Towards the Reusability and Compositionality of Causal Representations

Davide Talon
PAVIS, Istituto Italiano di Tecnologia (IIT)
University of Genova
talon.davide@gmail.com

Phillip Lippe
QUVA Lab, University of Amsterdam

Stuart James
Department of Computer Science, Durham University
PAVIS, Istituto Italiano di Tecnologia (IIT)

Alessio Del Bue
PAVIS, Istituto Italiano di Tecnologia (IIT)

Sara Magliacane
AMLab, University of Amsterdam
MIT-IBM Watson AI Lab

Abstract

Causal Representation Learning (CRL) aims at identifying high-level causal factors and their relationships from high-dimensional observations, e.g., images. While most CRL works focus on learning causal representations in a single environment, in this work we instead propose a first step towards learning causal representations from temporal sequences of images that can be adapted in a new environment, or composed across multiple related environments. In particular, we introduce DECAF, a framework that detects which causal factors can be reused and which need to be adapted from previously learned causal representations. Our approach is based on the availability of intervention targets, that indicate which variables are perturbed at each time step. Experiments on three benchmark datasets show that integrating our framework with four state-of-the-art CRL approaches leads to accurate representations in a new environment with only a few samples.

1 Introduction

Causal Representation Learning (CRL) [15, 17, 28, 35] aims at identifying high-level causal factors and their relationships from underlying low-level observations, e.g., images. While learning structured and disentangled representations has proved effective for interpretability, efficiency and fairness of deep learning models [10, 20, 31], most methods assume independent factors of variation. This assumption is often not met in real-world applications, which hinders the generalization capabilities of these methods [4, 5, 27, 30]. CRL generalizes the disentanglement setting by considering potential causal relations between the latent causal variables. Recent works rely on auxiliary variables [11, 19], non-stationarity [34, 35], sparsity [14, 15], intervention targets [16–18] and counterfactuals [2, 32] to identify the causal factors. Causal representations retain the *modular* nature of the associated causal generative model: an external change, i.e., an intervention, on a specific target variable will not affect the *causal mechanism*, i.e., the conditional distribution of any other variable given its parents [24].

While most CRL works focus on learning causal representations in a single environment, in this work we instead propose a first step towards learning causal representations from temporal sequences of images that can be adapted in new environments, or composed across multiple related environments.

Accepted to the NeurIPS 2023 Workshop on Causal Representation Learning.

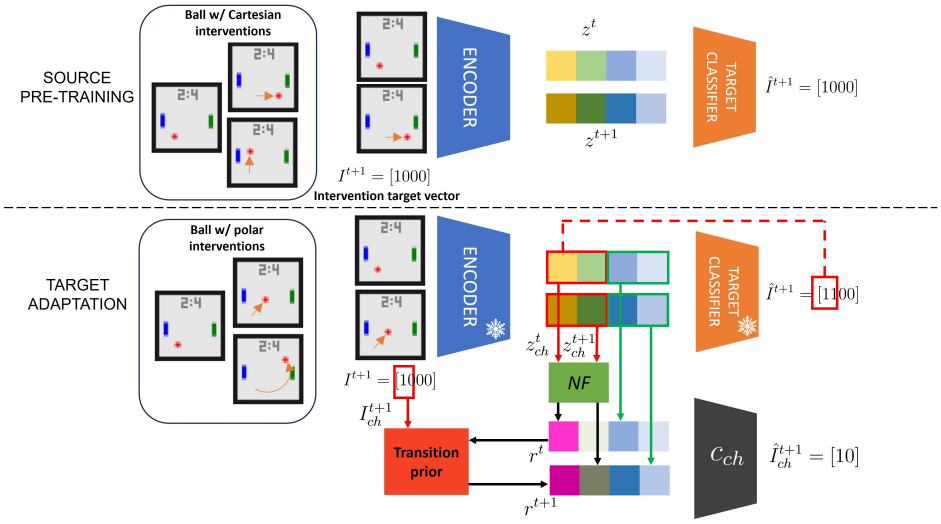


Figure 1: Overview of our approach for the *adaptation task* in Pong, where the *source* environment on which we learn the initial causal representation models the position of the ball as Cartesian coordinates, while the *target* environment uses polar coordinates for the ball position.

We are motivated by leveraging the implicit modularity of causal representations, as well as many real-world applications in which we want an agent to leverage its previous knowledge and adapt to changes in the environment with the least interactions possible.

In particular, we consider the TempoRal Intervened Sequences (TRIS) setting [17]. In this setting we observe temporal sequences of high-dimensional observations of an underlying causal system, and at each time step any of the causal variables might be intervened. We also assume that we have labels for which variables were intervened at each time step, represented as a binary *intervention target* vector. We leverage this information in DECAF (DEtect Changes and Adapt Factors), a framework that detects which causal factors can be reused and which need to be adapted from previously learned causal representations. DECAF can be combined with any CRL approach that works in TRIS.

To motivate our approach, we show an application of our framework for the *adaptation task* in Pong in Fig. 1. In the *source* environment, we exploit the available intervention targets I^t at each timestep t to learn the causal representation of the system, including the position of the two paddles and the ball. In this environment the position of the ball is measured in Cartesian coordinates x and y. Instead, in the *target* environment, the dynamics of the ball are modelled in polar coordinates, radius r and angle θ. Hence also the available interventions in this environment are on changing either the radius or the angle of a ball. In this setting, DECAF first learns a causal representation learned in the source domain using a standard CRL approach with an *encoder*. It also trains a *target classifier* to predict the intervened variables I^{t+1} at time $t+1$ from the predicted latent states z^t at time t and z^{t+1} at time $t+1$. When applying the target classifier to the new environment, DECAF exploits the discrepancies in the predicted and the intervened targets to detect which of the causal factors need to be adapted. Only these factors are then adapted by training a normalizing flow (NF) with a *transition prior* and an *auxiliary target classifier* that enforces that each newly learned latent variable models at most one intervention target. The other causal factors can be directly used in the new environment. As we show in the experiments, we can use a similar approach also in *compositional* settings in which we can combine representations from multiple source environments.

The contribution of this work is three-fold: (i) we formalize a generative model for the changes across environments for which we can *adapt* or *compose* causal representations, (ii) we propose DECAF (DEtect Changes and Adapt Factors), a framework that detects changes, adapts and composes causal representations, (iii) we validate the benefits of repurposing learned causal representations on three existing CRL benchmarks, for which we develop several adaptation and composition tasks.

2 Background

We assume our data follow the TempoRal Intervened Sequence (TRIS) setting [17]. In this setting we assume that there is an underlying unobserved causal system, and at each time step there can be an intervention on a set of causal variables. We only observe a time series of high-dimensional observations of it and the labels describing which variables have been intervened on, the *intervention targets*. Here we summarize the assumptions, and refer to [17] for details.

Latent causal process. We assume the latent causal process can be described by a Dynamic Bayesian Network (DBN) [3, 23] over a set of K multidimensional causal variables $(C_1, \dots, C_K)$ that generates the data at hand. At each time step, we only allow that a variable C_i^t can be potentially a parent of a variable C_j^{t+1} for $i, j \in [\![1..K]\!]$, i.e. the DBN is first-order Markov and has no instantaneous effects, and the causal relations are stationary, i.e., the causal parents repeat across all timesteps. In other words, each causal variable follows the structural causal equation $C_i^t = f_i(\text{pa}(C_i^t), \epsilon_i)$ for $i = [\![1..K]\!]$, where pa() are the parents, which are a subset of the variables in the previous time step, and ϵ_i is its exogenous noise. We assume the noises ϵ_i for $i = [\![1..K]\!]$ to be mutually independent. Causal factors can be multivariate, i.e., $C_i \in \mathcal{D}_i^{M_i}$ with $M_i \geq 1$ where $\mathcal{D}_i$ is $\mathbb{R}$ for continuous variables and $\mathbb{Z}$ for discrete ones. Hence, the causal factor space is defined as $\mathcal{C} = \mathcal{D}_1^{M_1} \times \mathcal{D}_2^{M_2} \times ... \times \mathcal{D}_K^{M_K}$. We denote as $C^t = (C_1^t, \dots, C_K^t)$ the causal factors at time step t.

Interventions. We assume that the causal system can be subjected to an intervention at each time step and that if it happens, we know the intervention targets. In particular, a binary vector $I^t \in \{0, 1\}^K$ indicates that a variable C_i^t is intervened upon iff $I_i^t = 1$. Intervention values are unobserved. Interventions can be *soft* [6], e.g. inducing a change in the mechanism of the intervened variables without necessarily making the target, or *hard*, e.g. do-interventions $\text{do}(C_i = c_i)$ [24]. Multiple variables can be intervened simultaneously. We model potential dependencies between intervention targets with an unobserved regime variable R^t [22]. We assume faithfulness of the distribution, hence there are no further independences than those given by the causal graph.

Observation function. At each time step t, we observe a high-dimensional observation of the latent causal factors. Let $f : \mathcal{C} \times \mathcal{U} \to \mathcal{X}$ be the invertible observation function from the space of factors $\mathcal{C}$ and noises $\mathcal{U}$ to the observation space $\mathcal{X}$. We define the high-dimensional observation $X^t = f(C_1^t, C_2^t, ..., C_K^t, U^t)$, where $U^t \in \mathcal{U}$.

Adaptation of CRL approaches to TRIS. Since the TRIS setting was originally developed for CITRIS [17], we can use it as is in this setting. We also adapt three other state-of-the-art CRL methods to work in the TRIS setting. iVAE [11] assumes that the causal variables are conditionally independent given some auxiliary information. In TRIS, this information can be provided by $\{C^t, I^{t+1}\}$. LEAP [35] leverages nonstationarity that is captured by a categorical auxiliary variable u, which can be represented with the intervention target vector I^{t+1}. Given actions with unknown targets, DMSVAE [15] identifies the causal factors when the underlying causal graph has a sparse structure. In TRIS, we consider the information target vector as the action itself.

Since we have multidimensional causal variables, we also need to learn a mapping from a latent space $\mathcal{Z} \subseteq \mathbb{R}^M$ with $M \geq K + 1$ to the causal space $\mathcal{C}$. We call this mapping the assignment function $\psi : [\![1..M]\!] \to [\![0..K]\!]$. We denote the latent variables assigned to a causal variable C_i^t as $z_{\psi_i}^t$ for $i \in [\![1..K]\!]$, while we denote with $z_{\psi_0}^t$ the latent variables that are not assigned to any causal variable. CITRIS learns ψ as part of its training, but iVAE, LEAP and DMSVAE do not and their identifiability is up to permutation and element-wise transformation. To compare them, we then use supervision to match the latent space learned by iVAE, LEAP and DMSVAE with the ground truth causal variables.

3 A simple generative model of environments for adaptation and composition

In this section we propose a simple generative model for changes across environments, for which our framework will be able to *adapt* and *compose* causal representations.

3.1 Adaption of causal representations.

For simplicity, we assume that we have two environments, the source S and the target T. We assume there is an underlying latent causal process with *underlying causal variables* C^t that is the same for both environments. In the source, we consider a set of *source causal variables* C_S^t, which are an invertible function of the underlying causal variables C^t. Similarly, we consider a set of *target causal variables* C_T^t, which are an invertible function of C^t. In general, we will assume that some of underlying causal variables C_{sh}^t are *shared* across the environments and with the underlying causal model, while others C_{ch}^t can change across the environments and w.r.t. the underlying causal model.

More formally, we will assume that the underlying causal variables C^t with size K can be partitioned in C_{ch}^t with size K_{ch} and C_{sh}^t with size K_{sh}. The source causal variables C_S^t can be then defined as $C_S^t = (h_S(C_{ch}^t), C_{sh}^t)$, where h_S is an invertible function. Similarly, the target causal variables are defined as $C_T^t = (h_T(C_{ch}^t), C_{sh}^t)$ for an invertible h_T. We denote with K_S the number of source causal variables and with K_T the number of target causal variables. The number of causal variables may change between source and target, as well as with respect to the underlying causal variables. Hence, we allow for refinement or coarsening of variables. However, the invertible functions h_l, h_s imply that the joint dimensionality of the causal variables is always constant.

3.2 Composition of causal representations.

We can extend the same notation to the case of composition, in which there are multiple source environments and a single target environment. We again assume that there is an underlying causal model with variables C^t. Let $C_{S_i}^t$ be the source causal variables of one of the L sources, and define $C_{sh_i}^t$ as the shared causal variables between the S_i-th source and the target environment. We assume that the target causal variables C_T^t are a composition of source causal variables that have been independently learned on the source environments. More formally:

$$C_T^t = (h_T(C_{ch_T}^t), C_{sh}^t, C_{sh_1}^t, \dots, C_{sh_L}^t), \tag{1}$$

where C_{sh}^t are the target causal variables shared with the underlying causal graph and $C_{ch_T}^t$ are the causal variables that are changed in the target environment with respect to the underlying causal variables through the invertible function h_T. If the shared causal variables C_{sh_i} are not disjoint, then the intersections will still be identical, and we can remove the duplicates.

4 Detection, Adaptation and Composition of Factors

Here we describe our framework DECAF and show how it adapts or composes causal representations in environments that follow our generative model. We first introduce how we detect the changed causal variables, based on the discrepancies in predicting the intervention targets. We then describe how we adapt the changed factors with a normalizing flow and how we compose causal representations.

Changing variable detection. Using a CRL approach adapted to the TRIS setting, as described in Sec. 2, we can learn a causal representation on the source data. We also learn a *target classifier* [17] that predicts the next step intervention targets I_i^{t+1} from the current latent state z^t and the next step latent state assigned to the causal variable C_i, which we denote as $z_{\psi_i}^{t+1}$. Intuitively, when we run the target classifier in the target environment, we expect that its accuracy would drop for the causal variables that have changed from the source to the target. In particular, for $k \in [\![1..K]\!]$, we define $X_{S,I_k=1} := \{X_S^t \mid I_k^t = 1, t \in [\![1..T]\!]\}$ as the set of observations on the source environment S in which C_k has been intervened upon. Similarly let $X_{T,I_k=1} := \{X_T^t \mid I_k^t = 1, t \in [\![1..T]\!]\}$ be the set of observations on the target environment T in which C_k has been intervened upon.

We define $\texttt{FPR}_{S,i}^k(j)$ and $\texttt{FNR}_{S,i}^k(j)$ as the False Positive Rate and False Negative Rate for intervention predictions of the classifier on the *source environment*, taking as inputs the latents learned from $X_{S,I_k=1}$ and the latent assigned to the variable z_{ψ_i} and predicting the intervention target I_j. Similarly, we define the False Positive Rate and the False Negative Rate for intervention predictions on the *target environment* as $\texttt{FPR}_{T,i}^k(j)$ and $\texttt{FNR}_{T,i}^k(j)$. We detect the changing causal factors $\hat{C}_{ch}$ by considering differences in false positive rates or false negative rates greater than threshold τ:

$$\hat{C}_{ch} = \{j \mid \exists\, i, k \in [\![1..K]\!] \; s.t. \; |\texttt{FPR}_{T,i}^k(j) - \texttt{FPR}_{S,i}^k(j)| > \tau \vee |\texttt{FNR}_{T,i}^k(j) - \texttt{FNR}_{S,i}^k(j)| > \tau\}. \tag{2}$$

As the target classifier generally predicts an intervention when the dynamics differ from the learnt observational ones, it tends to over-predict interventions in unseen environments. Thus, we found that using FPR to consistently outperforms using FNR and apply it throughout our experiments.

Adaptation. Once we have identified the changing causal variables C_{ch}, we adapt their representation $z_{ch} \in \mathbb{R}^{M_{ch}}$ by a Normalizing Flow (NF) [26]. The Normalizing Flow maps z_{ch} to a new representation $r \in \mathbb{R}^{M_{ch}}$ with the same dimensionality, while guaranteeing invertibility between the representations. Similarly to CITRIS [17], we train this flow with a transition prior p_ϕ parameterized by ϕ and condition each latent on exactly one intervention target I_{ch} of the changing variables:

$$p_\phi(r^{t+1} \mid r^t, I_{ch}^{t+1}) = \prod_{C_{ch_i} \in C_{ch}} p_\phi(r_{\psi_{ch_i}}^{t+1} \mid r^t, I_{ch_i}^{t+1}), \tag{3}$$

where ψ_{ch_i} is the learnt assignment of the components of z_{ch} to the causal variable C_{ch_i}, $\psi_{ch} : [\![1..M_{ch}]\!] \to [\![1..K_{ch}]\!]$. The model is optimized to maximize the log-likelihood of the target samples:

$$\mathcal{L}_{\text{MLE}}^{\phi,\omega} = \log p_{z_{ch}}(z_{ch}) = \log p_\phi(\texttt{NF}_\omega(z_{ch})) + \log \left| \det \frac{d\,\texttt{NF}_\omega(z_{ch})}{d\,z_{ch}} \right|, \tag{4}$$

where $\texttt{NF}_\omega$ represents the normalizing flow with parameters ω, and the original representation z_{ch} is kept frozen. During inference, we construct the final representation by replacing the changed causal variables z_{ch} with the adapted representation $r = \texttt{NF}_\omega(z_{ch})$.

Composing causal factors. Besides adapting causal representations, we can also try to compose the representations that we have identified across a set of source environments, to form the causal representation of a new target environment. More formally, consider the representation of L source environments $z_{S_l}, l = [\![1..L]\!]$. First, we detect the causal variables C_{sh_l} that are shared between each source representation C_{S_l} and the target using the changing variable detection described previously. In a second phase, we then concatenate the latent representation of all identified shared variables, i.e. $z_{\text{target}} = \{z_{sh_l} | l \in [\![1..L]\!]\}$. With that, we construct a representation that identifies the causal variables in the target environment if all causal variables can be found in the provided source environments.

5 Experiments

We evaluate DECAF on three benchmark datasets: the Voronoi Benchmark [18] which is a synthetic image dataset with random neural networks modeling the causal mechanisms; InterventionalPong [17] which simulates an Atari Pong game [1] with six causal variables; and Temporal Causal3DIdent [17, 32], a visually challenging dataset of 3D rendered objects with 10 causal variables describing the object and scene properties, see App. A.1. On these datasets, we apply DECAF to four different CRL approaches: CITRISVAE [17], LEAP [35], DMSVAE [15] and iVAE [11].

Baselines. For adapting causal representations from a source to a target environment, we compare DECAF to two simple adaptation baselines for reference, for each of the CRL methods: (1) `0shot`, where the model trained on the source environment is frozen and directly evaluated on the target data, and (2) `ft`, where the source model is fine-tuned on the target data.

Evaluation metrics. We evaluate the approaches based on the correlation between inferred latents and the ground truth causal factors, as estimated using the R^2 coefficient of determination [33] and Spearman's rank correlation [29]. For methods that only identify the causal variables up to permutations, we follow previous works [15, 17, 19] by assigning latents to the ground truth causal variable with the highest correlation. This results in a correlation matrix where the diagonal shows the correlation between matched learned and ground truth causal variables (high better), and off-diagonal elements the correlation to other variables (lower better). We propose a summary metric similar to the F1 score that combines the *average diagonal correlation* `diag` and the average max off-diagonal correlation `off_diag` through a harmonic mean. `diag` is intuitively similar to recall, while $(1 - \texttt{off_diag})$ is similar to precision. We define then the *Combined Correlation* (CC) as:

$$CC = 2\,\frac{\texttt{diag}(1 - \texttt{off_diag})}{\texttt{diag} + (1 - \texttt{off_diag})}. \tag{5}$$

A perfectly disentangled model achieves a score of $CC = 1$, while it decreases for models that have low correlation between its identified latents and the ground truth causal variables (low `diag`), or large cross correlation across variables (high `off_diag`). Full results are reported in App. B.

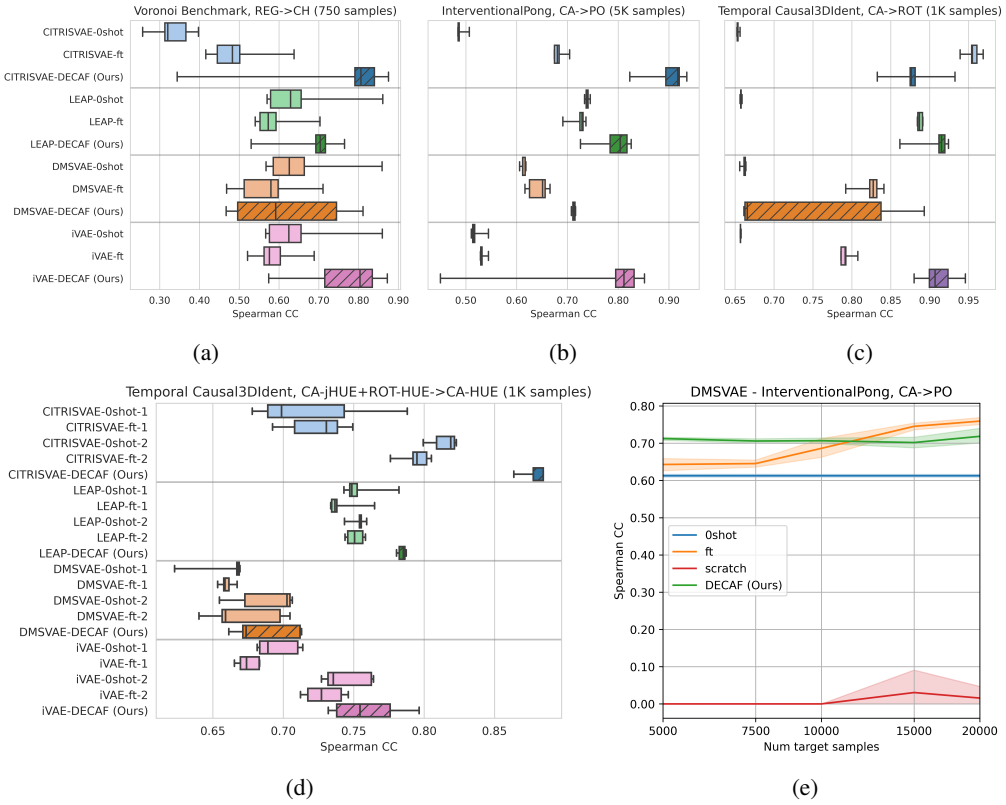


Figure 2: Spearman Combined Correlation (↑) of inferred latents to the ground truth changed variables. In the boxplot, CRL approaches are color-coded. The proposed method has a darker color. Adaptation in **(a)** Voronoi Benchmark, **(b)** InterventionalPong, **(c)** Temporal Causal3DIdent; **(d)** Composition for Temporal Causal3dIdent with sources `CA-jHUE` and `ROT-HUE`; **(e)** Effect of target samples on adaptation in InterventionalPong for DMSVAE.

Adaptation of causal representations. We conduct experiments using the Voronoi Benchmark with six causal variables, implementing the change from a source environment (REG) to the target (CH) by applying a randomly initialized NF on three causal variables. This simulates a coordinate system change for these three variables, with interventions being applied in the new system. Results on 750 data points from the target dataset over five seeds are presented in Fig. 2a. Both the baselines and the DECAF approaches show high dependency on the source-to-target variation, as evidenced by the performance variance. Yet, all DECAF approaches achieve a high CC score and outperform the baselines for CITRIS, LEAP and iVAE, showing its benefit and efficiency of adapting its source representation. In the InterventionalPong dataset, we change the ball coordinate system from Cartesian in the source domain to polar in the target (`CA` $\rightarrow$ `PO`), providing 5K samples in the target environment. For all considered CRL methods, the combination with DECAF beats the adaptation baselines as seen in Fig. 2b, although notable performance drops are observed for both iVAE and LEAP methods, where for one seed, the classifier fails to separate intervention targets. With more samples, `ft` can catch up to DECAF as seen in Fig. 2e. Yet, the low performance of training from scratch shows the importance of adaptive representations. Finally, in Temporal Causal3DIdent, we investigate the adaptation of the object position variables to a ROTated (ROT) x-y coordinate system, `CA` $\rightarrow$ `ROT` with 1K total samples, see Fig. 2c averaged over 5 seeds. As can be noted, DECAF is still competitive in the more visually complex scenario. While fine-tuning proves effective for CITRISVAE, in the other settings DECAF improves over the baselines. Notably, the DMSVAE approach exhibits high variance, with three runs detecting only one of the two changed variables.

Composition of causal representations. In the Temporal Causal3DIdent dataset, we consider two source environments: one with Cartesian position and jointly intervened hue (CA-jHUE), and another with a rotated coordinate system for position but independent interventions on hue (ROT-HUE), `CA-`

jHUE+ROT-HUE→ CA-HUE. The target environment composes the Cartesian position of the first source environment with the individual hue variables of the second environment, requiring the algorithms to identify which variables can be reused and combined from the sources. As shown in Fig. 2d, DECAF finds the correct variables to compose and, especially for CITRIS, provides significant gains over the baselines, while only requiring 1k samples. For results on other datasets, refer to App. B.

6 Conclusions

We introduce DECAF, a framework that is a first step towards adapting and composing causal representations. Our approach detects changing causal variables in a new environment and provides a method to adapt them with a limited amount of target samples.

7 Acknowledgments

We would like to thank Fan Feng, Gianluca Scarpellini and Andrea Maracani for the useful discussions. We gratefully acknowledge the HPC infrastructure and the Support Team at Fondazione Istituto Italiano di Tecnologia.

References

[1] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[2] Johann Brehmer, Pim De Haan, Phillip Lippe, and Taco Cohen. Weakly supervised causal representation learning. In *Advances in Neural Information Processing Systems*, 2022.

[3] Thomas Dean and Keiji Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence*, 5(2):142–150, 1989. doi: https://doi.org/10.1111/j.1467-8640.1989.tb00324.x.

[4] Andrea Dittadi, Frederik Träuble, Francesco Locatello, Manuel Wuthrich, Vaibhav Agrawal, Ole Winther, Stefan Bauer, and Bernhard Schölkopf. On the transfer of disentangled representations in realistic settings. In *International Conference on Learning Representations*, 2021.

[5] Andrea Dittadi, Samuele Papa, Michele De Vita, Bernhard Schölkopf, Ole Winther, and Francesco Locatello. Generalization and robustness implications in object-centric learning. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162, pages 5221–5285. PMLR, July 2022.

[6] Frederick Eberhardt. Causation and intervention. *Unpublished doctoral dissertation, Carnegie Mellon University*, page 93, 2007.

[7] Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. In *International conference on machine learning*, pages 881–889. PMLR, 2015.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[10] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*, 2017.

[11] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR, 2020.

[12] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

[13] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

[14] Sébastien Lachapelle, Tristan Deleu, Divyat Mahajan, Ioannis Mitliagkas, Yoshua Bengio, Simon Lacoste-Julien, and Quentin Bertrand. Synergies between disentanglement and sparsity: a multi-task learning perspective. *arXiv preprint arXiv:2211.14666*, 2022.

[15] Sebastien Lachapelle, Pau Rodriguez, Yash Sharma, Katie E Everett, Rémi LE PRIOL, Alexandre Lacoste, and Simon Lacoste-Julien. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA. In *First Conference on Causal Learning and Reasoning*, 2022.

[16] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M Asano, Taco Cohen, and Efstratios Gavves. Intervention Design for Causal Representation Learning. In *UAI 2022 Workshop on Causal Representation Learning*, 2022.

[17] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M Asano, Taco Cohen, and Stratis Gavves. CITRIS: Causal identifiability from temporal intervened sequences. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 13557–13603. PMLR, 17–23 Jul 2022.

[18] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M Asano, Taco Cohen, and Efstratios Gavves. Causal Representation Learning for Instantaneous and Temporal Effects in Interactive Systems. In *The Eleventh International Conference on Learning Representations*, 2023.

[19] Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M Asano, Taco Cohen, and Efstratios Gavves. BISCUIT: Causal representation learning from binary interactions. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pages 1263–1273. PMLR, 31 Jul–04 Aug 2023.

[20] Francesco Locatello, Gabriele Abbati, Thomas Rainforth, Stefan Bauer, Bernhard Schölkopf, and Olivier Bachem. On the fairness of disentangled representations. *Advances in neural information processing systems*, 32, 2019.

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[22] Joris M Mooij, Sara Magliacane, and Tom Claassen. Joint causal inference from multiple contexts. *The Journal of Machine Learning Research*, 21(1):3919–4026, 2020.

[23] Kevin Patrick Murphy. *Dynamic bayesian networks: representation, inference and learning*. University of California, Berkeley, 2002.

[24] Judea Pearl. *Causality*. Cambridge university press, 2009.

[25] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[26] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015.

[27] Karsten Roth, Mark Ibrahim, Zeynep Akata, Pascal Vincent, and Diane Bouchacourt. Disentanglement of correlated factors via hausdorff factorized support. *arXiv preprint arXiv:2210.07347*, 2022.

[28] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

[29] C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904.

[30] Frederik Träuble, Elliot Creager, Niki Kilbertus, Francesco Locatello, Andrea Dittadi, Anirudh Goyal, Bernhard Schölkopf, and Stefan Bauer. On disentangled representations learned from correlated data. In *International Conference on Machine Learning*, pages 10401–10412. PMLR, 2021.

[31] Sjoerd Van Steenkiste, Francesco Locatello, Jürgen Schmidhuber, and Olivier Bachem. Are disentangled representations helpful for abstract visual reasoning? *Advances in Neural Information Processing Systems*, 32, 2019.

[32] Julius Von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. *Advances in neural information processing systems*, 34: 16451–16467, 2021.

[33] S. Wright. *Correlation and Causation*. 1921.

[34] Weiran Yao, Guangyi Chen, and Kun Zhang. Temporally Disentangled Representation Learning. In *Advances in Neural Information Processing Systems 35, NeurIPS*, 2022.

[35] Weiran Yao, Yuewen Sun, Alex Ho, Changyin Sun, and Kun Zhang. Learning Temporally Causal Latent Processes from General Temporal Data. In *International Conference on Learning Representations*, 2022.

A Experimental Details

In this section we provide further details on the experimental setup, including information on the datasets A.1 and implementation details A.2.

A.1 Datasets

Voronoi benchmark We considered the non-instantenous version of the Voronoi benchmark [18] rendering colored Voronoi tiles whose colors are a mixed version of the ground truth generating factors. The underlying causal representation model is synthetically generated: starting from a random DAG, each variable is evaluated as sample from a Gaussian centered on the output of the mechanism randomly initialized neural network. Finally, the variables are mixed by a random normalizing flow and depicted as colors of a fixed-structure Voronoi diagram. We experimented with the 6 variables version of the dataset where all variables undergo perfect interventions. To allow for the change, we generated a version of the dataset where the 3 changed variables are fed to a randomly initialized NF. We denote with `REG` and `CH` the regular and changed versions of the dataset, respectively. In another version of the dataset, we enable for joint interventions on a group of 2 variables, while making sure there is no overlap between changed and coarse variables. We denote the coarse version of the dataset as `j`.

Interventional Pong We generated sequential data starting from InterventionalPong [17], based on the known Atari game Pong [1]. Six high-level causal variables underlie the generated data: `ball-pos-x`, `ball-pos-y`, `paddle-left-y`, `paddle-right-y`, `score-left`, `score-right`. The game dynamics follow two paddles playing one versus the other with the aim to score, i.e., let the ball go over the opponent's line of movement. The ball bounces on horizontal boundaries. Interventions are available for all causal variables, the scores are considered as a coarse variable. We generated multiple versions of the dataset, depending on different parameterizations of the interventions. Specifically, we considered a polar version where the ball position is represented in a polar coordinate system whose origin is the centre of the playground. Hence, the latents associated with the ball position ball-pos-x and ball-pos-y are replaced by `ball-pos-radius` and `ball-pos-angle`. Further, we considered coarse cases where a group of causal variables is always jointly intervened on and, hence, cannot be disentangled. In particular, we focused on the granularity of interventions associated with the paddles that could be independently (`PA`) or jointly (`jPA`) intervened.

Temporal Causal3DIdent We adapted the common benchmark of Temporal Causal3DIdent from [32] based on the temporal version in [17]. Samples visualize a rubber 3D object in the centre of a rendering scene. The dynamics are based on trigonometric functions. Observations follow 10 causal factors: `pos-x`, `pos-y`, `pos-z`, `rot-`α, `rot-`β, `rot-spotlight`, `hue-obj`, `hue-spotlight`, `hue-background`, `obj-shape`. All causal factors are subject to interventions. We adapted the dataset to support a different parameterization of the object position and different intervention granularities. Precisely, we generated a version of the dataset with rotated z-axis of 30 degrees. As a consequence, the xy coordinate system is rotated by 30 degrees anticlockwise. We indicate the rotated version of pos-x, pos-y with `rot-pos-x`, `rot-pos-y` respectively. We considered different levels of coarsening for the hue variables and denote as `jHUE` the version of the dataset where hue variables are jointly intervened.

A.2 Implementation details and hyperparameters

A.2.1 Source models

VAE architecture In order to have the different CRL approaches achieving their highest performance on source environments, we tested different variants of the same architecture. A convolutional encoder outputs the mean and standard deviation parameters of independent Gaussians. After sampling, the embeddings are decoded for reconstruction. For computational reasons, the specific architecture depends on the dataset. In Voronoi Benchmark and InterventionalPong the encoder is a 5-layer CNN + 2-layer MLP with a hidden dimension of 32. The decoder uses a symmetric architecture to the encoder (2-layer MLP and 5-layer deconv). In Temporal Causal3DIdent we followed the architecture in [17] and employed a 10-layer CNN and a 10-layer Resnet [8] decoder with a hidden dimension of 64.

On Voronoi Benchmark and Interventional Pong datasets we found an autoregressive flow prior [13, 26] to be beneficial on CITRISVAE and DMSVAE, following the architecture in [17]. The Gaussian samples from the encoder are fed to a 4-layer normalizing flow including Activation Normalization [12], Invertible 1×1 convolutions [12] and autoregressive affine coupling layers.

Transition prior The transition prior accepts as input the current time step and some auxiliary information to predict the next time step. In CITRISVAE the transition prior is a 2-layer MLP fed with z^t and I^{t+1} as input to predict z^{t+1}. Other baselines employ a 3-layer MLP. Following [17], we adapted the iVAE prior to accept as input the concatenation of the current time step z^t and the intervention target I^{t+1}. Similarly, both LEAP and DMSVAE priors accept as input a masked version of the concatenation $[z^t, I^{t+1}]$ where the mask is learned during training. Due to the density of the temporal graph of both Voronoi Benchmark and InternventionalPong, we found that restricting DMSVAE to learn the action mask only proved beneficial for the approach.

All the source models are trained with a batch size of 512 samples using AdamW [21] optimization with a learning rate of 1e-3 and Cosine Warmup scheduler. We used the Swish [9, 25] non-linearity. We regularized the source models to avoid overfitting on the source data by controlling for the source training epochs and adding a L^2-norm loss on the representation with hyperparameter β_{reg}. We summarized the used hyperparameters in Tab. 1.

A.2.2 Adaptation and Composition

Fine-tuning. The fine-tuning approach resumes the training of the model with the same causal representation strategy of the source model, e.g., a model pre-trained with the CITRISVAE strategy adapts to the new environment using the same CITRISVAE algorithm. Fine-tuning adapts the model with 2500 epochs and a batch size of 512 using AdamW optimizer with a learning rate of 1e-3 and Cosine Warmup scheduler.

Adaptation. We implemented the adaptation approach using an autoregressive normalizing flow [26] following [17]. The flow is based on the MADE [7] architecture with 16 neurons per latent variable. The flow includes Activation Normalization and 1×1 invertible convolutions. The depth of the flow depends on the dataset. As a flow prior, we employed a 2-layer autoregressive network that follows the same MADE architecture as the normalizing flow. For each latent variable, the flow outputs the parameters of a Gaussian distribution. DECAF adapts the model in 5000 epochs with a batch size of 1024 samples. We optimize using AdamW with a learning rate of 1e-2 and weight decay 5e-3. We applied the same Cosine Warmup scheduler as in the fine-tuning strategy.

Composition. DECAF stitches together the causal factor representations of modules that are detected to be invariant with respect to the target environement. Since the latent to factors assignment allows for a variable number of latents per factor, we cannot guarantee that the resulting representation matches the dimensionality of the pretrained autoencoder. To this end, we learn a projection function ρ projecting the representation to the same dimensionality as the source embedding. Thus, we freeze the representation model and learn it on the source data via reconstruction. In practice we parameterize ρ with a 2-layer feedforward network having 128 hidden dimensionality and Swish non-linearity. The projection function is trained with AdamW, a learning rate of 1e-3 and batch size 512.

We report the hyperparameters used for adaptation in Tab. 2.

B Full Results

Tables 3, 4, and 5 report the complete results on the adaptation setting using the correlation diagonal (diag) and off-diagonal (off-diag). Similarly, Tables 6, 7, 8 report the correlation metrics for the considered composition settings.

Voronoi benchmark/InterventionalPong				
Hyperparameter	CITRISVAE	LEAP	DMVAE	iVAE
Learning rate	—- 1e-3 —-			
Learning rate warmup	—- Cosine Warmup (100 steps) —-			
Optimizer	—- AdamW [21] —-			
Batch size	—- 512 —-			
Number of epochs	75(V)/125(P)	100(V)/200(P)	75(V)/175(P)	100(V)/ 200(P)
KLD Factor (β)	—- 1.0 —-			0.5
Num latents	—- 16 —-			
Model variant	VAE+NF	VAE	VAE+NF	VAE
Encoder	—- 5 layer CNN + 2 linear layers —-			
Prior layers	2	3	3	3
Decoder	—- 5 layer (deconv-)CNN + 2 linear layers —-			
Hidden dimensionality	—- 32 —-			
Activation function	—- Swish [25] —-			
Target classifier weight	2	—- n.a. —-		
Sparsity regularizer	n.a	—- 0.01 —-		n.a.
Discriminator weight	n.a	0.05	—- n.a. —-	

Temporal Causal3DIdent dataset				
Hyperparameter	CITRISVAE	LEAP	DMVAE	iVAE
Learning rate	—- 1e-3 —-			
Learning rate warmup	—- Cosine Warmup (100 steps) —-			
Optimizer	—- AdamW [25] —-			
Batch size	—- 512 —-			
Number of epochs	—- 600 —-			
KLD Factor (β)	—- 1 —-			
Num latents	—- 32 —-			
Model variant	VAE+NF	VAE	VAE+NF	VAE
Encoder	—- 10-layer CNN —-			
Prior layers	2	3	3	3
Decoder	—- 10-layer ResNet —-			
Hidden dimensionality	—- 64 —-			
Activation function	—- Swish [25] —-			
Target classifier weight	2	—- n.a. —-		
Sparsity regularizer	n.a	—- 0.01 —-		n.a.
Discriminator weight	n.a	0.1	—- n.a. —-	

Table 1: Summary of the hyperparameters for all source models trained on the Voronoi benchmark, InterventionalPong and Temporal Causal3DIdent dataset,

DECAF Adaptation			
Hyperparameter	Voronoi Benchmark	InterventionalPong	Temporal Causal3DIdent
Learning rate	—- 1e-2 —-		
Learning rate warmup	—- Cosine Warmup (100 steps) —-		
Optimizer	—- AdamW [21] —-		
Batch size	—- 1024 —-		
Number of epochs	—- 5000 —-		
KLD Factor (β)	—- 1 —-		
Hidden dimensionality	—- 64 —-		
Activation function	—- Swish [25] —-		
Target classifier weight	—- 2 —-		
Num flows	2	—- 4 —-	
At Least one (β_{ALO})	4	—- 2 —-	
L^2-Norm regularizer (β_{reg})	4	—- 2 —-	
Changed module threshold (τ)	0.15	0.2	0.1

Fine-tuning			
Hyperparameter	Voronoi Benchmark	InterventionalPong	Temporal Causal3dIdent
Learning rate	—- 1e-3 —-		
Learning rate warmup	—- Cosine Warmup (100 steps) —-		
Optimizer	—- AdamW [21] —-		
Batch size	—- 512 —-		
Number of epochs	—- 2500 —-		

DECAF Composition			
Hyperparameter	Voronoi Benchmark	InterventionalPong	Temporal Causal3DIdent
Learning rate	—- 1e-3 —-		
Learning rate warmup	—- Cosine Warmup (100 steps) —-		
Optimizer	—- AdamW [21] —-		
Batch size	—- 512 —-		
Number of epochs	—- 10 —-		
Changed module threshold (τ)	0.15	0.2	0.1

Table 2: Summary of the hyperparameters used for addressing the target environment.

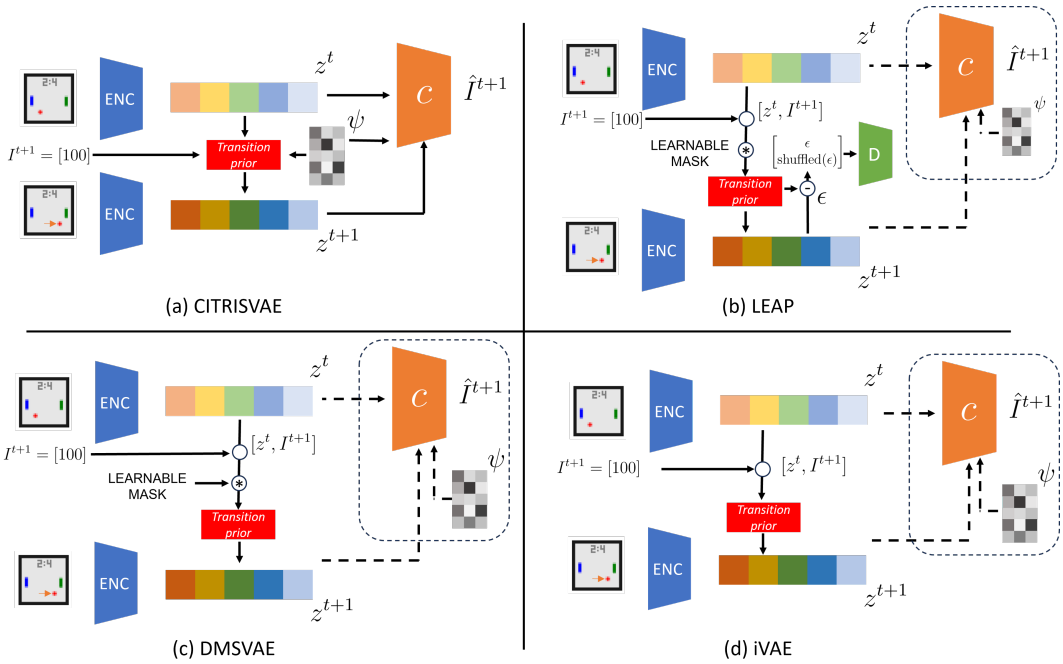
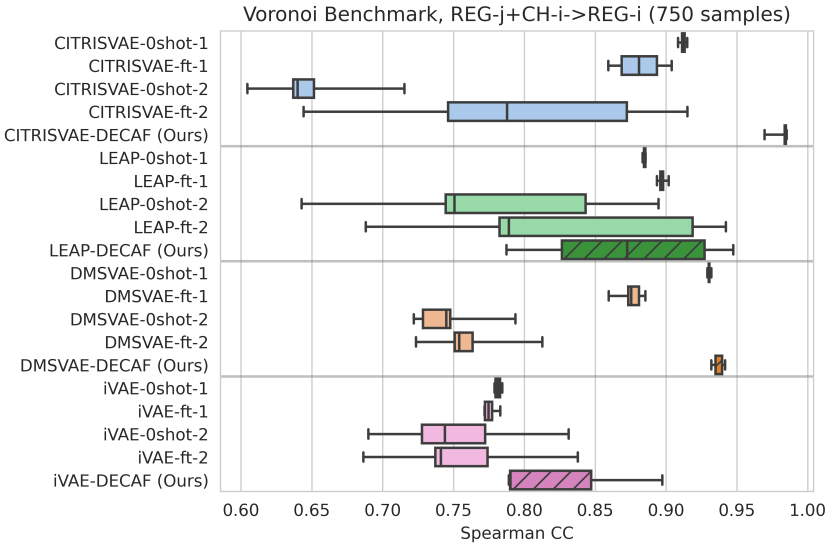
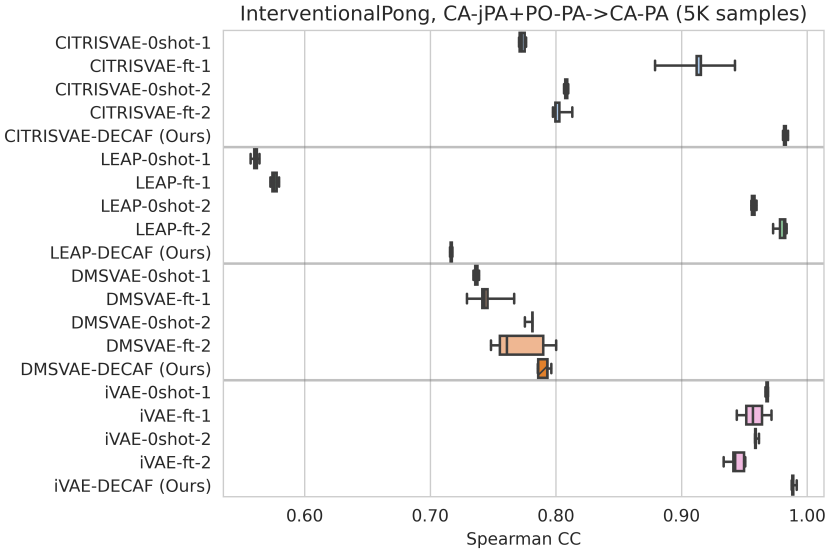


Figure 3: Visualization of the adaptation of the different CRL appoaches for the TRIS setting and learning of the target classifier on a pre-trained representation. (a) CITRISVAE [17] is used as is and makes available the classifier and the assignment ψ for later re-use. (b) LEAP [35]: since intervention targets are a source of non-stationarity, the previous time step and the intervention target are concatenated and masked to condition the LEAP transition prior. (c) DMSVAE [15] conditions the transition prior on the concatenation of previous time step and intervention targets, masked according to the learnt graph. (d) iVAE [11] conditions the prior on the concatenation of previous time step and intervention targets. For LEAP, DMSVAE and iVAE we learn a target classifier and the assignment ψ on top of the frozen representation.



(a) Voronoi Benchmark, REG-j+CH-i→REG-i (750 Samples)



(b) InterventionalPong, CA-jPA+PO-PA->CA-PA (5K Samples)

Figure 4: Composition of causal representations in (a) Voronoi Benchmark with sources `REG-j` and `CH-i`; (b) InterventionalPong with sources `CA-jPA` and `PO-PA`.

B.1 Other composition settings

In Figure 4 we report results on the composition setting in Voronoi Benchmark and InternvetionalPong.

B.2 Increasing number of target samples

In Figure 5 we report results on the adaptation of causal representations when increasing the number of target samples.

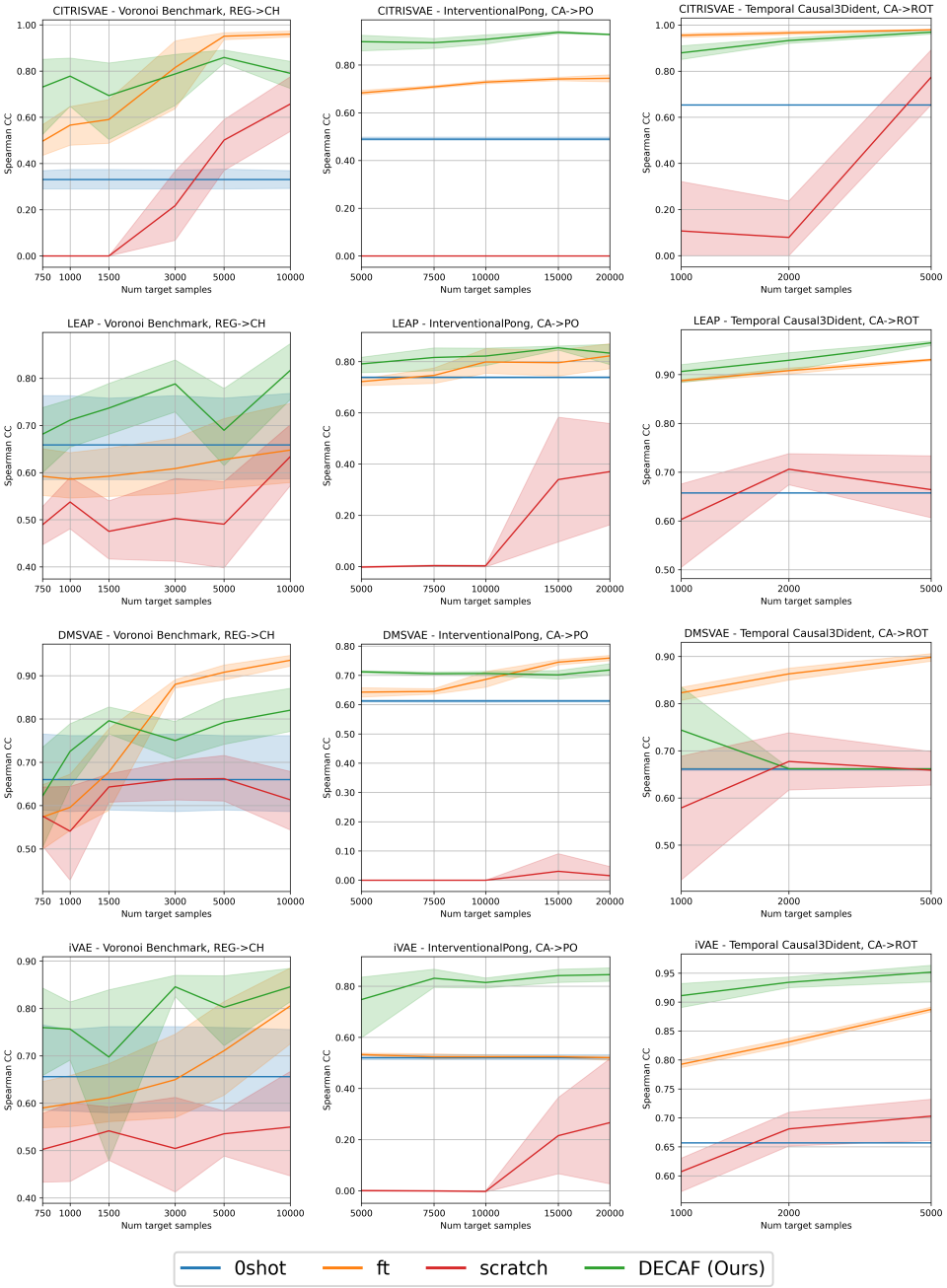


Figure 5: Adaptation with increasing number of target samples. **Rows**: CITRISVAE, LEAP, DMSVAE and iVAE. **Columns:** Voronoi Benchmark, InterventionalPong, Temporal Causal3dIdent.

Approach	Adaptation	R^2 diag ↑	R^2 off-diag ↓	Spearman diag ↑	Spearman off-diag ↓
CITRISVAE	0shot	0.24 ± 0.09	0.56 ± 0.05	0.40 ± 0.10	0.71 ± 0.03
	ft	0.37 ± 0.14	0.35 ± 0.07	0.57 ± 0.13	0.56 ± 0.06
	DECAF (Ours)	**0.72** ± 0.24	**0.18** ± 0.22	**0.83** ± 0.18	**0.34** ± 0.23
LEAP	0shot	0.67 ± 0.15	**0.23** ± 0.10	0.78 ± 0.13	**0.43** ± 0.12
	ft	0.60 ± 0.08	0.29 ± 0.07	0.74 ± 0.09	0.50 ± 0.07
	DECAF (Ours)	**0.79** ± 0.09	0.25 ± 0.12	**0.88** ± 0.06	0.44 ± 0.09
DMSVAE	0shot	**0.67** ± 0.16	**0.23** ± 0.10	**0.78** ± 0.13	**0.42** ± 0.12
	ft	0.63 ± 0.10	0.33 ± 0.10	**0.78** ± 0.06	0.55 ± 0.09
	DECAF (Ours)	0.54 ± 0.18	0.26 ± 0.14	0.70 ± 0.14	0.44 ± 0.16
iVAE	0shot	0.67 ± 0.16	0.23 ± 0.10	0.78 ± 0.14	0.43 ± 0.12
	ft	0.59 ± 0.08	0.29 ± 0.06	0.73 ± 0.09	0.50 ± 0.07
	DECAF (Ours)	**0.70** ± 0.16	**0.13** ± 0.11	**0.82** ± 0.11	**0.29** ± 0.13

Table 3: Voronoi Benchmark, REG -> CH (750 samples)

Approach	Adaptation	R^2 diag ↑	R^2 off-diag ↓	Spearman diag ↑	Spearman off-diag ↓
CITRISVAE	0shot	0.60 ± 0.01	0.60 ± 0.01	0.53 ± 0.01	0.55 ± 0.01
	ft	0.77 ± 0.01	0.34 ± 0.02	0.69 ± 0.01	0.33 ± 0.02
	DECAF (Ours)	**0.93** ± 0.03	**0.09** ± 0.04	**0.94** ± 0.03	**0.14** ± 0.06
LEAP	0shot	**0.85** ± 0.01	0.24 ± 0.01	**0.87** ± 0.01	0.36 ± 0.01
	ft	0.64 ± 0.02	**0.16** ± 0.02	0.72 ± 0.02	0.28 ± 0.02
	DECAF (Ours)	0.84 ± 0.04	0.18 ± 0.07	0.86 ± 0.06	**0.26** ± 0.03
DMSVAE	0shot	0.50 ± 0.01	0.25 ± 0.01	0.57 ± 0.00	0.33 ± 0.01
	ft	0.53 ± 0.04	0.18 ± 0.03	0.59 ± 0.04	0.30 ± 0.01
	DECAF (Ours)	**0.61** ± 0.01	**0.14** ± 0.01	**0.65** ± 0.01	**0.21** ± 0.01
iVAE	0shot	0.59 ± 0.04	0.53 ± 0.01	0.53 ± 0.03	0.49 ± 0.02
	ft	0.58 ± 0.03	0.51 ± 0.01	0.55 ± 0.02	0.48 ± 0.03
	DECAF (Ours)	**0.71** ± 0.17	**0.20** ± 0.19	**0.77** ± 0.19	**0.27** ± 0.15

Table 4: InterventionalPong, CA→PO (5K samples)

Approach	Adaptation	R^2 diag ↑	R^2 off-diag ↓	Spearman diag ↑	Spearman off-diag ↓
CITRISVAE	0shot	0.76 ± 0.00	0.28 ± 0.00	0.87 ± 0.00	0.48 ± 0.00
	ft	**0.95** ± 0.01	**0.01** ± 0.01	**0.98** ± 0.00	**0.06** ± 0.02
	DECAF (Ours)	0.92 ± 0.04	0.05 ± 0.03	0.96 ± 0.02	0.19 ± 0.06
LEAP	0shot	0.75 ± 0.00	0.28 ± 0.00	0.87 ± 0.00	0.47 ± 0.00
	ft	0.93 ± 0.00	0.07 ± 0.00	0.96 ± 0.00	0.18 ± 0.00
	DECAF (Ours)	**0.95** ± 0.01	**0.03** ± 0.02	**0.97** ± 0.01	**0.15** ± 0.04
DMSVAE	0shot	0.66 ± 0.03	0.23 ± 0.01	0.81 ± 0.02	0.44 ± 0.01
	ft	**0.81** ± 0.03	**0.09** ± 0.00	**0.90** ± 0.02	**0.24** ± 0.02
	DECAF (Ours)	0.73 ± 0.08	0.16 ± 0.10	0.85 ± 0.05	0.33 ± 0.15
iVAE	0shot	0.75 ± 0.00	0.28 ± 0.00	0.87 ± 0.00	0.47 ± 0.00
	ft	0.87 ± 0.00	0.15 ± 0.01	0.93 ± 0.00	0.31 ± 0.01
	DECAF (Ours)	**0.95** ± 0.02	**0.03** ± 0.01	**0.97** ± 0.01	**0.14** ± 0.04

Table 5: Temporal Causal3dIdent, CA→ROT (1K samples)

Approach	Adaptation	R^2 diag ↑	R^2 off-diag ↓	Spearman diag ↑	Spearman off-diag ↓
CITRISVAE	0shot-1	**0.99** ± 0.00	0.08 ± 0.00	**1.00** ± 0.00	0.16 ± 0.00
	ft-1	0.95 ± 0.01	0.06 ± 0.01	0.98 ± 0.00	0.20 ± 0.03
	0shot-2	0.60 ± 0.04	0.29 ± 0.05	0.69 ± 0.05	0.39 ± 0.04
	ft-2	0.79 ± 0.13	0.14 ± 0.10	0.87 ± 0.11	0.27 ± 0.11
	DECAF (Ours)	**0.99** ± 0.00	**0.00** ± 0.01	**1.00** ± 0.00	**0.03** ± 0.01
LEAP	0shot-1	0.91 ± 0.00	**0.08** ± 0.00	0.95 ± 0.00	0.17 ± 0.00
	ft-1	**0.93** ± 0.00	0.06 ± 0.00	**0.96** ± 0.00	**0.16** ± 0.00
	0shot-2	0.80 ± 0.13	0.15 ± 0.08	0.88 ± 0.08	0.30 ± 0.10
	ft-2	0.84 ± 0.13	0.11 ± 0.09	0.88 ± 0.11	0.22 ± 0.11
	DECAF (Ours)	0.90 ± 0.08	**0.08** ± 0.10	0.92 ± 0.09	**0.16** ± 0.11
DMSVAE	0shot-1	0.97 ± 0.00	**0.02** ± 0.00	**0.99** ± 0.00	0.12 ± 0.00
	ft-1	0.92 ± 0.01	0.08 ± 0.01	0.96 ± 0.00	0.20 ± 0.02
	0shot-2	0.77 ± 0.05	0.17 ± 0.03	0.85 ± 0.05	0.33 ± 0.03
	ft-2	0.78 ± 0.05	0.15 ± 0.03	0.87 ± 0.03	0.32 ± 0.03
	DECAF (Ours)	**0.98** ± 0.00	**0.02** ± 0.00	**0.99** ± 0.00	**0.11** ± 0.01
iVAE	0shot-1	**0.79** ± 0.00	0.17 ± 0.00	**0.87** ± 0.00	0.29 ± 0.00
	ft-1	**0.79** ± 0.00	0.15 ± 0.00	**0.87** ± 0.00	0.30 ± 0.01
	0shot-2	0.74 ± 0.09	0.17 ± 0.04	0.84 ± 0.06	0.32 ± 0.05
	ft-2	0.75 ± 0.09	0.16 ± 0.05	0.83 ± 0.07	0.30 ± 0.05
	DECAF (Ours)	0.73 ± 0.08	**0.04** ± 0.05	0.76 ± 0.10	**0.08** ± 0.08

Table 6: Voronoi Benchmark, REG-j+CH-i→REG-i (750 samples) with sources `REG-j` and `CH-i`.

Approach	Adaptation	R^2 diag ↑	R^2 off-diag ↓	Spearman diag ↑	Spearman off-diag ↓
CITRISVAE	0shot-1	0.80 ± 0.01	0.22 ± 0.00	0.88 ± 0.00	0.31 ± 0.00
	ft-1	0.92 ± 0.01	0.03 ± 0.01	0.95 ± 0.01	0.13 ± 0.03
	0shot-2	0.81 ± 0.00	0.17 ± 0.00	0.82 ± 0.00	0.20 ± 0.00
	ft-2	0.77 ± 0.01	0.10 ± 0.02	0.79 ± 0.02	0.18 ± 0.02
	DECAF (Ours)	**0.98** ± 0.00	**0.00** ± 0.00	**0.99** ± 0.00	**0.03** ± 0.00
LEAP	0shot-1	0.60 ± 0.01	0.42 ± 0.00	0.63 ± 0.00	0.49 ± 0.00
	ft-1	0.60 ± 0.01	0.40 ± 0.00	0.63 ± 0.00	0.47 ± 0.01
	0shot-2	0.98 ± 0.00	0.02 ± 0.00	0.99 ± 0.00	0.08 ± 0.00
	ft-2	**1.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00	**0.04** ± 0.01
	DECAF (Ours)	0.76 ± 0.00	0.30 ± 0.00	0.78 ± 0.00	0.34 ± 0.00
DMSVAE	0shot-1	0.76 ± 0.00	0.21 ± 0.00	0.83 ± 0.00	0.34 ± 0.00
	ft-1	0.79 ± 0.01	0.19 ± 0.01	0.83 ± 0.01	0.33 ± 0.02
	0shot-2	0.78 ± 0.00	0.13 ± 0.00	0.85 ± 0.00	0.28 ± 0.00
	ft-2	0.76 ± 0.02	**0.10** ± 0.03	0.80 ± 0.02	**0.26** ± 0.03
	DECAF (Ours)	**0.80** ± 0.00	0.13 ± 0.01	**0.86** ± 0.00	0.27 ± 0.01
iVAE	0shot-1	0.99 ± 0.00	0.01 ± 0.00	1.00 ± 0.00	0.06 ± 0.00
	ft-1	0.98 ± 0.00	0.01 ± 0.01	0.99 ± 0.00	0.08 ± 0.02
	0shot-2	0.96 ± 0.00	0.00 ± 0.00	0.97 ± 0.00	0.05 ± 0.00
	ft-2	0.98 ± 0.00	0.02 ± 0.01	0.99 ± 0.00	0.10 ± 0.01
	DECAF (Ours)	**1.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00	**0.02** ± 0.00

Table 7: InterventionalPong, CA-jPA+PO-PA→CA-PA (5K samples) with sources `CA-jPA` and `PO-PA`.

Approach	Adaptation	R^2 diag ↑	R^2 off-diag ↓	Spearman diag ↑	Spearman off-diag ↓
CITRISVAE	0shot-1	0.74 ± 0.06	0.25 ± 0.03	0.72 ± 0.05	0.28 ± 0.04
	ft-1	0.75 ± 0.03	0.21 ± 0.02	0.71 ± 0.03	0.26 ± 0.02
	0shot-2	0.83 ± 0.01	0.09 ± 0.00	0.84 ± 0.02	0.21 ± 0.01
	ft-2	0.80 ± 0.02	0.10 ± 0.01	0.78 ± 0.02	0.19 ± 0.02
	DECAF (Ours)	**0.88** ± 0.01	**0.04** ± 0.01	**0.88** ± 0.01	**0.12** ± 0.01
LEAP	0shot-1	0.75 ± 0.02	0.17 ± 0.02	0.74 ± 0.02	0.23 ± 0.01
	ft-1	0.72 ± 0.01	0.15 ± 0.01	0.71 ± 0.02	0.22 ± 0.01
	0shot-2	0.76 ± 0.00	0.15 ± 0.01	**0.78** ± 0.00	0.27 ± 0.01
	ft-2	0.74 ± 0.01	**0.12** ± 0.01	0.75 ± 0.01	0.24 ± 0.01
	DECAF (Ours)	**0.78** ± 0.00	**0.12** ± 0.01	**0.78** ± 0.00	**0.21** ± 0.01
DMSVAE	0shot-1	0.64 ± 0.02	0.25 ± 0.02	0.62 ± 0.02	0.29 ± 0.02
	ft-1	0.61 ± 0.02	**0.19** ± 0.01	0.60 ± 0.01	**0.27** ± 0.01
	0shot-2	**0.67** ± 0.03	0.21 ± 0.04	**0.66** ± 0.04	0.28 ± 0.01
	ft-2	0.63 ± 0.03	**0.19** ± 0.04	0.62 ± 0.04	**0.27** ± 0.01
	DECAF (Ours)	**0.67** ± 0.03	0.21 ± 0.02	**0.66** ± 0.04	0.28 ± 0.02
iVAE	0shot-1	0.68 ± 0.02	0.24 ± 0.01	0.64 ± 0.02	0.24 ± 0.01
	ft-1	0.64 ± 0.01	0.22 ± 0.00	0.62 ± 0.01	0.26 ± 0.00
	0shot-2	0.72 ± 0.02	0.16 ± 0.02	**0.73** ± 0.03	0.24 ± 0.01
	ft-2	0.71 ± 0.02	**0.14** ± 0.02	0.71 ± 0.02	0.26 ± 0.01
	DECAF (Ours)	**0.74** ± 0.02	0.15 ± 0.03	0.72 ± 0.03	**0.20** ± 0.02

Table 8: Temporal Causal3DIdent, CA-jHUE+ROT-HUE→CA-HUE (1K samples) with sources `CA-jHUE` and `ROT-HUE`.