南京大学
NANJING UNIVERSITY

人工智能导论

神经符号AI
(Neuro-Symbolic AI)

郭兰哲

南京大学 智能科学与技术学院

https://www.lamda.nju.edu.cn/guolz/IntroAI/fall2025/index.html

Email: guolz@nju.edu.cn

国际规划竞赛IPC(International Planning Competition)

- International Planning Competition，隶属ICAPS (International Conference on Automated Planning and Scheduling)会议
- 始于1998年，通常每2-3年举办一次
- 统一采用PDDL语言 作为领域标准
- 里程碑规划器：
  - **FF (Fast Forward) (IPC-2000)**：引入了忽略删除列表的启发式规划，现代规划器的鼻祖
  - **Fast Downward (IPC-2004)**：引入了 SAS+ 翻译和多启发式搜索架构，至今仍是学术界的基础框架
- 新趋势：
  - **HTN 复兴：分层规划重新引起重视**
  - **RL + Planning：传统搜索仍然强势，但RL正在冲击榜单**
  - **LLM+Planning：开始尝试基于大模型做规划任务**

https://ipc2023.github.io/

The International Planning Competition 2023

- Classical Tracks
  - Daniel Fišer, Saarland University
  - Florian Pommerening, University of Basel
- Learning Tracks
  - Jendrik Seipp, Linköping University
  - Javier Segovia-Aguas, Universitat Pompeu Fabra
- Probabilistic Tracks
  - Ayal Taitler, University of Toronto
  - Scott Sanner, University of Toronto
- Numeric Tracks
  - Joan Espasa Arxer, University of St Andrews
  - Enrico Scala, University of Brescia
- HTN Tracks
  - Ron Alford, MITRE
  - Dominik Schreiber, Karlsruhe Institute of Technology
  - Gregor Behnke, University of Amsterdam

LLM for Planning

Published in Transactions on Machine Learning Research (April/2025)

A Systematic Evaluation of the Planning and Scheduling Abilities of the Reasoning Model o1

Domain	Shots	Claude Models		OpenAI GPT-4 Models				LLaMA Models		Gemini Models	
		3.5 Sonnet	**3 Opus**	**4o**	**4o mini**	**4**	**4 Turbo**	**3.1 405B**	**3 70B**	**1.5 Pro**	**1 Pro**
Blocks world	One Shot	**346/600 (57.6%)**	289/600 (48.1%)	170/600 (28.3%)	49/600 (8.1%)	206/600 (34.3%)	138/600 (23%)	284/600 (47.3%)	76/600 (12.6%)	101/600 (16.8%)	68/600 (11.3%)
	Zero Shot	329/600 (54.8%)	356/600 (59.3%)	213/600 (35.5%)	53/600 (8.8%)	210/600 (34.6%)	241/600 (40.1%)	**376/600 (62.6%)**	205/600 (34.16%)	143/600 (23.8%)	3/600 (0.5%)
Mystery Blocks world	One Shot	19/600 (3.1%)	8/600 (1.3%)	5/600 (0.83%)	0/600 (0%)	**26/600 (4.3%)**	5/600 (0.83%)	21/600 (3.5%)	15/600 (2.5%)	-	2/500 (0.4%)
	Zero Shot	0/600 (0%)	0/600 (0%)	0/600 (0%)	0/600 (0%)	1/600 (0.16%)	1/600 (0.16%)	**5/600 (0.8%)**	0/600 (0%)	-	0/500 (0%)

https://openreview.net/pdf?id=FkKBxp0FhR

LLM for Planning

PlanBench: An Extensible Benchmark for Evaluating Large Language Models on Planning and Reasoning about Change

Karthik Valmeekam
School of Computing & AI
Arizona State University, Tempe.
kvalmeek@asu.edu

Matthew Marquez
School of Computing & AI
Arizona State University, Tempe.
mmarqu22@asu.edu

Alberto Olmo
School of Computing & AI
Arizona State University, Tempe.
aolmoher@asu.edu

Sarath Sreedharan*
Department of Computer Science,
Colorado State University, Fort Collins.
sarath.sreedharan@colostate.edu

Subbarao Kambhampati
School of Computing & AI
Arizona State University, Tempe.
rao@asu.edu

https://arxiv.org/pdf/2206.10498

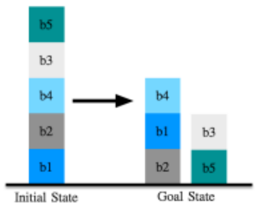
Task	Instances correct	
	GPT-4	I-GPT3
Plan Generation We showcase an instance and the respective plan as an example and prompt the machine with a new instance.	206/600 (34.3%)	41/600 (6.8%)
Cost-Optimal Planning We showcase an instance, the respective optimal plan and the associated cost as an example and prompt the machine with a new instance.	198/600 (33%)	35/600 (5.8%)
Plan Verification We showcase three instances and three distinct plans (goal reaching, non goal-reaching and inexecutable) and present the respective validation and explanations. We then present a new instance and a plan and ask the machine for to verify and provide an explanation, if needed.	352/600 (58.6%)	72/600 (12%)
Reasoning About Plan Execution We showcase an instance, an action sequence and the corresponding resulting state after executing the action sequence as an example. We then provide an instance and an executable action sequence and ask the machine to provide the resulting state.	191/600 (31.8%)	4/600 (0.6%)
Replanning We showcase an instance, the respective plan and present an unexpected change of the state. We then also present a new plan from the changed state. Finally, for a new instance we repeat the same except we ask the machine for the new plan.	289/600 (48.1%)	40/600 (6.6%)
Plan Generalization We showcase an instance and the respective plan as an example and prompt the machine with a new instance. The plans for both the instances can be generated by a fixed program containing loops and conditionals.	141/500 (28.2%)	49/500 (9.8%)
Plan Reuse We showcase an instance and the respective plan as an example and prompt the machine with an instance which requires only a certain prefix of the plan provided in the example.	392/600 (65.3%)	102/600 (17%)
Robustness to Goal Reformulation (Shuffling goal predicates) We showcase an instance and the respective plan as an example and prompt the machine with the same instance but shuffle the ordering of the goals.	461/600 (76.8%)	467/600 (77.8%)
Robustness to Goal Reformulation (Full → Partial) We showcase an instance with a fully specified goal state and the respective plan as an example and prompt the machine with the same instance but provide a partially specified goal state.	522/600 (87%)	467/600 (77.8%)
Robustness to Goal Reformulation (Partial → Full) We showcase an instance with a partially specified goal state and the respective plan as an example and prompt the machine with the same instance but provide a fully specified goal state.	348/600 (58%)	363/600 (60.5%)

前沿趋势：LLM + PDDL

LLM充当自然语言到形式化语言的翻译器，调用符号求解工具完成规划任务

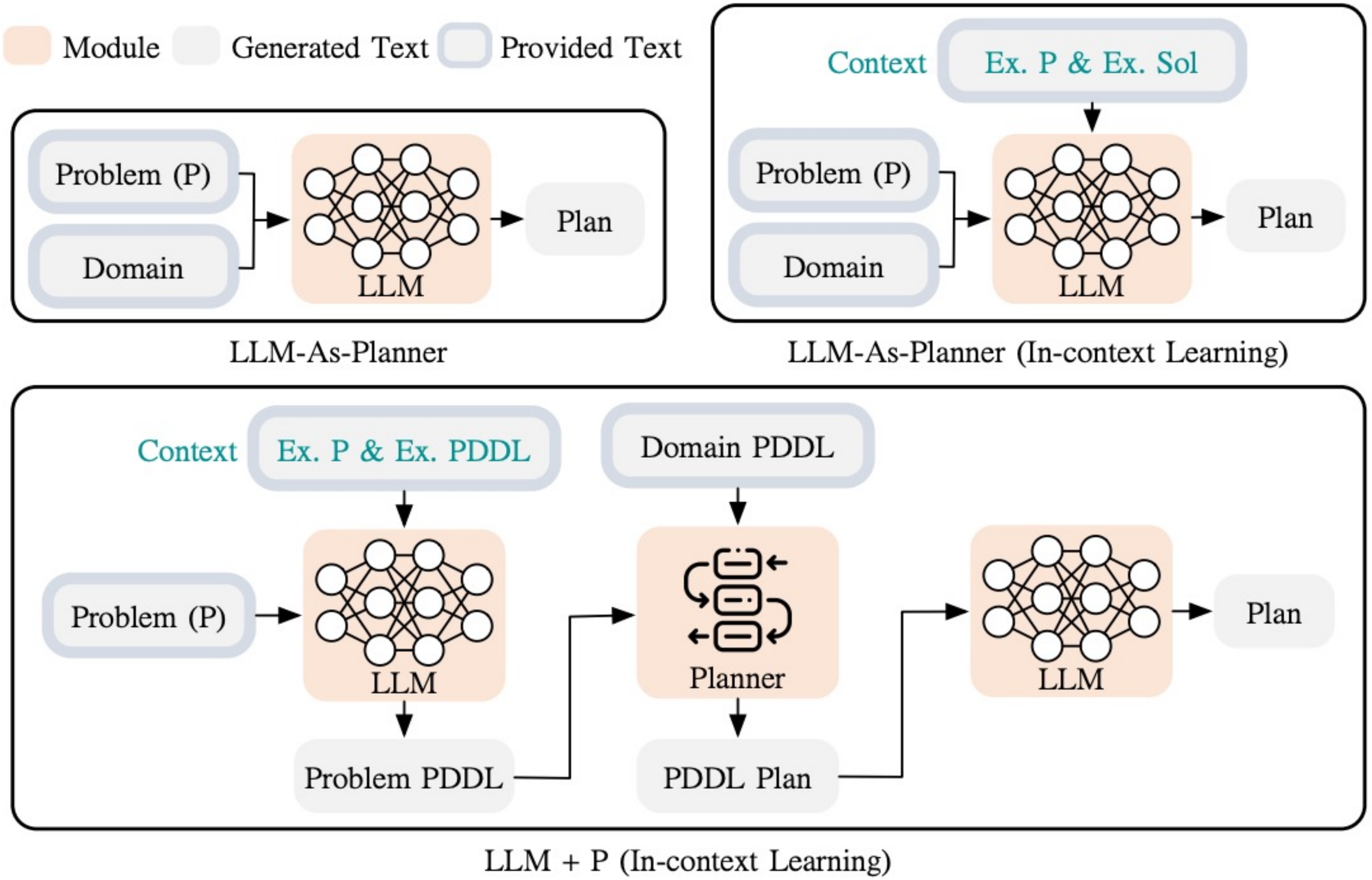
A Failure Example of GPT-4 in Planning

Problem (P1): You have 5 blocks. One cannot place more than one block on another block. b5 is on top of b3. b4 is on top of b2. b2 is on top of b1. b3 is on top of b4. b1 is on the table. b5 is clear. Your arm is empty.
Your goal is to move the blocks.
b1 should be on top of b2.
b3 should be on top of b5.
b4 should be on top of b1.



GPT-4:

```
1. Move b5 from b3 to the table.
2. Move b3 from b4 to b5.
3. Move b4 from b2 to the table.
4. Move b2 from b1 to b5.
5. Move b1 from the table to b2.
6. Move b4 from the table to b1.
```



LLM+P: Empowering Large Language Models with Optimal Planning Proficiency. https://arxiv.org/pdf/2304.11477

前沿趋势：LLM + PDDL

LLM充当自然语言到形式化语言的翻译器，调用符号求解工具完成规划任务

(a) grasp bottle (b) free gripper (c) grasp soup can (d) place soup can (e) re-grasp bottle (f) place bottle

Fig. 2: Demonstration of the optimal tidy-up plan. The robot starts at the coffee table and 1) picks up the bottle, 2) navigates to a room with the side table and the recycle bin, 3) puts down the bottle, 4) grasps the soup can, 5) puts the soup can in the recycle bin, 6) re-grasps the bottle, 7) navigates to the kitchen, 8) places the bottle in the pantry.

LLM+P: Empowering Large Language Models with Optimal Planning Proficiency. https://arxiv.org/pdf/2304.11477

符号主义人工智能的痛点：符号接地

Symbolic AI 的核心假设是：**智能是对符号的操作**

- **离散性与确定性**：On(Block A, Block B) 是一个非常明确、没有歧义的状态
- **组合性**：符号可以无限组合。学会了“拿”，学会了“苹果”，就能理解“拿苹果”，不需要重新训练
- **长程推理能力**：在数学证明、规划、逻辑谜题中，符号系统可以进行几十甚至上百步的推理
- **经典AI的困境**：对于一个纯符号系统，它知道 Apple 是一种 fruit，fruit 可以 eat。但如果你给它看一张苹果的照片，或者让机器人去抓一个苹果，它完全不知道这个符号对应现实中的哪一个物体
- **鸿沟**：**现实世界是连续的、嘈杂的（像素、声波、电压），而符号世界是离散的、抽象的，Symbolic AI缺乏一个将“连续信号”映射为“离散符号”的转换器**

THE SYMBOL GROUNDING PROBLEM

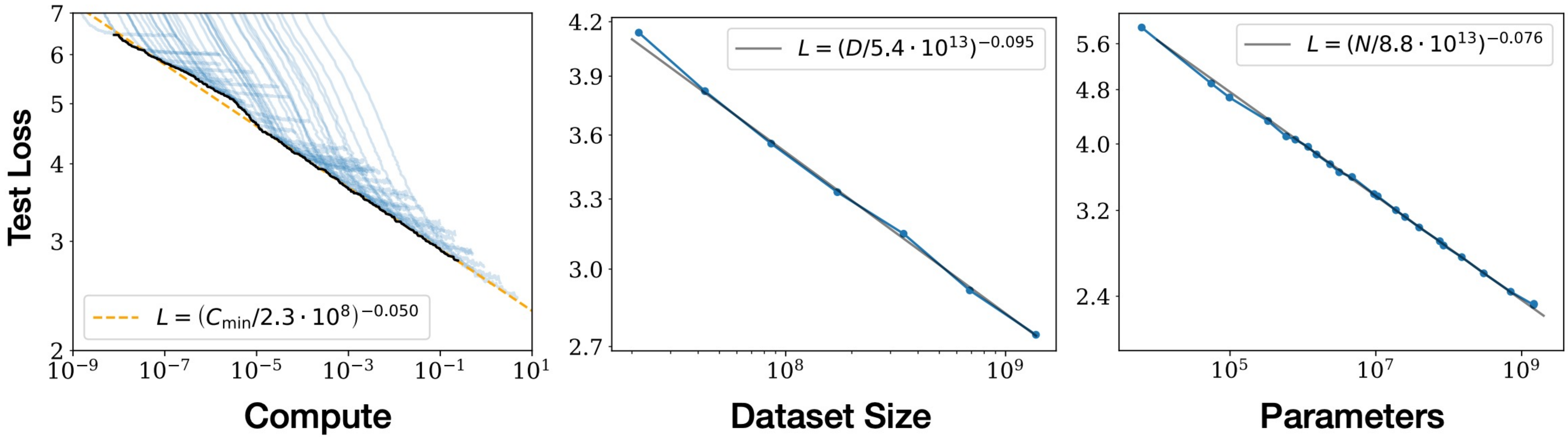
Stevan Harnad
Department of Psychology
Princeton University
Princeton NJ 08544
harnad@cogsci.soton.ac.uk

ABSTRACT: There has been much discussion recently about the scope and limits of purely symbolic models of the mind and about the proper role of connectionism in cognitive modeling. This paper describes the "symbol grounding problem": How can the semantic interpretation of a formal symbol system be made intrinsic to the system, rather than just parasitic on the meanings in our heads? How can the meanings of the meaningless symbol tokens, manipulated solely on the basis of their (arbitrary) shapes, be grounded in anything but other meaningless symbols? The problem is analogous to trying to learn Chinese from a Chinese/Chinese dictionary alone. A candidate solution is sketched: Symbolic representations must be grounded bottom–up in nonsymbolic representations of two kinds: (1) "iconic representations" , which are analogs of the proximal sensory projections of distal objects and events, and (2) "categorical representations" , which are learned and innate feature–detectors that pick out the invariant features of object and event categories from their sensory projections. Elementary symbols are the names of these object and event categories, assigned on the basis of their (nonsymbolic) categorical representations. Higher–order (3) "symbolic representations" , grounded in these elementary symbols, consist of symbol strings describing category membership relations (e.g., "An X is a Y that is Z"). Connectionism is one natural candidate for the mechanism that learns the invariant features underlying categorical representations, thereby connecting names to the proximal projections of the distal objects they stand for. In this way connectionism can be seen as a complementary component in a hybrid nonsymbolic/symbolic model of the mind, rather than a rival to purely symbolic modeling. Such a hybrid model would not have an autonomous symbolic "module," however; the symbolic functions would emerge as an intrinsically "dedicated" symbol system as a consequence of the bottom–up grounding of categories' names in their sensory representations. Symbol manipulation would be governed not just by the arbitrary shapes of the symbol tokens, but by the nonarbitrary shapes of the icons and category invariants in which they are grounded.

https://arxiv.org/html/cs/9906002

观点1：仅依赖大模型

大语言模型、多模态大模型已经展现初步的推理能力，尽管还存在幻觉、不可靠等问题，但只要能够持续Scaling，就能实现性能提升

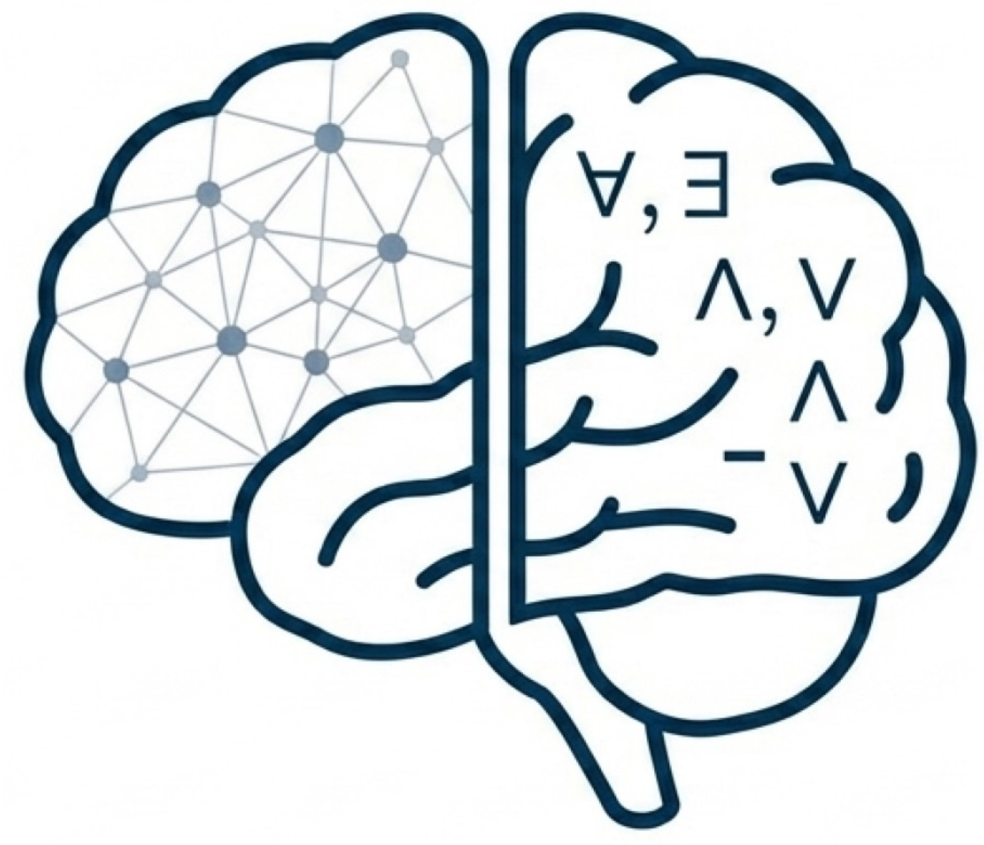


More data, More compute, Larger models

观点2：Neuro-Symbolic AI

Neural Networks

神经网络的感知、识别能力 (System 1)



Symbolic

符号推理的严谨逻辑、可靠性、可解释性

Neuro-Symbolic AI: 结合两种AI范式的优势，创造即聪明又可靠的下一代人工智能系统

快思考与慢思考(系统1与系统2)

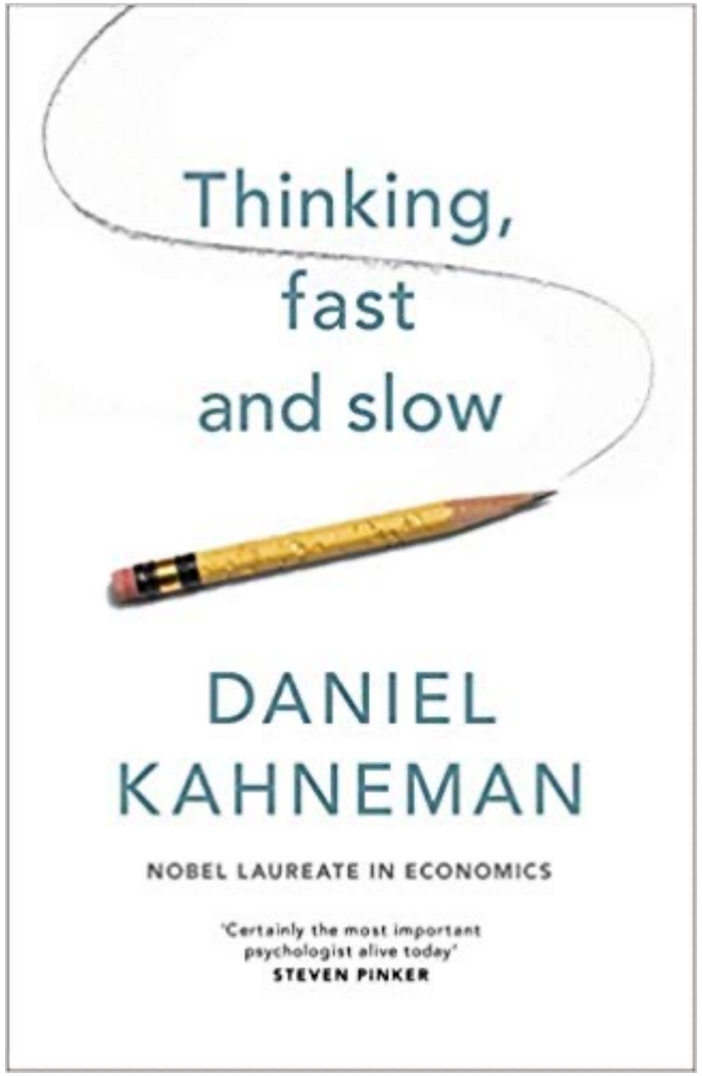
认知科学中的猜想：人类的认知分为系统1与系统2 [Kahneman, 2011]

你能多快判断一下命题的正误

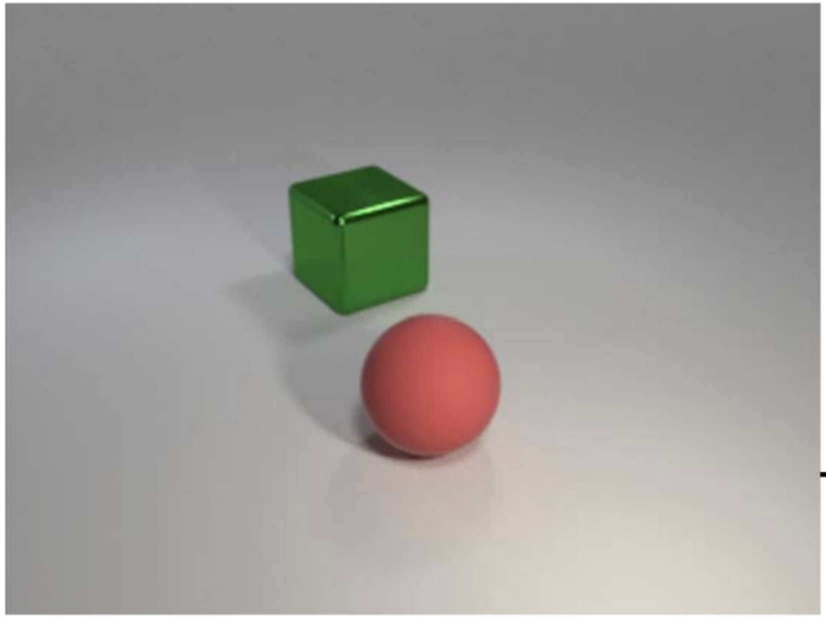
- 1 + 1 = 2
- 12345 * 54321 = 670592745
- 窗外的树有些是绿色的
- OOOOO ← 这里有5个圈
- OOOOOOOOOOOOOOOOOO ← 这里有18个圈
- “我只给自己不理发的人理发，所以我可以给自己理发”

快思考与慢思考(系统1与系统2)

- **Human Cognition:** System 1 (Neural Network) + System 2(Symbolic Reasoning)
- Neuro-Symbolic AI: Combine true symbolic reasoning with a neural network
- **Neural Network for Perception**
- **Symbolic for Rigorous Reasoning**



End-to-End Visual Reasoning



Visual Question Answering

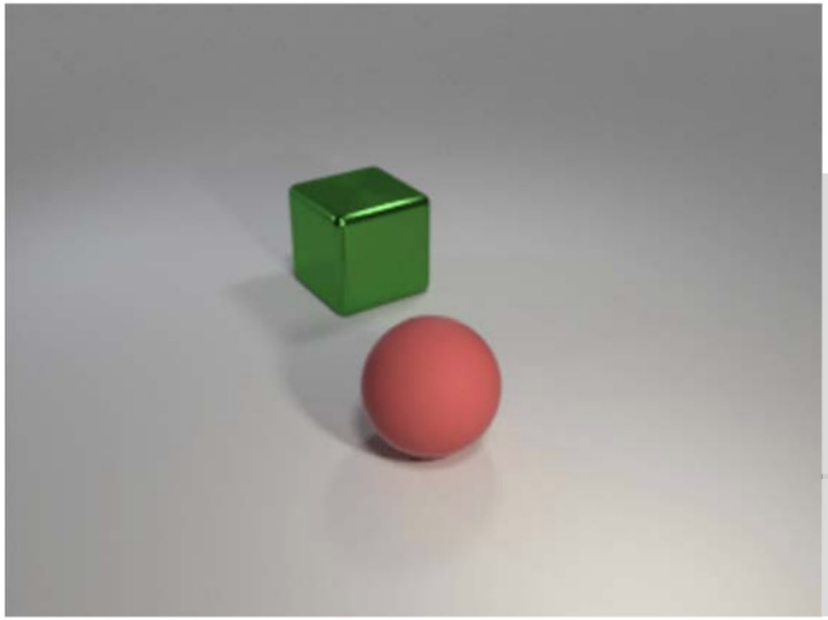
Q: What's the **shape** of the **red** object?

End-to-End
Neural Network

A: Sphere.

NMN [Andreas et al., 2016]
IEP [Johnson et al., 2017]
FiLM [Perez et al., 2018],
MAC [Hudson & Manning, 2018]
Stack-NMN [Hu et al., 2018]
TbD [Mascharka et al. 2018]

End-to-End Visual Reasoning

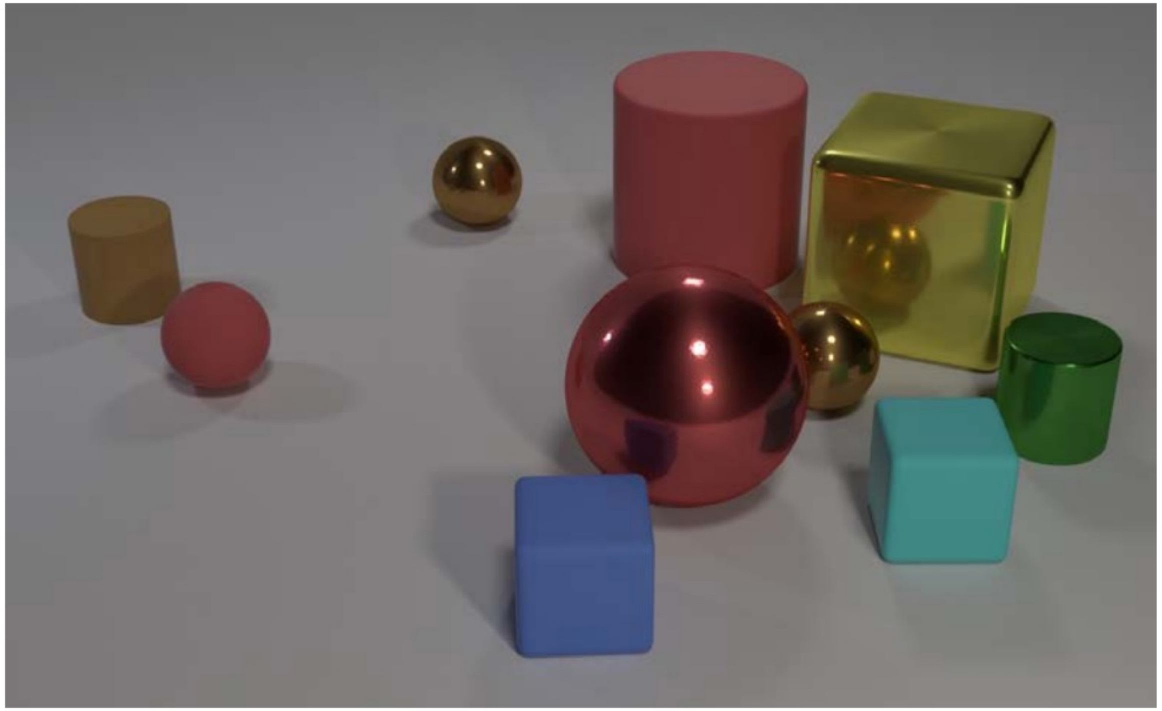


Visual Question Answering
Q: What's the **shape** of the **red** object?

Entangled
Concept ⟷ Reasoning
(e.g., colors, shapes) (e.g., count)

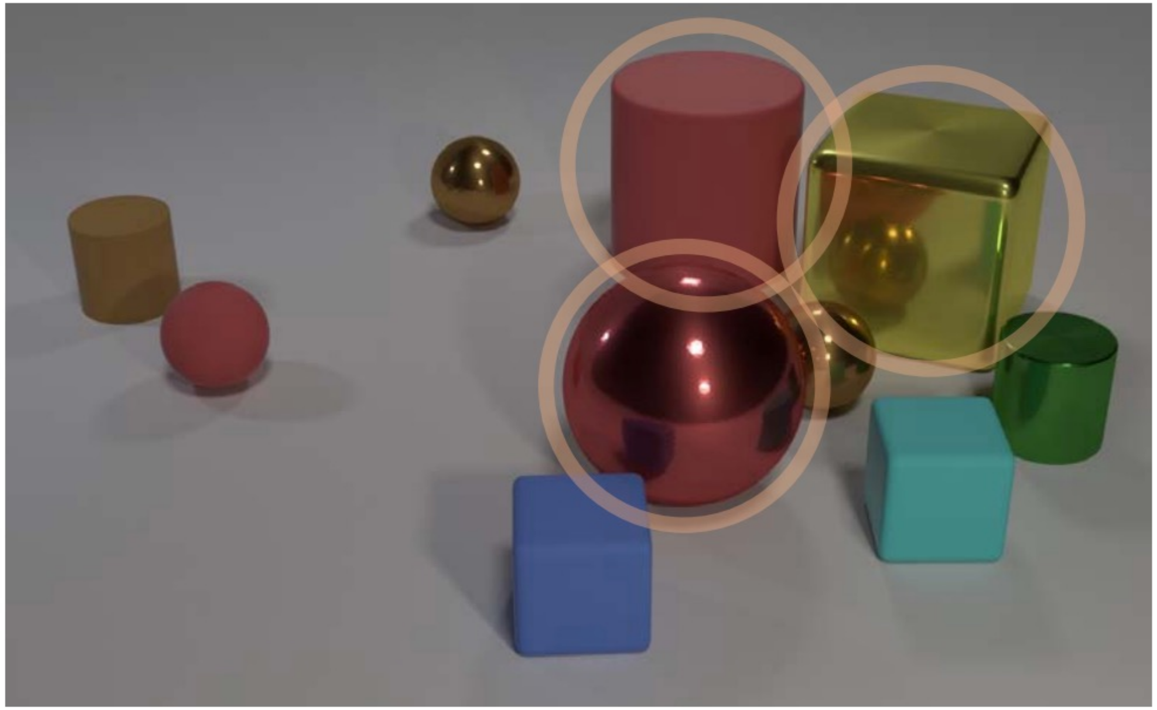
NMN [Andreas et al., 2016]
IEP [Johnson et al., 2017]
FiLM [Perez et al., 2018],
MAC [Hudson & Manning, 2018]
Stack-NMN [Hu et al., 2018]
TbD [Mascharka et al. 2018]

Human-Like Reasoning

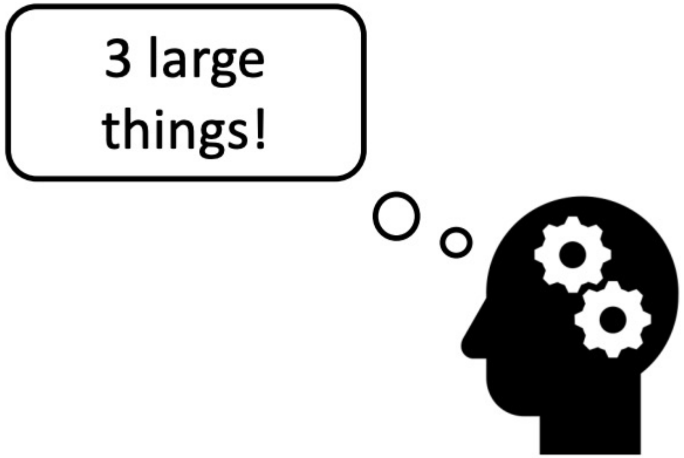


Question: *Are there an equal number of large things and metal spheres?*

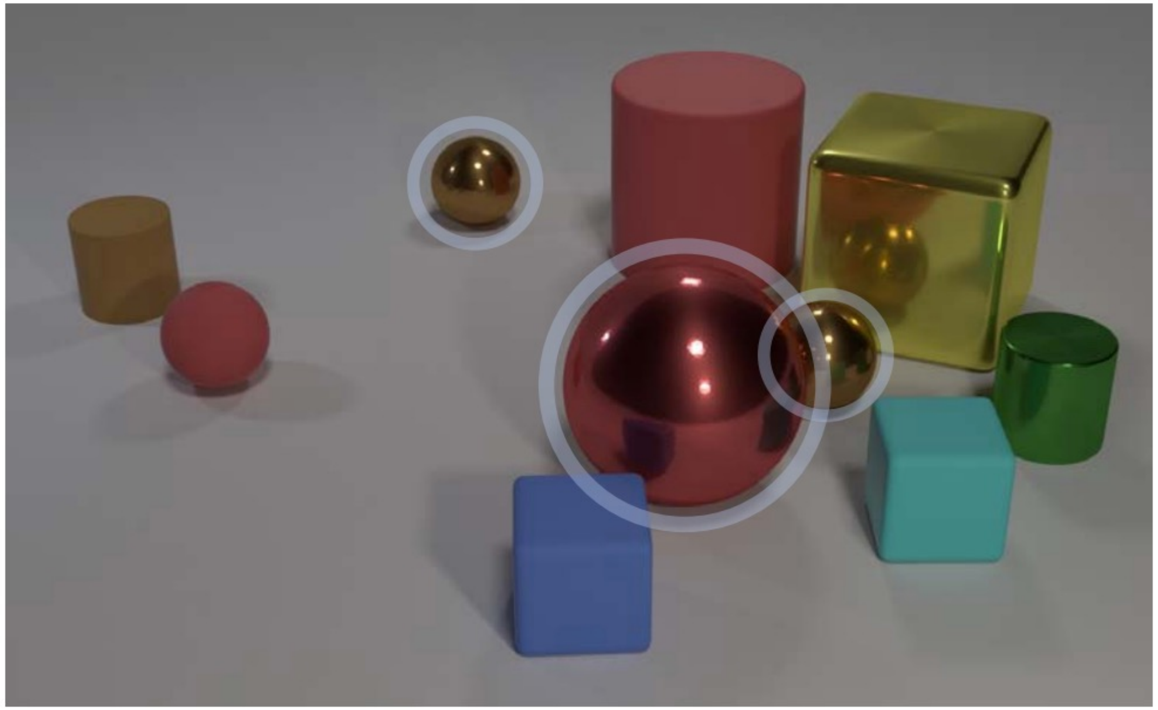
Human-Like Reasoning



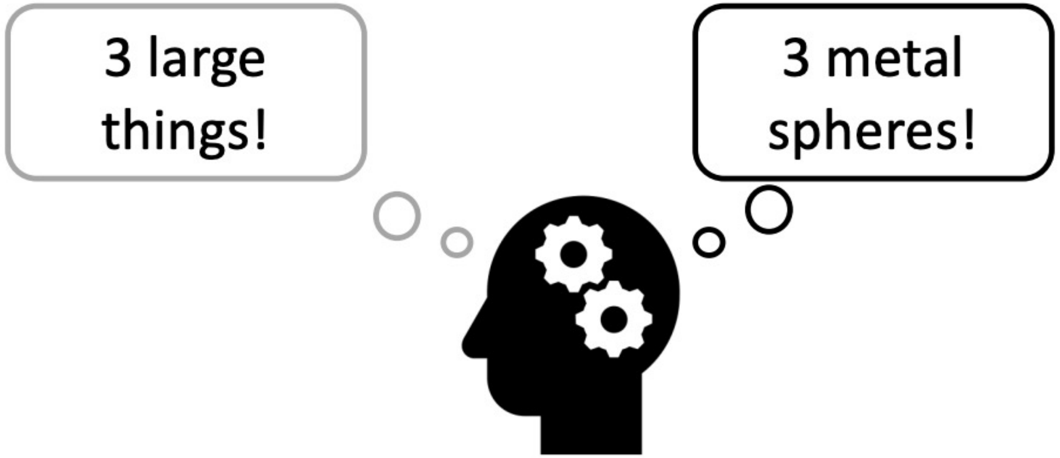
Question: *Are there an equal number of **large things** and metal spheres?*



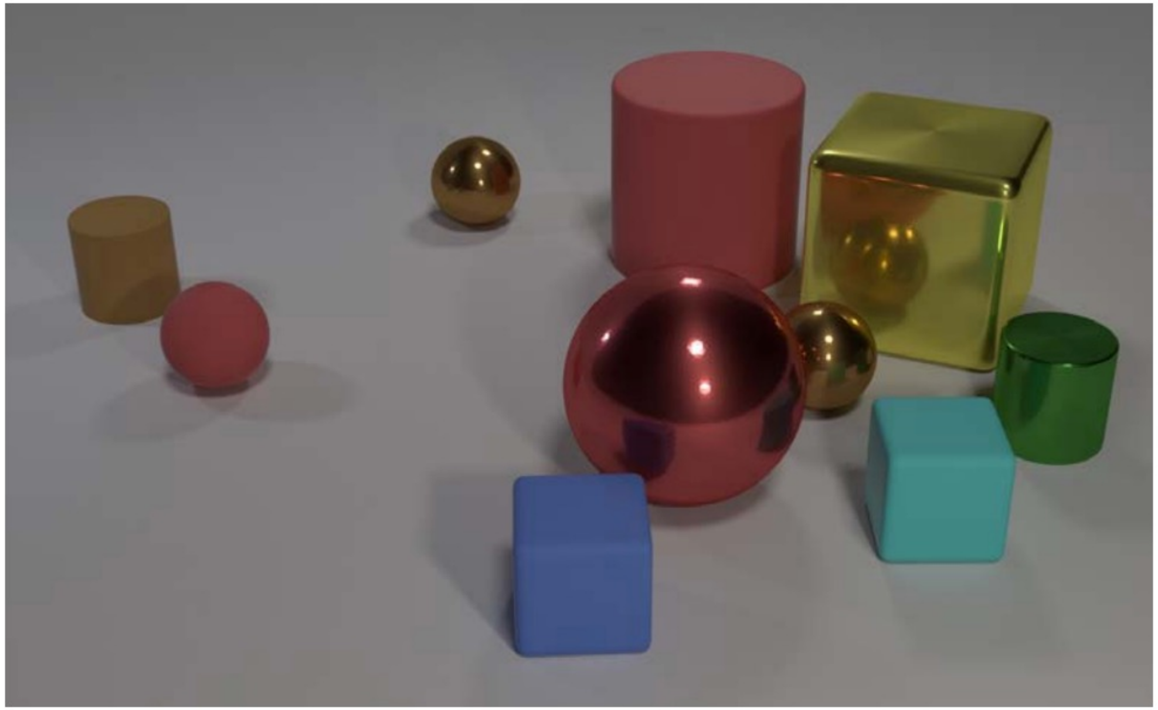
Human-Like Reasoning



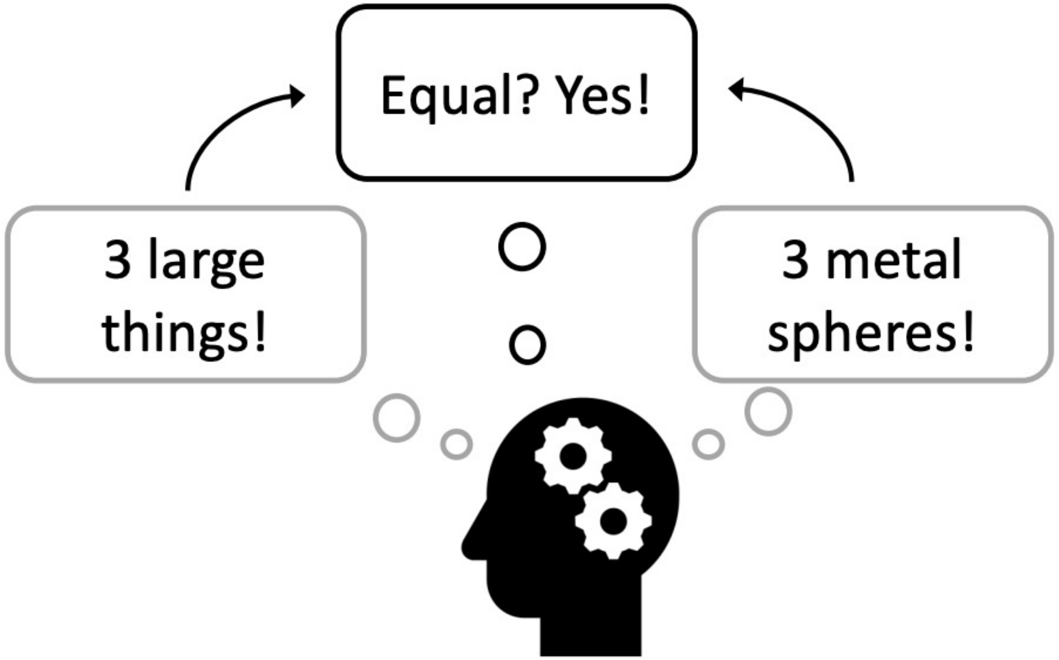
Question: *Are there an equal number of large things and **metal spheres**?*



Human-Like Reasoning



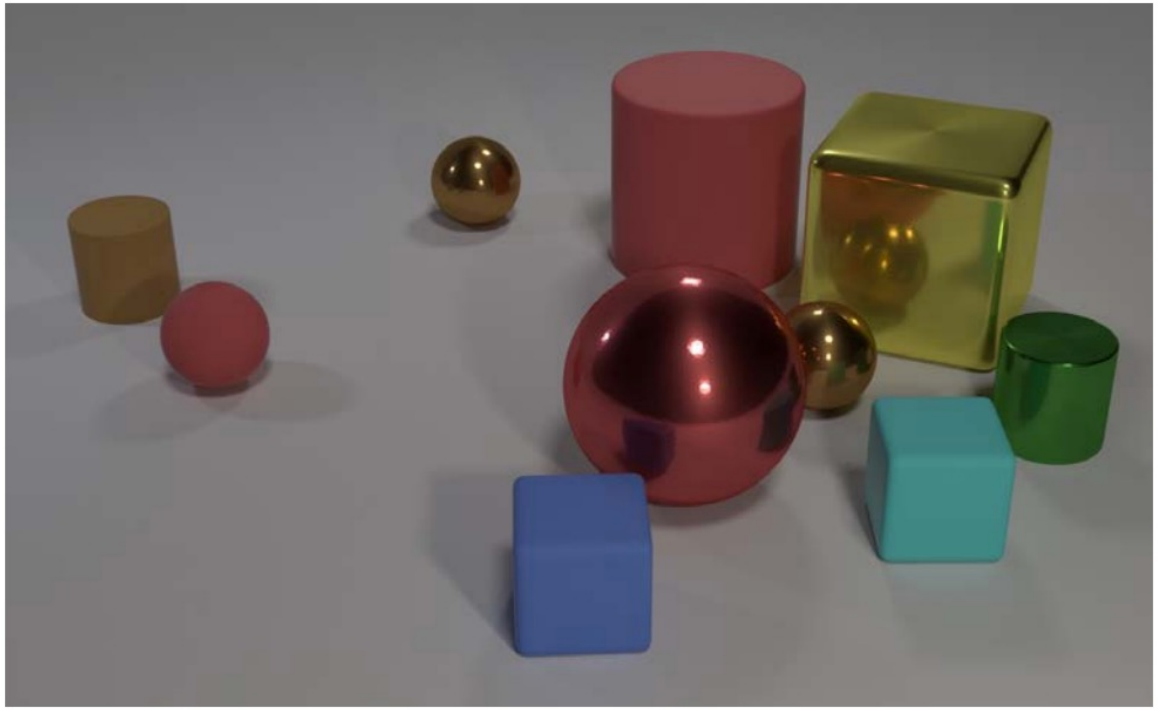
Question: *Are there an* ***equal number*** *of large things and metal spheres?*



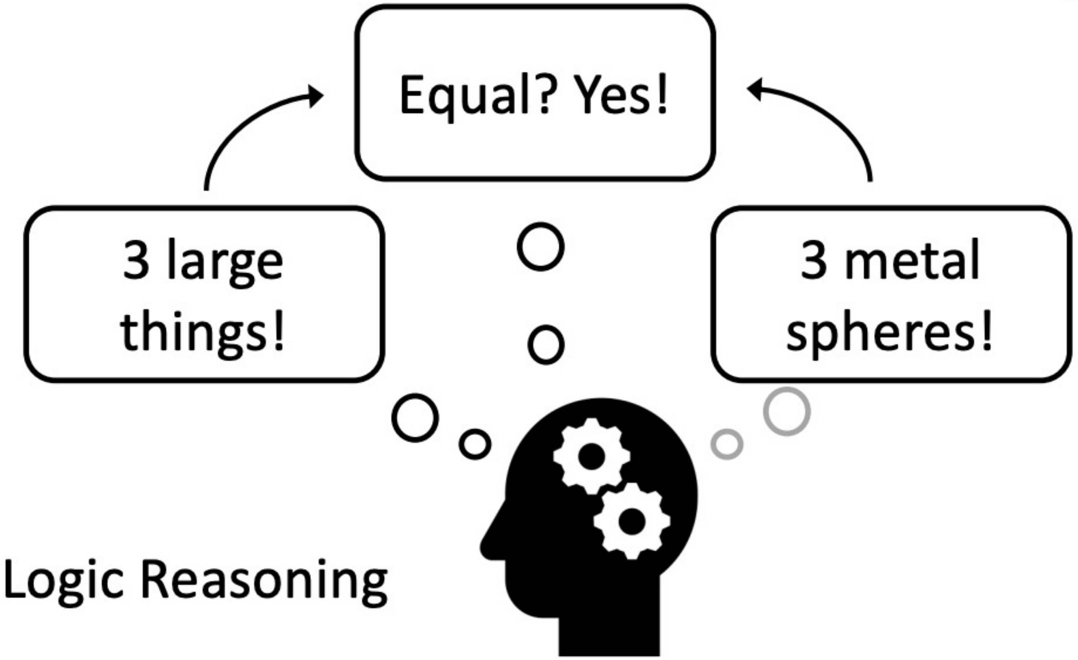
Human-Like Reasoning

Question Understanding

Question: *Are there an equal number of large things and metal spheres?*



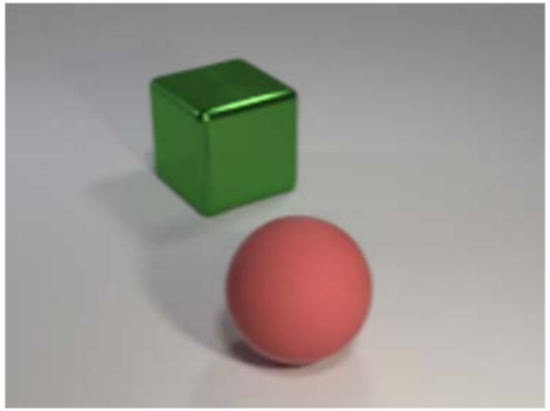
Visual Perception



Logic Reasoning

Neuro-Symbolic AI

Vision



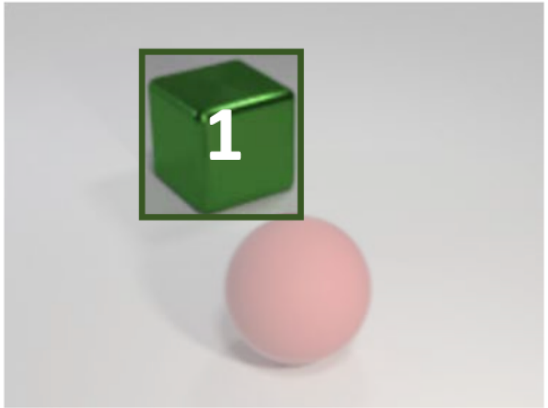
Scene Parsing →

Language

Q: What's the **shape** of the **red** object?

Neuro-Symbolic AI

Vision



Scene Parsing

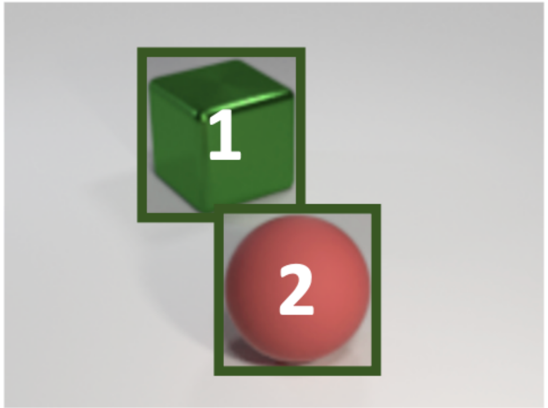
ID	Color	Shape	Material
1	Green	Cube	Metal

Language

Q: What's the **shape** of the **red** object?

Neuro-Symbolic AI

Vision



Scene Parsing →

ID	Color	Shape	Material
1	Green	Cube	Metal
2	Red	Sphere	Rubber

Language

Q: What's the **shape** of the **red** object?

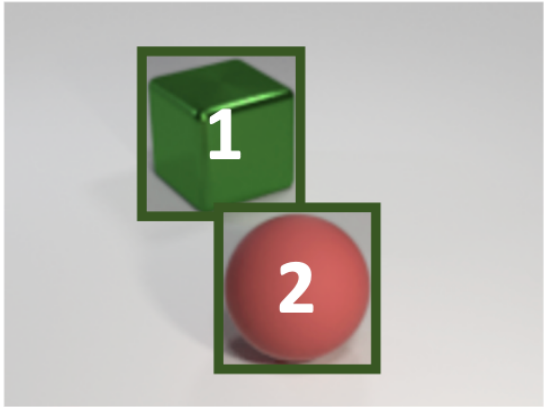
Semantic Parsing →

```
Filter(Red)
    ↓
Query(Shape)
```

Program

Neuro-Symbolic AI

Vision



Scene Parsing

ID	Color	Shape	Material
1	Green	Cube	Metal
2	Red	Sphere	Rubber

Symbolic Reasoning

Language

Q: What's the **shape** of the **red** object?

Semantic Parsing

```
Filter(Red)
↓
Query(Shape)
```

Program

Neuro-Symbolic AI

Vision

Scene Parsing

ID	Color	Shape	Material
1	Green	Cube	Metal
2	Red	Sphere	Rubber

Symbolic Reasoning

Language

Q: What's the shape of the red object?

Semantic Parsing

```
Filter(Red)
↓
Query(Shape)
```

Program

Neuro-Symbolic AI

Vision

Scene Parsing

ID	Color	Shape	Material
1	Green	Cube	Metal
2	*Red*	Sphere	Rubber

Symbolic Reasoning

Language

Q: What's the shape of the red object?

Semantic Parsing

```
Filter(Red)
Query(Shape)
```

Program

Neuro-Symbolic AI

Vision

Scene Parsing

ID	Color	Shape	Material
1	Green	Cube	Metal
2	Red	*Sphere*	Rubber

Symbolic Reasoning

Language

Q: What's the shape of the red object?

Semantic Parsing

```
Filter(Red)
↓
Query(Shape)
```

Program

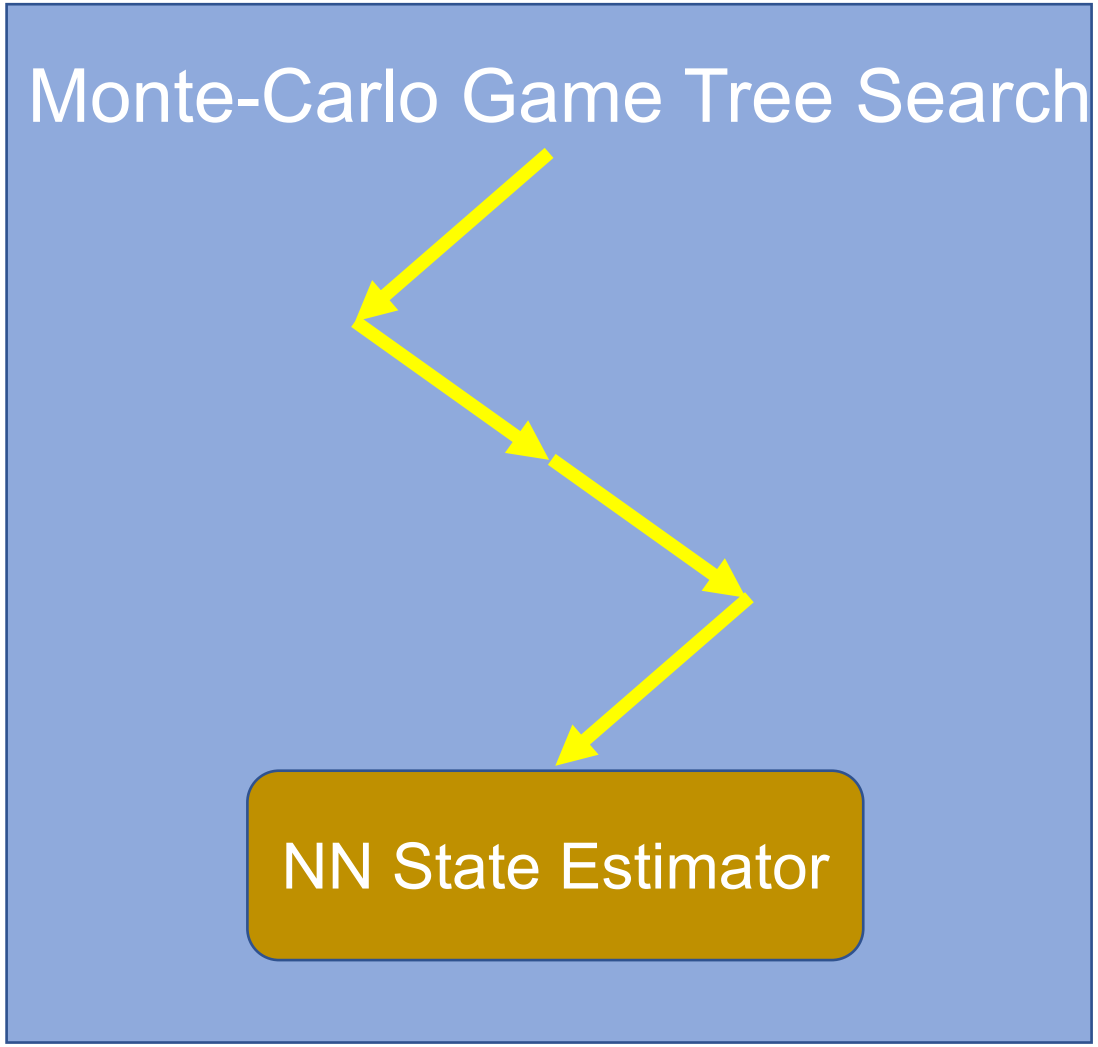
Sphere

Neuro-Symbolic AI (before LLM)

- Neuro helps Symbolic
- Neuro ↔️ Symoblic

Neuro for Symbolic

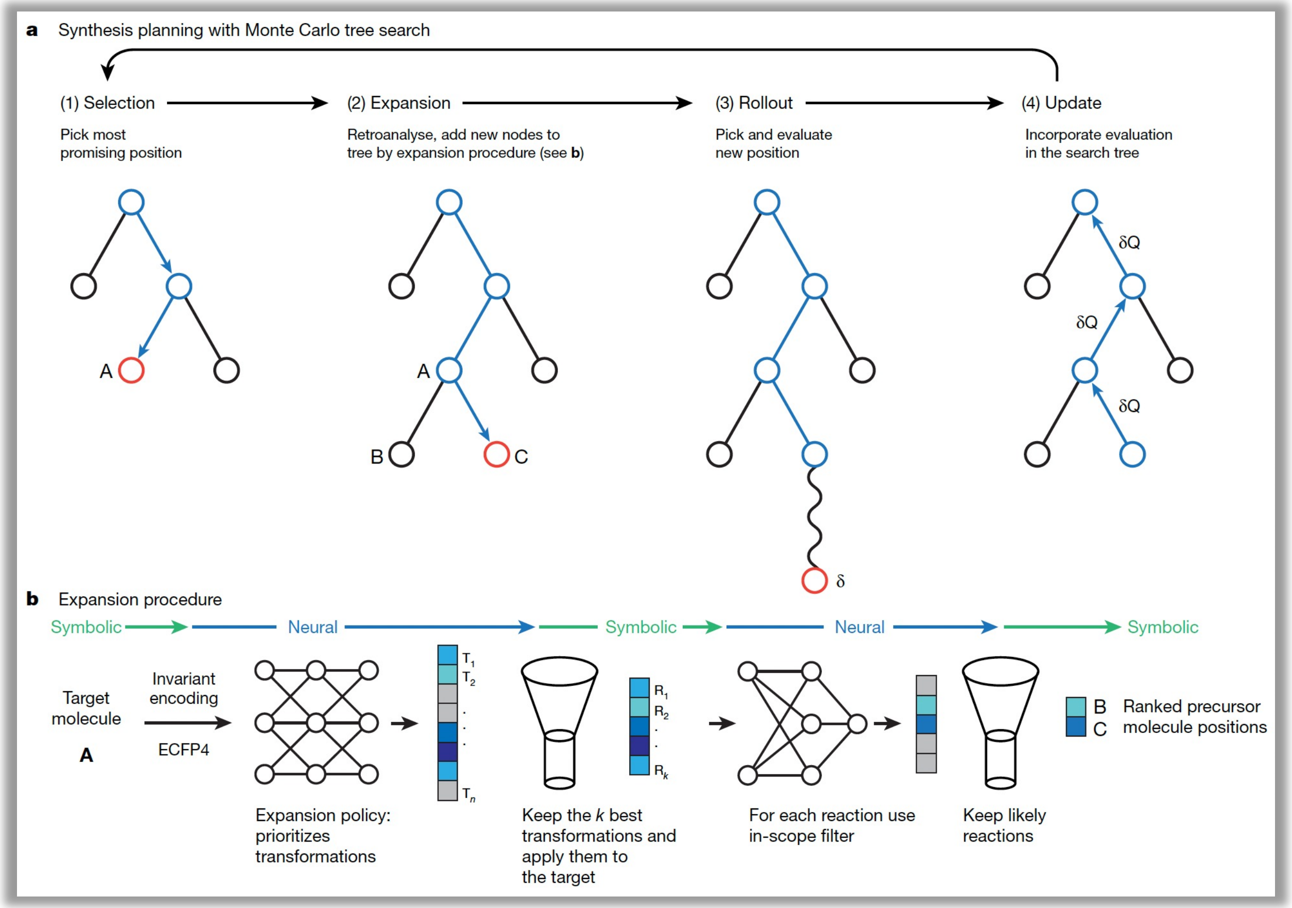
- AlphaGo(2016), AlphaZero(2017)
- 问题：传统符号搜索时间复杂度高
- 策略网络 (Policy Network)
  - 缩小搜索空间，只探索可能性大的动作
- 价值网络 (Value Network)
  - 节省大量模拟过程
- 神经网络提供启发式估计
- MCTS进行严谨符号化搜索
- 搜索结果辅助神经网络训练



Neuro for Symbolic

化学领域的“AlphaGo”：逆合成分析 (Retrosynthesis)

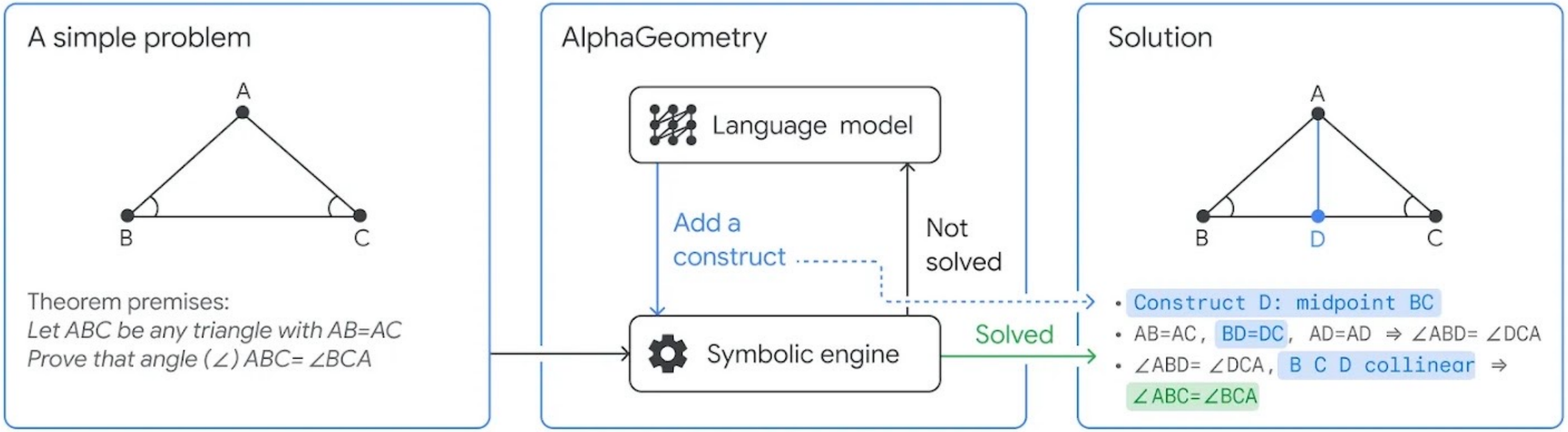
- 目标：给定一个复杂的目标分子（药物），如何用简单原料合成它?
- 符号空间：化学反应规则是离散的符号(例如：A物质＋B物质->C物质)，逆向搜索空间极其巨大
- Symbolic: 蒙特卡洛树搜索 (MCTS)，树的节点是分子，边是化学反应规则
- Neural: 训练一个深度神经网络，输入当前分子结构，预测哪种反应规则最有可能成功将其分解为更简单的片段（类似 AlphaGo 预测下一步棋）



[Planning Chemical Syntheses with Deep Neural Networks and Symbolic AI](). Nature 2019.

Neuro for Symbolic

- AlphaGeometry: An Olympiad-Level AI System for Geometry
- Neuro-Networks guide the symbolic engine by suggesting constructs



Solving olympiad geometry without human demonstrations. Nature 2024.

$\text{Neuro} \underset{\leftarrow}{\rightarrow} \text{Symbolic} \underset{\leftarrow}{\rightarrow}$

符号推理与神经网络联合优化?

- 神经网络依赖梯度优化，但是经典逻辑不可微，难以联合优化
- 实值逻辑(Real Logic)：一种将逻辑真值映射到连续实数域([0, 1])的数学框架

逻辑与(∧)

Gödel:

$A \wedge B = \min(A, B)$

Product:

$A \wedge B = A \cdot B$

Lukasiewicz:

$A \wedge B = \max(0, A + B - 1)$

逻辑或(∨)

Gödel:

$A \vee B = \max(A, B)$

Product:

$A \vee B = A + B - A \cdot B$

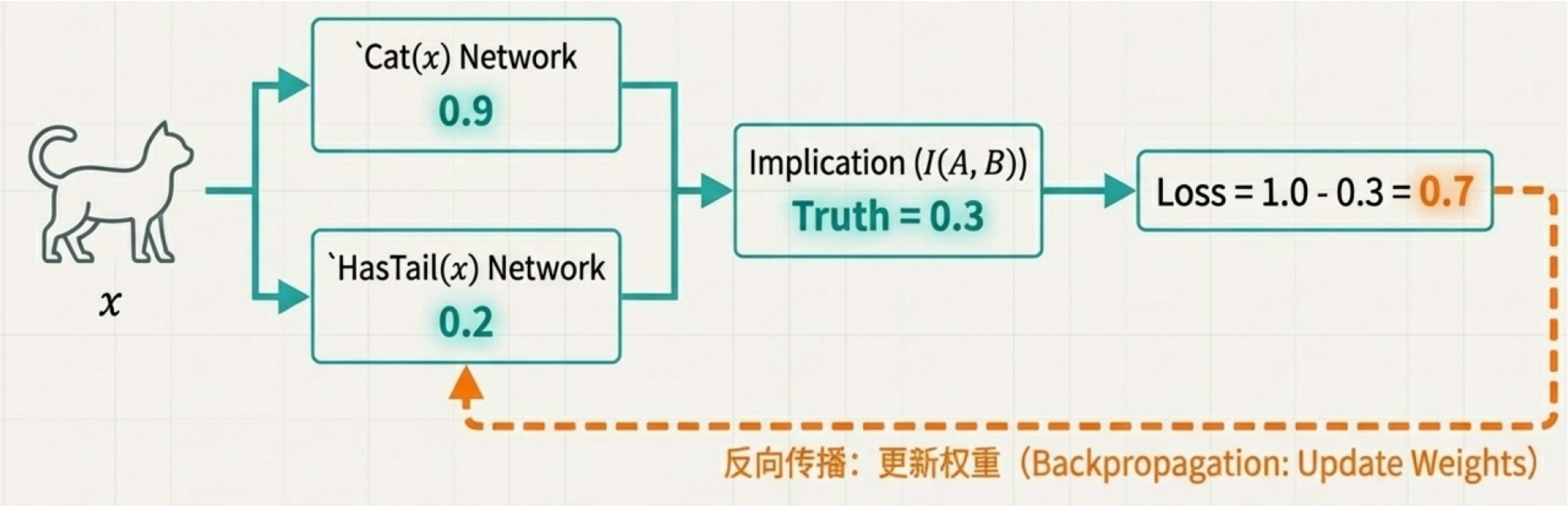
逻辑非(¬)

Standard:

$\neg A = 1 - A$

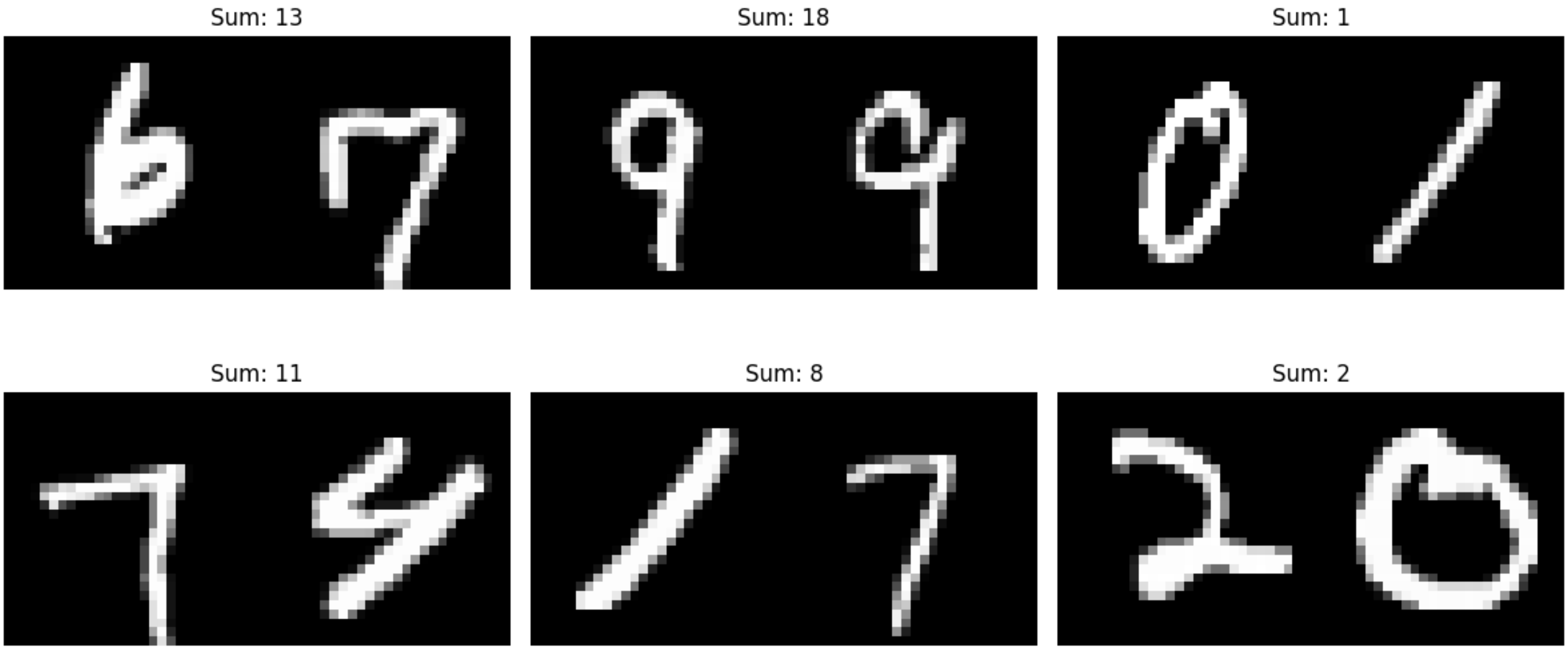
Neuro $\overset{\rightarrow}{\leftarrow}$ Symbolic $\overset{\longrightarrow}{\longleftarrow}$

∀x: Cat (x) → HasTail(x)



[Logic Tensor Networks](). Artificial Intelligence 2022.

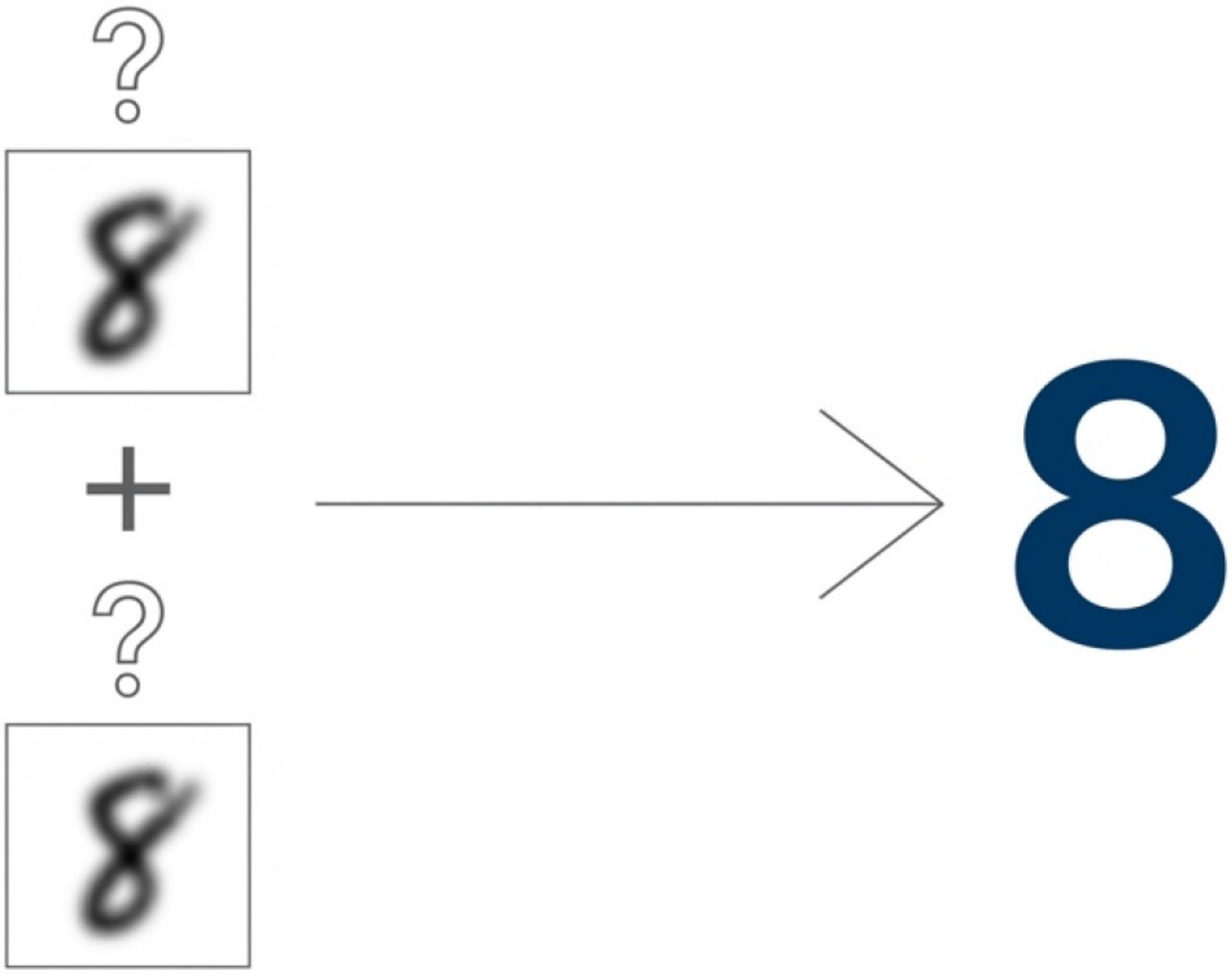
手写数字加法



手写数字加法

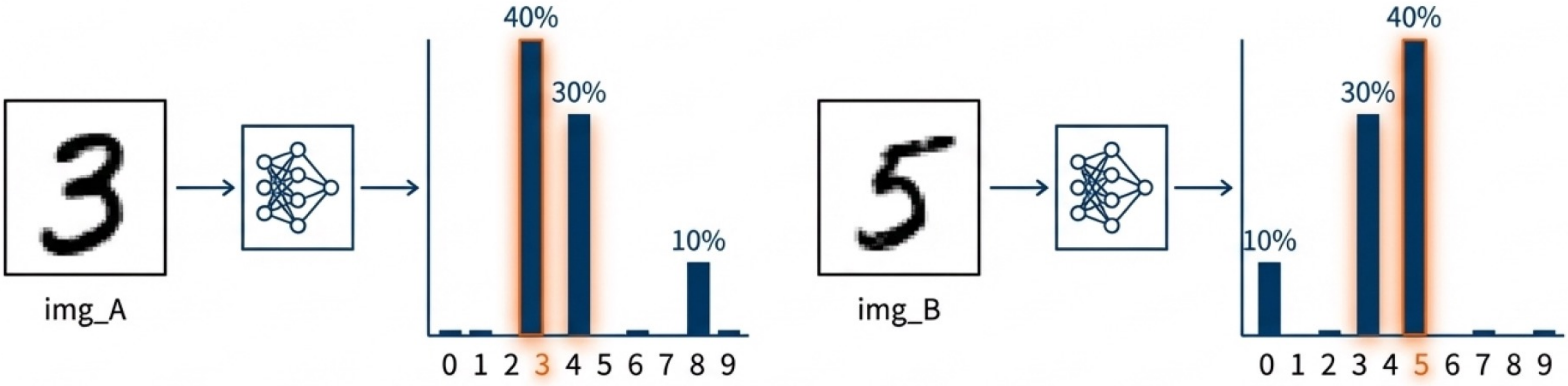
- **输入：** 两张手写数字图片
- **已知：** 它们的数字之和
- **缺失：** 每张图片各自独立的标签

我们能否在不看到单个数字标签的情况下，教会机器识别每个数字？



DeepProbLog

- 第一步：神经网络的初步猜测
- 神经网络接收图片，输出类别概率分布，由于未经训练，它的输出是模糊和不确定的

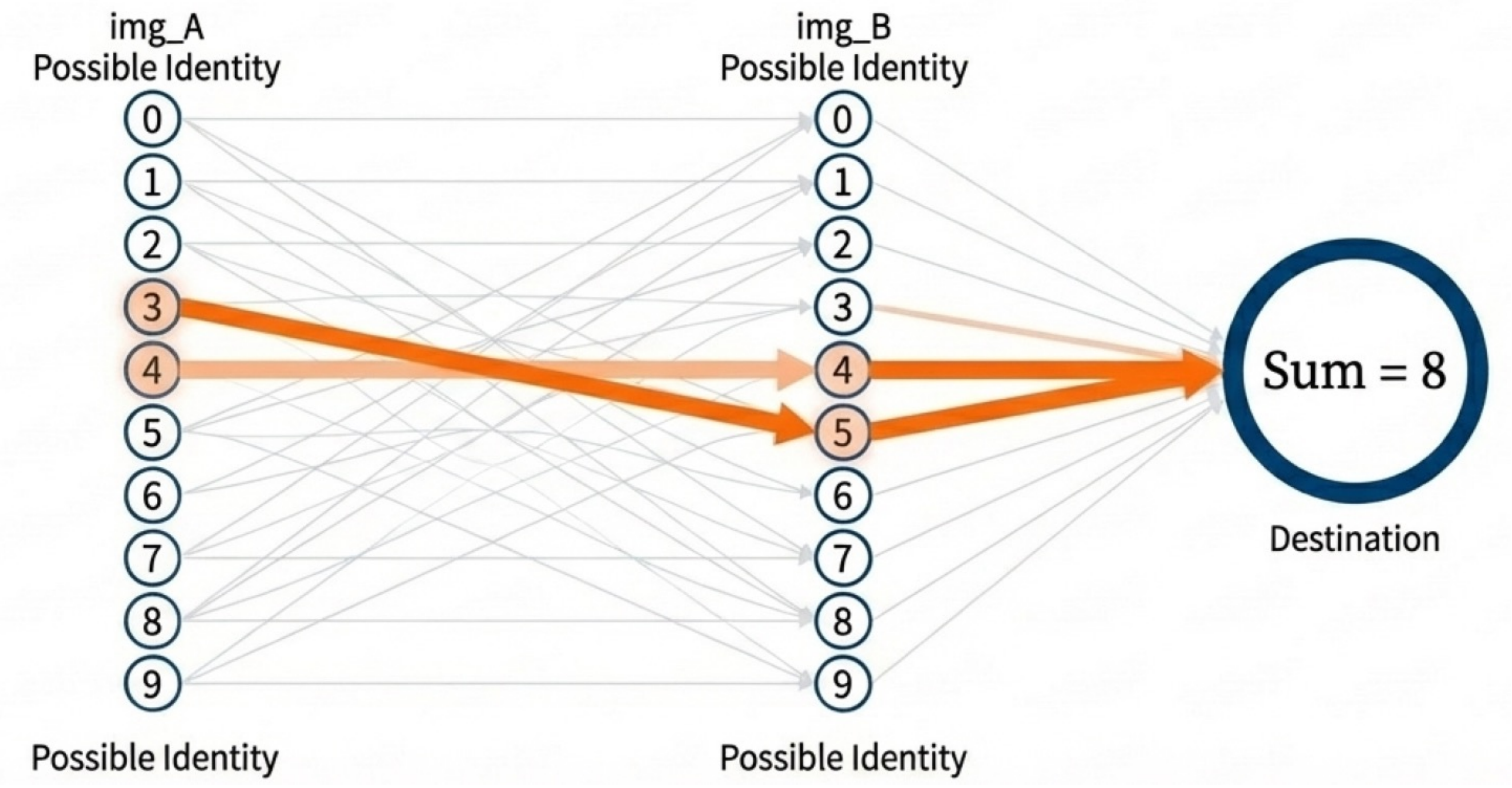


DeepProbLog: Neural Probabilistic Logic Programming. NeurIPS 2018.

DeepProbLog

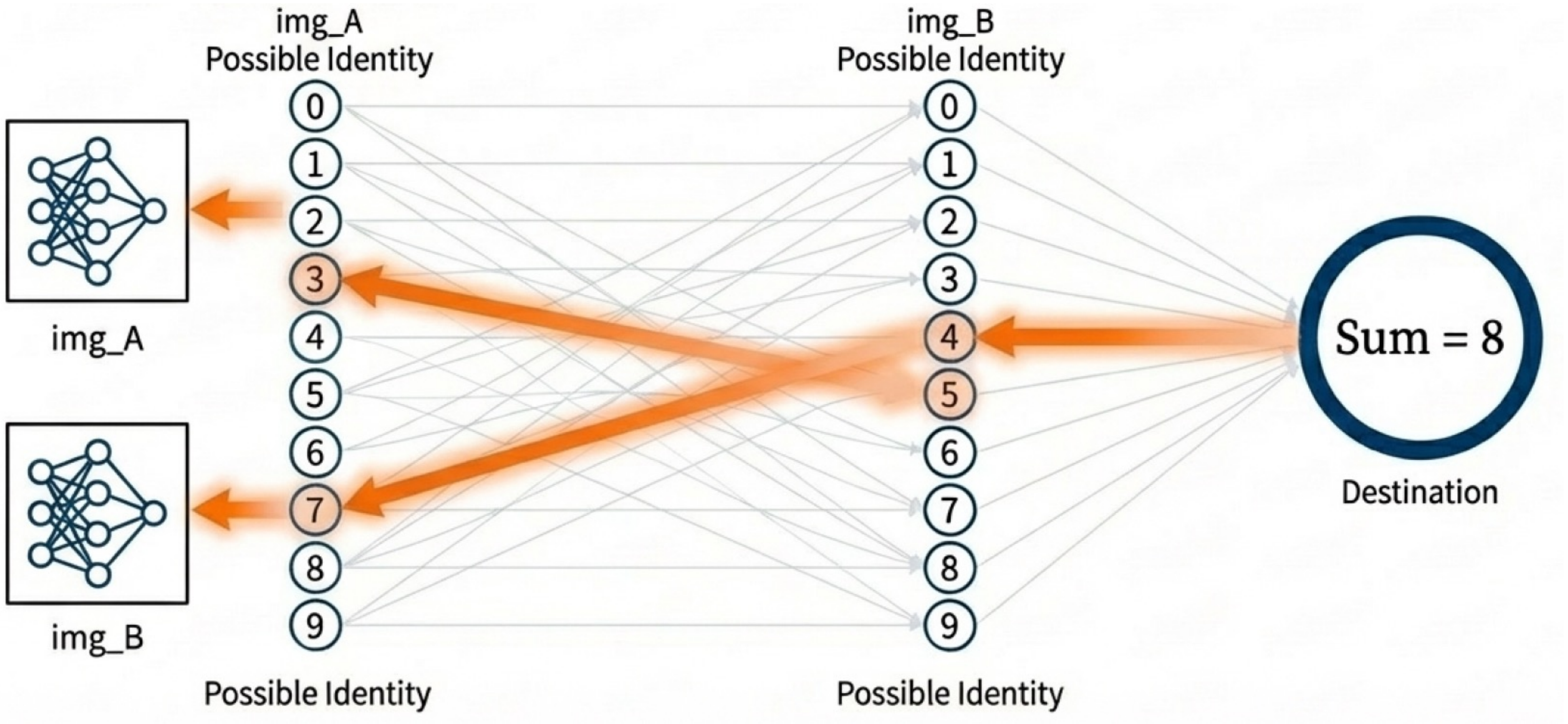
- **DeepProbLog引擎根据X+Y=8的规则，计算所有满足该规则的可能世界的概率**

$$P(sum{=}8) = P(d1{=}0, d2{=}8) + \cdots + P(d1{=}3, d2{=}5) + P(d1{=}4, d2{=}4) + \cdots$$



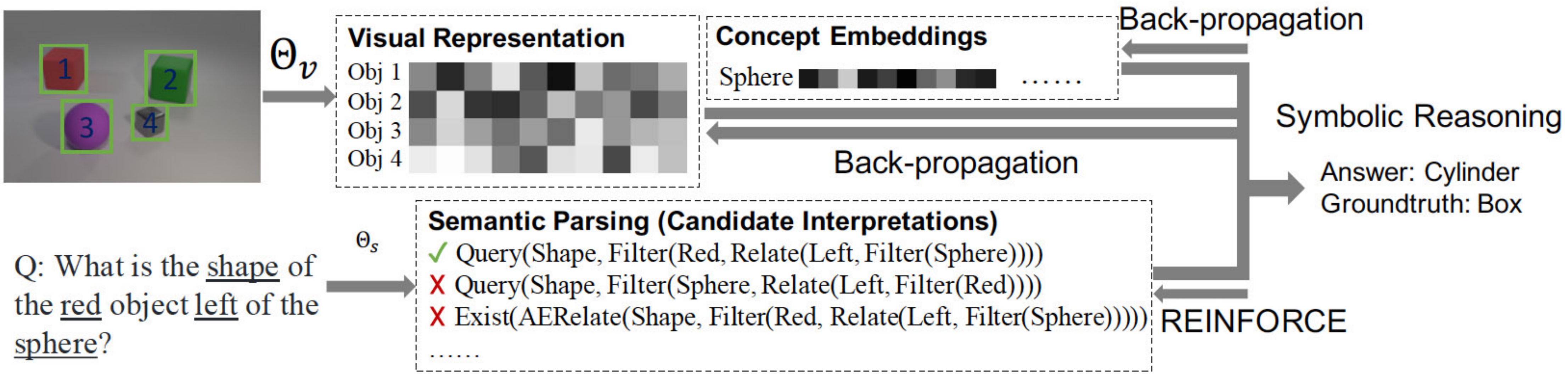
DeepProbLog

系统的目标是最大化P(sum=8)，根据损失函数反向更新权重



Neuro $\overset{\rightarrow}{\leftarrow}$ Symbolic $\overset{\rightarrow}{\leftarrow}$

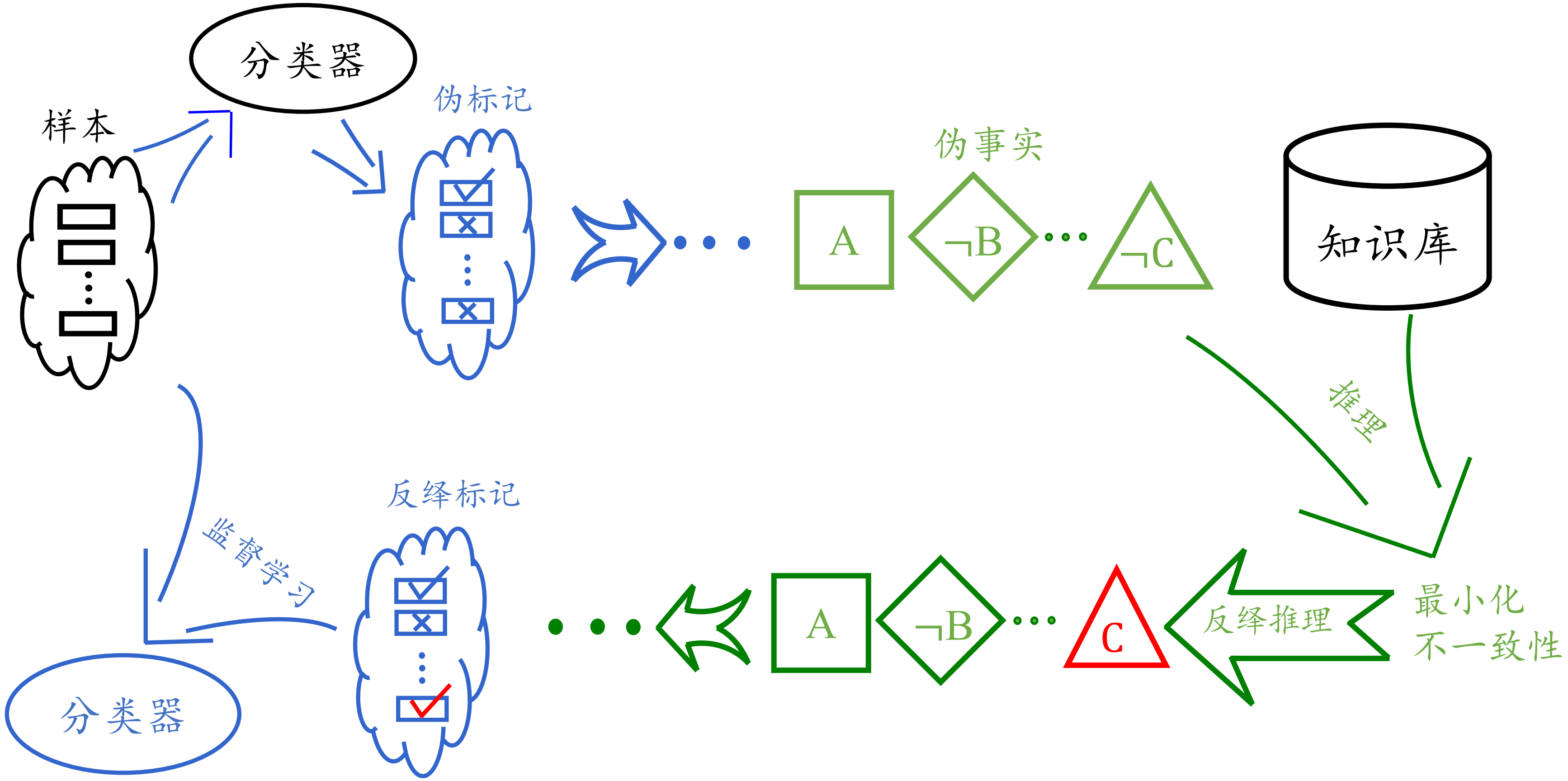
- Neuro-Symbolic Concept Learner (NS-CL)
- 神经网络负责感知，符号系统负责推理
- 通过反向传播端到端优化



The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision. ICLR 2019.

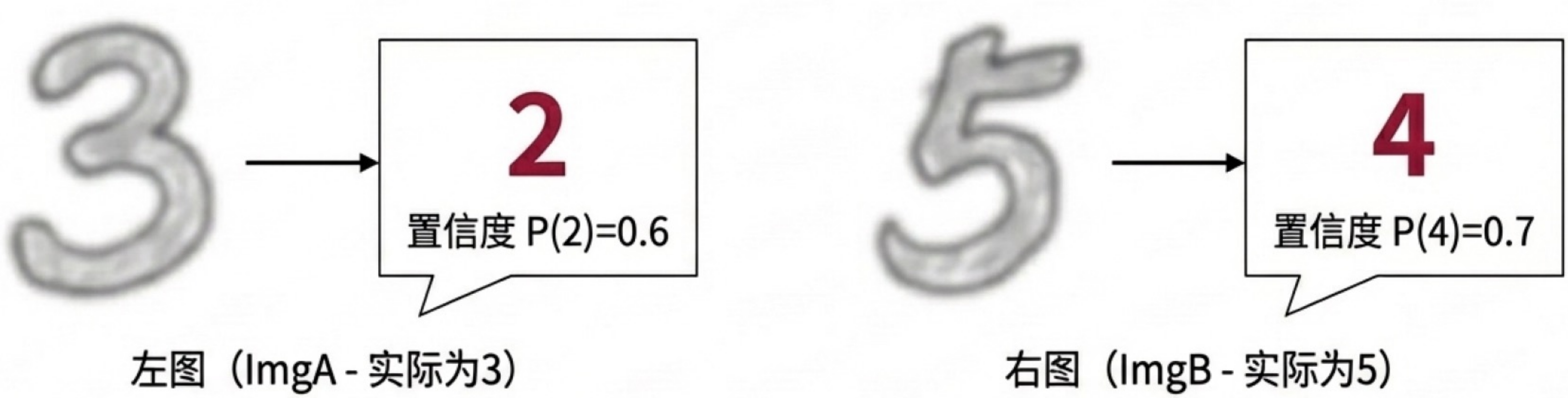
Neuro $\underset{\leftarrow}{\rightarrow}$ Symbolic $\underset{\leftarrow}{\rightarrow}$

- Abductive Learning (ABL)：无需逻辑可微



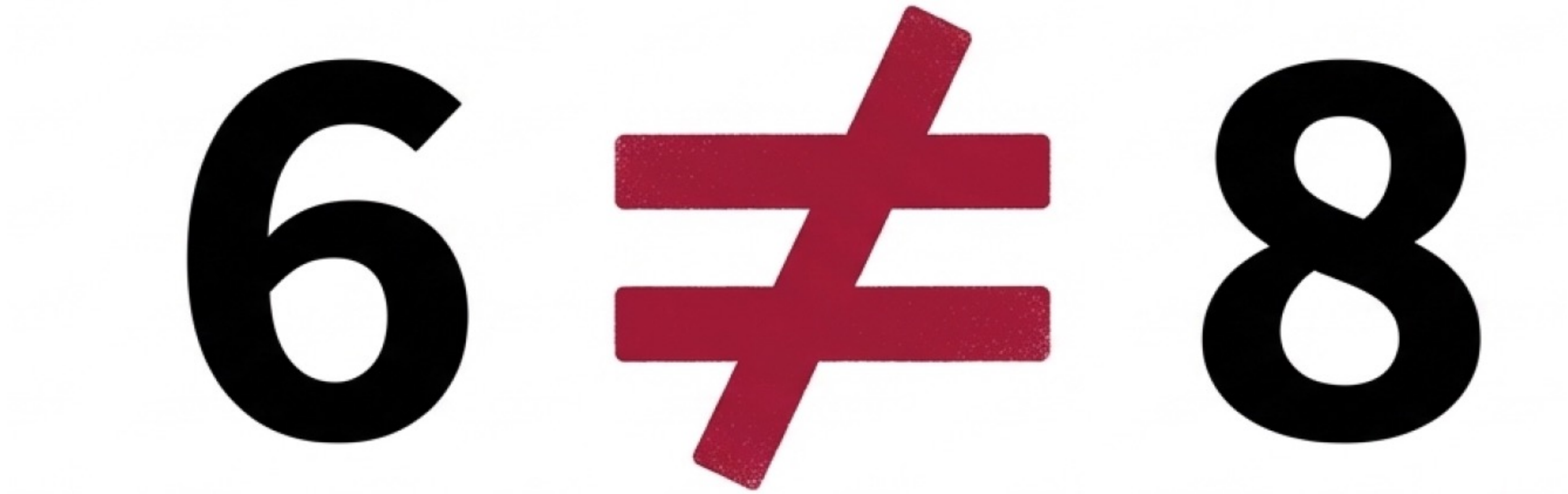
Abductive Learning (ABL)

步骤1：机器学习预测



Abductive Learning (ABL)

步骤2：知识库检查



计算：2+4=6

机器学习模型输出与
逻辑推理存在矛盾

Abductive Learning (ABL)

步骤3：反绎推理(Abduction)

已知最终结果必须为8，且加法规则是X+Y=Z

为了让X+Y成立，X,Y可能是什么？

生成一致性假设

H_1：(0, 8)

H_2：(1, 7)

H_3：(2, 6)

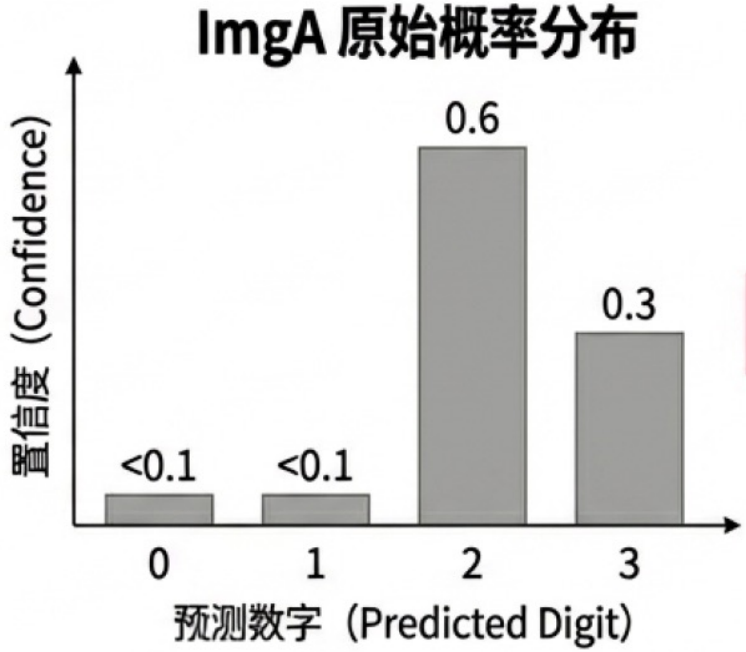
H_4：(3, 5)

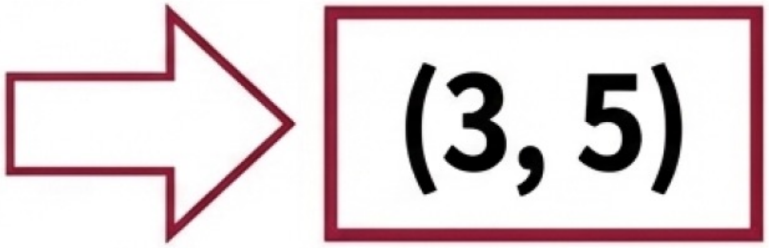
H_5：(4, 4)

Abductive Learning (ABL)

反绎修正

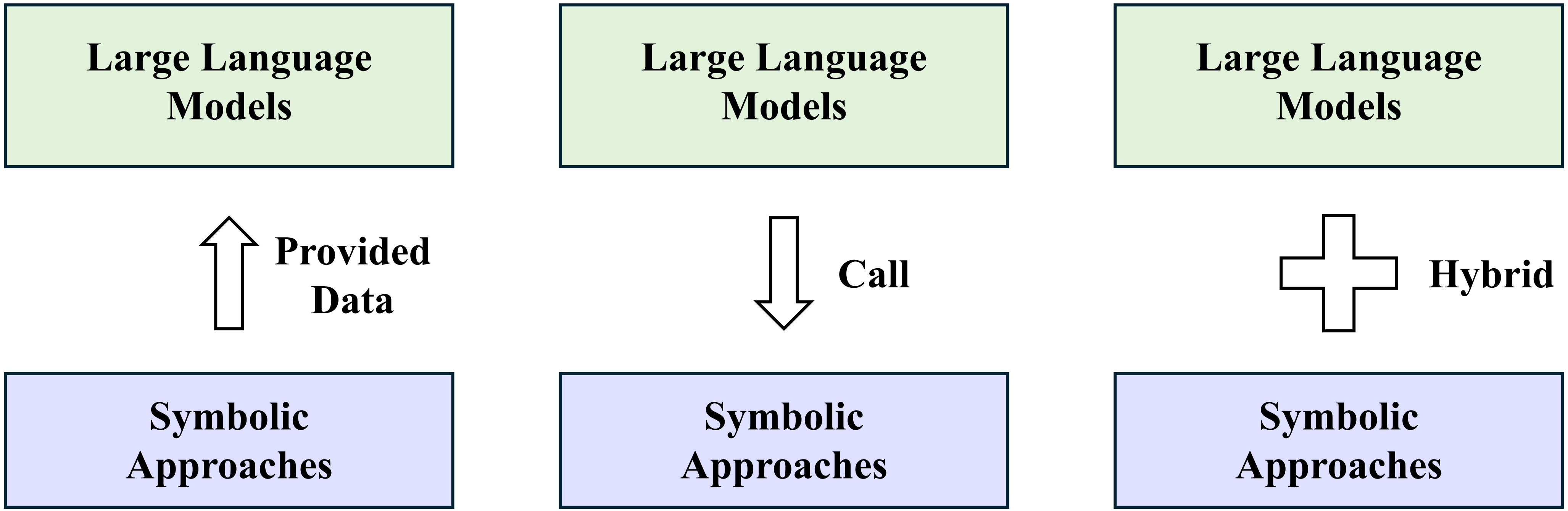
- H_1：(0, 8)
- H_2：(1, 7)
- H_3：(2, 6)
- **H_4：(3, 5)**
- H_5：(4, 4)





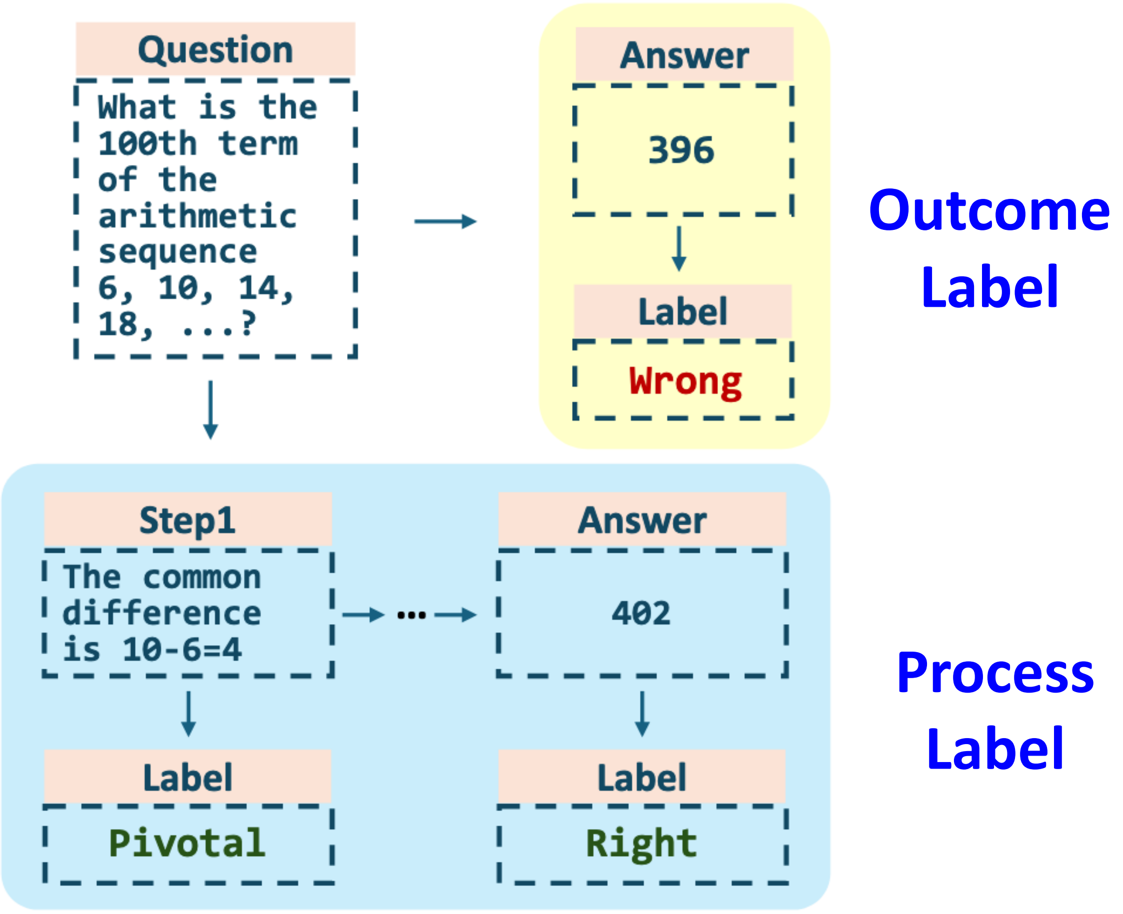
综合评估后，推理机选择（3, 5）作为最可靠的修正假设。

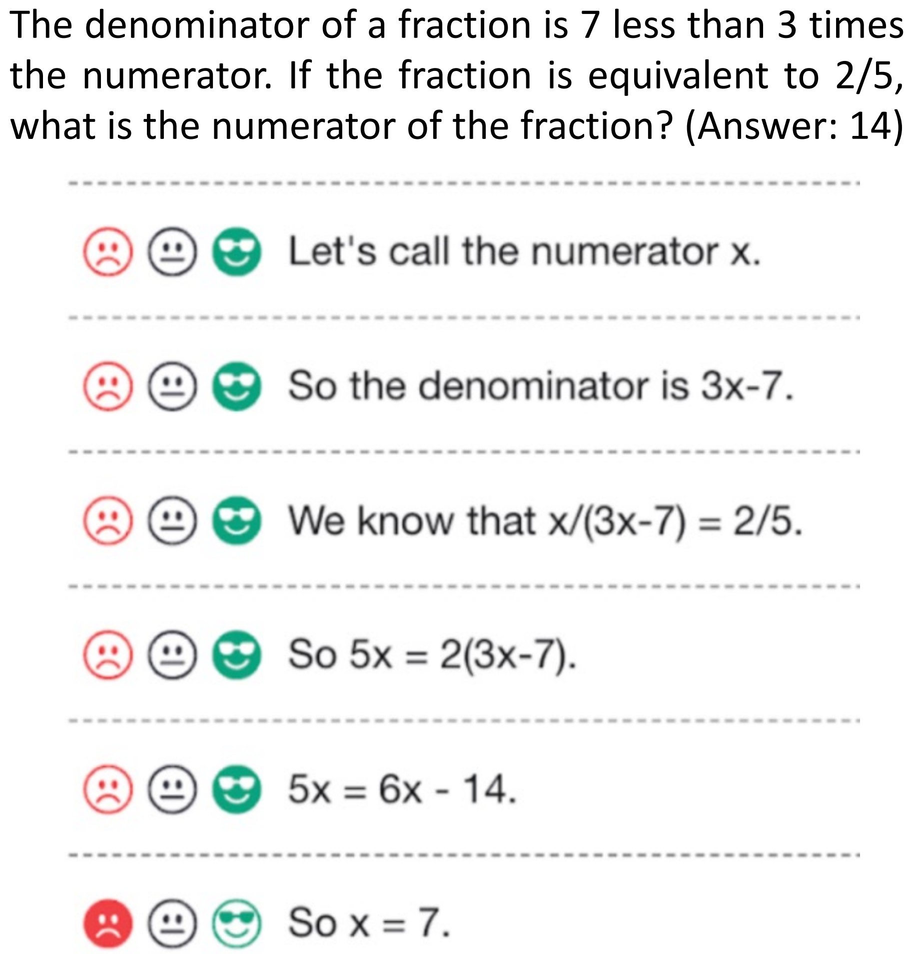
Neuro-Symbolic Methods with LLMs



Supervision for Reasoning & Planning Tasks

Data Collection, especially the label collection is very expensive. Different from classification tasks, reasoning requires not only the **outcome label** but also the **reasoning process label**

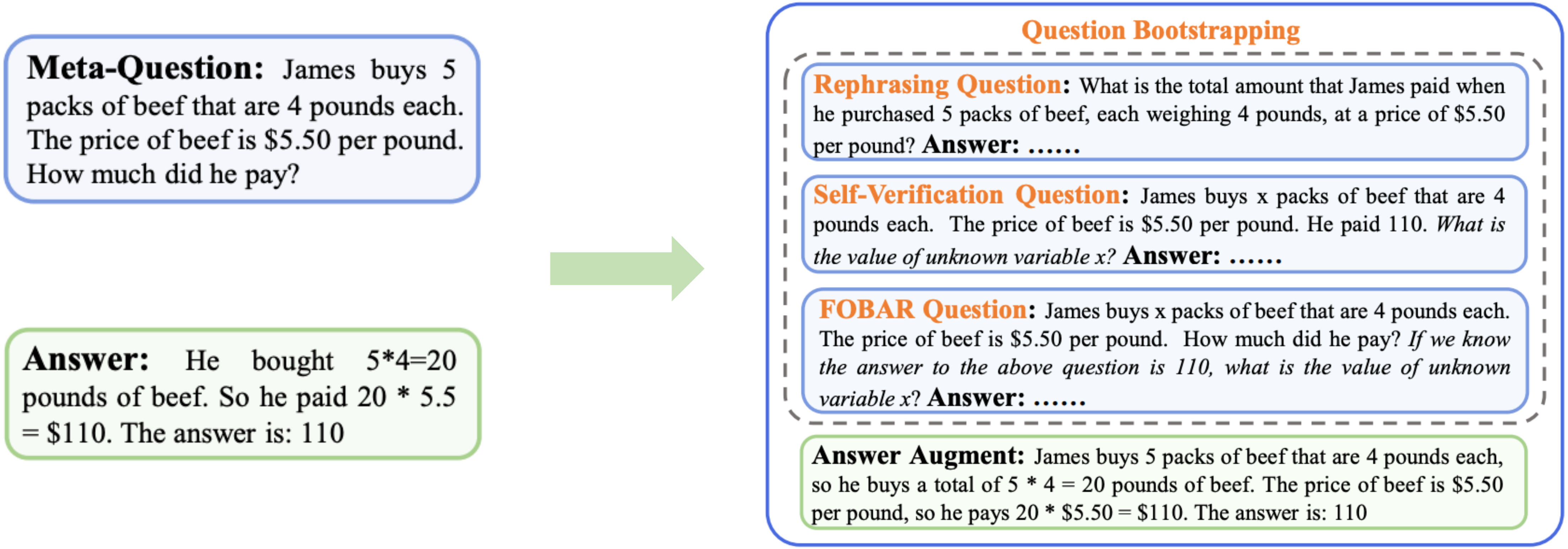




Automatic Data Generation

Many heuristic methods can be developed to generate large-scale datasets from seed examples

- Rephrasing, Template-based Augmentation, Numeric Perturbation, LLM Generation, etc



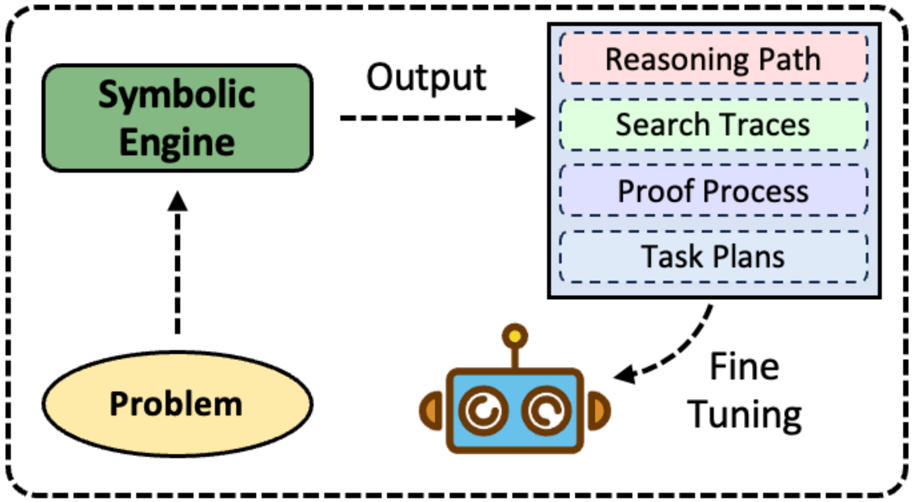
However, these data generation methods often fail to ensure data diversity and accuracy

Symbolic -> LLM

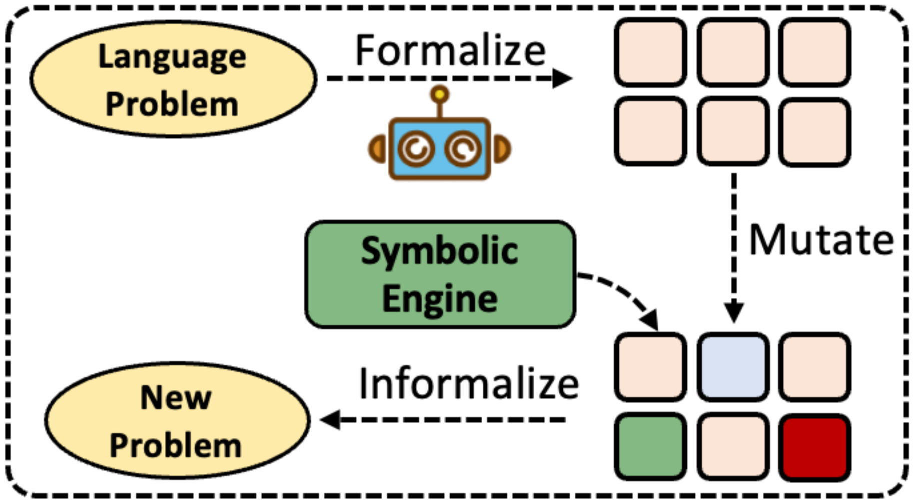
☐ Basic Ideas：

- Symbolic Generation, LLM Imitation： Utilize **symbolic engines** (e.g., logic solvers, search algorithms) to **generate reasoning paths**, and **fine-tune LLMs** to imitate the capabilities of symbolic methods
- LLM Formalize, Symbolic Augment： Employ LLM to transform **informal natural language data** into a **formal symbolic language**, and perform **data augmentation within the formal space**

Symbolic Generation, LLM Imitation

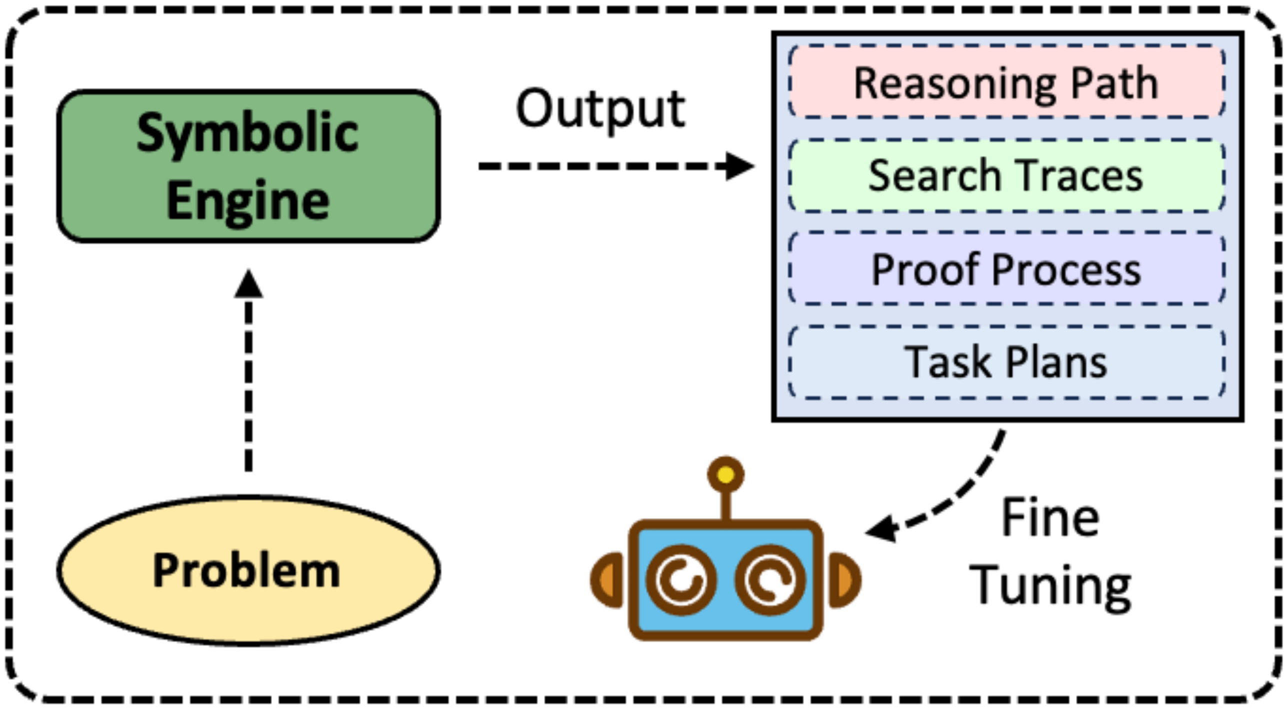


LLM Formalize, Symbolic Augment



Symbolic Generation, LLM Imitation

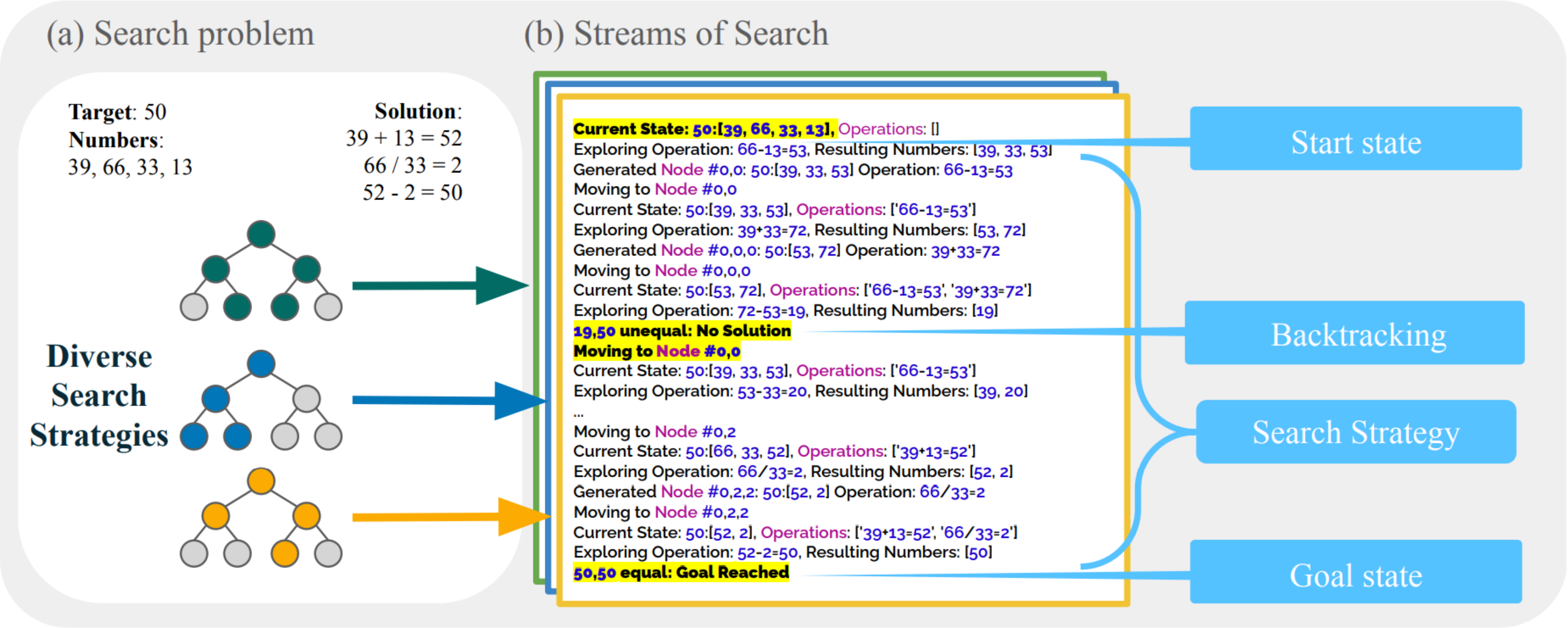
Symbolic Generation, LLM Imitation



Symbolic Generation, LLM Imitation

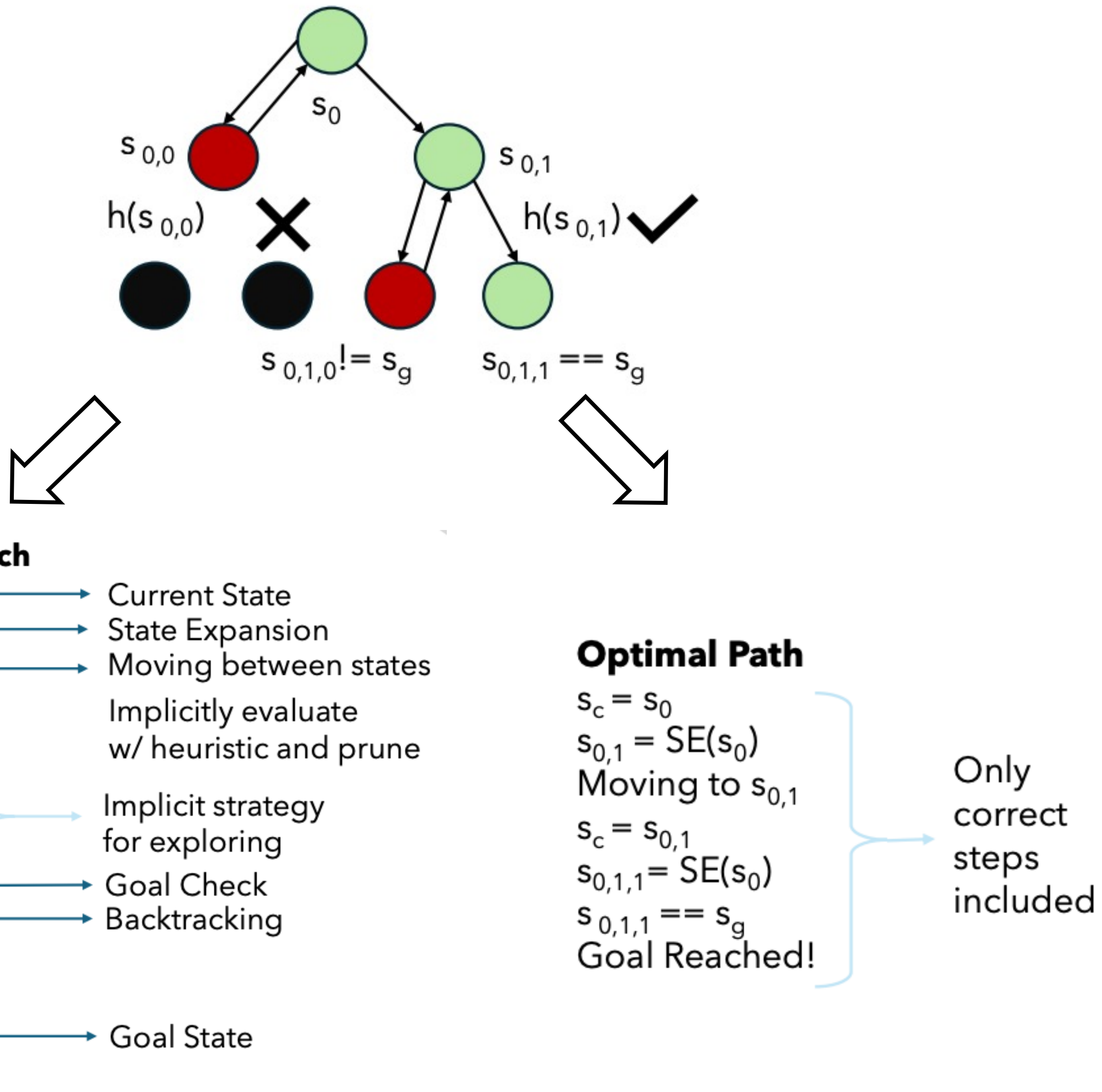
SOS: Use search algorithms to generate solution paths for **Countdown** problems,

And then fine-tune the LLMs on this data so that it acquires backtracking ability



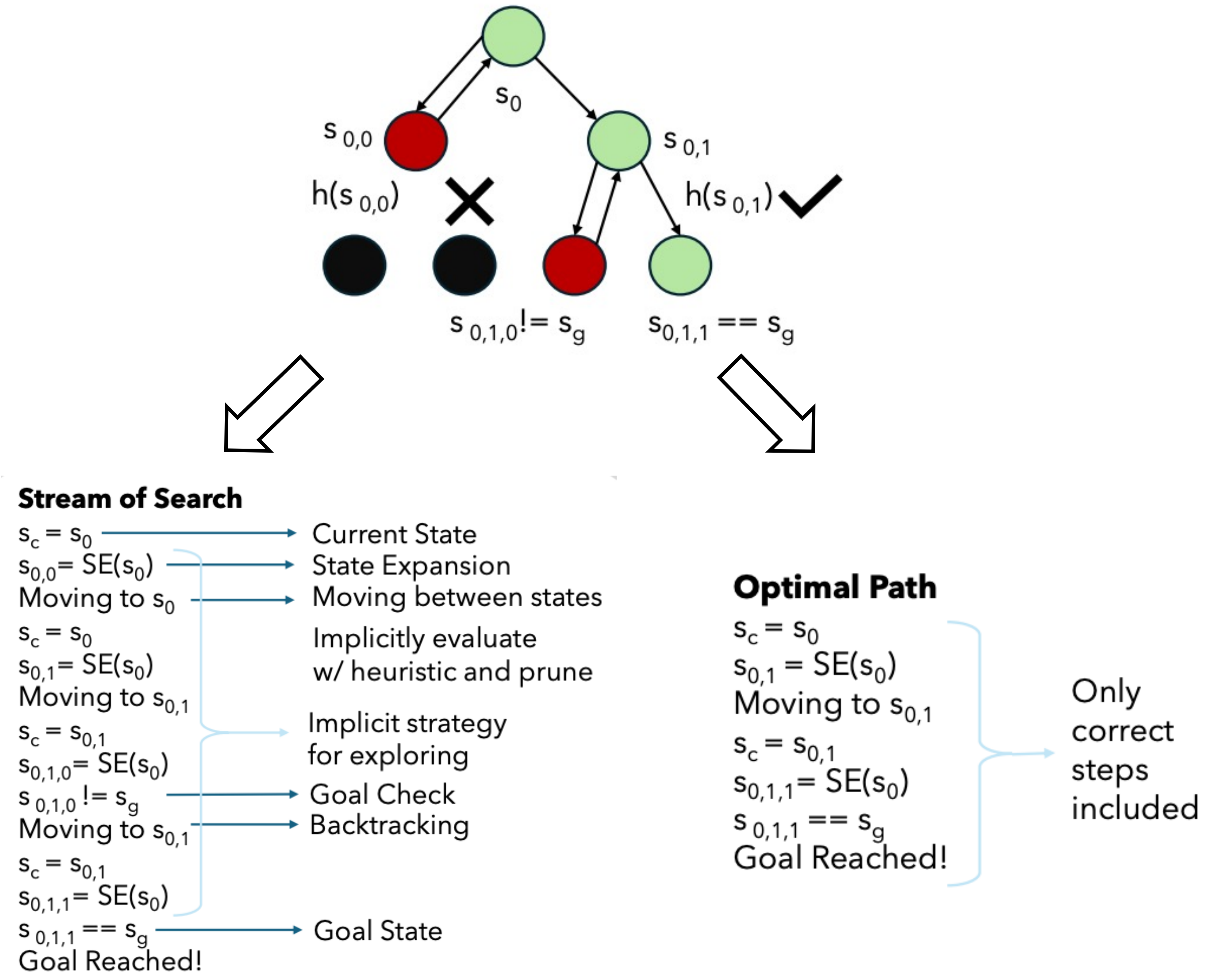
[1] Stream of Search (SoS): Learning to Search in Language. COLM 2024.

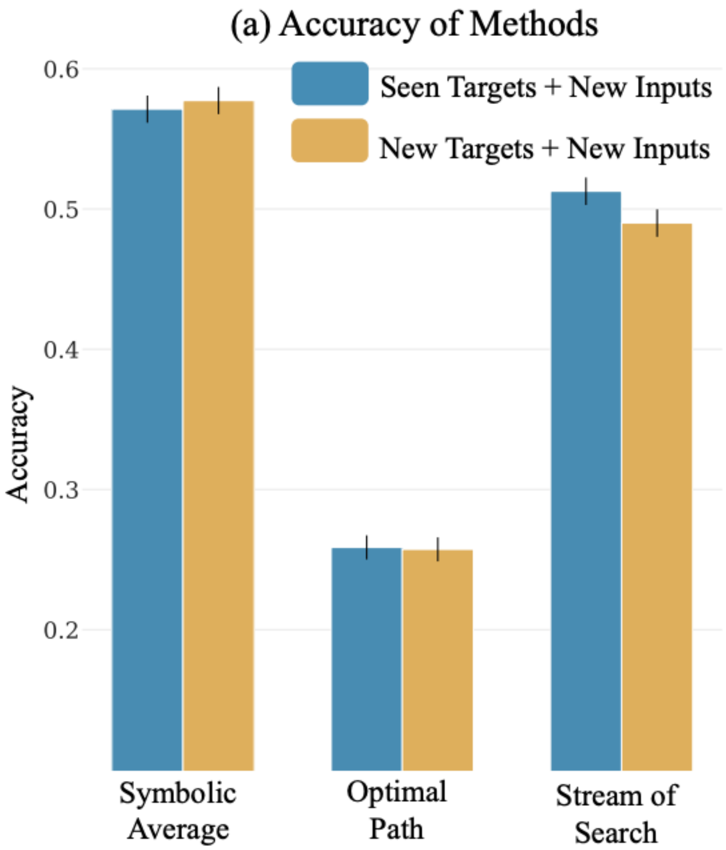
Symbolic Generation, LLM Imitation



[1] Stream of Search (SoS): Learning to Search in Language. COLM 2024.

Symbolic Generation, LLM Imitation





LLMs fine-tuned on

search process data

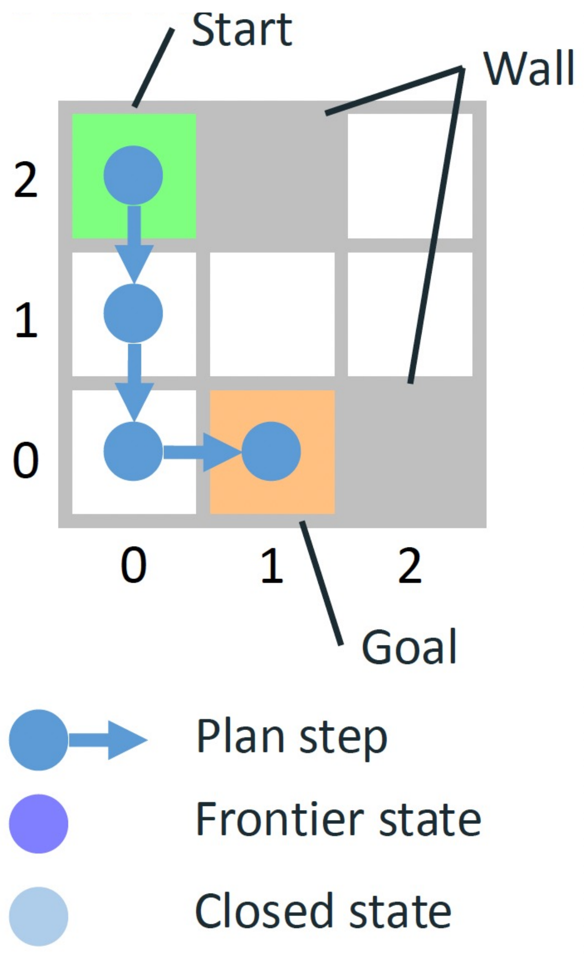
outperform those trained on

optimal path data

[1] Stream of Search (SoS): Learning to Search in Language. COLM 2024.

Symbolic Generation, LLM Imitation

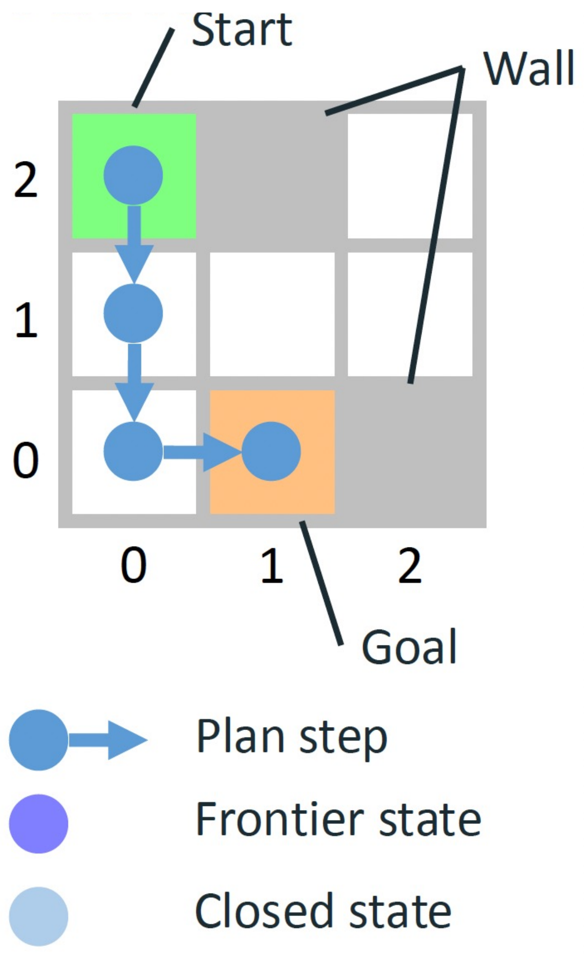
SearchFormer: Train transformer with A* generated traces for maze navigation tasks



[1] Beyond A∗ : Better Planning with Transformers via Search Dynamics Bootstrapping. COLM 2024.

Symbolic Generation, LLM Imitation

SearchFormer: Train transformer with A* generated traces for maze navigation tasks



<prompt>

```
bos
start 0 2
goal  1 0
wall  1 2
wall  2 0
eos
```

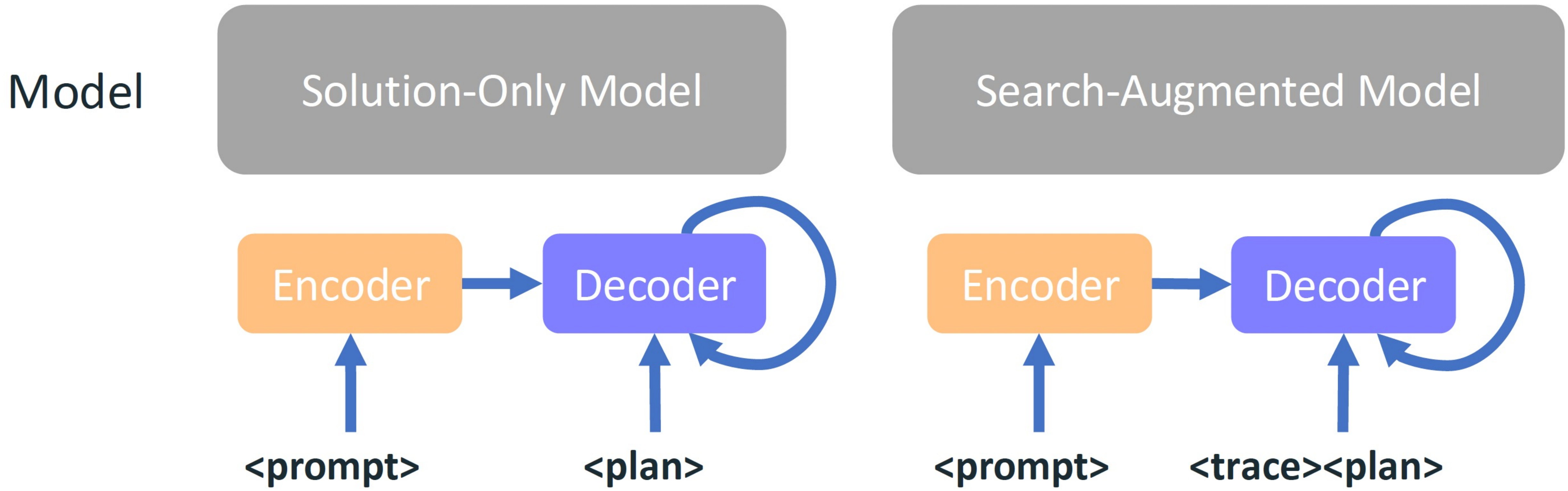
<trace><plan>

```
bos
create 0 2 c0 c3
close  0 2 c0 c3
create 0 1 c1 c2
close  0 1 c1 c2
create 0 0 c2 c1
create 1 1 c2 c1
close  0 0 c2 c1
create 1 0 c3 c0
close  1 0 c3 c0
plan   0 2
plan   0 1
plan   0 0
plan   1 0
eos
```

[1] Beyond A∗ : Better Planning with Transformers via Search Dynamics Bootstrapping. COLM 2024.

Symbolic Generation, LLM Imitation

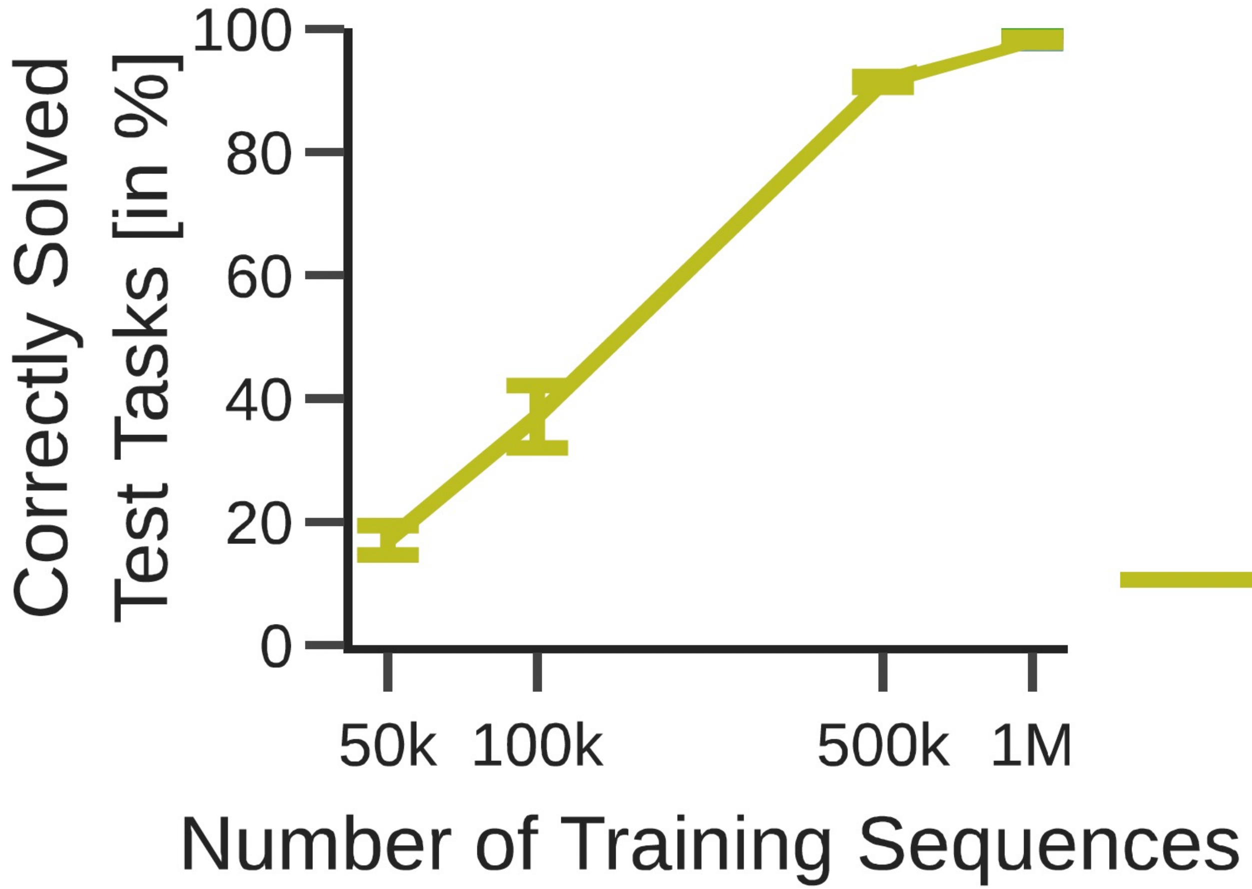
Train a Transformer to predict the next token via teacher forcing



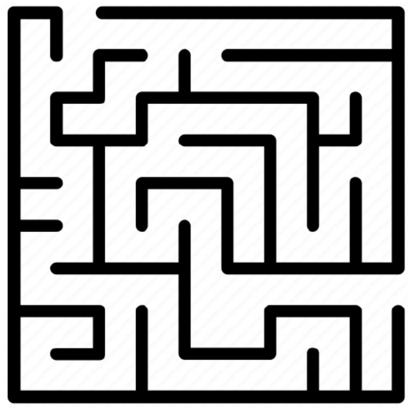
[1] Beyond A∗ : Better Planning with Transformers via Search Dynamics Bootstrapping. COLM 2024.

Symbolic Generation, LLM Imitation

Search-Augmented vs. Solution-Only Models



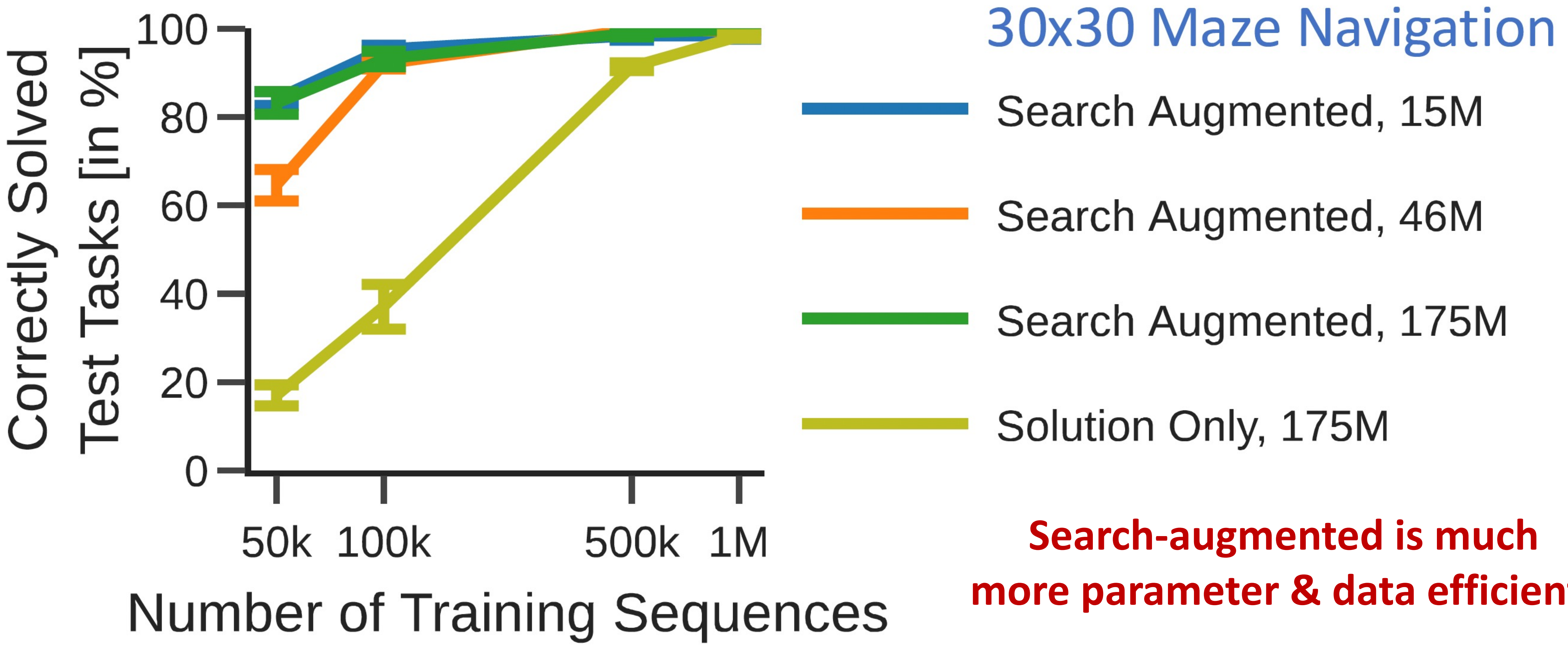
30x30 Maze Navigation



[1] Beyond A∗ : Better Planning with Transformers via Search Dynamics Bootstrapping. COLM 2024.

Symbolic Generation, LLM Imitation

Search-Augmented vs. Solution-Only Models

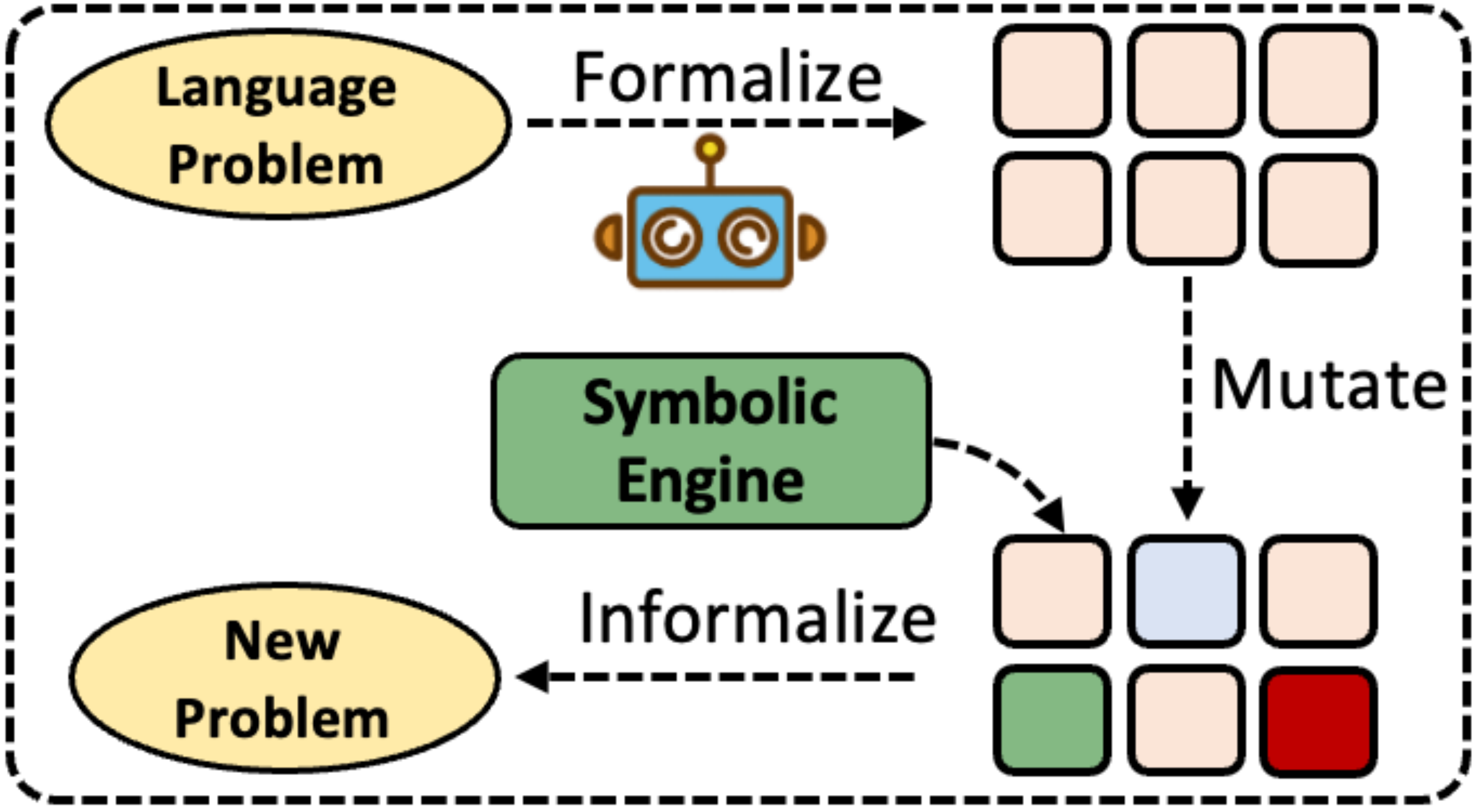


Search-augmented is much more parameter & data efficient!

[1] Beyond A∗ : Better Planning with Transformers via Search Dynamics Bootstrapping. COLM 2024.

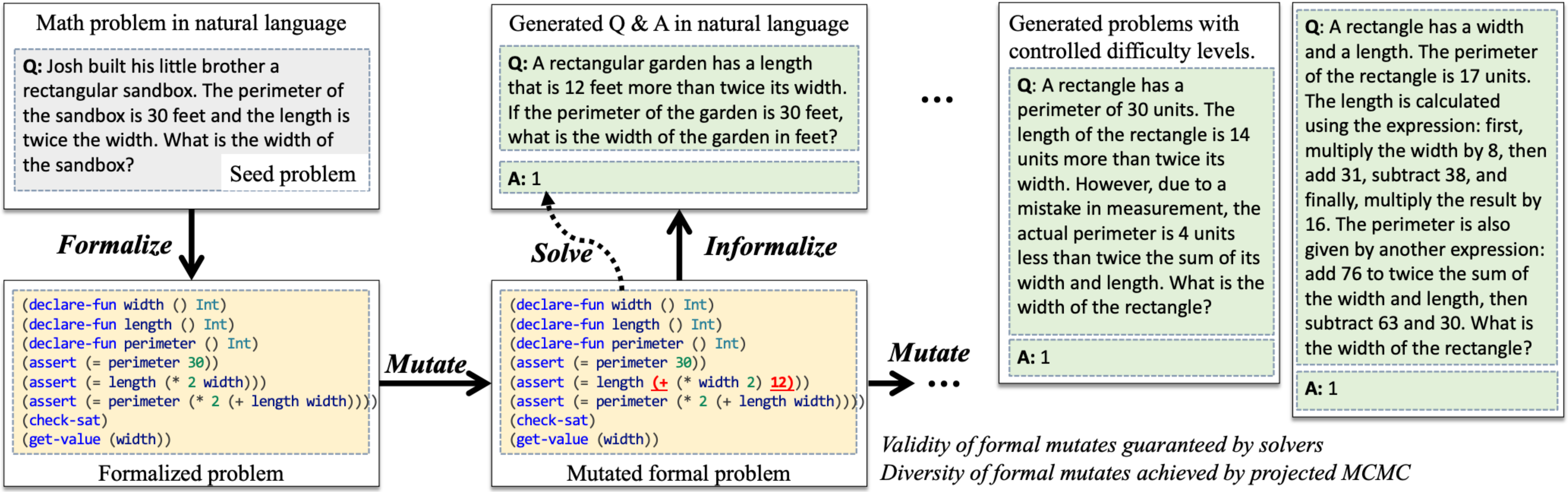
LLM Formalize, Symbolic Augment

LLM Formalize, Symbolic Augment



LLM Formalize, Symbolic Augment

NSDG： Convert math problems into **SMT-LIB language**, perform **data augmentation** within the formal space, and finally use an LLM to translate the formal language back into natural language



[1] **Neuro-Symbolic Data Generation for Math Reasoning**. NeurIPS 2024.

Neuro-Symbolic Data Generation for math reasoning

Supervised fine-tuning of the base model on the constructed dataset achieves consistent performance improvements

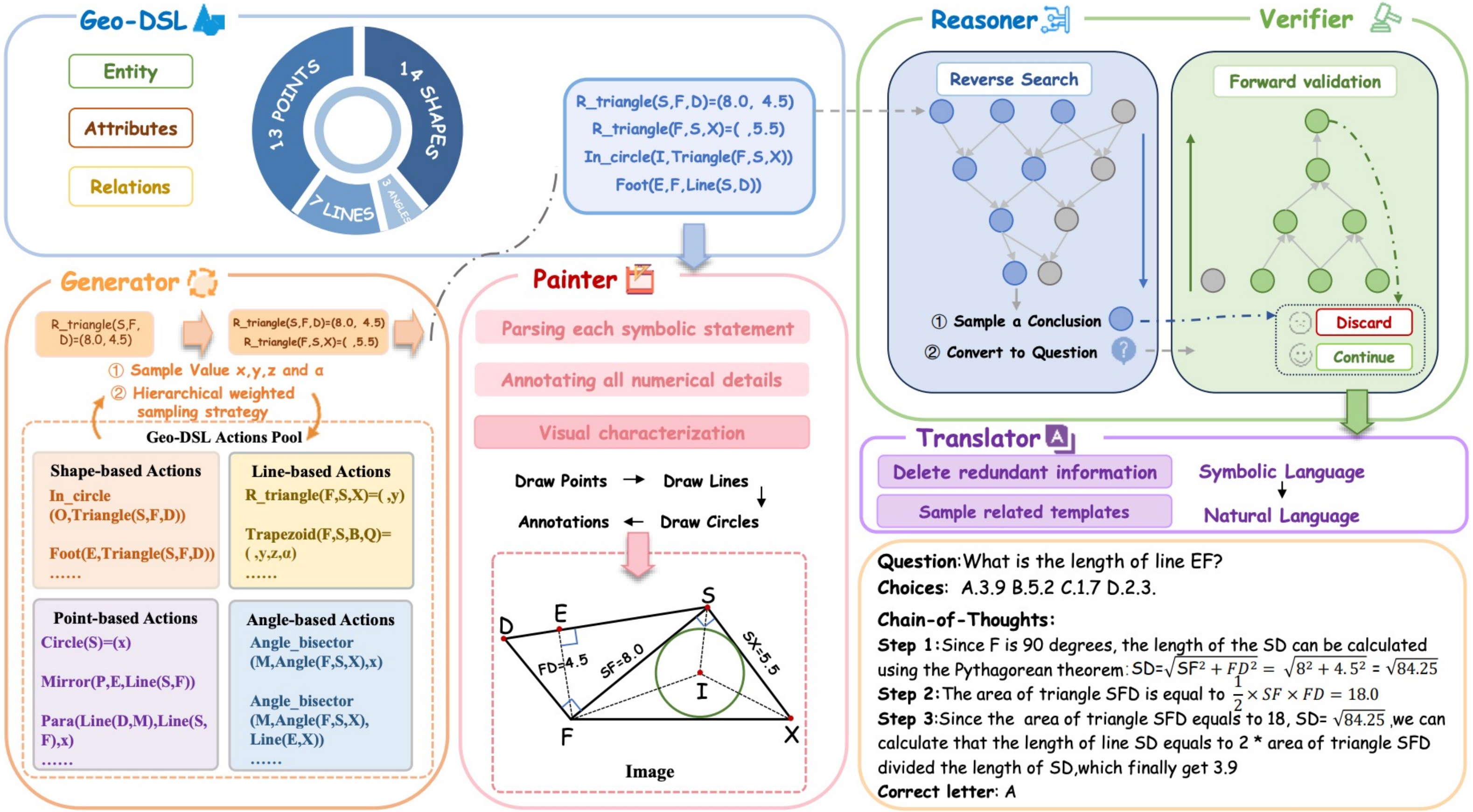
Model	#Dataset	LLaMA-2 7B Base		LLaMA-2 13B Base		Mistral 7B Base	
		GSM8K	MATH	GSM8K	MATH	GSM8K	MATH
WizardMath	>240K	54.9	10.7	63.9	14.0	83.2	33.0
MuggleMATH	157K	68.4	8.4	74.0	9.4	-	-
MAmmoTH†	260K	50.5	10.4	56.3	12.9	61.9	17.5
MetaMath	395K	66.5	19.8	72.3	22.4	77.7	28.2
Ours	860K	**79.0**	**30.4**	**84.1**	**33.7**	**86.8**	**37.3**
Δ		↑ 10.6	↑ 10.6	↑ 10.1	↑ 11.3	↑ 3.6	↑ 4.3

† Model performance is re-evaluated using Pass@1 of CoT prompt.

[1] **Neuro-Symbolic Data Generation for Math Reasoning**. NeurIPS 2024.

LLM Formalize, Symbolic Augment

NeSyGeo: A **DSL for plane geometry** was defined, and data augmentation was performed based on this DSL representation to generate a large number of high-quality geometric figures



[1] NeSyGeo: A Neuro-Symbolic Framework for Multimodal Geometric Reasoning Data Generation. AI4Math Workshop @ ICML 2025.

LLM Formalize, Symbolic Augment

Reinforcement Fine-tuning

Model	GeoQA	MathVision Angle	MathVision Area	MathVision Length	MathVerse Angle	MathVerse Area	MathVerse Length	MathVerse Plane Geometry
Qwen2.5-VL-3B	53.3	26.3	26.3	21.1	31.3	20.9	37.0	32.5
Qwen2.5-VL-3B+NeSyGeo	**55.7** (+2.4)	**26.3** (+0.0)	**42.1** (+15.8)	**26.3** (+5.2)	**32.6** (+1.3)	**23.5** (+2.6)	**37.2** (+0.2)	**35.5** (+3.0)
InternVL2.5-4B	61.9	36.8	31.6	26.3	31.5	22.7	31.9	30.7
InternVL2.5-4B+MAVIS	63.5 (+1.6)	31.6 (-5.2)	26.3 (-5.3)	31.6 (+5.3)	37.1 (+5.6)	20.9 (-1.8)	35.3 (+3.4)	33.7 (+3.0)
InternVL2.5-4B+R-CoT	63.3 (+1.4)	31.6 (-5.2)	31.6 (+0.0)	21.1 (-5.2)	31.2 (-0.3)	18.3 (-4.4)	34.3 (+2.4)	28.7 (-2.0)
InternVL2.5-4B+NeSyGeo	**69.2** (+7.3)	**42.1** (+5.3)	**36.8** (+5.2)	**26.3** (+0.0)	**39.9** (+8.4)	**24.9** (+2.2)	**36.1** (+4.2)	**36.7** (+6.0)
InternVL2.5-8B	66.2	36.8	36.8	21.1	36.9	23.1	34.8	36.6

Supervised Fine-Tuning

Model	GeoQA	Vision Intensive Angle	Vision Intensive Area	Vision Intensive Length	Vision Intensive Plane Geometry
Qwen2.5-VL-7B	69.4	43.0	27.5	46.2	44.1
Qwen2.5-VL-7B+NeSyGeo	**71.8** (+2.4)	**46.1** (+3.1)	**23.1** (-4.4)	**49.5** (+3.3)	**46.7** (+2.6)
LLaVA-Next-7B	22.6	28.5	6.6	16.5	20.4
LLaVA-Next-7B+NeSyGeo	**26.1** (+3.5)	**30.6** (+2.1)	**7.7** (+1.1)	**19.2** (+2.7)	**22.9** (+2.5)

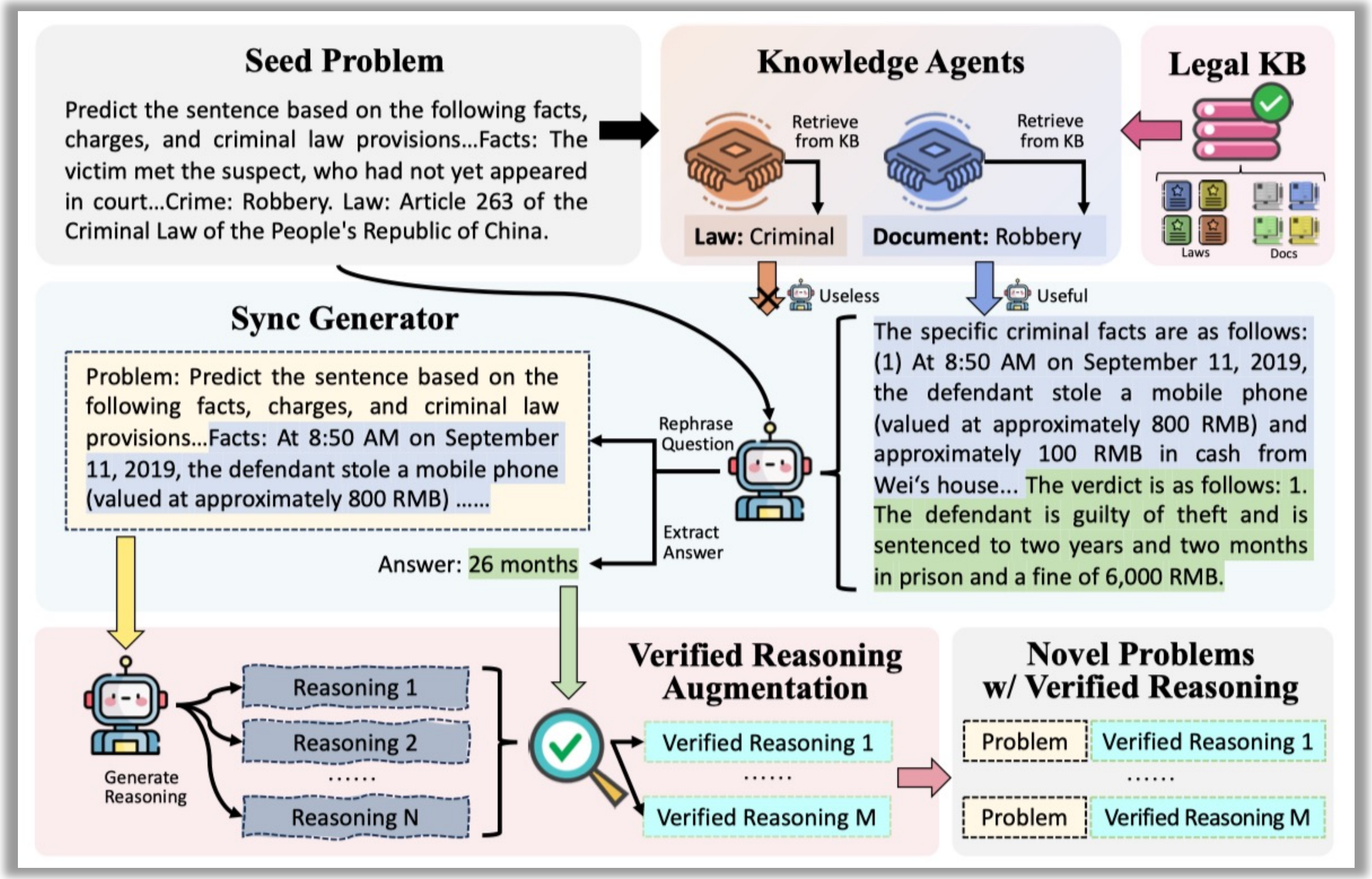
[1] NeSyGeo: A Neuro-Symbolic Framework for Multimodal Geometric Reasoning Data Generation. AI4Math Workshop @ ICML 2025.

Application: Chinese Legal Reasoning

LawGPT: Knowledge-Enhanced Legal Reasoning Data Augmentation

For legal reasoning, no suitable symbolic language representation

- LLM generates text data
- Verifies the synthetic data via abductive reasoning based on a legal knowledge base
- Improve the quality of data iteratively



[1] **LawGPT: A Chinese Legal Knowledge-Enhanced Large Language Model**. ICLR 2025 Workshop @ SCI-FM.

Application: Chinese Legal Reasoning

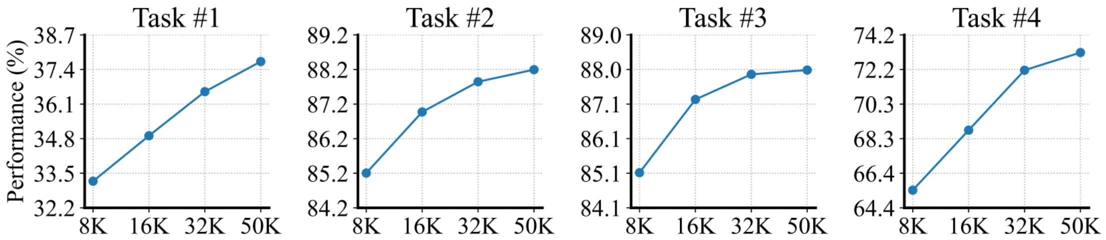
Compared to general and legal-specific LLMs, performance is improved across multiple legal reasoning tasks

Models	#Parameters	Task #1	Task #2	Task #3	Task #4	Average
General LLMs						
Deepseek V3	671B	38.1	87.5	86.8	84.4	74.2
GPT-4	-	27.5	82.6	81.9	77.6	67.4
GPT-3.5 Turbo	-	31.3	78.7	76.8	61.2	62.0
Legal-Specific LLMs						
Lexilaw	7B	35.8	78.1	74.9	35.8	56.1
HanFei	7B	33.6	73.1	69.6	39.4	53.9
FuziMingcha	7B	22.2	77.2	75.5	47.2	55.5
WisdomInterrogatory	7B	32.0	80.4	81.1	17.4	52.7
LaywerLLaMA	13B	25.9	74.2	75.5	39.2	53.7
ChatLaw	13B	31.6	76.2	73.6	41.4	55.7
ChatLaw	33B	26.0	67.0	53.6	41.6	47.1
LawGPT	0.5B	33.1	86.8	86.6	62.0	67.1
LawGPT	1.5B	35.7	87.4	87.3	68.0	69.6
LawGPT	3B	**37.7**	**88.2**	**88.0**	**73.2**	**71.8**

Scalability of the data generation:

More data, higher performance

Models	#Parameters	Task #1	Task #2	Task #3	Task #4	Average
Qwen-2.5	0.5B	27.9	81.2	80.1	45.0	58.6
LawGPT	0.5B	33.1	86.8	86.6	62.0	67.1
Δ Performance		↑ 5.2	↑ 5.6	↑ 6.5	↑ 14.0	↑ 9.5
Qwen-2.5	1.5B	29.9	82.4	82.3	49.0	60.9
LawGPT	1.5B	35.7	87.4	87.3	68.0	69.6
Δ Performance		↑ 5.8	↑ 5.0	↑ 5.0	↑ 19.0	↑ 8.7
Qwen-2.5	3.0B	28.7	81.7	79.9	56.0	61.6
LawGPT	3.0B	37.7	88.2	88.0	73.2	71.8
Δ Performance		↑ 9.0	↑ 6.5	↑ 8.1	↑ 17.2	↑ 10.2



[1] Zhi Zhou, Jiang-Xin Shi, Peng-Xiao Song, Xiao-Wen Yang, Yi-Xuan Jin, Lan-Zhe Guo, Yu-Feng Li. **LawGPT: A Chinese Legal Knowledge-Enhanced Large Language Model**. ICLR 2025 Workshop @ SCI-FM.

Summary & Open Problems

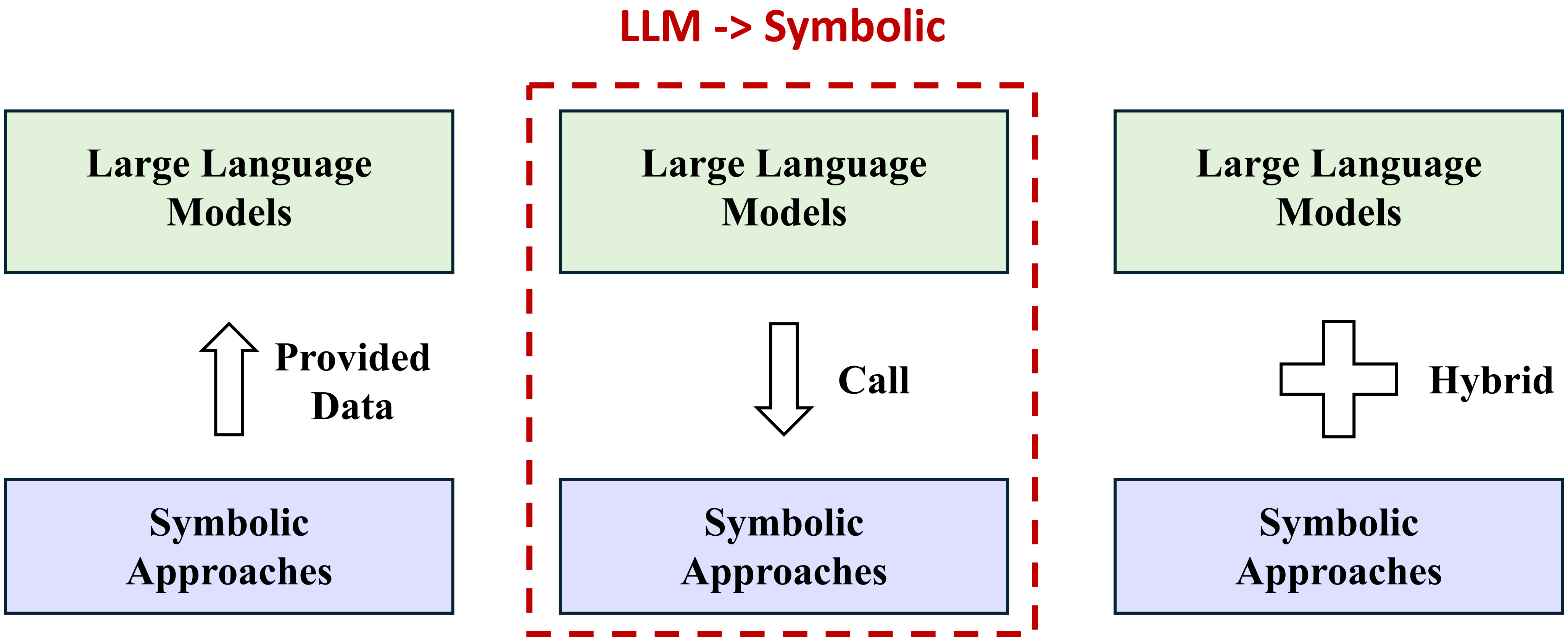
☐ Key Idea:

- Generate reasoning path with symbolic methods directly
- Augment data in the symbolic language represented space

☐ Open Problems:

- How to evaluate the quality of reasoning data?
- How to select the optimal subset of the training data?
- Can symbolic methods help data evaluation and selection?
- ……

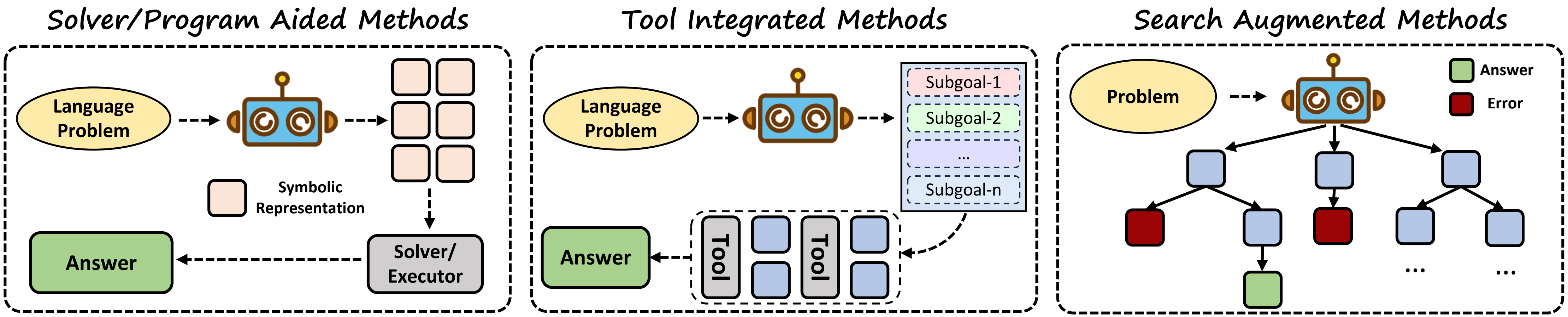
Option Two: Neuro-Symbolic Methods



LLM -> Symbolic

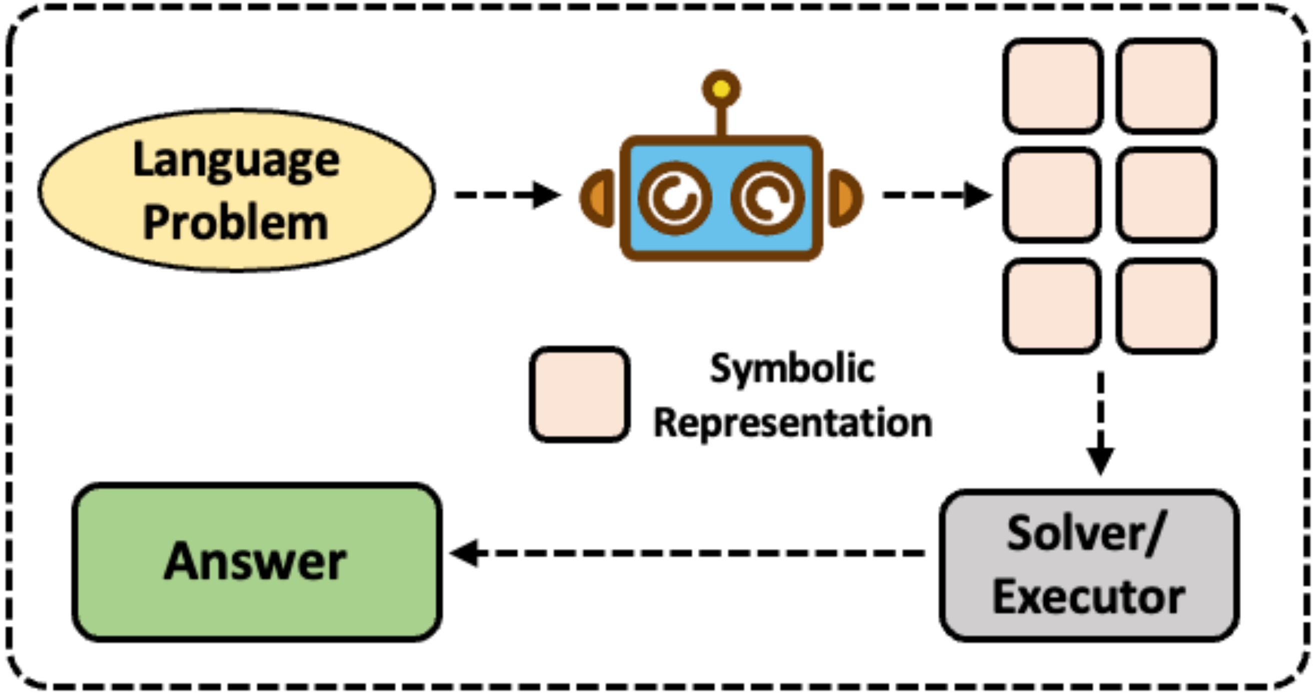
Basic Ideas:

- Solver/Program Aided Methods: LLM **converts Natural Language into Formal Language**, then call a symbolic solver
- Tool Integrated Methods: LLM **decompose original problem into multiple steps**, where intermediate steps can call external tools for solving, such as symbolic solvers, calculators, APIs, etc.
- Search Augmented Methods: LLM's decoding process is a search problem and can be optimized via different search strategies, e.g., MCTS, A*, etc.



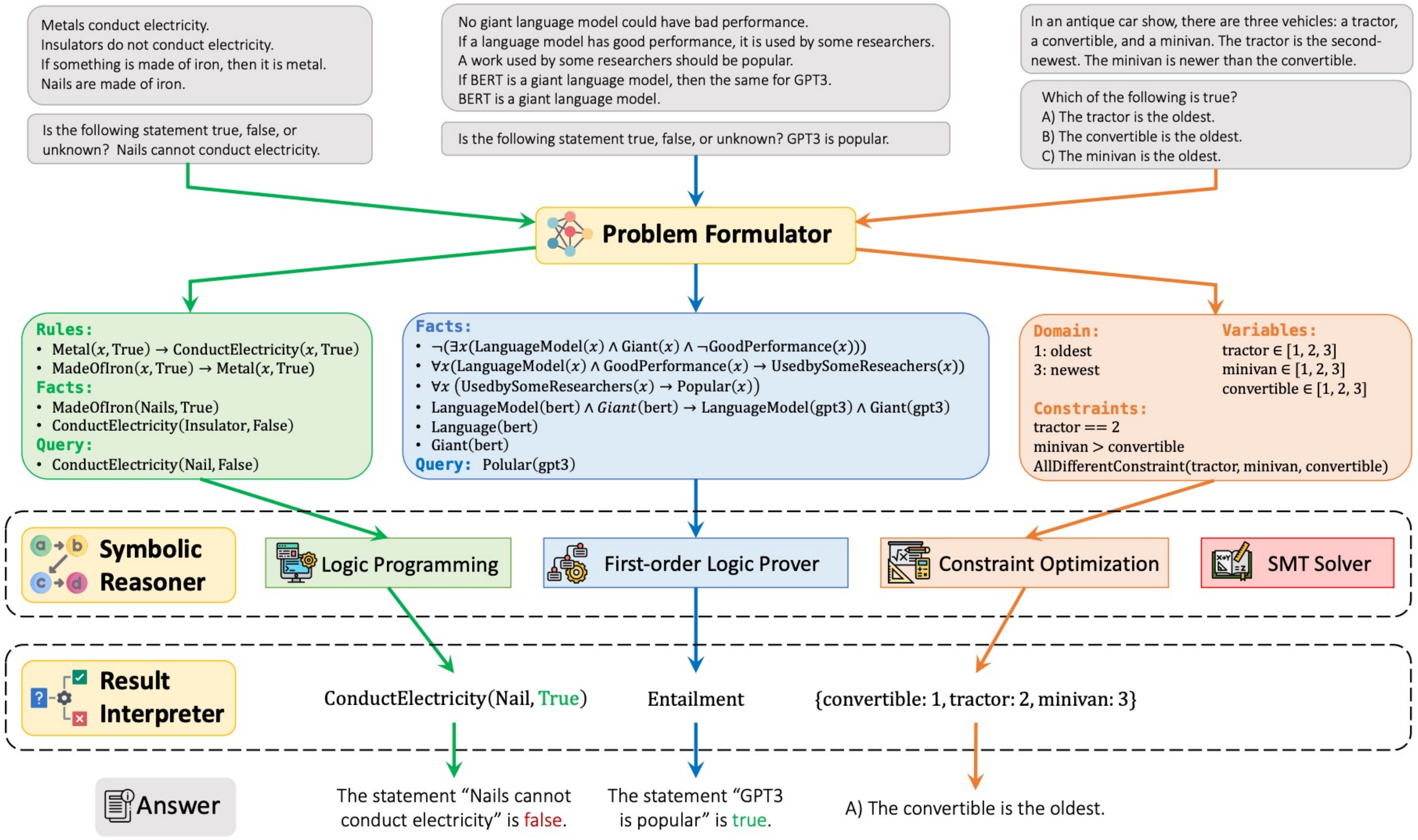
Solver/Program Aided Methods

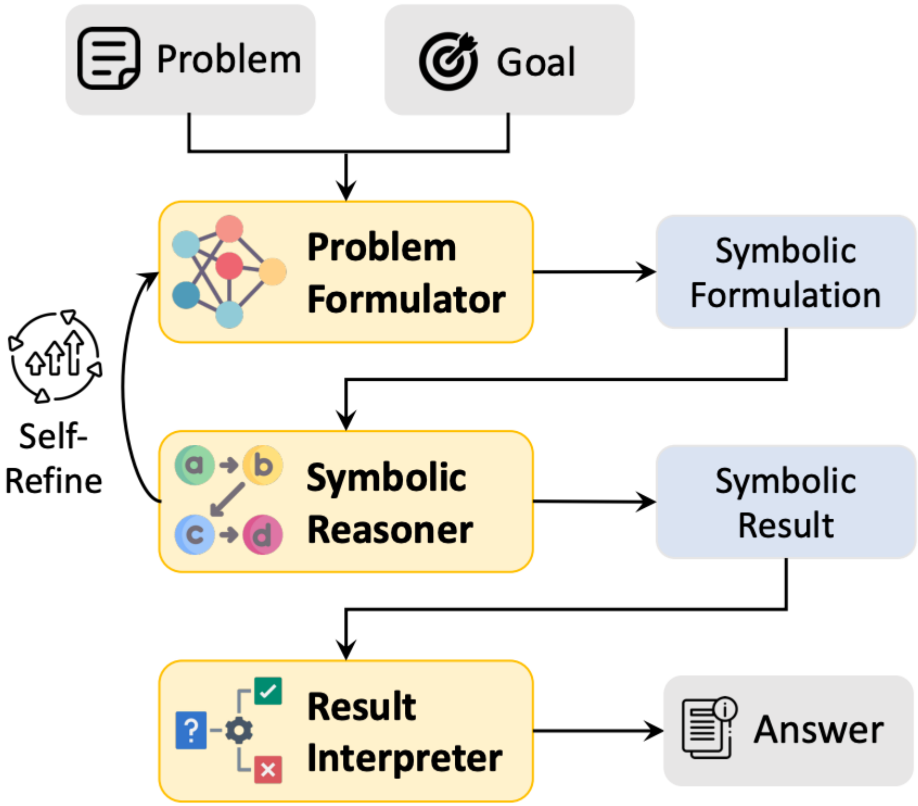
Solver/Program Aided Methods



Symbolic Solver Aided Methods

Convert natural Language into formal Language, then call a symbolic reasoner, such as **First-order Logic Prover**, **SMT solver**, **Constraint Optimizer**, etc.

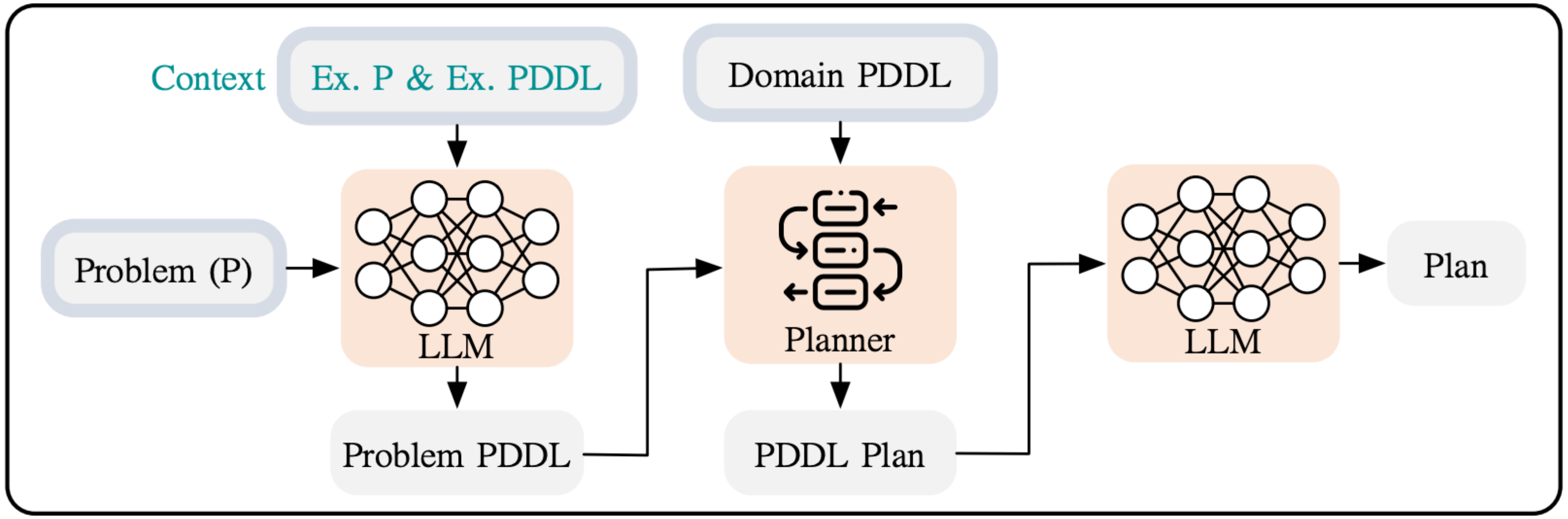


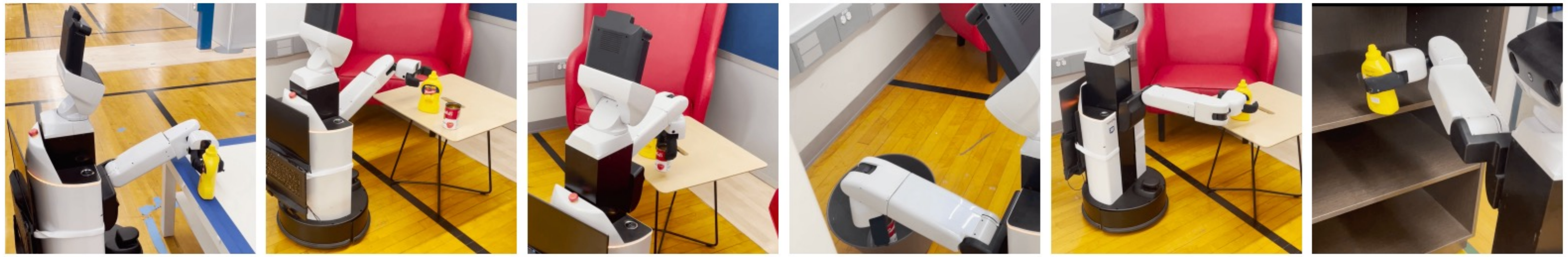


[1] LOGIC-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning. EMNLP 2023.

Symbolic Solver Aided Methods

For planning tasks, first **generate the PDDL descriptions**, then **use a PDDL Planner** (e.g., FastDownward) to produce the plan





(a) grasp bottle (b) free gripper (c) grasp soup can (d) place soup can (e) re-grasp bottle (f) place bottle

Tidy-Up Problem PDDL Generated by LLM+P

Problem (P): You are a home robot with one gripper. The distance between coffee table and side table is 10. The distance between coffee table and pantry is 20... You are at the coffee table. There is a mustard bottle... Your goal is to move objects to their destinations...

Problem PDDL generated by LLM+P:

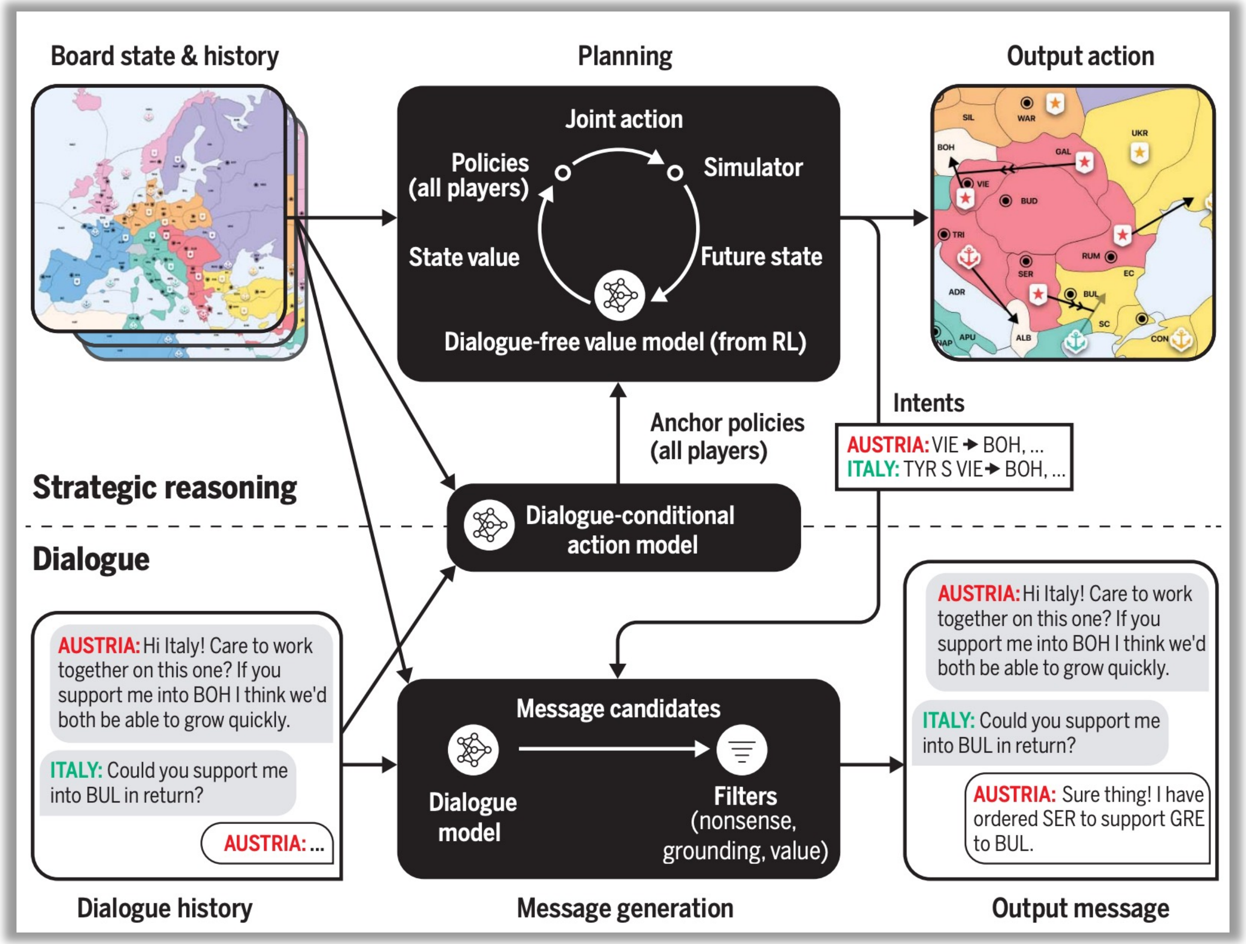
```
(:objects coffee-table side-table
recycle-bin pantry - location
mustard-bottle soup-can - object)
(:init (= (total-cost) 0) (=
(distance coffee-table side-table)
10) (= (distance coffee-table
pantry) 20) ... (robot-at
coffee-table) (at mustard-bottle
coffee-table) (at soup-can
side-table) (hand-empty) )
(:goal (and (at mustard-bottle
pantry) (at soup-can recycle-bin)))
(:metric minimize (total-cost)) )
```

[1] LLM+P: Empowering Large Language Models with Optimal Planning Proficiency. Arxiv 2023.

Symbolic Solver Aided Methods

外交风云：沟通与策略桌游

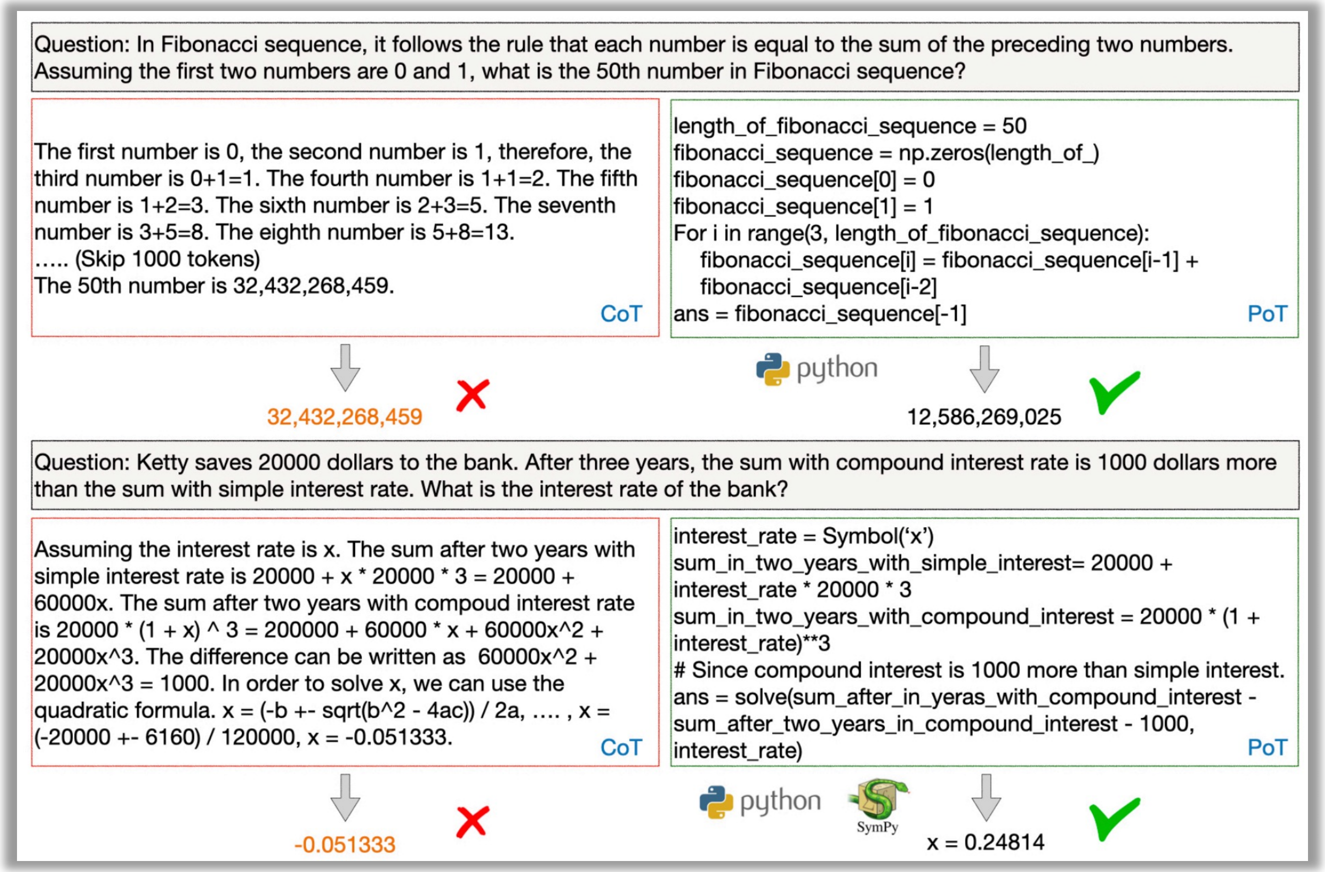
- 神经网络：
  - 理解玩家消息
  - 与用户进行对话
- 符号推理
  - 进行策略推理，生成最佳行动
  - 游戏规则引擎+博弈论求解器

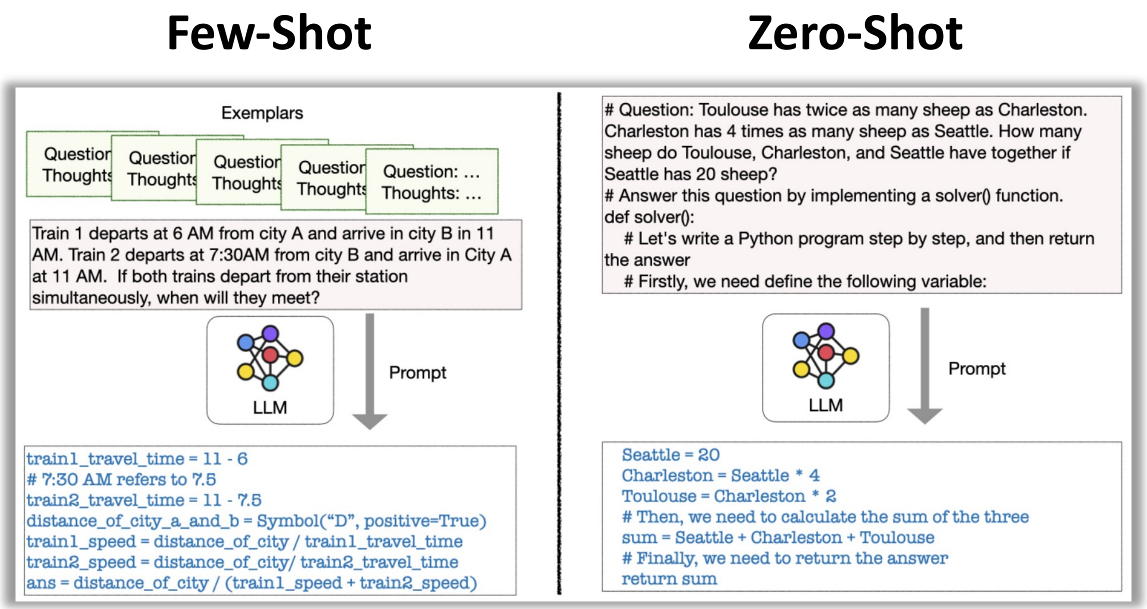


[Human-level play in the game of Diplomacy by combining language models with strategic reasoning](). Science 2022.

Program Aided Methods

Convert the reasoning process expressed in natural language into a **programming language** (e.g., Python, SQL, etc.), and then **execute the program interpreter**

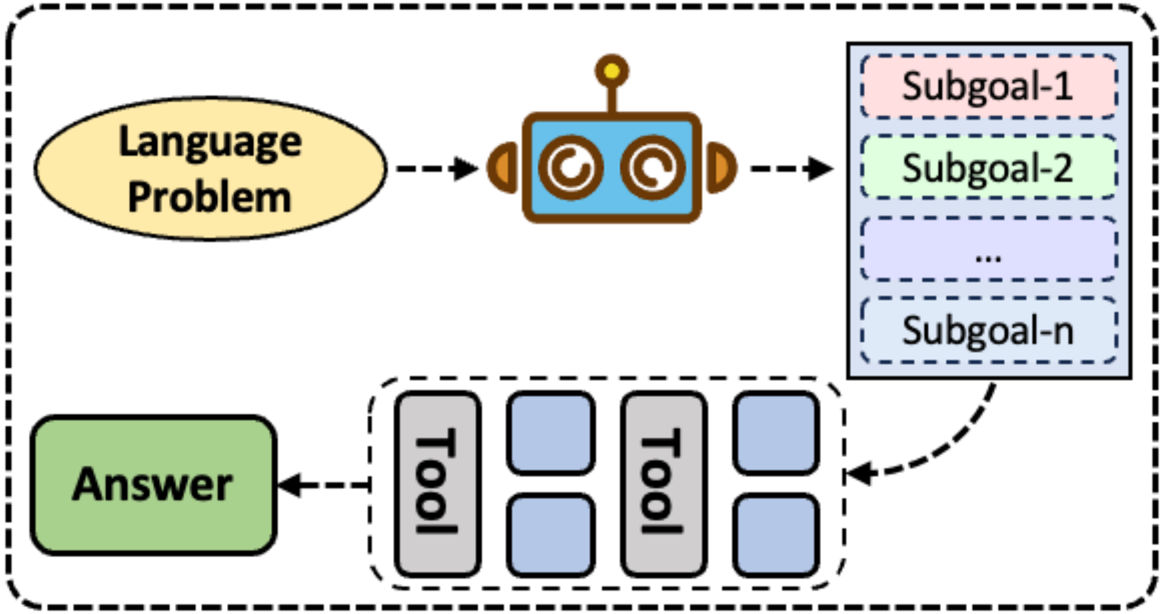




[1] PAL: Program-aided Language Models. ICML 2023.
[2] Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks. TMLR 2023.

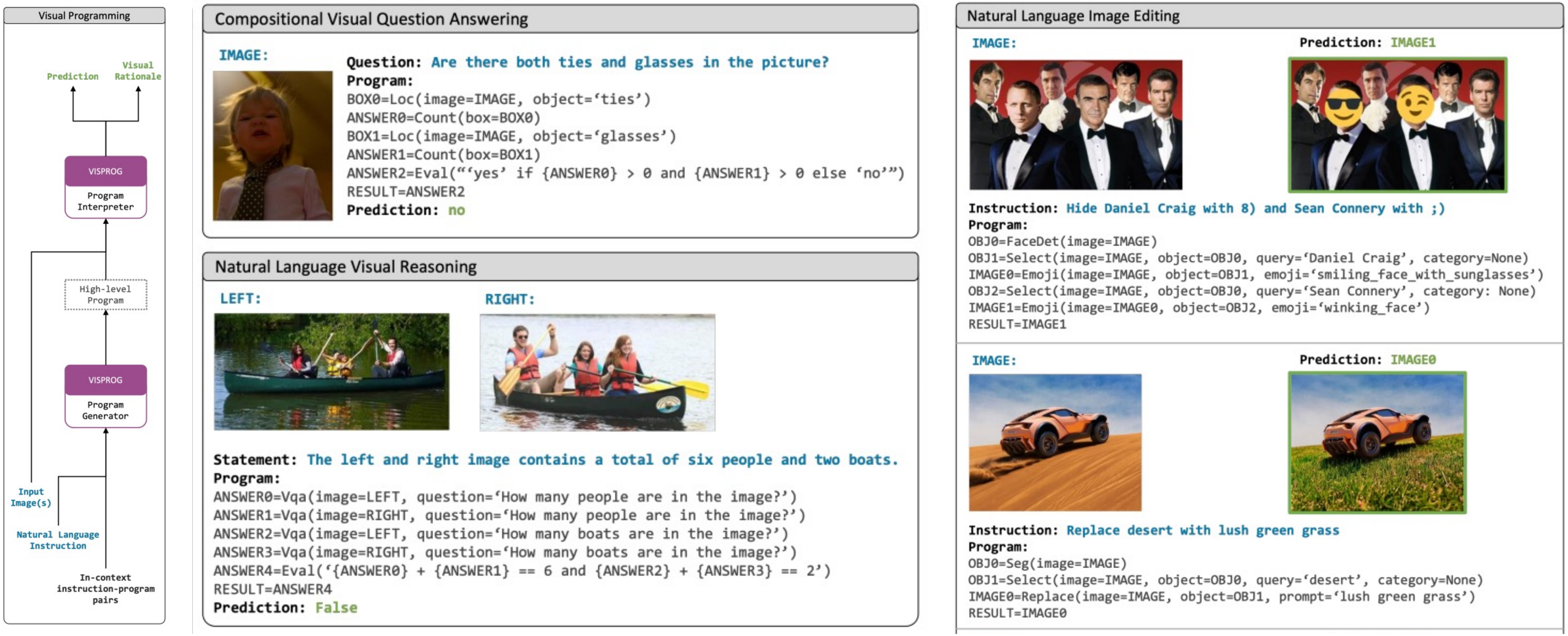
Tool Integrated Methods

Tool Integrated Methods



Tool Integrated Methods

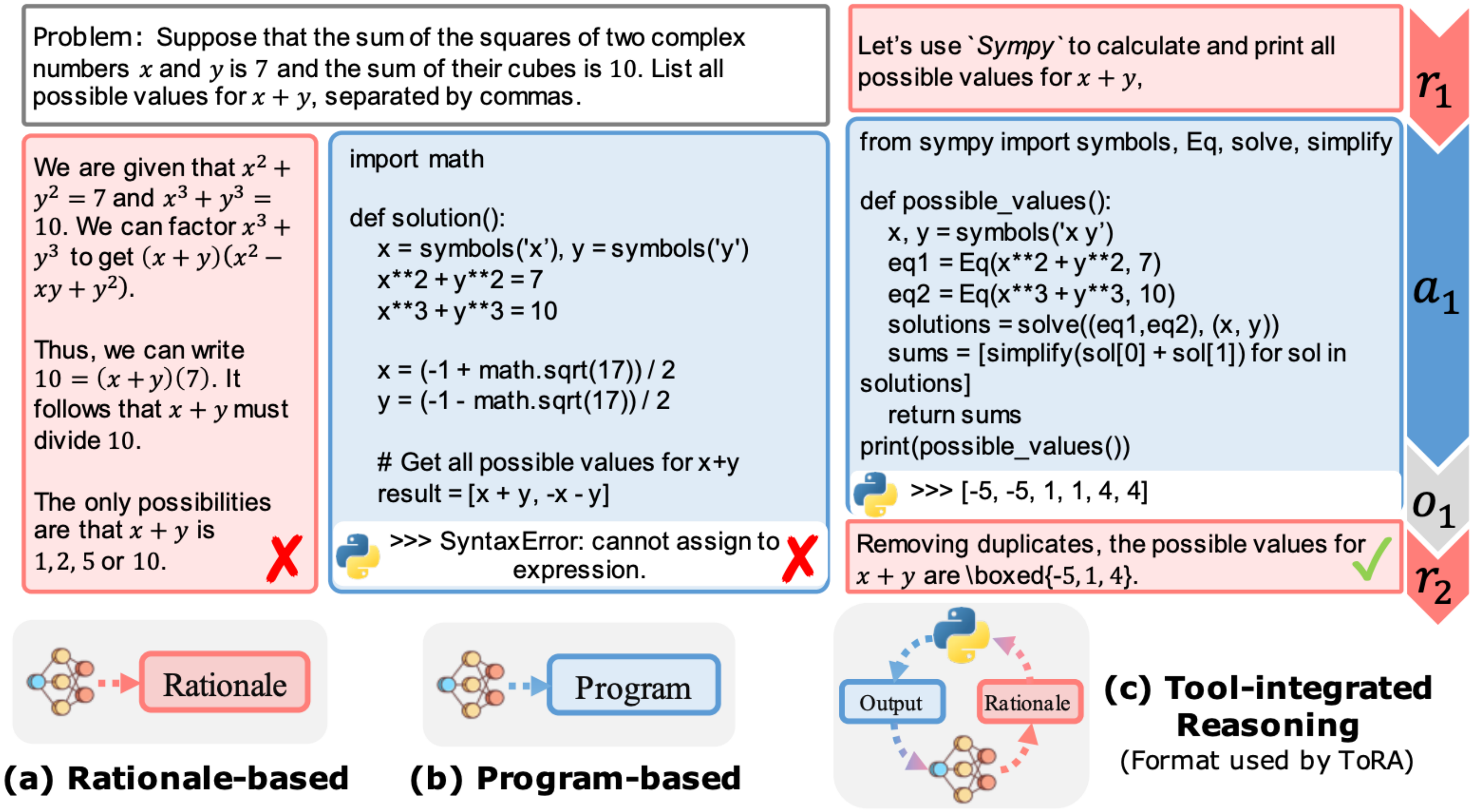
LLMs **decompose the reasoning task** and **invoke tools/APIs/models to accomplish subgoals**



[1] Visual Programming: Compositional visual reasoning without training. CVPR 2023 (Best Paper).

Tool Integrated Methods

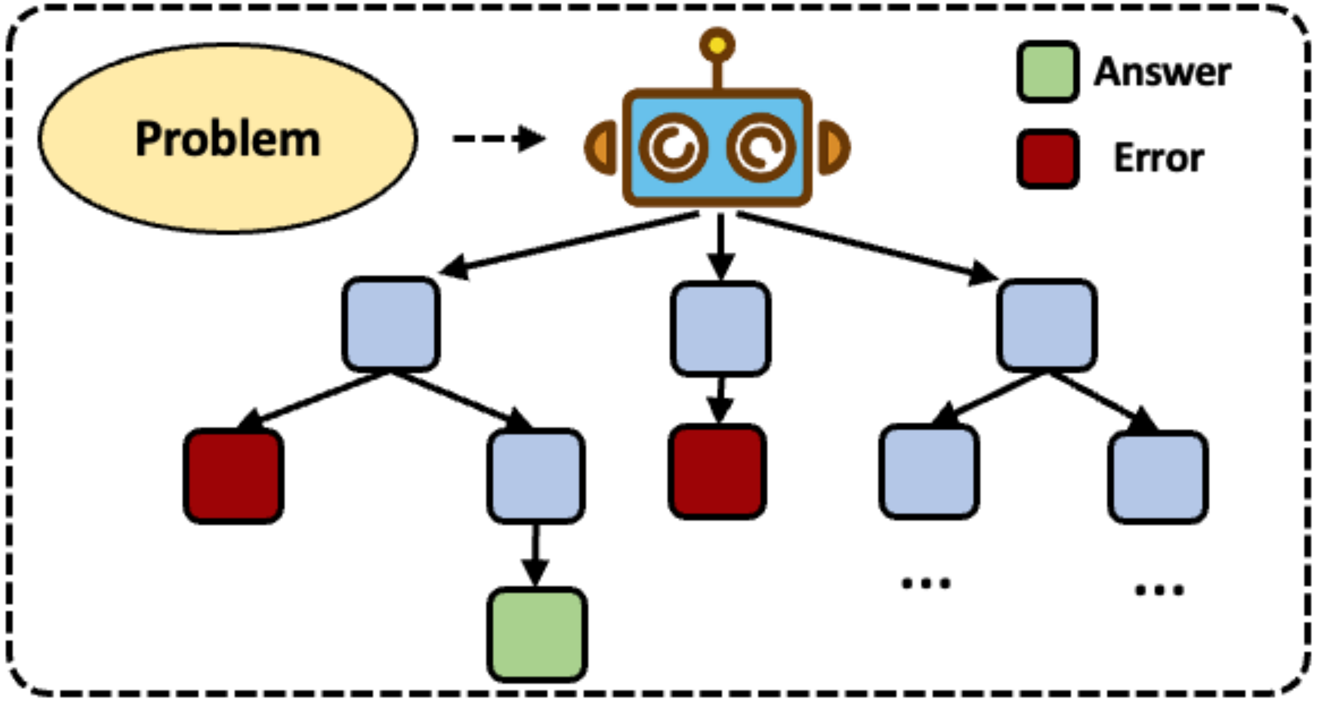
LLMs **decompose the reasoning task** and **invoke tools/APIs/models to accomplish subgoals**



[1] ToRA: A Tool-Integrated Reasoning Agent for Mathematical Problem Solving. ICLR 2024.

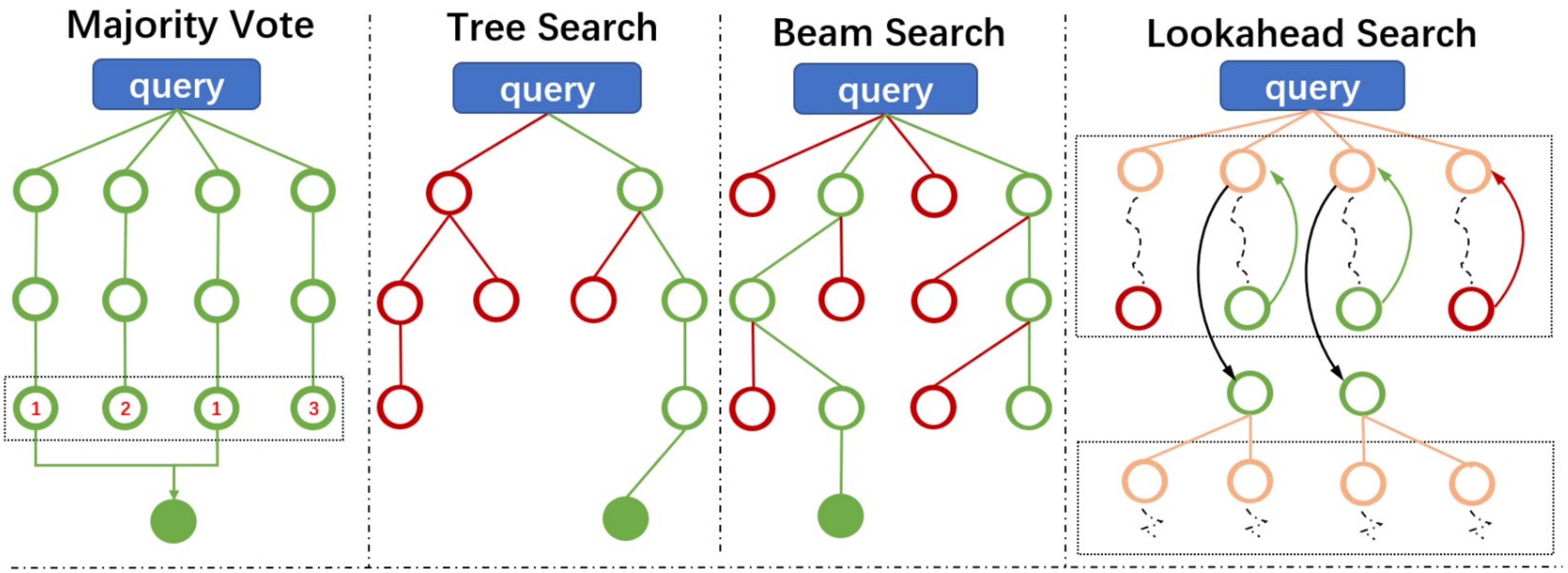
Search Augmented Methods

Search Augmented Methods



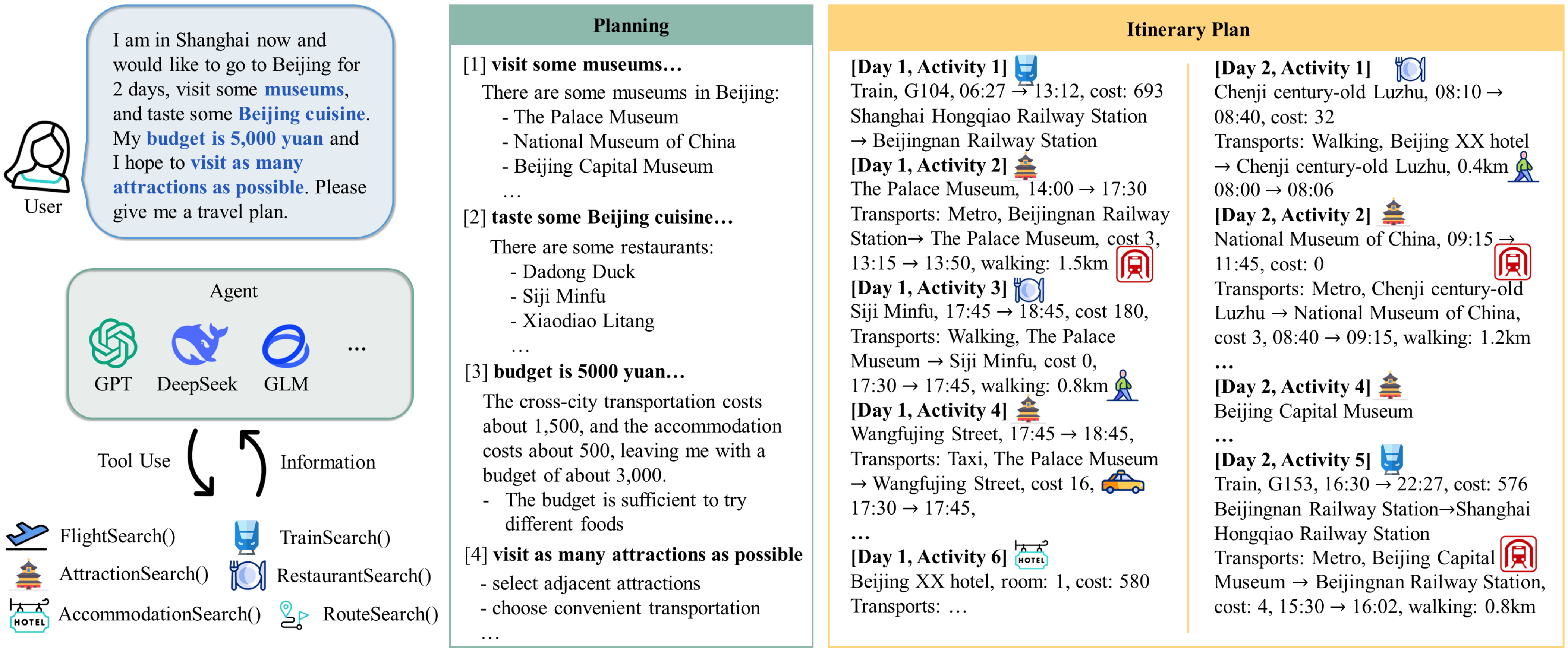
Search Augmented Methods

Since LLMs rely on **next-token prediction**, search algorithms can be applied over candidate tokens to optimize the reasoning path



Application: China Travel Planning

Given user requirements, including travel preferences, budget, and other constraints, generate a reasonable travel plan.



[1] **ChinaTravel: A Real-World Benchmark for Language Agents in Chinese Travel Planning**. https://arxiv.org/pdf/2412.13682

Application: China Travel Planning

Environment constraints: transportation; attraction opening hours and ticket prices; restaurant hours, prices and categories; hotel information; temporal and spatial constraints, etc.

Evaluation Metrics	Environment Constraints
Cross-city Transportation	Available Trains or Airplanes across cities.
	Correct information of cost and schedule.
Inner-city Transportation	Available Metro, Taxi or Walking between different positions.
	Correct information of cost, distance and duration
Attractions	Available Attractions in the target city, visiting in their open time.
	Attraction choices should not be repeated throughout the trip.
	Correct information of cost.
Restaurants	Available Restruants in the target city, visiting in their open time.
	Restaurant choices should not be repeated throughout the trip.
	Breakfast, lunch, and dinner are served at their designated meal times.
	Correct information of cost.
Accommodation	Available Accommodation in the target city.
	Room information to meet headcounts.
Time	The given activity events occur in chronological order.
Space	Events at different positions should provide transport information.

Name	Syntax	Description
variables	$x, y, z, \cdots$	Variables that refer to activities in the travel planning domain.
not	*not expr*	The negation of an Boolean-valued expression.
and,or	$expr_1$ *and* $expr_2$	The conjunction/disjunction of an Boolean-valued expression.
<, >, ==	$expr_1 < expr_2$	Return an expression with built-in number comparison functions.
$+, -, *, /$	$expr_1 + expr_2$	Return an expression with built-in number calculation functions.
attributes	$cost(var)$	A function that takes activities as inputs and returns the attributes, such as cost, type or time.
relation	$dist(expr_1, expr_2)$	A function that takes locations as inputs and returns the distance.
effect	$var = expr$	An assignment affects a variable *var* with the expression *expr*.
union, inter, diff	$uni(\{var\}_1, \{var\}_2)$	Return a set with the built-in union/intersection/difference operations of given two sets.
enumerate	*for var in* $\{var\}$	Enumerate all variables in the collection $\{var\}$.
when	*if expr : effect*	The conditional effect takes a Boolean-valued condition of the expression *expr*, and the effect *effect*.

Domain-Specific Language (DSL) for logical constraints

```
# 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
```

(a) Dining expenses.

```
# Arriving in Shanghai should be before
6 PM on the second day.
return_time = 0
for act_i in day_activities(plan, 2):
  typ = activity_type(act_i)
  dest = transport_destination(act_i)
  if (typ=="train" or typ=="airplane")
and des=="Shanghai": return_time ==
activity_endtime(act_i)
return return_time < "18:00"
```

(b) Arrived Time.

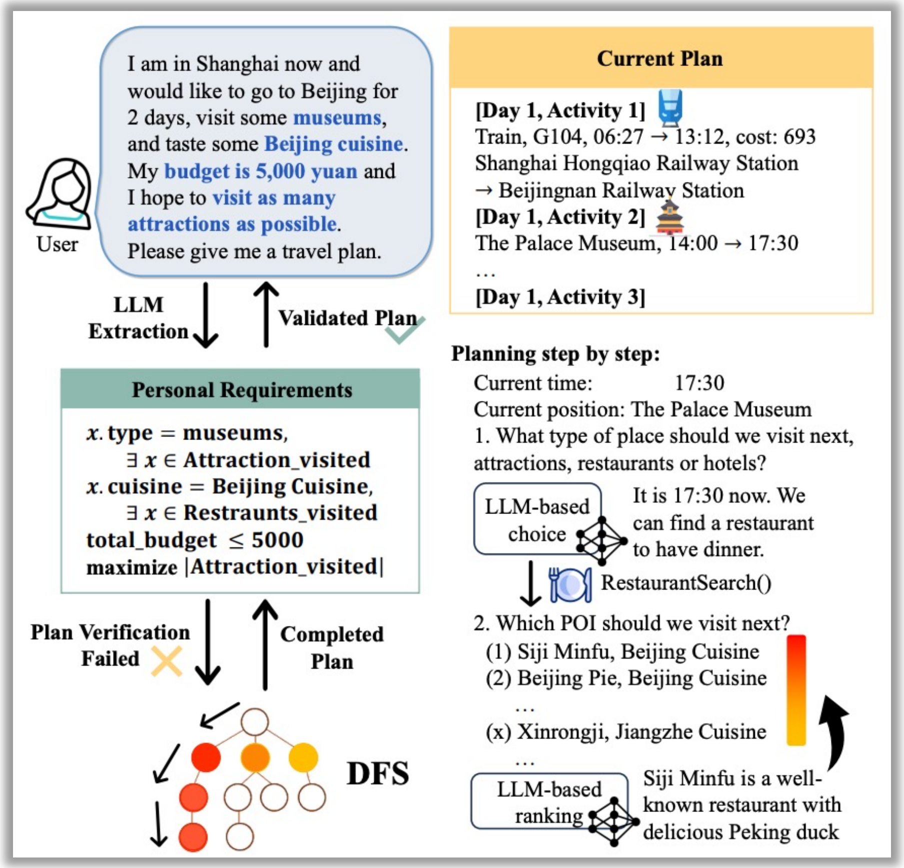
Examples of DSL expressions for logical constraints

[1] **ChinaTravel: A Real-World Benchmark for Language Agents in Chinese Travel Planning**. https://arxiv.org/pdf/2412.13682

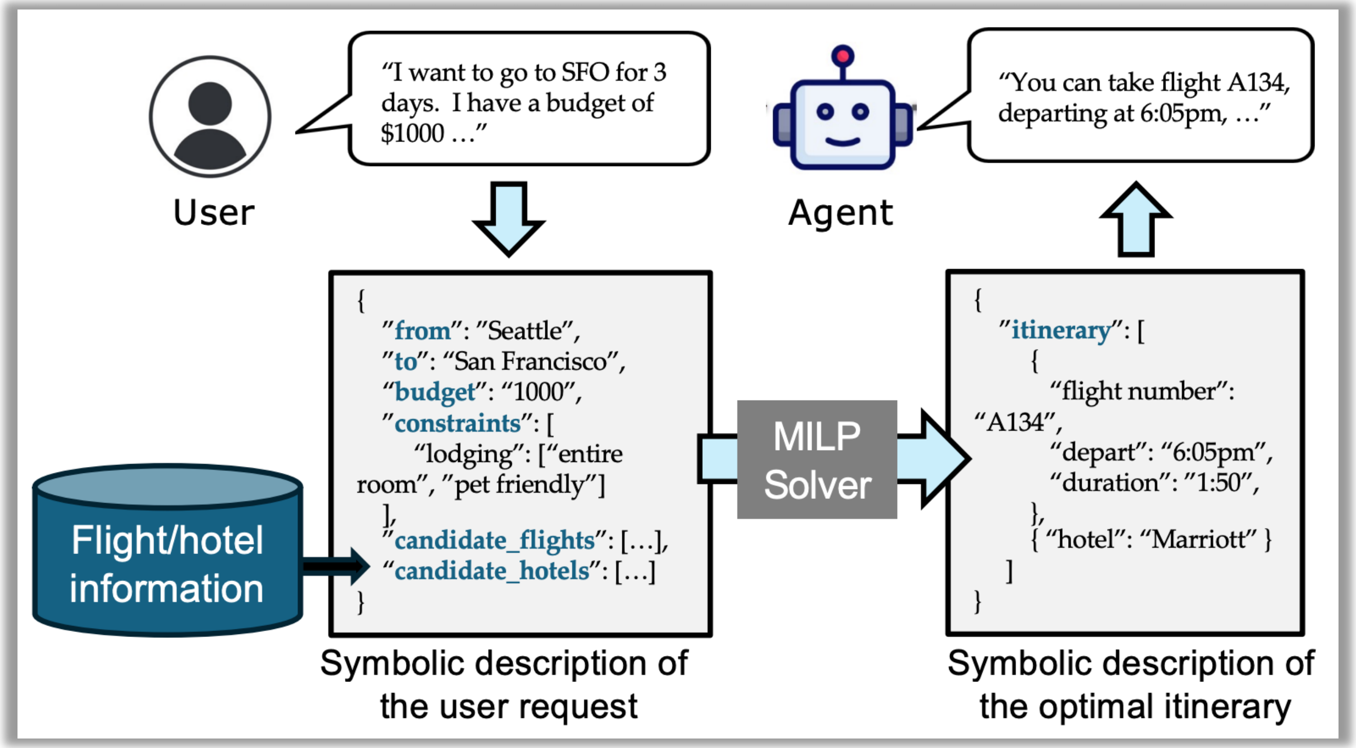
Application: China Travel Planning

Neuro-Symbolic Methods: LLM +Search; LLM + Constraint Optimizer

LLM understands the query, and then models travel planning as a **next-POI prediction** problem and solves it **using search algorithms**



LLM understands the query, and then model travel planning as a **mixed-integer optimization problem** and solve it **using an MILP solver**



Application: China Travel Planning

Pure LLM methods

		DR	EPR Mic.	EPR Mac.	LPR Mic.	LPR Mac.	C-LPR	FPR	DR	EPR Mic.	EPR Mac.	LPR Mic.	LPR Mac.	C-LPR	FPR
		Easy (#300)							Human-Val (#154)						
Act		70.4	49.9	0	64.6	30.8	0	0	-						
		97.5	70.8	0	86.8	68.8	0	0	-						
ReAct (zero-shot)		43.3	40.8	0	41.9	19.6	0	0	36.4	29.5	0.65	35.2	16.2	0.38	0
		95.4	48.2	0	71.3	32.9	0	0	**96.1**	50.5	0	**72.4**	32.5	0	0
ReAct (one-shot)		77.5	68.3	6.25	74.1	52.5	5.77	5.42	55.2	**57.3**	2.60	64.6	44.2	1.71	2.60
		94.2	68.1	0	**89.4**	**70.8**	0	0	69.5	46.3	0	63.6	46.8	0	0
NeSy Planning		75.3	**75.3**	75.3	70.4	52.6	70.4	52.6	51.9	53.2	52.5	47.0	**37.6**	46.5	**37.0**
		75.0	73.6	**64.0**	73.5	63.3	**61.7**	**60.6**	45.4	50.1	45.4	40.9	29.8	**38.5**	27.9
		72.3	67.0	34.0	70.4	49.6	32.6	28.3	42.8	47.4	42.2	36.2	27.2	34.4	25.3
		32.0	31.9	31.3	29.1	21.0	28.3	21.0	25.9	25.8	24.0	22.3	12.3	20.5	11.0
		30.3	30.3	30.3	27.6	19.6	27.6	19.6	37.6	38.2	37.6	32.7	18.8	32.2	18.8

Neuro-Symbolic Methods

TTG (oracle)		18.3	21.5	8.66	17.2	15.0	8.23	8.66	9.09	12.8	2.59	7.65	5.19	2.39	1.29
LLM-Modulo* (Oracle Verifier)		48.3	94.5	4.33	58.4	43.6	4.11	4.33	61.6	90.2	2.59	75.9	51.2	2.75	2.59
		91.6	88.2	7.66	**95.5**	**84.6**	7.66	7.00	91.5	87.2	3.24	**92.9**	**66.2**	2.87	3.24
		30.0	80.5	0.0	62.7	25.0	0.0	0.0	35.0	75.3	0.0	61.6	19.4	0.0	0.0
		28.6	69.4	0.0	55.2	8.33	0.0	0.0	19.4	74.1	0.0	43.4	5.19	0.0	0.0
		10.3	90.5	0.0	39.1	9.0	0.0	0.0	3.24	**92.2**	0.0	31.4	4.54	0.0	0.0
NeSy Planning* (Oracle Translation)		**82.6**	**81.7**	**75.0**	**82.2**	75.3	**75.0**	**74.0**	**58.4**	59.6	**57.7**	53.8	46.1	**52.0**	**45.4**
		66.6	66.7	66.0	64.6	63.6	64.6	62.6	52.6	46.9	42.9	47.6	40.9	43.9	40.9
		69.3	69.3	59.3	70.2	59.6	59.3	57.9	53.2	55.1	54.5	48.0	42.8	47.6	40.9
		52.6	52.6	52.6	50.4	45.3	50.4	45.6	40.9	42.8	42.8	37.7	28.5	37.7	27.9
		33.3	33.2	32.6	32.1	32.0	31.4	32.3	29.2	29.1	26.6	25.4	20.1	23.4	19.4
		Human-Test (#1000)							NeSy Planning* (Oracle Translation)						
NeSy Planning		**44.6**	**44.5**	**42.6**	**38.7**	**23.3**	**37.6**	**23.3**	**60.6**	**60.3**	**59.0**	**53.6**	**32.0**	**52.5**	**31.6**
		37.3	37.2	35.0	30.7	11.3	29.2	11.3	27.8	27.8	27.1	24.8	12.8	24.4	12.8
		36.6	36.5	34.6	29.6	6.43	28.5	6.43	41.1	41.1	40.6	34.6	13.8	34.2	13.8

Exploit search or optimization methods clearly improve the planning accuracy

Summary & Open Problems

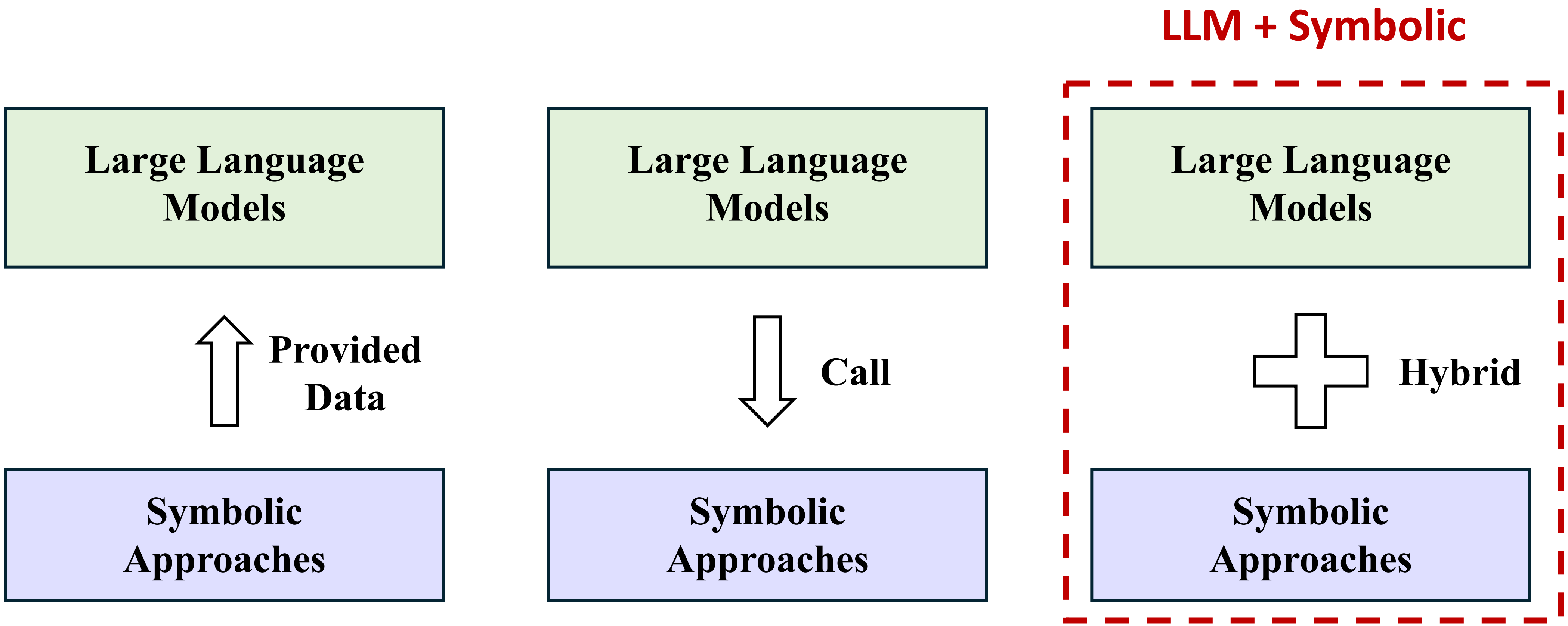
☐ Key Ideas:

- LLM converts natural language into symbolic Language
- Symbolic solvers solve the formal reasoning and planning

☐ Open Problems:

- Quality of Natural Language to Symbolic Language
- Open Environment with new symbolic concept
- Limited Generalization of Symbolic Solvers: how to automatically select or learn the most suitable symbolic representation

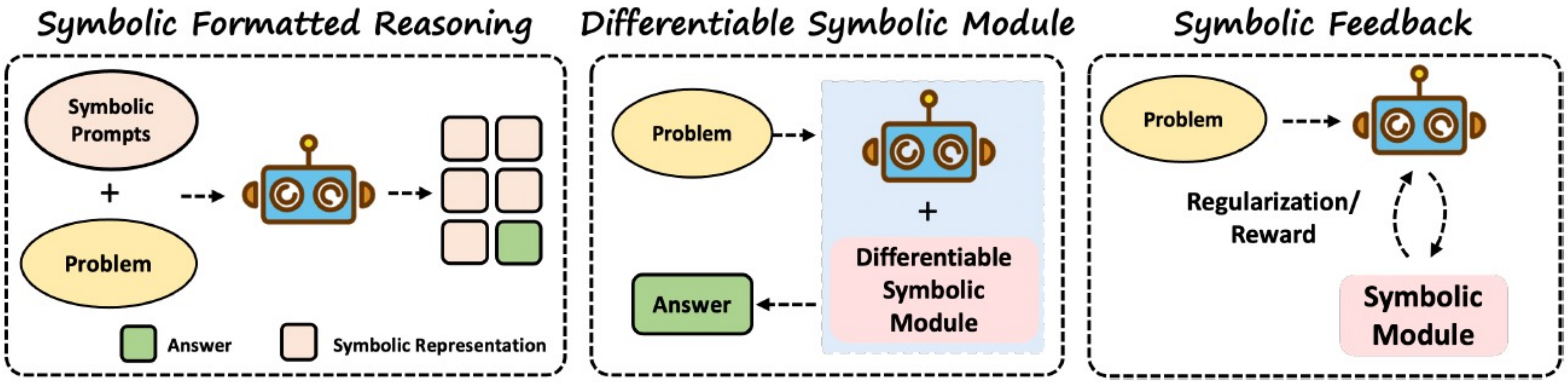
Neuro-Symbolic Methods



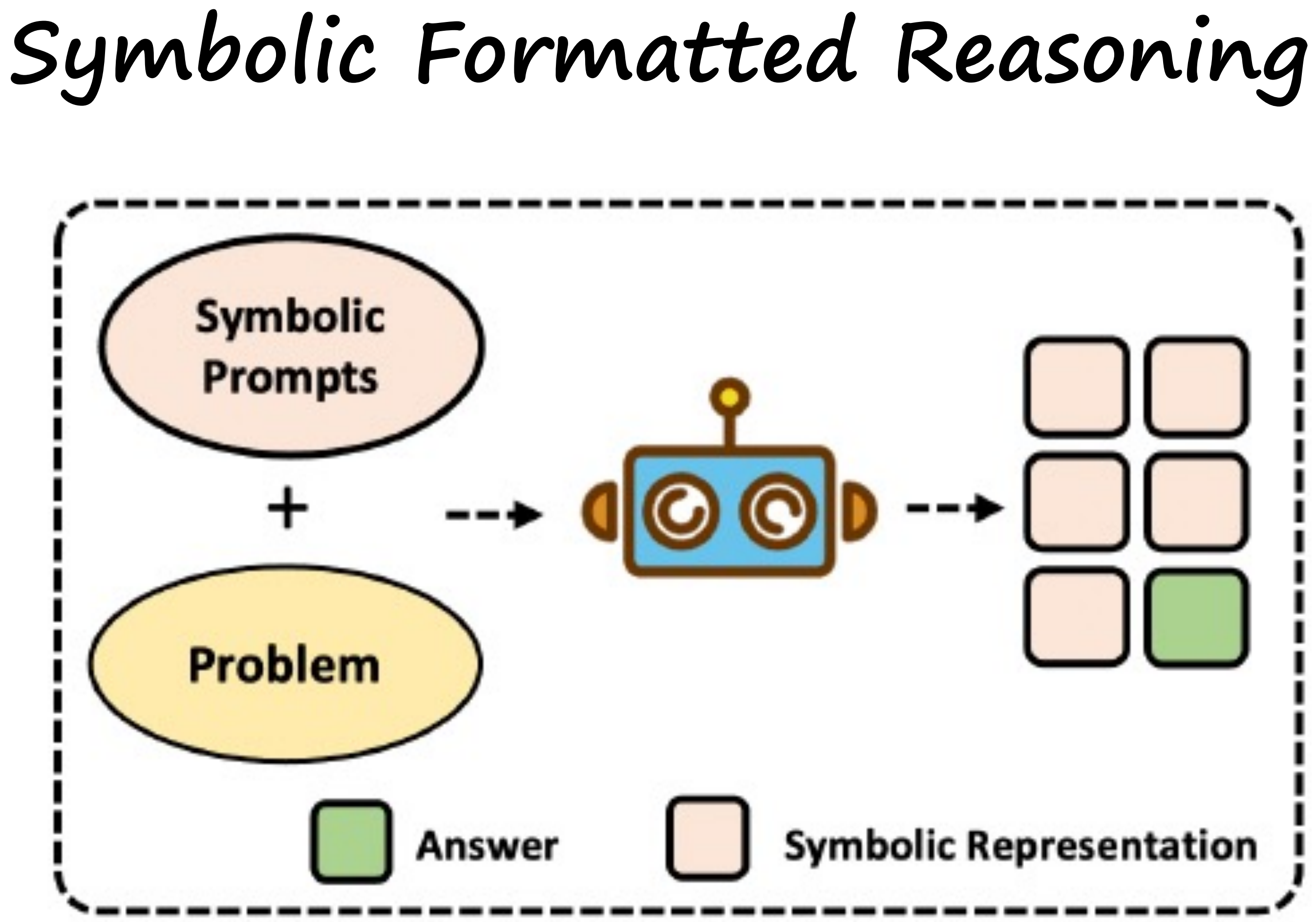
LLM + Symbolic

☐ Basic Ideas：

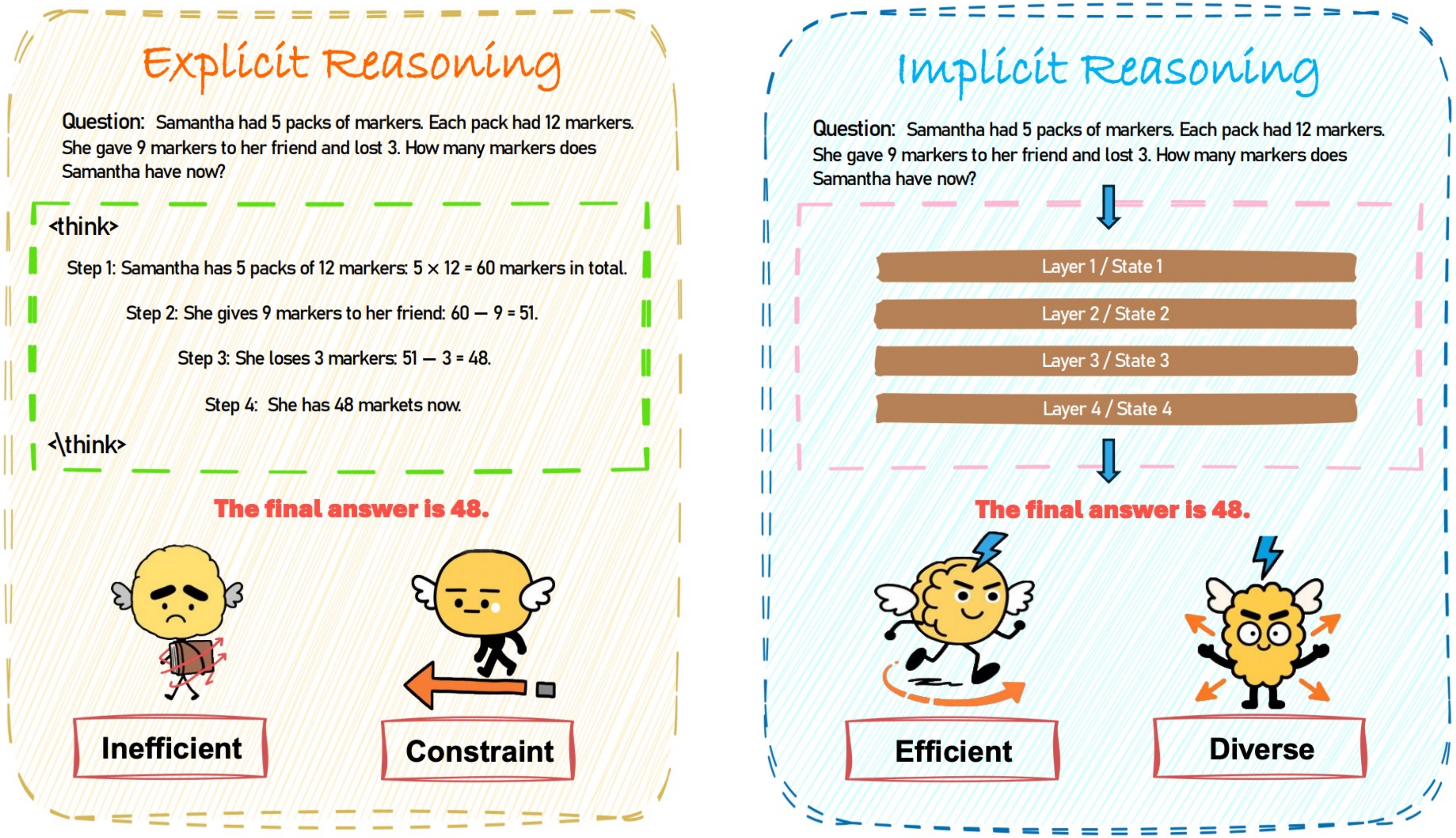
- Symbolic Formatted Reasoning： representing the reasoning process in symbolic language, instead of the natural language
- Differentiable Symbolic Module： Design differentiable symbolic modules to enable joint optimization with neural networks?
- Symbolic Feedback： Symbolic feedback provides more fine-grained losses or rewards, thereby guiding the optimization process



Symbolic Formatted Reasoning



Explicit Reasoning vs. Implicit Reasoning



Explicit reasoning express the reasoning process via natural language

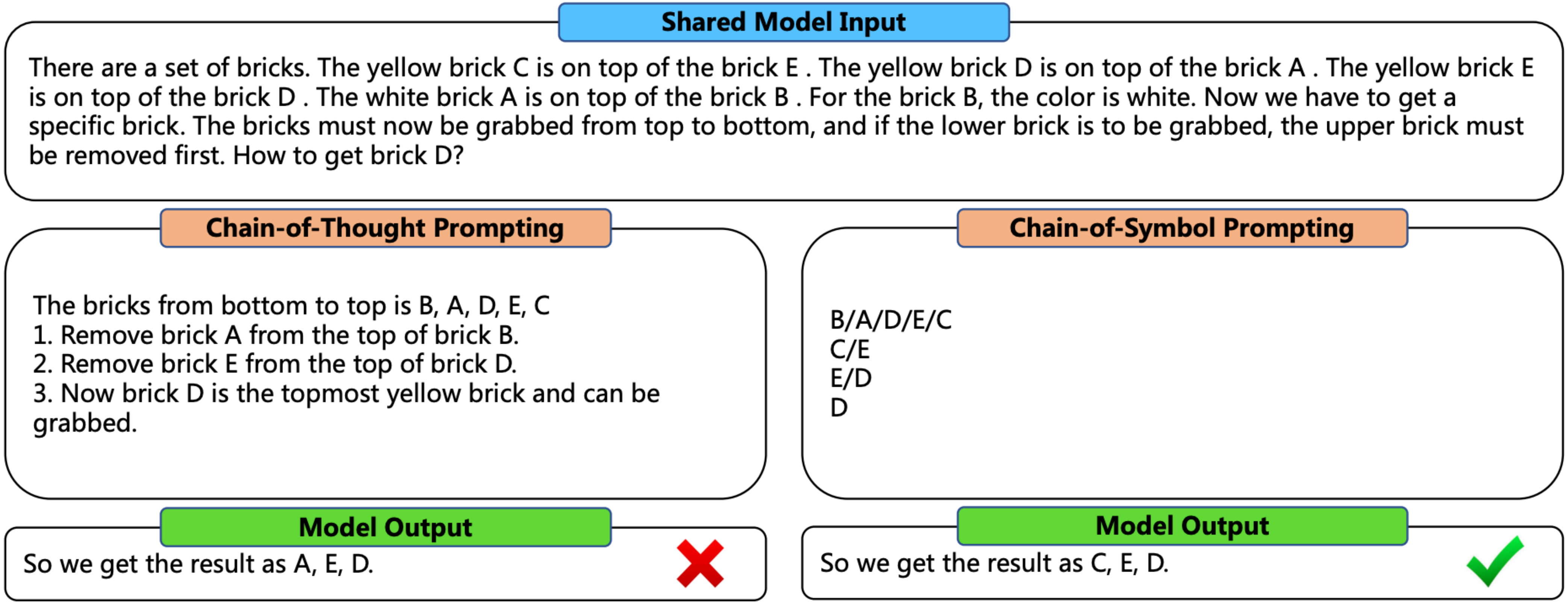
Implicit reasoning handles the reasoning process internally across different layers or states

Symbolic Formatted Reasoning

How about represent the reasoning process with symbolic language?

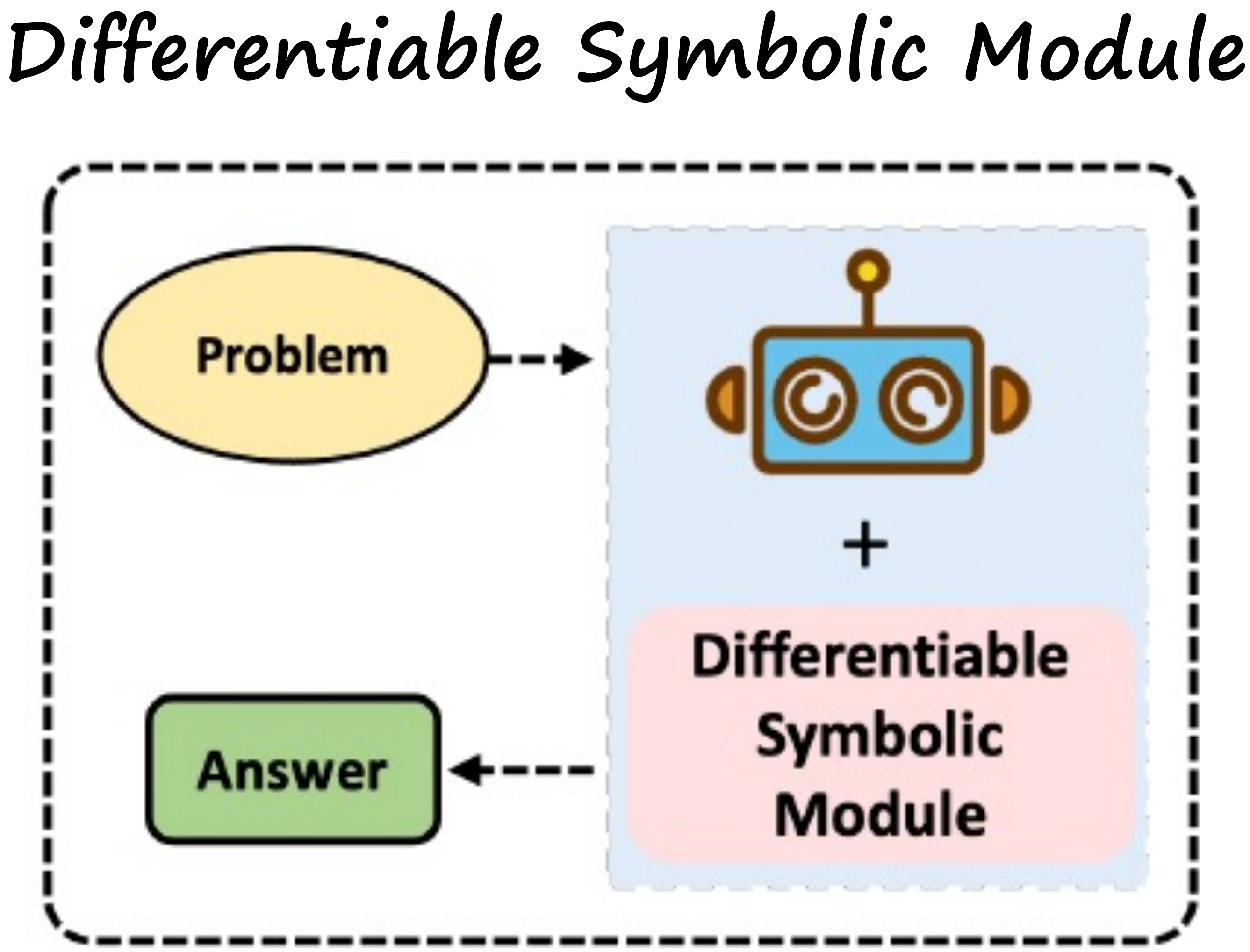
Symbolic Formatted Reasoning

A straightforward example: Chain-of-Symbol



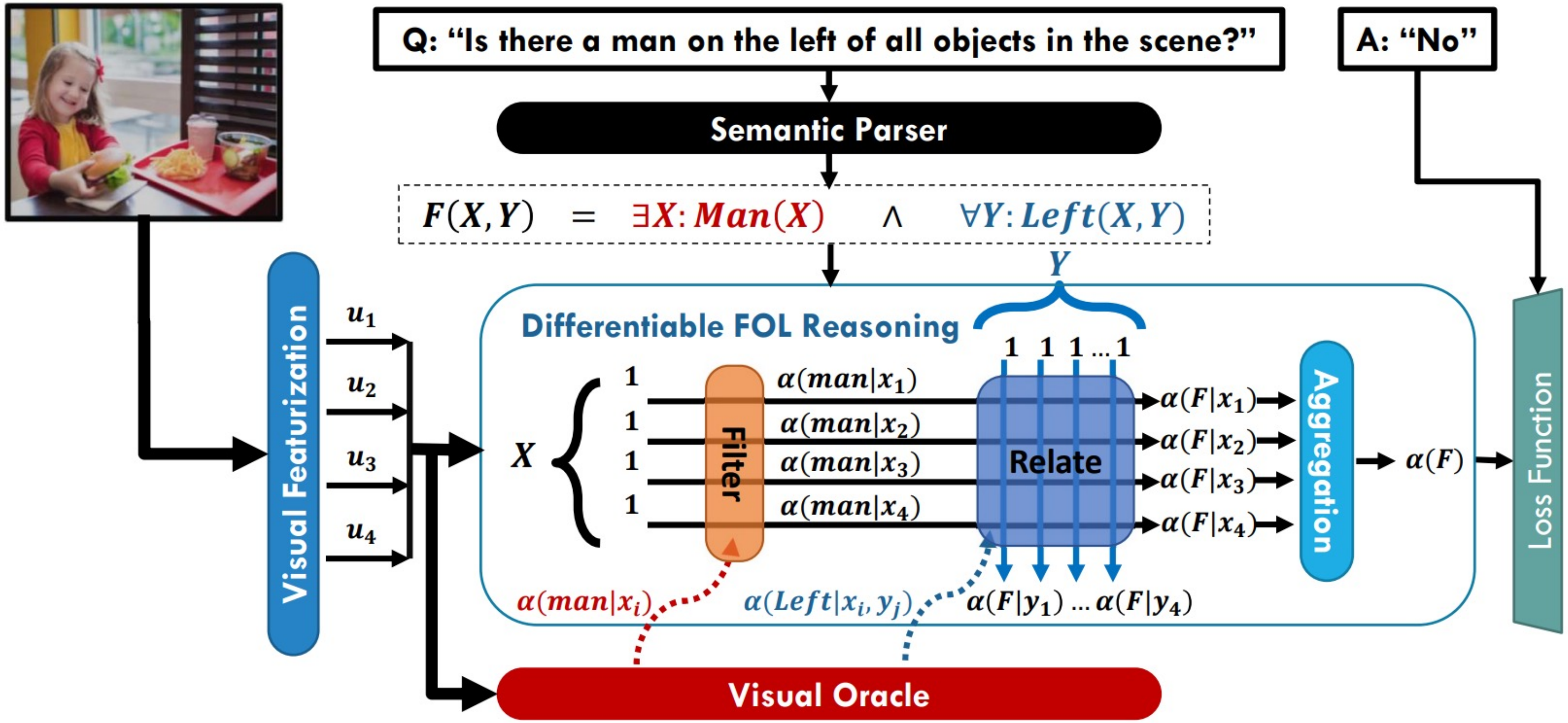
[1] Chain-of-Symbol Prompting for Spatial Reasoning in Large Language Models. COLM 2024.

Differentiable Symbolic Module



Differential Symbolic Module

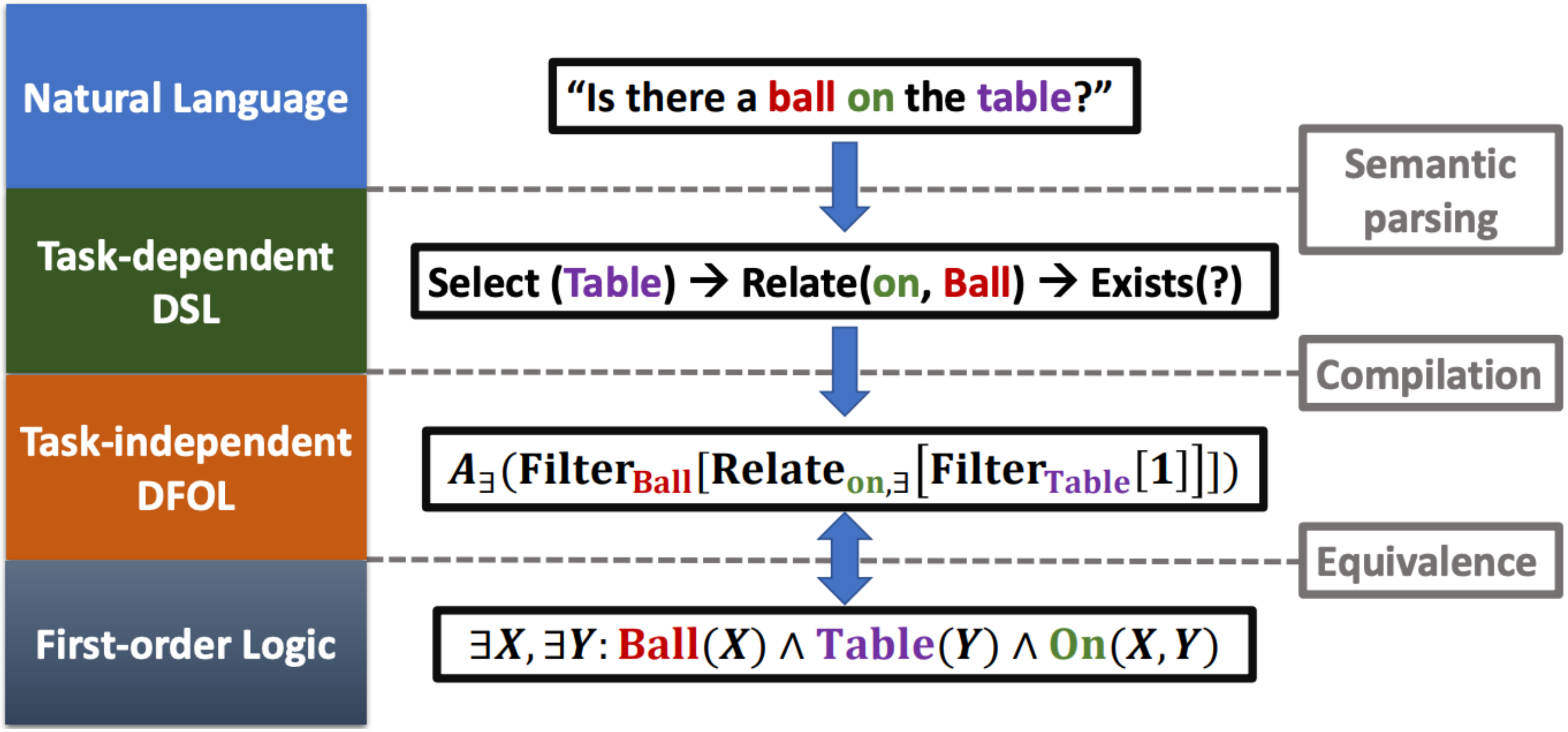
Disentangle perception and reasoning: use **LLMs or MLLMs for perception** and **symbolic systems for reasoning**, and **make logical reasoning differentiable** by incorporating techniques such as **fuzzy logic**



[1] Neuro-Symbolic Visual Reasoning: Disentangling "Visual" from "Reasoning". ICML 2020.

Differential Symbolic Module

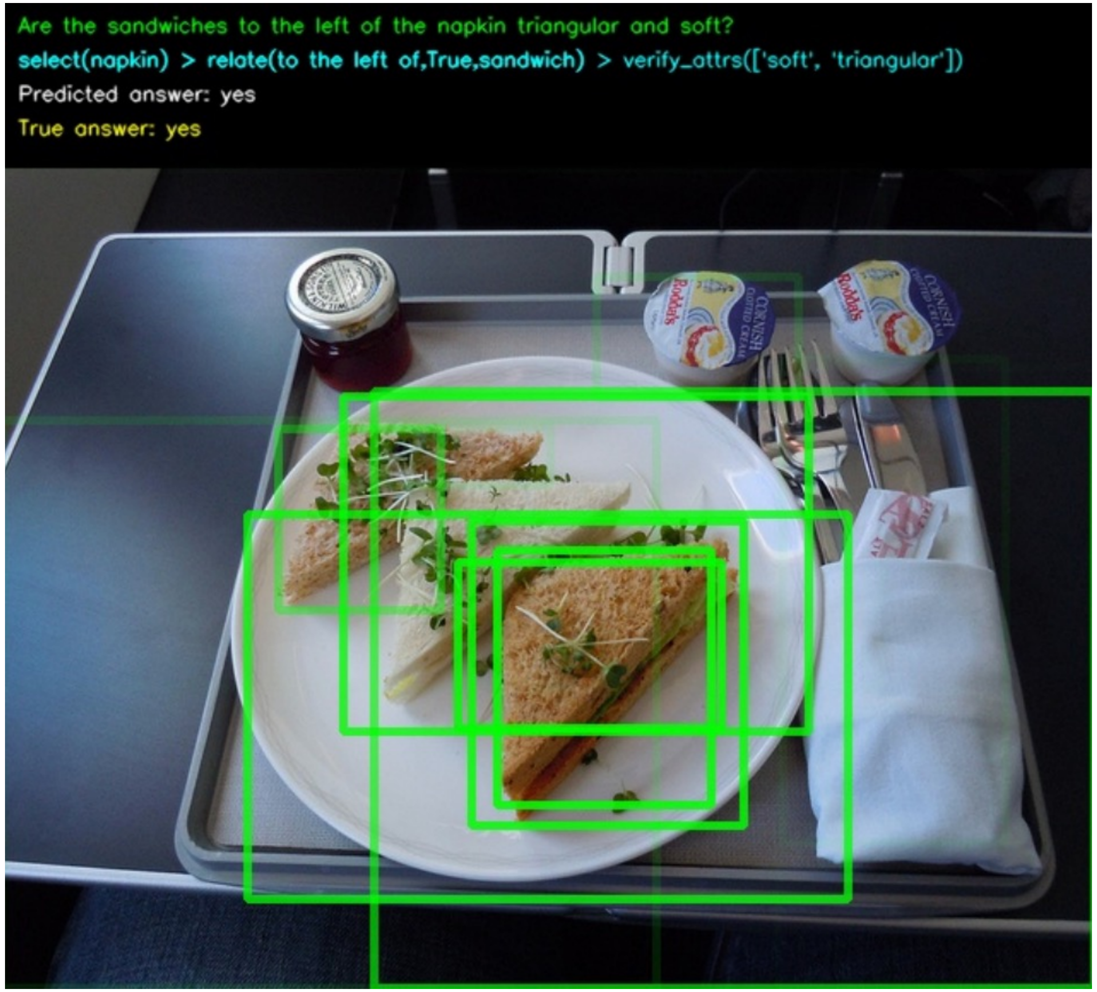
An example of the language system: natural language -> DSL program -> DFOL <->FOL formula



[1] Neuro-Symbolic Visual Reasoning: Disentangling "Visual" from "Reasoning". ICML 2020.

Differential Symbolic Module

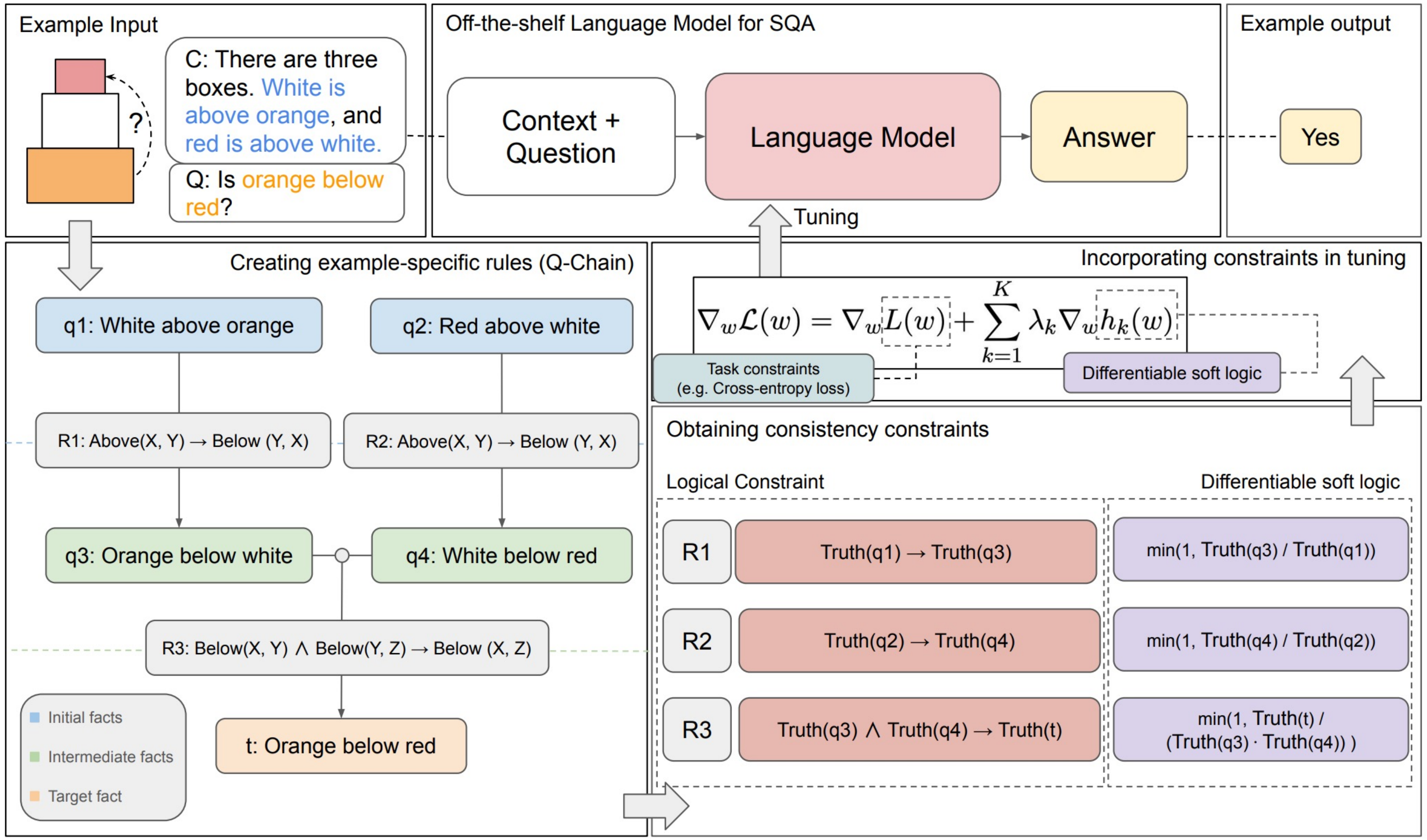
Some Visual Reasoning Examples



[1] Neuro-Symbolic Visual Reasoning: Disentangling "Visual" from "Reasoning". ICML 2020.

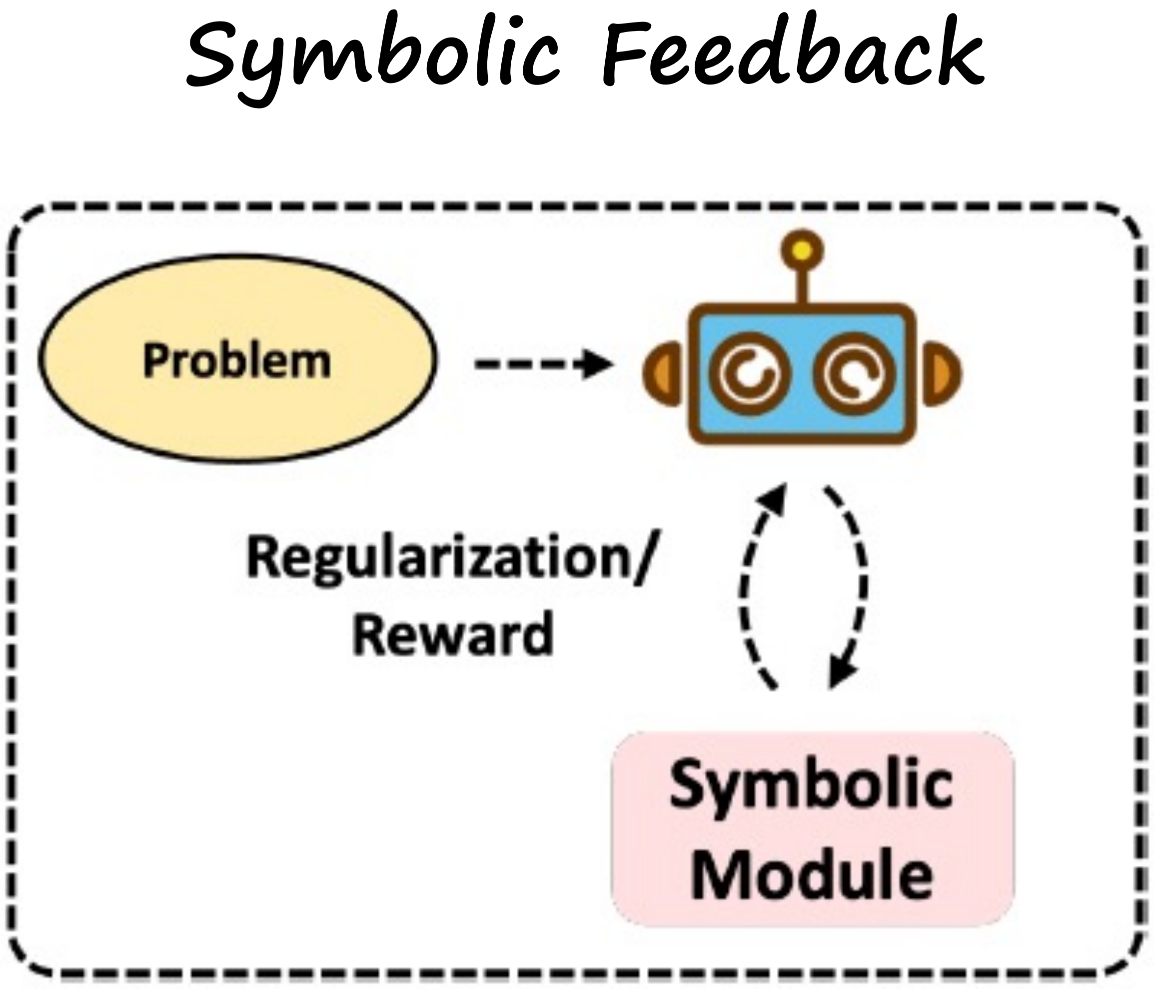
Differential Symbolic Module

Convert logical constraints into differentiable soft logic and add them to the objective as a regularization term



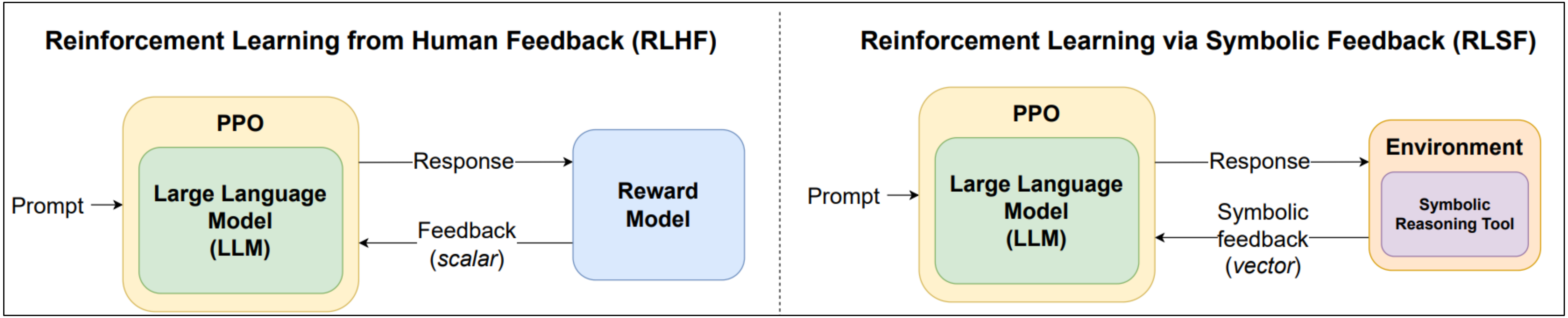
[1] Neuro-symbolic Training for Spatial Reasoning over Natural Language. Arxiv 2025.

Symbolic Feedback



Symbolic Feedback

RLVR: For reinforcement Learning, introduce symbolic tools (e.g., theorem provers, logic solvers, code compilers) instead of a reward model to provide **verifiable rewards**

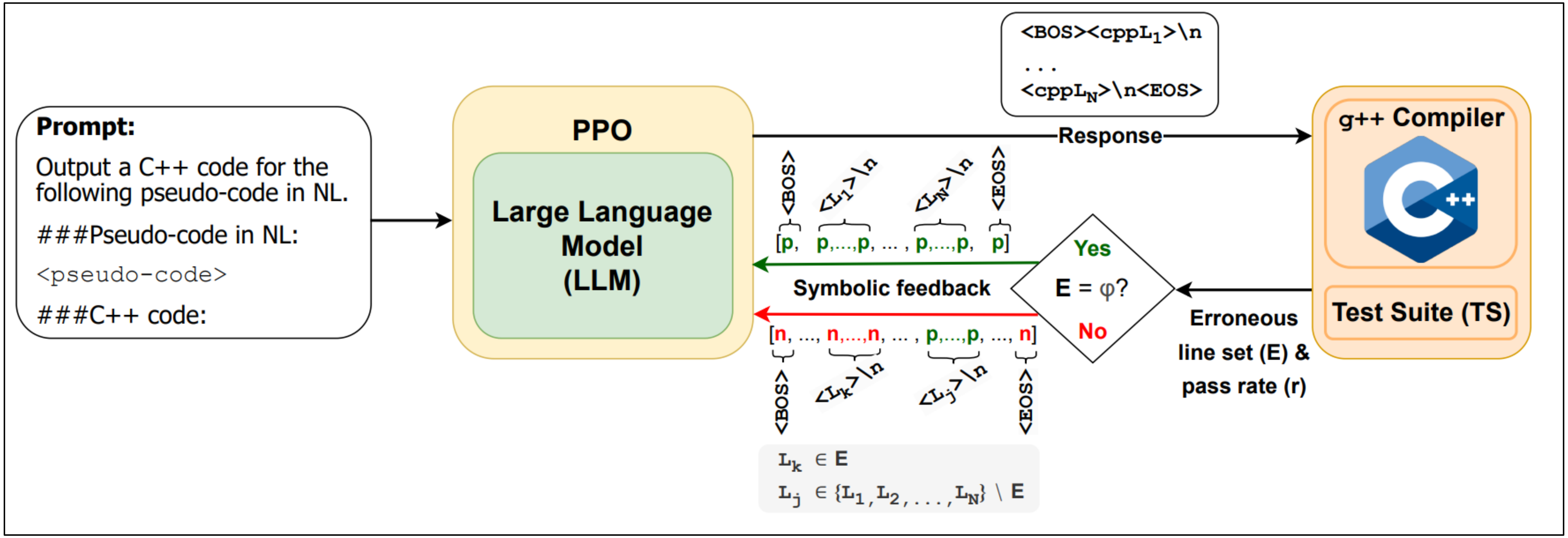


[1] RLSF: Reinforcement Learning via Symbolic Feedback. Arxiv 2024.

Symbolic Feedback

Example task: Natural Language Pseudo-code to C++ Code

- Task: Automated code synthesis from natural language descriptions
- Symbolic Feedback: g++ Compiler and Test Suite provide feedback

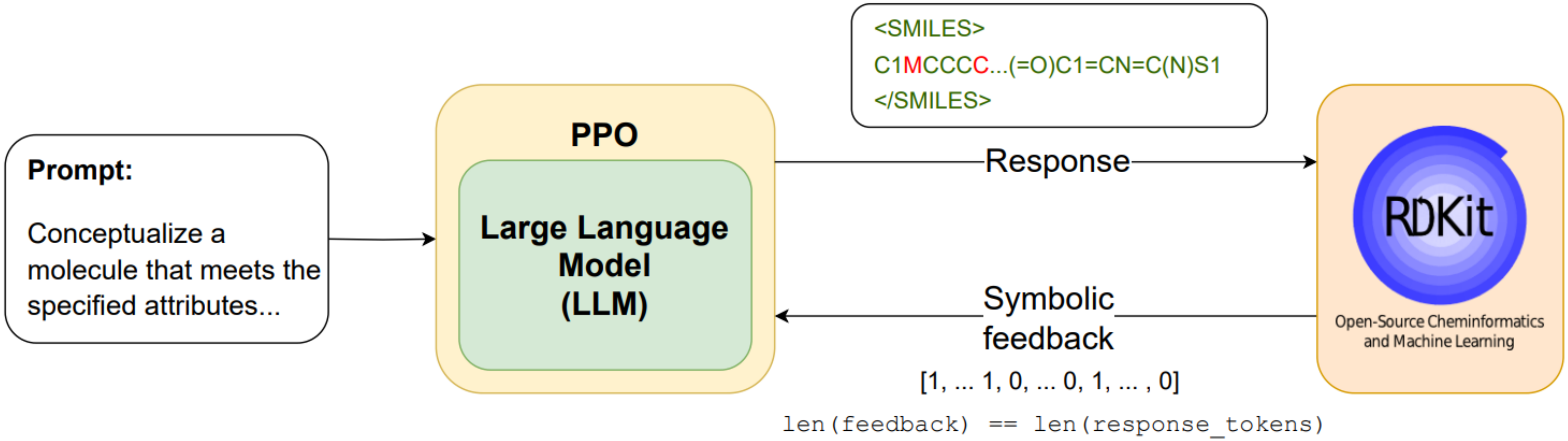


[1] RLSF: Reinforcement Learning via Symbolic Feedback. Arxiv 2024.

Symbolic Feedback

Example task: Chemistry (Molecule Generation, Forward Synthesis and Retrosynthesis)

- Task: Molecular Generation (MG), Forward Synthesis (FS), and Retrosynthesis (RS)
- Symbolic Feedback: RDKit, a widely adopted cheminformatics toolkit



[1] RLSF: Reinforcement Learning via Symbolic Feedback. Arxiv 2024.

Summary & Open Problems

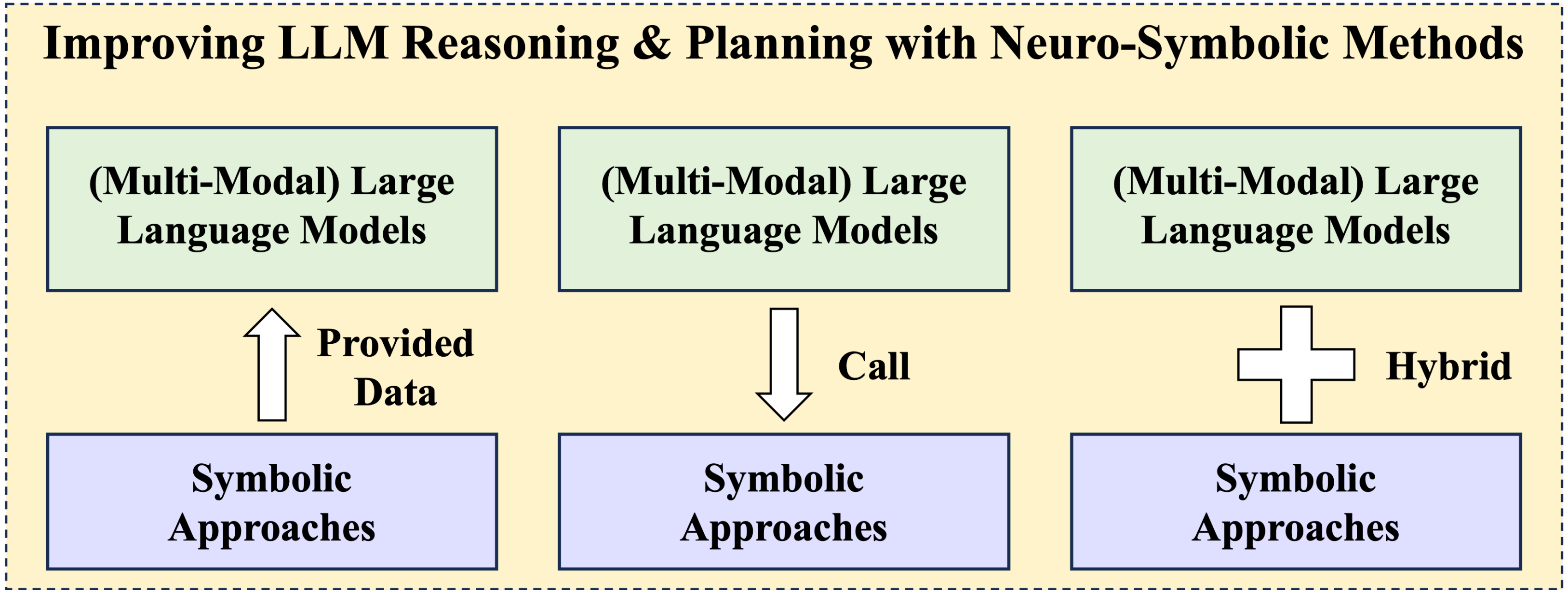
☐ Key Ideas:

- Represent the reasoning process with symbolic representation
- Symbolic systems provides feedback for reinforcement learning
- Differentiable symbolic reasoning modules

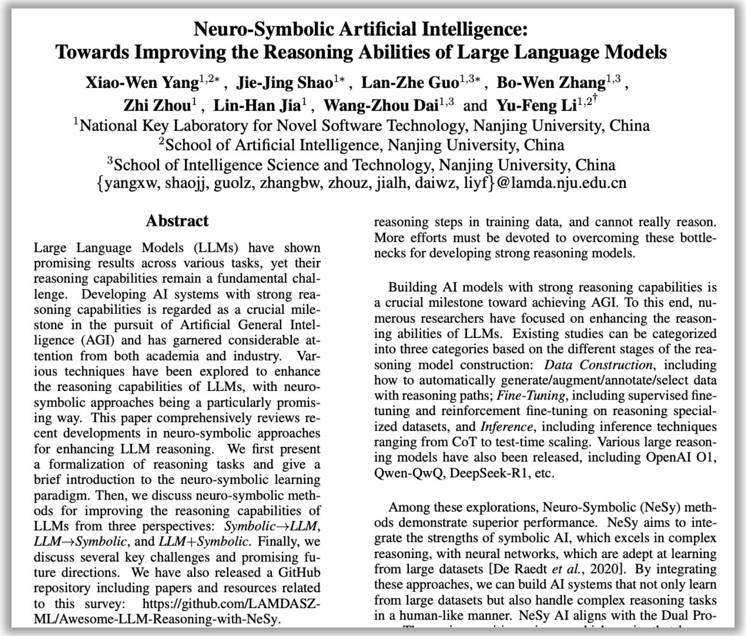
☐ Open Problems:

- Differentiability of symbolic reasoning
- More efficient joint optimization techniques
- The interpretability of the reasoning process
- ……

Summary



Our Survey in IJCAI 2025



Neuro-Symbolic Artificial Intelligence: Towards Improving the Reasoning Abilities of Large Language Models

Xiao-Wen Yang[1,2*], Jie-Jing Shao[1*], Lan-Zhe Guo[1,3*], Bo-Wen Zhang[1,3], Zhi Zhou[1], Lin-Han Jia[1], Wang-Zhou Dai[1,3] and Yu-Feng Li[1,2†]
[1]National Key Laboratory for Novel Software Technology, Nanjing University, China
[2]School of Artificial Intelligence, Nanjing University, China
[3]School of Intelligence Science and Technology, Nanjing University, China
{yangxw, shaojj, guolz, zhangbw, zhouz, jialh, daiwz, liyf}@lamda.nju.edu.cn

Abstract

Large Language Models (LLMs) have shown promising results across various tasks, yet their reasoning capabilities remain a fundamental challenge. Developing AI systems with strong reasoning capabilities is regarded as a crucial milestone in the pursuit of Artificial General Intelligence (AGI) and has garnered considerable attention from both academia and industry. Various techniques have been explored to enhance the reasoning capabilities of LLMs, with neuro-symbolic approaches being a particularly promising way. This paper comprehensively reviews recent developments in neuro-symbolic approaches for enhancing LLM reasoning. We first present a formalization of reasoning tasks and give a brief introduction to the neuro-symbolic learning paradigm. Then, we discuss neuro-symbolic methods for improving the reasoning capabilities of LLMs from three perspectives: *Symbolic→LLM*, *LLM→Symbolic*, and *LLM+Symbolic*. Finally, we discuss several key challenges and promising future directions. We have also released a GitHub repository including papers and resources related to this survey: https://github.com/LAMDASZ-ML/Awesome-LLM-Reasoning-with-NeSy.

reasoning steps in training data, and cannot really reason. More efforts must be devoted to overcoming these bottlenecks for developing strong reasoning models.

Building AI models with strong reasoning capabilities is a crucial milestone toward achieving AGI. To this end, numerous researchers have focused on enhancing the reasoning abilities of LLMs. Existing studies can be categorized into three categories based on the different stages of the reasoning model construction: *Data Construction*, including how to automatically generate/augment/annotate/select data with reasoning paths; *Fine-Tuning*, including supervised fine-tuning and reinforcement fine-tuning on reasoning specialized datasets, and *Inference*, including inference techniques ranging from CoT to test-time scaling. Various large reasoning models have also been released, including OpenAI O1, Qwen-QwQ, DeepSeek-R1, etc.

Among these explorations, Neuro-Symbolic (NeSy) methods demonstrate superior performance. NeSy aims to integrate the strengths of symbolic AI, which excels in complex reasoning, with neural networks, which are adept at learning from large datasets [De Raedt *et al.*, 2020]. By integrating these approaches, we can build AI systems that not only learn from large datasets but also handle complex reasoning tasks in a human-like manner. NeSy AI aligns with the Dual Pro-

Thanks For Your Lisenting!

GitHub Repo

https://github.com/LAMDASZ-ML/Awesome-Neuro-Symbolic-Learning-with-LLM