

# Generalized Deep Transfer Networks for Knowledge Propagation in Heterogeneous Domains

JINHUI TANG, XIANGBO SHU, and ZECHAO LI, Nanjing University of Science and Technology  
 GUO-JUN QI, University of Central Florida  
 JINGDONG WANG, Microsoft Research Asia

In recent years, deep neural networks have been successfully applied to model visual concepts and have achieved competitive performance on many tasks. Despite their impressive performance, traditional deep networks are subjected to the decayed performance under the condition of lacking sufficient training data. This problem becomes extremely severe for deep networks trained on a very small dataset, making them overfitting by capturing nonessential or noisy information in the training set. Toward this end, we propose a novel generalized deep transfer networks (DTNs), capable of transferring label information across heterogeneous domains, textual domain to visual domain. The proposed framework has the ability to adequately mitigate the problem of insufficient training images by bringing in rich labels from the textual domain. Specifically, to share the labels between two domains, we build parameter- and representation-shared layers. They are able to generate domain-specific and shared interdomain features, making this architecture flexible and powerful in capturing complex information from different domains jointly. To evaluate the proposed method, we release a new dataset extended from NUS-WIDE at <http://imag.njust.edu.cn/NUS-WIDE-128.html>. Experimental results on this dataset show the superior performance of the proposed DTNs compared to existing 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

General Terms: Design, Algorithms, Experimentation, Performance

Additional Key Words and Phrases: Heterogeneous-domain knowledge propagation, cross-domain label transfer, deep transfer network, image classification

## ACM Reference Format:

Jinhui Tang, Xiangbo Shu, Zechao Li, Guo-Jun Qi, and Jingdong Wang. 2016. Generalized deep transfer networks for knowledge propagation in heterogeneous domains. *ACM Trans. Multimedia Comput. Commun. Appl.* 12, 4s, Article 68 (November 2016), 22 pages.  
 DOI: <http://dx.doi.org/10.1145/2998574>

## 1. INTRODUCTION

It is a big challenge for existing machine learning methods to do the visual concept classification [Song et al. 2016] with a small amount of labeled data. Two main reasons lead to this issue: First, the classical handcrafted visual features of images cannot reveal

---

This work was partially supported by the 973 Program of China (project 2014CB347600), the National Natural Science Foundation of China (grants 61522203 and 61402228), and the National Ten Thousand Talent Program of China (Young Top-Notch Talent).

Authors' addresses: J. Tang, X. Shu (corresponding author), and Z. Li, School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, P.R. China; emails: [jinhuitang@njust.edu.cn](mailto:jinhuitang@njust.edu.cn), [shuxb104@gmail.com](mailto:shuxb104@gmail.com), [zechao.li@njust.edu.cn](mailto:zechao.li@njust.edu.cn); G.-J. Qi, College of Engineering and Computer Science, University of Central Florida, Orlando, FL 32816; email: [guojun.qi@ucf.edu](mailto:guojun.qi@ucf.edu); J. Wang, Microsoft Research, Beijing 100080, P. R. China; email: [jingdw@microsoft.com](mailto:jingdw@microsoft.com).

Permission to make digital or hard copies of part or all 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 show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or [permissions@acm.org](mailto:permissions@acm.org).

© 2016 ACM 1551-6857/2016/11-ART68 \$15.00

DOI: <http://dx.doi.org/10.1145/2998574>

the semantic information very well due to their limited representation power [Bengio et al. 2012]. Second, a good visual concept classification model needs to be trained on massive labeled data, but labeling the training data is time consuming [Tang et al. 2011].

To improve the representation power, deep networks have been extensively investigated in recent years [LeCun et al. 2015]. It has been proven that deep networks, composed of multiple nonlinear transformations, can learn high-level representations more so than traditional shallow models [Bengio 2009; Vincent et al. 2008; Li et al. 2015; Krizhevsky et al. 2012; Salakhutdinov and Hinton 2009]. Current popular deep network architectures include deep belief networks (DBNs) [Hinton et al. 2006], convolutional neural networks (CNNs) [Krizhevsky et al. 2012], stacked autoencoders (SAEs) [Vincent et al. 2010], deep Boltzmann machines (DBMs) [Salakhutdinov and Hinton 2009], and their variants [Weston et al. 2012].

Traditionally, deep networks have millions of parameters. For example, the representative AlexNet model [Krizhevsky et al. 2012] trained on 1,000 classes of ILSVRC 2012<sup>1</sup> has 60 million parameters. Such deep networks are prone to overfitting when we train them on a (extremely) small training image set. Thus, they should be trained on a very large scale database. Generally, if we attempt to train deep networks in such a scenario, one straightforward way is to employ the deep networks pretrained on ImageNet [Russakovsky et al. 2014] with the dropout strategy [Ba and Frey 2013] to extract the representations of images, then fine tuning is used to further improve the representation power [Tang et al. 2015]. However, the samples used for fine tuning should also be manually labeled, and they must be homogeneous with the training samples.

In this article, we consider another way to address this challenge by bringing in massive data of image-text pairs, which are weakly supervised and can be easily acquired from social Web sites. Then we learn the deep networks on the heterogeneous training data by transferring the label information from a textual domain to a visual domain. We aspire to answer the following question: how can the cross-domain data help to model visual concepts? We find that it is beneficial to explore the text information for the following two reasons: (1) the word features of text data are more directly related to the semantic concepts [Vinyals et al. 2015; Karpathy and Fei-Fei 2015; Jia et al. 2015], and (2) abundant labeled text documents are widely available on Web sites. They inspire us to transfer the discriminative knowledge from textual space (i.e., source domain) to visual space (i.e., target domain). By transferring the feature representation and label information from textual space, we can learn a semantic-intensive image feature representation directly related to visual concepts. This will greatly improve the performance of image classification. In other words, the rich information transferred from a textual domain can help to train complex deep networks for image classification even with a small amount of labeled training images. In this article, we present how to use transferred cross-domain information to train a powerful deep network, which is very competitive on image classification tasks.

To this end, we propose a novel deep network architecture with several parameter- and representation-shared layers (called *generalized layers*) that hierarchically learn to transfer the semantic knowledge from web texts to images, as shown in Figure 1. We call this architecture generalized deep transfer networks (DTNs) in this article. As a hierarchically nonlinear model, DTNs differ from existing shallow transfer learning methods [Zhu et al. 2011; Duan et al. 2012; Qi et al. 2011; Roy et al. 2012], which learn the heterogeneous feature presentations by linear models, such as matrix factorization, subspace learning, and linear transfer function. In DTNs, we model two SAEs that take

<sup>1</sup><http://www.image-net.org/challenges/LSVRC/2012/>.

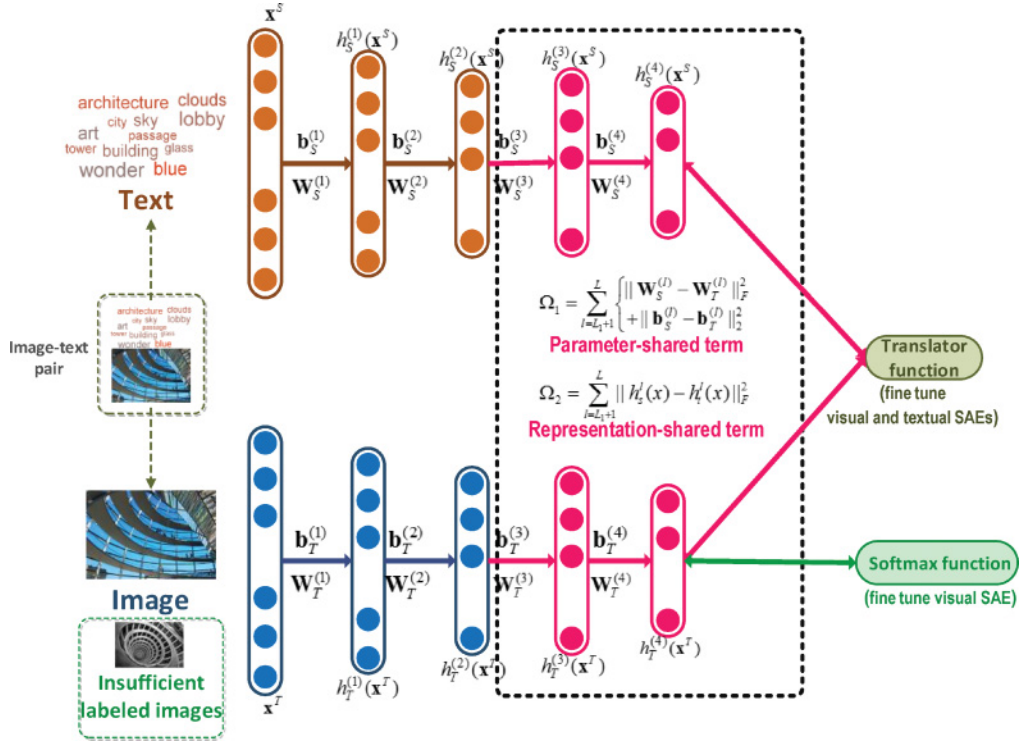


Fig. 1. Architecture of generalized DTNs. The generalized layers are parameter- and presentation-shared by imposing two penalty terms that decide the extent to which they should be shared. This architecture is quite flexible, enabling representation of both domain-specific features and shared interdomain features. The adopted dual fine tuning alternately starts from the objective function and softmax on the visual SAE.

texts and images as inputs, respectively, followed by multiple generalized layers. The outputs of the generalized layers in DTNs are used to learn the deep semantic translator function. This architecture is flexible to represent both domain-specific features and the shared features across domains.

Existing methods, such as multimodal deep networks [Ngiam et al. 2011; Feng et al. 2015], employ a representation-shared scheme that forces the learned representations of different modal data to be the same. This representation-shared scheme improves the performance to a certain extent. However, this scheme usually ignores the role of modality-specific features, such as background in images, or irrelevant words (e.g., preposition) in texts. To address this problem, we bring two parameter-shared schemes into the preliminary work [Shu et al. 2015]: the weakly shared scheme and the strongly shared scheme. The former forces the parameters of the last two layers in the two SAEs to be consistent with each other, whereas the latter forces the parameters of the last two layers in the two SAEs to be same with each other. It has been validated that the weakly shared scheme performs better than the strongly shared one. In this work, the proposed DTNs utilize the representation- and parameter-shared scheme, which makes the network architecture flexible and powerful in capturing complex information from different domains. In other words, by jointly sharing parameters and representations, this architecture can learn both domain-specific and shared interdomain representations. It is worth noting that the representation sharing in the representation- and parameter-shared scheme differing from the one in Ngiam et al. [2011] and Feng et al.

Table I. Four Variants of DTNs with Different Layer-Shared Schemes

Variant	Property
Strongly shared DTNs	Parameters of the last several layers in the two sub-networks are the same.
Parameter-shared DTNs	Parameters of the last several layers in the two sub-networks are weakly shared.
Representation-shared DTNs	Outputs of the last several layers in the two sub-networks are weakly shared.
Generalized DTNs (Parameter- and representation-shared DTNs)	Both parameters and outputs of the last several layers in the two sub networks are weakly shared.

[2015] controls the consistence of the outputs of the last several layers in the two SAEs. In Table I, we list four variants of DTNs: strongly shared DTNs, parameter-shared DTNs,<sup>2</sup> representation-shared DTNs [Feng et al. 2015], and our proposed generalized DTNs.

In this work, the proposed DTNs are trained in a novel way that minimizes the errors incurred by a cross-domain information transfer process. In particular, we explore various fine-tuning strategies to improve the performance of DTNs. To evaluate the proposed method, a new dataset extended from NUS-WIDE has been released (<http://imag.njust.edu.cn/NUS-WIDE-128.html>) by adding ground truth of 47 new concepts. In the training phase, the proposed DTNs are trained on the image-text pairs, which can be easily crawled from Internet, and a small number of labeled images. In the testing phase, the testing images without co-occurring texts and tags can be represented by the trained DTNs, and then they can be assigned the class labels by the translator function. Experimental results on this dataset show the superiority of the proposed DTNs over state-of-the-art methods.

The main contributions of this work are summarized as follows:

- We propose a novel DTN architecture equipped with the parameter- and representation-shared layers. The corresponding optimization procedure is also presented.
- We utilize the proposed DTNs to learn the semantic knowledge from Web texts and then transfer it to images by the learned translator function when there is a lack of sufficient training data in the visual domain.
- We present several variants of the proposed DTNs, with different sharing schemes (i.e., parameter-shared scheme, representation-shared scheme, and generalized scheme) and different fine-tuning strategies (i.e., single fine tuning and dual fine tuning).
- We release a new Web image dataset extended from NUS-WIDE [Chua et al. 2009] by extending the ground truth of the original 81 concepts to 128. This dataset can be used for the evaluation of social image retrieval, image annotation, multilabel image classification, social image tag refinement, cross-domain processing of image and text, and so forth.

Compared to the preliminary work [Shu et al. 2015], this work has the following improvements: (1) a more generalized DTN equipped with the generalized layers is

<sup>2</sup>In this article, the parameter-shared scheme refers to the weakly shared scheme only.

proposed; (2) more competitive results of the generalized DTNs on more concepts are provided; (3) the performance of strongly shared DTNs, parameter-shared DTNs, representation-shared DTNs, and generalized DTNs by experiments are compared and discussed; (4) a new dataset for social image retrieval, multilabel image classification, and cross-domain processing of image and text (cross-domain retrieval, transfer learning, etc.) is released.

The rest of the article is organized as follows. Section 2 reviews related work. Section 3 defines the problems and tasks of our work. Section 4 elaborates on the proposed DTNs, including the network architecture, transfer mechanism, objective function, and optimization procedure. Experimental results and analyses are presented in Section 5. Finally, Section 6 concludes this work and discusses future work.

## 2. RELATED WORK

This section briefly reviews related works on deep learning and transfer learning, as well as their alliance.

### 2.1. Deep Learning and SAEs

Deep learning has been comprehensively reviewed and discussed in the literature [Bengio 2009; LeCun et al. 2015] (e.g., CNNs [Krizhevsky et al. 2012], SAEs [Vincent et al. 2010], and DBMs [Salakhutdinov and Hinton 2009]). In particular, SAEs are well known to learn useful deep representations. Classical SAEs linearly stack multiple layers of autoencoders together to learn higher-level representations. The high-level representation learned by SAEs can be also used as input in a stand-alone supervised learning model (e.g., support vector machines (SVMs), softmax, and logistic regression). Considering the superior performance of SAEs in representation learning, we adopt them as basic building blocks to learn multimodal representations, on which a novel multilayered translator function will be built to transfer discriminative information across heterogeneous domains.

Many variants of SAEs have been proposed [Hong et al. 2015; Kalmanovich and Chechik 2014; Ba and Frey 2013; Rasmus et al. 2015]. For example, Kalmanovich et al. [2014] investigated the training scheme of SAEs and described a gradual training scheme that improves the reconstruction error under a fixed training budget compared to classical SAEs. Ba and Frey [2013] proposed an adaptive dropout for SAEs by approximately computing local expectations of binary dropout variables and computing derivatives using back-propagation and stochastic gradient descent. Gao et al. [2015] proposed a new supervised autoencoder, which is a new type of building block for deep architectures. Rasmus et al. [2015] proposed a novel SAE model to support supervised learning that is compatible with supervised learning by using the unsupervised denoising task as an auxiliary training objective function. As one of powerful representation learning methods, SAEs have been widely used for face recognition [Kan et al. 2014; Gao et al. 2015], motion detection [Xu et al. 2014], multimedia retrieval [Ou et al. 2014; Feng et al. 2014; Li and Tang 2015], pose analysis [Hong et al. 2015], and so forth.

### 2.2. Transfer Learning

Transfer learning [Raina et al. 2007] aims to improve the performance of a learning task with little or no additional supervised information in a target domain by propagating the knowledge from source domains with abundant training data. This has been comprehensively reviewed in Pan and Yang [2010]. Transfer learning can be mainly categorized into two classes: homogeneous transfer learning (domain adaptation) [Donahue et al. 2013; Long et al. 2014; Ni et al. 2013; Jiang et al. 2009] in a single domain but with different distributions in training and testing sets, and heterogeneous transfer learning [Zhu et al. 2011; Duan et al. 2012; Qi et al. 2011] across different



modalities. Specifically, homogeneous transfer learning transfers knowledge in the same modality, whereas heterogeneous transfer learning transfers knowledge across different modalities. Moreover, zero-shot learning [Gavves et al. 2015] is also a specific transfer learning in which there is no training data in the target domain. In this article, we focus on the heterogeneous transfer learning (HTL) problem, where there is insufficient training data in the target domain available. Our focus is a novel transfer learning scenario where the heterogeneous data in different domains is not aligned and a cross-domain alignment must be learned before the information can be transferred across different domains. This will make the transfer learning problem more challenging compared to the existing scenarios.

### 2.3. Alliance of Deep Learning and Transfer Learning

The alliance of deep learning and transfer learning is comprehensively reviewed in Bengio [2012]. Glorot et al. [2011] employed stack denoising autoencoders (SDAs) to learn hidden feature representation for homogeneous cross-domain sentiment classification. Zhang et al. [2015b] proposed a deep neural network structure 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. [2012] proposed marginalized SDAs. Socher et al. [2013] attempted to study heterogeneous transfer learning, although they focused on learning a zero-shot image representation rather than directly transferring cross-modal information. On the contrary, the proposed DTNs consider a practical HTL scenario where the visual concepts can be directly modeled from the text labels. In this way, we can fully utilize the rich cross-domain information to train the proposed DTNs, as well as avoid overfitting.

The related deep neural networks (multimodal deep learning [Ngiam et al. 2011; Sohn et al. 2014; Hong et al. 2015], multimodal learning with DBMs [Srivastava and Salakhutdinov 2012], etc.) share the top layer of modality-specific subnetworks and learn joint representations across multiple modalities. In particular, Wang et al. [2014a] proposed deeply coupled autoencoder networks to learn two deep networks' embedded intraclass compactness and interclass penalty with each other in each layer. Kandaswamy et al. [2014] proposed stacked denoising autoencoders in both supervised and unsupervised ways to improve the effectiveness of the transfer learning. Yang et al. [2015] proposed a boosted multifeature learning approach to iteratively learn multiple representations within a boosting procedure for cross-domain image classification, sentiment classification, and spam filtering. The key difference between the proposed DTNs and existing approaches is that DTNs not only consider weakly shared representations but also consider weakly shared parameters. This scheme makes the model more robust for domain-specific features.

### 3. PROBLEM DEFINITION

In this article, we take the textual domain and visual domain as the source domain and target domain, respectively. For each concept, we have a set of image-text pairs  $\mathcal{C} = \{(\mathcal{T}_i^S, \mathbf{x}_i^T)\}_{i=1}^{N_c}$  (the superscript  $S$  and  $T$  denote the source domain and target domain, respectively, in this work), where the visual feature vector  $\mathbf{x}_i^T \in \mathbb{R}^b$  denotes the image in the  $i$ -th pair and the tag set  $\mathcal{T}_i^S$  includes the tags corresponding to the image  $\mathbf{x}_i^T$ . We use the textual feature  $\mathbf{x}_i^S \in \mathbb{R}^a$  to denote the text in the  $i$ -th pair, where its element  $x_{ij}^S = 1$  if the set  $\mathcal{T}_i^S$  includes the  $j$ -th tag of the predefined tag set  $\mathcal{T} = \bigcup \mathcal{T}_i^S$ ; otherwise,  $x_{ij}^S = 0$ . For  $\mathbf{x}_i^S$ , the corresponding label  $y_i = +1$  if  $\mathcal{T}_i^S$  includes the concept label; otherwise,

$y_i = -1$ . By exploring this set of image-text pairs, the alignment between texts and images can be revealed, which facilitates the label transfer between these two domains. We also have a extremely small training image set  $\mathcal{A}_T = \{(\mathbf{x}_t^T, y_t^T)\}_{t=1}^{N_T}$  for each concept, where  $\mathbf{x}_t^T \in \mathbb{R}^b$  denotes a manually labeled image and  $y_t^T \in \{+1, -1\}$  is its label. Our goal is to transfer the labels from the tag set to the images in the target domain for visual concept classification.

In particular, we jointly learn the deep representations of texts and images to effectively transfer the discriminative information from the source domain to the target domain. The core of the deep transfer learning process is an efficient cross-domain translator function that can transform the labels from the source domain to the target domain even with the challenge of scarcely labeled images in the target domain. By leveraging the learned translator function, we can solve the image classification task with an extremely small training set.

## 4. THE PROPOSED DEEP NETWORKS

### 4.1. The Architecture

For an input vector  $\mathbf{x}_0$ , the formulation of a classical autoencoder is composed 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 a loss function  $loss(\mathbf{x}_0, \tilde{h}(h(\mathbf{x}_0)))$ . Here,  $s_e(\cdot)$  and  $s_d(\cdot)$  denote the nonlinear activation function and decoder's activation function, respectively. Several autoencoders can be consecutively stacked to form the SAEs by feeding the hidden representation of the  $l$ -th autoencoder into the  $(l + 1)$ -th autoencoder.

Built on SAEs, we propose DTNs to transfer knowledge across two domains,<sup>3</sup> as shown in Figure 1. First we pretrain the textual SAE and visual SAE, respectively, which output hidden representations of these two domains via the nonlinear encoding of multiple layers. Formally, there are  $L + 1$  layers in DTNs, where  $L = L_1 + L_2$ . The first  $L_1$  (i.e.,  $L_1 = 3$  in Figure 1) layers of the textual SAE and visual SAE are structured and built separately. After that, the textual SAE and visual SAE begin to share the parameters and representations in the following  $L_2$  (i.e.,  $L_2 = 2$  in Figure 1) shared layers, which are called *generalized layers* in this article. Note that the inputs of the generalized layers are the outputs of the  $L_1$ -th layers of the textual SAE and visual SAE. These generalized layers provide a way to transfer the knowledge across two different domains.

Given a pair of input text  $\mathbf{x}_i^S$  and image  $\mathbf{x}_i^T$ , we use  $\mathbf{x}_{S_i}^{(l)} \in \mathbb{R}^{a_l}$  and  $\mathbf{x}_{T_i}^{(l)} \in \mathbb{R}^{b_l}$  to denote the outputs of the  $l$ -th hidden layers of the textual SAE and visual SAE, respectively. We set  $\mathbf{x}_{S_i}^{(0)} = \mathbf{x}_i^S$  and  $\mathbf{x}_{T_i}^{(0)} = \mathbf{x}_i^T$  as inputs for the two SAEs. For ease 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:

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

<sup>3</sup>It is worth pointing out that the SAE for each domain can be built on another deep network. For example, we can create the CNNs whose output is ingested into the visual SAE. In this article, to avoid notational and illustration clutter, we do not explicitly show this structure.

where  $h_S^{(l)}(\cdot)$  and  $h_T^{(l)}(\cdot)$  denote the  $l$ -th layer hidden representation in textual SAE and visual SAE, 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 textual SAE and visual SAE, respectively.

*Representation-shared scheme.* The first  $L_1$  layers in textual SAE and visual SAE are expected to learn the representations of text and image, respectively, whereas the following  $L_2$  layers provide shared information and structure. Following the basic idea of Wang et al. [2014a, 2014b], we formulate a penalty term to force the representations between the two subnetworks to be similar to each other. Then, providing a set of image-text pairs  $\{\mathbf{x}_i^S, \mathbf{x}_i^T\}_{i=1}^{N_C}$  of input text  $\mathbf{x}_i^S$  and image  $\mathbf{x}_i^T$ , we have

$$\Omega_1 = \sum_{i=1}^{N_C} \sum_{l=L_1+1}^L \left( \|h_S^{(l)}(\mathbf{x}_i^S) - h_T^{(l)}(\mathbf{x}_i^T)\|_2^2 \right), \quad (2)$$

which can measure the difference between outputs of the last  $L_2$  layers of two SAEs.

*Parameter-shared scheme.* We aim to control the parameters of the last  $L_2$  layers to be equal to a certain degree to balance the trade-off between modeling the shared cross-domain features and preserving necessary domain-specific details. Therefore, we define the penalty term

$$\Omega_2 = \sum_{l=L_1+1}^L \left( \|\mathbf{W}_S^{(l)} - \mathbf{W}_T^{(l)}\|_F^2 + \|\mathbf{b}_S^{(l)} - \mathbf{b}_T^{(l)}\|_2^2 \right). \quad (3)$$

Minimizing this term will minimize the difference between parameters of the last  $L_2$  layers in the textual SAE and visual SAE.

*Generalized scheme.* We aim to weakly share the parameters and representations in the proposed architecture. Therefore, we combine the representation-shared penalty in Equation (2) and the parameter-shared penalty in Equation (3) to obtain the generalized penalty corresponding to the generalized scheme as follows:

$$\begin{aligned} \Omega &= \Omega_1 + \Omega_2 \\ &= \sum_{i=1}^{N_C} \sum_{l=L_1+1}^L \left( \|h_S^{(l)}(\mathbf{x}_i^S) - h_T^{(l)}(\mathbf{x}_i^T)\|_2^2 \right) + \sum_{l=L_1+1}^L \left( \|\mathbf{W}_S^{(l)} - \mathbf{W}_T^{(l)}\|_F^2 + \|\mathbf{b}_S^{(l)} - \mathbf{b}_T^{(l)}\|_2^2 \right). \end{aligned} \quad (4)$$

## 4.2. Deep Transfer Mechanism

The goal of the proposed DTNs is to transfer the label information of  $\{\mathcal{T}_i^S\}_{i=1}^{N_C}$  (source domain) to each image  $\mathbf{x}^T$  in the target domain. For this purpose, we define the translator function as an inner product  $(h_S^{(L)}(\mathbf{x}_i^S))' h_T^{(L)}(\mathbf{x}^T)$ , where the prime (') denotes the transpose operation. This translator function is used to propagate the labels from the source domain to the target domain as follows:

$$f(\mathbf{x}^T) = \sum_i^{N_C} y_i^S \left( h_S^{(L)}(\mathbf{x}_i^S) \right)' h_T^{(L)}(\mathbf{x}^T), \quad (5)$$

which combines all source labels weighted by the corresponding translator function. For the binary classification task considered in this article,  $f(\mathbf{x}^T)$  is also a discriminant function, whose sign predicts the label (" +1" or " -1") of the corresponding image  $\mathbf{x}^T$  in the target domain.

We learn the parameters in the proposed DTNs by minimizing the classification loss incurred by the preceding label transfer process and maximizing the consistency of



**ALGORITHM 1:** Training of Generalized DTNs

**Input:**  $\{(\mathbf{x}_i^S, \mathbf{x}_i^T)\}_{i=1}^{N_C}$ ,  $\mathcal{A}_T = \{(\mathbf{x}_t^T, y_t^T)\}_{t=1}^{N_T}$ ,  $L_1$ ,  $L_2$ ,  $\gamma$ ,  $\lambda$ ,  $\mu$ , dimension per layers and maxIter.

**Output:** Parameter set  $\Theta$ .

**Initialization:** Initializing parameters in set  $\Theta$ ,  $iter \leftarrow 1$ ;

// pre-training

**for**  $l = 1, 2, \dots, L$  **do**

    Pretraining textual SAE and visual SAE with inputs  $\{\mathbf{x}_i^S\}_{i=1}^{N_C}$  and  $\{\mathbf{x}_i^T\}_{i=1}^{N_C}$ , respectively.

**end**

Extracting hidden representations on pretrained DTNs for all text and image examples.

// dual fine tuning

**repeat**

**for**  $l = 1, 2, \dots, L$  **do**

        Updating  $\mathbf{W}_S^{(l)}, \mathbf{b}_S^{(l)}$  using Equations (9) and (10);

        Updating  $\mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}$  using Equations (11) and (12);

**end**

**for**  $l = 1, 2, \dots, L_1$  **do**

        Fine tuning  $\mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}$  with the back-propagated errors from the softmax output layer of the visual SAE;

        Updating  $\mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}$  using Equations (11) and (12);

**end**

    Extracting hidden representations on DTNs for all text and image samples;

**until** Convergence;

the predicted labels of the image and text in each pair to capture the cross-domain alignment information. We elaborate on these two criteria next:

—*Empirical loss on the manually labeled small training set.* We have an extremely small training set  $\mathcal{A}_T = \{(\mathbf{x}_t^T, y_t^T)\}_{t=1}^{N_T}$  and aim to minimize the training errors incurred by the translator function  $f$  on this set  $\mathcal{A}_T$ ,

$$\min J_1 = \sum_{t=1}^{N_T} \ell(y_t^T \cdot f(\mathbf{x}_t^T)), \quad (6)$$

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

—*Empirical loss on the image-text pairs.* Provided with a set of image-text pairs  $\{(\mathbf{x}_i^S, \mathbf{x}_i^T)\}_{i=1}^{N_C}$ , we maximize the alignment between the image and text in each pair by minimizing the objective function

$$\min J_2 = \sum_{i=1}^{N_C} \chi\left(\left(h_S^{(L)}(\mathbf{x}_i^S)\right)' h_T^{(L)}(\mathbf{x}_i^T)\right), \quad (7)$$

where  $\chi(x) = \exp(-x)$  is an exponential loss function.

### 4.3. Objective Function and Optimization

Following the analysis in the last section, we can minimize an objective function  $J$  to learn the parameter set  $\Theta = \{\mathbf{W}_S^{(l)}, \mathbf{b}_S^{(l)}, \mathbf{W}_T^{(l)}, \mathbf{b}_T^{(l)}\}_{l=1}^L$  of DTNs

$$\arg \min_{\Theta} J = J_1 + \eta J_2 + \frac{\gamma}{2} \Omega + \frac{\lambda}{2} \Psi, \quad (8)$$

**ALGORITHM 2:** Testing Phase**Input:** Testing image  $\mathbf{x}$ , trained parameter set  $\Theta$ .**Output:** Concept label  $y$  of image of  $\mathbf{x}$ .Extracting the representation  $h_T^{(L)}(\mathbf{x})$  of  $\mathbf{x}$  in set  $\Theta$ ;  
Predicting  $y$  by Equation (5).

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 to avoid overfitting. The parameters  $\eta$ ,  $\gamma$ , and  $\lambda$  weight the importance of the alignment between the image and text in each pair, the generalized-shared penalty, and the overfitting regularization, respectively.

To train the proposed DTNs, we first take each layer as an autoencoder and pretrain in a greedy fashion. Then, the pretrained DTNs are fine tuned according to the objective function (8) by employing the available supervision information. In the fine-tuning step, we implement a back-propagation process starting from the top output layers down through the whole DTNs to adjust all parameters. Each parameter in  $\Theta$  is updated by stochastic gradient descent in the back-propagation algorithm as follows:

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

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

$$\mathbf{W}_T^{(l)} = \mathbf{W}_T^{(l)} - \mu \frac{\partial J}{\partial \mathbf{W}_T^{(l)}}, \quad (11)$$

$$\mathbf{b}_T^{(l)} = \mathbf{b}_T^{(l)} - \mu \frac{\partial J}{\partial \mathbf{b}_T^{(l)}}, \quad (12)$$

where  $\mu$  is the learning rate. 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 generalized layers can be computed as follows:

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

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

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

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

$$\frac{\partial J}{\partial \mathbf{b}_T^{(l)}} = \sum_{t=1}^{N_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_T^{(l)}} + \lambda \mathbf{b}_T^{(l)}; \quad (16)$$

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

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

Table II. Various Versions of the Proposed DTNs in This Article

Name	Single Fine Tuning	Dual Fine Tuning	Representation-Shared Scheme	Parameter-Shared Scheme
sig-DTNs	✓			✓
duft-DTNs		✓		✓
sig-tDTNs	✓		✓	✓
duft-tDTNs		✓	✓	✓

Note: Sig-DTNs and Duft-DTNs were originally proposed in the preliminary work [Shu et al. 2015].

$$\frac{\partial J}{\partial \mathbf{b}_S^{(l)}} = \sum_{t=1}^{N_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_S^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_S^{(l)}} + \gamma \left( \sum_{i=1}^{N_C} \frac{\partial \kappa}{\partial h_S^{(l)}(\cdot)} \cdot \frac{\partial h_S^{(l)}(\cdot)}{\partial \mathbf{b}_S^{(l)}} + \mathbf{b}_S^{(l)} - \mathbf{b}_T^{(l)} \right) + \lambda \mathbf{b}_S^{(l)}, \quad (18)$$

$$\frac{\partial J}{\partial \mathbf{w}_T^{(l)}} = \sum_{t=1}^{N_T} \frac{\partial \ell(u_t)}{\partial \mathbf{w}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{w}_T^{(l)}} + \gamma \left( \sum_{i=1}^{N_C} \frac{\partial \kappa}{\partial h_S^{(l)}(\cdot)} \cdot \frac{\partial h_S^{(l)}(\cdot)}{\partial \mathbf{w}_T^{(l)}} + \mathbf{w}_T^{(l)} - \mathbf{w}_S^{(l)} \right) + \lambda \mathbf{w}_T^{(l)}, \quad (19)$$

$$\frac{\partial J}{\partial \mathbf{b}_T^{(l)}} = \sum_{t=1}^{N_T} \frac{\partial \ell(u_t)}{\partial \mathbf{b}_T^{(l)}} + \eta \sum_{i=1}^{N_C} \frac{\partial \chi(v_i)}{\partial \mathbf{b}_T^{(l)}} + \gamma \left( \sum_{i=1}^{N_C} \frac{\partial \kappa}{\partial h_S^{(l)}(\cdot)} \cdot \frac{\partial h_S^{(l)}(\cdot)}{\partial \mathbf{b}_T^{(l)}} + \mathbf{b}_S^{(l)} - \mathbf{b}_T^{(l)} \right) + \lambda \mathbf{b}_T^{(l)}. \quad (20)$$

We define  $\kappa = \|h_S^{(l)}(\mathbf{x}_i^S) - h_T^{(l)}(\mathbf{x}_i^T)\|_2^2$ ,  $u_t = y_t^T \cdot f(\mathbf{x}_t^T)$  and  $v_i = (h_S^{(L)}(\mathbf{x}_i^S))' h_T^{(L)}(\mathbf{x}_i^T)$  in Equations (13) through (20). The derivatives of  $\kappa$ ,  $\ell(u_t)$ , and  $\chi(v_i)$  over the network parameters can be computed in a similar back-propagation fashion of the conventional neural networks. For simplicity, the preceding fine-tuning strategy is called *single fine tuning* in this article.

Single fine tuning is more effective for updating the textual SAE than the visual SAE, as the concept labels are more directly related to the textual features. To adequately tune the visual SAE in conjunction with the tuning of the textual SAE, we add an additional softmax layer on the visual SAE that outputs the image labels, as shown in Figure 1. Then supervision information in  $\mathcal{A}_T$  is exploited again to incrementally fine tune the parameters in the visual SAE.

We implement the fine tuning by alternately minimizing the loss of objective function  $J$  and the loss of the softmax function on the top of the visual SAE, respectively. In contrast to single fine tuning, we refer to such a fine-tuning strategy as *dual fine tuning* in this article. We extend the generalized DTNs to two versions by using different fine-tuning strategies, namely sig-tDTNs (using single fine tuning) and duft-tDTNs (using dual fine tuning). The configuration difference of sig-tDTNs and duft-tDTNs with sig-DTNs and duft-DTNs [Shu et al. 2015] is shown in Table II.

We compare these various versions of DTNs in Section 5. The detailed training procedure of dual fine-tuned DTNs (duft-tDTNs) is described in Algorithm 1, whereas the testing procedure for image classification is described in Algorithm 2. The convergence criterion is that the iteration steps stop when the relative cost of the objective function is smaller than a predefined threshold.

## 5. EXPERIMENTS









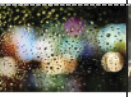


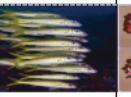



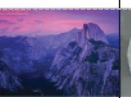


### 5.1. Datasets

To evaluate the effectiveness of the proposed DTNs, we released a new dataset extended from the widely used NUS-WIDE [Chua et al. 2009]. The original NUS-WIDE contains 269,648 images, the corresponding 5,018 textual tags collected from

Table III. Supplemented 47 Concepts in the NUS-WIDE-128 Dataset Compared to the Previous Version [Chua et al. 2009]

<i>architecture</i>	<i>art</i>	<i>bicycles</i>	<i>boys</i>	<i>bus</i>	<i>butterfly</i>	<i>chair</i>	<i>children</i>
<i>church</i>	<i>deer</i>	<i>door</i>	<i>dragons</i>	<i>elephants</i>	<i>farm</i>	<i>fence</i>	<i>field</i>
<i>fireworks</i>	<i>football</i>	<i>girls</i>	<i>hills</i>	<i>ice</i>	<i>indoor</i>	<i>island</i>	<i>landscape</i>
<i>man</i>	<i>museum</i>	<i>nature</i>	<i>outdoor</i>	<i>park</i>	<i>pets</i>	<i>portrait</i>	<i>rain</i>
<i>restaurant</i>	<i>rivers</i>	<i>room</i>	<i>school</i>	<i>seals</i>	<i>ships</i>	<i>smoke</i>	<i>telephone</i>
<i>truck</i>	<i>village</i>	<i>watch</i>	<i>winter</i>	<i>woman</i>	<i>wood</i>	<i>zoo</i>	

Table IV. Exhibition of Image-Text Pairs With Top-10 High-Frequency Tags for Some Concepts

Concept	<i>park</i>	<i>smoke</i>	<i>rain</i>	<i>cat</i>	<i>mountain</i>	<i>fish</i>	<i>ice</i>
Image-text pair	 Park, city, blue, vacation, sky, people, lifestyle, happy, play, outdoor	 smoke, newyork, man, night, dark, fire, bravo	 rain, road, street, man, black, color, wet, yellow, umbrella, wind	 cat, cute, smile, face, look, laughing, pose, happy, kitten feline	 mountain, tree, blue, mountains alps, color, clouds, bravo	 fish, orange, pet, fish, eyes, kiss, spice, bubble	 ice, winter, red, Italy, white, cold, colour, sport, team, game
Image-text pair	 Park, usa, chicago, tower, car, architecture, parkinglot, skyscraper	 smoke, woman, white, abstract, black, dancing, flash	 rain, water, lights, airport, colour, sydney, raindrop, wet	 cat, animal, zoo, interestingness, tiger, wildlife	 mountain, blue, sky, white, mountain, snow, sport, clouds	 fish, school, beautiful, yellow, wow, cool, nice, water	 ice, cake, sweet, studio, food chocolate, icecream, sell, food
Image-text pair	 park, green, home, nature, public, field, grass, outside, land	 smoke, smoking, fire, light, brazil, man, color, hand, art	 rain, sky, storm, tree, weather clouds, wind, light, thunder	 cat, tiger, animal, wow, nose, zoo, eyes	 mountain, pink, sunset, sky, horizontal, view cloud, landscape, evening, bravo,	 fish, food, delicious, sea, dish, bravo, taste, ocean, market, japan	 Ice, cold, icecream, teeth, food, cheer, eyes, fun, drawing
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Flickr.com, and 81 manually annotated concepts. To make this dataset more applicable, we employ several annotators to manually label an additional 47 concepts, as shown in Table III. For each concept, all 269,648 images are labeled by one annotator first. Then, the results are checked by two other annotators. Finally, we have ground truth for 128 concepts. We call this dataset NUS-WIDE-128, and we released it at <http://imag.njust.edu.cn/NUS-WIDE-128.html>.

The data used in the experiments is selected from NUS-WIDE-128 and ImageNet [Deng et al. 2009]. For many concepts in NUS-WIDE-128, the image-text pairs are very few. Thus, among the 128 concepts, we selected 21 representative ones to conduct experiments for image classification, including *birds*, *building*, *cars*, *cat*, *dog*, *fish*, *flowers*, *horses*, *mountain*, *plane*, *dragons*, *outdoor*, *park*, *fireworks*, *village*, *ice*, *rain*, *rivers*, *smoke*, *art*, and *landscape*. For each concept, we collected 3,000 image-text pairs, of which 1,500 pairs are relevant to this concept and the rest of the pairs are irrelevant. We show the exemplary image-text pairs in Table IV (only the top-10 high-frequency words are selected for the table due to space limitations). We can see that some of the text is incorrect or ambiguous when describing the corresponding image. For example, in the fourth column of Table IV, the second and third images are related to “tiger”

Table V. Detailed Construction of SD and CD in This Article

Name	Training data		Testing data
	Text-Image co-occurrence pairs (# 3,000 for each concept)	Labeled images (# 10 for each concept)	Images with ground truth (# 1,000 for each concept)
Single dataset (SD)	From NUS-WIDE-128	From NUS-WIDE-128	From NUS-WIDE-128
Cross dataset (CD)	From NUS-WIDE-128	From ImageNet	From ImageNet

(a) *Birds* images from NUS-WIDE-128(b) *Birds* images from ImageNetFig. 2. Examples of *birds* images from NUS-WIDE-128 (a) and ImageNet (b).

but are tagged with “cat.” This observation is also discussed in Tang et al. [2016]. In addition, for each concept, we collected 10 labeled images from either NUS-WIDE-128 or ImageNet for training. The testing images are from either NUS-WIDE-128 or ImageNet as well.

Finally, we constructed two types of datasets, as shown in Table V, to evaluate the proposed DTNs:

- Single Dataset (SD)*: We used the collected training data from NUS-WIDE-128 to train the DTNs. Then we used the trained DTNs to transfer the textual labels to annotate the testing images in NUS-WIDE-128. For each concept, the training set contains 3,000 image-text pairs and 10 labeled images, whereas the testing set contains 1,000 images from NUS-WIDE-128. This is a challenge for image classification, as the images on Flickr.com are uploaded by amateur photographers. These images have large differences in resolution sizes, and many of them contain a cluttered background.
- Cross Dataset (CD)*: In this setting, we trained the DTNs with the collected training data from NUS-WIDE-128 as well. But we used the trained DTNs to transfer the textual labels from NUS-WIDE-128 to annotate the images in ImageNet. For each concept, the training set contains 3,000 image-text pairs from NUS-WIDE-128 and 10 labeled images from ImageNet, whereas the testing set contains 1,000 images from ImageNet. This setting can test the generalization ability of DTNs across different image datasets. Figure 2 shows some examples of *birds* images from NUS-WIDE-128 and ImageNet. We can see that the images from NUS-WIDE-128 are visually diverse, whereas those images of the same concept from ImageNet are more visually consistent.



## 5.2. Comparing Methods

We compare the proposed generalized DTNs (sig-tDTNs and duft-tDTNs) to the following methods:

- (1) *SVM*: SVM is the conventional shallow structured classifier [Zhang et al. 2015a] and is set as the baseline for comparisons.
- (2) *SAE*: SAEs learn the image representations, and we connect them to a logistic output layer that predicts labels of the testing images. Both SVMs and SAEs only use the manually labeled training set, without any image-text pair involved. We also compare the proposed DTNs with other transfer learning methods.
- (3) *HTL* [Zhu et al. 2011]: HTL maps each image and the text into a latent vector space via a formulated implicit distance function. HTL also makes use of the occurrence information between images and texts.
- (4) *Translator from text to image (TTI)* [Qi et al. 2011]: TTI learns a translator on image-text pairs and a small size of labeled image set. Then it effectively converts the semantic information from texts to images for the image classification task. Both HTL and TTI are proposed to transfer the label information from the text set to the image set.
- (5) *Parameter-shared DTNs (two versions: sig-DTNs and duft-DTNs)* [Shu et al. 2015]: sig-DTNs and duft-DTNs can transfer information across heterogeneous domains (i.e., from the textual domain to the visual domain). In particular, the parameter-shared layers capture the complex representation of data in different domains by utilizing both shared interdomain and domain-specific knowledge.

For each concept, we compare the performance of different methods to varying numbers of training images, ranging from 2 to 10. The same number of images from other concepts are used as the negative samples. In SVMs and SAEs, the image-text pairs do not join the training process. The whole process is repeated 10 times, and the average accuracy is reported. We also analyze the accuracy of the proposed DTNs with varying numbers of image-text pairs. We adopt the twofold cross validation to tune the parameters. We run the validation process 10 times and choose the parameters that achieve the best average validation accuracy.

## 5.3. Results

In this work, we train five-layer DTNs, in which the number of neurons at each layer is  $1,226 \rightarrow 618 \rightarrow 128 \rightarrow 128 \rightarrow 60$  and  $1,000 \rightarrow 512 \rightarrow 128 \rightarrow 128 \rightarrow 60$  from the bottom up in the visual SAE and textual SAE, respectively. The last two layers are the generalized layers. In the proposed DTNs, 1,000 words are extracted and stemmed from the textual parts, and their frequencies are input into the textual SAE. In addition, we extract 4,096-dim CNN features by AlexNet [Krizhevsky et al. 2012; Jia et al. 2014] for the images. To keep the numbers of the input neurons of the visual SAE and textual SAE close, as well as to keep the cumulative energy above 85%, we use PCA to reduce the image features from 4,096 dimensions to 1,226 dimensions as input of the visual SAE.

The accuracies of different methods over all concepts with varying numbers of manually labeled images are shown in Figures 3 and 4. Either on SD or CD, the accuracies obtained by the traditional SVM and SAE methods are lower than the other compared methods, no matter how many training images are used, as neither of them explores the textual information. Among all of these methods, the proposed DTNs perform the best.

The subplots in Figures 3 and 4 show the accuracy of the 21 concepts. We can see that the DTNs outperform other compared methods, especially when there is an extremely

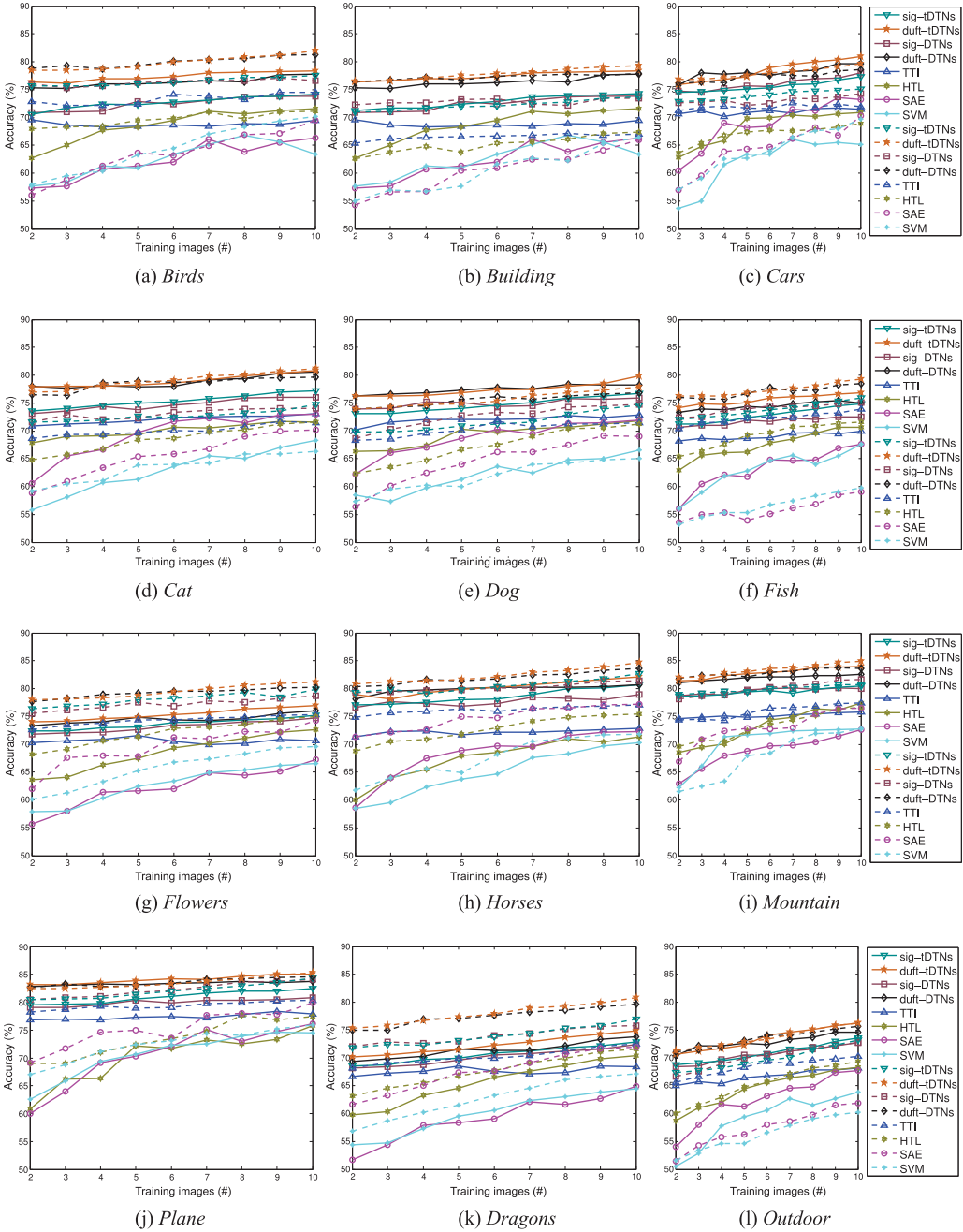


Fig. 3. Accuracy (%) of different methods with the number of auxiliary labeled images. The solid line and dotted line denote the results on the SD and CD datasets, respectively.

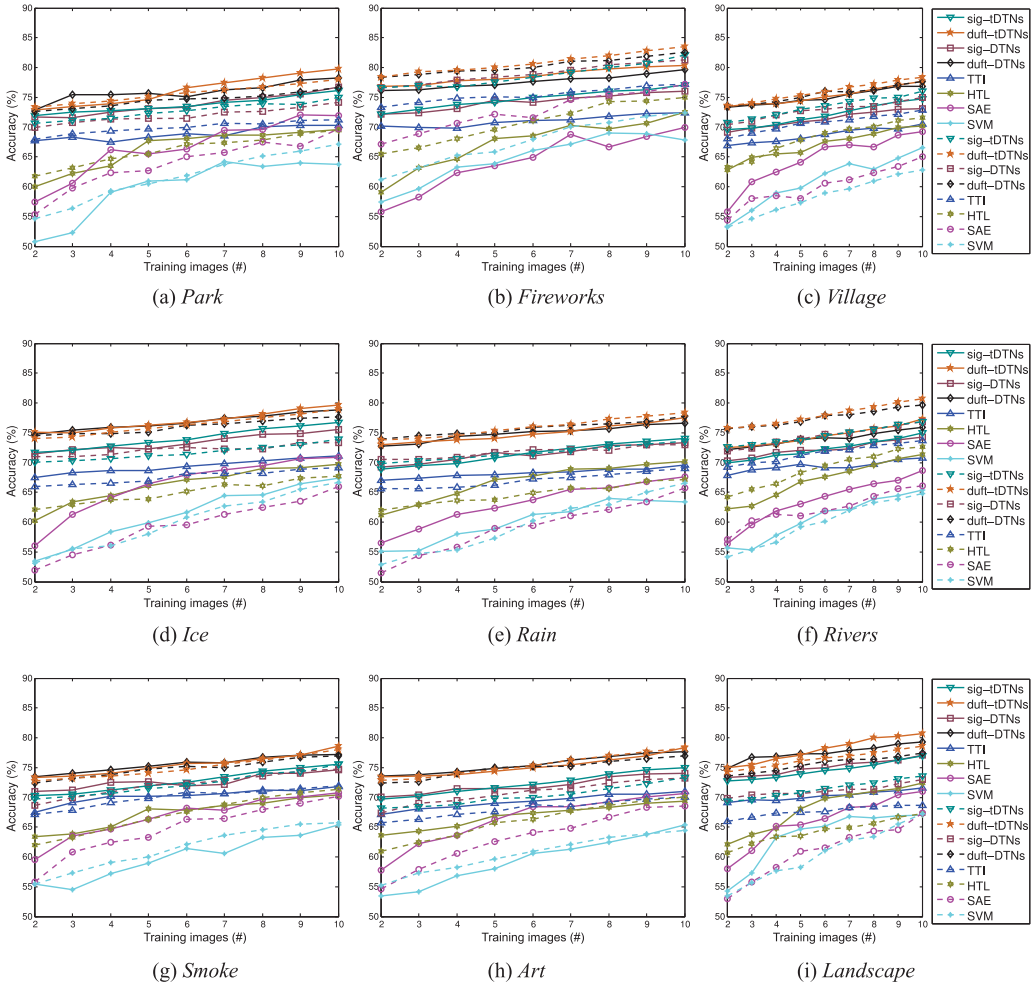


Fig. 4. Accuracy (%) of different methods with the number of auxiliary labeled images. The solid line and dotted line denote the results on the SD and CD datasets, respectively.

small amount of training data. This is also consistent with our earlier assertions that the proposed DTNs can work well even in the case when there is an insufficient amount of prelabeled training samples by utilizing the co-occurrence information between texts and images. We also observe that DTNs with dual fine tuning (i.e., duft-tDTNs and duft-DTNs) perform better than the corresponding DTNs with single fine tuning (i.e., sig-tDTNs and sig-DTNs). This is because dual fine tuning uses an extra step to tune the visual SAE with the training images. This better balances the trade-off between the fine tuning of the textual SAE and the visual SAE.

Table VI also reports the average accuracy of various methods over all 21 concepts with varying numbers of training images. Results in both experimental settings are reported in this table. We can see that duft-tDTNs achieve better performance than other methods and other variants of DTNs. In particular, an interesting discovery is that the accuracy of DTNs on the cross dataset (CD) is comparable to the accuracy of DTNs on the single dataset (SD). This suggests that DTNs trained with one dataset (NUS-WIDE-128) are well generalized to transfer the label information between

Table VI. Average Accuracy (%) of Various Methods Versus the Number of Training Images on SD and CD, Respectively

Training Images (#)	Single Dataset (SD)							
	SVM	SAE	HTL	TTI	sig-DTN	duft-DTN	sig-tDTN	duft-tDTN
2	55.92	57.	62.	69.27	72.16	74.91	72.19	<b>75.26</b>
3	57.29	61.01	64.39	69.74	72.46	<b>75.60</b>	72.54	75.47
4	60.83	63.97	65.70	69.78	72.98	75.95	73.03	<b>76.46</b>
5	62.04	64.91	67.87	70.32	73.59	76.35	73.51	<b>76.86</b>
6	63.67	66.28	68.80	70.32	73.62	76.46	74.06	<b>77.52</b>
7	64.99	67.89	69.66	70.39	74.23	76.88	74.63	<b>78.08</b>
8	65.59	67.83	70.11	70.88	74.78	77.24	75.23	<b>78.70</b>
9	66.21	69.22	70.75	71.09	75.04	77.91	75.70	<b>79.16</b>
10	66.60	69.99	71.41	71.39	75.45	78.13	76.23	<b>79.76</b>

Training Images	Cross Dataset (CD)							
	SVM	SAE	HTL	TTI	sig-DTN	duft-DTN	sig-tDTN	duft-tDTN
2	56.75	58.06	64.19	69.51	72.48	75.87	72.62	<b>76.58</b>
3	58.51	61.02	65.32	70.10	73.05	76.22	72.91	<b>76.87</b>
4	59.91	62.73	66.45	70.54	73.56	76.73	73.19	<b>77.36</b>
5	61.23	64.04	67.56	71.01	74.07	77.23	73.79	<b>77.85</b>
6	63.02	64.95	68.61	71.53	74.51	77.85	74.23	<b>78.42</b>
7	64.60	66.17	69.51	71.74	74.81	77.98	74.78	<b>79.04</b>
8	65.46	67.46	70.18	72.04	75.17	78.33	75.41	<b>79.57</b>
9	66.50	68.21	70.95	72.47	75.75	78.85	75.82	<b>80.17</b>
10	67.27	69.59	71.40	72.70	76.04	79.20	76.63	<b>80.74</b>

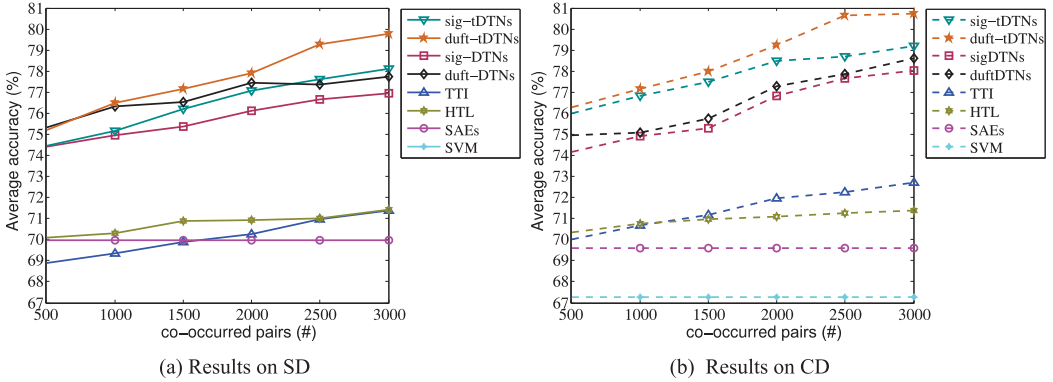


Fig. 5. Number of image-text pairs versus average accuracy (%) on SD and CD, respectively.

different datasets (i.e., from NUS-WIDE-128 to ImageNet). This is a very useful property in many real applications when we have to handle data from different sources.

The preceding results of DTNs are obtained by using 3,000 image-text pairs. Since the image-text pairs play an important role in connecting heterogeneous domains in the proposed DTNs, we examine the performance of the proposed DTNs with varying numbers of image-text pairs (i.e., {500, 1,000, 1,500, 2,000, 2,500, 3,000}). The number of pre-labeled training images is fixed to 10. The results are shown in Figure 5. We can see that the average accuracy of TTI and DTNs (sig-DTNs, duft-DTNs, sig-tDTNs, and duft-tDTNs) on both SD and CD are increased with the increment of the image-text pairs. This suggests that more pairs tend to provide more information to better model DTNs.

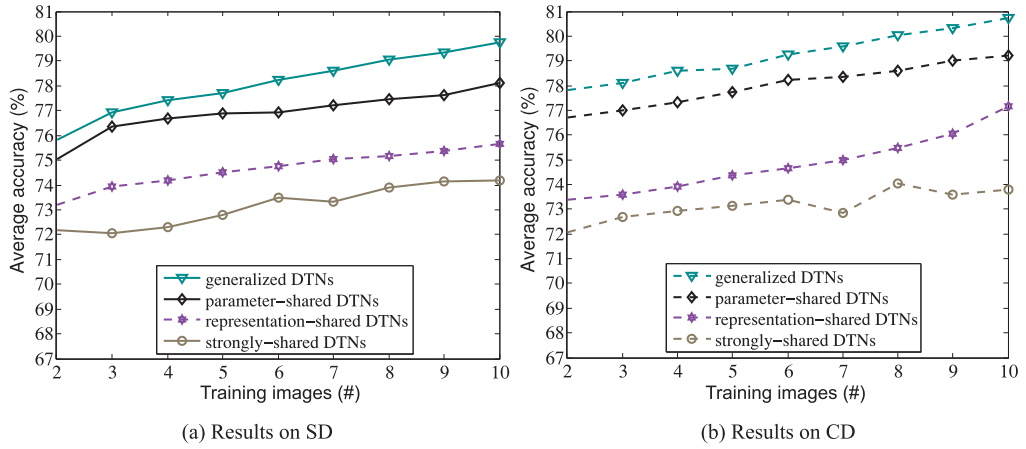


Fig. 6. Comparison results of generalized DTNs with generalized layers, parameter-shared DTNs, representation-shared DTNs, and strongly shared DTNs.

#### 5.4. Discussion on the Generalized Layers

To illustrate the superiority of the generalized DTNs, we also compare the performance of the generalized DTNs, the parameter-shared DTNs, the representation-shared DTNs, and the strongly shared DTNs. For strongly shared DTNs, we set  $\mathbf{W}_S^{(l)} = \mathbf{W}_T^{(l)}$  and  $\mathbf{b}_S^{(l)} = \mathbf{b}_T^{(l)}$  for the textual SAE and visual SAE in 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 performance of these four variants of DTNs with different layer-shared schemes. We can see that the strongly shared DTNs gain the lowest average accuracy, and the average accuracy of the generalized DTNs are higher than other ones on both SD and CD. This confirms that the equipped generalized layers are more generalized for modeling the representation of heterogeneous data than the parameter-shared layers, the representation-shared layers, and the strongly shared layers. In addition, the performance of the parameter-shared DTNs is more competitive than the representation-shared DTNs on both SD and CD. This illustrates that the parameter-shared layers are more suitable to model DTNs than the representation-shared layers.

#### 5.5. Parameter Sensitivity

In the experiments, parameters  $\gamma$  and  $\lambda$  of objective function  $J$  in Equation (8) are chosen from  $\gamma \in \{0, 0.5, 1.0, 2.0\}$  and  $\lambda \in \{0.1, 0.5, 1.0, 2.0\}$ , respectively, by a cross-validation procedure. Conventionally, we set  $\eta = 1$  to equally weight the two types of loss in Equation (8). Here, we study their impact on performance in Figure 7. When  $\gamma = 0$ , the average accuracy is the lowest, because the textual SAE and visual SAE are completely independent without any shared layers. This structure fails to model the joint visual and textual representation, and it is unable to transfer labels across heterogeneous domains. In addition,  $\lambda$  can also improve the accuracy when it is set to a proper value to regularize the model. The best accuracy on both SD and CD is achieved when  $\gamma = 1$  and  $\lambda = 0.5$ . The accuracy with different values of parameters do not vary too much. This suggests that DTNs are insensitive to the parameters.



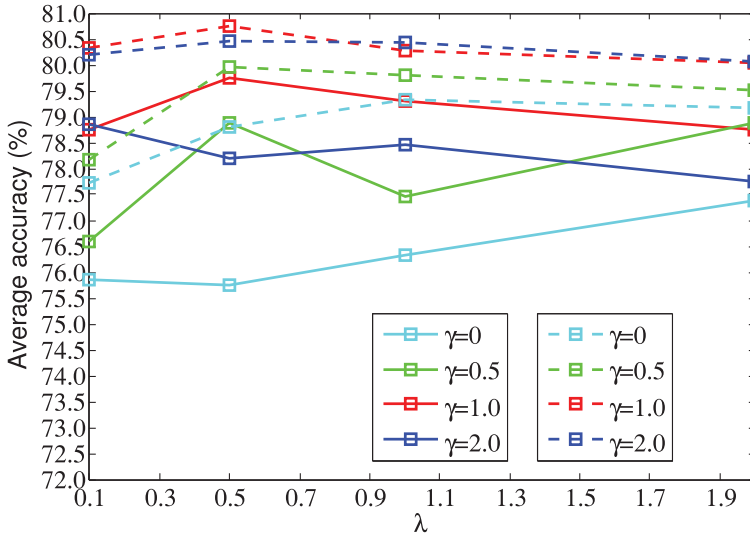


Fig. 7. Parametric sensitivity versus average accuracy (%) with different parameters  $\gamma$  and  $\lambda$  on 10 training images. The solid line and dotted line denote results on SD and CD, respectively.

## 6. CONCLUSIONS AND FUTURE WORK

In this article, we proposed a type of novel DTN equipped with several generalized layers to transfer the cross-domain information from the textual domain (i.e., source domain) to the visual domain (i.e., target domain). The equipped generalized layers have more powerful ability to capture complex representation of data from different domains with both shared interdomain and domain-specific knowledge than the strongly shared layers, parameter-shared layers, and representation-shared layers. The proposed DTNs are trained in a novel way that directly minimizes the loss incurred by the translator function. We also proposed a dual fine tuning strategy that is implemented by alternately minimizing the loss of objective function and the loss of softmax function on the top of visual SAE, respectively. To evaluate the proposed DTNs, a new dataset extended from NUS-WIDE has been released at <http://imag.njust.edu.cn/NUS-WIDE-128.html>. We show the superior results of the proposed DTNs with various versions over baselines and other state-of-the-art methods.

In future work, we will consider two directions. First, we will explore the end-to-end fine-tuning strategy. Specifically, we will add several convolutional layers and pooling layers under the bottom of the visual SAE, which are alternately stacked. Thus, the visual SAE becomes the CNN architecture, which can be fine tuned from the objective function to the input images in an end-to-end way. Second, we will use more data from multiple source domains. We can propagate the semantic knowledge from as least two types of source domain data to the target-domain data. Thus, how to design the effective multiple types of subnetworks with the representation-shared scheme and parameter-shared scheme is the main challenge.

## REFERENCES

- Jimmy Ba and Brendan Frey. 2013. Adaptive dropout for training deep neural networks. In *Advances in Neural Information Processing Systems 26 (NIPS'13)*.
- Yoshua Bengio. 2009. Learning deep architectures for AI. *Foundations and Trends in Machine Learning* 2, 1, 1–127.

- Yoshua Bengio. 2012. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of the International Conference on Machine Learning (ICML'12)*. 17–36.
- Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. 2012. Unsupervised feature learning and deep learning: A review and new perspectives. arXiv:1206.5538v1.
- Minmin Chen, Zhixiang Xu, Fei Sha, and Kilian Q. Weinberger. 2012. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML'12)*.
- Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. 2009. NUS-WIDE: A real-world Web image database from National University of Singapore. In *Proceedings of the Conference on Image and Video Retrieval (CIVR'09)*.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'09)*.
- Jeff Donahue, Judy Hoffman, Erik Rodner, Kate Saenko, and Trevor Darrell. 2013. Semi-supervised domain adaptation with instance constraints. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'13)*.
- Lixin Duan, Dong Xu, and Ivor W. Tsang. 2012. Learning with augmented features for heterogeneous domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML'12)*.
- Fangxiang Feng, Ruifan Li, and Xiaojie Wang. 2015. Deep correspondence restricted Boltzmann machine for cross-modal retrieval. *Neurocomputing* 154, 50–60.
- Fangxiang Feng, Xiaojie Wang, and Ruifan Li. 2014. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM'14)*. 7–16.
- Shenghua Gao, Yuting Zhang, Kui Jia, Jiwen Lu, and Yingying Zhang. 2015. Single sample face recognition via learning deep supervised autoencoders. *IEEE Transactions on Information Forensics and Security* 10, 10, 2108–2118.
- Efstathios Gavves, Thomas Mensink, Tatiana Tommasi, Cees G. M. Snoek, and Tinne Tuytelaars. 2015. Active transfer learning with zero-shot priors: Reusing past datasets for future tasks. In *Proceedings of the International Conference on Computer Vision (ICCV'15)*.
- Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the International Conference on Machine Learning (ICML'11)*.
- Geoffrey Hinton, Simon Osindero, and Yee-Whye Teh. 2006. A fast learning algorithm for deep belief nets. *Neural Computation* 18, 7, 1527–1554.
- Chaoqun Hong, Jun Yu, Jian Wan, Dacheng Tao, and Meng Wang. 2015. Multimodal deep autoencoder for human pose recovery. *IEEE Transactions on Image Processing* 24, 12, 5659–5670.
- Xu Jia, Efstathios Gavves, Basura Fernando, and Tinne Tuytelaars. 2015. Guiding long-short term memory for image caption generation. arXiv:1509.04942.
- Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM'14)*. 675–678.
- Yu-Gang Jiang, Chong-Wah Ngo, and Shih-Fu Chang. 2009. Semantic context transfer across heterogeneous sources for domain adaptive video search. In *Proceedings of the 17th ACM International Conference on Multimedia (MM'09)*. 155–164.
- Alexander Kalmanovich and Gal Chechik. 2014. Gradual training of deep denoising auto encoders. arXiv:1412.6257.
- Meina Kan, Shiguang Shan, Hong Chang, and Xilin Chen. 2014. Stacked progressive auto-encoders (SPA-E) for face recognition across poses. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'14)*.
- Chetak Kandaswamy, Lynette M. Silva, Luis Alexandre, Ricardo Sousa, Jorge M. Santos, and Joaquim Marques de Sá. 2014. Improving transfer learning accuracy by reusing stacked denoising autoencoders. In *Proceedings of the International Conference on Systems, Man, and Cybernetics (SMC'14)*. 1380–1387.
- Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'15)*.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25 (NIPS'12)*.
- Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553, 436–444.
- Zechao Li, Jing Liu, Jinhui Tang, and Hanqing Lu. 2015. Robust structured subspace learning for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, 10, 2085–2098.

- Zechao Li and Jinhui Tang. 2015. Weakly supervised deep metric learning for community-contributed image retrieval. *IEEE Transactions on Multimedia* 17, 11, 1989–1999.
- Mingsheng Long, Jianmin Wang, Guiguang Ding, Jianguang Sun, and Philip S. Yu. 2014. Transfer joint matching for unsupervised domain adaptation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'14)*.
- Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. 2011. Multimodal deep learning. In *Proceedings of the International Conference on Machine Learning (ICML'11)*.
- Jie Ni, Qiang Qiu, and Rama Chellappa. 2013. Subspace interpolation via dictionary learning for unsupervised domain adaptation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'13)*.
- Xinyu Ou, Lingyu Yan, Hefei Ling, Cong Liu, and Maolin Liu. 2014. Inductive transfer deep hashing for image retrieval. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM'14)*. 969–972.
- Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10, 1345–1359.
- Guo-Jun Qi, Charu Aggarwal, and Thomas Huang. 2011. Towards semantic knowledge propagation from text corpus to Web images. In *Proceedings of the International Conference on World Wide Web (WWW'11)*.
- Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. 2007. Self-taught learning: Transfer learning from unlabeled data. In *Proceedings of the International Conference on Machine Learning (ICML'07)*.
- Antti Rasmus, Harri Valpola, and Tapani Raiko. 2015. Lateral connections in denoising autoencoders support supervised learning. arXiv:1504.08215.
- Suman Deb Roy, Tao Mei, Wenjun Zeng, and Shipeng Li. 2012. SocialTransfer: Cross-domain transfer learning from social streams for media applications. In *Proceedings of the 20th ACM International Conference on Multimedia (MM'12)*. 649–658.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, et al. 2014. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 3, 211–252.
- Ruslan Salakhutdinov and Geoffrey E. Hinton. 2009. Deep Boltzmann machines. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS'09)*.
- Xiangbo Shu, Guo-Jun Qi, Jinhui Tang, and Jingdong Wang. 2015. Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation. In *Proceedings of the 23rd ACM International Conference on Multimedia (MM'15)*. 35–44.
- Richard Socher, Milind Ganjoo, Christopher D. Manning, and Andrew Ng. 2013. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems 26 (NIPS'13)*.
- Kihyuk Sohn, Wenling Shang, and Honglak Lee. 2014. Improved multimodal deep learning with variation of information. In *Advances in Neural Information Processing Systems 27 (NIPS'14)*.
- Xiaonan Song, Jianguang Zhang, Yahong Han, and Jianmin Jiang. 2016. Semi-supervised feature selection via hierarchical regression for Web image classification. *Multimedia Systems* 22, 1, 41–49.
- Nitish Srivastava and Ruslan Salakhutdinov. 2012. Multimodal learning with deep Boltzmann machines. In *Advances in Neural Information Processing Systems 25 (NIPS'12)*.
- Jinhui Tang, Richang Hong, Shuicheng Yan, Tat-Seng Chua, Guo-Jun Qi, and Ramesh Jain. 2011. Image annotation by kNN-sparse graph-based label propagation over noisily tagged Web images. *ACM Transactions on Intelligent Systems and Technology* 2, 2, 14.
- Jinhui Tang, Lu Jin, Zechao Li, and Shenghua Gao. 2015. RGB-D object recognition via incorporating latent data structure and prior knowledge. *IEEE Transactions on Multimedia* 17, 11, 1899–1908.
- Jinhui Tang, Xiangbo Shu, Guo-Jun Qi, Zechao Li, Meng Wang, Shuicheng Yan, and Ramesh Jain. 2016. Tri-clustered tensor completion for social-aware image tag refinement. *IEEE Transaction on Pattern Analysis and Machine Intelligence* PP, 99, 1. DOI: <http://dx.doi.org/10.1109/TPAMI.2016.2608882>
- Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the International Conference on Machine Learning (ICML'08)*.
- Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research* 11, 3371–3408.
- Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'15)*.

- Wen Wang, Zhen Cui, Hong Chang, Shiguang Shan, and Xilin Chen. 2014a. Deeply coupled auto-encoder networks for cross-view classification. arXiv:1402.2031.
- Wei Wang, Beng Chin Ooi, Xiaoyan Yang, Dongxiang Zhang, and Yueting Zhuang. 2014b. Effective multi-modal retrieval based on stacked auto-encoders. *Proceedings of the VLDB Endowment* 7, 8, 1–12.
- Jason Weston, Frédéric Ratle, Hossein Mobahi, and Ronan Collobert. 2012. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*. Springer, 639–655.
- Pei Xu, Mao Ye, Xue Li, Qihe Liu, Yi Yang, and Jian Ding. 2014. Dynamic background learning through deep auto-encoder networks. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM'14)*. 107–116.
- Xiaoshan Yang, Tianzhu Zhang, Changsheng Xu, and Ming-Hsuan Yang. 2015. Boosted multifeature learning for cross-domain transfer. *ACM Transactions on Multimedia Computing, Communications, and Applications* 11, 3, 35.
- Xu Zhang, Felix Kinnam Yu, Shih-Fu Chang, and Shengjin Wang. 2015b. Deep transfer network: Unsupervised domain adaptation. arXiv:1503.00591.
- Yi Zhang, Jinchang Ren, and Jianmin Jiang. 2015a. Combining MLC and SVM classifiers for learning based decision making: Analysis and evaluations. *Computational Intelligence and Neuroscience* 2015, Article No. 44.
- Yin Zhu, Yuqiang Chen, Zhongqi Lu, Sinno Jialin Pan, Gui-Rong Xue, Yong Yu, and Qiang Yang. 2011. Heterogeneous transfer learning for image classification. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'11)*.

Received January 2016; revised September 2016; accepted September 2016