Recurrent Independent Mechanisms

Anirudh Goyal
Mila, University of Montreal

Collaborators: Alex Lamb, Jordan Hoffman, Shagun Sodhani, Sergey Levine, Bernhard Scholkopf, Yoshua Bengio

May 4, 2021

Missing from Current Dynamical Systems ?

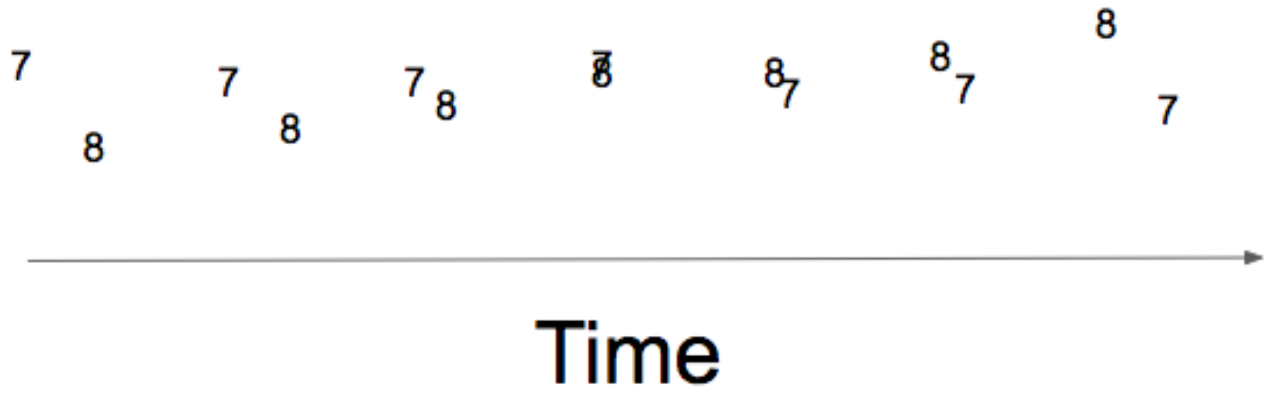
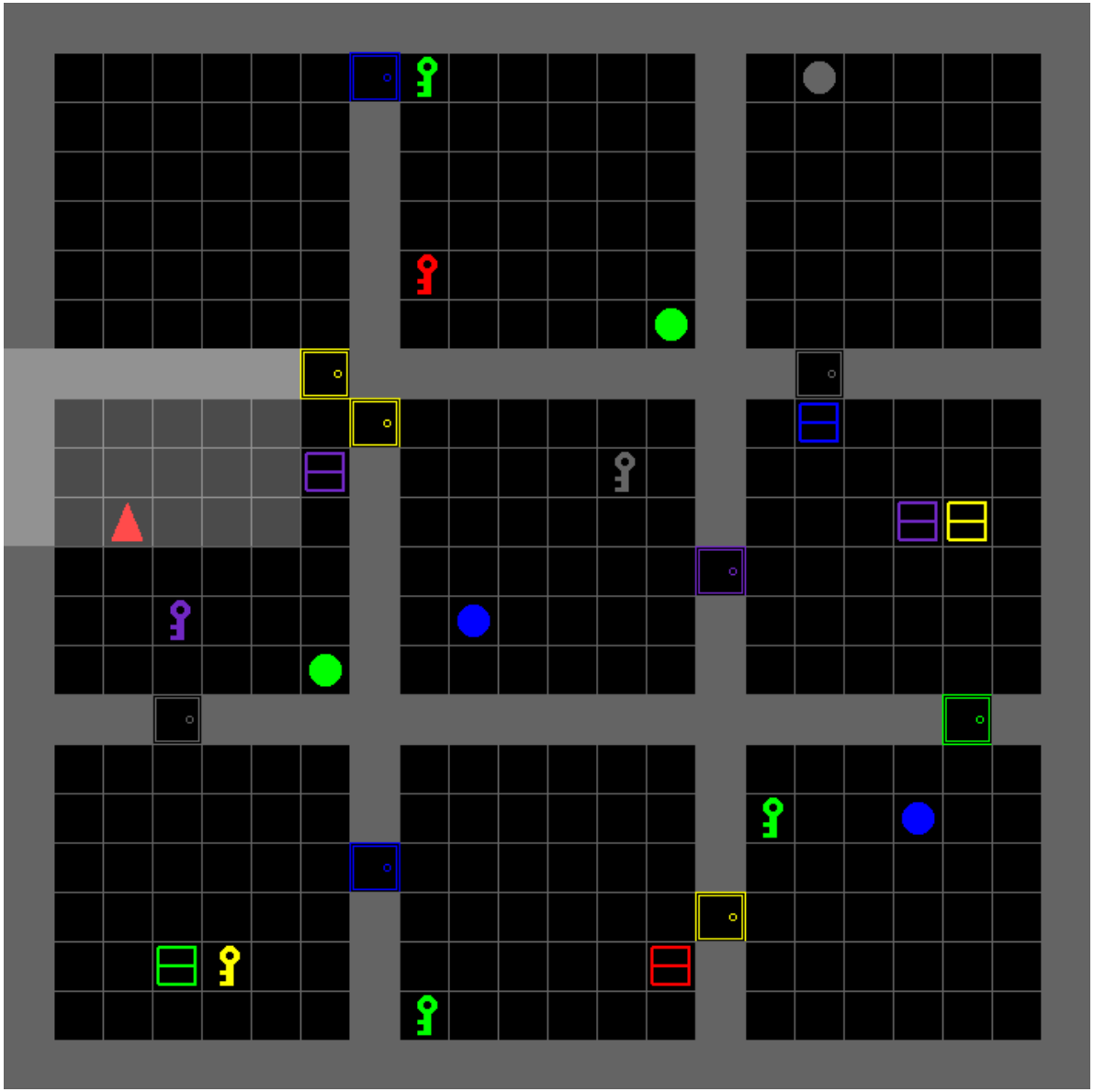


Figure 1

- Normal neural networks generally put all of these computational streams together.
- Hard for the model to share information in a dynamic way (maybe inclined towards either always sharing information or doing it in a fixed way).

Data Generating Distribution



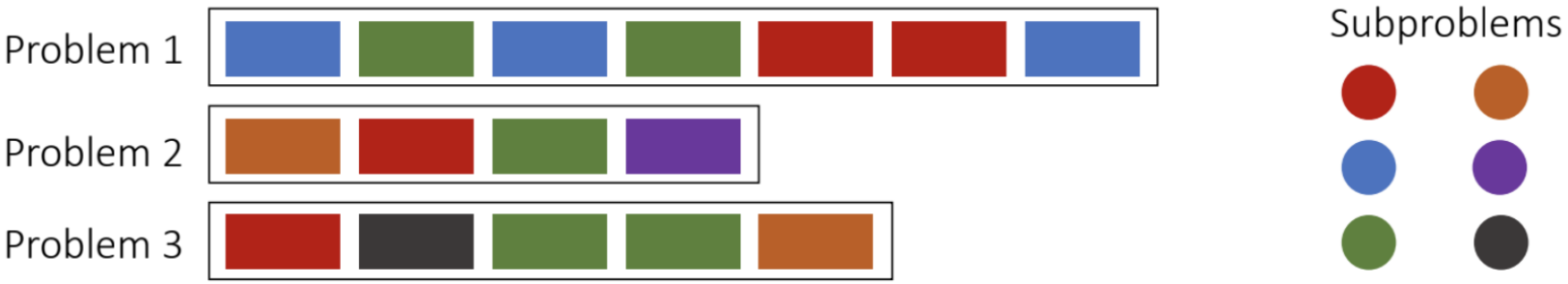
Task distribution - Diverse, growing, compositional.

Need for Out of Distribution Generalization

Learning Agents face non-stationarities. Changes in distribution due to:

- Their actions.
- Actions of other agents.

Systematic Generalization/Compositional Generalization



- Dynamically recombine existing concepts.
- Even when new combinations have 0 probability under training distribution.

Question: How to learn and then re-compose reusable computations ?

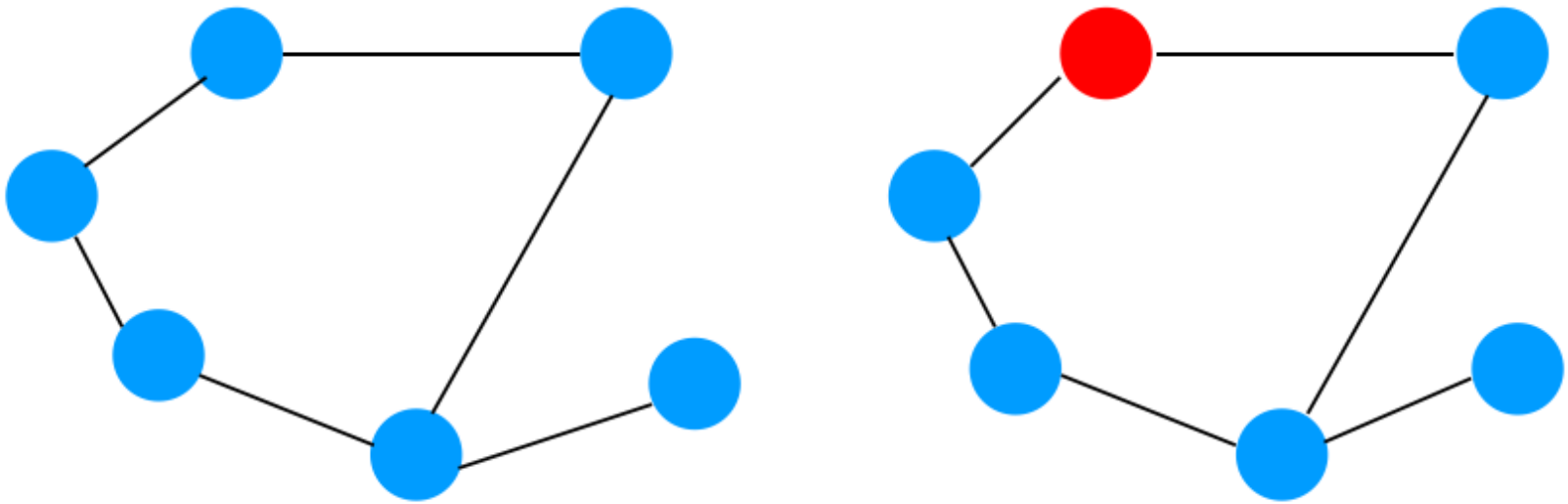
Independent Mechanisms

$$p(X_1, \ldots, X_n) = \prod_{i=1}^{m} p(X_i \mid PA_i). \tag{1}$$

- Interventions are feasible if the mechanisms (i.e., causal conditions) are independent. Changing one conditional $p(X_i \mid PA_i)$ does not change other $p(X_j \mid PA_j)$ for $(i \neq j)$, they remain invariant.
- if all PA_i are empty, the factors are statistically independent. (disentanglement as independence).

Separating Knowledge in Small Re-Usable Pieces

- Mechanisms which can be used combinatorially.
- Mechanisms which are stable versus non-stationary subject to non-stationarity.



Learning Representations, where change can be localized.

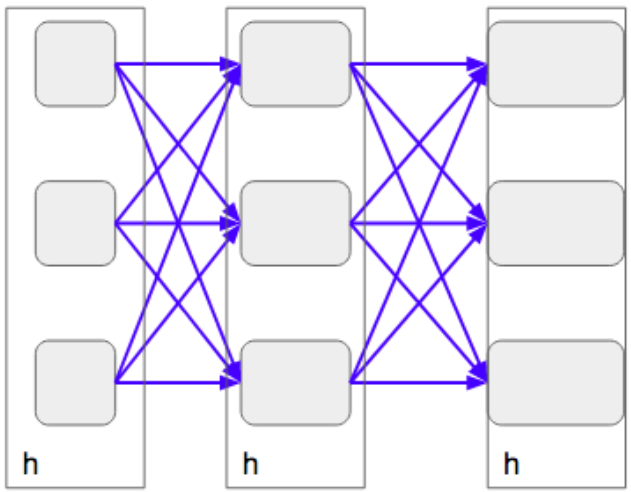
Why could it help to have separated dynamics?

- Reuse, Recompose, Repurpose.
- Learning independent mechanisms, buys invariance.
- Invariance buys extrapolation.
- Better transfer and continual learning (if a new task shares some dynamics but not others)
- Makes it easy for the model to keep two processes separate, which could also make long-term information storage much easier (no interference).

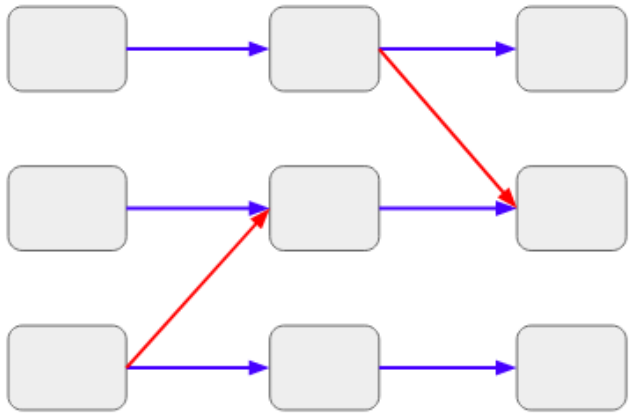
What form of knowledge representation would support these goals ?: What kind of assumptions on the joint distribution of high level variables ?

Ensemble of Sparingly Interacting Modules

Intuitive Figure



Normal RNN



RIMs

Desiderata for Learning Independent Mechanisms

- **Seeking relevant information:** Each Module only attends to relevant information.
- **Capacity:** Limiting the capacity of each module.
- **Diversity:** Diversity among the different modules i.e. capture different aspects of the world.
- **Coherence between different mechanisms:** Need to think about "coherence" between different mechanisms.

Multiheaded Key Value Attention: Manipulate sets of objects

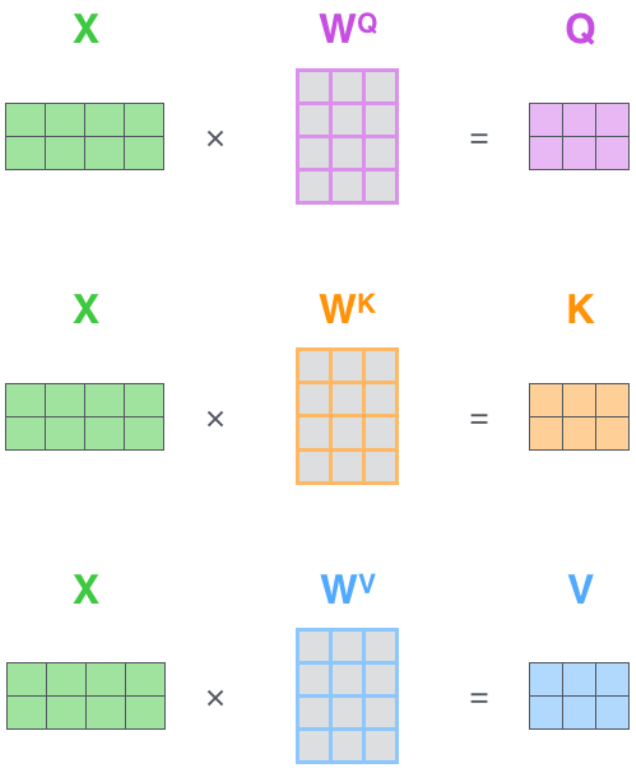


Figure 2: Linear Projection

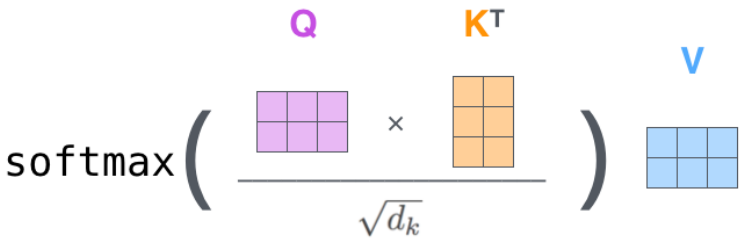
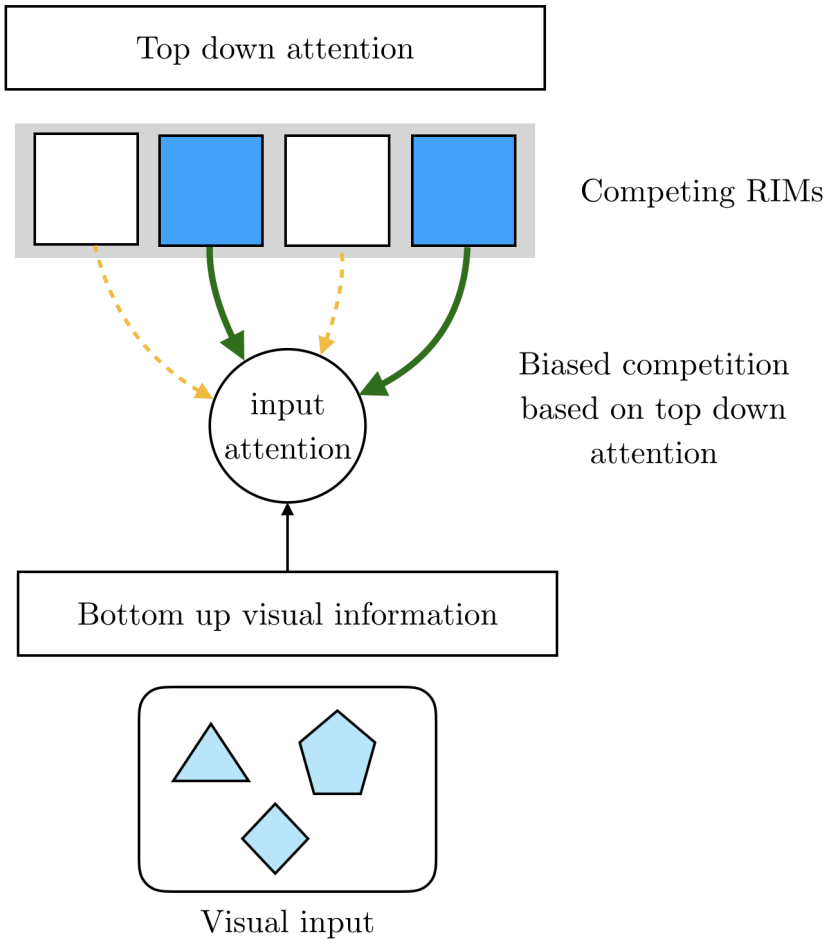


Figure 3: Softmax Calculation

Recurrent Independent Mechanisms

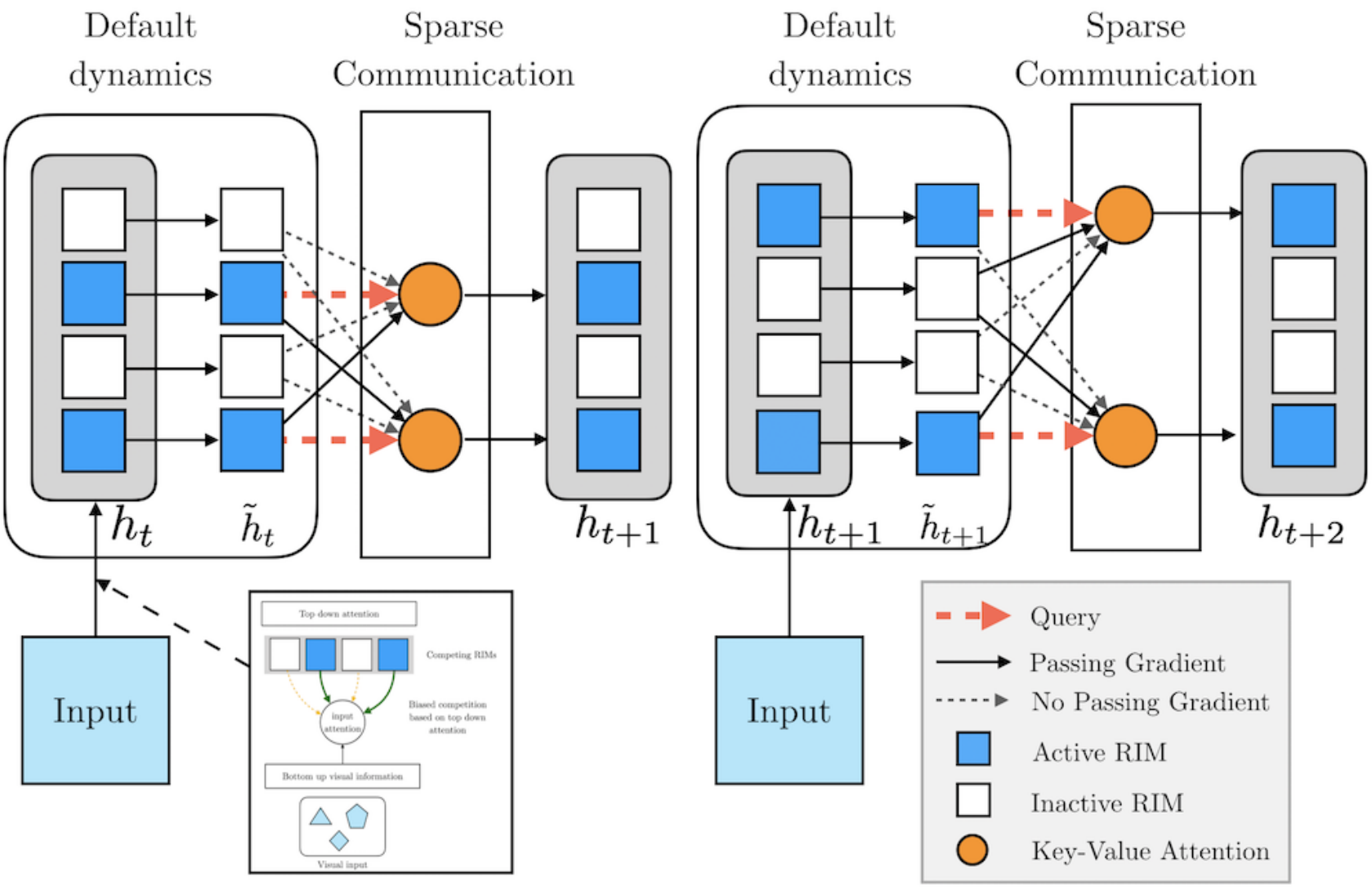
- Data Dependent Activation of Mechanisms
  - Top Down Activation of Mechanisms
- Coherence between representations of different mechanisms.
  - Active Mechanisms communicate with other mechanisms.
- Inactive Mechanisms follow the *default* dynamics.

Top Down Competition: Attentive Information Seeking



- The top-down filter not only enhances the target but also suppresses other stimuli, including the distractors.

Recurrent Independent Mechanisms



Experiments: Diagnostic Interventions

- **Robustness to Distractors:** Ignore Irrelevant Information.
- **Event Based Representations:** Specialization over temporal patterns.
- **Object Based Representations:** Specialize over objects and generalize over them.

Specialization over temporal patters: MNIST Resolution

- Motivated by the intuition that RIMs activate on relevant part of the sequence.
- Generalization to images of resolutions different from those seen during training.
- On CIFAR10, a particular RIM learns to attend to the foreground, and ignore the background.

Sequential MNIST	k_T	k_A	h_{size}	16 x 16 Accuracy	19 x 19 Accuracy	24 x 24 Accuracy
RIMs	6	6	600	85.5	56.2	30.9
	6	5	600	88.3	43.1	22.1
	6	4	600	**90.0**	**73.4**	**38.1**
LSTM	-	-	300	86.8	42.3	25.2
	-	-	600	84.5	52.2	21.9
EntNet	-	-	-	89.2	52.4	23.5
RMC	-	-	-	89.58	54.23	27.75
DNC	-	-	-	87.2	44.1	19.8
Transformers	-	-	-	**91.2**	51.6	22.9

Figure 4: MNIST Resolution Results

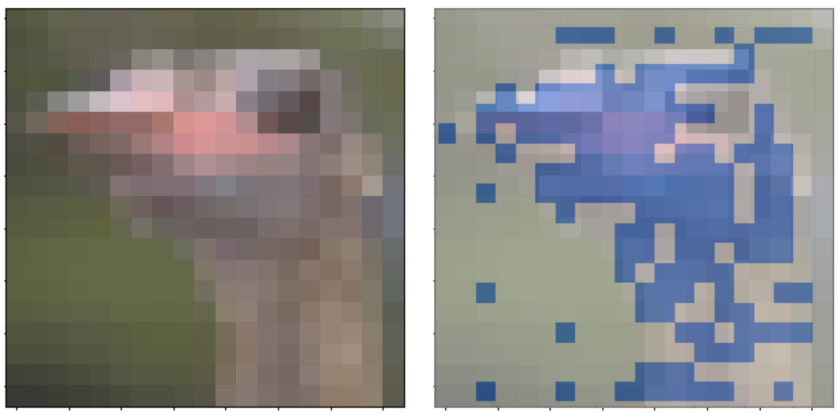


Figure 5: Vizualization for CIFAR

Specialization over objects and generalize over them

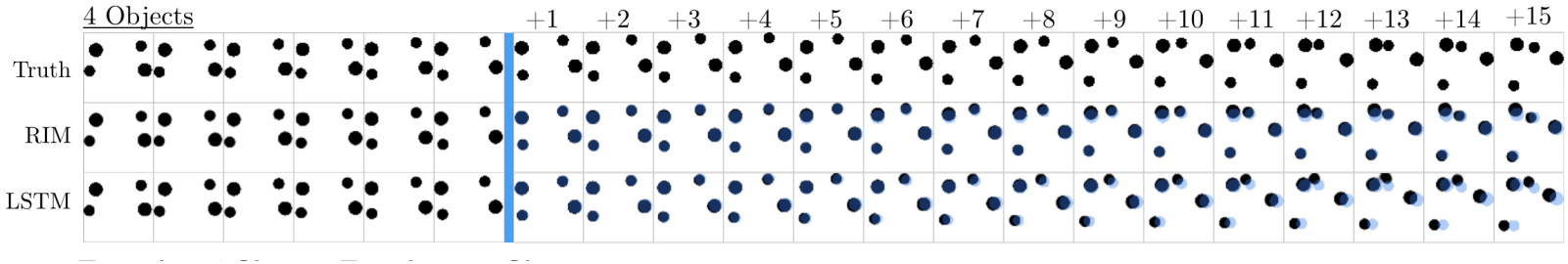


Figure 6: Predicting Movement of Bouncing Balls. The first 15 frames of ground truth are given (last 6 of those shown) and then the system is rolled out for the next 15 time steps. We find that RIMs perform better than the LSTMs (predictions are in black, ground truth in blue). Notice the blurring of LSTM predictions.

RIMs improve Generalization in Complex Atari Environments

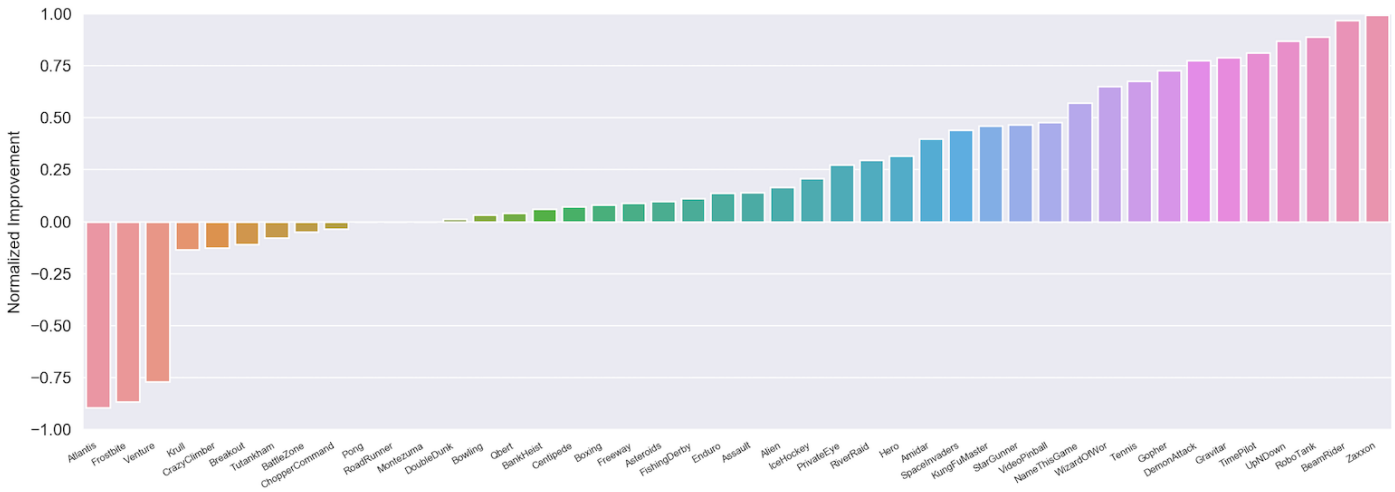


Figure 7: RIMs-PPO relative score improvement over LSTM-PPO baseline across all Atari games averaged over 3 trials per game. In both cases, PPO was used with the exact same settings, and the only change is the choice of recurrent architecture.

Summary: Useful Properties of RIMs

- Processing sets of elements rather than fixed-size vectors as 'states' of the computation.
- Computation is sparse and modular.
- Computation is dynamic rather than static.
- Inputs and outputs of these modules are sets of objects.
- Inputs and outputs are similar to variables in logic and arguments in programming, in that the attention-driven control flow selects which actual objects will be fed as input to activated modules.