



Demystifying Neuro-Symbolic AI for Software-Hardware Co-Design

Zishen Wan

PhD Student @ School of ECE, Georgia Tech

Advisors: Prof. Arijit Raychowdhury, Prof. Tushar Krishna



Seminar @ Google, April 17, 2025

Executive Summary

- **Understand** neuro-symbolic workloads from architecture and system perspective.
- Identify **optimization opportunities** for neuro-symbolic systems.
- Demonstrate orders of scalability and efficiency improvement of neuro-symbolic workload via **co-designed** system.

Neural Networks in Our Daily Life



Image Recognition



Speech Recognition



Language Translation



Autonomous Vehicle



Medical Diagnosis



Financial Services

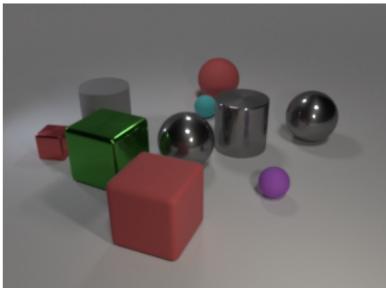


Recommendation Systems



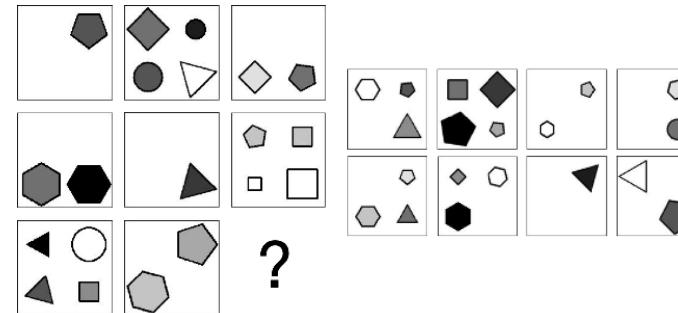
ChatGPT

But... Is That Enough?



(i) Remove all gray spheres. How many spheres are there? (3), (ii) Take away 3 cubes. How many objects are there? (7), (iii) How many blocks must be removed to get 1 block? (2)

Complex Question Answering
NN accuracy: 50%



Abstract Reasoning
NN accuracy: 53%



Interactive Learning
NN accuracy: 71%

Scenario

Imagine that a stranger will give Hank one thousand dollars to break all the windows in his neighbor's house without his neighbor's permission. Hank carries out the stranger's request.

Imagine that there are five people who are waiting in line to use a single-occupancy bathroom at a concert venue. Someone at the back of the line needs to throw up immediately. That person skips to the front of the line instead of waiting in the back.

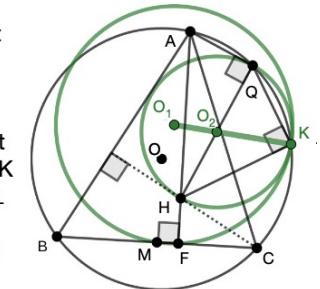
At a summer camp, there is a pool. Right next to the pool is a tent where the kids at the camp have art class. The camp made a rule that there would be no cannonballing in the pool so that the art wouldn't get ruined by the splashing water. Today, there is a bee attacking this kid, and she needs to jump into the water quickly. This kid cannonballs into the pool.



Ethical Decision Making
NN accuracy: 65%

IMO 2015 P3

"Let ABC be an acute triangle. Let (O) be its circumcircle, H its orthocenter, and F the foot of the altitude from A. Let M be the midpoint of BC. Let Q be the point on (O) such that QH \perp QA and let K be the point on (O) such that KH \perp KQ. Prove that the circumcircles (O_1) and (O_2) of triangles FKM and KQH are tangent to each other."



Automated Theorem Proving
NN accuracy: 20%

Farmer John has N cows ($2 \leq N \leq 10^5$). Each cow has a breed that is either Guernsey or Holstein. As is often the case, the cows are standing in a line, numbered $1 \dots N$ in this order.

Over the course of the day, each cow writes down a list of cows. Specifically, cow i 's list contains the range of cows starting with herself (cow i) up to and including cow E_i ($i \leq E_i \leq N$).

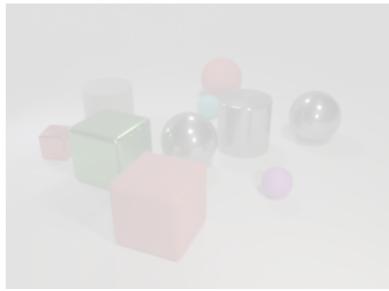
FJ has recently discovered that each breed of cow has exactly one distinct leader. FJ does not know who the leaders are, but he knows that each leader must have a list that includes all the cows of their breed, or the other breed's leader (or both).

Help FJ count the number of pairs of cows that could be leaders. It is guaranteed that there is at least one possible pair.

Problem

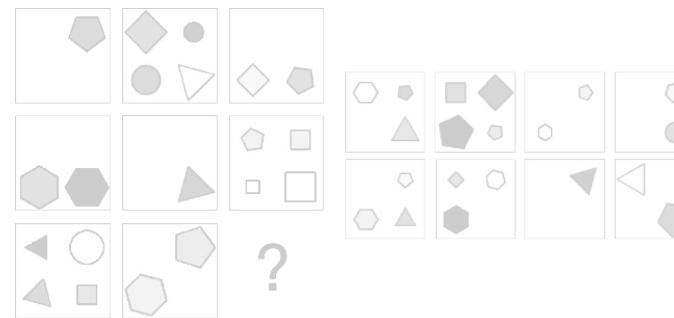
Competitive Programming
NN accuracy: 8.7%

But... Is That Enough?



Complex Question Answering
NN accuracy: 50%

- (i) Remove all gray spheres. How many spheres are there? (3), (ii) Take away 3 cubes. How many objects are there? (7), (iii) How many blocks must be removed to get 1 block? (2)



Abstract Reasoning
NN accuracy: 13%



Interactive Learning
NN accuracy: 71%

Neuro-Symbolic AI

Scenario

Imagine that a stranger will give Hank one thousand dollars to break all the windows in his neighbor's house without his neighbor's permission. Hank carries out the stranger's request.

Imagine that there are five people who are waiting in line to use a single-occupancy bathroom at a concert venue. Someone at the back of the line needs to throw up immediately. That person skips to the front of the line instead of waiting in the back.

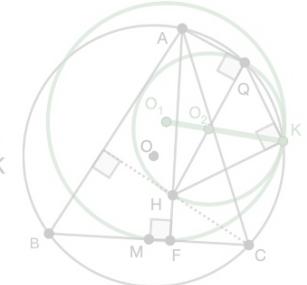
At a summer camp, there is a pool. Right next to the pool is a tent where the kids at the camp have art class. The camp made a rule that there would be no cannonballing in the pool so that the art wouldn't get ruined by the splashing water. Today, there is a bee attacking this kid, and she needs to jump into the water quickly. This kid cannonballs into the pool.



Ethical Decision Making
NN accuracy: 65%

IMO 2015 P3

"Let ABC be an acute triangle. Let (O) be its circumcircle, H its orthocenter, and F the foot of the altitude from A. Let M be the midpoint of BC. Let Q be the point on (O) such that QH \perp QA and let K be the point on (O) such that KH \perp KQ. Prove that the circumcircles (O_1) and (O_2) of triangles FKM and KQH are tangent to each other."



Automated Theorem Proving
NN accuracy: 20%

Farmer John has N cows ($2 \leq N \leq 10^5$). Each cow has a breed that is either Guernsey or Holstein. As is often the case, the cows are standing in a line, numbered $1 \dots N$ in this order.

Over the course of the day, each cow writes down a list of cows. Specifically, cow i 's list contains the range of cows starting with herself (cow i) up to and including cow E_i ($i \leq E_i \leq N$).

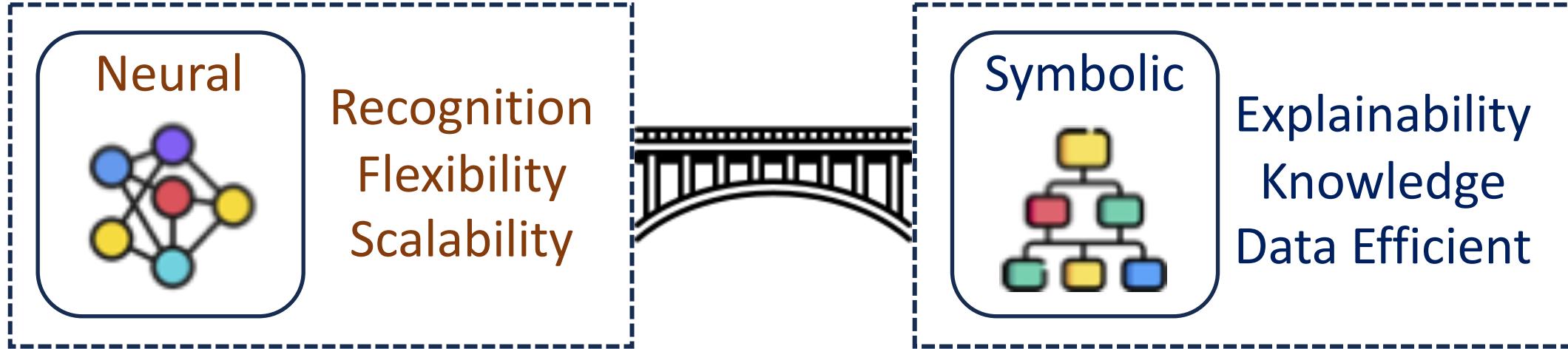
FJ has recently discovered that each breed of cow has exactly one distinct leader. FJ does not know who the leaders are, but he knows that each leader must have a list that includes all the cows of their breed, or the other breed's leader (or both).

Help FJ count the number of pairs of cows that could be leaders. It is guaranteed that there is at least one possible pair.

Problem

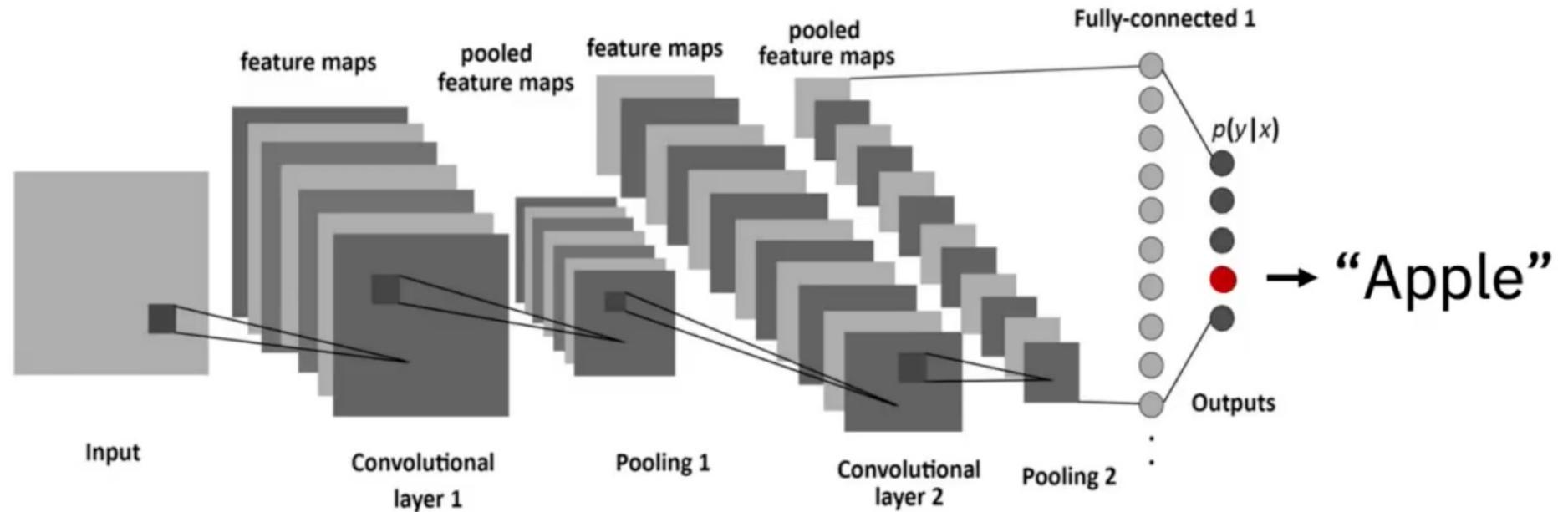
Competitive Programming
NN accuracy: 8.7%

What is Neuro-Symbolic AI?



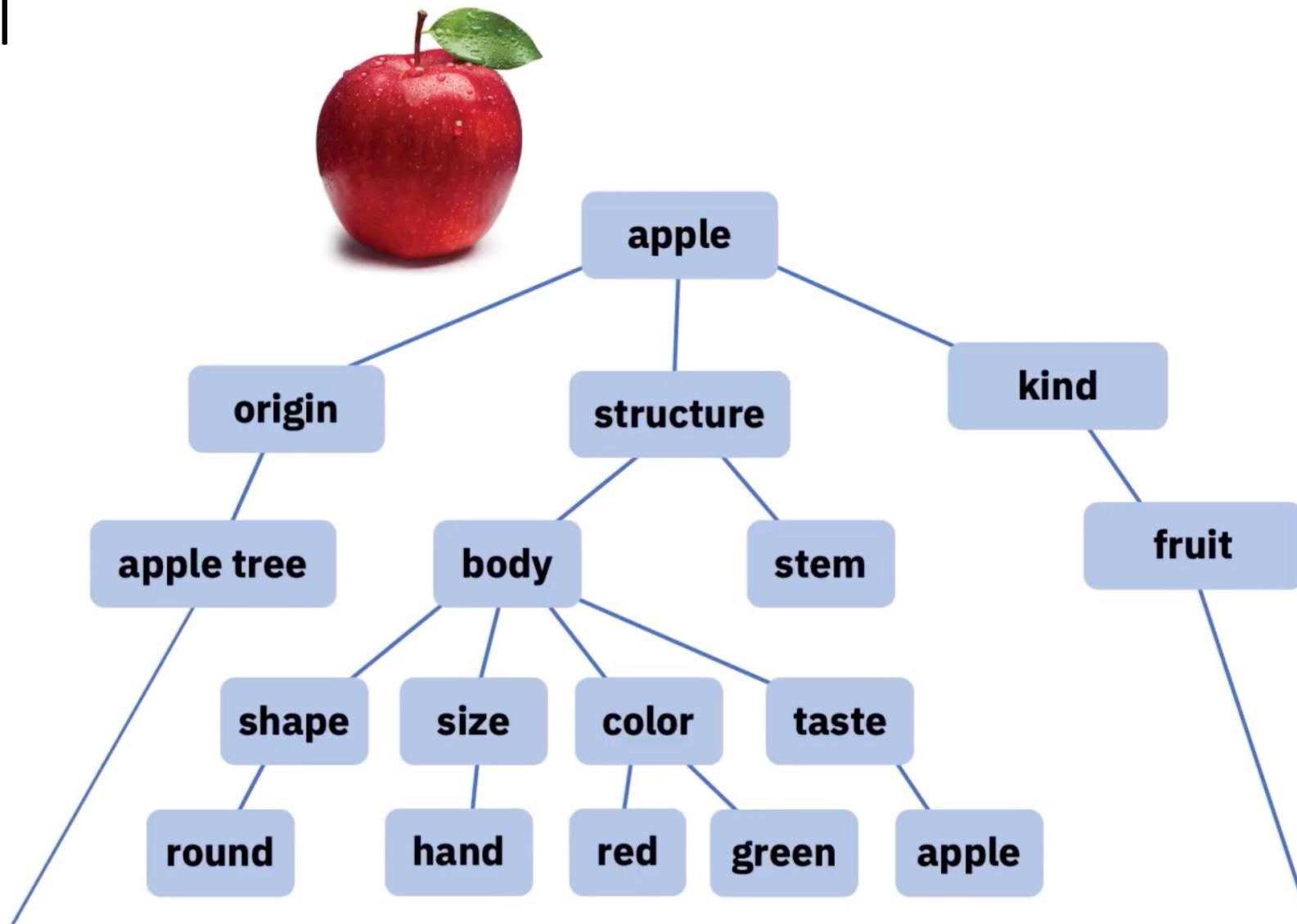
Towards Cognitive and Trustworthy AI Systems

Neural Network



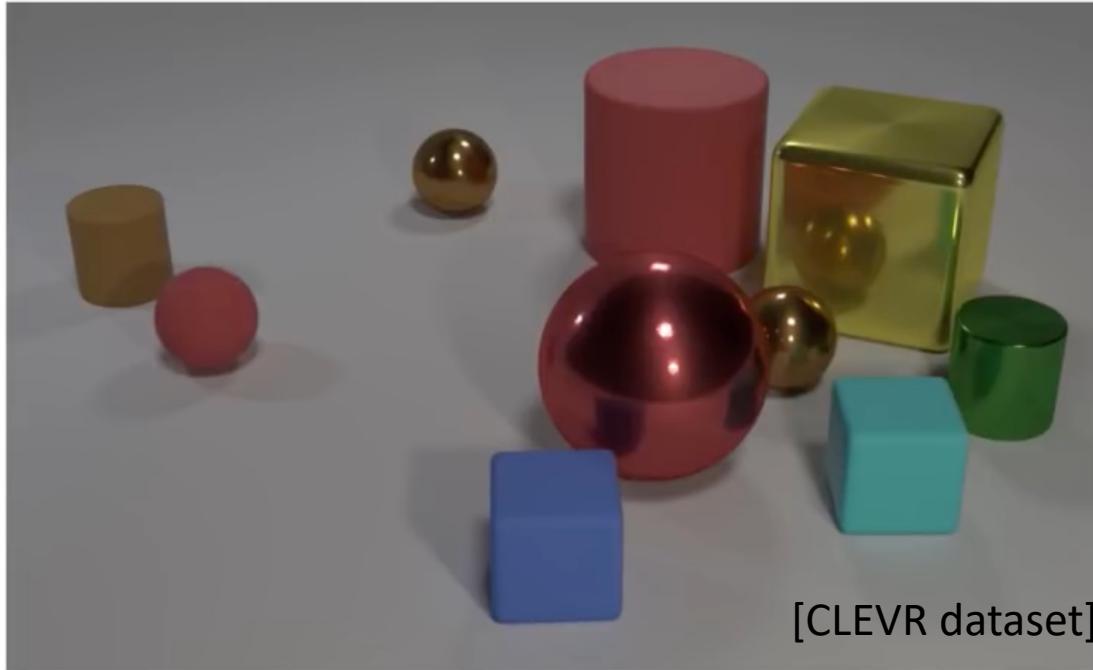
Slide Adapted from MIT 6.S191: Neurosymbolic AI

Symbolic AI



Slide Adapted from MIT 6.S191: Neurosymbolic AI

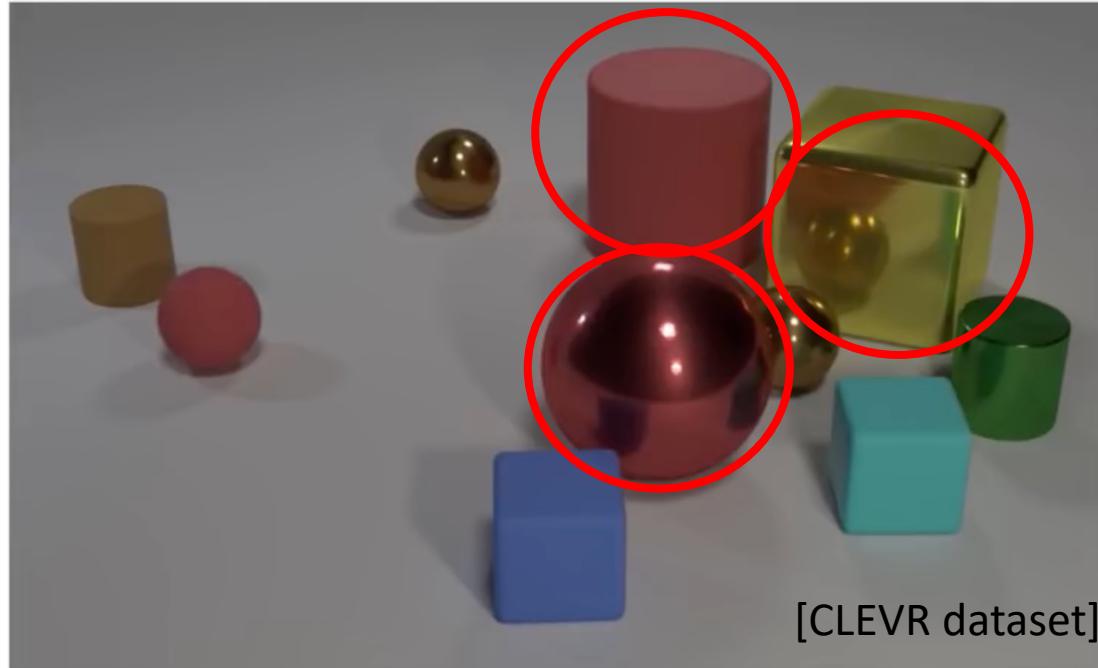
Neuro-Symbolic AI Example: Visual Reasoning



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

Slide Adapted from MIT 6.S191: Neurosymbolic AI

Neuro-Symbolic AI Example: Visual Reasoning



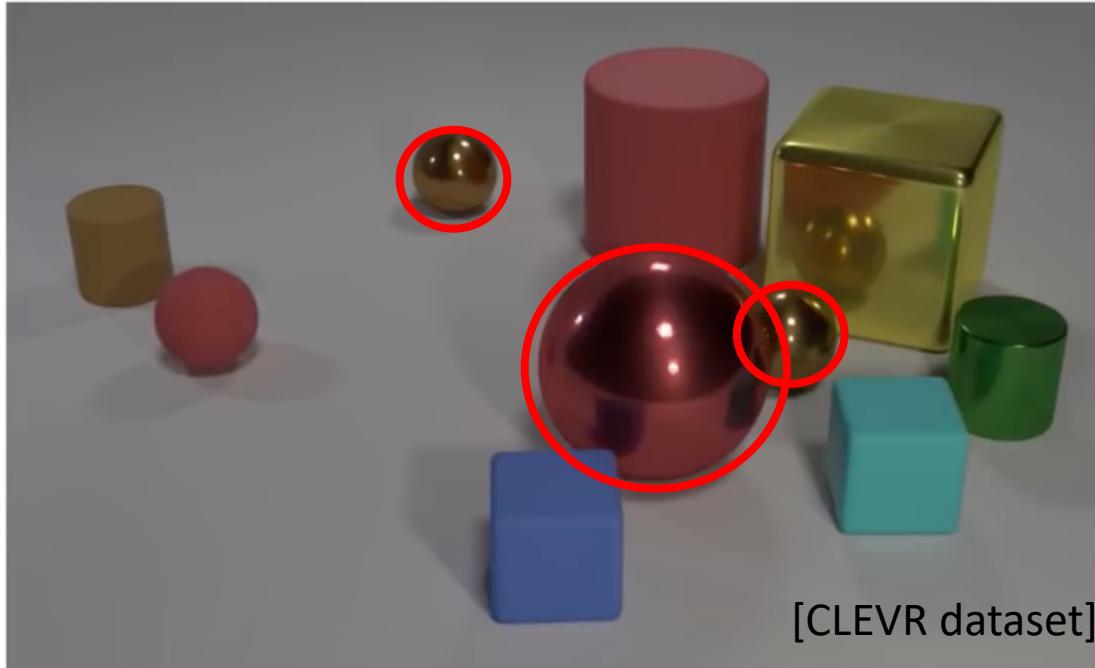
Question: *Are there an equal number of large things and metal spheres?*

3 large
things!



Slide Adapted from MIT 6.S191: Neurosymbolic AI

Neuro-Symbolic AI Example: Visual Reasoning



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

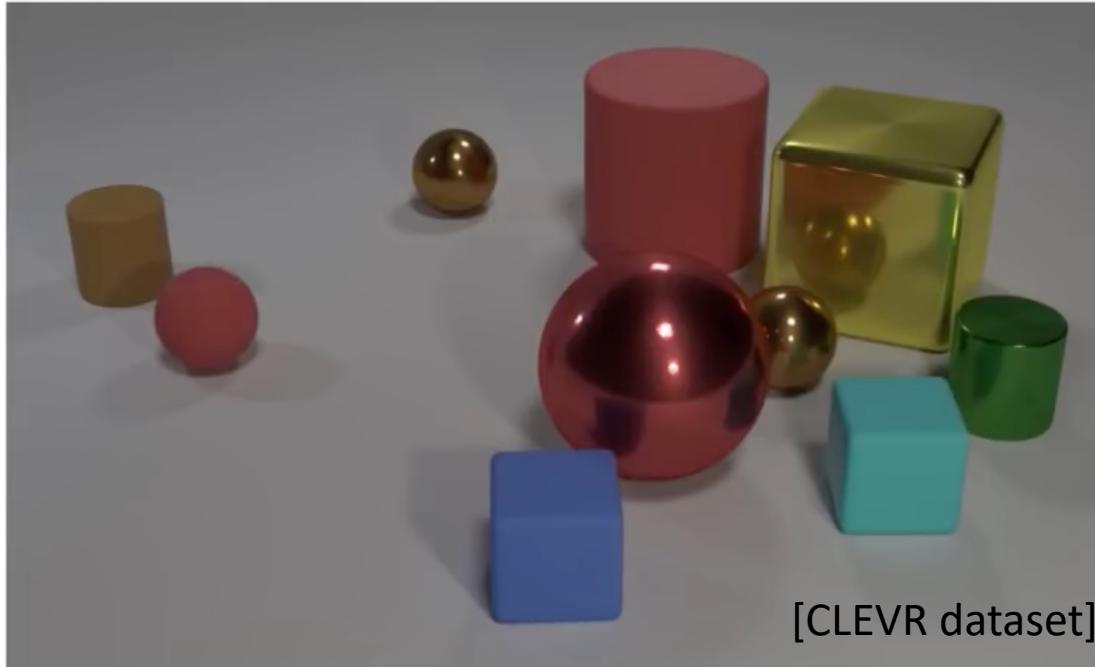
3 large things!

3 metal spheres!

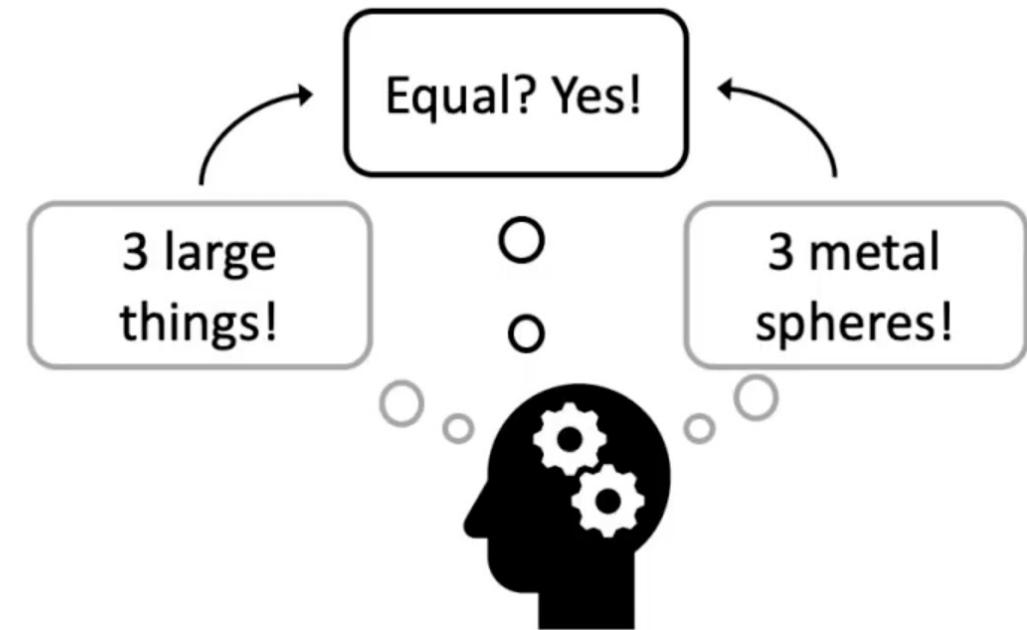


Slide Adapted from MIT 6.S191: Neurosymbolic AI

Neuro-Symbolic AI Example: Visual Reasoning

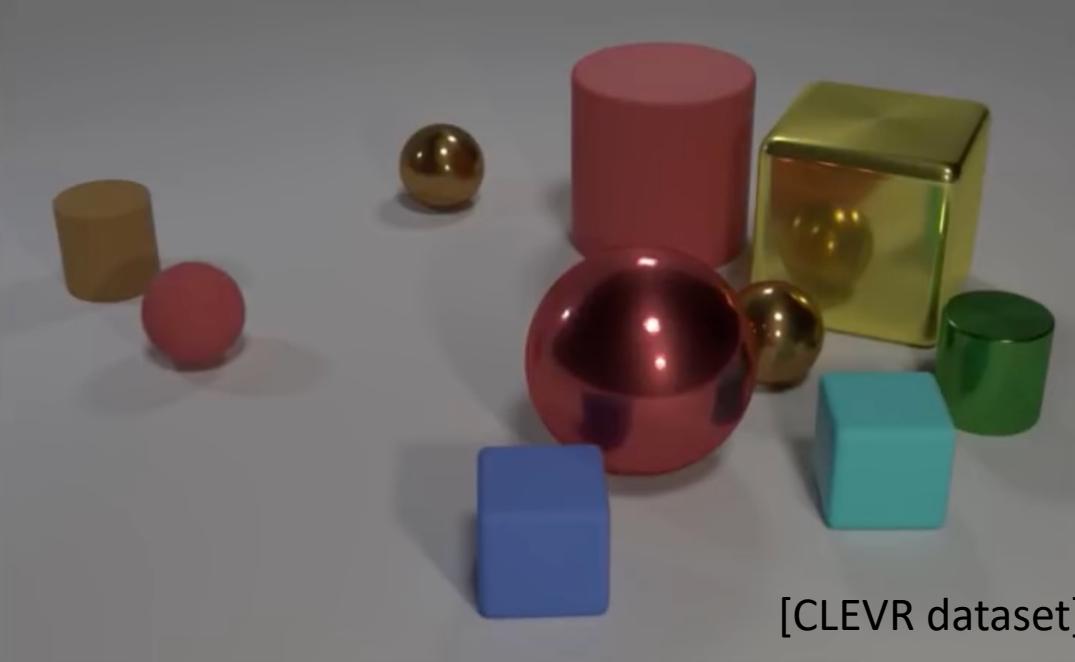


Question: *Are there an equal number of large things and metal spheres?*



Slide Adapted from MIT 6.S191: Neurosymbolic AI

Neuro-Symbolic AI Example: Visual Reasoning

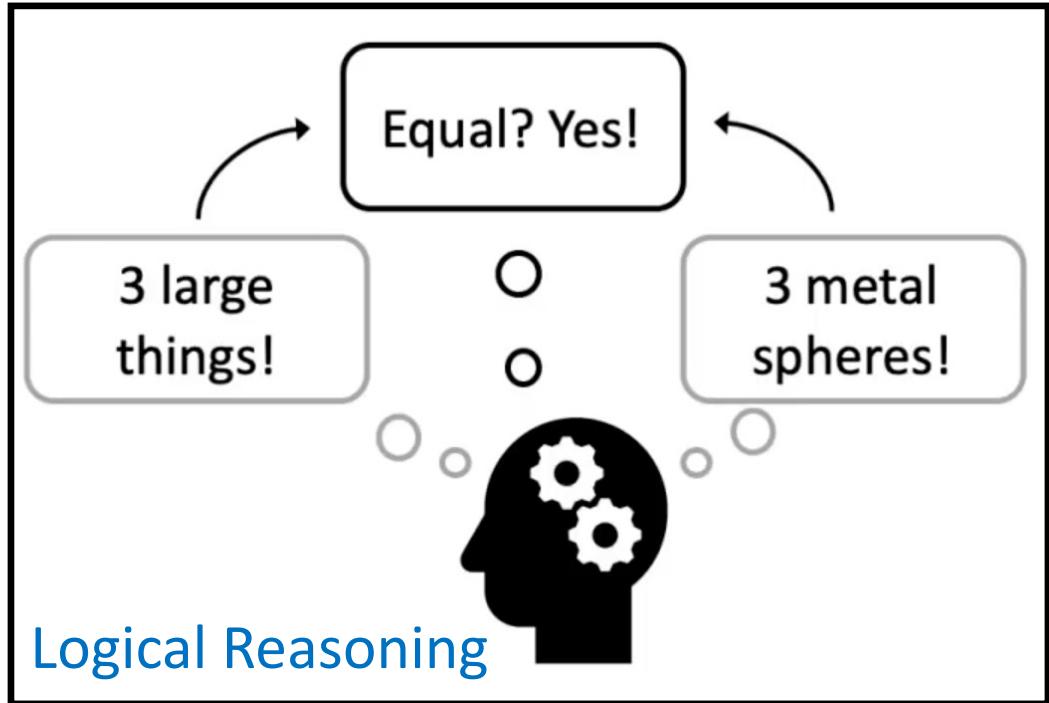


[CLEVR dataset]

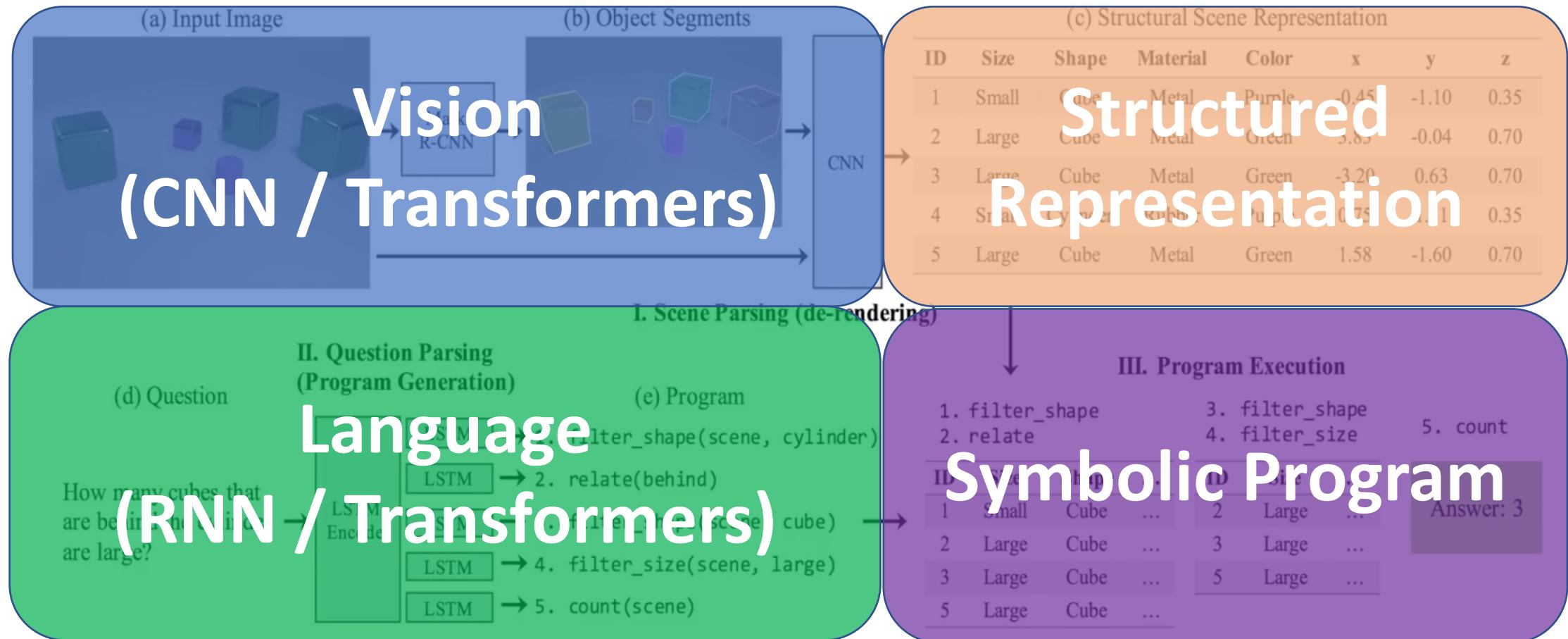
Visual Perception

Question Understanding

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



Neuro-Symbolic AI Example: Visual Reasoning



Other Examples

≡ Google DeepMind

AlphaGeometry: An Olympiad-level AI system for geometry

17 JANUARY 2024

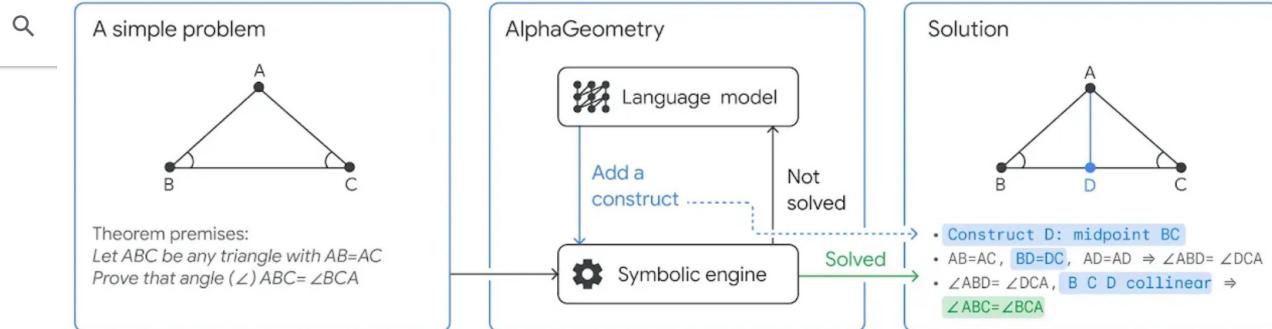
Trieu Trinh and Thang Luong

Share



AlphaGeometry adopts a neuro-symbolic approach

AlphaGeometry is a neuro-symbolic system made up of a neural language model and a symbolic deduction engine, which work together to find proofs for complex geometry theorems. Akin to the idea of "[thinking, fast and slow](#)", one system provides fast, "intuitive" ideas, and the other, more deliberate, rational decision-making.



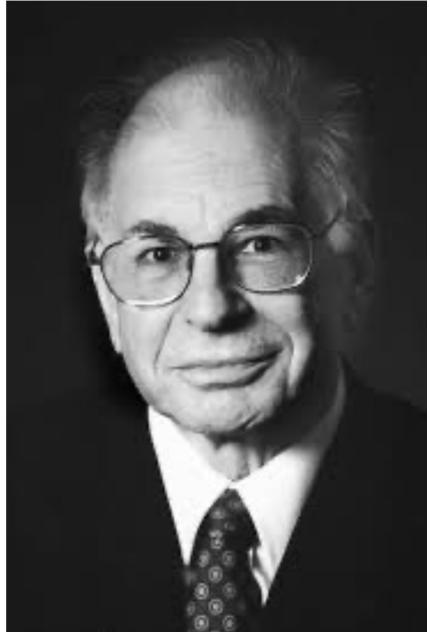
LLM: construct auxiliary points and lines
Symbolic: deductive reasoning

Eval on 30 Int. Math Olympics (IMO) problems:

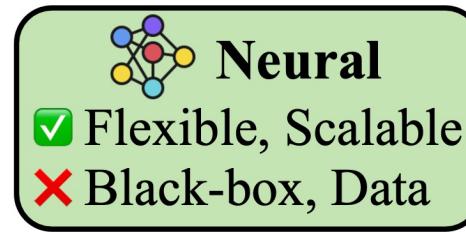
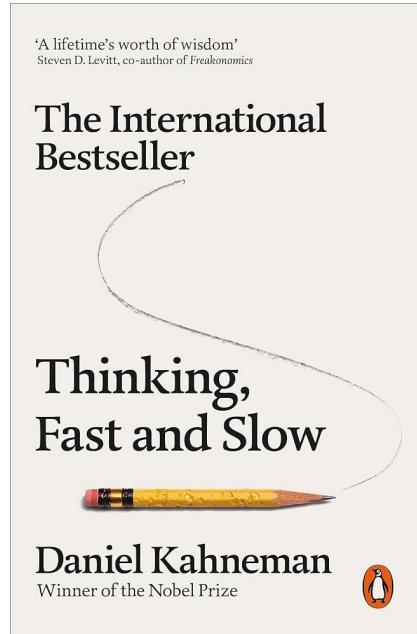
- GPT-4: 0/30
- AlphaGeometry (Neuro-Symbolic): 25/30
- Human Gold Medalist: 26/30

Trinh et al, "Solving Olympiad Geometry without Human Demonstrations", Nature 2024

Relationship to Human Minds

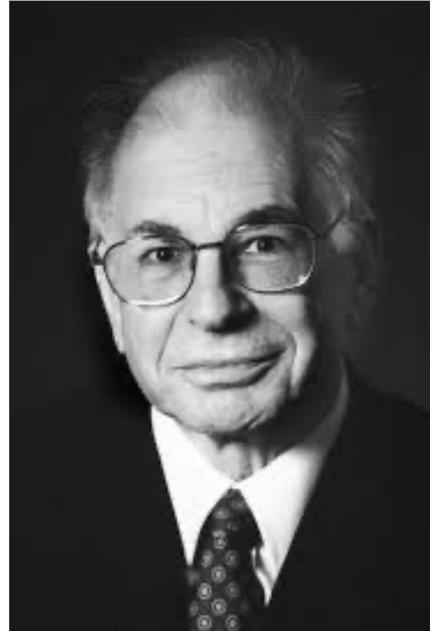


Daniel Kahneman
(1934-2024)

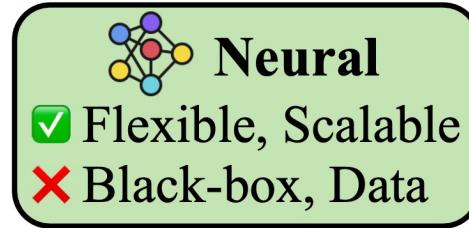
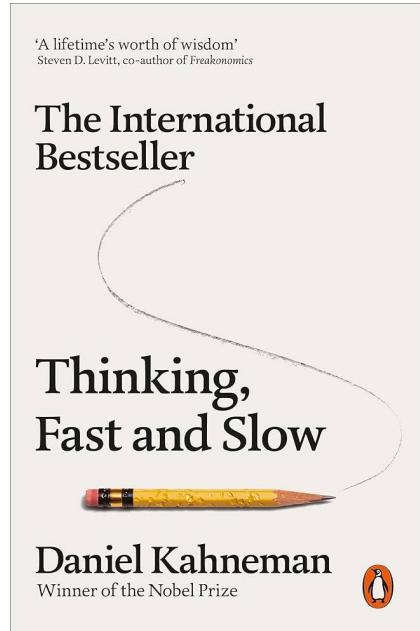


*System 1: thinking fast
(intuitive perception)*

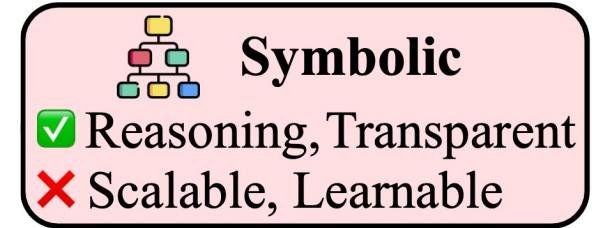
Relationship to Human Minds



Daniel Kahneman
(1934-2024)



*System 1: thinking fast
(intuitive perception)*

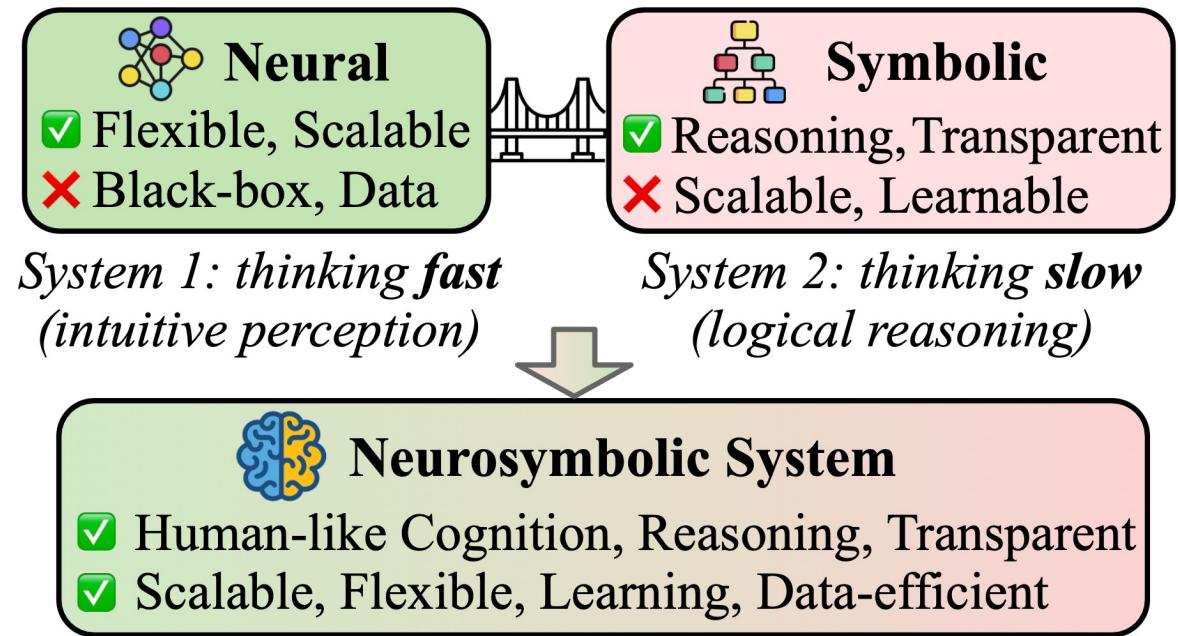
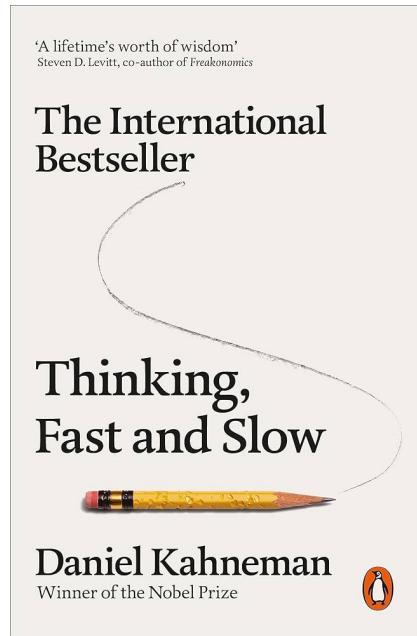


*System 2: thinking slow
(logical reasoning)*

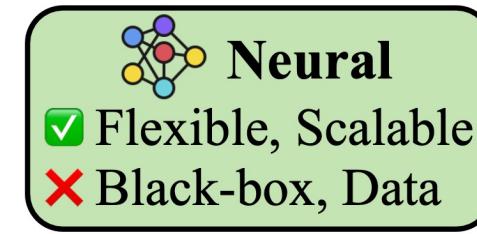
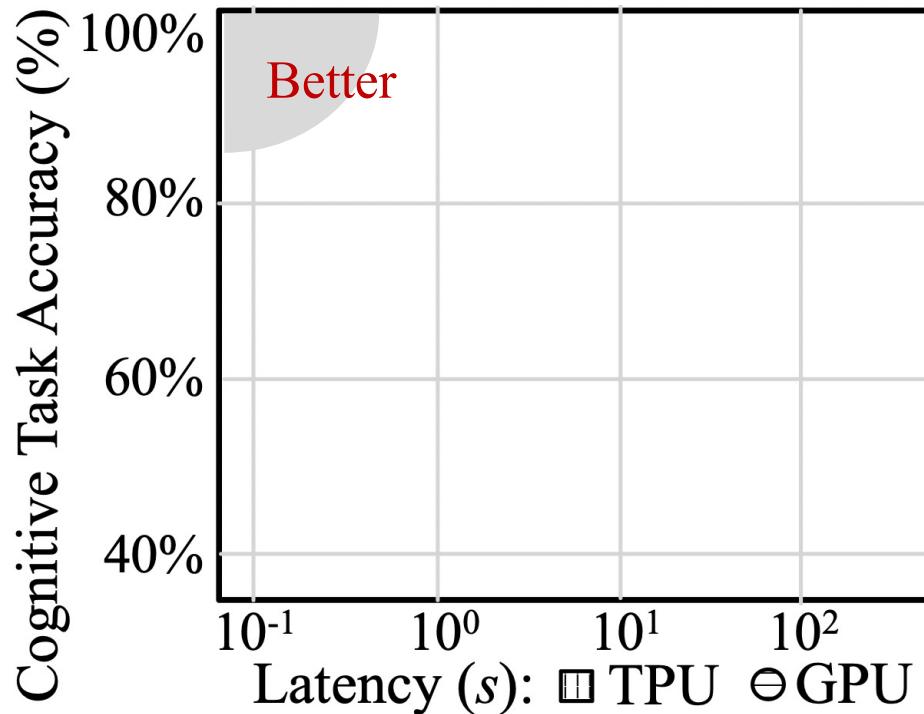
Relationship to Human Minds



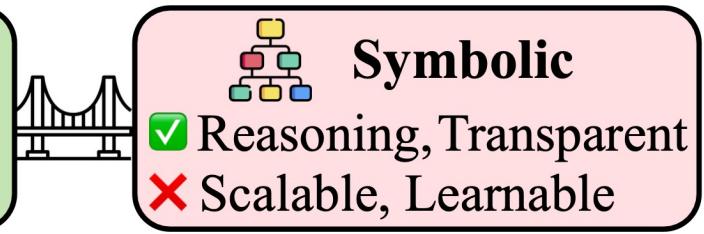
Daniel Kahneman
(1934-2024)



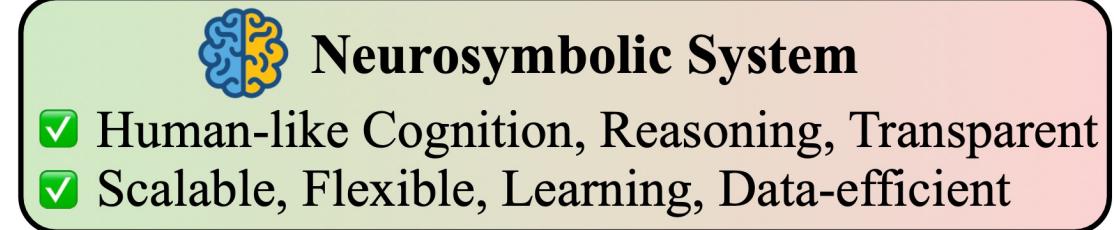
However.. From Computing Perspective



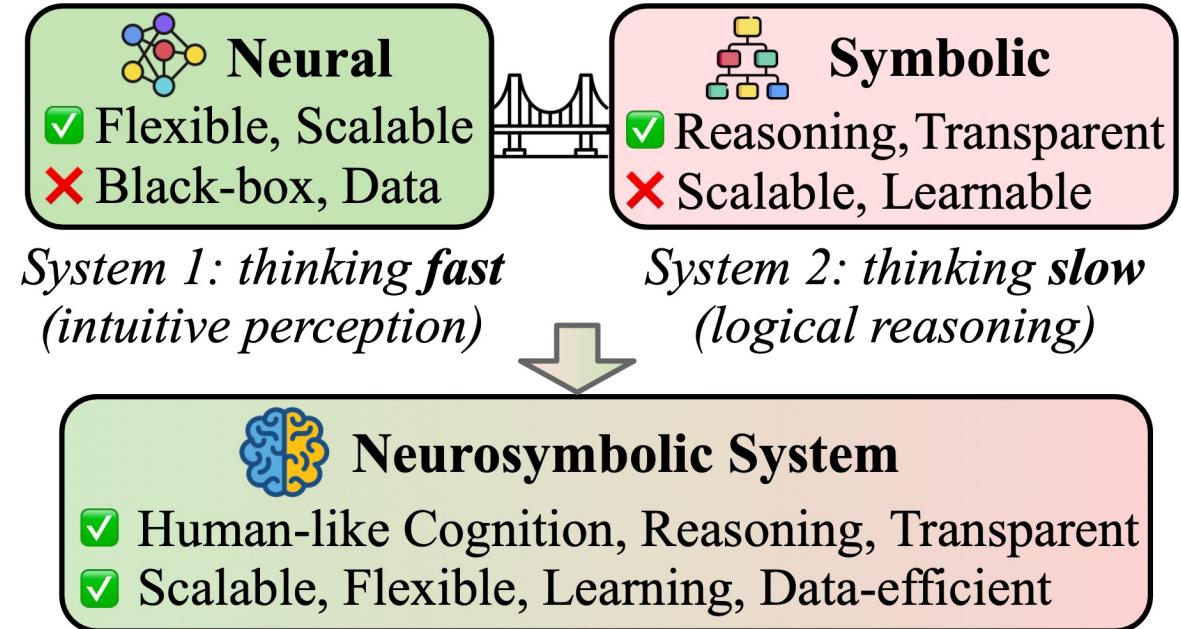
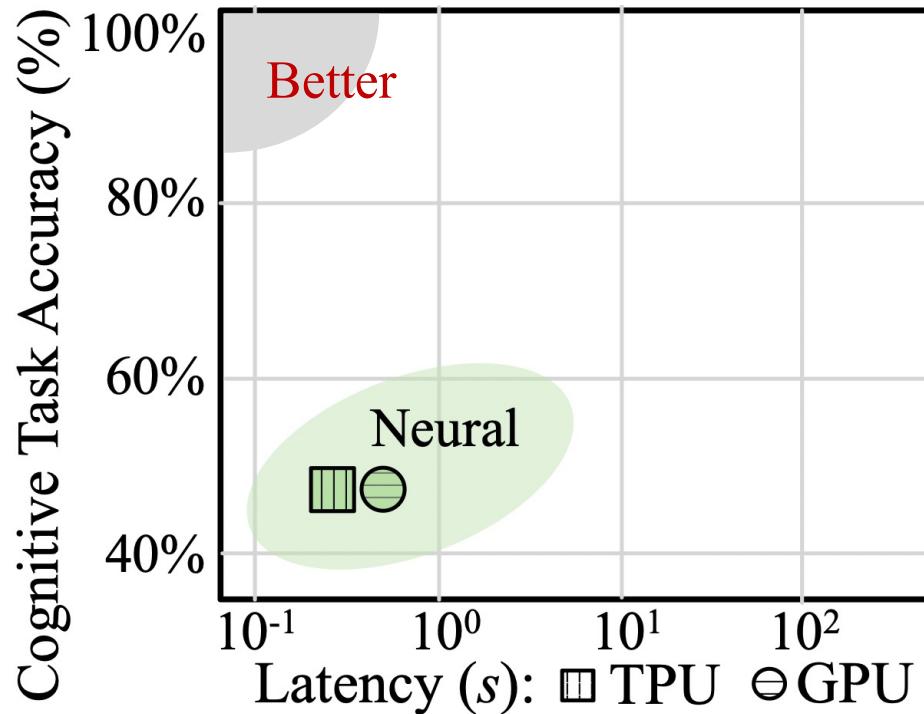
*System 1: thinking fast
(intuitive perception)*



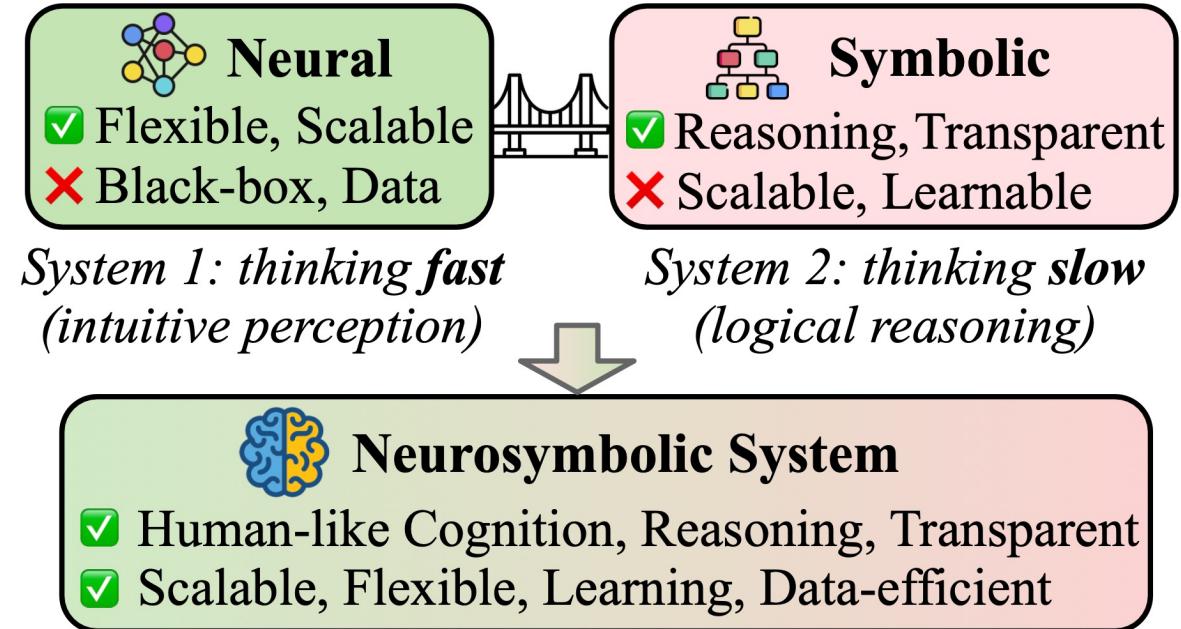
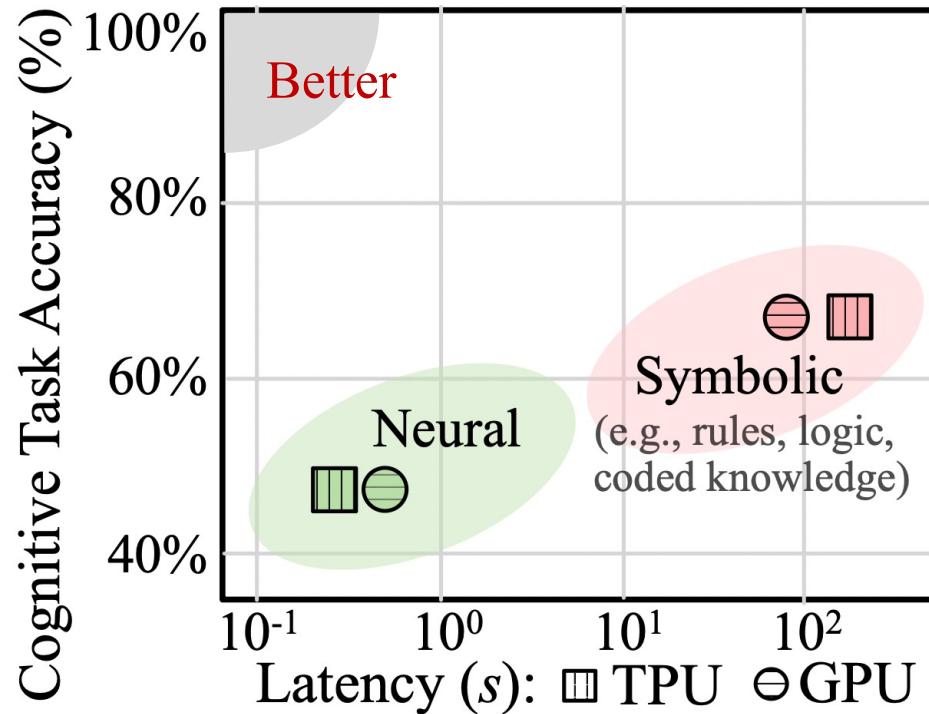
*System 2: thinking slow
(logical reasoning)*



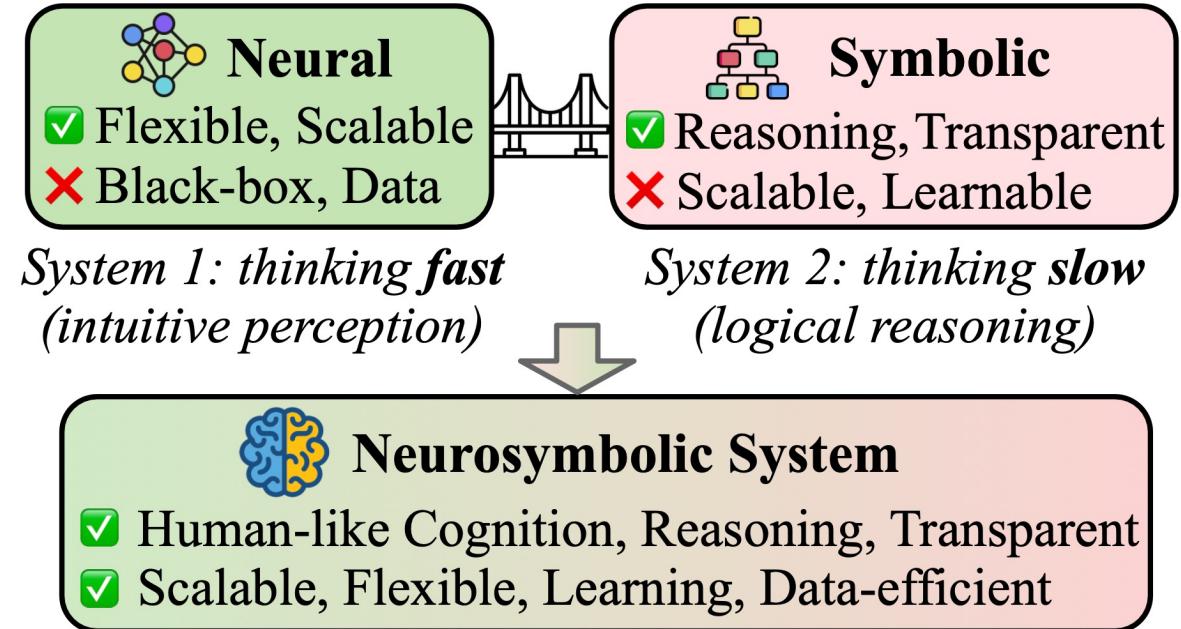
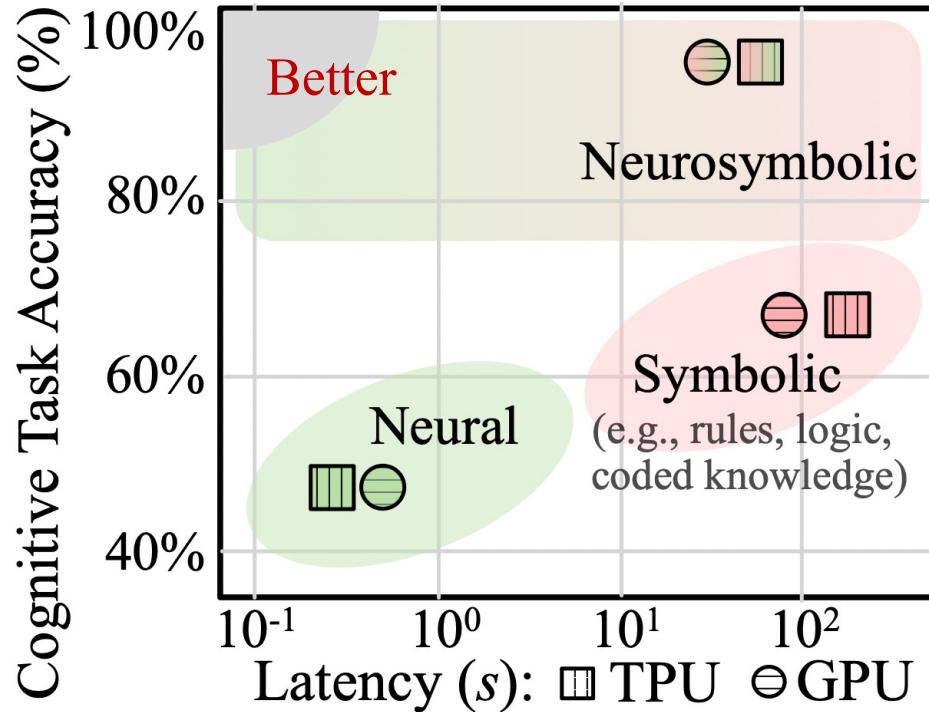
However.. From Computing Perspective



However.. From Computing Perspective



However.. From Computing Perspective

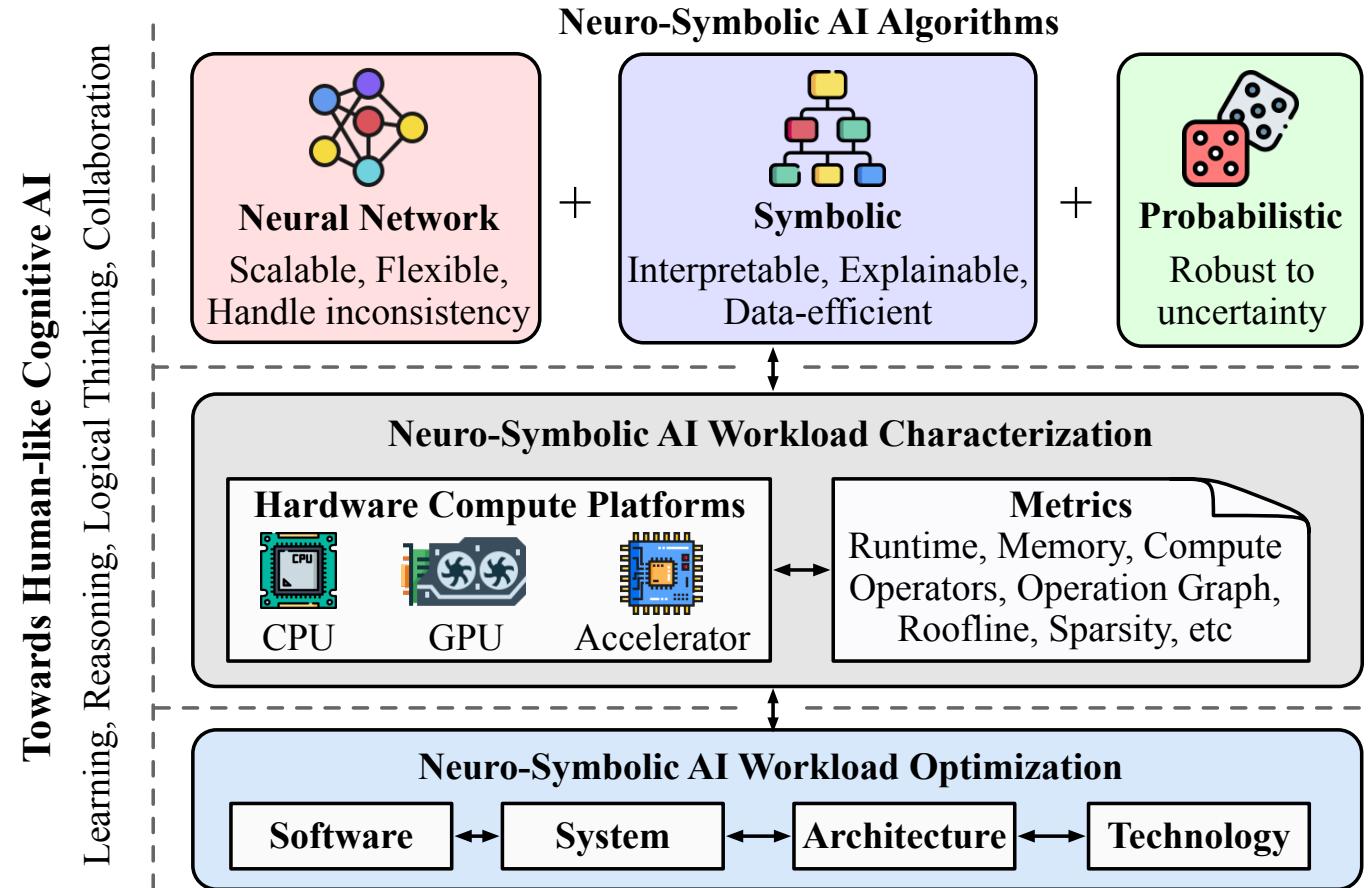


This talk: Demystify Neuro-Symbolic AI for SW/HW Co-Design

Characterize Neuro-Symbolic Workloads

Identify Potential Inefficiency Reasons

Optimize Neuro-Symbolic Systems via Co-Design

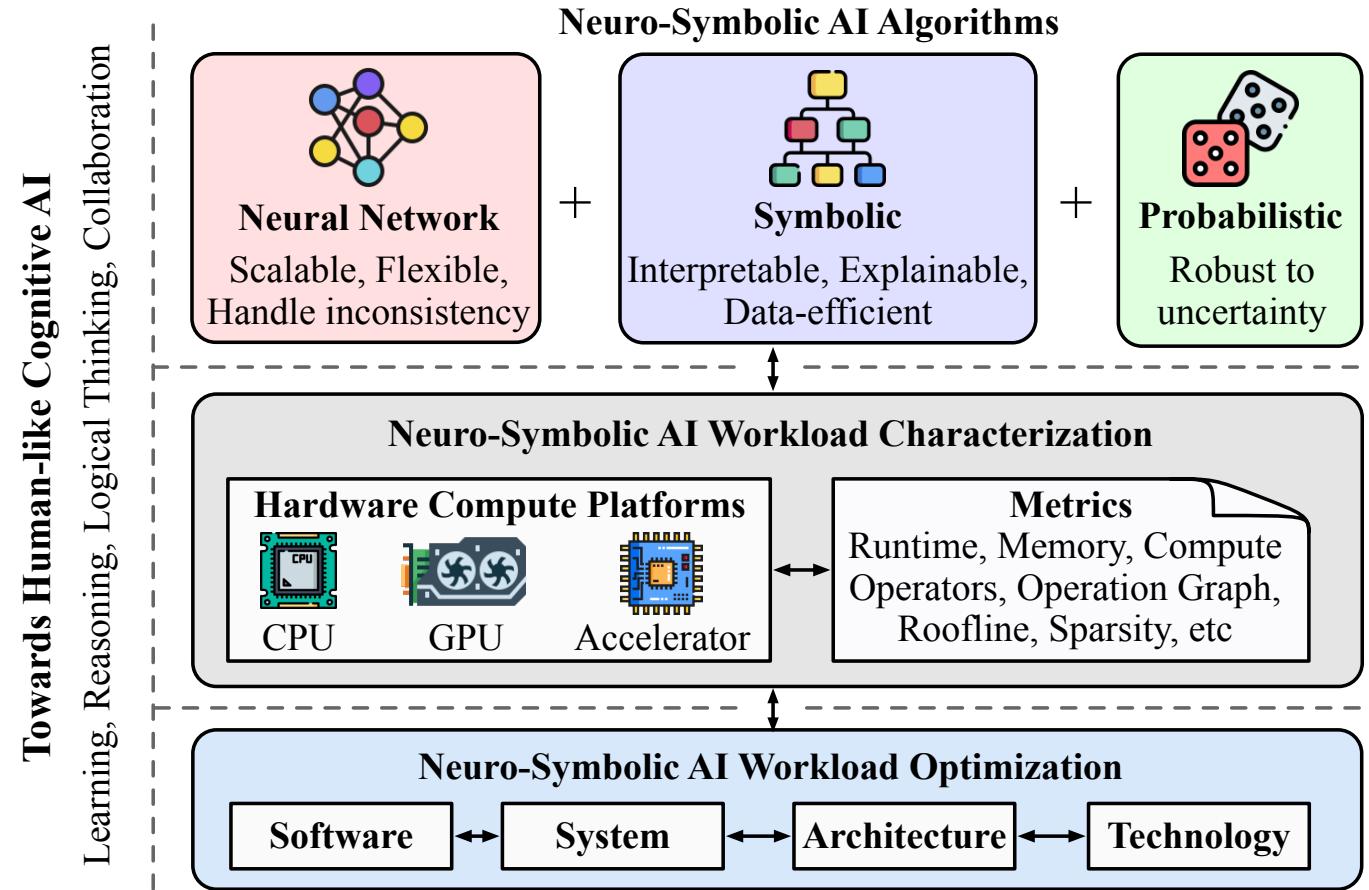


This talk: Demystify Neuro-Symbolic AI for SW/HW Co-Design

Characterize Neuro-Symbolic Workloads

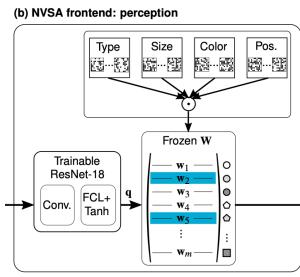
Identify Potential Inefficiency Reasons

Optimize Neuro-Symbolic Systems via Co-Design

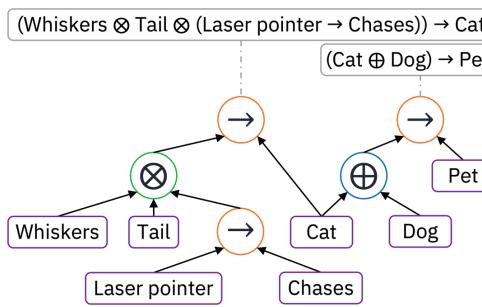
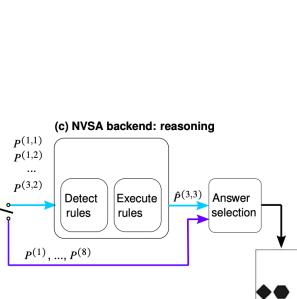


Zishen Wan, Che-Kai Liu, Hanchen Yang, Ritik Raj, Chaojian Li, Haoran You, Yonggan Fu, Cheng Wan, Ananda Samajdar, Celine Lin, Tushar Krishna, Arijit Raychowdhury, "Workload and Characterization of Neuro-Symbolic AI", in *ISPASS 2024*

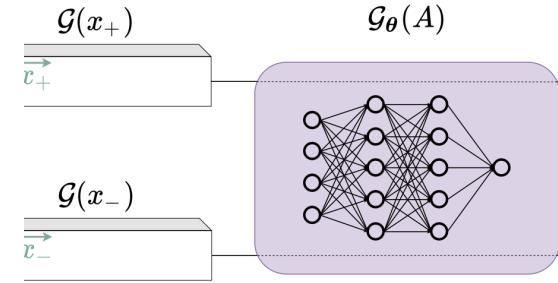
Lots of Neuro-Symbolic Algorithms



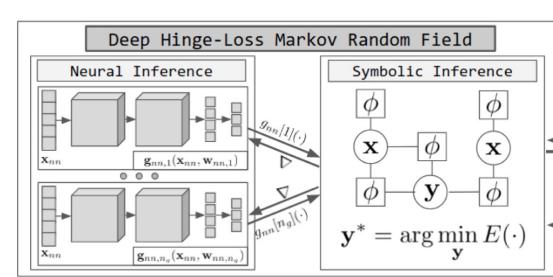
Neuro-Vector-Symbolic Arch



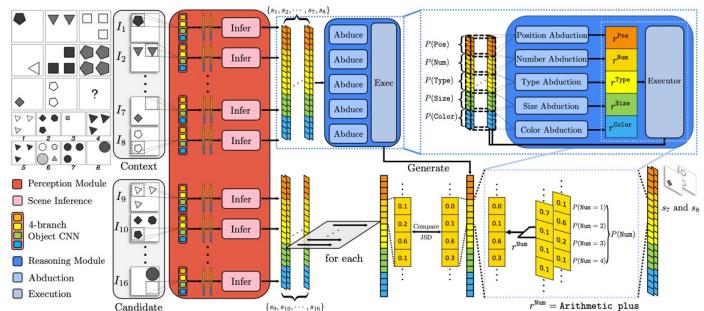
Logical Neural Network



Logical Tensor Network



Neural Probabilistic Soft Logic



Probabilistic Abduction

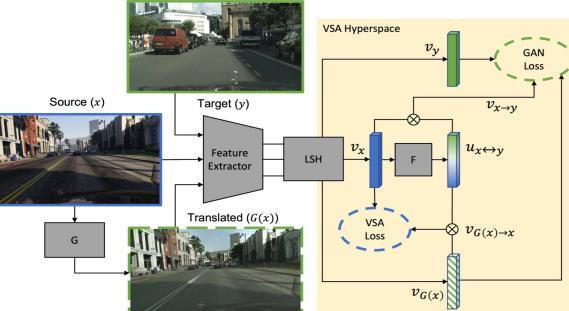
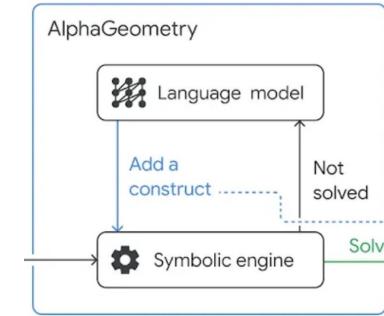
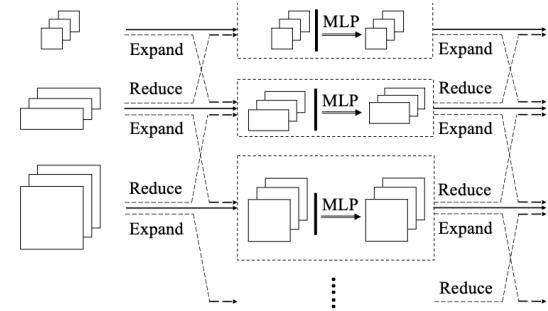


Image Translation via VSA



AlphaGeometry



Neural Logical Machine

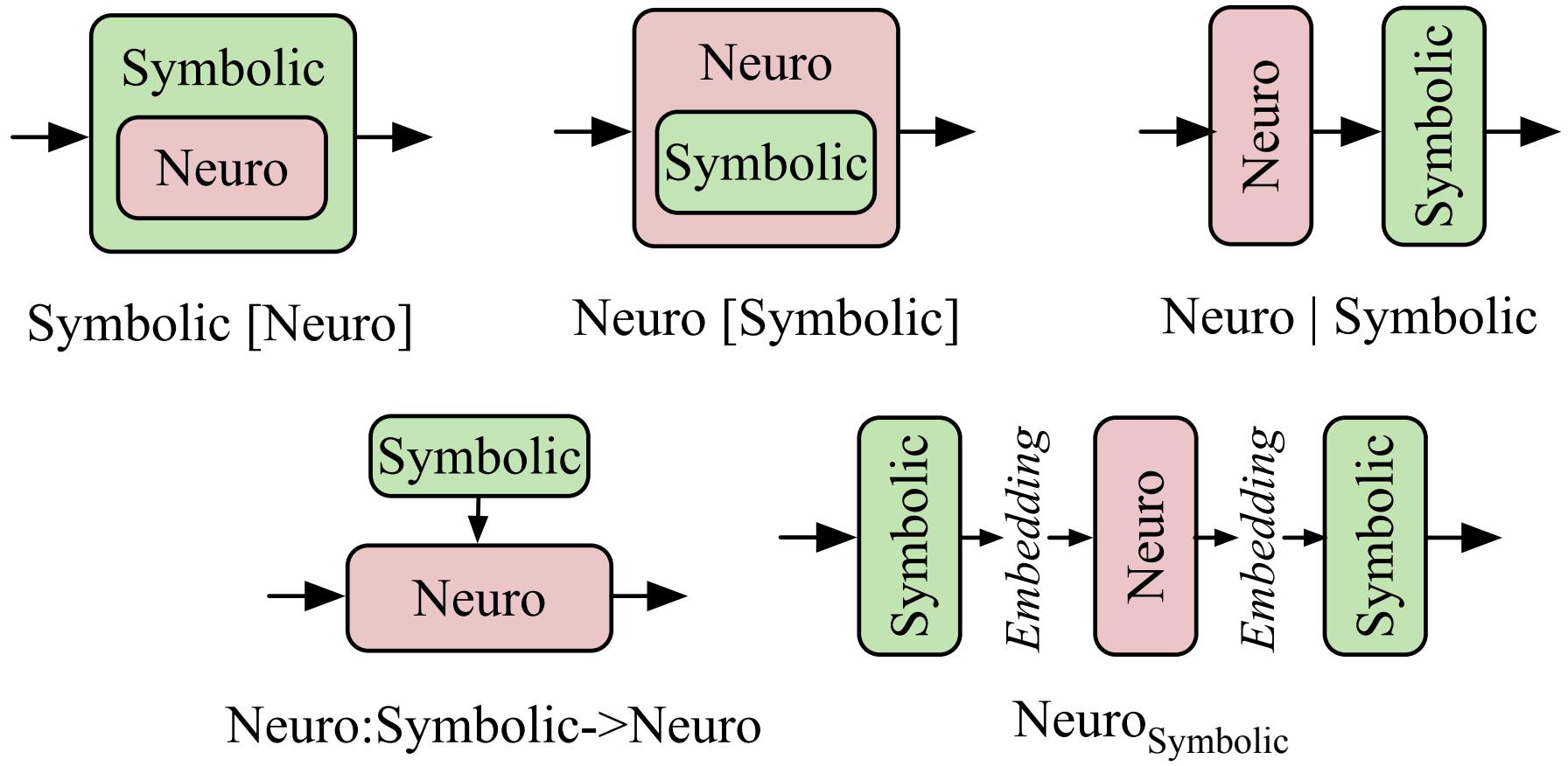
Neuro

MLP, ConvNet, Transformer, etc

Symbolic

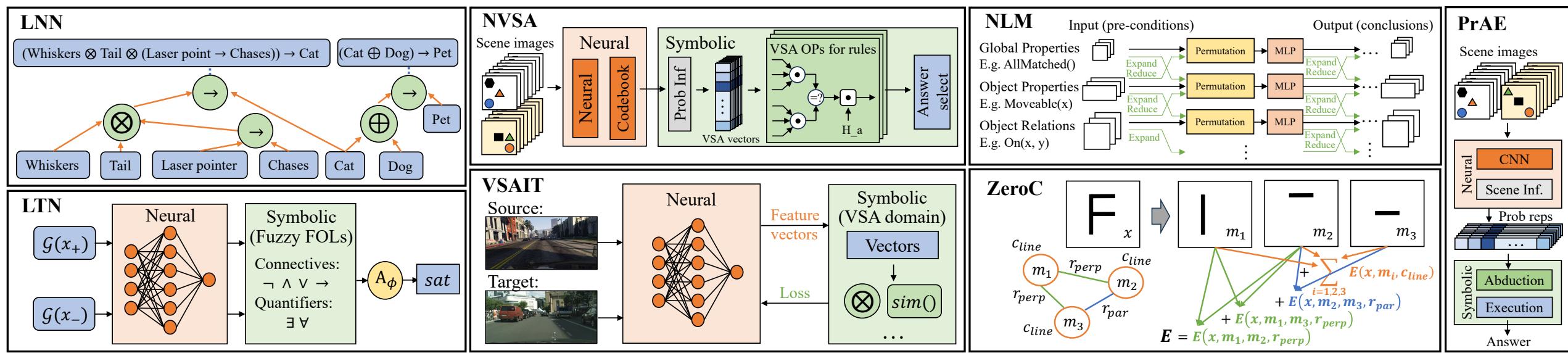
Vector, Fuzzy logic, Knowledge graph, Decision tree, etc

Neuro-Symbolic AI Workload Category

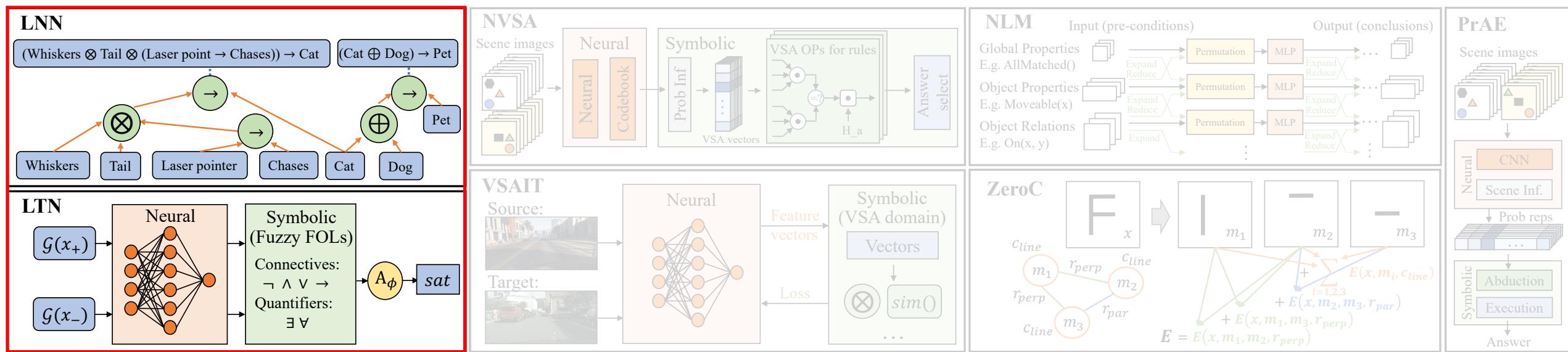


Inspired by Henry Kautz's terminology

Selected Neuro-Symbolic Workloads

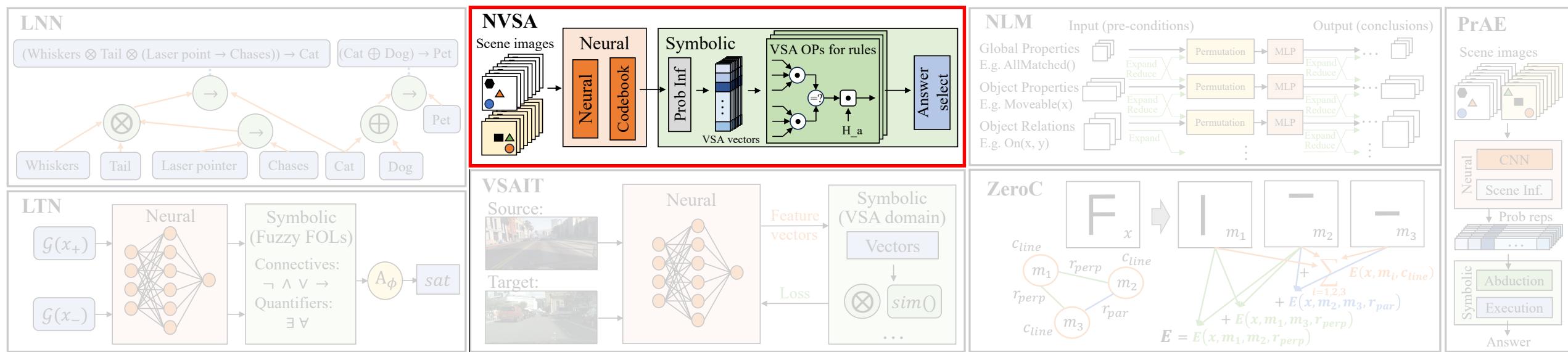


Selected Neuro-Symbolic Workloads



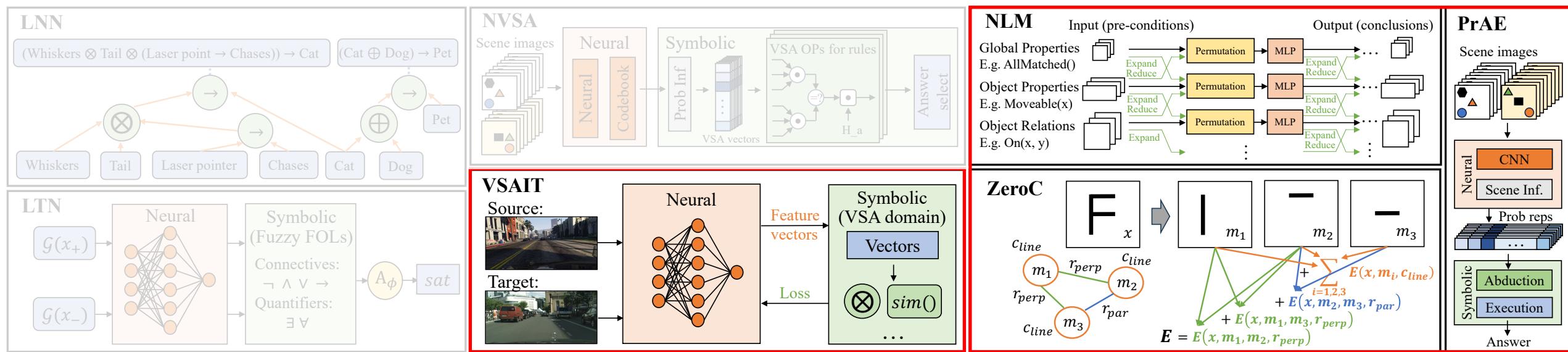
Representative Neuro-Symbolic AI Workloads	Logic Neural Network [30]	Logic Tensor Network [34]
Abbreviation	LNN	LTN
Neuro-Symbolic Category	Neuro:Symbolic \rightarrow Neuro	NeuroSymbolic
Learning Approach	Supervised	Supervised/Unsupervised
Deployment Scenario	Application	Learning and reasoning, Full theorem prover
	Advantage vs. Neural Model	Higher interoperability, resilience to incomplete knowledge, generalization
	Dataset	LUBM benchmark [40], TPTP benchmark [41]
Computation Pattern	Datatype	FP32
	Neuro	Graph
	Symbolic	FOL/Logical operation

Selected Neuro-Symbolic Workloads



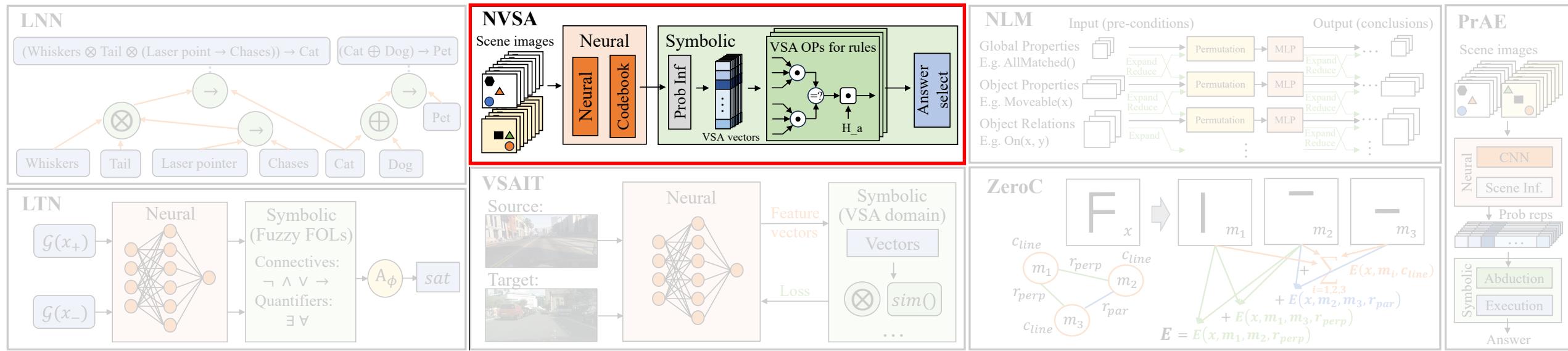
Representative Neuro-Symbolic AI Workloads	Logic Neural Network [30]	Logic Tensor Network [34]	Neuro-Vector-Symbolic Architecture [4]
Abbreviation	LNN	LTN	NVSA
Neuro-Symbolic Category	Neuro:Symbolic→Neuro	NeuroSymbolic	Neuro Symbolic
Learning Approach	Supervised	Supervised/Unsupervised	Supervised/Unsupervised
Deployment Scenario	Application	Learning and reasoning, Full theorem prover	Querying, learning, reasoning (relational and embedding learning, query answering)
	Advantage vs. Neural Model	Higher interoperability, resilience to incomplete knowledge, generalization	Higher data efficiency, comprehensibility, out-of-distribution generalization
	Dataset	LUBM benchmark [40], TPTP benchmark [41]	UCI [42], Leptograpus crabs [43], DeepProbLog [44]
Computation Pattern	Datatype	FP32	FP32
	Neuro	Graph	MLP
	Symbolic	FOL/Logical operation	VSA/Vector operation

Selected Neuro-Symbolic Workloads



Representative Neuro-Symbolic AI Workloads	Logic Neural Network [30]	Logic Tensor Network [34]	Neuro-Vector-Symbolic Architecture [4]	Vector Symbolic Architecture Image2Image Translation [7]	Neural Logic Machine [38]	Zero-shot Concept Recognition and Acquisition [37]	Probabilistic Abduction and Execution [23]
Abbreviation	LNN	LTN	NVSA	VSAIT	NLM	ZeroC	PrAE
Neuro-Symbolic Category	Neuro:Symbolic→Neuro	Neuro Symbolic	Neuro Symbolic	Neuro Symbolic	Neuro[Symbolic]	Neuro[Symbolic]	Neuro Symbolic
Learning Approach	Supervised	Supervised/Unsupervised	Supervised/Unsupervised	Supervised	Supervised/Unsupervised	Supervised	Supervised/Unsupervised
Deployment Scenario	Application	Learning and reasoning, Full theorem prover	Querying, learning, reasoning (relational and embedding learning, query answering)	Fluid intelligence, Abstract reasoning	Unpaired image-to-image translation	Relational reasoning, Decision making	Cross-domain classification and detection, Concept acquisition
	Advantage vs. Neural Model	Higher interoperability, resilience to incomplete knowledge, generalization	Higher data efficiency, comprehensibility, out-of-distribution generalization	Higher joint representations efficiency, abstract reasoning capability, transparency	Address semantic flipping and hallucinations issue in unpaired image translation tasks	Higher generalization, logic reasoning, deduction, explainability capability	Higher generalization, concept acquisition and recognition, compositionality capability
	Dataset	LUBM benchmark [40], TPTP benchmark [41]	UCI [42], Leptograpus crabs [43], DeepProbLog [44]	RAVEN [21], I-RAVEN [22], PGM [45]	GTA [47], Cityscapes [48], Google Maps dataset [49]	Family graph reasoning, sorting, path finding [46]	Abstraction reasoning [50], Hierarchical-concept corpus [51]
Computation Pattern	Datatype	FP32	FP32	FP32	FP32	INT64	FP32
	Neuro	Graph	MLP	ConvNet	ConvNet	Sequential tensor	Energy-based network
	Symbolic	FOL/Logical operation	FOL/Logical operation	VSA/Vector operation	VSA/Vector operation	FOL/Logical operation	Graph, vector operation

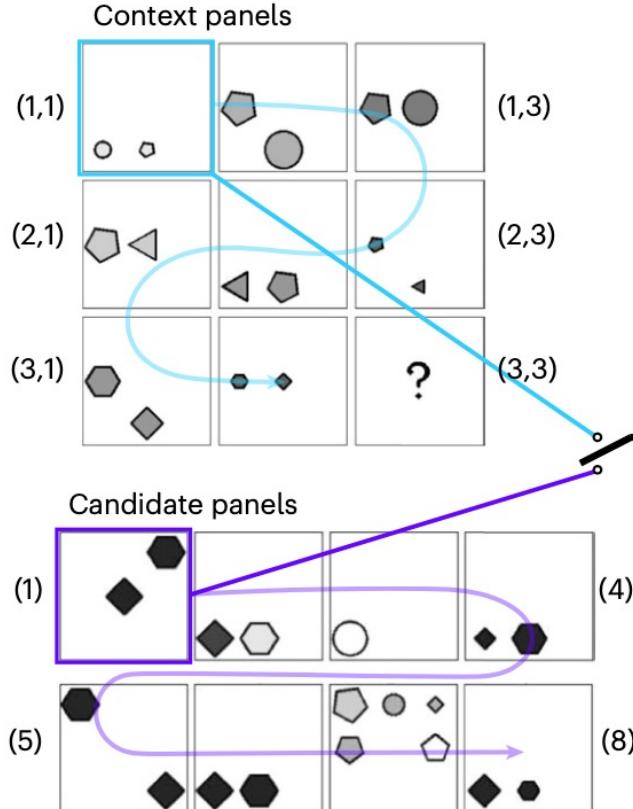
Example: Neuro-Vector-Symbolic Architecture (NVSA)



Representative Neuro-Symbolic AI Workloads	Logic Neural Network [30]	Logic Tensor Network [34]	Neuro-Vector-Symbolic Architecture [4]	Vector Symbolic Architecture Image2Image Translation [7]	Neural Logic Machine [38]	Zero-shot Concept Recognition and Acquisition [37]	Probabilistic Abduction and Execution [23]
Abbreviation	LNN	LTN	NVSA	VSAIT	NLM	ZeroC	PrAE
Neuro-Symbolic Category	Neuro:Symbolic→Neuro	Neuro Symbolic	Neuro Symbolic	Neuro Symbolic	Neuro[Symbolic]	Neuro[Symbolic]	Neuro Symbolic
Learning Approach	Supervised	Supervised/Unsupervised	Supervised/Unsupervised	Supervised	Supervised/Unsupervised	Supervised	Supervised/Unsupervised
Deployment Scenario	Application	Learning and reasoning, Full theorem prover	Querying, learning, reasoning (relational and embedding learning, query answering)	Fluid intelligence, Abstract reasoning	Unpaired image-to-image translation	Relational reasoning, Decision making	Cross-domain classification and detection, Concept acquisition
	Advantage vs. Neural Model	Higher interoperability, resilience to incomplete knowledge, generalization	Higher data efficiency, comprehensibility, out-of-distribution generalization	Higher joint representations efficiency, abstract reasoning capability, transparency	Address semantic flipping and hallucinations issue in unpaired image translation tasks	Higher generalization, logic reasoning, deduction, explainability capability	Higher generalization, concept acquisition and recognition, compositionality capability
	Dataset	LUBM benchmark [40], TPTP benchmark [41]	UCI [42], Leptograpsus crabs [43], DeepProbLog [44]	RAVEN [21], I-RAVEN [22], PGM [45]	GTA [47], Cityscapes [48], Google Maps dataset [49]	Family graph reasoning, sorting, path finding [46]	Abstraction reasoning [50], Hierarchical-concept corpus [51]
Computation Pattern	Datatype	FP32	FP32	FP32	FP32	FP32	INT64
	Neuro	Graph	MLP	ConvNet	ConvNet	Sequential tensor	Energy-based network
	Symbolic	FOL/Logical operation	FOL/Logical operation	VSA/Vector operation	VSA/Vector operation	FOL/Logical operation	Graph, vector operation

Example: Neuro-Vector-Symbolic Architecture

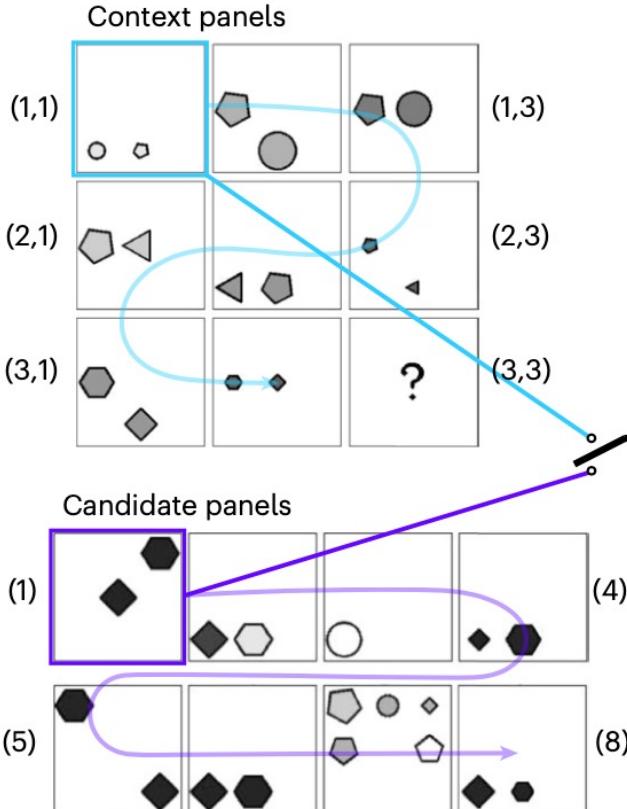
RAVEN example test



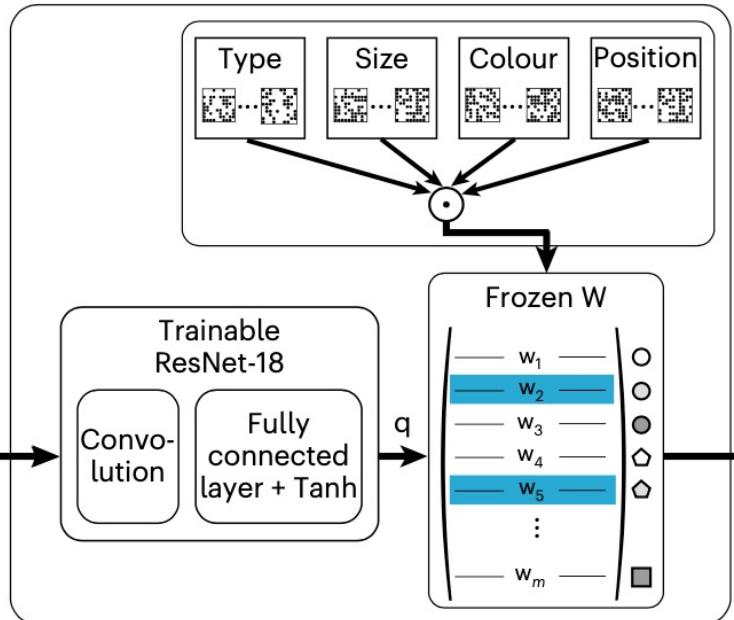
Hersche, et al. "A neuro-vector-symbolic architecture for solving Raven's progressive matrices". In Nature Machine Intelligence, 2023

Example: Neuro-Vector-Symbolic Architecture

RAVEN example test



NVSA frontend: perception

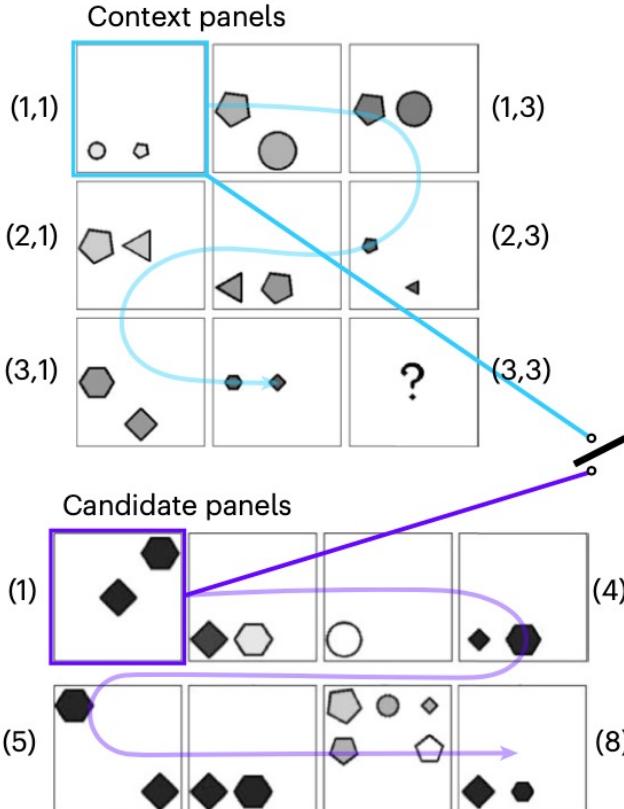


Neuro Perception

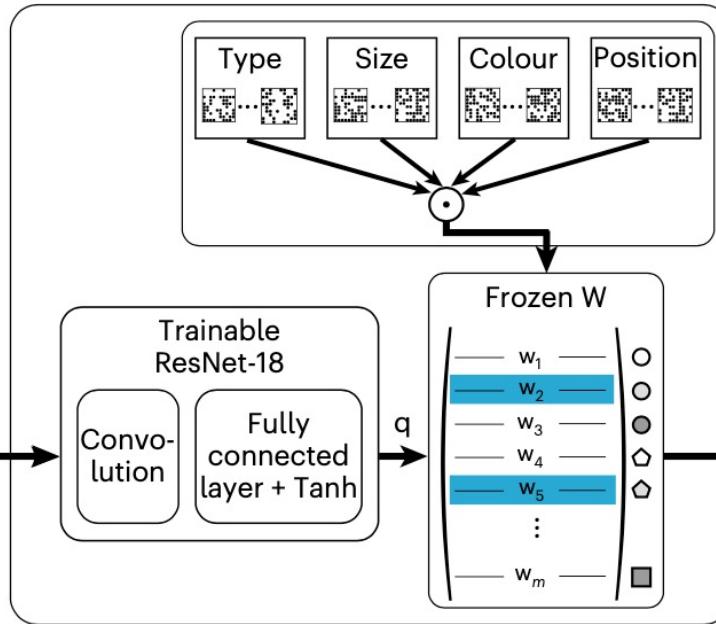
Hersche, et al. "A neuro-vector-symbolic architecture for solving Raven's progressive matrices". In Nature Machine Intelligence, 2023

Example: Neuro-Vector-Symbolic Architecture

RAVEN example test

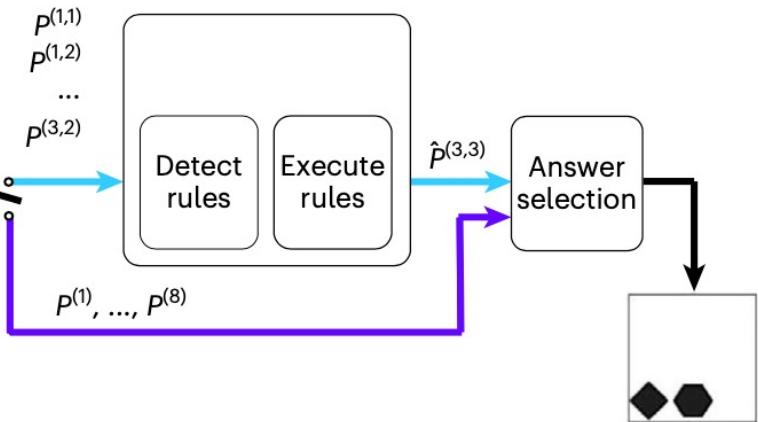


NVSA frontend: perception



Neuro Perception

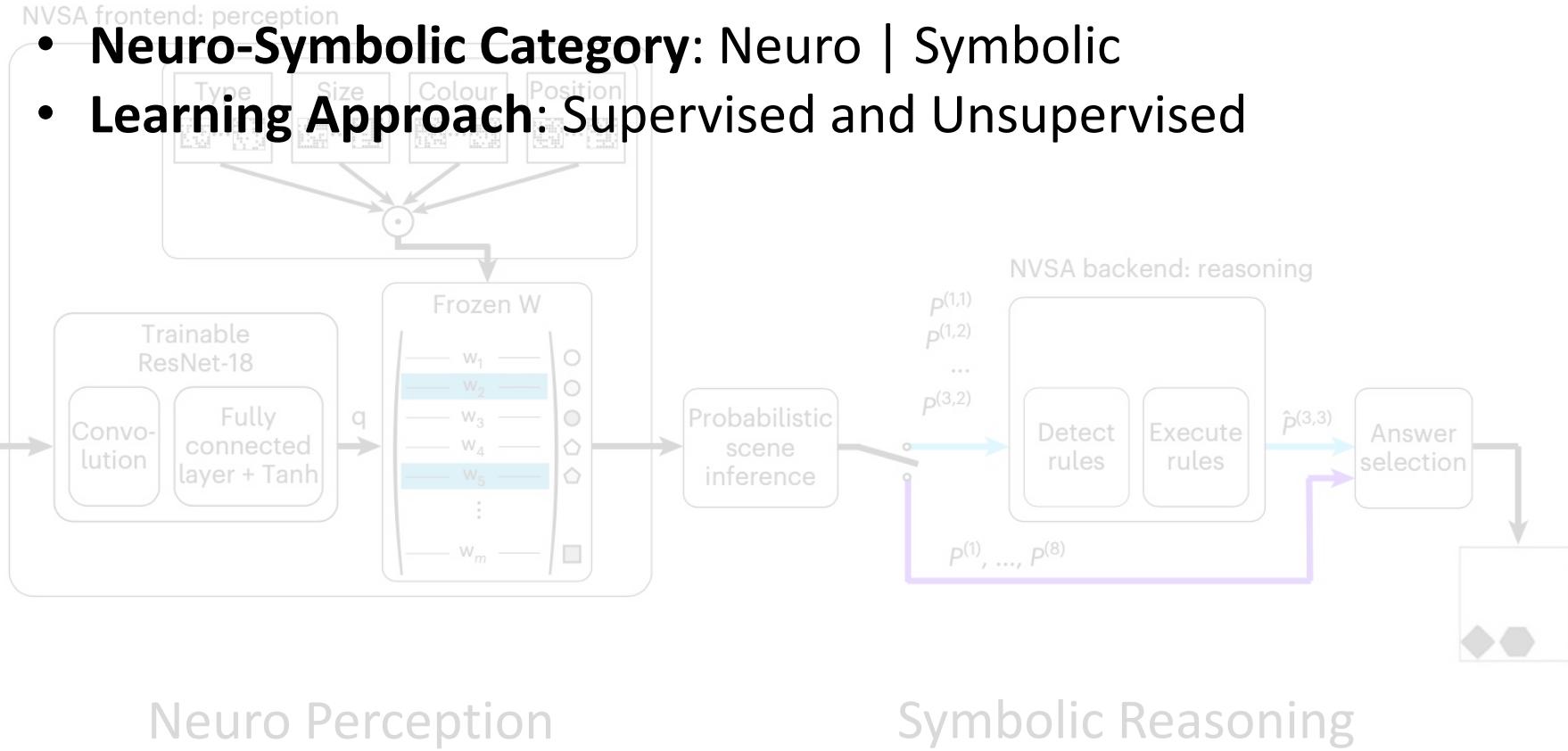
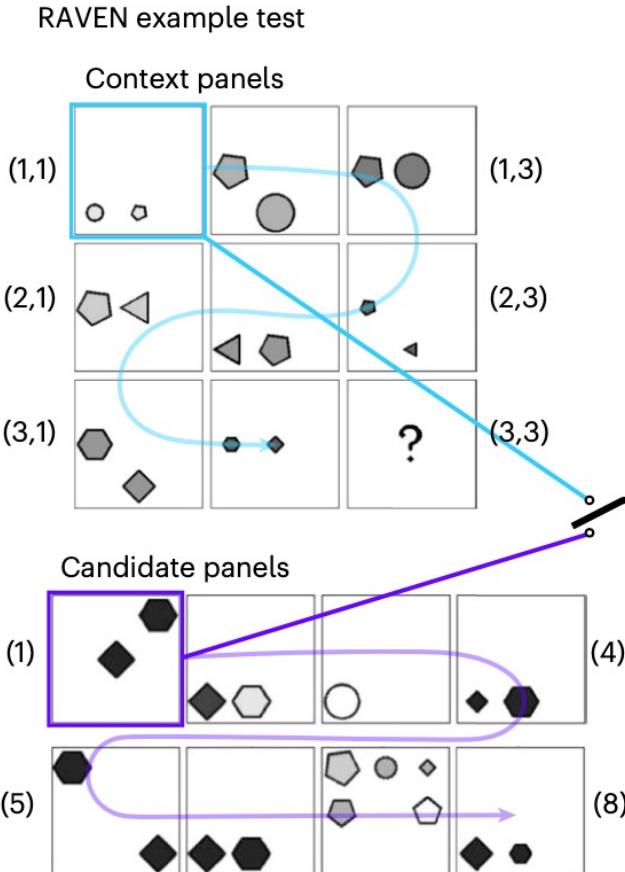
NVSA backend: reasoning



Symbolic Reasoning

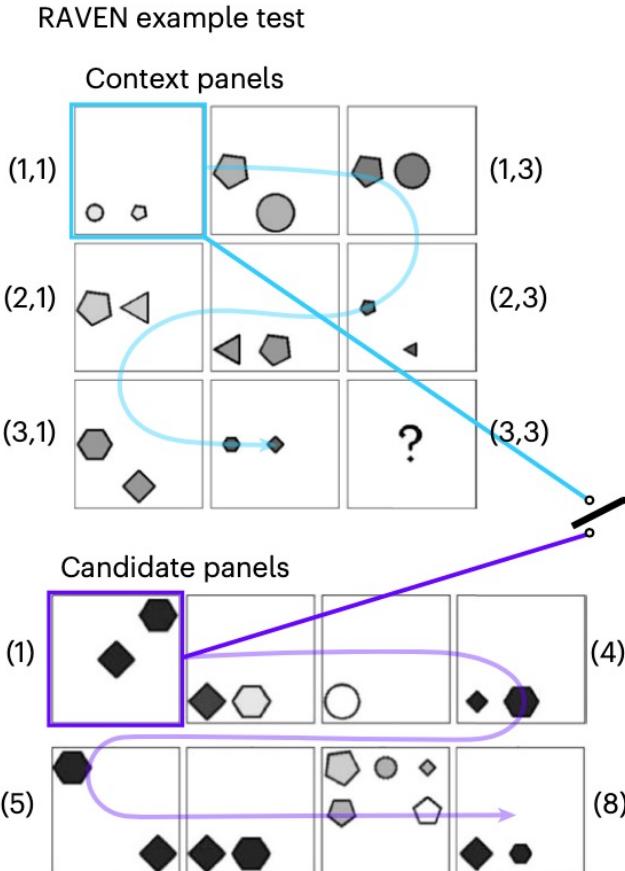
Hersche, et al. "A neuro-vector-symbolic architecture for solving Raven's progressive matrices". In Nature Machine Intelligence, 2023

Example: Neuro-Vector-Symbolic Architecture



Hersche, et al. "A neuro-vector-symbolic architecture for solving Raven's progressive matrices". In Nature Machine Intelligence, 2023

Example: Neuro-Vector-Symbolic Architecture

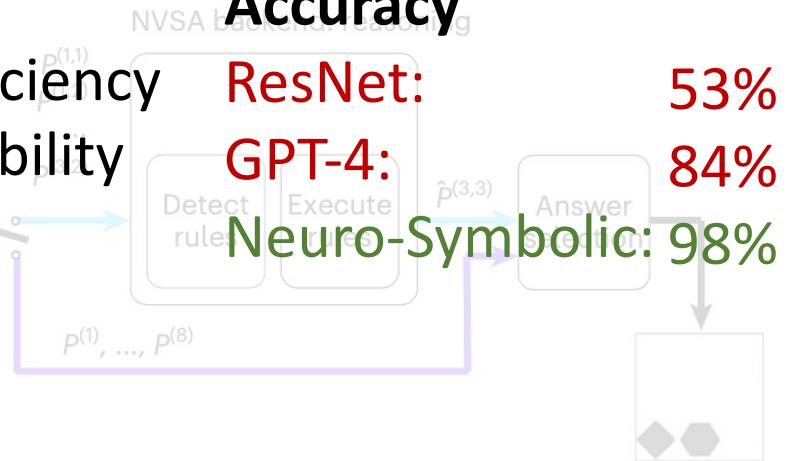


NVSA frontend: perception

- **Neuro-Symbolic Category:** Neuro | Symbolic
- **Learning Approach:** Supervised and Unsupervised
- **Application:** Fluid Intelligence, Abstract reasoning
- **Advantages over Neural Model:**
 - Higher joint representation efficiency
 - Higher abstract reasoning capability
 - Higher transparency
- **Dataset:** RAVEN, I-RAVEN, PGM

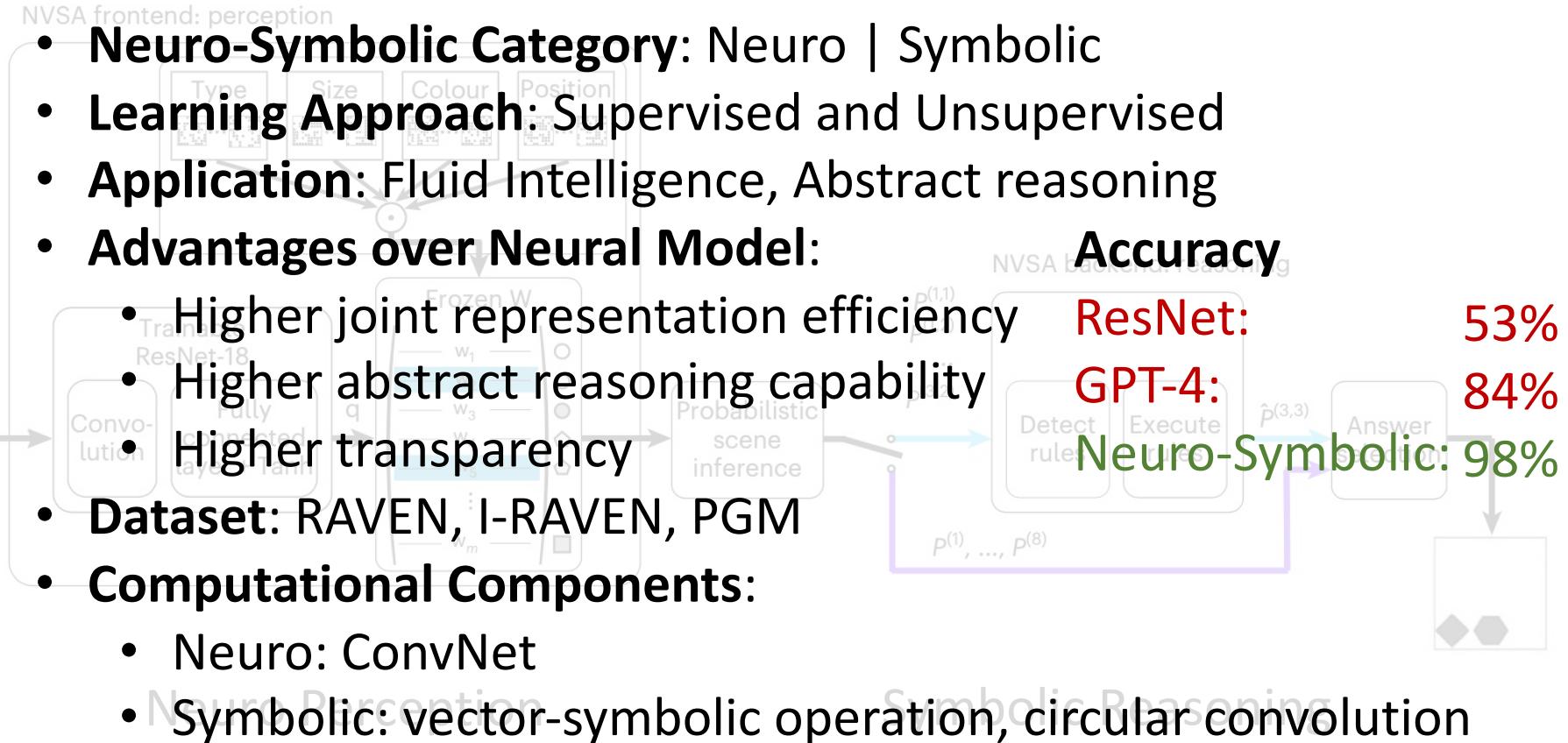
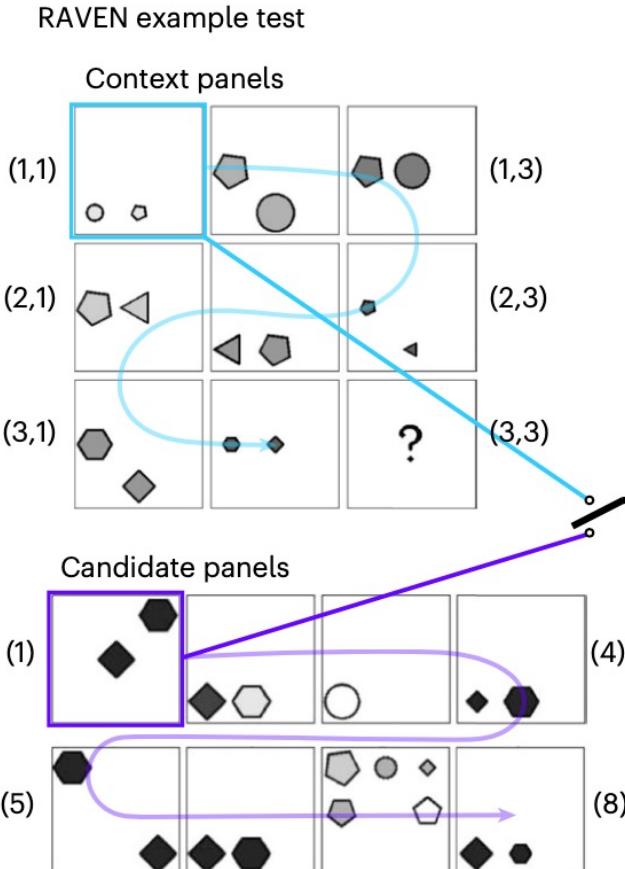
Neuro Perception

Symbolic Reasoning



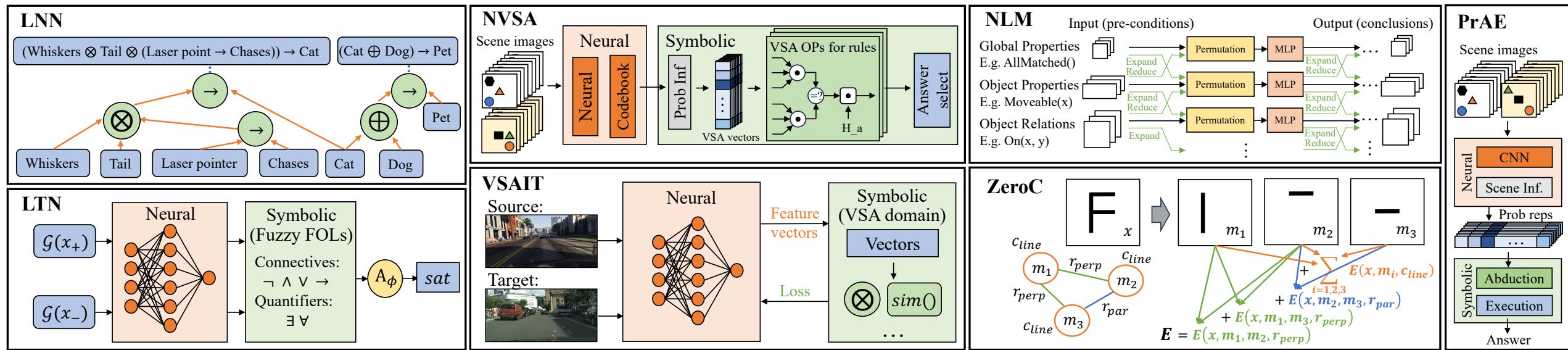
Hersche, et al. "A neuro-vector-symbolic architecture for solving Raven's progressive matrices". In Nature Machine Intelligence, 2023

Example: Neuro-Vector-Symbolic Architecture



Hersche, et al. "A neuro-vector-symbolic architecture for solving Raven's progressive matrices". In Nature Machine Intelligence, 2023

Example: Neuro-Vector-Symbolic Architecture (NVSA)



Representative Neuro-Symbolic AI Workloads	Logic Neural Network [30]	Logic Tensor Network [34]	Neuro-Vector-Symbolic Architecture [4]	Vector Symbolic Architecture Image2Image Translation [7]	Neural Logic Machine [38]	Zero-shot Concept Recognition and Acquisition [37]	Probabilistic Abduction and Execution [23]
Abbreviation	LNN	LTN	NVSA	VSAIT	NLM	ZeroC	PrAE
Neuro-Symbolic Category	Neuro:Symbolic→Neuro	Neuro Symbolic	Neuro Symbolic	Neuro Symbolic	Neuro[Symbolic]	Neuro[Symbolic]	Neuro Symbolic
Learning Approach	Supervised	Supervised/Unsupervised	Supervised/Unsupervised	Supervised	Supervised/Unsupervised	Supervised	Supervised/Unsupervised
Deployment Scenario	Application	Learning and reasoning, Full theorem prover	Querying, learning, reasoning (relational and embedding learning, query answering)	Fluid intelligence, Abstract reasoning	Unpaired image-to-image translation	Relational reasoning, Decision making	Cross-domain classification and detection, Concept acquisition
	Advantage vs. Neural Model	Higher interoperability, resilience to incomplete knowledge, generalization	Higher data efficiency, comprehensibility, out-of-distribution generalization	Higher joint representations efficiency, abstract reasoning capability, transparency	Address semantic flipping and hallucinations issue in unpaired image translation tasks	Higher generalization, logic reasoning, deduction, explainability capability	Higher generalization, concept acquisition and recognition, compositionality capability
	Dataset	LUBM benchmark [40], TPTP benchmark [41]	UCI [42], Leptograpsus crabs [43], DeepProbLog [44]	RAVEN [21], I-RAVEN [22], PGM [45]	GTA [47], Cityscapes [48], Google Maps dataset [49]	Family graph reasoning, sorting, path finding [46]	Abstraction reasoning [50], Hierarchical-concept corpus [51]
Computation Pattern	Datatype	FP32	FP32	FP32	FP32	FP32	INT64
	Neuro	Graph	MLP	ConvNet	ConvNet	Sequential tensor	Energy-based network
	Symbolic	FOL/Logical operation	FOL/Logical operation	VSA/Vector operation	VSA/Vector operation	FOL/Logical operation	Graph, vector operation



Research Question:

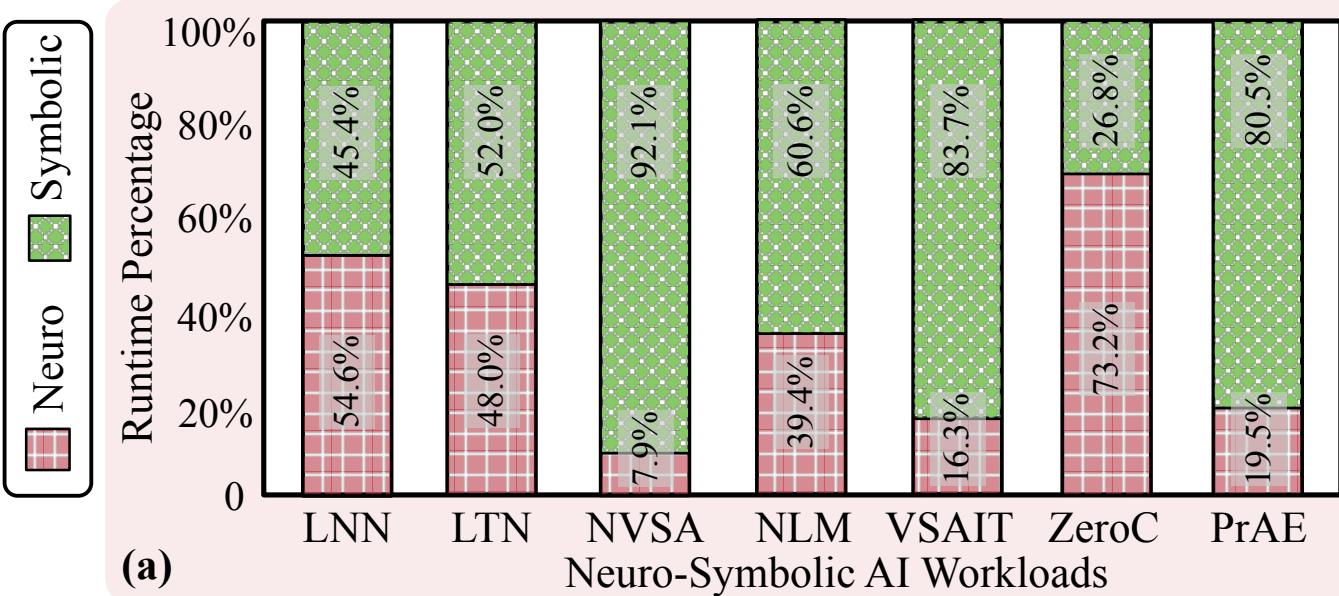
What's the **system implications** of neuro-symbolic workloads?

Why neuro-symbolic workloads are **inefficient** on off-the-shelf hardware?

Workload Characterization - Runtime

Profiling setup: CPU+GPU system, using pytorch profiler, seven neuro-symbolic workloads

- End-to-end runtime latency analysis:

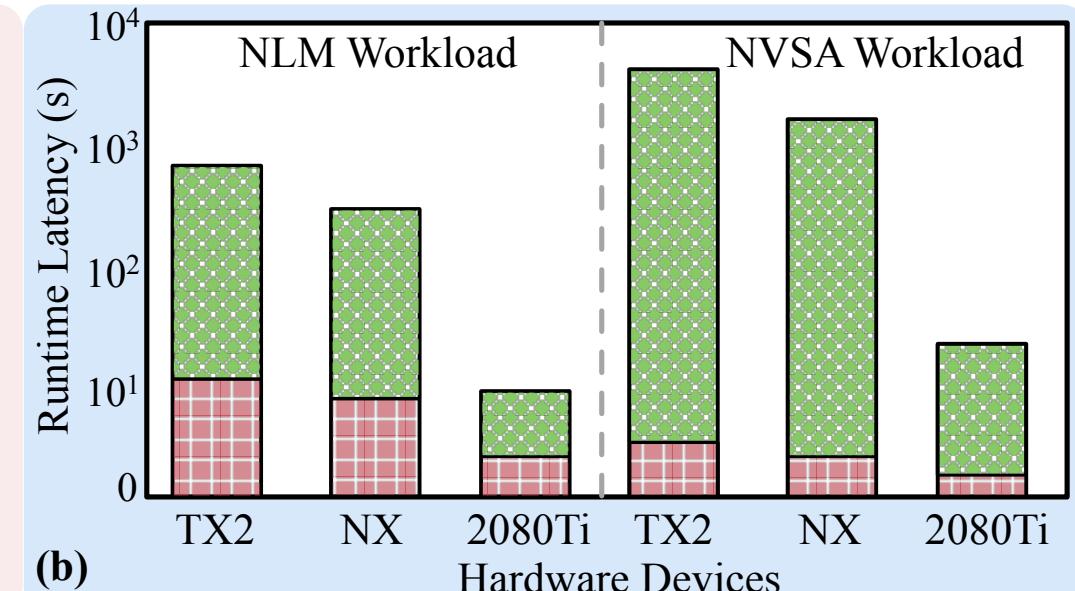
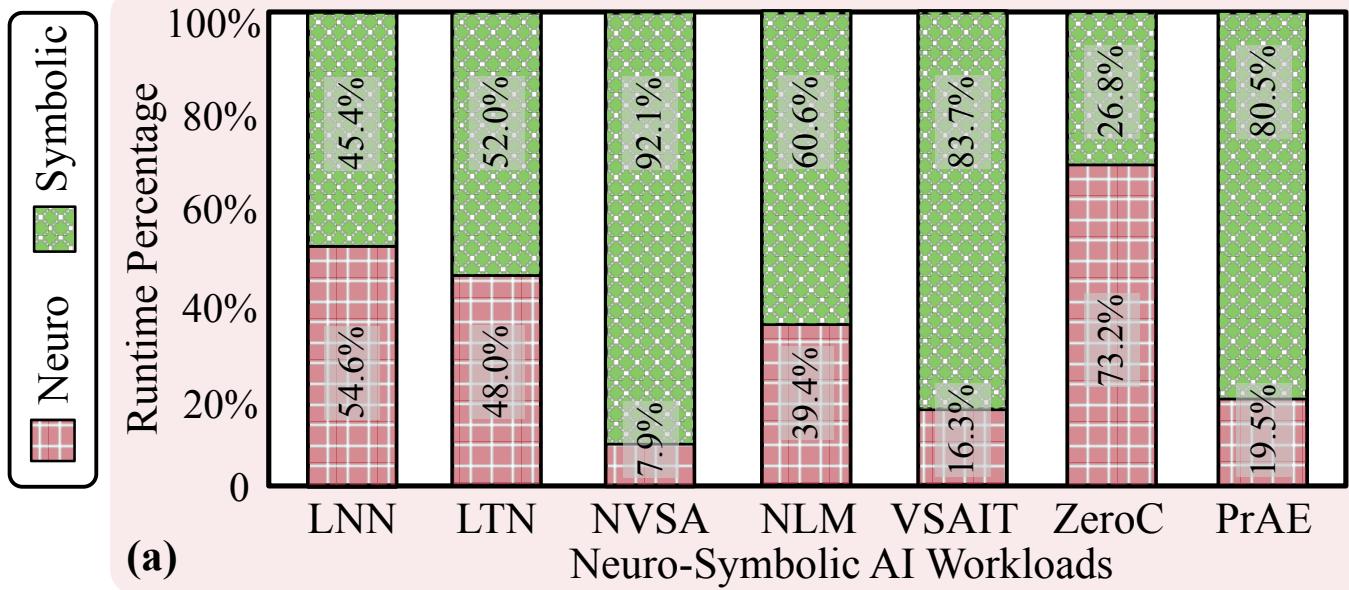


Neuro-symbolic workload exhibits **high latency** compared to neural models;

Workload Characterization - Runtime

Profiling setup: CPU+GPU system, using pytorch profiler, seven neuro-symbolic workloads

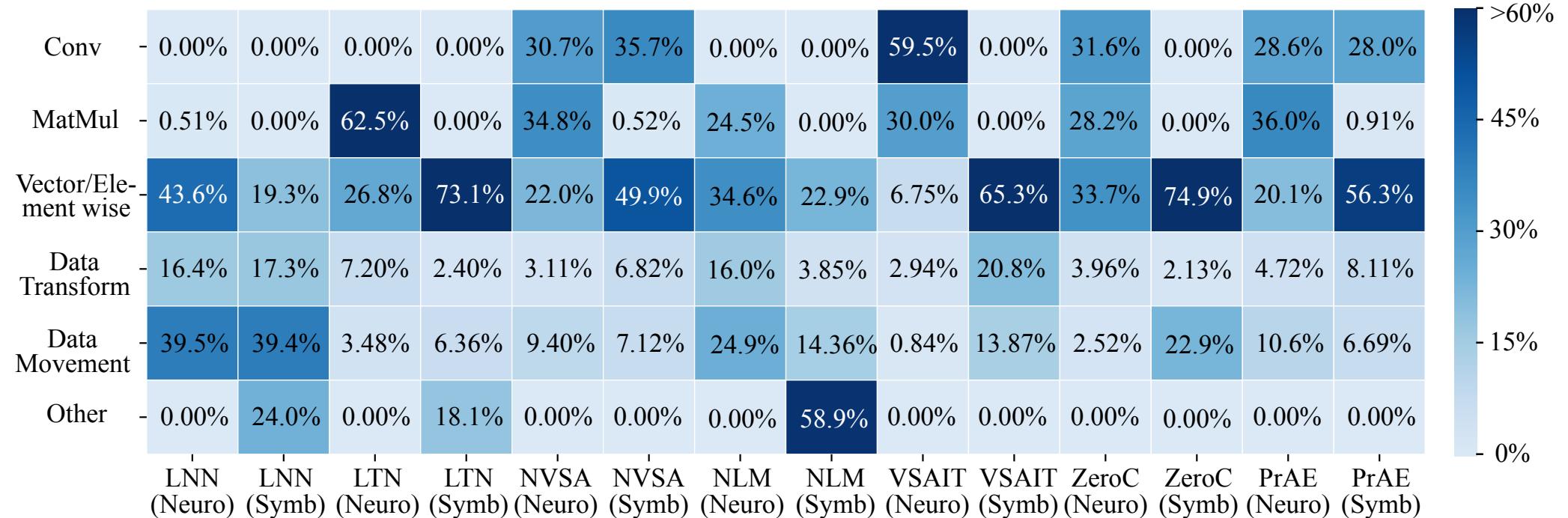
- End-to-end runtime latency analysis:



Neuro-symbolic workload exhibits **high latency** compared to neural models;
Symbolic component is executed **inefficiently** across off-the-shelf CPU/GPUs

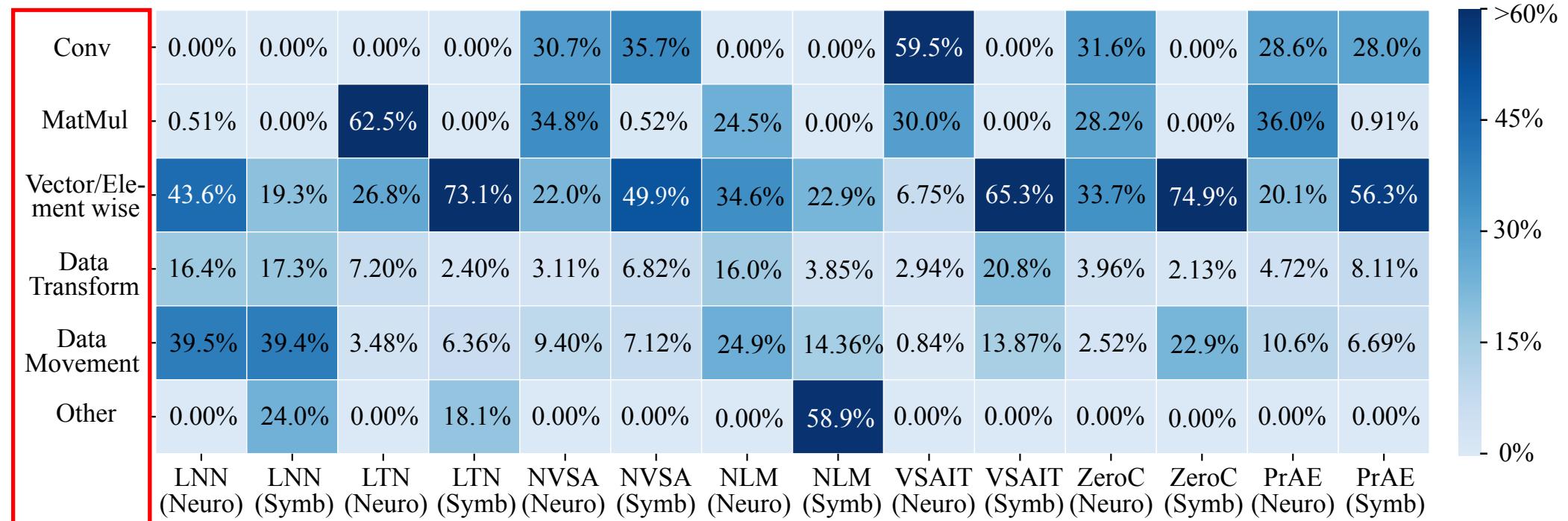
Workload Characterization - Operator

- Compute operator analysis:



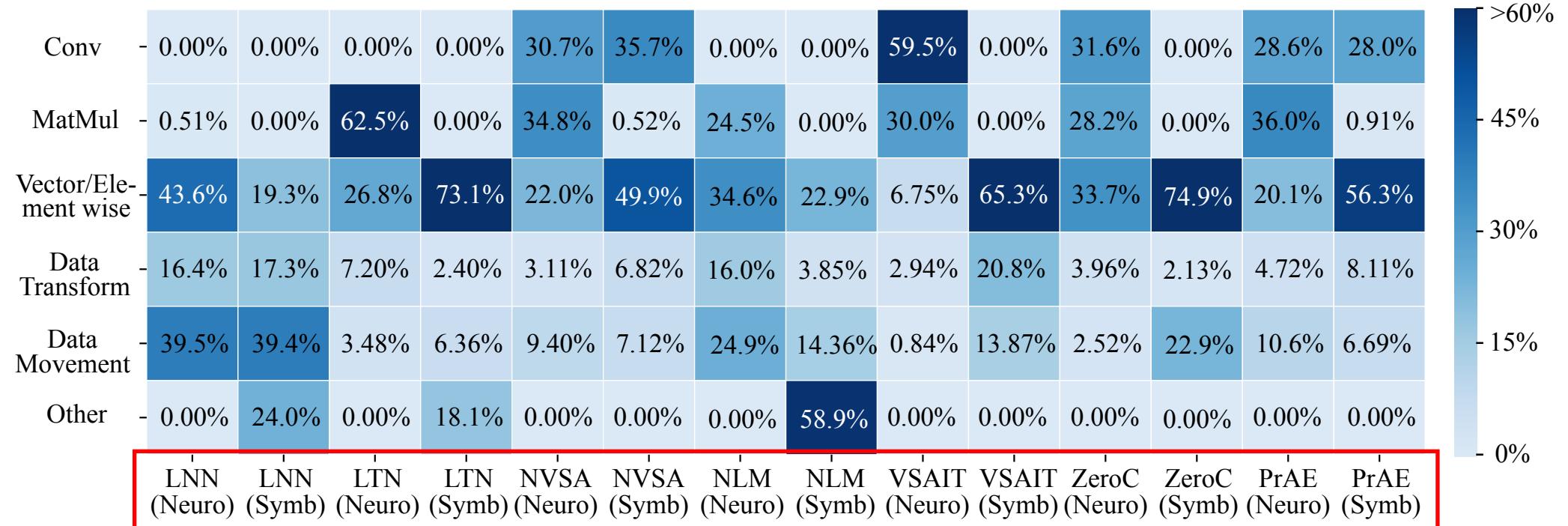
Workload Characterization - Operator

- Compute operator analysis:



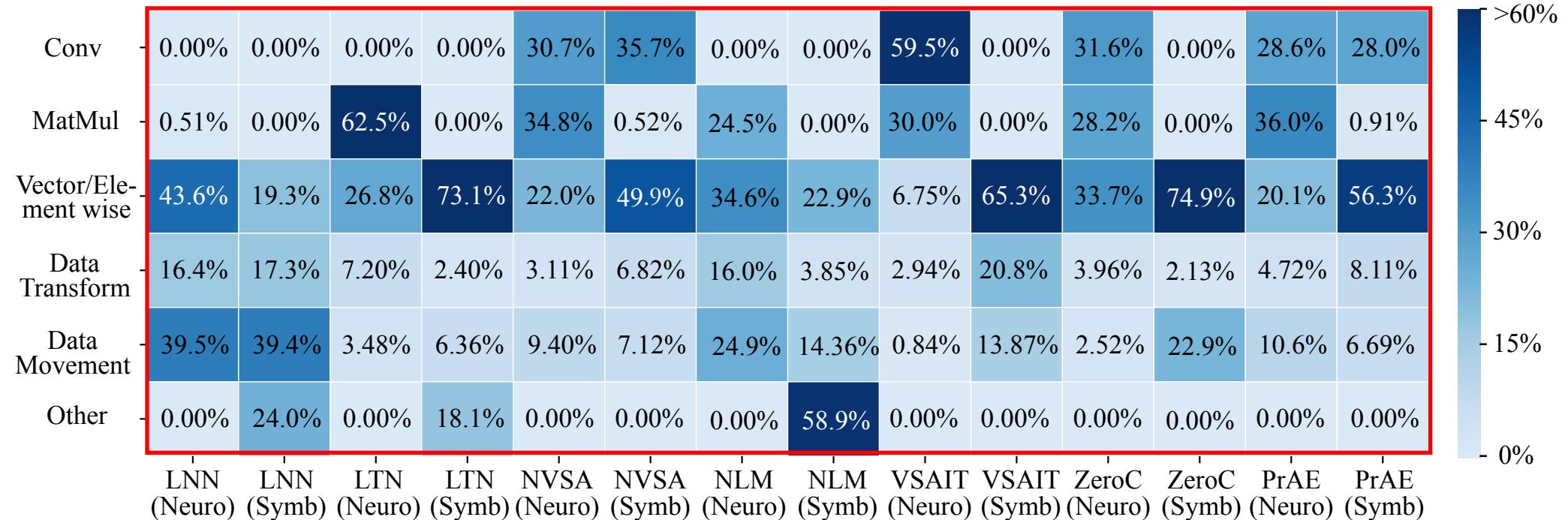
Workload Characterization - Operator

- Compute operator analysis:



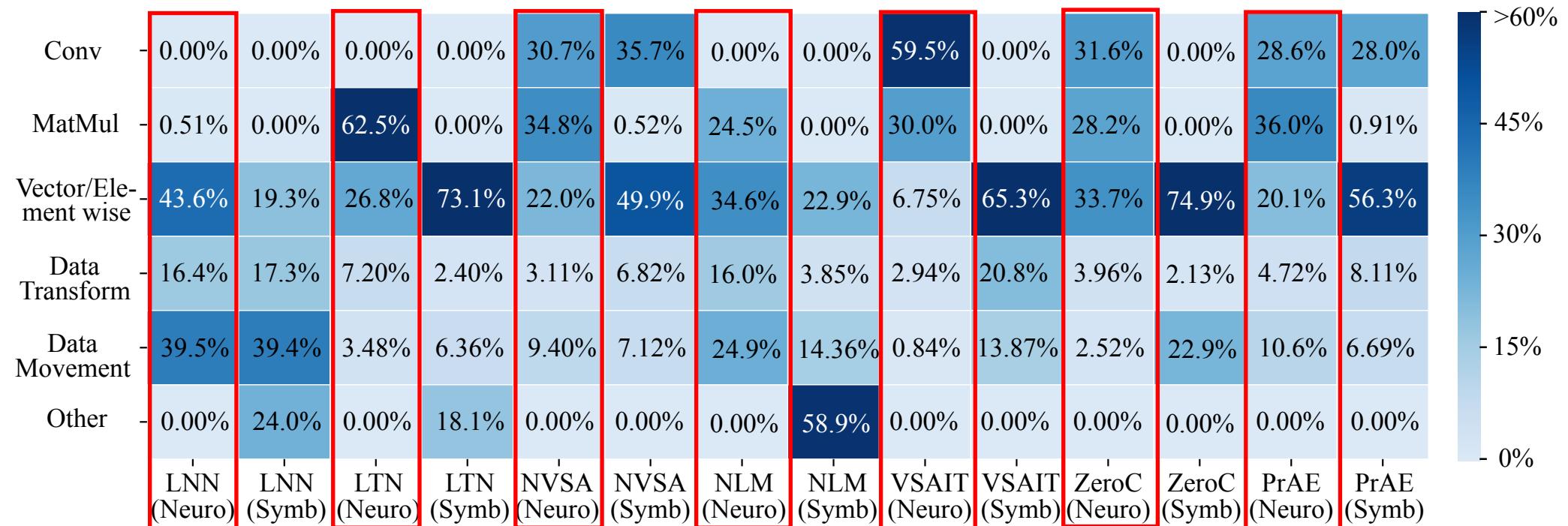
Workload Characterization - Operator

- Compute operator analysis:



Workload Characterization - Operator

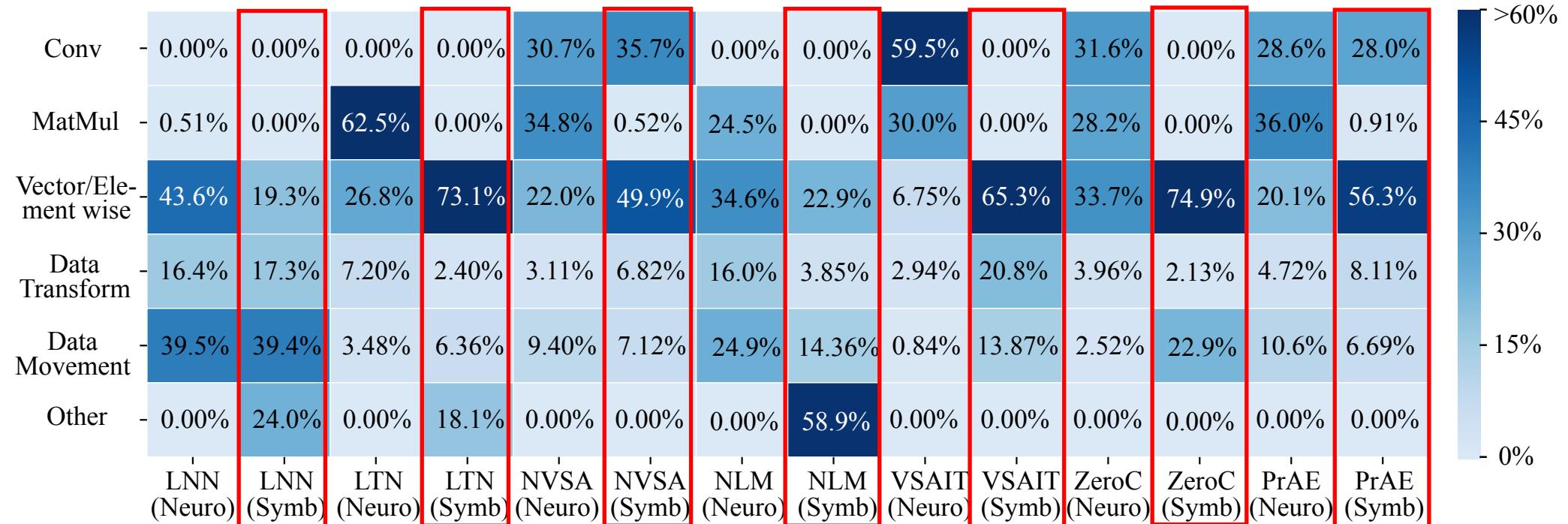
- Compute operator analysis:



Neural dominated by MatMul and Conv operations;

Workload Characterization - Operator

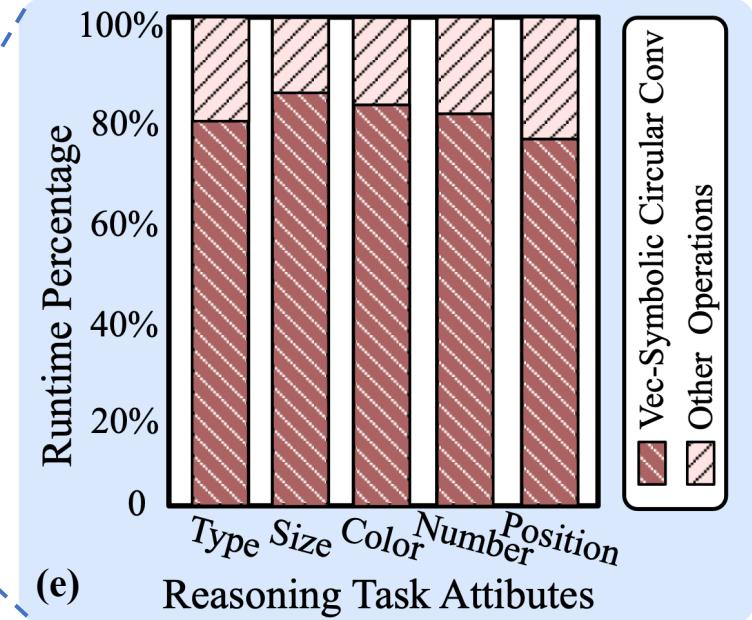
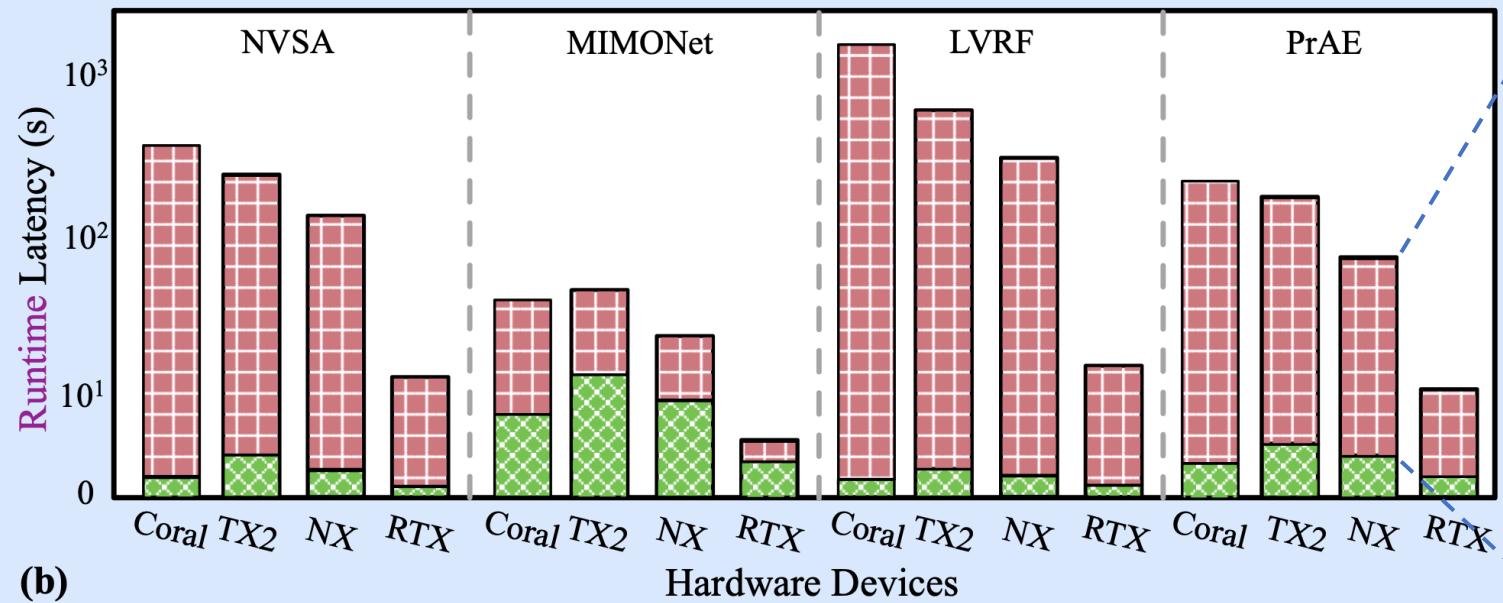
- Compute operator analysis:



Neural dominated by MatMul and Conv operations; Symbolic dominated by vector/element-wise and logical operations

Workload Characterization - Operator

■ Neuro ■ Symbolic



One example of dominated symbolic operation is **vector-symbolic circular convolutions**

Workload Characterization – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)				
Compute Throughput (%)				
ALU Utilization (%)				
L1 Cache Hit Rate (%)				
L2 Cache Hit Rate (%)				
L1 Cache Throughput (%)				
L2 Cache Throughput (%)				
DRAM BW Utilization (%)				

Why system Inefficiency?

Workload Characterization – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	95.1	92.9	3.0	2.3
ALU Utilization (%)	90.1	48.3	5.9	4.5
L1 Cache Hit Rate (%)				
L2 Cache Hit Rate (%)				
L1 Cache Throughput (%)				
L2 Cache Throughput (%)				
DRAM BW Utilization (%)				

Symbolic exhibits low ALU utilization,

Workload Characterization – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	95.1	92.9	3.0	2.3
ALU Utilization (%)	90.1	48.3	5.9	4.5
L1 Cache Hit Rate (%)	1.6	51.6	29.5	33.3
L2 Cache Hit Rate (%)	86.8	65.5	48.6	34.3
L1 Cache Throughput (%)				
L2 Cache Throughput (%)				
DRAM BW Utilization (%)				

Symbolic exhibits low ALU utilization, low cache hit rate,

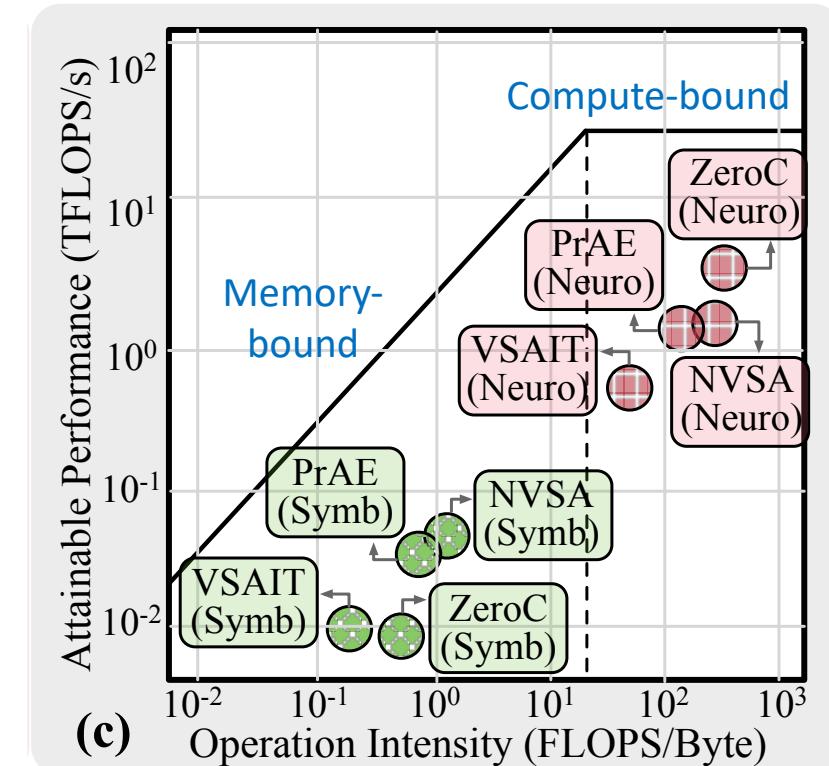
Workload Characterization – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	95.1	92.9	3.0	2.3
ALU Utilization (%)	90.1	48.3	5.9	4.5
L1 Cache Hit Rate (%)	1.6	51.6	29.5	33.3
L2 Cache Hit Rate (%)	86.8	65.5	48.6	34.3
L1 Cache Throughput (%)	79.7	82.6	28.4	10.8
L2 Cache Throughput (%)	19.2	17.5	29.8	22.8
DRAM BW Utilization (%)	14.9	24.2	90.9	78.4

Symbolic exhibits low ALU utilization, low cache hit rate, **massive data transfer**, resulting in hardware underutilization and inefficiency

Workload Characterization – Kernel Behavior

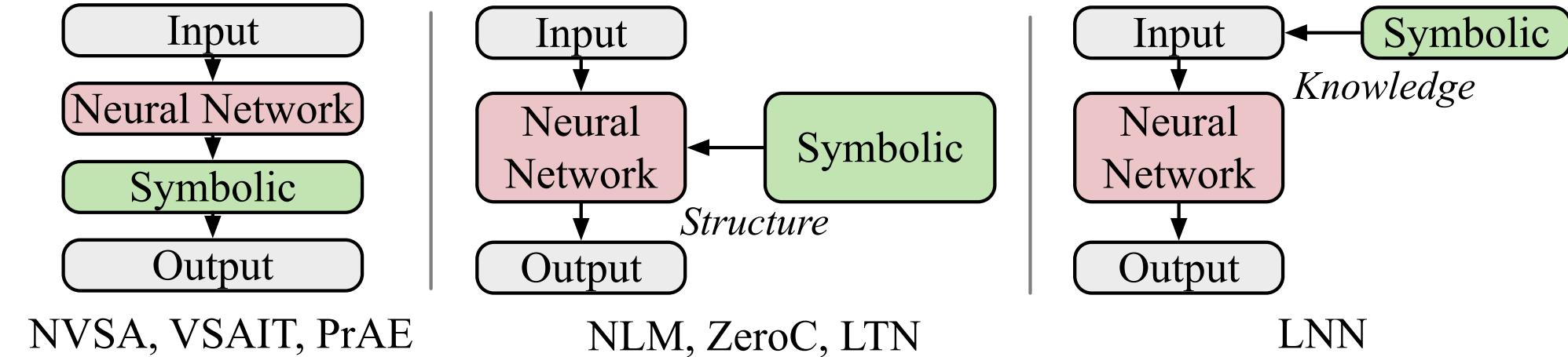
	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	95.1	92.9	3.0	2.3
ALU Utilization (%)	90.1	48.3	5.9	4.5
L1 Cache Hit Rate (%)	1.6	51.6	29.5	33.3
L2 Cache Hit Rate (%)	86.8	65.5	48.6	34.3
L1 Cache Throughput (%)	79.7	82.6	28.4	10.8
L2 Cache Throughput (%)	19.2	17.5	29.8	22.8
DRAM BW Utilization (%)	14.9	24.2	90.9	78.4



Neuro operations are **compute-bounded**, symbolic operations are **memory-bounded**.

Workload Characterization – Control Flow

- Data Dependence Graph analysis:



Neuro and symbolic components interaction requires **complex control flow**

Neural Network vs. Neuro-Symbolic

	Neural Network	Neuro-Symbolic

Neural Network vs. Neuro-Symbolic

	Neural Network	Neuro-Symbolic
Runtime	[Neural Network] < [Neuro-Symbolic]	

Neural Network vs. Neuro-Symbolic

	Neural Network	Neuro-Symbolic
Runtime	[Neural Network] < [Neural-Symbolic]	
Compute Kernels	Neural kernels (Conv, MatMul, etc)	Heterogenous neural and symbolic kernels (vector, element, MatMul, graph, logic, etc)

Neural Network vs. Neuro-Symbolic

	Neural Network	Neuro-Symbolic
Runtime	[Neural Network] < [Neural-Symbolic]	
Compute Kernels	Neural kernels (Conv, MatMul, etc)	Heterogenous neural and symbolic kernels (vector, element, MatMul, graph, logic, etc)
Hardware Efficiency	Efficient on GPU/TPU	Inefficient on CPU/GPU/TPU (low ALU utilization, low L1 cache hit rate, high data movement, etc)

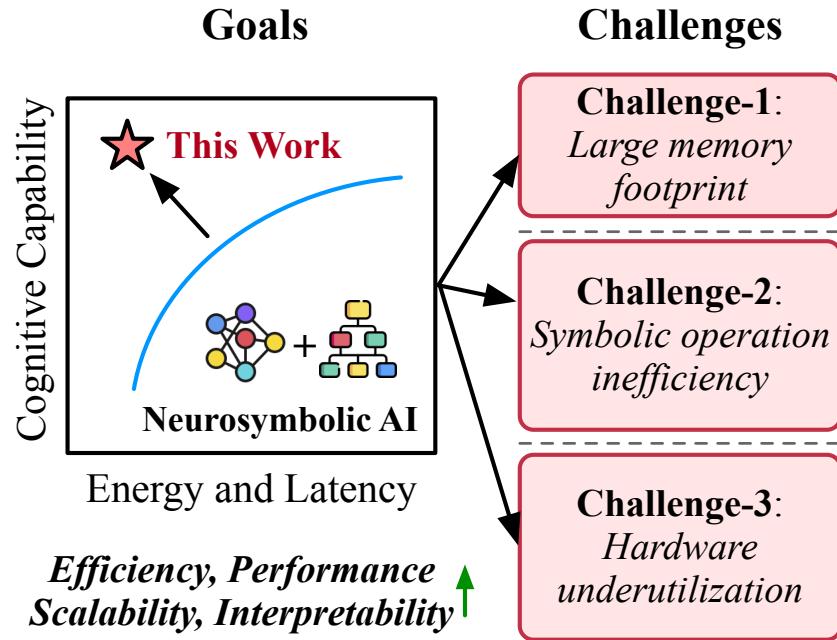
Neural Network vs. Neuro-Symbolic

	Neural Network	Neuro-Symbolic
Runtime	[Neural Network] < [Neural-Symbolic]	
Compute Kernels	Neural kernels (Conv, MatMul, etc)	Heterogenous neural and symbolic kernels (vector, element, MatMul, graph, logic, etc)
Hardware Efficiency	Efficient on GPU/TPU	Inefficient on CPU/GPU/TPU (low ALU utilization, low L1 cache hit rate, high data movement, etc)
System Bound	Compute-bound / Memory-bound	Memory-bound

Neural Network vs. Neuro-Symbolic

	Neural Network	Neuro-Symbolic
Runtime	[Neural Network] < [Neural-Symbolic]	
Compute Kernels	Neural kernels (Conv, MatMul, etc)	Heterogenous neural and symbolic kernels (vector, element, MatMul, graph, logic, etc)
Hardware Efficiency	Efficient on GPU/TPU	Inefficient on CPU/GPU/TPU (low ALU utilization, low L1 cache hit rate, high data movement, etc)
System Bound	Compute-bound / Memory-bound	Memory-bound
Dataflow	Simple flow control, High parallelism	Complex flow control, Low parallelism

Challenges of Neuro-Symbolic AI

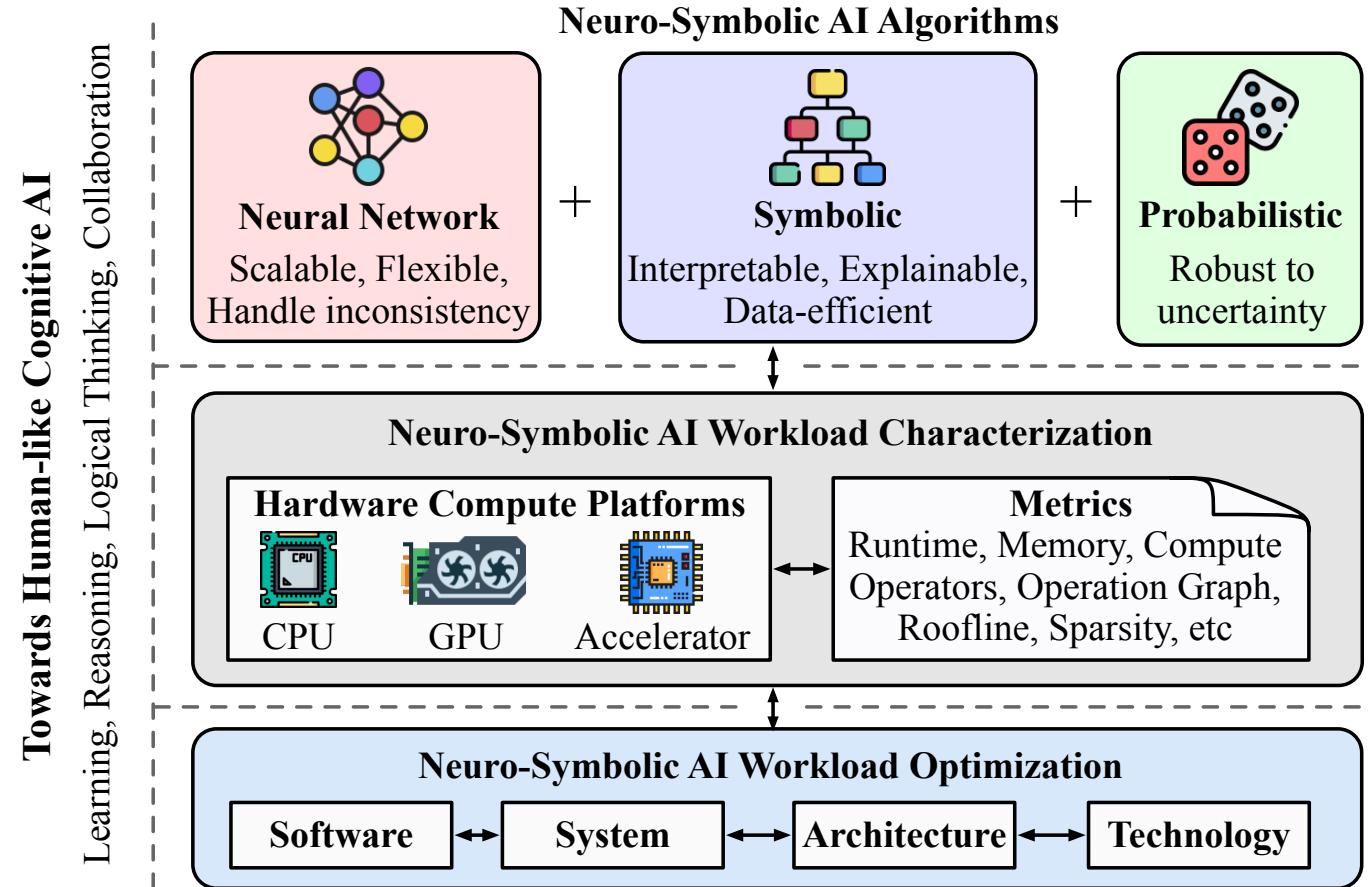


This talk: Demystify Neuro-Symbolic AI for SW/HW Co-Design

Characterize Neuro-Symbolic Workloads

Identify Potential Inefficiency Reasons

Optimize Neuro-Symbolic Systems via Co-Design

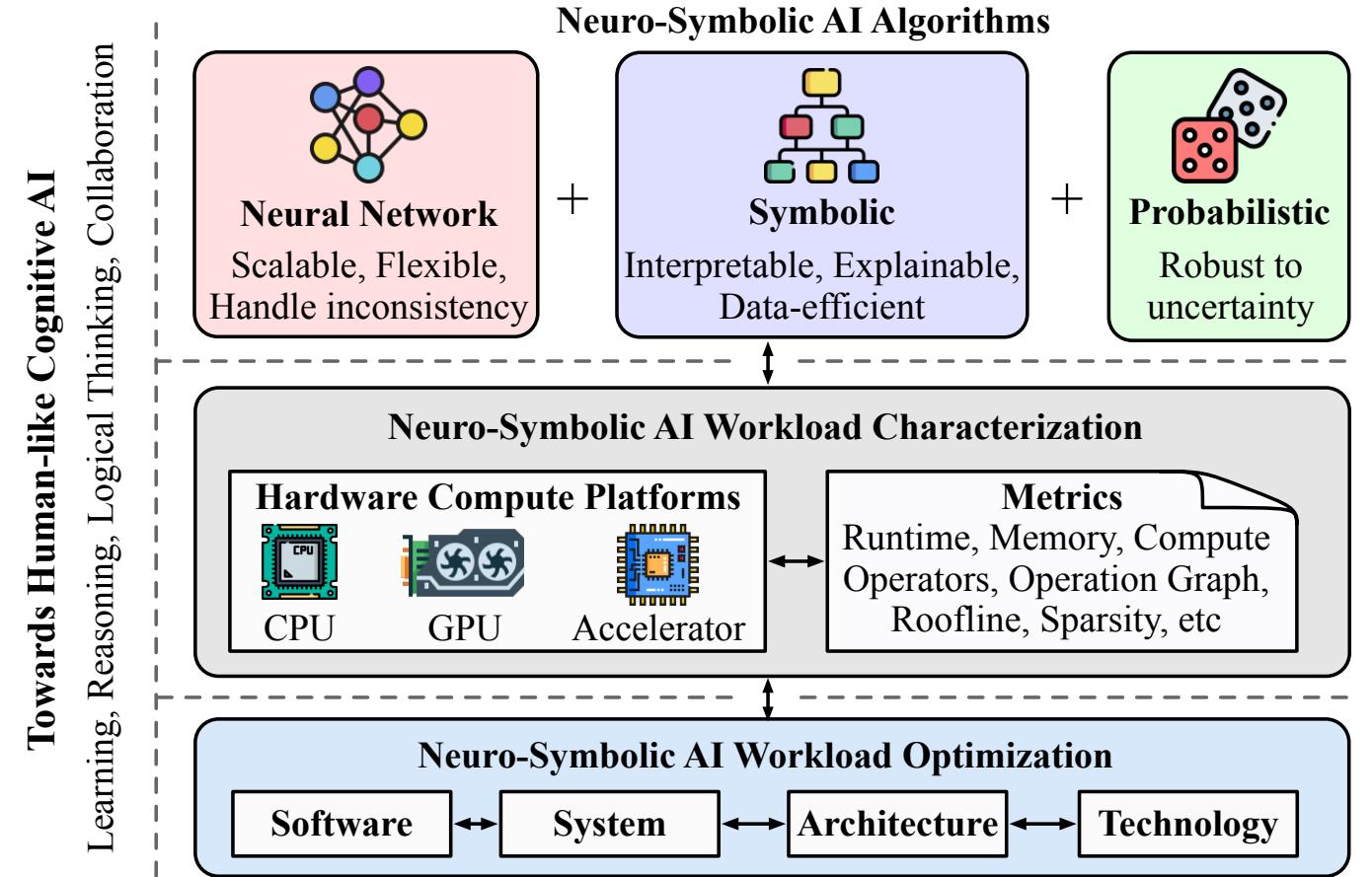


This talk: Demystify Neuro-Symbolic AI for SW/HW Co-Design

Characterize Neuro-Symbolic Workloads

Identify Potential Inefficiency Reasons

Optimize Neuro-Symbolic Systems via Co-Design



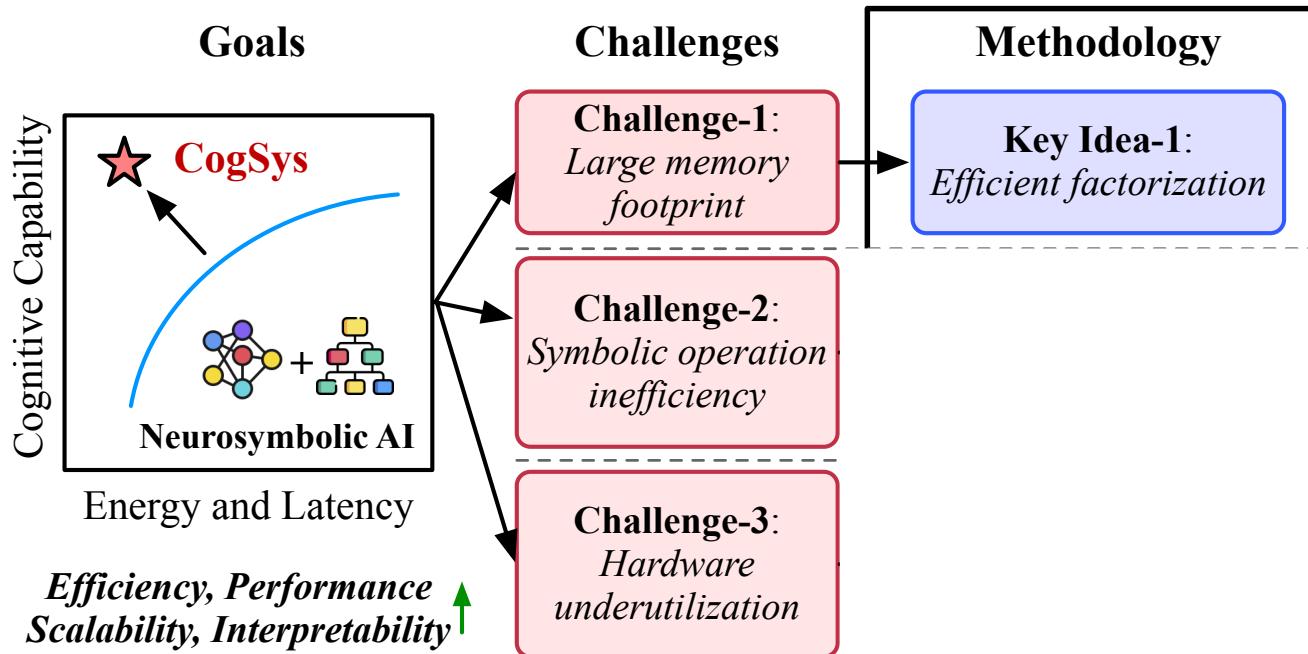
Zishen Wan*, Hanchen Yang*, Ritik Raj*, Che-Kai Liu, Ananda Samajdar, Arijit Raychowdhury, Tushar Krishna, “CogSys: Efficient and Scalable Neurosymbolic Cognition System via Algorithm-Hardware Co-Design”, in HPCA 2025



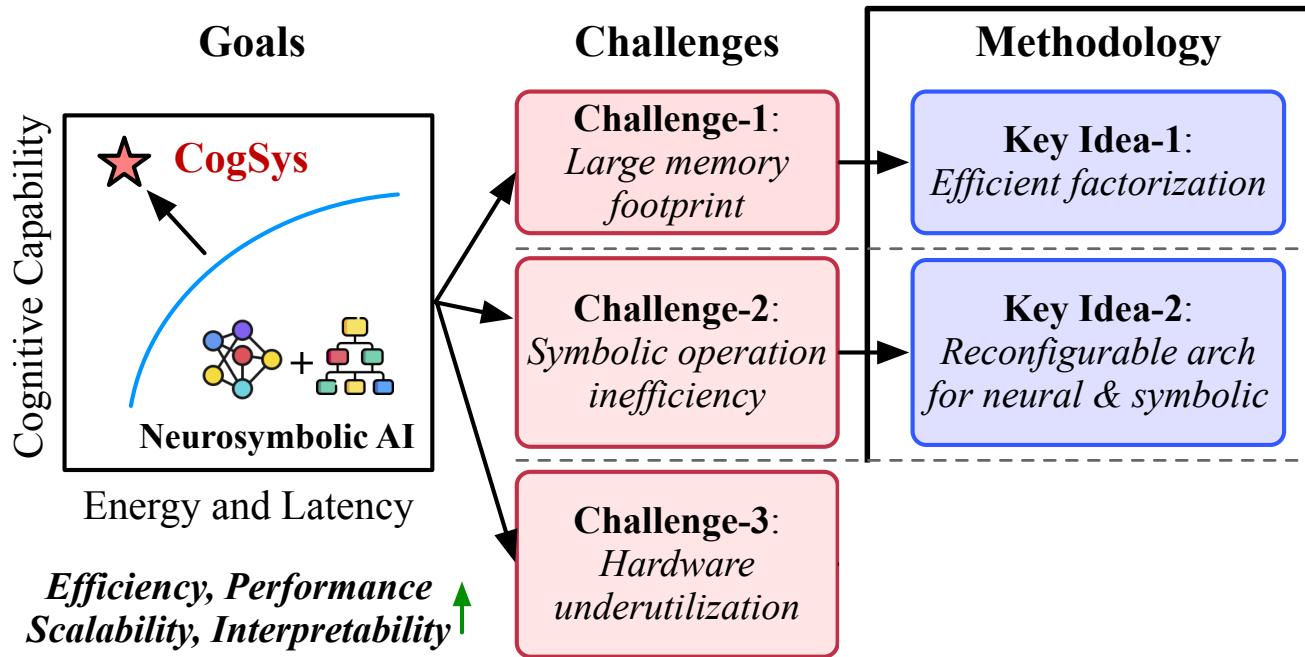
Research Question:

How to enhance the **efficiency and scalability** of neuro-symbolic systems?

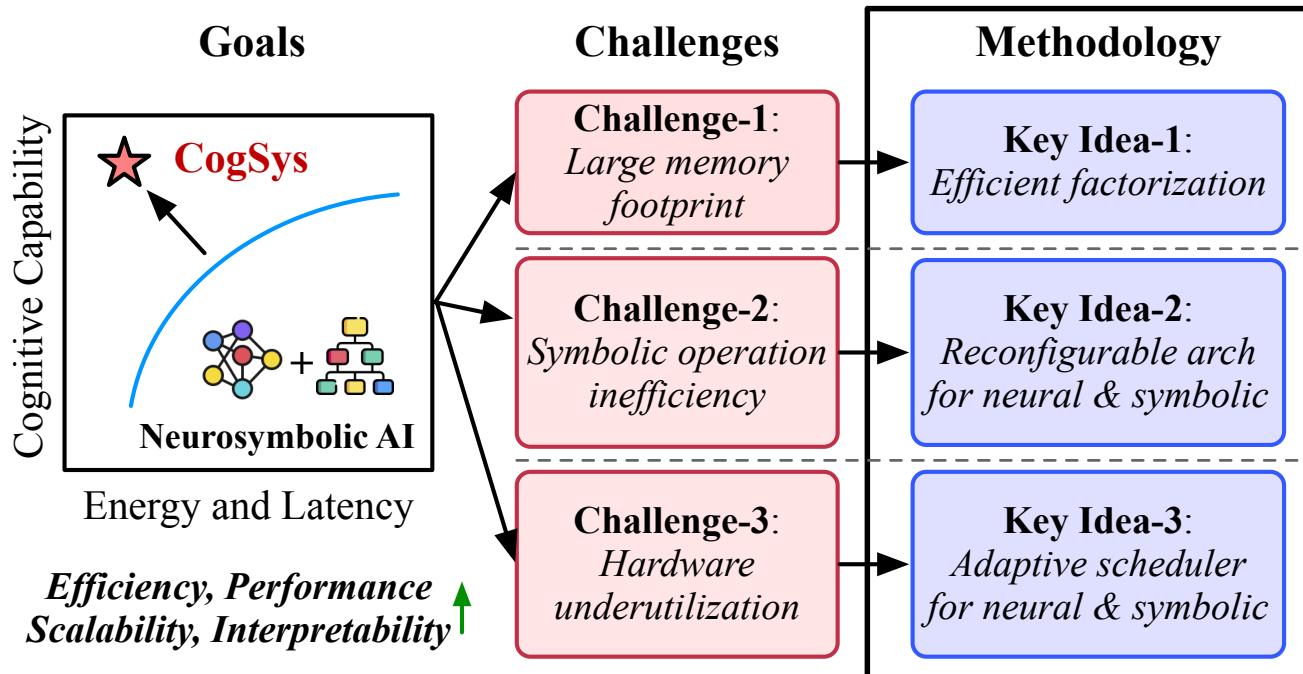
CogSys: Co-Design for Neuro-Symbolic AI



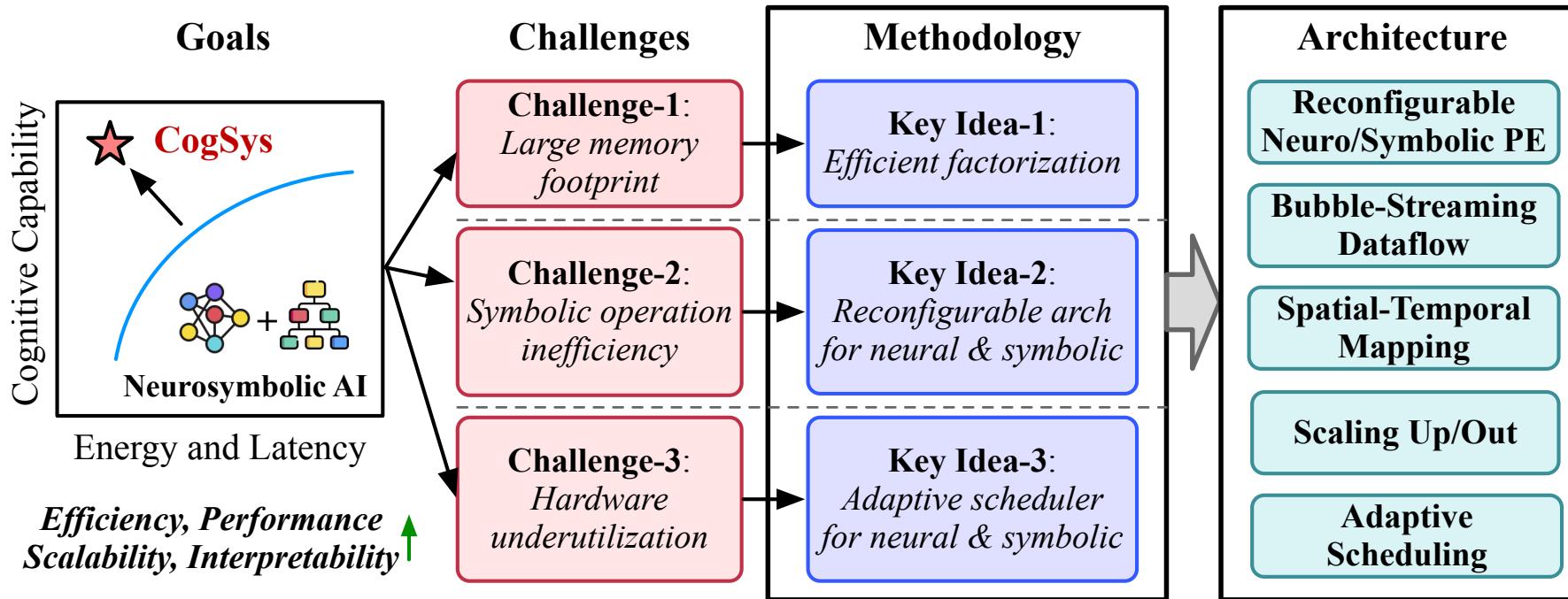
CogSys: Co-Design for Neuro-Symbolic AI



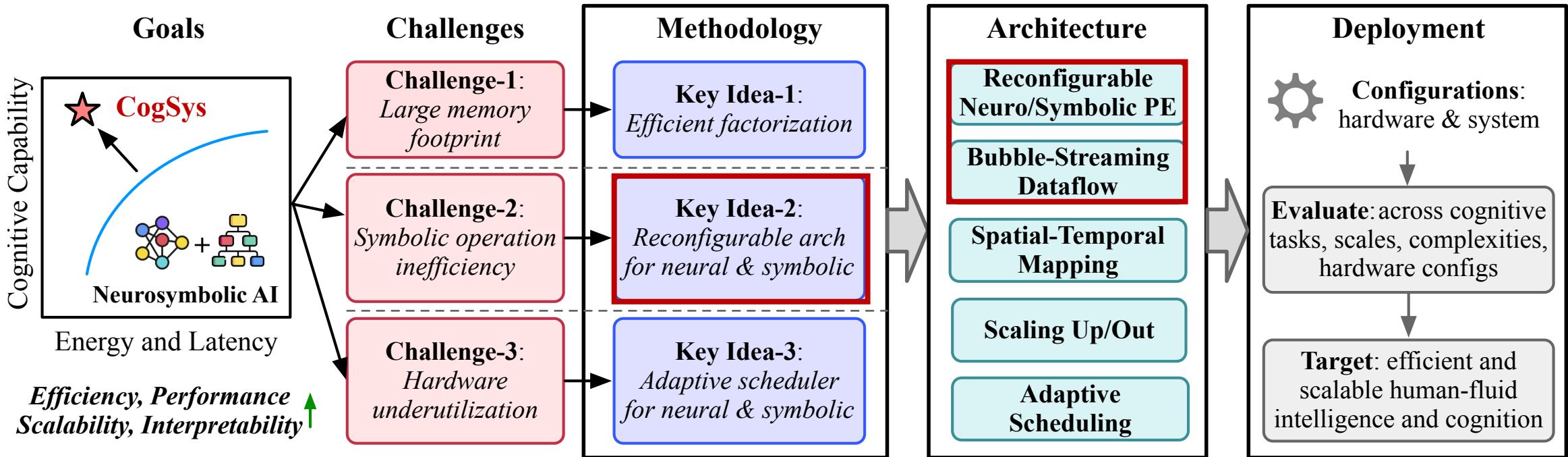
CogSys: Co-Design for Neuro-Symbolic AI



CogSys: Co-Design for Neuro-Symbolic AI

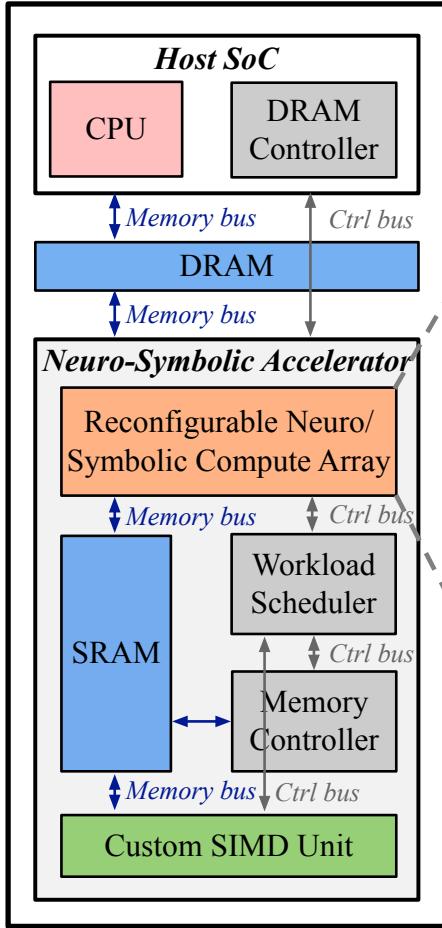


CogSys: Co-Design for Neuro-Symbolic AI

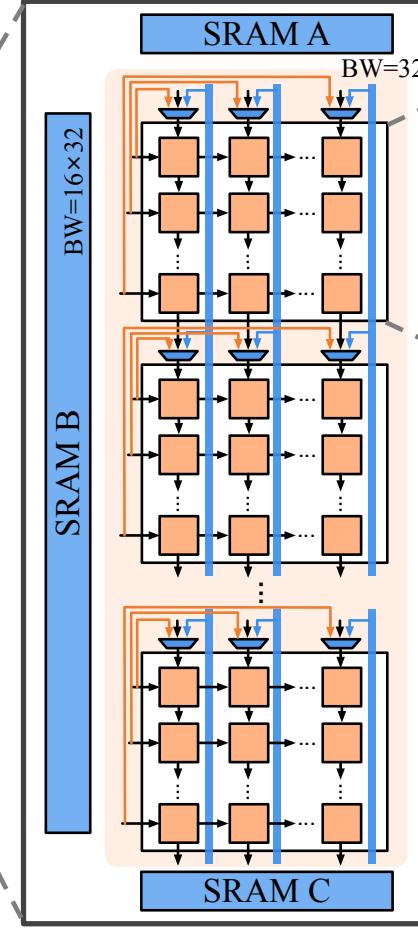


Hardware Architecture Overview

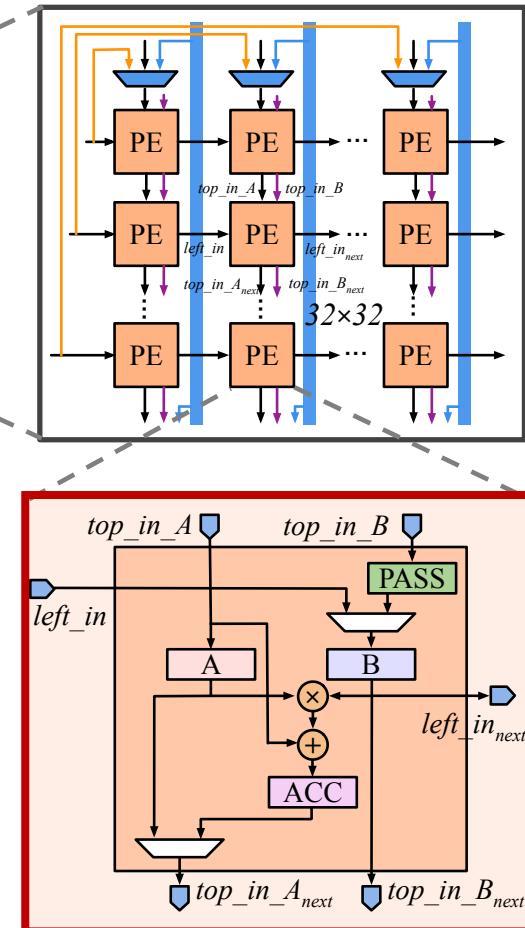
(a) Overall Architecture



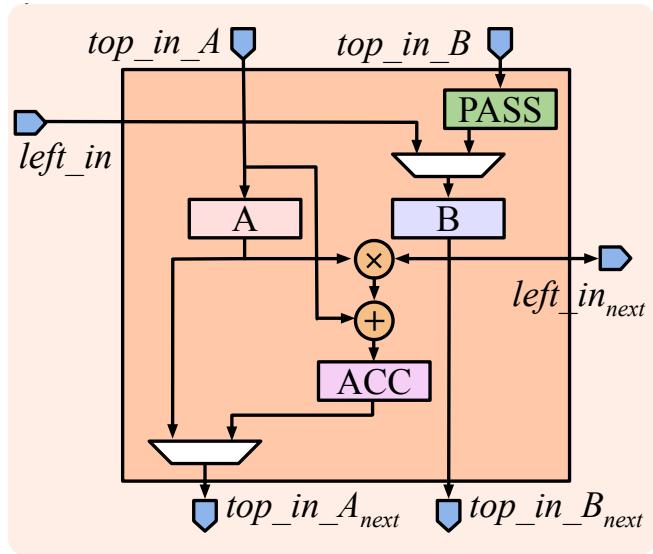
(b) Scalable Compute Array



(c) Reconfig. Neuro/Symbolic PE



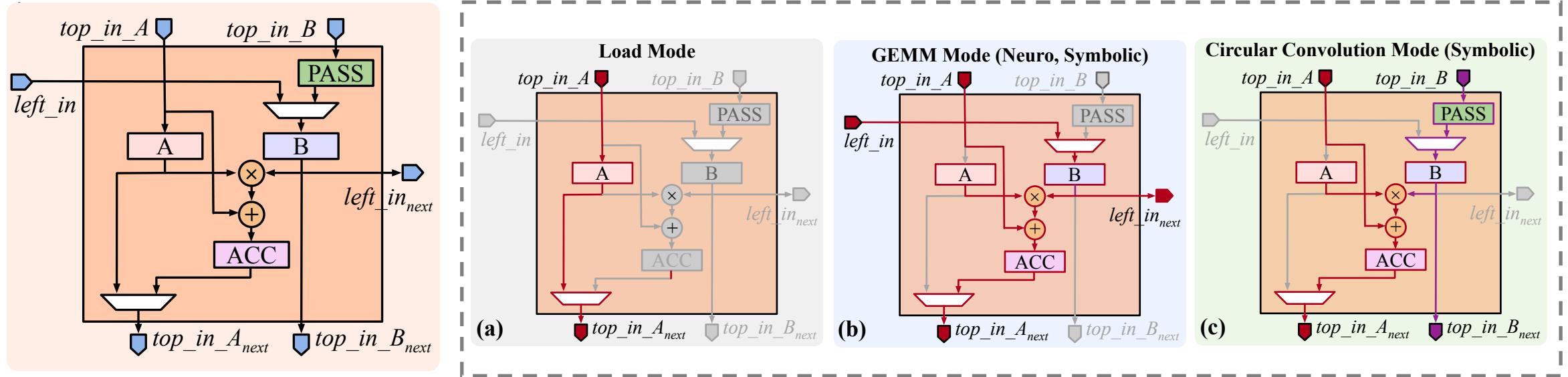
Reconfigurable Neuro/Symbolic PE



Micro-architecture of
reconfigurable neuro/symbolic PE

Reconfigurable neuro/symbolic PE incurs **low area overhead** based on systolic array PE;

Reconfigurable Neuro/Symbolic PE



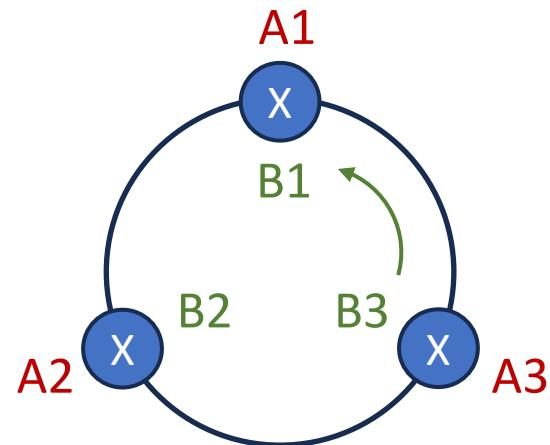
Micro-architecture of
reconfigurable neuro/symbolic PE

Operation mode of
reconfigurable neuro/symbolic PE

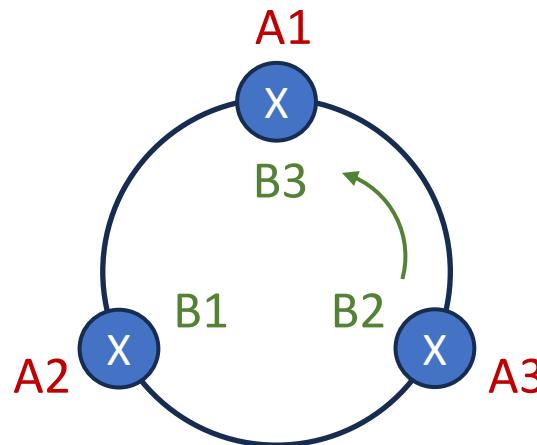
Reconfigurable neuro/symbolic PE incurs **low area overhead** based on systolic array PE;
The PE is reconfigurable for **three operation modes**: load, neuro, symbolic

What is Circular Convolution?

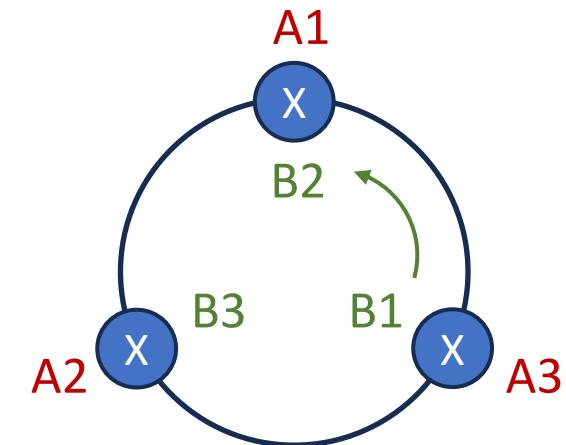
$$\begin{bmatrix} A1 \\ A2 \\ A3 \end{bmatrix} \odot \begin{bmatrix} B1 \\ B2 \\ B3 \end{bmatrix} = \begin{bmatrix} A1B1 + A2B2 + A3B3 \\ A1B3 + A2B1 + A3B2 \\ A1B2 + A2B3 + A3B1 \end{bmatrix}$$



$A1B1 + A2B2 + A3B3$



$A1B3 + A2B1 + A3B2$



$A1B2 + A2B3 + A3B1$

Bubble Streaming Dataflow

Vector-Symbolic Circular Convolution Example (3 CircConv):

$$\text{CircConv \#1: } (A_1, A_2, A_3) \odot (B_1, B_2, B_3)$$

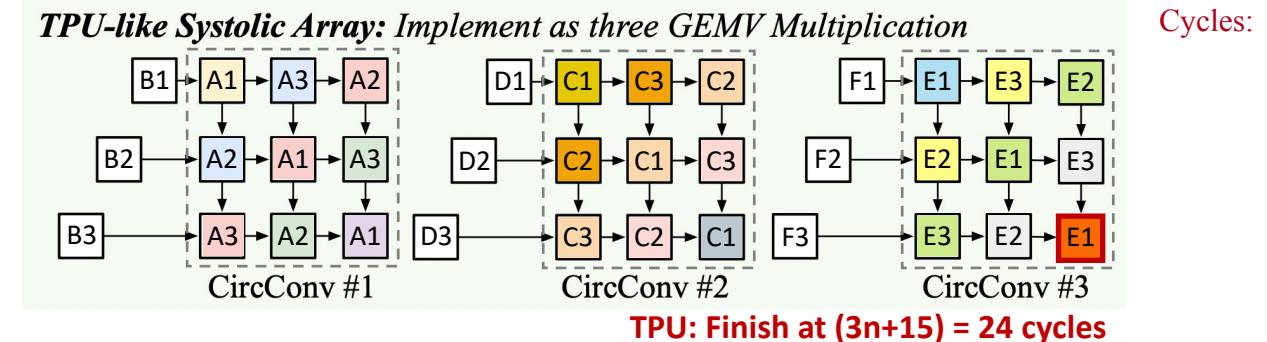
$$\text{CircConv \#2: } (C_1, C_2, C_3) \odot (D_1, D_2, D_3)$$

$$\text{CircConv \#3: } (E_1, E_2, E_3) \odot (F_1, F_2, F_3)$$

CircConv #1 Computation:

$$(A_1, A_2, A_3) \odot (B_1, B_2, B_3) =$$

$$(A_1B_1 + A_2B_2 + A_3B_3, A_1B_3 + A_2B_1 + A_3B_2, A_1B_2 + A_2B_3 + A_3B_1)$$



For symbolic operation:

- TPU-like array **suffers from** low parallelism & high memory access;

Bubble Streaming Dataflow

Vector-Symbolic Circular Convolution Example (3 CircConv):

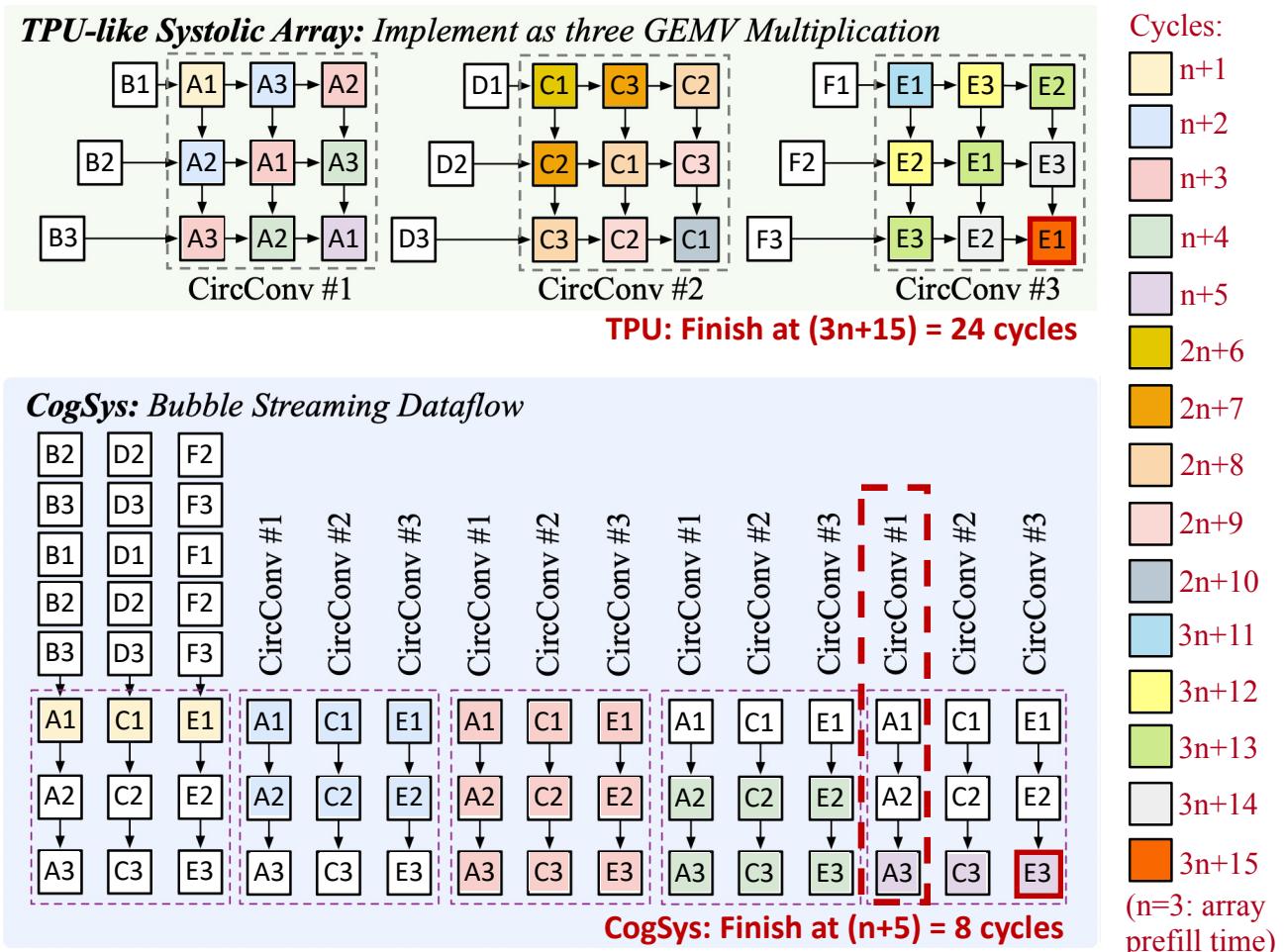
- CircConv #1: $(A_1, A_2, A_3) \odot (B_1, B_2, B_3)$
- CircConv #2: $(C_1, C_2, C_3) \odot (D_1, D_2, D_3)$
- CircConv #3: $(E_1, E_2, E_3) \odot (F_1, F_2, F_3)$

CircConv #1 Computation:

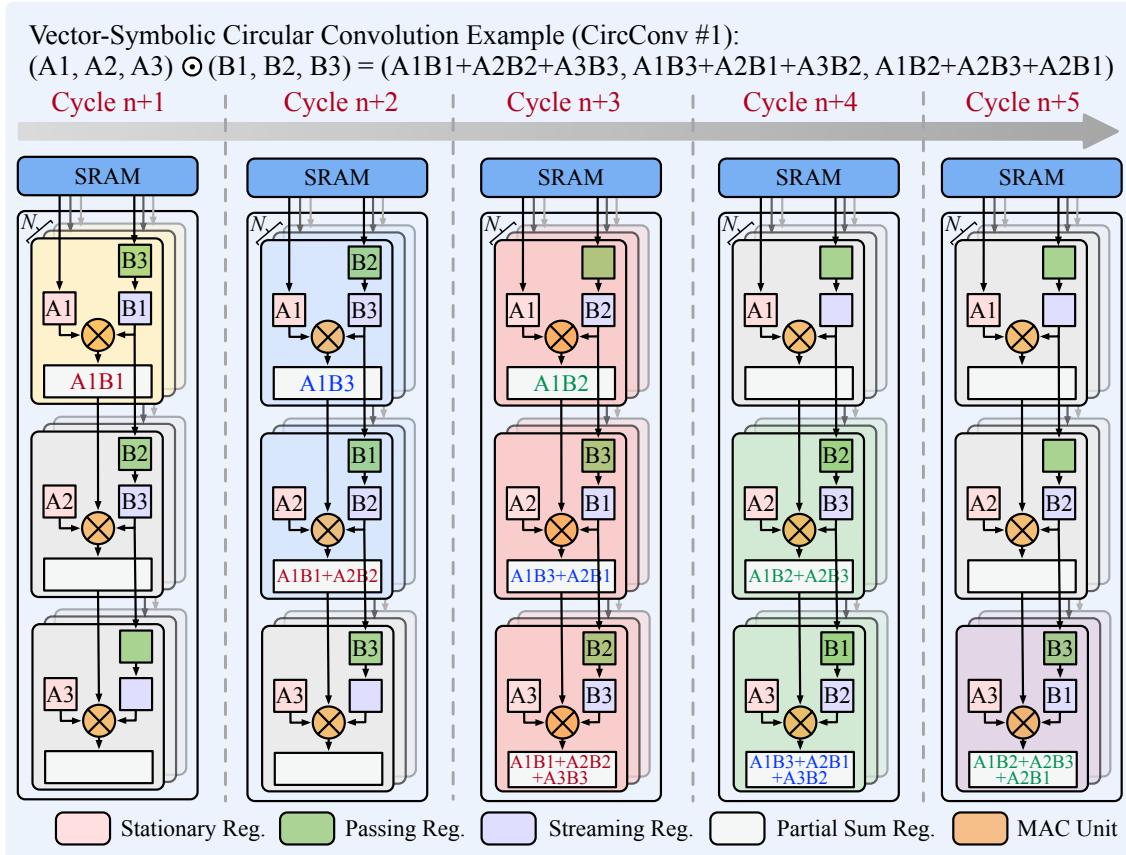
$$(A_1, A_2, A_3) \odot (B_1, B_2, B_3) = \\ (A_1B_1 + A_2B_2 + A_3B_3, A_1B_3 + A_2B_1 + A_3B_2, A_1B_2 + A_2B_3 + A_3B_1)$$

For symbolic operation:

- TPU-like array **suffers from** low parallelism & high memory access;
- Bubble streaming dataflow **improve parallelism, arithmetic intensity, and data reuse.**

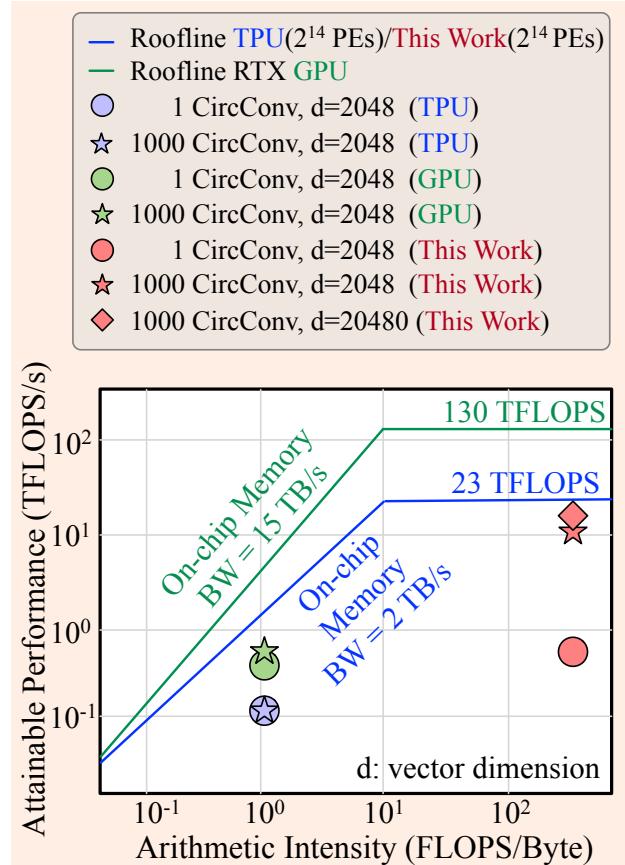
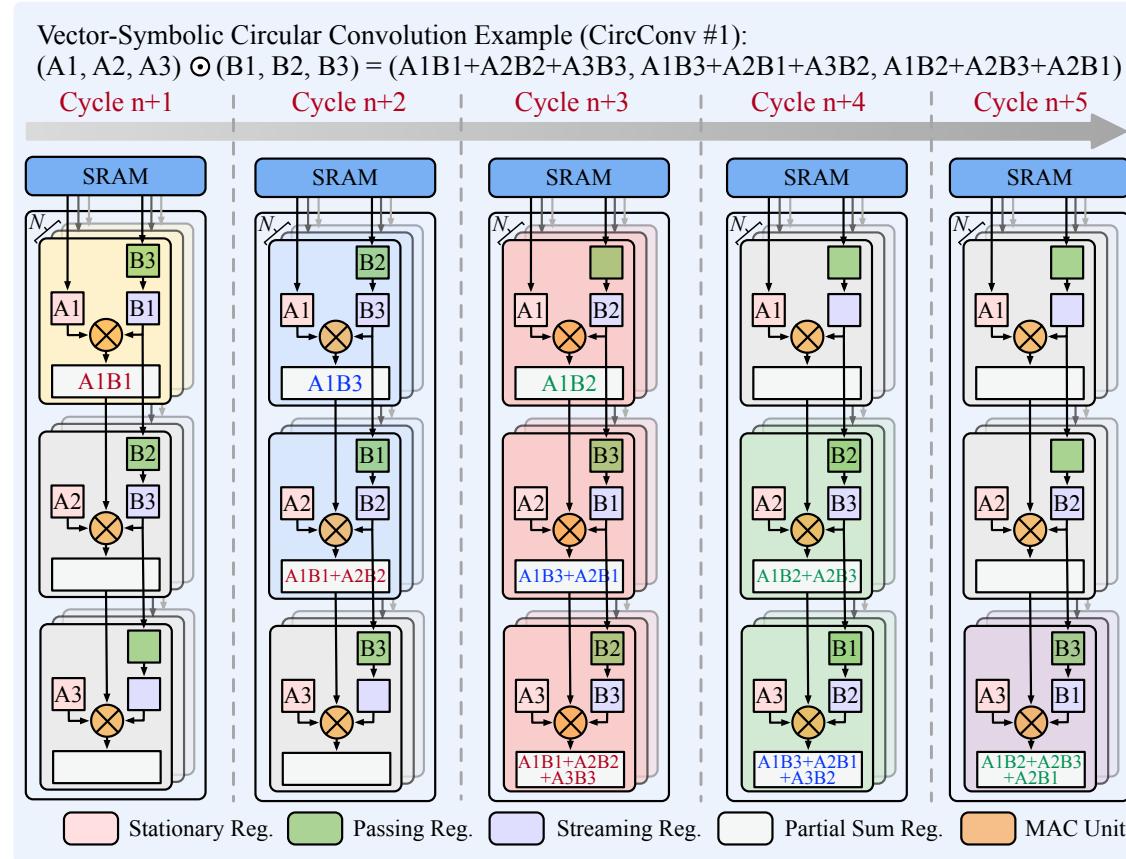


Bubble Streaming Dataflow



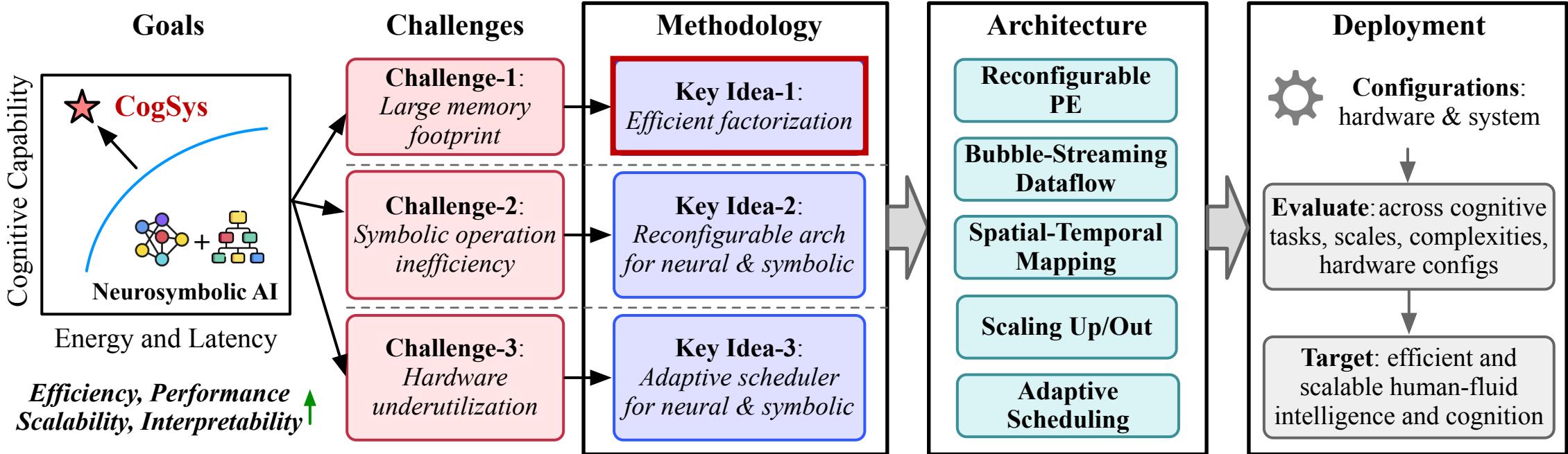
Bubble streaming dataflow flow improve parallelism, arithmetic intensity, and data reuse

Bubble Streaming Dataflow

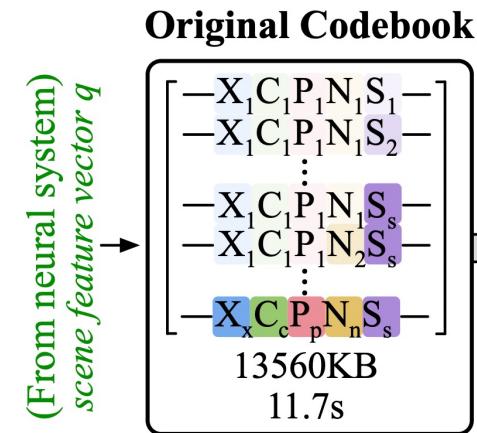


Bubble streaming dataflow flow improve parallelism, arithmetic intensity, and data reuse

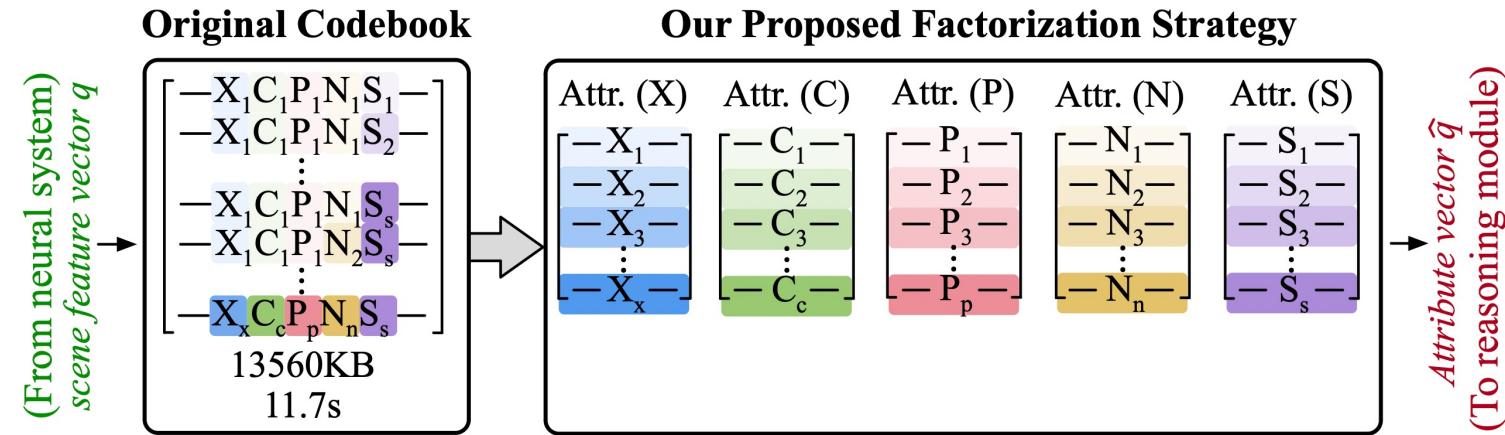
Our Methodology



Algorithm Optimization – Efficient Factorization

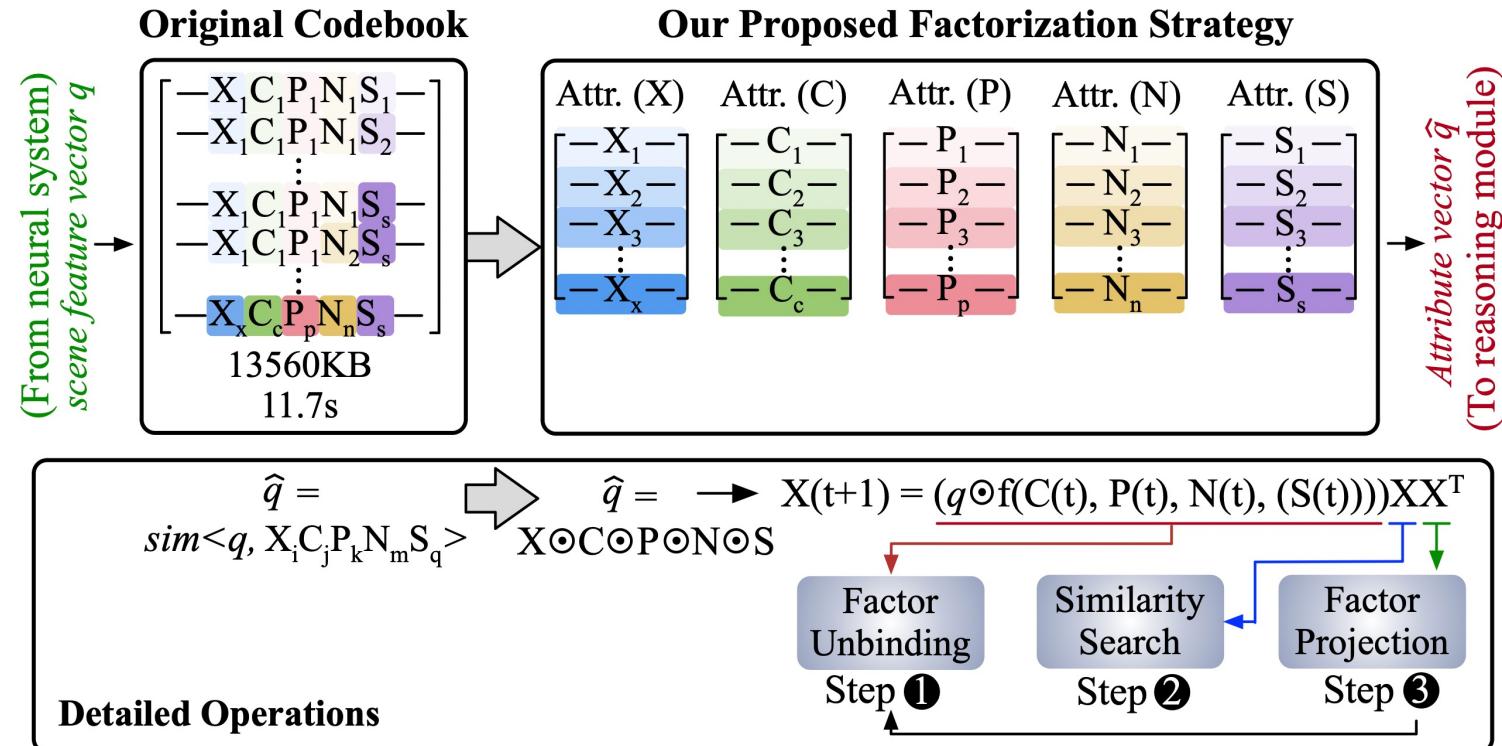


Algorithm Optimization – Efficient Factorization



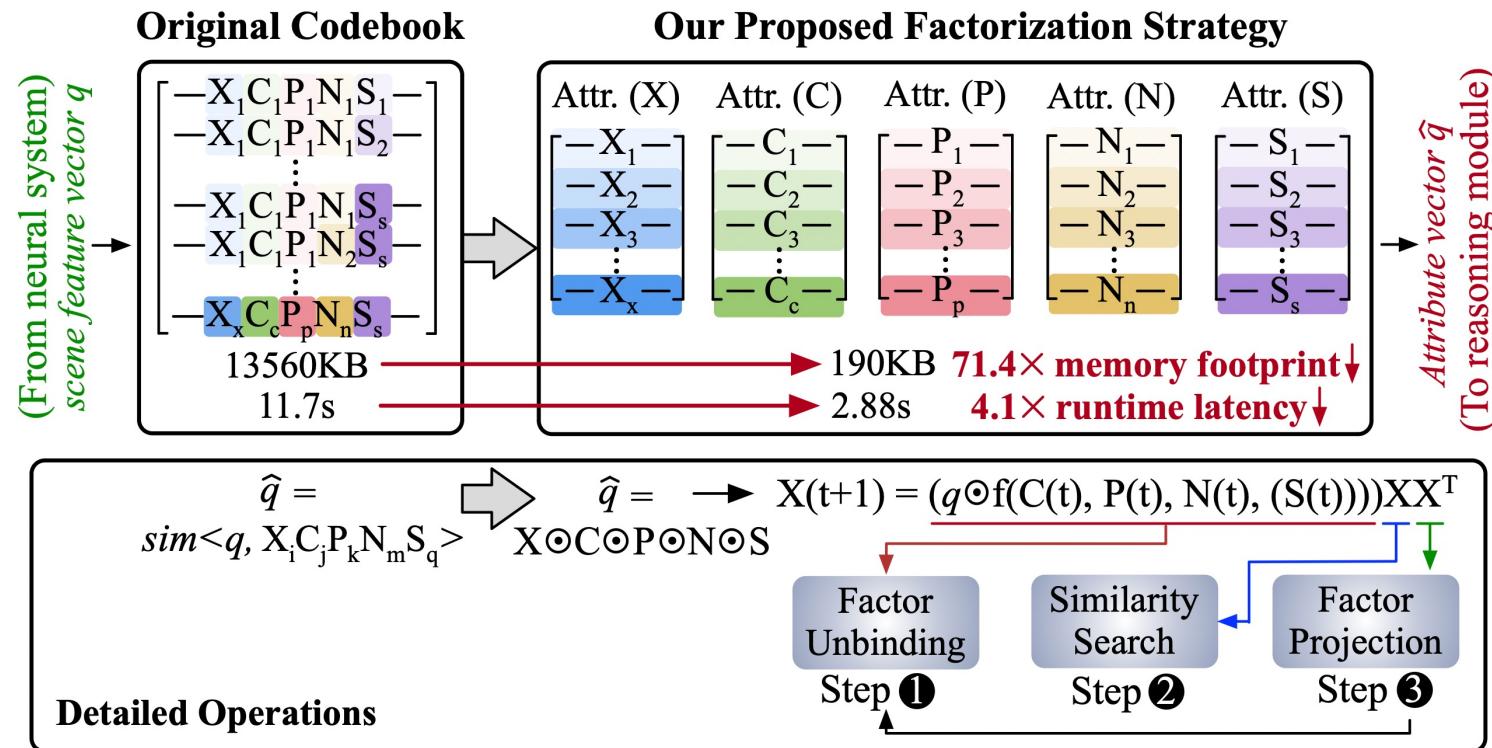
Factorization **disentangles** large symbolic knowledge codebook into small volume of attributes

Algorithm Optimization – Efficient Factorization



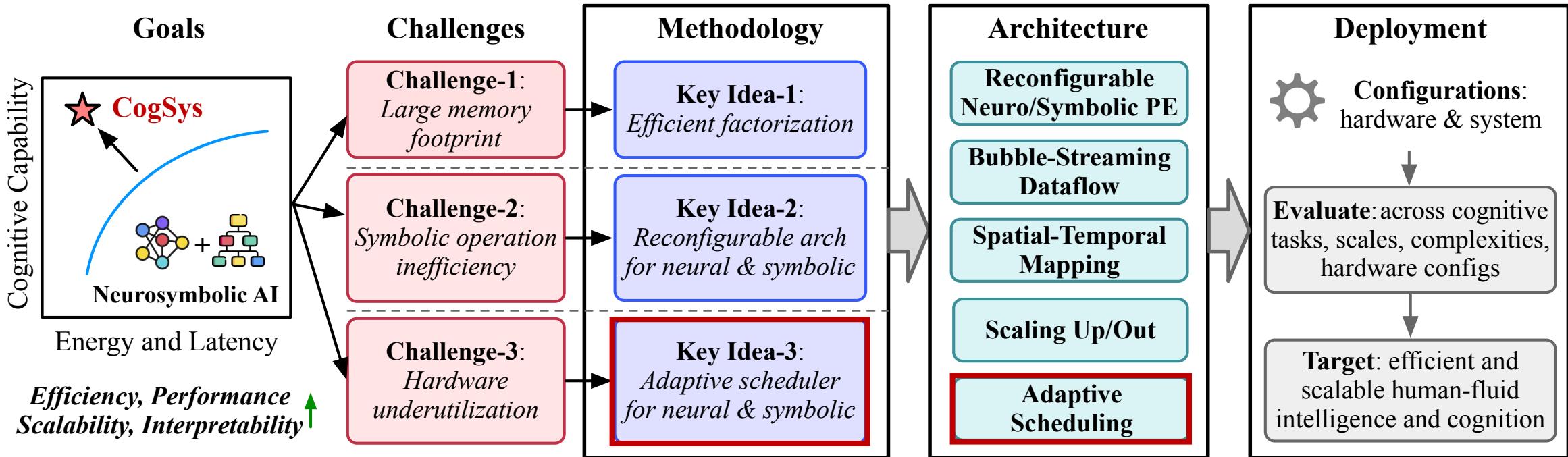
Factorization **disentangles** large symbolic knowledge codebook into small volume of attributes

Algorithm Optimization – Efficient Factorization

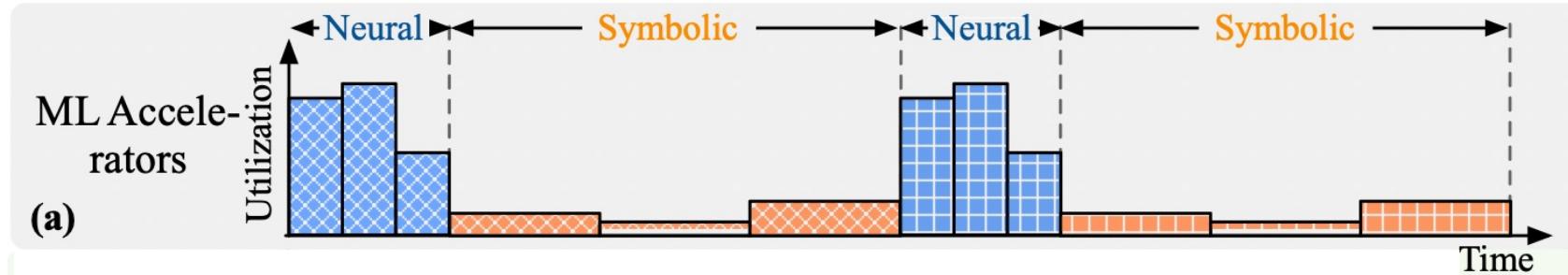


Factorization **disentangles** large symbolic knowledge codebook into small volume of attributes, thus **reducing computational time and space complexity**

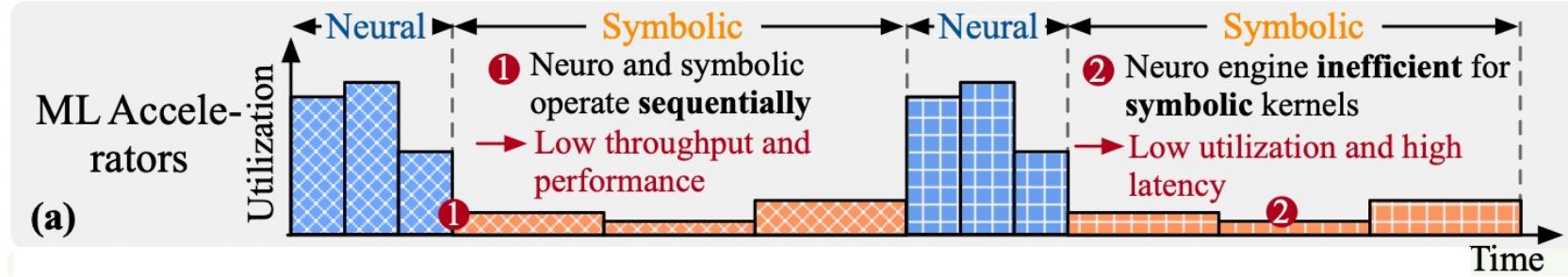
CogSys: Co-Design for Neuro-Symbolic AI



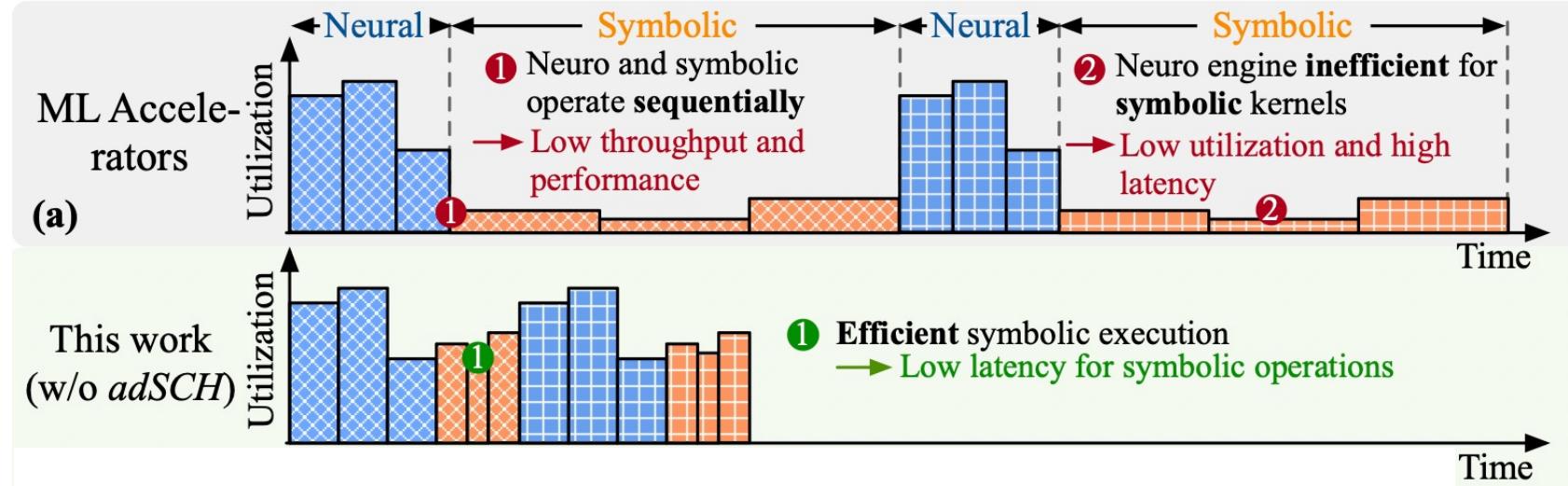
System Optimization - Adaptive Scheduling



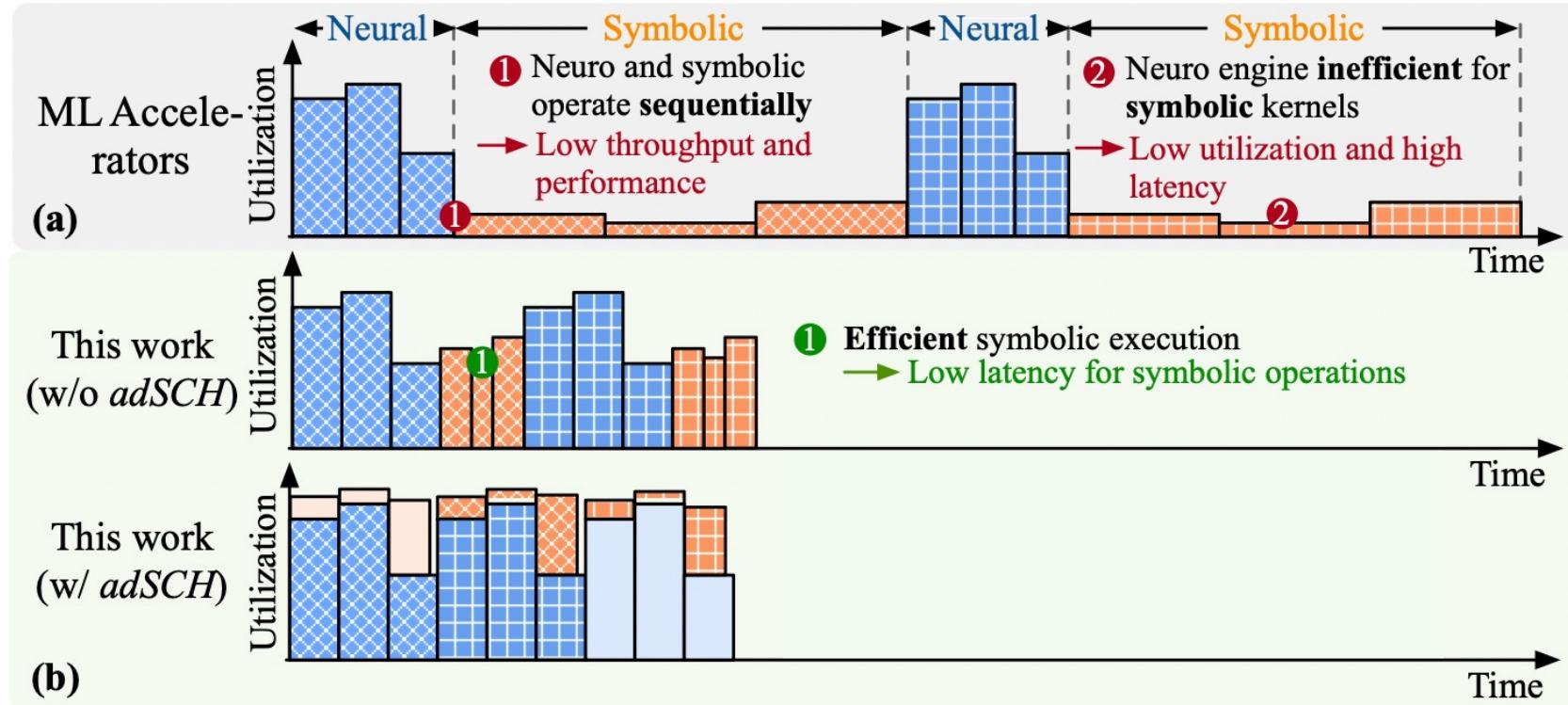
System Optimization - Adaptive Scheduling



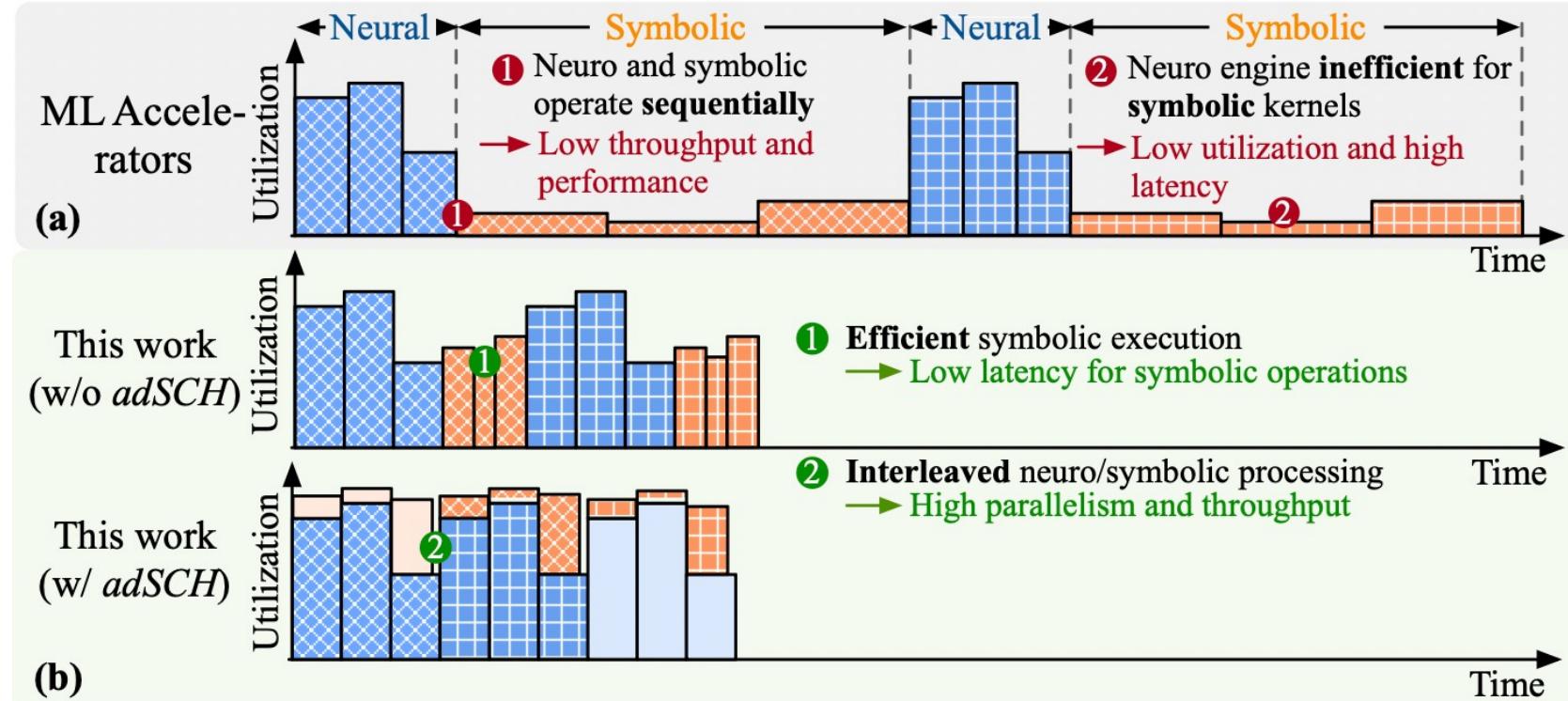
System Optimization - Adaptive Scheduling



System Optimization - Adaptive Scheduling

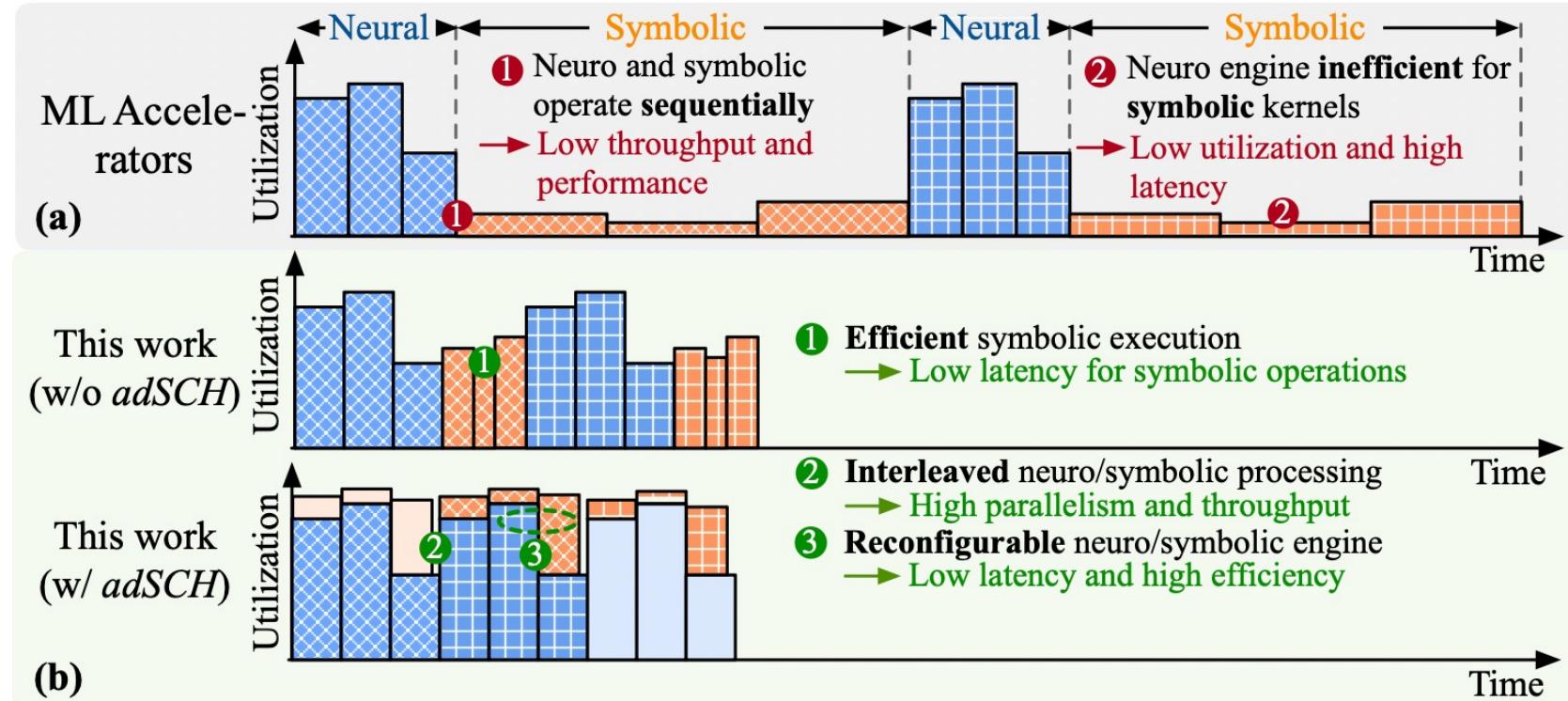


System Optimization - Adaptive Scheduling



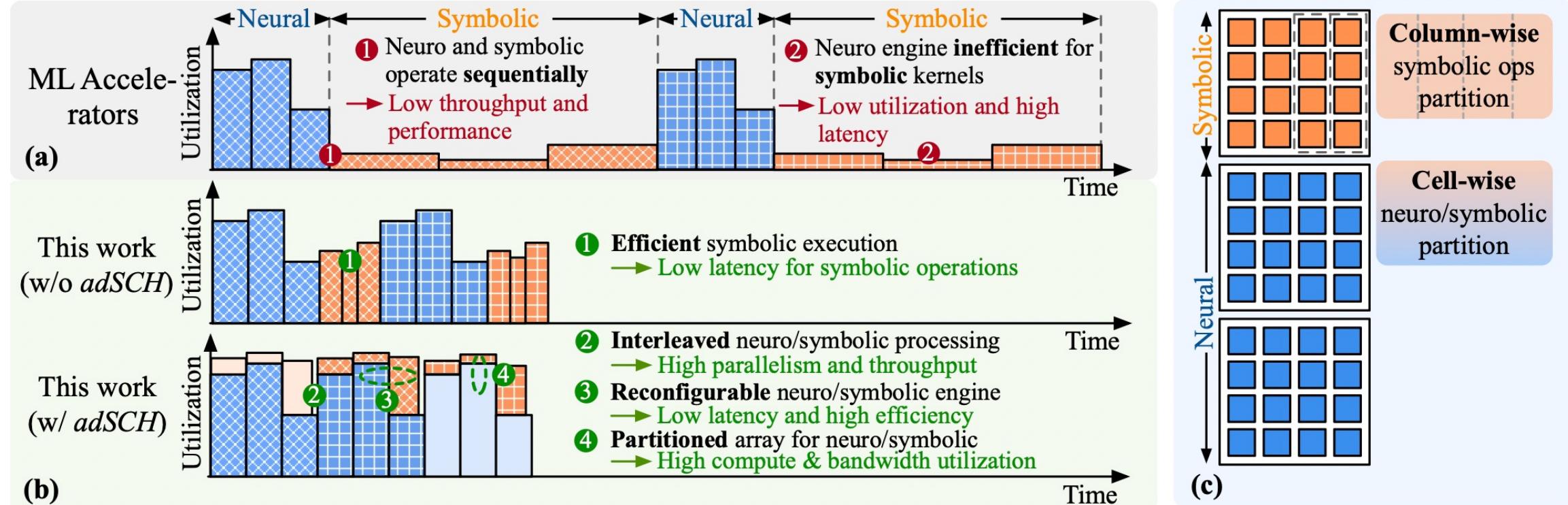
Adaptive scheduling enables **interleaved**

System Optimization - Adaptive Scheduling



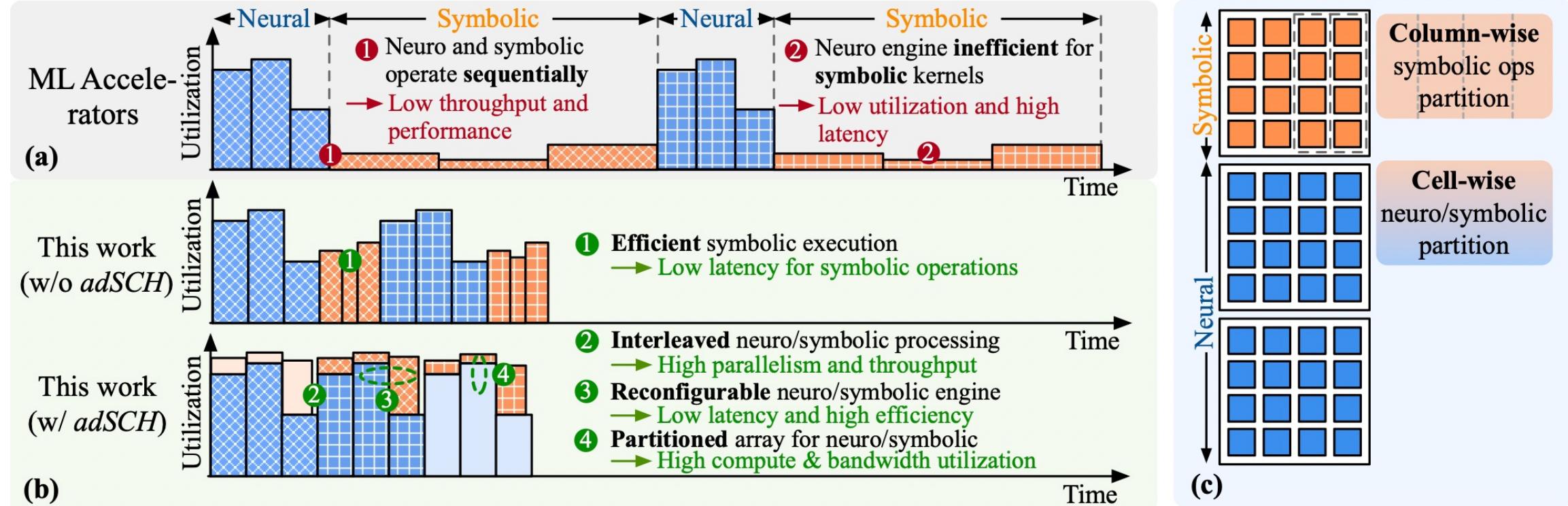
Adaptive scheduling enables **interleaved** and **reconfigurable** neuro/symbolic processing

System Optimization - Adaptive Scheduling



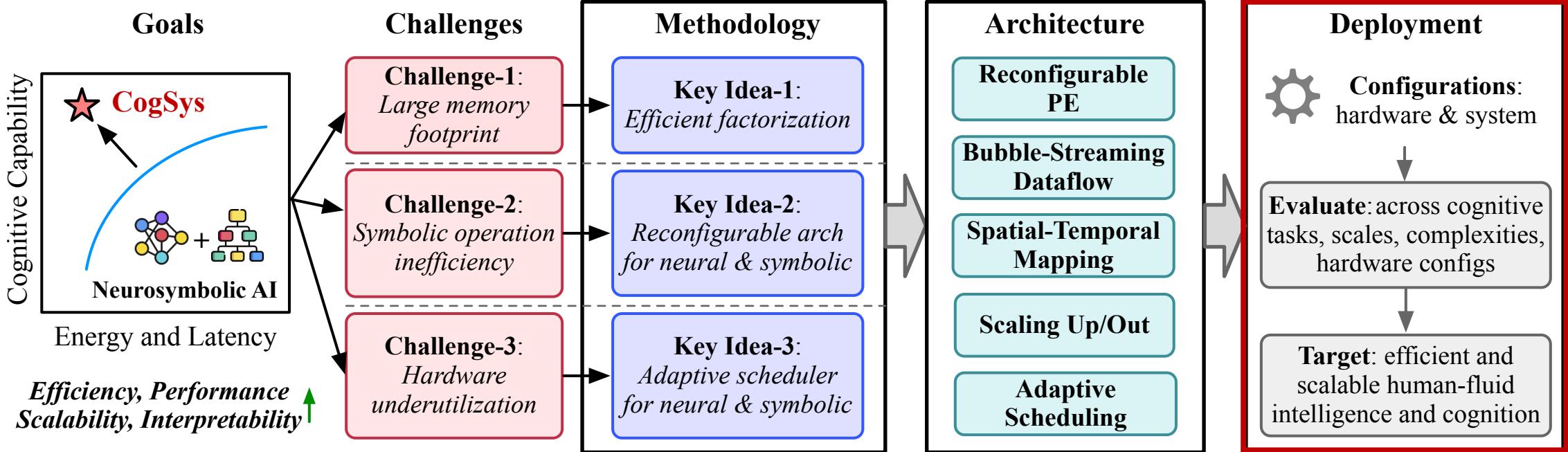
Adaptive scheduling enables **interleaved** and **reconfigurable** neuro/symbolic processing with **partitioned array**

System Optimization - Adaptive Scheduling



Adaptive scheduling enables **interleaved** and **reconfigurable** neuro/symbolic processing with **partitioned array**, improving parallelism, latency, efficiency, and utilization

CogSys: Co-Design for Neuro-Symbolic AI



Evaluation – Setup and Accelerator Layout

Layout of Neuro-Symbolic Accelerator



Accelerator Specs

Technology	28 nm	Frequency	600 MHz
#Arrays	16	Voltage	1 V
Size of Each Array	32x32	Power	1.48 W
SRAM	4.5 MB	Area	4.9 mm ²

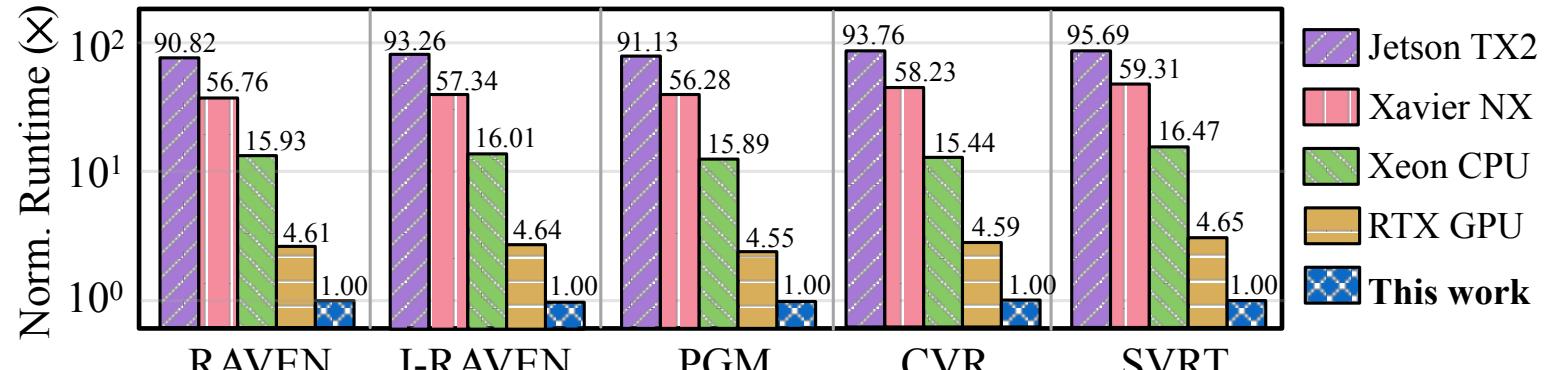
- **Task:** Cognitive reasoning tasks
- **Reasoning datasets:**
 - RAVEN, I-RAVEN, PGM, CVR, SVRT
- **Neuro-symbolic workloads:**
 - NVSA, MIMONet, LVRF
- **Hardware baseline:**
 - Jetson TX2, Xavier NX, RTX GPU, Xeon CPU
 - ML accelerators (TPU, MTIA, Gemmini)

Evaluation – Algorithm Performance

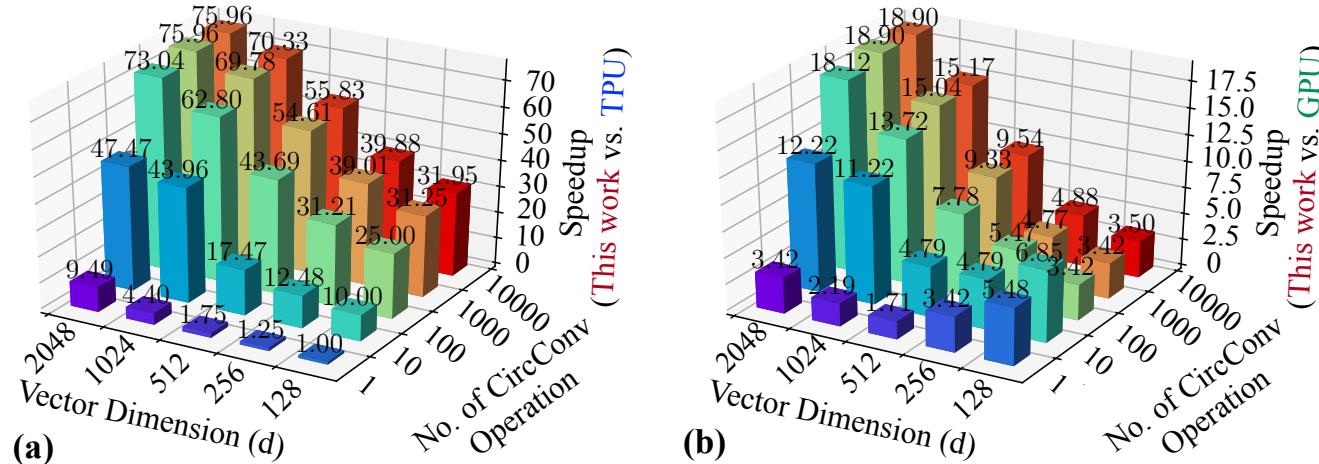
Dataset	Neurosymbolic Model			Non-neurosymbolic		Human
	NVSA	Our Design (+Algo Opt.)	Our Design (+Quant.)	ResNet18	GPT-4	
RAVEN	98.5%	98.9%	98.7%	53.4%	89.0%	84.4%
I-RAVEN	99.0%	99.0%	98.8%	40.3%	86.0%	78.6%
PGM	68.3%	68.7%	68.4%	36.8%	56.0%	N/A
#Parameters	38 MB	32 MB	8 MB	42 MB	1.7 TB	N/A

- **Better Reasoning Capability:** neurosymbolic methods achieve high accuracy across reasoning tasks than NNs and human.
- **Smaller Memory Footprint:** neurosymbolic methods consume much less #parameter than NNs (e.g., LLM).

Evaluation – Hardware Performance

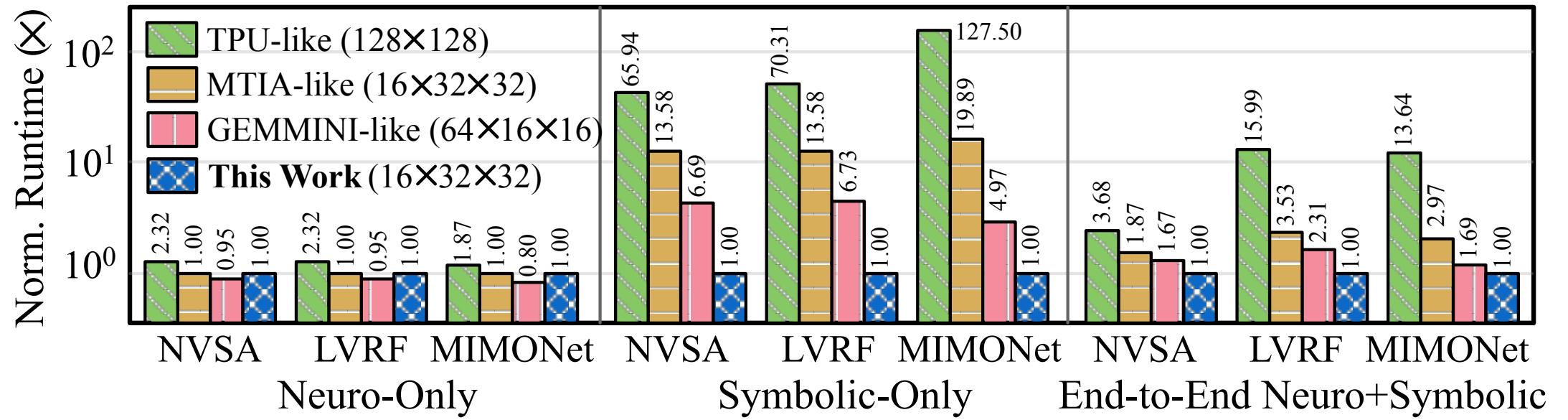


4x - 90x speedup
compared to CPU/GPU



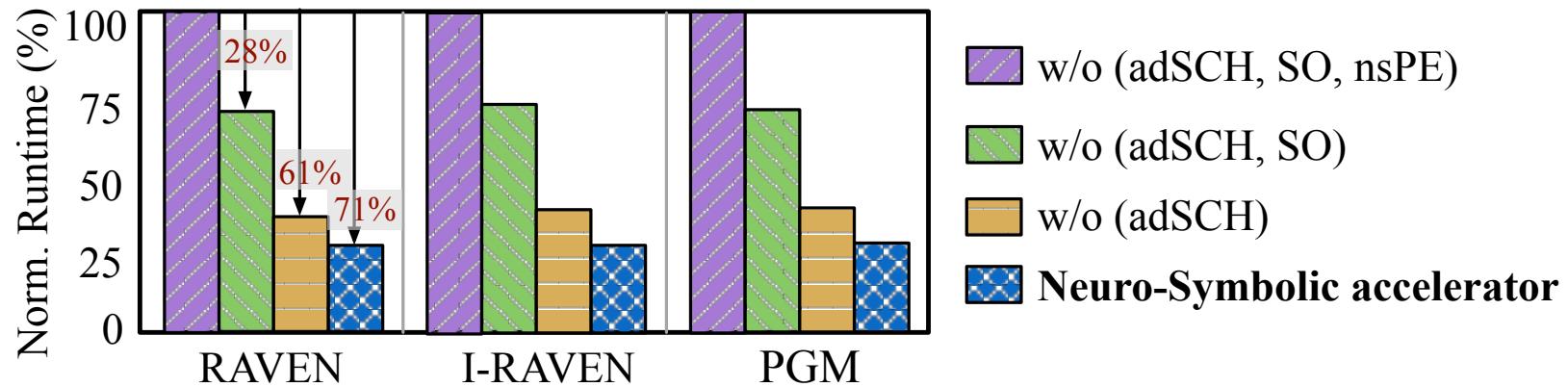
Symbolic operation:
75x speedup to TPU
18x speedup to GPU

Evaluation – Hardware Performance



Compared with ML accelerators: similar neuro latency, **7-120x symbolic speedup**,
2-16x end-to-end neuro-symbolic speedup

Evaluation – Ablation Study



Proposed **scheduling**,
reconfigurable **PE**,
bubble streaming
dataflow are effective

Neurosymbolic Cognitive Solution Algorithm @ Hardware	Normalized Runtime (%) on RAVEN I-RAVEN PGM CVR SVRT				
NVSA @ Xavier NX	100	100	100	100	100
Proposed Algorithm @ Xavier NX	89.5%	88.9%	90.7%	87.6%	88.4%
Proposed Algorithm @ Proposed Accelerator	1.76%	1.74%	1.78%	1.72%	1.69%

**Algorithm-system-
hardware co-design**
is critical



Key Observations:

Compared with systolic arrays that only support neural,
CogSys provides **reconfigurable support for neural and
symbolic** operations with **only 4.8% area overhead**.

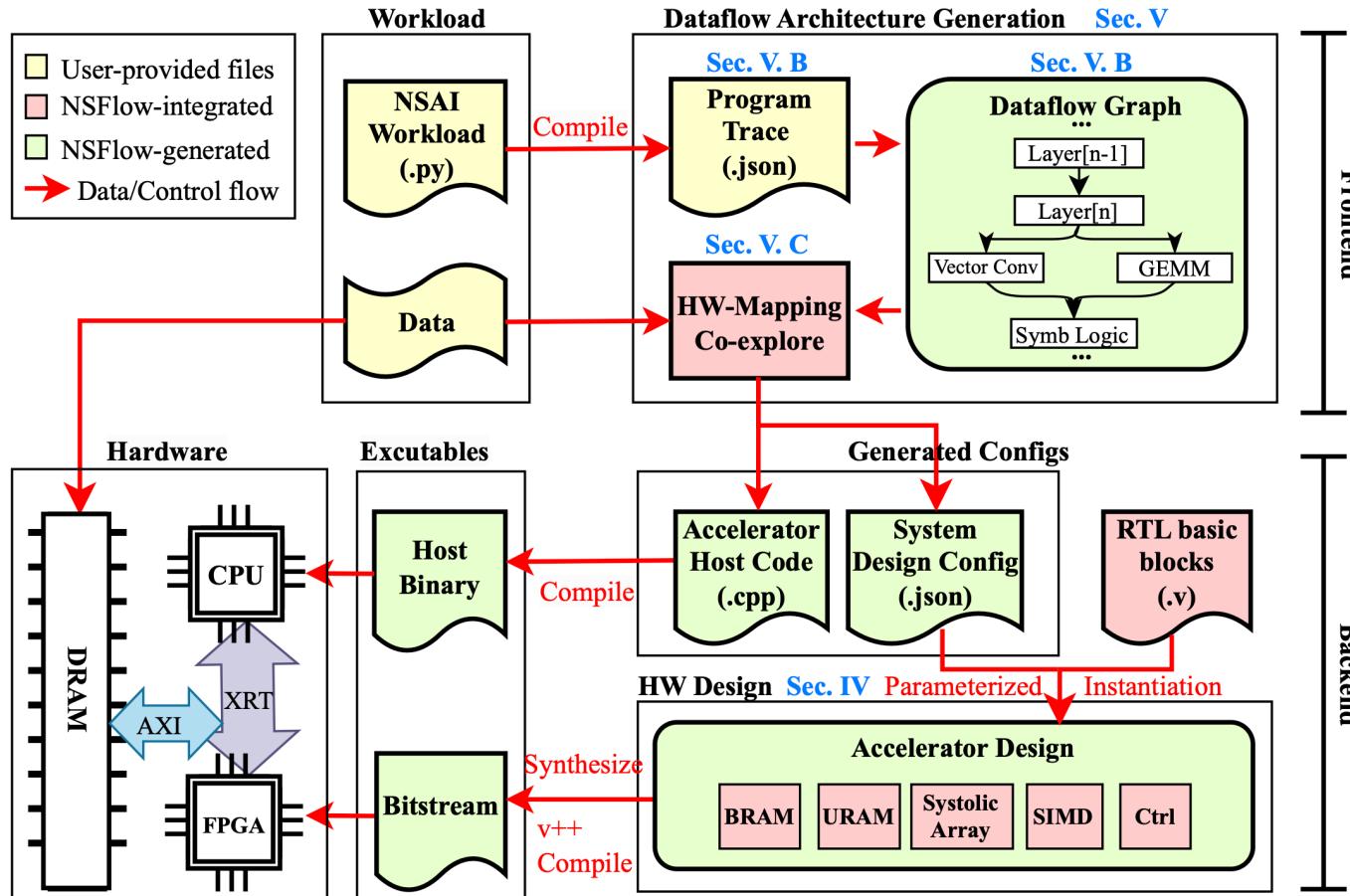
Our design achieves **0.3s latency** per cognition task, with
1.18W power consumption.



Research Question:

How to **automate** this neuro-symbolic
architecture **design** process?

End-to-End FPGA Deployment for Neuro-Symbolic AI



Frontend: dataflow arch generator

- Step 1: Extract execution trace
- Step 2: Generate dataflow graph
- Step 3: HW-mapping co-exploration

Backend: FPGA deployment

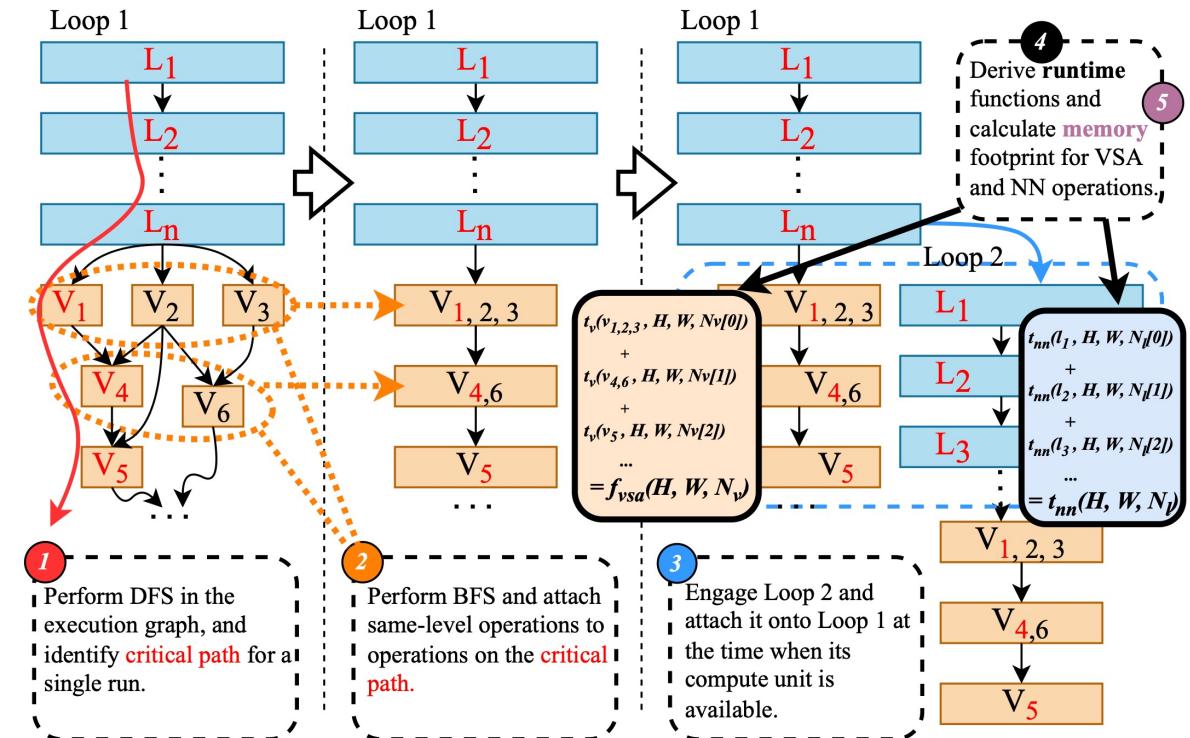
- Step 1: Pre-define hardware template
- Step 2: Configure design parameters
- Step 3: Synthesize and compile RTL

Hanchen Yang*, Zishen Wan*, Ritik Raj, Joongun Park, Ziwei Li, Ananda Samajdar, Arijit Raychowdhury, Tushar Krishna, “**NSFlow: An End-to-End FPGA Framework with Scalable Dataflow Architecture for Neuro-Symbolic AI**”, in DAC 2025

Frontend – Dataflow architecture Generation

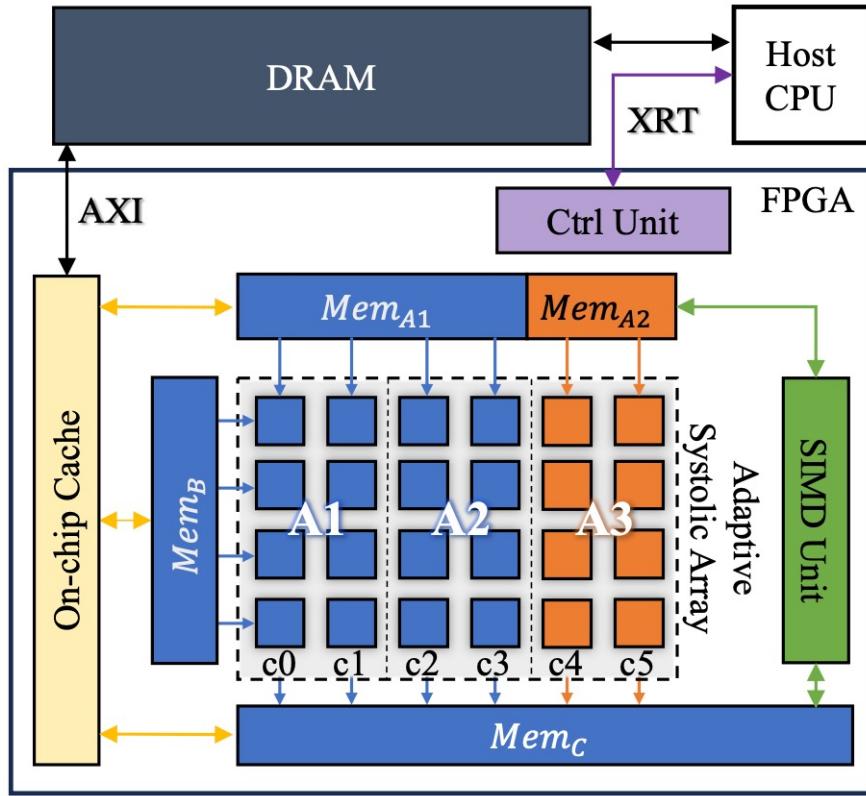
```
graph():
    ...
    // Neuro Operation - CNN (Resnet18)
    %relu_1[16, 64, 160, 160] : call_module[relu](args = (%bn1
        [16, 64, 160, 160]))
    %maxpool_1[16, 64, 160, 160] : call_module[maxpool](args =
        (%relu_1[16, 64, 160, 160]))
    %conv2d_1[16, 64, 160, 160] : call_module[conv2d](args =
        (%maxpool_1[16, 64, 160, 160]))
    ...
    // Symbolic Operations
    // Inverse binding of two block codes vectors by
    // blockwise circular correlation
    %inv_binding_circular_1[1, 4, 256] : call_function[nvsa.
        inv_binding_circular](args = (%vec_0[1, 4, 256], %
        vec_1[1, 4, 256]))
    %inv_binding_circular_2[1, 4, 256] : call_function[nvsa.
        inv_binding_circular](args = (%vec_3[1, 4, 256], %
        vec_4[1, 4, 256]))
    // Compute similarity between two block codes vectors
    %match_prob_1[1] : call_function[nvsa.match_prob](args
        = (%inv_binding_circular_1[1, 4, 256], %vec_2
        [1, 4, 256]))
    // Compute similarity between a dictionary and a batch
    // of query vectors
    %match_prob_multi_batched_1[1]: call_function[nvsa.
        match_prob_multi_batched](args = (%
            inv_binding_circular_2[1, 4, 256], %vec_5[7, 4, 256]))
    %sum_1[1] : call_function[torch.sum](args = (%
        match_prob_multi_batched_1[1]))
    %clamp_1[1] : call_function[torch.clamp](args = (%sum_1
        [1]))
    %mul_1[1] : call_function[operator.mul](args = (%
        match_prob_1[1], %clamp_1[1]))
    ...
    ...
```

Extract workload execution trace

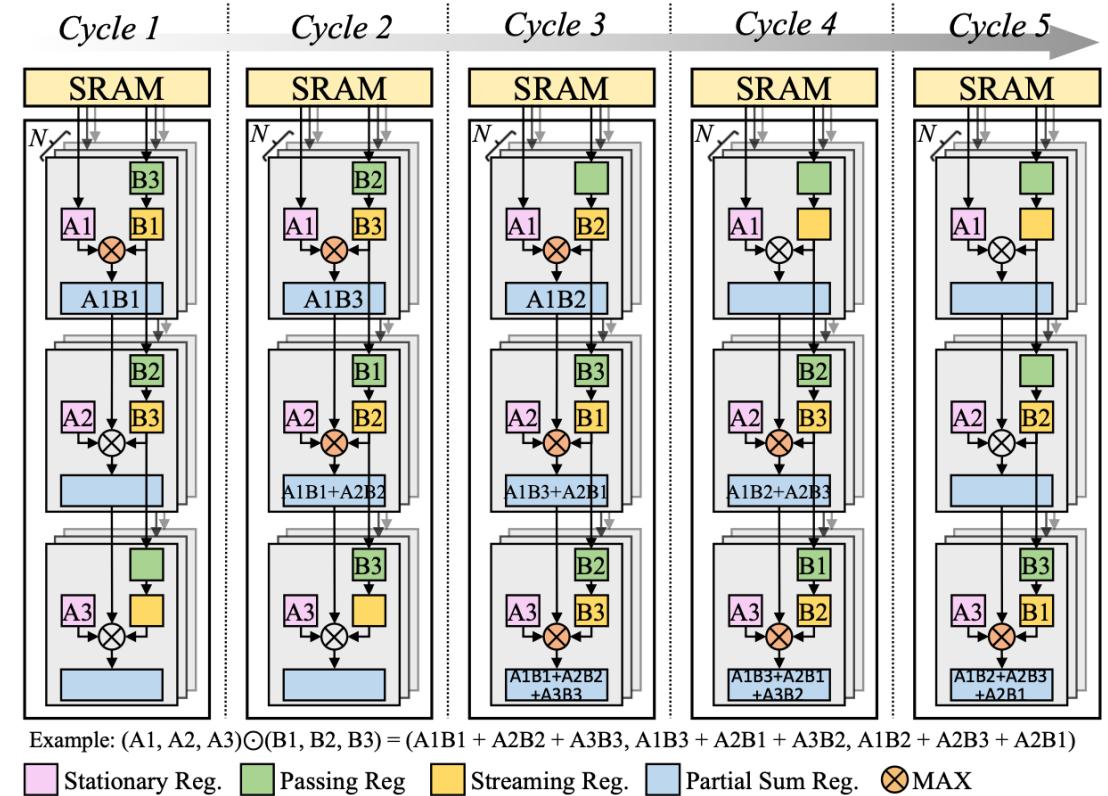


Generate dataflow graph &
two-stage HW-mapping co-exploration

Backend – FPGA Deployment



Pre-defined architecture template



Dataflow & configure design parameters

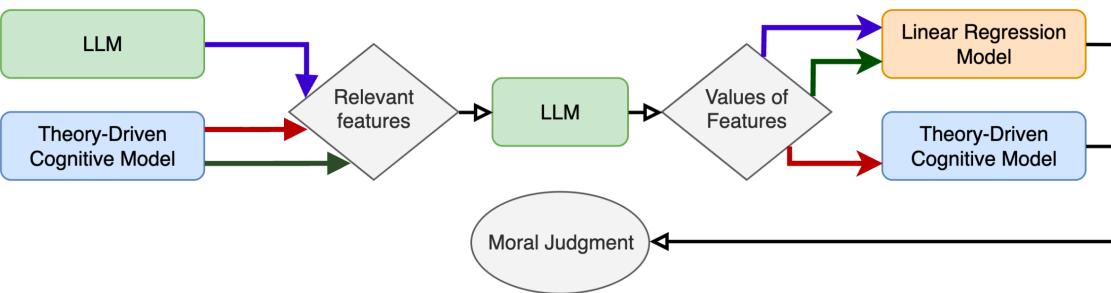


Research Question:

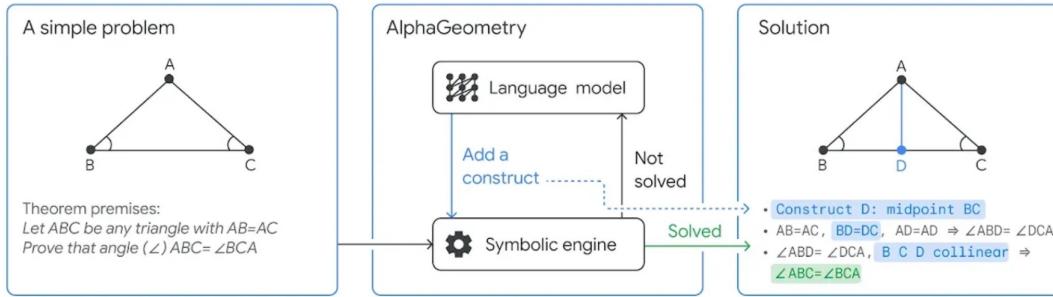
Can we **tape out** the design and explore how emerging **memory technology** can further help?

Temporarily withheld due to chip under review

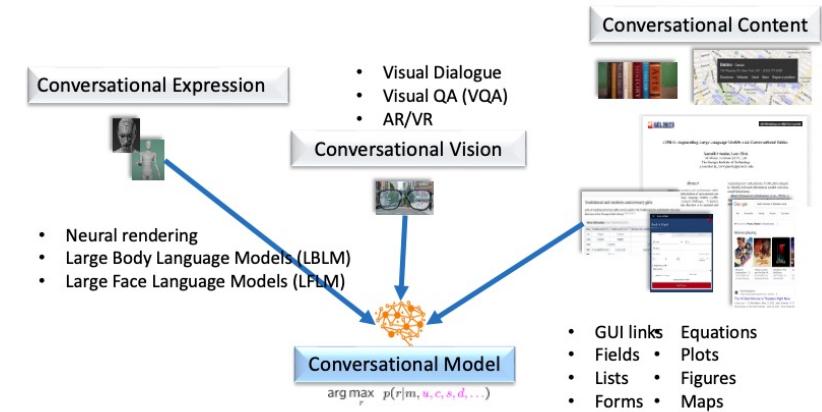
Looking Ahead: LLM + Neurosymbolic



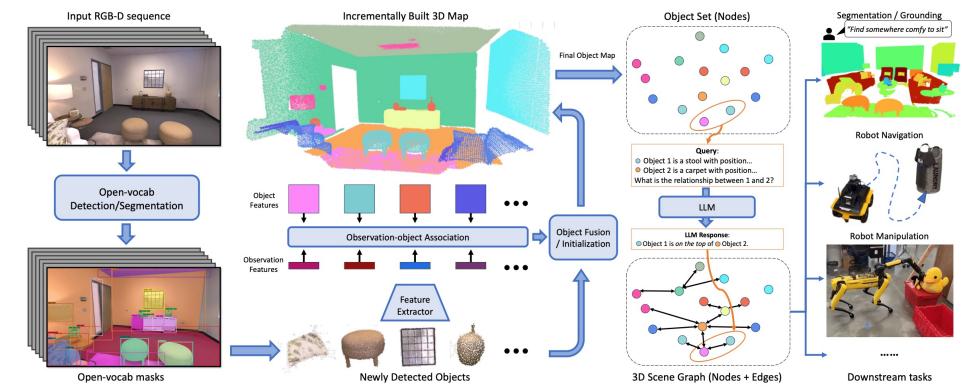
Towards safe and trustworthy AI System:
LLM + cognitive model for human moral judgment



Towards logical reasoning AI System:
LLM + symbolic solver for scientific computing



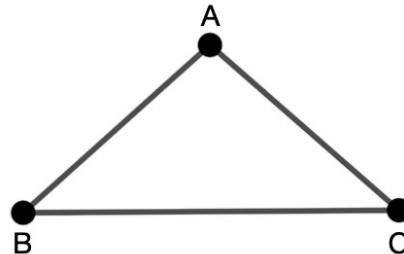
Towards human-centered AI System:
LLM + knowledge base for conversational reasoning



Towards intelligent AI System:
LLM + concept graph for intelligent autonomous system

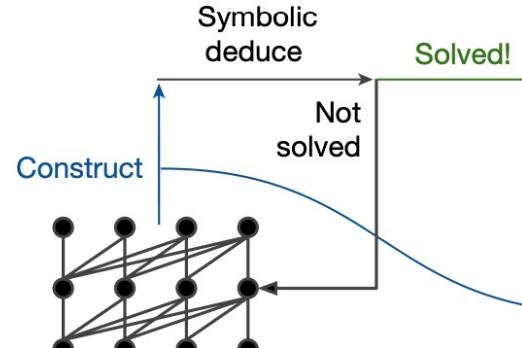
Example: How Neurosymbolic Advances LLM Reasoning

a A simple problem

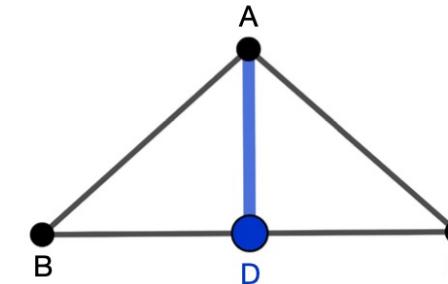


"Let ABC be any triangle with $AB = AC$.
Prove that $\angle ABC = \angle BCA$."

b AlphaGeometry



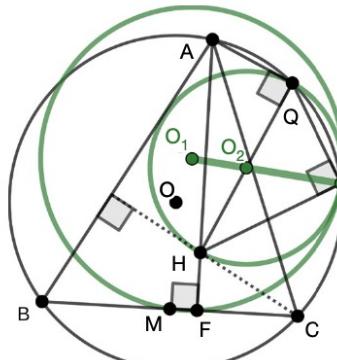
d Solution



Construct D: midpoint BC,
 $AB=AC$, $BD = DC$, $AD=AD \Rightarrow \angle ABD=\angle DCA$ [1]
[1], $B C D$ collinear $\Rightarrow \angle ABC=\angle BCA$

e IMO 2015 P3

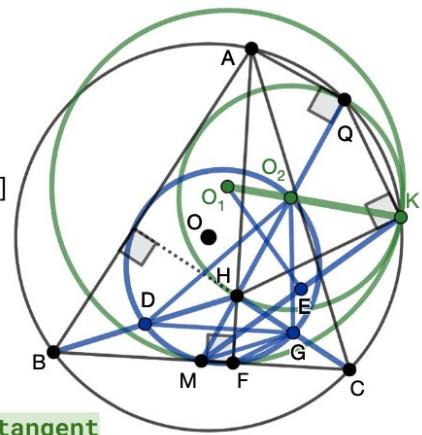
"Let ABC be an acute triangle. Let (O) be its circumcircle, H its orthocenter, and F the foot of the altitude from A. Let M be the midpoint of BC. Let Q be the point on (O) such that $QH \perp QA$ and let K be the point on (O) such that $KH \perp KQ$. Prove that the circumcircles (O_1) and (O_2) of triangles FKM and KQH are tangent to each other."



Alpha-
Geometry

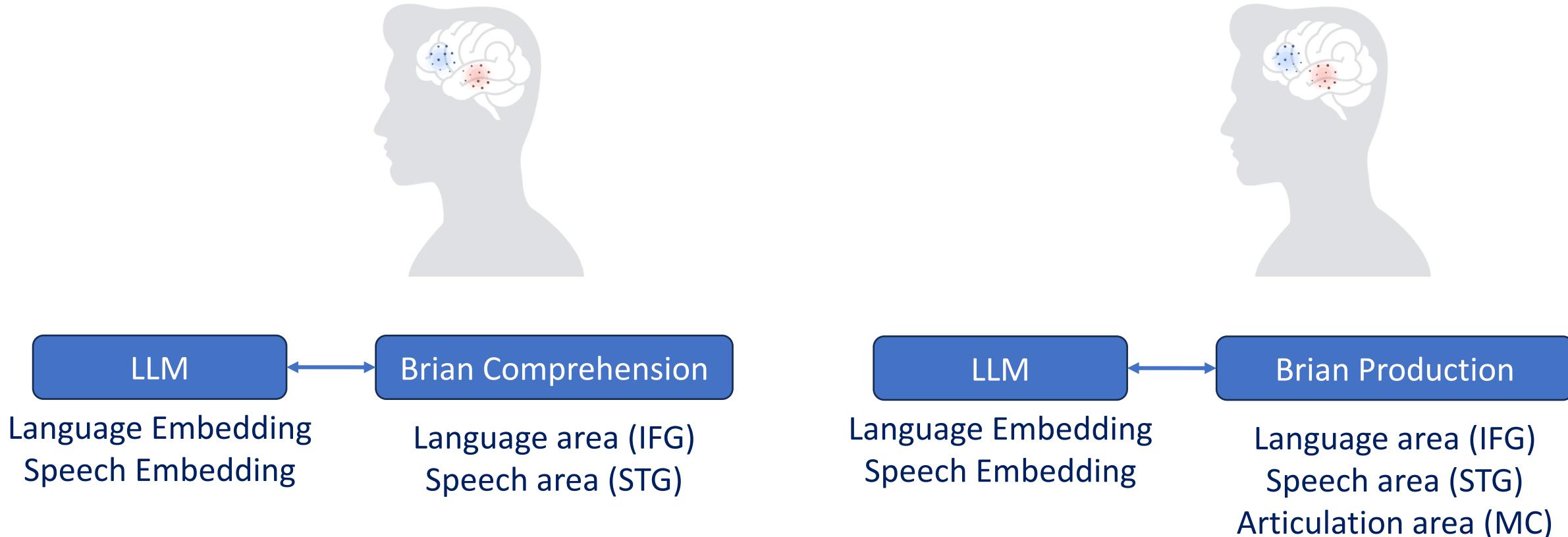
f Solution

Construct D: midpoint BH [a]
[a], O_2 midpoint HQ $\Rightarrow BQ \parallel O_2D$ [20]
...
Construct G: midpoint HC [b] ...
 $\angle GMD = \angle GO_2D \Rightarrow M O_2 G D$ cyclic [26]
[a], [b] $\Rightarrow BC \parallel DG$ [30]
...
Construct E: midpoint MK [c]
..., [c] $\Rightarrow \angle KFC = \angle KO_1E$ [104]
...
 $\angle FKQ_1 = \angle FKQ_2 \Rightarrow KO_1 \parallel KO_2$ [109]
[109] $\Rightarrow O_1 O_2 K$ collinear $\Rightarrow (O_1)(O_2)$ tangent

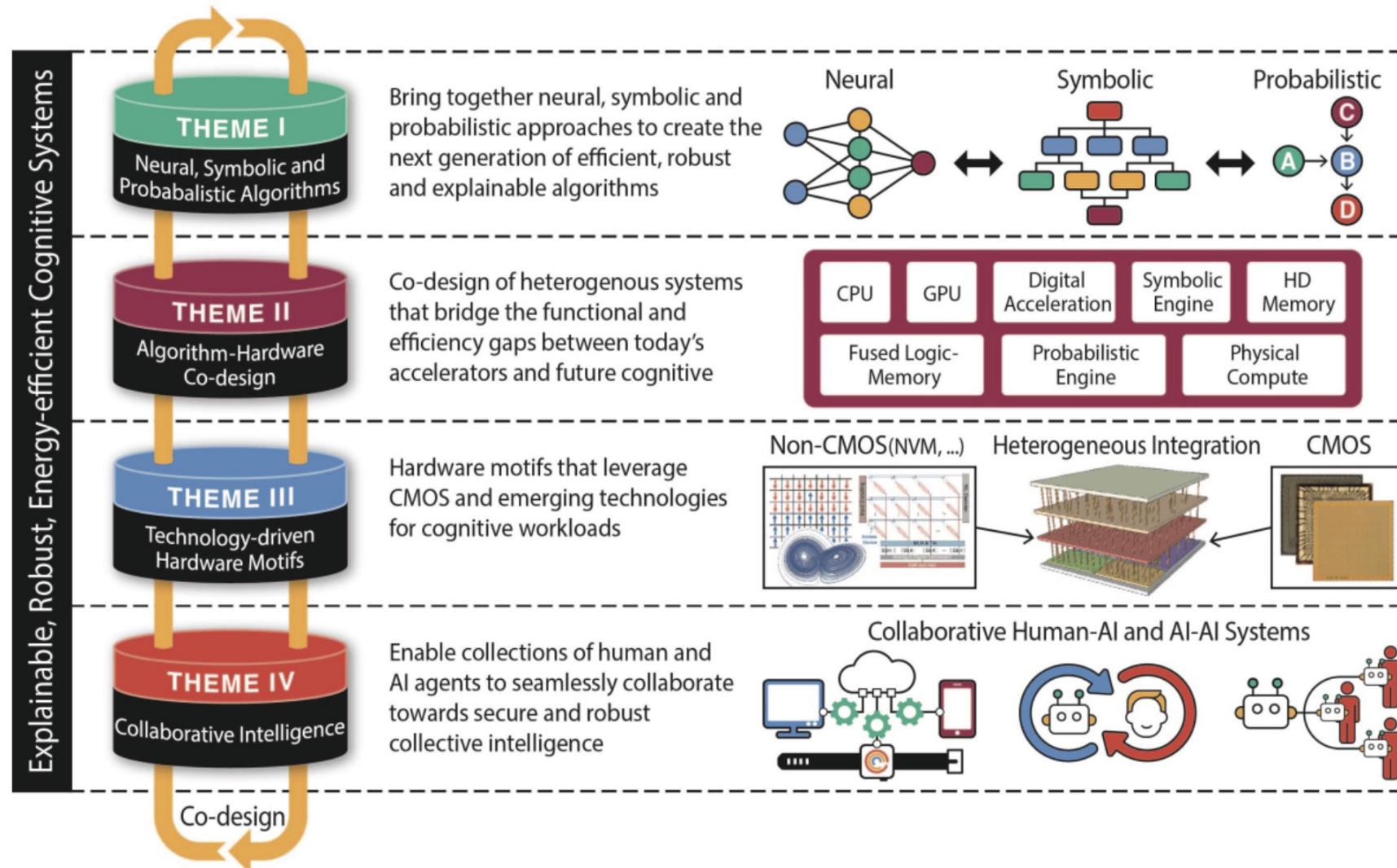


Trinh et al, "Solving Olympiad Geometry without Human Demonstrations", Nature 2024

Vision: LLM Representation <-> Human Brian



Approach: Co-Design of Cognitive Systems

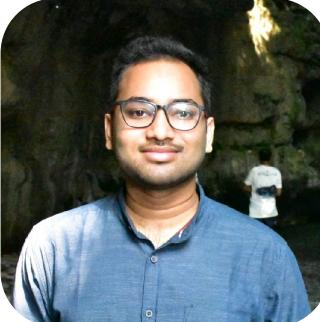


Our Team

Students



Zishen Wan



Ritik Raj



Hanchen Yang



Che-Kai Liu

PIs



Arijit Raychowdhury



Tushar Krishna

Collaborators



Ananda Samajdar

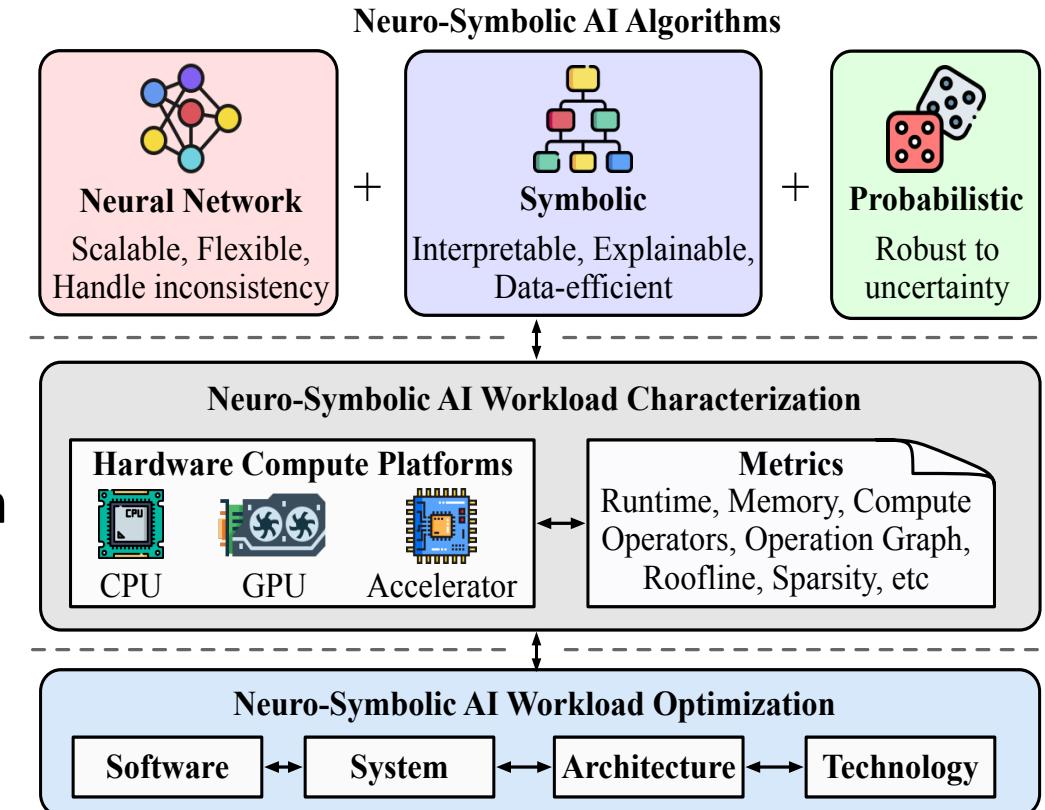
Summary

- **Motivation**

- Neuro-symbolic AI is a promising paradigm towards next-generation cognitive AI
- Challenge: inefficiency on off-the-shelf HW

- **Approach**

- Characterize neuro-symbolic workloads
- Identify potential inefficiency reasons
- Optimize neuro-symbolic system via co-design
 - **Architecture:** Reconfig PE, dataflow, mapping
 - **System:** adaptive workload scheduling
 - **FPGA:** automated end-to-end FPGA deployment
 - **ASIC SoC:** programmable neuro-symbolic SoC
 - Achieve **efficient and scalable neuro-symbolic** execution across agentic reasoning tasks





Demystifying Neuro-Symbolic AI for Software-Hardware Co-Design

Zishen Wan

PhD Student @ School of ECE, Georgia Tech

Advisors: Prof. Arijit Raychowdhury, Prof. Tushar Krishna

Web: <https://zishenwan.github.io>

Email: zishenwan@gatech.edu

Seminar @ Google, April 17, 2025

