A Representation Learning Approach for Domain Adaptation

Pascal Germain

INRIA Paris (SIERRA Team)

TAO Seminar

INRIA Saclay / CNRS / Université Paris-Sud

March 1, 2016

Joint Work with...

- François Laviolette
- Mario Marchand
- Hana Ajakan

} Université Laval, Québec, Canada

- Hugo Larochelle

} Université de Sherbrooke, Québec, Canada and Twitter

- Yaroslav Ganin
- Evgeniya Ustinova
- Victor Lempitsky

} Skolkovo Institute of Science and Technology, Moscow Region, Russia

Plan

1. Domain Adaptation Setting
2. Theoretical Foundations
3. Domain-Adversarial Neural Network (DANN)
4. Empirical Results with “Shallow” Networks
5. Empirical Results with “Deep” Networks
6. Conclusion

Example

Book critics
(target)

??? **The end of the series.**
This book was written to provoke those who wanted Adams to continue the trilogy but I loved it. Aurthor setteled down on a bob fearing planet where he has aquired the prestigous...
Read more
Published on Mar 18 2002 by dan

??? **Mostly Harmless is Underrated**
I think most of the reviews for this book downplay it seriously. While the ending is kind of disappointing, the book overall is wonderful.
Read more
Published on Jan 22 2002 by A Big Adams Fan

??? **Please pretend this book was never written.**
I have long been a fan of the Hitchhikers series as they are comic genius. The book Mostly Harmless, however, should never have come about. It is frustration at its peak. Read more
Published on Jan 14 2002 by Paul Norrod

??? **Kinda like horror movies...**
...in that the last one usually isn't all that appealing. I liked it fine, with some of Adams's wit, but it was a bit disappointing. Read more
Published on Nov 4 2001 by Kristopher Vincent

??? **A Terrible End to A Great Series**
The ending for this books was so bad that I vowed never to read another Douglas Adams book. Adams was obviously sick and tired of the series and used this book to kill it off with...
Read more
Published on Oct 17 2001 by David A. Lessnau

Movie critics
(source)

0 **An insult to Douglas Adams' memory**
I agree entirely with "darkgenius'" comments. This movie is a travesty of the book and the TV series; a cutesy version totally lacking in the wit and satire of the original. Read more
Published 5 months ago by John W Beare

1 **Don't Panic!**
If you haven't listened to the BBC radio-play, this isn't bad! Purists, no doubt, will dispute my verdict but the fact of the matter is THGTTG (see title) does have Douglas Adams'...
Read more
Published on Mar 13 2011 by Sid Matheson

1 **On Blu-ray, even better**
I've seen this movie on TV and wanted to add it to my collection. I couldn't find it locally so when I saw it on amazon and on Blu-ray, I picked it up. Read more
Published on April 18 2009 by J. W. Little

0 **An insult to Douglas Adams' memory**
The filmmaker's reverence for Adams' legacy? What kind of rubbish statement is that? As a loyal fan of Douglas Adams for more than a quarter of a century, I was appalled and...
Read more
Published on Aug 22 2006 by Daniel Jolley

Learning algorithm

Classifier

0

Our Domain Adaptation Setting

Classification task

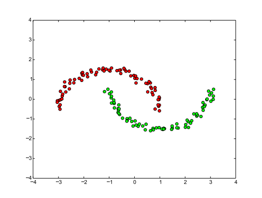
- Input space : $\mathcal{X} \subseteq \mathbb{R}^d$
- Labels : $\mathcal{Y} = \{0, 1, 2, \ldots, L\}$

Two different data distributions

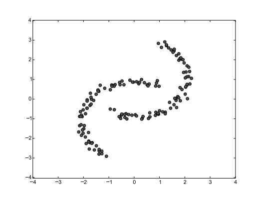
- Source domain : $\mathcal{D}_S$
- Target domain : $\mathcal{D}_T$

A **domain adaptation** learning algorithm is provided with

a **labeled source sample**
$S = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^n \sim (\mathcal{D}_S)^n,$



an **unlabeled target sample**
$T = \{\mathbf{x}_i^t\}_{i=1}^{n'} \sim (\mathcal{D}_T)^{n'}.$



The goal is to build a classifier $\eta : \mathcal{X} \to \mathcal{Y}$ with a low **target risk**

$$R_{\mathcal{D}_T}(\eta) \stackrel{\text{def}}{=} \Pr_{(\mathbf{x}^t, y^t) \sim \mathcal{D}_T} \left[\eta(\mathbf{x}^t) \neq y^t \right].$$

Domain Adaptation

Question

In which context can we adapt from source $\mathcal{D}_S$ to target $\mathcal{D}_T$?

Rough Answer

When domains $\mathcal{D}_S$ and $\mathcal{D}_T$ are ≪similar≫.

Tool

Notion of "distance" $d(\mathcal{D}_S, \mathcal{D}_T)$ between domains.

Two approaches to conceive learning algorithms

1. Find a hypothesis $\eta \in \mathcal{H}$ such that $d_\eta(\mathcal{D}_S, \mathcal{D}_T)$ and $R_{\mathcal{D}_S}(\eta)$ are small.
2. Modify the representation of the examples :
   $\Rightarrow$ Find a function $\mathbf{h}$ such that $d_\mathcal{H}(\mathbf{h}(\mathcal{D}_S), \mathbf{h}(\mathcal{D}_T))$ is small ;
   and a $\eta \in \mathcal{H}$ such that $R_{\mathbf{h}(\mathcal{D}_S)}(\eta)$ is small.

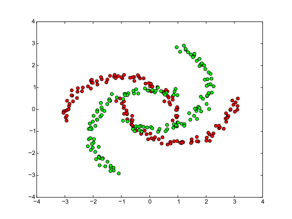
Divergence between source and target domains

Definition (Ben David et al., 2006)

Given two domain distributions $\mathcal{D}_S$ and $\mathcal{D}_T$, and a **hypothesis class** $\mathcal{H}$, the $\mathcal{H}$-**divergence** between $\mathcal{D}_S$ and $\mathcal{D}_T$ is

$$d_{\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) \overset{\text{def}}{=} 2 \sup_{\eta \in \mathcal{H}} \left| \Pr_{\mathbf{x}^s \sim \mathcal{D}_S} \left[\eta(\mathbf{x}^s) = 1\right] + \Pr_{\mathbf{x}^t \sim \mathcal{D}_T} \left[\eta(\mathbf{x}^t) = 0\right] - 1 \right|.$$

The $\mathcal{H}$-**divergence** measures the ability of an hypothesis class $\mathcal{H}$ to **discriminate** between source $\mathcal{D}_S$ and target $\mathcal{D}_T$ distributions.



Bound on the target risk

Theorem (Ben David et al., 2006)

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

$$R_{\mathcal{D}_T}(\eta) \;\leq\; \widehat{R}_S(\eta) + \frac{4}{n}\sqrt{d\log\tfrac{2e\,n}{d} + \log\tfrac{4}{\delta}} + \widehat{d}_{\mathcal{H}}(S,T) + \frac{4}{n^2}\sqrt{d\log\tfrac{2\,n}{d} + \log\tfrac{4}{\delta}} + \beta$$

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

Empirical risk on the source sample :

$$\widehat{R}_S(\eta) \;\stackrel{\text{def}}{=}\; \frac{1}{n}\sum_{i=1}^{n} I[\eta(\mathbf{x}_i^s) \neq y_i^s]\,.$$

Empirical $\mathcal{H}$-divergence :

$$\widehat{d}_{\mathcal{H}}(S,T) \;\stackrel{\text{def}}{=}\; 2\max_{\eta\in\mathcal{H}}\left[\frac{1}{n}\sum_{i=1}^{n} I[\eta(\mathbf{x}_i^s)=1] + \frac{1}{n'}\sum_{i=1}^{n'} I[\eta(\mathbf{x}_i^t)=0] - 1\right].$$

Bound on the target risk

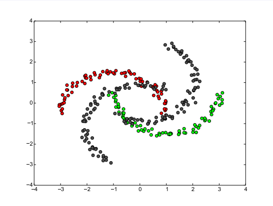
Theorem (Ben David et al., 2006)

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

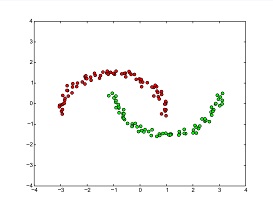
$$R_{\mathcal{D}_T}(\eta) \;\leq\; \widehat{R}_S(\eta) + \frac{4}{n}\sqrt{d \log \frac{2e\,n}{d} + \log \frac{4}{\delta}} + \widehat{d}_{\mathcal{H}}(S,T) + \frac{4}{n^2}\sqrt{d \log \frac{2\,n}{d} + \log \frac{4}{\delta}} + \beta$$

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

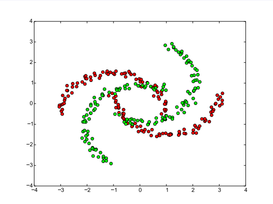
Target risk $R_{\mathcal{D}_T}(\eta)$ is low
if, given S and T,



$\widehat{R}_S(\eta)$ is small,
i.e., $\eta \in \mathcal{H}$ is good on



and $\widehat{d}_{\mathcal{H}}(S,T)$ is small,
i.e., all $\eta' \in \mathcal{H}$ are bad on



Standard Neural Network

Let consider a neural network architecture with one hidden layer

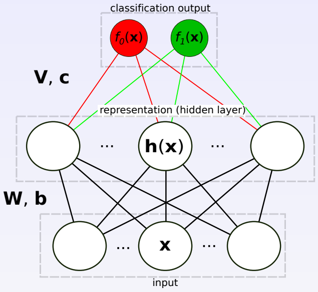
$$\mathbf{h}(\mathbf{x}) \;=\; \mathrm{sigm}(\mathbf{W}\mathbf{x}+\mathbf{b})\,, \quad \text{and} \quad \mathbf{f}(\mathbf{h}(\mathbf{x})) \;=\; \mathrm{softmax}(\mathbf{V}\mathbf{h}(\mathbf{x})+\mathbf{c})\,.$$

$$\min_{\mathbf{W},\mathbf{V},\mathbf{b},\mathbf{c}} \underbrace{\left[\frac{1}{n}\sum_{i=1}^{n} -\log\left(f_{y_i^s}\left(\mathbf{h}(\mathbf{x}_i^s)\right)\right)\right]}_{\text{source loss}}\,.$$

where $f_y(\mathbf{h}(\mathbf{x}))$ denotes the conditional probability that the neural network assigns $\mathbf{x}$ to class y.

Given a source sample $S = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^n \sim (\mathcal{D}_S)^n$,

1. Pick a $\mathbf{x}^s \in S$
2. Update $(\mathbf{V}, \mathbf{c})$ towards $\mathbf{f}(\mathbf{h}(\mathbf{x}^s)) = y^s$
3. Update $(\mathbf{W}, \mathbf{b})$ towards $\mathbf{f}(\mathbf{h}(\mathbf{x}^s)) = y^s$



The hidden layer learns a **representation** $\mathbf{h}(\cdot)$ from which linear hypothesis $\mathbf{f}(\cdot)$ can **classify source examples**.

Domain-Adversarial Neural Network (DANN)

Empirical $\mathcal{H}$-divergence

$$\widehat{d}_{\mathcal{H}}(S, T) \stackrel{\text{def}}{=} 2 \max_{\eta \in \mathcal{H}} \left[\frac{1}{n} \sum_{i=1}^{n} I[\eta(\mathbf{x}_i^s) = 1] + \frac{1}{n'} \sum_{i=1}^{n'} I[\eta(\mathbf{x}_i^t) = 0] - 1 \right].$$

Given a representation output by the hidden layer $\mathbf{h}(\cdot)$, we estimate the $\mathcal{H}$-divergence by

$$\widehat{d}_{\mathcal{H}}\Big(\mathbf{h}(S), \mathbf{h}(T)\Big) \approx 2 \max_{\mathbf{u}, d} \left[\frac{1}{n} \sum_{i=1}^{n} \log\big(o(\mathbf{h}(\mathbf{x}_i^s))\big) + \frac{1}{n'} \sum_{i=1}^{n'} \log\big(1 - o(\mathbf{h}(\mathbf{x}_i^t))\big) - 1 \right].$$

where $o(\mathbf{h}(\mathbf{x}))$ is a logistic regressor that "tries" to detect if $\mathbf{x}$ is from the **source domain** $\left(o(\mathbf{h}(\mathbf{x})) > \frac{1}{2}\right)$ or **target domain** $\left(o(\mathbf{h}(\mathbf{x})) < \frac{1}{2}\right)$:

$$o(\mathbf{h}(\mathbf{x})) \stackrel{\text{def}}{=} \mathrm{sigm}(\mathbf{u}^\top \mathbf{h}(\mathbf{x}) + d).$$

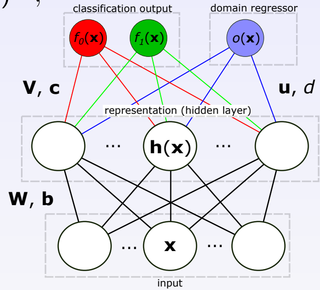
Domain-Adversarial Neural Network (DANN)

$$\min_{\mathbf{W},\mathbf{V},\mathbf{b},\mathbf{c}} \Bigg[\underbrace{\frac{1}{n}\sum_{i=1}^{n} -\log\Big(f_{y_i^s}(\mathbf{h}(\mathbf{x}_i^s))\Big)}_{\text{source loss}} + \lambda \underbrace{\max_{\mathbf{u},d} \Bigg(\frac{1}{n}\sum_{i=1}^{n}\log\big(o(\mathbf{h}(\mathbf{x}_i^s))\big) + \frac{1}{n'}\sum_{i=1}^{n'}\log\big(1-o(\mathbf{h}(\mathbf{x}_i^t))\big)\Bigg)}_{\text{adaptation regularizer}} \Bigg],$$

where $\lambda > 0$ weights the domain adaptation regularization term.

Given a source sample $S = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n'} \sim (\mathcal{D}_S)^{n'}$,
and a target sample $T = \{(\mathbf{x}_i^t)\}_{i=1}^{n} \sim (\mathcal{D}_T)^{n}$,

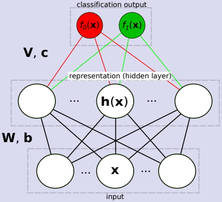
1. Pick a $\mathbf{x}^s \in S$ and $\mathbf{x}^t \in T$
2. Update $(\mathbf{V}, \mathbf{c})$ towards $\mathbf{f}(\mathbf{h}(\mathbf{x}^s)) = y^s$
3. Update $(\mathbf{W}, \mathbf{b})$ towards $\mathbf{f}(\mathbf{h}(\mathbf{x}^s)) = y^s$
4. Update $(\mathbf{u}, d)$ towards $o(\mathbf{h}(\mathbf{x}^s)) = 1$ and $o(\mathbf{h}(\mathbf{x}^t)) = 0$
5. Update $(\mathbf{W}, \mathbf{b})$ towards $o(\mathbf{h}(\mathbf{x}^s)) = 0$ and $o(\mathbf{h}(\mathbf{x}^t)) = 1$



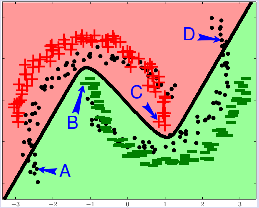
DANN finds a representation $\mathbf{h}(\cdot)$ that are good on S; but unable to discriminate between S and T.

Toy Dataset

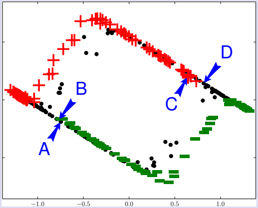
Standard Neural Network (NN)



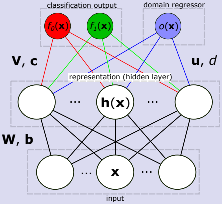
Classification output : $\mathbf{f}(\mathbf{h}(\mathbf{x}))$

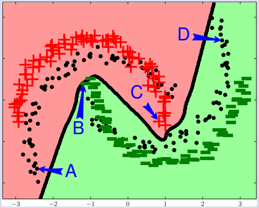


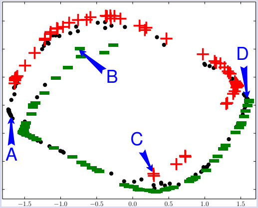
Representation PCA



Domain-Adversarial Neural Networks (DANN)

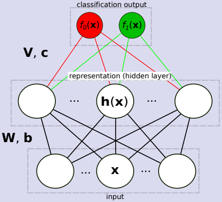


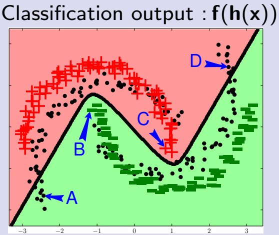


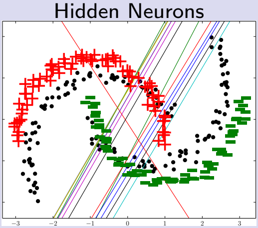


Toy Dataset

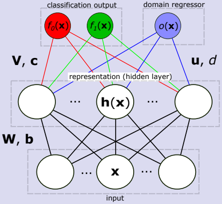
Standard Neural Network (NN)

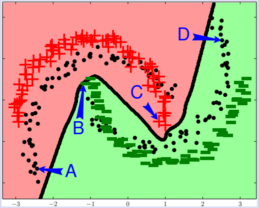


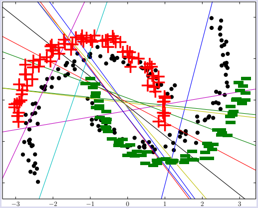




Domain-Adversarial Neural Networks (DANN)



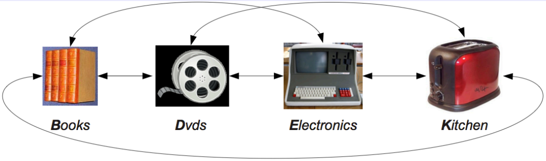




Amazon Reviews

Input : product review (bag of words)

Output : positive or negative rating.



Model Selection by *Reverse Validation* (inspired by Zhong et al., 2010)

For each tuple of hyperparameters :

- Split S, T into **training sets** S', T' and **validation sets** S_V, T_V.
- Learn classifier η on (labeled) source S' and (unlabeled) target T'.
- Learn **reverse classifier** η_r on **self-labeled** $\{(\mathbf{x}, \eta(\mathbf{x}))\}_{\mathbf{x} \in T'}$ as source and unlabeled part of S' as target.
- Compute the **reverse validation risk** $\widehat{R}_{S_V}(\eta_r)$.

Amazon Reviews

Source	Target	DANN	NN	SVM
books	dvd	.784	.790	**.799**
books	electronics	.733	.747	**.748**
books	kitchen	**.779**	.778	.769
dvd	books	.723	.720	**.743**
dvd	electronics	**.754**	.732	.748
dvd	kitchen	**.783**	.778	.746
electronics	books	**.713**	.709	.705
electronics	dvd	**.738**	.733	.726
electronics	kitchen	**.854**	**.854**	.847
kitchen	books	**.709**	.708	.707
kitchen	dvd	**.740**	.739	.736
kitchen	electronics	**.843**	.841	.842

Marginalized Stacked Denoising Autoencoders (mSDA)

Question

Does DANN can be combined with other representation learning techniques for domain adaptation ?

The autoencoders mSDA (Chen et al. 2012) provides a new common representation for source and target (unsupervised)

With **mSDA+SVM**, Chen et al. (2012) obtained *state-of-the-art* results on Amazon Reviews :
– Train a linear SVM on mSDA source representations.

We try **mSDA+DANN** :
– Train DANN on source representations and target representations.

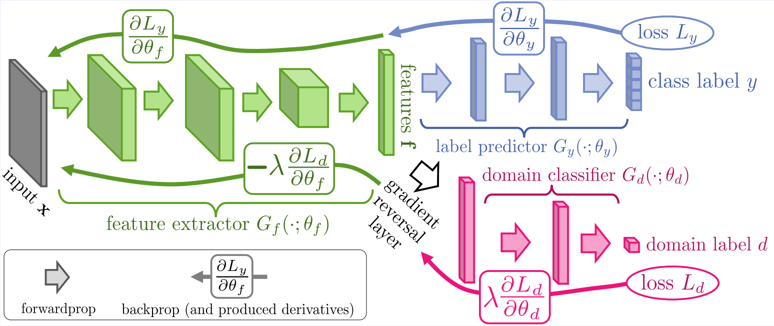
Amazon Reviews

		Original data			mSDA representation		
Source	Target	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**

Deeper and deeper...

To appear in JMLR : **Domain-Adversarial Neural Networks.**

by Ganin, Ustinova, Ajakan, Germain, Larochelle, Laviolette, Marchand and Lempitsky

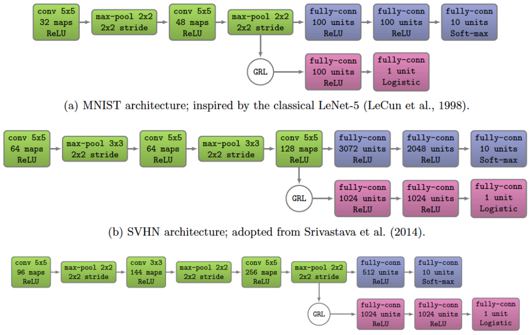


Preprint on arXiv : `http://arxiv.org/abs/1505.07818`

Gradient Reversal Layer

Implemented in `Caffe` Deep Learning Package (Jia et al. 2014) :

$$\mathcal{R}(\mathbf{x}) = \mathbf{x}\,, \qquad \frac{d\mathcal{R}}{d\mathbf{x}} = -\mathbf{I}\,.$$



(a) MNIST architecture; inspired by the classical LeNet-5 (LeCun et al., 1998).

(b) SVHN architecture; adopted from Srivastava et al. (2014).

(c) GTSRB architecture; we used the single-CNN baseline from Cireşan et al. (2012) as our starting point.

Digits and Traffic Signs Recognition

	MNIST	Syn Numbers	SVHN	Syn Signs
Source / Target				
	MNIST-M	SVHN	MNIST	GTSRB

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	.8644	.5932	.8165
DANN		**.7666**	**.9109**	**.7385**	**.8865**
Train on target		.9596	.9220	.9942	.9980

Office Dataset

Images from three domains : Amazon, DSLR camera, and Webcam

31 labels : chair, cup, laptop, keyboard, ...

Method (Source / Target)	Amazon / Webcam	DSLR / Webcam	Webcam / DSLR
GFK(PLS, PCA) (Gong et al. 2012)	.197	.497	.6631
SA (Fernando et al., 2013)	.450	.648	.699
DLID (Chopra et al., 2013)	.519	.782	.899
DDC (Tzeng et al., 2014)	.618	.950	.985
DAN (Long and Wang, 2015)	.685	.960	.990
Source only	.642	.961	.978
DANN	**.730**	**.964**	**.992**

Summary

We learn a new representation that is

1. accurate on the source domain ; but
2. unable to discriminate between source and target domains.

Our method is :

- Directly based on the seminal theory of domain adaptation of Ben-David et al. (2006).
- Easy to implement in any neural network architectures.
- Achieving state-of-the-art results on several benchmarks.

Future work :

- Multi source / multi target domain adaptation.
- Other network architectures (beyond classification).