NeurIPS 2021

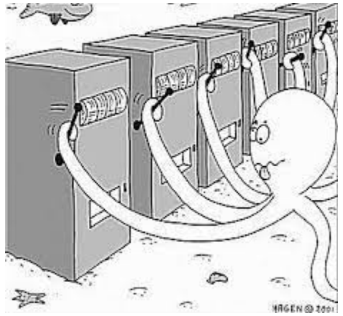
Dynamic Bottleneck for Robust Self-Supervised Exploration

Chenjia Bai[1] Lingxiao Wang[2] Lei Han[3] Animesh Garg[4] Jianye Hao[5] Peng Liu[1] Zhaoran Wang[2]

[1]Harbin Institute of Technology
[2]Northwestern University
[3]Tencent Robotics X
[4]University of Toronto
[5]Tianjin University

Motivation

- The tradeoff between exploration and exploitation has long been a major challenge in Reinforcement Learning (RL).
- Self-supervised exploration: extrinsic rewards are entirely unavailable.
- Previous methods becomes unstable when the states are noisy, e.g., containing dynamics-irrelevant information.

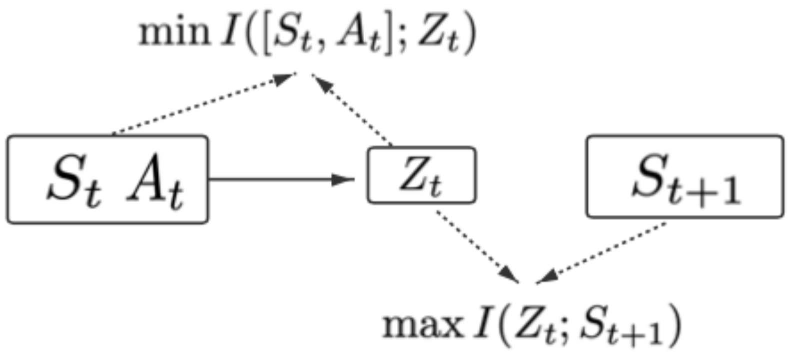


Bandit Problem

NEURAL INFORMATION PROCESSING SYSTEMS

Method

- We propose a Dynamic Bottleneck (DB) model, which generates a dynamics-relevant representation Z_t of the current state-action pair (S_t, A_t)through the Information-Bottleneck (IB) principle.
- DB acquires dynamics-relevant information and discards dynamics-irrelevant features simultaneously.
- We maximize the MI term $I(Z_t; S_{t+1})$, and minimize the MI term $I([S_t, A_t]; Z_t)$



Method

- Maximize the MI term $I(Z_t; S_{t+1})$
  - Predictive objective with Momentum encoder

$$I_{\text{pred}} \triangleq \mathbb{E}_{p(z_t, s_{t+1})}[\log q(s_{t+1} | z_t; \psi)].$$

  - Contrastive Objective

$$I(Z_t; S_{t+1}) \geq \mathbb{E}_{p(z_t, s_{t+1})} \mathbb{E}_{S^-} \left[\log \frac{\exp(h(z_t, s_{t+1}))}{\sum_{s_j \in S^- \cup s_{t+1}} \exp(h(z_t, s_j))} \right] \triangleq I_{\text{nce}}.$$

bilinear function as the score function

$$h(z_t, s_{t+1}) = f_o^P(\bar{q}(z_t; \psi))^\top \mathcal{W} f_m^P(s_{t+1}),$$

Method

- Minimize the MI term $I([S_t, A_t]; Z_t)$
  - minimizing a tractable upper bound

$$
\begin{aligned}
I([S_t, A_t]; Z_t) &= \mathbb{E}_{p(s_t,a_t)}\left[\frac{p(z_t|s_t,a_t)}{p(z_t)}\right] = \mathbb{E}_{p(s_t,a_t)}\left[\frac{p(z_t|s_t,a_t)}{q(z_t)}\right] - D_{\mathrm{KL}}\left[p(z_t)\|q(z_t)\right] \\
&\leq \mathbb{E}_{p(s_t,a_t)}\left[D_{\mathrm{KL}}[p(z_t|s_t,a_t)\|q(z_t)]\right] \triangleq I_{\mathrm{upper}},
\end{aligned}
$$

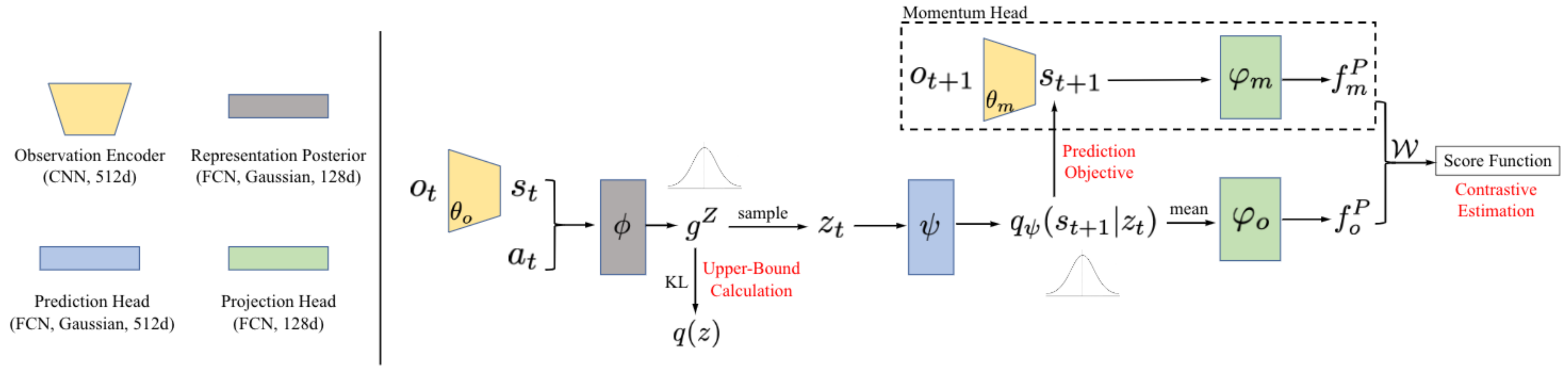
  - a standard spherical Gaussian distribution $q(z) = N(0, I)$ as the approximation

Method

- The final loss for training the DB model

$$\min_{\theta_o,\phi,\psi,\varphi_o,\mathcal{W}} \mathcal{L}_{\mathrm{DB}} = \alpha_1 I_{\mathrm{upper}} - \alpha_2 I_{\mathrm{pred}} - \alpha_3 I_{\mathrm{nce}},$$

- Architecture
  - observation encoder , representation posterior, prediction head, projection heads



Method

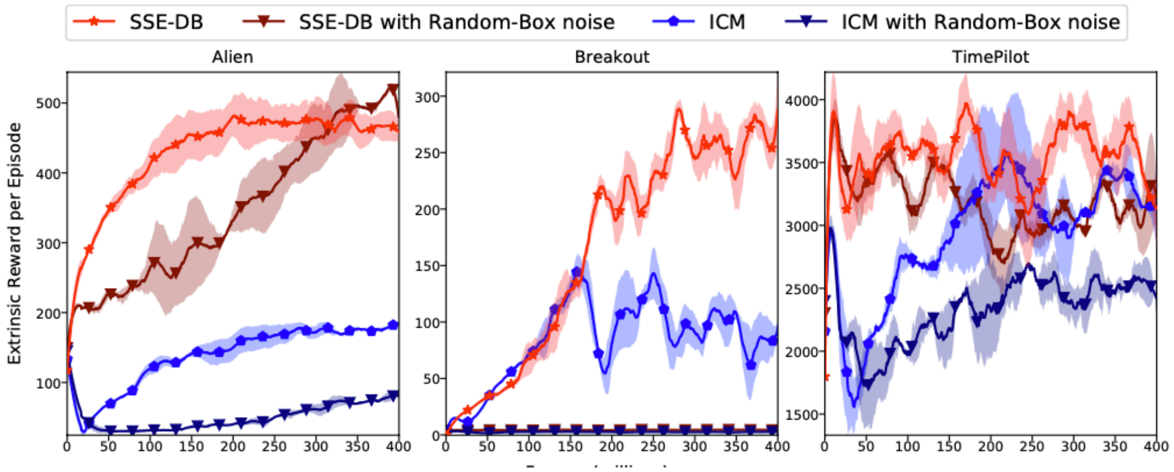
- Exploration based on DB
  - The DB-bonus
$$r^{\mathrm{db}}(s_t,a_t) \triangleq I\big(\Theta;(s_t,a_t,S_{t+1})|\mathcal{D}_m\big)^{1/2}$$
$$= \Big[\mathcal{H}\big((s_t,a_t,S_{t+1})|\mathcal{D}_m\big) - \mathcal{H}\big((s_t,a_t,S_{t+1})|\Theta,\mathcal{D}_m\big)\Big]^{1/2}.$$
  - Connection to UCB-bonus in linear MDPs and visiting count in tabular MDP
$$\beta_0/\sqrt{2}\cdot r_t^{\mathrm{ucb}} \le I(W_t;(s_t,a_t,S_{t+1})|\mathcal{D}_m)^{1/2} \le \beta_0\cdot r_t^{\mathrm{ucb}},$$
$$r^{\mathrm{db}}(s_t,a_t) \approx \frac{\sqrt{|\mathcal{S}|/2}}{\sqrt{N_{s_t,a_t}+\lambda}} = \beta_0\cdot r^{\mathrm{count}}(s_t,a_t)$$
  - Empirical estimation
$$r^{\mathrm{db}}(s_t,a_t) \ge \Big[\mathcal{H}\big(g(s_t,a_t,S_{t+1})|\mathcal{D}_m\big) - \mathcal{H}\big(g(s_t,a_t,S_{t+1})|\Theta,\mathcal{D}_m\big)\Big]^{1/2} \triangleq r_l^{\mathrm{db}}(s_t,a_t),$$
$$r_l^{\mathrm{db}}(s_t,a_t) = \Big[\mathcal{H}\big(g^{\mathrm{margin}}\big) - \mathcal{H}\big(g^Z(s_t,a_t;\phi)\big)\Big]^{1/2} = \mathbb{E}_{\Theta}D_{\mathrm{KL}}\big[g^Z(z_t|s_t,a_t;\phi)\|g^{\mathrm{margin}}\big]^{1/2}$$

NEURAL INFORMATION PROCESSING SYSTEMS

Experiment

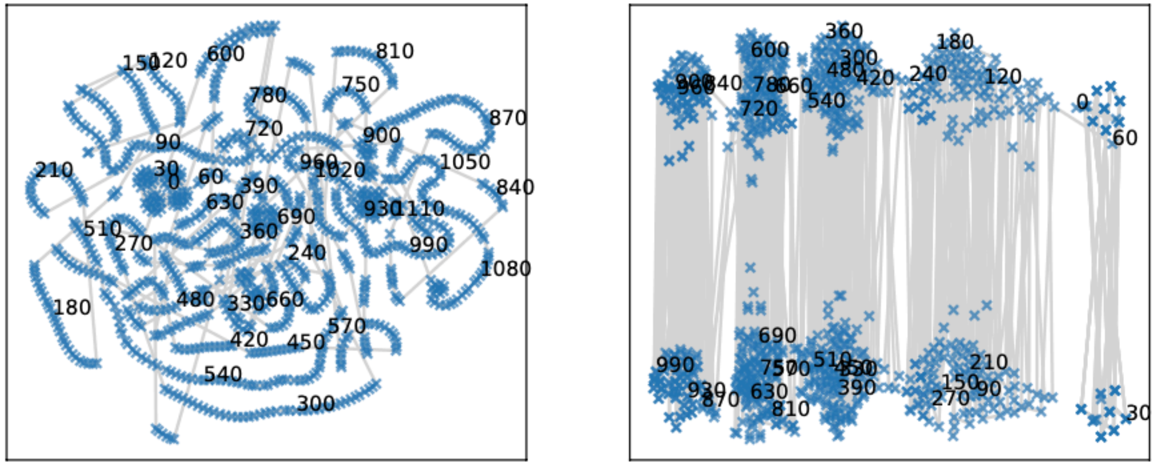
- Analyze the robustness of SSE-DB to observation noises
- Distractors for the observations
  - Random Box noise
  - Pixel-level noise

Normal Observation | Random-Box Noise | Pixel Noise



Experiment

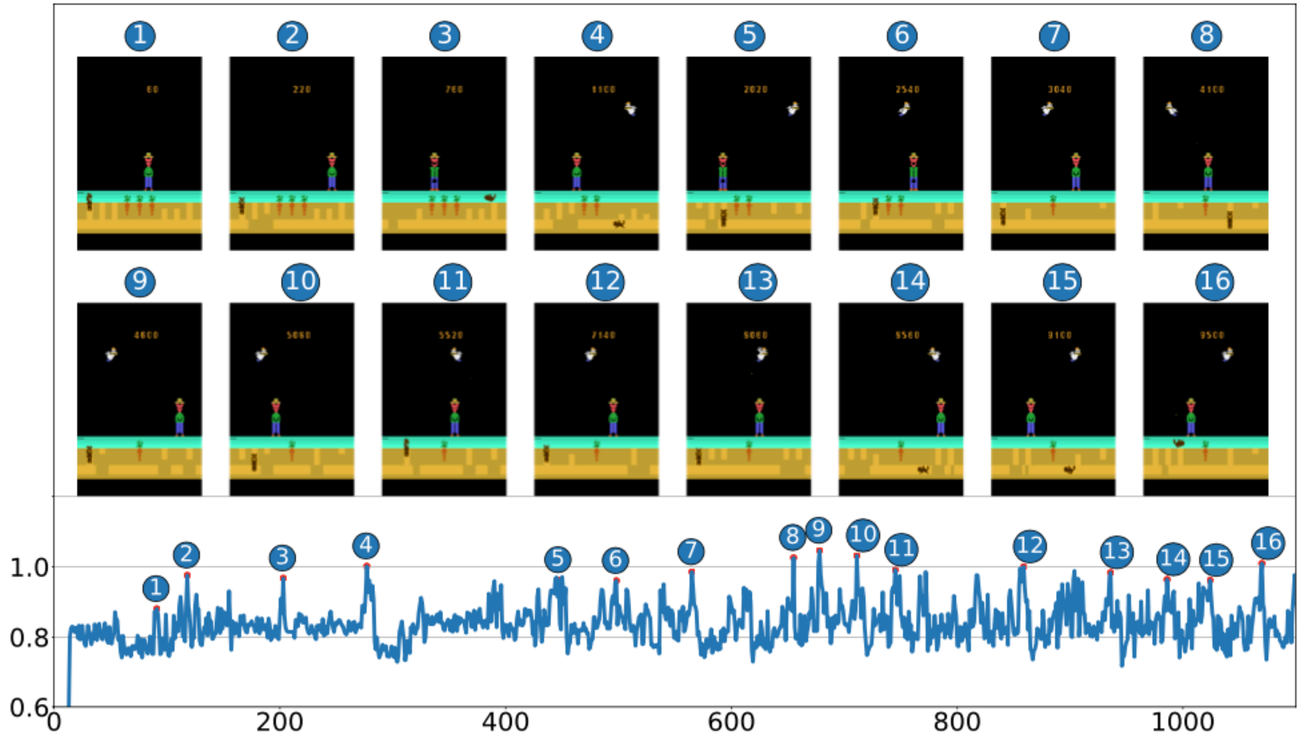
- Representations align temporally-consecutive movements
- Each segment of a curve corresponds to a semantic component



(a) DB representations with Random-Box (b) ICM representations with Random-Box

Experiment

- DB-bonus encourages the agent to explore the informative transitions
- Gopher
  - DB-bonus correspond to scenarios that the gopher makes a hole to the surface



Thanks for your attention!