Inversely Learning Transferable Rewards via Abstracted States

Yikang Gui[1] and **Prashant Doshi**[2]
[1]THINC Lab, School of Computing
University of Georgia, Athens, GA 30602
[2] School of Computing and Institute for AI
University of Georgia, Athens, GA 30602
{yikang.gui, pdoshi}@uga.edu

Abstract

Inverse reinforcement learning (IRL) has made significant progress in inferring reward functions from expert demonstrations. However, a key challenge to its use remains: how can we learn reward functions that generalize across related but distinct tasks. In this paper, we address this by focusing on *transferable* IRL, learning intrinsic rewards that can drive effective behavior in unseen but structurally aligned environments. Our method relies on a variational autoencoder to learn an abstract representation of the state space shared across multiple source tasks. This abstract space captures high-level features that are invariant between tasks, allowing the learning of a unified abstract reward function. The learned reward is then used to train policies in a separate, previously unseen target task without requiring new demonstrations in the target task. We evaluate our approach in multiple environments from Gymnasium and AssistiveGym, demonstrating that the learned abstract rewards consistently support successful policy learning in novel task settings.

1 Introduction

The objective of inverse reinforcement learning (IRL) is one of abductive reasoning: to infer the reward function that best explains observed expert trajectories. This is challenging because available data are often sparse, admitting many potential solutions (some degenerate), and the learned reward functions often fail to generalize when transferred to other tasks with distribution shifts. Despite these challenges, significant progress has been made in the last decade toward learning the underlying reward functions in discrete and continuous domains [Arora and Doshi, 2021]. A central challenge in IRL is to learn reward functions that capture *task-invariant behavioral structure*: preferences that remain relevant across aligned tasks unseen during training. This contributes to the transferability of learned rewards, a characteristic of a general solution and key to IRL's appeal over related techniques such as behavior cloning.

In this paper, we introduce a new method that generalizes IRL to unseen tasks that exhibit commonality with the observed ones in terms of shared intrinsic preferences. Abstractions offer a powerful representation toward generalization [Allen *et al.*, 2021], and we introduce the novel concept of an *abstract reward function*. To illustrate, consider the Ant domain from OpenAI Gymnasium [Schulman *et al.*, 2016]. Let two Ant environments with differing pairs of disabled legs act as source environments and an Ant environment with another pair of disabled legs as the target. As the source and target ants have different disabled legs, the marginal state distributions of the sources differ from the target's, which makes it difficult to transfer a learned reward function. However, if we focus on the ant's torso instead of its legs, the marginal state distribution of the torso remains largely invariant across both the sources and the target environment. Consequently, recovering a reward function based on the torso, which is the *abstraction*, allows the function to be transferred across varying morphologies.

Our method leverages observed behavior data from two or more tasks as input to a specialized variational autoencoder (VAE) framework. A single encoder is coupled with multiple task-specific decoders to reconstruct the trajectories unique to each source instance. We demonstrate how the common latent variable(s) of this VAE model can be interpreted and shaped as an abstract reward function that governs the task behaviors across instances of the domain. Crucially, two or more aligned task behaviors are required to disentangle shared task-invariant behavioral structure from task-specific dynamics. Furthermore, we incorporate a discriminator to regularize the latent space, ensuring it is structured by optimality information (e.g., distinguishing expert from suboptimal trajectories). This adversarial component is important because it forces the latent representation to prioritize features that are salient to the reward signal, rather than in mere reconstruction of state-action pairs.

We evaluate our method for *transferable IRL*, labeled TraIRL, on two benchmarks: Gymnasium domains [Schulman *et al.*, 2016] and AssistiveGym for collaborative robots [Erickson *et al.*, 2020]. We utilize trajectories from multiple tasks in each domain as input to the VAE and show how the inversely learned abstract reward function can help successfully learn a high-quality behavior in another aligned task of the domain or even for a different domain. These results open a new frontier for methods that can learn abstract rewards via IRL to offer a level of generalizability not previously seen in the literature.

2 Related Work

Extant transfer learning for IRL struggles with mismatches in environment dynamics, which limits reward transferability. Tanwani and Billard [2013] introduce an approach to learn diverse strategies from multiple experts by exploiting shared knowledge. However, it assumes unchanged dynamics between experts, which limits its applicability to settings with dynamics mismatch. I2L [Tanmay and Jian, 2020] is designed for state-only imitation learning and addresses transition dynamics mismatches by using a prioritized trajectory buffer and by optimizing a lower bound on the expert's state-action visitation distribution. While empirically effective, it lacks theoretical guarantees on reward transferability and does not formally justify how the learned reward generalizes across different dynamics. Viano *et al.* [2024] analyzes maximum causal entropy (MCE) IRL under transition dynamics mismatch, deriving necessary and sufficient conditions for transferability and providing a tight bound on performance degradation. It offers a robust MCE-based IRL algorithm but struggles to generalize under action-space shifts due to its reliance on matching state-action occupancy measures. Yoo and Seo [2022] studies transferable reward learning from multi-task demonstrations. However, their framework learns a latent subtask-conditioned reward and policy through *a posteriori* subtask variable whereas TraIRL learns a shared cross-task state abstraction and a single stationary reward over the learned abstract manifold.

Rewards learned by adversarial IRL and IL methods may not transfer across environments. AIRL [Fu *et al.*, 2018], f-MAX [Ghasemipour *et al.*, 2020] and f-IRL [Ni *et al.*, 2021] claim that their learned reward functions generalize to unseen or dynamically different environments. But these claims are not supported by explicit structural frameworks or theoretical guarantees, which leaves the transferability unpredictable. Furthermore, the learned rewards are tied to specific expert policy trajectories, which prevents their use in training new policies from scratch. In contrast, IQ-Learn [Garg *et al.*, 2021] is non-adversarial and learns soft Q-functions from expert data, which offers improved stability and efficiency. However, its reliance on Q-functions, which are action dependent, limits generalization to state-only reward functions. Unlike meta-IRL approaches which condition on latent task variables, TraIRL learns a single stationary reward function intended for deployment without task inference or adaptation.

Reward identifiability remains a central theoretical challenge in IRL. A fundamental ambiguity in IRL is that multiple reward functions can induce the same expert behavior, making the expert's underlying reward generally non-identifiable without additional assumptions. Prior theoretical work studies this issue from complementary perspectives. Kim *et al.* [2021] analyze reward identifiability in deterministic MDPs under the MaxEntRL objective, Rolland *et al.* [2022] characterize when rewards become identifiable or generalizable from multiple experts under structured MDP assumptions, and Schlaginhaufen and Kamgarpour [2024] provide conditions under which rewards recovered by regularized IRL remain transferable under changes in transition laws. TraIRL is orthogonal to these identifiability-focused analyses: rather than claiming full recovery of the ground-truth reward, it targets a transferable abstract reward. Specifically, TraIRL uses multi-task demonstrations to learn a shared cross-task state abstraction and then recovers a single stationary state-only reward over the learned abstract manifold.

Existing successor feature matching algorithms lack transferable feature functions, which limits their ability to generalize to unseen tasks. SFM [Jain *et al.*, 2025] introduces successor feature matching into IRL while avoiding adversarial training. However, its feature functions are not designed for transfer learning. Our method can be viewed as an extension of SFM, where we incorporate a transferable abstract feature function. In Appendix D.2 in the supplement, we compare SFM as a backbone with f-IRL as a backbone. Crucially, TraIRL differs from SFM-based approaches in that the abstraction itself is learned jointly from multiple tasks and is explicitly optimized for reward transfer, rather than being treated as a fixed feature space.

3 Background

We briefly review MCE based IRL [Ziebart *et al.*, 2010] as it informs our method. The entropy-regularized Markov decision process (MDP) is characterized by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma, \rho_0)$. Here, $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, respectively, while $\gamma \in (0, 1)$ is the discount factor. In the standard IRL context, the dynamics modeled by the transition distribution $\mathcal{T}(s'|a, s)$, the initial state distribution $\rho_0(s)$, and the reward function $r(s, a)$ are not known, and can only be determined through interaction with the environment. Optimal policy π under the maximum entropy framework maximizes the objective $\pi^* = \arg\max_\pi \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T} \gamma^t (r(s_t, a_t) + H(\pi(\cdot|s_t)))\right]$, where $\tau \triangleq (s_0, a_0, s_1, a_1, \ldots, s_T, a_T)$ denotes a sequence of states and actions induced by the policy and transition function, and $H(\pi(\cdot|s))$ is the entropy of the action distribution due to policy π for state s.

The method f-IRL [Ni *et al.*, 2021] integrates f-divergence to improve scalability and robustness. f-IRL relies on a generator-discriminator schema to recover a stationary reward function by matching the expert's state marginal distribution (also called *state density* or occupancy distribution). This approach builds on and improves on the state marginal matching (SMM) algorithm [Lee *et al.*, 2019] for IRL. We rely on a variant of f-IRL that minimizes the 1-Wasserstein distance between the state marginals, as this distance is an integral probability metric:

$$\mathcal{L}_{\mathcal{F}}(\boldsymbol{\theta}) = \mathcal{D}_{\mathcal{F}}(\rho_E || \rho_{\boldsymbol{\theta}}) = W_1(\rho_E(s)\,,\, \rho_{\boldsymbol{\theta}}(s)), \qquad (1)$$

where $\mathcal{D}_{\mathcal{F}}$ is a divergence measure between distributions, W_1 is the 1-Wasserstein distance, ρ_E and $\rho_{\boldsymbol{\theta}}$ denote the state densities of the expert and the soft-optimal learner under the reward function $R_{\boldsymbol{\theta}}$. Another advantage of f-IRL is its separation of the reward and discriminator networks, effectively forming a distillation model. The discriminator serves as the stronger model, while the reward function distills its information yielding better generalization than the single-discriminator design used in most adversarial IRL methods.

4 Learning Transferable Rewards via Abstraction

We present TraIRL in this section and its schematic in Figure 1. The approach learns a state-only abstracted reward function optimized for general transfer from expert trajectories in source tasks to unseen target environments. This abstract reward function is leveraged to train a policy in the target domain using RL without additional expert trajectories.

4.1 Problem Definition

We are provided with a set of expert trajectories which may be induced from policies of multiple source MDPs $\{\mathcal{M}^i\}_{i=1}^n$, each corresponding to a distinct but related task. The goal is to infer a shared intrinsic reward function that generalizes to a previously unseen target MDP $\mathcal{M}_T$. Critically, this allows an agent to perform the task effectively without access to expert demonstrations in the target task.

To support such transfer, we define a shared *abstracted state space* $\bar{\mathcal{S}}$ that captures task-invariant features across the source MDPs. This is formalized below:

Definition 1 (Cross-task abstraction)**.** *The ground MDP for task i is defined as $\mathcal{M}^i = (\mathcal{S}^i, \mathcal{A}^i, \mathcal{P}^i, \mathcal{R}^i, \gamma^i, \rho_0^i)$. A cross-task abstraction is defined by a mapping $\phi : \mathcal{S}^i \to \bar{\mathcal{S}}$, where $\phi(s^i) \in \bar{\mathcal{S}}$ denotes the abstracted state corresponding to the ground state s^i. The inverse mapping $\psi^i(\bar{s})$ denotes the set of ground states of task i that are mapped to the abstract state $\bar{s} \in \bar{\mathcal{S}}$.*

TraIRL assumes that the source and target tasks share a common abstract manifold though this manifold is unknown *a priori* and learned from source demonstrations. Our objective is to learn an abstract reward function $\bar{\mathcal{R}} : \bar{\mathcal{S}} \to \mathbb{R}$ and a cross-task abstraction ϕ such that the composition $\bar{\mathcal{R}} \circ \phi : \mathcal{S}^i \to \mathbb{R}$ induces expert behaviors across source tasks $i = 1$ to n when used to substitute $\mathcal{R}^i$ in $\mathcal{M}^i$. The abstract reward function $\bar{\mathcal{R}}$ and the cross-task abstraction ϕ are then utilized to guide policy search in the target MDP $\mathcal{M}^T$, assuming that $\mathcal{M}^T$ shares the same abstract manifold $\bar{\mathcal{S}}$.

4.2 Learning Abstraction via Multi-Head VAE

To enable reward transfer across tasks, we extract an abstract representation that captures the intrinsic, task-invariant structure from expert demonstrations. In TraIRL, we realize the cross-task abstraction function ϕ using the encoder of a VAE. Specifically, we employ a *single* task-agnostic encoder $p_{\boldsymbol{\phi}}$ shared across all source tasks and *n* task-specific decoders $\{q_{\boldsymbol{\psi}^i}\}_{i=1}^n$. The encoder $p_{\boldsymbol{\phi}}(z|s)$ maps a ground state $s \in \mathcal{S}^i$ to a latent variable $z \in \bar{\mathcal{S}}$, which serves as the abstracted state $\phi(s)$. The corresponding decoder $q_{\boldsymbol{\psi}}(s|z)$ approximates the inverse mapping $\psi(z)$, reconstructing the original ground state from its abstracted state.

To facilitate the distillation of task-invariant features, we incorporate a gated linear unit (GLU) at the input layer of the encoder. For any given ground state s, the gated transformation is defined as:

$$\tilde{s} = \sigma(sV + b_v) \odot (sW + b_w) \tag{2}$$

While the gating mechanism enables the encoder to dynamically mask task-specific noise, the multi-task reconstruction objective is critical for maintaining the topological continuity of the abstracted state manifold. Without this reconstruction constraint, the manifold risks collapsing into disjoint clusters as we discuss in Section 4.3. Such collapse would result in excessive separation between expert trajectories within the abstracted space, creating a discontinuity that hinders the learner policy from effectively sampling and reaching high-reward states. By enforcing ground state reconstruction, the encoder preserves a dense and continuous abstracted state manifold that bridges the gap between agent exploration and expert behavior.

For our multi-task setting, the overall VAE objective is

$$\mathcal{L}_{\text{VAE}}(\boldsymbol{\phi}, \boldsymbol{\psi}^1, \ldots, \boldsymbol{\psi}^n; \boldsymbol{\tau}) = \sum\nolimits_{i=1}^{n} \mathcal{L}^i_{\text{VAE}}(\boldsymbol{\phi}, \boldsymbol{\psi}^i; \boldsymbol{\tau}^i) =$$
$$\sum_{i=1}^{n} \mathbb{E}_{z\sim p_{\boldsymbol{\phi}}(z|s^i)} \log q_{\psi^i}(s^i|z) - \lambda_{\mathcal{D}} \mathcal{D}_{\text{KL}} \left(p_{\boldsymbol{\phi}}(z|s^i) \| p(z) \right) \tag{3}$$

where $\boldsymbol{\tau} \triangleq \langle \tau^1, \ldots, \tau^n \rangle$ denotes the collection of expert trajectories from n source tasks and each trajectory $\tau^i = \{s_0^i, s_1^i, \ldots, s_T^i\}$ consists of a sequence of states from the i-th source environment. In the rest of the paper, s_t^i denotes the state at time step t from task i, and τ^i denotes a trajectory from task i. The prior $p(z)$ is a normal distribution $\mathcal{N}(0, I)$, and $\lambda_{\mathcal{D}}$ controls the weight of the KL regularization term.

4.3 Structuring the Abstract State Space

While the VAE objective ensures a representative abstraction through reconstruction, it is optimality-agnostic as it preserves environmental features without distinguishing expert behaviors from suboptimal trajectories. To ensure that the abstracted state space is semantically structured for reward recovery, we introduce a discriminator-guided regularization mechanism. This allows the manifold to become optimality-aware by explicitly embedding the distributional differences between expert and learner occupancy measures.

We employ Wasserstein GAN with spectral normalization to estimate the 1-Wasserstein distance between the abstracted state distributions of the expert and the learner. The objective function of WGAN in our multi-task setting is

$$\mathcal{L}_{\text{WGAN}}(\boldsymbol{\phi}, \boldsymbol{\omega}\,; \boldsymbol{\tau}) = \sum\nolimits_{i=1}^{n} \mathcal{L}^i_{\text{WGAN}}(\boldsymbol{\phi}, \boldsymbol{\omega}\,; \boldsymbol{\tau}^i) =$$
$$\sum\nolimits_{i=1}^{n} \mathbb{E}_{p_{\boldsymbol{\phi}}(z|s), \rho_L(s^i)}[D_{\boldsymbol{\omega}}(z)] - \mathbb{E}_{p_{\boldsymbol{\phi}}(z|s), \rho_E(s^i)}[D_{\boldsymbol{\omega}}(z)] \tag{4}$$

where $\rho_L(s^i)$ and $\rho_E(s^i)$ denote the state densities of the learner and expert in the i-th source task, respectively, and $D_{\boldsymbol{\omega}}$ denotes the discriminator. By distinguishing between these trajectories, the discriminator imposes an optimality-relevant topology on the abstracted state space, which facilitates the learning of a generalizable reward function.

4.4 Robust Transferable Reward Learning via Abstracted States

Once the abstracted state manifold is structured, we recover a shared reward function $R_{\boldsymbol{\theta}}(z)$ defined over this manifold. It is important to note that TraIRL is not strictly coupled to a specific IRL framework; however, we adopt f-IRL as our underlying backbone. This choice is motivated by the functional decoupling of the reward and discriminator networks, which provides two primary advantages for transfer learning.

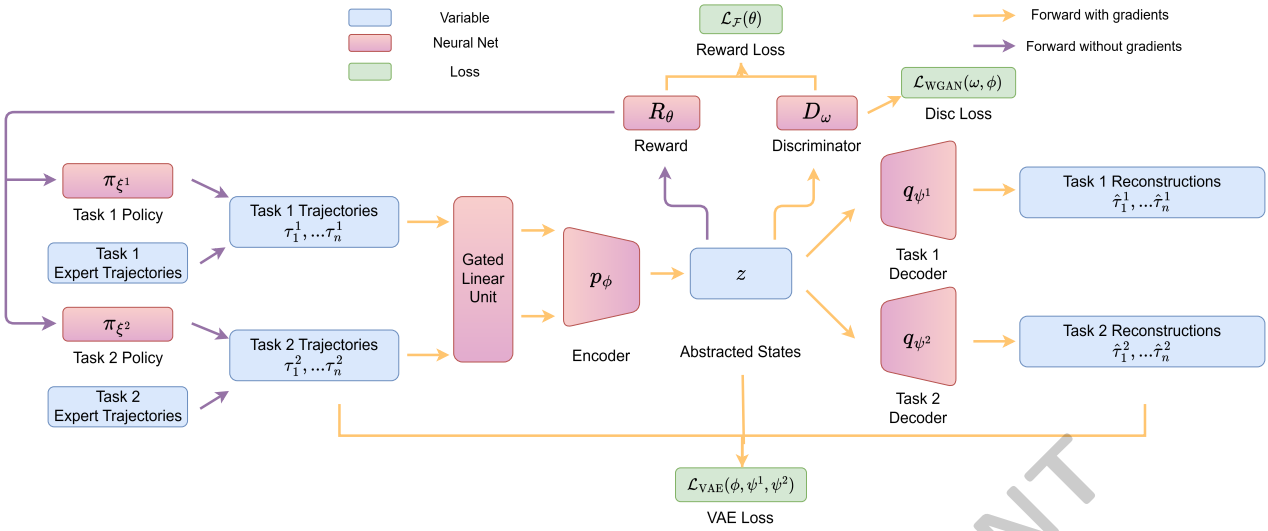


Figure 1: Overview of TraIRL. Expert and learner trajectories in multiple source tasks are mapped to shared abstract states via a shared encoder. A discriminator compares the abstracted state densities to estimate the 1-Wasserstein distance between the expert and learner, which guides learning a reward function over the abstract space through a covariance-based objective. The learned reward function then optimizes the learner policy.

First, the separation allows the reward function to act as a stationary optimization target, analogous to a target network in deep reinforcement learning. By decoupling the reward from the high-variance fluctuations of the adversarial discriminator, we ensure the recovered reward is robust to data variance and provides a stable signal for the learner policy. Second, this architecture induces a natural distillation process. By forcing the reward function to approximate the density misalignments captured by the discriminator, we filter out task-specific artifacts and overfitted features, thus ensuring that only generalizable, task-invariant intent is preserved.

In our multi-task adaptation, the reward function $R_{\boldsymbol{\theta}}(z)$ is trained to minimize the 1-Wasserstein distance between the abstracted state densities of the expert and the learner. The objective function is defined as:

$$\mathcal{L}_{\mathcal{F}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} W_1(\rho_E(z^i), \rho_L(z^i)) = \sum_{i=1}^{n} \mathbb{E}_{p_{\boldsymbol{\phi}}(z|s), \rho_E(s^i)}[D_{\boldsymbol{\omega}}(z)] - \mathbb{E}_{p_{\boldsymbol{\phi}}(z|s), \rho_L(s^i)}[D_{\boldsymbol{\omega}}(z)]. \quad (5)$$

Notably, while the parameters $\boldsymbol{\theta}$ are not explicitly present in the objective, they are implicitly coupled to the objective through the learner's occupancy measure $\rho_L(s^i)$, which is induced by the learner policy optimized under $R_{\boldsymbol{\theta}}$. During reward optimization, the abstraction encoder is frozen to prevent representation drift and ensure the reward is optimized over a stable manifold.

Theorem 1 (Gradient of reward function)**.** *The analytic gradient of our objective function $\mathcal{L}_{\mathcal{F}}(\boldsymbol{\theta})$ presented in Eq. 5 w.r.t $\boldsymbol{\theta}$ can be derived as:*

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{F}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} cov_{\hat{\rho}(s^i), p_{\boldsymbol{\phi}}(z|s)} \left(D_{\boldsymbol{\omega}}(z), \nabla_{\boldsymbol{\theta}} R_{\boldsymbol{\theta}}(z)\right),$$

where $\hat{\rho}(s^i) = \frac{1}{2}(\rho_L(s^i) + \rho_E(s^i))$.

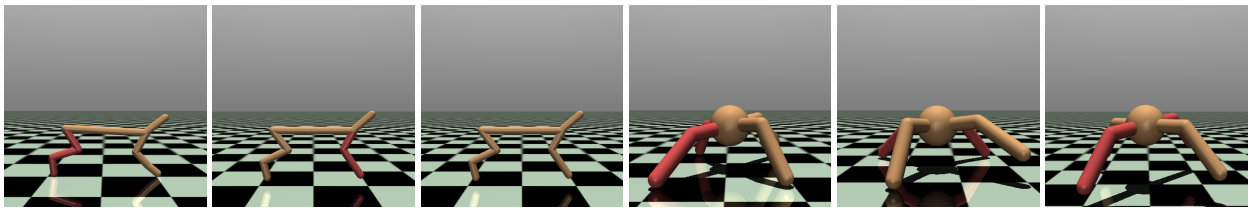
Theorem 1 (proof in Appendix B.1 of supplement) demonstrates that the reward gradient is computed as the covariance between the discriminator's density-matching signal and the reward's parameter sensitivity. This formulation is advantageous as it does not require backpropagation through the environment transition dynamics. During this phase, the encoder $p_{\boldsymbol{\phi}}$ is held stationary to prevent representation drift, ensuring the reward is optimized over a stable and continuous manifold. This decoupling prevents the reward from overfitting to task-specific artifacts, thereby enhancing its transferability.

The overall objective function for TraIRL across n source tasks is a weighted combination of the three objective functions defined previously:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\omega}, \boldsymbol{\phi}, \boldsymbol{\psi}^1, \ldots, \boldsymbol{\psi}^n) = \lambda_{\text{VAE}} \mathcal{L}_{\text{VAE}}(\boldsymbol{\phi}, \boldsymbol{\psi}^1, \ldots, \boldsymbol{\psi}^n) - \lambda_{\mathcal{F}} \mathcal{L}_{\mathcal{F}}(\boldsymbol{\theta}) + \lambda_{\text{WGAN}} \mathcal{L}_{\text{WGAN}}(\boldsymbol{\phi}, \boldsymbol{\omega}) \quad (6)$$

where $\lambda_{\text{VAE}}, \lambda_{\mathcal{F}}$ and λ_{WGAN} are the hyperparameters for $\mathcal{L}_{\text{VAE}}, \mathcal{L}_{\mathcal{F}}, \mathcal{L}_{\text{WGAN}}$, respectively.

Finally, Fu *et al.* [2018] demonstrates that disentangling rewards from task dynamics is essential for transfer, which requires the reward to satisfy a decomposability condition. While this condition is frequently violated in ground-level MDPs due to task-specific transition artifacts, TraIRL induces an abstracted state manifold designed to satisfy this property. By optimizing $R_{\boldsymbol{\theta}}(z)$ over a manifold trained for cross-task invariance, TraIRL aims to recover a reward that is dynamics-agnostic relative to the abstracted transitions. We provide a theoretical analysis and a concrete example in Appendix B.3 demonstrating a case where decomposability fails in the ground MDP but is successfully satisfied within the induced abstract MDP, thereby enabling robust transfer across environments with significantly different dynamics. The corresponding algorithm optimizing Eq. 6 is in Appendix A.



(a) (b) (c) (d) (e) (f)

Figure 2: Source and target tasks from MuJoCo domains. Red legs are the disabled legs of the robots. Frames (a,b) depict source tasks of running with a disabled leg in **Half Cheetah** while (c) represents the target environment with no disability. Similarly, frames (d,e) show the source tasks of running with different pairs of disabled legs in **Ant**, whereas (f) shows the target of running with another pair of disabled legs.

4.5 Analytical Framework for Reward Transferability

We aim to formally characterize sufficient conditions under which reward transfer is possible when using TraIRL, in order to interpret empirical successes and failures.

Definition 2 (Reward transferability)**.** *Define a reward function $R_{\boldsymbol{\theta}}$ learned for states $\mathcal{S}^i$ of the source task $\mathcal{T}^i$ as **transferable** to a target task $\mathcal{T}^t$ iff for a small constant $\epsilon > 0$,*

$$W_1(\rho^*_{\mathcal{T}^t}(z), \rho_{\mathcal{T}^i}(z)) \leq \epsilon, \tag{7}$$

where $\rho^*_{\mathcal{T}^t}(z)$ is the abstract state density induced by the (soft-)optimal policy $\pi^*_{\mathcal{T}^t}$ in the target task and $\rho_{\mathcal{T}^i}(z)$ is the abstract state density induced by the policy $\pi_{\mathcal{T}^i}$ obtained by optimizing for $R_{\boldsymbol{\theta}}$ in the source task $\mathcal{T}^i$.

Definition 2 states that the reward function $R_{\boldsymbol{\theta}}$ is transferable if the policy obtained by optimizing $R_{\boldsymbol{\theta}}$ in the source task yields an abstract state density that closely matches the one induced by the (soft-)optimal policy in the target task. Next, Theorem 2 delineates the conditions under which the reward function learned using TraIRL is transferable from source tasks to a target task.

Theorem 2 (Applicability of TraIRL)**.** *Let $R_{\boldsymbol{\theta}}$ denote the reward function learned by optimizing Eq. 6, $\rho^*_{\mathcal{T}^i}(z)$ denotes the abstract state density of the expert in the i-th source task $\mathcal{T}^i$, ϵ be the threshold from Def. 2, and $\alpha \in (0, 1)$. If, for every i,*

$$W_1(\rho^*_{\mathcal{T}^t}(z), \rho^*_{\mathcal{T}^i}(z)) \leq \alpha\epsilon \tag{8}$$
$$W_1(\rho_{\mathcal{T}^i}(z), \rho^*_{\mathcal{T}^i}(z)) \leq (1-\alpha)\epsilon \tag{9}$$

*then the reward function $R_{\boldsymbol{\theta}}$ is **transferable** to the target task, enabling effective policy learning.*

Sketch of Proof. Wasserstein distance, W_1, satisfies the *triangle inequality*. Applying it to Eqs. 8 and 9, we derive Eq. 7. This satisfies the transferable reward condition in Def. 2. □

Eq. 8 should be understood as an assumption on the task family, not a condition enforced or verified during training. Despite its simplicity, this decomposition is useful in practice: it explains failure modes (Appendix D.9), motivates the use of abstraction to reduce Wasserstein distances (Table 1b), and guides the choice of multiple source tasks. Theorem 2 identifies two fundamental requirements for cross-task reward transfer. First, Eq. 8 represents the *structural alignment assumption*: it requires that the optimal behaviors across source and target tasks map to similar regions in the abstracted manifold. This validates the invariance of our latent representation. Second, Eq. 9 represents *recoverability correctness*: it ensures that the IRL process effectively distills the source expert's intent into $R_{\boldsymbol{\theta}}$ such that the resulting learner policy can replicate the expert's occupancy in the abstract space.

5 Experiments

We implement our framework in PyTorch and evaluate its performance across MuJoCo [Todorov *et al.*, 2012] and Assistive Gym [Erickson *et al.*, 2020] benchmarks. TraIRL is trained to convergence using 50 trajectories from multiple source tasks with distinct dynamics. Policy performance is measured by the mean and standard deviation of accumulated rewards over 25 evaluation episodes. For a fair comparison, we benchmark against three state-of-the-art techniques: AIRL-ME [Buening *et al.*, 2022], RIME [Chae *et al.*, 2022], and I2L. While RIME and I2L are specifically designed for dynamics shifts, they lack the abstraction mechanisms utilized by TraIRL. All algorithms receive the default Gymnasium observation space, including joint angles and velocities, as input.

Model architecture and implementation We employ a multilayer perceptron with Tanh as the activation function for both the encoder and decoder in VAE and to represent the reward function. We use Soft Actor-Critic (SAC) [Haarnoja *et al.*, 2018] as the backbone RL algorithm. Further details regarding the model architecture and hyperparameters to aid reproducibility are available in Appendix C.

5.1 Evaluations in MuJoCo-Gym

We use the **Half Cheetah** and **Ant** domains in MuJoCo-Gym. Figure 2 illustrates the source and target tasks, which differ in dynamics between the sources and between the sources and the target. Specifically, these differences in dynamics arise from *disabling* different pairs of legs. Disabled legs are indicated in red in the frames of Fig. 2. While the action space remains unchanged across the environments, as the input actions are directly applied to all joints regardless of whether a leg is disabled, the dynamics differ because the disabled legs cannot respond to the input actions.

With a single source task, abstraction learning is not well defined and tends to entangle reward-relevant and dynamics-specific features (Appendix D.4). Multiple sources allow for

	Sources		Target
	Run (rear disabled)	Run (front disabled)	Run (no disability)
f-IRL	3,252.72 ± 67.9	3,523.01 ± 91.1	3,582.10 ± 94.3
AIRL-ME	4,014.52 ± 79.3	3,905.38 ± 73.1	3,725.63 ± 80.9
RIME	4,271.67 ± 36.1	4,129.19 ± 83.4	4,061.22 ± 58.6
I2L	4,396.43 ± 45.4	4,518.66 ± 52.2	4,512.71 ± 66.4
TraIRL	4,404.07 ± 57.6	4,359.35 ± 99.2	**5,835.11 ± 74.0**
Expert	5,052.25 ± 25.4	5,499.07 ± 156.1	6,420.38 ± 37.9

(a)

Domain	Pair	Abstraction	Ground
HalfCheetah	S1–S2	**0.36**	1.37
	S1–T	**0.62**	2.83
	S2–T	**0.55**	2.10
Ant	S1–S2	**0.33**	1.78
	S1–T1	**0.71**	2.82
	S1–T2	**0.79**	2.97

(b)

Table 1: Transfer performance and abstraction analysis. (a) Mean cumulative rewards with standard deviation for HalfCheetah. (b) Cross-task Wasserstein distances W_1, where S1 and S2 denote source tasks, and T, T1, and T2 denote target tasks.

	Sources		Targets		
	Leg 1,2 disabled	Leg 0,3 disabled	Leg 1,3 disabled	Leg 0,2 disabled	Half Cheetah (One-Shot)
f-IRL	2,456.17 ± 85.0	1,146.39 ± 95.8	1,598.24 ± 44.9	1,652.89 ± 74.3	3871.57 ± 115.1
AIRL-ME	2,389.65 ± 52.0	2,231.09 ± 87.6	2,098.11 ± 99.3	2,190.12 ± 50.2	4963.55 ± 215.7
RIME	2,681.67 ± 49.9	2,708.14 ± 81.5	2,190.71 ± 61.9	2,188.00 ± 59.8	4623.12 ± 155.0
I2L	2,831.28 ± 36.4	2,786.90 ± 79.4	2,585.32 ± 84.5	2,618.32 ± 92.4	4789.89 ± 90.4
TraIRL	2,714.18 ± 35.9	2,936.52 ± 95.5	**2,917.92 ± 79.3**	**3,156.54 ± 63.1**	**5,378.78 ± 61.7**
Expert	3,312.12 ± 304.3	3,303.99 ± 341.0	3,369.05 ± 216.8	3,590.57 ± 158.2	6,420.38 ± 37.9

Table 2: Mean cumulative rewards with standard deviation for **Ant**. TraIRL achieves the highest cumulative rewards in all the target tasks. Comprehensive experiments including more source and target tasks can be found in Appendix D.10.

invariance that cannot be explained by dynamics alone, enabling the disentanglement of task-invariant behavioral structure. In the target task, policies are trained exclusively using the learned reward function without access to the environment's ground-truth reward or expert demonstrations. For evaluation only, we report performance using the environment's true reward function, following standard practice in IRL benchmarks.

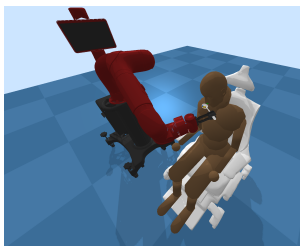
Reward Transferability. Tables 1a and 2 report our results using TraIRL on the baselines in Half Cheetah and Ant domains, respectively. These show the mean cumulative rewards obtained in the target environment as well as in the two source environments. The optimal policy's (expert) rewards for each are reported as well. Observe that the rewards learned by TraIRL achieve the highest average return compared to all baselines in the target tasks for both domains. As such, the abstracted rewards learned by TraIRL from the two sources are most transferable and correct. I2L achieves the next best performance on the target task but remains significantly lower than TraIRL's (Student's paired t-test, $p < 0.01$). More detailed experiments, including 10 source and 5 target tasks, are provided in Appendix D.10. Also refer to Appendix D.9 for the failure case of TraIRL when the structural alignment assumption (Eq. 8) is violated.

We evaluate TraIRL in a more challenging cross-domain setting: transfer from Ant to the Half Cheetah domain. We empirically demonstrate the transferability between domains with different dynamics and states of inversely learned rewards. Despite these differences, TraIRL shows strong performance in the target task with one-shot transfer, as shown in the last column of Table 2. Details of this experiment are provided in Appendix D.6. Furthermore, Appendix D.1 provides a semantic analysis of the abstract state space showing that an abstraction trained on quadrupeds captures transferable structure useful for bipedal locomotion, rather than merely encoding peculiarities of the expert data such as joint angles.

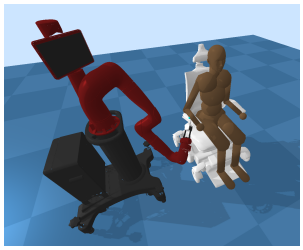
Benefit of Abstraction. To understand why TraIRL performs better, we aim to gain some insight into our novel abstraction concept. In Table 1b, we report the 1-Wasserstein distance (W_1) between the abstracted state densities of the two source tasks and between the densities of each source and the target, $W_1(\rho^*_{\mathcal{T}^i}(z), \rho^*_{\mathcal{T}^t}(z))$, for the Half Cheetah and Ant. We compare these to the corresponding W_1 distances between the ground state densities. To obtain the W_1 between ground state densities, we train a variant of f-IRL (see Appendix A.2 of Ni *et al.* [2021]). Although expert trajectories for the target are not available in the core transfer setting, the Wasserstein distances between abstract expert occupancies of source tasks (Table 2) serve as an empirical proxy for alignment. When these distances are large, we observe degraded transfer performance (Appendix D.9).

Next, we perform an abstraction sensitivity analysis to investigate the underlying mechanisms of the learned abstraction. The details of the procedure and comprehensive results are provided in Appendix D. As illustrated in Figure 3, the abstract representation exhibits the highest sensitivity to the x-velocity of the front tip. This finding is intuitive as this dimension directly captures the robot's directional progress and velocity, aligning with the forward-speed component of the ground-truth reward function. The fact that the encoder prioritizes this feature suggests that the abstraction process effectively distills task-relevant dynamics while maintaining

Figure 3: Visualization of abstraction sensitivity on the HalfCheetah target task. The x-axis indexes ground-state dimensions, and darker colors indicate lower similarity, meaning the abstraction is more sensitive to that dimension.



(a) FeedingSawyer



(b) ScratchItchSawyer

	Sources		**Target**
	Feeding task 1	Feeding task 2	Scratch itch
f-IRL	-10.63 $\pm$ 16.79	-15.3 $\pm$ 22.78	-21.35 $\pm$ 14.89
AIRL-ME	-6.35 $\pm$ 25.1	3.59 $\pm$ 17.3	-17.32 $\pm$ 11.6
RIME	8.01 $\pm$ 6.0	8.66 $\pm$ 9.7	-11.64 $\pm$ 4.2
I2L	9.32 $\pm$ 11.8	9.11 $\pm$ 11.4	-10.07 $\pm$ 8.0
TraIRL	8.32 $\pm$ 7.9	9.56 $\pm$ 10.5	**-3.82 $\pm$ 3.3**
Expert	11.29 $\pm$ 5.3	12.77 $\pm$ 4.2	-1.18 $\pm$ 5.7

(c)

Figure 4: Two tasks in the feeding domain of **Assistive Gym** (a) are used as sources to learn a reward function that is transferred to perform the task of scratching an itch shown in (b). (c) Learned policy returns with standard deviation in the **Assistive Gym** environments. TraIRL shows the highest return in the target task.

invariance to less critical state dimensions.

For each comparison in Table 1b, the abstracted state densities yield a consistently lower W_1 distance compared to the ground state densities. This indicates that the abstraction reduces task-specific variability in the abstract state space, bringing the source and target closer in terms of their occupancy measures. In particular, the W_1 distance between the abstract state densities of optimal policies in the source and target tasks, $W_1(\rho^*_{\mathcal{T}^i}(z), \rho^*_{\mathcal{T}^t}(z))$, corresponds to Eq. 8 in Theorem 2. A smaller distance implies a tighter bound on the transfer error ϵ and therefore reflects better generalization of the abstract state space across tasks.

We explored the sensitivity of TraIRL's performance to hyperparameter values and conducted ablation studies. The results are reported in Appendix D.3. The visualizations of the abstracted state manifolds are reported in Appendix F.

5.2 Evaluation in Human-Robot Assistive Gym

To give an indication of the utility of TraIRL in the real-world, we illustrate its use in human-robot collaboration using the realistic Assistive Gym testbed [Erickson *et al.*, 2020]. Similar to MuJoCo, Assistive Gym is a physics-based simulation framework but for studying human-robot interaction and robotic assistance by the collaborative robot Sawyer.

For the source tasks, we select *FeedingSawyer*, a simulation environment in which the collaborative robot is tasked with feeding a disabled human (Fig. 4a) The two source tasks differ in the condition of the disabled human. The human is static in one while the human has tremors in the other, which cause the target area to move leading to a shifting goal. The robot's challenge is to adapt to these differing human conditions while executing the feeding task. Our target task is a different task, *ScratchItchSawyer*, in which Sawyer is tasked with scratching a disabled human's itch (Fig. 4b). Although this task differs from feeding in terms of its specific goal, it is also aligned in that the robot must move its end effector precisely to a designated target area. This abstract information should be represented by the learned reward function.

We report the performance of TraIRL and the baselines in transferring the reward function inversely learned from the two feeding tasks, to learn how to scratch the person's itch. Table 4c gives the mean cumulative reward from forward RL in the target using the learned rewards. Notice that TraIRL yields the policy with the highest value and close to the optimal, indicating transferable rewards. The transferred reward function exhibits a strong positive linear relationship with the ground-truth rewards, as indicated by a Pearson correlation coefficient of 0.86 ($p < .001$). Among the baselines, I2L and RIME achieve comparative transfer, with performance close to each other on the source tasks but still weaker on the target. AIRL-ME lags further behind, showing limited ability to generalize. In contrast, TraIRL consistently outperforms all three, indicating that the abstracted reward function provides superior transferability.

6 Conclusion

TraIRL represents a significant advancement of IRL by introducing a principled approach to inversely learn transferable reward functions from demonstrations of multiple aligned tasks. The key contribution lies in its ability to extract invariant abstractions that model the structure intrinsic to multiple tasks, which makes them transferable to aligned target tasks. TraIRL's analytical properties delineate the transfer applicability of the abstracted rewards and the experiments validate the transferability in both formative and use-inspired contexts. By explicitly leveraging multiple source tasks, TraIRL mitigates the entanglement between task-specific dynamics and reward-relevant structure, enabling more robust generalization under dynamics shift.

Acknowledgments

This research was supported in part by NSF grants IIS-1830421 and IIS-2312657 both to Doshi. We acknowledge useful discussions with Prasanth Suresh that informed parts of this paper.

References

[Allen *et al.*, 2021] Cameron Allen, Neev Parikh, Omer Gottesman, and George Konidaris. Learning markov state abstractions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 34:8229–8241, 2021.

[Arora and Doshi, 2021] Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, 2021.

[Buening *et al.*, 2022] Thomas Kleine Buening, Victor Villin, and Christos Dimitrakakis. Environment design for inverse reinforcement learning. In *Proceedings of the 41st International Conference on Machine Learning*, 2022.

[Chae *et al.*, 2022] Jongseong Chae, Seungyul Han, Whiyoung Jung, Myungsik Cho, Sungho Choi, and Youngchul Sung. Robust imitation learning against variations in environment dynamics. *The 39th International Conference on Machine Learning*, pages 2828–2852, 2022.

[Erickson *et al.*, 2020] Zackory Erickson, Vamsee Gangaram, Ariel Kapusta, C. Karen Liu, and Charles C. Kemp. Assistive gym: A physics simulation framework for assistive robotics. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

[Fu *et al.*, 2018] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adverserial inverse reinforcement learning. In *International Conference on Learning Representations*, 2018.

[Garg *et al.*, 2021] Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation. *Advances in Neural Information Processing Systems*, 34:4028–4039, 2021.

[Ghasemipour *et al.*, 2020] Seyed Kamyar Seyed Ghasemipour, Richard Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. In *The 3rd Conference on robot learning*, 2020.

[Haarnoja *et al.*, 2018] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[Jain *et al.*, 2025] Arnav Kumar Jain, Harley Wiltzer, Jesse Farebrother, Irina Rish, Glen Berseth, and Sanjiban Choudhury. Non-adversarial inverse reinforcement learning via successor feature matching. In *The International Conference on Learning Representations*, 2025.

[Kim *et al.*, 2021] Kuno Kim, Shivam Garg, Kirankumar Shiragur, and Stefano Ermon. Reward identification in inverse reinforcement learning. *Proceedings of the 38th International Conference on Machine Learning*, 139:5496–5505, 2021.

[Lee *et al.*, 2019] Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. In *arXiv*, 2019.

[Ni *et al.*, 2021] Tianwei Ni, Harshit Sikchi, Yufei Wang, Tejus Gupta, Lisa Lee, and Ben Eysenbach. f-irl: Inverse reinforcement learning via state marginal matching. In *The 4th Conference on Robot Learning*, 2021.

[Rolland *et al.*, 2022] Paul Rolland, Luca Viano, Norman Schürhoff, Boris Nikolov, and Volkan Cevher. Identifiability and generalizability from multiple experts in inverse reinforcement learning. *Advances in Neural Information Processing Systems*, 35:550–564, 2022.

[Schlaginhaufen and Kamgarpour, 2024] Andreas Schlaginhaufen and Maryam Kamgarpour. Towards the transferability of rewards recovered via regularized inverse reinforcement learning. In *Advances in Neural Information Processing Systems*, 2024.

[Schulman *et al.*, 2016] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *4th International Conference on Learning Representations*, 2016.

[Tanmay and Jian, 2020] Gangwani Tanmay and Peng Jian. State-only imitation with transition dynamics mismatch. In *The 8th International Conference on Learning Representations*, 2020.

[Tanwani and Billard, 2013] Ajay Kumar Tanwani and Aude Billard. Transfer in inverse reinforcement learning for multiple strategies. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3244–3250, 2013.

[Todorov *et al.*, 2012] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.

[Viano *et al.*, 2024] Luca Viano, Yu-Ting Huang, Parameswaran Kamalaruban, Adrian Weller, and Volkan Cevher. Robust inverse reinforcement learning under transition dynamics mismatch. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2024.

[Yoo and Seo, 2022] Se-Wook Yoo and Seung-Woo Seo. Learning multi-task transferable rewards via variational inverse reinforcement learning. *2022 International Conference on Robotics and Automation (ICRA)*, pages 434–440, 2022.

[Ziebart *et al.*, 2010] Brian D. Ziebart, J. Andrew Bagnell, and Anind K. Dey. Modeling interaction via the principle of maximum causal entropy. In *Proceedings of the 27th International Conference on Machine Learning*, 2010.