Towards an Information Theoretic Framework of Context-Based Offline Meta-Reinforcement Learning

Lanqing Li[1,2*], **Hai Zhang**[3*], **Xinyu Zhang**[4], **Shatong Zhu**[3], **Yang Yu**[2], **Junqiao Zhao**[3†], **Pheng-Ann Heng**[2]
[1] Zhejiang Lab
[2] The Chinese University of Hong Kong
[3] Tongji University
[4] Stony Brook University

lanqingli1993@gmail.com, {zhanghai12138, zhushatong, zhaojunqiao}@tongji.edu.cn, zhang146@cs.stonybrook.edu, {yangyu, pheng}@cse.cuhk.edu.hk

Abstract

As a marriage between offline RL and meta-RL, the advent of offline meta-reinforcement learning (OMRL) has shown great promise in enabling RL agents to multi-task and quickly adapt while acquiring knowledge safely. Among which, context-based OMRL (COMRL) as a popular paradigm, aims to learn a universal policy conditioned on effective task representations. In this work, by examining several key milestones in the field of COMRL, we propose to integrate these seemingly independent methodologies into a unified framework. Most importantly, we show that the pre-existing COMRL algorithms are essentially optimizing the same mutual information objective between the task variable $\boldsymbol{M}$ and its latent representation $\boldsymbol{Z}$ by implementing various approximate bounds. Such theoretical insight offers ample design freedom for novel algorithms. As demonstrations, we propose a supervised and a self-supervised implementation of $I(\boldsymbol{Z}; \boldsymbol{M})$, and empirically show that the corresponding optimization algorithms exhibit remarkable generalization across a broad spectrum of RL benchmarks, context shift scenarios, data qualities and deep learning architectures. This work lays the information theoretic foundation for COMRL methods, leading to a better understanding of task representation learning in the context of reinforcement learning [1].

1 Introduction

The ability to swiftly learn and generalize to new tasks is a hallmark of human intelligence. In pursuit of this high-level artificial intelligence (AI), the paradigm of meta-reinforcement learning (RL) proposes to train AI agents in a trial-and-error manner by interacting with multiple external environments. In order to quickly adapt to the unknown, the agents need to integrate prior knowledge with minimal experience (namely the context) collected from the new tasks or environments, without over-fitting to the new data. This meta-RL mechanism has been adopted in many applications such as games [1, 2], robotics [3, 4] and drug discovery [5].

However, for data collection, classical meta-RL usually requires enormous online explorations of the environments [6, 7], which is impractical in many safety-critical scenarios such as healthcare [8], autonomous driving [9] and robotic manipulation [10]. As a remedy, offline RL [11] enables agents

*Equal contribution. This work was primarily done when Hai Zhang worked as an intern at Zhejiang Lab.
†Corresponding Author
[1]The open-source code is at https://github.com/betray12138/UNICORN.git.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

to learn from logged experience only, therefore circumventing risky or costly online interactions. Recently, offline meta-RL (OMRL) [12–14] has emerged as a novel paradigm to significantly extend the applicable range of RL by "killing two birds in one stone": it builds powerful agents that can quickly learn and adapt by meta-learning, while leveraging offline RL mechanism to ensure a safe and efficient optimization procedure. In the context of classical supervised or self-supervised learning, which is de facto offline, OMRL is reminiscent of the multi-task learning [15], meta-training [6] and fine-tuning [16–18] of pre-trained large models. We envision it as a cornerstone of RL foundation models [19–21] in the future.

Along the line of OMRL research, context-based offline meta-reinforcement learning (COMRL) is a popular paradigm that seeks optimal meta-policy conditioning on the context of Markov Decision Processes (MDPs). Intuitively, the crux of COMRL lies in learning effective task representations, hence enabling the agent to react optimally and adaptively in various contexts. To this end, one of the earliest COMRL algorithms FOCAL [12] proposes to capture the structure of task representations by distance metric learning. From a geometric perspective, it essentially performs clustering by repelling latent embeddings of different tasks while pulling together those from the same task, therefore ensuring consistent and distinguishable task representations.

Despite its effectiveness, FOCAL is reported to be vulnerable to context shifts [22], i.e., when testing on out-of-distribution (OOD) data (Fig. 1). Such problems are particularly challenging for OMRL, since any context shift incurred at test time can not be rectified in the fully offline setting, which may result in severely degraded generalization performance [22, 23]. To alleviate the problem, follow-up works such as CORRO [24] reformulates the task representation learning of COMRL as maximizing the mutual information $I(\boldsymbol{Z};\boldsymbol{M})$ between the task variable $\boldsymbol{M}$ and its latent $\boldsymbol{Z}$. It then approximates $I(\boldsymbol{Z};\boldsymbol{M})$ by an InfoNCE [25] contrastive loss, where the positive and negative pairs are conditioned on the state-action tuples $(\boldsymbol{s},\boldsymbol{a})$. Inspired by CORRO, a recently proposed method CSRO [23] introduces an additional mutual information term between $\boldsymbol{Z}$ and $(\boldsymbol{s},\boldsymbol{a})$. By explicitly minimizing it along with the FOCAL objective, CSRO is demonstrated to achieve the state-of-the-art (SOTA) generalization performance on various MuJoCo [26] benchmarks.

Contributions In this paper, following the recent development and storyline of COMRL, we present a Unified Information Theoretic Framework of Context-Based Offline Meta-Reinforcement Learning (UNICORN) encompassing pre-existing methods. We first prove that the objectives of FOCAL, CORRO and CSRO operate as the upper bound, lower bound of $I(\boldsymbol{Z};\boldsymbol{M})$ and their linear interpolation respectively, which provides a nontrivial theoretical unification of these methods.

Second, by the aforementioned insight and an analysis of the COMRL causal structures, we shed light on how CORRO and CSRO improve context-shift robustness compared to their predecessors by trading off causal and spurious correlations between $\boldsymbol{Z}$ and input data $\boldsymbol{X}$.

Lastly, by examining eight related meta-RL methods (Table 1) concerning their objectives and implementations, we highlight the potential design choices of novel algorithms offered by our framework. As examples, we investigate two instantiated algorithms, one supervised and the other self-supervised, and demonstrate experimentally that they achieve competitive in-distribution and exceptional OOD generalization performance on a wide range of RL domains, OOD settings, data qualities and model architectures. Our framework provides a principled roadmap to novel COMRL algorithms by seeking better approximations/regularizations of $I(\boldsymbol{Z};\boldsymbol{M})$, as well as new implementations to further combat context shift.

2 Method

2.1 Preliminaries, Problem Statement and Related Work

We consider MDP modeled as $M = (\mathcal{S}, \mathcal{A}, T, R, \rho_0, \gamma, H)$ with state space $\mathcal{S}$, action space $\mathcal{A}$, transition function $T(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$, bounded reward function $R(\boldsymbol{s},\boldsymbol{a})$, initial state distribution $\rho_0(\boldsymbol{s})$, discount factor $\gamma \in (0,1)$ and H the horizon. The goal is to find a policy $\pi : \mathcal{S} \to \mathcal{A}$ to maximize the expected return. Starting from the initial state, for each time step, the agent performs an action sampled from π, then the environment updates the state with T and returns a reward with R. We denote the marginal state distribution at time t as $\mu_\pi^t(\boldsymbol{s})$. The V-function and Q-function are given by

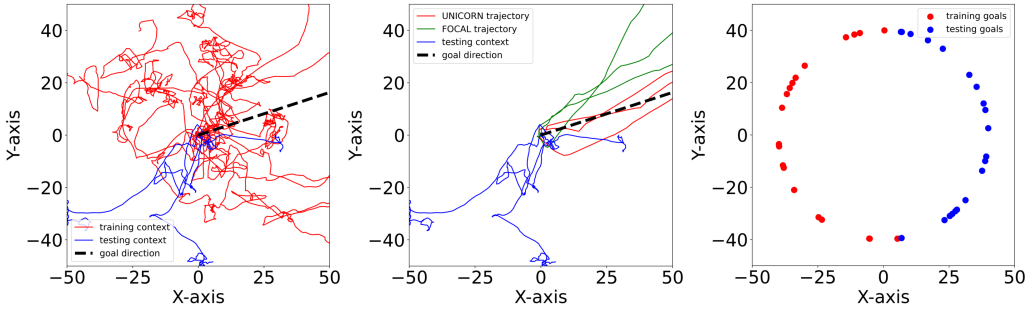

Figure 1: **Context shift of COMRL in Ant-Dir**. **Left**: Given a task M^i specified by a goal direction (dashed line), the RL agent is trained on data generated by a variety of behavior policies trained on the *same* task M^i (red). At test time, however, the context might be collected by behavior policies trained on *different* tasks $\{M^j\}$ (blue), causing a context shift of OOD behavior policies (Section 3.3). **Middle**: Against OOD context, UNICORN (red) is more robust than baselines such as FOCAL (green) in terms of navigating the Ant robot towards the right direction. **Right**: Besides behavior policy, the task distribution (e.g., goal positions in Ant) can induce significant context shift (Section 4.2), which is also a challenging scenario for COMRL models to generalize.

$$V_\pi(\boldsymbol{s}) = \sum_{t=0}^{H-1} \gamma^t \mathbb{E}_{\boldsymbol{s}_t \sim \mu_\pi^t(\boldsymbol{s}), \boldsymbol{a}_t \sim \pi}[R(\boldsymbol{s}_t, \boldsymbol{a}_t)], \tag{1}$$

$$Q_\pi(\boldsymbol{s}, \boldsymbol{a}) = R(\boldsymbol{s}, \boldsymbol{a}) + \gamma \mathbb{E}_{\boldsymbol{s}' \sim T(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})}[V_\pi(\boldsymbol{s}')]. \tag{2}$$

In this paper, we restrict to scenarios where tasks share the *same* state and action space and can only be distinguished via reward and transition functions. Each OMRL task is defined as an instance of MDP: $M^i = (\mathcal{S}, \mathcal{A}, T^i, R^i, \rho_0, \gamma, H) \in \mathcal{M}$, where $\mathcal{M}$ is the set of all possible MDPs to be considered. For each task index i, the offline dataset $X^i = \{(\boldsymbol{s}_j^i, \boldsymbol{a}_j^i, r_j^i, \boldsymbol{s'}_j^i)\}$ is collected by a behavior policy π_β^i. For meta-learning, given a trajectory segment $\boldsymbol{c}_{1:n}^i = \{(\boldsymbol{s}_j^i, \boldsymbol{a}_j^i, r_j^i, \boldsymbol{s'}_j^i)\}_{j=1}^n$ of length n as the context of M^i, COMRL employs a context encoder $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{c}_{1:n}^i)$ to obtain the latent representation $\boldsymbol{z}^i$ of task M^i. If $\boldsymbol{z}$ contains sufficient information about the task identity, COMRL can be treated as a special case of RL on partially-observed MDP [27], where $\boldsymbol{z}$ is interpreted as a faithful representation of the unobserved state. By conditioning on $\boldsymbol{z}$, the learning of universal policy $\pi_{\boldsymbol{\theta}}(\cdot|\boldsymbol{s}, \boldsymbol{z})$ and value function $V_\pi(\boldsymbol{s}, \boldsymbol{z})$ [28] become regular RL, and optimality can be attained by Bellman updates with theoretical guarantees. To this end, FOCAL [12] proposes to *decouple* COMRL into the upstream *task representation learning* and downstream offline RL. For the former, which can be treated independently, FOCAL employs metric learning to achieve effective clustering of task embeddings:

$$\mathcal{L}_{\text{FOCAL}} = \min_{\boldsymbol{\phi}} \mathbb{E}_{i,j} \left\{ \mathbb{1}\{i = j\} ||\boldsymbol{z}^i - \boldsymbol{z}^j||_2^2 + \mathbb{1}\{i \neq j\} \frac{\beta}{||\boldsymbol{z}^i - \boldsymbol{z}^j||_2^n + \epsilon} \right\}. \tag{3}$$

However, it is empirically shown that FOCAL struggles to generalize in the presence of context shift [22]. This problem has been formulated in several earliest studies of OMRL. Li et al. [29] observed that when there are large divergence of the state-action distributions among the offline datasets, due to shortcut learning [30], the task encoder may learn to ignore the causal information like rewards and *spuriously correlate* primarily state-action pairs to the task identity, leading to poor generalization performance. Dorfman et al. [14] concurrently identified a related problem which they termed MDP ambiguity in the context of Bayesian offline RL. An example is illustrated in Figure 1. The problem is further exacerbated in the fully offline setting, as the testing distribution is fixed and cannot be augmented by online exploration, allowing no theoretical guarantee for the context shift.

To mitigate context shift, a subsequent algorithm CORRO [24] proposes to optimize a lower bound of $I(\boldsymbol{Z}; \boldsymbol{M})$ in the form of an InfoNCE [25] contrastive loss

$$\mathcal{L}_{\text{CORRO}} = \min_{\boldsymbol{\phi}} \mathbb{E}_{\boldsymbol{x}, \boldsymbol{z}} \left[-\log \left(\frac{h(\boldsymbol{x}, \boldsymbol{z})}{\sum_{M^* \in \mathcal{M}} h(\boldsymbol{x}^*, \boldsymbol{z})} \right) \right], \tag{4}$$

where $\boldsymbol{x} = (\boldsymbol{s}, \boldsymbol{a}, r, \boldsymbol{s}')$, $\boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, $h(\boldsymbol{x}, \boldsymbol{z}) = \frac{P(\boldsymbol{z}|\boldsymbol{x})}{P(\boldsymbol{z})} \approx \frac{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}{P(\boldsymbol{z})}$, and $\boldsymbol{x}^* = (\boldsymbol{s}, \boldsymbol{a}, r^*, \boldsymbol{s}'^*)$ as a transition tuple generated for task $M^* \in \mathcal{M}$ conditioned on the same $(\boldsymbol{s}, \boldsymbol{a})$. To further combat the *spurious correlation* between the task representation and behavior-policy-induced state-actions, a recently proposed method CSRO [23] introduces a CLUB upper bound [31] of the mutual information between $\boldsymbol{z}$ and $(\boldsymbol{s}, \boldsymbol{a})$, to regularize the FOCAL objective:

$$\mathcal{L}_{\text{CSRO}} = \min_{\boldsymbol{\phi}} \left\{ \mathcal{L}_{\text{FOCAL}} + \lambda L_{\text{CLUB}} \right\}, \tag{5}$$

$$L_{\text{CLUB}} = \mathbb{E}_i \left[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{s}_i, \boldsymbol{a}_i) - \mathbb{E}_j \left[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}_j|\boldsymbol{s}_i, \boldsymbol{a}_i) \right] \right]. \tag{6}$$

In the next section, we will show how these algorithms are inherently connected and how their context-shift robustness gets improved incrementally.

2.2 A Unified Information Theoretic Framework

We start with a formal definition of *task representation learning* in COMRL:

Definition 2.1 (Task Representation Learning)**.** *Given an input context variable $\boldsymbol{X} \in \mathcal{X}$ and its associated task/MDP random variable $\boldsymbol{M} \in \mathcal{M}$, task representation learning in COMRL aims to find a sufficient statistics $\boldsymbol{Z}$ of $\boldsymbol{X}$ with respect $\boldsymbol{M}$.*

In pure statistical terms, Definition 2.1 implies that an ideal representation $\boldsymbol{Z}$ is a mapping of $\boldsymbol{X}$ that captures the mutual information $I(\boldsymbol{X}; \boldsymbol{M})$. We therefore define the following dependency structures in terms of directed graphical models:

Definition 2.2 (Causal Decomposition)**.** *The dependency graphs of COMRL is given by Fig. 2, where $\boldsymbol{X}_b$ and $\boldsymbol{X}_t$ are the behavior-related $(\boldsymbol{s}, \boldsymbol{a})$-component and task-related $(\boldsymbol{s}', r)$-component of the context $\boldsymbol{X}$, with $\boldsymbol{X} = (\boldsymbol{X}_t, \boldsymbol{X}_b)$.*

For the first graph, $\boldsymbol{M} \to \boldsymbol{X} \to \boldsymbol{Z}$ forms a Markov chain, which satisfies $I(\boldsymbol{Z}; \boldsymbol{M}|\boldsymbol{X}) = 0$. To interpret the second graph, given an MDP $M \sim \boldsymbol{M}$, the state-action component of $\boldsymbol{X}$ is primarily captured by the behavior policy π_β: $\boldsymbol{s} \sim \mu_{\pi_\beta}(\boldsymbol{s}), \boldsymbol{a} \sim \pi_\beta$. The only exception is when tasks differ in initial state distribution ρ_0 or transition dynamics T, in which case the state variable $\boldsymbol{S}$ also depends on $\boldsymbol{M}$. We therefore define it as the *behavior-related* component, which should be *weakly* causally related (dashed lines) to $\boldsymbol{M}$ and $\boldsymbol{Z}$. Moreover, when $\boldsymbol{X}_b$ is given, $\boldsymbol{X}_t$ is fully characterized by the transition function $T : (\boldsymbol{s}, \boldsymbol{a}) \to \boldsymbol{s}'$ and reward function $R : (\boldsymbol{s}, \boldsymbol{a}) \to r$ of M, which should be *strongly* causally related (solid lines) to $\boldsymbol{M}$ and $\boldsymbol{Z}$ and therefore be defined as the *task-related* component.

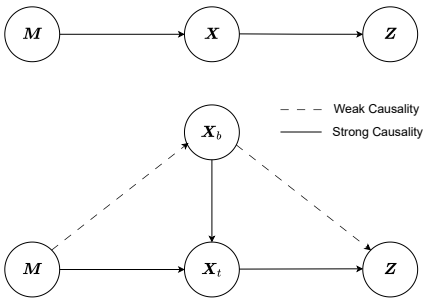

Figure 2: Graphical Models of COMRL.

Accordingly, we name $I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b)$ and $I(\boldsymbol{Z}; \boldsymbol{X}_b)$ the *primary* and *lesser* causality in our problem respectively. With this prior knowledge and assumption, we present the central theorem of this paper:

Theorem 2.3 (Central Theorem)**.** *Let $\equiv$ denote equality up to a constant, then*

$$\underbrace{I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b)}_{\text{primary causality}} \quad \leq \quad I(\boldsymbol{Z}; \boldsymbol{M}) \quad \leq \quad I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b) + I(\boldsymbol{Z}; \boldsymbol{X}_b) = \quad \underbrace{I(\boldsymbol{Z}; \boldsymbol{X})}_{\text{primary + lesser causality}}$$

holds up to a constant, where

1. $\mathcal{L}_{\text{FOCAL}} \equiv -I(\boldsymbol{Z}; \boldsymbol{X})$.
2. $\mathcal{L}_{\text{CORRO}} \equiv -I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b)$.
3. $\mathcal{L}_{\text{CSRO}} \geq -\left((1-\lambda) I(\boldsymbol{Z}; \boldsymbol{X}) + \lambda I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b)\right)$.

Proof. See Appendix B. □

Theorem 2.3 induces several key observations. Firstly, the FOCAL and CORRO objectives operate as an upper bound and a lower bound of the relevant information $I(\boldsymbol{Z}; \boldsymbol{M})$ respectively. Since

one would like to maximize $I(\boldsymbol{Z};\boldsymbol{M})$ according to Definition 2.1, CORRO, which maximizes the lower bound $I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)$, can effectively optimize $I(\boldsymbol{Z};\boldsymbol{M})$ with theoretical assurance. However, FOCAL which maximizes the upper bound $I(\boldsymbol{Z};\boldsymbol{X}_t,\boldsymbol{X}_b)$ provides *no guarantee* for $I(\boldsymbol{Z};\boldsymbol{M})$. Mathematically since $I(\boldsymbol{Z};\boldsymbol{X}) = I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b) + I(\boldsymbol{Z};\boldsymbol{X}_b)$, maximizing the FOCAL objective may instead significantly elevates the lesser causality $I(\boldsymbol{Z};\boldsymbol{X}_b)$, which is undesirable since it contains *spurious correlation* between the task representation $\boldsymbol{Z}$ and behavior policy π_β. This explains *why FOCAL is less robust to context shift compared to CORRO*.

Secondly, CSRO as the latest COMRL algorithm, inherently optimizes a linear combination of the FOCAL and CORRO objectives. In the $0 \leq \lambda \leq 1$ regime, the CSRO objective becomes a convex interpolation of the upper bound $I(\boldsymbol{Z};\boldsymbol{X})$ and the lower bound $I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)$ of $I(\boldsymbol{Z};\boldsymbol{M})$, which in essence, enforces a *trade-off* between the causal ($\boldsymbol{Z}$ with T & ρ_0) and spurious ($\boldsymbol{Z}$ with π_β) correlation contained in $I(\boldsymbol{Z};\boldsymbol{X}_b)$. This accounts for the *improved performance of CSRO compared to FOCAL and CORRO*.

2.3 Instantiations of UNICORN

By providing a unified view of pre-existing COMRL algorithms, Theorem 2.3 opens up avenues for novel algorithmic implementations by seeking alternative approximations of the true objective $I(\boldsymbol{Z};\boldsymbol{M})$. To demonstrate the impact of our proposed UNICORN framework, we discuss two instantiations as follows:

Supervised UNICORN $I(\boldsymbol{Z};\boldsymbol{M})$ can be re-expressed as

$$\begin{aligned} I(\boldsymbol{Z};\boldsymbol{M}) &= H(\boldsymbol{M}) - H(\boldsymbol{M}|\boldsymbol{Z}) \equiv -H(\boldsymbol{M}|\boldsymbol{Z}) \\ &= \mathbb{E}_{\boldsymbol{z}}\mathbb{E}_{M\sim p(M|\boldsymbol{z})}\left[\log p(M|\boldsymbol{z})\right] = -\mathbb{E}_{\boldsymbol{z}}\left[H(\boldsymbol{M}|\boldsymbol{Z}=\boldsymbol{z})\right]. \end{aligned} \tag{7}$$

where $H(\cdot)$ is entropy. Since in practice, each $\boldsymbol{z}^i$ of sample $\boldsymbol{x}^i$ is collected within a specific task M^i, minimizing the parameterized entropy $H_{\boldsymbol{\theta}}(\boldsymbol{M}|\boldsymbol{Z}=\boldsymbol{z}^i)$ is equivalent to finding an optimal function $p_{\boldsymbol{\theta}}(M|\boldsymbol{z})$ which correctly assigns the ground-truth label M^i to $\boldsymbol{z}^i$, i.e., optimizing $p_{\boldsymbol{\theta}}(M|\boldsymbol{z})$ towards a delta function $\delta(M-M^i)$ for continuous $\boldsymbol{M}$ or an indicator function $\mathbb{1}(M=M^i)$ for discrete $\boldsymbol{M}$. This implies that

$$\arg\min_{\theta} H_{\boldsymbol{\theta}}(\boldsymbol{M}|\boldsymbol{Z}=\boldsymbol{z}^i) = \arg\max_{\theta} \log p_{\boldsymbol{\theta}}(M^i|\boldsymbol{z}^i). \tag{8}$$

Suppose the task label M is given during training and a total of n_M training tasks $\{M^i\}_{i=1}^{n_M}$ are drawn from the task distribution $p(M)$ for meta-training. Under this supervised scenario, by substituting Eqn 8 into 7, we have

$$\begin{aligned} \arg\max_{\theta} I(\boldsymbol{Z};\boldsymbol{M}) &= \arg\max_{\theta} \mathbb{E}_{\boldsymbol{z}}\mathbb{E}_{M}\left[\delta(M-M^i)\log p_{\boldsymbol{\theta}}(M^i|\boldsymbol{z})\right] \\ &\simeq \arg\max_{\theta} \mathbb{E}_{\boldsymbol{z}}\left[\sum_{i=1}^{n_M}\mathbb{1}(M^i=M)\log p_{\boldsymbol{\theta}}(M^i|\boldsymbol{z})\right], \end{aligned} \tag{9}$$

which is precisely the negative cross-entropy loss $H(\boldsymbol{M}, P(\boldsymbol{M}|\boldsymbol{X}))$ for n_M-way classification with feature $\boldsymbol{z}$ and classifier $p_{\boldsymbol{\theta}}$. We therefore define the objective of supervised UNICORN as

$$\begin{aligned} \mathcal{L}_{\text{UNICORN-SUP}} &\coloneqq H(\boldsymbol{M}, P(\boldsymbol{M}|\boldsymbol{X})) \\ &= -\mathbb{E}_{\boldsymbol{x},\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\sum_{j=1}^{n_M}\mathbb{1}(M^i=M)\log p_{\boldsymbol{\theta}}(M^i|\boldsymbol{z})\right]. \end{aligned} \tag{10}$$

Note that $\mathcal{L}_{\text{UNICORN-SUP}}$ is convex and operates as a finite-sample approximation of $-I(\boldsymbol{Z};\boldsymbol{M})$, for which we give the following theorem:

Theorem 2.4 (Concentration bound for supervised UNICORN)**.** *Denote by $\hat{I}(\boldsymbol{Z};\boldsymbol{M})$ the empirical estimate of $I(\boldsymbol{Z};\boldsymbol{M})$ by n_M tasks, $\bar{I}(\boldsymbol{Z};\boldsymbol{M})$ the expectation, then with probability at least $1-\delta$,*

$$\left|\hat{I}(\boldsymbol{Z};\boldsymbol{M}) - \bar{I}(\boldsymbol{Z};\boldsymbol{M})\right| \leq \sqrt{\frac{\text{Var}(H(\boldsymbol{Z}|\boldsymbol{M}))}{n_M\delta}}. \tag{11}$$

Proof. See Appendix B. □

Theorem 2.4 bounds the finite-sample estimation error of the empirical risk $\hat{I}(\boldsymbol{Z};\boldsymbol{M})$ with n_M task instances drawn from the real task distribution $p(\boldsymbol{M})$. The supervised UNICORN has the merit of directly estimating and optimizing the real objective $I(\boldsymbol{Z};\boldsymbol{M})$, which requires explicit knowledge of the task label M^i and a substantial amount of task instances according to Theorem 2.4. To trade-off computation and performance, we choose to sample 20 training tasks for all RL environments in our experiments.

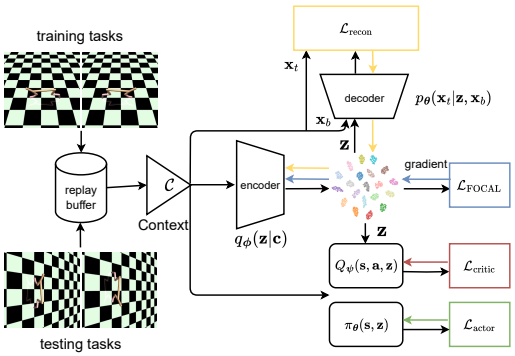

Figure 3: **Meta-learning procedure of UNICORN-SS**. The supervised variant UNICORN-SUP simply replaces the decoder by a classifier $p_{\boldsymbol{\theta}}(M|\boldsymbol{z})$ and optimize a cross-entropy loss instead of $\mathcal{L}_{\text{recon}}$ and $\mathcal{L}_{\text{FOCAL}}$.

Self-supervised UNICORN In practice, offline RL datasets may often be collected with limited knowledge of the task specifications or labels. In this scenario, previous works implement self-supervised learning [32] to obtain effective representation $\boldsymbol{Z}$, such as the contrastive-based FOCAL/CORRO to optimize $I(\boldsymbol{Z};\boldsymbol{X})$/$I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)$ respectively; or generative approaches like VariBAD [33]/BOReL [14] to reconstruct the trajectories $\boldsymbol{X}$ by variational inference, which is equivalent to maximizing $I(\boldsymbol{X}_t;\boldsymbol{Z},\boldsymbol{X}_b)$. By Theorem 2.3, these methods optimize a relatively loose upper/lower bound of $I(\boldsymbol{Z};\boldsymbol{M})$, which can be improved by a convex combination of the two bounds:

$$I(\boldsymbol{Z};\boldsymbol{M}) \approx \alpha I(\boldsymbol{Z};\boldsymbol{X}) + (1-\alpha) I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b), \tag{12}$$

where $0 \leq \alpha \leq 1$ is a hyperparameter. Implementing each term in Eqn 12 allows ample decision choices, such as the contrastive losses in Eqs. (3), (4) and (6) or autoregressive generation via Decision Transformer [34, 35] or RNN [33]. For demonstration, in this paper we employ a *contrastive* objective $\mathcal{L}_{\text{FOCAL}}$ as in Eq. (3) for $I(\boldsymbol{Z};\boldsymbol{X})$ while approximate $I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)$ by reconstruction. By the chain rule of mutual information [36]:

$$\begin{aligned} I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b) &= I(\boldsymbol{X}_t;\boldsymbol{Z},\boldsymbol{X}_b) - I(\boldsymbol{X}_t;\boldsymbol{X}_b) \\ &\equiv I(\boldsymbol{X}_t;\boldsymbol{Z},\boldsymbol{X}_b), \end{aligned} \tag{13}$$

since $I(\boldsymbol{X}_t;\boldsymbol{X}_b)$ is a constant when $\boldsymbol{X}_t$ and $\boldsymbol{X}_b$ are drawn from a fixed distribution as in offline RL. Moreover, by definition of mutual information:

$$\begin{aligned} I(\boldsymbol{X_t};\boldsymbol{Z},\boldsymbol{X_b}) &= \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log \frac{p(\boldsymbol{x_t}|\boldsymbol{z},\boldsymbol{x_b})}{p(\boldsymbol{x_t})}\right] \\ &\equiv \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log p(\boldsymbol{x_t}|\boldsymbol{z},\boldsymbol{x_b})\right] \\ &\geq \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x_t},\boldsymbol{x_b})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x_t}|\boldsymbol{z},\boldsymbol{x_b})\right], \end{aligned} \tag{14}$$

which induces a *generative* objective $\mathcal{L}_{\text{recon}} := -I(\boldsymbol{X}_t;\boldsymbol{Z},\boldsymbol{X}_b)$ by reconstructing $\boldsymbol{X}_t$ with a decoder network $p_{\boldsymbol{\theta}}(\cdot|\boldsymbol{z},\boldsymbol{x_b})$ conditioning on $\boldsymbol{Z}$ and $\boldsymbol{X_b}$. As a result, the proposed unsupervised UNICORN objective can be rescaled as Eqn 15. The influence of the hyper-parameter $\frac{\alpha}{1-\alpha}$ is shown in Appendix C.4.

$$\mathcal{L}_{\text{UNICORN-SS}} := \mathcal{L}_{\text{recon}} + \frac{\alpha}{1-\alpha}\mathcal{L}_{\text{FOCAL}}. \tag{15}$$

We summarize our learning procedure in Fig. 3 with pseudo-code in Algorithms 1 and 2. A holistic comparison of our proposed algorithms with related contextual meta-RL methods is shown in Table 1. The extra KL divergence can be interpreted as the result of a variational approximation to an information bottleneck [37, 38] that constrains the mutual information between $\boldsymbol{Z}$ and $\boldsymbol{X}$, which we found unnecessary in our offline setting (see ablation in Table 6.) Behavior regularized actor critic [39] is employed to tackle the bootstrapping error [40] for downstream offline RL implementation.

3 Experiments

For brevity, we name our proposed supervised and self-supervised algorithms UNICORN-SUP and UNICORN-SS respectively. To evaluate their effectiveness, our main experiments are organized to

Table 1: Comparison between UNICORN instantiations and related existing contextual meta-RL methods. For clarity, "Representation Learning Objective" only lists the loss functions of $\boldsymbol{Z}$ that are independent of the downstream RL tasks. Note that $I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b) \equiv I(\boldsymbol{X}_t; \boldsymbol{Z}, \boldsymbol{X}_b)$ holds *only* for offline RL.

Method	Setting	Representation Learning Objective	Implementation	Context $\boldsymbol{X}$
UNICORN-SUP	Offline	$I(\boldsymbol{Z}; \boldsymbol{M})$	Predictive	Transition
UNICORN-SS	Offline	$\alpha I(\boldsymbol{Z}; \boldsymbol{X}) + (1-\alpha) I(\boldsymbol{X}_t; \boldsymbol{Z}, \boldsymbol{X}_b)$	Contrastive+Generative	Transition
FOCAL [12, 22]	Offline	$I(\boldsymbol{Z}; \boldsymbol{X})$	Contrastive	Transition
CORRO [24]	Offline	$I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b)$	Contrastive	Transition
CSRO [23]	Offline	$(1-\lambda) I(\boldsymbol{Z}; \boldsymbol{X}) + \lambda I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{X}_b)$	Contrastive	Transition
GENTLE [41]	Offline	$I(\boldsymbol{X}_t; \boldsymbol{Z}, \boldsymbol{X}_b)$	Generative	Transition
BOReL [14]	Offline	$I(\boldsymbol{X}_t; \boldsymbol{Z}, \boldsymbol{X}_b) - D_{\mathrm{KL}}(q_\phi(\boldsymbol{Z}|\boldsymbol{X})\|\|p_\theta(\boldsymbol{Z}))$	Generative	Trajectory
VariBAD [33]	Online	$I(\boldsymbol{X}_t; \boldsymbol{Z}, \boldsymbol{X}_b) - D_{\mathrm{KL}}(q_\phi(\boldsymbol{Z}|\boldsymbol{X})\|\|p_\theta(\boldsymbol{Z}))$	Generative	Trajectory
PEARL [7]	Online	$-D_{\mathrm{KL}}(q_\phi(\boldsymbol{Z}|\boldsymbol{X})\|\|p_\theta(\boldsymbol{Z}))$	N/A	Transition
ContraBAR [42]	Offline&Online	$I(\boldsymbol{Z}; \boldsymbol{X}_t|\boldsymbol{A})$	Contrastive	Trajectory

address three primary inquiries regarding the core advantages of UNICORN: (1) How does UNICORN perform on in-distribution tasks? (2) How well can UNICORN generalize to data collected by OOD behavior policies? (3) Can UNICORN outperform consistently across datasets of different qualities? The performance is measured by the average return across 20 testing tasks randomly sampled for each environment.

3.1 Experimental Setup

We adopt MuJoCo [26] and MetaWorld [3] benchmark to evaluate our method, involving six robotic locomotion tasks which require adaptation across reward functions (HalfCheetah-Dir, HalfCheetah-Vel, Ant-Dir, Reach) or across dynamics (Hopper-Param, Walker-Param). We compare UNICORN with six competitive OMRL algorithms. These baselines are categorized as context-based, gradient-based and transformer-based methods:

FOCAL, CORRO, CSRO are context-based methods that can be seen as special cases or approximations of UNICORN, see details in Section 2.

Supervised is a *context-based* method, directly using actor-critic loss to train the policy/value networks and task encoder end-to-end. We find it a competitive baseline across all benchmarks.

MACAW [13] is a *gradient-based* method, using supervised advantage-weighted regression for both the inner and outer loop of meta-training.

Prompt-DT [43] is a *transformer-based* method, taking context as the prompt of a Decision Transformer (DT) [34, 35] to solve OMRL as a conditional sequence modeling problem.

We perform our evaluation in a few-shot manner. For gradient-based methods, we first update the meta-policy with a batch of testing context and then evaluate the agent. For transformer-based methods, we utilize the trajectory segment as the prompt to condition the auto-regressive rollout.

Table 2: **Average testing returns of UNICORN against baselines on datasets collected by IID and OOD behavior policies.** Each result is averaged by 6 random seeds. The best is **bolded** and the second best is underlined.

Algorithm	HalfCheetah-Dir		HalfCheetah-Vel		Ant-Dir		Hopper-Param		Walker-Param		Reach	
	IID	OOD	IID	OOD	IID	OOD	IID	OOD	IID	OOD	IID	OOD
UNICORN-SS	**1307±26**	**1296±24**	**-22±1**	-94±5	267±14	236±18	**316±6**	**304±11**	**419±44**	**407±46**	**2775±241**	2604±183
UNICORN-SUP	1296±20	1130±76	-25±3	**-91±5**	250±4	**239±16**	312±4	302±12	322±28	312±39	2681±111	2641±140
CSRO	1180±228	458±253	-28±1	-102±5	**276±19**	233±12	310±6	301±10	310±58	279±65	2720±235	**2801±182**
CORRO	704±450	245±146	-37±3	-112±2	148±13	120±12	283±8	272±13	277±38	213±48	2468±175	2322±327
FOCAL	1186±272	861±253	-22±1	-97±2	217±29	173±24	302±4	297±13	308±98	286±91	2424±256	2316±303
Supervised	962±356	782±429	-24±1	-104±1	238±39	202±38	306±10	294±8	256±60	210±28	2489±248	2283±205
MACAW	1155±10	450±6	-56±2	-188±1	26±3	0±0	218±6	205±2	141±9	130±5	2431±157	1728±79
Prompt-DT	1176±40	-25±9	-118±66	-249±21	1±0	0±0	234±5	202±5	185±9	156±17	2165±85	1896±111

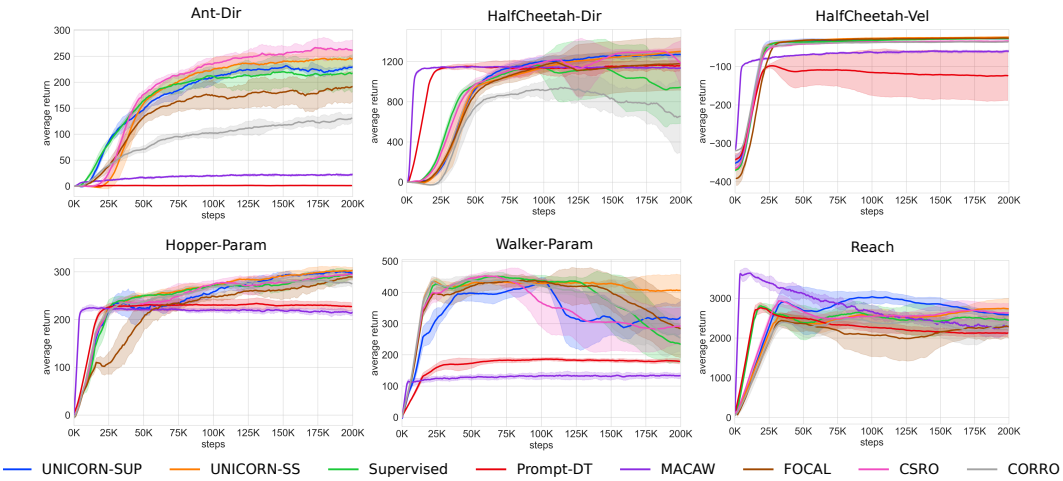

Figure 4: **Testing returns of UNICORN against baselines on six benchmarks.** Solid curves refer to the mean performance of trials over 6 random seeds, and the shaded areas characterize the standard deviation of these trials.

3.2 Few-Shot Generalization to In-Distribution Data

Denote by $\{\pi_\beta^{i,t}\}_{i=1:N_M,t=0:N}$ the checkpoints of behavior policies for all tasks logged during data collection (i.e., training SAC agents), where i refers to the task index and t represents training iteration. In the in-distribution (IID) setting, for each test task M^j, we use behavior policies only from the *same tasks* $\{\pi_\beta^{j,t}\}_{j,t=0:N}$ to collect the context and the conditioned rollouts, as visualized in Figure 1. Figure 4 illustrates the learning curves of UNICORN vs. all baselines, which correspond to the IID entries of Table 2. The results demonstrate that UNICORN variants, especially UNICORN-SS, consistently achieve SOTA performance across all benchmarks. The observation that the asymptotic performance of CSRO is comparable to UNICORN is expected since it also optimizes a linear combination of the lower and upper bound of $I(\boldsymbol{Z};\boldsymbol{M})$ by Theorem 2.3, with a different implementation choice. Notably, UNICORN exhibits remarkable stability, especially on HalfCheetah-Dir and Walker-Param, where the other context-based methods suffer a significant performance decline during the later training process. Although the gradient-based MACAW and the transformer-based Prompt-DT show faster convergence, their asymptotic performance leaves much to be desired. Moreover, the average training time for MACAW is almost three times longer than the context-based methods.

3.3 Few-Shot Generalization to Out-of-Distribution Behavior Policies

To evaluate OOD generalization, for each test task M^j, we use *all* behavior policies in $\{\pi_\beta^{i,t}\}_{i=1:N_M,t=0:N}$ to collect the OOD contexts and the conditioned rollouts, as visualized in Figure 1. The OOD testing performance is measured by averaging returns across all testing tasks, as shown in the OOD entries of Table 2. While all methods suffer notable decline when facing OOD context, UNICORN maintains the strongest performance by a large margin. Another observation is that the context-based methods are in general significantly more robust compared to the gradient-based MACAW and transformer-based Prompt-DT. This also delineates the storyline of COMRL development from 2021 to date

$$\underset{(2021)}{\text{FOCAL}} \rightarrow \underset{(2022)}{\text{CORRO}} \rightarrow \underset{(2023)}{\text{CSRO}} \rightarrow \underset{(2024)}{\text{UNICORN}}$$

as a roadmap for pursuing more robust and generalizable task representation learning for COMRL.

3.4 Influence of Data Quality

To test whether the UNICORN framework can be applied to different data distributions, we collect three types of datasets whose size is equal to the mixed dataset used in Section 3.2 and 3.3: random, medium and expert, which are characterized by the quality of the behavior policies.

As shown in Table 3, UNICORN demonstrates SOTA performance on all types of datasets. Notably, despite CSRO having better IID results than UNICORN on Ant-Dir under the mixed distribution, UNICORN surpasses CSRO by a large margin under the narrow distribution (medium and expert) and random distribution.

We observe that CORRO fails on all three datasets. This may be due to the reliance of CORRO on the negative sample generator. Under narrow data distribution or poor random data distribution, it might be more challenging to generate negative samples of good quality. As for Prompt-DT, its performance improves significantly with increased data quality, which is expected since the decision transformer adopts a behavior-cloning-like supervised learning style.

Table 3: **UNICORN vs. baselines on Ant-Dir datasets of various qualities.** Each result is averaged by 6 random seeds. The best is **bolded** and the second best is underlined.

Algorithm	Random		Medium		Expert	
	IID	OOD	IID	OOD	IID	OOD
UNICORN-SS	**81±18**	**62±6**	**220±23**	**243±10**	**279±10**	**262±13**
UNICORN-SUP	75±15	60±5	140±11	126±32	247±15	229±19
CSRO	2±3	0±1	166±10	198±17	252±39	202±45
CORRO	1±1	0±0	8±5	-7±2	-4±10	-14±9
FOCAL	67±26	44±10	171±84	187±86	229±42	246±20
Supervised	65±6	47±12	149±50	110±80	249±33	215±60
MACAW	3±1	0±0	28±2	1±1	88±43	1±1
Prompt-DT	1±0	0±0	2±4	0±1	78±15	1±2

4 Discussion

This section presents more empirical evidence on the applicability of the UNICORN framework.

4.1 Is UNICORN Model-Agnostic?

As UNICORN tackles task representation learning from a general information theoretic perspective, it is in principle *model-agnostic*. Hence a straightforward idea is to transfer UNICORN to other model architectures like DT, which we envision to be a promising backbone for large-scale training of RL foundation models.

Table 4: **DT implementation of COMRL on HalfCheetah-Dir and Hopper-Param.** Each result is averaged by 6 random seeds.

Algorithm	HalfCheetah-Dir		Hopper-Param	
	IID	OOD	IID	OOD
UNICORN-SS	1307±26	1296±24	316±6	304±11
UNICORN-SS-DT	1233±10	1186±43	304±4	291±4
UNICORN-SUP-DT	1227±21	1065±57	308±6	297±2
FOCAL-DT	1209±33	652±36	293±4	284±5
Prompt-DT	1177±40	-25±9	234±5	203±5

We employ a simple implementation by embedding the task representation vector $\boldsymbol{z}$ as the first token in sequence to prompt the learning and generation of DT. Instead of the unlearnable prompt used in the original Prompt-DT, we enforce task representation learning of the prompt by optimizing $\mathcal{L}_{\text{UNICORN-SS}}$, $\mathcal{L}_{\text{UNICORN-SUP}}$ and $\mathcal{L}_{\text{FOCAL}}$. We name these variants as UNICORN-SS-DT, UNICORN-SUP-DT and FOCAL-DT respectively.

As shown in Table 4 and Figure 7, applying UNICORN and FOCAL on DT results in significant improvement in both IID and OOD generalization, with UNICORN being the superior option. Despite the gap between our implementation of UNICORN-DT and its MLP counterpart in terms of asymptotic performance, we expect UNICORN-DT to extrapolate favorably in the regime of greater dataset size and model parameters due to the scaling law of transformer [44] and DT [45]. We leave this verification to future work.

4.2 Can UNICORN be Exploited for Model-Based Paradigms?

With decoder $p_\theta(\boldsymbol{x}_t|\boldsymbol{z}, \boldsymbol{x}_b)$, UNICORN-SS can potentially enable a model-based paradigm by generating data using customized latent $\boldsymbol{z}$, which can be especially useful for generalization to OOD tasks. We implement this idea on Ant-Dir by sorting the tasks according to the polar angle of their goal direction ($\theta \in [0, 2\pi)$), and construct task-level OOD by taking 20 consecutive tasks in the middle for training and the rest for testing (Figure 1). Gaussian noise $\boldsymbol{\epsilon}$ is applied to the real task representation $\boldsymbol{z}$ and $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{z}+\boldsymbol{\epsilon})$ is then fed to the decoder to generate $(\boldsymbol{s}', r)$. The imaginary rollouts $\{(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}'_t, r_t)\}_{t=1}^n$ are used during the training of RL agent. We adopt the conventional ensemble technique in model-based RL [46, 47] to stabilize the training process.

As shown in Figure 5, UNICORN-SS-enabled model-based RL is the only method to achieve positive performance (indicating positive few-shot transfer) in this extremely challenging scenario where context shift is induced by OOD tasks.

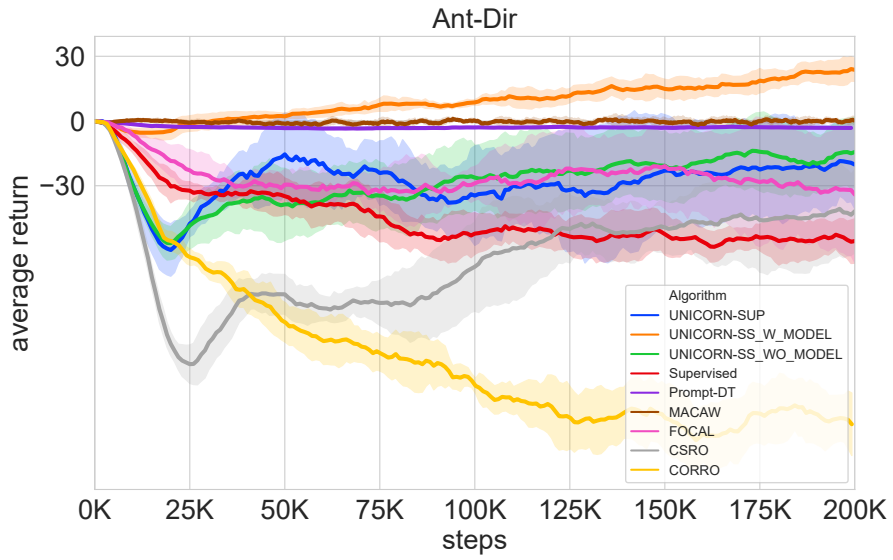

Figure 5: **Testing returns for OOD tasks.** The learning curve is averaged by 6 random seeds.

5 Conclusion & Limitation

In this paper, we unify three major context-based offline meta-RL algorithms, FOCAL, CORRO and CSRO (potentially a lot more) into a single framework from an information theoretic representation learning perspective. Our theory offers valuable insight on how these methods are inherently connected and incrementally evolved in pursuit of better generalization against context shift. Based on the proposed framework, we instantiate two novel algorithms called UNICORN-SUP and UNICORN-SS which are demonstrated to be remarkably robust and broadly applicable through extensive experiments. We believe our study offers potential for new implementations, optimality bounds and algorithms for both fully-offline and offline-to-online COMRL. Moreover, we expect our framework to incorporate almost all existing COMRL methods that tackle task representation learning. Since according to Definition 2.1, as long as the method tries to solve COMRL by learning a sufficient statistic $\boldsymbol{Z}$ of $\boldsymbol{X}$ w.r.t. $\boldsymbol{M}$, it will eventually come down to optimizing an information-theoretic objective equivalent to $I(\boldsymbol{Z}, \boldsymbol{M})$, or a lower/upper bound approximation like the ones introduced by Theorem 2.3, up to some regularizations or constraints. With such mathematical generalization as well as the empirical evidence of being model-agnostic (see Sec 4.1), we envision UNICORN as a nascent pre-training paradigm of foundation models for decision making [21].

Limitation Since our framework assumes a decoupling of task representation learning and offline policy optimization, it does not directly elucidate how high-quality representations are indicative of higher downstream RL performance. Another limitation would be the scale of the experiments. Our offline datasets cover at most 40 tasks and $\sim 450k$ transition tuples, which might be the reason why UNICORN-SUP is inferior to UNICORN-SS (n_M too small in Theorem 2.4). However, we expect our conclusion to extrapolate to larger task sets, dataset sizes and model parameters, for which the DT variants may demonstrate better scaling. Lastly, in principle, the information theoretic formalization in this work should be directly applicable to online RL. However, many of our key derivations rely on the static assumption of $\boldsymbol{M}$ and $\boldsymbol{X}$ (see Appendix B for details), which are apparently violated in the online scenario. Extending our framework to online RL is interesting and nontrivial, which we leave for future work.

Acknowledgments and Disclosure of Funding

The authors would like to thank the anonymous reviewers for their insightful comments and suggestions. The work described in this paper was supported partially by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project Reference Number: T45-401/22-N), by a grant from the Hong Kong Innovation and Technology Fund (Project Reference Number: MHP/086/21) and by the National Key Research and Development Program of China (No. 2020YFA0711402).

References

[1] Mandi Zhao, Pieter Abbeel, and Stephen James. On the effectiveness of fine-tuning versus meta-reinforcement learning. *Advances in Neural Information Processing Systems*, 35:26519–26531, 2022.

[2] Jacob Beck, Risto Vuorio, Evan Zheran Liu, Zheng Xiong, Luisa Zintgraf, Chelsea Finn, and Shimon Whiteson. A survey of meta-reinforcement learning. *arXiv preprint arXiv:2301.08028*, 2023.

[3] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pages 1094–1100. PMLR, 2020.

[4] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.

[5] Leo Feng, Padideh Nouri, Aneri Muni, Yoshua Bengio, and Pierre-Luc Bacon. Designing biological sequences via meta-reinforcement learning and bayesian optimization. *arXiv preprint arXiv:2209.06259*, 2022.

[6] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[7] Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, and Deirdre Quillen. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *International conference on machine learning*, pages 5331–5340. PMLR, 2019.

[8] Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature medicine*, 25(1):16–18, 2019.

[9] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A Al Sallab, Senthil Yogamani, and Patrick Pérez. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):4909–4926, 2021.

[10] Di Zhang, Bowen Lv, Hai Zhang, Feifan Yang, Junqiao Zhao, Hang Yu, Chang Huang, Hongtu Zhou, Chen Ye, and Changjun Jiang. Focus on what matters: Separated models for visual-based rl generalization. *arXiv preprint arXiv:2410.10834*, 2024.

[11] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

[12] Lanqing Li, Rui Yang, and Dijun Luo. Focal: Efficient fully-offline meta-reinforcement learning via distance metric learning and behavior regularization. In *International Conference on Learning Representations*, 2021.

[13] Eric Mitchell, Rafael Rafailov, Xue Bin Peng, Sergey Levine, and Chelsea Finn. Offline meta-reinforcement learning with advantage weighting. In *International Conference on Machine Learning*, pages 7780–7791. PMLR, 2021.

[14] Ron Dorfman, Idan Shenfeld, and Aviv Tamar. Offline meta reinforcement learning–identifiability challenges and effective data collection strategies. *Advances in Neural Information Processing Systems*, 34:4607–4618, 2021.

[15] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

[16] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021.

[17] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, 2021.

[18] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, 2021.

[19] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[20] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gómez Colmenarejo, Alexander Novikov, Gabriel Barth-maron, Mai Giménez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *Transactions on Machine Learning Research*, 2022.

[21] Sherry Yang, Ofir Nachum, Yilun Du, Jason Wei, Pieter Abbeel, and Dale Schuurmans. Foundation models for decision making: Problems, methods, and opportunities. *arXiv preprint arXiv:2303.04129*, 2023.

[22] Lanqing Li, Yuanhao Huang, Mingzhe Chen, Siteng Luo, Dijun Luo, and Junzhou Huang. Provably improved context-based offline meta-rl with attention and contrastive learning. *arXiv preprint arXiv:2102.10774*, 2021.

[23] Yunkai Gao, Rui Zhang, Jiaming Guo, Fan Wu, Qi Yi, Shaohui Peng, Siming Lan, Ruizhi Chen, Zidong Du, Xing Hu, et al. Context shift reduction for offline meta-reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[24] Haoqi Yuan and Zongqing Lu. Robust task representations for offline meta-reinforcement learning via contrastive learning. In *International Conference on Machine Learning*, pages 25747–25759. PMLR, 2022.

[25] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[26] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

[27] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

[28] Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *International conference on machine learning*, pages 1312–1320. PMLR, 2015.

[29] Jiachen Li, Quan Vuong, Shuang Liu, Minghua Liu, Kamil Ciosek, Henrik Christensen, and Hao Su. Multi-task batch reinforcement learning with metric learning. *Advances in Neural Information Processing Systems*, 33:6197–6210, 2020.

[30] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[31] Pengyu Cheng, Weituo Hao, Shuyang Dai, Jiachang Liu, Zhe Gan, and Lawrence Carin. Club: A contrastive log-ratio upper bound of mutual information. In *International conference on machine learning*, pages 1779–1788. PMLR, 2020.

[32] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *IEEE transactions on knowledge and data engineering*, 35(1):857–876, 2021.

[33] Luisa Zintgraf, Kyriacos Shiarlis, Maximilian Igl, Sebastian Schulze, Yarin Gal, Katja Hofmann, and Shimon Whiteson. Varibad: A very good method for bayes-adaptive deep rl via meta-learning. In *International Conference on Learning Representations*, 2020.

[34] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

[35] Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. *Advances in neural information processing systems*, 34:1273–1286, 2021.

[36] Raymond W Yeung. *Information theory and network coding*. Springer Science & Business Media, 2008.

[37] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*, pages 1–5. IEEE, 2015.

[38] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017.

[39] Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

[40] Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

[41] Renzhe Zhou, Chen-Xiao Gao, Zongzhang Zhang, and Yang Yu. Generalizable task representation learning for offline meta-reinforcement learning with data limitations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17132–17140, 2024.

[42] Era Choshen and Aviv Tamar. Contrabar: Contrastive bayes-adaptive deep rl. In *International Conference on Machine Learning*, 2023.

[43] Mengdi Xu, Yikang Shen, Shun Zhang, Yuchen Lu, Ding Zhao, Joshua Tenenbaum, and Chuang Gan. Prompting decision transformer for few-shot policy generalization. In *international conference on machine learning*, pages 24631–24645. PMLR, 2022.

[44] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[45] Kuang-Huei Lee, Ofir Nachum, Mengjiao Sherry Yang, Lisa Lee, Daniel Freeman, Sergio Guadarrama, Ian Fischer, Winnie Xu, Eric Jang, Henryk Michalewski, et al. Multi-game decision transformers. *Advances in Neural Information Processing Systems*, 35:27921–27936, 2022.

[46] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems*, 33:14129–14142, 2020.

[47] Hai Zhang, Hang Yu, Junqiao Zhao, Di Zhang, Chang Huang, Hongtu Zhou, Xiao Zhang, and Chen Ye. How to fine-tune the model: Unified model shift and model bias policy optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[48] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

A Pseudo-Code

Algorithm 1 UNICORN Meta-training

Require: Offline Datasets $\boldsymbol{X} = \{X^i\}_{i=1}^{N_{\text{train}}}$, training tasks $\boldsymbol{M} = \{M^i\}_{i=1}^{N_{\text{train}}}$, initialized learned policy $\pi_{\boldsymbol{\omega}}$, Q function $Q_{\boldsymbol{\psi}}$, context encoder $q_{\boldsymbol{\phi}}$, classifier/decoder $p_{\boldsymbol{\theta}}$, hyper-parameter α, task batch size N_{tb}, learning rates β_1, β_2, β_3, β_4

while not done **do**
  Sample task batch $\{M^j\}_{j=1}^{N_{tb}} \sim \boldsymbol{M}$ and the corresponding replay buffer $\mathcal{R} = \{X^j\}_{j=1}^{N_{tb}} \sim \boldsymbol{X}$
  for step in each iter **do**
    // Train the context encoder and decoder
    Sample context $\{\boldsymbol{c}^j\}_{j=1}^{N_{tb}} \sim \mathcal{R}$
    Obtain task embeddings $\{\boldsymbol{z}^j \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{c}^j)\}_{j=1}^{N_{tb}}$
    Estimate $\mathcal{L}_{\text{UNICORN}}$ ($\mathcal{L}_{\text{UNICORN-SUP}}$ or $\mathcal{L}_{\text{UNICORN-SS}}$)
    $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \beta_1 \nabla_{\boldsymbol{\phi}} \mathcal{L}_{\text{UNICORN}}$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \beta_2 \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{UNICORN}}$
    // Train the actor and critic
    Detach the task embeddings $\{\boldsymbol{z}^j\}_{j=1}^{N_{tb}}$
    Sample training data $\{\boldsymbol{d}^j\}_{j=1}^{N_{tb}} \sim \mathcal{R}$
    Estimate $\mathcal{L}_{\text{actor}}$, $\mathcal{L}_{\text{critic}}$
    $\boldsymbol{\omega} \leftarrow \boldsymbol{\omega} - \beta_3 \nabla_{\boldsymbol{\omega}} \mathcal{L}_{\text{actor}}$
    $\boldsymbol{\psi} \leftarrow \boldsymbol{\psi} - \beta_4 \nabla_{\boldsymbol{\psi}} \mathcal{L}_{\text{critic}}$
  end for
end while

Algorithm 2 UNICORN Meta-testing

Require: Offline Datasets $\boldsymbol{X} = \{X^i\}_{i=1}^{N_{\text{test}}}$, testing tasks $\boldsymbol{M} = \{M^i\}_{i=1}^{N_{\text{test}}}$, trained policy $\pi_{\boldsymbol{\omega}}$ and context encoder $q_{\boldsymbol{\phi}}$

for each M^i **do**
  Sample context $\boldsymbol{c}^i \sim X^i$
  Obtain task embedding $\boldsymbol{z}^i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{c}^i)$
  Rollout policy $\pi_{\boldsymbol{\omega}}(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{z}^i)$ for evaluation
end for

B Proofs

B.1 Proof of Theorem 2.3

We first prove the following lemma:

Lemma B.1. *For COMRL, $I(\boldsymbol{Z};\boldsymbol{M}) \geq I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_b)$; or equivalently, $I(\boldsymbol{Z};\boldsymbol{M};\boldsymbol{X}_b) \geq 0$, both up to a constant.*

Proof.

$$
\begin{aligned}
& I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_t,\boldsymbol{X}_b) = 0 \\
\Longrightarrow & I(\boldsymbol{Z};\boldsymbol{M}) - I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_t,\boldsymbol{X}_b) = I(\boldsymbol{M};\boldsymbol{X}_b,\boldsymbol{X}_t) - I(\boldsymbol{M};\boldsymbol{X}_b,\boldsymbol{X}_t|\boldsymbol{Z}) \geq 0 \\
\Longrightarrow & I(\boldsymbol{M};\boldsymbol{X}_b) + I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b) - I(\boldsymbol{M};\boldsymbol{X}_b|\boldsymbol{Z}) - I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b,\boldsymbol{Z}) \geq 0 \\
\Longrightarrow & I(\boldsymbol{M};\boldsymbol{X}_b) - I(\boldsymbol{M};\boldsymbol{X}_b|\boldsymbol{Z}) \geq \underbrace{-I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b)}_{\text{const}} + \underbrace{I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b,\boldsymbol{Z})}_{\geq 0} \geq \text{const.}
\end{aligned}
$$

Now, since $I(\boldsymbol{Z};\boldsymbol{M}) - I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_b) = I(\boldsymbol{M};\boldsymbol{X}_b) - I(\boldsymbol{M};\boldsymbol{X}_b|\boldsymbol{Z})$, we have $I(\boldsymbol{Z};\boldsymbol{M}) - I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_b) = I(\boldsymbol{Z};\boldsymbol{M};\boldsymbol{X}_b) \geq \text{const} \equiv 0$. □

With Lemma B.1, we proceed to prove Theorem 2.3 as follows:

Proof. $I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b) \equiv -\mathcal{L}_{\text{CORRO}} \leq I(\boldsymbol{Z};\boldsymbol{M})$ is shown in the Theorem 4.1. of CORRO [24] and GENTLE [41] with strong assumptions. We hereby present a proof with *no* assumptions. Given $I(\boldsymbol{Z};\boldsymbol{M};\boldsymbol{X}_b) \geq 0$ (Lemma B.1), we have

$$
\begin{aligned}
I(\boldsymbol{Z};\boldsymbol{M}) &= I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_b) + I(\boldsymbol{Z};\boldsymbol{M};\boldsymbol{X}_b) \\
&\geq I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_b) \\
&= I(\boldsymbol{M};\boldsymbol{Z},\boldsymbol{X}_t|\boldsymbol{X}_b) - I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{Z},\boldsymbol{X}_b) \\
&= \underbrace{I(\boldsymbol{Z};\boldsymbol{M}|\boldsymbol{X}_t,\boldsymbol{X}_b)}_{0 \text{ by } \boldsymbol{M}\to\boldsymbol{X}\to\boldsymbol{Z}} + I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b) - I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{Z},\boldsymbol{X}_b) \\
&= I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b) - I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{Z},\boldsymbol{X}_b) \\
&= I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b) - H(\boldsymbol{X}_t|\boldsymbol{Z},\boldsymbol{X}_b) + H(\boldsymbol{X}_t|\boldsymbol{M},\boldsymbol{Z},\boldsymbol{X}_b) \\
&\geq I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b) - H(\boldsymbol{X}_t|\boldsymbol{Z},\boldsymbol{X}_b) \\
&= \underbrace{I(\boldsymbol{M};\boldsymbol{X}_t|\boldsymbol{X}_b)}_{\text{const}} - \underbrace{H(\boldsymbol{X}_t)}_{\text{const}} + I(\boldsymbol{X}_t;\boldsymbol{Z},\boldsymbol{X}_b) \\
&\equiv I(\boldsymbol{X}_t;\boldsymbol{Z},\boldsymbol{X}_b) \\
&= I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b) + \underbrace{I(\boldsymbol{X}_t;\boldsymbol{X}_b)}_{\text{const}} \\
&\equiv I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b),
\end{aligned}
$$

as desired. The third and fourth lines are obtained by direct application of the chain rule of mutual information [36]. All mutual information terms without $\boldsymbol{Z}$ are held constant in the fully offline scenario.

$I(\boldsymbol{Z};\boldsymbol{M}) \leq I(\boldsymbol{Z};\boldsymbol{X})$ is a direct consequence of the Data Processing Inequality [36] and the Markov chain $\boldsymbol{M} \to \boldsymbol{X} \to \boldsymbol{Z}$ in Definition 2.2. We now prove the three claims regarding FOCAL, CORRO and CSRO:

1. $\mathcal{L}_{\text{FOCAL}} \equiv -I(\boldsymbol{Z};\boldsymbol{X})$.

$$
\begin{aligned}
I(\boldsymbol{Z};\boldsymbol{X}) &:= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{p(\boldsymbol{z},\boldsymbol{x})}{p(\boldsymbol{z})p(\boldsymbol{x})}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{1}{\frac{p(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})}|\mathcal{M}|}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{1}{\frac{p(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})}|\mathcal{M}|\mathbb{E}_{\boldsymbol{x}}\left[\frac{p(\boldsymbol{z}|\boldsymbol{x})}{p(\boldsymbol{z})}\right]}\right)\right] + \log(|\mathcal{M}|) \\
&\approx \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{1}{\frac{p(\boldsymbol{z})}{p(\boldsymbol{z}|\boldsymbol{x})}\sum_{M^i\in\mathcal{M}}\mathbb{E}_{\boldsymbol{x}^i\sim X^i}\left[\frac{p(\boldsymbol{z}|\boldsymbol{x}^i)}{p(\boldsymbol{z})}\right]}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{\frac{p(\boldsymbol{z}|\boldsymbol{x})}{p(\boldsymbol{z})}}{\sum_{M^*\in\mathcal{M}}\mathbb{E}_{\boldsymbol{x}^i\sim X^i}\left[\frac{p(\boldsymbol{z}|\boldsymbol{x}^i)}{p(\boldsymbol{z})}\right]}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{h(\boldsymbol{x},\boldsymbol{z})}{\sum_{M^i\in\mathcal{M}}h(\boldsymbol{x}^i,\boldsymbol{z})}\right)\right] + \log(|\mathcal{M}|)
\end{aligned}
$$

The first term on RHS is precisely the supervised contrastive learning objective introduced by a variant of FOCAL [22], which is equivalent to the negative distance metric learning loss $\mathcal{L}_{\text{FOCAL}}$ with the effect of pushing away embeddings of different tasks while pulling together those from the same task. Therefore we have

$$
\begin{aligned}
I(\boldsymbol{Z};\boldsymbol{X}) &= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{h(\boldsymbol{x},\boldsymbol{z})}{\sum_{M^i\in\mathcal{M}} h(\boldsymbol{x}^i,\boldsymbol{z})}\right)\right] + \text{const} \\
&\equiv -\mathcal{L}_{\text{FOCAL}}.
\end{aligned}
$$

2. $\mathcal{L}_{\text{CORRO}} \equiv -I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)$.

$$
\begin{aligned}
I(\boldsymbol{Z};\boldsymbol{X_t}|\boldsymbol{X_b}) &:= \mathbb{E}_{\boldsymbol{x}_t,\boldsymbol{x}_b,\boldsymbol{z}}\left[\log\left(\frac{p(\boldsymbol{z},\boldsymbol{x_t}|\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_b})p(\boldsymbol{x_t}|\boldsymbol{x_b})}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log\left(\frac{1}{\frac{p(\boldsymbol{z}|\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_t},\boldsymbol{x_b})}|\mathcal{M}|}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log\left(\frac{1}{\frac{p(\boldsymbol{z}|\boldsymbol{x}_b)}{p(\boldsymbol{z}|\boldsymbol{X})}|\mathcal{M}|\mathbb{E}_{M^*\in\mathcal{M}}\left[\frac{p(\boldsymbol{z}|\boldsymbol{x}_t^*,\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_b})}\right]}\right)\right] + \log(|\mathcal{M}|) \\
&\approx \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log\left(\frac{1}{\frac{p(\boldsymbol{z}|\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_t},\boldsymbol{x_b})}\sum_{M^*\in\mathcal{M}}\frac{p(\boldsymbol{z}|\boldsymbol{x}_t^*,\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_b})}}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log\left(\frac{\frac{p(\boldsymbol{z}|\boldsymbol{x_t},\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_b})}}{\sum_{M^*\in\mathcal{M}}\frac{p(\boldsymbol{z}|\boldsymbol{x}_t^*,\boldsymbol{x_b})}{p(\boldsymbol{z}|\boldsymbol{x_b})}}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x_t},\boldsymbol{x_b},\boldsymbol{z}}\left[\log\left(\frac{\frac{p(\boldsymbol{z}|\boldsymbol{x_t},\boldsymbol{x_b})}{p(z)}}{\sum_{M^*\in\mathcal{M}}\frac{p(\boldsymbol{z}|\boldsymbol{x}_t^*,\boldsymbol{x_b})}{p(z)}}\right)\right] + \log(|\mathcal{M}|) \\
&= \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}\left[\log\left(\frac{h(\boldsymbol{x},\boldsymbol{z})}{\sum_{M^*\in\mathcal{M}} h(\boldsymbol{x}^*,\boldsymbol{z})}\right)\right] + \log(|\mathcal{M}|) \\
&\equiv -\mathcal{L}_{\text{CORRO}}.
\end{aligned}
$$

3. $\mathcal{L}_{\text{CSRO}} \geq (\lambda-1)I(\boldsymbol{Z};\boldsymbol{X}) - \lambda I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)$.

   The CLUB loss in Eqn 6 operates as a upper bound of $I(\boldsymbol{Z};\boldsymbol{X}_b)$ [31]. By Eqn 5 we have

$$
\begin{aligned}
\mathcal{L}_{\text{CSRO}} &= \mathcal{L}_{\text{FOCAL}} + \lambda L_{\text{CLUB}} \\
&\geq \mathcal{L}_{\text{FOCAL}} + \lambda I(\boldsymbol{Z};\boldsymbol{X}_b) \\
&= \mathcal{L}_{\text{FOCAL}} + \lambda \underbrace{\left(I(\boldsymbol{Z};\boldsymbol{X}) - I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)\right)}_{\text{chain rule of mutual information}} \\
&\overset{1}{=} -I(\boldsymbol{Z};\boldsymbol{X}) + \lambda(I(\boldsymbol{Z};\boldsymbol{X}) - I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b)) \\
&= (\lambda-1)I(\boldsymbol{Z};\boldsymbol{X}) - \lambda I(\boldsymbol{Z};\boldsymbol{X}_t|\boldsymbol{X}_b).
\end{aligned}
$$

□

B.2 Proof of Theorem 2.4

Proof. Denote by $n_Z := \sum_{i=1}^{n_M} |M^i||X^i|$ the number of context samples, we have

$$
\begin{aligned}
I(\boldsymbol{Z};\boldsymbol{M}) &= H(\boldsymbol{Z}) - H(\boldsymbol{Z}|\boldsymbol{M}) && (16) \\
&\simeq -\sum_{i=1}^{n_Z}\log p(\boldsymbol{z}_i) - \mathbb{E}_M\left[H(\boldsymbol{Z}|\boldsymbol{M}=M)\right] && (17) \\
&\simeq -\sum_{i=1}^{n_Z}\log p(\boldsymbol{z}_i) - \sum_{j=1}^{n_M} H(\boldsymbol{Z}|\boldsymbol{M}=M^j). && (18)
\end{aligned}
$$

Since $n_Z \gg n_M$, the concentration characteristic of $I(\boldsymbol{Z};\boldsymbol{M})$ is dominated by the second term. If we ignore the approximation error of the first term, by the Chebyshev's inequality, for any $\epsilon > 0$,

$$\Pr\left(\left|\hat{I}(\boldsymbol{Z};\boldsymbol{M}) - \bar{I}(\boldsymbol{Z};\boldsymbol{M})\right| \geq \epsilon\right) \leq \frac{\mathrm{Var}(H(\boldsymbol{Z}|\boldsymbol{M}))}{n_M \epsilon^2}. \tag{19}$$

Or equivalently, with probability at least $1-\delta$,

$$\left|\hat{I}(\boldsymbol{Z};\boldsymbol{M}) - \bar{I}(\boldsymbol{Z};\boldsymbol{M})\right| \leq \sqrt{\frac{\mathrm{Var}(H(\boldsymbol{Z}|\boldsymbol{M}))}{n_M \delta}}. \tag{20}$$

Remark: Consider a practical implementation of $p(\boldsymbol{z}|M)$ as a multivariate Gaussian $\mathcal{N}(\boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M)$, which gives $H(\boldsymbol{Z}|\boldsymbol{M}) = \frac{1}{2}\log[(2\pi e)^{\dim Z}|\boldsymbol{\Sigma}_M|]$ [36]. Substituting into Eqn 20, we have with probability at least $1-\delta$,

$$\left|\hat{I}(\boldsymbol{Z};\boldsymbol{M}) - \bar{I}(\boldsymbol{Z};\boldsymbol{M})\right| \leq \sqrt{\frac{\mathrm{Var}(\log|\boldsymbol{\Sigma}_M|)}{2 n_M \delta}}. \tag{21}$$

□

C Further Experiments

C.1 UNICORN-SS-0: A Label-free Version

When $\alpha = 0$, UNICORN-SS task representation learning reduces to the minimization of the reconstruction loss only, which becomes a *label-free* algorithm (i.e., it does not require the knowledge of task identities/labels to optimize, as opposed to the contrastive learning in previous methods). We name this special case UNICORN-SS-0, which is equivalent to a concurrent method GENTLE [41]. For a fair comparison, we use a variant of a Bayesian OMRL method BOReL [14] that does not utilize oracle reward functions, as a label-free baseline. At test time, we modify BOReL to use offline datasets rather than the online data collected by interacting with the environment to infer the task information[2]. As shown in Table 5, UNICORN-SS-0 is also competitive with BOReL.

Table 5: UNICORN-SS-0 compared to another label-free COMRL method, BOReL, on Ant-Dir.

Algorithm	Ant-Dir	
	IID	OOD
UNICORN-SS-0	220±16	200±9
BOReL	206±18	187±10

To further validate that the KL constraint $D_{KL}(q_\theta(z|x)|\mathcal{N}(0,I))$ is unnecessary, we weight the KL constraint to UNICORN-SS-0. Table 6 shows that KL constraint would reduce the performance under the offline setting.

C.2 Visualization of Task Embeddings

For better interpretation, we visualize the task representations by 2D projection of the embedding vectors via t-SNE [48]. For each test task, we generate a trajectory using each policy in $\{\pi_\beta^{i,t}\}_{i,t}$ and infer their task representations $\boldsymbol{z}$. This setting aligns with our OOD experiments in the main text.

Since the UNICORN-SS objective operates as a convex combination of the reconstruction loss (UNICORN-SS-0) and the FOCAL loss, we compare representations of UNICORN-SS to these

[2]Experiment shows that the usage of offline dataset rather than online data has little effect on the results of BOReL.

Table 6: UNICORN-SS-0 compared to other KL constraint weight variants.

Algorithm	Ant-Dir	
	IID	OOD
UNICORN-SS-0	220±16	200±9
KL-0.1	215±11	189±12
KL-0.5	202±17	182±13
KL-1.0	194±11	161±7
KL-5.0	162±14	143±10

two extremes. As shown in Figure 6, compared to UNICORN-SS-0, UNICORN-SS provides evidently more distinguishable task representations for OOD data, emphasizing the necessity of joint optimization with the FOCAL loss. On the other hand, despite the seemingly higher quality of task representations of FOCAL, the performance of FOCAL is much worse than UNICORN-SS. We speculate that since FOCAL blindly separates embeddings from different tasks, it may fail to capture shared structure between similar tasks, whereas UNICORN-SS is able to do so by generative modeling via the reconstruction loss.

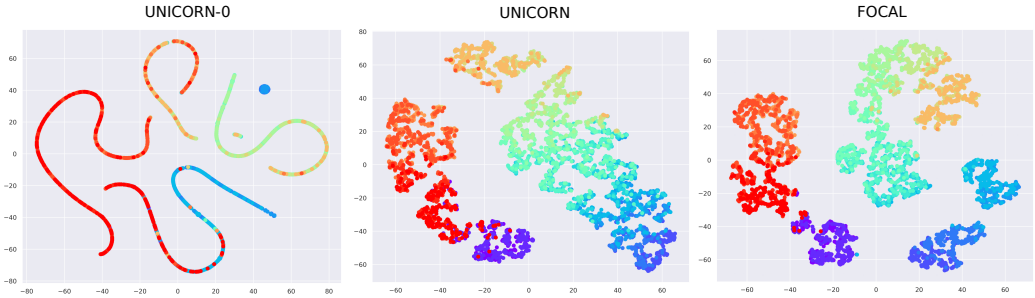

Figure 6: The 2D projection of the learned task representation space in Ant-Dir. Points are uniformly sampled from out-of-distribution data. Tasks of given goals from 0 to 6 are mapped to rainbow colors, ranging from purple to red.

C.3 More on UNICORN-DT

We provide the learning curves of the results in Section 4.1 here. As shown in Figure 7, UNICORN-SS-DT and UNICORN-SUP-DT demonstrate a faster convergence speed compared to vanilla UNICORN-SS and a higher asymptotic performance compared to vanilla Prompt-DT.

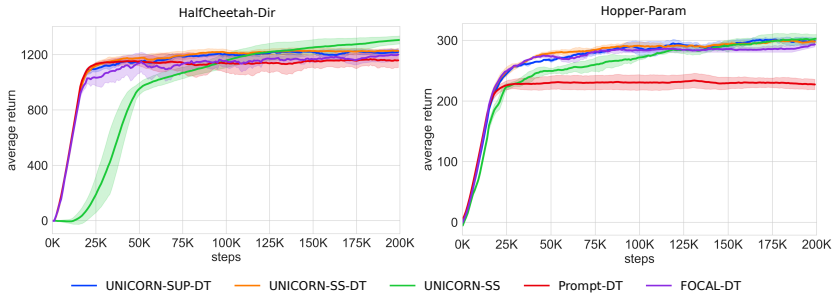

Figure 7: Test returns of UNICORN-SUP-DT and UNICORN-SS-DT against UNICORN-SS, Prompt-DT and FOCAL-DT on 2 benchmarks. The learning curve is averaged by 6 random seeds.

C.4 Ablation Study

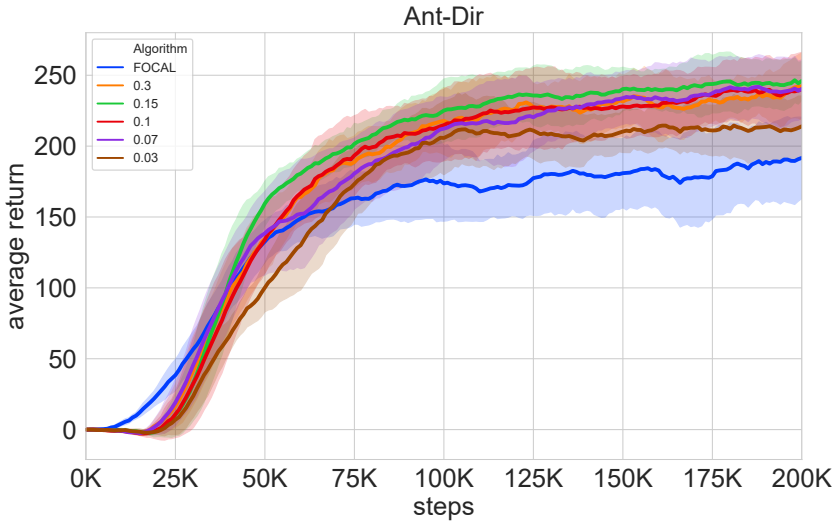

Figure 8: Different hyper-parameter settings of $\frac{\alpha}{1-\alpha}$ on Ant-Dir. The learning curve is averaged by 6 random seeds.

To illustrate the effect of hyper-parameter $\frac{\alpha}{1-\alpha}$ on the asymptotic performance, we set the following ablation study. Figure 8 shows that as $\frac{\alpha}{1-\alpha}$ increases, the performance gradually increases but when it goes excessively the performance would decrease (FOCAL means $\frac{\alpha}{1-\alpha} \to \infty$), which validates our proposed theory. We also find that UNICORN-SS can maintain relatively stable performance over a range of $\frac{\alpha}{1-\alpha}$ values, thus alleviating the exhaustion from parameter-tuning.

D Experimental Details

Table 7 lists the necessary hyper-parameters that we used to produce the experimental results.

Table 7: Configurations and hyper-parameters used in the training process.

Configurations	Ant-Dir	HalfCheetah-Dir	HalfCheetah-Vel	Hopper-Param	Walker-Param	Reach
dataset size	1e5	2e5	2e5	3e5	4.5e5	1e5
task representation dimension	5	5	5	40	32	2
weight $\frac{\alpha}{1-\alpha}$	0.15	0.15	0.15	1.5	1.5	0.3
training steps	200k					
task batch size	16					
RL batch size	256					
context training size	1 trajectory (200 steps)					1 trajectory (500 steps)
learning rate	3e-4					
RL network width	256					
RL network depth	3					
encoder width	64			128		64

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, the main contributions are clearly laid out at the end of introduction. See Section 2 for our theory and Section 3 for our experimental verification.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, see Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Yes, see for example Section 2.2 and Appendix B for a complete proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Yes, see Appendix D for experimental details. Source code is provided in the Supplementary Material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Yes, see Supplementary Material for the source code and README instructions.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes, see for example Appendix C.4 for hyperparameter study.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Yes, all figures and tables contain error bars.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See README in Supplementary Material for computing environment information.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We make sure to preserve anonymity and conform to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The societal/general impacts of offline meta-RL are discussed in Section 1.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We only use open-source benchmarks for experiments: MuJoCo (Apache-2.0 license) and Meta-World (MIT license).

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subject.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.