Robust Task Representations for Offline Meta-Reinforcement Learning via Contrastive Learning

Haoqi Yuan, Zongqing Lu
School of Computer Science, Peking University

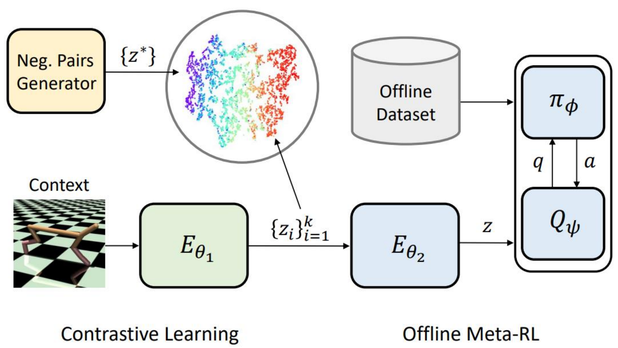
Introduction

Offline meta-reinforcement learning (OMRL) is a promising RL paradigm that learns from offline multi-task experience to adapt to new tasks. In OMRL, the difference of the context collection policy between training and test makes **task representation** and adaptation unstable, which remains an understudied problem in the **fully offline** setting.

In **our work**:

- We propose CORRO for learning robust task representations with fully offline datasets. It extracts task information from the distribution of transitions, which is jointly determined by the behavior policy and task.
- We introduce bi-level task encoder, contrastive learning and methods for negative pairs generation.
- We empirically show the advantages of CORRO over prior methods, especially on the generalization to out-of-distribution behavior policies.

Framework



CORRO is a context-based OMRL framework. It trains an agent using pre-collected offline multi-task datasets.

- Transition encoder: extracts latent representations for all the transition tuples in the context trajectory.
- Aggregator: gathers all the latent codes into a task representation.
- Policy and Q-function: they are conditioned on the task representation to make decisions. They are trained with offline RL method, together with the aggregator.

We propose a contrastive task representation learning method to train the transition encoder, with several methods to generate negative pairs.

Contrastive Task Representation Learning

Main idea: we formalize the learning objective for the transition encoder to be maximizing mutual information between the representation and the task:

$$\max I(z;M) = \mathbb{E}_{z,M}\left[\log \frac{p(M|z)}{p(M)}\right]$$

The transition encoder aims at maximally reducing task uncertainty while minimally preserving task-irrelevant information.

Solution: the mutual information is intractable. We derive a lower bound:

$$I(z;M) - \log(N) \geq \mathbb{E}_{\mathcal{M},x,z}\left[\log\left(\frac{h(x,z)}{\sum_{M^*\in\mathcal{M}} h(x^*,z)}\right)\right]$$

$\mathcal{M}$ is the training tasks set. $h(x,z) = \frac{P(Z|X)}{P(Z)}$. $x = (s,a,r,s')$ is a transition tuple in the first task, $x^* = (s,a,r^*,s^{*\prime})$ is a transition tuple in task M^*, where (s,a) follows some distribution.

We introduce **contrastive learning** to approximately optimize the objective:

$$\max_{\theta_1} \sum_{\substack{M_i\in\mathcal{M}\\ x,x'\in X_i}} \left[\log\left(\frac{\exp(S(z,z'))}{\sum_{M^*\in\mathcal{M}}\exp(S(z,z^*))}\right)\right]$$

X_i is a task dataset. S is a score function. z, z' are representations of x, x'. z^* is the representation of a sample x^* in task M^*. x^* and x share the same (s,a).

- Positive pair: to maximize the score of (z,z'), the encoder should extract similar features for samples in the same task.
- Negative pairs: to minimize the score of (z,z^*), the encoder should capture the essential variance of rewards and state transitions between different tasks.

Negative Pairs Generation

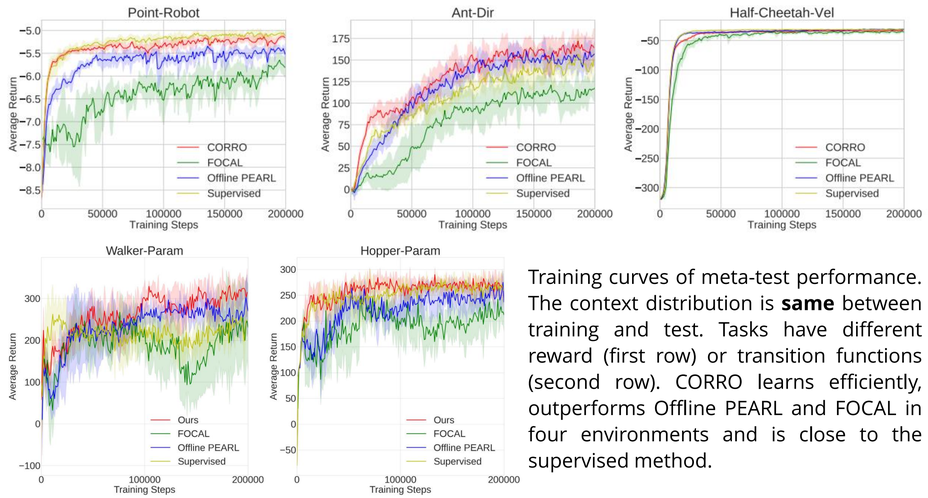
Problem: without access to the reward and transition functions in all the tasks, how to generate the negative samples z^*?

Solution: we propose methods to approximate the negative samples distribution and generate diverse samples.

- **Generative modeling:** use the offline data distribution $P_{\cup_i^N X_i}(r,s'|s,a)$ to approximate the target distribution. Train a conditional VAE to generate diverse x^*.
- **Reward randomization:** if tasks only differ in reward functions, we add a random perturbation to the reward to imitate the sample from other tasks. $r^* = r + \nu, \nu \sim P(\nu)$.

When the state-action distributions in all the tasks are similar, generative modeling can approximate the target distribution. When the overlap of state-action pairs between tasks is small, CVAE may collapse. In this case, reward randomization can still generate diverse samples to support contrastive learning.

Experiments



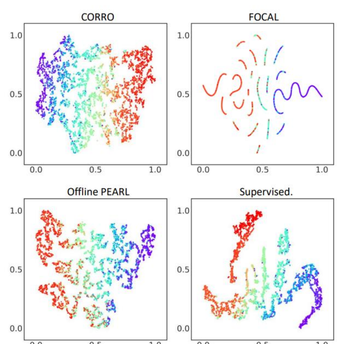
Training curves of meta-test performance. The context distribution is **same** between training and test. Tasks have different reward (first row) or transition functions (second row). CORRO learns efficiently, outperforms Offline PEARL and FOCAL in four environments and is close to the supervised method.

Environment	Supervised.		Offline PEARL		FOCAL		CORRO	
	IID	*OOD*	*IID*	*OOD*	*IID*	*OOD*	*IID*	*OOD*
Point-Robot	**-4.89**±0.10	**-5.84**±0.14	-5.4±0.17	-6.74±0.19	-6.06±0.42	-7.34±0.20	**-5.19**±0.05	**-6.39**±0.05
Ant-Dir	136±17.6	131.7±11.4	155.4±24.4	141.5±11.3	109.8±12.8	53.5±16.4	**156.8**±35.2	**154.7**±25.8
Half-Cheetah-Vel	**-31.6**±0.7	**-32.1**±0.9	**-31.2**±0.5	-242.7±6.0	-38.0±4.0	-204.1±9.5	-33.7±1.1	**-89.7**±7.4
Walker-Param	232.7±29.2	221.2±43.4	259.1±48.2	254.7±35.8	225.4±56.4	193.3±151.5	**301.5**±37.9	**284.0**±19.3
Hopper-Param	**269.2**±20.3	251.9±28.8	244.0±18.5	236.6±18.5	195.6±62.3	199.7±51.9	**267.6**±25.6	**268.0**±13.8

To measure the robustness of task representations, we propose OOD test: meta-test with **out-of-distribution** context collection policies. CORRO outperforms baselines and has smaller gap between IID and OOD.

Method	Contrastive Loss	*IID* Return	*OOD* Return
Generative	0.07	**-33.7**±1.1	-89.7±7.4
Randomize	0.83	-34.3±1.5	**-84.5**±1.3
Relabeling	0.04	-40.8±1.5	-245.3±12.9
None	1.20	-34.1±2.4	-97.6±3.1
Generative	2.83	-9.41±0.42	-9.42±0.42
Randomize	0.54	**-5.19**±0.05	**-6.39**±0.05
Relabeling	0.04	-9.22±0.24	-9.27±0.22
None	1.46	-5.24±0.27	-6.52±0.08

Choice of negative pairs generation methods in Half-Cheetah-Vel and Point-Robot.



Latent space visualization in Half-Cheetah-Vel.

Acknowledgements: We would like to thank the anonymous reviewers for their useful comments to improve our work. This work is supported in part by NSF China under grant 61872009.