École Polytechnique Fédérale de Lausanne

Investigating independence regularization for compositional out-of-distribution classification

by Ceraolo Roberto

Optional Semester Project Report

Prof. Alahi Alexandre
Project Advisor

Liu Yuejiang
Project Supervisor

EPFL IIC ENAC - VITA
CH-1015 Lausanne

January 18, 2023

Contents

Chapter 1

Introduction

Causal representation learning is a subfield of machine learning that focuses on learning representations of causal relationships among variables in a system. In other words, it aims to uncover the underlying mechanisms that govern how different variables in a system are related to each other and how changes in one variable can cause changes in others. The main goal of causal representation learning is to learn a causal graph, also known as a structural causal model, which is a directed acyclic graph (DAG) that represents the causal relationships among variables in a system. Each node in the graph represents a variable, and each directed edge represents a causal relationship between two variables. In order to ensure that the learned representations are accurate and generalizable, it is important to use appropriate evaluation metrics and consider potential confounding factors, such as unobserved variables, that can bias the results.

In general, creating machine learning models able to generalize to unseen distributions of data is a crucial problem if we want to unlock new possibilities of applications for AI. Indeed, there are several cases in which we cannot be sure that the training data contains all possible states of the world, for instance when dealing with autonomous driving systems.

In Causal Triplet [4], the authors propose a causal representation learning benchmark featuring pairs of images [2]. In each pair, an image contains several visual objects, one of which is intervened upon with a certain action, giving place to the other image in the pair. One of the challenges they tackle is to create a model able to predict the action class and to generalize to unseen compositions of object-action pairs. This is called compositional out-of-distribution (ood hereinafter) generalization. The core idea is that the model needs to learn the factors having a causal effect on the image itself. In order to do so, it needs to avoid learning spurious correlations in the data, namely memorizing links between objects and actions seen during training. To do so, we want the model to encourage statistical independence between the spuriously correlated features.

Previous literature in causal representation learning uses mainly two methods to encourage independence between features: a loss factor based on Adversarial Training, as in Causal Triplet,

and one based on the Hilbert-Schmidt Independence Criterion (HSIC), namely in [1] and [6].

Adversarial training (AT) was shown to be sometimes unnecessary or even counterproductive. Moyer et al. [5] claim that AT adds an extra layer of complexity, due to the second optimization that needs to be done, in parallel to the first one. They also found that in some settings in which representations of data are invariant to changes in specified factors, adversarial training can be unnecessary. Also, in some cases, bad approximations, coming from weak adversaries, may provide bad gradient information, leading to poor performances.

This is why other alternatives, like HSIC, are worth exploring. This motivated the Semester Project, in which the HSIC loss was investigated in the context of Causal Triplet. The main phases of the project comprised:

- an in-depth theoretical study of the functioning of HSIC, understanding how and why it should encourage independence
- A theoretical comparison between HSIC and conditional HSIC, two different versions used in previous literature, to understand the most appropriate for the problem at hand
- An investigation of the balancing of classes, as the conditional HSIC relies on
- The implementation of the conditional HSIC
- The experimentation.

Chapter 2

Related work

There are four main clusters of previous literature that must be taken into account to fully frame the project. Firstly, we will outline what the HSIC measure is, how it works and how can it be helpful in this context. Secondly, we cover a previous seminal work in which the HSIC measure was used in a Causal Representation Learning setting. Thirdly, we look into a subsequent paper tackling the same problem (out-of-distribution generalization) via prototype propagation, together with HSIC. We also highlight their different approaches to the use of the HSIC measure to encourage independence. Finally, the Causal Triplet approach, using Adversarial Loss, is described.

2.1 HSIC

The Hilbert-Schmidt Independence Criterion (HSIC) [3] is a widely used method for measuring statistical dependence between two random variables. It is based on the Hilbert-Schmidt norm of the cross-covariance operator of the two variables, which is defined as the sum of the squares of the elements of the cross-covariance operator. The HSIC is a scalar value that ranges between 0 and 1, with 0 indicating independence and 1 indicating complete dependence.

One of the main advantages of the HSIC is that it is a non-parametric method, which means it does not require any assumptions about the underlying distributions of the variables. It can be used to test independence between two variables in both discrete and continuous settings, and it can also be extended to handle multiple variables.

The authors of the paper introducing HSIC, also provide an empirical estimator of it, since directly computing the HSIC is hard. Additionally, they provide a bias on the empirical estimator, which, assuming m is the number of datapoints on which is computed, behaves as $O(m^{-}1)$. Additional theoretical details about HSIC are available in the appendix.

In the present work, we used the empirical HSIC to build a regularization factor added to the loss of the main model, in order to encourage independence between the spuriously correlated features. In this work, when we mention the "HSIC regularizer" we mean the regularizer that uses the empirical estimator of HSIC.

2.2 HSIC in Causal Representation Learning

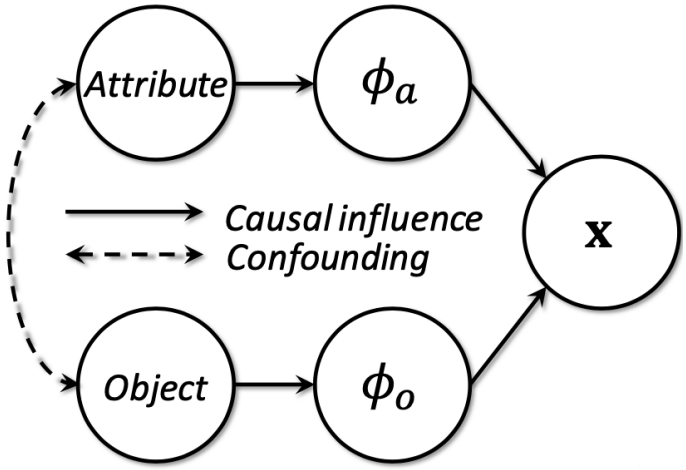
Yuval Atzmon and his colleagues, in the seminal paper "A causal view of compositional zero-shot recognition" [1] tackle a problem which is very close to Causal Triplet, and they frame it through the means of Causal Representation Learning.

Their paper presents a novel approach to compositional zero-shot recognition. They take a causal perspective on the problem, which means that it focuses on understanding the underlying cause-and-effect relationships between the objects in the image. In their setting, each image has two labels: an "Attribute" and an "Object" one.

The proposed approach uses a causal graphical model to represent the relationships between the objects in an image, and it uses this model to reason about unseen object compositions. By modeling the relationships between the objects in this way, the approach is able to reason about the unseen compositions in a more interpretable and robust way than previous methods.

The paper presents experimental results that demonstrate the effectiveness of the proposed approach on a number of benchmark datasets. The results show that the proposed approach outperforms previous methods and achieves state-of-the-art results in compositional zero-shot recognition. They claim the generative process of an image is the one shown in figure 2.1. They draw a confounding link between the object and the attribute, that prevents the model to generalize to unseen object-attribute combinations. Intuitively, if a model is shown thousands of white swans during training, it will infer that "being white" is one of the "causes" to be a swan, so it will hardly generalize to a black swan. In order to remove the confounding link, they use the HSIC: they add it as a regularization term in the objective function of their model. The authors of the paper argue that by promoting independence between the object representations and the object classes, the model will be more robust to unseen object compositions.

In their work, the authors introduce an embedding model that learns representations of two basic visual components of images. Such representations are disentagled and were learnt from confounded training data. The goal of the model is to predict unseen combinations of object-attribute pairs. The two "semantic" embedding spaces learnt are respectively for attributes and for objects. In the two spaces, each attribute has a dense prototypical representation ϕ_a belonging to a space Φ_a and each object has its representation ϕ_o belonging to a space Φ_o. When trying to infer such real-world attribute and object features, the model generates the embeddings $\hat{\phi}_o$ and $\hat{\phi}_a$.



Atzmon et al. in their paper, show the causal diagram (DAG, directed acyclic graph) in fig. 1 to describe the real-world generative process that created the images. The two categorical variables Attribute and Object cause the core features ϕ_a and ϕ_o which, together, cause the image to be as it is.

Figure 2.1: Causal diagram of the real-world process generating the image - Atzmon et al.

Because of the rules of *d-separation*, given the above causal graph, we have that $\phi_a \perp\!\!\!\perp O \mid A$, where O is the object label and A is the attribute's one. A more detailed explaination of this result is available in 5.2.

Nevertheless, ϕ_a and ϕ_o are not observed, we have their estimates $\hat{\phi}_o$ and $\hat{\phi}_a$ learnt by the model. The goal of the training process is to make the estimates as similar as possible to the unobserved core features. In order to do so, the key idea of the independence loss regularization in the paper is to try to make the estimates follow the causal relationship of the "theoretical" causal model they pictured in the graph. We stated above (and shown in the appendix) that, in these settings, the following holds: $\phi_a \perp\!\!\!\perp O \mid A$. Similarly, it can be shown that given the DAG in 2.1, we have: $\phi_o \perp\!\!\!\perp A \mid O$. For the estimates, in order to encourage $\hat{\phi}_a \perp\!\!\!\perp O \mid A$ the authors of the paper add the following independence component to the loss: $\frac{1}{|A|}\sum_{a\in A}\text{HSIC}(\hat{\phi}_a|A=a, O|A=a)$. By doing so, they are penalizing possible dependencies between the learnt embedding of the attribute and the object label, conditioned on the attribute label. Hence, they are encouraging the embeddings to behave following the causal graph of the latent representations.

2.3 HSIC in Prototype Propagation

"Independent Prototype Propagation for Zero-Shot Compositionality" [6] is a paper that presents a novel method for zero-shot compositionality. The proposed method, called Independent Prototype Propagation (IPP), aims to learn independent object prototypes for different object classes and to propagate these prototypes to unseen object compositions.

The IPP approach is based on the idea of independently training object prototypes for different object classes and then using these prototypes to reason about unseen object compositions. The approach uses a neural network to learn these prototypes and it uses a novel loss function that combines the negative log-likelihood of the data with the HSIC to encourage independence between the prototypes.

The authors of the paper evaluate the proposed method on a number of benchmark datasets and show that it outperforms previous methods in terms of zero-shot compositionality recognition. They also demonstrate that the use of the HSIC regularization term leads to better generalization performance and more interpretable results when dealing with unseen object compositions.

Ruis et. al, in their work, approach the compostional generalization problem in a different manner with respect to Atzmon and his co-authors. They do not frame the problem in a causal perspective. After using a ResNet-18 to find the local image patches x_{ij}, they compare the latter with prototype vectors through a similarity function and they obtain compatibility scores. Then, such scores are passed through a softmax function and then a weighted sum z_k of the feature map is computed, using as weights the outputs of the softmax. Such element is finally passed to HSIC, together with the label with which we want to encourage independence. This way independence between attribute and object prototypes is promoted. This implies that each patch will be affected by the independence loss depending on the strength of its appearance, since the output of the softmax is based on the similarity score. The formula for their independence loss is as follows:

$$\mathcal{L}_{hsic} = \lambda_h \frac{\text{HSIC}\left(z_a, O\right) + \text{HSIC}\left(z_o, A\right)}{2}$$

where z_a and z_o are respectively the softmax-pooled outputs of the attribute and the object prototype layers, A and O are the attribute and the object one-hot-encoded labels, and λ_h is a coefficient to be tuned. The striking differences between this approach and the one shown in [1] are: the absence of conditionality and the usage of a weighted feature map. While Atzmon and his colleagues considered the conditional independence between the feature map and the label when computing the independence loss, as it is suggested by their causal graph, Ruis and his co-authors considered the direct independence between the weighted feature map and the object (or attribute) label.

2.4 Independence via Adversarial Loss

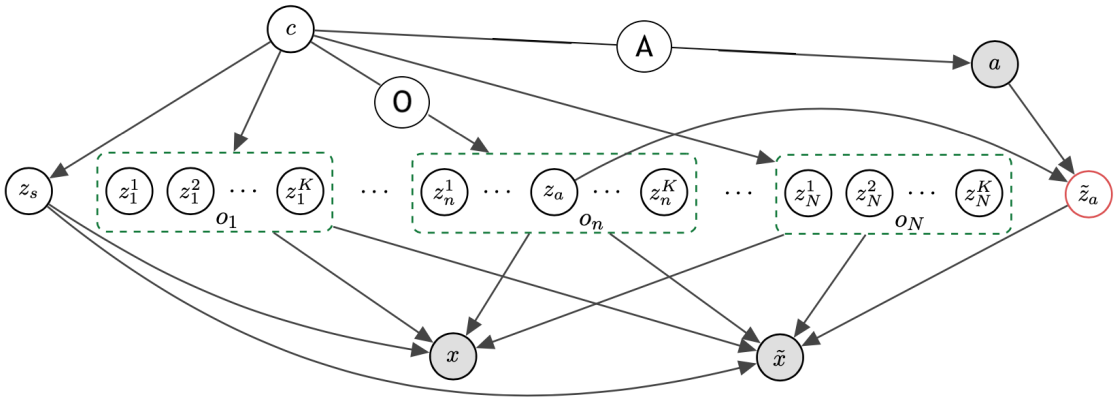


Figure 2.2: Causal diagram of the real-world process generating the image - Causal Triplet. The graph from the paper was slightly modified in order to explicitly introduce the object and action labels, useful to derive structural equations.

In Causal Triplet [4], the setting of the problem is put in a causal perspective, as was in the paper by Atzmon et al. [1]. Let's focus on the modelling of the data generation process, in order to understand the causal implications. The current assumptions are made:

- The set of latent vectors that cause the image consists of both global scene-level variables z_s and local object-level variables z_n^k, where n is the object index and k is latent index within the object group
- Each action has an effect only on a subset of the latent factors z_a. The action a is the intervention that was done on the image x, and $\tilde{x}$ is the image after the intervention. Specifically, z_a is the latent factor that was impacted by the intervention. $\tilde{z}_a$ is the factor after the intervention
- The latent factors and the actions are not independent due to some unobserved confounders c.

They model the process generating the data following the causal diagram in 2.2. In order to prevent the model to learn the spurious correlations between the object and the action label, they introduce a regularizer, based on adversarial training. The Independence component they add to the loss is the following:

$$\min_{f,\pi,\phi} \max_{\psi} \mathcal{L}_a(f,\pi,\phi) - \lambda_i \mathcal{L}_o(f,\pi,\psi)$$

where $\mathcal{L}_o$ is a cross-entropy loss for object classification, and λ_i is the coefficient of the independence loss. They find a significant performance increase of 14% on unseen compositions. The authors also note two necessary assumptions that also represent limitations to this loss:

- Given a dataset, the couple of features which are spuriously correlated must be known a priori
- The label of the spurious attribute must be provided.

It is worth noting that both the above assumptions need to hold also in the case of HSIC regularization. Given previous encouraging results about the usage of HSIC to promote independence, and given some downsides of Adversarial Training shown in [5] and mentioned here in **??**, HSIC regularization is worth exploring in Causal Triplet setting.

Chapter 3

Method

3.1 Conditional HSIC

Our goal was to implement HSIC in Causal Triplet in order to minimize the mututal information between the object and the action label. Given the previous literature, two main ways to do it could be viable: encouraging direct independence or conditional.
Let's consider again the CausalTriplet's DAG 2.2. Objects o_i are the constituting objects of the image, o_n is the one which is intervened upon. A and O are respectively the action and object labels.
In this case, through the independence loss, we want to promote statistical independence between the learned representation of the action and the respective object classes. The goal is to prevent the model to learn spurious correlations between the action and the object label, in order to improve its generalization capabilities. The independence relations we can infer are similar to what we found in the paper from Atzmon et al. Let's consider, the node O, representing the label of the object that was intervened upon. We want to verify the possible dependence between the object o_n and the action embedding a. From the DAG we can notice that there are two paths connecting the two nodes. If we include the action label A in the conditioning set, the two paths are *d-separated* (notes about *d-separation* are available at 5.2):

- The path connecting o_n and a that goes through c and A contains a chain, and the middle node (A) is in the conditioning set
- the path that goes through $\tilde{z}_a$ contains a collider such that the middle node is not in the conditioning set, nor its descendants are.

Hence, we can safely affirm that, given the above causal graph, $a \perp\!\!\!\perp O \mid A$, where O is the object label, A is the attribute's one and a is the latent action embedding.
In order to promote the above to hold with the non-latent representations of the factors, we need to ensure the following: $\hat{a} \perp\!\!\!\perp O \mid A$, where $\hat{a}$ is the learned representation of the action embedding. In order to do so, we add the following independence component to the loss: $\frac{1}{|A|}\sum_{a\in A}\text{HSIC}(\hat{a}|A=a, O|A=a)$. By doing so, we are penalizing possible dependencies between the learnt embedding of the action and the object label, conditioned on the action label.

Hence, we are encouraging the embeddings to behave following the causal graph of the latent representations.

3.2 Labels' distribution

Given the way we defined the conditional HSIC measure, it is computed for batches of points sharing the same action label. This means that biases in the distributions of classes among the datapoints may hurt the accuracy of the HSIC. Indeed, it can be shown [3] that the bias of the empirical estimator of HSIC grows as $O(m^{-1})$, where m is the number of points on which it is computed. Hence, we need to have enough instances in each batch to avoid biased results. A second, but not for importance, objective to have when looking at the classes, is to have similar class distributions between the training and the out-of-distribution sets. Indeed, the goal of the model is not to learn to predict in cases of different classes distributions, but different classes combinations.
This motivated an investigation about the labels' distribution. In Causal Triplet's settings, there are two distributions that need to be taken into account: the distribution of the action-object labels combinations and the distribution of the action labels. The optimal situation would be to have balanced classes in both cases, but it proved to be hard to balance them both at the same time.
From an investigation on the distribution of classes, we found significant differences between some labels. Two interventions were applied to the data in order to improve the balancing:

- The action, object combinations that had less than 100 instances in the dataset were discarded
- The action, object combinations that had more than 1000 instances were subsampled to 500

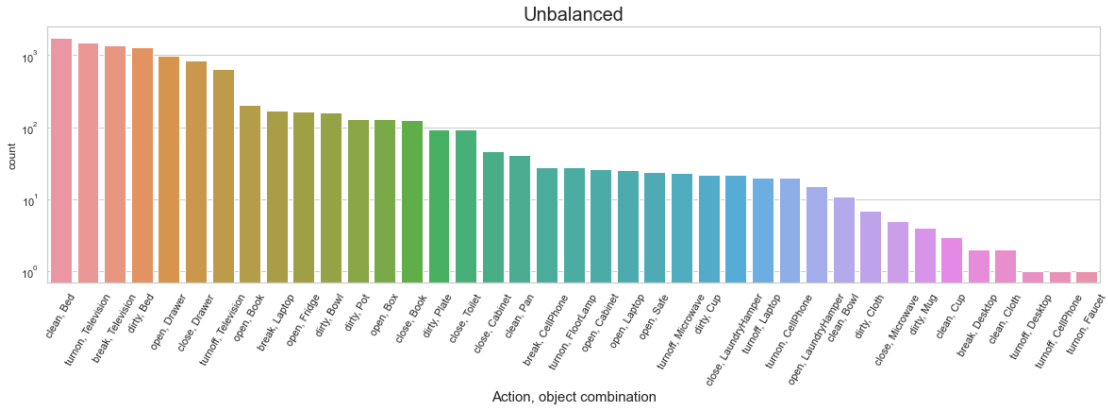


Figure 3.1: Train set combinations unbalanced

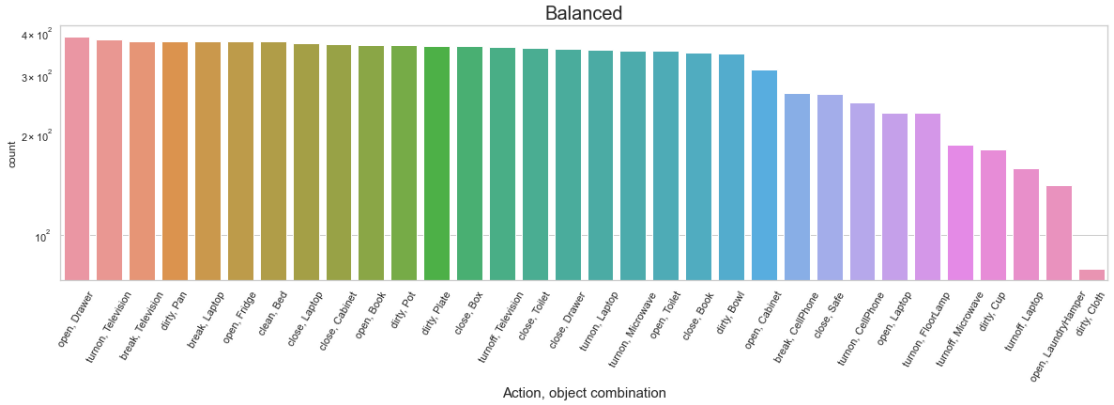


Figure 3.2: Train set combinations balanced

In figure 3.1 and 3.2 we see the effect of the new balancing procedure on the distribution of combinations in the training set. Clearly, there is an improvement in terms of differences in the number of instances per combination. Similar effects were obtained in the ood set.
For what concerns the action classes instead, in the training set we noticed a bit more imbalance after the cutoff and the subsampling, but a good distribution in the ood set. As mentioned above, balancing both the combinations and the action classes entails some trade-offs, and the proposed solution was deemed a good mid-point. All the distributions before and after the balancing are shown in the appendix 5.3.

Chapter 4

Experiment

In all experiments, the model was run with several different initializations (seeds) due to the partly stochastic nature of the model. All the experiments were run for 50 epochs, and the best ood accuracy over all the epochs was taken into consideration. The three main clusters of experiments we report here are the following: the effect of the cutoff and the subsampling described in 3.2, the effect of the HSIC regularizer, with different values for its coefficient, the effect of the Adversarial regularizer, as a mean of comparison, again with several different values for its coefficient.
After some coarse-grained testing for the fine-tuning of the coefficients, we found that sensible values, both for Adversarial and for HSIC regularizers, are between 0.1 and 1. Hence, in 4.2 we report the accuracies obtained with the following values for the coefficients: 0.01, 0.1, 0.2, 0.5 and 1.

4.1 Effect of balancing

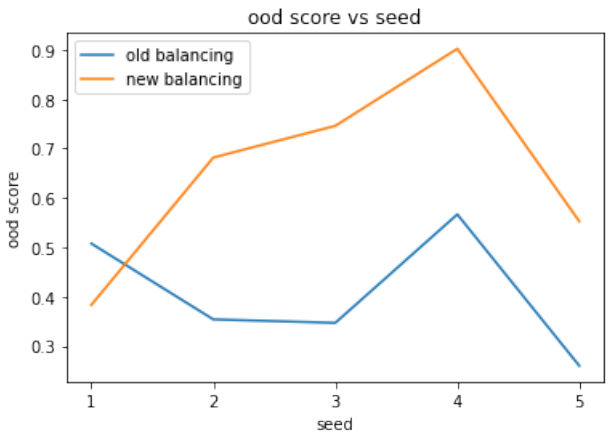


Figure 4.1: Comparison before and after the balancing procedure

The improvement after the new balancing proves that the model was actually struggling in classifying the less represented actions and combinations. Part of the improvement is also due to the fact that the number of classes decreased. In figure 4.1, we can see the comparison in out-of-distribution accuracy between before and after the cutoff and subsampling. Clearly, all other parameters were kept equal. The values shown in this graph were obtained without Adversarial regularization, but with HSIC regularization with coefficient 10.

4.2 HSIC and Adversarial

We now report the results coming from the introduction of the HSIC regularizer, and Adversarial regularizer with respect to the baseline. In all experiments, we noticed a large variability depending on the random seed.

4.2.1 Out-of-distribution

In figure 4.2 we plot the ood comparison, for each seed and for several different values of the coefficients. We notice that in most cases both regularizers help the model generalize to unseen compositions, sometimes with a large gain in performance. We also note that in the cases (e.g. seeds 4 and 5) in which the baseline is already high, between 80 and 90% of accuracy, the regularizers do not help, and may even hurt the performances.

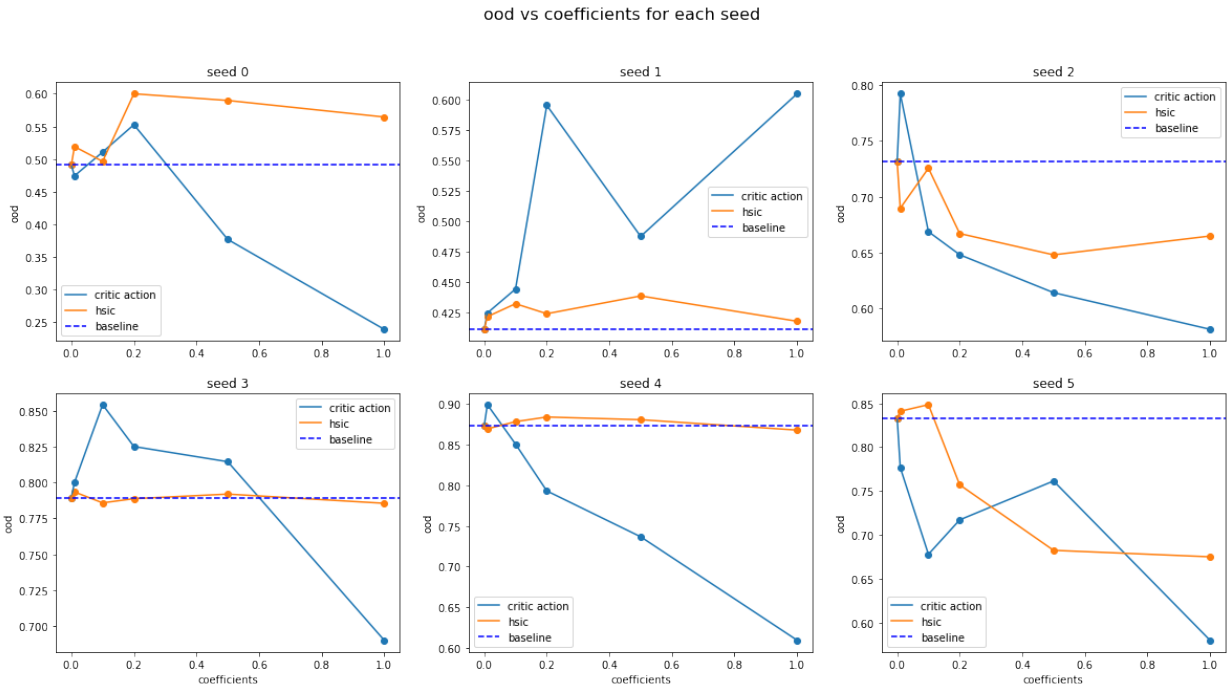


Figure 4.2: OOD comparison baseline, HSIC-regularized and Adversarial-regularized model, with several seeds and coefficients

The ood results are also reported in table 4.1. Since there was no single coefficient performing consistently better, the best coefficient is specified in each entry of the table.

4.2.2 In-distribution

Secondly, we are also interested in the impact that the regularizers have on the in-distribution accuracy. In fact, the goal is to build a model that is able to generalize to unseen compositions, without excessive loss in performance when classifying seen compositions of actions and objects. The following graph in 4.3 reports the in-distribution accuracies in the same fashion as before.

	seed = 0	seed = 1	seed = 2	seed = 3	seed = 4	seed = 5
baseline	49.05%	41.12%	73.12%	78.95%	87.35%	83.33%
Adversarial	55.26% $\lambda_a = 0.2$	60.47% $\lambda_a = 1.0$	79.18% $\lambda_a = 0.01$	85.43% $\lambda_a = 0.1$	89.92% $\lambda_a = 0.01$	77.66% $\lambda_a = 0.01$
HSIC	60.01% $\lambda_H = 0.2$	43.86 % $\lambda_H = 0.5$	72.54% $\lambda_H = 0.1$	79.34% $\lambda_H = 0.01$	88.40% $\lambda_H = 0.2$	84.84% $\lambda_H = 0.1$

Table 4.1: Out-of-distribution accuracies obtained for each seed

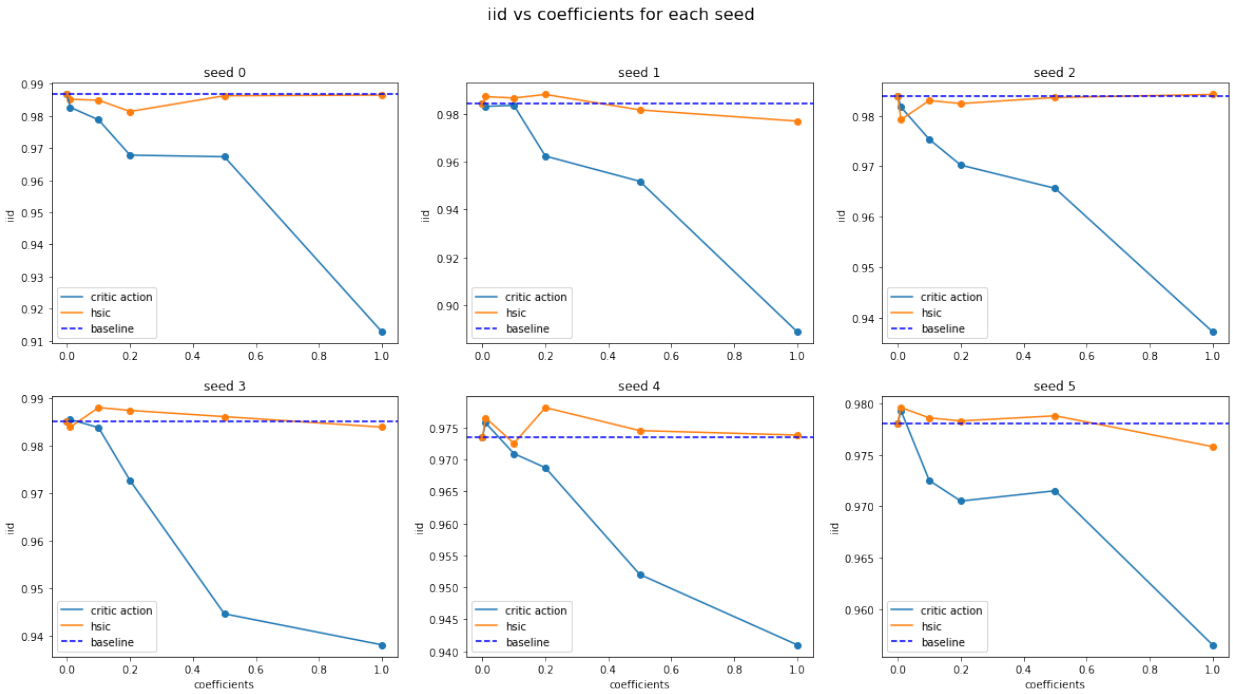


Figure 4.3: IID comparison baseline, HSIC-regularized and Adversarial-regularized model, with several seeds and coefficients

We notice that, in most cases the adversarial regularizer hurts the in-distribution performance more than the HSIC. Such difference may be due to the fact that Adversarial Loss is stronger in isolating the information flow between the spuriously correlated features, or maybe because on average the absolute values of the loss are greater, and so they tend to have a greater impact on the overall loss, other things being equal.

Chapter 5

Conclusion

Causal representation learning is promising for compositional generalization problems, but it also entails several challenges that still need to be addressed. Thanks to this project, we understood that it's crucial to have balanced classes both from the target label distribution and the combinations' distribution. Also, we investigated the Hilbert-Schmidt Independence Criterion, with both its conditional and non-conditional implementations. It revealed to be promising, but still not very stable in its performance gains in out-of-distribution accuracy. Finally, we compared its performances to the Adversarial regularizer, which is delivering significant performance gains as well.

There are several possible further developments that could be pursued. First of all, it would be interesting to further investigate the reason for such variability in the performance of the HSIC regularizer. Part of the investigation could be reproducing the results of [1] and [6] and check the variance of the results they obtain, given different initializations. Then, it could be interesting to try a simultaneous regularization coming from adversarial and hsic regularizers, as it could bring additional improvement, in the case the two measures capture different characteristic of the mutual information flow between the spuriously correlated features.

Finally, the two weaknesses of these measures mentioned in [4] could be addressed. First of all, finding a way to spot the spuriously correlated features without domain knowledge. In most cases indeed, it is hard to say a priori which features are causing the model to underperform in unseen compositions because of confounding. One possible way to do it could be to split the dataset at hand in 3 parts, two for training and one for ood testing. This way, one of the two training set could be used to provide information about the spurious correlations and the other one could be used as normal training set. Secondly, having a way to avoid reliance on the labels for the confounded features could drastically increase the fields of applications of Causal Representation Learning.

Bibliography

[1] Yuval Atzmon, Felix Kreuk, Uri Shalit, and Gal Chechik. “A causal view of compositional zero-shot recognition”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020.

[2] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. “ProcTHOR: Large-Scale Embodied AI Using Procedural Generation”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2022.

[3] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. “Measuring Statistical Dependence with Hilbert-Schmidt Norms”. In: *Proceedings of the 16th international conference on Algorithmic Learning Theory*. 2005.

[4] Yuejiang Liu, Alexandre Alahi, Chris Russell, Max Horn, Dominik Zietlow, Bernhard Schölkopf, and Francesco Locatello. “Causal Triplet: An Open Challenge for Intervention-centric Causal Representation Learning”. In: *Conference on Causal Learning and Reasoning (CLeaR)*. 2023.

[5] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Greg Ver Steeg, and Aram Galstyan. “Invariant representations without adversarial training”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2018.

[6] Frank Ruis, Gertjan Burghouts, and Doina Bucur. “Independent Prototype Propagation for Zero-Shot Compositionality”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2021.

Appendix

5.1 HSIC theoretical notes

Here, we define more in detail what the HSIC is. In order to define the HSIC operator, we need to first define: Reproducing Kernel Hilbert Spaces, the Cross-covariance operator and the Hilbert-Schmidt norm.

Definition 1 *A function $k(\cdot,\cdot)$ is a reproducing kernel of a Hilbert space $\mathcal{H}$ if $\forall f \in \mathcal{H}, f(x) = \langle k(x,\cdot), f(\cdot)\rangle$. If an Hilbert space $\mathcal{H}$ is equipped with a reproducing kernel, whose span is dense in $\mathcal{H}$, then $\mathcal{H}$ is a Kernel Reproducing Hilbert Space (KRHS)*

The cross-covariance operator is basically a generalization of the cross-covariance matrix to infinite dimensional feature spaces.

Definition 2 *Given two KRHS $\mathcal{F}$ and $\mathcal{G}$ with kernels k and l and feature maps ϕ and ψ, the cross-covariance operator is the following:*

$$C_{xy} := \mathbf{E}_{x,y}\left[(\phi(x) - \mu_x) \otimes (\psi(y) - \mu_y)\right] = \mathbf{E}_{x,y}[\phi(x) \otimes \psi(y)] - \mu_x \otimes \mu_y \tag{5.1}$$

The Hilbert-Schimdt norm is a generalization of the Frobenius norm on matrices.

Definition 3 *Let $C : \mathcal{G} \to \mathcal{F}$ be a linear operator, u_i and v_j respectively the orthonormal bases of $\mathcal{F}$ and $\mathcal{G}$. Then the HS norm of C is:*

$$\|C\|^2_{\mathrm{HS}} := \sum_{i,j} \langle Cv_i, u_j\rangle^2_{\mathcal{F}} \tag{5.2}$$

Then, we can define the HSIC as the HS norm of the Cross-Covariance operator.

Definition 4 *Let $\mathcal{F}$ and $\mathcal{G}$ be two KRHS, the Hilbert-Schmidt Independence Criterion is defined as:*

$$\mathrm{HSIC}\left(p_{xy}, \mathcal{F}, \mathcal{G}\right) := \|C_{xy}\|^2_{\mathrm{HS}} \tag{5.3}$$

Finally, since directly computing the HSIC is hard, the authors of the paper propose an empirical estimator for it, and they also prove that its bias is bounded.

Definition 5 *Empirical estimator of HSIC:*

$$\mathrm{HSIC}\left(p_{xy}, \mathcal{F}, \mathcal{G}\right) := \|C_{xy}\|^2_{\mathrm{HS}} \tag{5.4}$$

Finally, the bias of the estimator is the following:

Definition 6 *Bias of the empirical estimator of HSIC, where m is the number of datapoints extracted from p_{xy}:*

$$\mathrm{HSIC}\left(p_{xy}, \mathcal{F}, \mathcal{G}\right) = \mathbf{E}_Z[\mathrm{HSIC}(Z, \mathcal{F}, \mathcal{G})] + O\left(m^{-1}\right) \tag{5.5}$$

5.2 Causality reasoning

This paragraph contains some supplementary material for the causality conclusion that we mentioned in 2.2. By investigating the causality implications of the graph, we can better understand the reasoning of the authors of the paper.
Recalling the definition of d-separation for a path in a causal diagram:

Definition 7 *A path r is d-separated by a set of nodes Z iff one of the following conditions holds:*

- *r contains a chain $V_i \to V_j \to V_k$ or a fork $V_i \leftarrow V_j \to V_k$ such that V_j is in Z*
- *r contains a collider $V_i \to V_j \leftarrow V_k$ such that V_j is not in Z and no descendant of V_j is in Z*

Based on the 7, we can define the d-separation of two nodes:

Definition 8 *Two nodes V_i and V_j are d-separated by a set of nodes Z if all paths between V_i and V_j are d-separated by Z. We write d-separation between two nodes as: $(V_i \perp\!\!\!\perp V_j \mid Z)_G$*

Finally, recalling a theorem that links d-separation to conditional statistical independence:

Theorem 1 *$(V_i \perp\!\!\!\perp V_j \mid Z)_G$ implies $V_i \perp\!\!\!\perp V_j \mid Z$*

Looking now at the causal diagram in 2.1, we investigate the conditional independence between ϕ_a and the object label (and the same reasoning can be applied to ϕ_o and the attribute label). There are two paths connecting the nodes ϕ_a and *Object*: one passes through *Attribute*, and the other one goes through *x*. We can see that if we put *Attribute* in the conditioning set Z, both paths are d-separated:

- In the first case, the path contains a chain such that the middle node is in the conditioning set
- In the second case, the path contains a collider such that the middle node is not in the conditioning set, nor its descendants are (in this case there are no descendants).

5.3 Class balancing

This section is the supplementary material to the section 3.2. We show the impact of the new class balancing procedure on the distribution of action, object combinations and action classes, both in training and in ood set.

5.3.1 Training set

For the training set, the change in combinations distribution was already shown in 3.2. The change in action class distributions is the following.

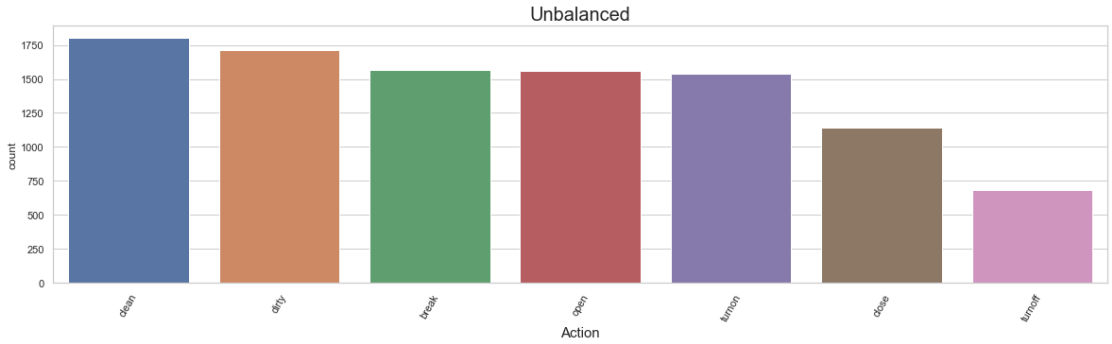


Figure 5.1: Train set actions unbalanced

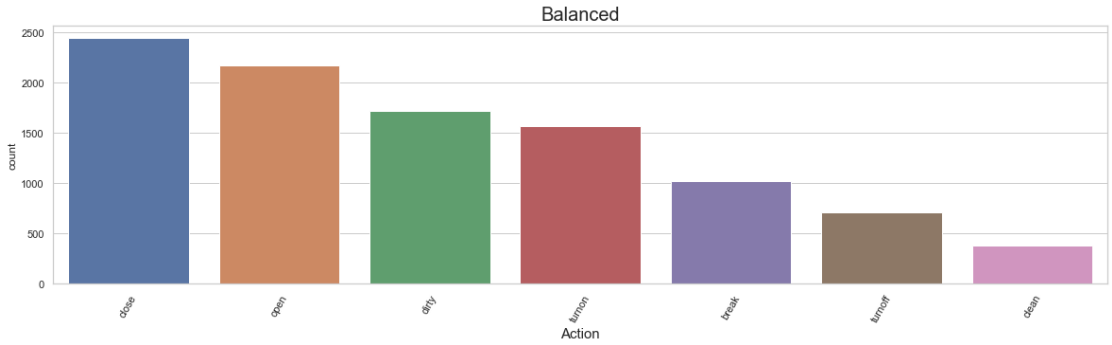


Figure 5.2: Train set actions balanced

5.3.2 Ood set

For the out of distribution set, this is the balancing gain.

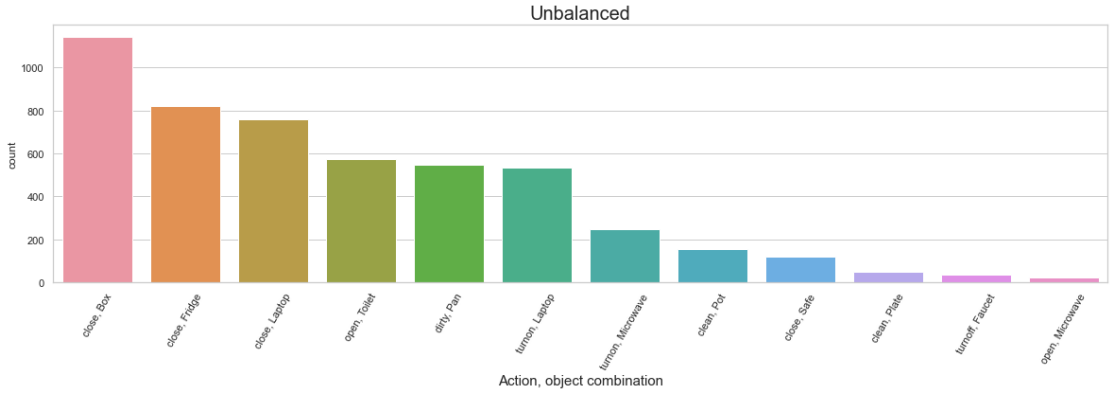


Figure 5.3: Ood set combinations unbalanced

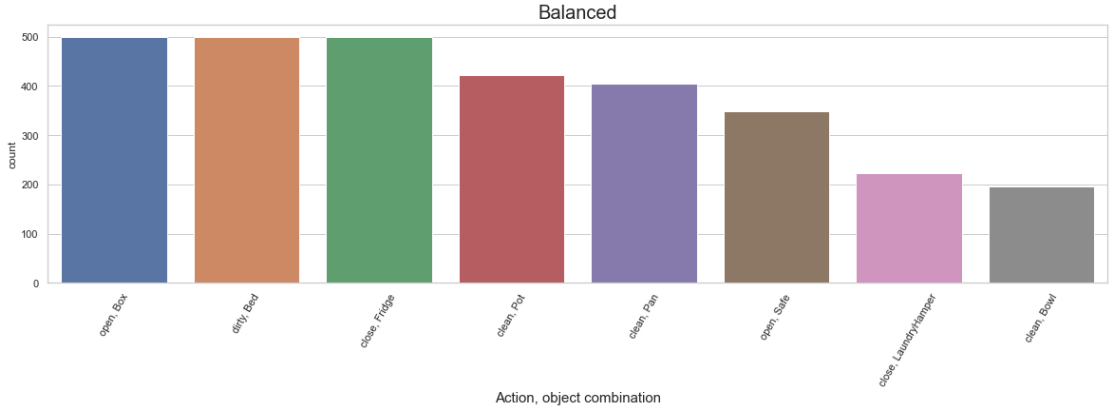


Figure 5.4: Ood set combinations balanced

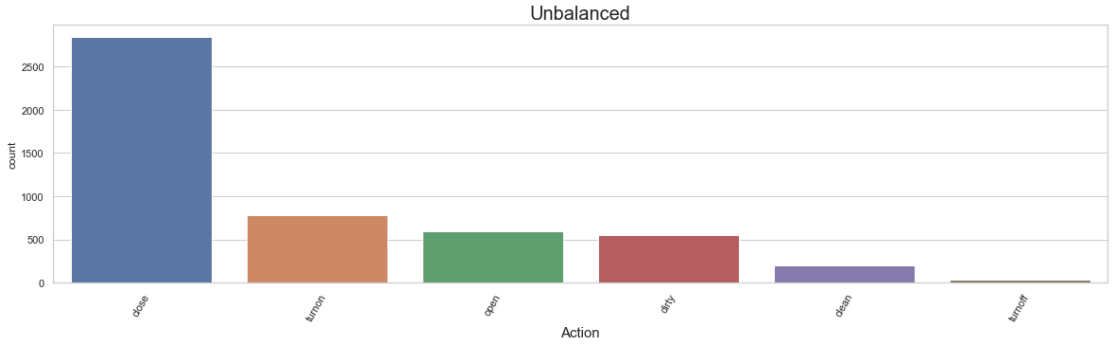


Figure 5.5: Ood set actions unbalanced

Figure 5.6: Ood set actions balanced