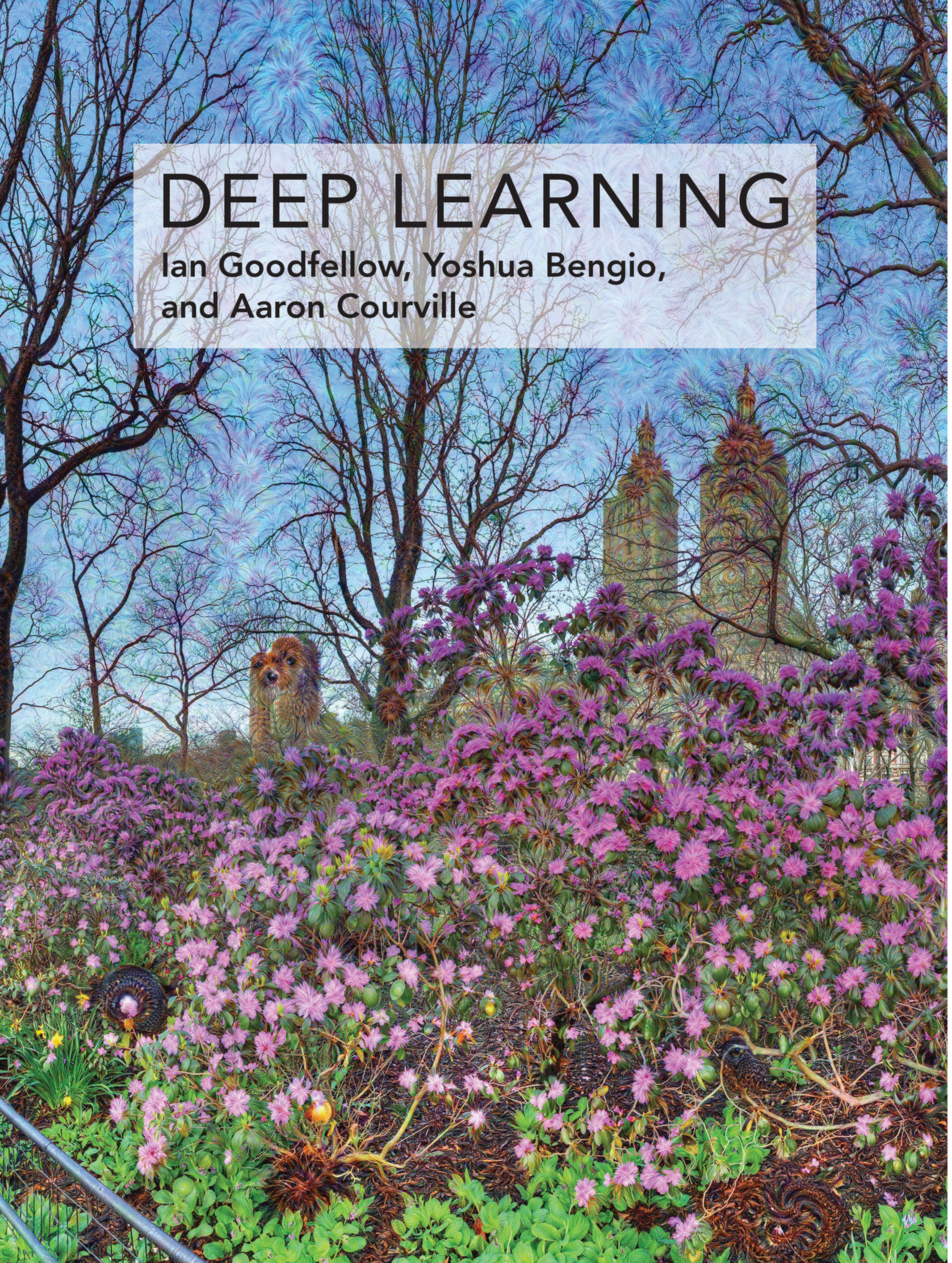


TOWARDS CAUSAL REPRESENTATION LEARNING:

AN AI & DEEP LEARNING PERSPECTIVE ON CAUSALITY

YOSHUA BENGIO

30 November 2020
Mila Causality Inference Lectures

Mila
Université de Montréal
CIFAR | ICRA
CANADIAN INSTITUTE FOR ADVANCED RESEARCH
INSTITUT CANADIEN DE RECHERCHES AVANCÉES

WHAT IS MISSING TOWARDS HUMAN-LEVEL AI?

- AI systems which actually understand the variables they manipulate (including language, perception and action)
- What does ’understanding’ mean?
    - They capture **causality**
    - They capture how the world works
    - They understand abstract actions and how use them to control
    - They can reason and plan, even in novel scenarios
    - They can explain what happened (inference, credit assignment)
    - They can generalize out-of-distribution

Missing from Current ML: Understanding & Generalization Beyond the Training Distribution

- Learning theory only deals with generalization within the same distribution
- Models learn but do not generalize well (or have high sample complexity when adapting) to modified distributions, non-stationarities, etc.

Missing from Current ML: Understanding & Generalization Beyond the Training Distribution

- If not iid, need alternative assumptions, otherwise no reason to expect generalization
- How do distributions change?
- What knowledge can be re-used?

COMPOSITIONALITY HELPS IID AND OOD GENERALIZATION

Different forms of compositionality
each with different exponential advantages

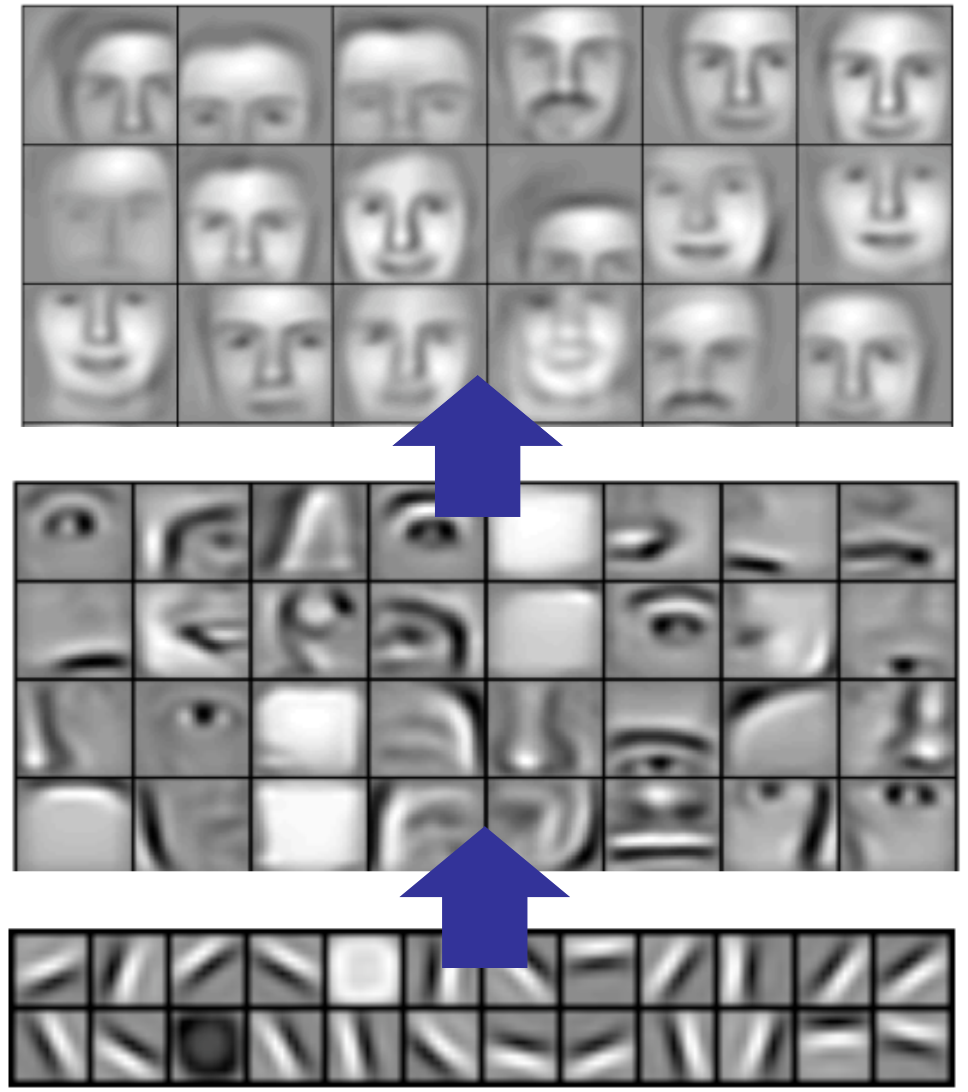
- Distributed representations

  (Pascanu et al ICLR 2014)

- Composition of layers in deep nets

  (Montufar et al NeurIPS 2014)

- **Systematic generalization in language, analogies, abstract reasoning? TBD**



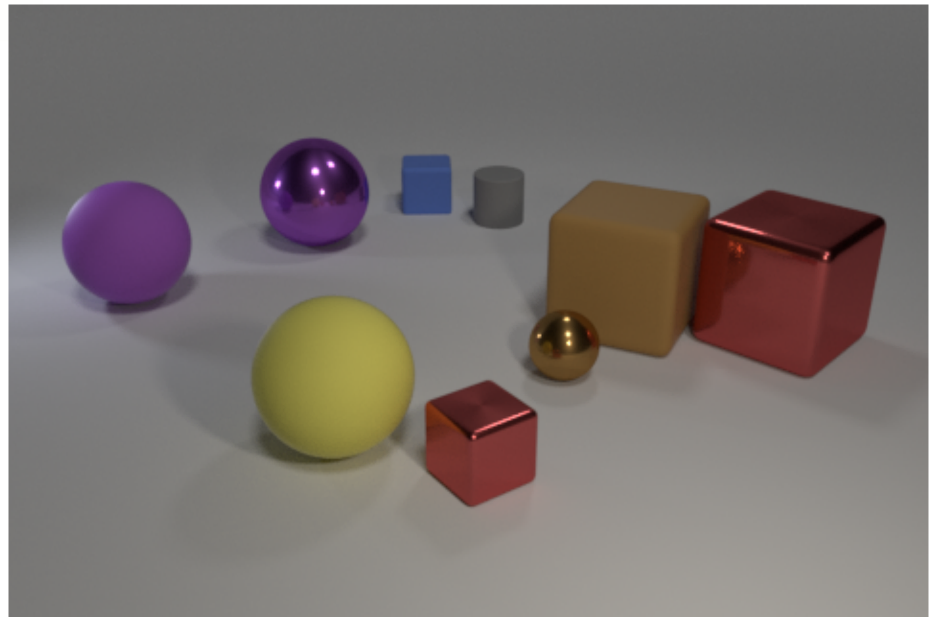
(Lee, Grosse, Ranganath & Ng, ICML 2009)

SYSTEMATIC GENERALIZATION

- Studied in linguistics
- **Dynamically recombine existing concepts**
- <u>Even when new combinations have 0 probability under training distribution</u>
  - E.g. Science fiction scenarios
  - E.g. Driving in an unknown city
- Not very successful with current DL, which can ”overfit” the training **distribution**

  (Lake & Baroni 2017)
  (Bahdanau et al & Courville ICLR 2019)
  CLOSURE: (Bahdanau et al & Courville arXiv:1912.05783) on CLEVR

(Lake et al 2015)



CONSCIOUS PROCESSING HELPS HUMANS DEAL WITH OOD SETTINGS

Faced with novel or rare situations, humans call upon conscious attention to combine on-the-fly the appropriate pieces of knowledge, to reason with them and imagine solutions.

→ we do not follow our habitual routines, we think hard to solve problems.

AGENT LEARNING NEEDS **OOD GENERALIZATION**

Agents face non-stationarities

Changes in distribution due to

- their actions
- ***ESPECIALLY:***
  - ***actions of other agents***
- different places, times, sensors, actuators, goals, policies, etc.

Multi-agent systems: many changes in distribution
Ood generalization needed for continual learning

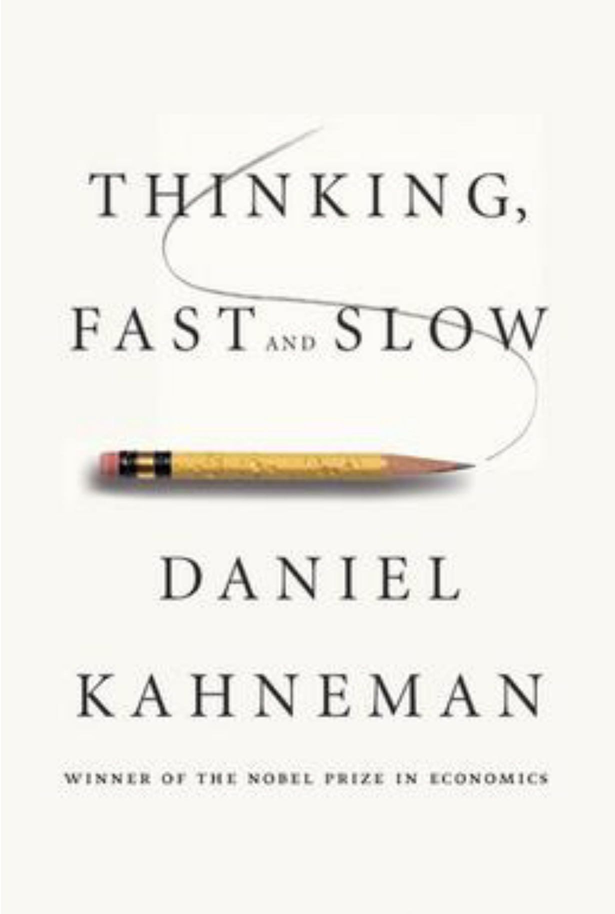
SYSTEM 1 VS. SYSTEM 2 COGNITION

2 systems (and categories of cognitive tasks):

Manipulates high-level / semantic concepts, which can be recombined combinatorially

System 1

- Intuitive, fast, **UNCONSCIOUS**, 1-step parallel, non-linguistic, habitual
- Implicit knowledge
- Current DL



System 2

- Slow, logical, **sequential**, **CONSCIOUS**, linguistic, algorithmic, planning, reasoning
- Explicit knowledge
- DL 2.0

IMPLICIT VS VERBALIZABLE KNOWLEDGE: UNDERLYING ASSUMPTIONS BEHIND VERBALIZABLE KNOWLEDGE

- Most knowledge in our brain is implicit and **not verbalizable** (hence the explainability challenge, even for humans)

- Some of our knowledge is verbalizable and we can reason and plan explicitly with it

- The concepts manipulated in this way are those we can name with language

- Properties of joint distribution between these concepts and their change over time?

➔ clarify these assumptions as priors to be able to embed them in ML architectures and training frameworks which bridge perception and reasoning

Independent Mechanisms

Scholkopf et al 2012

- Knowledge can be decomposed in informationally independent pieces (modules, mechanisms)
- Any causal intervention normally affects just one such mechanism
- Any other factorization would not have that property
- Mechanisms can be used in many instances (e.g. same law of gravity)

Mila

SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- Distributional changes due to localized causal interventions (in semantic space)
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared ’generic rules’ across instances (as arguments), requiring variables & indirection
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

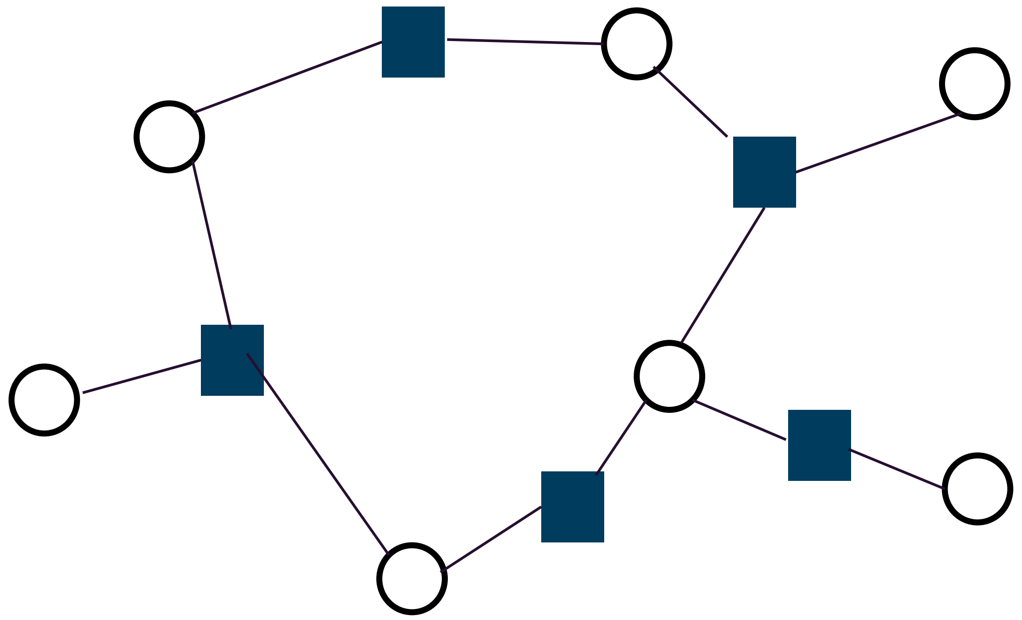
SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- **Sparse factor graph in space of high-level semantic variables**
- Semantic variables are causal: agents, intentions, controllable objects
- Distributional changes due to localized causal interventions (in semantic space)
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared 'rules' across instance tuples (as arguments), requiring variables & indirection
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

CONSCIOUSNESS **PRIOR**
➔ SPARSE FACTOR GRAPH

Bengio 2017, arXiv:1709.08568

- Property of **high-level variables which we manipulate with language**: *we can predict some given very few others*
  - E.g. "if I drop the ball, it will fall on the ground”
- **Disentangled factors** $\neq$marginally independent, e.g. ball & hand
- **Prior**: sparse factor graph joint distribution between high-level variables
- Inference involves few variables at a time, selected by **attention mechanism** and memory retrieval



Mila

SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- Sparse factor graph in space of high-level semantic variables
- **Semantic variables are causal: agents, intentions, controllable objects**
- Distributional changes due to localized causal interventions (in semantic space)
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared ’rules’ across instance tuples (as arguments), requiring variables & indirection
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

WHAT CAUSAL VARIABLES?

- Physics: position and momentum of every particle
    - Computationally intractable
- Scientists (and other humans) invent higher-level abstraction which make it easier to model causal structure of the world
- Can ML also do it?
    - Human brains are complex machines
    - Hence it is feasible

Mila

AGENCY TO GUIDE REPRESENTATION LEARNING & DISENTANGLING

(E. Bengio et al, 2017; V. Thomas et al, 2017; more recently see Kim et al ICML 2019)

Some factors (e.g. objects) correspond to 'independently controllable' aspects of the world

- Maximize mutual information between intentions (goal-conditioned policies) and changes in the state (trajectories), conditioned on the current state.

Can only be discovered by acting in the world

- *Control linked to notion of objects & agents*
- *Causal but agent-specific & subjective: affordances*

FROM PERCEPTION TO MODELLING THE WORLD AT THE SEMANTIC-LEVEL

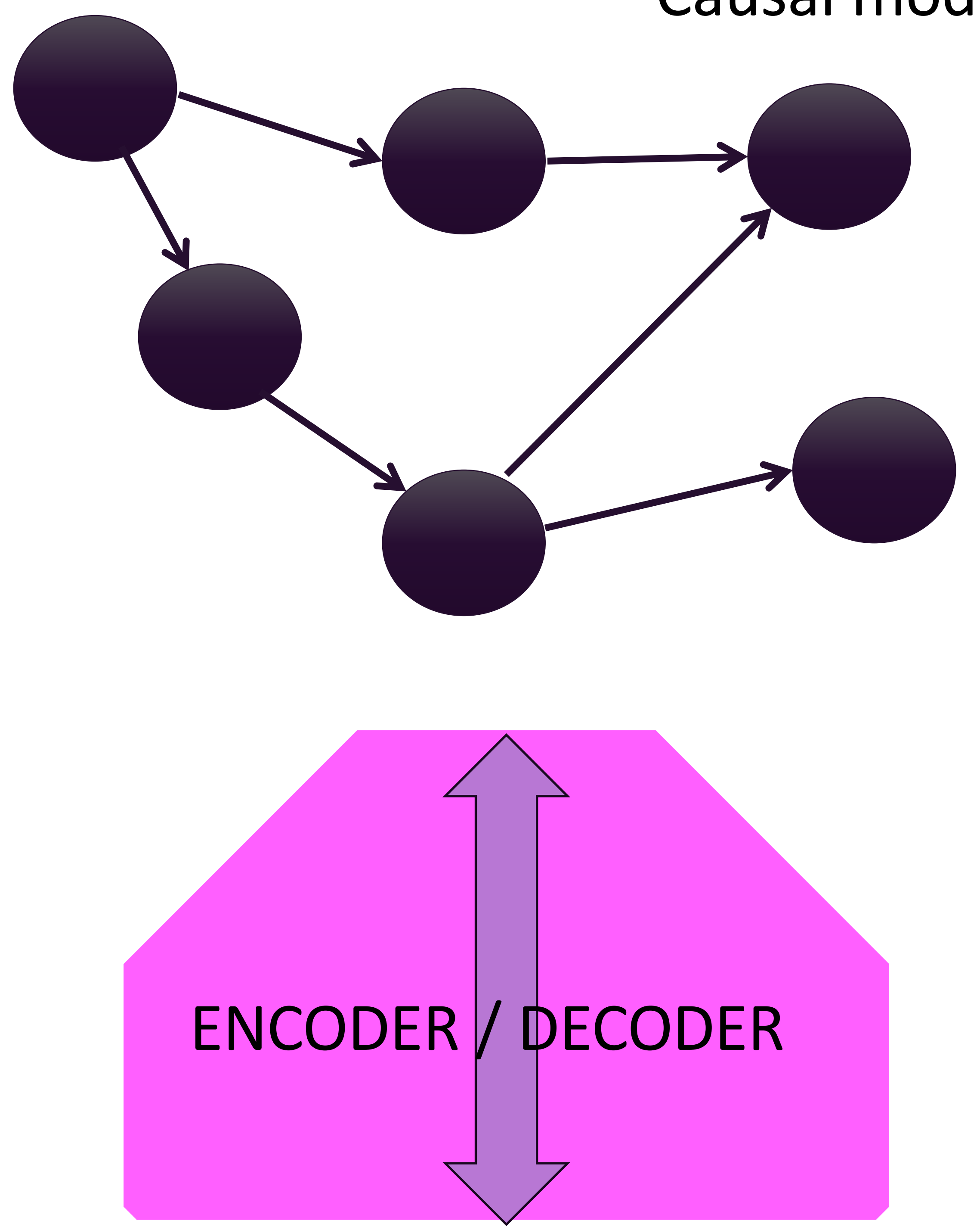
What are the right representations? Causal variables explaining the data

How to discover them (as a function of observed data)?

How to discover their causal relationship, the causal graph?

How are actions corresponding to causal interventions?

How is raw sensory data mapped to high-level causal variables and how do high-level causal variables turn into low-level actions and partial observations?

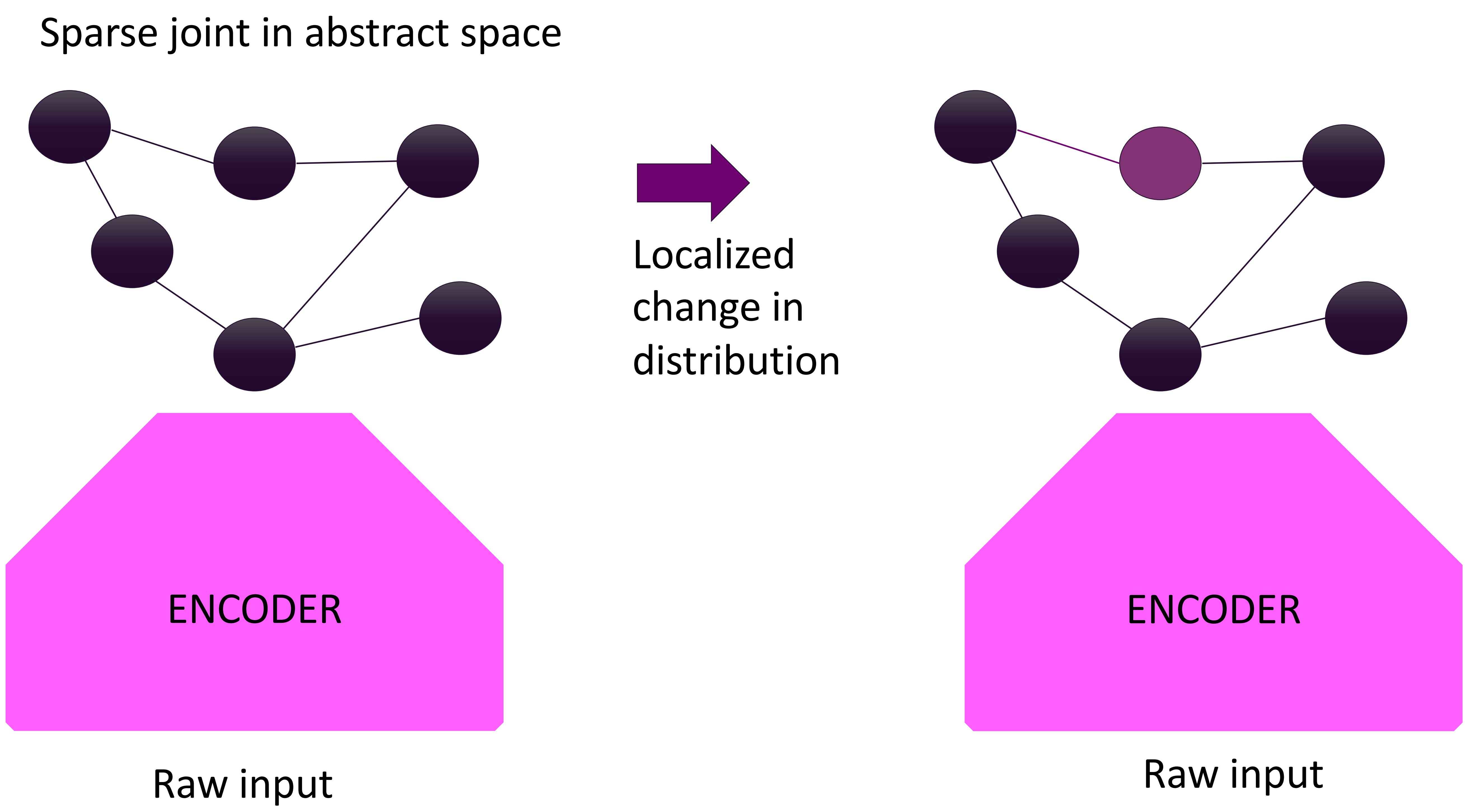


SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- **Distributional changes due to localized causal interventions (in semantic space)**
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared 'rules' across instance tuples (as arguments), requiring variables & indirection
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

INDEPENDENT MECHANISMS: SPARSE CHANGE IN ABSTRACT LATENT SPACE

Change may be drastic in pixel space but tiny in semantic space of causal variables



WHAT **CAUSES** CHANGES IN DISTRIBUTION?

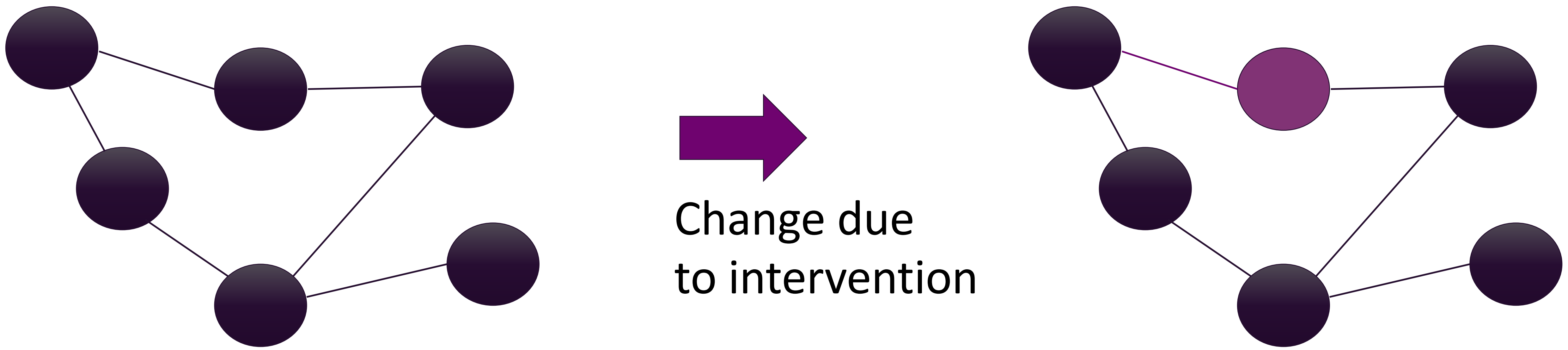
Underlying physics: actions are localized in space and time.

Hypothesis to replace iid assumption:

changes = consequence of an intervention on few causes or mechanisms

Extends the hypothesis of (informationally) Independent Mechanisms *(Scholkopf et al 2012)*

➔ **local inference or adaptation in the right model**



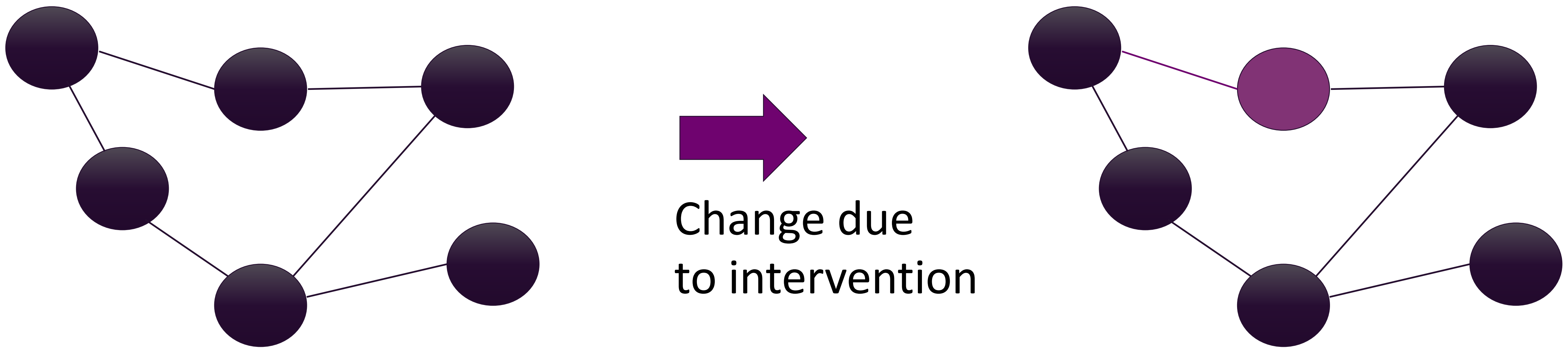
COUNTING ARGUMENT:
LOCALIZED CHANGE→OOD TRANSFER

Good representation of variables and mechanisms + localized change hypothesis

→ few bits need to be accounted for (by inference or adaptation)

→ few observations (of modified distribution) are required

→ good ood generalization/fast transfer/small ood sample complexity



CAUSAL INDUCTION FROM INTERVENTION DATA

Recovery of causal model from data

Observational data:

- Distinguishes causal models *only up to Markov equivalence class*

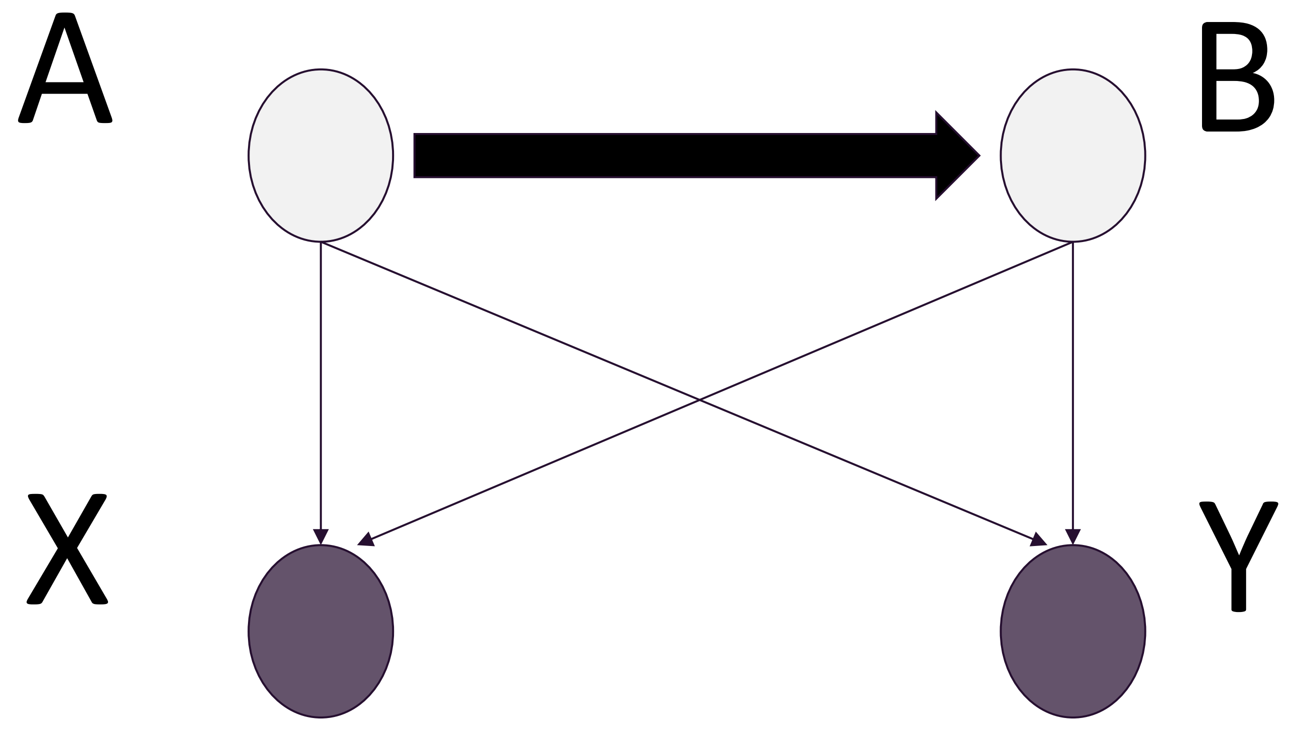
Intervention data:

- What causal induction requires
- Most work assumes *known*-intervention data
- Real world: Other agents or environment can intervene
  - Hence, interventions unknown
- How to handle unknown intervention?
  - *Infer* it

EXAMPLE: DISCOVERING CAUSE AND EFFECT
= **HOW TO FACTORIZE A JOINT DISTRIBUTION?**

A Meta-Transfer Objective for Learning to Disentangle Causal Mechanisms

- Learning whether A causes B or vice-versa
- Learning to disentangle (A,B) from observed (X,Y)
- Exploit changes in distribution and speed of adaptation to guess causal direction



ICLR 2020: A Meta-Transfer Objective for Learning to Disentangle Causal Mechanisms, Bengio, Deleu, Rahaman, Ke, Lachapelle, Bilaniuk, Goyal, Pal ArXiv:1901.10912

Experimental setup

- Consider two r.v. A and B, where **A causes B**.

- The correct causal model decomposes as

$$p(A, B) = p(A)p(B \mid A)$$

- Consider two distributions, where **only p(A) changes** and p(B | A) remains unchanged (covariate shift).
    - A **training** distribution p
    - A **transfer** distribution $\tilde{p}$

- If we train a model using data from p using the correct decomposition, then **adaptation on $\tilde{p}$ is fast** because

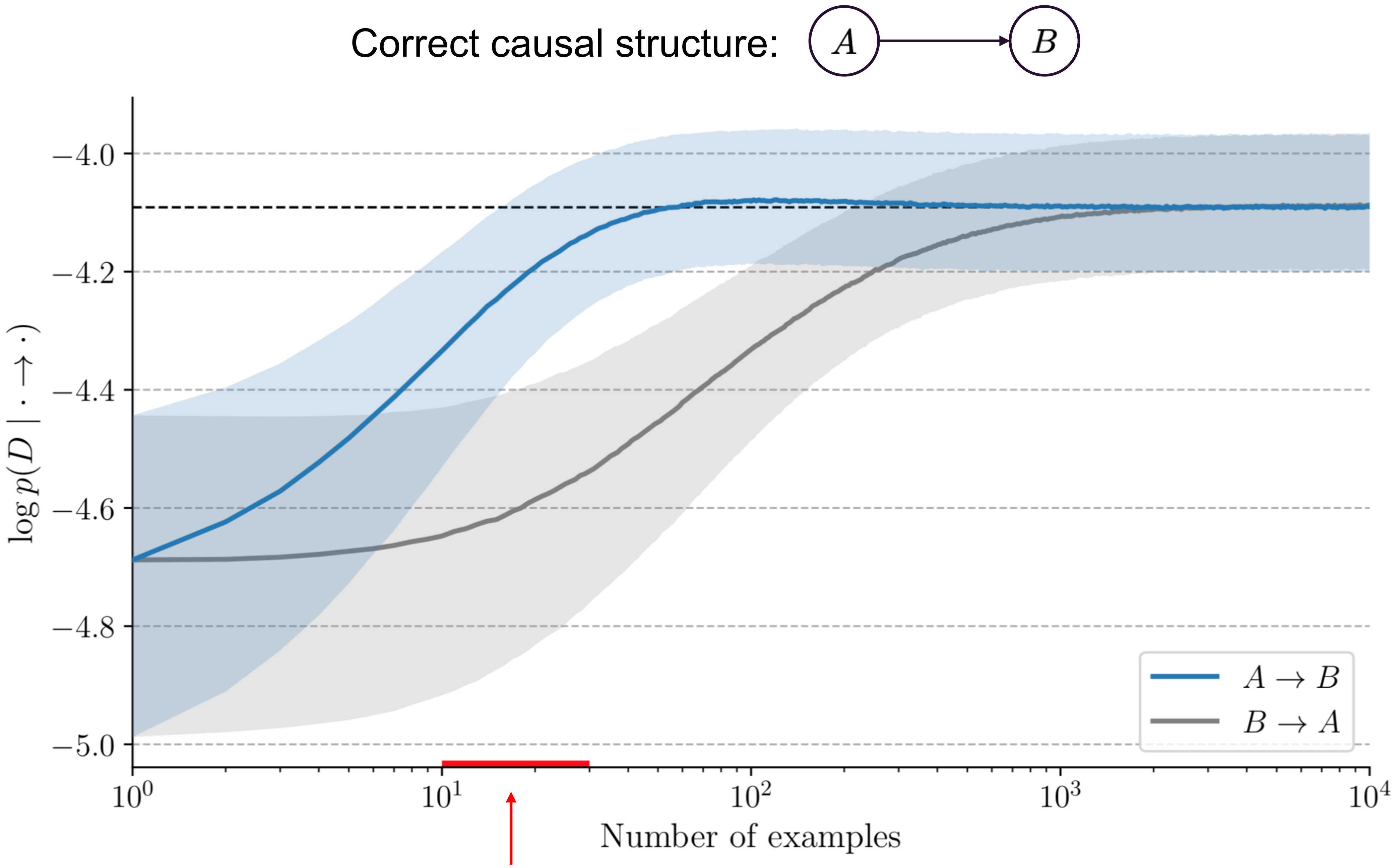
$$\mathbb{E}_{\tilde{p}(B|A)}\left[\frac{\partial \log p_\theta(B \mid A)}{\partial \theta}\right] = 0$$

$$\text{when} \quad p_\theta(B \mid A) = \tilde{p}(B \mid A)$$

$$A \longrightarrow B$$

Wrong knowledge factorization leads to poor transfer

- With the wrong factorization $p(B)p(A \mid B)$ a change in p(A) influences all the modules.
    - Poor transfer: all the parameters need to be adapted.
- This is the normal situation with standard neural networks: every parameter participates to every relationship between all the variables.
- This causes **catastrophic forgetting**, **poor transfer**, **difficulties with continual learning or domain adaptation**, etc.
- Use the speed of adaptation as a way to find the correct factorization.



- Faster online adaptation to modified distribution = lower NLL regret
- Effect of the correct factorization is most evident with only a few samples from modified distribution

Mila

The Meta-Transfer Objective

- Quantify the speed of adaptation with the **online likelihood**

$$\mathcal{L}_G(\mathcal{D}_{int}) = \prod_{t=1}^{T} p(\mathbf{x}_t\,;\,\theta_G^{(t)}, G)$$

$$\begin{aligned}\theta_G^{(1)} &= \hat{\theta}_G^{ML}(\mathcal{D}_{obs})\\ \theta_G^{(t+1)} &= \theta_G^{(t)} + \alpha \nabla_\theta \log p(\mathbf{x}_t\,;\,\theta_G^{(t)}, G)\end{aligned}$$

  - Adaptation with gradient ascent
  - D_{obs} is a large **training dataset** sampled from p
  - D_{int} is a small **transfer dataset** sampled from $\tilde{p}$

- **Smooth parametrization** of the causal structure

$$\mathcal{R}(\mathcal{D}_{int}) = -\log\left[\sigma(\gamma)\mathcal{L}_{A\to B}(\mathcal{D}_{int}) + (1-\sigma(\gamma))\mathcal{L}_{B\to A}(\mathcal{D}_{int})\right]$$

  - Structural (meta-)parameter γ
  - If $\sigma(\gamma) = 1$, then the correct structure is recovered.

The Meta-Transfer Objective gradient

Proposition 2. *The gradient of the negative log-likelihood of the transfer data $\mathcal{D}_{int}$ in Equation (5) wrt. the structural parameter γ is given by*

$$\frac{\partial \mathcal{R}}{\partial \gamma} = p(A \to B) - p(A \to B \mid \mathcal{D}_{int}), \tag{6}$$

where $p(A \to B \mid \mathcal{D}_{int})$ is the posterior probability of the hypothesis $A \to B$ (when the alternative is $B \to A$). Furthermore, this can be equivalently written as

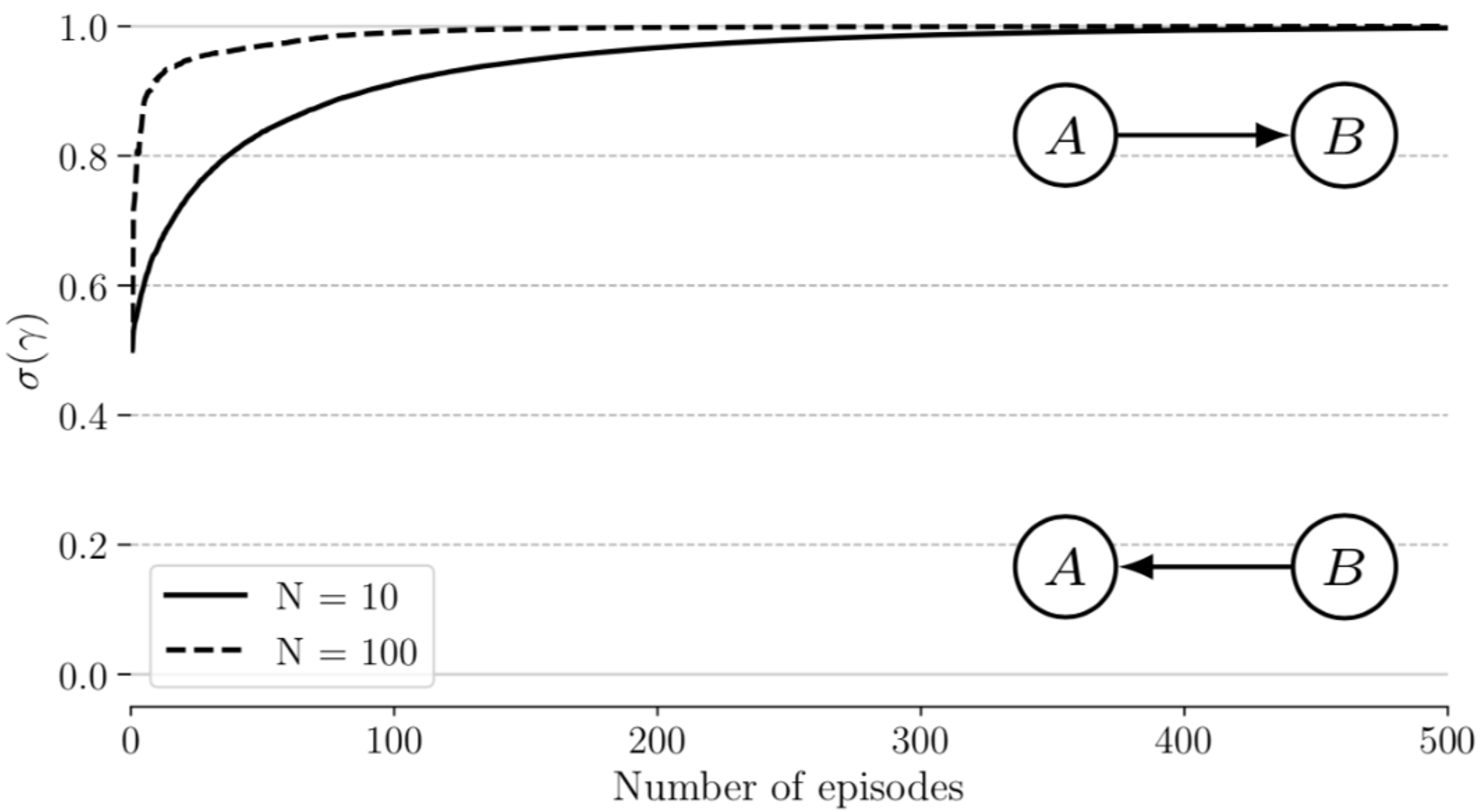
$$\frac{\partial \mathcal{R}}{\partial \gamma} = \sigma(\gamma) - \sigma(\gamma + \Delta), \tag{7}$$

where $\Delta = \log \mathcal{L}_{A \to B}(\mathcal{D}_{int}) - \log \mathcal{L}_{B \to A}(\mathcal{D}_{int})$ is the difference between the online log-likelihoods of the two hypotheses on the transfer data $\mathcal{D}_{int}$.

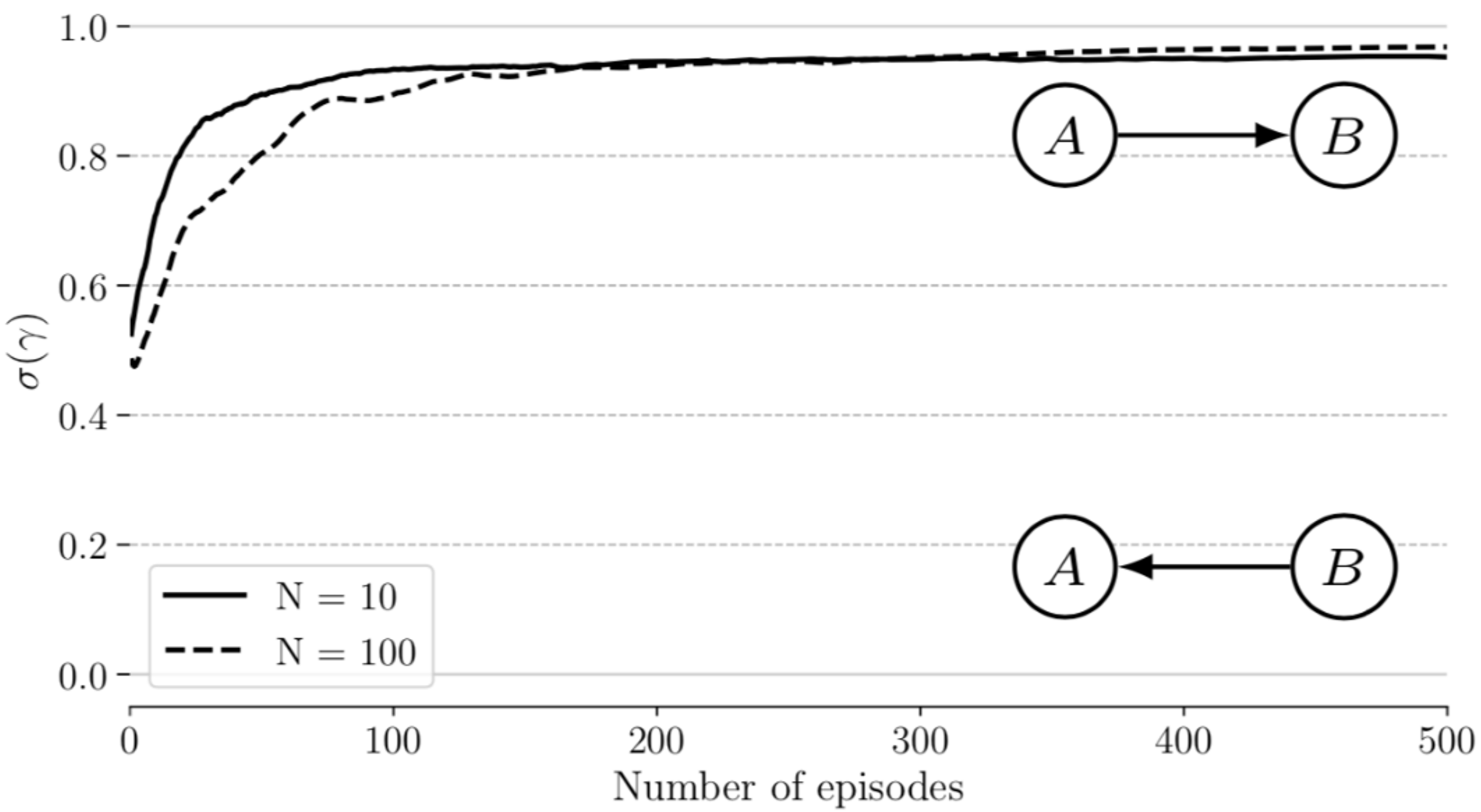
$$\mathcal{R}(\mathcal{D}_{int}) = -\log\left[\sigma(\gamma)\mathcal{L}_{A \to B}(\mathcal{D}_{int}) + (1 - \sigma(\gamma))\mathcal{L}_{B \to A}(\mathcal{D}_{int})\right]$$

Can be optimized wrt. γ with gradient descent

Experimental results - Discrete variables



Tabular representation of marginals and conditionals of bivariate model

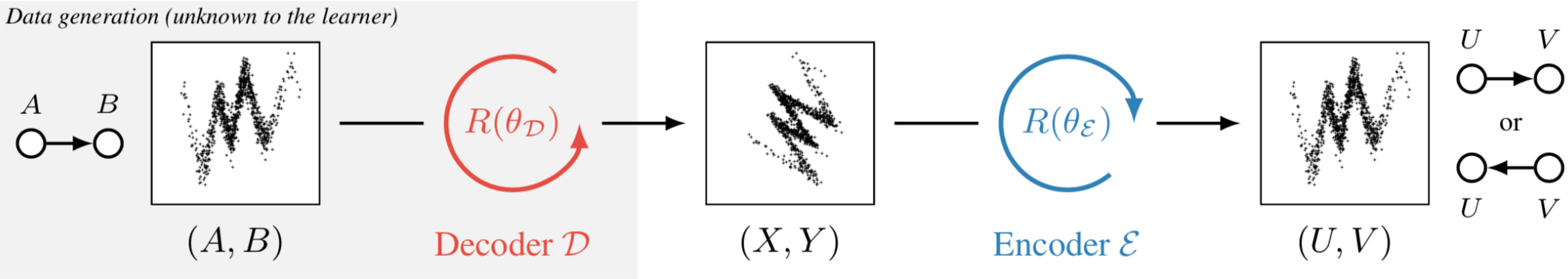


Each conditional is represented by a one-hidden-layer MLP with one-hot inputs, softmax outputs

+ Experiments on Linear Gaussian & Continuous multimodal variables (see Appendix).

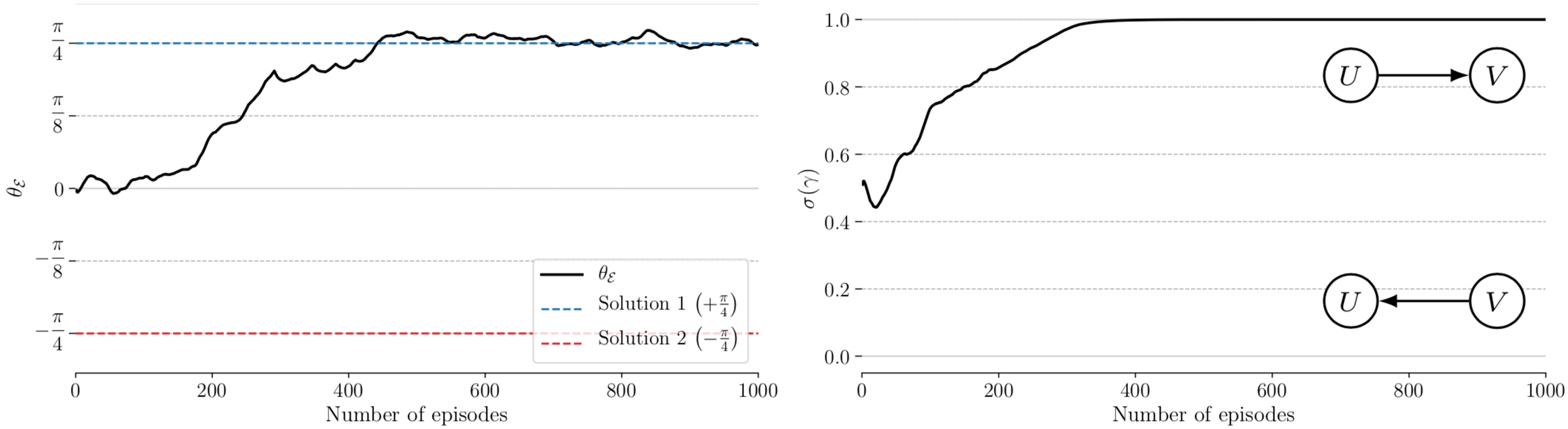
Disentangling the causes

- Realistic settings: causal variables are not directly observed.
- Need to learn an encoder which maps raw data to causal space.
- Consider both the encoder parameters and the causal graph structural parameters as meta-parameters trained together wrt proposed meta-transfer objective.



- Simplest possible scenario: linear mixing (rotating decoder) and unmixing (rotating decoder)

Experimental results - Disentangling the causes



- Recovers the correct encoder parameter (left), up to permutation.
- Simultaneously recovers causal direction (right).

Mila

DISCOVERING LARGER CAUSAL GRAPHS

Learning Neural Causal Models from Unknown Interventions

Ke, Bilaniuk, Goyal, Bauer, Scholkopf, Larochelle, Pal & Bengio 2019 arXiv:1910.01075

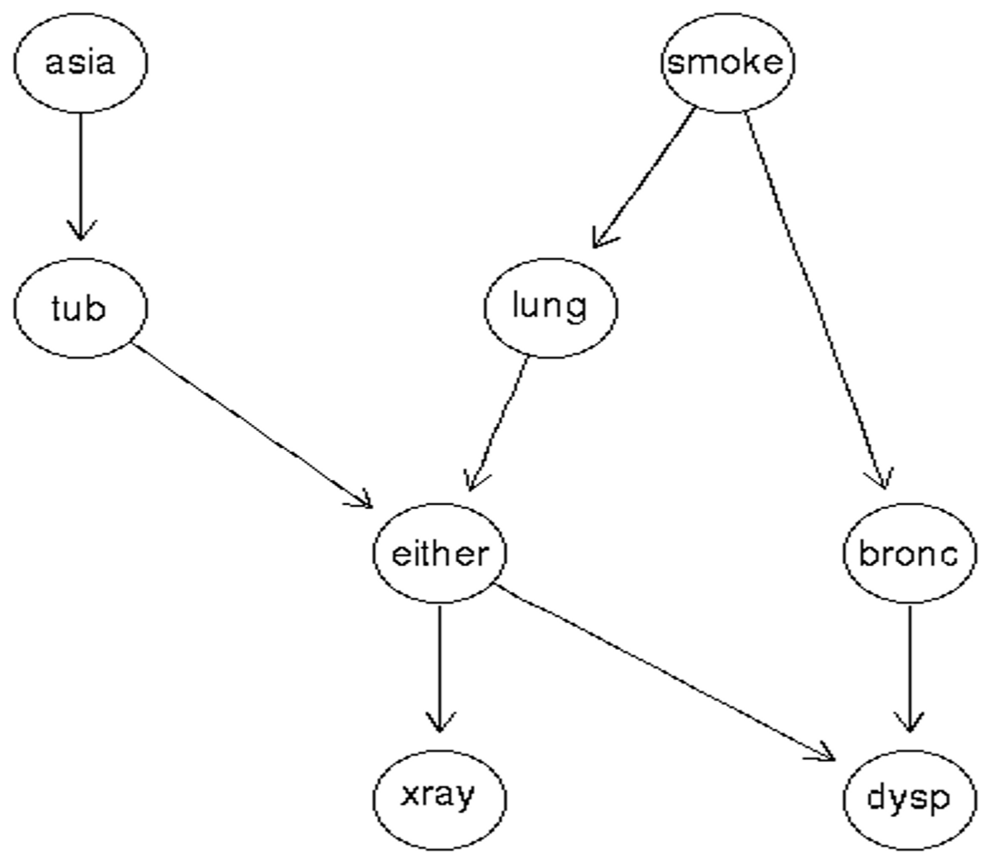
- Learning small causal graphs, avoid exponential explosion of # of graphs by parametrizing factorized distribution over graphs

- With enough observations of changes in distribution: perfect recovery of the causal graph without knowing the intervention; converges faster on sparser graphs

- Inference over the intervention:
  faster causal discovery

Asia graph, CE on ground truth edges, comparison against other causal induction methods

Our method	(Eaton & Murphy, 2007a)	(Peters et al., 2016)	(Zheng et al., 2018)
0.0	0.0	10.7	3.1



HOW TO FACTORIZE AND LEARN THE BELIEF DISTRIBUTION OVER GRAPHS

Learning Neural Causal Models from Unknown Interventions *Ke et al 2019 arXiv:1910.01075*

Dependency Structure Discovery from Interventions *Ke et al 2020, submitted*

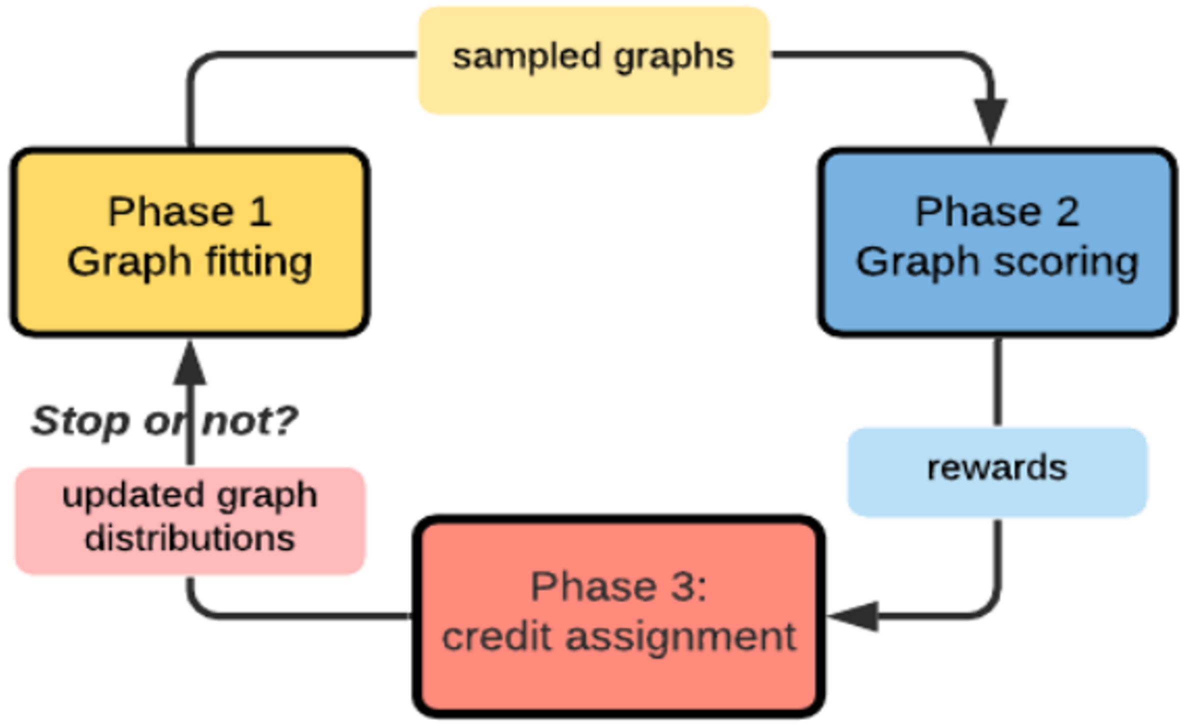


Figure 2: Workflow for our proposed method SDI. Phase 1 samples graphs under the model's current belief about the edge structure and fits parameters to observational data. Phase 2 scores a small set of graphs against interventional data and assigns rewards according to graphs' ability to predict interventions. Phase 3 uses the rewards from Phase 2 to update the beliefs about the edge structure. If the believed edge probabilities have all saturated near 0 or 1, the method has converged.

Use neural networks to present/ learn causal models

Parameters:

- Structural parameters
- Functional parameters

Method overview:

Iterate:

1. Phase 1: Graph fitting on *observational* data
2. Phase 2: Graph scoring on *interventional* data
3. Phase 3: Credit assignment to structural parameters

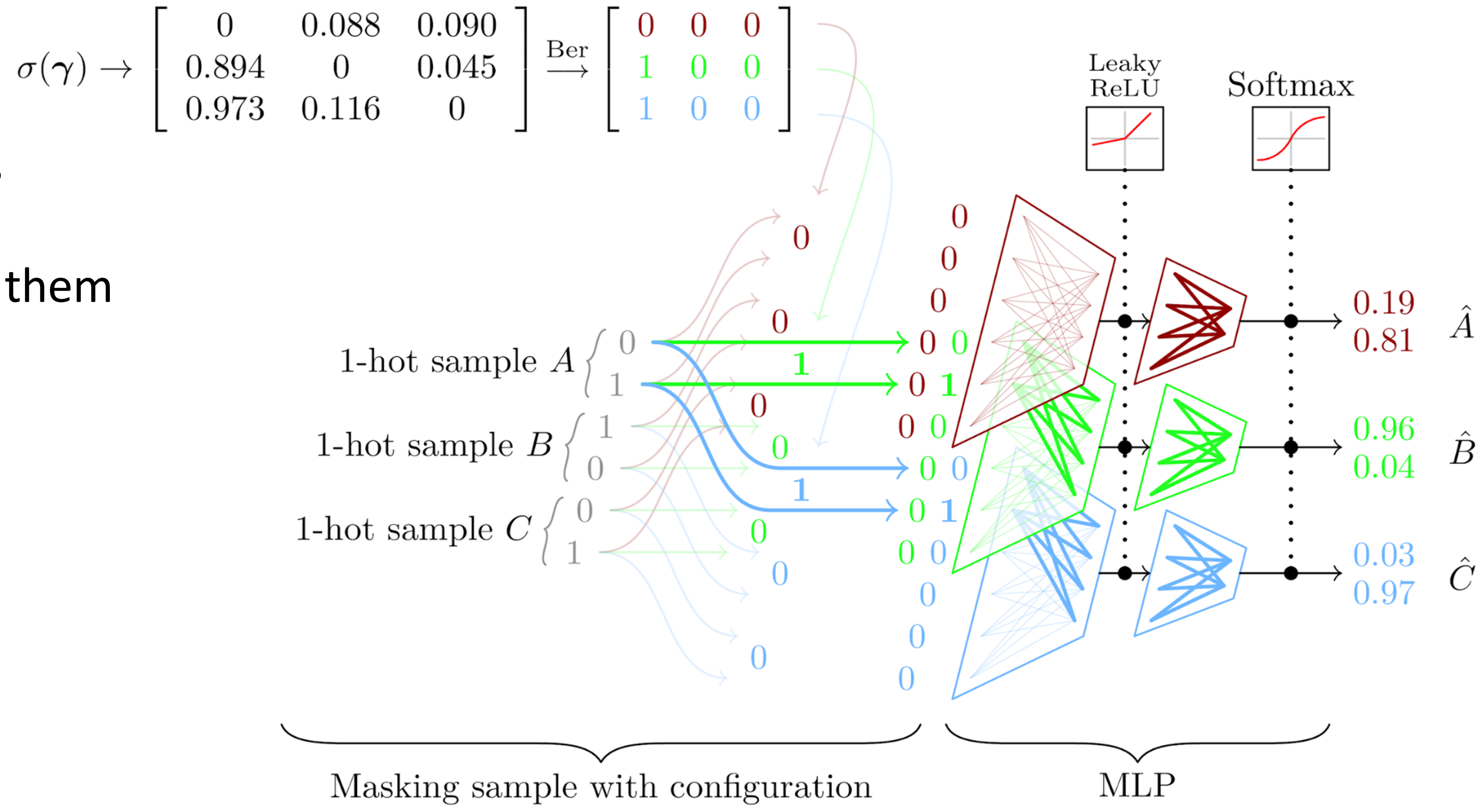
$$g_{ij} = \frac{\sum_k (\sigma(\gamma_{ij}) - c_{ij}^{(k)}) \mathcal{L}_{C_{,i}^{(k)}}(X)}{\sum_k \mathcal{L}_{C_{,i}^{(k)}}(X)}, \quad \forall i,j \in \{0,\ldots,M-1\}$$

MODEL ARCHITECTURE

Use N neural networks to represent causal graph with N variables

Each neural network models:

- Who are the direct causal parents
  - *Structural parameters*
- What is the relationship between them
  - *Functional parameters*



EXPERIMENTAL RESULTS

Table 1: **Baseline comparisons:** Structural Hamming Distance (SHD) (lower is better) for learned and ground-truth edges on various graphs from both synthetic and real datasets, compared to [33], [48], [14], [11] and [10]. The proposed method (Structural Discovery from Interventions (SDI)) is run on random seeds $1-5$ and we pick the worst performing model out of the random seeds in the table. OOM: out of memory. Our proposed method correctly recovers the true causal graph, with the exception of Sachs and full13, and it significantly outperforms all other baseline methods.

Method M	Asia 8	Sachs 11	collider 8	chain 13	jungle 13	collider 13	full 13
Zheng et al. [10]	14	22	18	39	22	24	71
Yu et al. [11]	10	19	7	14	16	12	77
Heinze-Deml et al. [48]	8	17	7	12	12	7	28
Peters et al. [33]	5	17	2	2	8	2	16
Eaton and Murphy [49]	0	OOM	7	OOM	OOM	OOM	OOM
Proposed Method (SDI)	0	6	0	0	0	0	7

[10] Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, pages 9472–9483, 2018.

[11] Yue Yu, Jie Chen, Tian Gao, and Mo Yu. Dag-gnn: Dag structure learning with graph neural networks. *arXiv preprint arXiv:1904.10098*, 2019.

[48] Christina Heinze-Deml, Jonas Peters, and Nicolai Meinshausen. Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2), 2018.

[33] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012, 2016.

[49] Daniel Eaton and Kevin Murphy. Belief net structure learning from uncertain interventions. *J Mach Learn Res*, 1:1–48, 2007.

EXPERIMENTAL RESULTS

Table 1: **Baseline comparisons:** Structural Hamming Distance (SHD) (lower is better) for learned and ground-truth edges on various graphs from both synthetic and real datasets, compared to [33], [48], [14], [11] and [10]. The proposed method (Structural Discovery from Interventions (SDI)) is run on random seeds $1-5$ and we pick the worst performing model out of the random seeds in the table. OOM: out of memory. Our proposed method correctly recovers the true causal graph, with the exception of Sachs and full13, and it significantly outperforms all other baseline methods.

Method	`Asia`	`Sachs`	`collider`	`chain`	`jungle`	`collider`	`full`
M	8	11	8	13	13	13	13
Zheng et al. [10]	14	22	18	39	22	24	71
Yu et al. [11]	10	19	7	14	16	12	77
Heinze-Deml et al. [48]	8	17	7	12	12	7	28
Peters et al. [33]	5	17	2	2	8	2	16
Eaton and Murphy [49]	0	OOM	7	OOM	OOM	OOM	OOM
Proposed Method (SDI)	0	6	0	0	0	0	7

[10] Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, pages 9472–9483, 2018.

[11] Yue Yu, Jie Chen, Tian Gao, and Mo Yu. Dag-gnn: Dag structure learning with graph neural networks. *arXiv preprint arXiv:1904.10098*, 2019.

[48] Christina Heinze-Deml, Jonas Peters, and Nicolai Meinshausen. Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2), 2018.

[33] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012, 2016.

[49] Daniel Eaton and Kevin Murphy. Belief net structure learning from uncertain interventions. *J Mach Learn Res*, 1:1–48, 2007.

CONVERGENCE RATE FOR DIFFERENT GRAPHS

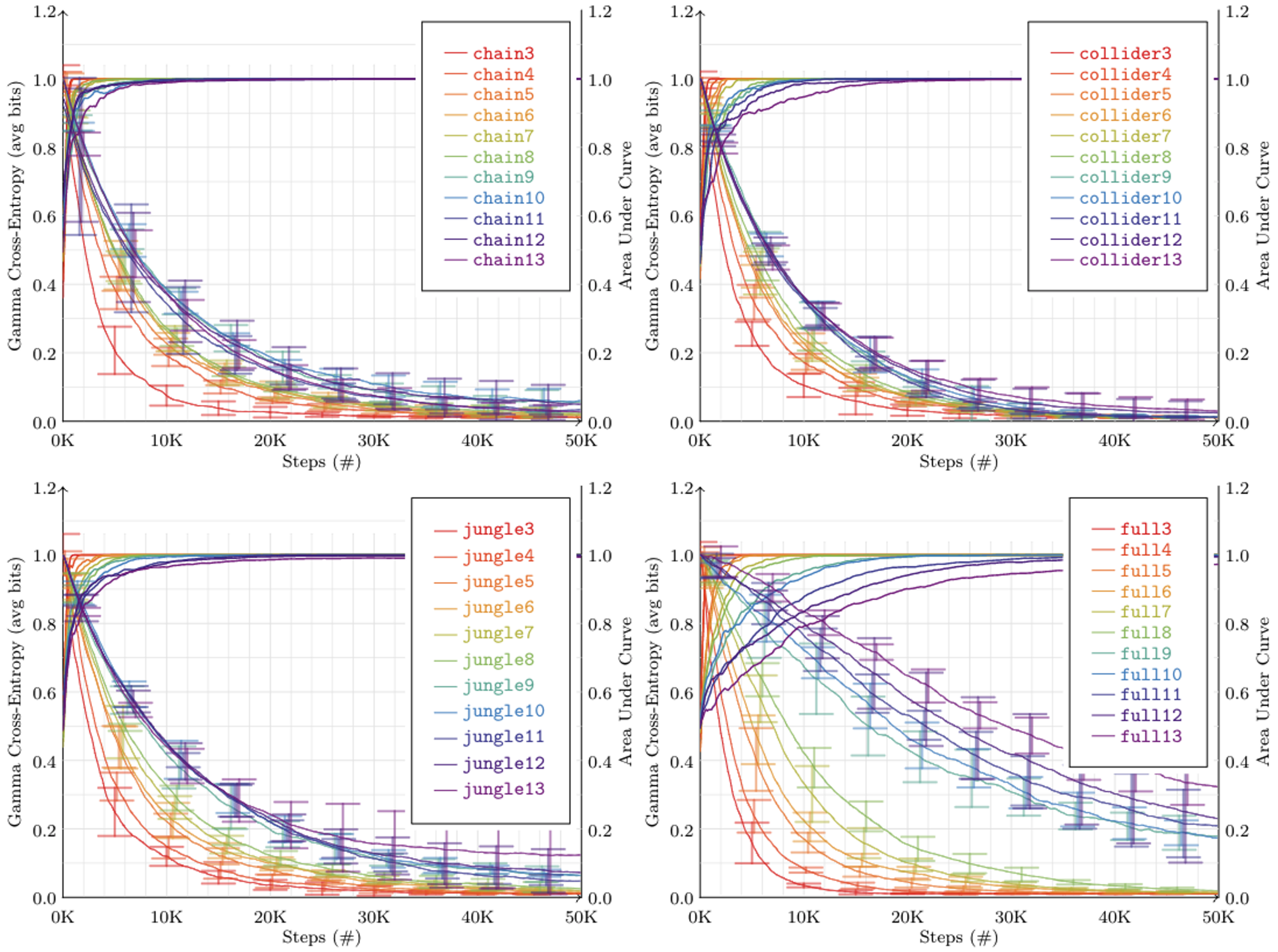


Figure 10: Cross entropy (CE) and Area-Under-Curve (AUC/AUROC) for edge probabilities of learned graph against ground-truth for synthetic SCMs. Error bars represent $\pm 1\sigma$ over PRNG seeds 1-5. **Left to right, up to down**: chainM,jungleM,fullM, $M = 3 \ldots 8$ ($9 \ldots 13$ in Appendix 7.6.1). Graphs (3-13 variables) all learn perfectly with AUROC reaching 1.0. However, denser graphs (fullM) take longer to converge.

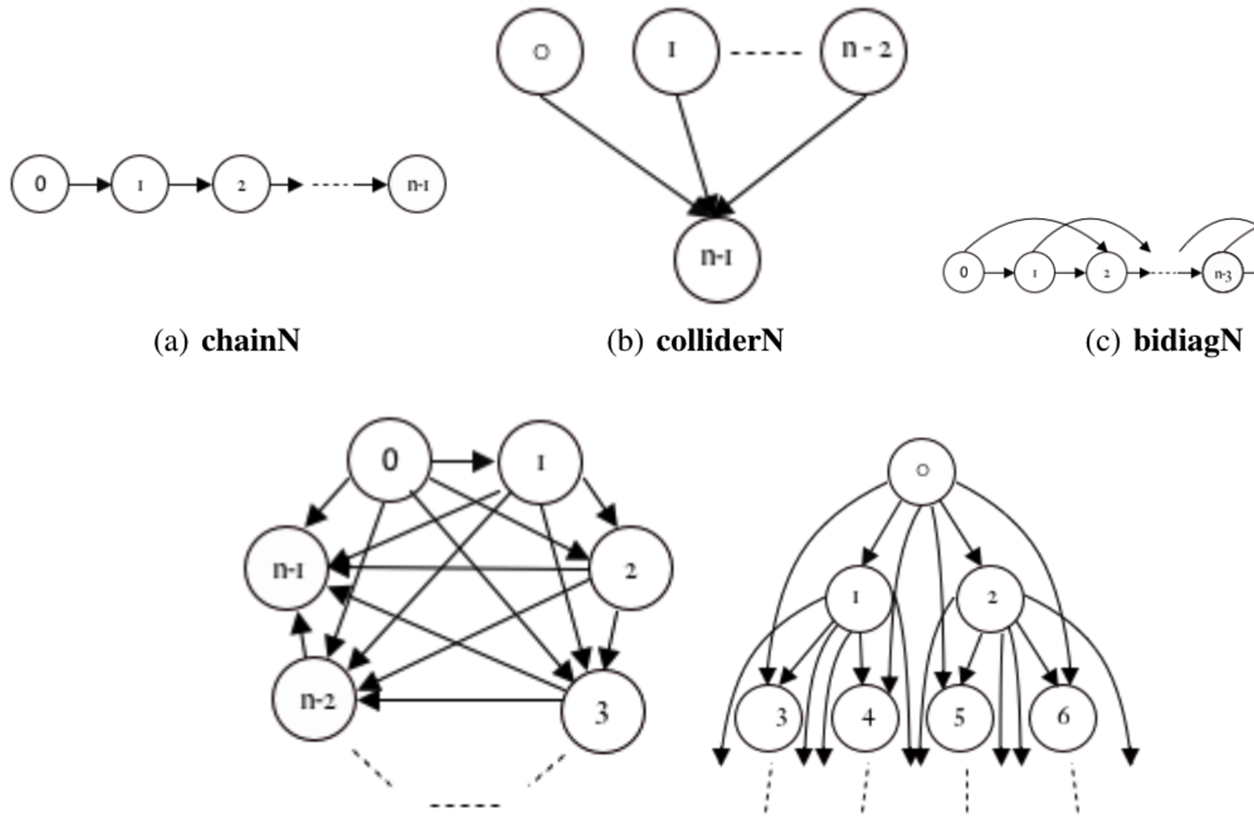


Figure 6: Figures for various synthetic graphs. chain, collider, bidiagonal, full and jungle graph.

DENSER GRAPHS ARE MORE CHALLENGING

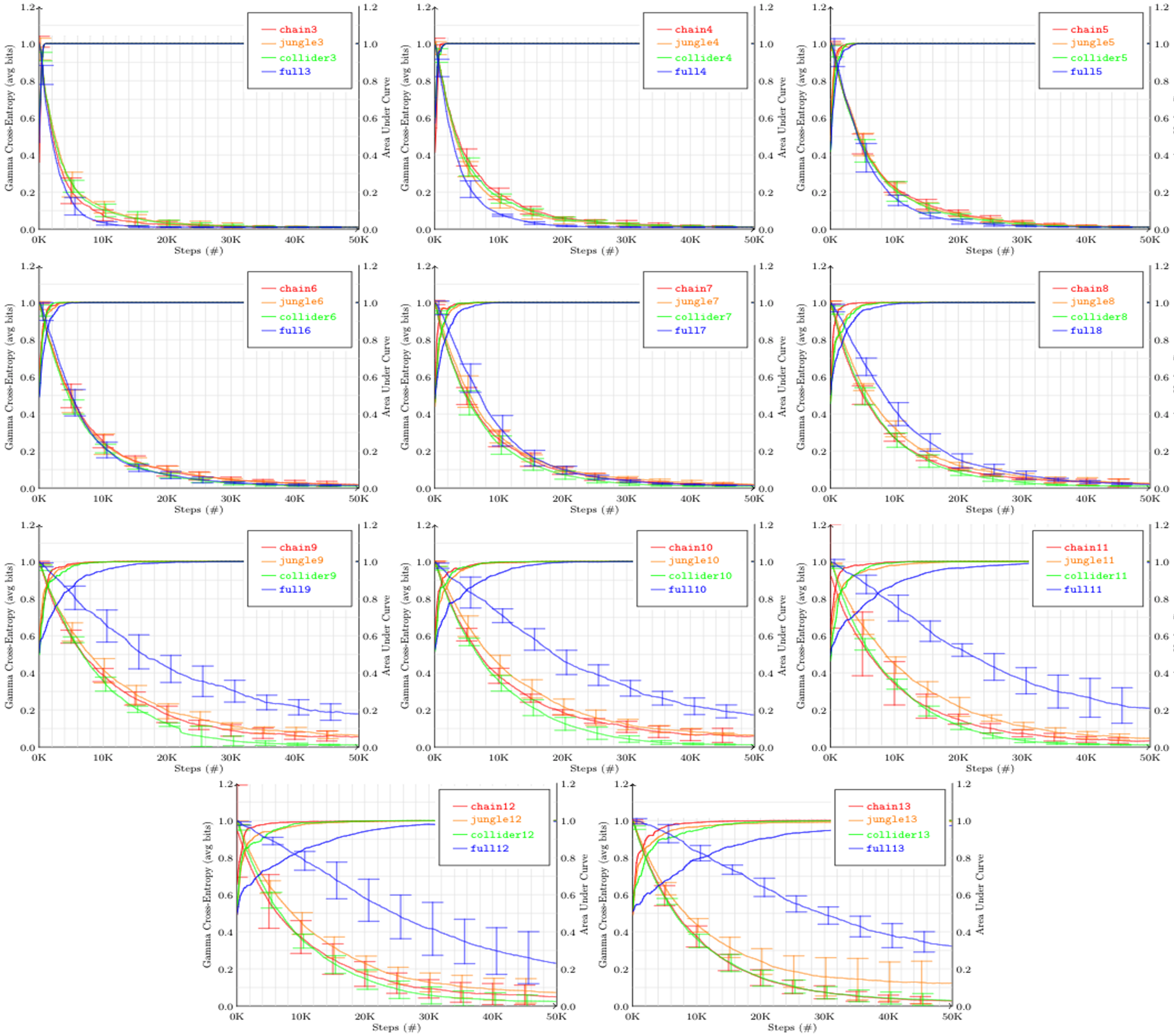


Figure 12: **Left to right, top to bottom** Average cross-entropy loss of edge beliefs $\sigma(\gamma)$ and Area-Under-Curve throughout training for the synthetic graphs `chainN`, `jungleN`, `colliderN` and `fullN`, N=3-13, grouped by graph size. Error bars represent $\pm 1\sigma$ over PRNG seeds 1-5.

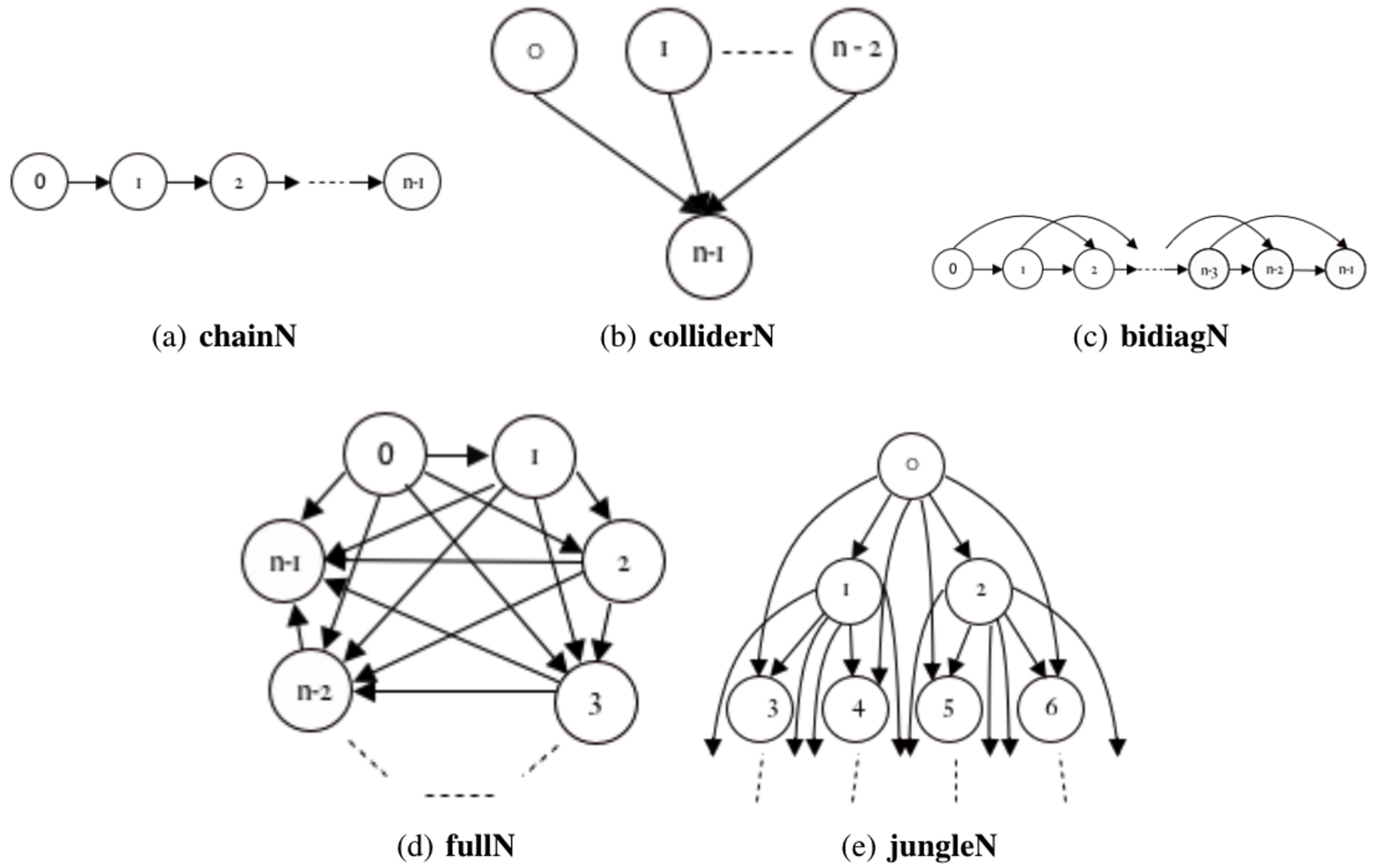
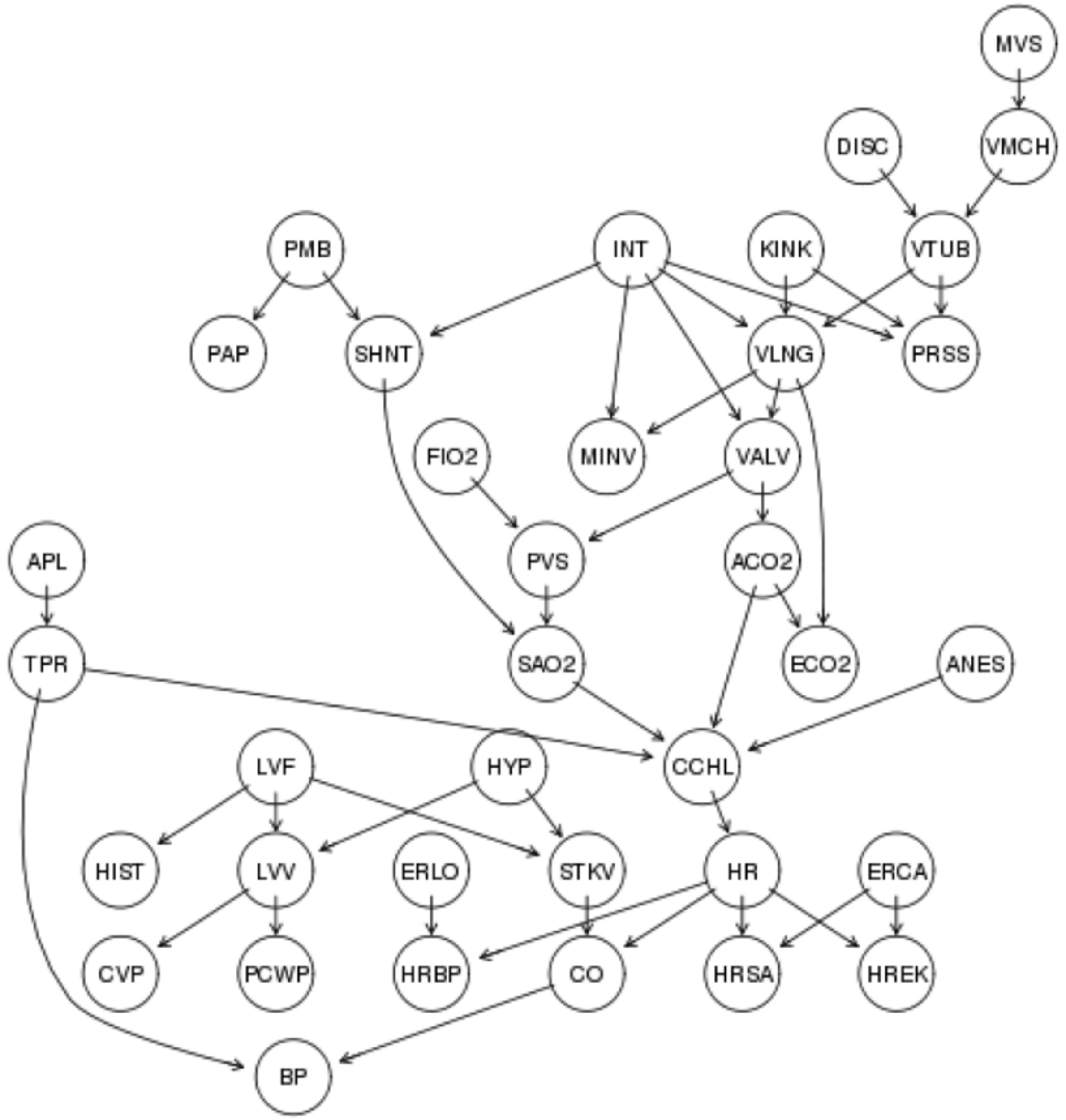


Figure 6: Figures for various synthetic graphs. `chain`, `collider`, `bidiagonal`, `full` and `jungle` graph.

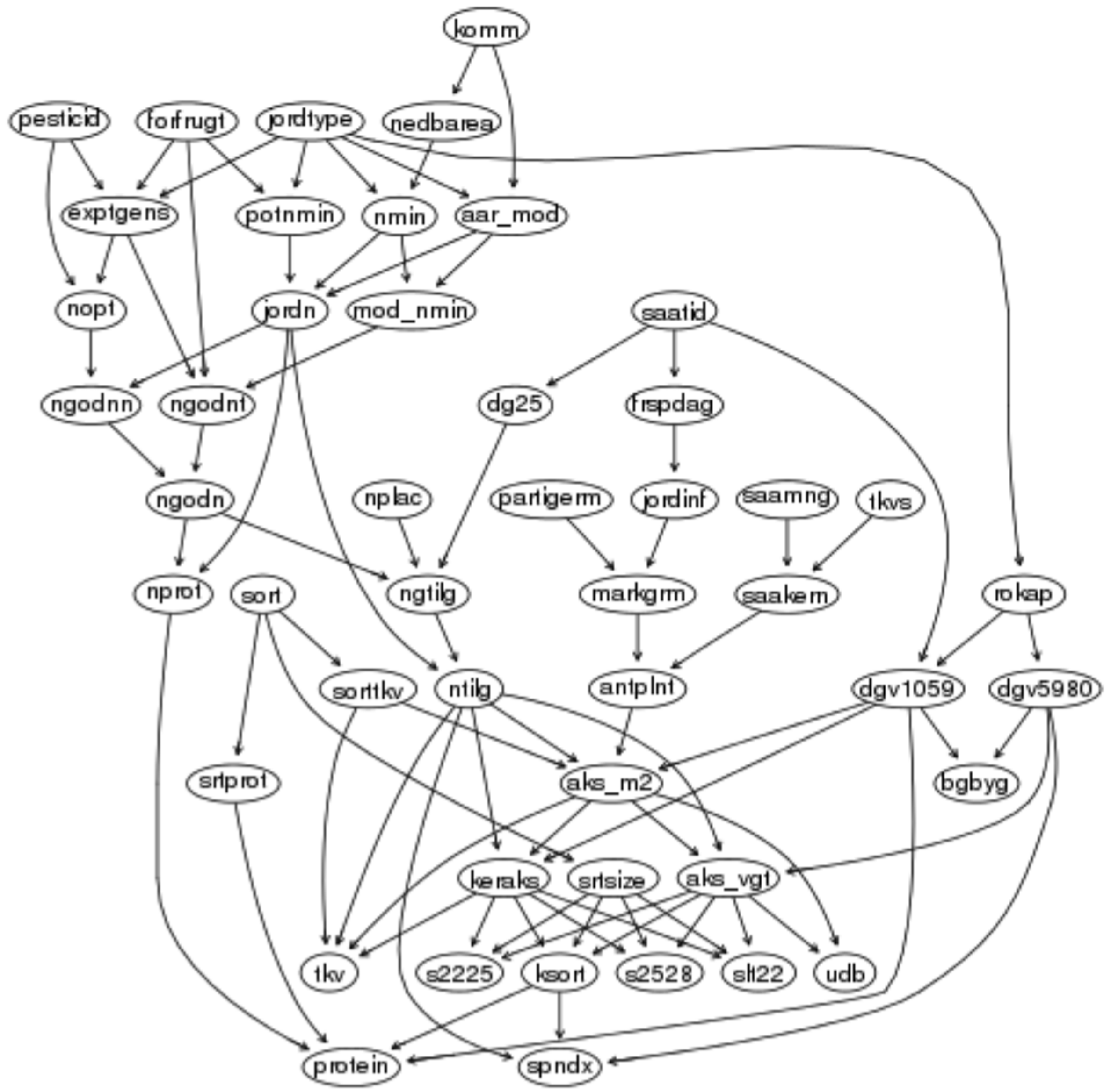
PARTIAL GRAPH RECOVERY

Table 4: **Partial Graph Recovery** on Alarm [51] and Barley [52]. The model is asked to predict 50 edges in Barley and 40 edges in Alarm. The accuracy is measured in Structural Hamming Distance (SHD). SDI achieved over 90% accuracy on both graphs.

Graph	**Alarm**	**Barley**
Number of variables	37	48
Total Edges	46	84
Edges to recover	40	50
Recovered Edges	37	45
Errors (in SHD)	3	5



Alarm [1].



Barclay [2].

[1]. I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper. The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks. In Proceedings of the 2nd European Conference on Artificial Intelligence in Medicine, pages 247-256. Springer-Verlag, 1989.

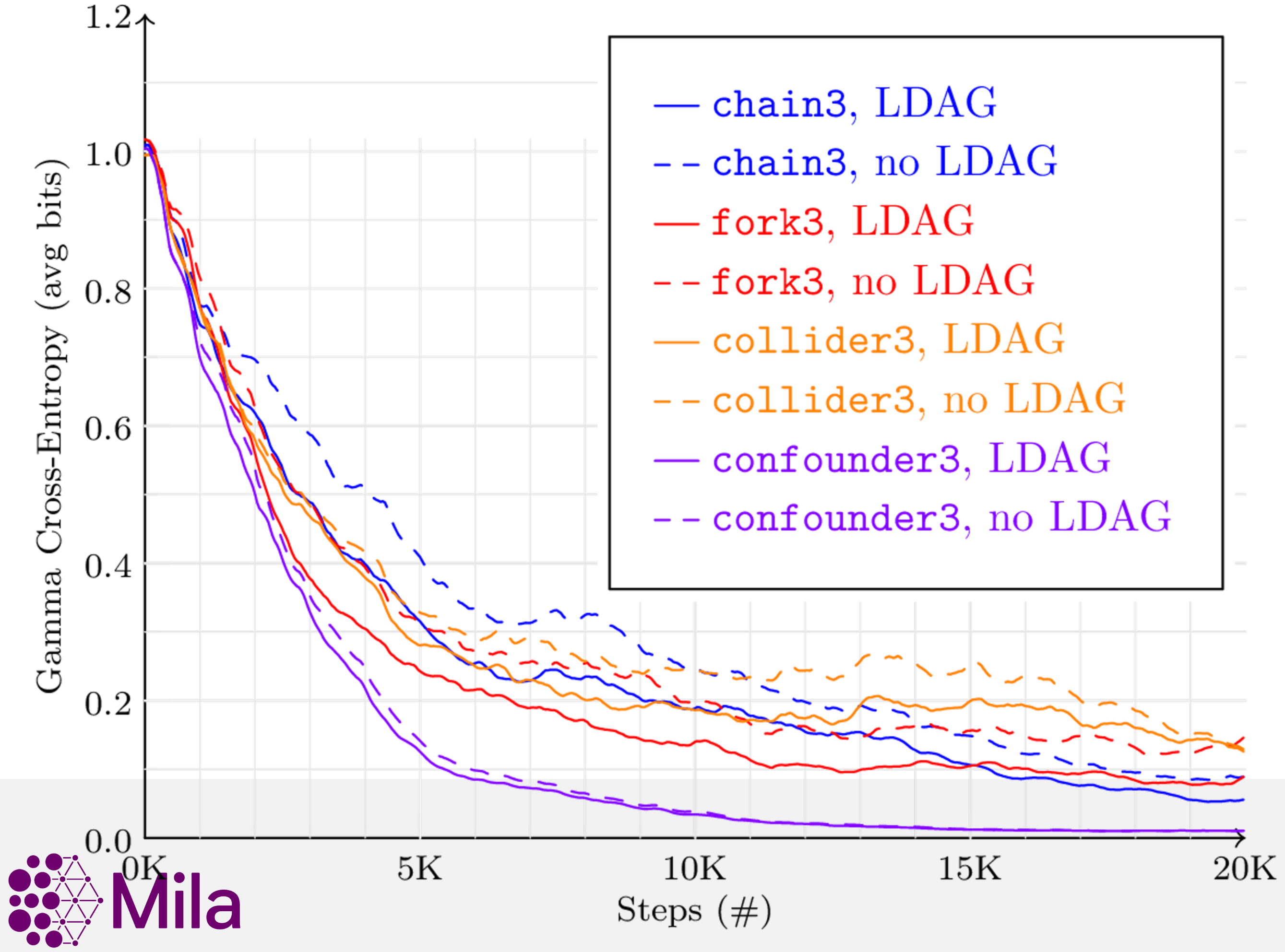
[2]. Preliminary model for barley developed under the project: "Production of beer from Danish malting barley grown without the use of pesticides" by Kristian Kristensen , Ilse A. Rasmussen and others.

ABLATIONS

Generalizing to previously unseen interventions:

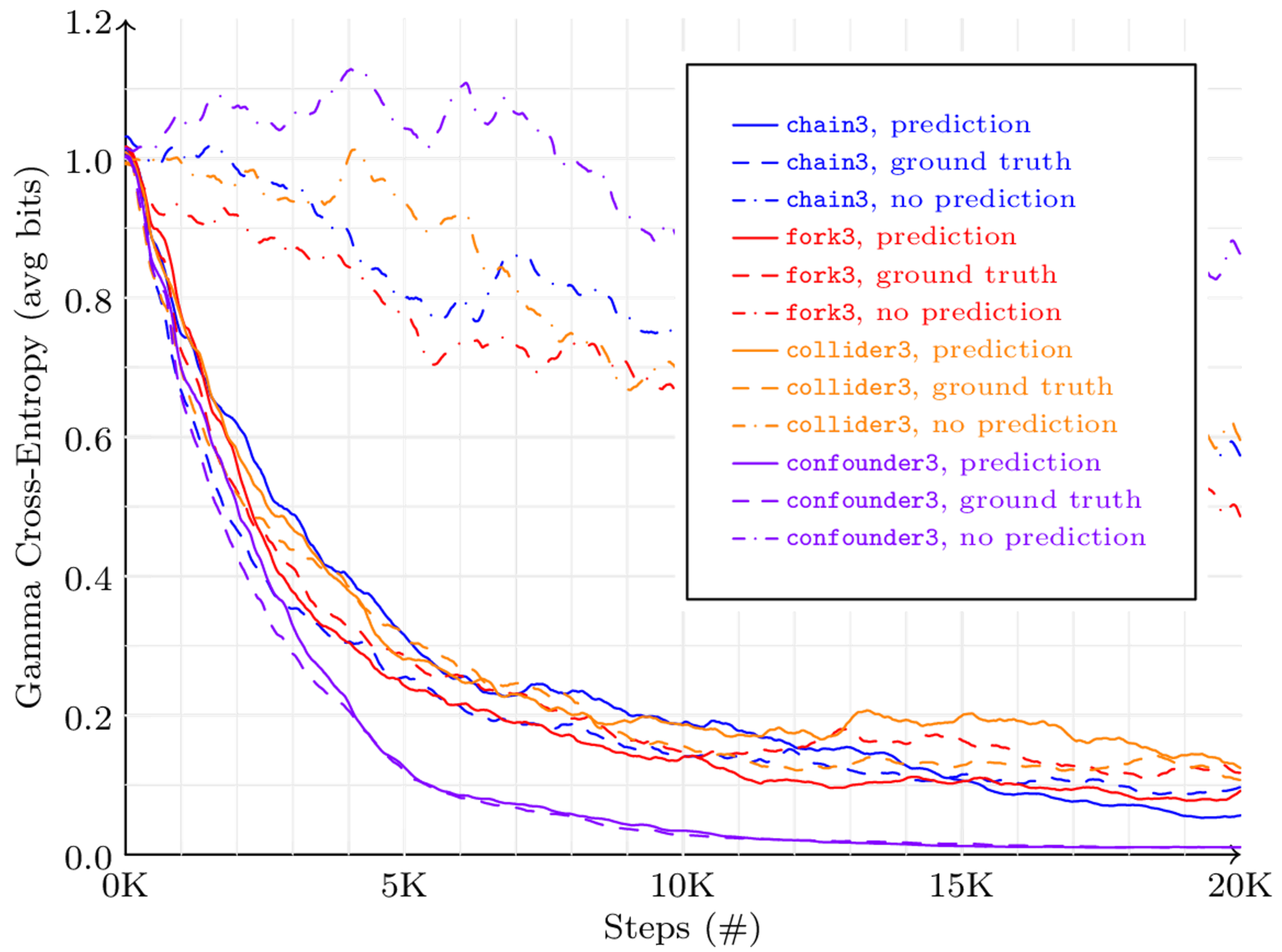
Table 2: **Evaluating the consequences of a previously unseen intervention:** (test log-likelihood under intervention)

	fork3	chain3	confounder3	collider3
Baseline	-0.5036	-0.4562	-0.3628	-0.5082
SDI	-0.4502	-0.3801	-0.2819	-0.4677



Importance of acyclic (LDAG) regularizer:

Importance of predicting the intervention:



OBSERVING OTHER AGENTS

- Can infants figure out causal structure in spite of being almost passive observers?
- Yes, if they exploit and infer the interventions made by other agents

- Our approach does not require the learner to know what the action/intervention was (but it could do inference over interventions)
- But more efficient learning if you can experiment and thus test hypotheses about cause & effect

SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- Distributional changes due to localized causal interventions (in semantic space)
- **Simple mapping between high-level semantic variables / thoughts and words / sentences**
- Shared 'rules' across instance tuples (as arguments), requiring variables & indirection
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

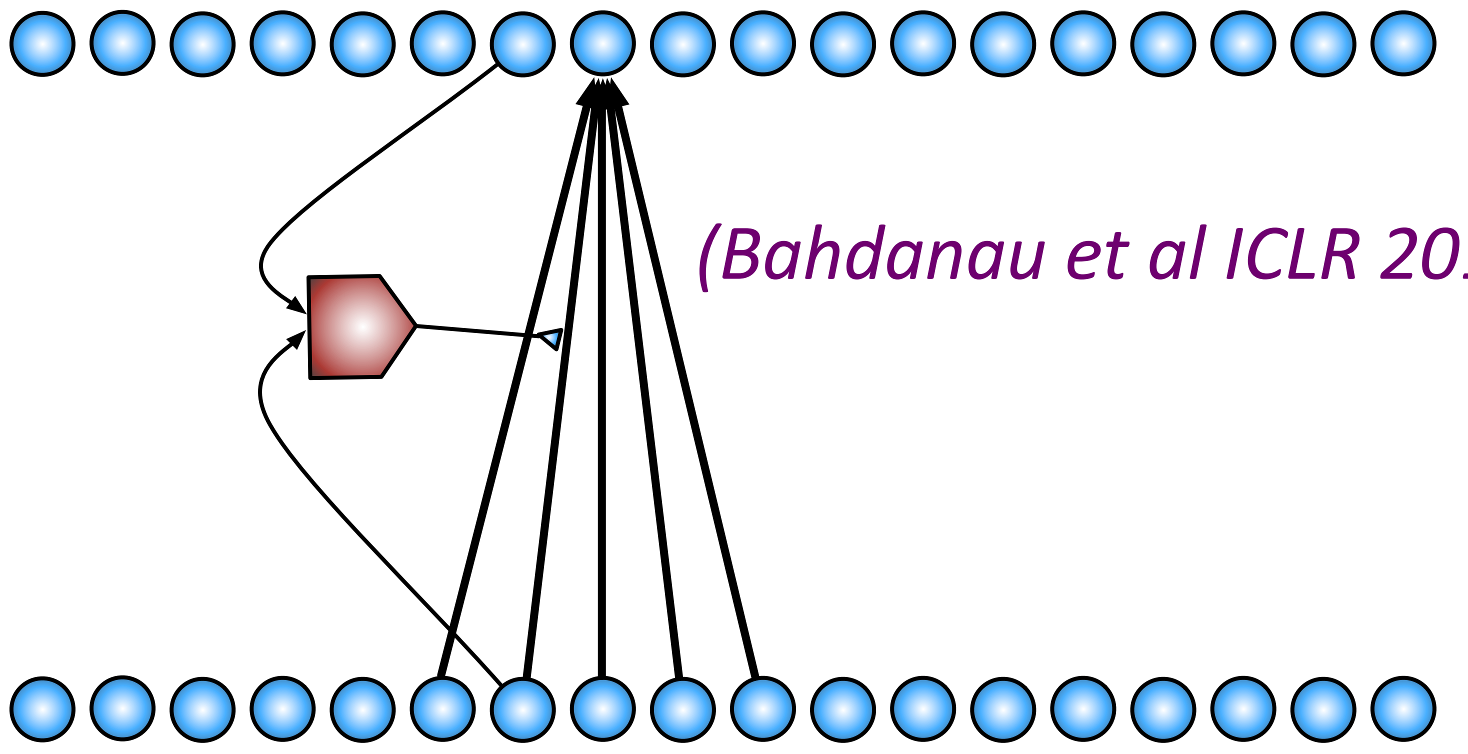
THOUGHTS, CONSCIOUSNESS, LANGUAGE

- Consciousness: from humans reporting
- High-level representations $\Leftrightarrow$ language
- High-level concepts: meaning anchored in low-level perception and action → **tie system 1 & 2**
- Grounded high-level concepts

  → better natural language understanding

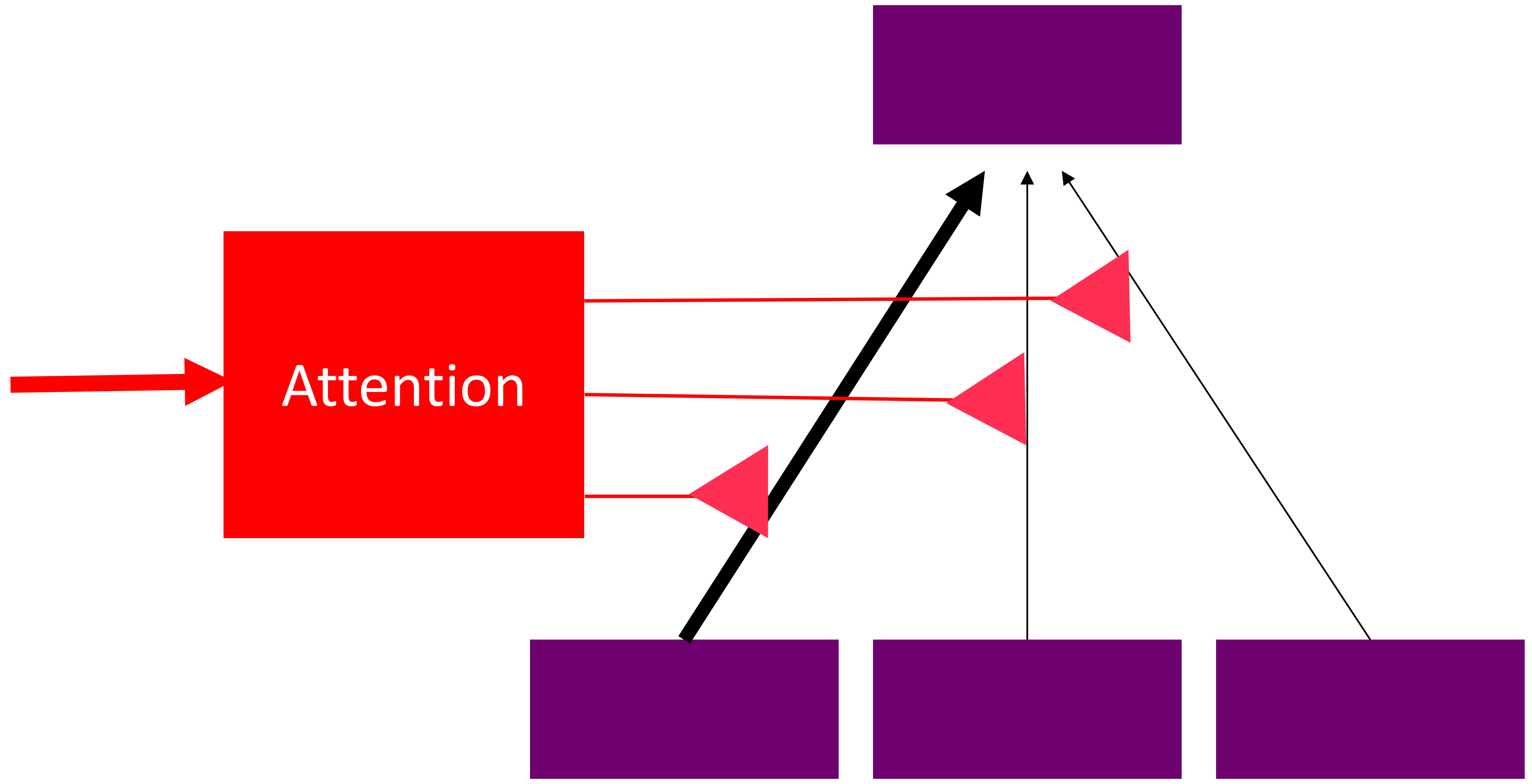
- **Grounded language learning** e.g. BabyAI: *(Chevalier-Boisvert and al ICLR 2019)*

CORE INGREDIENT FOR CONSCIOUS PROCESSING: **ATTENTION**

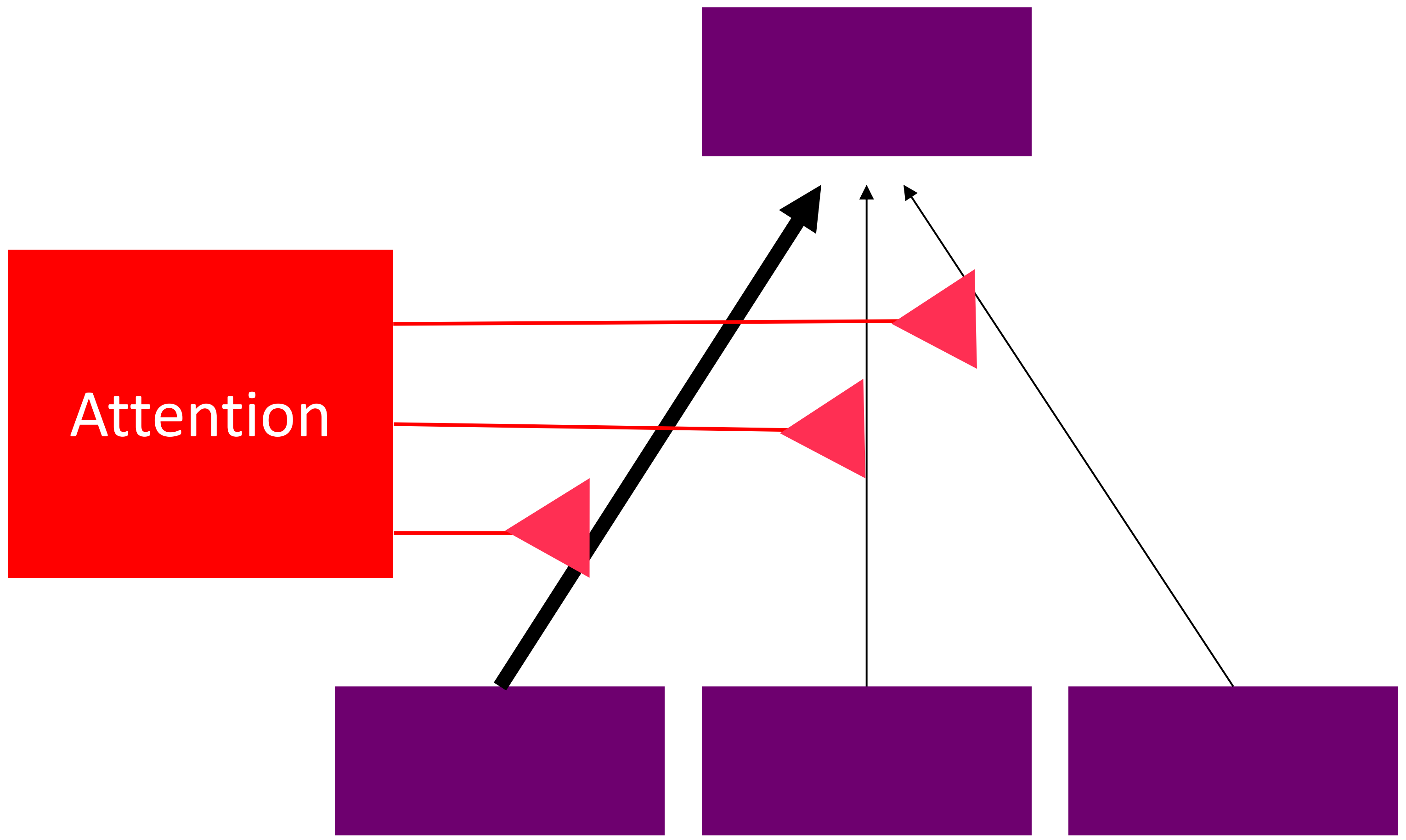


(Bahdanau et al ICLR 2015)

- **Focus** on a one or a few elements at a time
- **Content-based soft attention** is convenient, can backprop to *learn where to attend*
- Attention is an **internal action**, needs a **learned attention policy** *(Egger et al 2019)*
- Operating on unordered SETS of (key, value) pairs
- SOTA in NLP



FROM ATTENTION TO **INDIRECTION**



- Attention = dynamic connection
- Receiver gets the selected value
- Value of what? From where?

  → Also send ‘name’ (or key) of sender
- Keep track of 'named’ objects: indirection
- Manipulate sets of objects (transformers)

P.S. contrary to convnets doing object recognition, sequential tasks involving memory and attention typically involve a more difficult optimization problem, and fighting underfitting (including the issue of long-term dependencies)

RIMS: MODULARIZE COMPUTATION AND OPERATE ON SETS OF NAMED AND TYPED OBJECTS

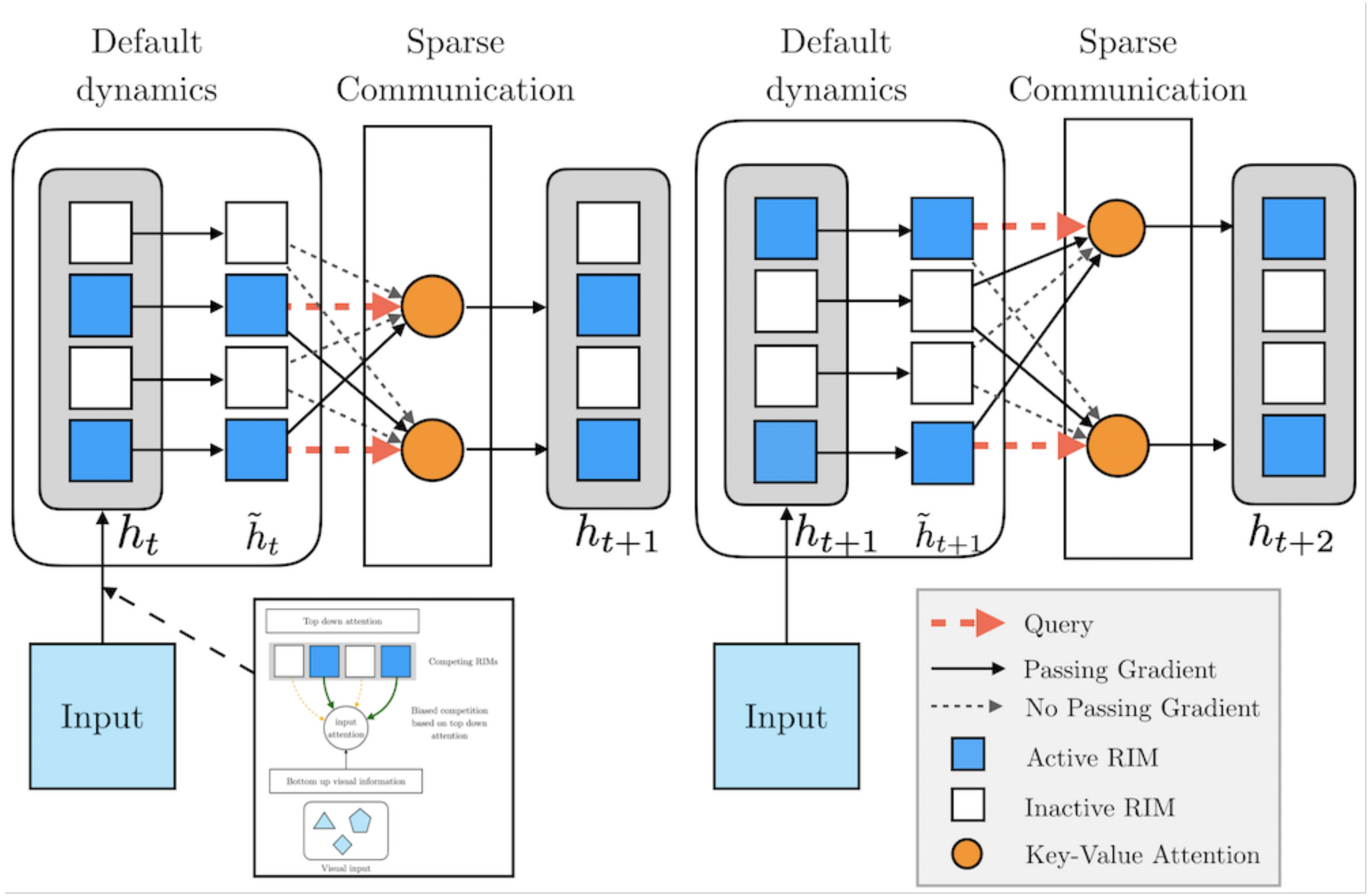
Recurrent Independent Mechanisms

Goyal et al 2019, arXiv:1909.10893

Multiple recurrent sparsely interacting modules, each with their own dynamics, with object (key/value pairs) input/outputs selected by multi-head attention

Results: better ood generalization

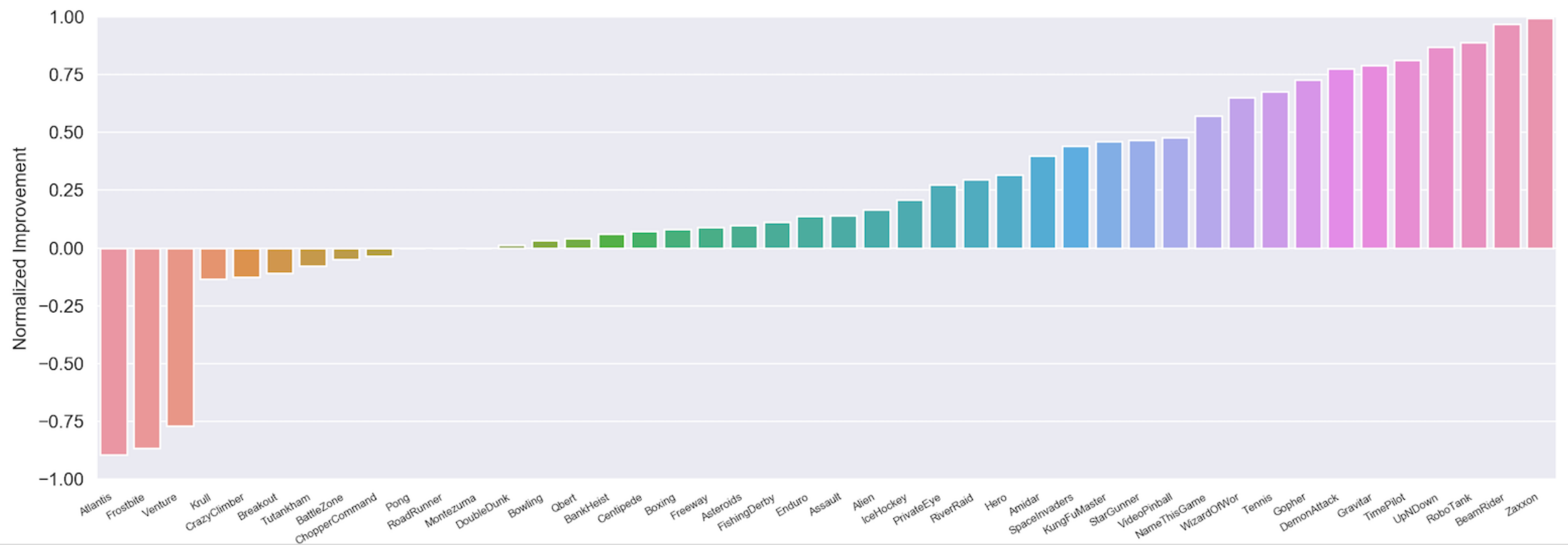
Ongoing work: hierarchy, top-down broadcasting, spatial layout of modules



Builds on rich recent litterature on object-centric representations (mostly for images)

RESULTS WITH **RECURRENT INDEPENDENT MECHANISMS**

- RIMs drop-in replacement for LSTMs in PPO baseline over all Atari games.
- Above 0 (horizontal axis) = improvement over LSTM.



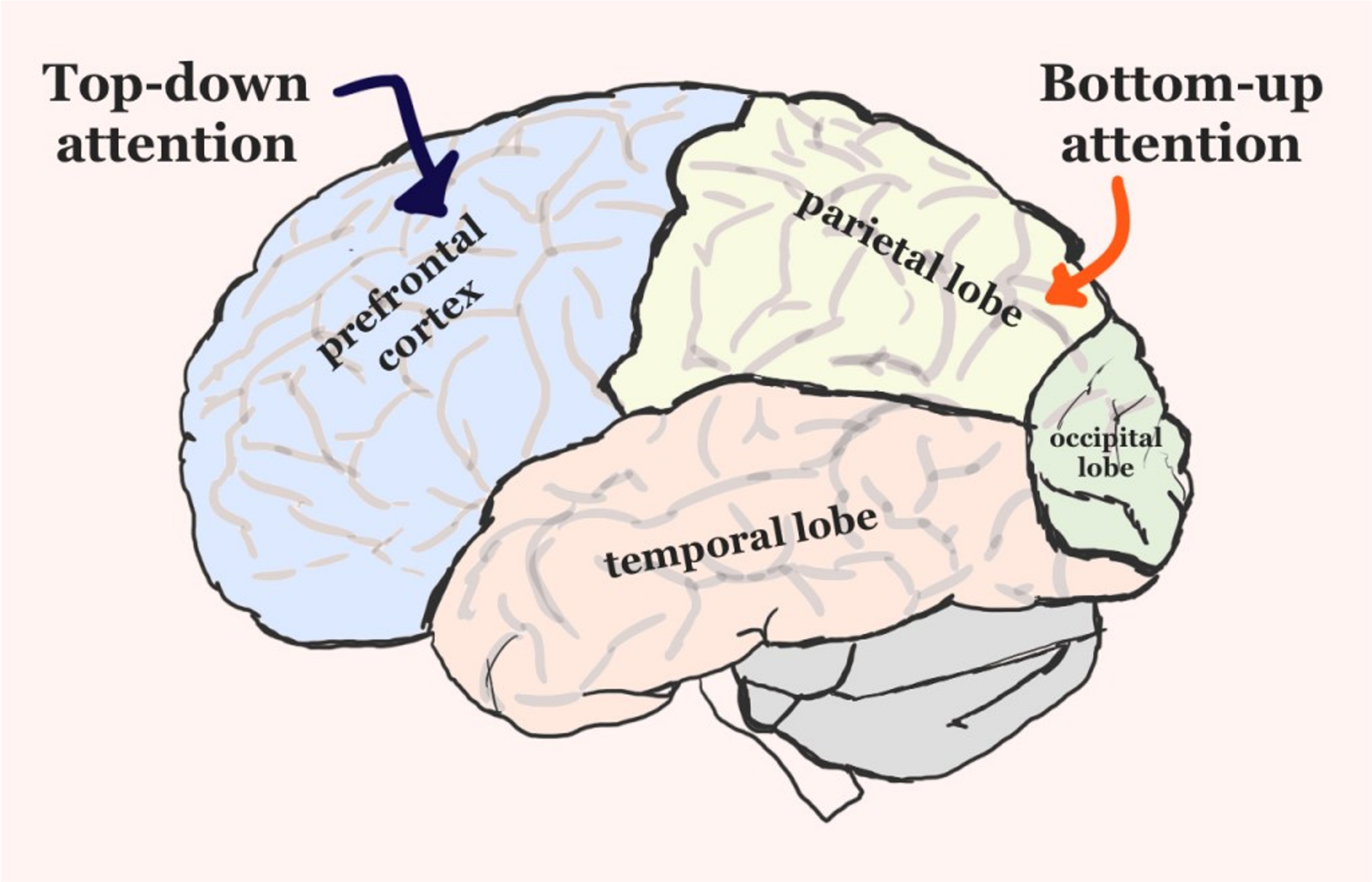
FROM ATTENTION TO **CONSCIOUSNESS**

C-word not taboo anymore in cognitive neuroscience

Global Workspace Theory

(Baars 1988++, Dehaene 2003++)

- Bottleneck of conscious processing
  - *WHY A BOTTLENECK?*
- Selected item is broadcast, stored in short-term memory, conditions perception and action
- System 2-like sequential processing, conscious reasoning & planning & imagination
- Can only run 1 simulation at a time, unlike a movie, only few abstract concepts involved at each step



Modules + Global Workspace

Adding a shared global workspace similar to the GWT greatly improves RIMs

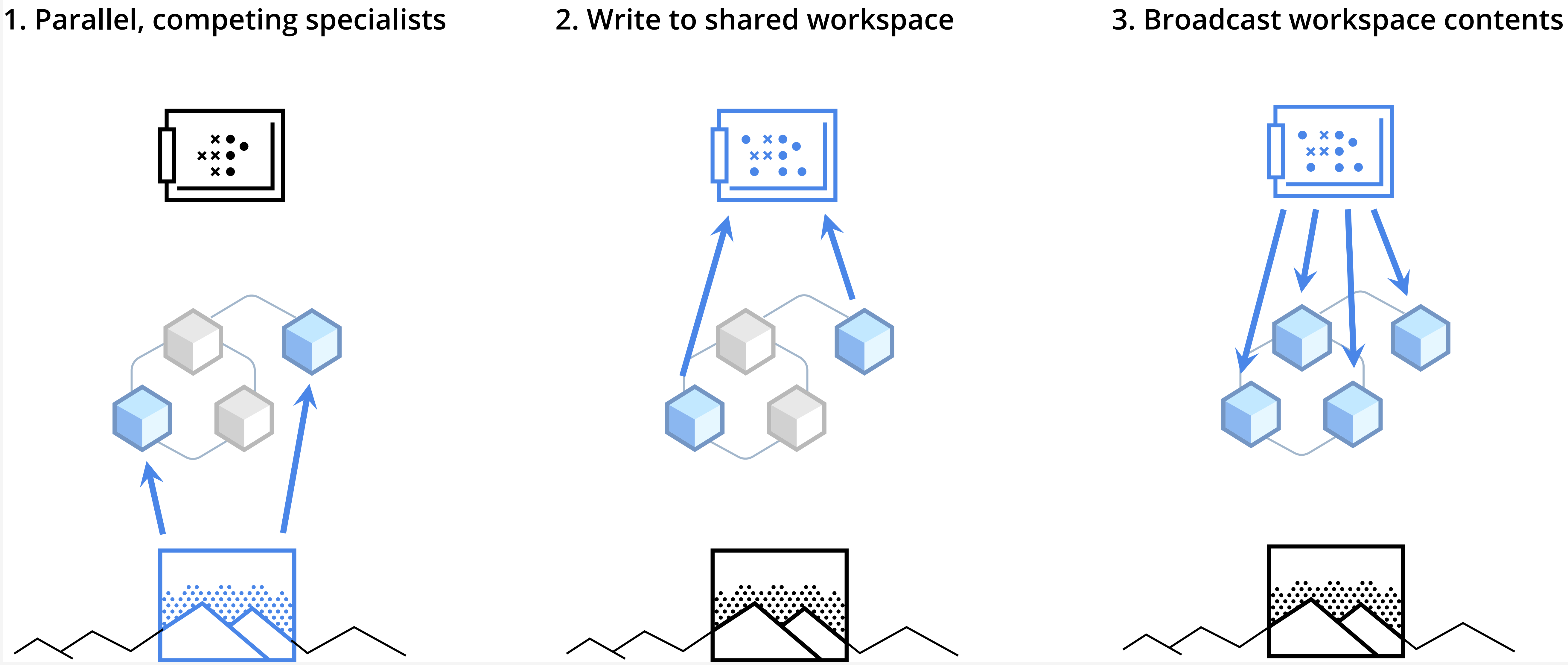
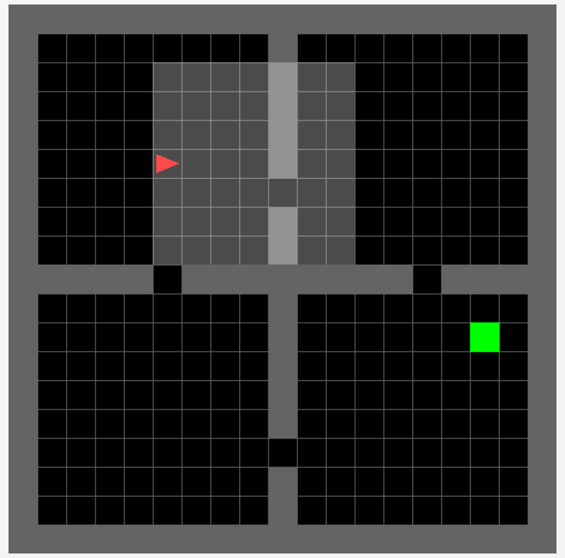
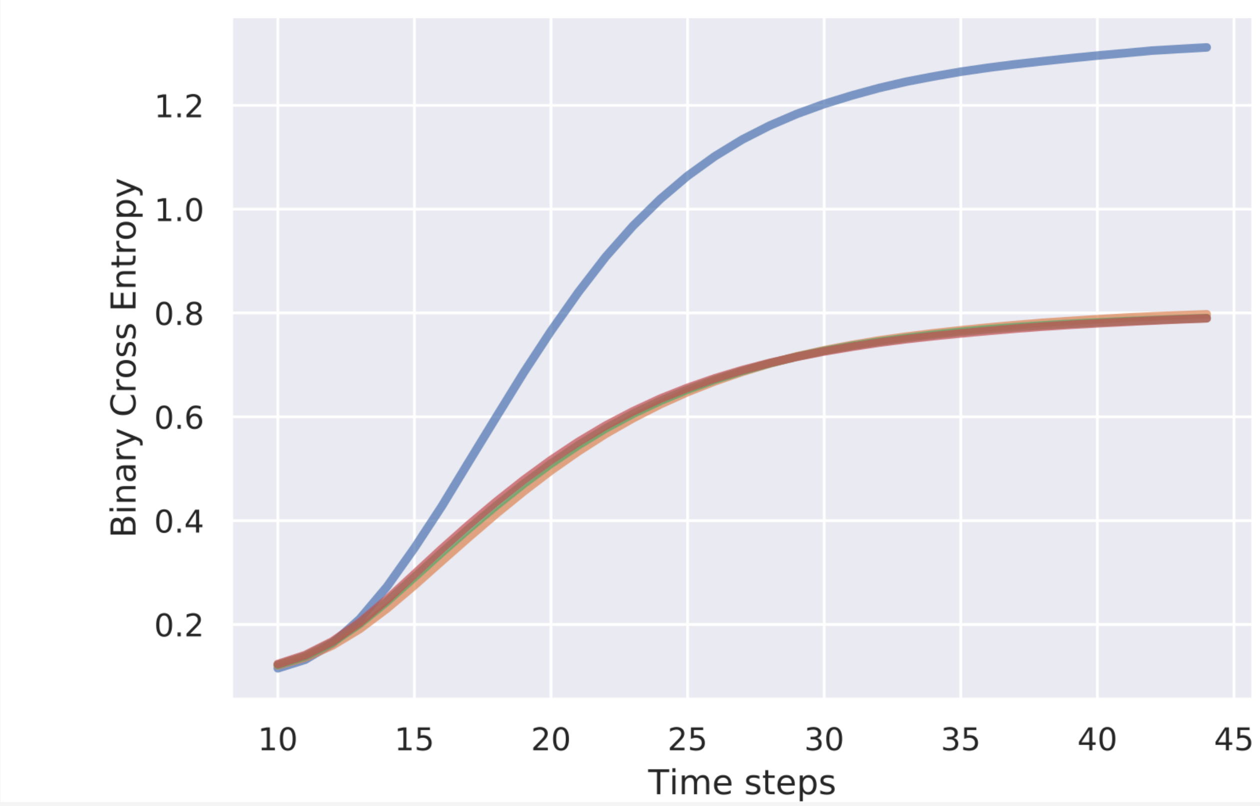


Table 2: **FourRoom Navigation Task:** Success Rate of the proposed method vs. the baselines on the FourRoom navigation environment illustrated on the right, with the agent in red, its field of visibility greyed out, and the object to get in green.

RIMs	RMC	LSTM	Ours
0.72 ± 0.02	0.67 ± 0.05	0.62 ± 0.02	0.96 ± 0.02





without GW

with GW

Tracking bouncing balls (Goyal et al 2020, submitted)

SCHEMAS AND SLOTS

Separate values (slots) from rules (schemas)

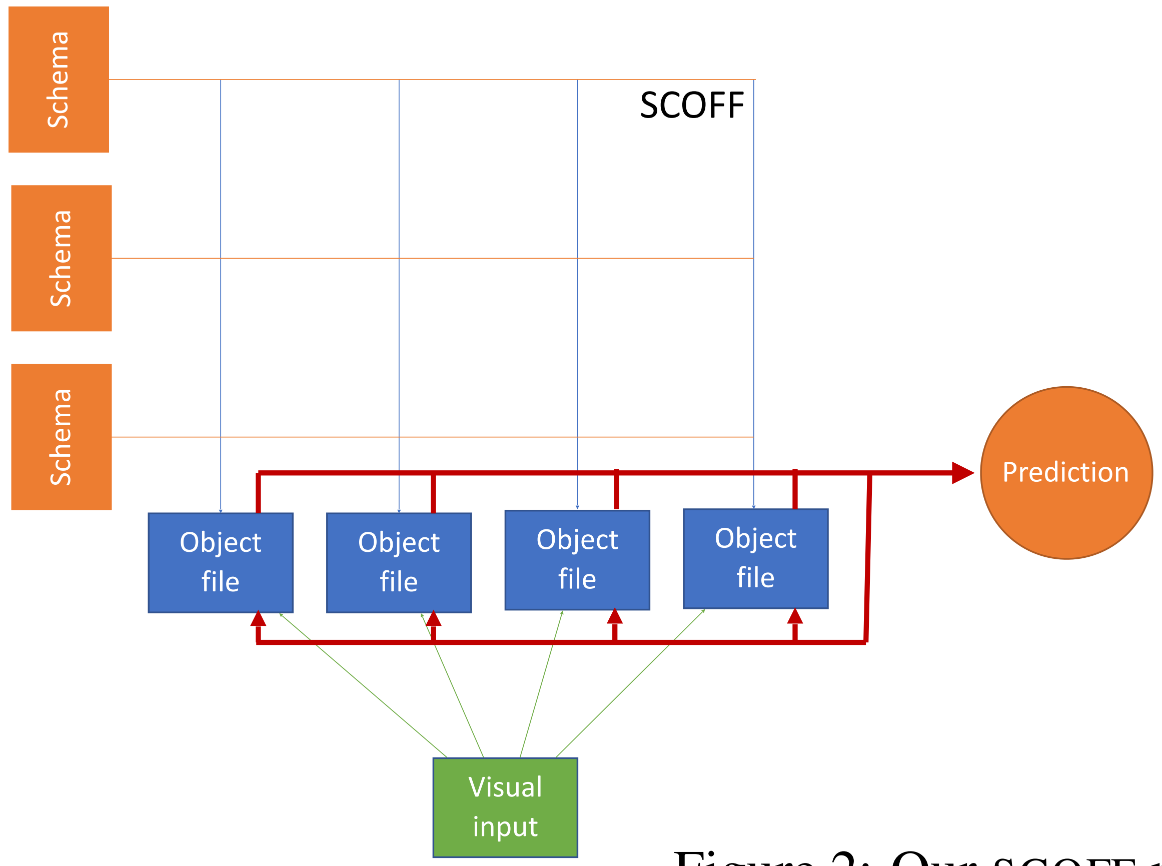


Figure 2: Our SCOFF model. Schemata are sets of parameters that specify the dynamics of objects. Object files are active modules that maintain the time-varying state of an object, seek information from the input, and select schemata for updating.

Object Files	Schema 1 Pacman	Schema 2 Normal Ghost	Schema 3 Scared Ghost
Top Frame			
A	✓		
B		✓	
C		✓	
D		✓	
E		✓	
Bottom Frame			
A	✓		
B			✓
C			✓
D			✓
E			✓

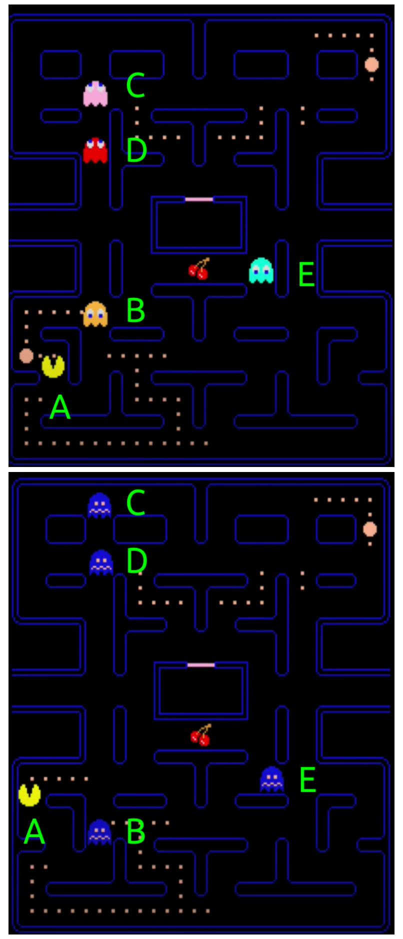


Figure 1: As a motivating example, we show two successive frames of the game PacMan and show how procedural and declarative knowledge must be dynamically factorized. The "B" ghost has a persistent object file (with its location and velocity), yet its procedure mostly depends on whether it is in its *scared* or *normal* routine.

Object Files and Schemata: factorizing declarative and procedural knowledge in dynamical systems

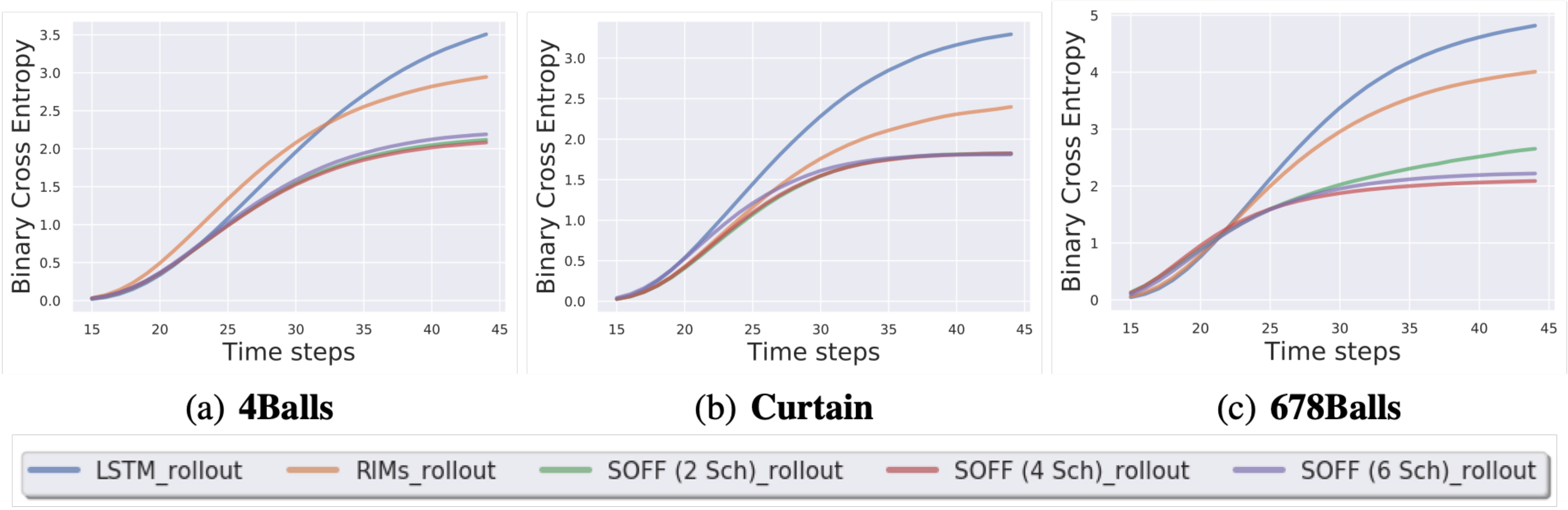
Lamb, Goyal, Blundell, Mozer, Beaudoin, Levine & Bengio, submitted, 2020

SCHEMAS AND SLOTS: RESULTS

Separate values (slots) from rules (schemas)

Number of Values	*LSTM*	*RIMS*	SCOFF
2	0.8731	0.0007	0.0005
3	1.3017	0.0009	0.0007
4	1.6789	0.0014	0.0013
5	2.0334	0.0045	0.0030
8	4.8872	0.0555	0.0191
9	7.3730	0.1958	0.0379
10	11.3595	0.8904	0.0539

Table 1: **Adding Task:** Mean test set error on 200 length sequences with number of numbers to add varying among $\{2, 3, 4, 5, 8, 9, 10\}$. The models are trained to add a mixture of two and four numbers from sequences of length 50.



(a) **4Balls** (b) **Curtain** (c) **678Balls**

(d) Comparison with RIMs and LSTM (sch = schemata)

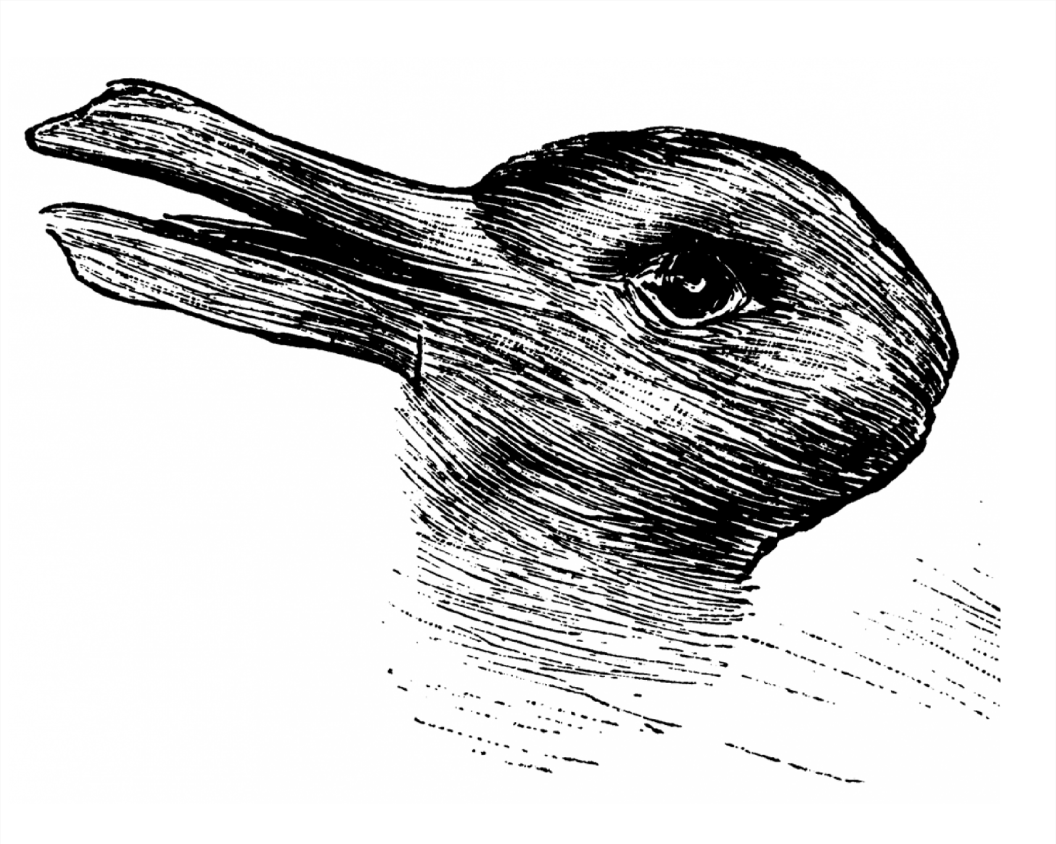
Figure 5: **Bouncing ball motion:** Prediction error comparison of SCOFF, LSTM, and RIMs. Given 10 frames of ground truth, the model predicts the rollout over the next 35 steps. SCOFF performs better than LSTM and RIMs in accurately predicting the dynamics. The advantage of SCOFF is amplified as the number of balls increases—(a) versus (c).

Experiments on Baby AI RL tasks show that slots specialize on objects (like a key) and schematas specialize on procedures (like opening a door) or object detection (like being triggered when the key is in view).

Object Files and Schemata: factorizing declarative and procedural knowledge in dynamical systems
Lamb, Goyal, Blundell, Mozer, Beaudoin, Levine & Bengio, submitted, 2020

Learning to Combine Top-Down and Bottom-Up Signals

Sarthak Mittal, Alex Lamb, Anirudh Goyal, Vikram Voleti, Murray Shanahan, Guillaume Lajoie, Michael Mozer, Yoshua Bengio, ICML 2020



Properly combining the contextual information and prior with the bottom-up signal can be useful even at the lower levels of perceptual processing and changes the lower-level interpretations.

Learning to Combine Top-Down and Bottom-Up Signals

ICML'2020

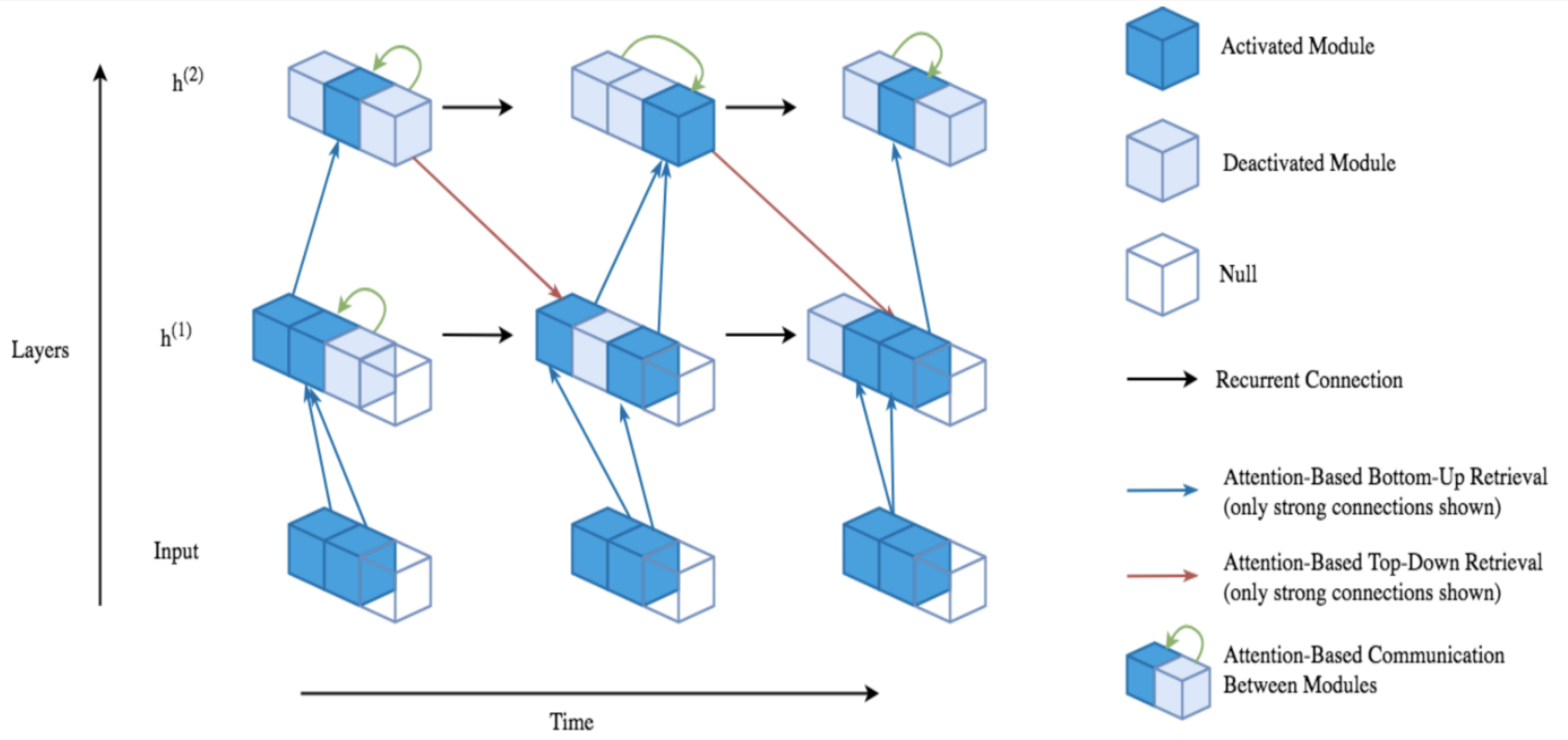


Figure 2: Proposed architecture. Bidirectional connections to provide top-down information (red arrows); Sparse Activation of modules (dark blue - active); Communication within each layer (green arrows)

Mila

Learning to Combine Top-Down and Bottom-Up Signals

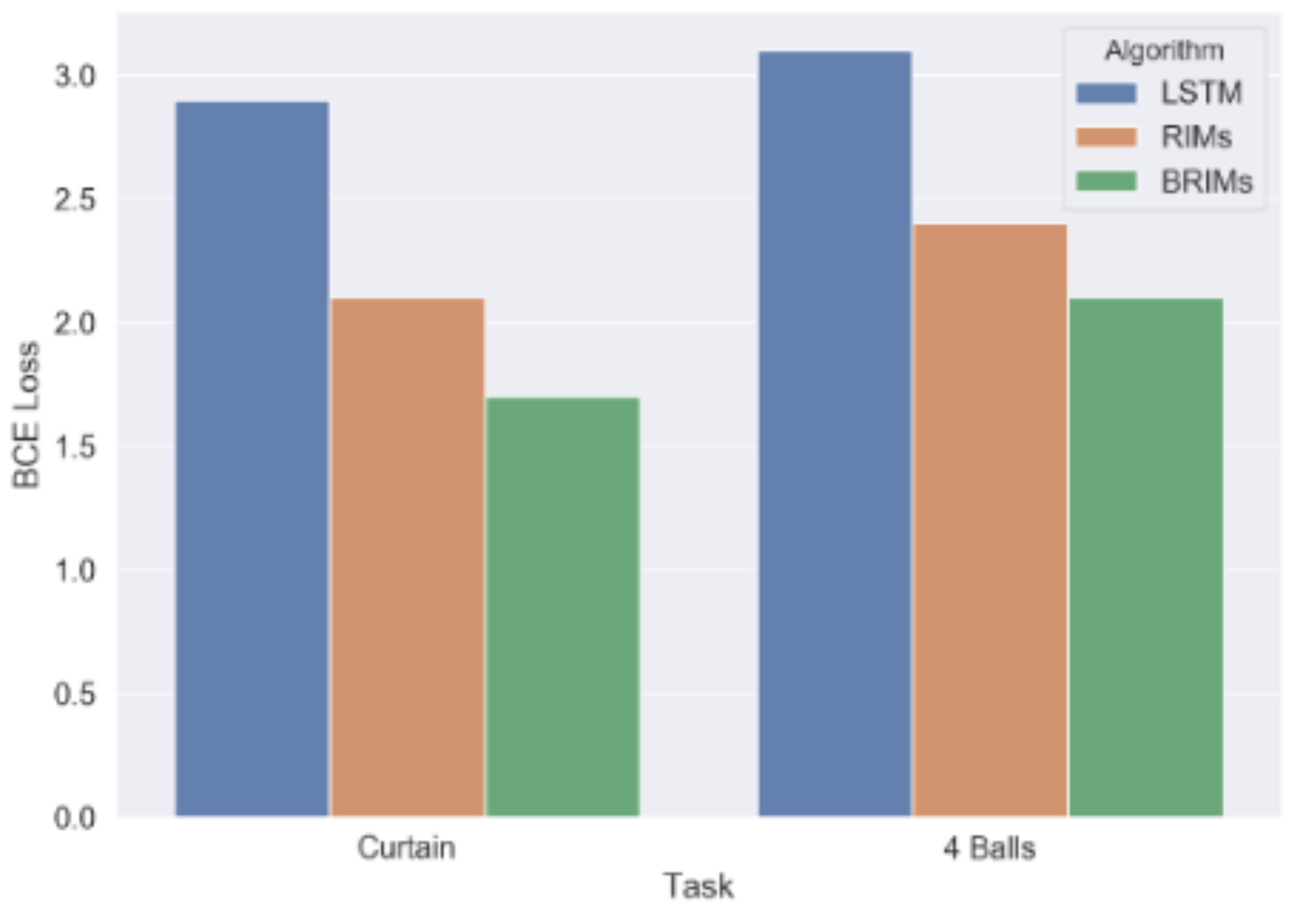
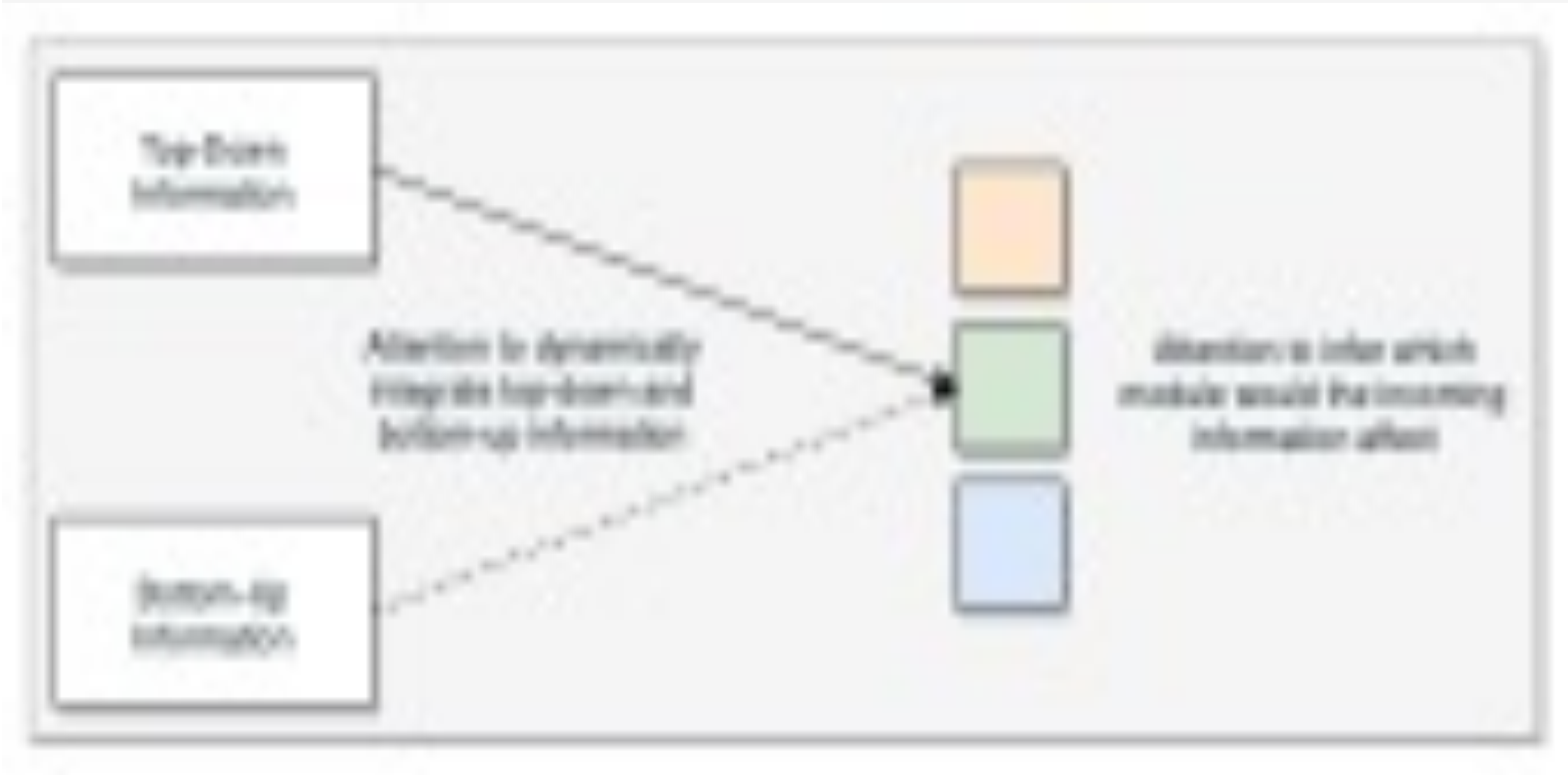


Figure 12: Performance on Bouncing Balls task. The task has multiple balls bouncing around so each ball has its own independent dynamics. They react only through collisions. Curtain provides examples with occlusion.



Use of Key-Value attention to integrate top-down and bottom-up information in context-dependent and dynamic way and to infer a sparse relationship between the incoming observations and the set-structured state representation.

Learning to Combine Top-Down and Bottom-Up Signals

Algorithm	*Properties*	*19 x 19*	*24 x 24*	*32 x 32*
LSTM	—	54.4	44.0	32.2
LSTM	H	57.0	46.8	33.2
LSTM	H+B	56.5	52.2	42.1
LSTM	H+A	56.7	51.5	40.0
LSTM	H+A+B	59.9	54.6	43.0
RMC	A	49.9	44.3	31.3
RIMs	A+M	56.9	51.4	40.1
Hierarchical RIMs	H+A+M	57.2	54.6	46.8
MLD-RIMs	H+A+M	56.8	53.1	44.5
BRIMs (ours)	H+A+B+M	60.1	57.7	52.2

Figure 10: We train sequential models on CIFAR10 where they see one pixel at a time. The models are trained on 16×16 resolution and then evaluated on 19×19, 24×24 and 32×32 resolutions. We see that BRIMs generalize very well across changes in sequence length.

Learning to Combine Top-Down and Bottom-Up Signals

Environment	*LSTM*	*RIMs*	*BRIMs (ours)*
Alien	1612 ± 44	2152 ± 81	4102 ± 400
Amidar	1000 ± 58	1800 ± 43	2454 ± 100
Assault	4000 ± 213	5400 ± 312	5700 ± 320
Asterix	3090 ± 420	21040 ± 548	30700 ± 3200
Asteroids	1611 ± 200	3801 ± 89	2000 ± 300
Atlantis	3.28M ± 0.20M	3.5M ± 0.12M	3.9M ± 0.05M
BankHeist	1153 ± 23	1195 ± 4	1155 ± 20
BattleZone	21000 ± 232	22000 ± 324	25000 ± 414
BeamRider	698 ± 100	5320 ± 300	4000 ± 323
MsPacMan	4598 ± 100	3920 ± 500	5900 ± 1000

Figure 11: We replace LSTM with BRIMs in an RL agent trained with PPO and show that BRIMs outperform their competitors on a set of randomly chosen Atari games.

Noisy Inputs: more attention to top-down signals

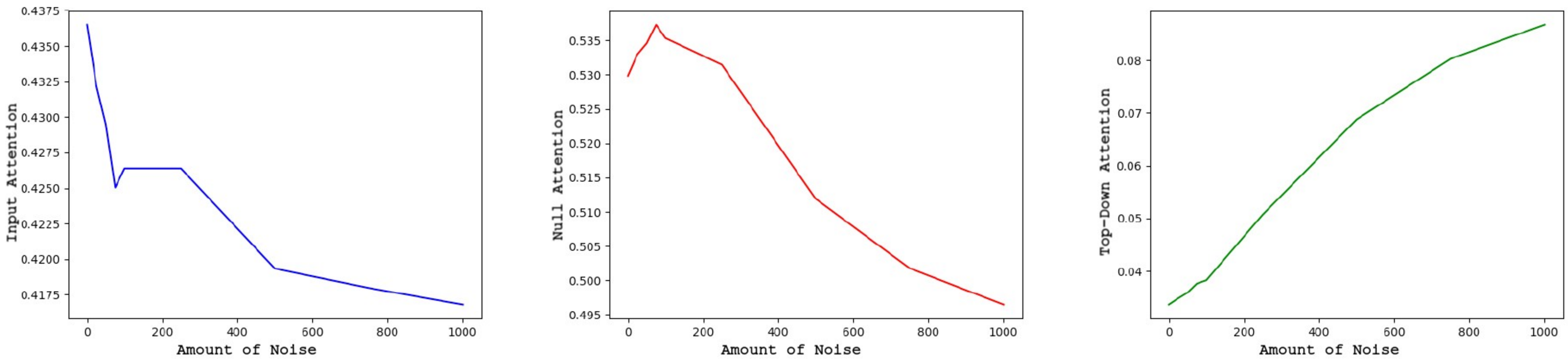


Figure 14: Attention given to input (left), zero vector (middle), and top-level (right), as a function of noise injected into CIFAR images. We see that as the amount of noise increases in the image, the model's reliance on higher level information increases. This is in line with our hypothesis that top-down modulation should be queried more in case of uncertainty.

SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- Distributional changes due to localized causal interventions (in semantic space)
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared ’rules’ across instance tuples (as arguments), requiring variables & indirection
- Meaning (e.g. causal graph or an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

SCHEMAS AND SLOTS

Separate values (slots) from rules (schemas)

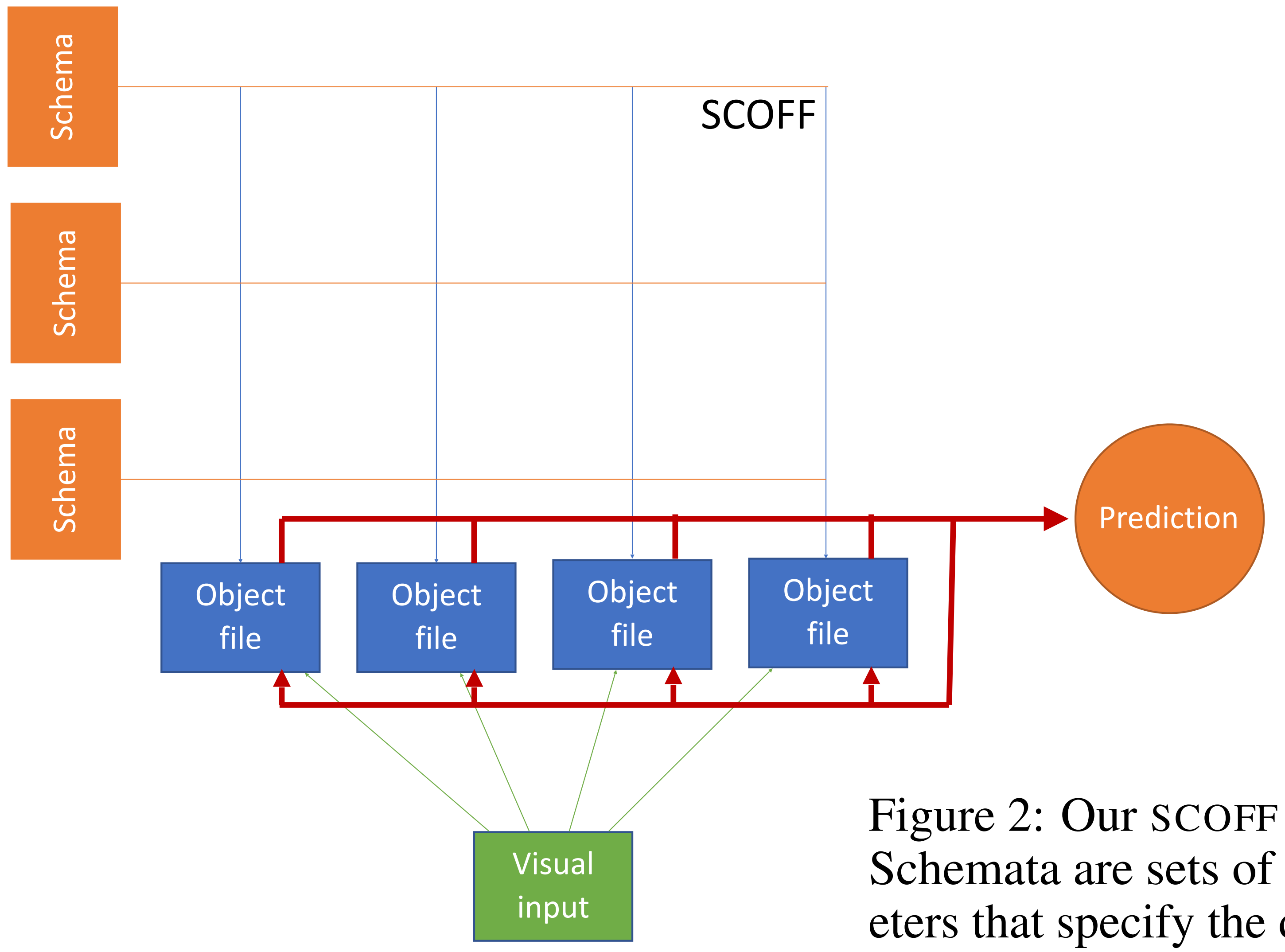


Figure 2: Our SCOFF model. Schemata are sets of parameters that specify the dynamics of objects. Object files are active modules that maintain the time-varying state of an object, seek information from the input, and select schemata for updating.

Object Files	Schema 1 Pacman	Schema 2 Normal Ghost	Schema 3 Scared Ghost
Top Frame			
A	✓		
B		✓	
C		✓	
D		✓	
E		✓	
Bottom Frame			
A	✓		
B			✓
C			✓
D			✓
E			✓

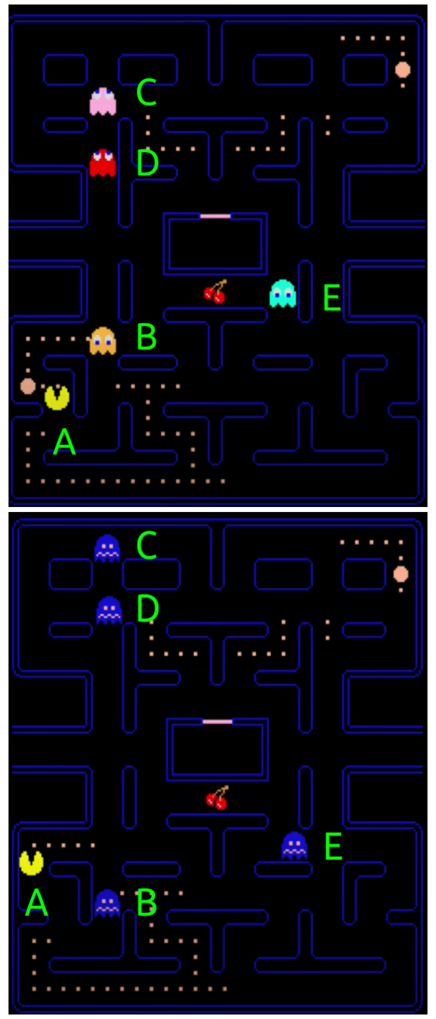


Figure 1: As a motivating example, we show two successive frames of the game PacMan and show how procedural and declarative knowledge must be dynamically factorized. The "B" ghost has a persistent object file (with its location and velocity), yet its procedure mostly depends on whether it is in its *scared* or *normal* routine.

Object Files and Schemata: factorizing declarative and procedural knowledge in dynamical systems

Goyal, Lamb, Gampa, Blundell, Mozer, Beaudoin, Levine & Bengio, submitted, 2020

SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- Distributional changes due to localized causal interventions (in semantic space)
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared 'rules' across instance tuples (as arguments), requiring variables & indirection
- **Meaning (e.g. causal graph or an encoder) is stable & robust wrt changes in distribution**
- Credit assignment is only over short causal chains

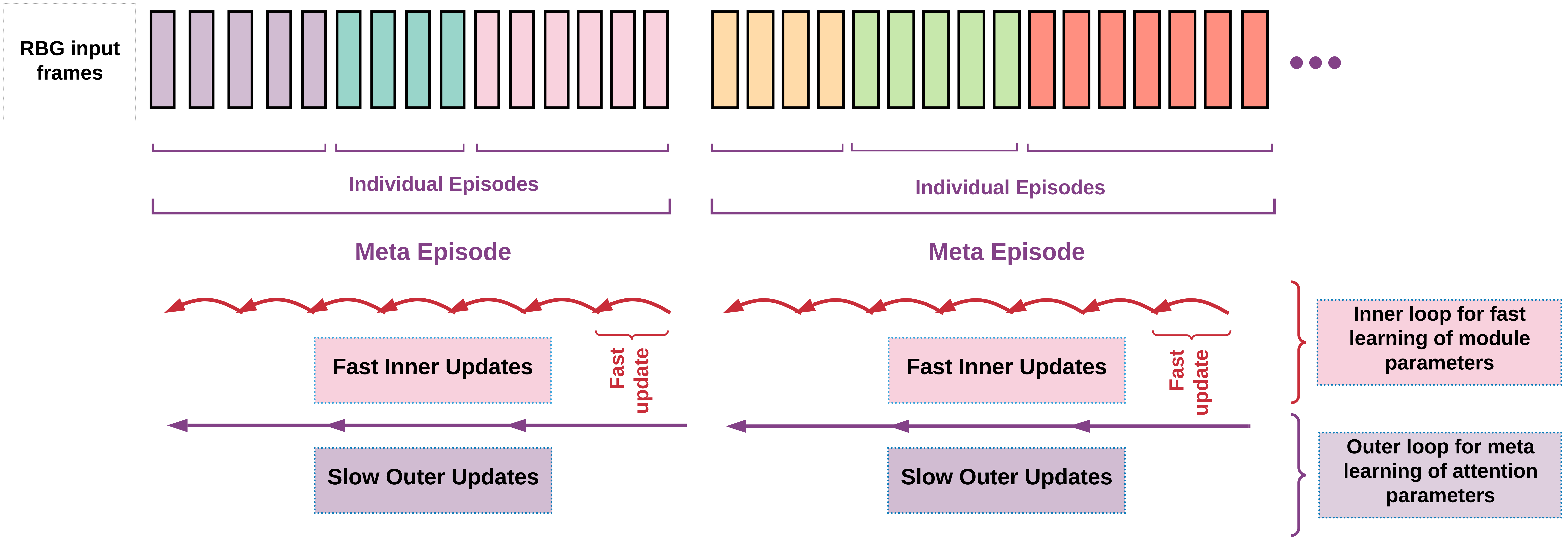
Fast and slow weights

- Slow weights
  - Ground-truth causal graph

$$\mathcal{R} = -\mathbb{E}_{X \sim D_{\text{int}}} [\log \mathbb{E}_{C \sim \text{Ber}(\gamma)} [\prod_i \mathcal{L}_{C,i}(X; \theta_{\text{slow}})]]$$

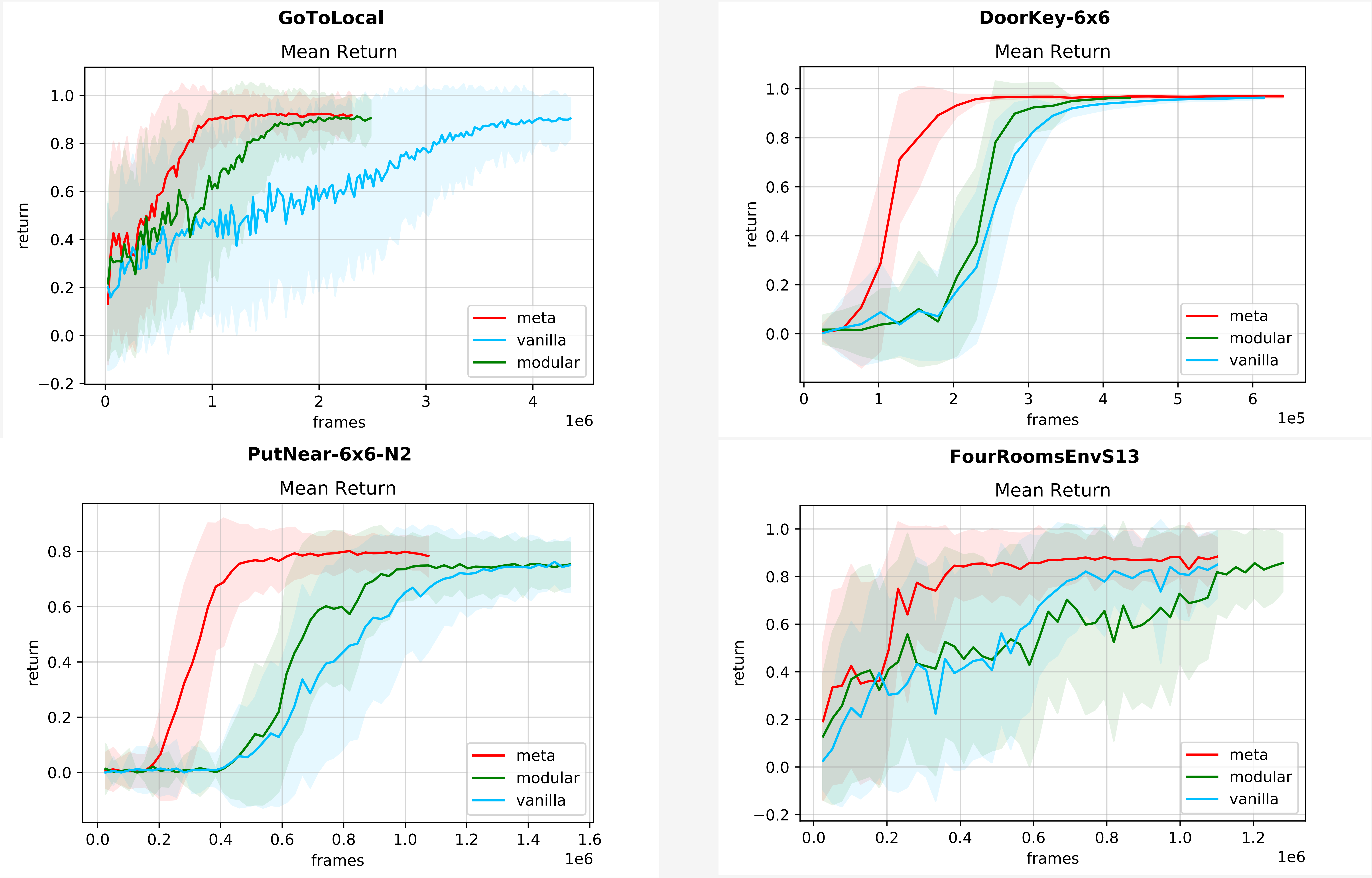
- Fast weights
  - Adapt to local (intervention) changes

Meta-Learning = Multiple Time Scales of Learning



Mila

Meta-Attention Networks



RIMs + meta-learning

Fast learning: modules

Slow learning: attention mechanism

Experiments on Baby AI tasks (Kanika Madan et al 2020, submitted)

SOME SYSTEM 2 INDUCTIVE PRIORS
all inspired by human cognition

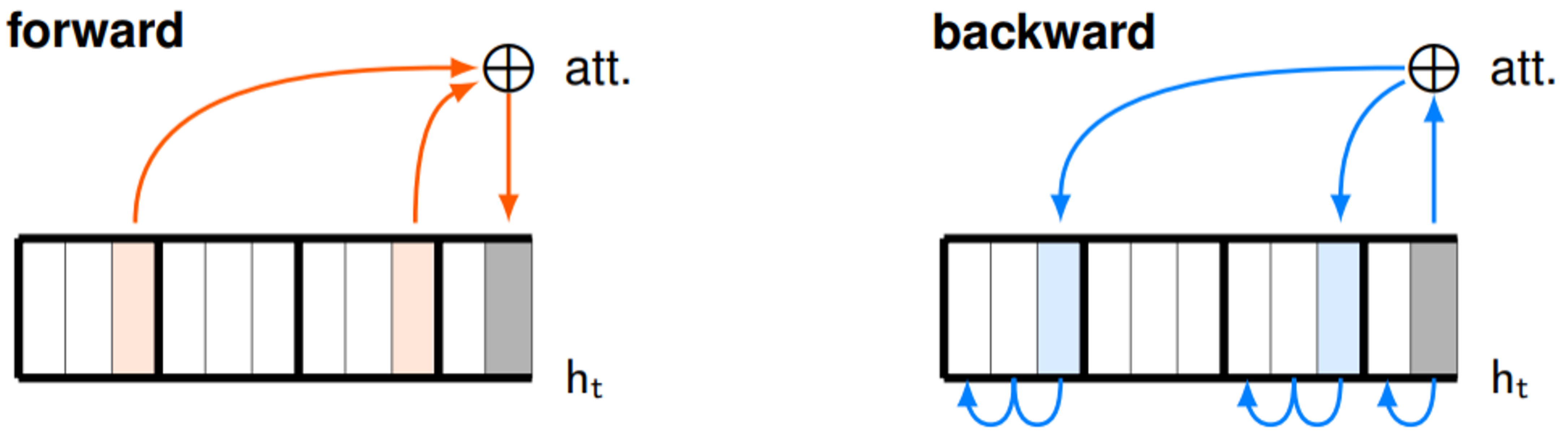
- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- Distributional changes due to localized causal interventions (in semantic space)
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared ’rules’ across instance tuples (as arguments), requiring variables & indirection
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

Sparse Attentive Backtracking

Rosemary Ke, Anirudh Goyal, Olexa Bilaniuk, Jonathan Binas, Mike Mozer, Yoshua Bengio,

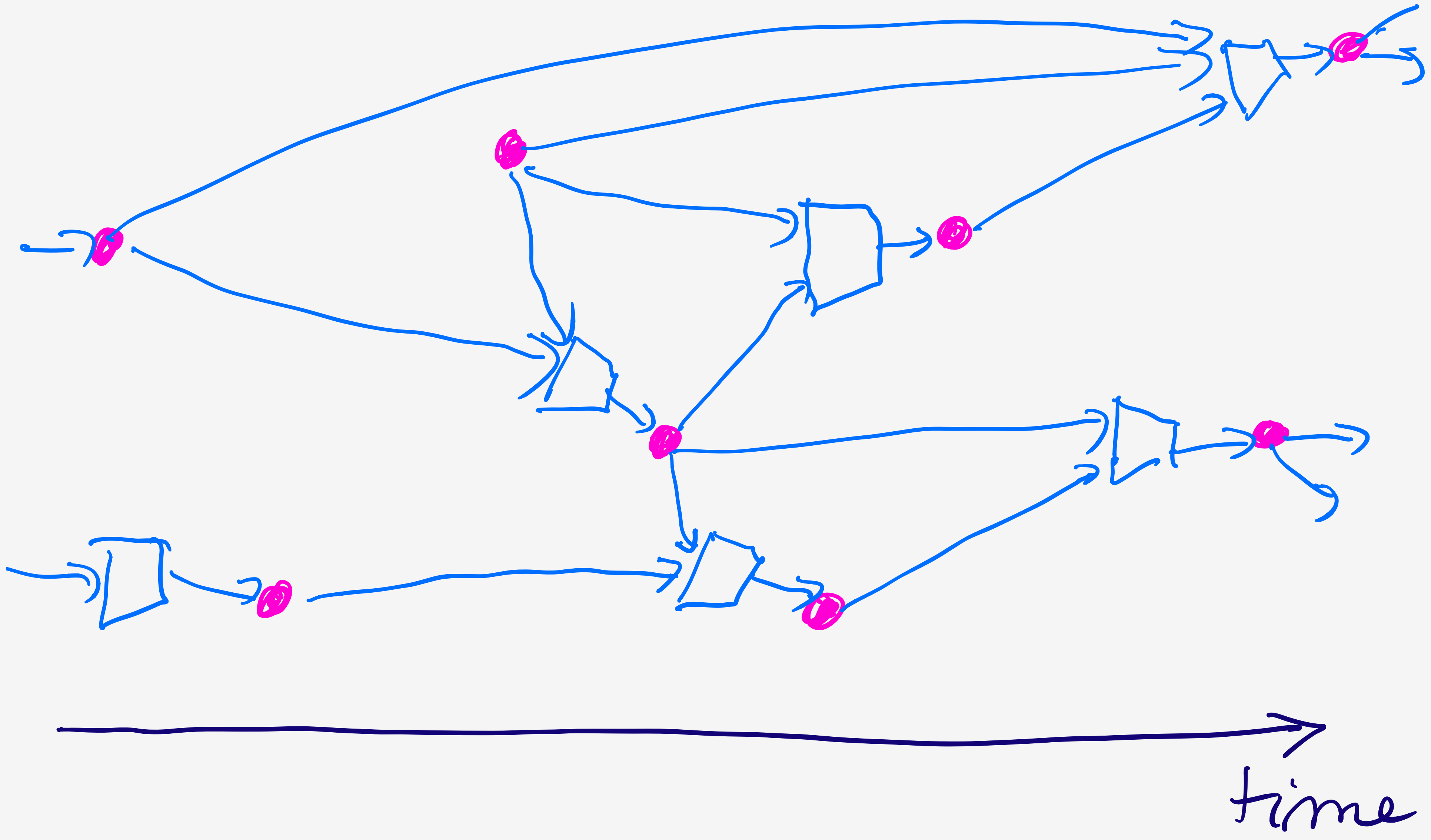
NeurIPS 2018

The attention mechanism of the associative memory picks up past memories which match (associate with) the current state, maybe be an alternative to BPTT for learning very long-term dependencies.



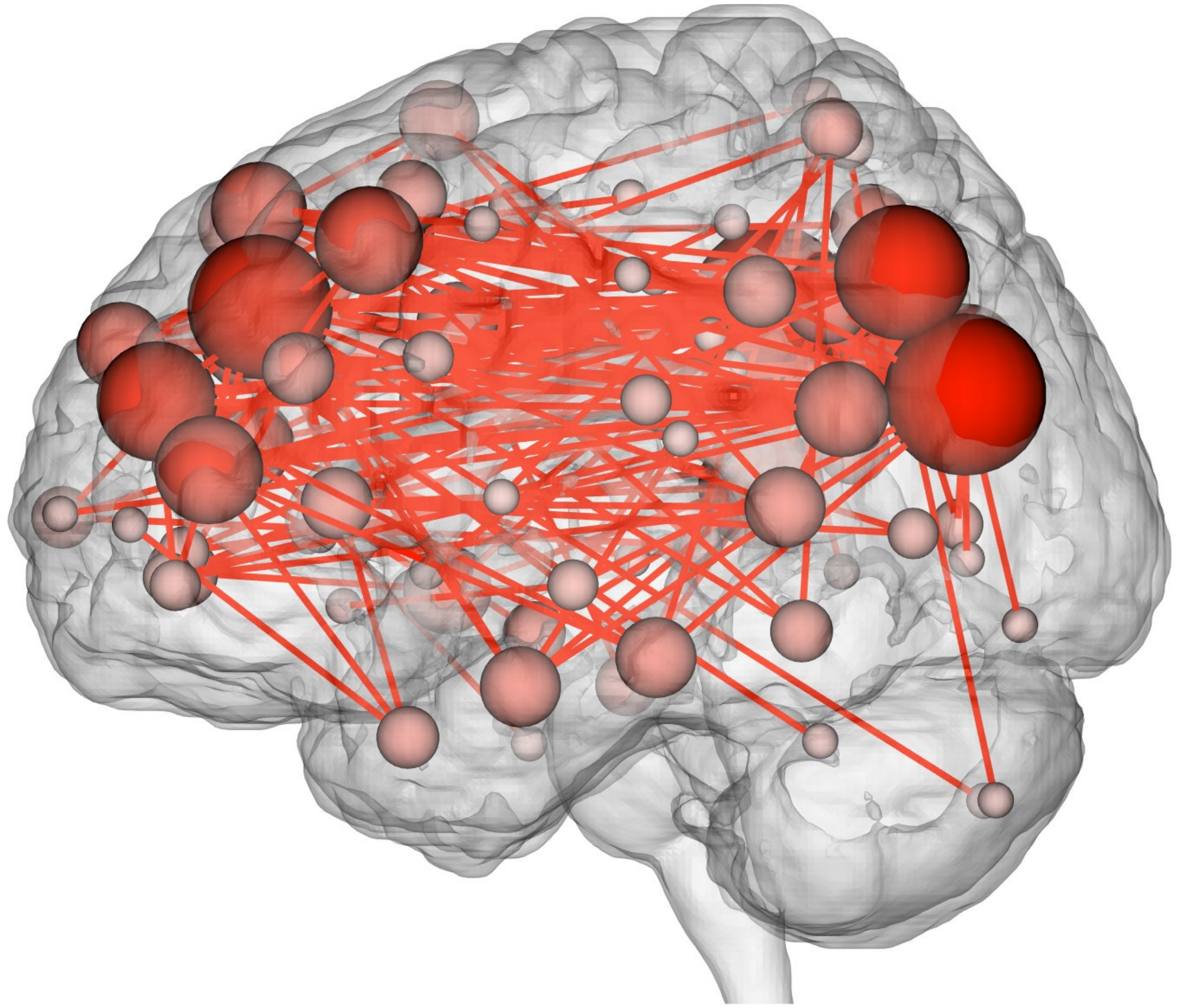
Causal reasoning over events factor graph

- Node of graph = event at particular time, involving a small set of variables
- Factor = causal mechanism
- Directed edges: from past to future, causal direction



LEARNING TO REASON

- Reasoning and planning is inference and is inherently computationally expensive
- Brains do not use exhaustive search but instead generate good candidates
- Conscious processing seems involved in evaluating them for global coherence across the brain's modules
- Attention mechanisms are part of the reasoning policy, converting declarative knowledge into selective computations for inference and decision-making



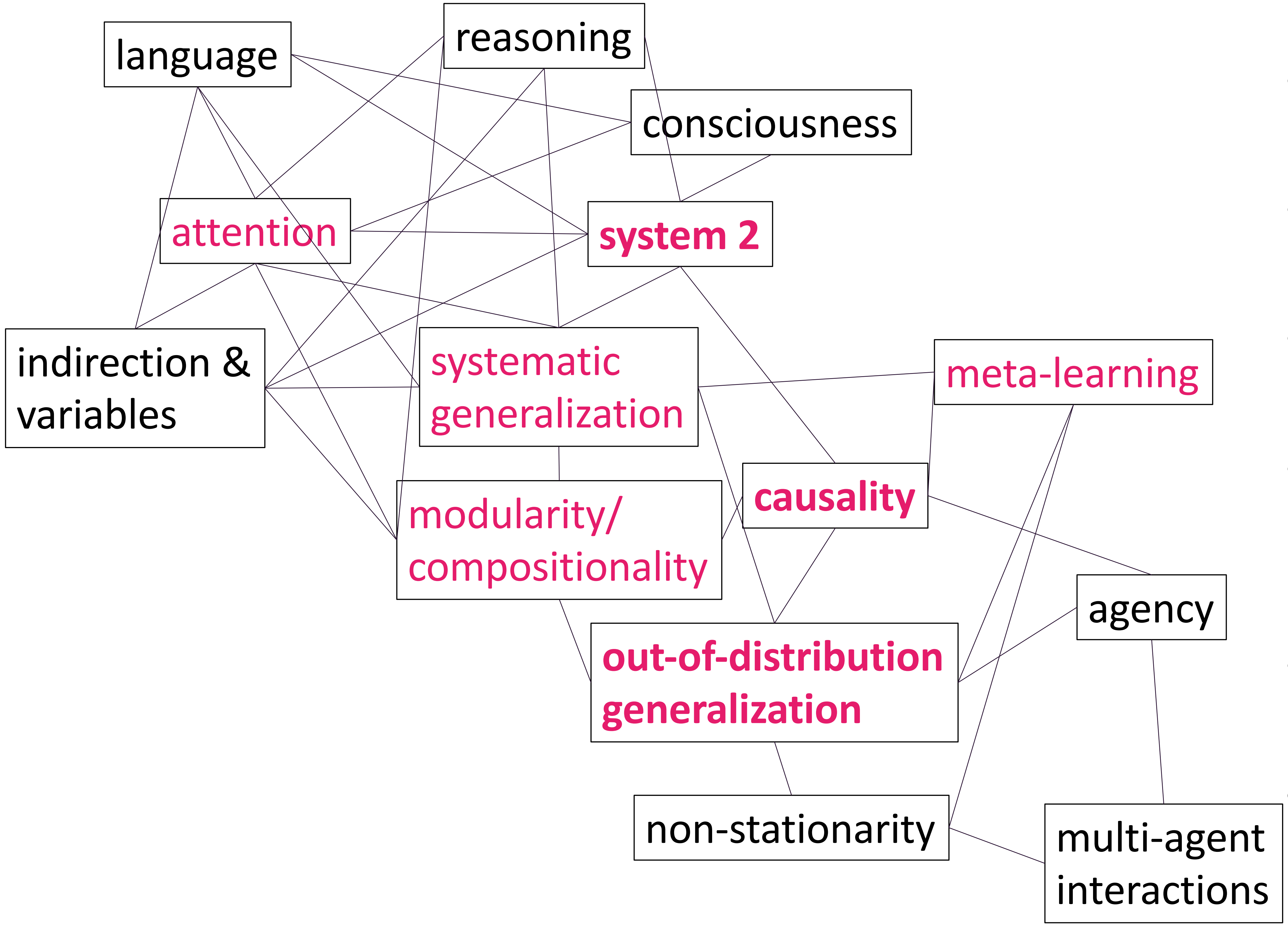
CONTRAST WITH **THE SYMBOLIC AI PROGRAM**

Avoid pitfalls of classical AI rule-based symbol-manipulation

- Need efficient large-scale learning
- Need semantic grounding in system 1 (implicit knowledge)
- Need distributed representations for generalization
- Need efficient = trained search (also system 1)
- Need uncertainty handling

But want

- Systematic generalization
- Factorizing knowledge in small exchangeable pieces
- Manipulating variables, instances, references & indirection



CONSCIOUSNESS PRIORS

- Sparse factor graph in space of high-level semantic variables
- Semantic variables are causal: agents, intentions, controllable objects
- Simple mapping between high-level semantic variables / thoughts and words / sentences
- Shared 'rules' across instance tuples (as arguments), w/ variables & indirection

- Distributional changes due to localized causal interventions (in semantic space)
- Meaning (e.g. grounded by an encoder) is stable & robust wrt changes in distribution
- Credit assignment is only over short causal chains

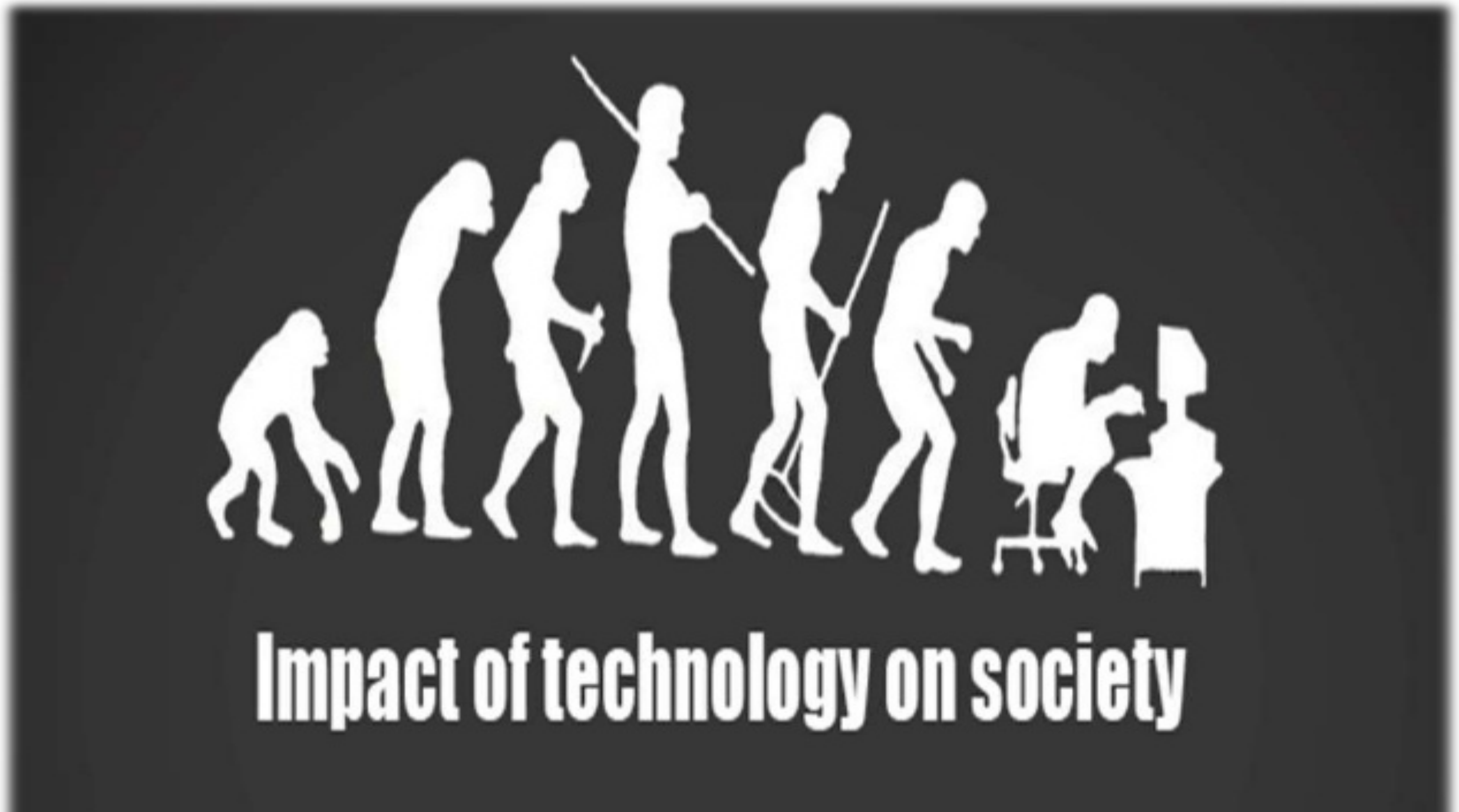
We have a responsibility

ML going out of labs, into society

- ML is not just a research question anymore
- ML-based products are being designed and deployed

  → new responsibility for AI scientists and engineers

  → wisdom race, as power of technology increases



THANK YOU!

Mila

Université de Montréal

McGill

Québec

CIFAR