CNCC | CNCC Tutorial

神经符号学习：理论、方法与应用

戴望州 南京大学
李昊轩 北京大学
丁尧相 浙江大学
郭兰哲 南京大学

2026 年 10 月 21 日

国家战略

中华人民共和国中央人民政府
www.gov.cn

首页 | 简 | 繁 | EN | 登录 | 邮箱 | 无障碍

首页 > 国务院公报 > 2017年第22号

字号：默认 大 超大 | 打印

国 务 院 关 于 印 发
新一代人工智能发展规划的通知

国发〔2017〕35号

各省、自治区、直辖市人民政府，国务院各部委、各直属机构：
现将《新一代人工智能发展规划》印发给你们，请认真贯彻执行。

国 务 院
2017年7月8日

3. 统筹布局人工智能创新平台。

建设布局人工智能创新平台，强化对人工智能研发应用的基础支撑。人工智能开源软硬件基础平台重点建设支持知识推理、概率统计、深度学习等人工智能范式的统一计算框架平台，形成促进人工智能软件、硬件和智能云之间相互协同的生态链。群体智能服务平台重点建设基于互联网大规模协作的知识资源管理与开放式共享工具，形成面向产学研用创新环节的群智众创平台和服务环境。混合增强智能支撑平台重点建设支持大规模训练的异构实时计算引擎和新型计算集群，为复杂智能计算提供服务化、系统化平台和解决方案。自主无人系统支撑平台重点建设面向自主无人系统复杂环境下环境感知、自主协同控制、智能决策等人工智能共性核心技术的支撑系统，形成开放式、模块化、可重构的自主无人系统开发与试验环境。人工智能基础数据与安全检测平台重点建设面向人工智能的公共数据资源库、标准测试数据集、云服务平台等，形成人工智能算法与平台安全性测试评估的方法、技术、规范和工具集。促进各类通用软件和技术平台的开源开放。各类平台要按照军民深度融合的要求和相关规定，推进军民共享共用。

我国国务院在 2017 年印发的《新一代人工智能发展规划》就明确指出新一代人工智能的基础理论和关键共性技术体系，其核心之一即为建设支持知识推理、概率统计等机器学习范式的统一计算框架。

Association for the Advancement of Artificial Intelligence

AAAI 2025 PRESIDENTIAL PANEL ON THE
Future of AI Research

Table of Contents

6 Introduction
8 Panel Members & Contributors
10 AI Reasoning
14 AI Factuality & Trustworthiness
18 AI Agents
22 AI Evaluation
26 AI Ethics & Safety
32 Embodied AI
36 AI & Cognitive Science
40 Hardware & AI
44 AI for Social Good
48 AI & Sustainability
54 AI for Scientific Discovery
58 Artificial General Intelligence (AGI)
64 AI Perception vs. Reality
68 Diversity of AI Research Approaches
72 Research Beyond the AI Research Comm
76 Role of Academia
80 Geopolitical Aspects & Implications of A

AI Reasoning

The ability to reason has been a salient characteristic of human intelligence, and there is a critical need for verifiable reasoning in AI systems.

Main Takeaways

- Reasoning has always been seen as a core characteristic of human intelligence. Reasoning is used to derive new information from given base knowledge; this new information is guaranteed correct when sound formal reasoning is used, otherwise it is merely plausible.
- AI research has led to a range of automated reasoning techniques. These reasoning techniques have given rise to AI algorithms and systems, including SAT, SMT, and constraints solvers as well as probabilistic graphical models, all of which play a key role in critical real-world applications.
- While large pre-trained systems (such as LLMs) have made impressive advancements in their reasoning capabilities, more research is needed to guarantee correctness and depth of the reasoning performed by them; such guarantees are particularly important for autonomously operating AI agents.

Finally, 61.8% of survey participants estimated the minimal percentage of symbolic AI techniques required for reaching human-level reasoning to be at least 50% (with 24.8% estimating it at 75% or more, compared to 38.2% estimating it at 25% or below). What remains unclear is the degree

主席报告调研显示，61.8%的AI研究者认为AI要达到人类水平的推理，符号主义AI技术（逻辑推理、搜索、规划）应至少在AI系统内占比50%以上

美国人工智能协会 AAAI 2025 主席小组报告指出，基于大规模预训练模型的AI智能体（Agent）自主运行亟须更具正确性和深度保证的推理技术。

学术前沿

Luc De Raedt

INVITED TALKS IJCAI 2025 MONTREAL



KU Leuven

Neurosymbolic AI : combining Data and Knowledge

Location: Montreal | **Day:** August 21st | **Time:** 9:00:00

Abstract: The focus in AI today is very much on using just data for learning, but one should not learn what one already knows. The challenge therefore is to use the avaible knowledge to guide and constrain the learning, and to reason with the resulting models in a trustworthy manner. This requires the integration of symbolic AI with machine learning, which is the focus of neurosymbolic AI, often touted as the next wave in AI. I will argue that Neurosymbolic AI = Logic + Probability + Neural Networks. This will allow me to specify a high-level recipe for incorporating background knowledge into any neural network approach. The recipe starts from neural networks, interprets them at the symbol level by viewing them as "neural predicates" (or relations) and adding logical knowledge layers on top of them. The glue is provided by a probabilistic interpretation of the logic. Probability is interpreted broadly, it provides the quantitative differentiable component necessary to connect the symbolic and subsymbolic levels. Finally, I will show how the recipe and its ingredients can be used to develop the foundations of neurosymbolic AI.

Bio: Prof. Dr. Luc De Raedt is Director of Leuven.AI, the KU Leuven Institute for AI, full professor of Computer Science at KU Leuven, and guest professor at Örebro University (Sweden) at the Center for Applied Autonomous Sensor Systems in the Wallenberg AI, Autonomous Systems and Software Program. He is working on the integration of machine learning and machine reasoning techniques, also known under the term neurosymbolic AI. He has chaired the main European and International Machine Learning and Artificial Intelligence conferences (IJCAI, ECAI, ICML and ECMLPKDD) and is a fellow of EurAI, AAAI and ELLIS, and member of Royal Flemish Academy of Belgium. He received ERC Advanced Grants in 2015 and 2023.

神经符号AI-数据与知识的结合

Bernhard Schölkopf



Max Planck Institute for Intelligent Systems and ELLIS Institute Tübingen

From ML for science to causal digital twins

Location: Montreal | **Day:** August 19th | **Time:** 10:00:00

Abstract: Machine learning is being deployed across the sciences, changing the very way we perform scientific inference. This not only allows us to scrutinise datasets too large for human analysis, but also expands the domain of problems amenable to mathematical modeling toward greater complexity. However, standard machine learning has weaknesses when it comes to discovering causal relationships as opposed to potentially spurious correlations. I will discuss the associated failure modes as well as some success stories (with a particular focus on astronomy), including some that use methods of causal machine learning.

Bio: Bernhard Schölkopf studies machine learning and causal inference, with applications in fields ranging from astronomy to robotics. Trained in physics and mathematics, he earned a Ph.D. in computer science in 1997 and became a Max Planck director in 2001. A professor at ETH Zurich and a Fellow of the CIFAR program Learning in Machines and Brains, he has received the ACM-AAAI Allen Newell Award, the BBVA Foundation Frontiers of Knowledge Award, and the Royal Society Milner Award. He co-founded the MLSS series of Machine Learning Summer Schools, the ELLIS Society, and helped start the Journal of Machine Learning Research, an early milestone in open access and today the field's flagship journal.

从面向科学发现的机器学习到因果数字孪生

25年8月召开的国际人工智能联合会（IJCAI 25）两个主旨报告指出机器学习应当与经典的知识推理/因果推断相结合。

NEWS FEATURE | 25 November 2025

nature

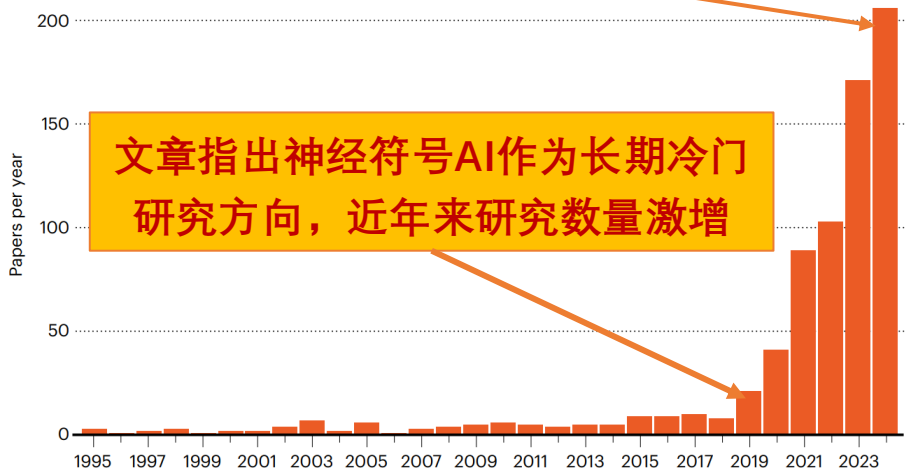
How good old-fashioned AI could spark the field's next revolution

Blending logic systems with the neural networks that power large language models is one of the hottest trends in artificial intelligence.

Nature于25年11月底发表专题报道，指出"老一套"的符号AI与神经网络的融合可能在AI领域掀起下一次范式革命。

GOING UP AND UP
The number of papers on neurosymbolic artificial intelligence has shot up over the past five years.

尽管收获学界关注，这仍然是一个年轻、体量较小且需要大量原创探索的研究领域



文章指出神经符号AI作为长期冷门研究方向，近年来研究数量激增

Nicola Jones, How good old-fashioned AI could spark the field's next revolution, *Nature*, 2025.11

符号主义 vs 联结主义



Deep Blue
IBM, 1997



AlphaGo
Deepmind, 2016

符号主义 vs 联结主义



《2001：太空漫游》
Stanley Kubrick, 1965

"Good afternoon, gentlemen. I am a HAL 9000 computer."

HAL 9000, **a Heuristically programmed ALgorithmic computer.**

顾问：Marvin Minsky
AI奠基者之一，1969图灵奖



《底特律：成为人类》
Quantum Dynamics, 2018

"I find it fascinating that we can create something that will **learn by itself, learning things even we can't understand.**"

David Cage, Quantum Dynamics

过去 → 现在

知识+推理

达特茅斯会议
J. McKarthy et al., 1955

逻辑理论家
H. Simon & A. Newell, 1950s
1975年图灵奖

专家系统
E. Feigenbaum, 1960s
1994年图灵奖

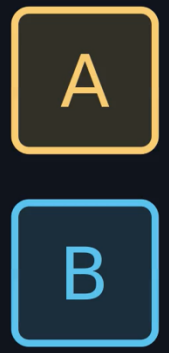
数据+学习

计算学习理论
L. Valiant, 1980s
2010年图灵奖

因果学习
J. Pearl, 1980s
2011年图灵奖

深度学习
Y. LeCun, G. Hinton & Y. Bengio, 2000s
2019年图灵奖

One world. Two ways to represent it.



A is above B. What should an intelligent system store?

Conceptual framing: Newell & Simon (1976); Hinton, Neural Networks for Machine Learning, Lecture 4

Symbols preserve identity and argument roles

$$\mathrm{Above}(A, B)$$

A symbolic expression has parts with explicit roles: a relation and its arguments.

Symbol structures: Newell & Simon (1976) • Symbolic learning: Muggleton & De Raedt (1994)

One rule. New objects. A checkable derivation.

$$\forall x, y, z: \quad \mathrm{Above}(x, y) \land \mathrm{Above}(y, z) \Rightarrow \mathrm{Above}(x, z)$$

A rule states how a relation composes. Match its variables to the facts.

Distributed means many-to-many

one-hot: one active coordinate



A one-hot code assigns each item its own coordinate. This is a localist encoding.

A geometry of similarity



Learning can place useful similarities into the geometry of a representation.

Learned representations: Bengio et al. (2003) • Toy two-dimensional illustration

Decades of rivalry—not a smooth convergence

Selected milestones · spacing not to scale

In 1969, Minsky and Papert analyzed limits of perceptrons—not of all possible neural networks.

Minsky & Papert; Rumelhart et al.; Fodor & Pylyshyn; Smolensky; Lake & Baroni; Woydt et al.

The same pieces can mean different things



Two scenes contain the same objects. Their relationship is reversed.

An explicit counterexample for unordered pooling—not a claim that neural networks cannot encode relations

Generated code is symbolic machinery

“Stack these blocks.”



An LLM can propose a program. The program has variables, control flow, and explicit operations.

Program execution: Gao et al., PAL (2023) • Robot programs: Liang et al., Code as Policies (2023)

Neuro-symbolic AI inherits the disagreement

A working hybrid can combine learned representations, explicit programs, and checks.

Engineering integration: PAL; Code as Policies • The systematicity debate remains a distinct scientific question

讲座安排

Classical Neuro-Symbolic Learning Paradigm	Wang-Zhou Dai, NJU	1. Taxonomy of Neuro-Symbolic Learning. 2. Abductive Learning: A brief Introduction.
Neuro-Symbolic Learning LLM Reasoning	Haoxuan Li, PKU	1. The status and challenges to the logical reasoning abilities of LLMs. 2. Evaluation for LLM logical reasoning abilities. 3. Improvement for LLM logical reasoning abilities with Neuro-Symbolic LearningMethods.
Q&A and Discussion	10min	Discussion
Neuro-Symbolic Learning for Multi-Modal Understanding and Generation	Yao-Xiang Ding, ZJU	1. How symbolic modules can improve the faithfulness, interpretability, and generalizability of multimodal understanding and generation models. 2. Symbolic intermediate representations: The key to bridge symbolic reasoning and neural understanding/generation. 3. State-of-the-art for neuro-symbolic multimodal understanding and generation.
Neuro-Symbolic Learning Agent	Lan-Zhe Guo, NJU	1. A brief introduction to neuro-symbolic agent. 2. Neuro-Symbolic Agentic Environment. 3. Neuro-Symbolic Learning for Long-Horizon Agentic Planning. 4. Neuro-Symbolic World Models.
Q&A and Discussion	15min	Interactive discussion and future directions.

A Brief History of Neuro-Symbolic Learning

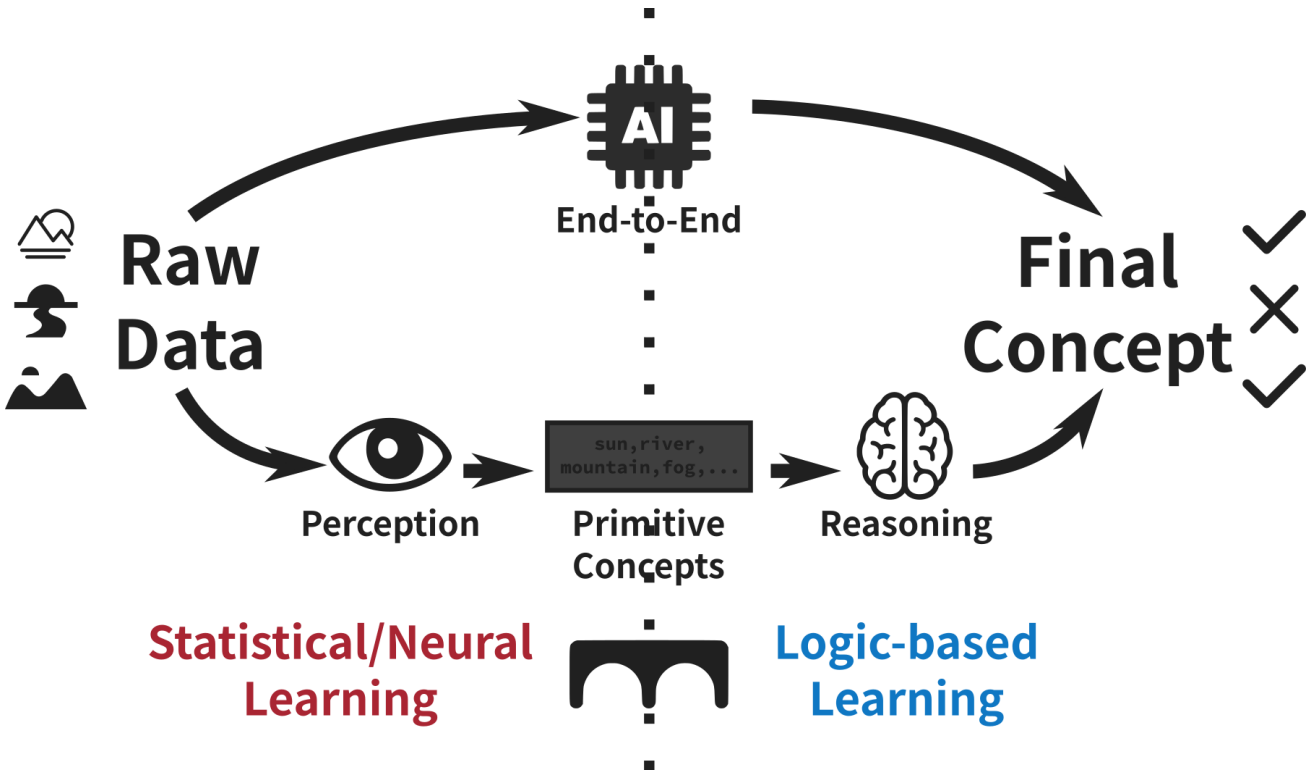
Wang-Zhou Dai

Nanjing University

CNCC 2026 | Tutorial of NeSy Learning

21 October 2026, Chengdu

Learning and reasoning



The interface connects perception to meaning.

De Raedt et al. (2026), pp. 99–103

Learning

Recognise patterns in raw data.

Reasoning

Apply explicit knowledge to a particular situation.

A shared origin in logic

1943

- Idealised neurons implement logical operations.
- Network structure determines the computation.
- Later neuro-symbolic systems add learning to this connection.

A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

WARREN S. MCCULLOCH and WALTER H. PITTS

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.

McCulloch & Pitts (1943)

Selected milestones

Period	Representative systems	Main question
1990s	KBANN, C-IL^{2}P	How can rules initialise a network?
2013–2019	NTN, TensorLog, NTP, ∂ILP, NLM	How can inference support gradient learning?
2014–2019	Memory networks, NTM, DNC, NALU	Which computational structure helps learning?
2018 onwards	DeepProbLog, ABL	How can learners and solvers cooperate?

Dates refer to the representative publications discussed in this tutorial.

The Manifesto's organising questions

Where does reasoning happen?	What passes across the interface?
1. Within neural computation	Rules shape representations, architecture or loss.
2. In a symbolic solver	Predictions enter the solver. Feedback may train the learner.
3. In a solver with neural guidance	Both components work with proofs, plans or other shared structures.

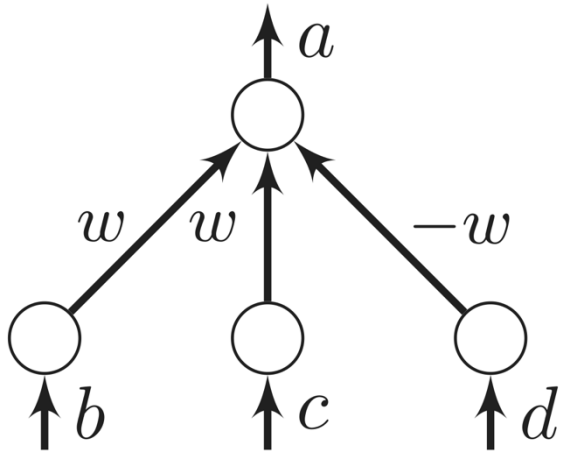
A second question: does reasoning improve learning?

De Raedt et al. (2026), pp. 103–105

Knowledge-based model construction

KBANN

- Compile propositional rules into connections and initial weights.
- Use examples to refine the network through backpropagation.
- Treat prior knowledge as an initial bias that learning can revise.



Rules become part of the learner.

Towell & Shavlik (1994), KBANN

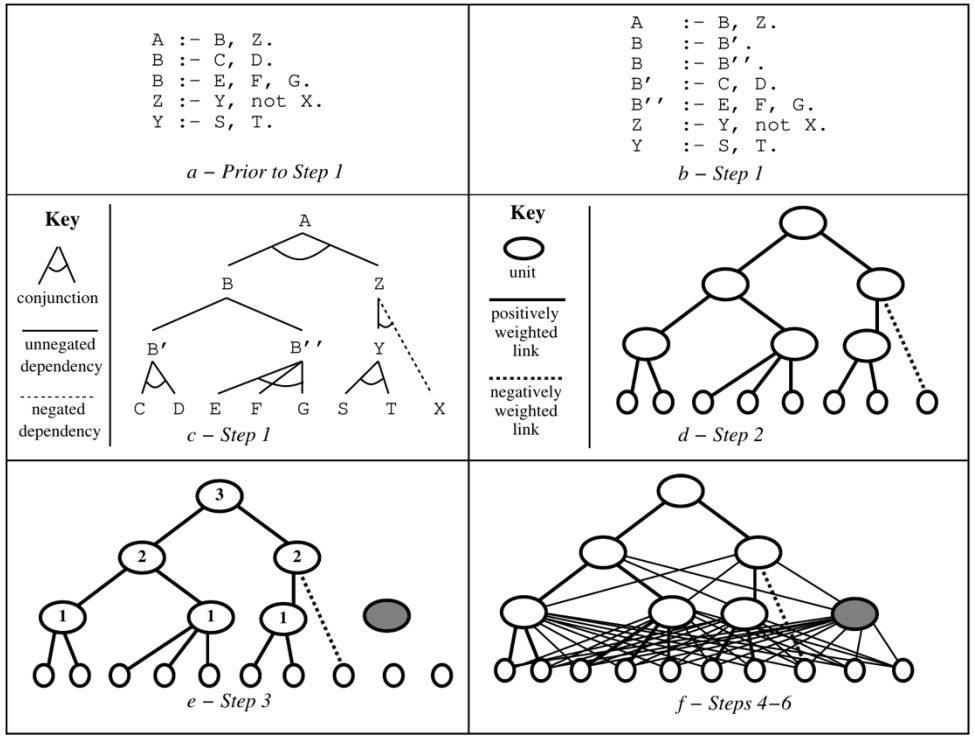
KBANN: rules become a network

Logical structure supplies the starting architecture.

1. Translate the rules

2. Add learning capacity

3. Refine with examples



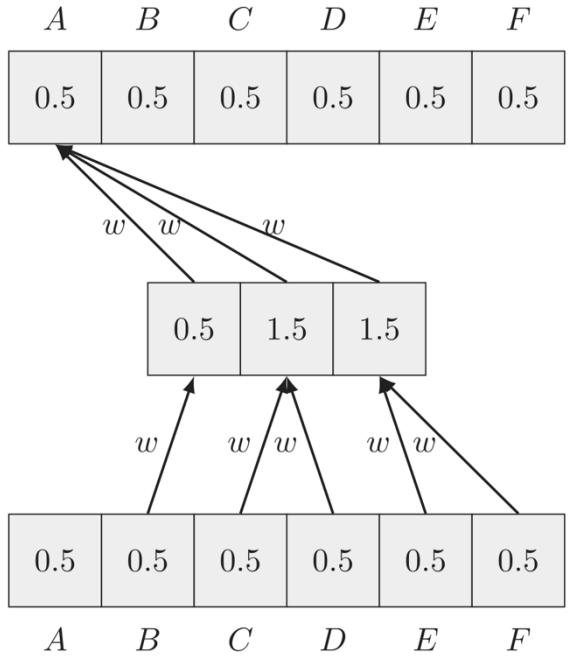
How much logical meaning survives learning?

Towell & Shavlik (1994)

Logic programs in neural networks

Logical inference as network dynamics

- Atoms correspond to units and clauses to intermediate computations.
- Repeated updates emulate consequence propagation.
- C-IL²P adds learning and the extraction of a revised logic program.



Hölldobler & Kalinke (1994) · d'Avila Garcez & Zaverucha (1999)

Relational knowledge and rule extraction

CILP++

- Construct bottom clauses from relational examples and background knowledge.
- Propositionalise the resulting features for neural learning.

Rule extraction

- Express a trained network's behaviour as symbolic rules.
- Assess fidelity to the network and readability for people.

The translation can run in both directions.

França et al. (2014) · Andrews et al. (1995) · d'Avila Garcez et al. (2001) · Zhou et al. (2003)

Representing symbols numerically

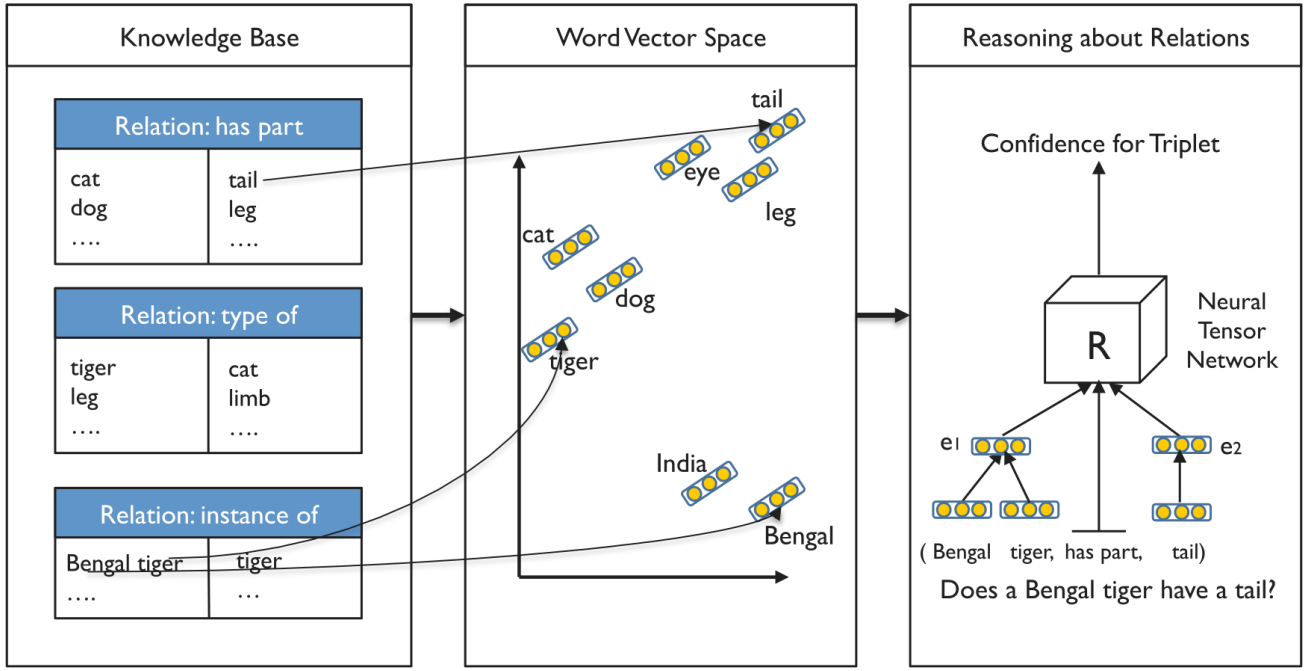
Representation	Meaning of a dimension	Typical advantage
Propositional features	An explicit relational feature or clause	Features can remain interpretable.
One-hot symbols	A particular entity or symbol	Identity stays explicit.
Distributed embeddings	A learned latent coordinate	Similar symbols can share statistical strength.

The representation influences what "reasoning" can mean.

Lavrač et al. (2020), Propositionalization and Embeddings: Two Sides of the Same Coin

Neural Tensor Networks

Learn how plausible a knowledge-base triple is.



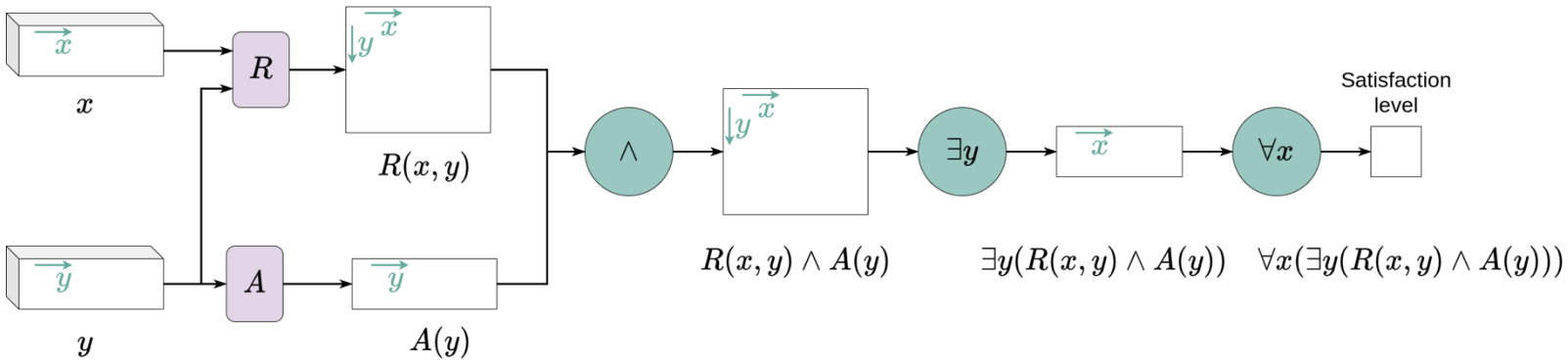
A high score for a relation is a prediction, rather than a proof.

Socher et al. (2013)

Logic Tensor Networks

- Ground entities and predicates in tensors.
- Optimise a differentiable measure of rule satisfaction.

e.g. if R denotes the predicate for $friends$, and A denotes the predicate for $Italian$,
the following computational graph translates the English sentence "everybody has a friend that is Italian".



The choice of fuzzy semantics affects both inference and optimisation.

Badreddine et al. (2022)

TensorLog

Compile relational inference into sparse linear algebra.

$$grandparent(X, Z) \leftarrow parent(X, Y) \wedge parent(Y, Z)$$

$$s(a) = e_{\mathrm{a}}{}^{\mathrm{T}} M_{\mathrm{p}} M_{\mathrm{p}}$$

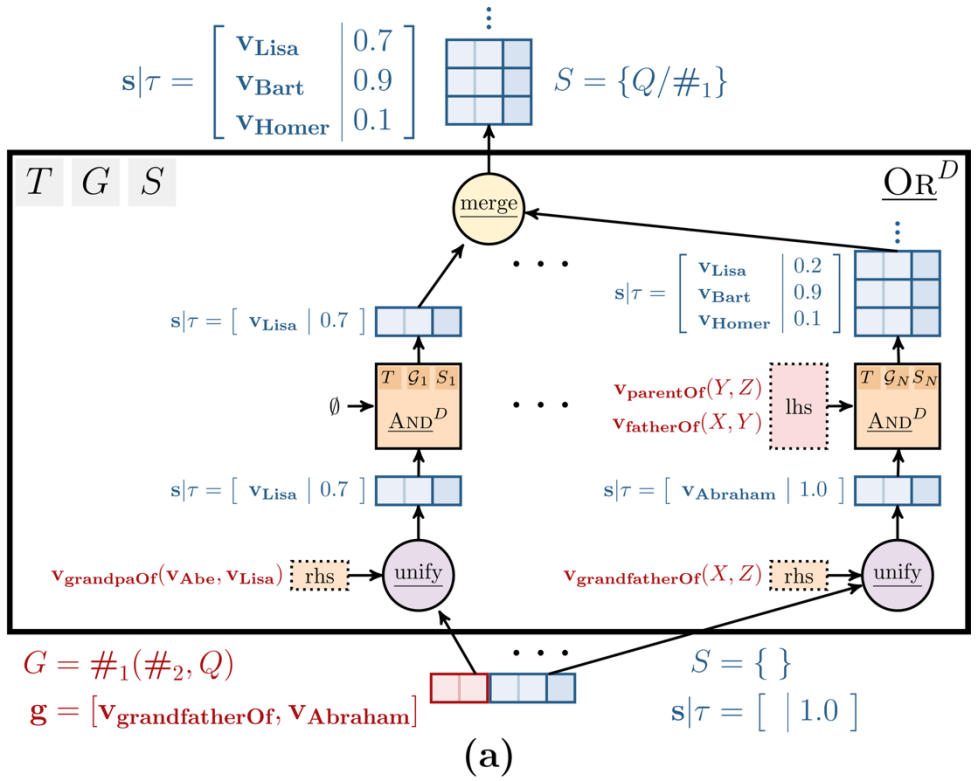
M_{p} encodes parent relations. A one-hot vector selects a row.
Matrix composition follows a relational chain.

Symbol identity remains explicit, while the computation admits gradients.

Cohen (2016)

Neural Theorem Provers

- Retain the structure of backward chaining.
- Replace exact symbol matching with similarity between embeddings.
- Learn embeddings and rule parameters through a proof score.



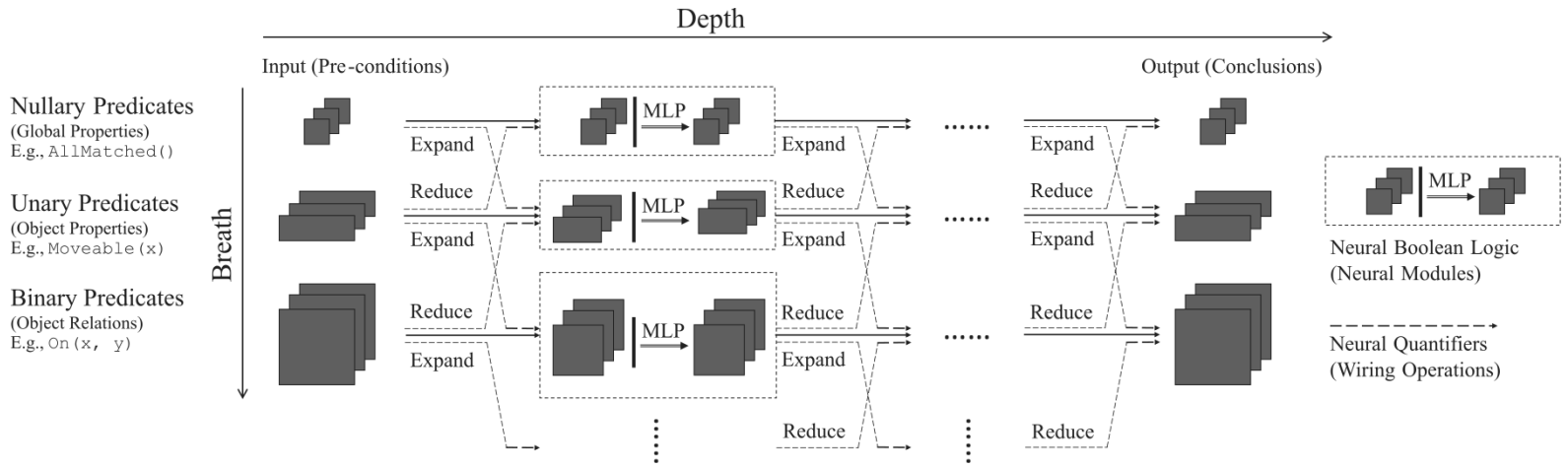
The proof structure is symbolic.
The matching operation is soft.

Rocktäschel & Riedel (2017)

Learning relational computations

∂ILP learns weights over candidate logical clauses.

NLM builds relational computation around predicate arity.



Structural bias can help a model generalise to larger relational problems.

Evans & Grefenstette (2018), ∂ILP · Dong et al. (2019), Neural Logic Machines

What survives the translation?

Route	What knowledge contributes	What needs checking
Rule-initialised networks	Architecture and initial parameters	Whether learning preserves the intended rules
Embeddings and soft logic	Similarity or differentiable constraints	Meaning of scores and truth degrees
Differentiable inference	A structured computation over relations	Language restrictions and inference semantics
Special neural modules	Memory or algorithmic operations	Robustness and generalisation

A comparison of mechanisms, rather than a ranking of systems.

Keeping a symbolic solver

1. Perception or translation

Map an image or a natural-language problem into symbols.

2. Formal inference

Apply a solver to the explicit model and symbolic input.

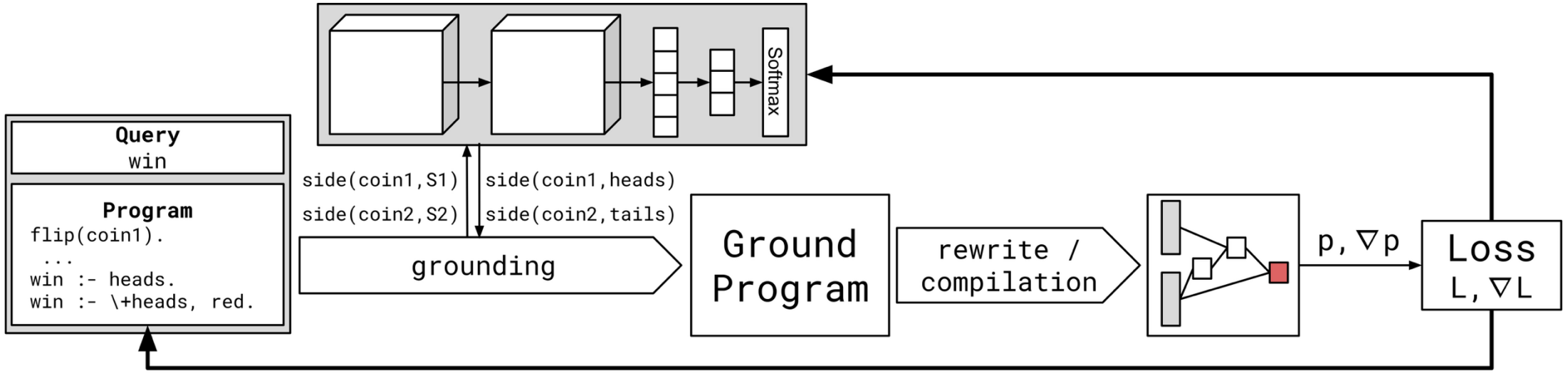
3. A learning interface

Use reasoning outcomes to improve perception or translation.

De Raedt et al. (2026), category A

DeepProbLog

Neural predicates connect perception to probabilistic logic.

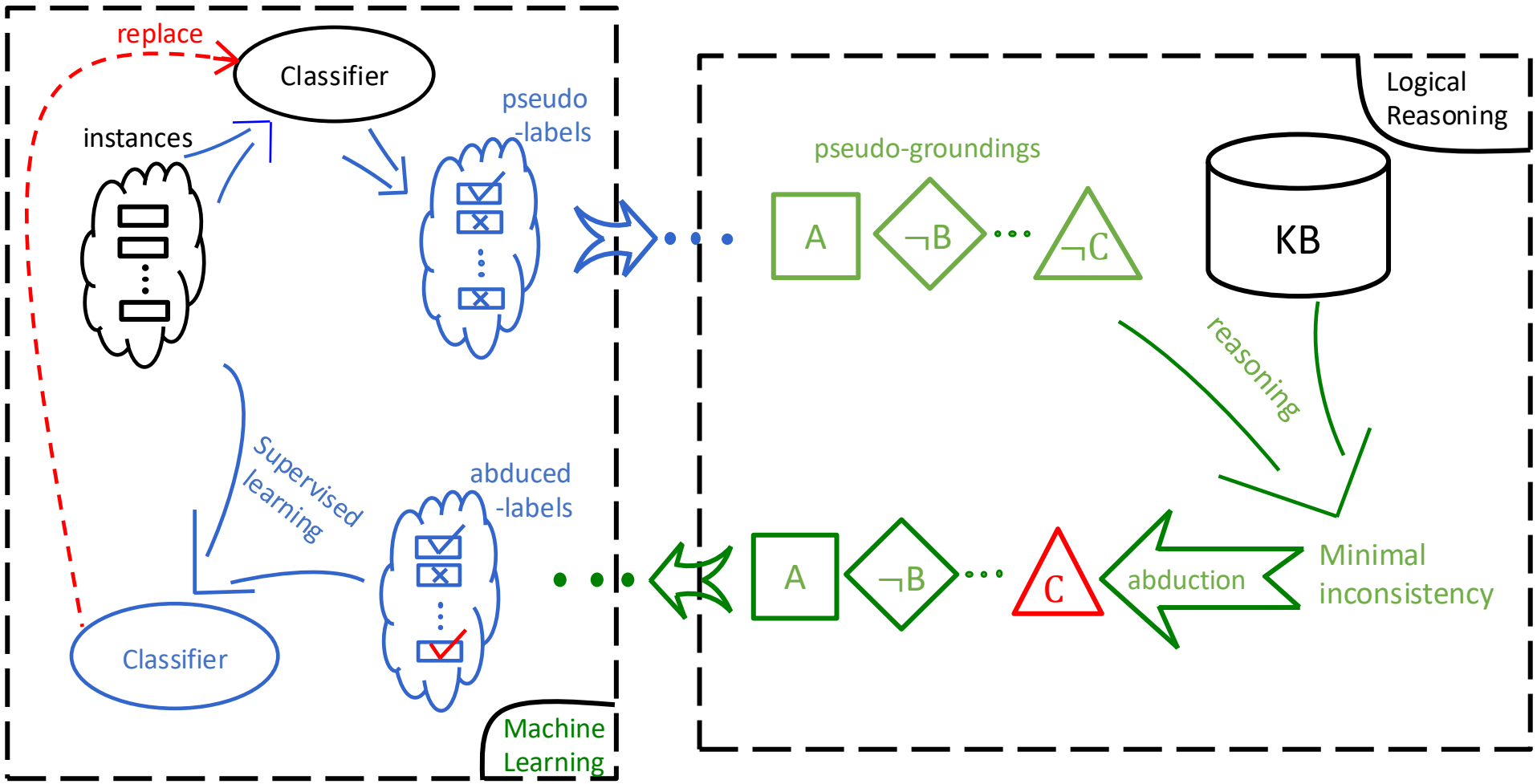


(a) The learning pipeline.

The task loss trains perception through probabilistic inference.

Manhaeve et al. (2018, 2021)

Abductive Learning (ABL)



Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. Science China Information Sciences, 2019.
W.-Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou. Bridging machine learning and logical reasoning by abductive learning. NeurIPS, 2019.

Learning from the answer

$$digit(x_1) + digit(x_2) = 5$$

Observed: two digit images and their sum, 5.

$$P(sum = 5 \mid x_1, x_2) = \sum p_\theta(a \mid x_1)\, p_\theta(b \mid x_2)$$

The sum ranges over digit pairs with a + b = 5.

The answer trains perception through the reasoning computation.

DeepProbLog: an illustrative addition task

Two ways to return a learning signal

Interface	Solver contribution	Learner update
Differentiable reasoning	Compute an answer score and its sensitivity to neural outputs.	Backpropagate a task-level loss.
Abductive feedback	Infer or revise symbols that explain the observed target.	Train on the inferred supervision.

Bidirectional learning is the key idea.
The feedback need not be a derivative through every solver operation.

De Raedt et al. (2026), p. 103

A learner and a solver improve one another

A Learning and Reasoning Manifesto

Luc De Raedt[1,4(✉)], Fredrik Heintz[2], Kristian Kersting[3], and Giuseppe Marra[1]

[1] KU Leuven, Leuven, Belgium
{luc.deraedt,giuseppe.marra}@kuleuven.be
[2] Linköping University, Linköping, Sweden
fredrik.heintz@liu.se
[3] TU Darmstadt, hessian.AI, and DFKI, Darmstadt, Germany
kersting@cs.tu-darmstadt.de
[4] Örebro University, Örebro, Sweden

Only when Artificial Intelligence can rival Arno Intelligence will it truly succeed. But as Arno manifestly learns and reasons, so too must Artificial Intelligence. This manifesto is dedicated to that pursuit, and to Arno — who is inspiringly intelligent.

Abstract. This manifesto provides a perspective on artificial intelligence in which the ability to both learn and reason are central. It then examines the state-the-of-the-art in artificial intelligence while focusing on foundation models and neurosymbolic artificial intelligence.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2026
M. van Leeuwen and J. Vreeken (Eds.): Arno Siebes Festschrift, LNCS 16067, pp. 97–108, 2026.
https://doi.org/10.1007/978-3-032-03028-3_6



Luc De Raedt

Elected Member of the Royal Flemish Academy of Belgium

EurAI Fellow, AAAI Fellow, ELLIS Fellow, and former IJCAI Trustee

ECML Chair and founder

IJCAI 2022 Program Chair

In his 2026 paper as first and corresponding author, abductive learning is commented as:

standard gradient-descent techniques. More broadly, it implies each component can compute a learning signal to improve the other, perhaps by synthesizing a training example as seen in abduction-based systems [45]. This leads to a tight and principled integration between the two models, and the solver and the neural network. This is the ultimate type of neurosymbolic AI system as it has the perception abilities of neural nets, the reasoning abilities of logic, and elegantly bridges the gap between the two. It is however also the most challenging type of system to develop and numerous research questions are still open.

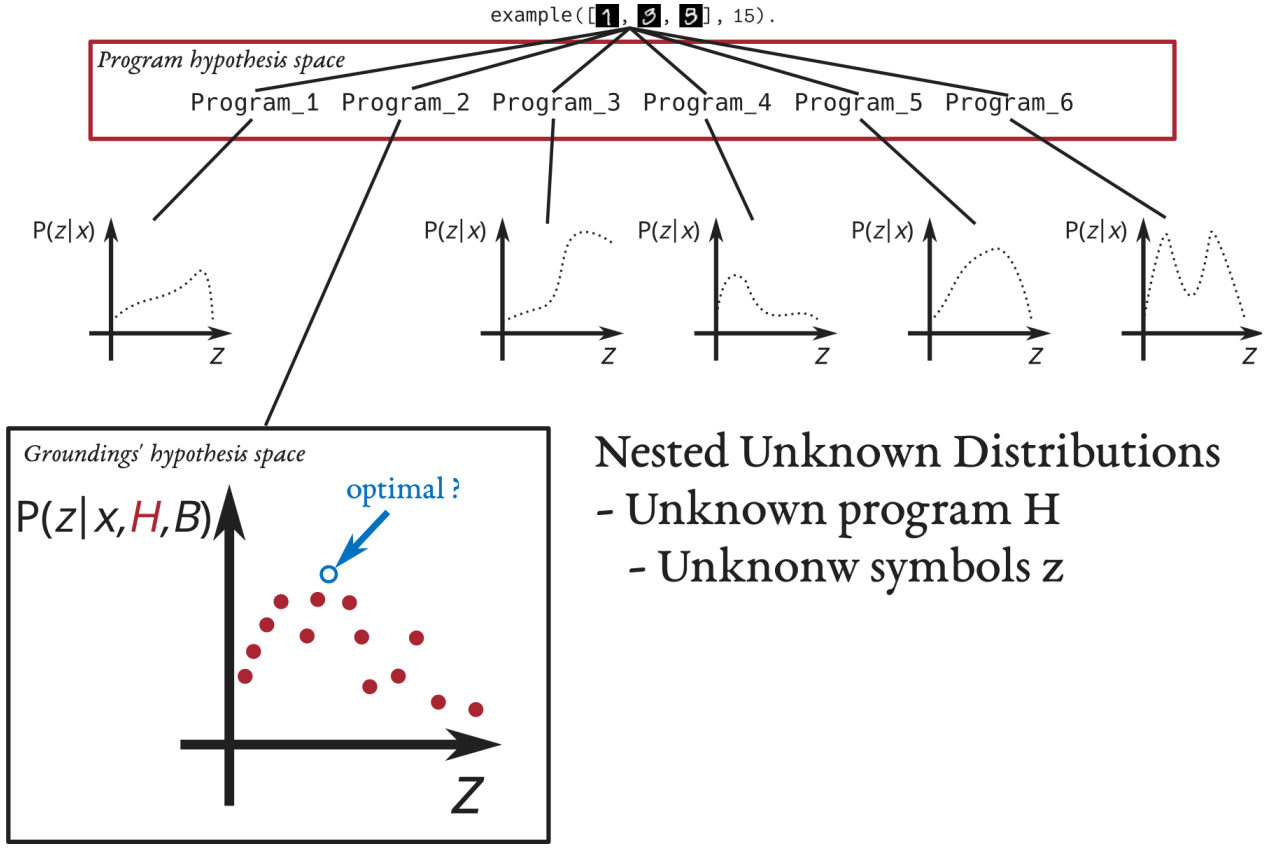
45. Hu, W.C., Dai, W.Z., Jiang, Y., Zhou, Z.H.: Efficient rectification of neuro-symbolic reasoning inconsistencies by abductive reflection. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, no. 16, pp. 17333–17341 (2025)

“Most challenging and open”

- The primitive symbols may be uncertain.
- The knowledge or program may also be incomplete.

Abduction revises the grounding.
Induction revises the knowledge.

The two searches constrain one another.



Dai & Muggleton (2021)

The historical change in emphasis

Emphasis	Representative examples	Central contribution
Put knowledge into a learner	KBANN, C-IL^{2}P, CILP++	Rules shape how a model learns.
Make structured computation trainable	NTN, LTN, TensorLog, NTP, NLM	Neural learning gains relational structure.
Give learning algorithmic structure	Memory networks, NTM, NALU	Useful operations become architectural biases.
Make learners and solvers cooperate	DeepProbLog, ABL	Reasoning returns a learning signal.

A synthesis of the lecture's four streams using the Manifesto's interface perspective.

A Gentle Introduction to Abductive Learning

Wang-Zhou Dai, Yu-Xuan Huang, En-Hao Gao, Wen-Chao Hu

Nanjing University

CNCC 2026 | Tutorial of NeSy Learning

October 2026, Chengdu

Outline

Part I. Introduction

- The Abductive Learning (ABL) Framework

Part II. Examples of Application

- Semi-Supervised Abductive Learning and Its Application to Theft Judicial Sentencing

Part III. Method & Theory

- Fast Abductive Learning by Similarity-based Consistency Optimization
- Enabling Knowledge Refinement upon New Concepts in Abductive Learning
- Abductive Learning with Ground Knowledge Base
- Efficient Rectification of Neuro-Symbolic Reasoning Inconsistencies by Abductive Reflection
- Deciphering Raw Data in Neuro-Symbolic Learning with Provable Guarantees

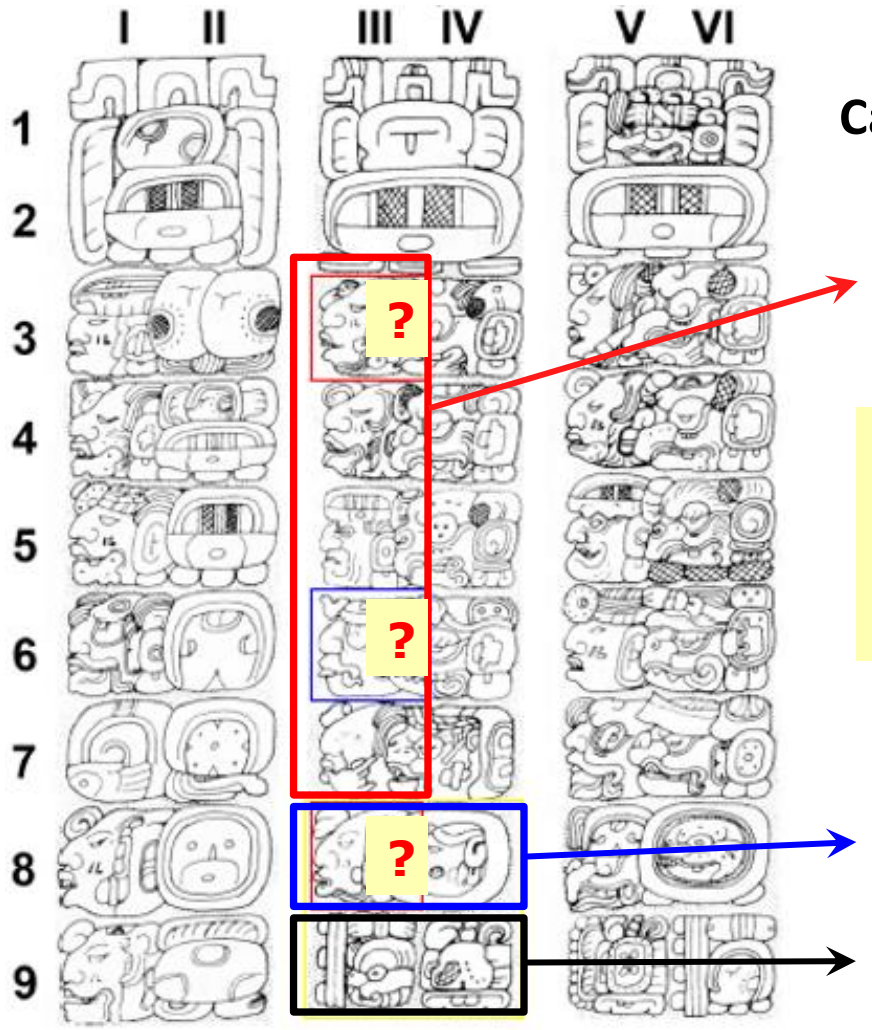
Part IV. Tool

- ABLkit: A Python Toolkit for Abductive Learning

Part I. Introduction

- The Abductive Learning Framework

An Example: Deciphering the Mayan Calendar System



Three Mayan Calendar systems

Long Count date :
?. 18 . 5 . ? . 0

They must be **consistent**
→
What are the **?**'s

Tzolk'in date : **? Ahau**

Haab'date : **13 Mac**

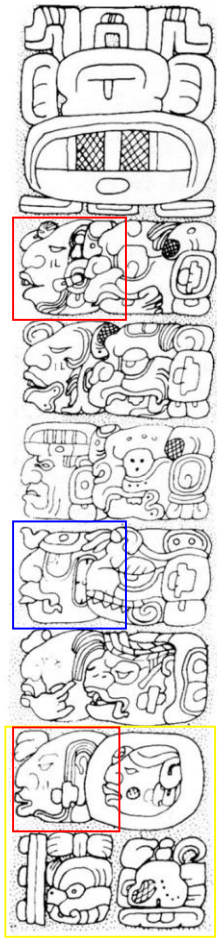


The Interface Groundings

Image → Numbers → *Equation*

Perception → Grounding → *Reasoning*

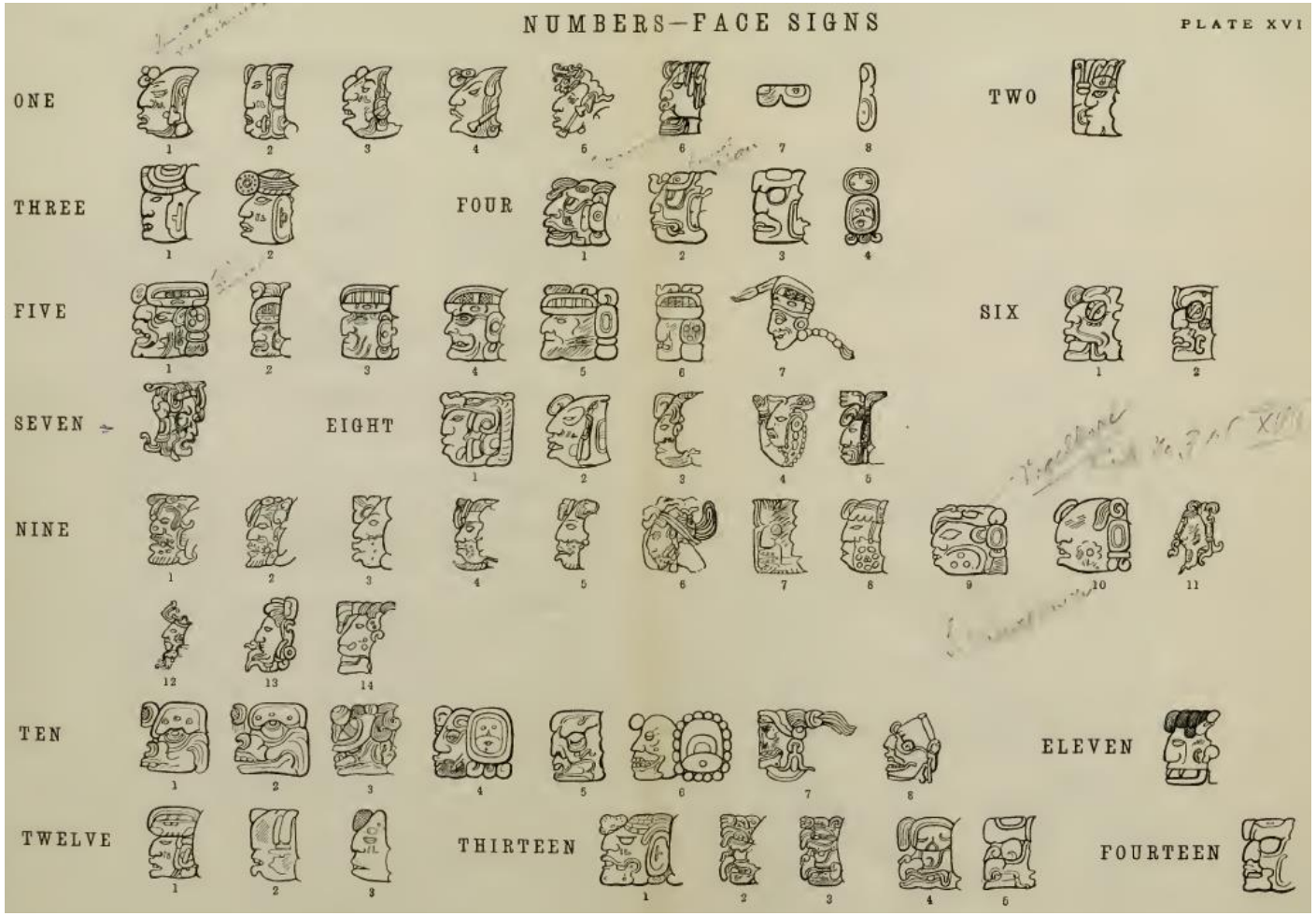
Cracking the glyphs



Column 1.	Column 2.
9.18.5.0.0., 9 Ahau 13 Mac.	4 Ahau 13 Ceh (18)
9.18.5.1.0., 9 Ahau 13 Mac.	11 Ahau 13 Mac (18)
9.18.5.2.0., 9 Ahau 13 Mac.	5 Ahau 13 Kankin (18)
9.18.5.3.0., 9 Ahau 13 Mac.	12 Ahau 13 Muan (18)
9.18.5.4.0., 9 Ahau 13 Mac.	6 Ahau 13 Pax (18)
9.18.5.5.0., 9 Ahau 13 Mac.	13 Ahau 13 Kayab (18)
8.18.5.8.0., 8 Ahau 13 Mac	9 Ahau 3 Zac (40)
8.18.5.9.0., 8 Ahau 13 Mac	3 Ahau 3 Ceh (40)
8.18.5.10.0., 8 Ahau 13 Mac	10 Ahau 3 Mac (40)
8.18.5.11.0., 8 Ahau 13 Mac	4 Ahau 3 Kankin (40)
1.18.5.2.0., 1 Ahau 13 Mac	13 Ahau 13 Zac (34)
1.18.5.3.0., 1 Ahau 13 Mac	7 Ahau 13 Ceh (34)
1.18.5.4.0., 1 Ahau 13 Mac	1 Ahau 13 Mac (34)
1.18.5.5.0., 1 Ahau 13 Mac	8 Ahau 13 Kankin (34)
1.18.5.6.0., 1 Ahau 13 Mac	2 Ahau 13 Muan (34)
1.18.5.7.0., 1 Ahau 13 Mac	9 Ahau 13 Pax (34)

Bowditch, Charles Pickering. The numeration, calendar systems and astronomical knowledge of the Mayas. 1910.

The Cracked Ground Mapping



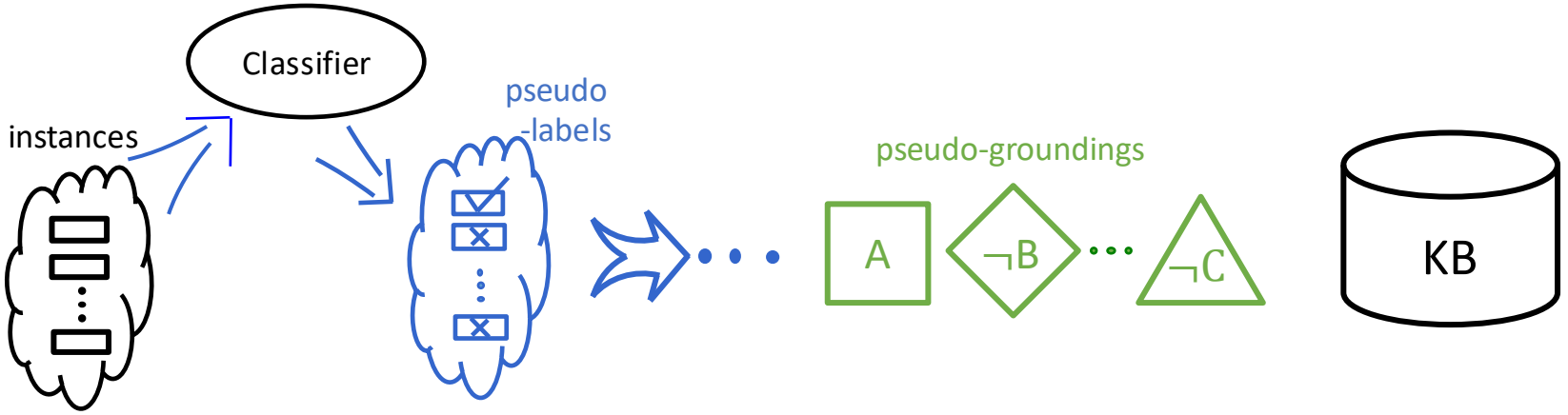
Abductive Reasoning

Perception	Optimization	Reasoning
Glyphs (image) $\mapsto$ Numbers (symbol)		Creation + LC = Tz, Hb Calculation rules: 20-based

"Equations should be correct"

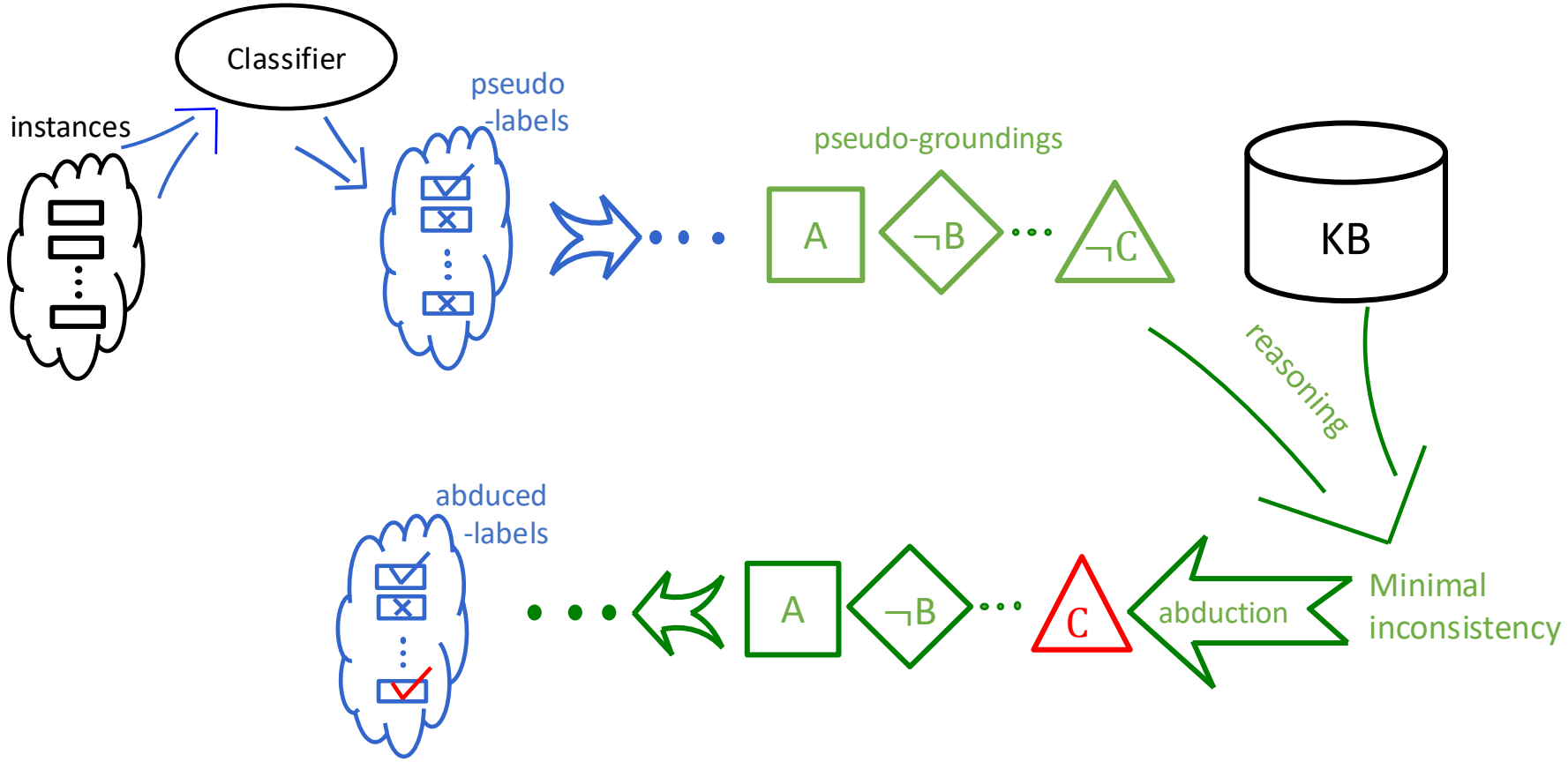
Starts from incomplete observations, proceeds to the likeliest possible explanation for them

Abductive Learning (ABL)



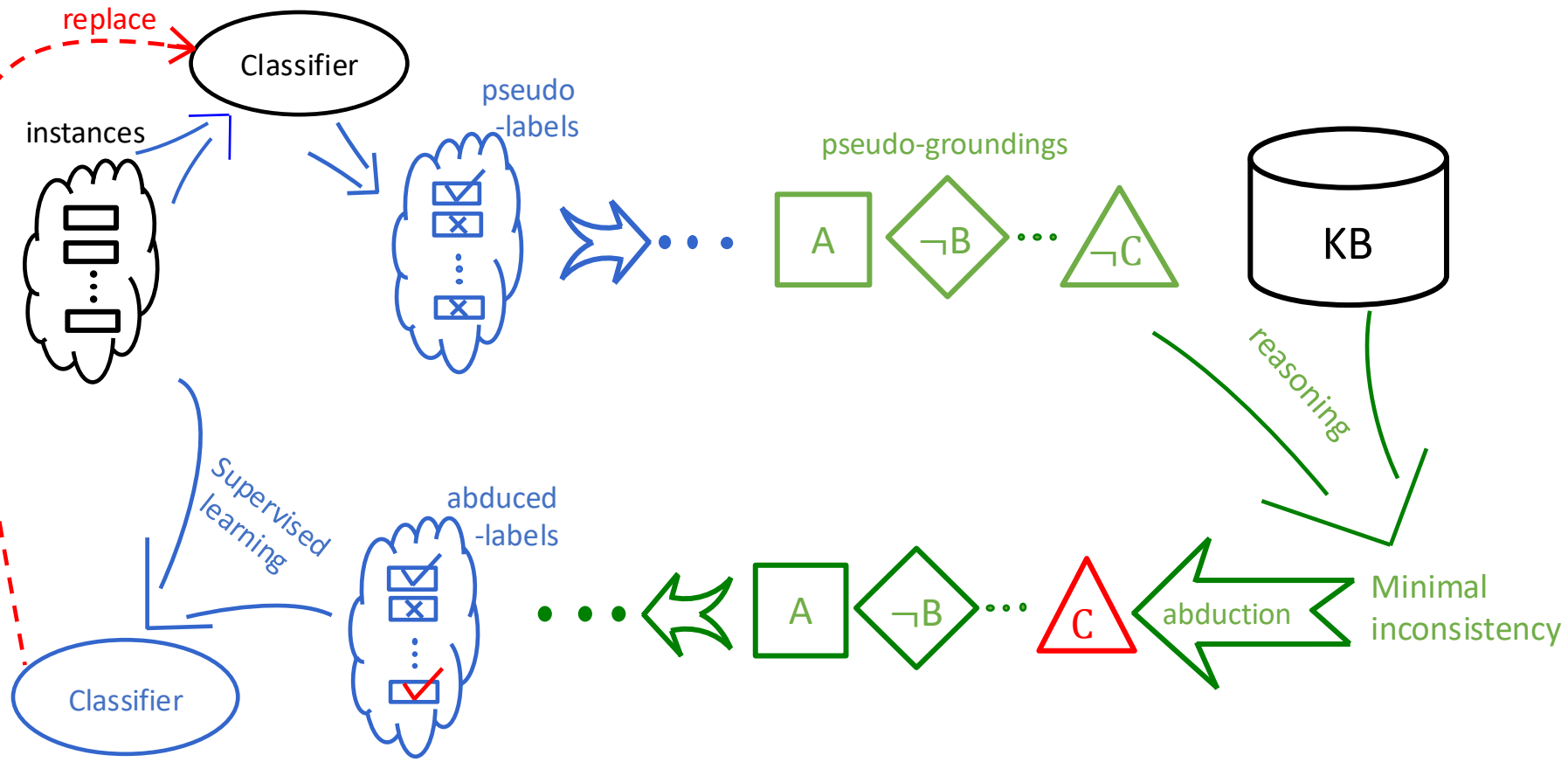
Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. Science China Information Sciences, 2019.
W.-Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou. Bridging machine learning and logical reasoning by abductive learning. NeurIPS, 2019.

Abductive Learning (ABL)



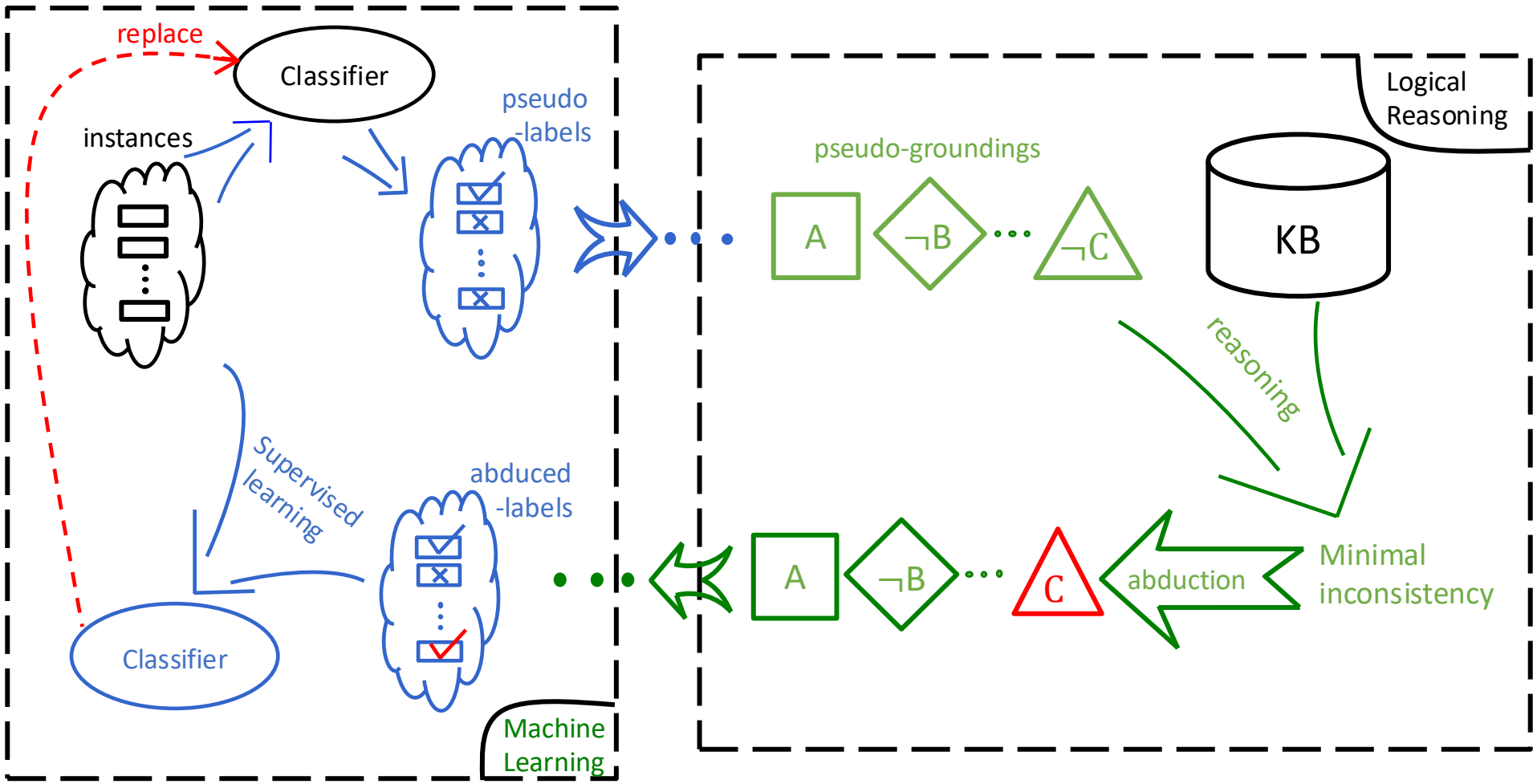
Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. Science China Information Sciences, 2019.
W.-Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou. Bridging machine learning and logical reasoning by abductive learning. NeurIPS, 2019.

Abductive Learning (ABL)



Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. Science China Information Sciences, 2019.
W.-Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou. Bridging machine learning and logical reasoning by abductive learning. NeurIPS, 2019.

Abductive Learning (ABL)



Zhi-Hua Zhou. Abductive learning: Towards bridging machine learning and logical reasoning. Science China Information Sciences, 2019.
W.-Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou. Bridging machine learning and logical reasoning by abductive learning. NeurIPS, 2019.

Abductive Learning (ABL)

Training examples: $\langle x, y \rangle$

$$\text{sum}([\text{}], 15).$$

1. Machine learning (e.g., neural net):

$$z = f(x; \theta) = \text{Sigmoid}(P_\theta(z|x))$$

   - Learns a perception model mapping raw data (x) $\longrightarrow$ logic facts (z)

2. Logical Reasoning (e.g., logic program):

$$B \cup z \vDash \text{y}$$

   - **Inference (Deduction)**: $x \overset{f}{\Rightarrow} z \overset{B}{\Rightarrow} y$
   - Learning (Abduction): $\text{y} \overset{B}{\Rightarrow} z \overset{x}{\Rightarrow} f$

3. **Optimization**: maximizes the consistency of z and f w.r.t. $\langle x, y \rangle$ and B

W.-Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou. Bridging machine learning and logical reasoning by abductive learning. NeurIPS, 2019.

Abductive Learning (ABL)

Intuitively, the abduction of the pseudo-labels provides a gradient-like feedback to the machine learning model.

Feed forward $x \overset{f_{\theta_i}}{\Longrightarrow} \hat{z} \overset{B}{\Rightarrow} \hat{y}$

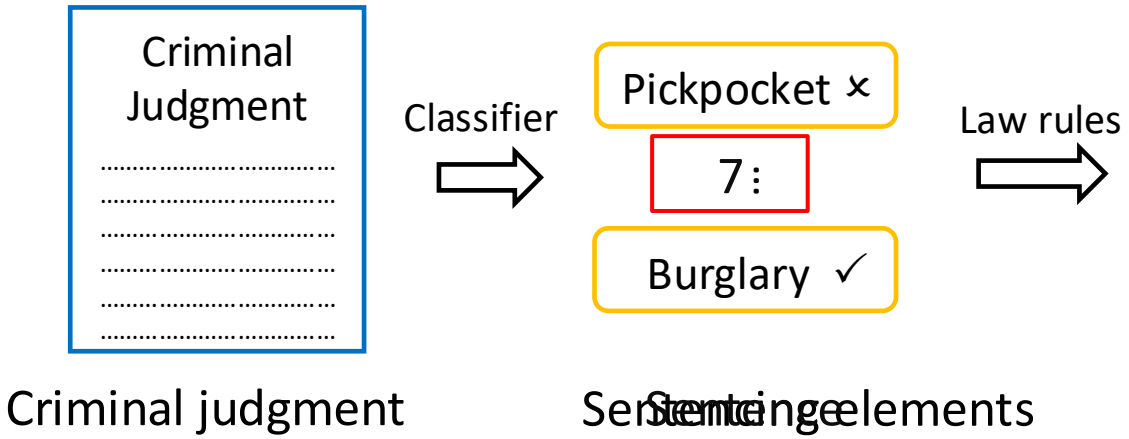
Feed back $\text{dist}(y^*, \hat{y}) \overset{B}{\Rightarrow} \hat{z}' \overset{x}{\Rightarrow} f_{\theta_{i+1}}$

Part II. Examples of Application

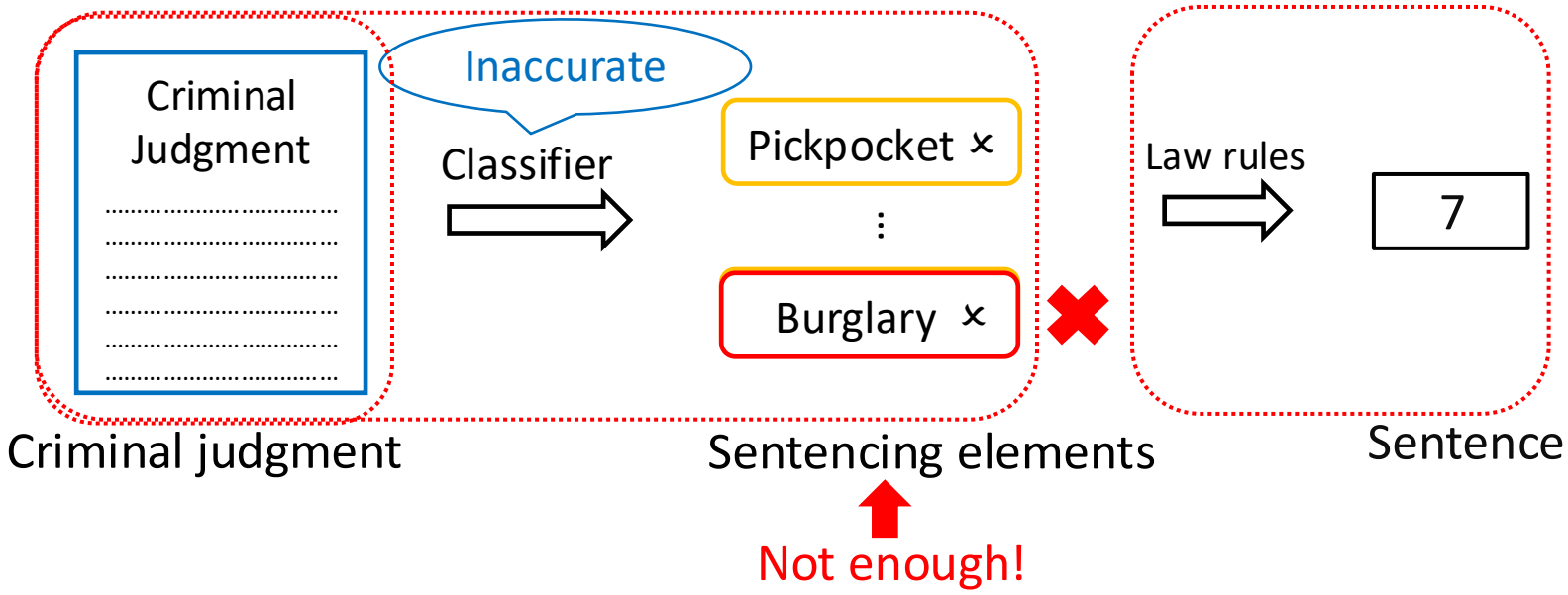
- Semi-Supervised Abductive Learning and Its Application to Theft Judicial Sentencing

Theft Judicial Sentencing

- Check the quality of sentences made by judges in local court
- Extract sentencing elements + Predict judicial sentence



Challenge



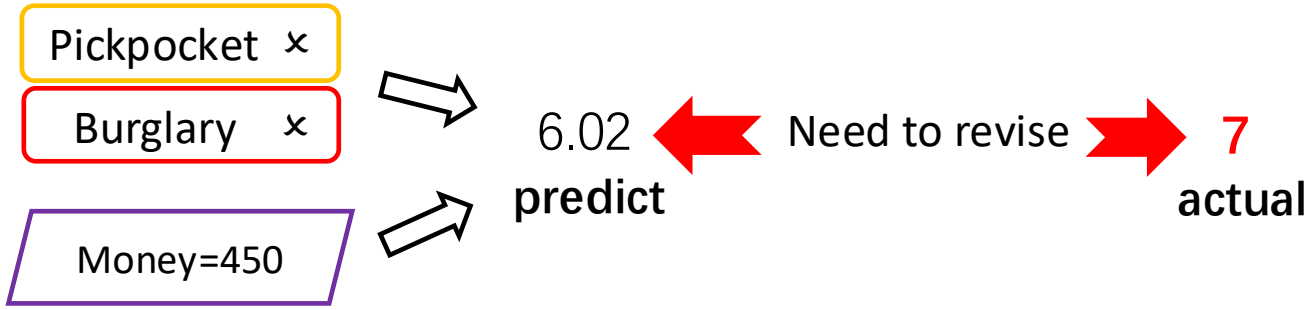
- Few labeled data: time-consuming, needs legal professionals

→ Inaccurate classifier (E.g., the classifier may misclassify "Burglary" to "No-Burglary")

Two information sources

- A lot of **unlabeled data**: cheap, easy to collect
- Domain **knowledge**: symbolic law rules

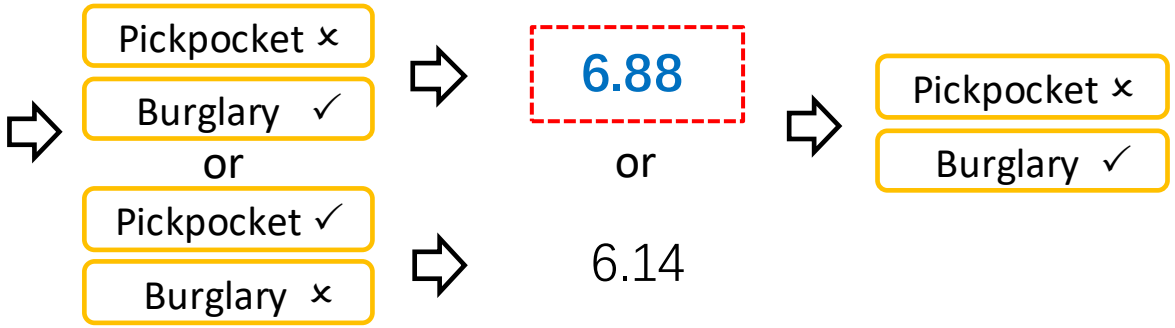
Abduction example



Formalized Law Rules

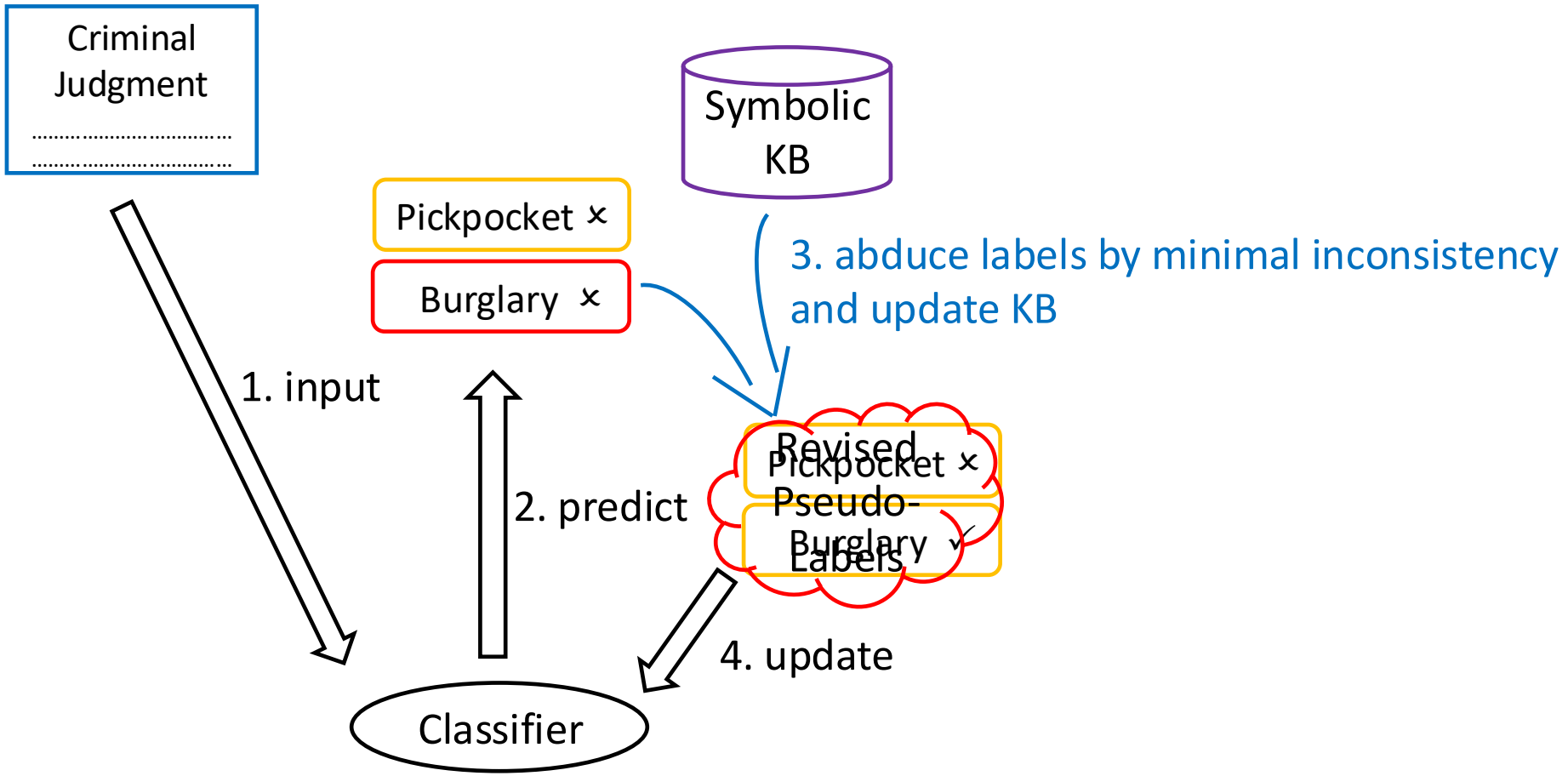
$$penalty(X,Y) \wedge leq(Y,36) \leftarrow (money(X,m) \wedge leq(m,1000)) \vee element(X,pickpocket) \vee element(X,burglary)$$

$$False \leftarrow element(X,pickpocket) \wedge element(X,burglary)$$



1. Generate hypotheses 2. Minimal inconsistency 3. Final result

Abduction example



Results

The classifier

TABLE I
F1-SCORE OF BERT, AND MAE & MSE OF PREDICTED SENTENCE.

Method	F1	MAE	MSE
BERT-10	0.811±0.010	0.867±0.032	1.204±0.123
PL-10	0.814±0.006	0.862±0.035	1.155±0.107
Tri-10	0.812±0.016	0.840±0.066	1.155±0.107
ABL-10	0.824±0.014	0.873±0.102	1.376±0.217
SS-ABL-10	**0.862±0.005**	**0.824±0.017**	**1.146±0.049**
BERT-50	0.857±0.006	0.830±0.020	1.065±0.044
PL-50	0.858±0.010	0.832±0.035	1.045±0.154
Tri-50	0.861±0.007	0.810±0.021	0.994±0.046
ABL-50	0.860±0.003	0.842±0.029	1.082±0.110
SS-ABL-50	**0.865±0.007**	**0.788±0.027**	**0.959±0.091**
BERT-100	0.863±0.003	0.821±0.014	1.011±0.031
ABL-100	0.867±0.008	0.822±0.030	1.007±0.093

- The performance of SS-ABL-10 and SS-ABL-50 are **superior** to other methods
- When the label rate is **low**, the gain is much **greater**
- SS-ABL-10 is even **superior** to BERT-50 (more labeled data but no unlabeled data)

Results

The predicted sentence

TABLE I
F1-SCORE OF BERT, AND MAE & MSE OF PREDICTED SENTENCE.

Method	F1	MAE	MSE
BERT-10	0.811±0.010	0.867±0.032	1.204±0.123
PL-10	0.814±0.006	0.862±0.035	1.155±0.107
Tri-10	0.812±0.016	0.840±0.066	1.155±0.107
ABL-10	0.824±0.014	0.873±0.102	1.376±0.217
SS-ABL-10	**0.862±0.005**	**0.824±0.017**	**1.146±0.049**
BERT-50	0.857±0.006	0.830±0.020	1.065±0.044
PL-50	0.858±0.010	0.832±0.035	1.045±0.154
Tri-50	0.861±0.007	0.810±0.021	0.994±0.046
ABL-50	0.860±0.003	0.842±0.029	1.082±0.110
SS-ABL-50	**0.865±0.007**	**0.788±0.027**	**0.959±0.091**
BERT-100	0.863±0.003	0.821±0.014	1.011±0.031
ABL-100	0.867±0.008	0.822±0.030	1.007±0.093

- Mean Absolute Error (MAE) and Mean Square Error (MSE) between actual and predicted value
- The SS-ABL-based approaches **significantly outperform** other methods

Results

TABLE IV
SENTENCE RULES LEARNED BY SS-ABL.

Benchmark sentence=0.64m+6 (m:the amount of money involved in thousands)	
Heavier or lesser punishment rules	Heavier or lesser punishment rules
If *burglary* then benchmark+6%	If *confession* then benchmark-14%
If *forgiveness* then benchmark-15%	If *juvenile* then benchmark-15%
If *no loss* then benchmark-7%	If *pickpocket* then benchmark+15%
If *surrender* then benchmark-10%	If *recidivism* then benchmark+27%

- Explain **how knowledge base reasons** to produce final sentence prediction
- **Highly interpretable,** can be easily understood and examined by human, provide explanatory assistance to judge sentencing

Part III. Method & Theory

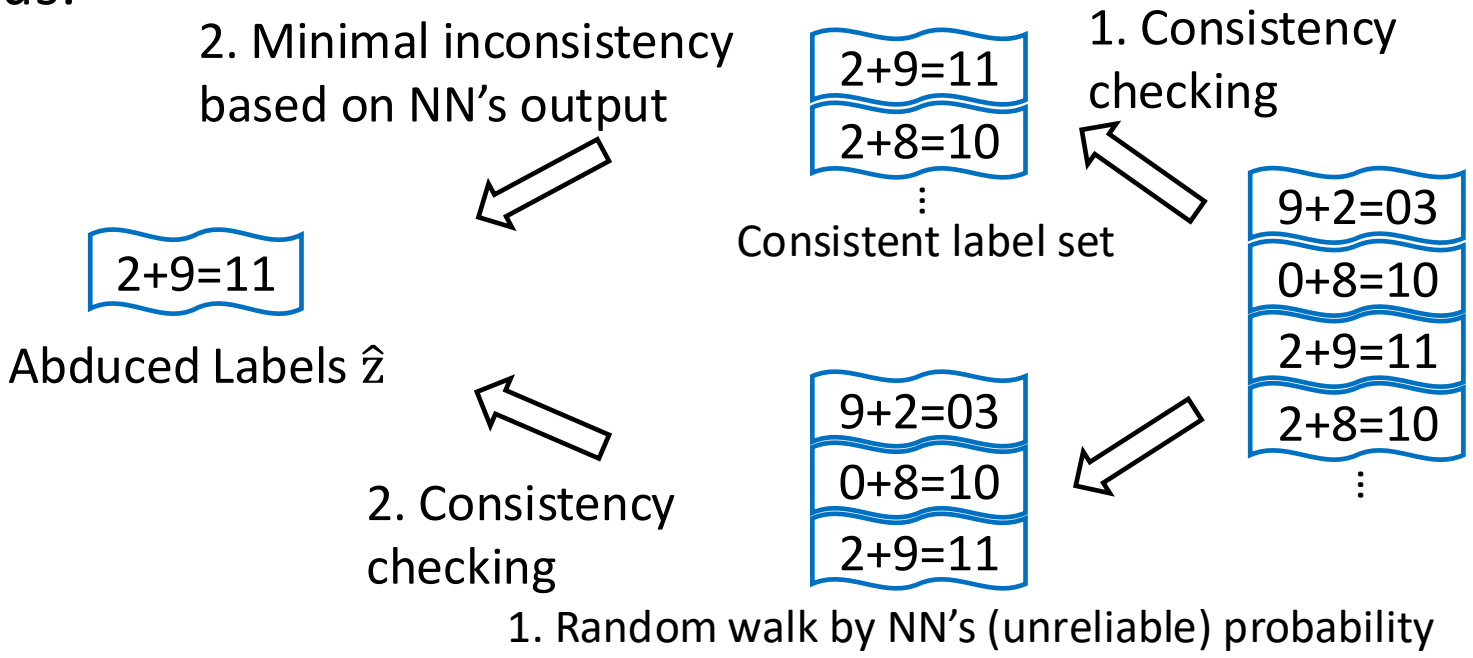
- Fast Abductive Learning by Similarity-based Consistency Optimization
- Efficient Rectification of Neuro-Symbolic Reasoning Inconsistencies by Abductive Reflection
- Enabling Abductive Learning to Exploit Knowledge Graph
- Abductive Learning with Ground Knowledge Base
- Deciphering Raw Data in Neuro-Symbolic Learning with Provable Guarantees

Challenge 1: Joint Optimization Bottlenecked by Logic Inference

- Fast Abductive Learning by Similarity-based Consistency Optimization
- Efficient Rectification of Neuro-Symbolic Reasoning Inconsistencies by Abductive Reflection

Consistency Measure

- Previous:



- Do not consider feature space
- Unreliable when under-trained



- Good measure:

Model's output + Background knowledge + Input features

Consistency Measure



Raw Input
(Ground Truth: 2+9=11)

Neural Network

Predict

2+9=10

Pseudo Labels

Knowledge Base

Abduce

2+8=10 0.82

2+9=11 0.76

1+9=10 0.73

⋮

Multiple answers with similar confidences, which one to choose?

Visual Similarity Helps Search Pruning



Raw Input
(Ground Truth: 2+9=11)

Neural Network

Model Update

Predict

2+9=10

Pseudo Labels

Knowledge Base

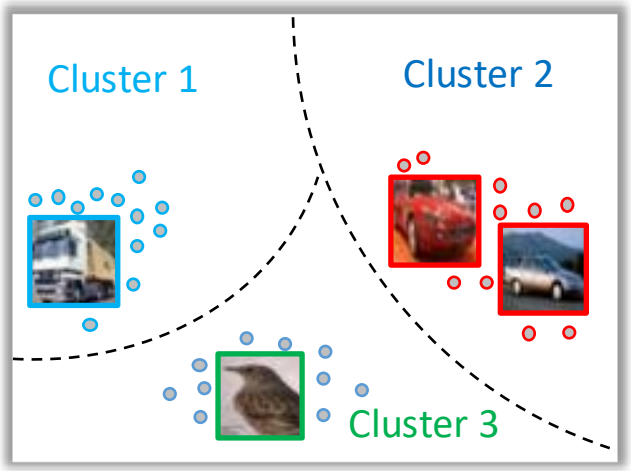
Abduce

2+9=11

Abduced Labels

New Constraints

$x_5 = x_6 \wedge x_3 \neq x_6 \wedge \cdots$



Embedding Space

 and  should share **the same** label!

 and  should have **different** labels!

.....

Collect more samples before model update

- The abduction problem

$$
\begin{aligned}
\max_{\bar{\boldsymbol{Z}}} \quad & \text{Score}(\boldsymbol{X}, \bar{\boldsymbol{Z}}), \\
s.t. \quad & \boldsymbol{X} = (\boldsymbol{x}^{\langle 1 \rangle}, \boldsymbol{x}^{\langle 2 \rangle}, \cdots, \boldsymbol{x}^{\langle m \rangle}), \\
& \bar{\boldsymbol{Z}} \in \mathbb{A}^{\langle 1 \rangle} \times \mathbb{A}^{\langle 2 \rangle} \times \cdots \times \mathbb{A}^{\langle m \rangle}, \\
& \mathbb{A}^{\langle k \rangle} = \{\bar{\boldsymbol{z}} \mid KB \cup \bar{\boldsymbol{z}} \models y^{\langle k \rangle}\}.
\end{aligned}
$$

- Combinatorial optimization problem where the search space grows exponentially with m
- ABLSim uses **beam search** to solve this optimization problem greedily

Results

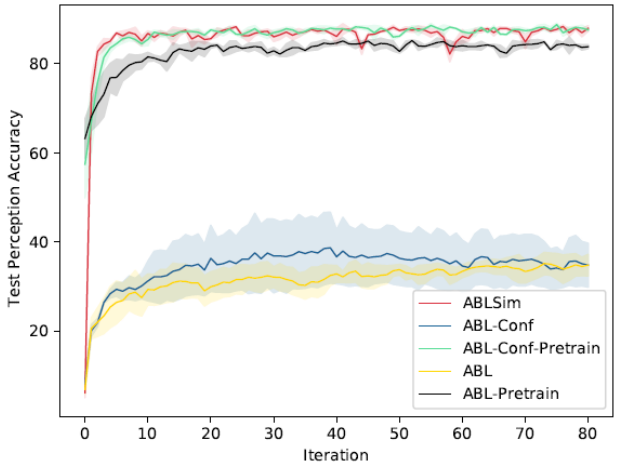
- **MNIST (CIFAR-10) Addition**
- **Handwritten Formula Recognition (HWF)**

	Method	Addition	Addition (CIFAR)	HWF	HWF (CIFAR)
Acc / %	DeepProbLog	96.5±0.5	21.6±1.7	32.2±0.6	15.2±2.6
	NGS-dft	39.9±54.1	38.7±35.1	99.6±0.2	23.8±6.3
	NGS-opt	98.5±0.3	88.7±0.8	99.6±0.2	66.0±14.5
	ABLSim (ours)	**98.8±0.1**	**88.9±0.5**	**99.9±0.1**	**88.4±0.7**
Time / s	DeepProbLog	396±3	time out	time out	time out
	NGS-dft	time out	time out	299±36	time out
	NGS-opt	46±4	6954±558	240±7	time out
	ABLSim (ours)	**42±5**	**6066±79**	**130±4**	**7263±122**

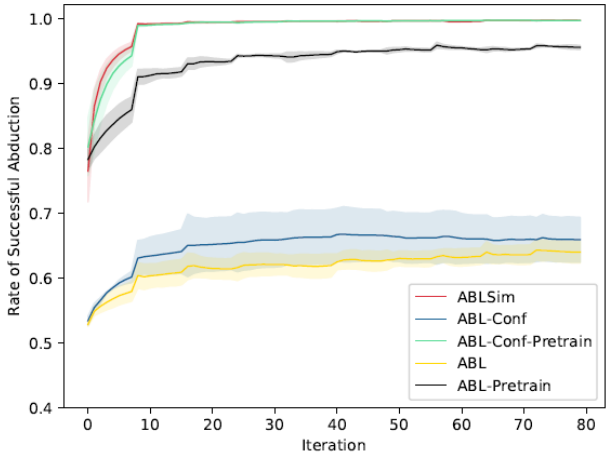
- ABLSim solves all tasks efficiently and achieves a higher accuracy than SOTA models

Results

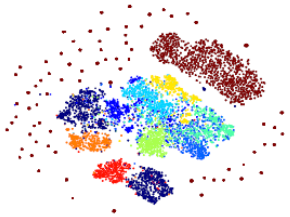
- **CIFAR-10 Decimal Equation Decipherment**



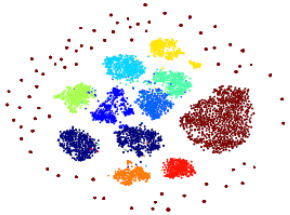
(a) Perception model test accuracy.



(b) Rate of successful abduction.



(c) 1st iteration



(d) 10th iteration

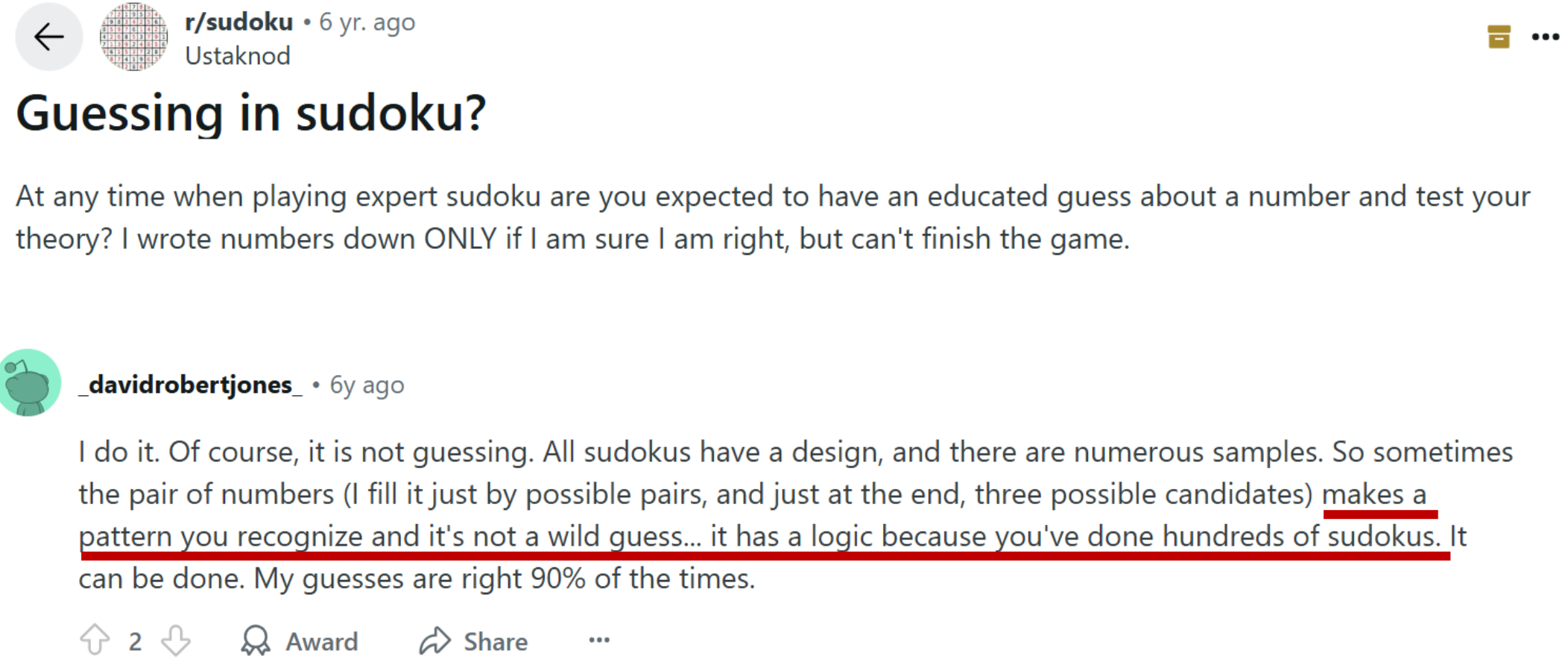
Figure 3: Learning curves (a & b) and the t-SNE visualization of the learned embeddings (c & d).

- Converges much faster and achieves higher accuracy than other methods
- The embeddings of classes are improved after the neural net is updated with the abduced labels

Challenge 1: Joint Optimization Bottlenecked by Logic Inference

- Fast Abductive Learning by Similarity-based Consistency Optimization
- Efficient Rectification of Neuro-Symbolic Reasoning Inconsistencies by Abductive Reflection

Pruning Search Space by "Learning Pattern Recognition Intuitions"



r/sudoku • 6 yr. ago
Ustaknod

Guessing in sudoku?

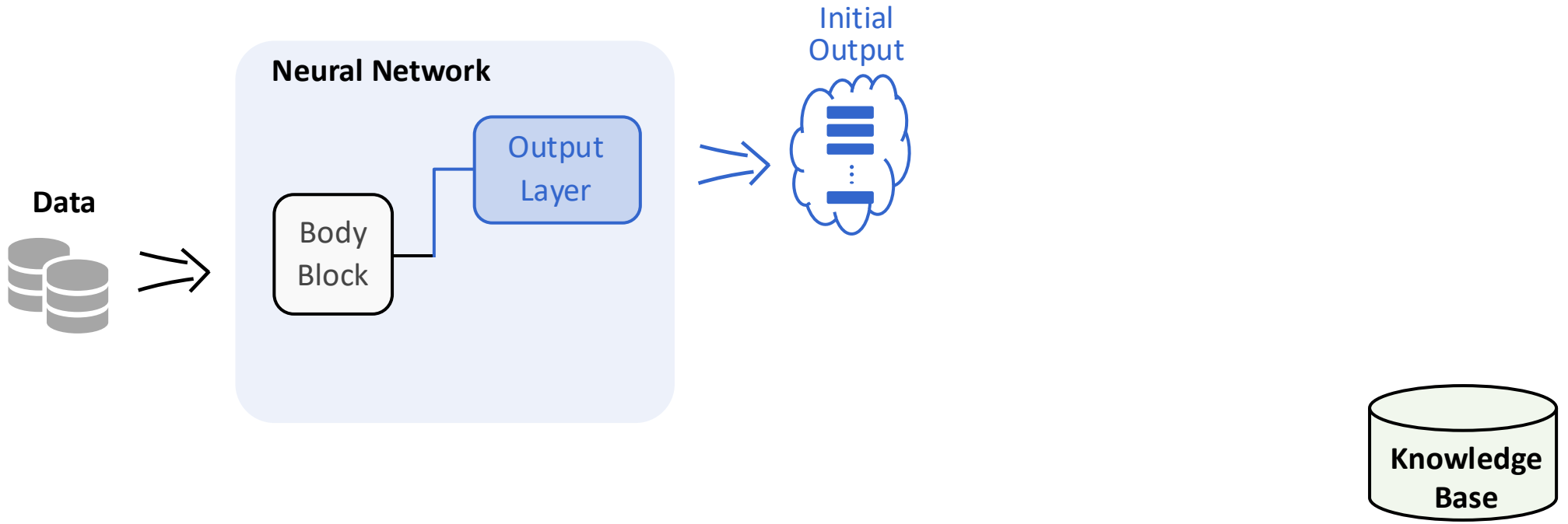
At any time when playing expert sudoku are you expected to have an educated guess about a number and test your theory? I wrote numbers down ONLY if I am sure I am right, but can't finish the game.

davidrobertjones • 6y ago

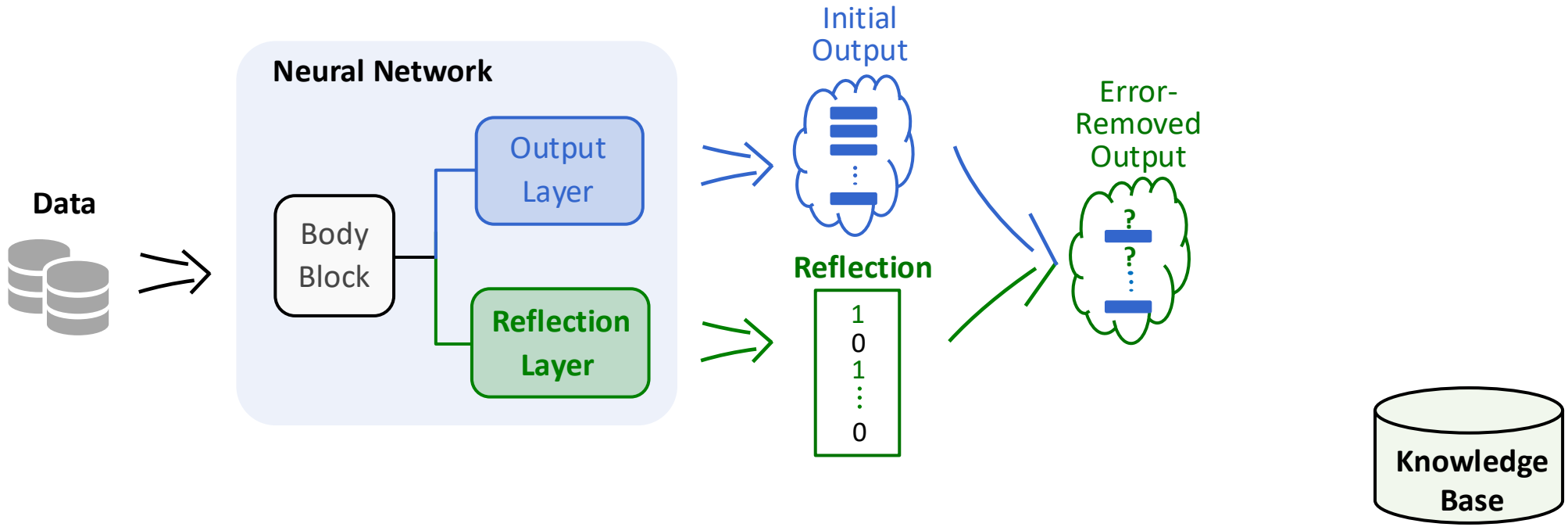
I do it. Of course, it is not guessing. All sudokus have a design, and there are numerous samples. So sometimes the pair of numbers (I fill it just by possible pairs, and just at the end, three possible candidates) makes a pattern you recognize and it's not a wild guess... it has a logic because you've done hundreds of sudokus. It can be done. My guesses are right 90% of the times.

2 Award Share

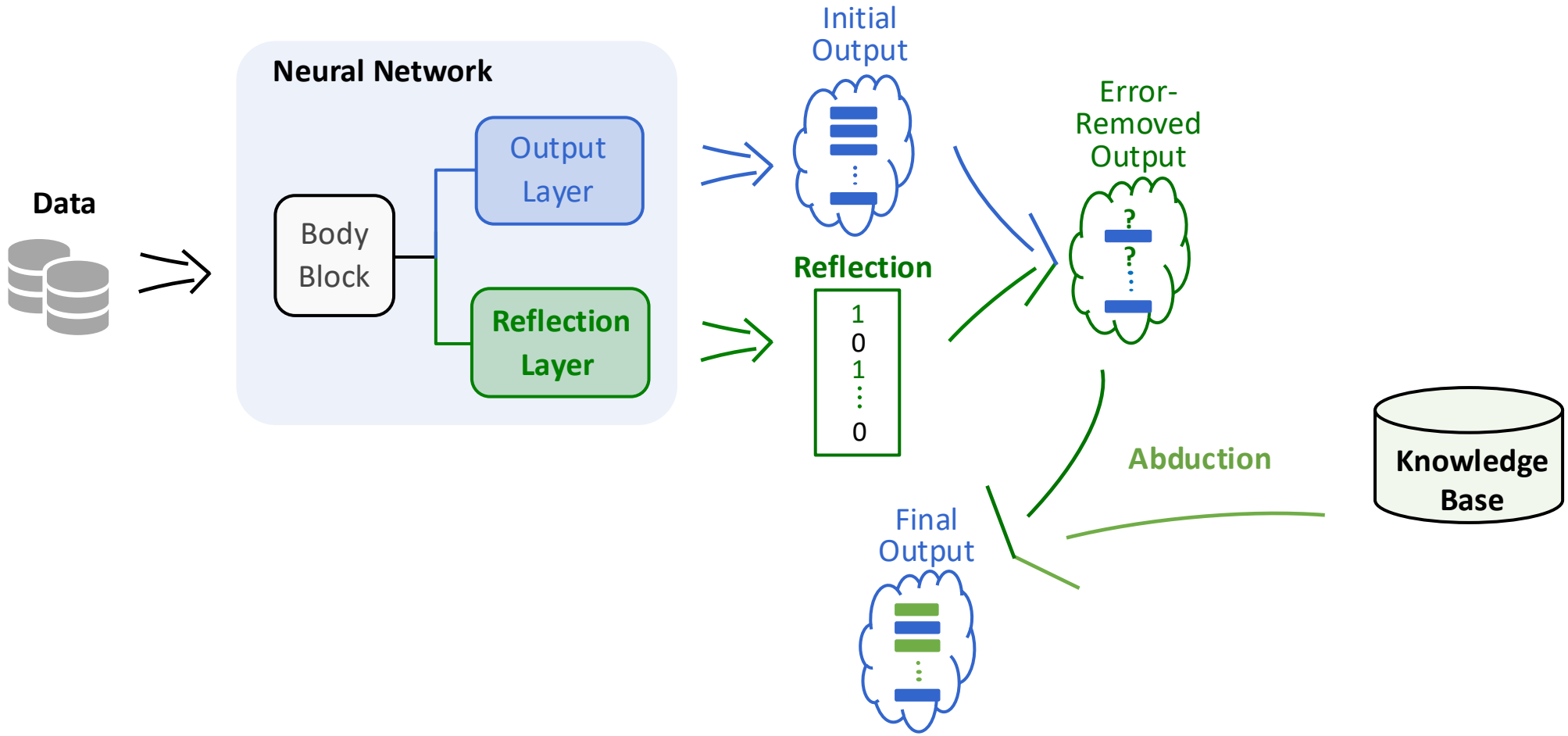
Abductive Learning with Reflection: Efficient Rectification of Neuro-Symbolic Inconsistencies



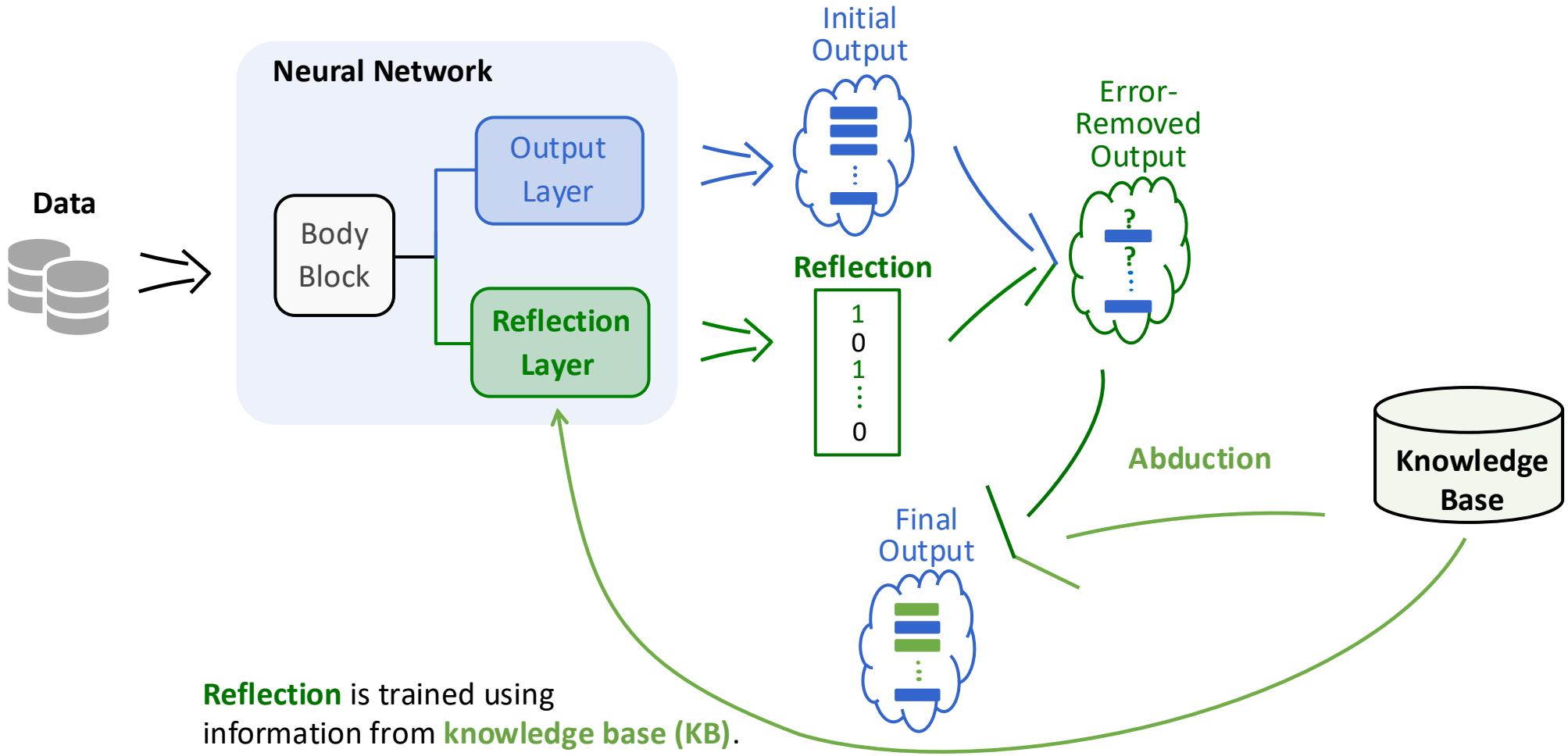
Abductive Learning with Reflection: Efficient Rectification of Neuro-Symbolic Inconsistencies



Abductive Learning with Reflection: Efficient Rectification of Neuro-Symbolic Inconsistencies



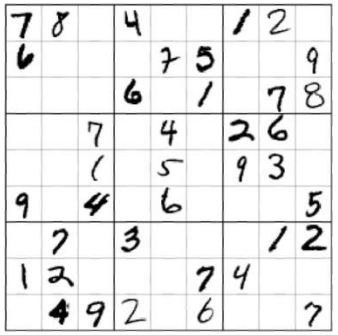
Abductive Learning with Reflection: Efficient Rectification of Neuro-Symbolic Inconsistencies



Experiment 1 Solving Sudoku

7	8	0	4	0	0	1	2	0
6	0	0	0	7	5	0	0	9
0	0	0	6	0	1	0	7	8
0	0	7	0	4	0	2	6	0
0	0	1	0	5	0	9	3	0
9	0	4	0	6	0	0	0	5
0	7	0	3	0	0	0	1	2
1	2	0	0	0	7	4	0	0
0	4	9	2	0	6	0	0	7

x (Symbolic)



x (Visual)

7	8	5	4	3	9	1	2	6
6	1	2	8	7	5	3	4	9
4	9	3	6	2	1	5	7	8
8	5	7	9	4	3	2	6	1
2	6	1	7	5	8	9	3	4
9	3	4	1	6	2	7	8	5
5	7	8	3	9	4	6	1	2
1	2	6	5	8	7	4	9	3
3	4	9	2	1	6	8	5	7

y

Neural Network

Body block f_1 — Graph neural network

Output layer f_2
Reflection layer R — Linear layer

Domain Knowledge

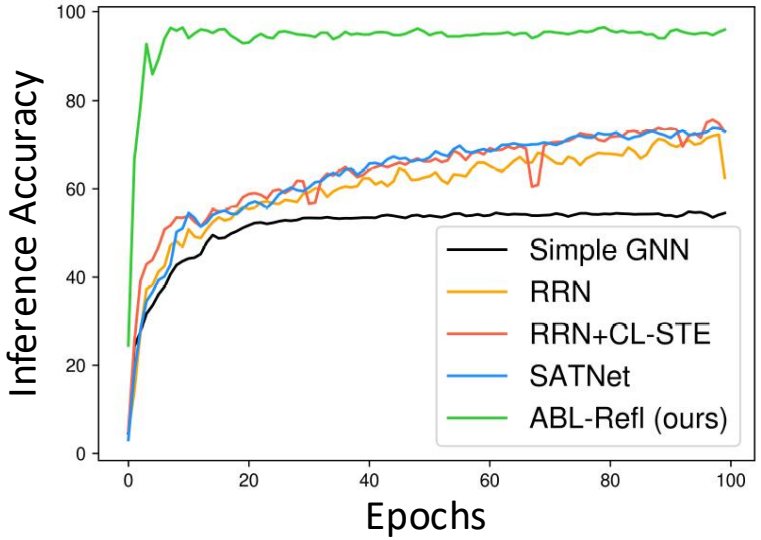
$\mathcal{KB}$ — No duplicate numbers are allowed in any row, column, or 3×3 subgrid.

Solver — MiniSAT (propositional logic) / Prolog with CLP(FD) (first-order logic)

Experiment 1 Solving Sudoku

1. Compared to neuro-symbolic methods

Method	Training Time (min)	Inference Time (s)	Inference Accuracy
RRN	114.8±7.8	0.19±0.01	73.1±1.2
CL-STE	173.6±9.9	0.19±0.02	76.5±1.8
SATNet	140.3±6.8	0.11±0.01	74.1±0.4
ABL-Refl	**109.8**±10.8	0.22±0.02	**97.4**±0.3



Labeled data	Unlabeled data	Test accuracy (%)
20K	0	97.4±0.3
10K	10K	96.3±0.3
5K	15K	95.8±0.6
2K	18K	94.7±0.8

Achieves better **reasoning accuracy** in only **a few epochs** (<10) with less labeled data.

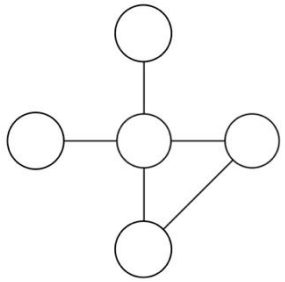
Experiment 1 Solving Sudoku

2. Compared to symbolic solver

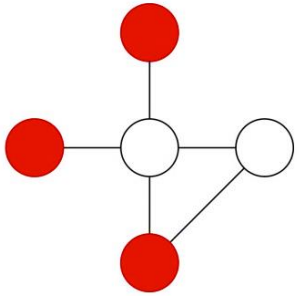
$\mathcal{KB}$ Form	Solver	Method	Inference Accuracy	Inference Time (s)		
				NN Time	Abduction Time	Overall Time
Propositional logic	MiniSAT	Solver only	100±0	-	0.227±0.024	0.227±0.024
		ABL-Refl	97.4±0.3	0.021±0.004	**0.196**±0.015	**0.217**±0.019
First-order logic	Prolog with CLP(FD)	Solver only	100±0	-	105.81±5.62	105.81±5.62
		ABL-Refl	97.4±0.3	0.021±0.004	**31.86**±1.88	**31.88**±1.89

Enhances the **efficiency of symbolic solvers**,
applicable for both propositional logic (MiniSAT) and first-order logic (Prolog).

Experiment 2 Solving Combinatorial Problems



x



y

Domain Knowledge

$\mathcal{KB}$ Mathematical definition of clique/independent set…

Solver Gurobi

Method	Dataset (Graph nums./Avg. nodes per graph/Avg. edges per graph)			
	ENZYMES (600/33/62)	PROTEINS (1113/39/73)	IMDB-Binary (1000/19/97)	COLLAB (5000/74/2457)
Erdos	0.883±0.156	0.905±0.133	0.936±0.175	0.852±0.212
Neural SFE	0.933±0.148	0.926±0.165	0.961±0.143	0.781±0.316
ABL-Refl	**0.991**±0.017	**0.985**±0.020	**0.979**±0.029	**0.982**±0.015

Capable of handling scalable data scenarios, and a wide range of domain knowledge.

Challenge 2: Imperfect Knowledge Base

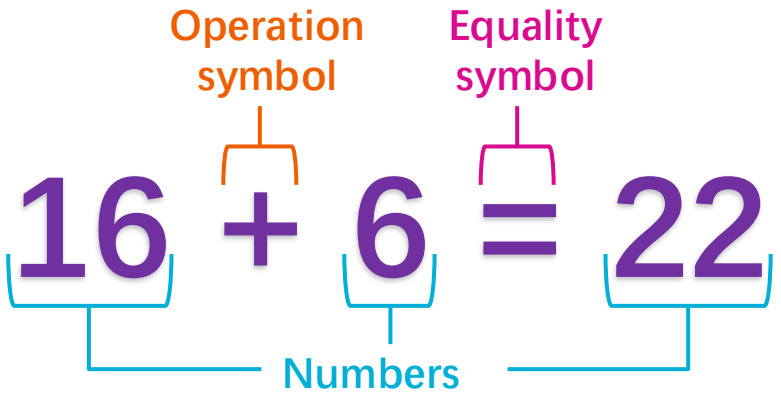
- Enabling Knowledge Refinement upon New Concepts in Abductive Learning

ABL in Closed Environment

Closed Environment

- A fixed set of symbols are enough to represent the concepts in the domain

Example



Concepts in the task

- Provided with all the numbers and operator symbols that could appear

ABL in Open Environment

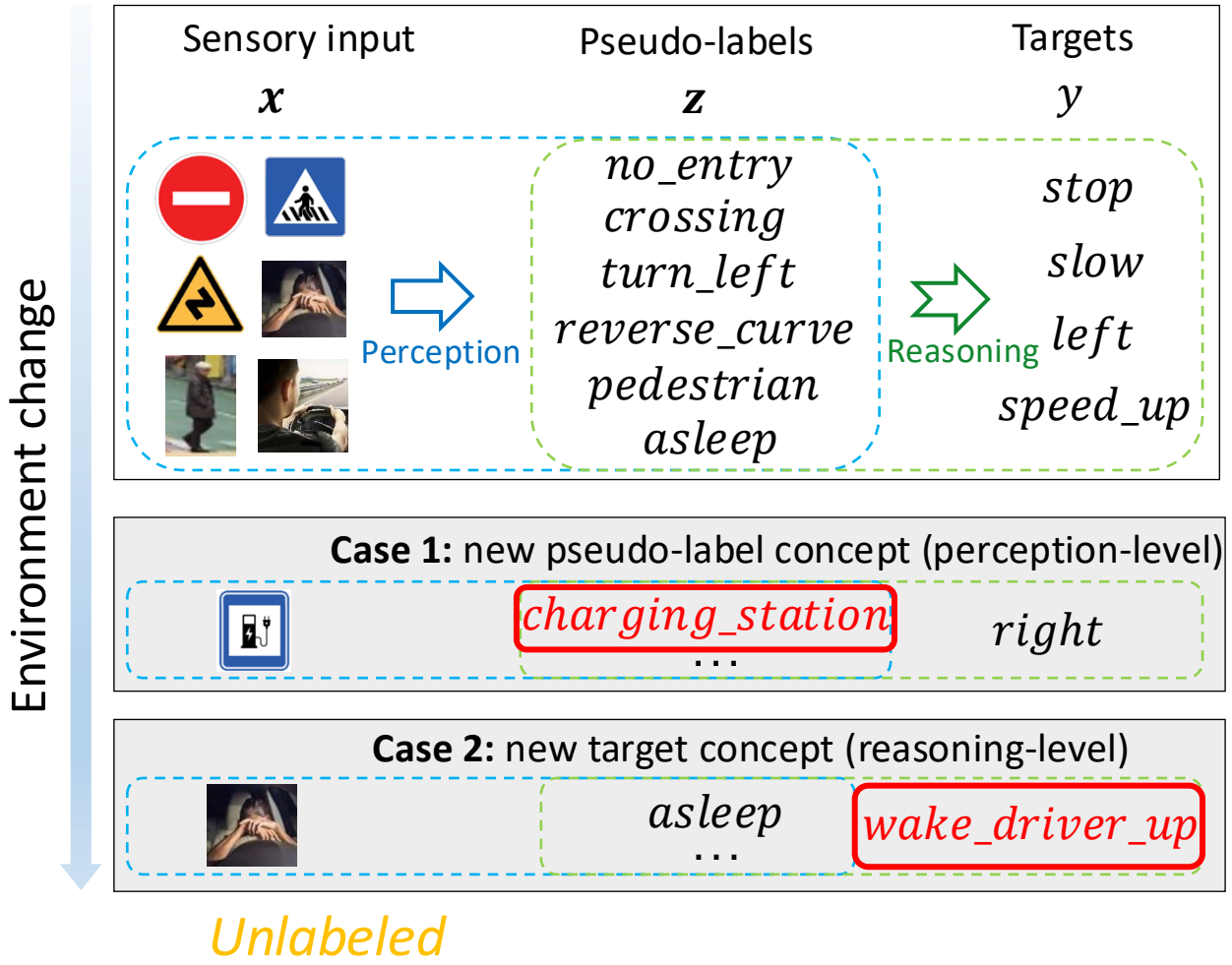
Open Environment

- New concepts may emerge

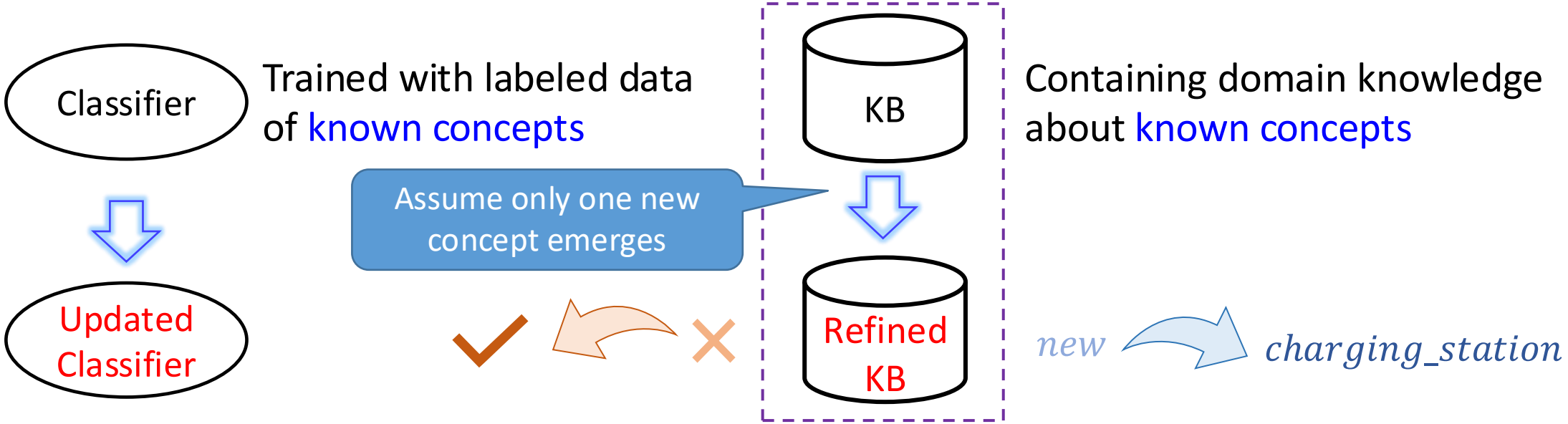
Example



- Ignoring new concepts can lead to significant performance decay
- Crucial for ABL to identify and internalize the new concepts



ABL in Open Environment



Main challenges

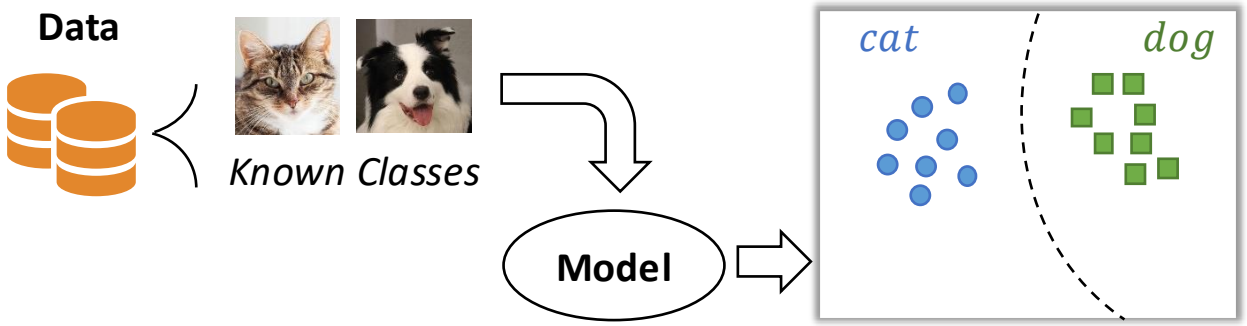
- **Conflictive**: Identify new concepts and add them to existing KB may encounter potential conflicts

Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning

$dog(X) \leftarrow \cdots$

$cat(X) \leftarrow sharp_claws(X) \wedge \cdots$

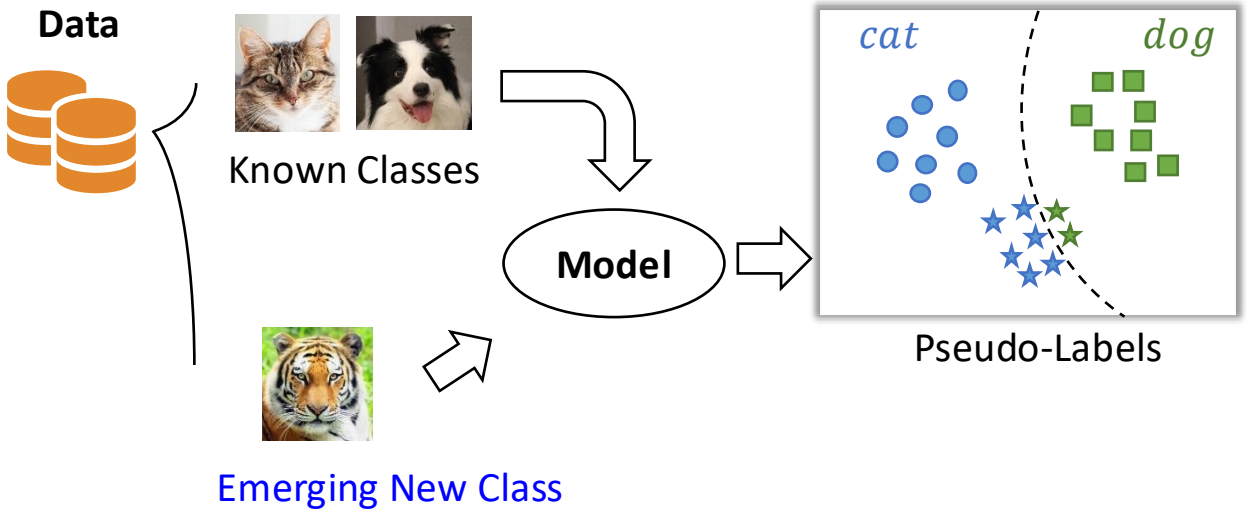
Initial Knowledge Base



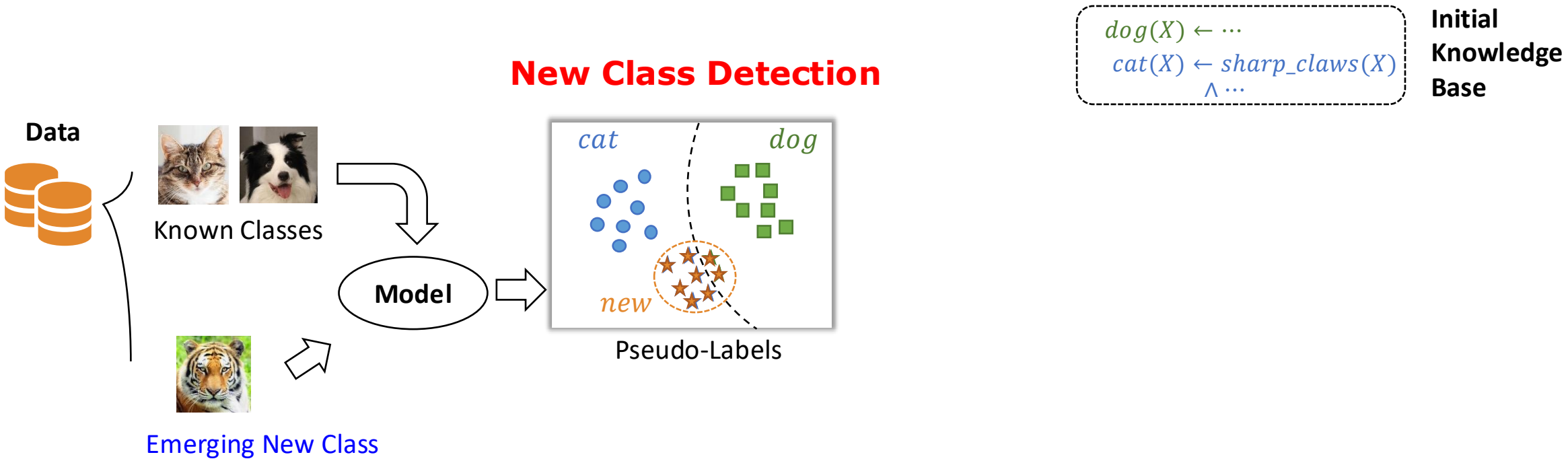
Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning

$dog(X) \leftarrow \cdots$
$cat(X) \leftarrow sharp_claws(X) \wedge \cdots$

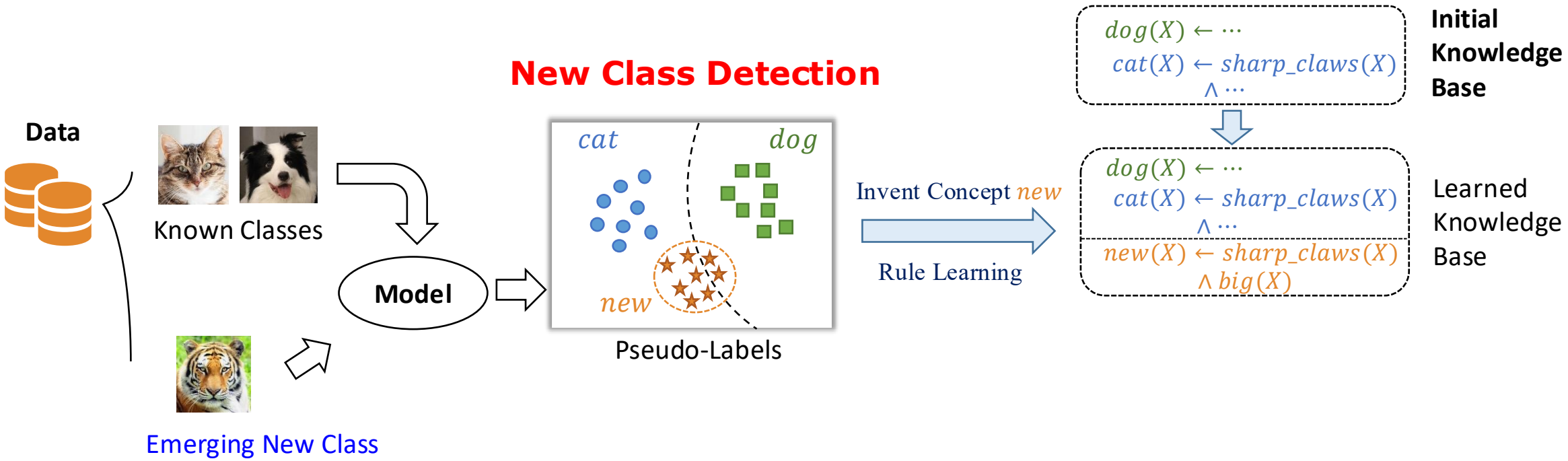
Initial Knowledge Base



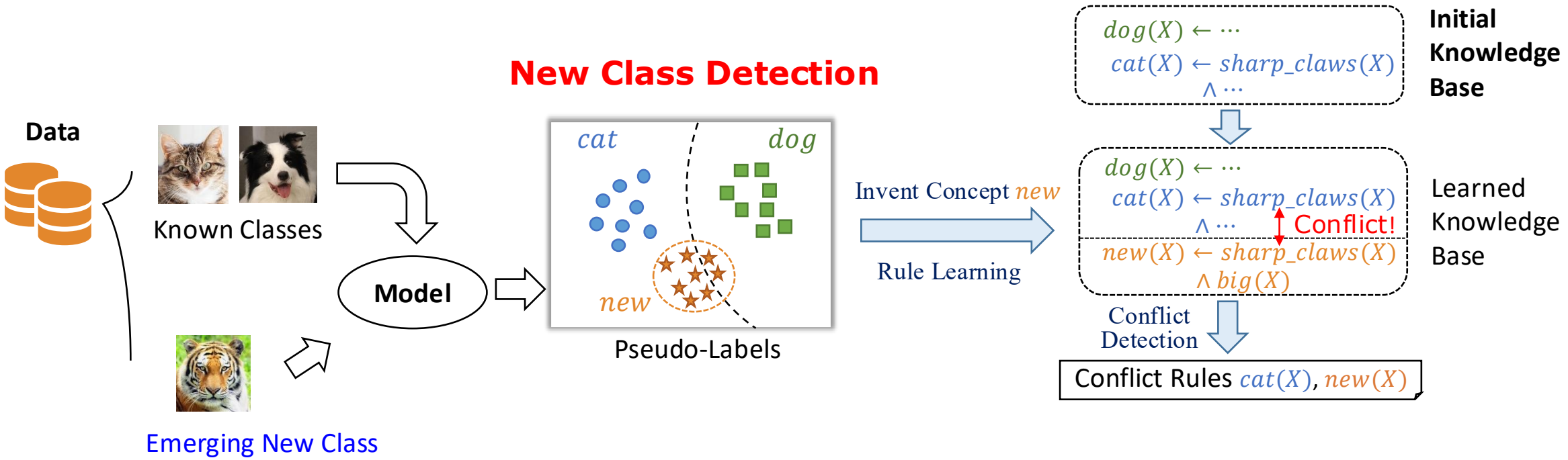
Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning



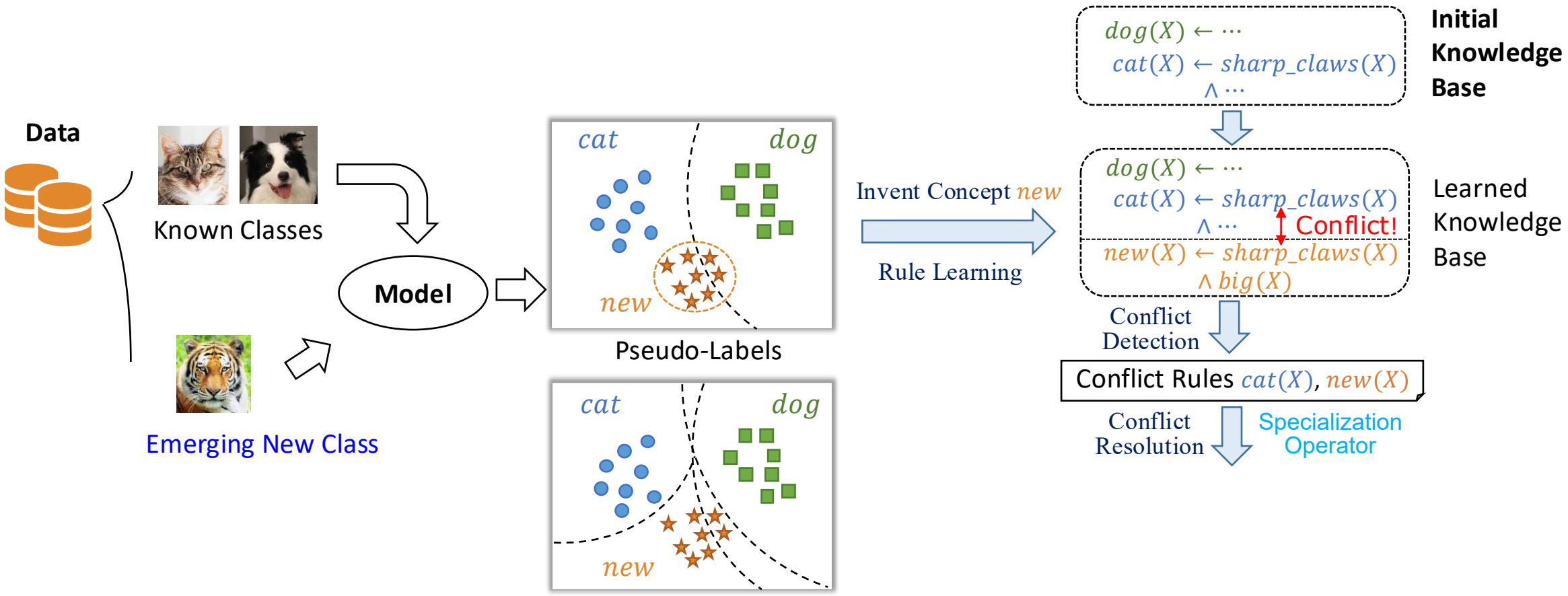
Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning



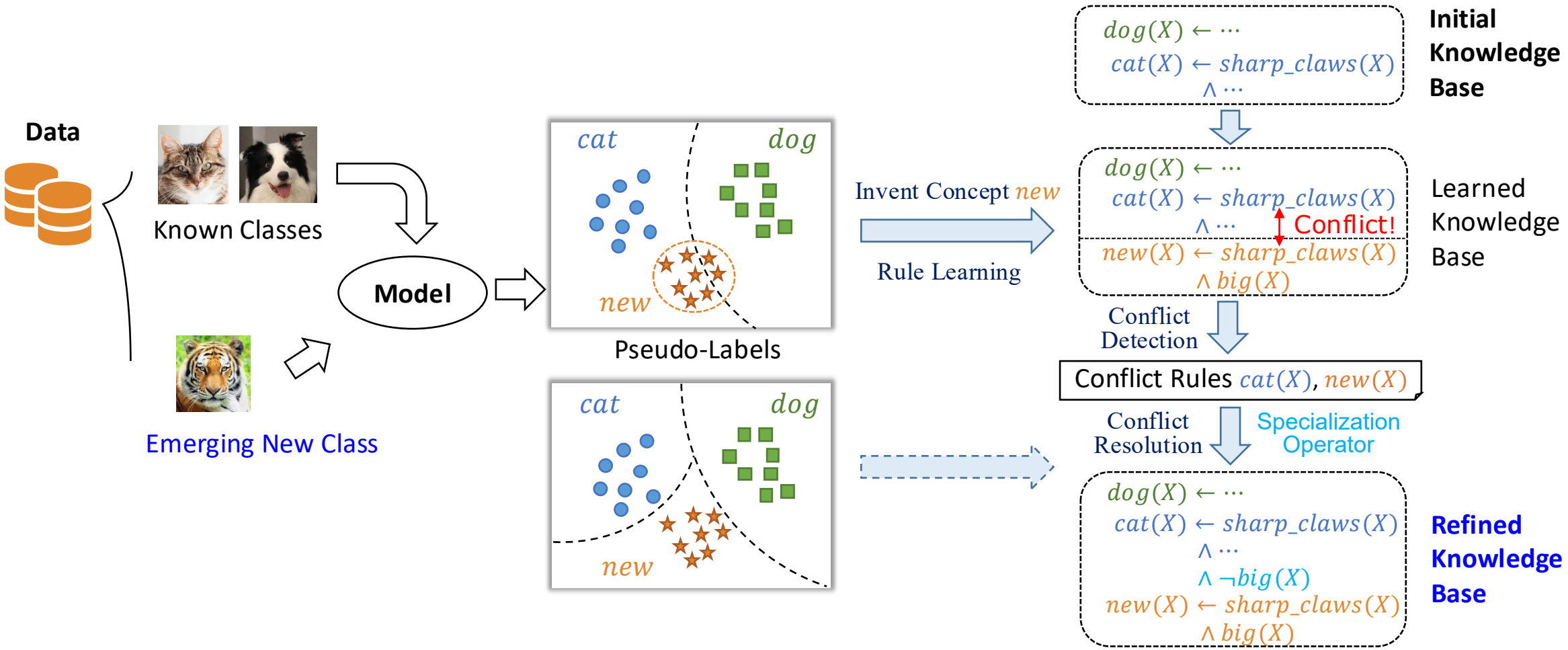
Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning



Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning



Knowledge Refinement by New Concept Invention corresponding to NewClass identified in machine learning

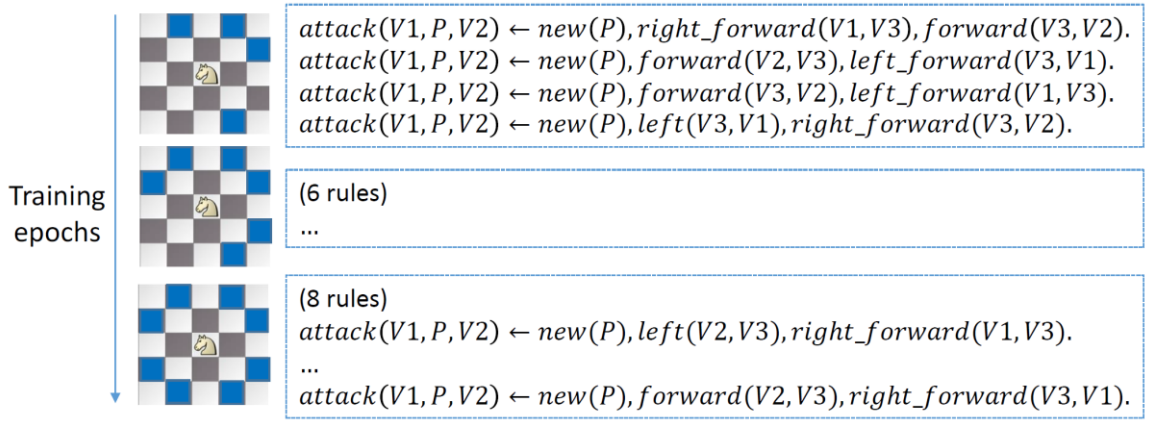


Results (Learned rules)

Task: Less-Than with New Digits

$$succ(X, Y) \leftarrow seven(X) \wedge new(Y).$$
$$succ(X, Y) \leftarrow new(X) \wedge nine(Y).$$

Task: Chess with New Pieces



Task: Multiples of Three

$$even([X_1, X_2]) \leftarrow divisible(X_1, 2) \wedge divisible(X_2, 2) \wedge \neg divisible(X_1, 3).$$
$$even([X_1, X_2]) \leftarrow divisible(X_1, 2) \wedge divisible(X_2, 2) \wedge \neg divisible(X_2, 3).$$

- ABL_{nc} learned the correct rules in all three tasks
- The quality of rules improves as training goes on
- The conflict rules are resolved

Results (Machine Learning Model)

Task	Less-Than with New Digits		Chess with New Pieces		Multiples of Three	
	Perception Acc.	Reasoning Acc.	Perception Acc.	Reasoning Acc.	Perception Acc.	Reasoning Acc.
CNN	0.889±0.006	0.915±0.016	0.831±0.001	0.558±0.029	0.903±0.008	0.601±0.019
CNN_{new}	0.946±0.014	0.876±0.013	0.924±0.014	0.910±0.006	0.903±0.008	0.323±0.012
ABL	0.959±0.015	0.904±0.009	0.930±0.015	0.937±0.009	0.938±0.005	0.639±0.011
ABL_{nc}	**0.975±0.005**	**0.986±0.005**	**0.990±0.006**	**0.984±0.005**	**0.960±0.008**	**0.963±0.004**

Table 1: Test accuracy on perception and reasoning of different methods.

- ABL_{nc} achieves higher accuracy than other methods in all tasks

Question: Does Formal Knowledge Always Help?

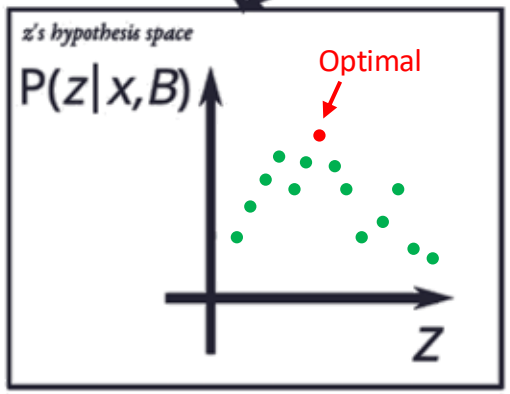
- Abductive Learning with Ground Knowledge Base
- Deciphering Raw Data in Neuro-Symbolic Learning with Provable Guarantees

The Abductive Optimization

Observing evidence $\langle x, y \rangle$, and maximizes the *consistency* of $H \cup z$ and f w.r.t. the evidence and background knowledge:

example([, ,], 15).

Domain Knowledge:
a correct program

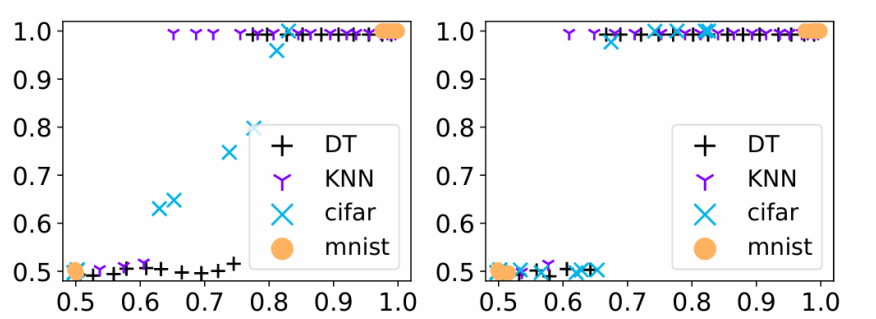


Optimization in this unknown distribution:

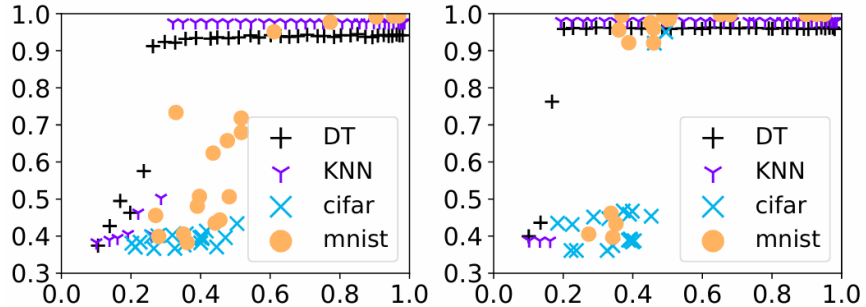
- exact inference [Manhaeve et al., 2018]
- evolutionary algorithm [Dai et al., 2019]
- Reinforcement learning [Dong et al., 2019]
- MCMC sampling [Li et al., 2020]

Reason

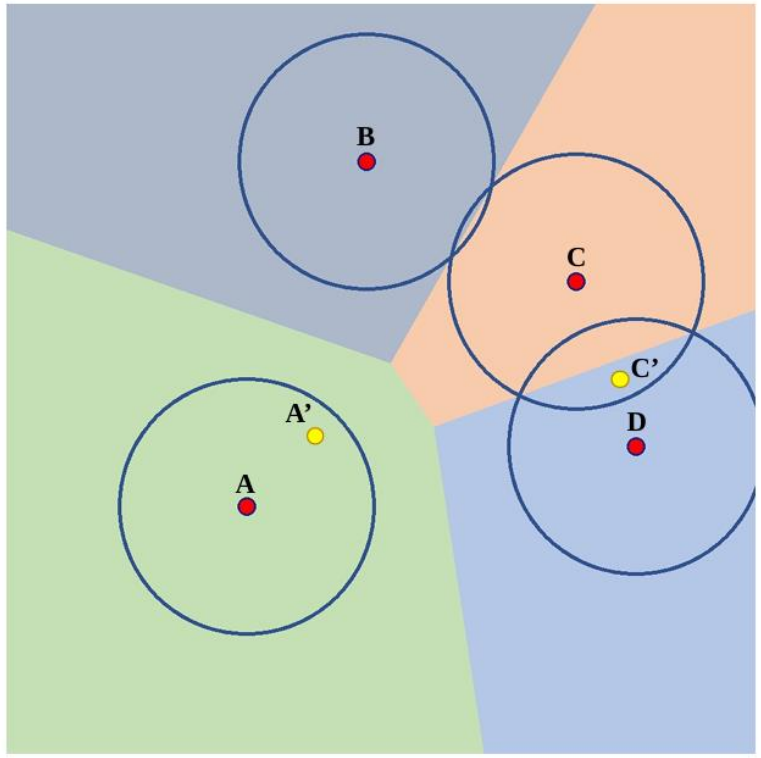
A weak perception model may get the abduction trapped in local optima.



(a) Hamming w/o confidence. (b) Hamming w/ confidence.



(c) Equation w/o confidence. (d) Equation w/ confidence.



Learning improves reasoning.

An example space with good geometry helps,

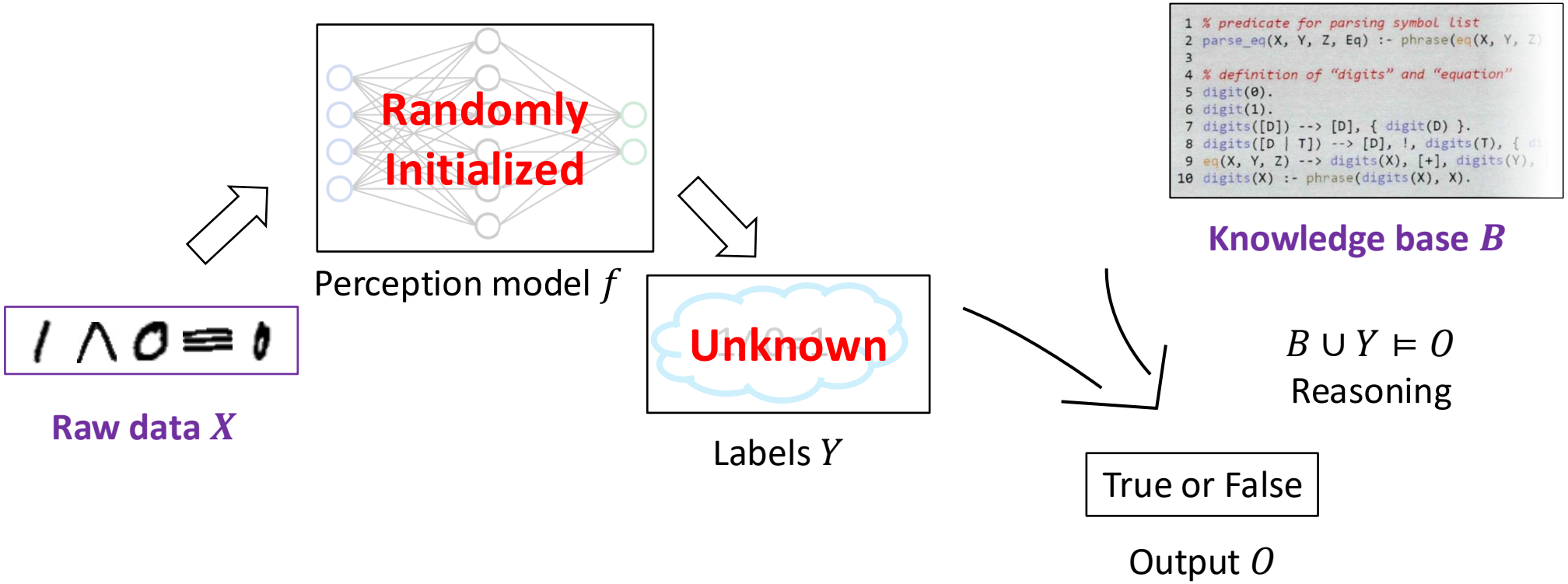
and pre-train is very useful.

Reasoning improves learning?

How to measure the abduced hypothesis space?

When does knowledge improves learning even without any pre-train?

Mystery



- When the *true labels* are **unknown** and the *model* is **randomly initialised**, abductive learning may still **work**; learned models are **accurate** in many cases

How is this feasible?

Mystery

- "The Cold Start Problem"

- Preliminary experiments show that:

Sometimes it works. Sometimes it doesn't

- **Raw data:** 
- **Knowledge:** **"They are all conjunctive equations"**

- **Raw data:** 
- **Knowledge:** **"They are all XOR equations"**

Our Starting Point

- The information from knowledge reasoning is *coarse*:
  - For a raw sequence such as 
  - The learner only know that it belongs to one of the following

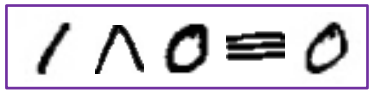
0 ∧ 0 = 0	**0 ∧ 1 = 0**	**1 ∧ 0 = 0**	**1 ∧ 1 = 1**

- A *simple* yet *important* observation:

0 ∧ 0 = 0	**0 ∧ 1 = 0**	**1 ∧ 0 = 0**	**1 ∧ 1 = 1**

  - For any equation to be correct, the instance  must be "**=**"
  - This index information is called "location signals"

Location Signals as Bridges



- For any unlabelled instance x from a sequence $X = [x_0, x_1, x_2, x_3, x_4]$, the learner can observe its location ι from the index value. E.g.,

0 ∧ 0 = 0	**0 ∧ 1 = 0**	**1 ∧ 0 = 0**	**1 ∧ 1 = 1**

- If the x's location ι is "3", then its label y is "=";
- If the x's location ι is "1", then its label y is "∧";
- If the x's location ι is "4", then its label y is undetermined, but more likely to be "0" than "1". In ¾ cases, y is "0".

This calls for a formulation of (probabilistic) bridges

Formulation

$p(\iota|y)$ works as a bridge between $p(\iota|x)$ and $p(y|x)$

- $p(\iota|y)$ can be *directly* estimated from knowledge base (all groundings)
- For example, given the knowledge of candidate labels $\left\{\begin{matrix}[0,0,0],\\ [0,1,0],\\ [1,0,0],\\ [1,1,1]\end{matrix}\right\}$
  - We can observe $p(\iota = 2 \mid y = 0) = \frac{3}{7},\ p(\iota = 3 \mid y = 1) = \frac{1}{5},\ \cdots$
  - These probabilities are summarised into a matrix $Q = \begin{pmatrix} 2/7 & 2/7 & 3/7 \\ 2/5 & 2/5 & 1/5 \end{pmatrix}$

Q has full rank

Formulation

$p(\iota|y)$ works as a bridge between $p(\iota|x)$ and $p(y|x)$

- $p(\iota|y)$ can be *directly* computed from knowledge
- For example, given the knowledge of candidate labels $\left\{ \begin{matrix} [0,0,0], \\ [0,1,1], \\ [1,0,1], \\ [1,1,0] \end{matrix} \right\}$
  - We can observe $p(\iota = 2 \mid y = 0) = \frac{1}{3},\ p(\iota = 2 \mid y = 1) = \frac{1}{3},\ \cdots$
  - These probabilities are summarised into a matrix $Q = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$

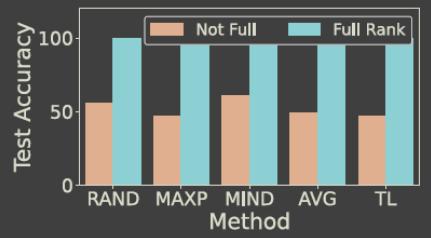
Q is not full rank

Theory

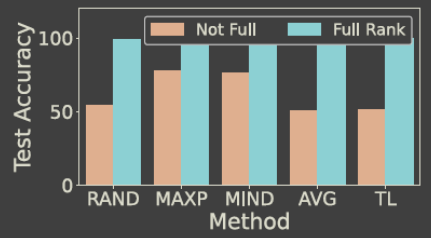
$p(\iota|y)$ works as a bridge between $p(\iota|x)$ and $p(y|x)$

Theorem 1. If the probability matrix Q, which represents $p(\iota|y)$, has full row rank, then the probability of true labels given an instance $p(y|x)$ can be uniquely identified from $p(\iota|x)$, recovering the Bayesian optimal classifier.

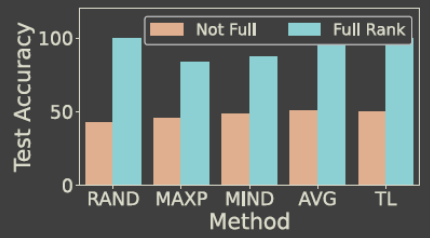
Results



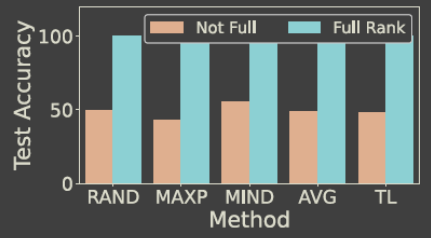
(a) DNF, $m = 3$



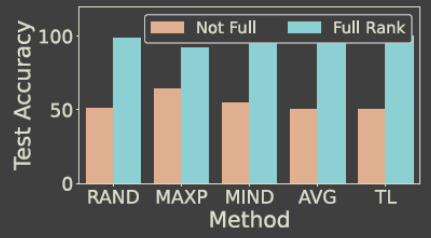
(b) DNF, $m = 4$



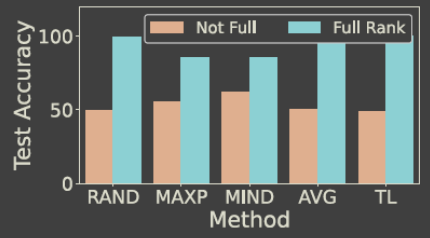
(c) DNF, $m = 5$



(d) CNF, $m = 3$



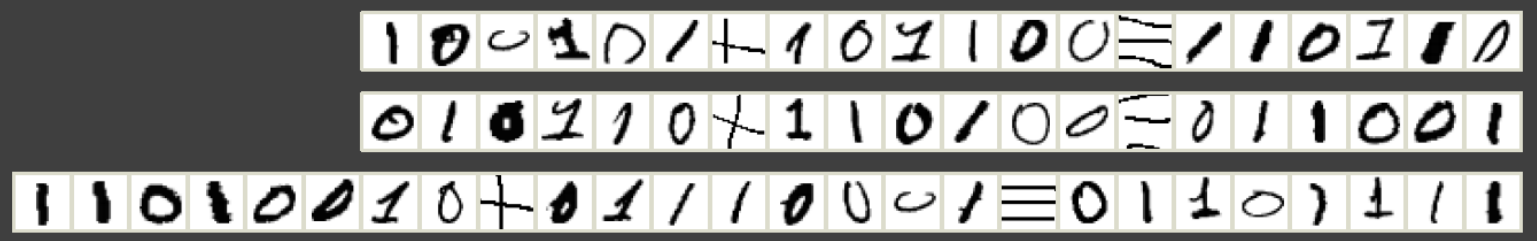
(e) CNF, $m = 4$

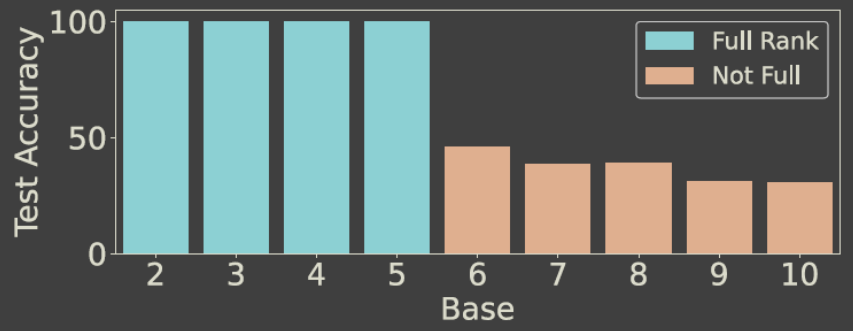


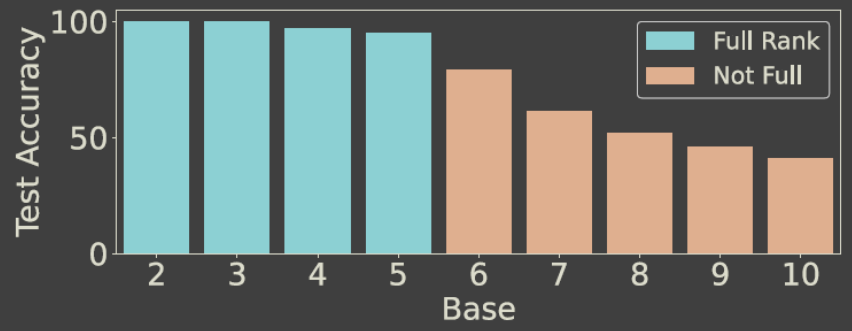
(f) CNF, $m = 5$

- Randomly generated CNFs/DNFs as target concepts with different Q matrices
- Abduction by `RAND` (random choice), `MAXP` (maximal perception probability), `MIND` (minimal Hamming distance), `AVG` (average/expectation over symbols)
- `TL`: Minimising the loss function derived from our bound

Results







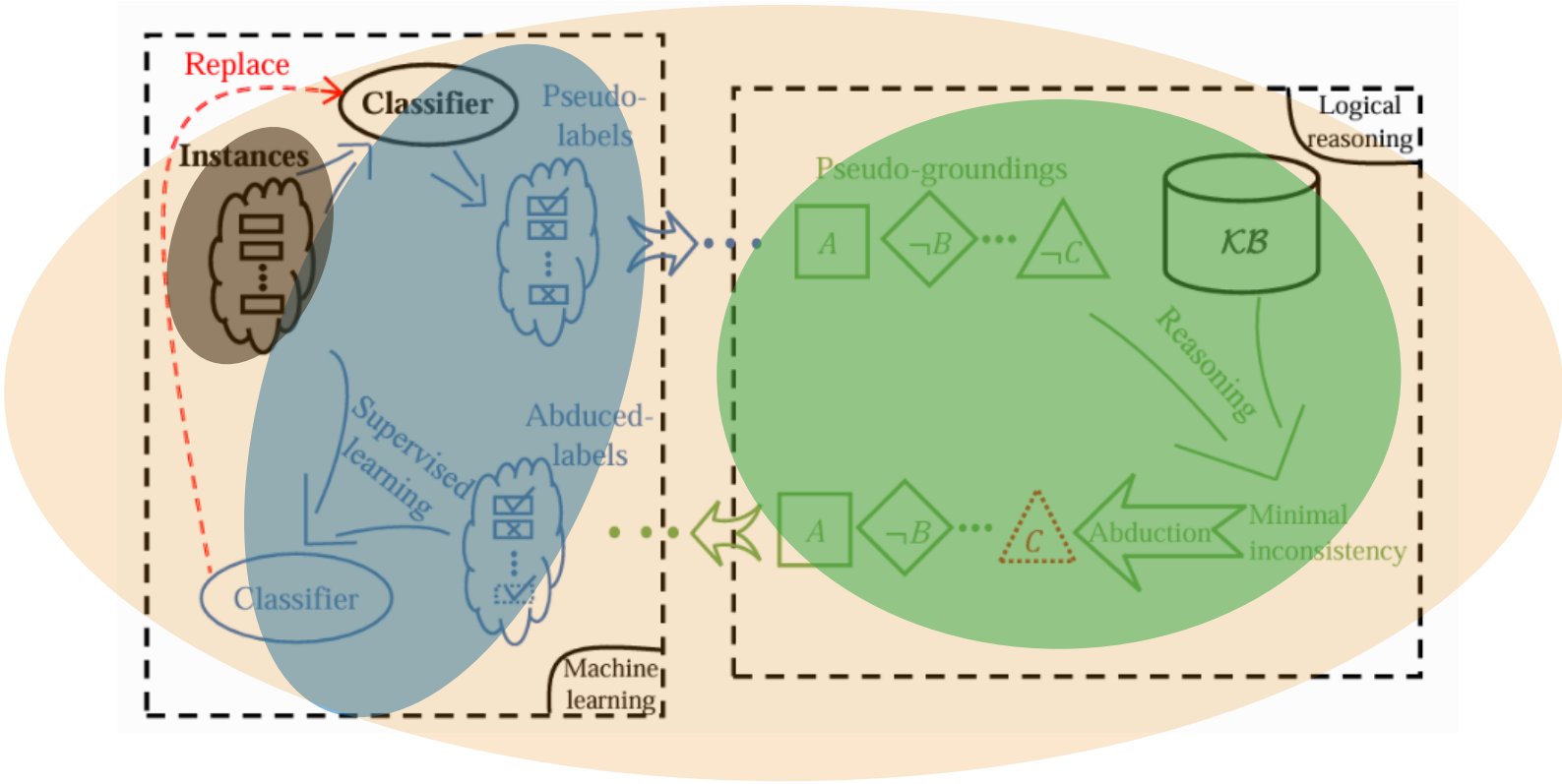
- *Handwritten Equation Deciphering* task with different number-based addition tasks
- Abduction with `MAXP` (left) and `TL` (right) strategy
- Cold start: NO pre-training for the perception model

Part IV. Tool

- ABLkit: A Python Toolkit for Abductive Learning

ABLkit：a toolkit for ABL

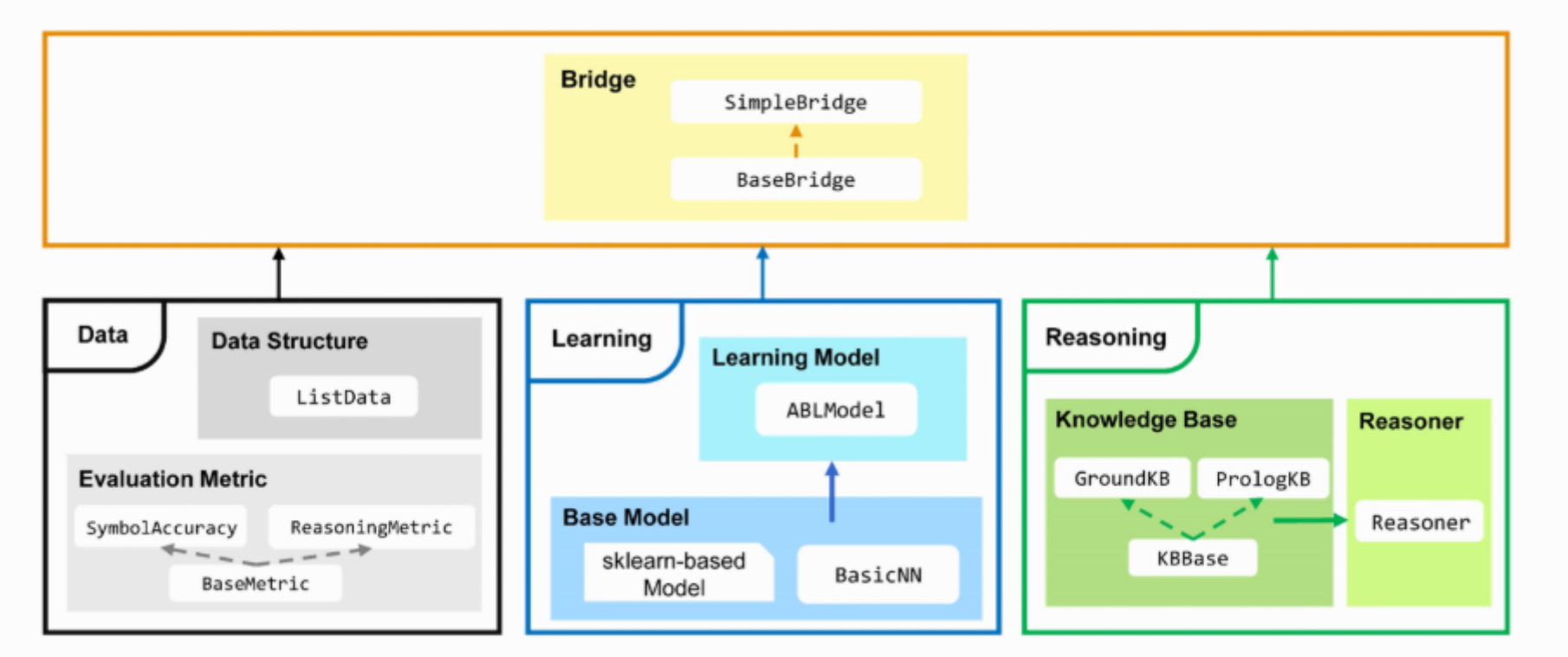
- An efficient and general-purpose open-source toolkit for abductive learning



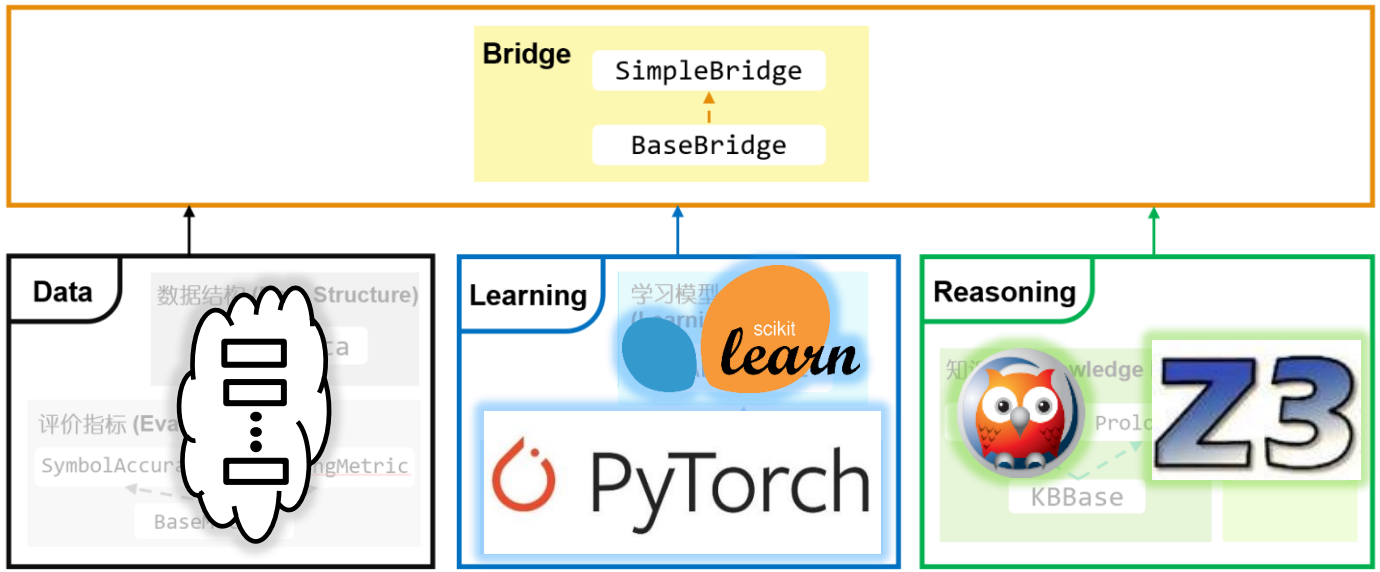
Four modules: Data + Learning + Reasoning + Bridge

Architecture

Four modules: Data + Learning + Reasoning + Bridge



Feature



High Flexibility Compatible with various machine learning modules and reasoning components.

Easy-to-Use Interface Provide **data**, **model**, and **knowledge**, and start with just a few lines of code.

Optimized Performance Optimization for high performance and accelerated training speed.

Feature

Method	Learning + Reasoning
NeurASP	PyTorch + ASP
DeepProbLog	PyTorch + ProbLog
DeepStochLog	PyTorch + SDCG
NGS	PyTorch + CFG
LTN	PyTorch + FL
ABLkit	**Unrestricted + Unrestricted**

While baseline methods are limited to using neural networks as the learning model and constrain the reasoning component to a specific language,

ABLkit supports almost any learning model or reasoning module suitable for the task.

Experiments

MNIST addition

Method	Accuracy	Time	Memory
NeurASP	96.2	966	3552
DeepProbLog	97.1	2045	3521
DeepStochLog	97.5	257	3545
LTN	97.5	251	3860
ABLkit	**98.1**	**47**	**2482**

The training time of classical methods exceeds that of ABLkit by more than **43 times**

Handwritten formula

Method	Accuracy on varied length					Time	Memory
	1	3	5	7	所有		
DeepProbLog	90.8	85.6	Timeout	Timeout	Timeout	Timeout	4315
DeepStochLog	92.8	87.5	92.1	Timeout	Timeout	Timeout	4355
NGS	91.2	89.1	92.7	5.2	98.4	426.2	3705
ABLkit	**94.0**	**89.7**	**96.5**	**97.2**	**98.6**	**77.3**	**3074**

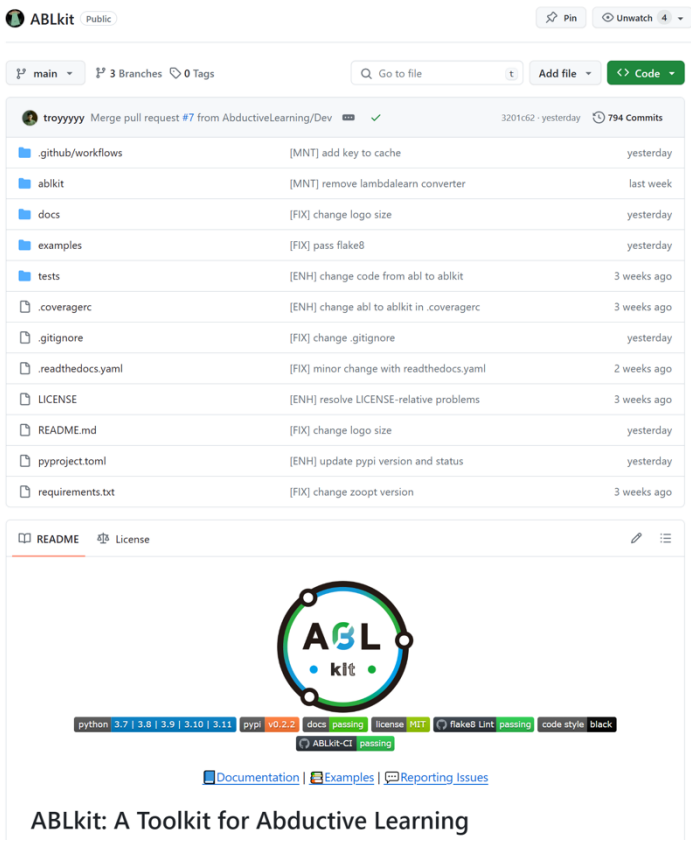
Often time out when handling longer equations

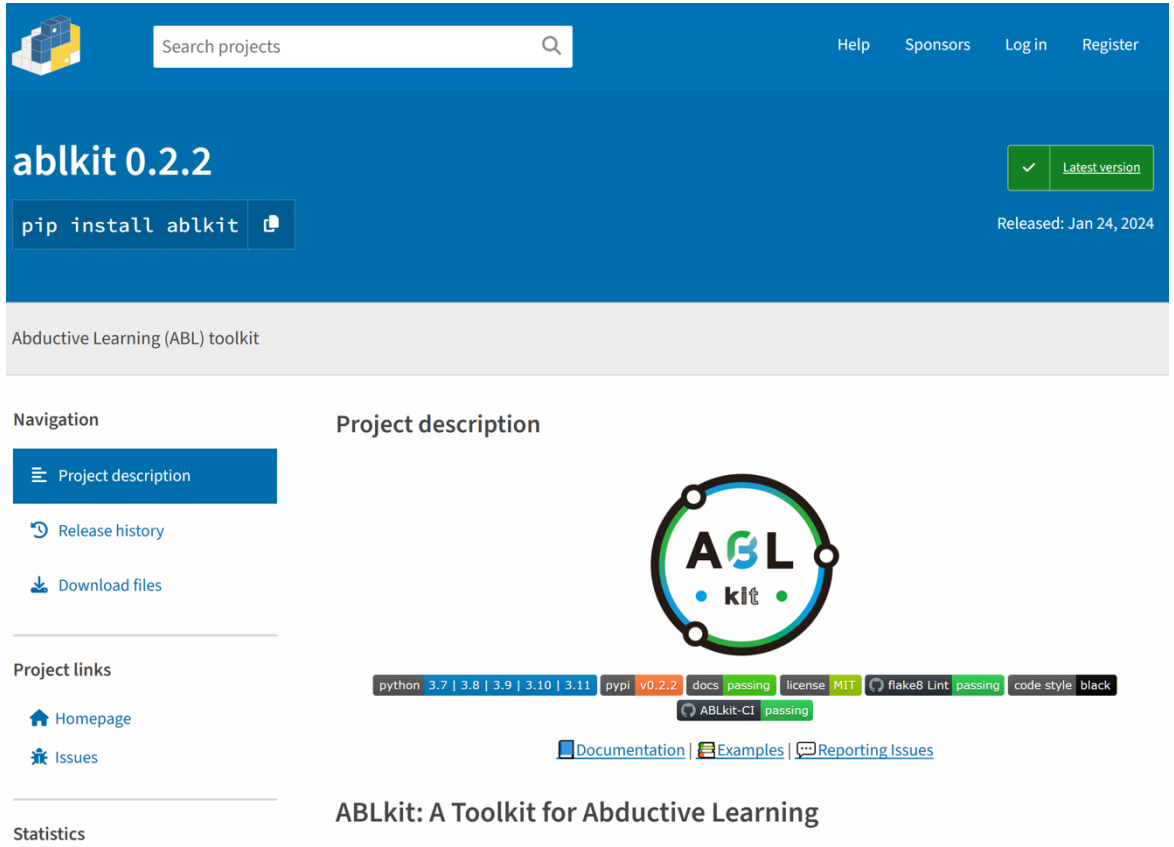
- ABLkit achieves **the highest accuracy on all datasets**, and significantly outperforming other methods in both **training time** and **memory consumption**.

The toolkit is now open-source

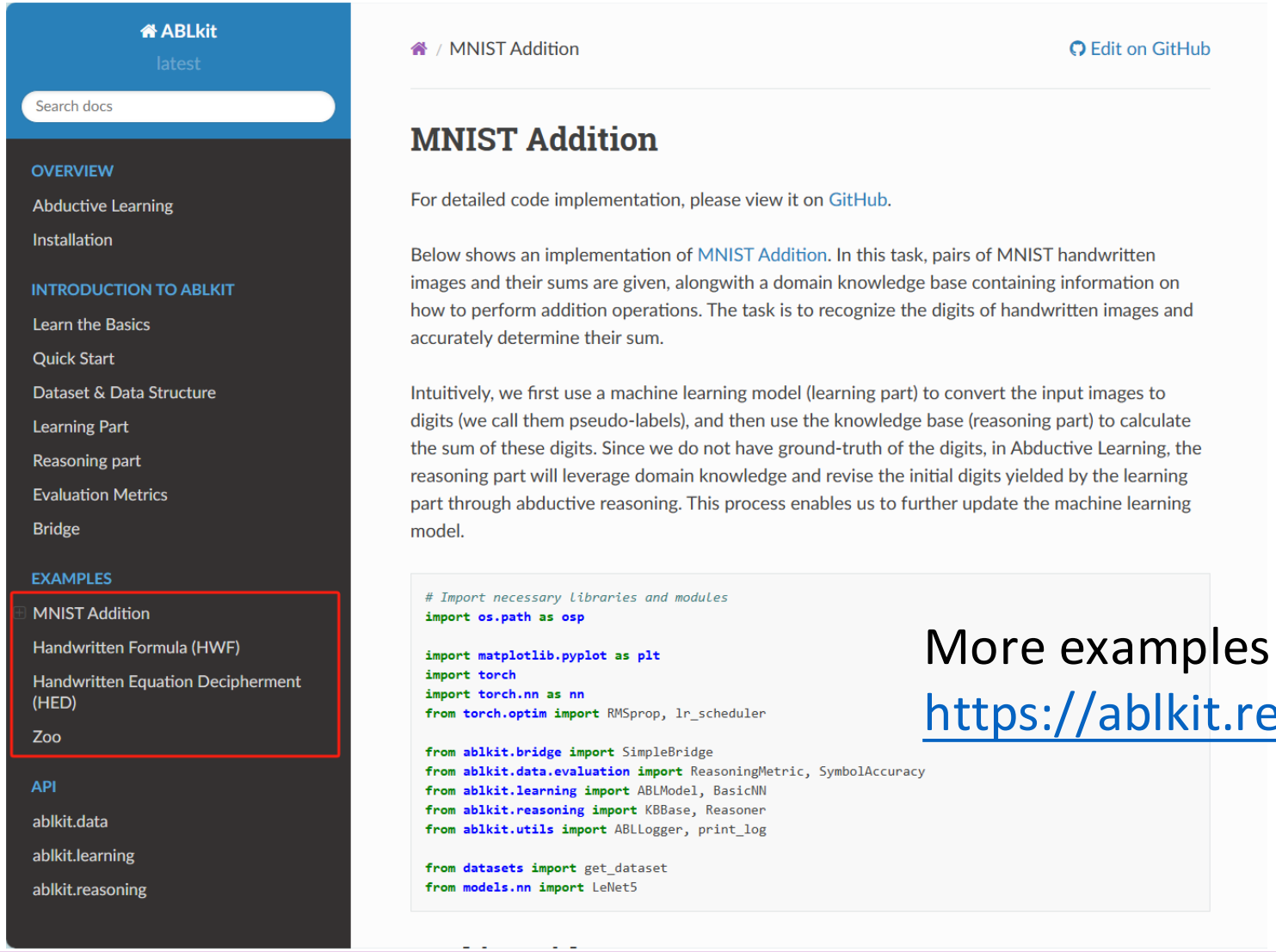
GitHub: https://github.com/AbductiveLearning/ABLKit

install: `pip install ablkit`





Documentation



More examples are shown in **documentation**
https://ablkit.readthedocs.io/en/latest/

To be continued…

Q&A Thanks!