Domain Adversarial Neural Networks (DANN)

Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario Marchhand

D´epartement d’informatique et de g´enie logiciel, Universit´e Laval, Qu´ebec, Canada
D´epartement d’informatique, Universit´e de Sherbrooke, Qu´ebec, Canada

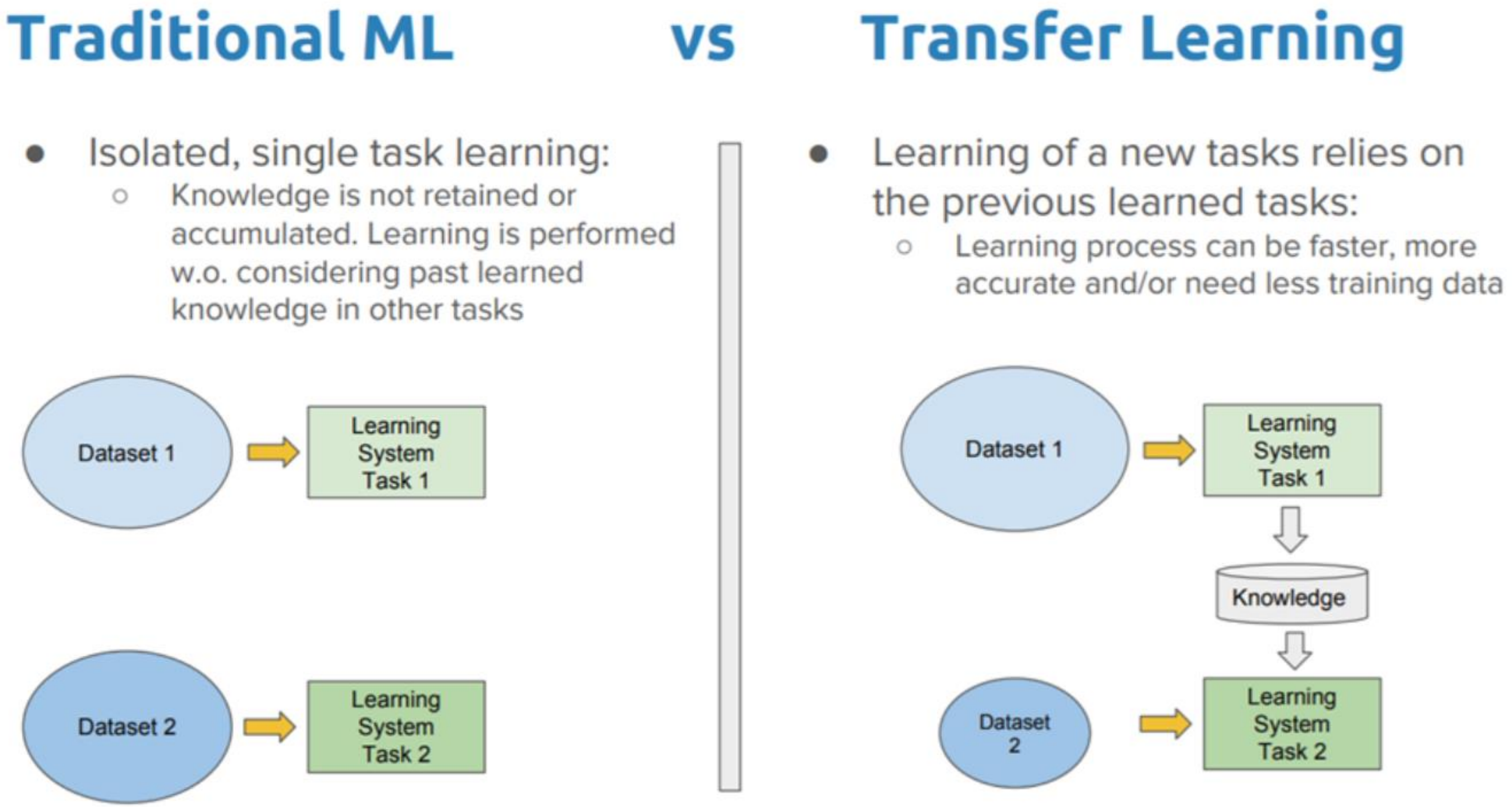
rayjang111@gmail.com
HyunSukJang

CONTENTS

01 Introduction

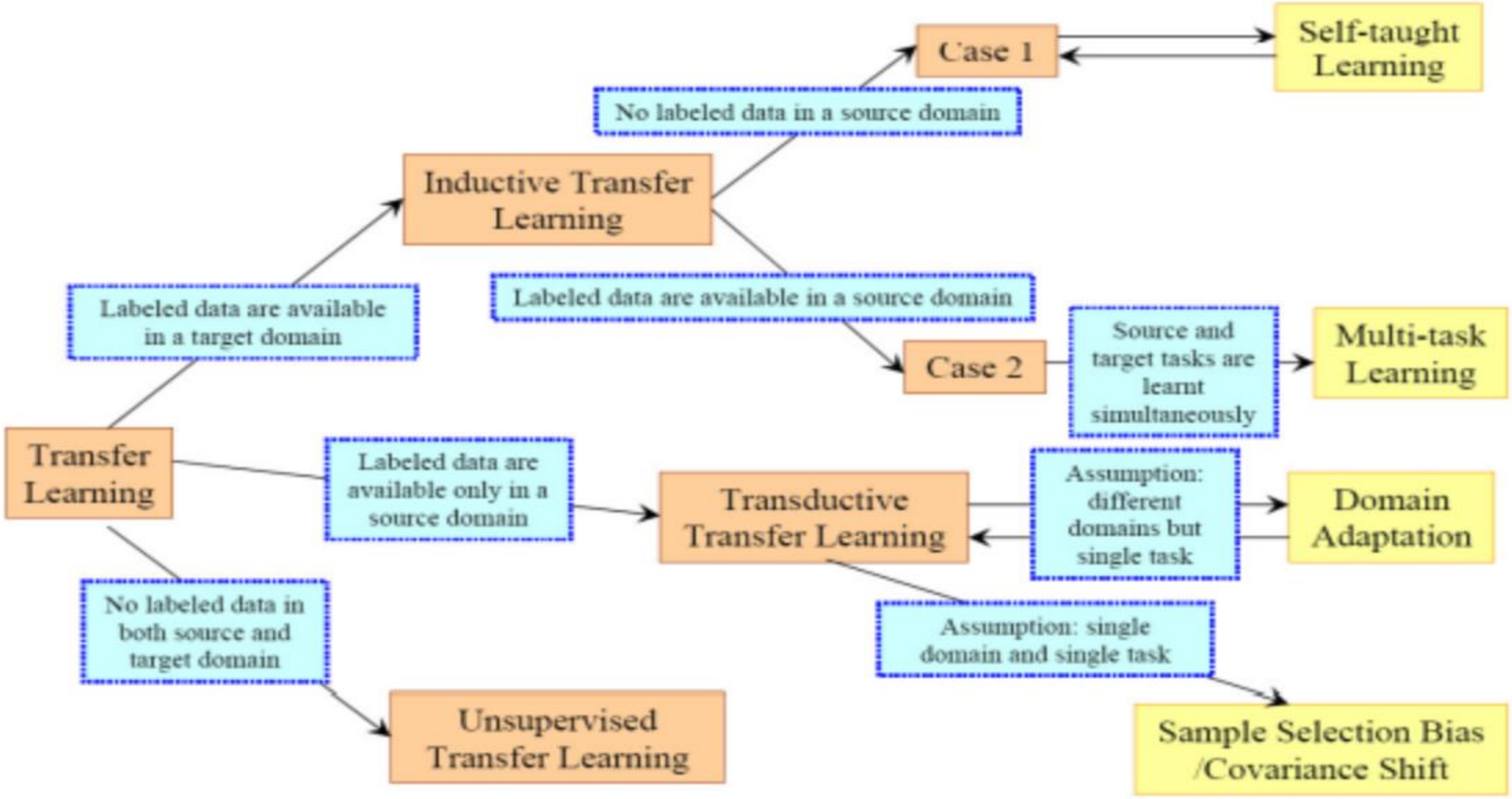
01. Introduction

What is Adaptation learning???



Traditional Learning vs Transfer Learning

01.Introduction



Transfer Learning Strategies

01. Introduction

02 DANN Model architecture

02. DANN Model architecture

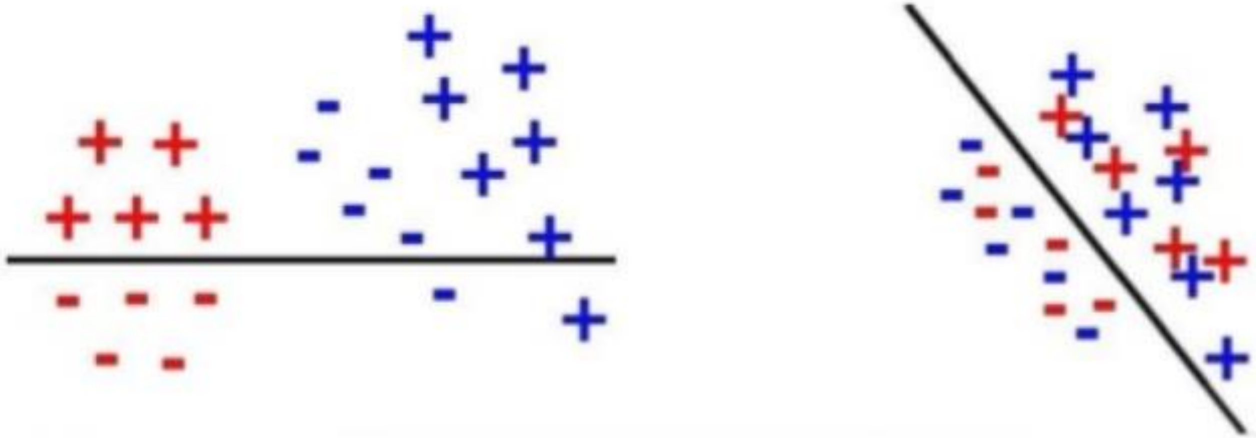
- **Labeled** Source Sample

  $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m_s}$ **Source** sample drawn i.i.d. from P_S

- **Unlabeled** Target Sample

  $T = \{\mathbf{x}_j\}_{j=1}^{m_t}$ **Target** Sample drawn i.i.d. from D_T
  optionnal: a few labeled target examples

If h is learned from **source** domain, how does it perform on **target** domain?



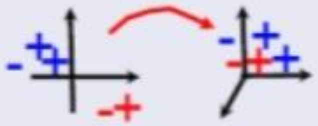
02. DANN Model architecture

Reweighting/Instance-based methods

Correct a sample bias by reweighting source labeled data: source instances close to target instances are more important.

Feature-based methods/Find new representation spaces

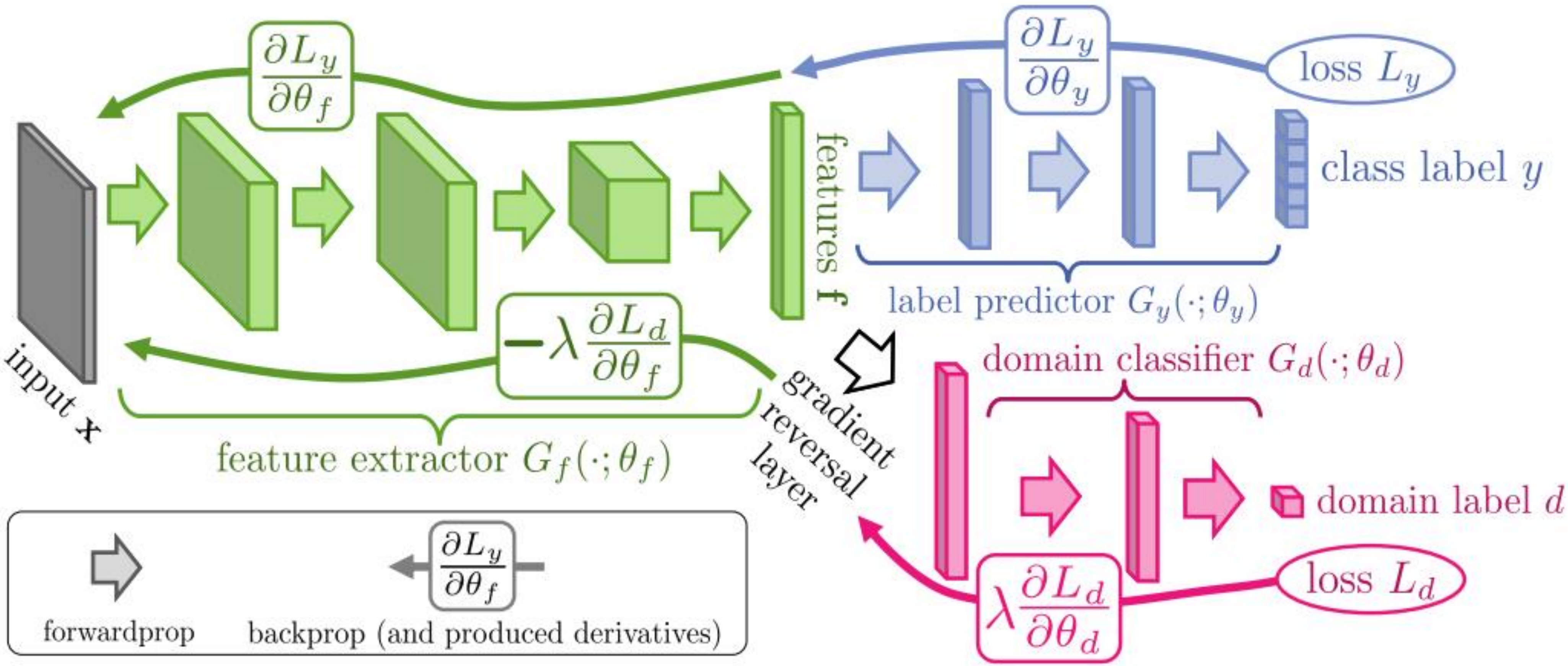
Find a common space where source and target are close (projection, new features, etc)



Ajustement/Iterative methods

Modify the model by incorporating pseudo-labeled information

02. DANN Model architecture

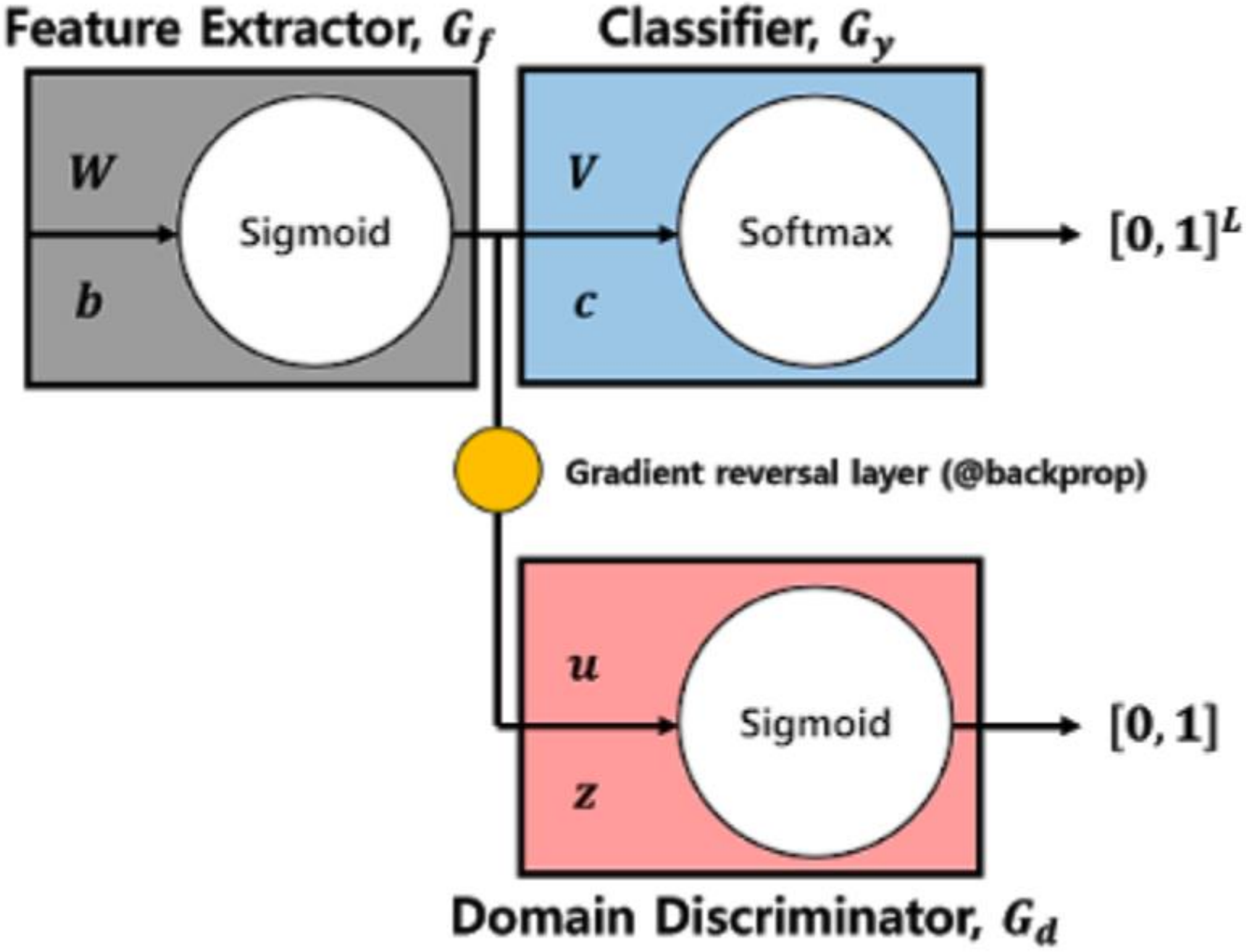


02. DANN Model architecture

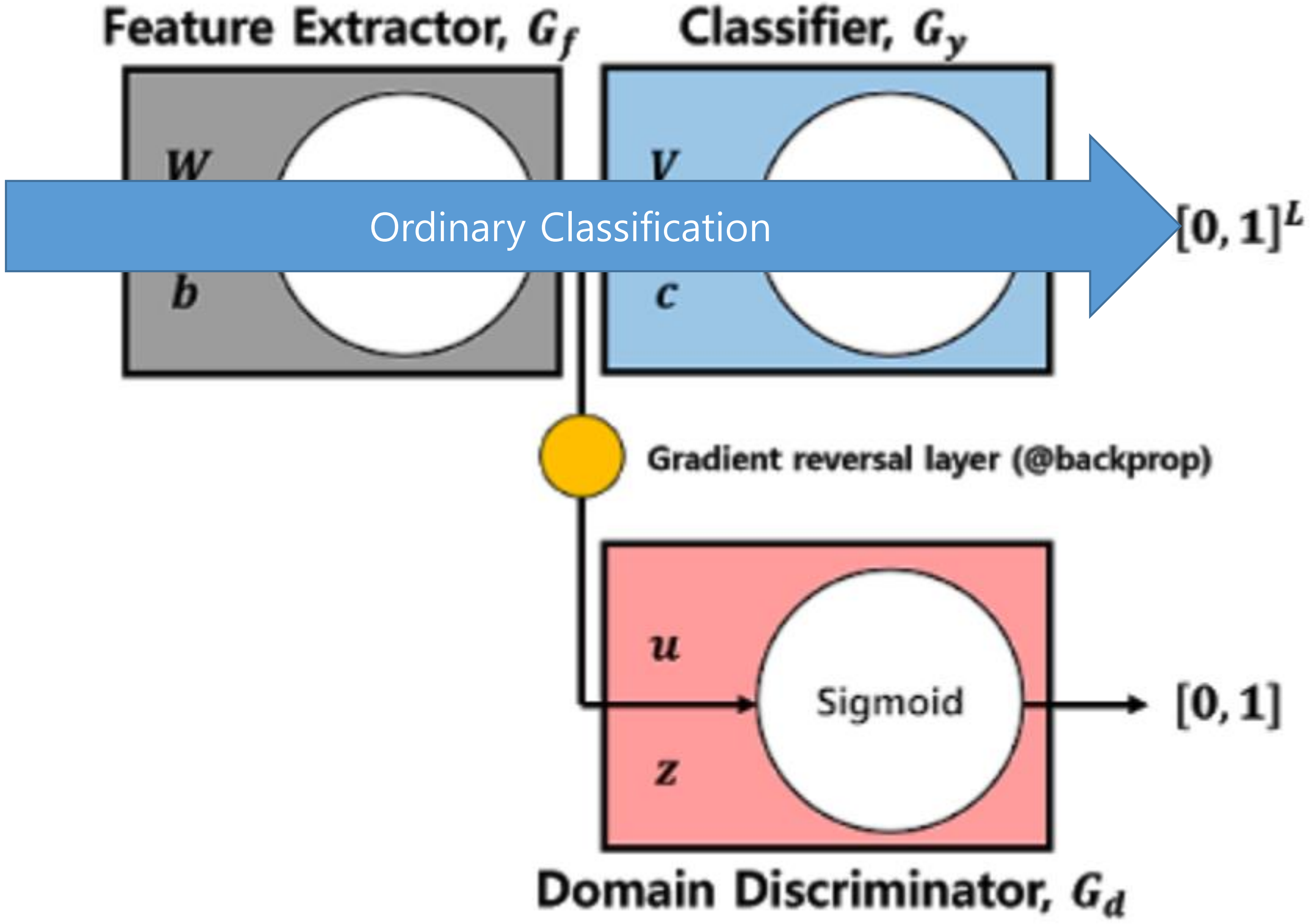
	Electronics	Video games
✓	(1) Compact; easy to operate; very good picture quality; looks sharp!	(2) A very good game! It is action packed and full of excitement. I am very much hooked on this game.
✓	(3) I purchased this unit from Circuit City and I was very excited about the quality of the picture. It is really nice and sharp.	(4) Very realistic shooting action and good plots. We played this and were hooked.
✗	(5) It is also quite blurry in very dark settings. I will never_buy HP again.	(6) It is so boring. I am extremely unhappy and will probably never_buy UbiSoft again.

- Source specific: *compact, sharp, blurry*.
- Target specific: *hooked, realistic, boring*.
- Domain independent: *good, excited, nice, never_buy, unhappy*.

02. DANN Model architecture

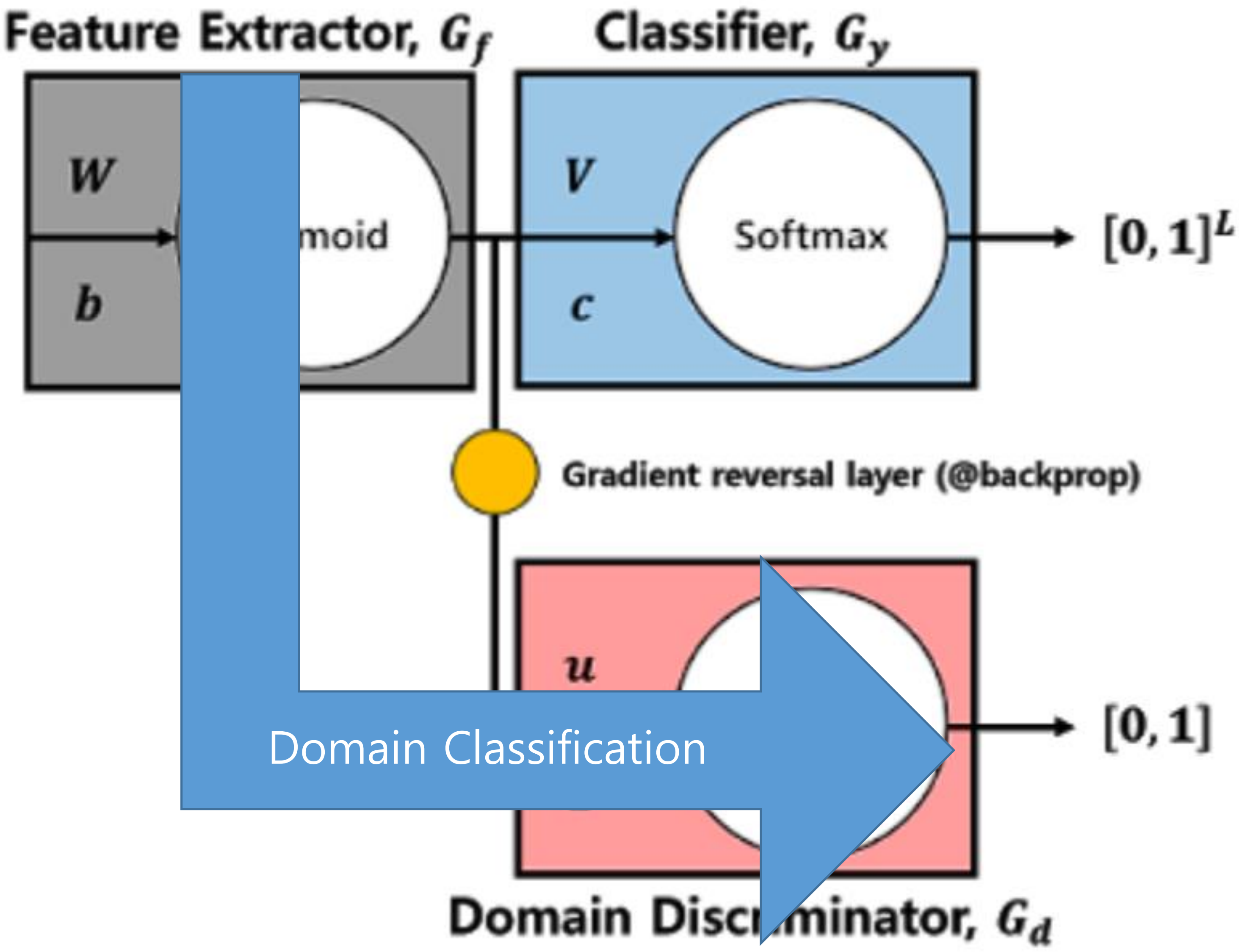


02. DANN Model architecture



0:Negative
1:Positive

02. DANN Model architecture



0: Video games
1:Electronics

02. DANN Model architecture

By minimizing classification error of source data& divergence between source domain and target domain
=>Features will be extracted to classify both source domain and target domain

03training&test method

Classical Test Error:

$$\epsilon_{\text{test}} \leq \hat{\epsilon}_{\text{train}} + \sqrt{\frac{\text{complexity}}{n}}$$

Measured on the same distribution!

Adaptation Target Error:

$$\epsilon_{\text{test}} \leq ??$$

Measured on a **new** distribution!

03. training&test method

Theorem 1.

(Ben-David et al.,2006) *Let $\mathcal{H}$ is a hypothesis class of VC dimension d. With probability $1-\delta$ over the choice of samples $S \sim (\mathbb{D}_S)^n$ and $T \sim (\mathbb{D}_T^X)^n$, for every $\eta \in \mathcal{H}$:*

$$R_{\mathbb{D}_T}(\eta) \leq R_S(\eta) + \sqrt{\frac{4}{m}(d \log \frac{2em}{d} + \log \frac{4}{\delta})} \tag{3}$$

$$+ \hat{d}_{\mathcal{H}}(S,T) + 4\sqrt{\frac{1}{m}(d \log \frac{2m}{d} + \log 4\delta)} + \beta, \tag{4}$$

with $\beta \geq \inf_{\eta^* \in \mathcal{H}} \left[R_{\mathbb{D}_S}(\eta^*) + R_{\mathbb{D}_T}(\eta^*)\right]$*, and*

$$R_S(\eta) = \frac{1}{m}\sum_{i=1}^{m} I\left[\eta(x_i) \neq y_i\right]$$

is the empirical source risk.

03. training&test method

Definition 1.

(Ben-David et al.,2006,2010; Kifer et al.,2004) *Given two domain distributions* $\mathbb{D}_S^X$ *and* $\mathbb{D}_T^X$ *over* X, *and a hypothesis class* $\mathcal{H}$, *the* $\mathcal{H}$*-divergence between* $\mathbb{D}_S^X$ *and* $\mathbb{D}_T^X$ *is*

$$\hat{h}_{\mathcal{H}}(\mathbb{D}_S^X, \mathbb{D}_T^X) = 2 \sup_{\eta \in \mathcal{H}} \left| \Pr_{x \sim \mathbb{D}_S^X} [\eta(x_i) = 1] - \Pr_{x \sim \mathbb{D}_T^X} [\eta(x_i) = 1] \right| .$$

03. training&test method

$$\hat{h}_{\mathcal{H}}(\mathbb{D}_S^X, \mathbb{D}_T^X) = 2 \sup_{\eta \in \mathcal{H}} \left| \Pr_{x \sim \mathbb{D}_S^X} [\eta(x_i) = 1] - \Pr_{x \sim \mathbb{D}_T^X} [\eta(x_i) = 1] \right|$$

$$= 2 \sup_{\eta \in \mathcal{H}} \left| \Pr_{x \sim \mathbb{D}_S^X} [\eta(x_i) = 1] + \Pr_{x \sim \mathbb{D}_T^X} [\eta(x_i) = 0] - 1 \right|$$

$$\hat{d}_{\mathcal{H}}(S, T) = 2 \left(1 - \min_{\eta \in \mathcal{H}} \underline{\left[\frac{1}{n} \sum_{i=1}^{n} I[\eta(x_i) = 1] + \frac{1}{n'} \sum_{i=n+1}^{N} I[\eta(x_i) = 0] \right]} \right),$$

$$\hat{h}_{\mathcal{H}}(S, T) = 2\,(1 - \underline{2\epsilon})$$

03. training&test method

$$E(W, V, b, c, u, z)$$

$$= \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_y^i(W, b, V, c) - \lambda\left(\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_d^i(W, b, u, z) + \frac{1}{n'}\sum_{i=n+1}^{N}\mathcal{L}_d^i(W, b, u, z)\right),$$

where we are seeking the parameters $\hat{W}, \hat{V}, \hat{b}, \hat{c}, \hat{u}, \hat{z}$ that deliver a saddle point given by

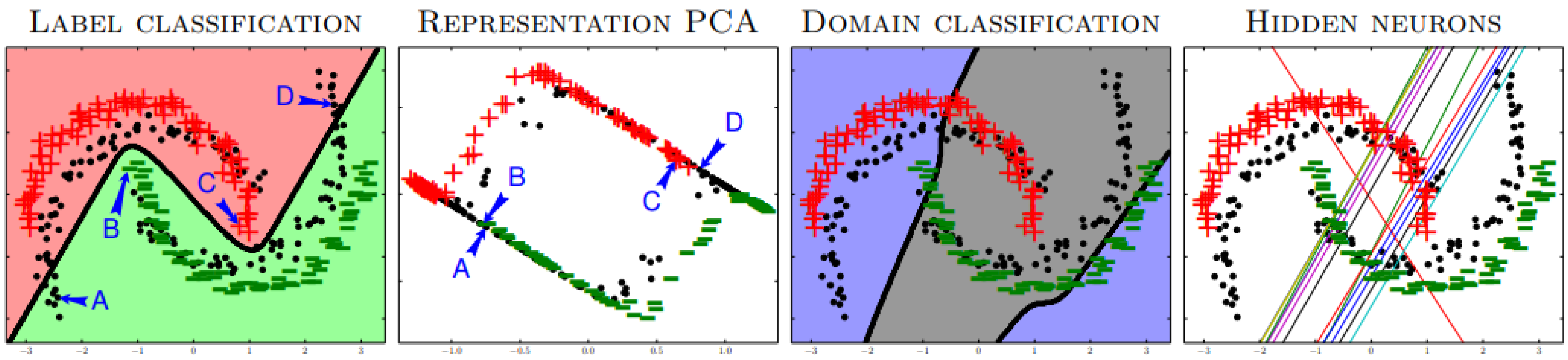
$$(\hat{W}, \hat{V}, \hat{b}, \hat{c}) = \underset{W,V,b,c}{arg\min}\, E(W, V, b, c, \hat{u}, \hat{z}), \tag{1}$$

$$(\hat{u}, \hat{z}) = \underset{\hat{u},\hat{z}}{arg\max}\, E(\hat{W}, \hat{V}, \hat{b}, \hat{c}, u, z). \tag{2}$$

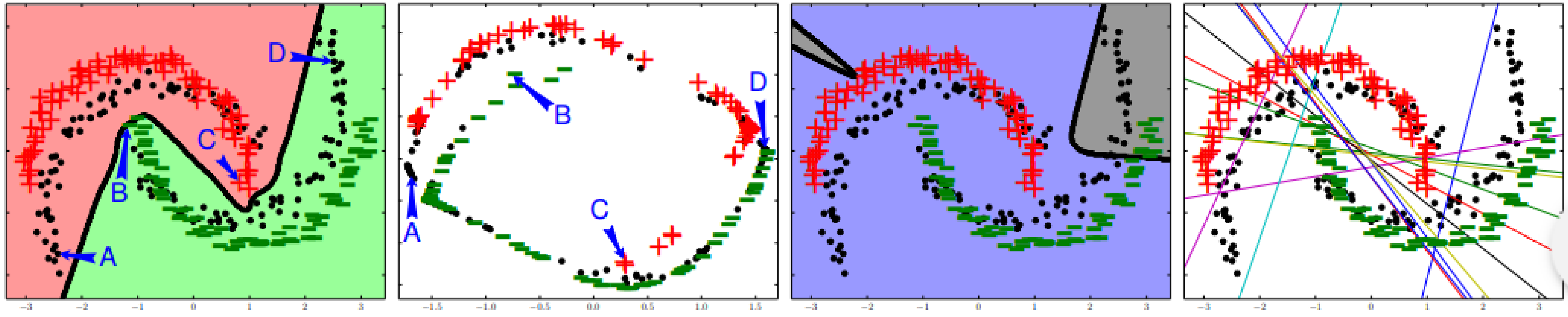
04 evaluation

04. evaluation

Toy problem



(a) Standard NN. For the "domain classification", we use a *non adversarial* domain regressor on the hidden neurons learned by the Standard NN. (This is equivalent to run Algorithm 1, without Lines 22 and 31)



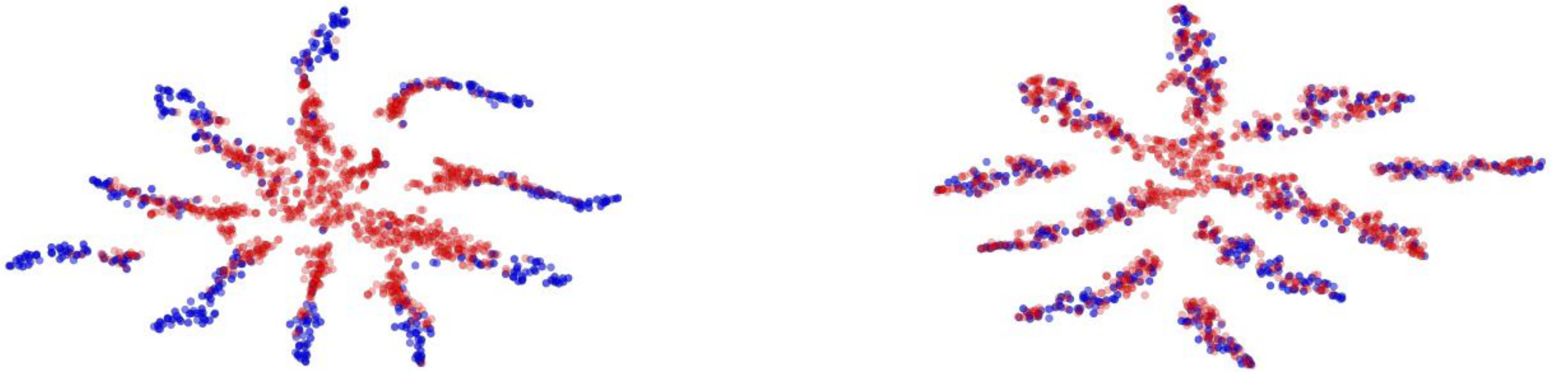
(b) DANN (Algorithm 1)

04. evaluation

MNIST → MNIST-M: top feature extractor layer

(a) Non-adapted (b) Adapted

Syn Numbers → SVHN: last hidden layer of the label predictor



(a) Non-adapted (b) Adapted

04. evaluation

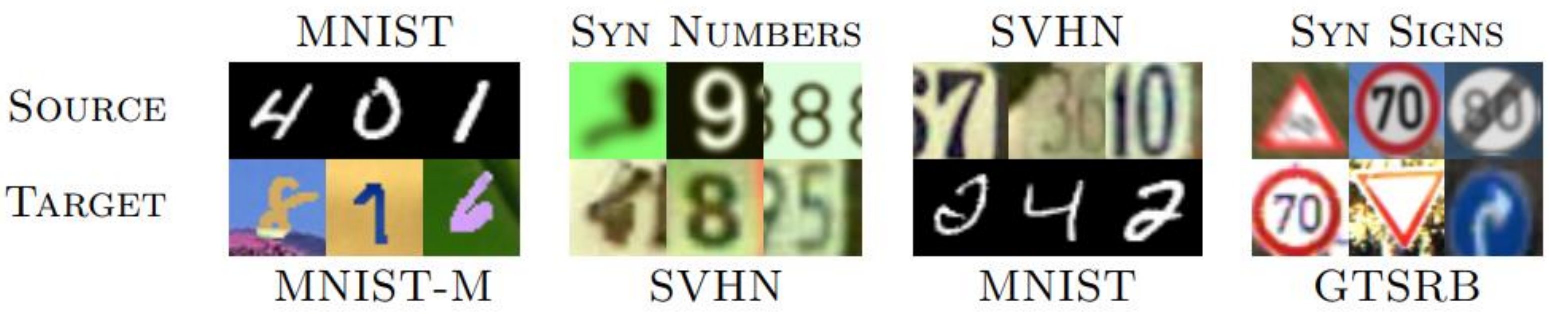


Figure 6: Examples of domain pairs used in the experiments. See Section 5.2.4 for details.

METHOD — SOURCE / TARGET	MNIST / MNIST-M	SYN NUMBERS / SVHN	SVHN / MNIST	SYN SIGNS / GTSRB
SOURCE ONLY	.5225	.8674	.5490	.7900
SA (Fernando et al., 2013)	.5690 (4.1%)	.8644 (−5.5%)	.5932 (9.9%)	.8165 (12.7%)
DANN	**.7666** (52.9%)	**.9109** (79.7%)	**.7385** (42.6%)	**.8865** (46.4%)
TRAIN ON TARGET	.9596	.9220	.9942	.9980

04. evaluation

Source	Target	Original data			mSDA representation		
		DANN	NN	SVM	DANN	NN	SVM
BOOKS	DVD	.784	.790	**.799**	.829	.824	**.830**
BOOKS	ELECTRONICS	.733	.747	**.748**	**.804**	.770	.766
BOOKS	KITCHEN	**.779**	.778	.769	**.843**	.842	.821
DVD	BOOKS	.723	.720	**.743**	.825	.823	**.826**
DVD	ELECTRONICS	**.754**	.732	.748	**.809**	.768	.739
DVD	KITCHEN	**.783**	.778	.746	.849	**.853**	.842
ELECTRONICS	BOOKS	**.713**	.709	.705	**.774**	.770	.762
ELECTRONICS	DVD	**.738**	.733	.726	**.781**	.759	.770
ELECTRONICS	KITCHEN	**.854**	**.854**	.847	.881	**.863**	.847
KITCHEN	BOOKS	**.709**	.708	.707	.718	.721	**.769**
KITCHEN	DVD	**.740**	.739	.736	**.789**	**.789**	.788
KITCHEN	ELECTRONICS	**.843**	.841	.842	.856	.850	**.861**

(a) Classification accuracy on the Amazon reviews data set

Thank you