Fast Generalized Distillation for Semi-Supervised Domain Adaptation

Shuang Ao, Xiang Li, Charles X. Ling
Department of Computer Science, The University of Western Ontario
sao@uwo.ca, lxiang2@uwo.ca, cling@csd.uwo.ca

Abstract

Semi-supervised domain adaptation (SDA) is a typical setting when we face the problem of domain adaptation in real applications. How to effectively utilize the unlabeled data is an important issue in SDA. Previous work requires access to the source data to measure the data distribution mismatch, which is ineffective, when the size of the source data is relatively large. In this paper, we propose a new paradigm, called *Generalized Distillation Semi-supervised Domain Adaptation* (GDSDA). We show that without accessing the source data, GDSDA can effectively utilize the unlabeled data to transfer the knowledge from the source models. Then we propose GDSDA-SVM which uses SVM as the base classifier and can efficiently solve the SDA problem. Experimental results show that GDSDA-SVM can effectively utilize the unlabeled data to transfer the knowledge between different domains under the SDA setting.

Introduction

Domain adaptation can be used in many real applications, which addresses the problem of learning a target domain with the help of a different but related source domain. In real applications, it can be expensive to obtain sufficient labeled examples while there are abundant unlabeled ones. *Semi-supervised domain adaptation* (SDA) tries to exploit the knowledge from the source domain and use a certain number of unlabeled examples and a few labeled ones from the target domain to learn a target model. Typically, the labeled examples in the target domain are too few to construct a good classifier alone. Therefore, an important issue in SDA is how to effectively utilize the unlabeled examples.

Previous work in SDA required access to the source data to measure the data distribution mismatch between the source and target domains (Duan et al. 2009; Donahue et al. 2013; Daumé III, Kumar, and Saha 2010; Yao et al. 2015). However, in some situations, we may not be able to access each of the source examples, for many reasons. When we use a large dataset as our source domain, for example, it is ineffective to compare each of the source examples with the target data to estimate the data distribution mismatch.

Recently, a framework called *Generalized Distillation* (**GD**)(Lopez-Paz et al. 2016) was proposed, which allows the knowledge to be transferred between different models effectively. GD contains two different models, the *teacher model* and the *student model*. The student model tries to learn from the teacher model by mimicking the outputs of the teacher model on the training data. Remarkably, in GD, the knowledge can be directly transferred from the teacher model to the student model without accessing the data used to train the teacher. Moreover, GD can be used to exploit the information of the unlabeled data in a semi-supervised scenario(Lopez-Paz et al. 2016). Given that GD has such ability, it is natural to ask the following two questions: (1) Can the GD framework be applied to solve the SDA problem? (2) How can we improve its effectiveness when we apply GD to real SDA applications?

To answer these two questions, in this paper, we first propose a new paradigm, called *Generalized Distillation Semi-supervised Domain Adaptation* (**GDSDA**), to solve the SDA problem. To answer the first question, we show that with GDSDA, the knowledge of the source models can be effectively transferred to the target domain using the unlabeled data. Specifically, the target model is trained with the help of the soft labels, which are the predictions of the target domain examples given by the source models. Therefore, without accessing each of the source examples, GDSDA is more efficient, especially when the source domain is relatively large and the source model is well-trained.

For the second question, we argue that the imitation parameter of GDSDA which controls the amount of knowledge transferred from the source model can greatly affect the performance of the target model. However, according to the previous work(Lopez-Paz et al. 2016; Tzeng et al. 2015), the imitation parameter is the hyperparameter which was determined by either a brute force search or domain knowledge in previous work. Therefore, we propose a novel imitation parameter estimation method for GDSDA, called GDSDA-SVM, which uses SVM as the base classifier and determines the imitation parameter efficiently. In particular, we use the Mean Square Error loss for GDSDA-SVM and show that the Leave-one-out cross validation (LOOCV) loss can be calculated in a closed form. By minimizing the LOOCV loss on the target data, we can find the optimal imitation parameter. In our experiments, we show that GDSDA-SVM can effectively find the optimal imitation parameter and achieve competitive performance compared to methods using brutal force search but with faster speed.

To summarize, the main contributions of this paper are: (1) we propose the paradigm of GDSDA that can directly transfer the knowledge from the source model with the help of unlabeled data for the SDA problems. (2) We propose the GDSDA-SVM which can effectively find the optimal imitation parameter for real SDA applications.

Related Work

As we use GD to solve SDA problem, we introduce related work in both GD and SDA areas.

In SDA, previous work tried to utilize the unlabeled data to improve the performance. (Yao et al. 2015) introduced a framework named Semi-supervised Domain Adaptation with Subspace Learning (SDASL) to correct data distribution mismatch and leverage unlabeled data. (Donahue et al. 2013) proposed a framework for adapting classifiers by "borrowing" the source data to the target domain using a combination of available labeled and unlabeled examples. (Daumé III, Kumar, and Saha 2010) show that augmenting the feature space of the data can compensate the domain shift. (Duan et al. 2009) proposed a method using the unlabeled data to measure the mismatch between different domains based on the maximum mean discrepancy.

There are also many studies related to GD for computer vision tasks. (Sharmanska, Quadrianto, and Lampert 2013) proposed a Rank Transfer method that uses attributes, annotator rationales, object bounding boxes, and textual descriptions as the privileged information for object recognition. (Motiian et al. 2016) proposed the information bottleneck method with privileged information (IBPI) that leverage the auxiliary information such as supplemental visual features, bounding box annotations and 3D skeleton tracking data to improve visual recognition performance. (Tzeng et al. 2015) proposed a CNN architecture for domain adaptation to leverage the knowledge from limited or no labeled data using the soft label. (Urban et al. 2016) used a small shallow net to mimick the output of a large deep net while using layer-wised distillation with ℓ_2 loss of the outputs of the student and teacher net. Similarly, (Luo et al. 2016) used ℓ_2 loss to train a compressed student model from the teacher model for face recognition.

Compared to previous work on SDA, our method only requires the output of the source models, which is more effective when the size of the source domain is relatively large and the source model is well-trained. Compared to other work in GD, our method GDSDA-SVM can effectively estimate the imitation parameter while previous work was limited to using either a brutal force search or domain knowledge.

Generalized Distillation for Semi-supervised Domain Adaptation

As previously mentioned, GDSDA is a paradigm using generalized distillation for semi-supervised domain adaptation.

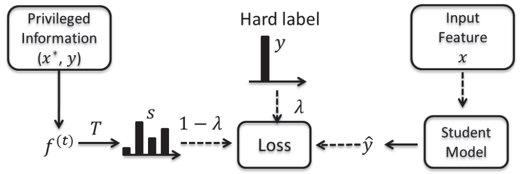


Figure 1: Illustration of Generalized Distillation training process.

In this section, we first give a brief review of generalized distillation. Then we show the process of GDSDA and demonstrate the reason why GDSDA can work for the SDA problem. Finally, we show the importance of the imitation parameter.

An overview of Generalized Distillation and GDSDA

Distillation (Hinton, Vinyals, and Dean 2014) and *Learning Using Privileged Information* (LUPI) (Vapnik and Izmailov 2015) are two paradigms that enable machines to learn from other machines. Both methods address the problem of how to build a student model that can learn from the advanced teacher models. Recently, (Lopez-Paz et al. 2016) proposed a framework called *generalized distillation* that unifies both methods and show that it can be applied in many scenarios.

In GD, the training data can be represented as a collection of the triples:

$$\{(x_1, x_1^*, y_1), (x_2, x_2^*, y_2) \ldots (x_n, x_n^*, y_n)\}$$

x^* is the privileged information for data x, which is only available in the training set and y is the corresponding label. Therefore, the goal of GD is to train a model, called student model with the guidance of the privileged information to predict the unseen example pair (x, y).

The process of generalized distillation is as follows: in step 1, a teacher model $f^{(t)}$ is trained using the input-output pairs $\{x_i^*, y_i\}_{i=1}^n$. In step 2, use $f^{(t)}$ to generate the soft label s_i for each training example x_i using the softmax function σ:

$$s_i = \sigma(f^{(t)}(x_i)/T) \tag{1}$$

where T is a parameter called temperature to control the smoothness of the soft label. In step 3, learn the student $f^{(s)}$ from the pairs $\{(x_i, y_i), (x_i, s_i)\}_{i=1}^n$ using:

$$\begin{aligned} f^{(s)} = & \mathop{\arg\min}_{f^{(s)} \in \mathcal{F}^{(s)}} \frac{1}{n} \sum_{i=1}^{n} \Big[\lambda \ell \Big(y_i, \sigma(f^{(s)}(x_i)) \Big) \\ & + (1-\lambda) \ell \Big(s_i, \sigma(f^{(s)}(x_i)) \Big) \Big] \end{aligned} \tag{2}$$

Here, $\ell(\cdot, \cdot)$ is the loss function and λ is the imitation parameter to balance the importance between the hard label y_i and the soft label s_i.

GD can be used in many scenarios such as multi-task learning, semi-supervised learning, and reinforcement learning. In domain adaptation, when we consider the source model as the teacher and the predictions of the target data

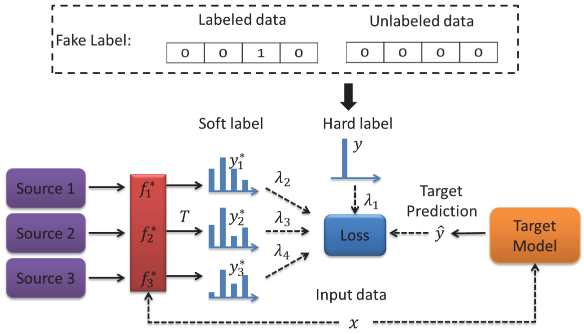


Figure 2: Illustration of GDSDA training process and our "fake label" strategy.

given by the source models as the privileged information, GD can be naturally applied to SDA. This leads to *Generalized Distillation Semi-supervised Domain Adaptation* (**GDSDA**). Moreover, in GDSDA, we also consider the multi-source scenario and extend the GD paradigm to fit this scenario. To be consistent with other work of domain adaptation, we use the *source model* and the *target model* to denote the teacher model and the student model.

Technically, when we apply GD to SDA, according to Eq. (2), each example is assigned with a hard label y (true label) and a soft label s (class probabilities from the teacher). However, in SDA, we are not able to obtain the hard labels of the unlabeled data. Here we follow the GD work(Lopez-Paz et al. 2016) and use the "fake label" strategy to label the unlabeled data: for the labeled examples, we use *one-hot* strategy to encode their labels while using all 0s as the label of the unlabeled examples (see Fig 2). Thus, each example in the target domain is assigned with a label. It is arguable that the "fake label" strategy would introduce extra noise and degrade the performance. However, we will show in our experiment that this noise can be well controlled by setting a proper value to the imitation parameter and we can still achieve improved performance (See the single source experiment).

Suppose we have $M-1$ source domains denoted as $D_s^{(j)} = \{X^{(j)}, Y^{(j)}\}_{j=1}^{M-1}$ and the target domain $D_t = \{X, Y\}$ encoded with the "fake label" strategy. The process of GDSDA is as follows:

1. Train the source models f_j^* for each of the $M-1$ domains with $\{X^{(j)}, Y^{(j)}\}$.
2. For each of the training example x_i in the target domain, generate the corresponding soft label y_{ij}^* with each of the source model f_j^* and the temperature $T > 0$.
3. Learn the target model f_t using the $(M+1)$-tuples $\{x_i, y_i, y_{i1}^*, \ldots, y_{i(M-1)}^*\}_{i=1}^{L}$ with the imitation parameters $\{\lambda_i\}_{i=1}^{M}$ using (3):

$$f_t(\lambda) = \arg\min_{f_t \in \mathcal{F}} \frac{1}{L} \sum_{i=1}^{L} \Bigg[\lambda_1 \ell\left(y_i, f_t(x_i)\right) + \sum_{j=1}^{M-1} \lambda_{j+1} \ell\left(y_{ij}^*, f_t(x_i)\right) \Bigg] \quad \text{s.t.} \quad \sum_i \lambda_i = 1 \tag{3}$$

Compared to other studies on SDA where each example of the source domain was used, by either re-weighting (Donahue et al. 2013; Duan et al. 2012) or augmentation (Daumé III, Kumar, and Saha 2010), GDSDA only requires the trained model from the source domain to generate the soft labels. Considering that it is more convenient to access the source model than each of the examples of the source domain, GDSDA can be more useful than those previous methods. For example, if we want to use ImageNet (Deng et al. 2009) as the source domain, it is almost impossible to access each of the millions of examples while there are many well trained models publicly available online that can be used for GDSDA. Also, GDSDA is able to handle the multi-class scenario while previous methods, such as SHFA(Duan, Xu, and Tsang 2012) only solved the binary classification problem of SDA. Moreover, GDSDA is compatible with any type of source model that is able to output the soft label (i.e., the class probabilities).

Why does GDSDA work

In this section, we demonstrate the scenarios where GDSDA can work. Before we provide our analysis, we first introduce two basic assumptions for GDSDA: the *assumption of distillation* and the *assumption of the source model*.

Assumption of Distillation: The capacity (or VC dimension) of the target model f_t is smaller than the capacity of source model f^*. This assumption is inherited from distillation (Lopez-Paz et al. 2016). **Assumption of the source model:** The source model f^* should work better than a target model f_t' trained only with the hard labels. This assumption is common, especially in SDA where the labeled examples are often too few to build a good target model. For example, when we only have one labeled example from each class in the target training set, it is reasonable to assume that the source model trained from another domain can perform better than the model trained only with the target training data on the target task. Based on these two assumptions, we will show that GDSDA can effectively leverage the source model and transfer the knowledge between different domains under the SDA setting.

According to the ERM principle(Vapnik 1999), a simple model has better generalization ability than the complex one, if they both have the same training error. As long as the target model f_t can achieve similar training error to that of the source model f^* on the target domain, considering the fact that the VC dimension of f_t is smaller than f^*, we can expect that the target model has better generalization ability. This process can be achieved by letting the target model mimick the output of the source model on the training data. It is worthy to notice that in this process, the target model only has to mimick the output of the source model (soft label) without considering the hard labels of the examples. In another word, GDSDA provide an effective way to utilize the unlabeled data.

Arguably, because of the domain shift, the source model is biased towards the source domain when we apply it to the target task. However, as suggested in (Hinton, Vinyals, and Dean 2014), we can use labeled data from the target domain to compensate for the domain shift and achieve a better performance on the target task with Eq. (3). Specifically, we use the imitation parameter λ to control the relative importance between the soft label and the hard label, which in turn reflects the similarity between the source and target tasks. For example, in Figure 2, when we set $\lambda_2 = 0$, we actually ignore the knowledge from source domain 1. As a result, GDSDA can compensate for the domain shift under the setting of SDA (for more details, please see the experiment section).

Key parameter: the imitation parameter

From the above analysis, we can see that GDSDA can effectively transfer the knowledge from the source to the target domain. In this section, we demonstrate that the imitation parameter can greatly affect the performance of the target model.

In GDSDA, we must decide the values of 2 parameters, the temperature T and the imitation parameter λ. The temperature T controls the smoothness of the soft label and the imitation parameter λ controls how much knowledge can be transferred from the source model. Previous work has addressed the importance of knowledge control in domain adaptation (Duan, Xu, and Tsang 2012; Duan et al. 2012). Without carefully controlling the amount of knowledge transferred from the source domain, the target model may suffer from degraded performance or even negative transfer (Pan and Yang 2010). How to choose the imitation parameter is crucial for GDSDA. In previous work, however, the imitation parameter was determined by either a brute force search (Lopez-Paz et al. 2016) or background knowledge (Tzeng et al. 2015). Meanwhile, in real applications, it is common that multiple source domains can be exploited. As suggested by (Tommasi, Orabona, and Caputo 2014), learning from multiple related sources simultaneously can significantly improve the performance of the target model. However, these previous methods become more difficult to apply when there are multiple sources and imitation parameters to be determined. For these reasons, it is ideal to find an approach that can determine the imitation parameter automatically.

GDSDA-SVM

As previously mentioned, it is important to find an approach that can effectively determine the imitation parameter. In this section, we propose our method GDSDA-SVM which uses SVM as the base classifier and can effectively estimate the imitation parameter by minimizing the cross-validation error on the target domain.

Distillation with multiple sources

As suggested in (Vapnik and Izmailov 2015), the optimal imitation parameter should be the one that can minimize the training error on the target domain. Based on that, we propose our method GDSDA-SVM which can effectively estimate the imitation parameter.

Instead of using hinge loss in our GDSDA-SVM, we use Mean Squared Error (MSE) as our loss function for the following two reasons: (1) several recently studies (Ba and Caruana 2014; Luo et al. 2016; Romero et al. 2015; Urban et al. 2016) show that MSE is also an efficient measurement for the target model to mimick the output of the source model. (2) MSE can provide a closed form cross-validation error estimation which allows us to estimate the imitation parameter effectively.

Suppose we have L examples $\{\mathbf{x}_j, \mathbf{y}_j\}_{j=1}^{L}$ from N classes in the target domain where $X \in R^{L\times d}, Y \in R^{L\times N}$. Meanwhile, there are $M-1$ source (teacher) models providing the soft labels $Y^* = \{\mathbf{y}^*_{ij} | j = 1, ..., L; i = 1, ..., M-1\}$ for each of the L examples. For simplicity, we concatenate the hard label Y and soft label Y^* into a new label matrix S, where:

$$S = [Y; Y^*] = [S_1; ...; S_M]; S_i \in R^{L\times N}$$

To solve this N-class classification problem, we adopt the One-vs-All strategy to build N binary SVMs. To build the nth binary SVM, we have to solve the following optimization problem:

$$\begin{aligned} \min_{w_n} \quad & \frac{1}{2}||\mathbf{w}_n||^2 + C\sum_j e_{jn}^2 \\ s.t. \quad & e_{jn} = \sum_i \lambda_i S_{ijn} - \mathbf{w}_n\mathbf{x}_j \end{aligned} \tag{4}$$

We use the KKT theorem (Cristianini and Shawe-Taylor 2000) and add dual sets of variables to the Lagrangian of the optimization problem:

$$\begin{aligned} \mathcal{L} = & \frac{1}{2}||\mathbf{w}_n||^2 + C\sum_j e_{jn}^2 \\ & + \sum_j \eta_{jn}\left(\sum_i \lambda_i S_{ijn} - \mathbf{w}_n\mathbf{x}_j - e_{jn}\right) \end{aligned} \tag{5}$$

To find the saddle point,

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}_n} = 0 \to \mathbf{w}_n = \sum_j \eta_{jn}\mathbf{x}_j; \quad \frac{\partial\mathcal{L}}{\partial e_{jn}} = 0 \to \eta_{jn} = 2Ce_{jn} \tag{6}$$

For each example $\mathbf{x}_j$ and its constraint of label S_{ijn}, we have $e_{jn} + \mathbf{w}_n\mathbf{x}_j = \sum_i \lambda_i S_{ijn}$. Replacing $\mathbf{w}_n$ and e_{jn}, we have:

$$\sum_j \eta_{jn}\mathbf{x}_j\mathbf{x}_i + \frac{\eta_{in}}{2C} = \sum_i \lambda_i S_{ijn} \tag{7}$$

Here we use Ω to denote the matrix $\Omega = [K + \frac{\mathbf{I}}{2C}]$ where K is the linear kernel matrix $K = \{\mathbf{x}_i\mathbf{x}_j | i, j \in 1 \ldots L\}$. Let Ω^{-1} be the inverse of matrix Ω and Ω_{jj}^{-1} be the jth diagonal element of Ω^{-1}. We have $\eta = \sum_i \lambda_i\Omega^{-1}S_i = \sum_i \lambda_i\eta'_i$. According to (Cawley 2006), the Leave-one-out (LOO) estimation of the example $\mathbf{x}_j$ for the nth binary SVM can be written as:

$$\hat{y}_{jn} = \sum_i \lambda_i \left(S_{ijn} - \frac{{\eta'}_{ijn}}{\Omega^{-1}_{jj}} \right) \tag{8}$$

Now for any given λ, we have found an efficient way to estimate the LOO prediction of each binary target model for example $\mathbf{x}_j$. In the following section, we will introduce how to find the optimal λ_i for each of the source models.

Cross-entropy loss for imitation parameter estimation

From the previous section, we have already found an effective solution to estimate the output of the SVM. The optimal imitation parameters can be found by solving the following optimization problem:

$$\begin{aligned} \min \quad & L_c(\lambda) = \frac{1}{2} \sum_{i}^{M} ||\lambda_i||^2 + \frac{1}{L} \sum_{j,n} \ell\left(y_{in}, \hat{y}_{jn}(\lambda)\right) \\ s.t. \quad & \sum \lambda_i = 1 \end{aligned} \tag{9}$$

Here we use the ℓ-2 regularization term to control the complexity of λs so that the target model can achieve better generalization performance. For the loss function $\ell(\cdot,\cdot)$, We choose the cross-entropy loss function.

$$\ell\left(y_{in}, \hat{y}_{jn}(\lambda)\right) = y_{in} \log(P_{jn}) \qquad P_{jn} = \frac{e^{\hat{y}_{jn}}}{\sum_h e^{\hat{y}_{jh}}} \tag{10}$$

Cross-entropy pays less attention to a single incorrect prediction which reduces the affect of the outliers in the training data. Moreover, cross-entropy works better for the unlabeled data with our "fake label" strategy. As we mentioned in our "fake label" strategy, we use 0s to encode the hard labels of the unlabeled examples. From (10) we can see that cross-entropy loss can automatically ignore penalties of the unlabeled examples and reduce the noise introduced by our "fake label" strategy. Let:

$$\mu_{ijn} := S_{ijn} - \frac{\eta'_{ijn}}{\Omega^{-1}_{jj}} \tag{11}$$

The derivative can be written as:

$$\frac{\partial \ell(\lambda)}{\partial \lambda_i} = \sum_n \mu_{ijn} \left(P_{jn} - y_{jn}\right) \tag{12}$$

We describe GDSDA-SVM in Algorithm 1. As the optimization problem (9) is strongly convex, it is easy to prove that Algorithm 2 can converge to the optimal λ with the rate of $O(\log(t)/t)$ where t is the optimization iteration. Due to the space limit, we are not able to provide the proof here.

Experiments

In this section, we show the empirical performance of our algorithm GDSDA-SVM on the Office benchmark dataset. Specifically, we provide the empirical results under two transfer scenarios: single source and multi-source transfer scenarios for GDSDA-SVM.

Algorithm 1 GDSDA-SVM

```
Input: Input examples $X = \{\mathbf{x}_1, ..., \mathbf{x}_L\}$, number of
    classes $N$, number of sources $M$, 3D label matrix, $S =
    [Y_1, Y_2, ..., Y_M]$ with size $L \times M \times N$, temperature $T$
Output: Target model $f_t = Wx$
    $\Omega = [K + \frac{\mathbf{I}}{2C}]$
    Find the imitation parameter $\lambda$ with Algorithm 2
    Generate new label $Y_{new} = \sum_i \lambda_i S_i$
    Calculate $\eta = \Omega^{-1} Y_{new}$
    Calculate $w_n = \sum_j \eta_{jn} x_j$
```

Algorithm 2 λ Optimization

```
Input: Input examples $X$, number of classes $N$, size of
    sources $M$, 3D label matrix $S$, temperature $T$, optimiza-
    tion iteration $iter$, Kernel matrix $\Omega$
Output: Imitation parameter $\lambda$
    Initialize $\lambda = \frac{1}{M}$,
    Let $S_n$ be the label matrix of $S$ for class $n$
    for Each label $S_n$ do
        Calculate $\eta'_n = \Omega^{-1} S_n$
    end for
    Calculate $\mu$ using (11)
    for $it \in \{1, ..., iter\}$ do
        Compute $\hat{y}_{jn}$ and $P_{jn}$ with (8) and (10)
        $\Delta_\lambda \leftarrow 0$
        for each $\mathbf{x}_j$ in $X$ do
            $\Delta_\lambda = \Delta_\lambda + \sum_n \mu_{ijn} (P_{jn} - y_{jn})$
        end for
        $\Delta_\lambda = \Delta_\lambda / L$, $\lambda = \lambda - \frac{1}{it}(\Delta_\lambda + \lambda)$
        $\lambda = \lambda / \sum \lambda_i$
    end for
```

Dataset: We use the domain adaptation benchmark dataset Office as our experiment dataset. There are 3 subsets in Office dataset, Webcam (795 examples), Amazon (2817 examples) and DSLR (498 examples), sharing 31 classes. We denote them as W, A and D respectively. In our experiments, we use DSLR and Webcam as the source domains and Amazon as the target domain. We use the features extracted from Alexnet (Krizhevsky, Sutskever, and Hinton 2012) FC7 as the input feature for both source and target domain. The source models are trained with multi-layer perception (MLP) on the whole source dataset.

Single Source for Office datasets

In this experiment, we compare our algorithm under the scenario where the source model is trained from a single source dataset. Specifically, we have two groups of experiments, transferring from Webcam to Amazon and from DSLR to Amazon. As we mentioned, there are significantly fewer labeled examples than unlabeled ones in real SDA applications. Therefore, in each group of experiment, there are only 31 labeled examples (1 per class) and some unlabeled examples (10, 15 and 20 per class) in the target domain.

To demonstrate the effectiveness of GDSDA-SVM, we show the performance of GDSDA using brute force to search the imitation parameter as the baseline. As there are two imitation parameters in this experiment, we use λ_1 and $1 - \lambda_1$ to denote the imitation parameter for hard and soft label respectively. Specifically, we search the imitation parameter λ_1 in the range $[0, 0.1, ..., 1]$ with different temperature T. Meanwhile, we show the performance of the source model (denoted as "Source") and the performance of a target model (denoted as "No transfer" using LIBLINEAR(Fan et al. 2008)) trained with only labeled examples of the target domain on the target task. We run each experiment 10 times and report the average result. For GDSDA-SVM, as we are not able to tune the temperature T, we empirically set $T = 20$ for all experiments in this subsection. The experimental results are shown in Figure 3.

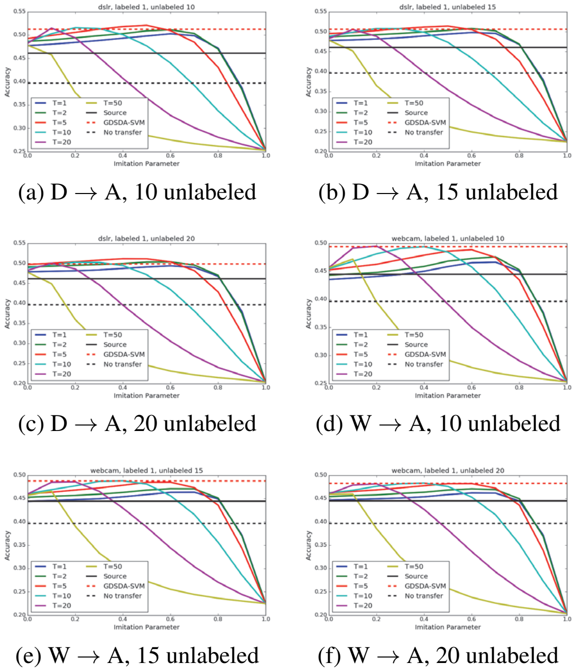
From the results of the brutal force search we can see that, the value of imitation parameter can greatly affect the performance of the target model. As we expected, without using any true label information of the target data, i.e. $\lambda_1 = 0$, GDSDA can still slightly outperform the source model. This means GDSDA can effectively transfer the knowledge between different domains with the unlabeled data. As we increase the value of imitation parameter, i.e. considering the hard labels from the target domain, the performance of GDSDA can be further improved. As we mentioned before, even though our "fake label" strategy would introduce extra noise, the noise can be limited by setting a proper value to imitation parameter and the target model can still achieve improved performance compared to the baselines.

Moreover, we can see that GDSDA-SVM can achieve competitive results compared to baselines using brutal force search in D→A experiments. In W→A experiments, it achieves the best performances on all 3 different unlabeled sizes. This indicates that we can efficiently (about 6 times faster than the brutal force search) obtain a good target model with GDSDA-SVM.

Multi-Source for Office datasets

In this experiment, we show the performance of GDSDA-SVM under the multi-source SDA scenario. Specifically, we use Amazon as the target domain which can leverage the knowledge of two source models trained from Webcam and DSLR. We use the similar settings as our single source experiment and perform 2 groups of experiments using 1 labeled and 2 labeled examples per class respectively. We use temperature $T = 5$. The results of multi-source GDSDA-SVM are denoted as SVM_Multi. Here we also include two single source GDSDA-SVMs obtained from the experiments above (SVM_w and SVM_d trained using Webcam and DSLR as the source respectively) as the baselines. Moreover, we show the best performance of the brutal force search model (SVM_BF). For SVM_BF, we search temperature in range $T = [1, 2, 5, 10, 20, 50]$ and each imitation parameter in range $[0, 0.1, ..., 1]$. The experiment results are shown in Figure 4.

From the results, we can see that, given 2 source models, SVM_Multi can outperform any single source model trained with GDSDA. This indicates GDSDA-SVM can still exploit the knowledge even in the complex multi-source scenario. Even though SVM_Multi performs slightly worse than the best result found by brutal force search in some experiments, considering their time consumption (GDSDA-SVM is around 30 times faster than brutal force search), SVM_Multi still has its advantage in real applications.



(a) D → A, 10 unlabeled

(b) D → A, 15 unlabeled

(c) D → A, 20 unlabeled

(d) W → A, 10 unlabeled

(e) W → A, 15 unlabeled

(f) W → A, 20 unlabeled

Figure 3: Experiment results on DSLR→Amazon and Webcam→Amazon when there are just one labeled examples per class. The X-axis denotes the imitation parameter of the hard label (i.e. λ_1 in Fig 2) and the corresponding imitation parameter of the soft label is set to $1 - \lambda_1$.

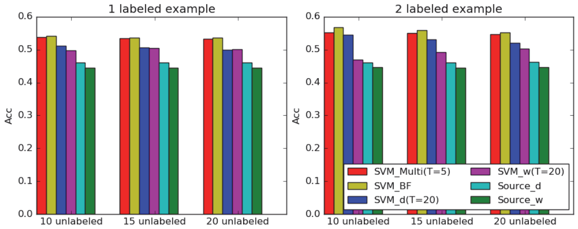


Figure 4: D+W→A, Multi-source results comparison.

Conclusion

In this paper, we propose a novel framework called *Generalized Distillation Semi-supervised Domain Adaptation* (GDSDA) that can effectively leverage the knowledge from the source domain for SDA problem without accessing to the source data. To make GDSDA more effective in real applications, we proposed our method called GDSDA-SVM and show that GDSDA-SVM can effectively determine the imitation parameter for GDSDA. In our future work, we plan to use a more advanced hyperparameter optimization method, which can optimize the imitation parameter λ and the temperature T in GDSDA simultaneously, and expect further performance improvement

Acknowledgments

We thank the anonymous reviewers for their valuable comments to improve this paper. This work is supported by Natural Sciences and Engineering Research Council of Canada (NSERC).

References

Ba, J., and Caruana, R. 2014. Do deep nets really need to be deep? In *Advances in neural information processing systems*, 2654–2662.

Cawley, G. C. 2006. Leave-one-out cross-validation based model selection criteria for weighted ls-svms. In *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, 1661–1668. IEEE.

Cristianini, N., and Shawe-Taylor, J. 2000. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.

Daumé III, H.; Kumar, A.; and Saha, A. 2010. Frustratingly easy semi-supervised domain adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, 53–59. Association for Computational Linguistics.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Donahue, J.; Hoffman, J.; Rodner, E.; Saenko, K.; and Darrell, T. 2013. Semi-supervised domain adaptation with instance constraints. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Duan, L.; Tsang, I. W.; Xu, D.; and Chua, T.-S. 2009. Domain adaptation from multiple sources via auxiliary classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 289–296. ACM.

Duan, L.; Xu, D.; Tsang, I. W.-H.; and Luo, J. 2012. Visual event recognition in videos by learning from web data. volume 34, 1667–1680. IEEE.

Duan, L.; Xu, D.; and Tsang, I. W. 2012. Learning with augmented features for heterogeneous domain adaptation. In *Proceedings of the International Conference on Machine Learning*, 711–718. Edinburgh, Scotland: Omnipress.

Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; and Lin, C.-J. 2008. Liblinear: A library for large linear classification. *Journal of machine learning research* 9(Aug):1871–1874.

Hinton, G.; Vinyals, O.; and Dean, J. 2014. Distilling the knowledge in a neural network. *NIPS Deep Learning and Representation Learning Workshop*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1106–1114.

Lopez-Paz, D.; Schölkopf, B.; Bottou, L.; and Vapnik, V. 2016. Unifying distillation and privileged information. In *International Conference on Learning Representations*.

Luo, P.; Zhu, Z.; Liu, Z.; Wang, X.; and Tang, X. 2016. Face model compression by distilling knowledge from neurons. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Motiian, S.; Piccirilli, M.; Adjeroh, D. A.; and Doretto, G. 2016. Information bottleneck learning using privileged information for visual recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22(10):1345–1359.

Romero, A.; Ballas, N.; Kahou, S. E.; Chassang, A.; Gatta, C.; and Bengio, Y. 2015. Fitnets: Hints for thin deep nets. In *In Proceedings of International Conference on Learning Representations*.

Sharmanska, V.; Quadrianto, N.; and Lampert, C. H. 2013. Learning to rank using privileged information. In *The IEEE International Conference on Computer Vision (ICCV)*.

Tommasi, T.; Orabona, F.; and Caputo, B. 2014. Learning categories from few examples with multi model knowledge transfer. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 36(5):928–941.

Tzeng, E.; Hoffman, J.; Darrell, T.; and Saenko, K. 2015. Simultaneous deep transfer across domains and tasks. In *The IEEE International Conference on Computer Vision (ICCV)*.

Urban, G.; Geras, K. J.; Kahou, S. E.; Aslan, O.; Wang, S.; Caruana, R.; rahman Mohamed, A.; Philipose, M.; and Richardson, M. 2016. Do deep convolutional nets really need to be deep (or even convolutional)? In *International Conference on Learning Representations (workshop track)*.

Vapnik, V., and Izmailov, R. 2015. Learning using privileged information: Similarity control and knowledge transfer. *Journal of Machine Learning Research* 16:2023–2049.

Vapnik, V. N. 1999. An overview of statistical learning theory. *IEEE Transactions on Neural Networks* 10(5):988–999.

Yao, T.; Pan, Y.; Ngo, C.-W.; Li, H.; and Mei, T. 2015. Semi-supervised domain adaptation with subspace learning for visual recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2142–2150.