Submission and Formatting Instructions for International Conference on Machine Learning (ICML 2025)

Firstname1 Lastname1 [*1] **Firstname2 Lastname2** [*1 2] **Firstname3 Lastname3** [2] **Firstname4 Lastname4** [3]
Firstname5 Lastname5 [1] **Firstname6 Lastname6** [3 1 2] **Firstname7 Lastname7** [2] **Firstname8 Lastname8** [3]
Firstname8 Lastname8 [1 2]

Abstract

This document provides a basic paper template and submission guidelines. Abstracts must be a single paragraph, ideally between 4–6 sentences long. Gross violations will trigger corrections at the camera-ready phase.

1. Related Literature

1.1. Background on Counterfactual Explanations

1.2. Learning Representations

For example, joint-energy models ...

1.3. Generalization and Robustness

Sauer & Geiger (2021) generate counterfactual images for MNIST and ImageNet through independent mechanisms (IM): each IM learns class-conditional input distributions over a specific lower-dimensional, semantically meaningful factor, such as *texture*, *shape* and *background*. The demonstrate that using these generated counterfactuals during classifier training improves model robustness. Similarly, Abbasnejad et al. (2020) argue that counterfactuals represent potentially useful training data in machine learning, especially in supervised settings where inputs may be reasonably mapped to multiple outputs. They, too, demonstrate the augmenting the training data of image classifiers can improve generalization. Teney et al. (2020) propose an approach using counterfactuals in training that does not rely on data augmentation: they argue that counterfactual pairs typically already exist in training datasets. Specifically, their approach relies on, firstly, identifying similar input samples with different annotations and, secondly, ensuring that the gradient of the classifier aligns with the vector between pairs of counterfactual inputs using the cosine distance as a loss function (referred to as *gradient supervision*) (this might be useful for our task as well). In the natural language processing (NLP) domain, counterfactuals have similarly been used to improve models through data augmentation: Wu et al. (2021), propose POLYJUICE, a general-purpose counterfactual generator for language models. They demonstrate empirically that augmenting training data through POLYJUICE counterfactuals improves robustness in a number of NLP tasks.

1.4. Link to Adversarial Training

Freiesleben (2022) propose two definitional differences between Adversarial Examples (AE) and Counterfactual Explanations (CE): firstly, and more importantly according to the authors, the term AE implies missclassification, which is not the case for CE (this might be a useful notion for use to distinguish between adversarials and explanations during training); secondly, they argue that closeness plays a more critical role in the context of CE but confess that even counterfactuals that are not close might be relevant explanations. Pawelczyk et al. (2022) show that CE and AE are equivalent under certain conditions and derive upper bounds on the distances between them.

1.5. Closely Related

Guo et al. (2023) are the first to propose end-to-end training pipeline that includes counterfactual explanations as part of the training prodeduce. In particular, they propose a specific network architecture that includes a predictor and CE generator network (akin a GAN?), where the parameters of the CE generator network are learnable. Counterfactuals are generated during each training iteration and fed back to the predictor network (here we are aligned). In contrast, we impose no restrictions on the neural network architecture at all. NB: to ensure the one-hot encoding of categorical features is maintained, they simple use softmax (might be interesting for CE.jl). Interestingly, the authors find that

[*] Equal contribution [1] Department of XXX, University of YYY, Location, Country [2] Company Name, Location, Country [3] School of ZZZ, Institute of WWW, Location, Country. Correspondence to: Firstname1 Lastname1 <first1.last1@xxx.edu>, Firstname2 Lastname2 <first2.last2@www.uk>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

their approach is sensitive to the choice of the loss function: only MSE seems to lead to good performance. They also demonstrate theoretically, that the objective function is difficult to optimize due to divergent gradients (because partial gradients with respect to the classification loss component and the counterfactual validity component point in opposite directions) and suffers from poor adversarial robustness. To mitigate these issues, the authors use block-wise gradient descent: they first update with respect to classification loss and then use a second update with respect to the other loss components (this might be useful for our task as well).

Ross et al. (2024) propose a way to train models that are guaranteed to provide recourse for individuals with high probability. The approach builds on adversarial training (here we are aligned), where in this context adversarial examples are actively encouraged to exist, but only target attacks with respect to the positive class. The proposed method allows for imposing a set of actionable recourse ex-ante: for example, users can impose mutability constraints for features (here we are aligned). NB: To solve their objective function more efficiently, they use a first-order Taylor approximation to approximate the recourse loss component (might be applicable in our case).

Luu & Inoue (2023) introduce Counterfactual Adversarial Training (CAT) with intention of improving generalization and robustness of language models. Specifically, they propose to proceed as follows: firstly, identify training samples that are subject to high predictive uncertainty (entropy); secondly, generate counterfactual explanations for those samples; and, finally, finetune the model on the augmented dataset that includes the generated counterfactuals.

References

Abbasnejad, E., Teney, D., Parvaneh, A., Shi, J., and van den Hengel, A. Counterfactual vision and language learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10041–10051, 2020. doi: 10.1109/CVPR42600.2020.01006.

Freiesleben, T. The intriguing relation between counterfactual explanations and adversarial examples. *Minds and Machines*, 32(1):77–109, 2022.

Guo, H., Nguyen, T. H., and Yadav, A. Counternet: End-to-end training of prediction aware counterfactual explanations. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, pp. 577–589, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701030. doi: 10.1145/3580305.3599290. URL https://doi.org/10.1145/3580305.3599290.

Luu, H. L. and Inoue, N. Counterfactual adversarial training for improving robustness of pre-trained language models. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pp. 881–888, 2023.

Pawelczyk, M., Agarwal, C., Joshi, S., Upadhyay, S., and Lakkaraju, H. Exploring counterfactual explanations through the lens of adversarial examples: A theoretical and empirical analysis. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 4574–4594. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/pawelczyk22a.html.

Ross, A., Lakkaraju, H., and Bastani, O. Learning models for actionable recourse. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9781713845393.

Sauer, A. and Geiger, A. Counterfactual generative networks, 2021. URL https://arxiv.org/abs/2101.06046.

Teney, D., Abbasnejad, E., and van den Hengel, A. Learning what makes a difference from counterfactual examples and gradient supervision. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X*, pp. 580–599, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-58606-5. doi: 10.1007/978-3-030-58607-2_34. URL https://doi.org/10.1007/978-3-030-58607-2_34.

Wu, T., Ribeiro, M. T., Heer, J., and Weld, D. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6707–6723, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.523. URL https://aclanthology.org/2021.acl-long.523.

A. You *can* have an appendix here.

You can have as much text here as you want. The main body must be at most 8 pages long. For the final version, one more page can be added. If you want, you can use an appendix like this one.

The `\onecolumn` command above can be kept in place if you prefer a one-column appendix, or can be removed if you prefer a two-column appendix. Apart from this possible change, the style (font size, spacing, margins, page numbering, etc.) should be kept the same as the main body.

run	**objective**	**lambda_energy_exper**	**lambda_energy_eval**	**generator_type**	**mean**	**std**
`Int64`	`String7`	`Float64`	`Float64`	`String7`	`Float64`	`Float64`
1	full	0.01	0.1	ecco	-1.18619	0.268524
1	full	0.01	0.1	generic	-1.27949	0.35677
1	full	0.01	0.1	omni	-1.3237	0.188125
1	full	0.01	0.1	revise	-17.8176	4.62883
1	full	0.01	0.5	ecco	-1.72938	0.606102