

Weakly-Shared Deep Transfer Networks for Heterogeneous-Domain Knowledge Propagation

Xiangbo Shu
Nanjing University of Science
and Technology, P.R. China
shuxb104@gmail.com

Jinhui Tang*
Nanjing University of Science
and Technology, P.R. China
jinhuitang@njust.edu.cn

Guo-Jun Qi
University of Central Florida
USA
guojun.qi@ucf.edu

Jingdong Wang
Microsoft Research
P.R. China
jingdw@microsoft.com

ABSTRACT

In recent years, deep networks have been successfully applied to model image concepts and achieved competitive performance on many data sets. In spite of impressive performance, the conventional deep networks can be subjected to the decayed performance if we have insufficient training examples. This problem becomes extremely severe for deep networks with powerful representation structure, making them prone to over fitting by capturing nonessential or noisy information in a small data set. In this paper, to address this challenge, we will develop a novel deep network structure, capable of transferring labeling information across heterogeneous domains, especially from text domain to image domain. This *weakly-shared Deep Transfer Networks* (DTNs) can adequately mitigate the problem of insufficient image training data by bringing in rich labels from the text domain.

Specifically, we present a novel architecture of DTNs to translate cross-domain information from text to image. To share the labels between two domains, we will build multiple weakly shared layers of features. It allows to represent both shared inter-domain features and domain-specific features, making this structure more flexible and powerful in capturing complex data of different domains jointly than the strongly shared layers. Experiments on real world dataset will show its competitive performance as compared with the other state-of-the-art methods.

Categories and Subject Descriptors

I.4.7 [Learning]: Parameter learning, Concept learning, Knowledge acquisition; H.2.5 [Database Applications]: Image Representation

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

MM'15, October 26–30, 2015, Brisbane, Australia.

© 2015 ACM. ISBN 978-1-4503-3459-4/15/10 ...\$15.00.

DOI: <http://dx.doi.org/10.1145/2733373.2806216>.

Keywords

Heterogeneous-domain knowledge propagation, cross-domain label transfer, deep transfer network, image classification

1. INTRODUCTION

Deep networks [11] formed by multiple layers of non-linear transformations can simulate the perception of human brain to represent high-level abstractions. Existing deep network architectures include *Deep Belief Networks* (DBNs) [11], *Convolutional Neural Networks* (CNNs) [11], *Stacked Auto-Encoders* (SAEs) [3], *Deep Boltzmann Machines* (DBMs) [27], as well as many of their variants. These deep networks have achieved a tremendous success in many areas, such as image classification [16], feature learning [25], collaborative filtering [28], face verification [32], etc.

However, these general deep networks are so powerful that they are prone to overfitting into minor and often noisy variations in a relatively small size of data set. This problem becomes extremely severe when we attempt to build deep networks to model images and their concepts without sufficient amount of data. Several strategies have been proposed to avoid the overfitting by reducing the unnecessary network complexity, e.g., dropout technique for randomly omitting some hidden units with a constant probability [12], corrupted input [33], greedy layer-wise training with sparse filtering [17, 19], alternative convolution and pooling layers as previous layers [16]. Although these strategies have achieved better generalization performance, they often sacrifice the representation power of deep networks to varying degrees.

Instead of trading the representation power of deep networks for reduced overfitting risk, we will consider another way to address this challenge of insufficient training data by bringing in labeling information from other modality, namely, the text document. We aspire to answer the question – How can the cross-modal data help model image concepts? We find it beneficial to explore the text information for at least twofold reasons: 1) the word features of text data are more directly related to semantic concepts inherent in class labels and interpret its concept intuitively; 2) abundant labeled text documents are more widely available on the websites. They inspire the heterogeneous transfer of discriminative knowledge from web text space (i.e., source domain) to image space (i.e., target domain). By transferring the feature representation and labeling information from text space, we will learn a semantic-intensive image feature representation directly related to image concepts, which can greatly improve the performance of image classification tasks. In other words, the rich information transferred from text can help

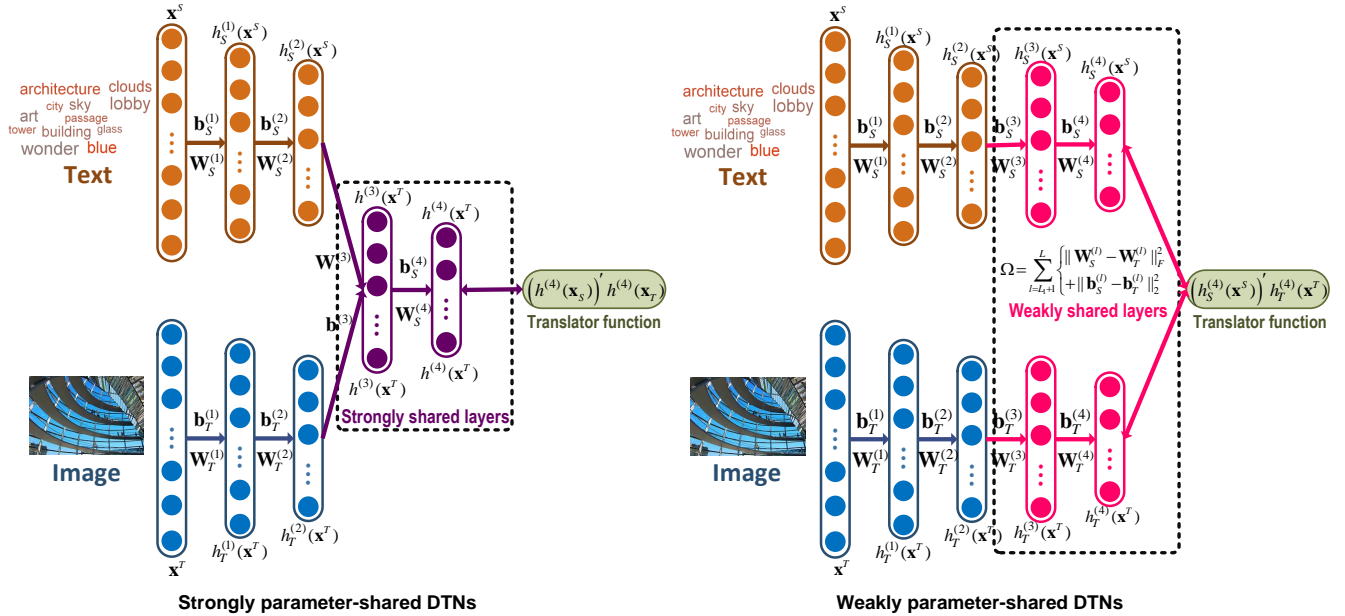


Figure 1: Architecture of *weakly-shared Deep Transfer Networks* (DTNs) with strongly parameter-shared layers (left figure) and weakly parameter-shared layers (right figure). In the left figure, DTNs use the parameter-shared layers (Layer 3 and 4) to model a set of shared features for two domains. On the contrary, in the right figure, weakly parameter-shared DTNs relax this constraint, allowing to use separate series of layers to model the features for two domains. These layers are weakly parameter-shared by imposing a regularizer that decides the extent to which they should be shared. This structure is much flexible, enabling to represent both domain-specific features and the shared features across domains.

train complex image deep networks even with little supervised image data. This is contrary to the popular strategy of trading the representation power of the deep networks for a better generalization performance in the literature. In this paper, we will present how to use transferred cross-modal information to train a powerful deep network without sacrificing structural richness, which will yield much competitive performance on image classification tasks.

To this end, we propose a novel deep network structure that hierarchically learns to transfer the semantic knowledge from web texts to images [23], namely *weakly-shared Deep Transfer Networks* (DTNs) in this paper. As a hierarchically non-linear model, DTNs differ from existing shallow transfer learning algorithms in the literature [37, 8, 24, 26] which learn the heterogeneous feature presentations by linear mathematics models, such as matrix factorization, subspace learning, and linear translator function. In DTNs, we model two SAEs that take a pair of text and image as input respectively, followed by multiple **parameter-sharing** network layers at the top. The output of the shared layer in DTNs yields the translator function that can be used to transfer cross-modal information. To the best of our knowledge, existing methods in literature like multimodal deep networks [20] are **representation-sharing**. Our parameter sharing scheme makes our networks more flexible, especially when the application requires more modality-specific features are learned. In other words, by sharing parameters, we allow deviation exists between the feature representations of different modalities. In contrast, in “representation-sharing structure”, a set of common features are constructed, which usually overestimates the importance of modality-specific features.

In this paper, we highlight the difference between the *strongly* parameter-shared layers and the *weakly* parameter-shared layers as

shown in Figure 1. Different from the strongly-shared layers, the weakly-shared layers allow the use of separate layers for different domains. The extent to which these layers are weakly-shared is adjustable by a regularizer modeling their difference. The benefit of weakly-shared layers is the flexibility of representing shared inter-domain features as well as domain-specific ones, making it more powerful in modeling complex data of multiple domains than strongly-shared layers.

We will show that DTNs are trained in a novel way that minimizes the errors incurred by a cross-modal information transfer process. In our experiments, we use image classification task to evaluate the effectiveness of trained DTNs. In testing phase, the test image without neither text nor label/tag can be represented by trained DTNs, and then assigned the class label by the trained translator. Extensive experiments on real world data sets show the effectiveness of DTNs compared with the other state-of-the-art methods. Figure 1 presents an overview of our proposed networks.

2. RELATED WORK

We briefly review some related works on deep learning and transfer learning in this section.

2.1 Deep Learning

Deep learning has been comprehensively reviewed and discussed in [1]. As one of the variants of deep learning models, SAEs have been widely used for face recognition [15], motion detection [35], multimedia retrieval [22, 9], etc. The classical SAEs linearly stack multiple layers of *Auto-Encoders* together to learn higher-level representation. The high-level representation output by SAEs can be used as input to a stand-alone supervised learning algorithm, e.g.,

Support Vector Machine, Softmax, Logistic Regression, etc. Consider their superior performance in feature learning, we will adopt SAEs as basic building blocks to model multi-modal representations upon which a novel multi-layered translator function will be built to transfer discriminative information across heterogeneous domains.

2.2 Transfer Learning

Transfer learning [23] aims to improve a learning task with little or no additional supervised information in a target domain by propagating the knowledge from the other source domains in which there are usually abundant training data. Transfer learning can be mainly categorized into two subsets: homogeneous transfer learning (domain adaptation) [7, 18, 21, 14] in a single domain but with different distributions in training and testing sets, and heterogeneous transfer learning [37, 8, 24] across different modalities. Most of existing transfer learning algorithms focus on the former class of transfer learning problem, while the latter one is more challenging. In this paper, we will focus on the second class of transfer learning problem. Our focus is a novel transfer learning scenario where the heterogeneous data in different domains is not aligned, and a cross-modal alignment must be learned before information can be transferred across different modalities. This will make the transfer learning problem more challenging as compared with the existing scenarios.

2.3 Alliance of Deep Learning and Transfer Learning

Deep learning has been explored to marry the transfer learning in literature [2]. For example, Glorot *et al.* [10] employed *Stack Denoising Autoencoders* (SDAs) to learn hidden feature representation for homogeneous cross-domain sentiment classification. Zhang *et al.* [36] proposed a deep neural network for domain adaptation by modeling and matching both the marginal and the conditional distribution between two homogeneous data. To reduce high computational cost and enhance the scalability of SDAs, Chen *et al.* [4] proposed marginalized SDAs. Socher *et al.* [29] first attempted to study heterogeneous transfer learning, though they focused on learning a zero-shot image representation than directly transferring cross-modal information. On the contrary, the proposed DTNs will consider a practical heterogeneous transfer learning scenario where the image concepts can be directly modeled from the text labels. This way, we can fully explore the rich cross-modal information to train densely connected deep transfer models while avoiding the overfitting problem.

The related deep representation networks (i.e., multimodal deep learning [20, 30], multimodal learning with DBMs [31], etc.) learning joint representations across multiple modalities are only shared on top of layers of modality-specific sub-networks. Deeply coupled auto-encoder networks [34] learn two deep networks embedded intra-class compactness and inter-class penalty with each other in each layer. **The key difference is that these existing models are “sharing representation”, while our method is “sharing parameters”, which makes our networks more robust for modality-specific features.**

3. PROBLEM DEFINITION

First, we consider text as source domain and image as target domain in this paper. Then we are given a labeled text data set $\mathcal{D}_S = \{(\bar{\mathbf{x}}_j^S, \bar{y}_j^S)\}_{j=1}^{N_S}$ in text domain, in which $\bar{\mathbf{x}}_j^S \in \mathbb{R}^a$ is the text data of source domain and $\bar{y}_j^S \in \{+1, -1\}$ is the corre-

sponding label¹. Our goal is to transfer the labels from the text data set to the target image domain for a classification task. To facilitate the label transfer process, we also have another co-occurrence set of paired texts and images $\mathcal{C} = \{(\mathbf{x}_i^S, \mathbf{x}_i^T)\}_{i=1}^{N_C}$, where $\mathbf{x}_i^S \in \mathbb{R}^a$ and $\mathbf{x}_i^T \in \mathbb{R}^b$ denote one text and image respectively. By exploring this co-occurrence set, we can reveal the alignment between texts and images, which will facilitate the label transfer between these two domains.

In particular, in this paper, we will jointly learn the deep representation of texts and images to effectively transfer the discriminative information from the source text domain to the target image domain. At the core of the deep transfer learning process is an efficient cross-modal translator which can transform the labels from text domain to image domain, even with the challenge of sparsely labeled target-domain data. By leveraging the learned deeply translator, we can solve the image classification task with extremely insufficient training data.

4. PROPOSED NETWORKS

4.1 The Architecture

For an input vector \mathbf{x}_0 , the formulation of a classical *Auto-Encoder* is comprised of an encoder function $h(\mathbf{x}) = s_e(\mathbf{W}\mathbf{x} + \mathbf{b})$ and a counterpart decoder function $\tilde{h}(\mathbf{x}) = s_d(\tilde{\mathbf{W}}\mathbf{x} + \tilde{\mathbf{b}})$ to minimize the reconstruction error of loss function $loss(\mathbf{x}_0, \tilde{h}(h(\mathbf{x}_0)))$. $s_e(\cdot)$ and $s_d(\cdot)$ are the non-linear activation function and decoder’s activation function respectively. Several auto-encoders can be consecutively stacked to form the *Stacked Auto-Encoders* (SAEs) by feeding the hidden representation of the l -th auto-encoder into the $l+1$ -th auto-encoder.

Built upon SAEs, we propose a Deep Transfer Networks² for information transfer across two domains, whose structure² is shown in Figure 1. First we pre-train two SAEs for text and image respectively, which output hidden representations of these two domains via the multiple layers of nonlinear encoding. Assume that these two SAEs have L_1 layers, where they are structured and built separately. After the first L_1 layers, the two SAEs begin to share their structure where L_2 shared layers are constructed as shown in Figure 1. These shared layers provide a way to transfer the information across two different domains represented by the two SAEs. In this case, the input into the shared layers is the (additive or multiplicative) mixture of outputs of the bottom L_1 layers by two SAEs.

However, we find this sort of *strongly* parameter-shared layers tend to over-mix the features learned from two heterogeneous domains, namely text and images. In other words, although there exist many shared features across different domains, it is risky to completely ignore the domain-specific features through the shared layers, since text and images often contain many elements that cannot be expressed by the same set of the neurons in the shared layers. In this paper, we relax such strongly shared structure, and propose to use *weakly* parameter-shared layers, which balance between the shared features and domain-specific ones.

Formally, there are $L+1$ layers in DTNs, where $L = L_1 + L_2$. Given a pair of input text \mathbf{x}_i^S and image \mathbf{x}_i^T , we use $\mathbf{x}_{S_i}^{(l)} \in \mathbb{R}^{a_i}$ and $\mathbf{x}_{T_i}^{(l)} \in \mathbb{R}^{b_i}$ to denote the latent representation of a hidden layer l

¹We reserve S and T in the superscript to denote the source domain and target domain respectively throughout of this paper.

²It is worth pointing out that the SAEs for each domain can be built upon another deep network. For example, we can create a Convolutional Neural Networks whose output is ingested into the SAEs for image domain. However, to avoid notational and illustration clutters, we do not explicitly show this structure.

for the two SAEs respectively. At the first layer, we set $\mathbf{x}_{S_i}^{(0)} = \mathbf{x}_i^S$ and $\mathbf{x}_{T_i}^{(0)} = \mathbf{x}_i^T$ as inputs. For easy of presentation, we drop the subscripts i of $\mathbf{x}_{S_i}^{(l-1)}$ and $\mathbf{x}_{T_i}^{(l-1)}$ ($l = 1, 2, \dots, L$) in the following. For $l = 1, 2, \dots, L$, the layer-wise processing of these two inputs through the whole network is defined as follows

$$\begin{aligned} \mathbf{x}_S^{(l)} &\triangleq h_S^{(l)}(\mathbf{x}^S) = s_e(\mathbf{W}_S^{(l)} \mathbf{x}_S^{(l-1)} + \mathbf{b}_S^{(l)}) \in \mathbb{R}^{a_l}, \\ \mathbf{x}_T^{(l)} &\triangleq h_T^{(l)}(\mathbf{x}^T) = s_e(\mathbf{W}_T^{(l)} \mathbf{x}_T^{(l-1)} + \mathbf{b}_T^{(l)}) \in \mathbb{R}^{b_l}, \end{aligned} \quad (1)$$

where $h_S^{(l)}(\cdot)$ and $h_T^{(l)}(\cdot)$ denote the l -th layer hidden representation in text and image SAEs respectively, $\{\mathbf{W}_S^{(l)}, \mathbf{b}_S^{(l)}\}_{l=1}^L$ and $\{\mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}\}_{l=1}^L$ are the parameters in text and image SAEs respectively.

The first L_1 layers are expected to learn the representations of text and image respectively, while the last L_2 layers provide shared representations. Under *weakly parameter-sharing* assumption, the parameters of the last L_2 layers should be set to be equal to a certain degree in order to trade off between modeling the shared cross-domain features and preserving necessary domain-specific details. Therefore, we propose the following penalty term Ω to quantify such trading-off

$$\Omega = \sum_{l=L_1+1}^L (\|\mathbf{W}_S^{(l)} - \mathbf{W}_T^{(l)}\|_F^2 + \|\mathbf{b}_S^{(l)} - \mathbf{b}_T^{(l)}\|_2^2). \quad (2)$$

Minimizing this term will minimize the difference of parameters in the last L_2 layers of two SAE networks. It will be added to the proposed objective function in the next section. A positive value of balancing coefficient will be multiplied with it in the objective function, which reflects the above trading-off for the weakly shared layers.

4.2 Deeply Transfer Mechanism

The goal of the proposed heterogenous domain transfer algorithm is to transfer the labels annotated on the source domain data set \mathcal{D}_S to annotate an arbitrary test data \mathbf{x}^T in the target domain. For this purpose, for a pair of image \mathbf{x}^T and text $\bar{\mathbf{x}}_j^S$ in \mathcal{D}_S , we define a translator function as the inner product $(h_S^{(L)}(\bar{\mathbf{x}}_j^S))' h_T^{(L)}(\mathbf{x}^T)$ between their weakly shared features output from the two SAEs, where $'$ denotes transpose operation in this paper. This translator function is used to propagate the labels from the source domain to the target domain as follows

$$f(\mathbf{x}^T) = \sum_j^{N_S} \bar{y}_j^S (h_S^{(L)}(\bar{\mathbf{x}}_j^S))' h_T^{(L)}(\mathbf{x}^T), \quad (3)$$

which combines all the source labels weighted by the corresponding translator functions. For the binary classification task considered in this paper, $f(\mathbf{x}^T)$ is also a discriminant function, whose sign predicts the class of the corresponding target domain data \mathbf{x}^T .

We learn the parameters in the proposed DTNs by minimizing the loss incurred by the above label transfer process, as well as maximizing the consistency between a set of co-occurrence pairs of texts and images to capture the cross-domain alignment information. We elaborate these two criteria below.

- **Empirical loss on the auxiliary set.** We have a small size of auxiliary training set $\mathcal{A}_T = \{(\tilde{\mathbf{x}}_t^T, \tilde{y}_t^T)\}_{t=1}^{\tilde{N}_T}$ in the target domain, where $\tilde{\mathbf{x}}_t^T \in \mathbb{R}^b$ of the target space and $\tilde{y}_t^T \in \{+1, -1\}$ is the label. We wish to minimize the training errors incurred by the label transfer function f on this set. We define the

Algorithm 1 Training of Weakly-Shared Deep Transfer Networks

Input: $\mathcal{D}_S = \{(\bar{\mathbf{x}}_j^S, \bar{y}_j^S)\}_{j=1}^{N_S}$, $\mathcal{C} = \{(\mathbf{x}_i^S, \mathbf{x}_i^T)\}_{i=1}^{N_C}$, $\mathcal{A}_T = \{(\tilde{\mathbf{x}}_t^T, \tilde{y}_t^T)\}_{t=1}^{\tilde{N}_T}$, $L_1, L_2, \gamma, \lambda, \mu$, dimension per layers and maxIter.
Output: Parameter set Θ .
Initialization: Initialize parameters in set Θ , $iter \leftarrow 1$.
1: // pre-training
2: **for** $l = 1, 2, \dots, L$ **do**
3: Pre-train text and image SAEs with inputs $\{\mathbf{x}_i^S\}_{i=1}^{N_C}$ and $\{\mathbf{x}_i^T\}_{i=1}^{N_C}$ respectively.
4: **end for**
5: Extract hidden representations on pre-trained DTNs for all text and image examples.
6: // dual fine tuning
7: **repeat**
8: **for** $l = 1, 2, \dots, L$ **do**
9: Update $\mathbf{W}_S^{(l)}, \mathbf{b}_S^{(l)}$ with Eq. (7) and (8).
10: Update $\mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}$ with Eq. (9) and (10).
11: **end for**
12: **for** $l = 1, 2, \dots, L_1$ **do**
13: Fine tune $\mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}$ with the back-propagated errors from the softmax output layer of image SAEs.
14: **end for**
15: Extract hidden representations on DTNs for all text and image examples.
16: $iter \leftarrow iter + 1$.
17: **until** Convergence or $iter > \text{maxIter}$.

training errors on \mathcal{A}_T as

$$\operatorname{argmin}_{\Theta} J_1 = \sum_{t=1}^{\tilde{N}_T} \ell(\tilde{y}_t^T \cdot f(\tilde{\mathbf{x}}_t^T)), \quad (4)$$

where we adopt a logistic loss function $\ell(x) = \log(1 + \exp(-x))$ to measure the cross-domain label transfer error.

The set of parameters of DTNs over which we will minimize J_1 include $\Theta = \{\mathbf{W}_S^{(l)}, \mathbf{b}_S^{(l)}, \mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}\}_{l=1}^L$ at each layer, which will determine the obtained weakly shared feature representations.

- **Empirical loss on co-occurrence pairs.** Given a set of co-occurrence pairs $\mathcal{C} = \{(\mathbf{x}_i^S, \mathbf{x}_i^T)\}_{i=1}^{N_C}$ of texts and images, we wish to maximize the alignment between each pair of examples in this set, yielding a translator function that can well capture the alignment between these co-occurrence pairs. Mathematically, we minimize the following objective function

$$\operatorname{argmin}_{\Theta} J_2 = \sum_{i=1}^{N_C} \chi\left(\left(h_S^{(L)}(\mathbf{x}_i^S)\right)' h_T^{(L)}(\mathbf{x}_i^T)\right), \quad (5)$$

where $\chi(x) = \exp(-x)$ is an exponential loss function, which can be seen as a measurement of mis-alignment between co-occurrence pairs caused by the translator function. Clearly, minimizing this loss function will ensure a large response of translator function over the co-occurrence set \mathcal{C} .

4.3 Objective Function and Optimization

After the above analyses, we propose the following objective function to learn the parameters of deep semantic translator as input of the top-layers of DTNs

$$\operatorname{argmin}_{\Theta} J = J_1 + \eta J_2 + \frac{\gamma}{2} \Omega + \frac{\lambda}{2} \Psi, \quad (6)$$

where $\Psi = \sum_{l=1}^L (\|\mathbf{W}_S^{(l)}\|_F^2 + \|\mathbf{b}_S^{(l)}\|_2^2 + \|\mathbf{W}_T^{(l)}\|_F^2 + \|\mathbf{b}_T^{(l)}\|_2^2)$ is the regularization term. The parameters λ , γ and η weigh the importance of different terms. In particular, η is the importance weight on alignment between the co-occurrence pairs, γ adjusts the weakly shared structure, and λ weighs the regularization term.

To train the DTNs, we first pre-train each layer per time in a greedy fashion by using the unsupervised data as in conventional auto-encoder algorithms. The pre-trained DTNs set up a good starting point that can be fine tuned according to the objective function (6) by employing the available supervision information.

We implement a back propagation process starting from the top output layers down through the whole DTNs to adjust all parameters. Each parameter in Θ is updated by stochastic gradient descent in back propagation algorithm below:

$$\mathbf{W}_S^{(l)} = \mathbf{W}_S^{(l)} - \mu \frac{\partial J}{\partial \mathbf{W}_S^{(l)}}, \quad (7)$$

$$\mathbf{b}_S^{(l)} = \mathbf{b}_S^{(l)} - \mu \frac{\partial J}{\partial \mathbf{b}_S^{(l)}}, \quad (8)$$

$$\mathbf{W}_T^{(l)} = \mathbf{W}_T^{(l)} - \mu \frac{\partial J}{\partial \mathbf{W}_T^{(l)}}, \quad (9)$$

$$\mathbf{b}_T^{(l)} = \mathbf{b}_T^{(l)} - \mu \frac{\partial J}{\partial \mathbf{b}_T^{(l)}}, \quad (10)$$

where μ is the learning rate. In more detail, the gradient of the objective function J with respect to the parameters $\{\mathbf{W}_S^{(l)}, \mathbf{b}_S^{(l)}, \mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}\}_{l=1}^{L_1}$ of the weakly shared layers can be computed as that:

(1) for $l = 1, 2, \dots, L_1$, we have

$$\frac{\partial J}{\partial \mathbf{W}_S^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{W}_S^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{W}_S^{(l)}} + \lambda \mathbf{W}_S^{(l)}, \quad (11)$$

$$\frac{\partial J}{\partial \mathbf{b}_S^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_S^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_S^{(l)}} + \lambda \mathbf{b}_S^{(l)}, \quad (12)$$

$$\frac{\partial J}{\partial \mathbf{W}_T^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{W}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{W}_T^{(l)}} + \lambda \mathbf{W}_T^{(l)}, \quad (13)$$

$$\frac{\partial J}{\partial \mathbf{b}_T^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_T^{(l)}} + \lambda \mathbf{b}_T^{(l)}; \quad (14)$$

(2) for $l = L_1 + 1, L_1 + 1, \dots, L$, we have

$$\frac{\partial J}{\partial \mathbf{W}_S^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{W}_S^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{W}_S^{(l)}} + \gamma (\mathbf{W}_S^{(l)} - \mathbf{W}_T^{(l)}) + \lambda \mathbf{W}_S^{(l)}, \quad (15)$$

$$\frac{\partial J}{\partial \mathbf{b}_S^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_S^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_S^{(l)}} + \gamma (\mathbf{b}_S^{(l)} - \mathbf{b}_T^{(l)}) + \lambda \mathbf{b}_S^{(l)}, \quad (16)$$

$$\frac{\partial J}{\partial \mathbf{W}_T^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{W}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{W}_T^{(l)}} + \gamma (\mathbf{W}_T^{(l)} - \mathbf{W}_S^{(l)}) + \lambda \mathbf{W}_T^{(l)}, \quad (17)$$

$$\frac{\partial J}{\partial \mathbf{b}_T^{(l)}} = \sum_{t=1}^{\tilde{N}_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_T^{(l)}} + \gamma (\mathbf{b}_T^{(l)} - \mathbf{b}_S^{(l)}) + \lambda \mathbf{b}_T^{(l)}. \quad (18)$$

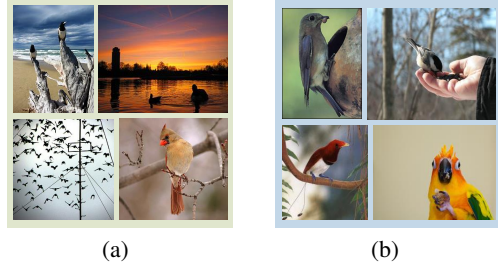


Figure 2: Examples of *birds* images from NUS-WIDE and ImageNet. (a) *birds* images from NUS-WIDE; (b) *birds* images from ImageNet.

We define $u_t = \tilde{y}_t^T \cdot f(\tilde{\mathbf{x}}_t^T)$ and $v_i = (h_S^{(L)}(\mathbf{x}_i^S))' h_T^{(L)}(\mathbf{x}_i^T)$ in above Eq. (11)~(18). The derivatives of $\ell(u_t)$ and $\chi(v_i)$ over the network parameters can be computed in a similar back propagation fashion as for the conventional neural networks. The gradient of the objective function J over the network parameters of the first L_1 layers of two separate SAEs can be computed similarly as in the above equations, except for the third term of RHS of each equation which accounts for the weakly shared layers only.

The above fine tuning strategy is more effective to update the text SAE than its image counterpart, since more number of text labels are involved to tune the networks. To more adequately tune the image SAE in conjunction with the tuning of text SAE, we exploit the supervision information in the training set \mathcal{A}_T to directly tune the image SAE. Specifically, we add an additional softmax layer upon image SAE that outputs the image labels. Then the labeled images in \mathcal{A}_T are used to compute the back-propagated errors to tune the parameters in the image SAE. These back-propagated errors are different from those computed from the objective function J that is based on label transfer process. Here the back-propagated errors only arise from the image labels that are intended to enhance the tuning of image SAE. This can avoid unbalanced tuning of text and image SAEs when much more text labels are used in label transfer process.

We alternate between the above two fine tuning strategies – minimizing the objective function J and tuning image SAE with labeled examples in \mathcal{A}_T . We call such mixed strategy *dual-fine tuning* in this paper. This is contrary to *single-fine tuned DTNs* (sigDTNs) that only trains the DTNs by minimizing J . We will compare these two methods in experiments. The detailed procedure of *dual-fine tuned DTNs* (duftDTNs) is described in Algorithm 1. The convergence criterion is that the iteration steps shall end when the number of iterations reaches the max or the relative cost of objective function is smaller than a predefined threshold.

5. EXPERIMENTS

5.1 Datasets

We conduct our experiments on NUS-WIDE data set [5], which consists of 269, 648 Flickr images and the associated text parts containing the user tags and comments annotated on each image. In our experiments, we use ten categories to evaluate the effectiveness on the image classification task, including *birds*, *building*, *cars*, *cat*, *dog*, *fish*, *flowers*, *horses*, *mountain* and *plane*. To train DTNs, we collect 1, 500 co-occurrence pairs of text and image for each category. The co-occurrence pairs of image and text with top-10 high-frequency words for 10 categories are shown in Table 1. The text descriptions of user tags in this co-occurrence set are labeled











<i>birds</i>	<i>building</i>	<i>cars</i>	<i>cat</i>	<i>dog</i>	<i>fish</i>	<i>flowers</i>	<i>horses</i>	<i>mountain</i>	<i>plane</i>
									
bird	sky	car	cat	dog	fish	flower	horse	mountain	airplane
nature	night	street	kitty	beach	sea	plant	foal	landscape	aircraft
sky	city	road	kitten	puppy	water	rose	nature	nature	plane
animal	architecture	locomotive	animal	pet	nature	color	bravo	cloud	aviation
wildlife	water	automobile	cute	running	color	spring	brazil	sky	airport
water	clouds	traffic	pet	animal	dark	nature	brasil	now	flying
flight	blue	vehicle	feline	water	pet	pink	argentina	blue	jet
animal	building	city	tabby	blue	ocean	grass	cloud	lake	sky
blue	sunset	police	nature	cute	swimming	cute	sky	water	flight
sea	skyline	train	white	nature	blue	beautiful	animal	tree	fighter

Table 1: Exhibition of co-occurrence pair of image and text with top-10 high-frequency words for all 10 categories.

# training images	Single-Dataset (SD) setting						Cross-Dataset (CD) setting					
	SVM	SAEs	HTL	TTI	sigDTNs	duftDTNs	SVM	SAEs	HTL	TTI	sigDTNs	duftDTNs
2	58.05	59.11	63.81	71.21	75.52	77.64	59.01	60.54	66.22	71.69	76.62	80.01
3	59.58	62.25	65.97	71.47	75.69	78.95	60.65	63.34	67.23	72.16	77.14	80.30
4	62.97	65.21	67.21	71.38	76.02	79.27	61.97	65.07	68.22	72.44	77.50	80.63
5	63.91	66.04	69.20	71.57	76.34	79.46	63.33	66.23	69.28	72.82	77.86	81.05
6	65.45	67.25	70.10	71.69	76.70	79.50	64.94	66.84	70.17	73.26	78.28	81.53
7	66.76	68.95	71.00	71.66	76.27	79.82	66.48	68.09	71.01	73.40	78.53	81.67
8	67.29	68.60	71.36	72.02	77.54	80.04	67.19	69.32	71.64	73.56	78.80	81.90
9	67.73	69.94	71.86	72.14	77.68	80.21	68.07	69.89	72.25	73.93	79.24	82.31
10	67.83	70.66	72.52	72.35	77.95	80.72	68.76	71.19	72.62	74.04	79.41	82.50

Table 2: Average accuracy (%) of various algorithms vs. number of training images in SD and CD settings.

by these ten categories. These text labels are transferred by the trained DTNs to label the images. A test set of images without any co-occurred texts are annotated with ground truth labels on NUS-WIDE dataset for evaluation purpose.

5.2 Experiment Settings

We consider two different experiment settings to evaluate the performance of the proposed DTNs.

- **Single-Dataset (SD) setting:** In this setting, we use NUS-WIDE data set to train the DTNs model. Then we use the trained DTNs to transfer the text labels to annotate the test images in NUS-WIDE data set. This is a challenging image annotation task since the images in Flickr are taken out of focus by amateur photographers. These images often have a wide range of resolution sizes, ranging from extremely small sizes (tens of pixels in each dimension) to very large sizes; many of them contain cluttered backgrounds. All of these factors make it difficult to model images with only limited visual information. To address this challenge, we use DTNs to transfer the text labels to enhance the discriminant information for modeling images in Flickr.
- **Cross-Dataset (CD) setting:** In this setting, we train the DTNs model with NUS-WIDE data set, but then use the trained DTNs to transfer the text labels from NUS-WIDE to annotate the images in ImageNet [6]. This setting can test the generalization ability of DTNs across different image data sets. For the sake of fair comparison, we test on ImageNet

with the same set of ten image categories and their subcategories as in NUS-WIDE. Figure 2 shows some examples of *birds* images from NUS-WIDE and ImageNet. We can see that the images from NUS-WIDE are visually diverse, while those images of the same category from ImageNet are more visually consistent with each other.

The performance of the proposed DTNs (with sigDTNs and duftDTNs) are compared with a set of state-of-the-art algorithms below. (1) SVMs: they are conventional shallow structured classifiers and set as the baseline for comparison; (2) SAEs: they represent deep structured network. SAEs model image content, and we connect them to a logistic output layer that directly gives the predicted labels on images. Both SVMs and SAEs only use image content, without any text labels involved. We also compare with the other heterogeneous transfer learning algorithms exploring text labels. (3) *Heterogeneous Transfer Learning* (HTL) [37]: it maps each image into a latent vector space via a formulated implicit distance function. HTL also makes use of the occurrence information between images and text documents as well as images and visual words. (4) *Translator from Text to Image* (TTI) [24]: it learns a translator on co-occurred pairs of text and images as well as a small size of training images and effectively convert the semantics from texts to images for image classification task. Both of these two algorithms explore the text labels to model images. TTI has been reported to achieve the state-of-the-art performances over Flickr dataset, however it only considers a shallow label transfer structure.

We compare the performances of different algorithms with varying numbers of training images, ranging from 2 to 10. In both experimental settings, SD and CD, the training images are random-

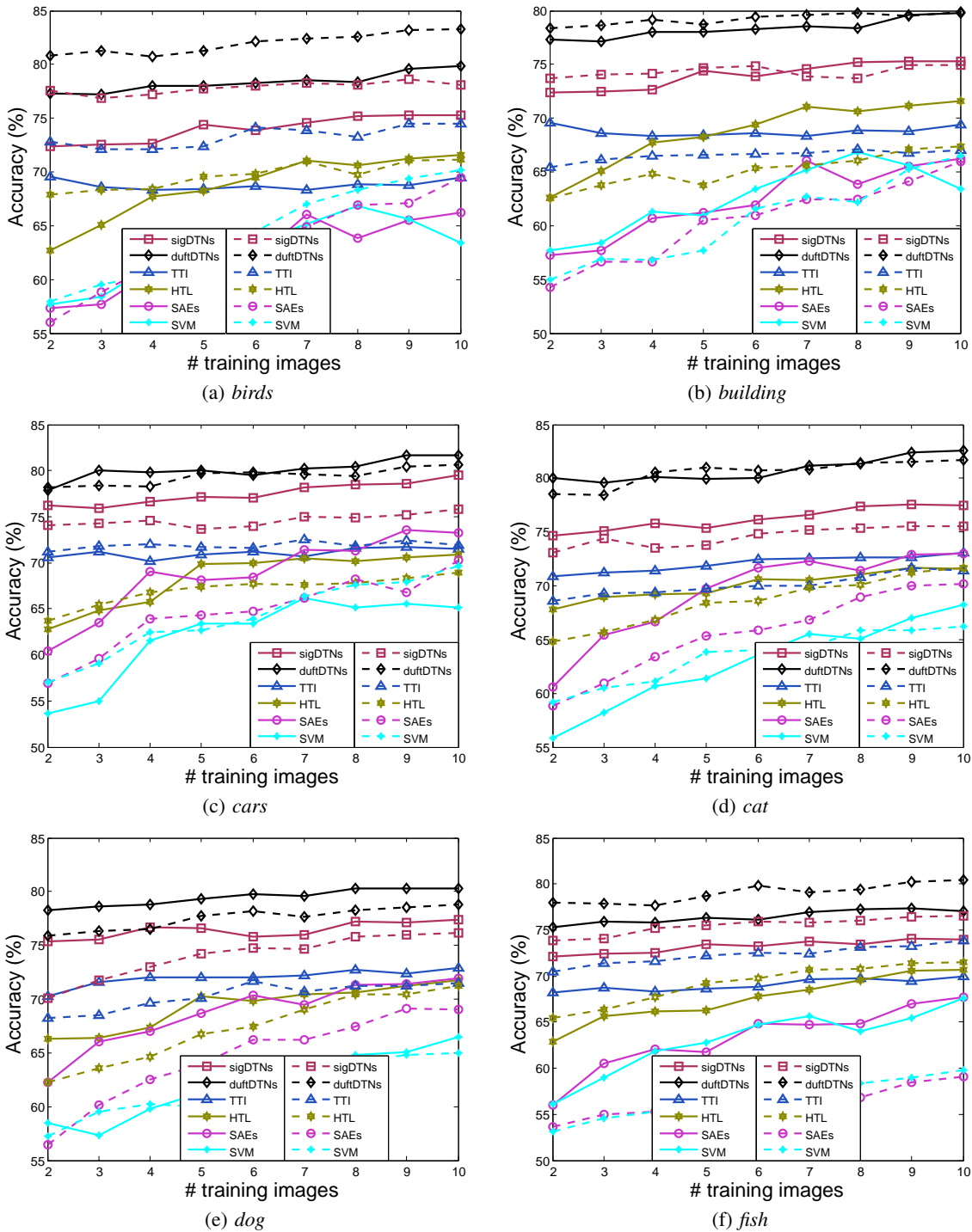


Figure 3: Accuracy (%) of different algorithms with number of auxiliary labeled images. Solid line and dotted line denote the two experiment settings *SD* and *CD*, respectively. This figure is followed by Figure 4.

ly selected, and the process is repeated ten times, and the average performance is reported.

We also compare the accuracy of the compared algorithms with the varying numbers of co-occurrence text-image pairs. All the parameters in our model are tuned based on a 2-fold cross-validation procedure on the training set, and the parameters are selected when the best performance is achieved.

5.3 Results

In this paper, we train 5-layer DTNs, in which the number of neurons are $1226 \rightarrow 618 \rightarrow 128 \rightarrow 128 \rightarrow 60$ and $1000 \rightarrow 512 \rightarrow 128 \rightarrow 128 \rightarrow 60$ at each layer from bottom up in images SAEs and text SAEs, respectively. The last three ones are weakly shared layers. At the input layers of DTNs, 1,000 words are extracted and stemmed from the text parts and their frequencies are input into text

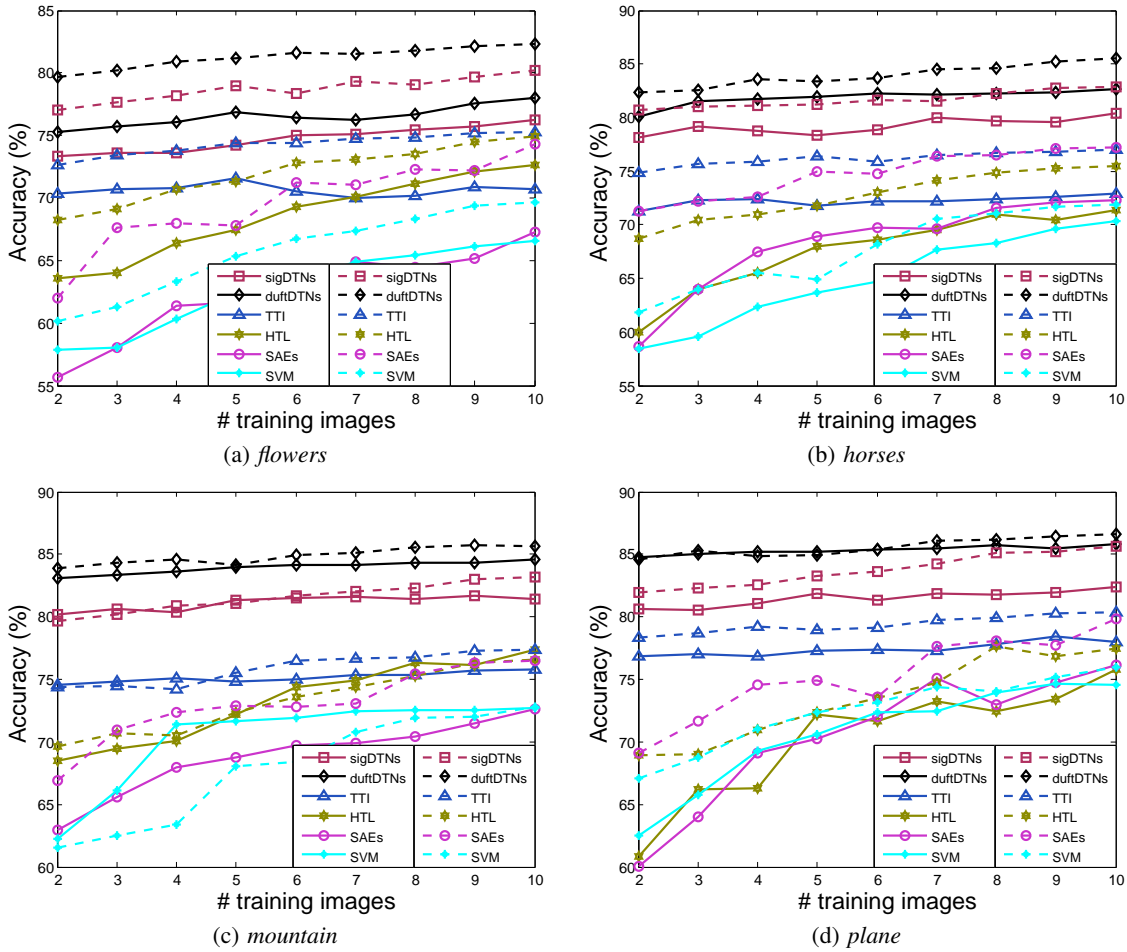


Figure 4: Accuracy (%) of different algorithms with number of auxiliary labeled images. Solid line and dotted line denote the two experiment settings SD and CD, respectively. This figure follows Figure 3.

SAEs. We extract the 4096 dims CNN features as an visual features for images by AlexNet [16, 13], and then reduce the dimensionality of CNN features into the 1226 dims by PCA.

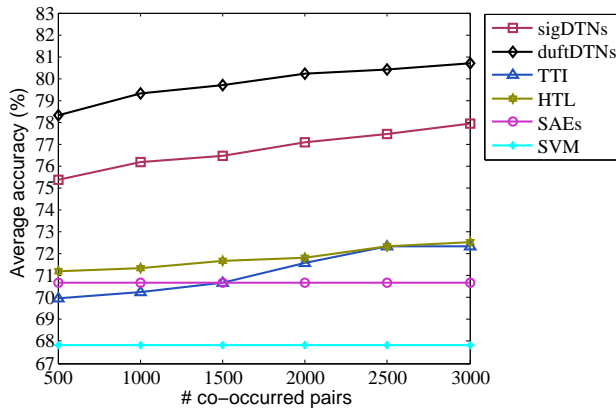
The accuracies over all categories with varying number of auxiliary training examples are plotted in Figure 3 and Figure 4. Either in SD setting or CD setting, the accuracy of traditional SVMs and SAEs algorithms is the lowest among all the compared algorithms no matter how many training images are used, since neither of them explores the heterogeneous domain. The performance of the compared transfer learning algorithms, including TTI, HTL and the proposed DTNs, is improved to different degrees. Among them, both of two proposed training algorithms, sigDTNs and duftDTNs, have performed the best.

All the subplots in Figure 3 and 4 show that the proposed DTNs, with both training algorithms of sigDTNs and duftDTNs, outperform the other compared algorithms. It illustrates the amazing advantages of the proposed DTNs when there are an extremely little amount of training data. This is also consistent with our earlier assertions that our DTNs can work well even in the insufficiency of auxiliary training examples, by exploring the co-occurrence information between text and image. We also observe that duftDTNs often performs better than sigDTNs. This is due to the fact that duftDTNs use an extra step to tune the image SAEs with the training images. This better balances between the fine tuning of text

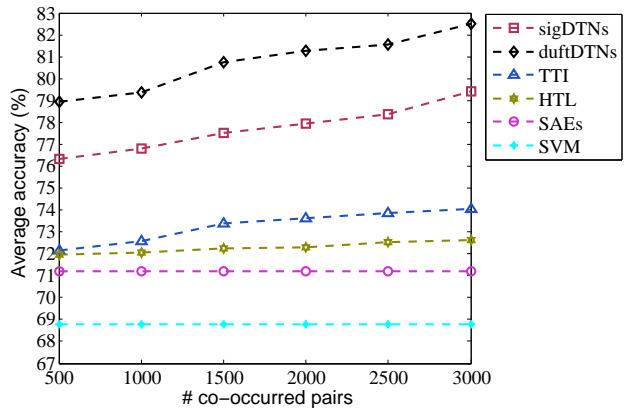
and image SAEs, when the text labels are much more than image labels.

Table 2 also reports the average accuracies of various methods over all 10 categories with varying number of training images. Results in both of two experiment settings are reported in this table. An interesting discovery is the accuracy in cross-dataset (CD) setting is much comparable with that in single-dataset (SD) setting. This suggests that the DTNs trained with one data set (NUS-WIDE) are well generalized to transfer the labels between data sets (i.e., from NUS-WIDE to ImageNet). This is a very useful property in many real applications when we have to handle data from different sources.

The above results for DTNs are obtained by using 3,000 co-occurrence pairs of text and image. Since the co-occurrence pairs play an important role as a bridge to connect heterogeneous domains in our proposed DTNs, we examine the effect with varying number of co-occurrence pairs of text and image in Figure 5, where the number of training images is fixed to 10. The average accuracy of TTI and DTNs in both SD and CD settings are increased with an increasing number of co-occurrence text-image pairs. This suggests that more co-occurrence pairs tend to provide more information to better model DTNs with improved performance.



(a) Experiment setting *SD*



(b) Experiment setting *CD*

Figure 5: Number of co-occurrence pairs vs. average accuracy (%) in SD and CD settings. (a) and (b) denote the two experiment settings *SD* and *CD* respectively.

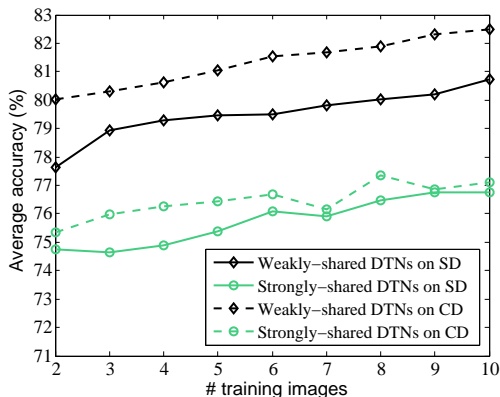


Figure 6: Comparison results of DTNs with weakly shared layers and strongly parameter-shared layers.

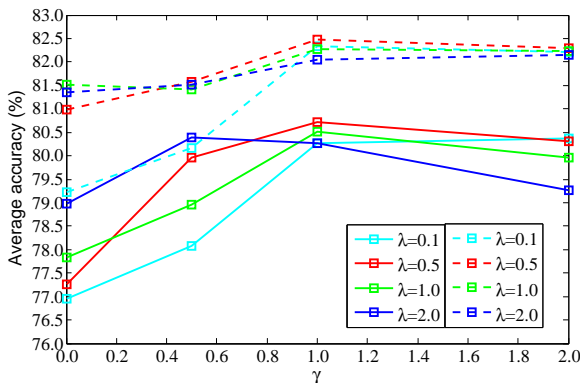


Figure 7: Parametric sensitivity vs. average accuracy (%) with different parameters γ and λ on 10 training images. Solid line and dotted line denote the SD and CD settings, respectively.

5.4 Strongly Parameter-Shared vs. Weakly Parameter-Shared Layers

To illustrate the superiority of the weakly parameter sharing, we also compare the performances of DTNs with strongly parameter-

shared layers and weakly parameter-shared layers. For strongly parameter-shared layers, we set $\mathbf{W}_S^{(l)} = \mathbf{W}_T^{(l)}$ and $\mathbf{b}_S^{(l)} = \mathbf{b}_T^{(l)}$ for text and image SAEs for the top layers $l = L_1 + 1, \dots, L$. In other words, these layers share the same neuron connections between two successive layers.

In this experiment, we also set the number of training images from 2 to 10. Figure 6 compares the performances of weakly parameter-shared layers and strongly parameter-shared layers. We can see that the performances of DTNs with weakly shared layers are further improved as compared with DTNs with strongly shared layers in both SD and CD settings. This confirms that proposed weakly shared layers are more suitable to model the DTNs than strongly shared layers.

5.5 Parameter Sensitivity

In the experiments, parameters γ and λ of object function J in Eq. (6) are chosen $\gamma \in \{0, 0.5, 1.0, 2.0\}$ and $\lambda \in \{0.1, 0.5, 1.0, 2.0\}$ respectively by a cross-validation procedure. Conventionally, we set $\eta = 1$ to equally weigh the two types of loss terms. Here, we study their impacts on the performances in Figure 7. When $\gamma = 0$, the average accuracy is the lowest. This is because in this case the text and image SAEs are completely independent without any shared layers. This structure fails to model the joint image and text representation, and is unable of transferring labels across heterogeneous domains. The accuracy increases rapidly when γ becomes large. On the other hand, λ can also improve the accuracy when it is set to a proper value to regularize the model. The best accuracy in both SD and CD settings is achieved when $\gamma = 1$ and $\lambda = 0.5$. The accuracies with different values of parameters are not varied very much, which suggests that DTNs are not very sensitive to these model parameters.

6. CONCLUSION

In this paper, we propose a type of novel *weakly-shared Deep Transfer Networks* (DTNs) to translate cross-domain information from text domain to image domain. The proposed DTNs with weakly parameter-shared layers can more powerfully capture complex representation of data of different domains with both shared inter-domain and domain-specific knowledge than the strongly parameter-shared layers. The DTNs are trained in a novel way that directly minimizes the loss incurred by a label transfer function. This yields a *dual-fine tuning* strategy to train DTNs from top down with

back-propagated errors that are derived from the label transfer loss alongside the loss from softmax output layer of image SAE to avoid unbalanced tuning. We show superior results of the proposed DTNs on extensive experiments as compared with the baselines and the other state-of-the-art methods.

7. ACKNOWLEDGEMENTS

This work was partially supported by the 973 Program (Project No. 2014CB347600), the National Natural Science Foundation of China (Grant No. 61402228), the Program for New Century Excellent Talents in University under Grant NCET-12-0632 and the Natural Science Fund for Distinguished Young Scholars of Jiangsu Province under Grant BK2012033.

8. REFERENCES

- [1] Y. Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.
- [2] Y. Bengio. Deep learning of representations for unsupervised and transfer learning. In *ICML*, pages 17–36, 2012.
- [3] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. *NIPS*, 19:153, 2007.
- [4] M. Chen, Z. Xu, F. Sha, and K. Q. Weinberger. Marginalized denoising autoencoders for domain adaptation. In *ICML*, pages 767–774, 2012.
- [5] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. Nus-wide: a real-world web image database from national university of singapore. In *CIVR*, page 48, 2009.
- [6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.
- [7] J. Donahue, J. Hoffman, E. Rodner, K. Saenko, and T. Darrell. Semi-supervised domain adaptation with instance constraints. In *CVPR*, pages 668–675, 2013.
- [8] L. Duan, D. Xu, and I. W. Tsang. Learning with augmented features for heterogeneous domain adaptation. In *ICML*, pages 711–718, 2012.
- [9] F. Feng, X. Wang, and R. Li. Cross-modal retrieval with correspondence autoencoder. In *ACM Multimedia*, pages 7–16, 2014.
- [10] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML*, pages 513–520, 2011.
- [11] G. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.
- [12] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv*, 2012.
- [13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, pages 675–678, 2014.
- [14] Y.-G. Jiang, C.-W. Ngo, and S.-F. Chang. Semantic context transfer across heterogeneous sources for domain adaptive video search. In *ACM Multimedia*, pages 155–164, 2009.
- [15] M. Kan, S. Shan, H. Chang, and X. Chen. Stacked progressive auto-encoders (spae) for face recognition across poses. In *CVPR*, pages 1883–1890, 2014.
- [16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.
- [17] H. Lee, C. Ekanadham, and A. Y. Ng. Sparse deep belief net model for visual area v2. In *NIPS*, pages 873–880, 2008.
- [18] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu. Transfer joint matching for unsupervised domain adaptation. In *CVPR*, pages 1410–1417, 2014.
- [19] J. Ngiam, Z. Chen, S. A. Bhaskar, P. W. Koh, and A. Y. Ng. Sparse filtering. In *NIPS*, pages 1125–1133, 2011.
- [20] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *ICML*, pages 689–696, 2011.
- [21] J. Ni, Q. Qiu, and R. Chellappa. Subspace interpolation via dictionary learning for unsupervised domain adaptation. In *CVPR*, pages 692–699, 2013.
- [22] X. Ou, L. Yan, H. Ling, C. Liu, and M. Liu. Inductive transfer deep hashing for image retrieval. In *ACM Multimedia*, pages 969–972, 2014.
- [23] S. J. Pan and Q. Yang. A survey on transfer learning. *TKDE*, 22(10):1345–1359, 2010.
- [24] G.-J. Qi, C. Aggarwal, and T. Huang. Towards semantic knowledge propagation from text corpus to web images. In *WWW*, pages 297–306, 2011.
- [25] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *ICML*, pages 833–840, 2011.
- [26] S. D. Roy, T. Mei, W. Zeng, and S. Li. Socialtransfer: cross-domain transfer learning from social streams for media applications. In *ACM Multimedia*, pages 649–658, 2012.
- [27] R. Salakhutdinov and G. E. Hinton. Deep boltzmann machines. In *AISTATS*, pages 448–455, 2009.
- [28] R. Salakhutdinov, A. Mnih, and G. Hinton. Restricted boltzmann machines for collaborative filtering. In *ICML*, pages 791–798, 2007.
- [29] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, pages 935–943, 2013.
- [30] K. Sohn, W. Shang, and H. Lee. Improved multimodal deep learning with variation of information. In *NIPS*, pages 2141–2149, 2014.
- [31] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *NIPS*, pages 2222–2230, 2012.
- [32] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1701–1708, 2014.
- [33] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *ICML*, pages 1096–1103, 2008.
- [34] W. Wang, Z. Cui, H. Chang, S. Shan, and X. Chen. Deeply coupled auto-encoder networks for cross-view classification. *arXiv*, 2014.
- [35] P. Xu, M. Ye, X. Li, Q. Liu, Y. Yang, and J. Ding. Dynamic background learning through deep auto-encoder networks. In *ACM Multimedia*, pages 107–116, 2014.
- [36] X. Zhang, F. X. Yu, S.-F. Chang, and S. Wang. Deep transfer network: Unsupervised domain adaptation. *arXiv*, 2015.
- [37] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G.-R. Xue, Y. Yu, and Q. Yang. Heterogeneous transfer learning for image classification. In *AAAI*, 2011.