FREE LUNCH FOR DOMAIN ADVERSARIAL TRAINING: ENVIRONMENT LABEL SMOOTHING

Yi-Fan Zhang[1,2,*], **Xue Wang**[3], **Jian Liang**[1,2],
Zhang Zhang[1,2], **Liang Wang**[1,2], **Rong Jin**[3,†], **Tieniu Tan**[1,2]
[1]National Laboratory of Pattern Recognition (NLPR), Institute of Automation
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)
[3] Machine Intelligence Technology, Alibaba Group.

ICLR 2023

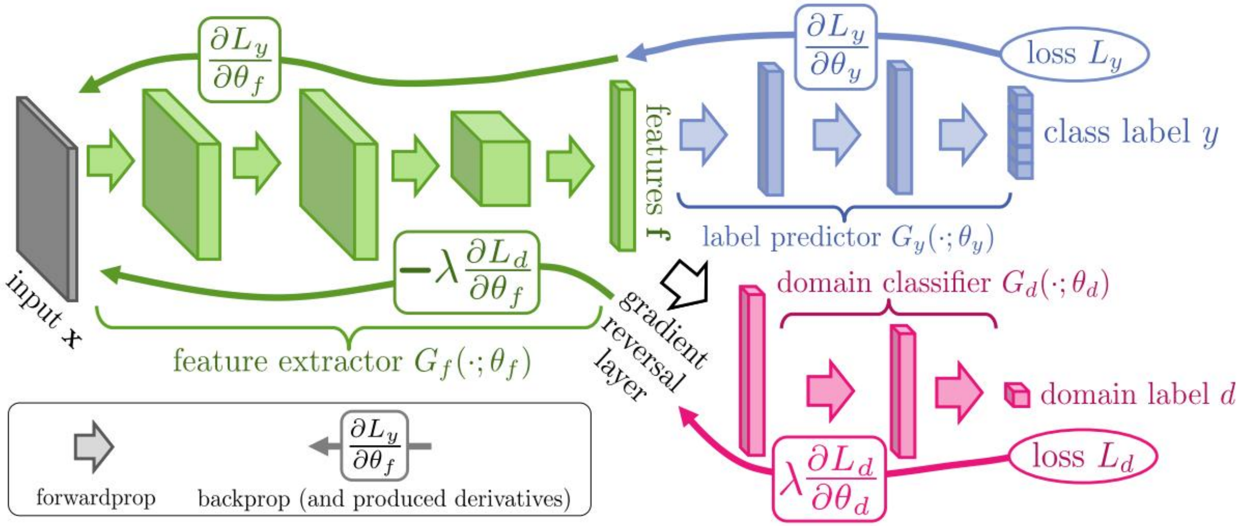
Reporter: 王启迅

2023/2/17

https://github.com/yfzhang114/Generalization-Causality

Introduction

Minimizing domain divergence by **Domain Adversarial Training (DAT)** is effective for extracting **domain-invariant features**, powerful for domain adaptation and domain generalization.



DANN framework

$$\mathcal{L}_{gen} := \min_{w,\phi} \mathcal{L}_t(w,\phi) - \lambda \mathcal{L}_{dom}(\eta)$$

$$\mathcal{L}_{disc} := \min_{\eta} \mathcal{L}_{dom}(\eta)$$

$$\mathcal{L}_t(w,\phi) = \frac{1}{n} \sum_{x_i, y_i \in \mathcal{D}_{tr}} \ell(w \circ \phi(x_i), y_i)$$

$$\mathcal{L}_{dom}(\eta) = -\frac{1}{n} \sum_{x_i, e_i \in \mathcal{D}_{tr}} \log(p_i)$$

Introduction

Problem of DANN: training instability

(i) Noise from **domain partition** (e.g. VLCS), and when the encoder gets better, the generated features from different domains are more **similar**, but features are still labeled differently.

(ii)DAT assign one-hot domain label, leading to highly oscillatory gradients.

Inspiration:

1. robust to environment-label noise
2. soft probability prediction by discriminator

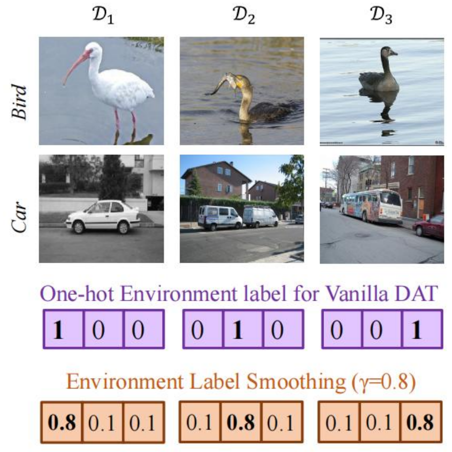


Figure 1: **A motivating example** of ELS with 3 domains on the VLCS dataset.

Proposed Method: Environment Label Smoothing (ELS)

M source domains $\{\mathcal{D}_i\}_{i=1}^{M}$, hypothesis $h = \hat{h} \circ g$, $g \in \mathcal{G}$ is the feature extractor mapping sample to representation space $\mathcal{Z}$,

$\hat{h} = \left(\hat{h}_1(\cdot), \dots, \hat{h}_M(\cdot)\right) \in \hat{\mathcal{H}} : \mathcal{Z} \to [0,1]^M; \sum_{i=1}^{M} \hat{h}_i(\cdot) = 1$ is the domain discriminator,

$\hat{h}' \in \hat{\mathcal{H}}' : \mathcal{Z} \to [0,1]^C; \sum_{i=1}^{C} \hat{h}'_i(\cdot) = 1$ is the classifier

DAT object:

$$\min_{\hat{h}', g} \max_{\hat{h}} \frac{1}{M} \sum_{i=1}^{M} \mathbb{E}_{\mathbf{x} \in \mathcal{D}_i} \left[\ell\left(\hat{h}' \circ g(\mathbf{x}), y \right) \right] + \lambda d_{\hat{h}, g}(\mathcal{D}_1, \dots, \mathcal{D}_M),$$

$$\max_{\hat{h} \in \hat{\mathcal{H}}} d_{\hat{h}, g}(\mathcal{D}_1, \dots, \mathcal{D}_M) = \max_{\hat{h} \in \mathcal{H}} \mathbb{E}_{\mathbf{x} \in \mathcal{D}_1} \log \hat{h}_1 \circ g(\mathbf{x}) + \cdots + \mathbb{E}_{\mathbf{x} \in \mathcal{D}_M} \log \hat{h}_M \circ g(\mathbf{x})$$

Proposed Method: Environment Label Smoothing (ELS)

After applying **ELS**, the maximization target is reformulated as:

$$
\begin{aligned}
\max_{\hat{h}\in\hat{\mathcal{H}}} d_{\hat{h},g,\gamma}(\mathcal{D}_1,\dots,\mathcal{D}_M) = \max_{\hat{h}\in\hat{\mathcal{H}}} \mathbb{E}_{\mathbf{x}\in\mathcal{D}_1}\left[\gamma \log \hat{h}_1 \circ g(\mathbf{x}) + \frac{(1-\gamma)}{M-1}\sum_{j=1;j\neq 1}^{M} \log\Big(\hat{h}_j \circ g(\mathbf{x})\Big)\right] + \cdots + \\
\mathbb{E}_{\mathbf{x}\in\mathcal{D}_M}\left[\gamma \log \hat{h}_M \circ g(\mathbf{x}) + \frac{(1-\gamma)}{M-1}\sum_{j=1;j\neq M}^{M} \log\Big(\hat{h}_j \circ g(\mathbf{x})\Big)\right]
\end{aligned}
$$

Theoretical Validation

- **Divergence minimization interpretation** (a 2-domain example)

Proposition 1. *Given two domain distributions $\mathcal{D}_S, \mathcal{D}_T$ over X, and a hypothesis class $\mathcal{H}$. We suppose $\hat{h} \in \hat{\mathcal{H}}$ the optimal discriminator with no constraint, denote the mixed distributions with hyper-parameter $\gamma \in [0.5, 1]$ as* $\begin{cases} \mathcal{D}_{S'} = \gamma \mathcal{D}_S + (1-\gamma)\mathcal{D}_T \\ \mathcal{D}_{T'} = \gamma \mathcal{D}_T + (1-\gamma)\mathcal{D}_S \end{cases}$. *Then minimizing domain divergence by adversarial training with* **ELS** *is equal to minimizing $2D_{JS}(\mathcal{D}_{S'} \| \mathcal{D}_{T'}) - 2\log 2$, where D_{JS} is the Jensen-Shanon (JS) divergence.*

- DANN results (Acuna et al., 2021): equal to minmizing $2D_{JS}(\mathcal{D}_S \| \mathcal{D}_T) - 2\log 2$
- Comparing to vanilla DANN: more flexible control on divergence minimization
- γ=1, original DAT;
- γ=0.5, $D_{JS}(\mathcal{D}_{S'} \| \mathcal{D}_{T'}) = 0$, the AT term does not provide gradient -> converge like ERM

- λ cannot affect the training dynamic, since it can only adjust $2\lambda \nabla D_{JS}(\mathcal{D}_S, \mathcal{D}_T)$
- e.g., when $\mathcal{D}_S, \mathcal{D}_T$ have disjoint support, $2\lambda \nabla D_{JS}(\mathcal{D}_S, \mathcal{D}_T)$=0 because $D_{JS}(\mathcal{D}_S, \mathcal{D}_T)$ is a constant, but $D_{JS}(\mathcal{D}_{S'} \| \mathcal{D}_{T'})$ is not.

Theoretical Validation-Training Stability

- ①The main source of training instability of GANs is the real and the generated distributions have **disjoint supports**. (Arjovsky & Bottou, 2017; Roth et al., 2017). Adding noise can extend the support of distributions.
- ELS can be seen as a kind of noise injection $\mathcal{D}_{S'} = \mathcal{D}_T + \gamma(\mathcal{D}_S - \mathcal{D}_T)$

- ② Nn vanilla DANN, as the discriminator gets better, **the gradient passed from discriminator to the encoder vanishes**, making training hard (Arjovsky & Bottou, 2017)
- ELS can relieve the gradient vanishing phenomenon

Proposition 5. *Denote $g(\theta;\cdot):\mathcal{X}\rightarrow\mathcal{Z}$ a differentiable function that induces distributions $\{\mathcal{D}_i^z\}_{i=1}^M$ with parameter θ, and $\{\hat{h}_i\}_{i=1}^M$ corresponding differentiable discriminators. If optimal discriminators for induced distributions exist, given any ϵ-optimal discriminator $\hat{h}_i$, we have $\sup_{\mathbf{z}\in\mathcal{Z}} \parallel \nabla_{\mathbf{z}}\hat{h}_i(\mathbf{z}) \parallel_2 + |\hat{h}_i(\mathbf{z}) - \hat{h}_i^*(\mathbf{z})| < \epsilon$, assume the Jacobian matrix of $g(\theta;\mathbf{x})$ given $\mathbf{x}$ is bounded by $\sup_{\mathbf{x}\in\mathcal{X}}[\parallel J_\theta(g(\theta;\mathbf{x})) \parallel_2] \leq C$, then we have*

$$\lim_{\epsilon\rightarrow 0} \parallel \nabla_\theta d_{\hat{h},g}(\mathcal{D}_1,\ldots,\mathcal{D}_M) \parallel_2 = 0 \quad (31)$$

$$\lim_{\epsilon\rightarrow 0} \parallel \nabla_\theta d_{\hat{h},g,\gamma}(\mathcal{D}_1,\ldots,\mathcal{D}_M) \parallel_2 < M(1-\gamma)C \quad (32)$$

Theoretical Validation-Training Stability

- ③Gradients of the encoder w.r.t. adversarial loss remain **highly oscillatory** in native DANN, which is an important reason for the instability of AT (Mescheder et al., 2018)
- ELS makes the gradient smoother and more stable.

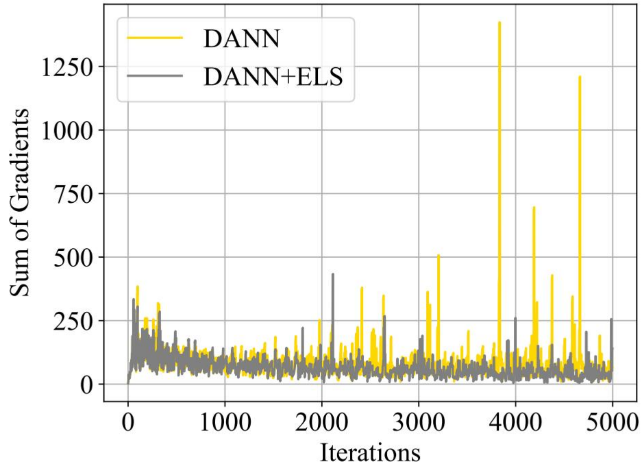


Figure 2: The sum of gradients provided to the encoder by the adversarial loss.

Theoretical Validation

- **ELS alleviate noisy domain label**
- noisy label $\tilde{y}$, noise rate $e = P(\tilde{y}=1 \mid y=0) = P(\tilde{y}=0 \mid y=1)$
- denote $f := \hat{h} \circ g$, $\tilde{y}^{\gamma}$ the smoothed noisy label
- minimizing the smoothed loss with noisy labels:

$$\min_f \mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathcal{D}}}[\ell(f(x),\tilde{y}^{\gamma})] = \min_f \mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathcal{D}}}[\gamma\ell(f(x),\tilde{y}) + (1-\gamma)\ell(f(x),1-\tilde{y})]$$
$$= \min_f \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\ell\left(f(x),y^{\gamma^*}\right)\right] + (\gamma^* - \gamma - e + 2\gamma e)\mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x),1-y) - \ell(f(x),y)]$$

- γ^* is the optimal smooth parameter on clean data.
- first term: risk under the clean label.
- second term: influence noisy labels: $\mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f(x),1-y) - \ell(f(x),y)]$

  (opposite of the optimization process)
  - without ELS: the weight is $\gamma^* - 1 + e$, high noise rate -> harmful contribution
  - with ELS: can tune $\gamma = \frac{\gamma^* - e}{1-2e}$ so that the second term =0

Theoretical Validation

- Gap: the analysis of proposition 1 and training stability are based on two unrealistic assumptions.
- (i) infinite data samples
- (ii) the discriminator is optimized over infinite-dimensional space

- In this paper, the author tries to analyze the gap $\left| d_{\hat{h},g}(\mathcal{D}_1, \ldots, \mathcal{D}_M) - d_{\hat{h},g}\left(\hat{\mathcal{D}}_1, \ldots, \hat{\mathcal{D}}_M\right) \right|$.

  the first attempt to study the empirical and parameterization gap of multi-domain AT

Proposition 6. *(Adapted from Theorem A.2 in (Arora et al., 2017)) Let* $\{\mathcal{D}_i\}_{i=1}^M$ *a set of distributions and* $\{\hat{\mathcal{D}}_i\}_{i=1}^M$ *be empirical versions with at least* n^* *samples each. We assume that the set of discriminators with softmax activation function* $\hat{h}(\theta;\cdot) = (\hat{h}_1(\theta_1,\cdot), \ldots, \hat{h}_M(\theta_M,\cdot)) \in \hat{\mathcal{H}} : \mathcal{Z} \to [0,1]^M; \sum_{i=1}^M \hat{h}_i(\theta_i;\cdot) = 1$[2] *are L-Lipschitz with respect to the parameters* θ *and use* p *denote the number of parameter* θ_i*. There is a universal constant* c *such that when* $n^* \geq \frac{cpM\log(Lp/\epsilon)}{\epsilon}$*, we have with probability at least* $1 - \exp(-p)$ *over the randomness of* $\{\hat{\mathcal{D}}_i\}_{i=1}^M$*,*

$$\mid d_{\hat{h},g}(\mathcal{D}_1, \ldots, \mathcal{D}_M) - d_{\hat{h},g}(\hat{\mathcal{D}}_1, \ldots, \hat{\mathcal{D}}_M) \mid \leq \epsilon \tag{48}$$

Theoretical Validation

- **Non-asymptotic convergence** (more precisely reveal the convergence of the dynamic system than the asymptotic analysis)
- (analyzing the convergence near a equilibrium point: converge to a local equilibrium point->satisfy conditions in Proposition 4.4.1 in [Bertsekas, 1999]->upper bound of lr->minimal step to converge)
- main results:

n_d

- (i) train_e the discriminator and encoder simulta$\eta \leq \frac{4}{\sqrt{n_d n_e c}}$, has **no guarantee of the non-asymptotic convergenc**$\eta \leq \frac{4}{\sqrt{n_d n_e c}} \frac{1}{2\gamma - 1}$
- (ii) alternately train the discriminator times once we train the encoder times -> **sublinear convergence** rate when lr
- (iii) ELS speeds up convergence:

Experiemnts

Table 1: **A summary on evaluation benchmarks.** Wg. acc. denotes worst group accuracy, 10 %/ acc. denotes 10th percentile accuracy. GIN (Xu et al., 2018) denotes Graph Isomorphism Networks, and CRNN (Gagnon-Audet et al., 2022) denotes convolutional recurrent neural networks.

Task	Dataset	Domains	Classes	Metric	Backbone	# Data Examples
Images Classification	Rotated MNIST	6 rotated angles	10	Avg. acc.	MNIST ConvNet	70,000
	PACS	4 image styles	7	Avg. acc.	ResNet50	9,991
	VLCS	4 image styles	5	Avg. acc.	ResNet50	10,729
	Office-31	3 image styles	31	Avg. acc.	ResNet50/ResNet18	4,110
	Office-Home	4 image styles	65	Avg. acc.	ResNet50/ViT	15,500
	Rotating MNIST	8 rotated angles	10	Avg. acc.	EncoderSTN	60,000
Image Retrieval	MS	5 locations	18,530	mAP, Rank m	MobileNet×1.4	121,738
Neural Language Processing	CivilComments	8 demographic groups	2	Avg/Wg acc.	DistillBERT	448,000
	Amazon	7676 reviewers	5	10 %/Avg/Wg acc.	DistillBERT	100,124
Genomics and Graph	RxRx1	51 experimental batch	1139	Wg/Avg/Test ID acc.	ResNet-50	125,510
	OGB-MolPCBA	120,084 molecular scaffold	128	Avg. acc.	GIN	437,929
Sequential Prediction	Spurious-Fourier	3 spurious correlations	2	Avg. acc.	LSTM	12,000
	HHAR	5 smart devices	6	Avg. acc.	Deep ConvNets	13,674

Experiemnts

Domain generalization

- DANN+ELS>DANN
- SOTA on VLCS

Table 5: **Rotating MNIST accuracy (%) at the source domain and each target domain.** X° denotes the domain whose images are Rotating by $[X^\circ, X^\circ + 45^\circ]$.

Algorithm	0°(Source)	45°	90°	135°	180°	225°	270°	315°	Average
	Rotating MNIST								
ERM (Vapnik, 1999)	99.2	79.7	26.8	31.6	35.1	37.0	28.6	76.2	45.0
ADDA (Tzeng et al., 2017)	97.6	70.7	22.2	32.6	38.2	31.5	20.9	65.8	40.3
DANN (Ganin et al., 2016)	98.4	**81.4**	38.9	35.4	40.0	43.4	48.8	77.3	52.1
CIDA (Wang et al., 2020)	**99.5**	80.0	33.2	**49.3**	**50.2**	**51.7**	**54.6**	**81.0**	57.1
DANN+ELS	98.4	**81.4**	**55.0**	39.9	43.7	45.9	53.7	78.7	**62.1**
↑	0.0	0.0	16.1	4.5	3.7	2.5	4.9	1.4	10.0

Table 3: The domain generalization accuracies (%) on VLCS, and PACS. ↑ denotes improvement of DANN+ELS compared to DANN.

Algorithm	PACS A	PACS C	PACS P	PACS S	PACS Avg	VLCS C	VLCS L	VLCS S	VLCS V	VLCS Avg
ERM (Vapnik, 1999)	87.8 ± 0.4	82.8 ± 0.5	97.6 ± 0.4	80.4 ± 0.6	87.2	97.7 ± 0.3	65.2 ± 0.4	73.2 ± 0.7	75.2 ± 0.4	77.8
IRM (Arjovsky et al., 2019)	85.7 ± 1.0	79.3 ± 1.1	97.6 ± 0.4	75.9 ± 1.0	84.6	97.6 ± 0.5	64.7 ± 1.1	69.7 ± 0.5	76.6 ± 0.7	77.2
DANN (Ganin et al., 2016)	85.4 ± 1.2	83.1 ± 0.8	96.3 ± 0.4	79.6 ± 0.8	86.1	98.6 ± 0.8	73.2 ± 1.1	72.8 ± 0.8	78.8 ± 1.2	80.8
ARM (Zhang et al., 2021b)	85.0 ± 1.2	81.4 ± 0.2	95.9 ± 0.3	80.9 ± 0.5	85.8	97.6 ± 0.6	66.5 ± 0.3	72.7 ± 0.6	74.4 ± 0.7	77.8
Fisher (Rame et al., 2021)	—	—	—	—	86.9	—	—	—	—	76.2
DDG (Zhang et al., 2021a)	**88.9 ± 0.6**	**85.0 ± 1.9**	97.2 ± 1.2	**84.3 ± 0.7**	**88.9**	**99.1 ± 0.6**	66.5 ± 0.3	73.3 ± 0.6	**80.9 ± 0.6**	80.0
DANN+ELS	87.8 ± 0.8	83.8 ± 1.6	97.1 ± 0.4	81.4 ± 1.3	87.5	**99.1 ± 0.3**	**73.2 ± 1.1**	**73.8 ± 0.9**	79.9 ± 0.9	**81.5**
↑	2.4	0.7	0.8	1.8	1.4	0.5	0	1	1.1	0.7

Experiemnts

Domain adaptation

- SDAT+ELS=SOTA

Table 2: **The domain adaptation accuracies (%) on Office-31**. ↑ denotes improvement of a method with ELS compared to that wo/ ELS.

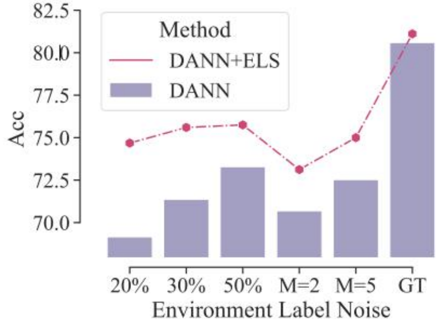
	A - W	D - W	W - D	A - D	D - A	W - A	Avg
				ResNet18			
ERM (Vapnik, 1999)	72.2	97.7	100.0	72.3	61.0	59.9	77.2
DANN (Ganin et al., 2016)	84.1	98.1	99.8	81.3	60.8	63.5	81.3
DANN+ELS	85.5	99.1	100.0	82.7	62.1	64.5	82.4
↑	1.4	1.0	0.2	1.4	1.3	1.1	1.1
SDAT (Rangwani et al., 2022)	87.8	98.7	100.0	82.5	73.0	72.7	85.8
SDAT+ELS	**88.9**	**99.3**	**100.0**	**83.9**	**74.1**	**73.9**	**86.7**
↑	1.1	0.5	0.0	1.4	1.1	1.2	0.9
				ResNet50			
ERM (Vapnik, 1999)	75.8	95.5	99.0	79.3	63.6	63.8	79.5
ADDA (Tzeng et al., 2017)	94.6	97.5	99.7	90.0	69.6	72.5	87.3
CDAN (Long et al., 2018)	93.8	98.5	100.0	89.9	73.4	70.4	87.7
MCC (Jin et al., 2020)	94.1	98.4	99.8	**95.6**	75.5	74.2	89.6
DANN (Ganin et al., 2016)	91.3	97.2	100.0	84.1	72.9	73.6	86.5
DANN+ELS	92.2	98.5	100.0	85.9	74.3	75.3	87.7
↑	0.9	1.3	0.0	1.8	1.4	1.7	1.2
SDAT (Rangwani et al., 2022)	92.7	98.9	100.0	93.0	78.5	75.7	89.8
SDAT+ELS	**93.6**	**99.0**	**100.0**	93.4	**78.7**	**77.5**	**90.4**
↑	0.9	0.1	0.0	0.4	0.2	1.8	0.6

Table 4: **Accuracy (%) on Office-Home for unsupervised DA** (with ResNet-50 and ViT backbone). SDAT+ELS outperforms other SOTA DA techniques and improves SDAT consistently.

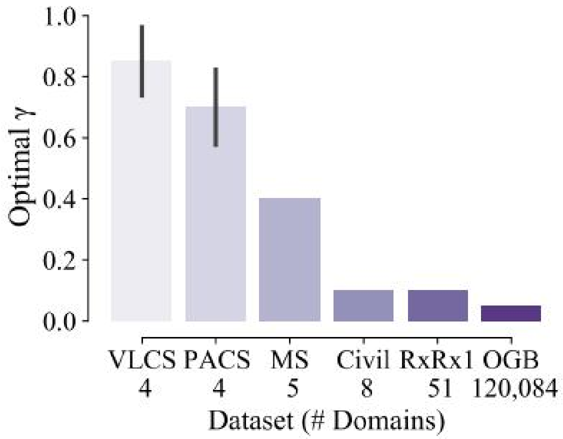
Method	Backbone	A-C	A-P	A-R	C-A	C-P	C-R	P-A	P-C	P-R	R-A	R-C	R-P	Avg
ResNet-50 (He et al., 2016)	ResNet-50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DANN (Ganin et al., 2016)		45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN (Long et al., 2018)		49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
MMD (Zhang et al., 2019)		54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	60.2	82.3	68.1
f-DAL (Acuna et al., 2021)		56.7	77.0	81.1	63.1	72.2	75.9	64.5	54.4	81.0	72.3	58.4	83.7	70.0
SRDC (Tang et al., 2020)		52.3	76.3	81.0	**69.5**	76.2	**78.0**	**68.7**	53.8	81.7	**76.3**	57.1	85.0	71.3
SDAT (Rangwani et al., 2022)		57.8	77.4	82.2	66.5	76.6	76.2	63.3	57.0	**82.2**	75.3	62.6	85.2	71.8
SDAT+ELS		**58.2**	**79.7**	**82.5**	67.5	**77.2**	77.2	64.6	**57.9**	**82.2**	75.4	**63.1**	**85.5**	**72.6**
↑		0.4	2.3	0.3	1.0	0.6	1.0	1.3	0.9	0.0	0.1	0.5	0.3	0.8
TVT (Yang et al., 2021)	ViT	**74.9**	86.6	89.5	82.8	87.9	88.3	79.8	71.9	90.1	85.5	74.6	90.6	83.6
CDAN (Long et al., 2018)		62.6	82.9	87.2	79.2	84.9	87.1	77.9	63.3	88.7	83.1	63.5	90.8	79.3
SDAT (Rangwani et al., 2022)		70.8	87.0	90.5	85.2	87.3	89.7	84.1	70.7	90.6	88.3	75.5	92.1	84.3
SDAT+ELS		72.1	**87.3**	**90.6**	**85.2**	**88.1**	**89.7**	**84.1**	**70.7**	**90.8**	**88.4**	**76.5**	**92.1**	**84.6**
↑		1.3	0.3	0.1	0.0	0.8	0.0	0.0	0.0	0.2	0.1	1.0	0.0	0.3

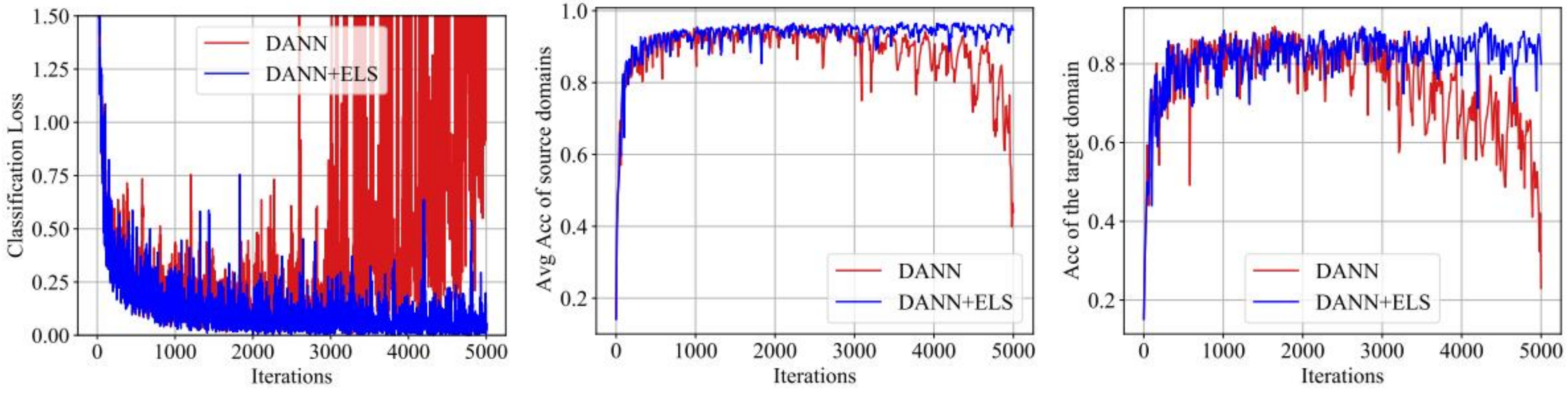
Experiemnts

- **Robustness to domain label noise**
- (GT: all labels are known)
- (M=2: partition all the training data randomly into two domains)



Another perspective of ELS:
avoid overconfident of discriminator
Discriminator more likely to be overconfident, smaller γ





(a) Classification loss. (b) Avg accuracy of source domains. (c) Acc on the target domain.

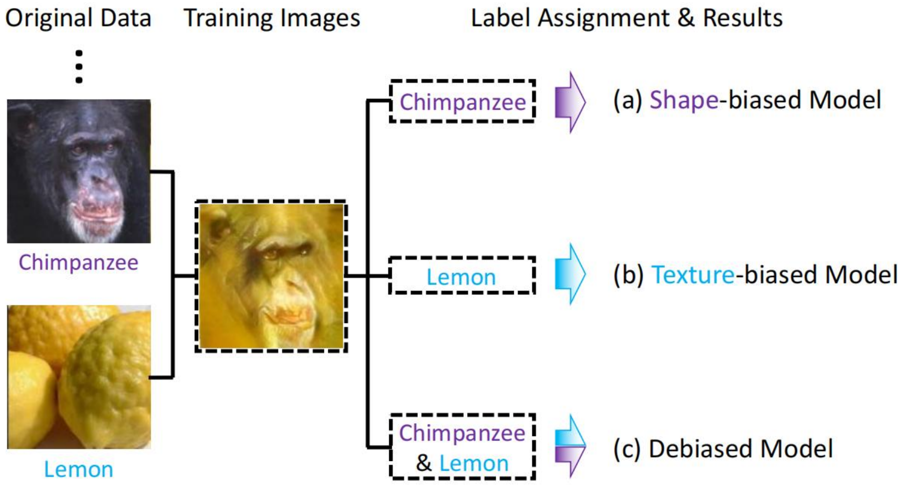
Figure 5: **Training statistics on PACS datasets.** Alternating GD with $n_d = 5, n_e = 1$ is used. All other parameters setting are the same and only on the default hyperparameters and without the fine-grained parametric search.

Insights

- Reduce the impact of noisy domain label
- Soft domain partition/soft label (Li et al., ICLR 2021)

$$\widetilde{y} = \gamma * y_s + (1 - \gamma) * y_t,$$

- Theoretical perspective



	IN Acc. ↑	IN-A Acc. ↑	IN-C mCE ↓	S-IN Acc. ↑	FGSM Acc. ↑
ResNet-50	76.4	2.0	75.0	7.4	17.1
CutMix + MoEx (Li et al., 2021)	79.0	8.0	74.8	**5.0**	41.0
DeepAugment + AugMix (Hendrycks et al., 2020)	**75.8**	3.9	53.6	21.2	18.8
SIN (Geirhos et al., 2019)	**60.2**	2.4	**77.3**	56.2	**5.6**
Shape-Texture Debiased Training (ours)	76.9	3.5	67.5	17.4	27.4