Unsupervised Domain Adaptation with Shared Latent Dynamics for Reinforcement Learning

Evgenii Nikishin[1,2*] nikishin.evg@gmail.com
Arsenii Ashukha[1,3] ars.ashuha@gmail.com
Dmitry Vetrov[2,3] dvetrov@hse.ru

[1]National Research University Higher School of Economics
[2]Samsung-HSE Laboratory, National Research University Higher School of Economics
[3]Samsung AI Center Moscow

Abstract

We propose a neural network architecture for domain adaptation in reinforcement learning. The architecture allows learning similar latent representations for similar observations from different environments without access to a one-to-one correspondence between the observations. The model achieves the alignment between the latent codes via learning shared dynamics for different environments and matching marginal distributions of latent codes. Furthermore, a single policy trained upon the latent representations from one environment acts optimally simultaneously for different environments.

1 Introduction

Reinforcement learning algorithms struggle to adapt quickly to new environments [Tzeng et al., 2015, Tobin et al., 2017]. In this paper, we consider a problem of leveraging representations learned in a source task S in a new, target task T. We assume that the source and target tasks have similar dynamics in *a hidden space*. As an example of such a source-target pair, consider an Atari game with observations x_S as a source environment, and a target environment with observations x_T that were obtained by inverting colors of x_S.

We propose a model that learns similar latent representations for similar pairs of observations x_S and x_T from different domains. The model uses shared dynamics in a latent space and adversarial matching of latent codes as a way to align the latent representations. Given the aligned latent space, the model aims to learn a policy upon latent representations that is optimal for both of the environments.

The related work includes [Gamrian and Goldberg, 2018] training a mapping in an unsupervised way directly between observations from source and target domains using CycleGANs [Zhu et al., 2017]. Ilse et al. [2019] proposes an architecture based on variational autoencoders [Kingma and Welling, 2013] that learns separately target-specific, domain-specific and reconstruction-specific features. A series of papers [Sadeghi and Levine, 2016, Tobin et al., 2017] proposes to increase the robustness of models by augmenting training datasets with randomized observations. The distinction of our model is exploiting the temporal structure of reinforcement learning and leveraging the assumption that source and target tasks may have similar dynamics in some hidden space.

We evaluate the proposed model on a pair of toy environments and demonstrate that the model can learn alignment between latent codes for different domains. Furthermore, a single policy upon the latent representations trained on data only from the source task acts optimally in both of the environments. We release PyTorch code at github.com/nikishin-evg/dyn_aae.

*Now student at Cornell University.

4th workshop on Bayesian Deep Learning (NeurIPS 2019), Vancouver, Canada.

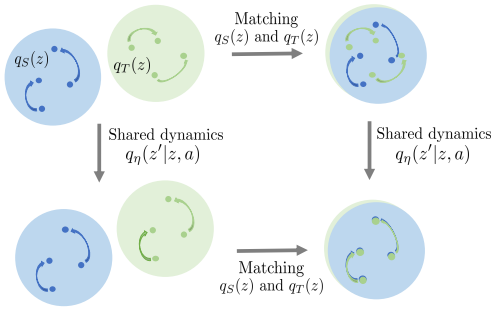

Figure 1: An illustration demonstrating the effects of introducing shared dynamics and the adversarial loss. Top left: aggregated posteriors $q_S(z)$ and $q_T(z)$. Top right: unaligned codes in matched $q_S(z)$ and $q_T(z)$. Bottom left: disjoint $q_S(z)$ and $q_T(z)$. Bottom right: aligned space.

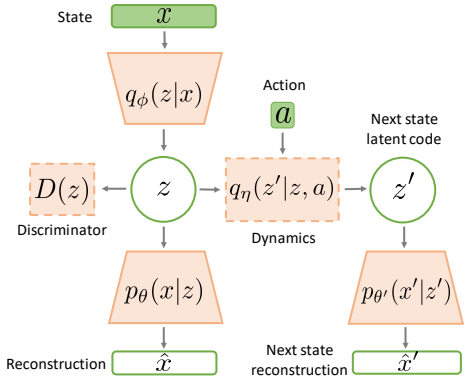

Figure 2: An instance of the proposed architecture. During training on the **source** environment, $D_S(z)$ distinguishes between samples from $q_S(z)$ and a prior distribution. During training on the **target** environment, $q_\eta(z'|z,a)$ is fixed and $D_T(z)$ distinguishes between samples from $q_S(z)$ and $q_T(z)$.

2 Domain adaptation using shared latent dynamics

The proposed model is based on variational autoencoders [Kingma and Welling, 2013]. We introduce an encoder-decoder pair $q_{\phi^S}(z|x), p_{\theta^S}(x|z)$ for a source environment S and an encoder-decoder pair $q_{\phi^T}(z|x), p_{\theta^T}(x|z)$ for a target environment T. Without additional requirements, latent codes for different domains would be irrelevant to each other. Thus we further extend the mode by two mechanisms that force latent codes for S and T to be aligned.

Adversarial posterior matching The first mechanism forces latent codes from different environments to belong to the same region of a latent space. As proposed in [Makhzani et al., 2015], we use a discriminator network $D(z)$ instead of KL regularization for matching latent codes. Let $q(z)$ denote an aggregated posterior $q(z) = \int q_\phi(z|x)p_{\text{data}}(x)dx$ induced by an encoder $q_\phi(z|x)$. A discriminator $D(z)$ is trained to classify latent codes sampled from the aggregated posterior $q(z)$ and from a prior distribution $p(z)$. The discriminator $D(z)$ is then used in a loss for encoder to guide the encoder to match $q(z)$ to $p(z)$.

During training the model on a source the choice of a prior distribution is arbitrary, for example, a standard gaussian. During the adaptation phase, we choose a prior to be the aggregated posterior $q_S(z)$ of the encoder for the source domain, forcing the aggregated posteriors $q_S(z)$ and $q_T(z)$ induced by encoders from different domains to be matched.

Shared latent dynamics Matched aggregated posteriors for two domains is a necessary but not a sufficient condition for learning aligned latent representations for both of the domains. As illustrated in Figure 1, aggregated posteriors $q_S(z)$ and $q_T(z)$ can be concentrated in the same region of a latent space, however, similar observations x_S and x_T would be encoded into different latent codes.

We further extend the model by introducing a neural network $q_\eta(z'|z,a)$ that learns the transition dynamics in the latent space. During training on the source, the parameters of dynamics are learned along with the parameters of other parts of the model through an additional x' reconstruction loss. Total loss for training the model—except $D_S(z)$—on the source task is given by:

$$\mathbb{E}_{q_{\phi^S}(z|x)}\left[\log p_{\theta^S}(x|z) + \log D_S(z) + \mathbb{E}_{q_\eta(z'|z,a)} \log p_{\theta'^S}(x'|z')\right] \rightarrow \max_{\phi^S,\theta^S,\theta'^S,\eta}. \tag{1}$$

During the adaptation phase, we freeze the parameters η of the dynamics, and only train encoder-decoder parts and $q_{\phi^T}(z|x), p_{\theta^T}(x|z), p_{\theta'^T}(x'|z')$ using the following loss:

$$\mathbb{E}_{q_{\phi^T}(z|x)}\left[\log p_{\theta^T}(x|z) + \log D_T(z) + \mathbb{E}_{q_\eta(z'|z,a)} \log p_{\theta'^T}(x'|z')\right] \rightarrow \max_{\phi^T,\theta^T,\theta'^T}, \tag{2}$$

where a discriminator $D_T(z)$ is trained to distinguish samples from $q_S(z)$ and $q_T(z)$. We summarize the proposed model for learning aligned representations for S and T in Figure 2.

Algorithm 1 Unsupervised Domain Adaptation with Shared Latent Dynamics

Input: $\mathcal{D}_S$, $\mathcal{D}_T$— transitions (x, a, x') from source and target domains.

1. Learning the latent space for source domain.
   Train $q_{\phi^S}(z|x)$, $p_{\theta^S}(x|z)$, $q_\eta(z'|z,a)$, $p_{\theta'^S}(x'|z')$ on samples from $\mathcal{D}_S$ to optimize loss 1.
   $D_S(z)$ is trained to distinguish between samples from $q_S(z)$ and a prior distribution.
2. The adaptation phase for the target domain.
   Train $q_{\phi^T}(z|x)$, $p_{\theta^T}(x|z)$, $p_{\theta'^T}(x'|z')$ using the freezed $q_\eta(z'|z,a)$ on samples from $\mathcal{D}_T$ to optimize loss 2.
   $D_T(z)$ is trained to distinguish between samples from $q_S(z)$ and $q_T(z)$.
3. Application phase: train a policy $\pi_\xi(a|z)$ on samples from $\mathcal{D}_S$ and use in the target task T.

The final goal of learning the aligned representations is to be able to have a single policy upon the latent space that will be optimal for both of the domains. Given a trained $q_{\phi^S}(z|x)$, we train a policy upon latent codes $\pi_\xi(a|z)$ on samples from S (e.g. collected following a previously trained policy) by optimizing

$$\mathbb{E}_{q_{\phi^S}(z|x)} \log \pi_\xi(a|z) \to \max_\xi .$$

Given a trained $q_{\phi^T}(z|x)$, we apply the policy $\pi_\xi(a|z)$ to the target task T. We summarize the proposed domain adaptation procedure in Algorithm 1.

3 Experiments

To evaluate the proposed model we design an artificial environment as follows:

1. observations x are given by MNIST digits dataset;
2. the action space consists of 3 actions: $\{-1, 0, +1\}$. -1 rotates an image by 90 degrees anticlockwise, 0 does not change an image, $+1$ rotates an image by 90 degrees clockwise;
3. the next observations x' are obtained after applying an action;
4. the reward $+1$ is given if an agent executes a correct action for a given x and 0 otherwise.

We randomly assign a correct action for every digit assuming that it corresponds to some optimal policy. For example, -1 will be assigned as an optimal action to all "4" digits in the dataset, 0 to all "6" digits and so on. To obtain the target environment, we invert the pixel values of all images in the dataset, while correct actions stay the same.

3.1 Learning aligned latent representations

After training both source and target parts of the model, we examine the alignment of latent codes between the domains. We demonstrate on Figure 3 that latent codes of digits from T are reconstructed into the same digits from S. Even though the dynamics network was trained on samples from S, its outputs are successfully reconstructed into x' for both domains.

3.2 Learning a policy upon latent spaces

We train a policy $\pi_\xi(a|z)$ upon a learned latent space using the data collected in the source environment $\mathcal{D}_S$. A policy $\pi_\xi(a|z)$ trained on the latent space obtained using the proposed model achieves near-optimal reward on the target dataset $\mathcal{D}_T$.

	Const	VAE	Adversarial	Dynamics	Model
Reward	0.40 ± 0	0.40 ± 0.03	0.45 ± 0.06	0.54 ± 0.07	0.81 ± 0.21

Table 1: Average rewards achieved by a constant policy (Const) and policies upon latent representations obtained by different models: without adversarial matching and shared dynamics (VAE), without shared dynamics (Adversarial), without adversarial matching (Dynamics), and the proposed model.

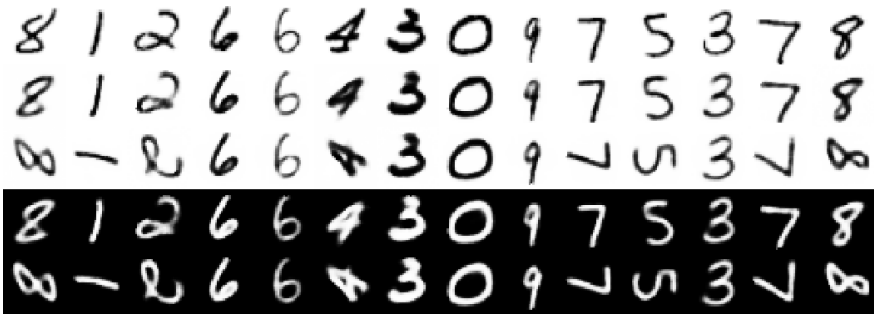

Figure 3: Cross-domain reconstructions. 1st row: samples of images from the target task T. 2nd row: outputs of $p_{\theta^T}(x|z)$ after sampling from $q_{\phi^T}(z|x)$. 3rd row: outputs of $p_{\theta'^T}(x'|z')$ after sampling from $q_{\phi^T}(z|x)$ and $q_\eta(z'|z,a)$. 4th row: outputs of $p_{\theta^S}(x|z)$ after sampling from $q_{\phi^T}(z|x)$. 5th row: outputs of $p_{\theta'^S}(x'|z')$ after sampling from $q_{\phi^T}(z|x)$ and $q_\eta(z'|z,a)$. We observe the same effects for samples from the source domain.

If we remove the adversarial mechanism and use KL regularization with standard gaussian, aggregated posteriors tend to be divided into two non-intersecting subregions of the latent space (a random forest classifier [Breiman, 2001] with default hyperparameters distinguishes samples from $q_S(z)$ and $q_T(z)$ with test accuracy close to 1).

If we remove the shared latent dynamics from the model then only aggregated posteriors are matched. We find that it is insufficient for learning the alignment and observe that similar digits from different domains have different latent codes (for example, a digit 0 from one domain may be reconstructed into digit 1 from another domain).

We compare rewards achieved in the target environment by policies upon different latent spaces in Table 1.

4 Conclusion

In this paper, we proposed a method for domain adaptation via matching latent representations of observations from different domains in an unsupervised manner. We demonstrated that the learned latent codes are decoded into similar observations from different domains and result near optimal actions in both of the domains.

The method is applicable under the assumption that there *exist* some latent representation capable of describing observations from both of the domains. We plan to apply the model to Atari games and experimentally explore the applicability of the model to more types of differences between source and target domains.

Acknowledgments

Evgenii Nikishin was supported by Samsung Research, Samsung Electronics. Dmitry Vetrov was supported by the Russian Science Foundation grant no. 19-71-30020.

References

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Shani Gamrian and Yoav Goldberg. Transfer learning for related reinforcement learning tasks via image-to-image translation. *arXiv preprint arXiv:1806.07377*, 2018.

Maximilian Ilse, Jakub M Tomczak, Christos Louizos, and Max Welling. Diva: Domain invariant variational autoencoders. *arXiv preprint arXiv:1905.10427*, 2019.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

Fereshteh Sadeghi and Sergey Levine. Cad2rl: Real single-image flight without a single real image. *arXiv preprint arXiv:1611.04201*, 2016.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30. IEEE, 2017.

Eric Tzeng, Coline Devin, Judy Hoffman, Chelsea Finn, Pieter Abbeel, Sergey Levine, Kate Saenko, and Trevor Darrell. Adapting deep visuomotor representations with weak pairwise constraints. *arXiv preprint arXiv:1511.07111*, 2015.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.