Relational Out-of-Distribution Generalization

Anonymous Author(s)
Affiliation
Address
email

Abstract

In out-of-distribution (OOD) generalization, domain relation is an important factor. It can provide a global view on the functionality among domains, *e.g.*, the protein domain in the binding affinity task or the geographical location domain in the weather forecast task. Existing work lacks the utilization of the domain relation; yet in this work, we want to explore how to incorporate such rich information into solving the distribution shift problem. Therefore, we propose READ, a general multi-head deep learning framework that harnesses domain relation to generalize to unseen domains in a structured learning and inference manner. In READ, each training domain shares a common backbone but learns one separate head. Built on a proposed explicit regularization, READ simulates the generalization process among heads, where a weighted ensemble prediction from heads irrelevant to input domain is calculated via domain relation and aligned with the target. To improve the reliability of domain relation, READ further leverages similarity metric learning to update initial relation. Empirically, we evaluate READ on three domain generalization benchmarks. The results indicate that READ consistently improves upon existing state-of-the-art methods on datasets from various fields.

1 Introduction

Distribution shift is a universal problem in the real-world scenarios [10, 14], where the test distribution is different from the training distribution. Yet, recent evidence suggests that deep neural networks can be sensitive to distribution shifts, exhibiting a dramatic performance degradation within new environments [4, 21, 26]. Thus, distribution shift is a challenging but rewarding task.

In this paper, we refer to this problem as the out-of-distribution (OOD) generalization and specifically focus on domain shifts. In domain shifts, the test data is from unseen domains, and a well-trained model should be able to possess the good generalization ability to test domains without seeing the data from those domains at training time. For example, in AI-aided Drug Discovery (AIDD), we train a model on data from a fixed set of known target proteins, which is treated as domains. Then we deploy the model to new targets with unseen data distribution. Recent work [12] has proven that existing OOD algorithms fail to generalize in this specific setting.

To improve model robustness under domain shifts, recent studies align training domains and learn domain-invariant representations or predictors [3, 18, 27]. Unfortunately, most invariant learning approaches do not exhibit substantial improvements compared to standard ERM [30] training on various real-world datasets [12, 37]. One potential reason in such failure cases is that some test domains only correlate with a few training domains. For every test domain, involving uncorrelated training domains to train a model is useless or even hurts the performance. To generalize on such data with correlations between domains, we formulate a novel problem called relational OOD, and introduce another promising direction in utilizing the domain relation to solve this task.

Thus, we propose **READ**, a relation-aware algorithm to harness domain relation in a structured learning and inference manner and improve out-of-distribution robustness. Specifically, we first

Submitted to Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

extract pairwise domain relations from the data source. Then, READ aims to optimize the objective function, which is composed of two parts: (1) the supervised loss, an empirical loss for each input and target pair; (2) the domain alignment regularization, an attention loss weighted by a score function based on both the ground-truth and learned domain relation. READ adds the second loss to mimic the cases where the tasks are unknown and out-of-distribution. Lastly during the test time, when given the new test domain, READ learns the relation between the new domain and all training domains, so as to weigh the objective function.

To sum up, our **main contributions** are: we investigate and formalize an important yet underexplored problem - OOD generalization with domain relation, and propose a effective multi-head deep learning framework called READ, which leverages domain relation for ensemble and alignment over domains. We empirically demonstrate the effectiveness of READ under domain shifts. By utilizing the domain relation, we observe that READ outperforms prior state-of-the-art invariant learning methods.

2 Relational Out-of-Distribution Generalization

2.1 Problem Formulation

In this section, we present our formulation of relational OOD problem. We focus on the domain shift setting, where the overall data distribution is drawn from a set of domains $\mathcal{D}$. Each domain $d \in \mathcal{D}$ corresponds with a dataset $(x_i, y_i, d)_{i=1}^{N_d}$ sampled from the domain-specific distribution p_d, where $x_i \in \mathcal{X}$ is the input feature and $y_i \in \mathcal{Y}$ is the prediction target. The relationship between domains is described by a domain graph with the weighted adjacency matrix $A = [A_{ij}]$, where i, j index nodes (domains) in the graph and A_{ij} holds the weight between i and j. The detailed data composition of relational OOD is shown in Appendix B. We split all domains into training domains $\mathcal{D}^{tr}$ and test domains $\mathcal{D}^{ts}$, where $\mathcal{D}^{tr} \cup \mathcal{D}^{ts} = \mathcal{D}$ and $\mathcal{D}^{tr} \cap \mathcal{D}^{ts} = \emptyset$. Our goal is to learn a robust and generalizable predictive model $f : \mathcal{X} \times \mathcal{D} \rightarrow \mathcal{Y}$ using data from the training domains $\mathcal{D}^{tr}$ and the given domain relation graph A to achieve a minimum prediction error on test domains $\mathcal{D}^{ts}$:

$$\min_{\theta} \mathbb{E}_{(x,y,d)\sim P^{ts}} [\ell(f_\theta(x, d, A), y)]. \tag{1}$$

2.2 Overall Pipeline

To better leverage the rich knowledge in domain relation, we would like a method that can build strong correlation among domains. To accomplish this, we introduce READ to ensemble and align over domains. The key idea motivating READ is to explicitly learn a collection of diverse functions that are consistent with training domain knowledge and link them with test domains via domain relation. As outlined in Figure 1, READ adopts an encoder-decoder architecture to extract features. Then, we replace original single-head predictor with a multi-head one, where each domain is assigned with a separate head. To better utilize the domain relation, we introduce a domain-relational learner, which is jointly trained with our prediction model. In practice, READ generates diverse results for each head, insert domain relations across heads, and leverage such relations for prediction.

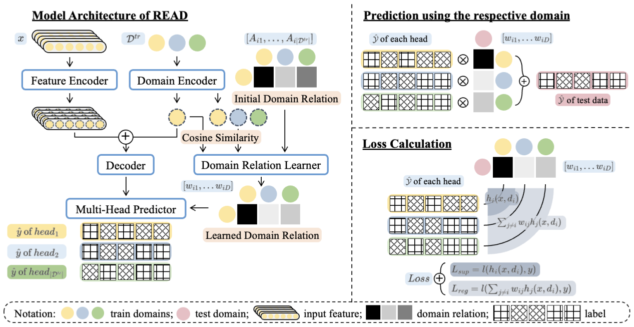

Model Architecture of READ
x
$\mathcal{D}^{tr}$
$[A_{i1}, \ldots, A_{i|\mathcal{D}^{tr}|}]$
Feature Encoder
Domain Encoder
Initial Domain Relation
Cosine Similarity
Decoder
Domain Relation Learner
$[w_{i1}, \ldots w_{iD}]$
Multi-Head Predictor
Learned Domain Relation
$\hat{y}$ of $head_1$
$\hat{y}$ of $head_2$
$\hat{y}$ of $head_{|\mathcal{D}^{tr}|}$
Prediction using the respective domain
$\hat{y}$ of each head
$[w_{i1}, \ldots w_{iD}]$
$\hat{y}$ of test data
Loss Calculation
$[w_{i1}, \ldots w_{iD}]$
$\hat{y}$ of each head
$h_i(x, d_i)$
$\sum_{j \neq i} w_{ij} h_j(x, d_i)$
Loss
$L_{sup} = l(h_i(x, d_i), y)$
$L_{reg} = l(\sum_{j \neq i} w_{ij} h_j(x, d_i), y)$
Notation: train domains; test domain; input feature; domain relation; label

Figure 1: An illustration of READ. **Left:** Model architecture of READ, where x is data from domain d_i. **Right:** The behavior of READ during training and inference.

2.3 Relational Ensemble and Alignment among Domain-Specific Heads

As described above, we build a specific head h_i for each training domain (i.e., $d \in \mathcal{D}^{tr}$). Here we denote the number of heads as $n = |\mathcal{D}^{tr}|$. To support our relational ensemble, two essential components are the predictions from each head and learned domain relation. Therefore, we first extract the features and estimate the outcomes with the encoder-decoder model f and prediction heads $h_i (i \in [n])$. Simultaneously, we produce weight w_{ij} for each domain index pair using domain relation learner. Next, we can ensemble and align over domains with these two components.

Model Learning. We train the model parameters θ by optimizing the regular supervised loss and additional regularization loss. For training data x from domain d_i, we first calculate the supervised loss using the i-th head. Then, we ensemble results from other heads via learned relation and mimic the generalization process with a domain alignment regularization forcing it to be consistent with target.

Therefore, for a data point (x, y, d_i) and initial relation A, we have:

$$Loss(x, d_i, y, A) = \mathcal{L}_{sup} + \lambda \cdot \mathcal{L}_{reg} = \ell(h_i(x, d_i, A), y) + \lambda \cdot \ell(\frac{\sum_{j \neq i} w_{ij} h_j(x, d_j, A)}{\sum_{j \neq i} w_{ij}}, y), \tag{2}$$

where λ is the regularization weight and ℓ is the loss function dedicated to a downstream task.

Model Inference. To infer outcomes on unseen test domain d_j, we would need to produce weighted ensemble among all heads. Given the prediction from head h_i and learned relation w_{ij} for each domain index pair (i, j), it is straightforward to ensemble as follows:

$$\hat{y} = \frac{\sum_{i=1}^{n} w_{ij} h_i(f(x, d_j, A))}{\sum_{i=1}^{n} w_{ij}} \tag{3}$$

where x is a data point on d_j, A is the initial domain relation, and f is the encoder-decoder model. Furthermore, since the learned domain relation is fixed in test, we only parallel calculate relations for all test domains once, indicating our domain relation learner brings little overhead during inference.

2.4 Similarity-based Domain Relation Learning

Following [17], we cast domain relation learning into a similarity metric function. As it takes domain representations (denoted as Z) and weighted adjacency matrix A as input and outputs a new relation denoted by A', we involve a two-step process. The first step follows the multi-head weighted cosine similarity learning method previously used by [7], while the second step incorporates the information of the original relation input.

We denote $\{\boldsymbol{w}_i\}_{i=1}^{m}$ as m independent learnable weight vectors. Without loss of generality, we compute the estimation for relation between domain index pair (u, v) as

$$a_{uv} = \frac{1}{m} \sum_{i=1}^{m} cos(\boldsymbol{w}_i \odot \boldsymbol{z}_u, \boldsymbol{w}_i \odot \boldsymbol{z}_v), \tag{4}$$

where z_u and z_v are respectively the representations of domain d_u and d_v.

Next, to inherit the information from original domain relation, we combine the learned relation A_{tmp} and initial relation A with smoothing parameter α to get output A':

$$A' = \alpha \times A + (1 - \alpha) \times A_{tmp}. \tag{5}$$

We use A' to replace A in Eqn. (2) and (3).

3 Experiment

In this section, we conduct comprehensive experiments to evaluate the effectiveness of READ. More experimental analysis can be find in Appendix G.

3.1 Datasets and Baselines

Following [19, 35] we evaluate READ on three benchmarks: (1) *DG-15*: 2D synthetic dataset, (2) *TPT-48*: weather prediction dataset, (3) *ChEMBL-STRING* (ChEMBL 50 and ChEMBL 100): drug discovery dataset. We present detailed descriptions of datasets in Appendix C.

Our main baselines are general-purpose methods with different learning strategies and categories including (1) *vanilla*: ERM [30], (2) *distributionally robust optimization*: GroupDRO [24], (3) *data augmentation*: Mixup [34], (4) *domain-invariant feature learning*: IRM [3], DANN [9], CORAL [27] (See Appendix D for more detail). For fair comparison, we adopt the same model architectures and same input x, y, d to the model (Eqn. (1)) for all approaches. All hyperparameters are selected via cross-validation. We list all hyperparameters in Appendix in E.

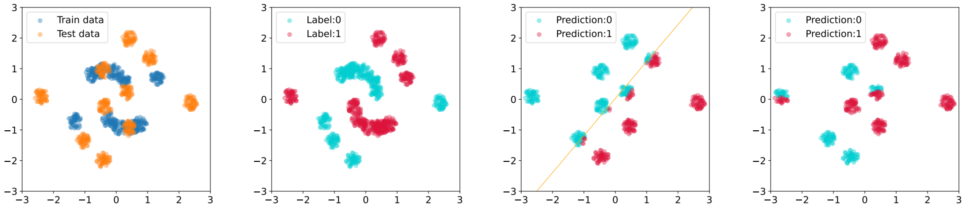
3.2 Experiment Results

DG-15. The performance of READ and prior methods on DG-15 is reported in Table 1. READ has an unparalleled advantage over all baselines with an outperformance of more than 30%. Moreover, previous methods perform even worse than random guess (50% accuracy), indicating the necessity of incorporating domain relation to transfer information among domains with high correlation.

Table 1: Results of domain shift on DG-15. Averaged accuracy is reported. See full table with standard deviation in Appendix F.1. We **bold** the best results and <u>underline</u> the second best results.

Algorithm	ERM	GroupDRO	Mixup	IRM	DANN	CORAL	**READ** (ours)
Accuracy	44.0%	<u>47.1%</u>	41.3%	43.9%	43.1%	43.5%	**77.5%**

For further understanding on how READ works on DG-15, Figure 2 visualizes data distribution on DG-15 and the corresponding predictions from GroupDRO and READ. Train/test split is depicted in Figure 2a. The comparison between Figure 2b with Figure 2c shows that GroupDRO learns a linear decision boundary that overfits the training domains under domain shift. In contrast, READ successfully generalize to test domains in Figure 2d except one without nearby training domain.



(a) : Data Split (b) : Label Distribution (c) : GroupDRO (d) : READ

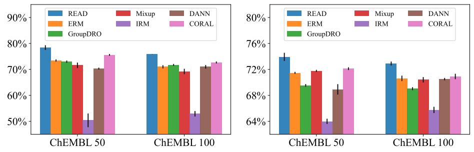
Figure 2: Visualization on DG-15. (a): Train/test split; (b): Label distribution; (c) Predictions from GroupDRO (our best baseline); (d) Predictions from READ.

TPT-48. Table 2 presents the MSE of our algorithm and previous methods on TPT-48. The results shows that Mixup, IRM and DANN achieve negative performance, highlighting the difficulty of tackling domain shift among geographic-related states. In comparison, READ achieves the lowest MSE, indicating its suitability for regression tasks. We present the full results in Appendix F.2.

Table 2: Results of domain shift on TPT-48. We report the average MSE here.

Algorithm	ERM	GroupDRO	Mixup	IRM	DANN	CORAL	**READ** (ours)
MSE	0.108	<u>0.096</u>	0.179	0.135	0.122	0.103	**0.091**

ChEMBL-STRING. Figure 3 shows the results of various methods on ChEMBL-STRING. READ outperforms previous methods in both ROC-AUC score and average accuracy, illustrating the effectiveness of READ. Therefore, READ provides a fast way to transfer knowledge to new proteins in drug discovery. We also observe that in this task, all previous OOD algorithms except CORAL exhibit no clear improvement over the simple ERM algorithm.



(a) : ROC-AUC (b) : Accuracy

Figure 3: Results of domain shifts on ChEMBL 50 and ChEMBL 100. **Left:** ROC-AUC score; **Right:** Average accuracy.

4 Conclusion

In this paper, we investigate relational OOD, a natural extension of classical domain shift problem. We propose an effective and efficient algorithm called READ to tackle this problem. READ aims to leverage ensemble and alignment domain-specific heads via domain relation. We evaluate the effectiveness of READ on three domain shift benchmarks from different fields, demonstrating its promise. Besides, detailed analyses verify that the performance gains caused by READ result from our proposed domain alignment regularization and relation learner.

References

[1] Climate change in the contiguous united states. https://github.com/washingtonpost/data-2C-beyond-the-limit-usa, 2020.

[2] Kartik Ahuja, Ethan Caballero, Dinghuai Zhang, Yoshua Bengio, Ioannis Mitliagkas, and Irina Rish. Invariance principle meets information bottleneck for out-of-distribution generalization. In *NeurIPS*, 2021.

[3] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[4] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *ECCV*, 2018.

[5] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando C Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2009.

[6] Shiyu Chang, Yang Zhang, Mo Yu, and T. Jaakkola. Invariant rationalization. *ArXiv*, abs/2003.09772, 2020.

[7] Yu Chen, Lingfei Wu, and Mohammed J. Zaki. Iterative deep graph learning for graph neural networks: Better and robust node embeddings. *ArXiv*, abs/2006.13009, 2020.

[8] Ching-Yao Chuang and Youssef Mroueh. Fair mixup: Fairness via interpolation. *ArXiv*, abs/2103.06503, 2021.

[9] Yaroslav Ganin, E. Ustinova, Hana Ajakan, Pascal Germain, H. Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. In *J. Mach. Learn. Res.*, 2016.

[10] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *ArXiv*, abs/2007.01434, 2021.

[11] Ruocheng Guo, Pengchuan Zhang, Hao Liu, and Emre Kıcıman. Out-of-distribution prediction with invariant risk minimization: The limitation and an effective fix. *ArXiv*, abs/2101.07732, 2021.

[12] Yuanfeng Ji, Lu Zhang, Jiaxiang Wu, Bing Wu, Long-Kai Huang, Tingyang Xu, Yu Rong, Lanqing Li, Jie Ren, Ding Xue, Houtim Lai, Shaoyong Xu, Jing Feng, Wei Liu, Ping Luo, Shuigeng Zhou, Junzhou Huang, Peilin Zhao, and Yatao Bian. Drugood: Out-of-distribution (ood) dataset curator and benchmark for ai-aided drug discovery - a focus on affinity prediction problems with noise annotations. *ArXiv*, abs/2201.09637, 2022.

[13] Kia Khezeli, Arno Blaas, Frank Soboczenski, Nicholas K. K. Chia, and John Kalantari. On invariance penalties for risk minimization. *ArXiv*, abs/2106.09777, 2021.

[14] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard L. Phillips, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts. In *ICML*, 2021.

[15] Masanori Koyama and Shoichiro Yamaguchi. Out-of-distribution generalization with maximal invariant predictor. *ArXiv*, abs/2008.01883, 2020.

[16] David Krueger, Ethan Caballero, Jörn-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Rémi Le Priol, and Aaron C. Courville. Out-of-distribution generalization via risk extrapolation (rex). In *ICML*, 2021.

[17] Danning Lao, Xinyu Yang, Qitian Wu, and Junchi Yan. Variational inference for training graph neural networks in low-data regime through joint structure-label estimation. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022.

[18] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex Chichung Kot. Domain generalization with adversarial feature learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018.

[19] Shengchao Liu, Meng Qu, Zuobai Zhang, Huiyu Cai, and Jian Tang. Structured multi-task learning for molecular property prediction. In *International Conference on Artificial Intelligence and Statistics*, pages 8906–8920. PMLR, 2022.

[20] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. *ArXiv*, abs/1502.02791, 2015.

[21] Massimiliano Mancini, Zeynep Akata, Elisa Ricci, and Barbara Caputo. Towards recognizing unseen categories in unseen domains. *ArXiv*, abs/2007.12256, 2020.

[22] David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, María Gordillo-Marañón, Fiona Hunter, Laura Junco, Grace Mugumbate, Milagros Rodriguez-Lopez, Francis Atkinson, Nicolas Bosc, Chris J Radoux, Aldo Segura-Cabrera, Anne Hersey, and Andrew R Leach. ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Research*, 47(D1):D930–D940, 11 2018.

[23] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12553–12562, 2020.

[24] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *ICLR*, 2020.

[25] Yang Shu, Zhangjie Cao, Chenyu Wang, Jianmin Wang, and Mingsheng Long. Open domain generalization with domain-augmented meta-learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9619–9628, 2021.

[26] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23:828–841, 2019.

[27] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.

[28] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1):D607–D613, 2019.

[29] Eric Tzeng, Judy Hoffman, N. Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *ArXiv*, abs/1412.3474, 2014.

[30] Vladimir Naumovich Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10 5:988–99, 1999.

[31] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C. Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, 2018.

[32] Russell S Vose, Scott Applequist, Mike Squires, Imke Durre, Matthew J Menne, Claude N. Jr. Williams, Chris Fenimore, Karin Gleason, and Derek Arndt. Gridded 5km ghcn-daily temperature and precipitation dataset (nclimgrid) version 1. *Maximum Temperature, Minimum Temperature, Average Temperature, and Precipitation*, 2014.

[33] Yufei Wang, Haoliang Li, and Alex Chichung Kot. Heterogeneous domain generalization via domain mixup. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3622–3626, 2020.

[34] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *AAAI*, 2020.

[35] Zihao Xu, Hao He, Guang-He Lee, Yuyang Wang, and Hao Wang. Graph-relational domain adaptation. *arXiv preprint:2202.03628*, 2022.

[36] Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *ArXiv*, abs/2001.00677, 2020.

[37] Huaxiu Yao, Caroline Choi, Yoonho Lee, Pang Wei Koh, and Chelsea Finn. Wild-time: A benchmark of in-the-wild distribution shift over time. In *ICML 2022 Shift Happens Workshop*, 2022.

[38] Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto L. Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2100–2110, 2019.

[39] Long Zhao, Ting Liu, Xi Peng, and Dimitris N. Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. *ArXiv*, abs/2010.08001, 2020.

[40] Fan Zhou, Zhuqing Jiang, Changjian Shui, Boyu Wang, and Brahim Chaib-draa. Domain generalization with optimal transport and metric learning. *ArXiv*, abs/2007.10573, 2020.

[41] Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. *ArXiv*, abs/2003.06054, 2020.

A Related Work

Here, we discuss related approaches that solve the OOD generalization from the following two categories:

Learning Invariant Representations. Inspired by unsupervised domain adaptation [5, 9], the first category of works aligns representations across domains to learn invariant representations. The major research line of this category aims to eliminate the domain dependency by minimizing the divergence of feature distributions with different distance metrics, e.g., maximum mean discrepancy [20, 29], an adversarial loss [9, 18], Wassertein distance [40]. Follow-up works applied data augmentation to (1) generate more domains and enhance the consistency of representations during training [25, 33, 34, 36, 38, 41] or (2) generate new domains in an adversarial way to imitate the challenging domains without using training domain information [23, 31, 39]. Instead of feature distributions, READ focuses on alignment among logits using domain relation, providing a fresh perspective to the field of domain alignment.

Learning Invariant Predictors.

Beyond using domain alignment to learning invariant representations, recent work aims to further enhance the correlations between the invariant representations and the labels [15], leading to invariant predictors. Representatively, motivated by casual inference, invariant risk minimization (IRM) [3] and its variants [2, 11, 13] aim to find a predictor that performs well across all domains through regularizations. Other follow-up works leverage regularizers to penalize the variance of risks across all domains [16], to align the gradient across domains [15], to smooth the cross-domain interpolation paths [8], or to involve game-theoretic invariant rationalization criterion [6].In contrast, READ encourages its prediction heads correlated to domains and ensembles them with domain relation to generate a test-domain-related predictor.

B Problem Formulation

In Figure 4, we provide an illustration of our relation OOD. Let $\mathcal{X}$ be the input (feature) space, $\mathcal{Y}$ be the target (label) space and $\mathcal{D}$ be the domain space, the dataset can be decomposed into several samples (x, y, d) from $\mathcal{X} \times \mathcal{Y} \times \mathcal{D}$, and a weighted adjacency matrix A representing the domain relation graph.

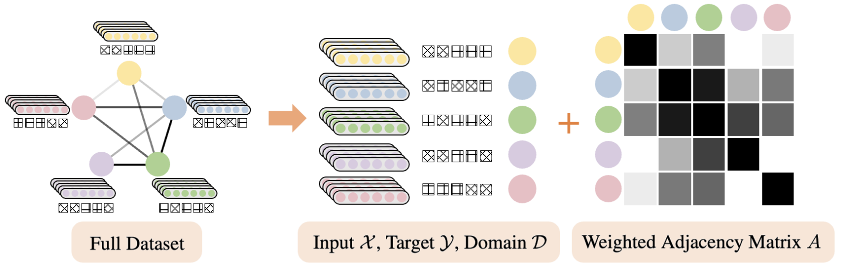

Full Dataset
Input $\mathcal{X}$, Target $\mathcal{Y}$, Domain $\mathcal{D}$
+
Weighted Adjacency Matrix A

Figure 4: An illustration of our problem formulation. Each color represents one domain. The transparency of edges indicates domain relation, with low transparency meaning close relation and vice versa.

C Detailed Description of Dataset

DG-15. Following [35], We start with a synthetic 2D binary classification dataset with 15 domains called DG-15. In each domain i, we randomly sample one point $p_i = (x_i, y_i)$ in the 2-dimensional space. The domain embedding is the angle of each point (i.e., $d_i = \arctan(\frac{y_i}{x_i})$). Next, 50 positive and 50 negative data points are generated from two different 2-dimensional Gaussian distributions $\mathcal{N}(p_i, \mathbf{I})$ and $\mathcal{N}(-p_i, \mathbf{I})$ respectively. In DG-15, we directly use the included angle of the two half-lines starting from the origin point and passing through p_i, p_j to construct the relation between domain i and j (i.e., $A_{ij} = \arctan(\frac{y_j}{x_j}) - \arctan(\frac{y_i}{x_i})$). The number of training, validation and test domains are 5, 5, 5 respectively.

TPT-48. TPT-48 is a real-world weather prediction dataset from the National Oceanic and Atmospheric Administration's Climate Divisional Database (nClimDiv) and Gridded 5km GHCN-Daily Temperature and Precipitation Dataset (nClimGrid) [32], where the monthly average temperature for the 48 contiguous states in the US from 2008 to 2019 is collected. We process the data following Washington Post [1] and focus on the regression task that forecasts the next 6 months' temperature based on previous first 6 months' temperature. The embedding of domain i is defined as the latitude and longitude of state i-th geographic center, which can be denoted as $d_i = (Lat_i, Lng_i)$. In TPT-48, we use a 0/1 adjacency matrix as the domain relation, and domain

i, j are connected (i.e., $A_{ij} = 1$) if they are adjacent states. We split all 48 states into 24 training, 12 validation and 12 test domains.

ChEMBL-STRING. We also consider a scientific dataset from the chemistry field. ChEMBL [22] is a dataset for the binding affinity task, which records the interaction between a small molecule and a target protein. Namely, each data point is a molecule, and each task is to predict if the molecule can interact with the protein accordingly. We cluster all tasks into several domains using their protein sequences. ChEMBL-STRING [19] was recently proposed. It adds the domain relation (*i.e.*, the protein relation) on ChEMBL by adopting the proteins using protein-protein interaction (PPI) scores from STRING [28]. The intuition is that the PPI is able to provide a large knowledge base that can be expected to connect the unseen proteins with the training proteins, enabling the fast knowledge generalization/transfer. Due to the sparsity of the domain relation in ChEMBL (i.e, the PPI scores of most protein pairs are 0), we densify the relation graph by iteratively filtering out proteins whose number of nonzero PPI is lower than a certain threshold. By setting the threshold value to 50 and 100, we obtain two relatively dense benchmark subsets called ChEMBL 50 and ChEMBL 100. The detailed statistics of ChEMBL 50 and 100 are listed in Table 3.

Table 3: Statistics about ChEMBL 50 and 100 datasets, where we use proteins as domains. Sparsity here is defined as the ratio of zero values in the relation graph.

Dataset	# Samples	# Proteins	Sparsity	# Train Proteins	# Valid Proteins	# Test Proteins
ChEMBL 50	87908	141	0.914	93	19	29
ChEMBL 100	58823	122	0.911	74	24	24

D Detailed Description of Baselines

In this work, we compare READ with several invariant learning approaches, i.e., ERM [30], GroupDRO [24], Mixup [34], IRM [3], DANN [9], CORAL [27]. GroupDRO optimizes the worst-domain loss. Inter-domain Mixup performs ERM on linear interpolations of examples from random pairs of domains and their labels. IRM learns invariant predictors that perform well across different domains. DANN employs an adversarial network to match feature distributions. CORAL matches the mean and covariance of feature distributions.

E Detailed Hyperparameters

Table 4: Hyperparameters for READ on all datasets.

Hyperparameters	DG-15	TRT-48	ChEMBL 50	ChEMBL 100
Learning Rate	1e-5	1e-4	1e-4	1e-4
Weight Decay	5e-4	5e-4	0	0
Batch Size	10	64	30	30
Epochs	30	40	100	100
Warm Start Epochs	5	10	10	10
Regularization Weight λ	0.5	0.5	0.5	0.5
Relation keep Ratio α	0.8	0.8	0.5	0.5

F Additional Experiment Results

F.1 Full Results on DG-15

Table 5: Full results of domain shift on DG-15.

Algorithm	ERM	GroupDRO	Mixup	IRM	DANN	CORAL	**READ** (ours)
Accuracy	44.0 ± 4.6%	47.1±9.0%	41.3±3.9%	43.9±5.1%	43.1±4.5%	43.5±1.5%	**77.5±2.5%**

F.2 Full Results on TPT-48

Table 6: Full results of domain shift on TPT-48.

Algorithm	ERM	GroupDRO	Mixup	IRM	DANN	CORAL	**READ** (ours)
MSE	0.108 ± 0.002	0.096 ± 0.001	0.179 ± 0.012	0.135 ± 0.003	0.122 ± 0.019	0.103 ± 0.004	**0.091** ± **0.003**

F.3 Full Results on ChEMBL-STRING

Table 7: Full results of domain shifts on ChEMBL 50 and 100.

	ChEMBL 50		ChEMBL 100	
	ROC-AUC	Accuracy	ROC-AUC	Accuracy
ERM	73.43 ± 0.40%	71.47 ± 0.10%	70.99 ± 0.61%	70.62 ± 0.42%
GroupDRO	72.99 ± 0.43%	69.54 ± 0.20%	71.75 ± 0.37%	69.03 ± 0.23%
Mixup	71.57 ± 1.06%	71.77 ± 0.17%	69.20 ± 1.04%	70.41 ± 0.41%
IRM	50.39 ± 2.62%	64.01 ± 0.40%	52.91 ± 1.05%	65.76 ± 0.48%
DANN	70.31 ± 0.35%	68.93 ± 0.80%	71.00 ± 0.67%	70.51 ± 0.18%
CORAL	75.60 ± 0.33%	72.11 ± 0.21%	72.69 ± 0.37%	70.91 ± 0.42%
READ (ours)	**77.98** ± **0.30%**	**74.07** ± **0.27%**	**75.56** ± **0.11%**	**73.02** ± **0.16%**

G Analysis

Finally, we do analysis on ChEMBL-STRING to understand the effect of each module in READ.

How do domain relation benefits ensemble inference?

We compare our weighted ensemble using domain relation with a simple uniform ensemble strategy during inference in Table 8. The performance gap shows the effectiveness of our strategy. In addition, we observe that the uniform ensemble strategy also outperforms ERM, indicating the potential of READ in scenarios that only relation between training domains is available.

Table 8: Comparison between ensemble strategies. ROC-AUC is reported.

	ChEMBL 50	ChEMBL 100
Uniform Ensemble	75.29 ± 0.21%	73.33 ± 0.12%
Weighted Ensemble	**78.42** ± **0.90%**	**75.85** ± **0.04%**

Does regularization improve performance?

By finetuning the regularization weight λ from 0 to 2, we analyze the effect of domain alignment regularization and test the model's sensitivity. The corresponding ROC-AUC and accuracy are respectively shown in Figure 5. First, when $\lambda = 0$ (i.e., no regularization), READ's performance is lower than those with a positive λ, confirming the effectiveness of the relational alignment. Moreover, when we vary λ between 0.5 and 2, λ around 0.5 achieves the best results, indicating that too large λ will result in weaker correlation between each domain and its corresponding head.

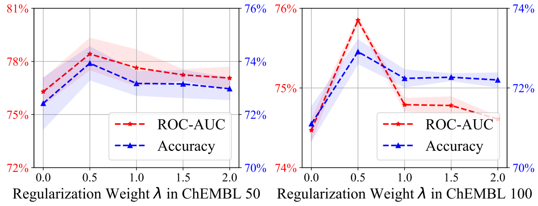


Figure 5: Sensitivity for regularization weight λ. ROC-AUC and Accuracy are reported.

Why we need to learn domain relation?

In Figure 6, we analyze the effect of domain relation learner. By increasing the value of relation keep ratio α from 0 to 1, we enforce our model to rely more on initial relation. As we can see, the single usage of either initial or learned domain relation leads to a decrease in performance, proving the significance of a combined relation. We conjecture that this is because the original relation is relatively accurate while the learned one is more task-related. Thus, combining these two relations guarantees a more comprehensive relation and improves the robustness of READ.

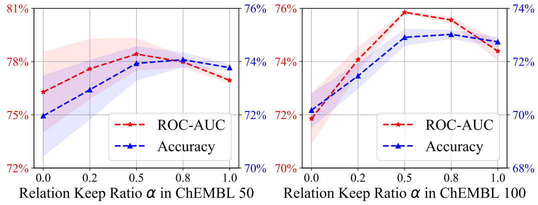


Figure 6: Analysis of relation keep ratio α on ChEMBL-STRING. Here we report ROC-AUC and Accuracy.