ICLR 2024 Spotlight

Improving Domain Generalization with Domain Relations

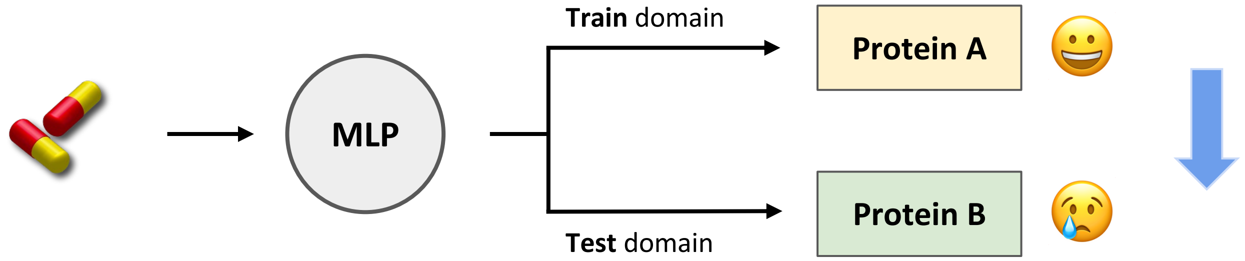
Huaxiu Yao[1,2*], Xinyu Yang[3*], Xinyi Pan[4], Shengchao Liu[5], Pang Wei Koh[6], Chelsea Finn[1]
[1]Stanford University, [2]UNC-Chapel Hill, [3]CMU, [4]UCLA, [5]Caltech, [6]University of Washington

24.03.21
Sangyoon Lee

Motivation

Domain Shift

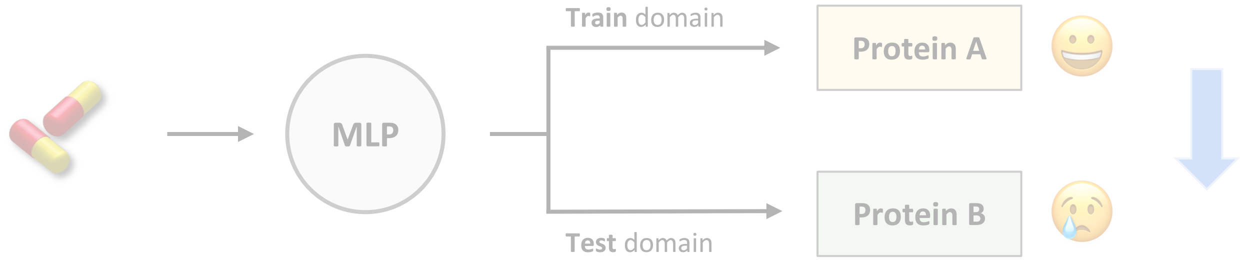
Applying a trained model to **new domains** (differ from training domains)



Motivation

Domain Shift

Applying a trained model to **new domains** (differ from training domains)



Robust model that can be **generalized** to overall domains.

Motivation

Domain Shift

Prior approach

→ **Single model** that is **domain invariant**

This work ...

Construct a **domain-specific model** for a *new domain* seen at test time.

(1) Model may perform better if they were *specialized to a given domain*.
(2) Different domains can exhibit *strong correlations* with non-general features.

Background

Out-of-Distribution Generalization

Problem: predicting the label $\boldsymbol{y} \in \boldsymbol{\mathcal{Y}}$ based on the input features $\boldsymbol{x} \in \boldsymbol{\mathcal{X}}$

$\boldsymbol{P^{tr}}$: train distribution, $\boldsymbol{P^{ts}}$: test distribution

Traditional Objective = $\arg\min_{\theta \in \Theta} \mathbb{E}_{(x,y)\,\sim \boldsymbol{P^{tr}}}[l(f_\theta(x), y)]$

Distribution shift → $\boldsymbol{P^{tr}} \neq \boldsymbol{P^{ts}}$

This work considers a setting where **overall distribution** is drawn from **a set of domains $\boldsymbol{\mathfrak{D}} = \{\mathbf{1}, \dots, \boldsymbol{D}\}$.**

→ Domain ID of training and test datapoints are available!

Background

Domain Relations and Domain Meta-Data

Key Idea: **Domain Relations** → Domain Shift

Domain meta-data $\mathcal{M} = \{m_i\}_{i=1}^{D}$

: depict the distinctive properties of each domain.

Domain relations (Domain Similarity Matrix $\mathcal{A} = \{a_{ij}\}_{i,j=1}^{D}$**)**

: similarity or relatedness between different domains.

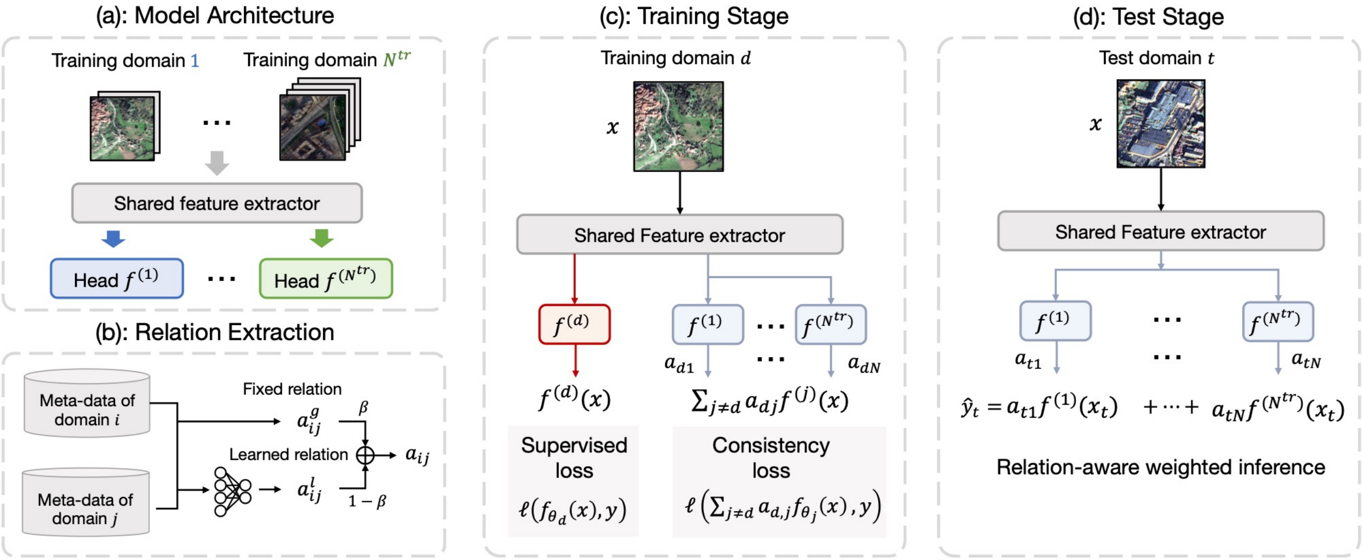
→ strength of the relationship between domains i and j

Method

Leveraging domain distances for out-of-domain generalization

$\boldsymbol{D^3G}$: Leveraging domain distances for out-of-domain generalization

→ improve *out-of-domain generalization* by constructing **domain- specific** models.

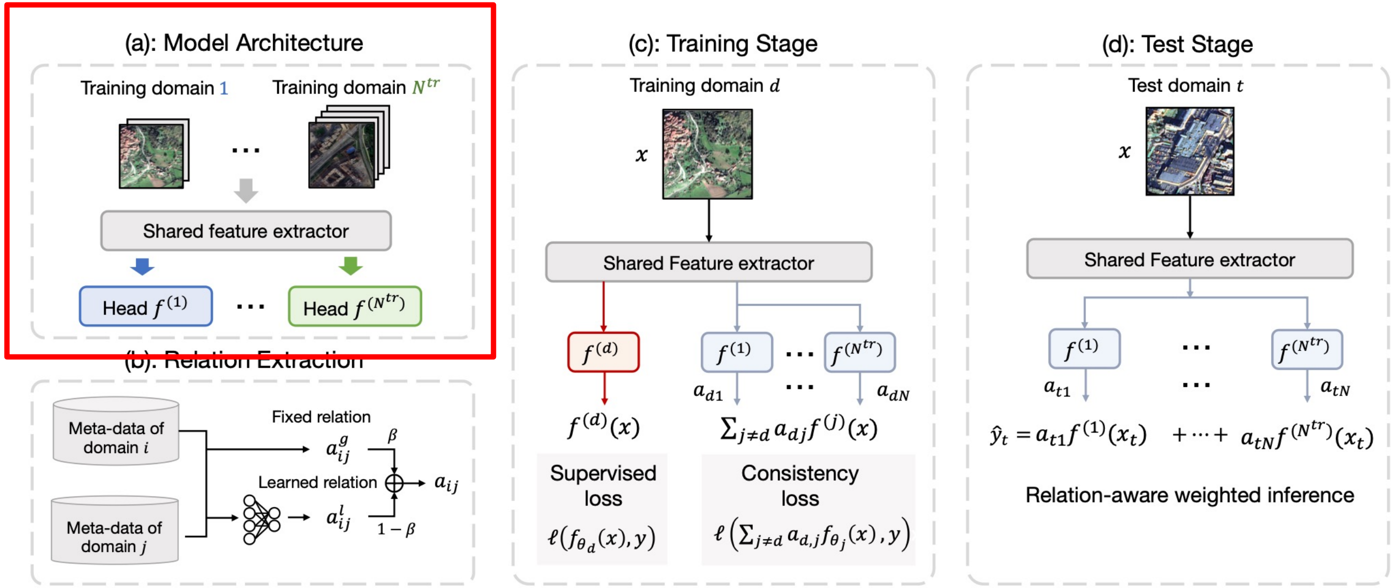


Method

Leveraging domain distances for out-of-domain generalization

$\boldsymbol{D^3G}$: Leveraging domain distances for out-of-domain generalization

→ improve *out-of-domain generalization* by constructing **domain- specific** models.

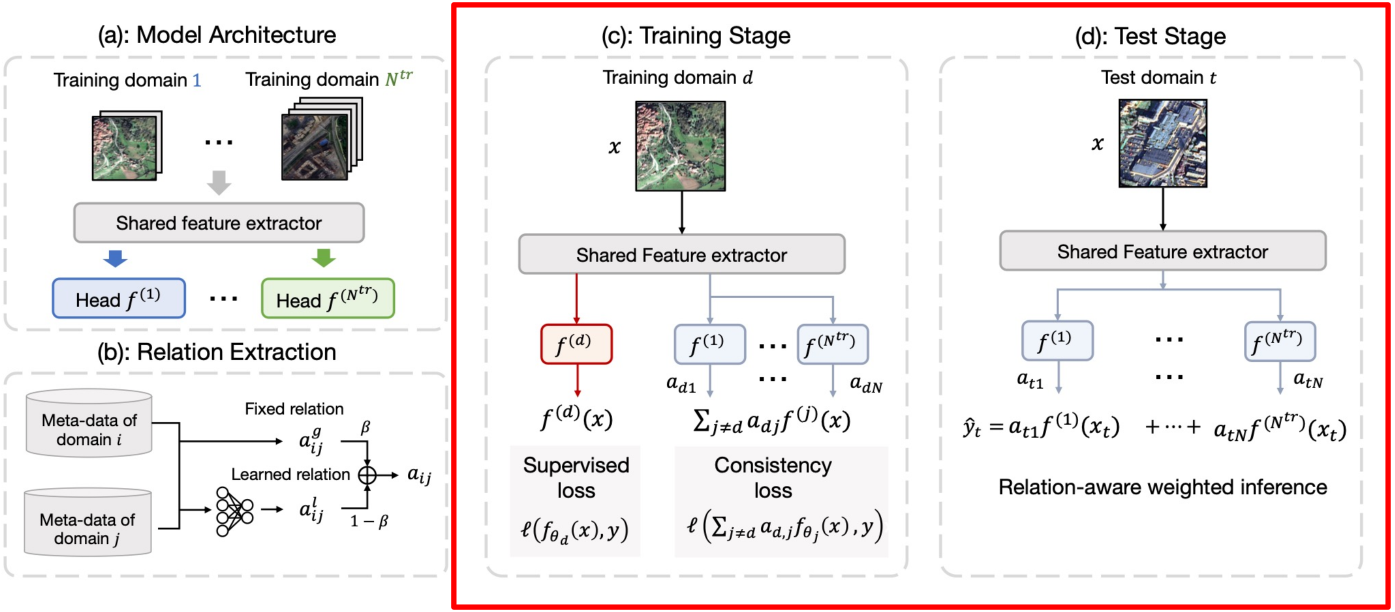


Method

Leveraging domain distances for out-of-domain generalization

D^3G : Leveraging domain distances for out-of-domain generalization

→ improve *out-of-domain generalization* by constructing **domain- specific** models.

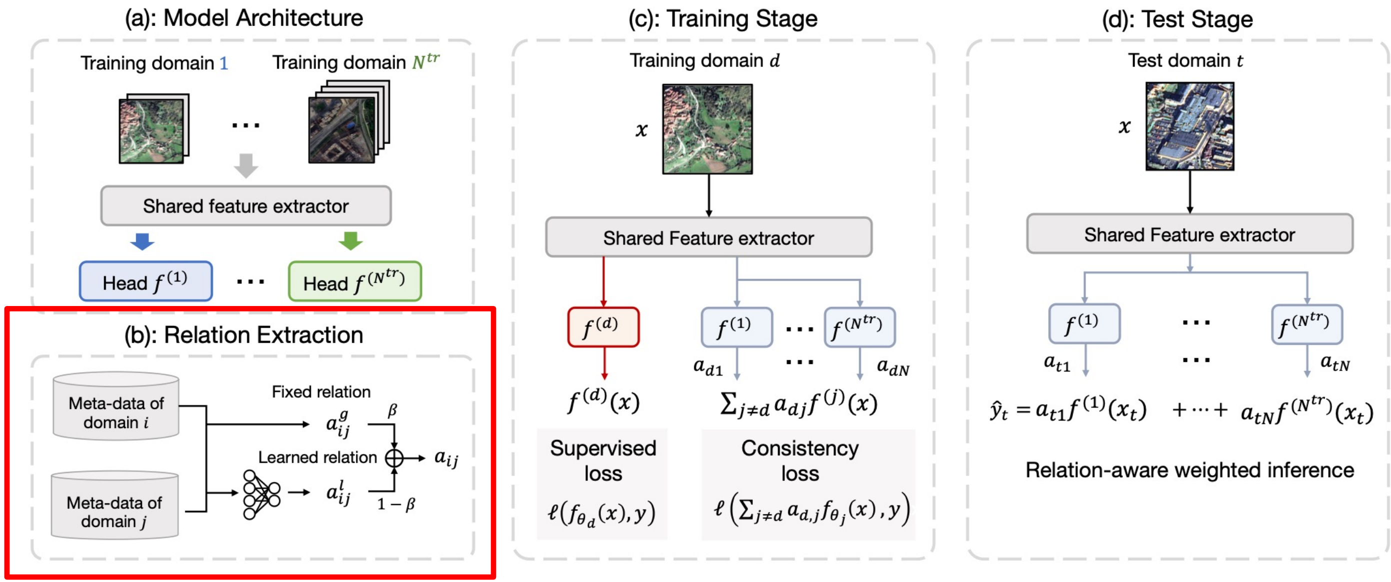


Method

Leveraging domain distances for out-of-domain generalization

D^3G : Leveraging domain distances for out-of-domain generalization

→ improve *out-of-domain generalization* by constructing **domain- specific** models.



Method

Building Domain-Specific Models

Training Stage.

Multi-headed neural network comprising N^{tr} heads. (N^{tr} : Number of training domains)

Input datapoint (x, y) from domain d → prediction made by d_{th} head : $\boldsymbol{f^{(d)}(x) = h^{(d)}(e(x))}$

Minimizing the **predictivie risk** :

$$\boldsymbol{L_{pred}} = \mathbb{E}_{d\in\mathcal{D}^{tr}}\mathbb{E}_{(x,y)\sim P_d}[l\left(f^{(d)}(x), y\right)].$$

Method

Building Domain-Specific Models

Training Stage. (with limited data)

Difficulties in training *domain-specific predictor*

→ similar domains tend to have similar predictive functions

Relation-aware consistency regularizer

$$\boldsymbol{L_{rel}} = \mathbb{E}_{d \in \mathcal{D}^{tr}} \mathbb{E}_{(x,y) \sim P_d} [l(\frac{\sum_{j=1, j \neq d}^{N^{tr}} a_{dj} f^{(j)}(x)}{\sum_{k=1, k \neq d}^{N^{tr}} a_{dk}}, y)].$$

$$\boldsymbol{L = L_{pred} + \lambda L_{rel}}$$

(1) rely more on predictions made by similar domains

(2) strengthen the relations between predictors and help training predictors for domains with insufficient data

Method

Building Domain-Specific Models

Test Stage.

$\boldsymbol{D^3G}$ constructs test domain-specific models based on the same assumption.

→ similar domains have similar predictive functions.

Prediction $\widehat{\boldsymbol{y}} = \frac{\sum_{d=1}^{N^{tr}} a_{dt} f^{(d)}(x)}{\sum_{k=1}^{N^{tr}} a_{kt}}$

Method

Extracting and Refining Domain Relations

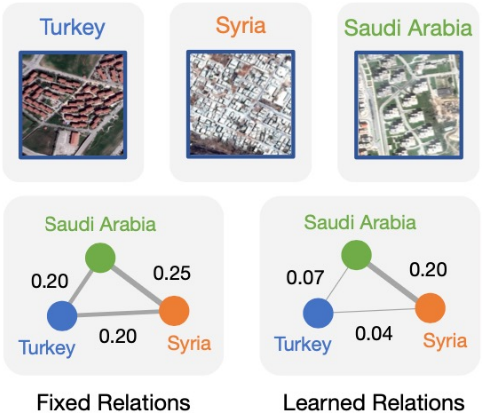
Domain Relations : $\mathcal{A} = \{a_{ij}\}_{i,j=1}^{D}$

→ Derived from **domain meta-data**

How to define?

Domain Relations = **Fixed Relations** + **Learned Relations**

$$a_{ij} = \beta a_{ij}^{g} + (1-\beta) a_{ij}^{l}$$



Method

Extracting and Refining Domain Relations

Fixed relations (directly collecting from domain-meta data)

(1) a relation graph in domain meta-data

(2) pairwise similarity calculated from each domain's meta-data

$\Rightarrow$ a_{ij}^{g}

Learned relations

fixed relations may not fully reflect accurate application-specific domain relation

$$\boldsymbol{a_{ij}^{l}} = \frac{\boldsymbol{1}}{\boldsymbol{R}}\sum_{\boldsymbol{r=1}}^{\boldsymbol{R}} \mathbf{cos}(\boldsymbol{w_r} \odot \boldsymbol{g}(\boldsymbol{m_i}), \boldsymbol{w_r} \odot \boldsymbol{g}(\boldsymbol{m_j}))$$

Algorithm

Training and Test Procedure of D^3G

Algorithm 1 Training and Test Procedure of D^{3}G

Require: Training and test data, relation combining coefficient β, loss balanced coefficient λ, meta-data $\{m_d\}_{d=1}^{D}$ of all domains, learning rate γ

/* *Training stage* */
Initialize all learnable parameters
Extract fixed relations $\{a_{ij}^{g}\}_{i,j=1}^{N^{tr}}$.
while not converge **do**
  Compute learned relations $\{a_{ij}^{l}\}_{i,j=1}^{N^{tr}}$ and obtain the final domain relations by equation 6.
  for each example (x, y, d) **do**
    Calculated supervised loss $\mathcal{L}_{pred}$ by equation 2.
    Computed consistency loss $\mathcal{L}_{rel}$ by equation 3 using domain relations.
  Update learnable parameters with learning rate γ.
/* *Test stage* */
for each test domain t **do**
  Obtain the relations between the test domain and training domains $\{a_{dt}\}_{d=1}^{N^{tr}}$
  for each example (x, y, t) **do**
    Computed the prediction $\hat{y}$ by equation 4.

Algorithm

Training and Test Procedure of D^3G

Algorithm 1 Training and Test Procedure of $\mathrm{D}^3\mathrm{G}$

Require: Training and test data, relation combining coefficient β, loss balanced coefficient λ, meta-data $\{m_d\}_{d=1}^{D}$ of all domains, learning rate γ

```
/* Training stage                                                                   */
Initialize all learnable parameters
Extract fixed relations {a^g_ij}_{i,j=1}^{N^tr}.
while not converge do
    Compute learned relations {a^l_ij}_{i,j=1}^{N^tr} and obtain the final domain relations by equation 6.
    for each example (x, y, d) do
        Calculated supervised loss L_pred by equation 2.
        Computed consistency loss L_rel by equation 3 using domain relations.
    Update learnable parameters with learning rate γ.
/* Test stage                                                                       */
for each test domain t do
    Obtain the relations between the test domain and training domains {a_dt}_{d=1}^{N^tr}
    for each example (x, y, t) do
        Computed the prediction ŷ by equation 4.
```

$$\boldsymbol{a_{ij}^l = \frac{1}{R}\sum_{r=1}^{R} \cos(w_r \odot g(m_i), w_r \odot g(m_j))}$$

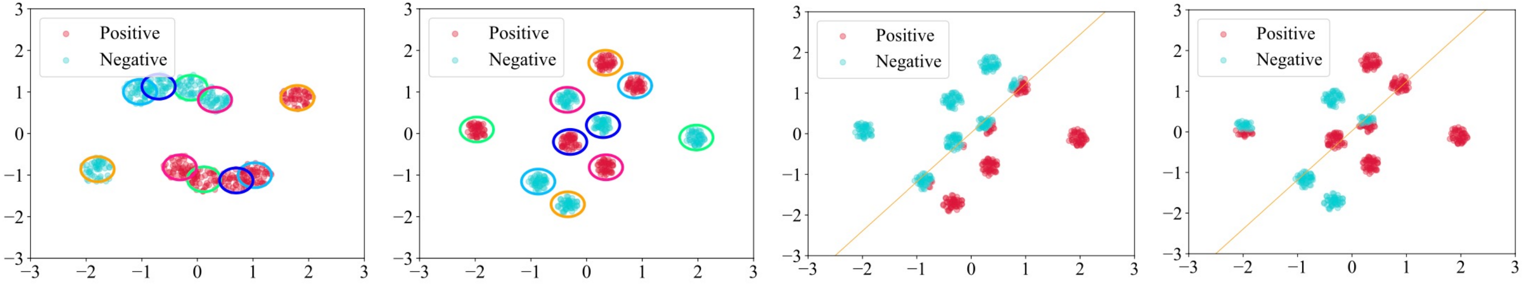
$$\boldsymbol{a_{ij} = \beta a_{ij}^g + (1-\beta) a_{ij}^l}$$

$$\boldsymbol{L = L_{pred} + \lambda L_{rel}}$$

$$\hat{\boldsymbol{y}} = \frac{\sum_{d=1}^{N^{tr}} a_{dt} f^{(d)}(x)}{\sum_{k=1}^{N^{tr}} a_{kt}}$$

Experiments

Illustrative Toy Task



(a) Training Distribution (b) Test Distribution (c) GroupDRO (d) D^{3}G

Model	ERM	GroupDRO	IRM	IB-IRM	IB-ERM	V-REx	DANN	CORAL
Accuracy	44.0%	47.7%	43.9%	45.4%	43.1%	44.0%	43.1%	43.5%
Model	MMD	RSC	CAD	SelfReg	Mixup	LISA	MAT	RaMoE
Accuracy	41.3%	58.2%	43.3%	40.7%	41.3%	47.4%	39.6%	53.7%
Model	mDSDI	AFFAR	GRDA	DRM	LLE	DDN	TRO	**D^{3}G (ours)**
Accuracy	45.2%	58.2%	59.5%	61.4%	58.8%	<u>66.2%</u>	56.3%	**77.5%**

Experiments

Performance between D^3G and other baselines

	TPT-48 (MSE ↓)		FMoW (Worst Acc. ↑)		ChEMBL-STRING (ROC-AUC ↑)	
	N (24) → S (24)	E (24) → W (24)	FMoW-Asia	FMoW-WILDS	$\text{PPI}_{>50}$	$\text{PPI}_{>100}$
	Region Shift	Region Shift	Region Shift	Region-Time Shift	Protein Shift	Protein Shift
ERM	0.445 ± 0.029	0.328 ± 0.033	26.05 ± 3.84%	34.87 ± 0.41%	74.11 ± 0.35%	71.91 ± 0.24%
GroupDRO	0.413 ± 0.045	0.434 ± 0.082	26.24 ± 1.85%	31.16 ± 2.12%	73.98 ± 0.25%	71.55 ± 0.59%
IRM	0.429 ± 0.043	0.262 ± 0.034	25.02 ± 2.38%	32.54 ± 1.92%	52.71 ± 0.50%	51.73 ± 1.54%
IB-IRM	0.416 ± 0.009	0.272 ± 0.026	26.30 ± 1.51%	34.94 ± 1.38%	52.12 ± 0.91%	52.33 ± 1.06%
IB-ERM	0.458 ± 0.032	0.273 ± 0.030	26.78 ± 1.34%	35.52 ± 0.79%	74.69 ± 0.14%	73.32 ± 0.21%
V-REx	0.412 ± 0.042	0.343 ± 0.021	26.63 ± 0.93%	37.64 ± 0.92%	71.46 ± 1.47%	69.37 ± 0.85%
DANN	0.394 ± 0.019	0.515 ± 0.156	25.62 ± 1.59%	33.78 ± 1.55%	73.49 ± 0.45%	72.22 ± 0.10%
CORAL	0.401 ± 0.022	0.283 ± 0.048	25.87 ± 1.97%	36.53 ± 0.15%	75.42 ± 0.15%	73.10 ± 0.14%
MMD	0.409 ± 0.067	0.279 ± 0.026	25.06 ± 2.19%	35.48 ± 1.81%	75.11 ± 0.27%	73.30 ± 0.50%
RSC	0.421 ± 0.040	0.330 ± 0.068	25.73 ± 0.70%	34.59 ± 0.42%	74.83 ± 0.68%	72.47 ± 0.38%
CAD	n/a	n/a	26.13 ± 1.82%	35.17 ± 1.73%	75.17 ± 0.64%	72.92 ± 0.39%
SelfReg	n/a	n/a	24.81 ± 1.77%	37.33 ± 0.87%	75.42 ± 0.42%	72.63 ± 0.71%
Mixup	0.574 ± 0.030	0.357 ± 0.011	26.99 ± 1.27%	35.67 ± 0.53%	74.40 ± 0.54%	71.31 ± 1.06%
LISA	0.467 ± 0.032	0.345 ± 0.014	26.05 ± 2.09%	34.59 ± 1.28%	74.30 ± 0.59%	71.45 ± 0.44%
MAT	0.423 ± 0.027	0.291 ± 0.024	25.92 ± 2.83%	35.07 ± 0.84%	74.73 ± 0.30%	72.07 ± 0.81%
AdaGraph	n/a	n/a	25.91 ± 0.59%	35.42 ± 0.55%	74.02 ± 0.42%	72.10 ± 0.06%
RaMoE	0.372 ± 0.035	0.311 ± 0.060	26.65 ± 0.46%	36.51 ± 0.71%	74.99 ± 0.22%	71.48 ± 0.49%
mDSDI	0.445 ± 0.027	0.315 ± 0.089	25.54 ± 0.46%	36.35 ± 0.45%	75.09 ± 0.47%	71.23 ± 0.69%
ADDAR	0.403 ± 0.061	0.287 ± 0.040	25.87 ± 1.01%	35.77 ± 0.70%	74.55 ± 0.54%	71.93 ± 0.33%
GRDA	0.373 ± 0.040	0.355 ± 0.068	26.57 ± 0.70%	34.41 ± 0.42%	75.01 ± 0.68%	73.57 ± 0.38%
DRM	0.571 ± 0.038	0.557 ± 0.027	25.22 ± 2.33%	36.39 ± 0.76%	74.34 ± 0.48%	72.41 ± 0.76%
LLE	0.603 ± 0.041	0.467 ± 0.047	26.37 ± 1.19%	35.83 ± 1.00%	74.01 ± 0.63%	71.68 ± 0.61%
DDN	0.537 ± 0.024	0.601 ± 0.038	26.77 ± 1.72%	35.13 ± 0.62%	75.17 ± 0.61%	72.71 ± 0.59%
TRO	0.371 ± 0.054	0.281 ± 0.066	26.87 ± 1.26%	37.48 ± 0.55%	74.85 ± 0.27%	72.49 ± 0.36%
D^3G (ours)	**0.342 ± 0.019**	**0.236 ± 0.063**	**28.12 ± 0.28%**	**39.47 ± 0.57%**	**78.67 ± 0.16%**	**77.24 ± 0.30%**

Domain-invariant learning approaches

Domain-specific learning approaches

Conclusion

Contributions

1) Novel method called $\boldsymbol{D^3G}$

2) Leverages the connections between different domains & employs **a domain-relationship aware weighting** system

(With **domain meta-data**)

3) Evaluate the **effectiveness** of $\boldsymbol{D^3G}$

Q&A

Out-of-distribution Generalization

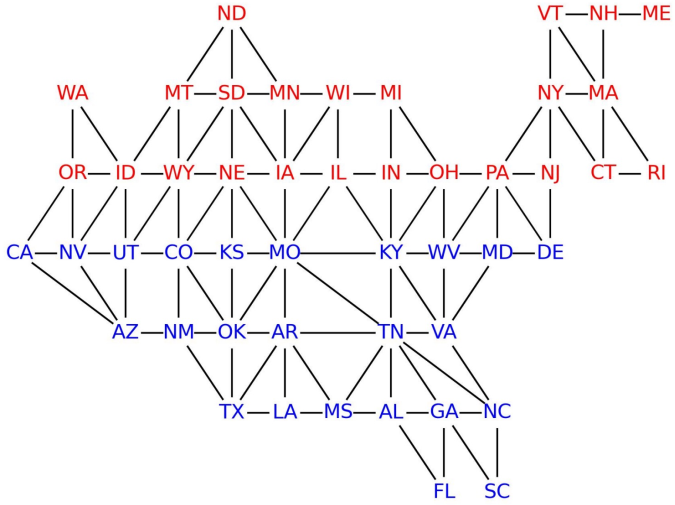
Previous works

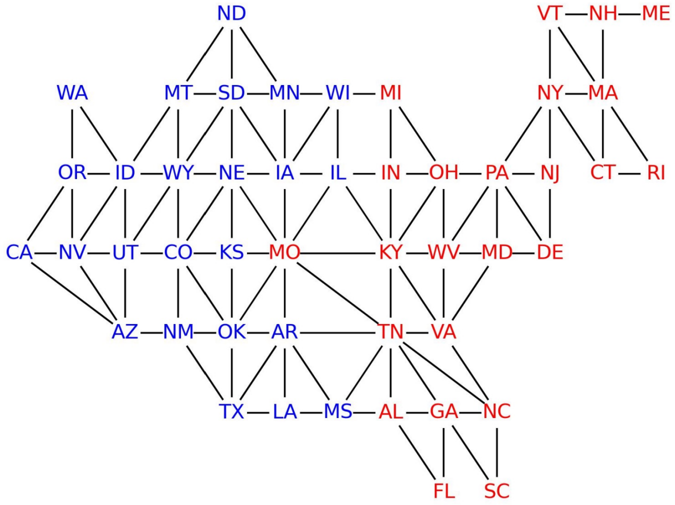
(1) minimizing the divergence of feature distributions

(2) generating more domains and enhancing the consistency among representations

(3) find a predictor that is invariant across domains by imposing an explicit regularizer

Appendix


ND
VT NH ME
WA MT SD MN WI MI NY MA
OR ID WY NE IA IL IN OH PA NJ CT RI
CA NV UT CO KS MO KY WV MD DE
AZ NM OK AR TN VA
TX LA MS AL GA NC
FL SC


ND
VT NH ME
WA MT SD MN WI MI NY MA
OR ID WY NE IA IL IN OH PA NJ CT RI
CA NV UT CO KS MO KY WV MD DE
AZ NM OK AR TN VA
TX LA MS AL GA NC
FL SC

Appendix

Table 10: Comparison of using different relations. The results on FMoW and ChEMBL-STRING are reported. In this case, when no relations are used, we take the average of predictions across all domains.

Fixed relations	Learned relations	FMoW (Worst Acc. ↑) FMoW-Asia	FMoW-WILDS	ChEMBL-STRING (ROC-AUC ↑) $\text{PPI}_{>50}$	$\text{PPI}_{>100}$
		$26.93 \pm 0.47\%$	$35.32 \pm 0.66\%$	$76.17 \pm 0.21\%$	$73.38 \pm 0.13\%$
✓		$27.43 \pm 0.41\%$	$39.37 \pm 0.34\%$	$77.66 \pm 0.32\%$	$76.59 \pm 0.40\%$
	✓	$21.18 \pm 2.30\%$	$36.41 \pm 1.09\%$	$77.09 \pm 0.94\%$	$75.57 \pm 1.20\%$
✓	✓	$\mathbf{28.12 \pm 0.28\%}$	$\mathbf{39.47 \pm 0.57\%}$	$\mathbf{78.67 \pm 0.16\%}$	$\mathbf{77.24 \pm 0.30\%}$

Table 11: Full results of comparison between D^3G with domain-specific fine-tuning.

Model	FMoW (Worst Acc. ↑) FMoW-Asia	FMoW-WILDS	ChEMBL (ROC-AUC ↑) $\text{PPI}_{>50}$	$\text{PPI}_{>100}$
ERM	$26.05 \pm 3.84\%$	$34.87 \pm 0.41\%$	$74.11 \pm 0.35\%$	$71.91 \pm 0.24\%$
CORAL	$25.87 \pm 1.97\%$	$36.53 \pm 0.15\%$	$75.42 \pm 0.15\%$	$73.10 \pm 0.14\%$
RW-FT	$27.03 \pm 1.03\%$	$36.39 \pm 1.28\%$	$76.31 \pm 0.35\%$	$74.30 \pm 0.40\%$
D^3G	$\mathbf{28.12 \pm 0.28\%}$	$\mathbf{39.47 \pm 0.57\%}$	$\mathbf{78.67 \pm 0.16\%}$	$\mathbf{77.24 \pm 0.30\%}$