Domain Adaptations for Computer Vision Applications

Oscar Beijbom,

obeijbom@ucsd.edu

Department of Computer Science and Engineering, University of California, San Diego.

Abstract

A basic assumption of statistical learning theory is that train and test data are drawn from the same underlying distribution. Unfortunately, this assumption doesn't hold in many applications. Instead, ample labeled data might exist in a particular 'source' domain while inference is needed in another, 'target' domain. Domain adaptation methods leverage labeled data from both domains to improve classification on unseen data in the target domain. In this work we survey domain transfer learning methods for various application domains with focus on recent work in Computer Vision.

1. Introduction

The shortage of labeled data is a fundamental problem in applied machine learning. While huge amounts of unlabeled data is constantly being generated and made available in many domains, the cost of acquiring data labels remains high. Even, worse, sometimes the situation makes it highly impractical or even impossible to acquire labelled data (e.g. when the underlying distribution is constantly changing).

Domain adaptation (sometimes referred to as domain transfer learning) approach this problem by leveraging labelled data in a related domain, hereafter referred to as 'source' domain, when learning a classifier for unseen data in a 'target' domain. The domains are assumed to be *related*, but not identical (in which case it becomes a standard machine learning problem).

This situation occur in many domains. A few examples are: event detection in across video corpora from different domains (e.g. different tv - stations), named entity recognition across different text corpora (e.g. sports text corpus and news corpus), object recognition in images acquired in different domains (webcam versus Amazon stock photos).

Domain adaptation (DA) only recently started receiving significant attention [DauméIII, 2009, Chelba and Acero, 2004, DauméIII and Marcu, 2006], in particular for computer vision applications [Saenko et al., 2010, Kulis et al., 2011, Gopalan et al., 2011, Jhuo et al., 2012, Duan et al.,

2009, Bergamo and Torresani, 2010], although related field, such as covariate shift [Shimodaira, 2000] has a longer history. It is perhaps indicative of the field being so new, that the proposed methods are of such different characteristic. To the best of our knowledge, there is only one previous survey of domain adaptation [Jiang, 2008], which focused on learning theory and natural language processing applications. Also [Pan and Yang, 2009] did a thorough survey on the related field of transfer learning.

1.1. Related Fields

As mentioned in the introduction, the shortage of labeled data is a fundamental problem for applied machine learning. It is important enough that several areas of research is devoted to various aspects of this problem. In the active learning paradigm, labels are acquired in an interactive fashion to maximize the benefit of each new label [Kapoor et al., 2007]. Related approaches include [Branson et al., 2011], where a 'human-in-the-loop' determines which labels to update, thus making the 'most' out of the acquired labels. Crowd sourcing through, e.g. Amazon mechanical turk (mTurk), allows for rapid collection of large amounts of labels, and much research is devoted to the efficient distribution of tasks and the interpretation and weighting of retrieved labels [Welinder et al., 2010]. Further areas include weakly supervised method, e.g. multiple instance learning [Dietterich et al., 1997] or latent structureal SVMs [Yu and Joachims, 2009] where the level of supervision is lower than the given task demands. Other approaches include semi-supervised learning that make use of small amounts of labelled data together with large amounts of unlabeled data. Notably the concept of co-training [Blum and Mitchell, 1998] is a popular approach.

More closely related to domain adaptation is transfer learning [Pan and Yang, 2009]. In transfer learning (TL) the marginal distribution of the source and target data are similar but different tasks are considered. To make this problem tractable, it is typically assumed a common prior on the model parameters across tasks. A computer vision example is 'one-shot learing' [Fei-Fei et al., 2006] where new visual categories are leaned using a single training example

by leveraging data from other labelled categories. This is different from domain adaptation where the marginal data distributions of source and target are different, but the task is similar.

Another related field is model-adaptation. Here, unlabeled data, sometimes referred to as *background* data or *auxiliary* data is used to regularize the class specific models. This paradigm has had much success in speaker verification [Reynolds et al., 2000], and has also been applied to computer vision problems [Dixit et al., 2011, L. Fei-Fei, 2007]. The methods used in this field, such as Adapted Gaussian Mixture Models [Reynolds et al., 2000] could trivially be used in a domain transfer setting by discarding source data labels, and letting the source data constitute the background model.

Cross-Modal classification / retrieval makes very similar assumptions on the data compared to DA but assume *instance*, rather than *class*, level relationship between the domains. In Cross-Modal classification, ample data from both domains are available at train time, and the unknown sample can come from any of the modalities.

2. Setup

In this section we introduce notation and provide a overview of the paper.

2.1. Notation

Let X denote the input, and Y the output random variable. Let $P(X, Y)$ denote the joint probability distribution of X and Y. Let, similarly $P(X)$ and $P(Y)$ denote the marginal probability distributions. In the domain adaptation scenario, as mentioned in the introduction, we have two distinct distributions. Let $P_s(X, Y)$ denote the *source* distribution where, typically, we have access to ample labelled data, and let $P_t(X, Y)$ be the *target* distribution that we seek to estimate. We also let $P(X = x, Y = y) \equiv P(x, y)$ refer to the joint probability, thus differentiating it from P(X, Y) that represents the probability distribution.

Data is available from three sets: labelled data from the source domain $\mathcal{S}_l = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{N^s}, \mathbf{x}^s \in \mathbb{R}^{d_s}$, drawn from a joint source probability distribution, $P_s(X, Y)$; labelled data from the target domain, $\mathcal{T}_l = \{(\mathbf{x}_i^{t,l}, y_i^t)\}_{i=1}^{N^{t,l}}, \mathbf{x}^{t,l} \in \mathbb{R}^{d_t}$, drawn from a joint target distribution, $P_t(X, Y)$; and unlabeled data, $\mathcal{T}_u = \{(\mathbf{x}_i^{t,i})\}_{i=1}^{N^{t,u}}, \mathbf{x}^{t,u} \in \mathbb{R}^{d_t}$ from a marginal target distribution $P_t(X)$. It is commonly assumed $N^s \gg N^{t,l}$, and $d_s = d_t$. The target and source labels are generally assumed to belong to the same space, e.g. for the k-class classification task, $y^s, y^t \in \mathbb{Z}^k$. We further let D^i be the data matrix for domain i, with one data sample per column, $i \in \{t, s\}$.

The goal of domain adaptation (DA) can thus be summarized as that of learning a function $\hat{y} = f(\mathbf{x}_u | \mathcal{D})$ that predicts the class, y_u of an unseen sample from the target with high probability, $P_t(Y = \hat{y} | X = \mathbf{x}_u)$. $\mathcal{D}$ is different depending on the data assumptions. In supervised DA, $\mathcal{D} = \mathcal{S}_l \cup \mathcal{T}_l$, in unsupervised DA $\mathcal{D} = \mathcal{S}_l \cup \mathcal{T}_u$, and semi supervised DA: $\mathcal{D} = \mathcal{S}_l \cup \mathcal{T}_l \cup \mathcal{T}_u$.

2.2. Overview

As mentioned in the introduction domain adaptation is a relatively new field. It is also relatively loosely defined with regards to e.g. how 'related' are the domains, and how 'few' labelled samples exist in the target domain. Also, the general problem statement applies to several application domains, such as natural language processing and computer vision. For all these reasons, there is a big variety in the proposed methods. Inspired by the categorization proposed in [Jiang, 2008], we begin by considering instance weighting methods for relaxation of the DA assumptions in Sec. 3. We consider methods utilizing the source data to regularize target models in Sec. 4. We then survey method seeking common representation across domains in Sec. 5. Section 6 make connections to transfer learning, and Sec. 7 briefly survey method for multi-modal learning.

3. Instance Weighting

Following [Jiang, 2008], we first consider two relaxations of the DA problem. For the analysis we will use the empirical risk minimization framework proposed by [Vapnik, 1999] for standard supervised data. Here we let $\theta \in \Theta$ be a model parameter from a given parameter space, and θ^* be the optimal parameter choice for the distribution $P(X, Y)$. Let further $l(x, y, \theta)$ be a loss function. In this framework we want to minimize

$$\theta^* = \arg\min_{\theta \in \Theta} \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} P(x, y) l(x, y, \theta) \quad (1)$$

$P(X, Y)$ is unknown but we can estimate it with the empirical distribution, $\tilde{P}(X, Y)$.

$$\tilde{\theta} = \arg\min_{\theta \in \Theta} \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} \tilde{P}(x, y) l(x, y, \theta) \quad (2)$$

$$= \arg\min_{\theta \in \Theta} \sum_{i=1}^{N} l(x_i, y_i, \theta). \quad (3)$$

[Jiang, 2008] extend this to the DA problem and arrive at the following formulation

$$\theta_t^* = \arg\min_{\theta \in \Theta} \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} P_t(x, y) l(x, y, \theta) \quad (4)$$

$$\approx \arg\min_{\theta \in \Theta} \sum_{i=1}^{N^s} \frac{P_t(x_i^t, y_i^t)}{P_s(x_i^s, y_i^s)} l(x_i^s, y_i^s, \theta). \quad (5)$$

We see that weighing the loss of (source) training sample by $\frac{P_t(x,y)}{P_s(x,y)}$ provides a solution that is consistent with the empirical risk minimization framework. Clearly, if we had a good estimate of $P_t(X, Y)$ we would already be done, so this doesn't really help us, but the formulation is useful for the discussion below. In the following we consider two relaxations of the DA problem formulation. Class imbalance: $P_t(X|Y = y) = P_s(X|Y = y)$ and covariate shift: $P_t(Y|X) = P_s(Y|X)$.

3.1. Class Imbalance

One way to relax the DA problem formulation is to assume $P_t(X|Y) = P_s(X|Y)$, but $P_t(Y) \neq P_s(Y)$. This is called class imbalance, population drift or sampling bias. Consider, for example, training data sampled from a remote sensing application. Test data collected a at a later occasion may have different class distribution due to a changed landscape. Taking the assumptions into account, the ratio $\frac{P_t(x,y)}{P_s(x,y)}$ becomes

$$
\begin{aligned}
\frac{P_t(x,y)}{P_s(x,y)} &= \frac{P_t(y)}{P_s(y)} \frac{P_t(x|y)}{P_s(x|y)} \qquad (6) \\
&= \frac{P_t(y)}{P_s(y)}, \qquad (7)
\end{aligned}
$$

and we only need to consider $\frac{P_t(y)}{P_s(y)}$. This approach was explored in [Lin et al., 2002]. We can also re-sample the data to make the class distributions equal.

3.2. Covariate Shift

Covariance shift [Shimodaira, 2000], is another relaxation of DA. Here, given an observation, the class distributions are same in the source and target domains, but the marginal data distributions are different. $P_t(Y|X) = P_s(Y|X)$, but $P_t(X) \neq P_s(X)$. This situation arise, for example, in active learning, where the $P_s(X)$ tend to be biased to lie near the margin of the classifier. At a first glance, this situation appears not to present a problem, since $P_t(Y|X) = P_s(Y|X)$, which we can estimate from the data. Here is why it becomes a problem in practice. Assuming, first of all, that the model family we use is mismatched to the data, i.e. regardless of what parameter we choose the model won't fit the underlying distribution. Under this assumption, covariate shift becomes a problem for the following reason. The optimal fit of the source data will be such that it minimize model error in the dense area of $P_s(X)$ (because these areas will dominate the error). Now, since $P_t(X)$ is different from $P_s(X)$, the learned model will not be optimal for the target data (again, since the model family is mismatched).

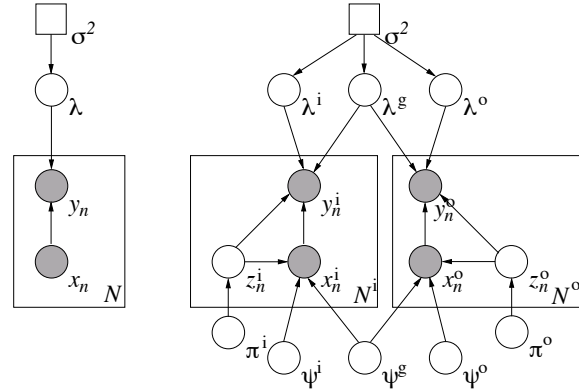As in the previous section, $\frac{P_t(x,y)}{P_s(x,y)}$ can be simplified un-

Figure 1. The MEGA model proposed in [DauméIII and Marcu, 2006]. This model assumes the data is in fact generated by three distributions, a target, a common and and source. The MEGA model learns a classifier for each space. Left is the standard logistic regression model.

der these assumptions

$$
\begin{aligned}
\frac{P_t(x,y)}{P_s(x,y)} &= \frac{P_t(x)}{P_s(x)} \frac{P_t(y|x)}{P_s(y|x)} \qquad (8) \\
&= \frac{P_t(x)}{P_s(x)}. \qquad (9)
\end{aligned}
$$

Again, a well founded solution can be identified by appropriate instance weighting of the loss function. [Shimodaira, 2000] explored this approach and show that the weighted model better estimate the data given a biased sampling function. The quantity $\frac{P_t(x)}{P_s(x)}$ can be estimated using *e.g.* non-parametric kernel estimation [Sugiyama and Mülcer, 2005, Shimodaira, 2000]. [Huang et al., 2007] proposed to directly estimate the ratio, i.e. the difference between the two distributions. They use the Kernel Mean Match,

$$
\left\| \frac{1}{N^s} \sum_{i=1}^{N^s} \beta_i \phi(\mathbf{x}_i^s) - \frac{1}{N^t} \sum_{i=1}^{N^t} \phi(\mathbf{x}_i^t) \right\|^2, \qquad (10)
$$

metric that measures the distribution distance in a Reproducing Kernel Hilbert Space.

4. Source Distribution As Prior

Often, the simplifying assumptions of the previous section doesn't hold. This section discuss method that use prior probabilities estimated on the source data to regularize the model. We first cover priors in the bayesian sense, and then some examples of discriminative methods.

4.1. Bayesian Priors

Maximum a posterior (MAP) estimation of model parameters is central in bayesian statistics. In this setting prior
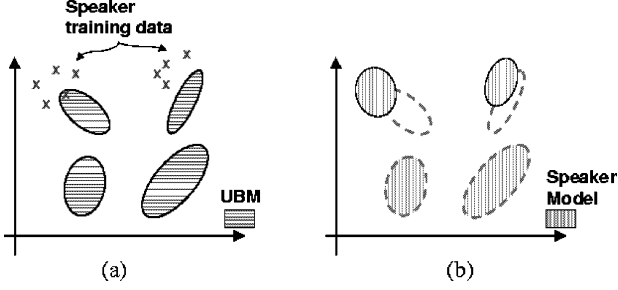
Figure 2. Figure from [Reynolds et al., 2000] illustrating the adapted GMM model. The left figure shows the universal GMM estimated from the background data together with the speaker-specific train data. The right shows the adapted model.

knowledge about the model can be incorporated in a prior probability of the parameters, $P(\theta)$. Specifically, instead of finding optimal parameters θ^* as

$$\theta^* = \arg\max_{\theta} \prod_{i=1}^{n} P(y_i|x_i; \theta), \qquad (11)$$

one solves

$$\theta^* = \arg\max_{\theta} P(\theta) \prod_{i=1}^{n} P(y_i|x_i, \theta). \qquad (12)$$

In domain adaptation we can estimate the prior probability from the source domain as

$$\theta^* = \arg\max_{\theta} P(\theta|\mathcal{S}_l) \prod_{i=1}^{N^{t,l}} P(y_i^t|x_i^t, \theta). \qquad (13)$$

[Chelba and Acero, 2004] pursued this approach in adapting a maximum entropy capitalizer. [DauméIII and Marcu, 2006] argued that this two step process (first estimating $P(\theta)$ from S_l and then estimating θ) was non-intuitive and suggested an ensemble model that considered three classifiers simultaneously, one for the target, one for the source and one for the joint portion of the data. This generative model, that they denote MEGA, is shown in Fig. 1.

When $P(\theta|\mathcal{S})$ is estimated with unlabeled data, the problem is technically no longer domain adaptation but rather model adaptation. Adapted Gaussian Mixture Models have successfully been applied to speaker verification [Reynolds et al., 2000, W. M. Campbell, 2006], and recently also for computer vision [Dixit et al., 2011]. Figure 2 show a schematic illustration of adapted GMM.

4.2. Discriminative Priors

In this section we survey work that investigate modifying the support vector machine (SVM) algorithms for the domain adaptation problem. These methods typically use a

already-trained SVM in the source domain as input to subsequent training. The source data is thus used to *regularize* the output model in a similar way as in Sec. 4.1.

[Yang et al., 2007] propose the adaptive support vector machine (ASVM). The basic idea is to learn a new decision boundary that is close to that learned in the source domain. The source data is thus acting as a regularizer on the final model. This method presume the existence of a SVM model $f^s(\mathbf{x})$ trained on the source domain data. They let the final decision function, $f(\mathbf{x})$ be the sum of $f^s(\mathbf{x})$ and $w^T \phi(\mathbf{x})$. The final classifier is attained by solving the following constrained optimization problem.

$$\min_{w} \frac{1}{2}||\mathbf{w}||_2^2 + C \sum_{i=1}^{N^{t,l}} \xi_i$$
$$s.t. \; \xi_i \geq 0$$
$$y_i(f^s(\mathbf{x}_i) + \mathbf{w}^T \mathbf{x}_i^t) \geq 1 - \xi_i.$$

One problem with this formulation is that it doesn't strive for a large margin, but rather a solution close to the source solution. This is only reasonable for situation where $P_t(X, Y)$ is similar to $P_s(X, Y)$. To address this [Jiang et al., 2008] proposed the Cross-Domain SVM (CDSVM). CDSVM relax the constraints that the final model need to be similar to the old one by only enforcing proximity where the support vectors of $f^s(x)$ are close to any of the target data. They do this by introducing additional constraints that the old support vectors, just like the target data points, should be correctly classified. These constraints are only active when the old support vectors are close to any part of the target data. Specifically, they solve the following constrained optimization problem

$$\min_{w} \frac{1}{2}||\mathbf{w}||_2^2 + C \sum_{i=1}^{|\mathcal{T}_l|} \xi_i + C \sum_{j=1}^{M} \sigma(\mathbf{v}_j^s, \mathcal{T}_l) \bar{\xi}_j$$
$$s.t. \; y_i(\mathbf{w}^T \phi(\mathbf{x}_i) - b) \geq 1 - \xi_i, \xi_i \geq 0, \forall (\mathbf{x}_i, y_i) \in \mathcal{T}_l$$
$$y^s i(\mathbf{w}^T \phi(\mathbf{v}_j^s) - b) \geq 1 - \bar{\xi}, \bar{\xi} \geq 0, \forall (\mathbf{v}_j^s, y_j^s) \in \mathcal{V}_s.$$

Here $\mathbf{v}_j^s \in \mathcal{V}_s$ are the support vectors of $f^s(x)$ with signs y_j^s. The authors let $\sigma(\mathbf{v}^s, \mathcal{T}_l)$ be a gaussian that determines which vectors are close. [Bergamo and Torresani, 2010] provides a survey of other SVM-based DA methods.

5. Common representation

The perhaps most intuitive way to do domain adaptation is to create a feature map such that the source and target distributions are aligned. In other words, finding functions $g_t(X)$ and $g_s(X)$ for which

$$P_t(Y = k|g_t(X) = x) = P_s(Y = k|g_s(X) = x)$$
$$\forall (y, x) \in (\mathcal{Y} \times \mathcal{X}) \quad (14)$$

where functions $g_t(X)$ and $g_s(X)$ might be equal, related or even identity, depending on the method. [Jiang, 2008] note that the entropy of $Y|g(X)$ is likely to increase compared to $Y|X$ since the feature representation usually is simpler after the mapping, and thus encode less information. This means that Bayes error is likely to increase, and a good algorithm for domain alignment should take this into account. A simple, and straight forward way of doing this is by feature selection. [Satpal and Sarawagi, 2006] proposed a method for this that remove features to minimize an approximated distance function between the source and target distributions. Specifically they minimize $\sum_{k \in K} d(E_{\mathcal{S}_l}^k, E_{\mathcal{T}_u}^k)$ where

$$E_{\mathcal{S}_l}^k = \sum_{(x_l, y_l) \in \mathcal{S}_l} \frac{f_k(\mathbf{x}_l, \mathbf{y}_l)}{N} \qquad (15)$$

$$E_{\mathcal{T}_u}^k = \sum_{(x_l) \in \mathcal{T}_u} \sum_y f_k(\mathbf{x}_l, \mathbf{y}) P(y|\mathbf{x}_l, \mathbf{w}), \qquad (16)$$

are the expectation of feature value k for $\mathcal{S}_l$ and $\mathcal{T}_u$. The objective function is

$$\arg\max_{\mathbf{w}, K} \sum_{(x,y) \in \mathcal{S}_l} \sum_{k \in K} w_k f_k(\mathbf{x}, y) - \log z_w(\mathbf{x}) \qquad (17)$$

$$s.t. \sum_{k \in K} d(E_{\mathcal{S}_l}^k, E_{\mathcal{T}_u}^k) \leq \epsilon \qquad (18)$$

[Duan et al., 2009] took another aproach towards the same goal. They follow [Borgwardt et al., 2006] and use the Maximum Mean Discrepancy (MMD) criterion, to compare data distributions based on the distance between the means of samples from the two domains in the Reproducing Kernel Hilbert Space (RKHS),

$$\text{dist}_k(D^s, D^t) = \left\| \frac{1}{N^s} \sum_{i=1}^{N^s} \phi(\mathbf{x}_i^s) - \frac{1}{N^t} \sum_{i=1}^{N^t} \phi(\mathbf{x}_i^t) \right\|^2. \quad (19)$$

The authors integrate this distance with the standard SVM loss function

$$[k^*, f^*] = \arg\min \Omega(\text{dist}_k(D^s, D^t)) + \sigma SVM_{k,f}(D), \quad (20)$$

thus jointly finding (1) a kernel that minimize $\text{dist}_k(D^s, D^t)$ and (2) a SVM decision function, $SVM_{k,f}$, that separate the data in kernel space. To make this tractable they iteratively solve for (the parameters of) a parameterized mixture of kernel functions and the (α parameters of) the SVM loss function. They show improvements on the TRECVID dataset over related approaches.

[Blitzer et al., 2006] proposed Structural Correspondence Learning (SCL). SCL finds a feature representation that maximize the correspondence between unlabeled data in source and target domain, by leveraging *pivot* feature that behave similarly in both domains. For example, if the word

on the right is 'required' then the query word is likely a noun. This, then, helps disambiguate words such as 'signal', which can be both a noun and an adjective. This algorithm works on unsupervised data, and therefore doesn't maximize the correspondence between $P_t(Y|g_t(X))$ and $P_s(Y|g_s(X))$ directly, but rather on related tasks.

Several recent papers from the computer vision community pursue this idea. [Saenko et al., 2010], and later [Kulis et al., 2011] proposed variations on a metric learning formulation, where they not only learn a mapping that aligns the feature spaces but that also maximize class separation.

$$\min_W r(W) + \lambda \sum_{i,j} c_{i,j}^W. \qquad (21)$$

Here D^t and D^s are the target and source (labelled) data matrices respectively, with one sample per row. Saenko *et al*. chose to $r(W)$ and $c_i()$ as,

$$r(W) = \text{tr}(W) - \log\det(W)$$
$$c_{i,j}^W = ||x_i^t, x_j^s||_W \leq u \mid y_i^t = y_j^s$$
$$||x_i^t, x_j^s||_W \geq l \mid y_i^t \neq y_j^s, \qquad (22)$$

where $||a, b||_Q$ is the Mahalanobis distance between a and b with respect to matrix Q. With these constraints this formulations is known as information theoretic metric learning (ITML) [Davis et al., 2007], and the algorithmic contribution of the paper is to enforce that each pair of datapoints are from the source and target domain, respectively. They state that this is crucial to ensure a domain transfer transform is learned.

The authors note that since $\log\det(W)$ is only defined for positive definite matrices, one can decompose W as $W = L^T L$. The mapping, therefore, is symmetric since $(D^t)^T W D^s = (D^t)^T L^T L D^s = (L D^t)^T (L D^s)$. [Kulis et al., 2011] address this by changing regularizer to the squared frobenius norm. They also changed the constraints to encode similarity of data samples rather than the Mahalanobis distance. The new formulation becomes

$$r(W) = \frac{1}{2} ||W||_F^2$$
$$c_{i,j}^W = \max(0, x_i^t W x_j^s - u)^2 \mid y_i^t = y_j^s$$
$$\max(0, l - x_i^t W x_j^s)^2 \mid y_i^t \neq y_j^s. \qquad (23)$$

Kulis *et al*. show how to kernelize this formulation. Their method show minor improvements on the 'Saenko Items' dataset (Table 2) compared to [Saenko et al., 2010, DauméIII, 2009] and baseline methods.

[Jhuo et al., 2012] very recently proposed a formulation where the goal is to map the source data, by a matrix $W \in \mathbb{R}^{d \times d}$, to an intermediate representation where each transformed sample can be reconstructed by a linear combination of the target data samples,
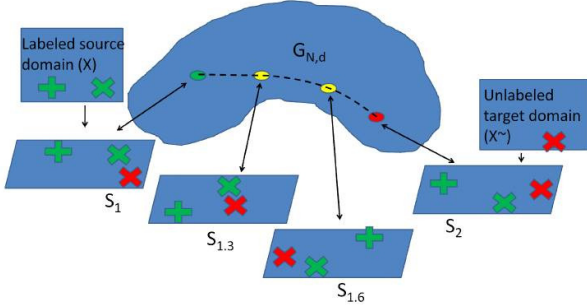
$$W D^s = D^t Z, \qquad (24)$$

Figure 3. Figure from [Gopalan et al., 2011] illustrating the proposed method.

where $Z \in \mathbb{R}^{n^{t,l} \times N^s}$. They propose the following formulation to solve for low rank solutions.

$$\min_{W,Z,E} \text{rank}(Z) + \alpha||E||_{2,1},$$
$$s.t. \; WS = TZ + W,$$
$$WW^T = I. \tag{25}$$

To solve this problem they relax the rank constraint to the nuclear norm and then apply a version of the Augmented Lagrange Multiplier (ALM) method [Lin et al., 2010].

Another recent method for computer vision also propose a mapping to a common representation [Gopalan et al., 2011]. Motivate by incremental learning, they create intermediate representation between the source and domain data by viewing the generative subspaces created from these domains as points on a Grassmanian manifold. Intermediate representations can then be recovered by sampling the geodesic path. The final feature representation is a stacked feature vector, from each location along the path. They use partial least squares to learn a model on this extended feature representation. Table 1 show the evaluation of this and several other discussed methods.

6. Transfer Learning

As mentioned in the introduction, transfer learning, sometimes called multi-task learning is different from DA. In transfer learning (TL) the joint probability of each task $\{P(Y_k, X)\}_{k=1}^m$ are different but there is only one marginal data distribution $P(X)$. Normally, the state space of the Y_k are assumed to be different, *e.g.* $\Omega(Y_1) \neq \Omega(Y_2)$. When learning class conditional models, $\{P(Y_k|X, \theta_k)\}_{k=1}^m$, it is typically assumed a common prior distribution of the variables $\theta_1 \ldots \theta_k \sim P_\Theta(\theta)$.

DA, while formally different, can be thought of as a special case of transfer learning with two tasks, one on the source, and one on the target, where $\Omega(Y_s) = \Omega(Y_t)$.

The classic paper by [DauméIII, 2009] can be viewed in this framework. Daumé propose a simple feature space augmentation by

$$\Phi^s(x) = \langle x, x, 0 \rangle \tag{26}$$
$$\Phi^t(x) = \langle x, 0, x \rangle \tag{27}$$

This 'frustratingly easy' method show promising performance doing named-entity recognition on several text datasets. Under a linear classification algorithm, this is equivalent to decomposing the model parameters for class k as $\sigma_c + \sigma_k$, where σ_c is shared by all domains. This formulation is basically identical to the one proposed by [Evgenious and Pontil, 2004] for the purpose of transfer learning. The authors [DauméIII, 2009] provide a different analysis in the paper, where they argue the similarity to the method of [Chelba and Acero, 2004].

7. Multi-Modal Learning

In this section, we discuss the concept of multi-modal learning. In this setting, correspondences are assumed to be on instance, rather then category, level. Also, here it is commonly assumed ample train data is available in both domains. Similarly to Sec. 5, the common goal of most methods is to estimate transformations L_t nad L_s so that $P_s((X^T L_s = x|Y = k) = P_t(X^T L_t = x|Y = k)$. This can be done by letting *e.g.* $L_s = I$, thus mapping the target domain to the source domain, or vice versa. One could also consider mapping both spaces into a common space. We will begin this section by reviewing Canonical Correlation Analysis, Principal Component Analysis, Linear Discriminant Analysis. We then consider recent work utilizing these methods [Sharma et al., 2012].

7.1. Background

In this section we recap the formulations of Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Canonical Correlation Analysis (CCA).

Principal Component Analysis: PCA, is a popular dimensionality reduction method that projects the data into direction of maximum variance. It can be derived as follows. Let $x_1 \ldots x_n$ be the input data. Let $\mathbf{w}_1$ be the desired projection direction. Let also $\mathbf{w}_1^T \mathbf{w}_1 = 1$. The mean of the projected data is $\mathbf{w}_1^T \bar{x}$, where $\bar{x} = \frac{1}{N} \sum_i^N x_i$. The variance of the projected data is

$$var(x) = \frac{1}{N} \sum_i^N (\mathbf{w}_1^T x_i - \mathbf{w}_1^T \bar{\mathbf{x}})^2, \tag{28}$$

which can be expressed in terms of the data covariance matrix,

$$S = \frac{1}{N} \sum_i^N (x_i - \bar{x})(x_i - \bar{x})^T, \tag{29}$$

Domain		Naive - train on all data	Metric learning (Supervised) [Kulis et al., 2011]		Manifold [Gopalan et al., 2011]		Low-rank (Supervised) [Jhuo et al., 2012]	A-SVM (Supervised) [Yang et al., 2007]
Source	Target	Naive	asymm	symm	Unsupervised	Supervised	RDALR	A-SVM
webcam	dslr	22.13 ± 1.2	25	27	19 ± 1.2	$\mathbf{37 \pm 2.3}$	32.89 ± 1.2	25.96 ± 0.7
dslr	webcam	32.17 ± 0.8	30	31	26 ± 0.8	$\mathbf{36 \pm 1.1}$	$\mathbf{36.85 \pm 1.3}$	33.01 ± 0.8
amazon	webcam	41.29 ± 1.3	48	44	39 ± 2.0	$\mathbf{57 \pm 3.5}$	50.71 ± 0.8	42.23 ± 0.9

Table 1. Evaluation of discussed method on the DA dataset introduced in [Kulis et al., 2011]. The naive method train on $\mathcal{S}_l \cup \mathcal{T}_l$. The methods are trained on 8 images per category (if source is *webcam* or *dslr* or 20 image (for *amazon*) from the source domain and 3 per category for the target domain. The best result for each experiment is marked in bold.

as $u^T S u$. We now maximize the projected variance with respect to w_1. The constrained maximization problem can be written as

$$\mathbf{w_1} = \arg\max_{\mathbf{w_1}} \mathbf{w_1}^T \Sigma_{tt} \mathbf{w_1}$$
$$\text{s.t. } \mathbf{w_1}^T \mathbf{w_1} = 1 \qquad (30)$$

Linear Discriminant Analysis: While PCA is very popular for unsupervised data dimensionality reduction purposes it is agnostic to class, and might project data in directions that are *unsuitable* for class discrimination. Linear Discriminant Analysis (LDA) finds projection directions by minimizing the within class scatter matrix while maximizing the between class scatter. LDA can be derived as follows. Let $\mathcal{X}_1 = \{\mathbf{x}_1^1, \dots \mathbf{x}_{l_1}^1\}$ and $\mathcal{X}_2 = \{\mathbf{x}_1^2, \dots \mathbf{x}_{l_2}^2\}$ be samples from two different classes. The data projection direction $\mathbf{w_1}$ is given by solving

$$\mathbf{w_1} = \arg\max_{\mathbf{w_1}} \mathbf{w_1}^T S_B \mathbf{w_1}$$
$$\text{s.t. } \mathbf{w_1}^T S_W \mathbf{w_1} = 1 \qquad (31)$$

where

$$S_B := (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T$$
$$S_W := \sum_{i=1,2} \sum_{\mathbf{x} \in \mathcal{X}_\rangle} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \qquad (32)$$

are the within and between scatter matrices, and m_i is the mean of samples in $\mathcal{X}_\rangle$. Note the very similar forms of Eq. (30) and Eq. (31). Eq. (30) is a regular eigenvalue problem, while Eq. (31) is a generalized eigenvalue problem.

Canonical Correlation Analysis: Developed by [Hotelling, 1936] CCA is a data analysis and dimensionality reduction method, that can be though of as a multimodal extension to PCA. CCA finds basis vectors for two sets of variables such that the correlations between the projections of the variables onto these basis variables are mutually maximized. Using notation from Sec. 2.1, we let D^s be a n^s by d^s matrix with rows x_i^s, and D^t similarly. CCA

finds projection directions w^s and w^t to maximize the correlation between the projected data. More formally, it finds projection directions by solving the following optimization problem

$$\rho = \max_{\mathbf{w^t}, \mathbf{w^s}} corr(D^t \mathbf{w^t}, D^s \mathbf{w^s}) \qquad (33)$$
$$= \max_{\mathbf{w^t}, \mathbf{w^s}} \frac{\langle D^t \mathbf{w^t}, D^s \mathbf{w^s} \rangle}{||D^t \mathbf{w^t}|| ||D^s \mathbf{w^s}||} \qquad (34)$$
$$= \max_{\mathbf{w^t}, \mathbf{w^s}} \frac{\mathbf{w^t}^T \Sigma_{ts} \mathbf{w^s}}{\sqrt{w_x^T \Sigma_{tt} \mathbf{w^t} \mathbf{w^s}^T \Sigma_{ss} \mathbf{w^s}}} \qquad (35)$$

where Σ_{ij} is the covariance matrix between data in domains i and j, $i, j \in \{s, t\}$. Note that this formulation requires the same number of samples from each domain, but not the same dimensionality. We also note that the optimization can be written as a constrained optimization as

$$\max_{\mathbf{w^t}, \mathbf{w^s}} \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix}^T \begin{bmatrix} 0 & \Sigma_{ts} \\ \Sigma_{st} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix} \qquad (36)$$
$$s.t. \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix}^T \begin{bmatrix} \Sigma_{tt} & 0 \\ 0 & \Sigma_{ss} \end{bmatrix} \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix} = 1 \qquad (37)$$

The optimization problem Eq. (33) (or Eq. (36)) can be formulated as a generalized eigenvalue problem which can be solved as efficiently as regular eigenvalue problems [Hardoon et al., 2004]. For more details [Hardoon et al., 2004] provides an excellent analysis on CCA and the kernelized version KCCA.

7.2. Generalized Multiview Analysis

GMA was proposed by [Sharma et al., 2012] as a unifying framework for learning multi modal discriminative linear projections. They argue that methods such as PCA and LDA do not handle multi-view data. On, the other hand, methods such as CCA is not supervised. Other methods such as SVM-2K [Farquhar et al., 2005], CDSVM [Jiang et al., 2008], ASVM [Yang et al., 2007] meet these criteria but do not generalize well to unseen classes. The unifying framework of GMA is

$$\max_{\mathbf{w^t},\mathbf{w^s}} \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix}^T \begin{bmatrix} A_t & \alpha Z_t Z_s^T \\ \alpha Z_s Z_t^T & \mu A_s \end{bmatrix} \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix} \quad (38)$$

$$s.t. \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix}^T \begin{bmatrix} B_t & 0 \\ 0 & \gamma B_s \end{bmatrix} \begin{bmatrix} \mathbf{w^t} \\ \mathbf{w^s} \end{bmatrix} = 1. \quad (39)$$

PCA can be recovered (in the i'th view) through this framework by setting $A_i = \Sigma_{ii}, B_i = I$. Similarly, LDA can be recovered $A_i = S^B, B_i = S^W$, with S^B and S^W defined as above. CCA can be recovered as $A_i = 0, B_i = D^i W^i (X^i)^T$, and $Z_i = X_i$.

Using this framework they propose two methods, Generalized Multiview LDA and Generalized Multiview Marginal Fisher Analysis (GMMFA). Here we recap only GMLDA. As noted above, LDA in the i'th view can be achieved by setting $A_i = S^B, B_i = S^W$. By setting $Z_i = M_i$, a matrix with columns that are class means, they enforce class mean alignment across classes. The authors also note that the two step process of LDA + CCA or vice versa needs to be considered as a baseline method. Similar approaches include [Rasiwasia et al., 2010] who introduced *semantic correlation matching*, which uses logistic regression to combine CCA with semantic matching. They also introduce the WikiText data set (Table 2).

The proposed method outperforms all baseline methods on multiPIE and VOC2007 and is on par with the domain specific approach by [Rasiwasia et al., 2010] on the WikiText dataset.

Name	Description	Instance or Class correspondence
MultiPIE	Face recognition data set containing face images under different Pose Illumination and Expression	Instance
WikiText	Each item is represented using a text and an image.	Instance
Pascal VOC 2007	5011 / 4952 (training / testing) image-tag pairs	Instance
Office dataset	Object images from Amazon, SLR and webcam	Class (subset with instance)

Table 2. Computer vision datasets for domain adaptation method benchmark.

References

Alessandro Bergamo and Lorenzo Torresani. Exploiting weakly-labeled web images to improve object classcation: a domain adaptation approach. *NIPS*, 2010. 1, 4

John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. *Conference on Empirical Methods in Natural Language Processing*, 2006. 5

Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Conf. on Computational learning theory*, 1998. 1

K. Borgwardt, A. Gretton, M. Rasch, H.-P. Kriegel, B. Schlkopf, and A. Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 2006. 5

Steve Branson, Pietro Perona, and Serge Belongie. Strong supervision from weak annotation: Interactive training of deformable part models. In *IEEE International Conference on Computer Vision (ICCV)*, 2011. 1

Ciprian Chelba and Alex Acero. Adaptation of maximumentropy capitalizer: Little data can helpalot. *Computer Speech And Language*, 2004. 1, 4, 6

Hal DauméIII. Frustratingly easy domain adaptation. *CoRR*, 2009. 1, 5, 6

Hal DauméIII and Daniel Marcu. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research*, 2006. 1, 3, 4

Jason Davis, Brian Kulis, Suvrit Sra, and Inderjit Dhillon. Information-theoretic metric learning. In *in NIPS 2006 Workshop on Learning to Compare Examples*, 2007. 5

Thomas G. Dietterich, Richard H. Lathrop, and Toms Lozano-Prez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31 – 71, 1997. ISSN 0004-3702. 1

M. Dixit, N. Rasiwasia, and N. Vasconcelos. Adapted gaussian models for image classification. *CVPR*, 2011. 2, 4

Lixin Duan, Ivor W. Tsang, Dong Xu, and Stephen J. Maybank. Domain transfer svm for video concept detection. *CVPR*, 2009. 1, 5

Theodoros Evgenious and Massimiliano Pontil. Regularized multi-task learning. *international conference on Knowledge discovery and data mining*, 2004. 6

Jason D.R. Farquhar, David R. Hardoon, Hongying Meng, and John Shawe-Taylor. Two view learning: Svm-2k, theory and practice. *NIPS*, 2005. 7

Li Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *PAMI*, 2006. 1

Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. *ICCV*, 2011. 1, 6, 7

David R. Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computing*, 2004. 7

Harold Hotelling. Relations between two sets of variates. *Biometrika*, 1936. 7

Jiayuan Huang, Arthur Gretton, Bernhard Schlkopf, Alexander J. Smola, and Karsten M. Borgwardt. Correcting sample selection bias by unlabeled data. In *In NIPS*. MIT Press, 2007. 3

I-Hong Jhuo, Bong Liu, D.T. Lee, and Shih-Fu Chang. Robust visual domain adaptation with low-rank reconstruction. *CVPR*, 2012. 1, 5, 7

Jing Jiang. A literature survey on domain adaptation of statistical classifiers. *Tech report*, 2008. 1, 2, 5

Wei Jiang, Eric Zavesky, Shih-Fu Chang, and Alex Loui. Cross-domain learning methods for high-level visual concept classification. *ICIP*, 2008. 4, 7

Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Active learning with gaussian processes for object categorization. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007. 1

B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. *CVPR*, 2011. 1, 5, 7

P. Perona L. Fei-Fei, R. Fergus. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 2007. 2

Yi Lin, Yoonkyung Lee, and Grace Wahbe. Support vector machines for classificaion in nonstandard situations. *Machine learning*, 2002. 3

Zhouchen Lin, Minming Chen, and Yi Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *Analysis*, 2010. 6

S. Jialin Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data*, 2009. 1

N. Rasiwasia, J. C. Pereira, E. Coviello, G. Doyle, G. R. G. Lanckriet, R. Levy, and N. Vasconcelos. A new approach to cross-modal multimedia retrieval. *Proceedings of the international conference on Multimedia*, 2010. 8

Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 2000. 2, 4

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visaul category models to new domains. *ECCV*, 2010. 1, 5

Eepkumar Satpal and Sunita Sarawagi. Domain adaptation of conditional probability models via feature subsetting, 2006. 5

A. Sharma, A. Kumar, and H. DauméIII. Generalized multiview analysis: A discriminative latent space. *CVPR*, 2012. 6, 7

Hidetoshi Shimodaira. Improving predictive inference under covariance shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 2000. 1, 3

Masashi Sugiyama and Klaus-Robert Mülcer. Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, 2005. 3

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1999. 2

D.A.Reynolds W. M. Campbell, D.E.Sturim. Support vector machines using gmm supervectors for speaker verificaion. *Signal Processing Letters*, 2006. 4

Peter Welinder, Steve Branson, Serge Belongie, and Pietro Perona. The multidimensional wisdom of crowds. In *Neural Information Processing Systems Conference (NIPS)*, 2010. 1

Jun Yang, Rong Yan, and Alexander G. Hauptmann. Cross-domain video concept detection using adaptive svms. *International Conference on Multimedia*, 2007. 4, 7

Chunnam John Yu and Thorsten Joachims. Learning structural svms with latent variables, 2009. 1