DASA: Domain Adaptation in Stacked Autoencoders using Systematic Dropout

Abhijit Guha Roy abhijit.guharoy@iitkgp.ac.in
Debdoot Sheet debdoot@ee.iitkgp.ernet.in
Department of Electrical Engineering, Indian Institute of Technology Kharagpur

Abstract

Domain adaptation deals with adapting behaviour of machine learning based systems trained using samples in source domain to their deployment in target domain where the statistics of samples in both domains are dissimilar. The task of directly training or adapting a learner in the target domain is challenged by lack of abundant labeled samples. In this paper we propose a technique for domain adaptation in stacked autoencoder (SAE) based deep neural networks (DNN) performed in two stages: (i) unsupervised weight adaptation using systematic dropouts in mini-batch training, (ii) supervised fine-tuning with limited number of labeled samples in target domain. We experimentally evaluate performance in the problem of retinal vessel segmentation where the SAE-DNN is trained using large number of labeled samples in the source domain (DRIVE dataset) and adapted using less number of labeled samples in target domain (STARE dataset).

1. Introduction

The under-performance of learning based systems during deployment stage can be attributed to dissimilarity in distribution of samples between the *source domain* on which the system is initially trained and the *target domain* on which it is deployed. Transfer learning is an active field of research which deals with transfer of knowledge between the *source* and *target domains* for addressing this challenge and enhancing performance of learning based systems (Pan & Yang, 2010), when it is challenging to train a system exclusively in the *target domain* due to unavailability of sufficient labeled samples. While domain adaptation (DA) have been primarily developed for simple reasoning and shallow network architectures, there exist few techniques for adapting deep networks with complex reasoning. In this paper we propose a systematic dropout based technique for adapting a stacked autoencoder (SAE) based deep neural network (DNN) for the purpose of vessel segmentation in retinal images (Abràmoff et al., 2010). Here the SAE-DNN is initially trained using ample number of samples in the *source domain* (DRIVE dataset) to evaluate efficacy of DA during deployment in the *target domain* (STARE dataset) where an insufficient number of labeled samples are available for reliable training exclusively in the *target domain*.

Related Work: Stacked AE (SAE) is created by hierarchically connecting hidden layers to learn hierarchical embedding in compressed representations. An SAE-DNN consists of encoding layers of an SAE followed by a target prediction layer for the purpose of regression or classification. With increase in demand for DA in SAE-DNNs different techniques have been proposed including marginalized training (Minmin et al., 2012), via graph regularization (Peng et al., 2013) and structured dropouts (Yang & Eisensteinl, 2014).

Challenge: The challenge of DA is to retain nodes common across *source* and *target domains*, while adapting the domain specific nodes using fewer number of labeled samples. Earlier methods (Minmin et al., 2012; Peng et al., 2013; Yang & Eisensteinl, 2014) are primarily challenged by their inability to re-tune nodes specific to the *source domain* to nodes specific for *target domain* for achieving desired performance, while they are able to only retain nodes or a thinned network which encode domain invariant hierarchical embeddings.

Approach: Here we propose a method for DA in SAE (DASA) using systematic dropout. The two stage

Appearing in Proceedings of the 2^{nd} Indian Workshop on Machine Learning, IIT Kanpur, India, 2016. Copyright 2016 by the author(s).

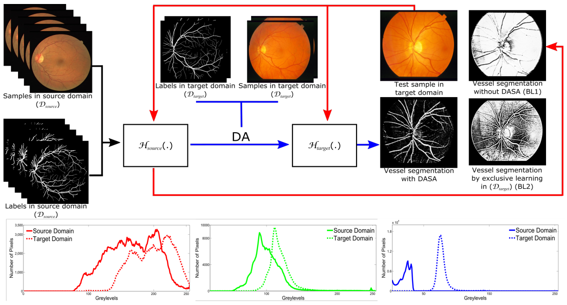

Samples in source domain ($\mathcal{D}_{source}$)
Labels in source domain ($\mathcal{D}_{source}$)
Labels in target domain ($\mathcal{D}_{target}$)
Samples in target domain ($\mathcal{D}_{target}$)
$\mathcal{H}_{source}(.)$
DA
$\mathcal{H}_{target}(.)$
Test sample in target domain
Vessel segmentation without DASA (BL1)
Vessel segmentation with DASA
Vessel segmentation by exclusive learning in ($\mathcal{D}_{target}$) (BL2)
Source Domain
Target Domain
Number of Pixels
Greylevels
0
1,000
2,000
3,000
3,500
100
200
250
Source Domain
Target Domain
Number of Pixels
Greylevels
0
4000
8000
10000
100
200
250
x 10^4
Source Domain
Target Domain
Number of Pixels
Greylevels
0
0.6
1.2
1.8
50
150
250

Figure 1. Overview of the process of DASA. The shifts in distribution of color statistics across samples in $\mathcal{D}_{source}$ and $\mathcal{D}_{target}$ are also illustrated.

method adapts a SAE-DNN trained in the *source domain* following (i) unsupervised weight adaptation using systematic dropouts in mini-batch training with abundant unlabeled samples in *target domain*, and (ii) supervised fine-tuning with limited number of labeled samples in *target domain*. The systematic dropout per mini-batch is introduced only in the representation encoding (hidden) layers and is guided by a saliency map defined by response of the neurons in the mini-batch under consideration. Error backpropagation and weight updates are however across all nodes and not only restricted to the post dropout activated nodes, contrary to classical randomized dropout approaches (Srivastava, et al., 2014). Thus having different dropout nodes across different mini-batches and weight updates across all nodes in the network, ascertains refinement of domain specific hierarchical embeddings while preserving domain invariant ones.

2. Problem Statement

Let us consider a retinal image represented in the RGB color space as $\mathcal{I}$, such that the pixel location $\mathbf{x} \in \mathcal{I}$ has the color vector $\mathbf{c}(\mathbf{x}) = \{r(\mathbf{x}), g(\mathbf{x}), b(\mathbf{x})\}$. $N(\mathbf{x})$ is a neighborhood of pixels centered at $\mathbf{x}$. The task of retinal vessel segmentation can be formally defined as assigning a class label $y \in \{\text{vessel}, \text{background}\}$ using a hypothesis model $\mathcal{H}(\mathcal{I}, \mathbf{x}, N(\mathbf{x}); \{\mathcal{I}\}_{\text{train}})$. When the statistics of samples in $\mathcal{I}$ is significantly dissimilar from $\mathcal{I}_{\text{train}}$, the performance of $\mathcal{H}(\cdot)$ is severely affected. Generally $\{\mathcal{I}\}_{\text{train}}$ is referred to as the *source domain* and $\mathcal{I}$ or the set of samples used during deployment belong to the *target domain*. The hypothesis $\mathcal{H}(\cdot)$ which optimally defines *source* and *target domains* are also referred to as $\mathcal{H}_{source}$ and $\mathcal{H}_{target}$. DA is formally defined as a transformation $\mathcal{H}_{source}$ to $\mathcal{H}_{target}$ as detailed in Fig. 1.

3. Exposition to the Solution

Let us consider the source domain as $\mathcal{D}_{source}$ with abundant labeled samples to train an SAE-DNN ($\mathcal{H}_{source}$) for the task of retinal vessel segmentation, and a target domain $\mathcal{D}_{target}$ with limited number of labeled samples and ample unlabeled samples, insufficient to learn $\mathcal{H}_{target}$ reliably as illustrated in Fig. 1. $\mathcal{D}_{source}$ and $\mathcal{D}_{target}$ are closely related, but exhibiting distribution shifts between samples of the *source* and *target domains*, thus resulting in under-performance of $\mathcal{H}_{source}$ in $\mathcal{D}_{target}$ as also illustrated in Fig. 1. The technique of generating $\mathcal{H}_{source}$ using $\mathcal{D}_{source}$, and subsequently adapting $\mathcal{H}_{source}$ to $\mathcal{H}_{target}$ via systematic dropout using $\mathcal{D}_{target}$ is explained in the following sections.

3.1. SAE-DNN learning in the source domain

AE is a single layer neural network that encodes the cardinal representations of a pattern $\mathbf{p} = \{p_k\}$ onto a transformed spaces $\mathbf{y} = \{y_j\}$ with $\mathbf{w} = \{w_{jk}\}$ denoting the connection weights between neurons, such that

$$\mathbf{y} = f_{\text{NL}}([\mathbf{w} \quad \mathbf{b}].[\mathbf{p} \ ; \ 1]) \tag{1}$$

where the cardinality of y denoted as $|\mathbf{y}| = J \times 1$, $|\mathbf{p}| = K \times 1$, $|\mathbf{w}| = J \times K$, and $\mathbf{b}$ is termed as the bias connection with $|\mathbf{b}| = J \times 1$. We choose $f_{\text{NL}}(\cdot)$ to be a sigmoid function defined as $f_{\text{NL}}(z) = 1/(1+\exp(-z))$. AE is characteristic with another associated function which is generally termed as the decoder unit such that

$$\hat{\mathbf{p}} = f_{\text{NL}}([\mathbf{w}' \quad \mathbf{b}'].[\mathbf{y} \ ; \ 1]) \tag{2}$$

where $|\hat{\mathbf{p}}| = |\mathbf{p}| = K \times 1$, $|\mathbf{w}'| = K \times J$ and $|\mathbf{b}'| = K \times 1$. When $|\mathbf{y}| << |\{\mathbf{p}_n\}|$, this network acts to store compressed representations of the pattern $\mathbf{p}$ encoded through the weights $\mathbf{W} = \{\mathbf{w}, \mathbf{b}, \mathbf{w}', \mathbf{b}'\}$. However the values of elements of these weight matrices are achieved through learning, and without the need of having class labels of the patterns $\mathbf{p}$, it follows unsupervised learning using some optimization algorithm, viz. stochastic gradient descent.

$$\{\mathbf{w}, \mathbf{b}, \mathbf{w}', \mathbf{b}'\} = \arg\min_{\mathbf{w},\mathbf{b},\mathbf{w}',\mathbf{b}'} (J(\mathbf{W})) \tag{3}$$

such that $J(\cdot)$ is the cost function used for optimization over all available patterns $\mathbf{p}_n \in \{\mathbf{p}(\mathbf{x}), \mathbf{x} \in \mathcal{I}\}$

$$J(\mathbf{W}) = \sum_n \|\mathbf{p}_n - \hat{\mathbf{p}}_n\| + \beta|\rho - \hat{\rho}_n| \tag{4}$$

where β regularizes the sparsity penalty, ρ is the imposed sparsity and $\hat{\rho}_n$ is the sparsity observed with the n^{th} pattern in the mini-batch.

The SAE-DNN consists of $L = 2$ cascade connected AEs followed by a softmax regression layer known as the target layer with $\mathbf{t}$ as its output. The number of output nodes in this layer is equal to the number of class labels such that $|\mathbf{t}| = |\Omega|$ and the complete DNN is represented as

$$\begin{aligned}\mathbf{t} \;\; = \;\; & f_{\text{NL}}\left([\mathbf{w}_3 \;\; \mathbf{b}_3].\left[f_{\text{NL}}\left([\mathbf{w}_2 \;\;\; \mathbf{b}_2].\left[f_{\text{NL}}\left([\mathbf{w}_1 \;\; \mathbf{b}_1].\right.\right.\right.\right.\right. \\ & \left.\left.\left.\left.\left.[\mathbf{p} \;\; 1]^T\right) \;\; 1\right]^T\right) \;\; 1\right]^T\right)\end{aligned} \tag{5}$$

where $\{\mathbf{W}_1 = \{\mathbf{w}_1, \mathbf{b}_1\}, \mathbf{W}_2 = \{\mathbf{w}_2, \mathbf{b}_2\}\}$ are the pretrained weights of the network obtained from the earlier section. The weights $\mathbf{W}_3 = \{\mathbf{w}_3, \mathbf{b}_3\}$ are randomly initialized and convergence of the DNN is achieved through supervised learning with the cost function

$$J(\mathbf{W}) = \sum_m \|\mathbf{t}_m - \Omega_m\| \tag{6}$$

during which all the weights $\mathbf{W} = \{\mathbf{W}_1 = \{\mathbf{w}_1, \mathbf{b}_1\}, \mathbf{W}_2 = \{\mathbf{w}_2, \mathbf{b}_2\}, \mathbf{W}_3 = \{\mathbf{w}_3, \mathbf{b}_3\}\}$ are updated to completely tune the DNN.

3.2. SAE-DNN adaptation in the target domain

Unupervised adaptation of SAE weights using systematic dropouts: The first stage of DA utilizes abundant unlabeled samples available in *target domain* to retain nodes which encode domain invariant hierarchical embeddings while re-tuning the nodes specific in *source domain* to those specific in *target domain*. We follow the concept of systematic node drop-outs during training (Srivastava, et al., 2014). The number of layers and number of nodes in the SAE-DNN however remains unchanged during domain adaptation. Fig. 2 illustrates the concept.

Weights connecting each of the hidden layers is imported from the SAE-DNN trained in $\mathcal{D}_{source}$ are updated in this stage using an auto-encoding mechanism. When each mini-batch in $\mathcal{D}_{target}$ is fed to this AE with one of the hidden layers from the SAE-DNN; some of the nodes in the hidden layer exhibit high response with most of the samples in the mini-batch, while some of the nodes exhibit low response. The nodes which exhibit high-response in the mini-batch are representative of domain invariant embeddings which need to be preserved, while the ones which exhibit low-response are specific to $\mathcal{D}_{source}$ and need to be adapted to $\mathcal{D}_{target}$. We set a free parameter $\tau \in [0, 1]$ defined as the transfer coefficient used for defining saliency metric ($\{s^l_j\} \in \mathbf{s}$) for the j^{th} node in the l^{th} layer as $s^l_j = 1$ if $y^l_j \geq \tau$ else $s^l_j = 0$ otherwise. Here $y^l_j \in \mathbf{y}$ as in (1), and we redefine (2) while preserving (4) and the original learning rules.

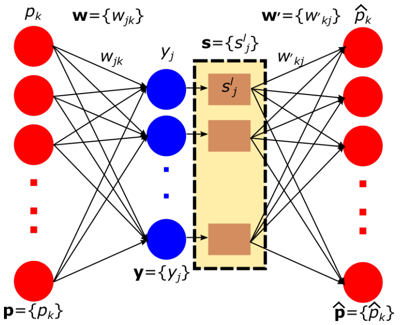


Figure 2. Illustration of the technique for unsupervised adaptation of SAE weights using systematic dropout.

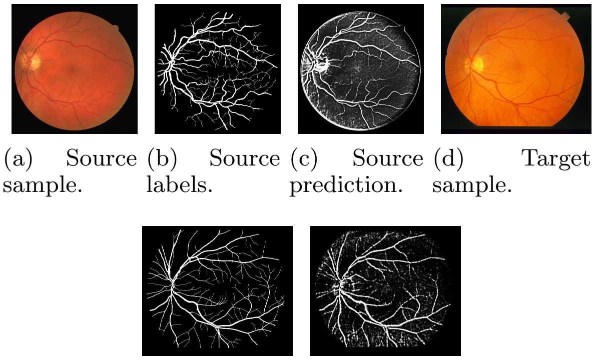
$$\hat{\mathbf{p}} = f_{\text{NL}}([\mathbf{w}' \;\; \mathbf{b}'].[\mathbf{y}.\mathbf{s} \; ; \; 1]) \tag{7}$$

Supervised fine tuning with limited number of labeled samples: The SAE-DNN with weight embeddings updated in the previous stage is now fine tuned using limited number of labeled samples in $\mathcal{D}_{target}$ following procedures in (5) and (6).

4. Experiments

SAE-DNN architecture: We have a two-layered architecture with $L = 2$ where AE$_1$ consists of 400 nodes and AE$_2$ consists of 100 nodes. The number of nodes at input is $15 \times 15 \times 3$ corresponding to the input with patch size of 15×15 in the color retinal images in RGB space. AEs are unsupervised pre-trained with learning rate of 0.3, over 50 epochs, $\beta = 0.1$ and $\rho = 0.04$. Supervised weight refinement of the SAE-DNN is performed with a learning rate of 0.1 over 200 epochs. The training configuration of learning rate and epochs were same in the *source* and *target* domains, with $\tau = 0.1$.

Source and target domains: The SAE-DNN is trained in $\mathcal{D}_{source}$ using 4% of the available patches from the 20 images in the training set in DRIVE dataset. DA is performed in $\mathcal{D}_{target}$ using (i) 4% of the available patches in 10 unlabeled images for unsupervised adaptation using systematic dropout and (ii) 4% of the available patches in 3 labeled images for fine tuning.



(a) Source sample. (b) Source labels. (c) Source prediction. (d) Target sample.

(e) Target labels. (f) Targe prediction with DA.

Figure 3. Performance of the vessel segmentation with (a-c) SAE-DNN on $\mathcal{D}_{source}$ (DRIVE), (d-f) DASA on $\mathcal{D}_{source}$ (STARE)

Baselines and comparison: We have experimented with the following SAE-DNN baseline (BL) configurations and training mechanisms for comparatively evaluating efficacy of DA: **BL1:** SAE-DNN trained in *source domain* and deployed in *target domain* without DA; **BL2:** SAE-DNN trained in *target domain* with limited samples and deployed in *target domain*.

5. Results and Discussion

The results comparing performance of the SAE-DNN are reported in terms of *logloss* and area under ROC curve as presented in Table 1, and DA aspects in Fig. 3.

	logloss	Area under ROC
Source domain	0.19 ± 0.05	0.90 ± 0.02
BL1	0.40 ± 0.31	0.86 ± 0.03
BL2	0.39 ± 0.68	0.87 ± 0.01
DASA	0.18 ± 0.02	0.92 ± 0.02

Table 1. Comparison of Performance with the baselines

Importance of transfer coefficient: The transfer coefficient τ drives quantum of knowledge transfer from the *source* to *target domains* by deciding on the amount of nodes to be dropped while adapting with ample unlabeled samples. This makes it a critical parameter to be set in DASA to avoid over-fitting and negative transfers as illustrated in Table. 2 where optimal $\tau = 0.1$. Generally $\tau \in [0, 1]$ with $\tau \to 0$ being associated with large margin transfer between domains when they are not very dissimilar, and $\tau \to 1$ being associated otherwise.

τ	0	0.05	0.1	0.15	0.2
logloss	0.39	0.24	0.18	0.21	0.32

Table 2. Variation of *logloss* in DA with variation of τ

6. Conclusion

We have presented DASA, a method for knowledge transfer in an SAE-DNN trained with ample labeled samples in *source domain* for application in *target domain* with less number of labeled samples insufficient to directly train to solve the task in hand. DASA is based on systematic droupout for adaptation being able to utilize (i) ample unlabeled samples and (ii) limited amount of labeled samples in *target domain*. We experimentally provide its efficacy to solve the problem of vessel segmentation when trained with DRIVE dataset (source domain) and adapted to deploy on STARE dataset (target domain). It is observed that DASA outperforms the different baselines. While systematic droupt is demonstrated on an SAE-DNN in DASA, it can be extended to other deep architectures as well.

Dual Submission: This paper has been presented at 3rd Asian Conference on Pattern Recognition, $3-6$ Nov. 2015 at Kuala Lumpur, Malaysia.

References

Minmin , C., Zhixiang, X., Kilian, W., and Fei,S. *Marginalized Denoising Autoencoders for Domain Adaptation*, *Proc. Int. Conf. Mach. Learn.*, pp. 767–774, 2012,

Pan, S.J. and Yang, Q. *A survey on transfer learning*, *IEEE Trans. Knowledge., Data Engg.*, vol. 22, no. 10, pp. 1345–1359, 2010.

Peng, Y., Wang, S. and Lu,B-L. *Marginalized Denoising Autoencoder via Graph Regularization for Domain Adaptation*, *Proc. Neural Inf. Process. Sys.*, pp. 156–163, 2013

Srivastava,, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov,R. *Dropout: A simple way to prevent neural networks from overfitting*, *J. Mach. Learn. Res.*, vol. 15, no. 1,pp. 1929–1958, 2014

Yang, Y. and Eisenstein, J. *Fast easy unsupervised domain adaptation with marginalized structured dropout*, *Proc. Assoc., Comput. Linguistics*, 2014.

Abràmoff, M.D., Garvin, M.K. and Sonka,M. *Retinal imaging and image analysis*, *IEEE Rev. Biomed. Engg.*, vol. 3, pp. 169–208, 2010