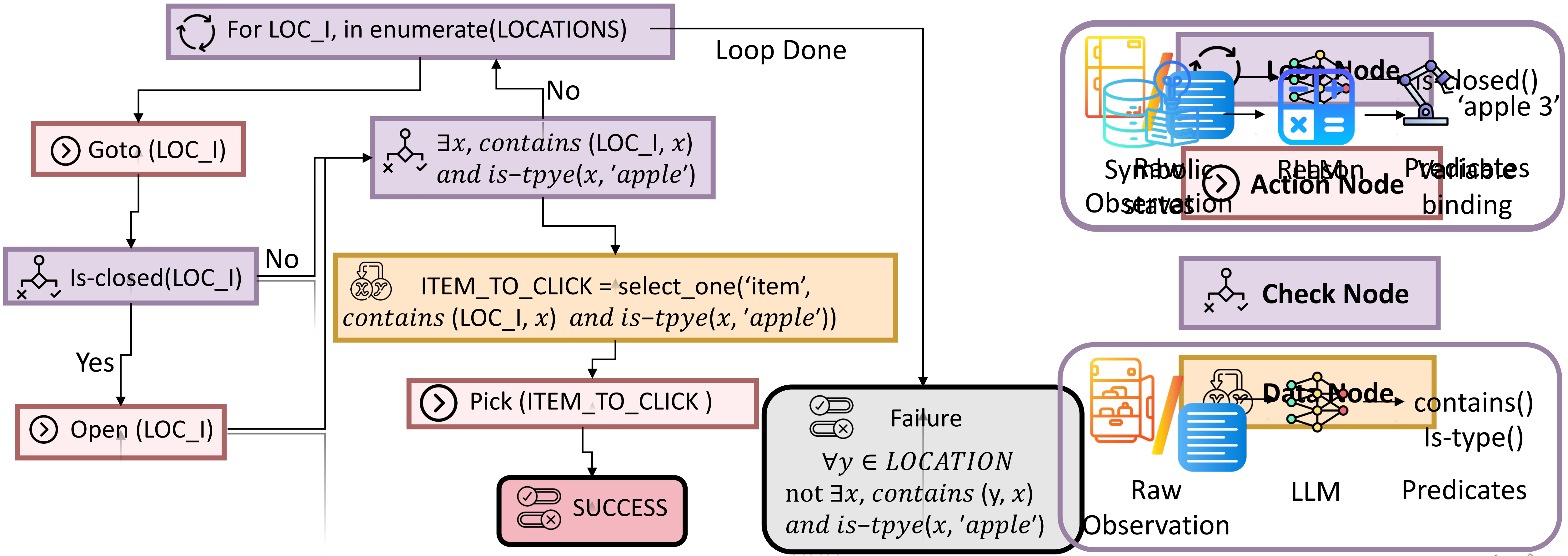
New state: the same script fails.



State-aware skill: check Has(Apple); pick if true, otherwise search.

Our approach: logic-grounded skill programs

Logic-Grounded Skills Encode Branches



Simple Skills with Broad State Support

Empirical support set

A trajectory contributes the states in which a candidate program achieves the target sub-goal.$\tau s_{h_g}, g' s_{h_{g'}} \models g' \tau$

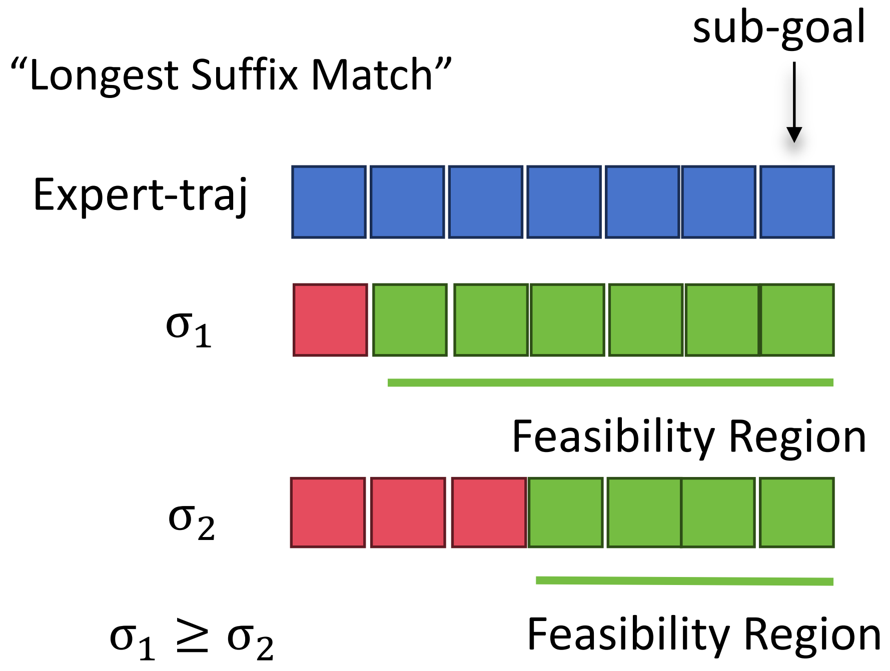
$$S_{g'}^{\tau} = \{ s_h \in \tau \mid 0 \leq h < h_{g'} \}$$

Empirical feasibility maximization

ERM + Complexity regularization

$$\max_{\sigma} \quad \left| \bigcup_{\tau \in \mathcal{D}} \widehat{\mathcal{R}}_{\sigma}^{\tau} \right| - |\sigma|$$

$$\widehat{\mathcal{R}}_{\sigma}^{\tau} \triangleq \left\{ s_h \in S_{g'}^{\tau} \;\middle|\; \exists \theta, \texttt{Consistent}(\tau, \sigma, \theta, h) = 1 \right\}.$$



NSI Consolidates Skills Across Traces

Empirical support set

A candidate skill is retained only when it expands feasible coverage under a complexity penalty.$\tau s_{h_{g'}}, g' s_{h_{g'}} \vDash g'\tau$

$$S_{g'}^{\tau} = \{ s_h \in \tau \mid 0 \le h < h_{g'} \}$$

Empirical feasibility maximization

ERM + Complexity regularization

$$\max_{\sigma} \quad \left| \bigcup_{\tau \in \mathcal{D}} \widehat{\mathcal{R}}_{\sigma}^{\tau} \right| - |\sigma|$$

$$\widehat{\mathcal{R}}_{\sigma}^{\tau} \triangleq \left\{ s_h \in S_{g'}^{\tau} \;\middle|\; \exists \theta, \texttt{Consistent}(\tau, \sigma, \theta, h) = 1 \right\}.$$

LLM Induction

$$\sigma^{t+1} \leftarrow LLM\ (\sigma^{t}, Error - Feedback)$$

Stage 1: Intra-Trajectory Consolidation

Initialization

Trace-to-code programs

Stage 2: Inter-Trajectory Generalization

Merge local experts into global skill

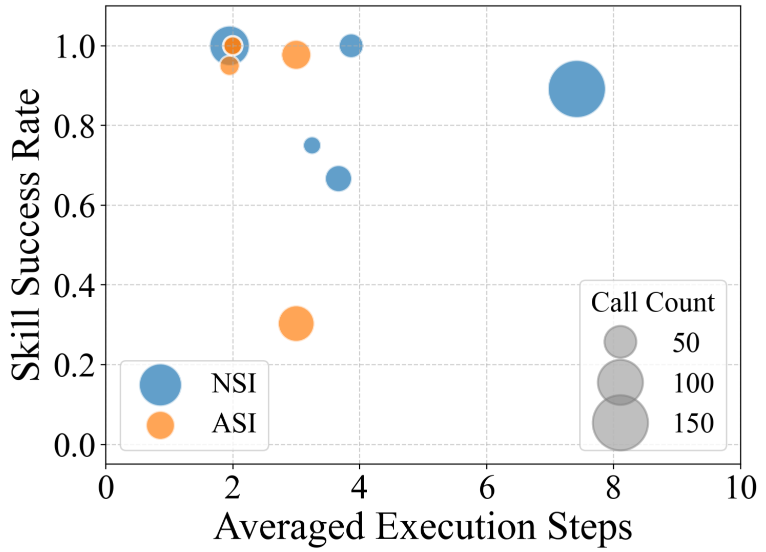
Greedy optimization with Feasibility Dominance

Accept only if coverage strictly expands

NSI Compresses Long-Horizon Tasks

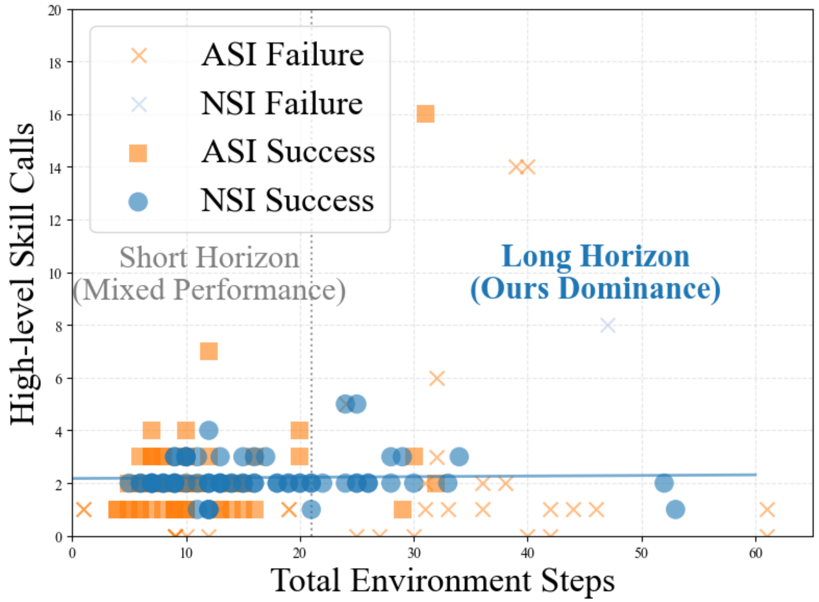
Horizon compression + state-aware execution

Execution Efficiency (ALFWorld)



Each skill encapsulates about 7.4 atomic actions, reducing planning steps and LLM calls.

Long-Horizon Robustness (ALFWorld)



ASI degrades beyond 22 steps; NSI remains effective beyond 53 steps.

This is evidence of robustness—not a formal guarantee of correct predicate grounding.

Lifting Traces to Logic: Results

- **Contribution:** Lifts long-horizon interaction traces into logic-grounded skills, keeping agent behavior consistent with state, preconditions, and physical constraints.
- **Role:** The symbolic layer serves as a verification layer between the LLM and the environment—checking preconditions, diagnosing failures, guiding repairs, and enabling skill reuse.
- **Implication:** Today's agentic skills often combine language and programs. NSI aligns language-based observations with programmatic execution logic, transforming experience into explicit, condition-aware execution procedures.

Selected references:

1. **Jie-Jing Shao**, Haiyan Yin, Yueming Lyu, Xingrui Yu, Lan-Zhe Guo, Ivor W. Tsang, James T. Kwok, Yu-Feng Li. **Lifting Traces to Logic: Programmatic Skill Induction with Neuro-Symbolic Learning for Long-Horizon Agentic Tasks**. **ICML 2026**.
2. **Jie-Jing Shao**, Hao-Ran Hao, Xiao-Wen Yang, and Yu-Feng Li. Abductive Learning for Neuro-Symbolic Grounded Imitation. **KDD 2025**.
3. **Jie-Jing Shao**, Hong-Jie You, Guohao Cai, Quanyu Dai, Zhenhua Dong, and Lan-Zhe Guo. Breaking the Self-Evaluation Barrier: Reinforced Neuro-Symbolic Planning with Large Language Models. **IJCAI 2025**.
4. Yang Chen, Hong-Jie You, **Jie-Jing Shao**, Xiao-Wen Yang, Ming Yang, Yu-Feng Li, Lan-Zhe Guo. **Re2 Agent: Reflective Rule Induction and Rule-Guided Refinement for Embodied Planning**. **NeurIPS 2025 Challenge on FM for Embodied Agents, Most Innovative Approach.**

Reliable LLM Agents Need Symbolic Structure

SPECIFY

Open goals
→ **executable constraints**

VALIDATE

Generated outputs
→ **objective checks**

EXECUTE

Past traces
→ **state-aware skills**

Semantic flexibility + explicit structure → reliable agent

Jie-Jing Shao | shaojj@lamda.nju.edu.cn | sh-jj.github.io