MR-CRL: Leveraging Predictive Representations for Contrastive Goal-Conditioned Reinforcement Learning

Muhammad Qasim Ali, Winnie Trandinh,
Howard Nguyen-Huu, Alexander Wong

Keywords: Contrastive reinforcement learning, model-based reinforcement learning, representation learning, self-supervised learning, contrastive learning, goal-conditioned reinforcement learning

Summary

Goal-conditioned reinforcement learning (GCRL) aims to train agents capable of achieving arbitrary goals, a task made significantly harder in offline settings where rewards and environment interaction are unavailable. Contrastive Reinforcement Learning (CRL) is a goal-conditioned framework that learns value functions through contrastive objectives, enabling effective policy learning from offline datasets without reward labels or environment interaction. In parallel, model-based reinforcement learning (MBRL) has shown that learning predictive representations of environment dynamics can significantly improve policy performance and sample efficiency. While both approaches learn features that anticipate future states, their integration remains underexplored. In this work, we investigate whether model-based predictive representations can enhance CRL's similarity-based value estimation. We propose Model-based Representations for Contrastive Reinforcement Learning (MR-CRL), a simple extension that augments CRL with predictive state and dynamics encoders trained using a novel cross-entropy loss objective over latent dynamics predictions. We evaluate multiple integration strategies within the CRL architecture and find that MR-CRL outperforms the original CRL baseline on 4 out of 18 tasks in the OGBench benchmark, with significant gains in both low- and high-dimensional environments. While gains are not universal, our results suggest that model-based inductive biases can enhance training goal-reaching on some tasks.

Contribution(s)

1. We propose MR-CRL, a simple extension to contrastive reinforcement learning that integrates model-based predictive representations into the critic architecture.
   Context: Context: MR-CRL draws on well-established insights from two distinct reinforcement learning paradigms: model-based RL and contrastive RL. We investigate the utility of incorporating model-based representations into a contrastive, model-free setting.
2. We introduce a cross-entropy loss for training predictive state and dynamics representations, drawing on techniques from self-supervised and model-based learning.
   Context: Context: Prior work in model-based representation learning for model-free reinforcement learning primarily uses L2 reconstruction losses. Our proposed loss, inspired by cross-entropy objectives in self-supervised and model-based RL literature, improves training stability and representation quality.
3. We evaluate multiple integration strategies for using model-based features within CRL's actor and critic networks and analyze their tradeoffs across tasks.
   Context: Context: Our ablation study helps clarify how predictive state and state-action embeddings influence contrastive value learning.
4. We show that MR-CRL improves over CRL in 4 out of 18 tasks in the OGBench benchmark, particularly in low-dimensional and structured settings.
   Context: Context: While improvements are not universal, results suggest that model-based inductive biases can benefit contrastive goal-conditioned RL in specific domains.

MR-CRL: Leveraging Predictive Representations for Contrastive Goal-Conditioned Reinforcement Learning

Muhammad Qasim Ali[1,†], **Winnie Trandinh**[2,†],
Howard Nguyen-Huu[1], **Alexander Wong**[1]
m45ali@uwaterloo.ca, trandint@mcmaster.ca,
{howard.nguyen-huu,a28wong}@uwaterloo.ca

[1]**Department of Systems Design Engineering, University of Waterloo, Canada**
[2]**Department of Computing and Software, McMaster University, Canada**

[†] Denotes equal contribution

Abstract

Goal-conditioned reinforcement learning (GCRL) aims to train agents capable of achieving arbitrary goals, a task made significantly harder in offline settings where rewards and environment interaction are unavailable. Contrastive Reinforcement Learning (CRL) is a goal-conditioned framework that learns value functions through contrastive objectives, enabling effective policy learning from offline datasets without reward labels or environment interaction. In parallel, model-based reinforcement learning (MBRL) has shown that learning predictive representations of environment dynamics can significantly improve policy performance and sample efficiency. While both approaches learn features that anticipate future states, their integration remains underexplored. In this work, we investigate whether model-based predictive representations can enhance CRL's similarity-based value estimation. We propose Model-based Representations for Contrastive Reinforcement Learning (MR-CRL), a simple extension that augments CRL with predictive state and dynamics encoders trained using a novel cross-entropy loss objective over latent dynamics predictions. We evaluate multiple integration strategies within the CRL architecture and find that MR-CRL outperforms the original CRL baseline on 4 out of 18 tasks in the OGBench benchmark, with significant gains in both low- and high-dimensional environments. While gains are not universal, our results suggest that model-based inductive biases can enhance training goal-reaching on some tasks.

1 Introduction

Goal-conditioned reinforcement learning (GCRL) seeks to train agents capable of reaching arbitrary goal states, enabling general-purpose policies across many tasks. When learning from offline datasets—without reward labels or online environment interaction—this problem becomes especially challenging.

Contrastive Reinforcement Learning (CRL), introduced by Eysenbach et al. (2022), offers an elegant solution to this challenge by learning state representations through contrastive objectives that directly encode goal-reaching behavior. Rather than relying on traditional temporal difference learning Sutton (1988), CRL trains a critic to discriminate between state-action pairs that lead to desired future states versus those that do not. The key insight is to parameterize the critic as an inner product between learned state-action embeddings and goal embeddings, creating a similarity metric that naturally captures goal-conditioned value functions. This approach enables agents to navigate to specified future states by leveraging the learned associations between current state-action pairs and their reachable outcomes.

Meanwhile, model-based reinforcement learning has demonstrated significant advances in sample efficiency and performance by learning predictive representations of environment dynamics. Methods such as Dreamer (Hafner et al., 2024) and SALE (Fujimoto et al., 2023) show that training neural networks to predict future states and representations can inject valuable inductive biases into policy learning. These predictive models capture temporal structure and controllable aspects of the environment, leading to more informed decision-making and improved generalization.

Notably, both CRL and MBRL learn representations that anticipate future states—yet these frameworks have remained largely disconnected. While CRL learns to associate state-action pairs with future goal states through contrastive objectives, model-based methods learn explicit predictive models of state transitions.

In this work, we investigate whether model-based predictive representations can enhance goal-conditioned policy learning within the CRL framework. We hypothesize that these predictive embeddings, which capture structured and temporally coherent information through dynamics prediction, will offer a beneficial inductive bias for CRL's similarity-based value functions.

We propose Model-based representations for Contrastive Reinforcement Learning (MR-CRL), a contrastive reinforcement learning framework enhanced with model-based representations. Our key contributions are:

- We introduce a novel training objective for model-based representation learning, using cross-entropy loss over latent dynamics predictions. This approach is inspired by insights from self-supervised learning methods (Grill et al., 2020; Oquab et al., 2023) and discrete embeddings in modern MBRL works (Hafner et al., 2024). Our proposed loss increases training stability of the model-based representations.
- We investigate multiple strategies for integrating learned state and dynamics embeddings into CRL's actor and critic networks, enabling richer goal-conditioned value estimation.
- We demonstrate that MR-CRL outperforms the original CRL baseline on 4 out of 18 tasks in the OGBench benchmark, with substantial improvements in both low-dimensional and high-dimensional environments.

2 Related Works

2.1 Representation Learning

The goal of representation learning is to learn a mapping from the input space to the latent space such that the mapping can be generalized to unseen data. Strong interest has risen especially in self-supervised methods with the promise of training on primarily unlabeled data. One such approach is done through autoencoders such as Masked Autoencoders (He et al., 2022) which learns to reconstruct the original input. Alternatively, the input can be randomly augmented to provide two variations of the same input, one of which is passed to the online network and the other to the target network, as is done in BYOL (Grill et al., 2020). The model is then trained to minimize the difference in features between the online and target network.

DINO is another effective method of representation learning in images that combines label-free knowledge distillation with self-supervised learning (Caron et al., 2021). Containing an architecturally identical student network and a teacher network, the student learns to match the teacher network's output by minimizing the cross-entropy loss, while the teacher's weights are updated through an exponential moving average (EMA) of the student's weights. To encourage local-global correlations, a multi-crop strategy is performed on the single input image to produce two global crops and several local crops. Only the two global crops are provided to the teacher, whereas the student receives all the image crops. DINO is then extended through the addition of iBOT losses (Zhou et al., 2022) and a more efficient implementation in DINOv2 (Oquab et al., 2023).

A different learning paradigm is done through contrastive learning. Instead of only augmenting the same image to extract positive samples, contrastive learning techniques such as MoCo and SimCLR uses both positive samples and negative samples, where the negative samples comes from different images (He et al., 2020; Chen et al., 2020). The features are learnt such that the similarity in features between positive samples are minimized, while the distance to negative samples are maximized. CLIP then extends this by combining text with images by training on image+caption pairs (Radford et al., 2021).

2.2 Reinforcement Learning

Reinforcement learning can be broadly split into model-based and model-free methods. The model-based method similar to our proposed method is the use of world models, first proposed by Ha & Schmidhuber (2018) and further developed by the Dreamer series of models (Hafner et al., 2019; 2020; 2024), which aims to learn a dynamics model that is then used to perform imaginary rollouts without the need to interact with the environment. TD-MPC extends this by learning the dynamics of a learnt latent space instead of operating directly on the pixel space as done by Dreamer (Hansen et al., 2022).

In contrast, a model-free approach that our work extends is Contrastive RL (CRL) (Eysenbach et al., 2022). The underlying principle behind CRL is to provide a method of learning features that can directly perform goal-conditioned RL, in contrast to decoupling the representation learning and goal-conditioned RL. This is accomplished by employing an actor critic method, where the critic is parameterized by two representations whose inner product represents the value function. Further work demonstrates that providing a single goal to CRL is sufficient for it to perform goal-conditioned RL, without the need for any intrinsic or extrinsic rewards (Liu et al., 2024).

2.3 Representation Learning in RL

Representation learning has been applied to RL in order to produce a latent space that the policy can more easily learn in. Methods of creating these state representations include using autoencoders (Finn et al., 2016) or through contrastive learning objectives (Laskin et al., 2020). SALE takes a different approach and learns both the state and state-action embeddings, thus providing representations of both the observation space and the dynamics model (Fujimoto et al., 2023). These embeddings are then appended to the input state and action. In contrast to world model methods, the embeddings are only used to improve the input to the value function, as opposed to using them for imaginary rollouts. This work is then extended in MR.Q, which foregoes the input state and action into the value function while incorporating the reward and termination losses into the learning of the state and state-action embeddings (Fujimoto et al., 2025).

3 Background

3.1 Model-Free Reinforcement Learning

Reinforcement learning (RL) is a framework in which an agent interacts with an environment $\mathcal{M}$, making sequential decisions to maximize cumulative reward r. At each time step t, the agent observes a state $s_t \in \mathcal{S}$, selects an action $a_t \in \mathcal{A}$, receives a reward $r(s_t, a_t)$, and transitions to a new state s_{t+1} according to the dynamics of the environment. The goal of the agent is to learn a policy $\pi(a|s)$ that maximizes expected discounted return over time.

Among the many classes of RL algorithms, this work focuses on *actor-critic* methods (Konda & Tsitsiklis, 1999). In these methods, a critic network estimates the value of a state-action pair, typically in the form of a Q-function:

$$Q(s,a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \,\middle|\, s_0 = s,\ a_0 = a\right], \tag{1}$$

where γ is a discount factor and the expectation is taken over trajectories induced by the policy and environment dynamics. The actor network uses the critic network to update the policy in a direction that increases expected returns.

In our work, we adopt the Deterministic Deep Policy Gradient (DDPG) algorithm (Lillicrap et al., 2019), a model-free actor-critic method for continuous control. The actor is represented by a deterministic policy $\pi(s;\theta^\pi)$, which maps states to actions. The actor is trained with loss:

$$\mathcal{L}_{\text{actor}}(\theta^\pi) = -\mathbb{E}_{s\sim\mathcal{D}}\left[Q(s, \pi(s;\theta^\pi))\right], \tag{2}$$

where $\mathcal{D}$ is the replay buffer. This loss encourages the actor to choose actions that maximize the critic's predicted value, i.e., the expected return.

3.2 Model-Based Representation Learning for Model-Free Reinforcement Learning

A growing body of work has investigated how predictive representations, learned via model-based objectives, can benefit model-free actor-critic algorithms (Fujimoto et al., 2023; 2025). These approaches aim to inject structure and temporal coherence into the learned embeddings, improving generalization and sample efficiency without explicitly planning over future trajectories.

In these approaches, two neural networks are used to encode latent dynamics: a state encoder $z_1 : \mathcal{S} \to \mathbb{R}^d$ that maps raw observations to compact representations, and a state-action encoder $z_2 : \mathbb{R}^d \times \mathcal{A} \to \mathbb{R}^d$ that predicts the next state's embedding from the current state representation $z_1(s)$ and action a. These encoders are trained with a dynamics reconstruction loss:

$$\mathcal{L}_{\text{dynamics}}(s_{t+1}, s_t, a_t) = \left\|\text{sg}(z_1'(s_{t+1})) - z_2(z_1(s_t), a_t)\right\|^2, \tag{3}$$

where sg denotes stopgrad and z_1' is a target encoder whose parameters are updated periodically every p iterations. This loss encourages z_2 to produce accurate predictions of the future latent state. This structured learning objective encourages representations that capture controllable aspects of the environment and are predictive of future states. It promotes temporally coherent representations that model the environment's dynamics without explicitly constructing a generative model.

To capture long-range and self-consistent representations, the final loss is summed over a rolled-out trajectory of length H:

$$\mathcal{L}_{\text{model}} = \sum_{h=1}^{H} \mathcal{L}_{\text{dynamics}}(s_{t+h}, s_{t+h-1}, a_{t+h-1}) \tag{4}$$

Once trained, the encoders can be used to augment the actor and critic networks in various ways. For the actor, one can condition the policy on both the raw state and its representation, i.e., $\pi(a \mid s, z_1(s))$, or even use the representation alone, $\pi(a \mid z_1(s))$. For the critic, value functions can be defined using a combination of state and action embeddings, $Q(s, a, z_1(s), z_2(z_1(s), a))$ or $Q(z_2(z_1(s), a))$ depending on whether raw inputs are retained. We explore several such architectural variants in Section 5.2.

3.3 Goal-Conditioned Reinforcement Learning

In goal-conditioned reinforcement learning (GCRL), the agent is tasked with reaching a specific goal state $s_g \in \mathcal{S}$. The environment is defined by states $s_t \in \mathcal{S}$, actions $a_t \in \mathcal{A}$, initial state distribution $p_0(s)$, transition dynamics $p(s_{t+1} \mid s_t, a_t)$, and a goal distribution $p_g(s_g)$. Each goal defines a different task, making GCRL a form of multi-task RL (Veeriah et al., 2018; Schaul et al., 2015).

We adopt a goal-reaching reward defined by the likelihood of arriving at the goal in the next time step: $r_g(s_t, a_t) = (1-\gamma)\, p(s_{t+1} = s_g \mid s_t, a_t)$. Such a reward definition avoids the need for user-defined distance metrics and aligns with prior work on goal-reaching objectives (Andrychowicz et al., 2018; Pong et al., 2020; Eysenbach et al., 2022). The agent learns a goal-conditioned policy $\pi(a \mid s, s_g)$ that maximizes the expected return over sampled goals:

$$\max_{\pi} \mathbb{E}_{s_g \sim p_g,\, \tau \sim \pi(\cdot \mid s_g)} \left[\sum_{t=0}^{\infty} \gamma^t r_g(s_t, a_t) \right]. \tag{5}$$

We define the corresponding goal-conditioned Q-function as

$$Q^{\pi}_{s_g}(s,a) = \mathbb{E}_{\tau \sim \pi(\cdot \mid s_g)} \left[\sum_{t'=t}^{\infty} \gamma^{t'-t} r_g(s_{t'}, a_{t'}) \mid s_t = s, a_t = a \right]. \tag{6}$$

This formulation allows learning policies that are capable of reaching and remaining at arbitrary goal states, using shared experience across many goals.

3.4 Contrastive Reinforcement Learning

Contrastive reinforcement learning (CRL) is a goal-conditioned reinforcement learning framework that learns value functions using contrastive objectives derived from self-supervised representation learning (Eysenbach et al., 2022). Rather than relying on traditional bootstrapped temporal difference (TD) targets, CRL reframes value estimation as a discriminative task: identifying whether a given state-action pair (s, a) leads to a specific goal state g in the future.

At the core of CRL is a critic function that measures the compatibility between state-action pairs and goal states. This critic is parameterized as a dot product between two learned representations:

$$f(s,a,g) = \phi(s,a)^{\top} \psi(g), \tag{7}$$

where $\phi(s,a)$ encodes the state-action pair and $\psi(g)$ encodes the goal. This similarity score serves as an unnormalized proxy for the Q-function:

$$Q^{\pi}_{g}(s,a) \propto \exp(f(s,a,g)). \tag{8}$$

The critic is trained via a contrastive binary classification loss, which encourages high similarity between state-action pairs and future goal states they actually reach (positive pairs), and low similarity for mismatched pairs (negative samples). This loss, known as the NCE-binary objective (Ma & Collins, 2018), is defined as:

$$\mathcal{L}_{\text{contrastive}} = \log \sigma\left(\phi(s,a)^{\top}\psi(s_f^{+})\right) + \log\left(1 - \sigma\left(\phi(s,a)^{\top}\psi(s_f^{-})\right)\right), \tag{9}$$

where s_f^{+} is a goal actually reached in the future from (s, a) and s_f^{-} is a randomly sampled goal state. Minimizing this loss encourages the critic to act as a probabilistic classifier that captures goal-reaching likelihoods under the current policy.

The actor is trained using the learned critic to select actions that increase the probability of reaching a desired goal. We adopt a deterministic actor, trained with a modified DDPG-style objective. Given the offline nature of the training setting, we add a behavior cloning loss to ensure policy stability. The full actor loss is:

$$\mathcal{L}_{\text{actor}} = -\mathbb{E}_{s,g}\left[f(s, \pi(s, g), g)\right] + \alpha\, \mathbb{E}_{s,a}\left[\|\pi(s, g) - a\|^2\right], \tag{10}$$

where α is a weighting coefficient for the behavior cloning regularization.

This contrastive formulation leads to a simple yet effective actor-critic algorithm for goal-reaching tasks. Unlike traditional TD-based RL, CRL requires no value bootstrapping, target networks, or auxiliary rewards, and is naturally suited for learning from static offline datasets.

4 Methodology

Our method trains state encoders to learn predictive representations with model-based losses, and uses the state encoder features within the critic networks of CRL. Prior works has proven that the learned Q-function of CRL is equivalent to the state occupancy measure (Eysenbach et al., 2022), and that by learning the proxy to the Q-function, the Markovian dynamics can also be learnt (Mazoure et al., 2023). Eysenbach et al. (2022) demonstrates the ability of CRL to learn non-linear environment dynamics within a simple visual maze-like environment. However, we hypothesize that using explicit predictive representations can help the critic learn more informative similarity scores between state-action pairs and future goal states using fewer training steps, thus enhancing the performance of CRL in offline settings.

4.1 Training the Encoder for Model-Based Representations

We incorporate the state encoders $z_1 : \mathcal{S} \to \mathbb{R}^d$ that maps raw observations to latent features, and a state-action encoder $z_2 : \mathbb{R}^d \times \mathcal{A} \to \mathbb{R}^d$ that predicts the future latent state from the current encoded state and action.

Our experiments found that training state encoders with L2 loss, as in the prior work of Fujimoto et al. (2023; 2025), did not yield significant gains over the CRL baseline. Furthermore, the state encoder losses (Eq. 4) were non-smooth and non-monotonic, indicating that the network was struggling to learn the dynamics. This is further explored within Section 5.2. We instead propose using cross-entropy (CE) loss to learn more effective representations. CE is advantageous for two reasons: (1) In self-supervised learning, especially in teacher-student frameworks like DINO and BYOL (Caron et al., 2021; Grill et al., 2020; Oquab et al., 2023; Hinton et al., 2015), CE encourages the student to match the teacher's semantic structure, yielding richer, transferable features. (2) In model-based RL, DreamerV3 (Hafner et al., 2024) uses discrete latents and KL-divergence (a form of CE) to enhance stability and generalization. These findings suggest CE promotes structure and information retention, making it well-suited for learning predictive encoders in RL.

Let $m(\cdot)$ denote a projection head, and τ_s, τ_t be the respective temperatures for the student and teacher outputs. The L2 dynamics loss (Eq. 3) is replaced by CE:

$$\mathcal{L}_{dynamics} = \text{CE}\left(\text{sg}(\text{softmax}(m'(z_1'(s_{t+1})/\tau_t))), \text{softmax}(m(z_2(z_1(s_t), a_t)/\tau_s))\right), \tag{11}$$

where m', z_1' denote slowly updated target networks and sg denotes stopgrad. Similar to prior model-based representation works, this loss is computed over horizon segments of length H to enforce temporal consistency.

4.2 Contrastive RL with Learned Encoders

To exploit these representations, we redefine the critic used in contrastive RL. Inspired by Fujimoto et al. (2023), we append the predictive features from the learned encoders as additional inputs to the existing critic encoders of CRL. The critic function $f(s, a, g)$ is now expressed as:

$$f(s, a, g) = \phi(s, a, z_1'(s), z_2'(z_1'(s), a))^\top \psi(g), \tag{12}$$

where ϕ, the critic's state-action encoder, is a function of the state, action, and their predictive embeddings computed using frozen target encoders z_1' and z_2', and $\psi(g)$ is the critic's goal encoder.

Since the critic encoder ϕ is trained to align with future goal states reachable from (s, a), it benefits from incorporating model-based features—specifically, the predictive embedding $z_2(z_1(s), a)$ and the compact state representation $z_1(s)$. Together, these components provide temporally rich information about future dynamics and concise, informative representations of the current state. Their effectiveness is validated in Section 5.1.

In the ablations in Section 5.2, we explored how the inclusion of model-based representations affects other components of the architecture—namely the goal encoder ψ and the policy network π. We also tested the effect of omitting $z_1(s)$ altogether. Our results indicate that while incorporating these changes can lead to performance gains on several tasks, they can also degrade performance on others. This highlights an important tradeoff: although model-based features introduce useful inductive bias, their utility can vary across environments depending on the nature of the task and the quality of the learned representations.

The actor and critic networks are trained following the standard CRL framework, using the DDPG objective for the actor and the contrastive binary classification loss (Eq. 9) for the critic.

4.3 Training Procedure

Networks	
Encoders	
State encoder	$z_1(s)$
State-action encoder	$z_2(a, z_1(s))$
Projector	$m(\cdot)$
Contrastive RL actor-critic	
Critic goal	$\psi(g)$
Critic state-action	$\phi(s, a, z_1(s), z_2(z_1(s), a))$
Policy	$a \sim \pi(s, g)$

Training Loop

for t in $1 : T$
  Sample batch (s, a, r, s', g) from replay buffer.
  Update encoder networks z_1, z_2, m with dynamics loss (Eq. 11).
  Update critic networks ϕ, ψ with contrastive loss (Eq. 9).
  Update policy $\pi(s, g)$ with actor loss (Eq. 2).
  if $t\%p = 0$ *then*
    Update target networks:
      $z_1', z_2', m' \leftarrow z_1, z_2, m$

Algorithm 1: Networks and training loop describing MR-CRL.

Algorithm 1 outlines the training procedure. At each training step, the encoders and CRL actor-critic networks are updated with a training batch. The weights of the frozen encoder target networks are copied every p training steps, similarly to Fujimoto et al. (2023).

5 Experiments

Benchmark We evaluate our method on the OGBench benchmark (Park et al., 2025), a diverse suite designed for offline goal-conditioned reinforcement learning. OGBench includes tasks that require long-horizon planning, stitching subgoals, and reasoning over noisy or suboptimal data, providing a comprehensive testbed for general-purpose policy learning. Our experiments span 18 tasks drawn from six environment families, covering both state-based locomotion and manipulation domains.

Specifically, we use three categories of manipulation tasks: Puzzle, which requires solving Lights Out-style grid puzzles via robot-arm button presses; Cube, which involves rearranging and stacking colored blocks; and Scene, which requires sequential manipulation of drawers, buttons, and other interactive objects. Each of these tasks features variants based on grid size or data type, such as *play* (expert-like) and *noisy* (highly noised expert trajectories) datasets. For locomotion, we use three maze-based navigation environments: PointMaze, AntMaze, and HumanoidMaze. These tasks involve controlling agents with increasing degrees of freedom—ranging from a 2D point mass to a quadrupedal ant and a 21-DoF humanoid robot—to reach goal locations in challenging mazes. Dataset variants include *navigate* (expert-like), *explore* (random), and *stitch* (disjoint segments), each presenting unique algorithmic challenges.

Baseline The main baseline we compare to is CRL (Eysenbach et al., 2022), as this work directly extends upon the original method.

Experimental Setup We set the behavior cloning weight to $\alpha = 0.15$ and the student temperature to $\tau_s = 1.0$. The teacher temperature τ_t is annealed using a cosine schedule, starting at 0.04 and increasing to 0.07 as done in Oquab et al. (2023). Following standard setup (Park et al., 2025), we train each agent for 1 million steps on offline datasets and evaluate performance over 20 episodes. Target networks are updated every $p = 250$ steps, and model-based representation losses are computed over rollout horizons of length $H = 15$, consistent with prior work (Fujimoto et al., 2023; 2025). A warmup period of 50k steps was included, where the target networks were updated every step without updating the actor and critic. All networks—z_1, z_2, ϕ, ψ, and π—are implemented as three-layer multilayer perceptrons (MLPs) with 512 hidden units per layer and GELU activations. We train using a batch size of 1024 and a learning rate of 10^{-4}.

5.1 Results

MR-CRL is compared against CRL as a baseline in Table 1. In the PointMaze environment, MR-CRL significantly outperforms the baseline with a score of 24 compared to the 0 from CRL. The

Environment	Dataset	CRL	MR-CRL
PointMaze	pointmaze-medium-stitch-v0	0 ± 1	$\mathbf{24 \pm 9}$
	pointmaze-large-stitch-v0	0 ± 0	$\mathbf{1 \pm 3}$
AntMaze	antmaze-large-navigate-v0	83 ± 4	84 ± 6
	antmaze-medium-stitch-v0	$\mathbf{53 \pm 6}$	32 ± 8
	antmaze-large-stitch-v0	$\mathbf{11 \pm 2}$	0 ± 0
	antmaze-medium-explore-v0	3 ± 2	$\mathbf{14 \pm 6}$
	antmaze-large-explore-v0	0 ± 0	$\mathbf{5 \pm 6}$
HumanoidMaze	humanoidmaze-medium-navigate-v0	60 ± 4	63 ± 7
	humanoidmaze-medium-stitch-v0	$\mathbf{36 \pm 2}$	20 ± 5
	humanoidmaze-large-stitch-v0	$\mathbf{4 \pm 1}$	0 ± 0
Cube	cube-single-play-v0	$\mathbf{19 \pm 2}$	0 ± 1
	cube-single-noisy-v0	$\mathbf{38 \pm 2}$	27 ± 4
	cube-double-noisy-v0	2 ± 1	2 ± 1
Scene	scene-play-v0	$\mathbf{19 \pm 2}$	0 ± 0
Puzzle	puzzle-3x3-play-v0	3 ± 1	$\mathbf{17 \pm 6}$
	puzzle-4x4-play-v0	0 ± 0	0 ± 0
	puzzle-3x3-noisy-v0	$\mathbf{30 \pm 6}$	4 ± 2
	puzzle-4x4-noisy-v0	0 ± 0	0 ± 0

Table 1: Results of experiments on OGBench datasets with standard deviations provided after the $\pm$ sign. Scores for CRL are taken directly from published work that uses eight seeds (Park et al., 2025), whereas scores for MR-CRL are averaged across four seeds. For methods whose difference in scores are statistically significant, as determined by a Welch's Test with significance level 0.05, the better score is **bolded**. MR-CRL outperforms CRL on 4/18 tasks but underperforms on 8/18 tasks.

improvement is explained through the consistently lower state-action encoder loss as shown in Figure 1. For a simple two-dimensional state space such as PointMaze, it is easy to learn the dynamics of the environment, as opposed to more complex higher-dimensional environments such as Cube and Puzzle. The strong state-action representations are then key to the performance improvement. Furthermore, this result shows that state and state-action embeddings are beneficial even for low-dimensional tasks, which was first proposed by Fujimoto et al. (2023).

Both CRL and MR-CRL perform comparably for antmaze-navigate, but differences appear between the stitch and explore variations, with CRL doing better on stitch and MR-CRL outperforming on explore. One interpretation of these results is that because the explore dataset contains random exploratory actions, it provides high coverage of the state-space which reduces the chances of out-of-distribution states during evaluation. This high coverage can then be captured in the state and state-action encoders, whereas CRL does not have an explicit mechanism to do so.

In the Cube and Scene environments, MR-CRL significantly underperforms CRL likely due to an overestimation of the state encoder which leads to an overconfident critic prediction. Thus, the learnt embeddings are negatively impacting the critic's ability to learn strong contrastive representations.

Finally, for the Puzzle environment, which is the highest dimensional environment, MR-CRL greatly improves on the baseline for the play variant but underperforms on the noisy variant. One possible explanation of this result can be that because the noisy variant explores more of the high-dimensional state-space, it takes longer to learn the more vast dynamics. In contrast, the play variant's trajectories are more confined to a subset of expert trajectories, thus providing a smaller, simplified state space. Future work should explore if increasing the training time produces better results for the noisy variants and other environments with complex dynamics.

5.2 Ablations

Ablations were conducted on where the state and state-action encoders are used, as well as applying an exponential moving average on the update of the state and state-action encoders. Furthermore, L2 loss on the encoders was tested as opposed to the cross entropy loss used in the other models, while keeping the rest of the model the same. The different ablations are outlined and named in Table 2, with the results shown in Table 3. Although the results show that no one method is consistently better across all environments, some models perform better at specific environments. The *No state encoder* model performs noticeably better on the PointMaze and HumanoidMaze environments, which may suggest that focusing purely on learning the dynamics of the environment through the state-action encoder may be beneficial even though the critic network is smaller. Meanwhile, the *EMA target* model produces strong and consistent performance on puzzle-3x3-play and puzzle-3x3-noisy, while all other methods exhibit a difference in score between the two variants. This may imply that the EMA encoder may help with stabilizing the training especially when the state space is extremely large. However, this model is unable to solve the PointMaze environment.

$z_1(s)$ on actor	$z_1(s)$ on ϕ	$z_2(z_1(s), a)$ on ϕ	$z_1(s)$ on ψ	EMA encoder	L2 loss	Model name
✗	✓	✓	✗	✗	✗	Baseline
✗	✗	✓	✗	✗	✗	No state encoder
✗	✓	✓	✓	✗	✗	Goal encoder
✓	✓	✓	✗	✗	✗	Actor encoder
✗	✓	✓	✗	✓	✗	EMA target
✗	✓	✓	✗	✗	✓	L2 loss

Table 2: Ablations conducted and the associated model names.

Dataset	Baseline	No state encoder	Goal encoder	Actor encoder	EMA target	L2 loss
pointmaze-medium-stitch-v0	24	**32**	26	1	0	7
antmaze-large-navigate-v0	84	60	**90**	88	88	86
antmaze-medium-stitch-v0	32	34	23	28	33	**45**
antmaze-medium-explore-v0	14	21	16	18	**22**	5
humanoidmaze-medium-navigate-v0	63	**80**	67	64	65	43
humanoidmaze-medium-stitch-v0	20	30	23	23	29	**47**
cube-single-noisy-v0	27	28	**37**	21	26	22
cube-double-noisy-v0	2	2	1	2	5	**7**
puzzle-3x3-play-v0	17	16	**25**	7	21	20
puzzle-3x3-noisy-v0	4	4	0	**22**	**22**	0

Table 3: Results of ablations. Aside from the baseline model which is averaged across four seeds, all results are performed using a single seed. The best score in each environment is **bolded**. Relative to the baseline, all methods improve and lose performance on some tasks.

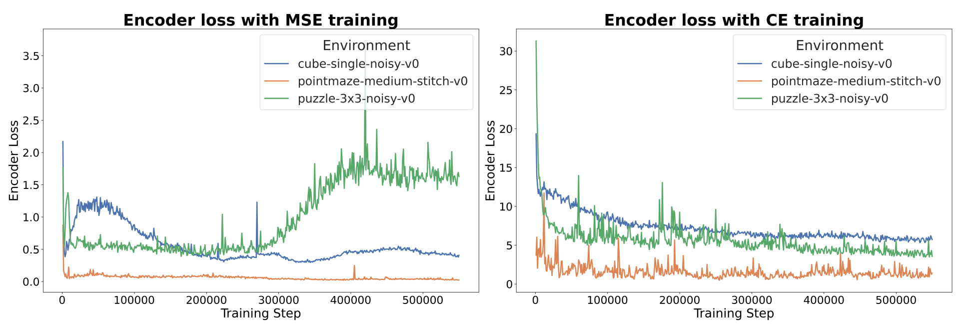

Figure 1: Comparison of the encoder loss (Eq. 4) curves when training encoders with L2 reconstruction loss (Eq. 3) versus the proposed CE loss (Eq. 11). The proposed CE loss yields much smoother loss curves with non-zero and non-saturating error.

With regards to the *L2 loss* model, it performs well on the antmaze-stitch and humanoid-stitch environments, but overall it does not outperform the baseline model and it only exceeds CRL on humanoid-medium-stitch. Figure 1 shows that the L2 encoder loss (Eq. 3) curve is much less smooth and monotonic than the CE encoder loss (Eq. 11) curve, showcasing the CE loss enhances training stability. Furthermore, the L2 encoder loss can degenerate to near-zero values on some tasks, while the CE loss stays non-zero.

In addition to ablations on the model architecture, some preliminary ablations were conducted on the training strategy. Initial tests showed that excluding the warmup period, where only the target networks are updated, caused a deterioration in the results. Future work will explore further ablations to the training strategy of the encoders, such as the scheduling of the target network updates.

6 Conclusion

We introduced MR-CRL, a simple yet effective extension to contrastive reinforcement learning that integrates model-based predictive representations. By training encoders with a novel cross-entropy loss and incorporating their outputs into the CRL architecture, we achieve improved performance on a subset of tasks in the OGBench benchmark. While not all tasks benefited from the representations, our results suggest that model-based inductive biases can enhance contrastive value learning.

References

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay, 2018. URL https://arxiv.org/abs/1707.01495.

Mathilde Caron, Hugo Touvron, Ishan Misra, Herv'e J'egou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021. URL https://api.semanticscholar.org/CorpusID:233444273.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

Benjamin Eysenbach, Tianjun Zhang, Sergey Levine, and Russ R Salakhutdinov. Contrastive learning as goal-conditioned reinforcement learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 35603–35620. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/e7663e974c4ee7a2b475a4775201ce1f-Paper-Conference.pdf.

Chelsea Finn, Xin Yu Tan, Yan Duan, Trevor Darrell, Sergey Levine, and Pieter Abbeel. Deep spatial autoencoders for visuomotor learning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 512–519. IEEE Press, 2016. DOI: 10.1109/ICRA.2016.7487173. URL https://doi.org/10.1109/ICRA.2016.7487173.

Scott Fujimoto, Wei-Di Chang, Edward Smith, Shixiang (Shane) Gu, Doina Precup, and David Meger. For sale: State-action representation learning for deep reinforcement learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 61573–61624. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/c20ac0df6c213db6d3a930fe9c7296c8-Paper-Conference.pdf.

Scott Fujimoto, Pierluca D'Oro, Amy Zhang, Yuandong Tian, and Michael Rabbat. Towards general-purpose model-free reinforcement learning, 2025. URL https://arxiv.org/abs/2501.16142.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020. URL https://arxiv.org/abs/2006.07733.

David Ha and Jürgen Schmidhuber. World models. *CoRR*, abs/1803.10122, 2018. URL http://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy P. Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *CoRR*, abs/1912.01603, 2019. URL http://arxiv.org/abs/1912.01603.

Danijar Hafner, Timothy P. Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *CoRR*, abs/2010.02193, 2020. URL https://arxiv.org/abs/2010.02193.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024. URL https://arxiv.org/abs/2301.04104.

Nicklas A Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8387–8406. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/hansen22a.html.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, June 2022.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. *Advances in neural information processing systems*, 12, 1999.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5639–5650. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/laskin20a.html.

Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning, 2019. URL https://arxiv.org/abs/1509.02971.

Grace Liu, Michael Tang, and Benjamin Eysenbach. A single goal is all you need: Skills and exploration emerge from contrastive rl without rewards, demonstrations, or subgoals. *CoRR*, abs/2408.05804, 2024. URL https://doi.org/10.48550/arXiv.2408.05804.

Zhuang Ma and Michael Collins. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency, 2018. URL https://arxiv.org/abs/1809.01812.

Bogdan Mazoure, Benjamin Eysenbach, Ofir Nachum, and Jonathan Tompson. Contrastive value learning: Implicit models for simple offline RL. In *7th Annual Conference on Robot Learning*, 2023. URL https://openreview.net/forum?id=oqOfLP6bJy.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

Seohong Park, Kevin Frans, Benjamin Eysenbach, and Sergey Levine. Ogbench: Benchmarking offline goal-conditioned rl. In *International Conference on Learning Representations (ICLR)*, 2025.

Vitchyr H. Pong, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine. Skew-fit: State-covering self-supervised reinforcement learning, 2020. URL https://arxiv.org/abs/1903.03698.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1312–1320, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/schaul15.html.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3 (1):9–44, 1988.

Vivek Veeriah, Junhyuk Oh, and Satinder Singh. Many-goals reinforcement learning, 2018. URL https://arxiv.org/abs/1806.09605.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer, 2022. URL https://arxiv.org/abs/2111.07832.