Chapter 21

Transfer Learning

Sinno Jialin Pan, Nanyang Technological University, Singapore, sinnopan@ntu.edu.sg

1.1 Introduction

Supervised machine learning techniques have already been widely studied and applied to various real-world applications. However, most existing supervised algorithms work well only under a common assumption: the training and test data are represented by the same features and drawn from the same distribution. Furthermore, the performance of these algorithms heavily rely on collecting high quality and sufficient labeled training data to train a statistical or computational model to make predictions on the future data [86, 57, 132]. However, in many real-world scenarios, labeled training data are in short supply or can only be obtained with expensive cost. This problem has become a major bottleneck of making machine learning methods more applicable in practice.

In the last decade, semi-supervised learning [167, 27, 89, 20, 63] techniques have been proposed to address the labeled data sparsity problem by making use of a large amount of unlabeled data to discover an intrinsic data structure to effectively propagate label information. Nevertheless, most semi-supervised methods require that the training data, including labeled and unlabeled data, and the test data are both from the same domain of interest, which implicitly assumes the training and test data are still represented in the same feature space and drawn from the same data distribution.

Instead of exploring unlabeled data to train a precise model, active learning, which is another branch in machine learning for reducing annotation effort of supervised learning, tries to design an active learner to pose queries, usually in the form of unlabeled data instances to be labeled by an oracle (e.g., a human annotator). The key idea behind active learning is that a machine learning algorithm can achieve greater accuracy with fewer training labels if it is allowed to choose the data from which it learns [71, 121]. However, most active learning methods assume that there is a budget for the active learner to pose queries in the domain of interest. In some real-world applications, the budget may be quite limited, which means that the labeled data queried by active learning may not be sufficient enough to learn an accurate classifier in the domain of interest.

Transfer learning, in contrast, allows the domains, tasks, and distributions used in training and testing to be different. The main idea behind transfer learning is to borrow labeled data or extract knowledge from some related domains to help a machine learning algorithm to achieve greater performance in the domain of interest [130, 97]. Thus, transfer learning can be referred to as a different strategy for learning models with minimal human supervision, compared to semi-supervised and active learning. In the real world, we can observe many examples of transfer learning. For example, we may find that learning to recognize apples might help to recognize pears. Similarly, learning to play the electronic organ may help facilitate learning the piano. Furthermore, in many engineering applications, it is expensive or impossible to collect sufficient training data to train models for use in each domain of interest. It would be more practical if one could reuse the training data which have been collected in some related domains/tasks or the knowledge that is already extracted from

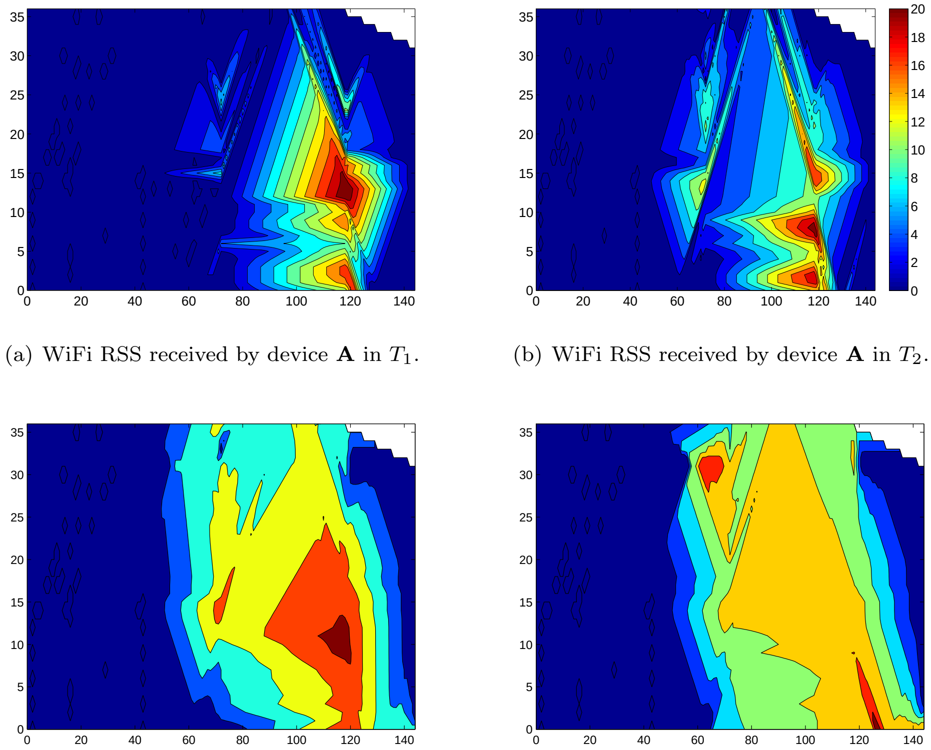
TABLE 1.1: Cross-domain sentiment classification examples: reviews of *Electronics* and *Video Games*. Boldfaces are domain-specific words that occur much more frequently in one domain than in the other one. "+" and "**-**" denote positive and negative sentiment respectively.

	Electronics	*Video Games*
+	**Compact**; easy to operate; very good picture quality; looks **sharp**!	A very good game! It is action packed and full of excitement. I am very much **hooked** on this game.
+	I purchased this unit from Circuit City and I was very excited about the quality of the picture. It is really nice and **sharp**.	Very **realistic** shooting action and good plots. We played this and were **hooked**.
-	It is also quite **blurry** in very dark settings. I will never buy HP again.	The game is so **boring**. I am extremely unhappy and will probably never buy UbiSoft again.

some related domains/tasks to learn a precise model for use in the domain of interest. In such cases, *knowledge transfer* or *transfer learning* between tasks or domains become more desirable and crucial.

Many diverse examples in knowledge engineering can be found where transfer learning can truly be beneficial. One example is sentiment classification, where our goal is to automatically classify reviews on a product, such as a brand of camera, into polarity categories (e.g., positive, negative or neural). In literature, supervised learning methods [100] have proven to be promising and widely used in sentiment classification. However, these methods are domain dependent, which means that a model built on one domain (e.g., reviews on a specific product with annotated polarity categories) by using these methods may perform poorly on another domain (e.g., reviews on another specific product without polarity categories). The reason is that one may use different domain-specific words to express opinions in different domains. Table 1.1 shows several review sentences of two domains: *Electronics* and *Video Games*. In the *Electronics* domain, one may use the words like "compact" and "sharp" to express positive sentiment and use "blurry" to express negative sentiment. While in the *Video Game* domain, the words like "hooked" and "realistic" indicate positive opinions and the word "boring" indicates negative opinion. Due to the mismatch of domain-specific words between domains, a sentiment classifier trained on one domain may not work well when directly applied to other domains. Therefore, cross-domain sentiment classification algorithms are highly desirable to reduce domain dependency and manually labeling cost by transferring knowledge from related domains to the domain of interest [18, 92, 51].

The need for transfer learning may also arise in applications of wireless sensor networks, where wireless data can be easily outdated over time or very different received by different devices. In these cases, the labeled data obtained in one time period or on one device may not follow the same distribution in a later time period or on another device. For example, in indoor WiFi-based localization, which aims to detect a mobile's current location based on previously collected WiFi data, it is very expensive to calibrate WiFi data for building a localization model in a large-scale environment because a user needs to label a large collection of WiFi signal data at each location. However, the values of WiFi signal strength may be a function of time, device or other dynamic factors. As shown in Figure 1.1, values of received signal strength (RSS) may differ across different time periods and mobile devices. As a result, a model trained in one time period or on one device may estimate locations poorly in another time period or on another device. To reduce the re-calibration effort, we might wish to adapt the localization model trained in one time period (the source domain)



(a) WiFi RSS received by device **A** in T_1.

(b) WiFi RSS received by device **A** in T_2.

(c) WiFi RSS received by device **B** in T_1.

(d) WiFi RSS received by device **B** in T_2.

FIGURE 1.1: Contours of RSS values over a 2-dimensional environment collected from a same access point in different time periods and received by different mobile devices. Different colors denote different values of signal strength.

for a new time period (the target domain), or to adapt the localization model trained on a mobile device (the source domain) for a new mobile device (the target domain) with little or without additional calibration [152, 91, 98, 165].

As a third example, transfer learning has shown to be promising for defect prediction in the area of software engineering, where the goal is to build a prediction model from data sets mined from software repositories, and the model is used to identify software defects. In the past few years, numerous effective software defect prediction approaches based on supervised machine learning techniques have been proposed and received a tremendous amount of attention [66, 83]. In practice, cross-project defect prediction is necessary. New projects often do not have enough defect data to build a prediction model. This cold-start is a well-known problem for recommender systems [116] and can be addressed by using cross-project defect prediction to build a prediction model using data from other projects. The model is then applied to new projects. However, as reported by some researchers that cross-project defect prediction often yields poor performance [170]. One of the main reasons for the poor cross-project prediction performance is the difference between the data distributions of the source and target projects. To improve the cross-project prediction performance with little additional human supervision, transfer learning techniques are again desirable, and have proven to be promising [87].

Generally speaking, transfer learning can be classified into two different fields: 1) transfer learning for classification, regression and clustering problems [97], and 2) transfer learning for reinforcement learning tasks [128]. In this chapter, we focus on transfer learning in data classification and its real-world applications. Furthermore, as first introduced in a survey article [97], there are three main research issues in transfer learning: 1) What to transfer, 2) How to transfer, and 3) When to transfer. Specifically, "What to transfer" asks which part of knowledge can be extracted and transferred across domains or tasks. Some knowledge is domain- or task- specific, which may not be observed in other domains or tasks, while some knowledge is common shared by different domains or tasks, which can be treated as a bridge for knowledge transfer across domains or tasks. After discovering which knowledge can be transferred, learning algorithms need to be developed to transfer the knowledge, which corresponds to the "how to transfer" issue. Different knowledge-transfer strategies lead to specific transfer learning approaches. "When to transfer" asks in which situations, transferring skills should be done. Likewise, we are interested in knowing in which situations, knowledge should not be transferred. In some situations, when the source domain and target domain are not related to each other, brute-force transfer may be unsuccessful. In the worst case, it may even hurt the performance of learning in the target domain, a situation which is often referred to as *negative transfer*. Therefore, the goal of "When to transfer" is to avoid *negative transfer* and then ensure *positive transfer*.

The rest of this chapter is organized as follows. In Section 1.2, we start by giving an overview of transfer learning including its brief history, definitions and different learning settings. In Sections 1.3-1.4, we summarize approaches into different categories based on two transfer learning settings, namely homogenous transfer learning and heterogeneous transfer learning. In Sections 1.5-1.6, we discuss the negative transfer issue and other research issues of transfer learning. After that, we show diverse real-world applications of transfer learning in Section 1.7. Finally, we give concluding remarks in Section 1.8.

1.2 Transfer Learning Overview

1.2.1 Background

The study of transfer learning is motivated by the fact that people can intelligently apply knowledge learned previously to solve new problems faster or with better solutions [47]. For example, if one is good at coding in C++ programming language, he/she may learn Java programming language fast. This is because both C++ and Java are object-oriented programming (OOP) languages, and share similar programming motivations. As another example, if one is good at playing table tennis, he/she may learn playing tennis fast because the skill sets of these two sports are overlapping. Formally, from a psychological point of view, the definition of transfer learning or learning of transfer is the study of the dependency of human conduct, learning, or performance on prior experience. More than 100 years ago, researchers has already explored how individuals would transfer in one context to another context that share similar characteristics [129].

The fundamental motivation for transfer learning in the field of machine learning is the need for lifelong machine learning methods that retain and reuse previously learned knowledge such that intelligent agencies can adapt to new environment or novel tasks effectively and efficiently with little human supervision. Informally, the definition of transfer learning in the field of machine learning is the ability of a system to recognize and apply knowledge and skills learned in previous domains or tasks to new domains or novel domains, which share some commonality.

1.2.2 Notations and Definitions

In this section, we follow the notations introduced in [97] to describe the problem statement of transfer learning. A *domain* $\mathcal{D}$ consists of two components: a feature space $\mathcal{X}$ and a marginal probability distribution $P(x)$, where $x \in \mathcal{X}$. In general, if two domains are different, then they may have different feature spaces or different marginal probability distributions. Given a specific domain, $\mathcal{D} = \{\mathcal{X}, P(x)\}$, a *task* $\mathcal{T}$ consists of two components: a label space $\mathcal{Y}$ and a predictive function $f(\cdot)$ (denoted by $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$). The function $f(\cdot)$ is a predictive function that can be used to make predictions on unseen instances $\{x^*\}$'s. From a probabilistic viewpoint, $f(x)$ can be written as $P(y|x)$. In classification, labels can be binary, i.e., $\mathcal{Y} = \{-1, +1\}$, or discrete values, i.e., multiple classes.

For simplicity, we only consider the case where there is one source domain $\mathcal{D}_S$, and one target domain $\mathcal{D}_T$, as this is by far the most popular of the research works in the literature. The issue of knowledge transfer from multiple source domains will be discussed in Section 1.6. More specifically, we denote $\mathbf{D}_S = \{(x_{S_i}, y_{S_i})\}_{i=1}^{n_S}$ the *source domain data*, where $x_{S_i} \in \mathcal{X}_S$ is the data instance and $y_{S_i} \in \mathcal{Y}_S$ is the corresponding class label. Similarly, we denote $\mathbf{D}_T = \{(x_{T_i}, y_{T_i})\}_{i=1}^{n_T}$ the target domain data, where the input x_{T_i} is in $\mathcal{X}_T$ and $y_{T_i} \in \mathcal{Y}_T$ is the corresponding output. In most cases, $0 \le n_T \ll n_S$. Based on the notations defined above, the definition of transfer learning can be defined as follows [97],

Definition 1. *Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$,* transfer learning *aims to help improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$.*

In the above definition, a domain is a pair $\mathcal{D} = \{\mathcal{X}, P(x)\}$. Thus the condition $\mathcal{D}_S \neq \mathcal{D}_T$ implies that either $\mathcal{X}_S \neq \mathcal{X}_T$ or $P(x_S) \neq P(x_T)$. Similarly, a task is defined as a pair $\mathcal{T} = \{\mathcal{Y}, P(y|x)\}$. Thus the condition $\mathcal{T}_S \neq \mathcal{T}_T$ implies that either $\mathcal{Y}_S \neq \mathcal{Y}_T$ or $P(y_S|x_S) \neq P(y_T|x_T)$. When the target and source domains are the same, i.e. $\mathcal{D}_S = \mathcal{D}_T$, and their learning tasks are the same, i.e., $\mathcal{T}_S = \mathcal{T}_T$, the learning problem becomes a traditional machine learning problem. Based on whether the feature spaces or label spaces are identical or not, we can further categorize transfer learning into two settings: 1) homogenous transfer learning, and 2) heterogenous transfer learning. In the following two sections, we give the definitions of these two settings and review their representative methods respectively.

1.3 Homogenous Transfer Learning

In this section, we first give an definition of homogenous transfer learning as follows,

Definition 2. *Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$,* homogenous transfer learning *aims to help improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{X}_S \bigcap \mathcal{X}_T \neq \emptyset$ and $\mathcal{Y}_S = \mathcal{Y}_T$, but $P(x_S) \neq P(x_T)$ or $P(y_S|x_S) \neq P(y_T|x_T)$.*

Based on the above definition, in homogenous transfer learning, the feature spaces between domains are overlapping, and the label spaces between tasks are identical. The difference between domains or tasks is caused by the marginal distributions or predictive distributions. Approaches to homogenous transfer learning can be summarized into four categories: 1) instance-based approach, 2) feature-representation-based approach, 3) model-parameter-based approach, and 4) relational-information-based approach. In the following sections, we describe the motivations of these approaches and introduce some representative methods of each approach.

1.3.1 Instance-based Approach

A motivation of the instance-based approach is that although the source domain labeled data cannot be reused directly, part of them can be reused for the target domain after re-weighting or re-sampling. An assumption behind the instance-based approach is that the source and target domains have a lot of overlapping features, which means that the domains share the same or similar support. Based on whether labeled data are required or not in the target domain, the instance-based approach can be further categorized into two contexts: 1) no target labeled data are available, and 2) a few target labeled data are available.

1.3.1.1 Case I: No Target Labeled Data

In the first context, no labeled data are required but a lot of unlabeled data are assumed to be available in the target domain. In this context, most instance-based methods are deployed based on an assumption that $P_S(y|x) = P_T(y|x)$, and motivated by importance sampling. To explain why importance sampling is crucial for this context of transfer learning, we first review the learning framework of empirical risk minimization (ERM) [132]. Given a task of interest, i.e., the target task, the goal of ERM is to learn an optimal parameter θ^* by minimizing the expected risk as follows,

$$\theta^* = \arg\min_{\theta\in\Theta} \mathbb{E}_{(x,y)\in P_T}[l(x,y,\theta)], \tag{1.1}$$

where $l(x,y,\theta)$ is a loss function that depends on the parameter θ. Since no labeled data are assumed to be available in the target domain, it is impossible to optimize (1.1) over target domain labeled data. It can be proved that the optimization problem (1.1) can be rewritten as follows,

$$\theta^* = \arg\min_{\theta\in\Theta} \mathbb{E}_{(x,y)\sim P_S}\left[\frac{P_T(x,y)}{P_S(x,y)} l(x,y,\theta)\right], \tag{1.2}$$

which aims to learn the optimal parameter θ^* by minimizing the weighted expected risk over source domain labeled data. As assumed $P_S(y|x) = P_T(y|x)$, by decomposing the joint distribution $P(x,y) = P(y|x)P(x)$, we obtain $\frac{P_T(x,y)}{P_S(x,y)} = \frac{P_T(x)}{P_S(x)}$. Hence, (1.2) can be further rewritten as

$$\theta^* = \arg\min_{\theta\in\Theta} \mathbb{E}_{(x,y)\sim P_S}\left[\frac{P_T(x)}{P_S(x)} l(x,y,\theta)\right], \tag{1.3}$$

where a weight of a source domain instance x is the ratio of the target and source domain marginal distributions at the data point x. Given a sample of source domain labeled data $\{(x_{S_i}, y_{S_i})\}_{i=1}^{n_S}$, by denoting $\beta(x) = \frac{P_T(x)}{P_S(x)}$, a regularized empirical objective of (1.3) can be formulated as

$$\theta^* = \arg\min_{\theta\in\Theta} \sum_{i=1}^{n_S} \beta(x_{S_i}) l(x_{S_i}, y_{S_i}, \theta) + \lambda\Omega(\theta), \tag{1.4}$$

where $\Omega(\theta)$ is a regularization term to avoid overfitting on the training sample. Therefore, a research issue on applying the ERM framework to transfer learning is how to estimate the weights $\{\beta(x)\}$'s. Intuitively, a simple solution is to first estimate $P_T(x)$ and $P_S(x)$ respectively, and thus calculate the ratio $\frac{P_T(x)}{P_S(x)}$ for each source domain instance x_{S_i}. However, density estimations on $P_T(x)$ and $P_S(x)$ are difficult, especially when data are high-dimensional and the data size is small. An alterative solution is to estimate $\frac{P_T(x)}{P_S(x)}$ directly.

In literature, there exist various ways to estimate $\frac{P_T(x)}{P_S(x)}$ directly. Here we introduce three representative methods. For more information on this context, readers may refer to [104]. Zadrozny [158] assumed that the difference in data distributions is caused by the data generation process. Specifically, the source domain data are assumed to be sampled from the target domain data following a rejection sampling process. Let $s \in \{0, 1\}$ be a selector variable to denote whether an instance in the target domain is selected to generate the source domain data or not, i.e., $s = 1$ denotes the instance is selected, otherwise unselected. In this way, the distribution of the selector variable maps the target distribution onto the source distribution as follows,

$$P_S(x) \propto P_T(x)P(s = 1|x).$$

Therefore, the weight $\beta(x)$ is propositional to $\frac{1}{P(s=1|x)}$. To estimate $\frac{1}{P(s=1|x)}$, Zadrozny proposed to consider all source domain data with labels 1's and all target domain data with labels 0's, and train a probabilistic classification model on this pseudo classification task to estimate $P(s = 1|x)$.

Huang *et al.* [59] proposed a different algorithm known as kernel-mean matching (KMM) to learn $\frac{P_S(x)}{P_T(x)}$ directly by matching the means between the source and target domain data in a reproducing-kernel Hilbert space (RKHS) [117]. Specifically, KMM makes use of Maximum Mean Discrepancy (MMD) introduced by Gretton *et al.* [52] as a distance measure between distributions. Given two samples, based on MMD, the distance between two sample distributions is simply the distance between the two mean elements in a RKHS. Therefore, the objective of KMM can be written as

$$\begin{aligned} \arg\min_{\beta} \quad & \left\| \frac{1}{n_S}\sum_{i=1}^{n_S}\beta(x_{S_i})\Phi(x_{S_i}) - \frac{1}{n_T}\sum_{j=1}^{n_T}\Phi(x_{T_j}) \right\|_{\mathcal{H}}, \\ s.t \quad & \beta(x_{S_i}) \in [0,\ B] \text{ and } \left| \frac{1}{n_S}\sum_{i=1}^{n_S}\beta(x_{S_i}) - 1 \right| \leq \epsilon. \end{aligned} \tag{1.5}$$

where B is the parameter to limit the discrepancy between $P_S(x)$ and $P_T(x)$, and ϵ is the nonnegative parameter to ensure the reweighted $P_S(x)$ to be close to a probability distribution. It can be showed that the optimization problem (1.5) can be transformed to a quadratic programming (QP) problem, and the optimal solutions $\{\beta(x_{S_i})\}$'s of (1.5) are equivalent to the ratio values $\left\{\frac{P_S(x_{S_i})}{P_T(x_{S_i})}\right\}$'s of (1.3) to be estimated.

As a third method, Sugiyama *et al.* [127] assumed that the ratio $\beta(x)$ can be estimated by the following linear model,

$$\widetilde{\beta}(x) = \sum_{\ell=1}^{b}\alpha_\ell\psi_\ell(x),$$

where $\{\psi_\ell(x)\}_{\ell=1}^{b}$ are the basic functions which are predefined, and the coefficients $\{\alpha_\ell\}_{\ell=1}^{b}$ are the parameters to be estimated. In this way, the problem of estimating $\beta(x)$ is transformed to the problem of estimating the parameters $\{\alpha_\ell\}_{\ell=1}^{b}$. By denoting $\widetilde{P}_T(x) = \widetilde{\beta}(x)P_S(x)$, the parameters can be learned by solving the following optimization problem,

$$\arg\min_{\{\alpha_\ell\}_{\ell=1}^{b}} l(P_T(x), \widetilde{P}_T(x)),$$

where $l(\cdot)$ is a loss function of the estimated target distribution $\widetilde{P}_T(x)$ to the ground truth target distribution $P_T(x)$. Different loss functions lead to various specific algorithms. For instance, Sugiyama *et al.* [127] proposed to use the Kullback-Leibler divergence as the loss function, while Kanamori *et al.* [65] proposed to use the least-squared loss as the loss function. Note that the ground truth of $P_S(x)$ and $P_T(x)$ are unknown. However, as shown in [127, 65], $P_S(x)$ and $P_T(x)$ can be eliminated when optimizing the parameters $\{\alpha_\ell\}_{\ell=1}^b$.

1.3.1.2 Case II: A Few Target Labeled Data

In the second context of the instance-based approach, a few target labeled data are assumed to be available. Different from the approaches in the first context, in this context, most approaches are proposed to weight the source domain data based on their contributions to the classification accuracy for the target domain.

Wu and Dietterich [142] integrated the source domain labeled data together with a few target domain labeled data into the standard Support Vector Machine (SVM) framework for improving the classification performance for the target domain as follows,

$$\begin{aligned} \arg\min_{w,\xi_S,\xi_T} \quad & \frac{1}{2}\|w\|_2^2 + \lambda_T \sum_{i=1}^{n_{T_l}} \xi_{T_i} + \lambda_S \sum_{i=1}^{n_S} \gamma_i \xi_{S_i}, \\ s.t. \quad & y_{S_i} w^\top x_{S_i} \geq 1 - \xi_{S_i},\ \xi_{S_i} \geq 0,\ i = 1, ..., n_S, \\ & y_{T_i} w^\top x_{T_i} \geq 1 - \xi_{T_i},\ \xi_{T_i} \geq 0,\ i = 1, ..., n_{T_l}, \end{aligned} \tag{1.6}$$

where n_{T_l} is the number of target domain labeled data, w is the model parameter, ξ_S and ξ_T are the slack variables to absorb errors on the source and target domain data respectively, λ_S and λ_T are the tradeoff parameters to balance the impact of different terms in the objective, and γ_i is the weight on the source domain instance x_{S_i}. There are various ways to set the values of $\{\gamma_i\}$'s. In [142], Wu and Dietterich proposed to simply set $\gamma_i = 1$ for each data point in the source domain. Jiang and Zhai [62] proposed a heuristic method to remove the "misleading" instances from the source domain, which is equivalent to setting $\gamma_i = 0$ for all "misleading" source domain instances and $\gamma_i = 1$ for the remaining instances. Note that the basic classifier used in [62] is a probabilistic model instead of SVM, but the idea is similar.

Dai *et al.* [38] proposed a boosting algorithm, known as TrAdaBoost, for transfer learning. TrAdaBoost is an extension of the AdaBoost algorithm [49]. The basic idea of TrAdaBoost attempts to iteratively re-weight the source domain data to reduce the effect of the "bad" source data while encourage the "good" source data to contribute more to the target domain. Specifically, for each round of boosting, TrAdaBoost uses the same strategy as AdaBoost to update weights of the target domain labeled data, while proposes a new mechanism to decrease the weights of misclassified source domain data.

1.3.2 Feature-representation-based Approach

As described in the previous section, for the instance-based approach, a common assumption is that the source and target domains have a lot of overlapping features. However, in many real-world applications, the source and target domains may only have some overlapping features, which means that many features may only have support in either the source or target domain. In this case, most instance-based methods may not work well. The feature-representation-based approach to transfer learning is promising to address this issue. An intuitive idea behind the feature-representation-based approach is to learn a "good" feature representation for the source and target domains such that based on the new representation, source domain labeled data can be reused for the target domain. In this sense, the knowledge to be transferred across domains is encoded into the learned feature representation. Specifically, the feature-representation-based approach aims to learn a mapping

$\varphi(\cdot)$ such that the difference between the source and target domain data after transformation, $\{\varphi(x_{S_i})\}$'s and $\{\varphi(x_{T_i})\}$'s, can be reduced. In general, there are two ways to learn such a mapping $\varphi(\cdot)$ for transfer learning. One is to encode specific domain or application knowledge into learning the mapping, the other is to propose a general method to learn the mapping without taking any domain or application knowledge into consideration.

1.3.2.1 Encoding Specific Knowledge for Feature Learning

In this section, we use sentiment classification as an example to present how to encode domain knowledge into feature learning. In sentiment classification, a domain denotes a class of objects or events in the world. For example, different types of products, such as *books*, *dvds* and *furniture*, can be regarded as different domains. Sentiment data are the text segments containing user opinions about objects, events and their properties of the domain. User sentiment may exist in the form of a sentence, paragraph or article, which is denoted by x_j. Alternatively, it corresponds with a sequence of words $v_1 v_2 ... v_{x_j}$, where w_i is a word from a vocabulary V. Here, we represent user sentiment data by a bag of words with $c(v_i, x_j)$ to denote the frequency of word v_i in x_j. Without loss of generality, we use a unified vocabulary W for all domains, and assume $|W| = m$.

For each sentiment data x_j, there is a corresponding label y_j, where $y_j = +1$ if the overall sentiment expressed in x_j is positive, and $y_j = -1$ if the overall sentiment expressed in x_j is negative. A pair of sentiment text and its corresponding sentiment polarity $\{x_j, y_j\}$ is called the *labeled sentiment data*. If x_j has no polarity assigned, it is *unlabeled sentiment data*. Note that besides positive and negative sentiment, there are also neutral and mixed sentiment data in practical applications. *Mixed polarity* means user sentiment is positive in some aspects but negative in other ones. *Neutral polarity* means that there is no sentiment expressed by users. In this chapter, we only focus on positive and negative sentiment data.

For simplicity, we assume that a sentiment classifier f is a linear function as

$$y^* = f(x) = \mathbf{sgn}(w^\top x),$$

where $x \in \mathbb{R}^{m \times 1}$, $\mathbf{sgn}(w^\top x) = +1$ if $w^\top x \geq 0$, otherwise, $\mathbf{sgn}(w^\top x) = -1$, and w is the weight vector of the classifier, which can be learned from a set of training data (i.e., pairs of sentiment data and their corresponding polarity labels).

Consider the example shown in Table 1.1 as an motivating example. We use the standard bag-of-words representation to represent sentiment data of the *Electronics* and *Video Games* domains. From Table 1.2, we observe that the difference between domains is caused by the frequency of the domain-specific words. On one hand, the domain-specific words in the *Electronics* domain, such as *compact*, *sharp* and *blurry*, cannot be observed in the *Video Games* domain. On the other hand, the domain-specific words in the *Video Games* domain, such as *hooked*, *realistic* and *boring*, cannot be observed in the *Electronics* domain. Suppose that the *Electronics* domain is the source domain and the *Video Games* domain is the target domain. Apparently, based on the three training sentences in the *Electronics* domain, the weights of the features *compact* and *sharp* are positive, the weight of the feature *blurry* are negative, and the weights of the features *hooked*, *realistic* and *boring* can be arbitrary or zeros if an ℓ_1-norm regularization term is performed on w for model training. However, an ideal weight vector for the *Video Games* domain are supposed to have positive weights on the features *hooked*, *realistic* and a negative weight on the feature *boring*. Therefore, a classifier learned from the *Electronics* domain may predict poorly or randomly on the *Video Games* domain data.

Generally speaking, in sentiment classification, features can be classified into three types: 1) source domain (i.e., the *Electronics* domain) specific features, such as *compact*, *sharp*, and *blurry*, 2) target domain (i.e., the Video Game domain) specific features, such as *hooked*,

TABLE 1.2: Bag-of-words representations of *electronics* and *video games* reviews. Only domain-specific features are considered.

		compact	sharp	blurry	hooked	realistic	boring
electronics	**+1**	1	1	0	0	0	0
	+1	0	1	0	0	0	0
	-1	0	0	1	0	0	0
video games	**+1**	0	0	0	1	0	0
	+1	0	0	0	1	1	0
	-1	0	0	0	0	0	1

realistic, and *boring*, and 3) domain independent features or pivot features, such as *good*, *excited*, *nice*, and *never_buy*. Based on these observations, an intuitive idea of feature learning is to align the source and target domain specific features to generate cluster- or group- based features by using the domain independent features as a bridge such that the difference between the source and target domain data based on the new feature representation can be reduced. For instance, if the domain specific features shown in Table 1.2 can be aligned in the way presented in Table 1.3, where the feature alignments are used as new features to represent the data, then apparently, a linear model learned from the source domain (i.e., the *Electronics* domain) can be used to make precise predictions on the target domain data (i.e., the *Video Game* domain).

TABLE 1.3: Using feature alignments as new new features to represent cross-domain data.

		sharp_hooked	compact_realistic	blurry_boring
electronics	**+1**	1	1	0
	+1	1	0	0
	-1	0	0	1
video games	**+1**	1	0	0
	+1	1	1	0
	-1	0	0	1

Therefore, there are two research issues to be addressed. A first issue is how to identify domain independent or pivot features. A second issue is how to utilize the domain independent features and domain knowledge to align domain specific features from the source and target domains to generate new features. Here the domain knowledge is that if two sentiment words co-occur frequently in one sentence or document, then their sentiment polarities tend to be the same with a high probability.

For identifying domain independent or pivot features, some researchers have proposed several heuristic approaches [18, 92]. For instance, Blitzer *et al.* [18] proposed to select pivot features based on the term frequency in both the source and target domains and the mutual dependence between the features and labels in the source domain. The idea is that a pivot feature should be discriminative to the source domain data and appear frequently in both the source and target domains. Pan *et al.* [92] proposed to select domain independent features based on the mutual dependence between features and domains. Specifically, by considering all instances in the source domain with labels 1's and all instances in the target domain with labels 0's, the mutual information can be used to measure the dependence between the features and the constructed *domain labels*. The motivation is that if a feature has high mutual dependence to the domains, then it is domain specific. Otherwise, it is domain independent.

For aligning domain specific features from the source and target domains to generate cross-domain features, Biltzer *et al.* [19] proposed the structural correspondence learning (SCL) method. SCL is motivated by a multi-task learning algorithm, alternating structure optimization (ASO) [4], which aims to learn common features underlying multiple tasks. Specifically, SCL first identifies a set of *pivot* features of size m, and then treats each *pivot* feature as a new output vector to construct a pseudo task with non-pivot features as inputs. After that, SCL learns m linear classifiers to model the relationships between the non-pivot features and the constructed output vectors as follows,

$$y_j = \mathbf{sgn}(w_j^\top x_{np}),\ j = 1, \ldots, m,$$

where y_j is an output vector constructed from a corresponding pivot feature, and x_{np} is a vector of non-pivot features. Finally, SCL performs the singular value decomposition (SVD) on the weight matrix $W = [w_1\ w_2\ \ldots\ w_m] \in \mathbb{R}^{q\times m}$, where q is the number of non-pivot features, such that $W = UDV^\top$, where $U_{q\times r}$ and $V_{r\times m}$ are the matrices of the left and right singular vectors. The matrix $D_{r\times r}$ is a diagonal matrix consists of non-negative singular values, which are ranked in non-increasing order. The matrix $U^\top_{[1:h,:]}$, where h is the number of features to be learned, is then used as a transformation to align domain-specific features to generate new features.

Pan *et al.* [92] proposed the Spectral Feature Alignment (SFA) method for aligning domain specific features, which shares a similar high-level motivation with SCL. Instead of constructing pseudo tasks to use model parameters to capture the correlations between domain-specific and domain-independent features, SFA aims to model the feature correlations using a bipartite graph. Specifically, in the bipartite graph, a set of nodes correspond to the domain independent features, and the other set of nodes correspond to domain specific features in either the source or target domain. There exist an edge connecting a domain specific feature and a domain independent feature, if they co-occur in the same document or within a predefined window. A number associated on an edge is the total number of the co-occurrence of the corresponding domain specific and domain independent features in the source and target domains. The motivation of using bipartite graph to model the feature correlations is that if two domain specific features have connections to more common domain independent features in the graph, they tend to be aligned or clustered together with a higher probability. Meanwhile, if two domain independent features have connections to more common domain specific features in the graph, they tend to be aligned together with a higher probability. After the bipartite graph is constructed, the spectral clustering algorithm [88] is applied on the graph to cluster domain specific features. In this way, the clusters can be treated as new features to represent cross-domain data.

1.3.2.2 Learning Features by Minimizing Distance between Distributions

In the previous section, we have shown how to encode domain knowledge into feature learning for transfer learning. However, in many real-work scenarios, domain knowledge is not available as input. In this case, general approaches to feature learning for transfer learning are required. In this section, we first introduce a feature learning approach to transfer learning based on distribution minimization in a latent space.

Note that in many real-world applications, the observed data are controlled by only a few latent factors. If the two domains are related to each other, they may share some latent factors (or components). Some of these common latent factors may cause the data distributions between domains to be different, while others may not. Meanwhile, some of these factors may capture the intrinsic structure or discriminative information underlying the original data, while others may not. If one can recover those common latent factors that do not cause much difference between data distributions and do preserve the properties of the original data, then one can treat the subspace spanned by these latent factors as a bridge to make knowledge transfer possible. Based on this motivation, Pan *et al.* [90] proposed a dimensionality reduction algorithm for transfer learning, whose high-level idea can be formulated as follows,

$$\begin{aligned} \min_{\varphi} \quad & \mathrm{Dist}(\varphi(\mathbf{X}_S), \varphi(\mathbf{X}_T)) + \lambda\Omega(\varphi) \\ \text{s.t.} \quad & \text{constraints on } \varphi(\mathbf{X}_S) \text{ and } \varphi(\mathbf{X}_T), \end{aligned} \tag{1.7}$$

where φ is the mapping to be learned, which maps the original data to a low-dimensional space. The first term in the objective of (1.7) aims to minimize the distance in distributions between the source and target domain data, $\Omega(\varphi)$ is a regularization term on the mapping φ, and the constraints are to ensure original data properties to be preserved.

Note that, in general, the optimization problem (1.7) is computationally intractable. To make it computationally solvable, Pan *et al.* [90] proposed to transform the optimization problem (1.7) to a kernel matrix learning problem, resulting in solving a semidefinite program (SDP). The proposed method is known as Maximum Mean Discrepancy Embedding (MMDE), which is based on the non-parametric measure MMD as introduced in Section 1.3.1.1. MMDE has proven to be effective in learning features for transfer learning. However, it has two major limitations: 1) Since it requires to solve a SDP, its computational cost is very expensive; 2) Since it formulates the kernel matrix learning problem in a transductive learning setting, it cannot generalize to out-of-sample instances. To address the limitations of MMDE, Pan *et al.* [95, 96] further relaxed the feature learning problem of MMDE to a generalized eigen-decomposition problem, which is much efficient and easily generalized to out-of-sample instances. Similarly, motivated by the idea of MMDE, Si *et al.* [125] proposed to use the Bregman divergence as the distance measure between sample distributions to minimize the distance between the source and target domain data in a latent space.

1.3.2.3 Learning Features Inspired by Multi-task Learning

Besides learning features by minimizing distance in distributions, another important branch of approaches to learning features for transfer learning is motivated by multi-task learning [24]. In multi-task learning, given multiple tasks with a few labeled training data for each task, the goal is to jointly learn individual classifiers for different tasks by exploring latent common features shared by the tasks. Without loss of generality, for each task, we assume that the corresponding classifier is linear, and can be written as

$$f(x) = \langle \theta, (U^\top x) \rangle = \theta^\top (U^\top x),$$

where $\theta \in \mathbb{R}^{k\times 1}$ is the individual model parameter to be learned, and $U \in \mathbb{R}^{m\times k}$ is the transformation shared by all task data, which maps original data to a k-dimensional feature space, and needs to be learned as well. Note that the setting of multi-task learning is different from that of transfer learning, where a lot of labeled training data are assumed to be available in a source domain, and the focus is to learn a more precise model for the target domain. However, the idea of common feature learning under different tasks can still be borrowed for learning features for transfer learning by assuming that a few labeled training data in the target domain are available. The high-level objective of feature learning based on multi-task learning can be formulated as follows,

$$\begin{aligned} \min_{U,\theta_S,\theta_T} \quad & \sum_{t\in\{S,T\}} \sum_{i=1}^{n_t} l(U^\top x_{t_i}, y_{t_i}, \theta_t) + \lambda\Omega(\Theta, U) \\ \text{s.t.} \quad & \text{constraints on } U, \end{aligned} \tag{1.8}$$

where $\Theta = [\theta_S \ \theta_T] \in \mathbb{R}^{k \times 2}$ and $\Omega(\Theta, U)$ is a regularization term on Θ and U. Based on different forms of $\Omega(\Theta, U)$ and different constraints on U, approaches to learning features based on multi-task learning can be generally classified into two categories. In a first category of approaches, U is assumed to be full rank, which means that $m = k$, and Θ is sparse. A motivation behind this is that the full-rank U is only to transform the data from original space to another space of the same dimensionality, where a few *good* features underlying different tasks can be found potentially, and the sparsity assumption on Θ is to select such *good* features and ignore those that are not helpful for the source and target tasks. One of the representative approaches in this category was proposed by Argyriou *et al.* [6], where the $\| \cdot \|_{2,1}$ norm is proposed to regularize the matrix form of the model parameters Θ,[1] and U is assumed to be orthogonal, which means that $U^\top U = UU^\top = I$. As shown in [6], the optimization problem can be transformed to a convex optimization formulation and solved efficiently. In a follow-up work, Argyriou *et al.* [8] proposed a new spectral function on Θ for multi-task feature learning.

In a second category of approaches, U is assumed to be row rank, which means that $k < m$, or $k \ll m$ in practice, and there are no sparsity assumptions on Θ. In this way, U transforms the original data to *good* common feature representations directly. Representative approaches in this category include the Alternating Structure Optimization (ASO) method, which has been mentioned in Section 1.3.2.1. As described, in ASO, the SVD is performed on the matrix of the source and target model-parameters to recover a low-dimensional predictive space as a common feature space. The ASO method has been applied successfully to several applications [18, 5]. However, the proposed optimization problem is non-convex and thus a global optimum is not guaranteed to be achieved. Chen *et al.* [30] presented an improved formulation, called iASO, by proposing a novel regularization term on U and Θ. Furthermore, in order to convert the new formulation into a convex formulation, in [30], Chen *et al.* proposed a convex alternating structure optimization (cASO) algorithm to solve the optimization problem.

1.3.2.4 Learning Features Inspired by Self-taught Learning

Besides borrowing ideas from multi-task learning, a third branch of feature learning approaches to transfer learning is inspired by self-taught learning [105]. In self-taught learning, a huge number of unlabeled data are assumed to be available, whose labels can be different from those of the task of interest. The goal is to learn a set of *higher-level* features such that based on these higher-level features, a classifier trained on a few labeled training data can perform well on the task of interest. In this branch of approaches, a common assumption is that large-scale unlabeled or labeled source domain data, which can come from a single source or multiple sources, are available as inputs, and a few labeled data in the target domain are available as well. Most methods consist of three steps: 1) To learn higher-level features from the large-scale source domain data with or without their label information; 2) To represent the target domain data based on the higher-level features; 3) To train a classifier on the new representations of the target domain data with corresponding labels. A key research issue in these approaches is how to learn higher-level features. Raina *et al.* [105] proposed to apply sparse coding [70], which is an unsupervised feature construction method, to learn the higher-level features for transfer learning. Glorot *et al.* [51] proposed to apply deep learning to learn the higher-level features for transfer learning. Note that the goal of deep learning is to generate hierarchical features from lower-level input features, where the features generated in higher layers are assumed to be more higher level.

[1] The $\| \cdot \|_{2,1}$-norm of Θ is defined as $\|\Theta\|_{2,1} = \sum_{i=1}^{m} \|\Theta^i\|_2^1$, where Θ^i is the i^{th} row of Θ.

1.3.2.5 Other Feature Learning Approaches

In addition to the above three branches of feature learning methods for transfer learning, Daumé III [39] proposed a simple feature augmentation method for transfer learning in the field of Natural Language Processing (NLP). The proposed method aims to augment each of the feature vectors of different domains to a high dimensional feature vector as follows,

$$\begin{aligned} \widetilde{x}_S &= [x_S \; x_S \; \mathbf{0}], \\ \widetilde{x}_T &= [x_T \; \mathbf{0} \; x_T], \end{aligned}$$

where x_S and x_T are original features vectors of the source and target domains respectively, and $\mathbf{0}$ is a vector of zeros, whose length is equivalent to that of the original feature vector. The idea is to reduce the difference between domains while ensure the similarity between data within domains is larger than that across different domains. In a follow-up work, Daumé III [40] extend the feature augmentation method in a semi-supervised learning manner. Dai *et al.* [36] proposed a co-clustering based algorithm to discover common feature clusters, such that label information can be propagated across different domains by using the common clusters as a bridge. Xue *et al.* [147] proposed a cross-domain text classification algorithm that extends the traditional probabilistic latent semantic analysis (PLSA) [58] algorithm to extract common topics underlying the source and target domain text data for transfer learning.

1.3.3 Model-parameter-based Approach

The first two categories of approaches to transfer learning are in the data level, where the instance-based approach tries to reuse the source domain data after re-sampling or re-weighting, while the feature-representation-based approach aims to find a good feature representation for both the source and target domains such that based on the new feature representation source domain data can be reused. Different from these two categories of approaches, a third category of approaches to transfer learning can be referred to as the model-parameter-based approach, which assumes that the source and target tasks share some parameters or prior distributions of the hyper-parameters of the models. A motivation of the model-parameter-based approach is that a well-trained source model has captured a lot of structure, which can be transferred to learn a more precise target model. In this way, the transferred knowledge is encoded into the model parameters. In the rest of this section, we first introduce a simple method to show how to transfer knowledge across tasks or domains through model parameters, and then describe a general framework of the model-parameter-based approach.

Without loss of generality, we assume that the classifier to be learned is linear and can be written as follows,

$$f(x) = \langle \theta, x \rangle = \theta^\top x = \sum_{i=1}^{m} \theta_i x_i.$$

Given a lot of labeled training data in the source domain and a few labeled training data in the target domain, we further assume that the source model parameter θ_S is well-trained, and our goal to exploit the structure captured by θ_S to learn a more precise model parameter θ_T from the target domain training data.

Evgeniou and Pontil [48] proposed that the model parameter can be decomposed into two parts, one is referred to as a task specific parameter, and the other is referred to as a common parameter. In this way, the source and target model parameters θ_S and θ_T can be decomposed as

$$\begin{aligned}\theta_S &= \theta_0 + v_S,\\ \theta_T &= \theta_0 + v_T,\end{aligned}$$

where θ_0 is the common parameter shared by the source and target classifiers, v_S and v_T are the specific parameters of the source and target classifiers respectively. Evgenious and Pontil further proposed to learn the common and specific parameters by solving the following optimization problem,

$$\arg\min_{\theta_S,\theta_T} \sum_{t\in\{S,T\}} \sum_{i=1}^{n_t} l(x_{t_i}, y_{t_i}, \theta_t) + \lambda\Omega(\theta_0, v_S, v_T), \tag{1.9}$$

where $\Omega(\theta_0, v_S, v_T)$ is the regularization term on θ_0, v_S and v_T, and $\lambda > 0$ is the corresponding trade-off parameter. The simple idea presented in (1.9) can be generalized to a framework of the model-parameter-based approach as follows,

$$\arg\min_{\Theta} \sum_{t\in\{S,T\}} \sum_{i=1}^{n_t} l(x_{t_i}, y_{t_i}, \theta_t) + \lambda_1 \mathrm{tr}(\Theta^\top\Theta) + \lambda_2 f(\Theta) \tag{1.10}$$

where $\Theta = [\theta_S\ \theta_T]$, $\mathrm{tr}(\Theta^\top\Theta)$ is a regularization on θ_S and θ_T to avoid overfiting, and $f(\Theta)$ is to model the correlations between θ_S and θ_T, which is used for knowledge transfer. Different forms of $f(\Theta)$ lead to various specific methods. It can be showed that in (1.9), $f(\Theta)$ can be defined by the following form,

$$f(\Theta) = \sum_{t\in\{S,T\}} \left\| \theta_t - \frac{1}{2}\sum_{s\in\{S,T\}} \theta_s \right\|_2^2. \tag{1.11}$$

Besides using (1.11), Zhang and Yeung [161] proposed to the following form to model the correlations between the source and target parameters,

$$f(\Theta) = \mathrm{tr}(\Theta^\top\Omega^{-1}\Theta), \tag{1.12}$$

where Ω is the covariance matrix to model the relationships between the source and target domains, which is unknown and needs to be learned with the constraints $\Omega \succeq 0$ and $\mathrm{tr}(\Omega) = 1$. Agarwal *et al.* [1] proposed to use a manifold of parameters to regularize the source and target parameters as follows,

$$f(\Theta) = \sum_{t\in\{S,T\}} \left\| \theta_t - \widetilde{\theta}_t^{\mathcal{M}} \right\|^2, \tag{1.13}$$

where $\widetilde{\theta}_S^{\mathcal{M}}$ and $\widetilde{\theta}_T^{\mathcal{M}}$ are the projections of the source parameter θ_S and target parameter θ_T on the manifold of parameters respectively.

Besides the framework introduced in (1.10), there are a number of methods that are based on non-parametric Bayesian modeling. For instance, Lawrence and Platt [69] proposed an efficient algorithm for transfer learning based on Gaussian Processes (GP) [108]. The proposed model tries to discover common parameters over different tasks, and an informative vector machine was introduced to solve large-scale problems. Bonilla *et al.* [21] also investigated multi-task learning in the context of GP. Bonilla *et al.* proposed to use a free-form covariance matrix over tasks to model inter-task dependencies, where a GP prior is used to induce the correlations between tasks. Schwaighofer *et al.* [118] proposed to use a hierarchical Bayesian framework (HB) together with GP for transfer learning.

1.3.4 Relational-information-based Approaches

A fourth category of approaches is referred to as the relational-information-based approach. Different from the other three categories, the relational-information-based approach assume that some relationships between objects (i.e., instances) are similar across domains or tasks, if these common relationships can be extracted, then they can be used for knowledge transfer. Note that in this category of approaches, data in the source and target domains are not required to be independent and identically distributed (i.i.d.).

Mihalkova *et al.* [84] proposed an algorithm known as TAMAR to transfer relational knowledge with Markov Logic Networks (MLNs) [110] across the source and target domains. MLNs is a statistical relational learning framework, which combines the compact expressiveness of first order logic with flexibility of probability. In MLNs, entities in a relational domain are represented by predicates and their relationships are represented in first-order logic. TAMAR is motivated by the fact that if two domains are related to each other, there may exist mappings to connect entities and their relationships from the source domain to the target domain. For example, a professor can be considered as playing a similar role in an academic domain as a manager in an industrial management domain. In addition, the relationship between a professor and his or her students is similar to that between a manager and his or her workers. Thus, there may exist a mapping from professor to manager and a mapping from the professor-student relationship to the manager-worker relationship. In this vein, TAMAR tries to use an MLN learned for the source domain to aid in the learning of an MLN for the target domain. In a follow-up work, Mihalkova *et al.* [85] extended TAMAR in a single-entity-centered manner, where only one entity in the target domain is required in training.

Instead of mapping first-order predicates across domains, Davis *et al.* [41] proposed a method based on second-order Markov logic to transfer relational knowledge. In second-order Markov Logic, predicates themselves can be variables. The motivation of the method is that though lower-level knowledge such as propositional logic or first-order logic is domain or task specific, higher-level knowledge such as second-order logic is general for different domains or tasks. Therefore, this method aims to generate a set of second-order logic formulas through second-order MLNs from the source domain, and use them as higher-level templates to instantiate first-order logic formulas in the target domain.

More recently, Li *et al.* [74] proposed a relation-information-based method for sentiment analysis across domains. In this method, syntactic relationships between topic and sentiment words are exploited to propagate label information across the source and target domains. The motivation behind this method is that though the sentiment and topic words used in the source and target domains may be different, the syntactic relationships between them may be similar or the same across domains. Based on a few sentiment and topic seeds in the target domain, together with the syntactic relationships extracted from the source domain by using NLP techniques, lexicons of topic and sentiment words can be expanded iteratively in the target domain.

Note that the relational-information-based approach introduced in this section aims to explore and exploit relationships between instances instead of the instances themselves for knowledge transfer. Therefore, the relational-information-based approach can be also applied to heterogeneous transfer learning problems that will be introduced in the following section, where the source and target feature or label spaces are different.

1.4 Heterogeneous Transfer Learning

In the previous section, we have introduced four categories of approaches to homogeneous transfer learning. Recently, some researchers have already started to consider transfer learning across heterogeneous feature spaces or non-identical label spaces. In this section, we start by giving a definition of heterogeneous transfer learning as follows,

Definition 3. *Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$,* heterogeneous transfer learning *aims to help improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{X}_S \bigcap \mathcal{X}_T = \emptyset$ or $\mathcal{Y}_S \neq \mathcal{Y}_T$.*

Based on the definition, heterogeneous transfer learning can be further categorized into two contexts: 1) approaches to transferring knowledge across heterogeneous feature spaces, and 2) approaches to transferring knowledge across different label spaces.

1.4.1 Heterogeneous Feature Spaces

How to transfer knowledge successfully across different feature spaces is an interesting issue. It is related to multi-view learning [20], which assumes that the features for each instance can be divided into several views, each with its own distinct feature space. Though multi-view learning techniques can be applied to model multi-modality data, it requires each instance in one view must have its correspondence in other views. In contrast, transfer learning across different feature spaces aims to solve the problem where the source and target domain data belong to two different feature spaces such as image vs. text, without correspondences across feature spaces. Recently, some heterogeneous transfer learning methods have been developed and applied to various applications, such as cross-language text classification [101, 77, 135], image classification [168, 33, 64], and object recognition [67, 115].

Transfer learning methods across heterogeneous feature spaces can be further classified into two categories. A first context of approaches is to learn a pair of feature mappings to transform the source and target domain heterogeneous data to a common latent space. Shi *et al.* [124] proposed a Heterogenous Spectral Mapping (HeMap) method to learn the pair of feature mappings based on spectral embedding, where label information is discarded in learning. Wang and Mahadevan [135] proposed a manifold alignment method to align heterogenous features in a latent space based on a manifold regularization term, which is denoted by DAMA in the sequel. In DAMA, label information is exploited to construct similarity matrix for manifold alignment. However, DADA only works on the data that have strong manifold structures, which limits its transferability on those data where the manifold assumption does not hold. More recently, Duan *et al.* [44] proposed a Heterogenous Feature Augmentation (HFA) method to augment homogeneous common features learned by a SVM-style approach with heterogeneous features of the source and target domains for transfer learning. The proposed formulation results in a semidefinite program (SDP), whose computational cost is very expensive.

Another context is to learn a feature mapping to transform heterogenous data from one domain to another domain directly. In [34, 101], the feature mappings are obtained based on some *translators* to construct corresponding features across domains. However, in general, such translators for corresponding features is not available and difficult to be constructed in many real-world applications. Kulis [67] proposed an Asymmetric Regularized Cross-domain transformation (ARC-t) method to learn a asymmetric transformation across domains based on metric learning. Similar to DAMA, ARC-t also utilizes the label information to construct similarity and dissimilarity constraints between instances from the source and target domains respectively. The formulated metric learning problem can be solved by an alternating optimization algorithm.

1.4.2 Different Label Spaces

In some real-world scenarios, the label spaces or categories of the source and target domain may not be the same. In this case, it is crucial to develop transfer learning methods to propagate knowledge across labels or categories. A common idea behind most existing approaches in this setting is to explore and exploit the relationships between the source and target categories such that label information can be propagated across domains. Shi *et al.* [123] proposed a risk-sensitive spectral partition (RSP) method to align the source and target categories based on spectral partitioning. Dai *et al.* [35] proposed a EigenTransfer framework to use a three-layer bipartite graph to model the relationships between instances, features and categories. Through the three-layer bipartite graph, label information can be propagated across different categories. Quadrianto *et al.* [103] proposed to maximize mutual information between labels across domains to identify their correspondences. Qi *et al.* [102] proposed an optimization algorithm to learn a parametric matrix to model the correlations between labels across domains based on the similarities between cross-domain input data. Xiang *et al.* [144] proposed a novel framework named source-selection-free transfer learning (SSFTL) to achieve knowledge transfer from a Web-scale auxiliary resource, e.g., Wikipedia, for universal text classification. The idea of SSFTL is to first pre-train a huge number of source classifiers from the auxiliary resource offline, then when a target task is given, whose labels may not be observed in the auxiliary resource, SSFTL makes use of social tagging data, e.g., Flick, to bridge the auxiliary labels and the target labels, and finally select relevant source classifiers to solve the target task automatically.

1.5 Transfer Bounds and Negative Transfer

For theoretical study of transfer learning, an important issue is to recognize the limit of the power of transfer learning. So far, most theoretical studies are focused on homogeneous transfer learning. There are some research works analyzing the generalization bound in a special setting of homogeneous transfer learning where only the marginal distributions, $P_S(x)$ and $P_T(x)$, of the source and target domain data are assumed to be different [13, 17, 12, 14]. Though the generalization bounds proved in different literatures are different slightly, there is a common conclusion that the generalization bound of a learning model in this setting consists of two terms, one is the error bound of the learning model on the source domain labeled data, the other is the bound on the distance between the source and target domains, more specifically the distance between marginal probability distributions between domains.

In a more general setting homogeneous transfer learning where the predictive distributions, $P_S(y|x)$ and $P_T(y|x)$, of the source and target domain data can be different, theoretical studies are more focused on the issue of transferability. That is to ask how to avoid negative transfer and then ensure a “safe transfer” of knowledge. Negative transfer happens when the source domain/task data contribute to the reduced performance of learning in the target domain/task. Though how to avoid negative transfer is a very important issue, few research works were proposed on this issue in the past. Rosenstein *et al.* [114] empirically showed that if two tasks are very dissimilar, then brute-force transfer may hurt the performance of the target task.

Recently, some research works have been explored to analyze relatedness among tasks using task clustering techniques, such as [15, 11], which may help provide guidance on how to avoid negative transfer automatically. Bakker and Heskes [11] adopted a Bayesian approach in which some of the model parameters are shared for all tasks and others are more loosely connected through a joint prior distribution that can be learned from the data. Thus, the data are clustered based on the task parameters, where tasks in the same cluster are supposed to be related to each other. Hassan Mahmud and Ray [80] analyzed the case of transfer learning using Kolmogorov complexity, where some theoretical bounds are proved. In particular, they used conditional Kolmogorov complexity to measure relatedness between tasks and transfer the "right" amount of information in a sequential transfer learning task under a Bayesian framework. Eaton *et al.* [46] proposed a novel graph-based method for knowledge transfer, where the relationships between source tasks are modeled by a graph using transferability as the metric. To transfer knowledge to a new task, one needs to map the target task to the graph and learn a target model on the graph by automatically determining the parameters to transfer to the new learning task.

More recently, Argyriou *et al.* [7] considered situations in which the learning tasks can be divided into groups. Tasks within each group are related by sharing a low-dimensional representation, which differs among different groups. As a result, tasks within a group can find it easier to transfer useful knowledge. Jacob *et al.* [60] presented a convex approach to cluster multi-task learning by designing a new spectral norm to penalize over a set of weights, each of which is associated to a task. Bonilla *et al.* [21] proposed a multi-task learning method based on Gaussian Process (GP), which provides a global approach to model and learn task relatedness in the form of a task covariance matrix. However, the optimization procedure introduced in [21] is non-convex, whose results may be sensitive to parameter initialization. Motivated by [21], Zhang and Yeung [161] proposed an improved regularization framework to model the negative and positive correlation between tasks, where the resultant optimization procedure is convex.

The above works [15, 11, 7, 60, 21, 161] on modeling task correlations are from the context of multi-task learning. However, in transfer learning, one may be particularly interested in transferring knowledge from one or more source tasks to a target task rather than learning these tasks simultaneously. The main concern of transfer learning is the learning performance in the target task only. Thus, we need to give an answer to the question that given a target task and a source task, whether transfer learning techniques should be applied or not. Cao *et al.* [23] proposed an Adaptive Transfer learning algorithm based on GP (AT-GP), which aims to adapt transfer learning schemes by automatically estimating the similarity between the source and target tasks. In AT-GP, a new semi-parametric kernel is designed to model correlations between tasks, and the learning procedure targets at improving performance of the target task only. Seah *et al.* [120] empirically studied the negative transfer problem by proposing a predictive distribution matching classifier based on SVMs to identify the regions of relevant source domain data where the predictive distributions maximally align with that of the target domain data, and thus avoid negative transfer.

1.6 Other Research Issues

Besides the negative transfer issue, in recent years, there are several other research issues of transfer learning that have attracted more and more attention from the machine learning community, which are summarized in the following sections.

1.6.1 Binary Classification vs. Multi-class Classification

Most existing transfer learning methods are proposed for binary classification. For multi-class classification problems, one has to first reduce the multi-class classification task into multiple binary classification tasks using the *one-vs-rest* or *one-vs-one* strategy, and then train multiple binary classifiers to solve them independently. Finally, predictions are made according to the outputs of all binary classifiers. In this way, the relationships between classes, which indeed can be used to further boost the performance in terms of classification accuracy, may not be fully explored and exploited. Recently, Pan *et al.* [94] proposed a Transfer Joint Embedding (TJE) method to map both the features and labels from the source and target domains to a common latent space such as the relationships between labels can be fully exploited for transfer learning in multi-class classification problems.

1.6.2 Knowledge Transfer from Multiple Source Domains

In Sections 1.3-1.4, the transfer learning methods described are focused on one-to-one transfer, which means that there are only one source domain and one target domain. However, in some real-world scenarios, we may have multiple sources at hand. Developing algorithms to make use of multiple sources for help learning models in the target domain is useful in practice. Yang *et al.* [150] and Duan *et al.* [42] proposed algorithms to learn a new SVM for the target domains by adapting SVMs learned from multiple source domains. Luo *et al.* [79] proposed to train a classifier for use in the target domain by maximizing the consensus of predictions from multiple sources. Mansour *et al.* [81] proposed a distribution weighted linear combination framework for learning from multiple sources. The main idea is to estimate the data distribution of each source to reweight the data of different source domains. Yao and Doretto [155] extended TrAdaBoost in a manner of multiple source domains. Theoretical studies on transfer learning from multiple source domains have also been presented in [81, 82, 12].

1.6.3 Transfer Learning meets Active Learning

As mentioned at the beginning of this Chapter, both active learning and transfer learning aim to learn a precise model with minimal human supervision for a target task. Several researchers have proposed to combine these two techniques together in order to learn a more precise model with even less supervision. Liao *et al.* [76] proposed a new active learning method to select the unlabeled data in a target domain to be labeled with the help of the source domain data. Shi *et al.* [122] applied an active learning algorithm to select important instances for transfer learning with TrAdaBoost [38] and standard SVM. In [26], Chan and NG proposed to adapt existing Word Sense Disambiguation (WSD) systems to a target domain by using domain adaptation techniques and employing an active learning strategy [71] to actively select examples from the target domain to be annotated. Harpale and Yang [56] proposed an active learning framework for the multi-task adaptive filtering [112] problem. They first applied a multi-task learning method to adaptive filtering, and then explore various active learning approaches to the adaptive filters to improve performance. Li *et al.* [75] proposed a novel multi-domain active learning framework to jointly actively query data instances from all domains to be labeled to build individual classifiers for each domain. Zhao *et al.* [163], proposed a framework to construct entity correspondences with limited budget by using active learning to facilitate knowledge transfer across different recommender systems.

1.7 Applications of Transfer Learning

Recently, transfer learning has been applied successfully to many classification problems in various application areas, such as Natural Language Processing (NLP), Information Retrieval (IR), recommendation systems, computer vision, image analysis, multimedia data mining, bioinformatics, activity recognition and wireless sensor networks.

1.7.1 NLP Applications

In the field of NLP, transfer learning, which is known as domain adaptation, has been widely studied for solving various tasks, such as name entity recognition [53, 9, 39, 111, 140, 141, 94], part-of-speech tagging [4, 19, 62, 39], sentiment classification [18, 92, 51], sentiment lexicon construction [74], word sense disambiguation [26, 2], coreference resolution [149], and relation extraction [61].

1.7.2 Web-based Applications

Information Retrieval (IR) is another application area where transfer learning techniques have been widely studied and applied. Typical Web applications of transfer learning include text classification [106, 16, 36, 37, 54, 137, 147, 29, 145, 153, 143], advertising [33, 32], learn to rank [10, 134, 28, 50] and recommender systems [73, 72, 22, 160, 99].

1.7.3 Sensor-based Applications

Transfer learning has also been explored to solve WiFi-based localization and sensor-based activity recognition problems [151]. For example, transfer learning techniques have been proposed to transfer WiFi-based localization models across time periods [91, 166, 156], space [93, 136] and mobile devices [165], respectively. Rashidi and Cook [107] and Zheng *et al.* [164] proposed to apply transfer learning techniques for solving indoor sensor-based activity recognition problems respectively.

1.7.4 Applications to Computer Vision

In the past decade, transfer learning techniques have also attracted more and more attention in the fields of computer vision, image and multimedia analysis. Applications of transfer learning in these fields include image classification [142, 126, 113, 131, 157, 68], image retrieval [78, 55, 31], face verification from images [138], age estimation from facial images [162], image semantic segmentation [133], video retrieval [148, 55], video concept detection [150, 43], event recognition from videos [45], and object recognition [115, 67].

1.7.5 Applications to Bioinformatics

In Bioinformatics, motivated by that different biological entities, such as organisms, genes, etc, may be related to each other from a biological point of view, some research works have been proposed to apply transfer learning techniques to solve various computational biological problems, such as identifying molecular association of phenotypic responses [159], splice site recognition of eukaryotic genomes [139], mRNA splicing [119], protein subcellular location prediction [146] and genetic association analysis [154].

1.7.6 Other Applications

Besides the above applications, Zhuo *et al.* [169] studied how to transfer domain knowledge to learn relational action models across domains in automated planning. Chai *et al.* [25] studied how to apply a GP based transfer learning method to solve the inverse dynamics problem for a robotic manipulator [25]. Alamgir *et al.* [3] applied transfer learning techniques to solve brain-computer interfaces problems. In [109], Raykar *et al.* proposed to jointly learn multiple different but conceptually related classifiers for computer aided design (CAD) using transfer learning. Nam *et al.* [87] adapted a transfer learning approach to cross-project defect prediction in the field of software engineering.

1.8 Concluding Remarks

In this chapter, we have reviewed a number of approaches to homogeneous and heterogeneous transfer learning based on different categories. Specifically, based on “what to transfer”, approaches to homogeneous transfer learning can be classified into four categories, namely the instance-based approach, the feature-representation-based approach, the model-parameter-based approach, and the relational-information-based approach. Based on whether the feature spaces or label spaces between the source and target domains are different or not, heterogeneous transfer learning can be further classified into two contexts: namely transfer learning across heterogeneous feature spaces, and transfer learning across different label spaces. Furthermore, we have also discussed current theoretical studies on transfer learning and some research issues of transfer learning. Finally, we have summarized classification applications of transfer learning in diverse knowledge engineering areas.

Transfer learning is still at an early but promising stage. As described in Sections 1.5-1.6, there exist many research issues needed to be addressed. Specially, though there are some theoretical studies on homogeneous transfer learning, theoretical studies on heterogeneous transfer learning are still missing. Furthermore, most existing works on combining transfer learning and active learning consists two steps, one is for transfer learning and the other is for active learning. How to integrate them into a unified framework is still an open issue. Finally, in the future, we expect to see more applications of transfer learning in novel areas.

Bibliography

[1] Arvind Agarwal, Hal Daumé III, and Samuel Gerber. Learning multiple tasks using manifold regularization. In *Advances in Neural Information Processing Systems 23*, pages 46–54. 2010.

[2] Eneko Agirre and Oier Lopez de Lacalle. On robustness and domain adaptation using svd for word sense disambiguation. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 17–24. ACL, June 2008.

[3] Morteza Alamgir, Moritz Grosse-Wentrup, and Yasemin Altun. Multitask learning for brain-computer interfaces. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, volume 9, pages 17–24. JMLR W&CP, May 2010.

[4] Rie K. Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, 2005.

[5] Rie Kubota Ando and Tong Zhang. A high-performance semi-supervised learning method for text chunking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 1–9. ACL, June 2005.

[6] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. In *Advances in Neural Information Processing Systems 19*, pages 41–48. MIT Press, 2007.

[7] Andreas Argyriou, Andreas Maurer, and Massimiliano Pontil. An algorithm for transfer learning in a heterogeneous environment. In *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 71–85. Springer, September 2008.

[8] Andreas Argyriou, Charles A. Micchelli, Massimiliano Pontil, and Yiming Ying. A spectral regularization framework for multi-task structure learning. In *Annual in Neural Information Processing Systems 20*, pages 25–32. MIT Press, 2008.

[9] Andrew Arnold, Ramesh Nallapati, and William W. Cohen. A comparative study of methods for transductive transfer learning. In *Workshops Proceedings of the 7th IEEE International Conference on Data Mining*, pages 77–82. IEEE Computer Society, 2007.

[10] Jing Bai, Ke Zhou, Guirong Xue, Hongyuan Zha, Gordon Sun, Belle Tseng, Zhaohui Zheng, and Yi Chang. Multi-task learning for learning to rank in web search. In *Proceeding of the 18th ACM conference on Information and knowledge management*, pages 1549–1552. ACM, 2009.

[11] Bart Bakker and Tom Heskes. Task clustering and gating for bayesian multitask learning. *Journal of Machine Learning Reserch*, 4:83–99, 2003.

[12] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jenn Wortman. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010.

[13] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Annual in Neural Information Processing Systems 19*, pages 137–144. MIT Press, 2007.

[14] Shai Ben-David, Tyler Lu, Teresa Luu, and David Pal. Impossibility theorems for domain adaptation. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, volume 9, pages 129–136. JMLR W&CP, May 2010.

[15] Shai Ben-David and Reba Schuller. Exploiting task relatedness for multiple task learning. In *Proceedings of the 16th Annual Conference on Learning Theory*, pages 825–830. Morgan Kaufmann Publishers Inc., August 2003.

[16] Steffen Bickel and Tobias Scheffer. Dirichlet-enhanced spam filtering based on biased samples. In *Advances in Neural Information Processing Systems 19*, pages 161–168. MIT Press, 2006.

[17] John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jenn Wortman. Learning bounds for domain adaptation. In *Annual in Neural Information Processing Systems 20*, pages 129–136. MIT Press, 2008.

[18] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 432–439. ACL, June 2007.

[19] John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language*, pages 120–128. ACL, July 2006.

[20] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Learning Theory*, pages 92–100, July 1998.

[21] Edwin Bonilla, Kian Ming Chai, and Chris Williams. Multi-task gaussian process prediction. In *Annual in Neural Information Processing Systems 20*, pages 153–160. MIT Press, 2008.

[22] Bin Cao, Nathan N. Liu, and Qiang Yang. Transfer learning for collective link prediction in multiple heterogenous domains. In *Proceedings of the 27th International Conference on Machine learning*, pages 159–166. Omnipress, June 2010.

[23] Bin Cao, Sinno Jialin Pan, Yu Zhang, Dit-Yan Yeung, and Qiang Yang. Adaptive transfer learning. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. AAAI Press, July 2010.

[24] Rich Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[25] Kian Ming A. Chai, Christopher K. I. Williams, Stefan Klanke, and Sethu Vijayakumar. Multi-task gaussian process learning of robot inverse dynamics. In *Advances in Neural Information Processing Systems 21*, pages 265–272. 2009.

[26] Yee Seng Chan and Hwee Tou Ng. Domain adaptation with active learning for word sense disambiguation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 49–56. ACL, June 2007.

[27] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-Supervised Learning.* MIT Press, Cambridge, MA, USA, 2006.

[28] Olivier Chapelle, Pannagadatta Shivaswamy, Srinivas Vadrevu, Kilian Weinberger, Ya Zhang, and Belle Tseng. Multi-task learning for boosting with application to web search ranking. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1189–1198. ACM, July 2010.

[29] Bo Chen, Wai Lam, Ivor W. Tsang, and Tak-Lam Wong. Extracting discriminative concepts for domain adaptation in text mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 179–188. ACM, June 2009.

[30] Jianhui Chen, Lei Tang, Jun Liu, and Jieping Ye. A convex formulation for learning shared structures from multiple tasks. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 137–144. ACM, June 2009.

[31] Lin Chen, Dong Xu, Ivor W. Tsang, and Jiebo Luo. Tag-based web photo retrieval improved by batch mode re-tagging. In *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, June 2010.

[32] Tianqi Chen, Jun Yan, Gui-Rong Xue, and Zheng Chen. Transfer learning for behavioral targeting. In *Proceedings of the 19th International Conference on World Wide Web*, pages 1077–1078. ACM, April 2010.

[33] Yuqiang Chen, Ou Jin, Gui-Rong Xue, Jia Chen, and Qiang Yang. Visual contextual advertising: Bringing textual advertisements to images. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence.* AAAI Press, July 2010.

[34] Wenyuan Dai, Yuqiang Chen, Gui-Rong Xue, Qiang Yang, and Yong Yu. Translated learning: Transfer learning across different feature spaces. In *Annual in Neural Information Processing Systems 21*, pages 353–360. 2009.

[35] Wenyuan Dai, Ou Jin, Gui-Rong Xue, Qiang Yang, and Yong Yu. Eigentransfer: a unified framework for transfer learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 25–31. ACM, June 2009.

[36] Wenyuan Dai, Guirong Xue, Qiang Yang, and Yong Yu. Co-clustering based classification for out-of-domain documents. In *Proceedings of the 13th ACM International Conference on Knowledge Discovery and Data Mining*, pages 210–219. ACM, August 2007.

[37] Wenyuan Dai, Guirong Xue, Qiang Yang, and Yong Yu. Transferring naive bayes classifiers for text classification. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pages 540–545. AAAI Press, July 2007.

[38] Wenyuan Dai, Qiang Yang, Guirong Xue, and Yong Yu. Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning*, pages 193–200. ACM, June 2007.

[39] Hal Daumé III. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263. ACL, June 2007.

[40] Hal Daumé III, Abhishek Kumar, and Avishek Saha. Co-regularization based semi-supervised domain adaptation. In *Advances in Neural Information Processing Systems 23*, pages 478–486. 2010.

[41] Jesse Davis and Pedro Domingos. Deep transfer via second-order markov logic. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 217–224. ACM, June 2009.

[42] Lixin Duan, Ivor W. Tsang, Dong Xu, and Tat-Seng Chua. Domain adaptation from multiple sources via auxiliary classifiers. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 289–296. ACM, June 2009.

[43] Lixin Duan, Ivor W. Tsang, Dong Xu, and Stephen J. Maybank. Domain transfer SVM for video concept detection. In *Proceedings of the 22nd IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1375–1381. IEEE, June 2009.

[44] Lixin Duan, Dong Xu, and Ivor W. Tsang. Learning with augmented features for heterogeneous domain adaptation. In *Proceedings of the 29th International Conference on Machine Learning*. icml.cc/Omnipress, June 2012.

[45] Lixin Duan, Dong Xu, Ivor W. Tsang, and Jiebo Luo. Visual event recognition in videos by learning from web data. In *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2010.

[46] Eric Eaton, Marie desJardins, and Terran Lane. Modeling transfer relationships between learning tasks for improved inductive transfer. In *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 317–332. Springer, September 2008.

[47] Henry C. Ellis. *The Transfer of Learning*. The Macmillan Company, New York, 1965.

[48] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi-task learning. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 109–117. ACM, August 2004.

[49] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Proceedings of the 2nd European Conference on Computational Learning Theory*, pages 23–37. Springer-Verlag, 1995.

[50] Wei Gao, Peng Cai, Kam-Fai Wong, and Aoying Zhou. Learning to rank only using training data from related domain. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169. ACM, July 2010.

[51] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning*, pages 513–520. Omnipress, 2011.

[52] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel method for the two-sample problem. In *Advances in Neural Information Processing Systems 19*, pages 513–520. MIT Press, 2007.

[53] Hong Lei Guo, Li Zhang, and Zhong Su. Empirical study on the performance stability of named entity recognition model across domains. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 509–516. ACL, July 2006.

[54] Rakesh Gupta and Lev Ratinov. Text categorization with knowledge transfer from heterogeneous data sources. In *Proceedings of the 23rd National Conference on Artificial Intelligence*, pages 842–847. AAAI Press, July 2008.

[55] Sunil Kumar Gupta, Dinh Phung, Brett Adams, Truyen Tran, and Svetha Venkatesh. Nonnegative shared subspace learning and its application to social media retrieval. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1169–1178. ACM, July 2010.

[56] Abhay Harpale and Yiming Yang. Active learning for multi-task adaptive filtering. In *Proceedings of the 27th International Conference on Machine learning*, pages 431–438. ACM, June 2010.

[57] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction.* Springer, 2 edition, 2009.

[58] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57. ACM, 1999.

[59] Jiayuan Huang, Alex Smola, Arthur Gretton, Karsten M. Borgwardt, and Bernhard Schölkopf. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems 19*, pages 601–608. MIT Press, 2007.

[60] Laurent Jacob, Francis Bach, and Jean-Philippe Vert. Clustered multi-task learning: A convex formulation. In *Advances in Neural Information Processing Systems 21*, pages 745–752. 2009.

[61] Jing Jiang. Multi-task transfer learning for weakly-supervised relation extraction. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1012–1020. ACL, August 2009.

[62] Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 264–271. ACL, June 2007.

[63] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*, pages 200–209. Morgan Kaufmann Pulishers Inc., June 1999.

[64] Guo jun Qi, Charu C. Aggarwal, and Thomas S. Huang. Towards semantic knowledge propagation from text corpus to web images. In *Proceedings of the 20th International Conference on World Wide Web*, pages 297–306. ACM, March 2011.

[65] Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10:1391–1445, 2009.

[66] Sunghun Kim, Jr. E. James Whitehead, and Yi Zhang. Classifying software changes: Clean or buggy? *IEEE Transactions on Software Engineering*, 34:181–196, March 2008.

[67] Brian Kulis, Kate Saenko, and Trevor Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition*, pages 1785–1792. IEEE, June 2011.

[68] Christoph H. Lampert and Oliver Krömer. Weakly-paired maximum covariance analysis for multimodal dimensionality reduction and transfer learning. In *Proceedings of the 11th European Conference on Computer Vision*, pages 566–579, September 2010.

[69] Neil D. Lawrence and John C. Platt. Learning to learn with the informative vector machine. In *Proceedings of the 21st International Conference on Machine Learning.* ACM, July 2004.

[70] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *Annual in Neural Information Processing Systems 19*, pages 801–808. MIT Press, 2007.

[71] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12. ACM/Springer, July 1994.

[72] Bin Li, Qiang Yang, and Xiangyang Xue. Can movies and books collaborate?: cross-domain collaborative filtering for sparsity reduction. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, pages 2052–2057. Morgan Kaufmann Publishers Inc., July 2009.

[73] Bin Li, Qiang Yang, and Xiangyang Xue. Transfer learning for collaborative filtering via a rating-matrix generative model. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 617–624. ACM, June 2009.

[74] Fangtao Li, Sinno Jialin Pan, Ou Jin, Qiang Yang, and Xiaoyan Zhu. Cross-domain co-extraction of sentiment and topic lexicons. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 410–419. ACL, July 2012.

[75] Lianghao Li, Xiaoming Jin, Sinno Jialin Pan, and Jian-Tao Sun. Multi-domain active learning for text classification. In *Preceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1086–1094. ACM, August 2012.

[76] Xuejun Liao, Ya Xue, and Lawrence Carin. Logistic regression with an auxiliary data source. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 505–512. ACM, August 2005.

[77] Xiao Ling, Gui-Rong Xue, Wenyuan Dai, Yun Jiang, Qiang Yang, and Yong Yu. Can chinese web pages be classified with english data source? In *Proceedings of the 17th International Conference on World Wide Web*, pages 969–978. ACM, April 2008.

[78] Yiming Liu, Dong Xu, Ivor W. Tsang, and Jiebo Luo. Using large-scale web data to facilitate textual query based retrieval of consumer photos. In *Proceedings of the 17th ACM International Conference on Multimedia*, pages 55–64. ACM, October 2009.

[79] Ping Luo, Fuzhen Zhuang, Hui Xiong, Yuhong Xiong, and Qing He. Transfer learning from multiple source domains via consensus regularization. In *Proceeding of the 17th ACM Conference on Information and Knowledge Management*, pages 103–112. ACM, October 2008.

[80] M. M. Hassan Mahmud and Sylvian R. Ray. Transfer learning using kolmogorov complexity: Basic theory and empirical evaluations. In *Annual in Neural Information Processing Systems 20*, pages 985–992. MIT Press, 2008.

[81] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation with multiple sources. In *Advances in Neural Information Processing Systems 21*, pages 1041–1048. 2009.

[82] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Multiple source adaptation and the renyi divergence. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, pages 367–374. AUAI Press, June 2009.

[83] Tim Menzies, Jeremy Greenwald, and Art Frank. Data mining static code attributes to learn defect predictors. *IEEE Transactions on Software Engineering*, 33:2–13, January 2007.

[84] Lilyana Mihalkova, Tuyen Huynh, and Raymond J. Mooney. Mapping and revising markov logic networks for transfer learning. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pages 608–614. AAAI Press, July 2007.

[85] Lilyana Mihalkova and Raymond J. Mooney. Transfer learning by mapping with minimal target data. In *Proceedings of the AAAI-2008 Workshop on Transfer Learning for Complex Tasks*, July 2008.

[86] Tom M. Mitchell. *Machine Learning.* McGraw-Hill, New York, 1997.

[87] Jaechang Nam, Sinno Jialin Pan, and Sunghun Kim. Transfer defect learning. In *Proceedings of the 35th International Conference on Software Engineering*, pages 382–391. IEEE/ACM, May 2013.

[88] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[89] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2-3):103–134, 2000.

[90] Sinno Jialin Pan, James T. Kwok, and Qiang Yang. Transfer learning via dimensionality reduction. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 677–682. AAAI Press, July 2008.

[91] Sinno Jialin Pan, James T. Kwok, Qiang Yang, and Jeffrey J. Pan. Adaptive localization in a dynamic WiFi environment through multi-view learning. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pages 1108–1113. AAAI Press, July 2007.

[92] Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Chen Zheng. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th International Conference on World Wide Web*, pages 751–760. ACM, April 2010.

[93] Sinno Jialin Pan, Dou Shen, Qiang Yang, and James T. Kwok. Transferring localization models across space. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 1383–1388. AAAI Press, July 2008.

[94] Sinno Jialin Pan, Zhiqiang Toh, and Jian Su. Transfer joint embedding for cross-domain named entity recognition. *ACM Transactions on Information Systems*, 31(2):7:1–7:27, May 2013.

[95] Sinno Jialin Pan, Ivor W. Tsang, James T. Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 1187–1192, July 2009.

[96] Sinno Jialin Pan, Ivor W. Tsang, James T. Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[97] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[98] Sinno Jialin Pan, Vincent W. Zheng, Qiang Yang, and Derek H. Hu. Transfer learning for WiFi-based indoor localization. In *Proceedings of the Workshop on Transfer Learning for Complex Task of the 23rd AAAI Conference on Artificial Intelligence*, July 2008.

[99] Weike Pan, Evan W. Xiang, Nathan N. Liu, and Qiang Yang. Transfer learning in collaborative filtering for sparsity reduction. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. AAAI Press, July 2010.

[100] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, pages 79–86. ACL, July 2002.

[101] Peter Prettenhofer and Benno Stein. Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1118–1127. ACL, July 2010.

[102] Guo-Jun Qi, Charu C. Aggarwal, Yong Rui, Qi Tian, Shiyu Chang, and Thomas S. Huang. Towards cross-category knowledge propagation for learning visual concepts. In *Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition*, pages 897–904. IEEE, June 2011.

[103] Novi Quadrianto, Alex J. Smola, Tiberio S. Caetano, S.V.N. Vishwanathan, and James Petterson. Multitask learning without label correspondences. In *Advances in Neural Information Processing Systems 23*, pages 1957–1965. Curran Associates, Inc., December 2010.

[104] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. MIT Press, 2009.

[105] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. Self-taught learning: Transfer learning from unlabeled data. In *Proceedings of the 24th International Conference on Machine Learning*, pages 759–766. ACM, June 2007.

[106] Rajat Raina, Andrew Y. Ng, and Daphne Koller. Constructing informative priors using transfer learning. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 713–720. ACM, June 2006.

[107] Parisa Rashidi and Diane J. Cook. Activity recognition based on home to home transfer learning. In *Proceedings of the Workshop on Plan, Activity, and Intent Recognition of the 24th AAAI Conference on Artificial Intelligence*. AAAI Press, July 2010.

[108] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

[109] Vikas C. Raykar, Balaji Krishnapuram, Jinbo Bi, Murat Dundar, and R. Bharat Rao. Bayesian multiple instance learning: automatic feature selection and inductive transfer. In *Proceedings of the 25th International Conference on Machine learning*, pages 808–815. ACM, July 2008.

[110] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning Journal*, 62(1-2):107–136, 2006.

[111] Alexander E. Richman and Patrick Schone. Mining Wiki resources for multilingual named entity recognition. In *Proceedings of 46th Annual Meeting of the Association of Computational Linguistics*, pages 1–9. ACL, June 2008.

[112] Stephen Robertson, Stephen Robertson, and Ian Soboroff. The trec 2002 filtering track report. In *Text REtrieval Conference*, 2001.

[113] Marcus Rohrbach, Michael Stark, György Szarvas, Iryna Gurevych, and Bernt Schiele. What helps where - and why? semantic relatedness for knowledge transfer. In *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*, pages 910–917. IEEE, June 2010.

[114] Michael T. Rosenstein, Zvika Marx, and Leslie Pack Kaelbling. To transfer or not to transfer. In *NIPS-05 Workshop on Inductive Transfer: 10 Years Later*, December 2005.

[115] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Proceedings of the 11th European Conference on Computer Vision*, pages 213–226. Springer, September 2010.

[116] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–260. ACM, August 2002.

[117] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

[118] Anton Schwaighofer, Volker Tresp, and Kai Yu. Learning gaussian process kernels via hierarchical bayes. In *Annual in Neural Information Processing Systems 17*, pages 1209–1216. MIT Press, 2005.

[119] Gabriele Schweikert, Christian Widmer, Bernhard Schölkopf, and Gunnar Rätsch. An empirical analysis of domain adaptation algorithms for genomic sequence analysis. In *Advances in Neural Information Processing Systems 21*, pages 1433–1440. 2009.

[120] Chun-Wei Seah, Yew-Soon Ong Ivor W. Tsang, and Kee-Khoon Lee. Predictive distribution matching SVM for multi-domain learning. In *Proceedings of 2010 European Conference on Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 231–247. Springer, September 2010.

[121] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[122] Xiaoxiao Shi, Wei Fan, and Jiangtao Ren. Actively transfer domain knowledge. In *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 342–357. Springer, September 2008.

[123] Xiaoxiao Shi, Wei Fan, Qiang Yang, and Jiangtao Ren. Relaxed transfer of different classes via spectral partition. In *Preceedings of the 2009 European Conference on Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 366–381. Springer, September 2009.

[124] Xiaoxiao Shi, Qi Liu, Wei Fan, Philip S. Yu, and Ruixin Zhu. Transfer learning on heterogenous feature spaces via spectral transformation. In *Proceedings of the 10th IEEE International Conference on Data Mining*, pages 1049–1054. IEEE Computer Society, December 2010.

[125] Si Si, Dacheng Tao, and Bo Geng. Bregman divergence-based regularization for transfer subspace learning. *IEEE Transaction Knowledge Data Engineering*, 22(7):929–942, 2010.

[126] Michael Stark, Michael Goesele, and Bernt Schiele. A shape-based object class model for knowledge transfer. In *Proceedings of 12th IEEE International Conference on Computer Vision*, pages 373–380. IEEE, September 2009.

[127] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Von Buenau, and Motoaki Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems 20*, pages 1433–1440. MIT Press, 2008.

[128] Matthew E. Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(1):1633–1685, 2009.

[129] Edward Lee Thorndike and Robert Sessions Woodworth. The influence of improvement in one mental function upon the efficiency of the other functions. *Psychological Review*, 8:247–261, 1901.

[130] Sebastian Thrun and Lorien Pratt, editors. *Learning to learn.* Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[131] Tatiana Tommasi, Francesco Orabona, and Barbara Caputo. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*, pages 3081–3088. IEEE, June 2010.

[132] Vladimir N. Vapnik. *Statistical Learning Theory.* Wiley-Interscience, New York, September 1998.

[133] Alexander Vezhnevets and Joachim Buhmann. Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*, pages 3249–3256. IEEE, June 2010.

[134] Bo Wang, Jie Tang, Wei Fan, Songcan Chen, Zi Yang, and Yanzhu Liu. Heterogeneous cross domain ranking in latent space. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, pages 987–996. ACM, November 2009.

[135] Chang Wang and Sridhar Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1541–1546. IJCAI/AAAI, July 2011.

[136] Hua-Yan Wang, Vincent W. Zheng, Junhui Zhao, and Qiang Yang. Indoor localization in multi-floor environments with reduced effort. In *Proceedings of the 8th Annual IEEE International Conference on Pervasive Computing and Communications*, pages 244–252. IEEE Computer Society, March 2010.

[137] Pu Wang, Carlotta Domeniconi, and Jian Hu. Using Wikipedia for co-clustering based cross-domain text classification. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 1085–1090. IEEE Computer Society, 2008.

[138] Xiaogang Wang, Cha Zhang, and Zhengyou Zhang. Boosted multi-task learning for face verification with applications to web image and video search. In *Proceedings of the 22nd IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 142–149. IEEE, June 2009.

[139] Christian Widmer, Jose Leiva, Yasemin Altun, and Gunnar Rätsch. Leveraging sequence classification by taxonomy-based multitask learning. In *Proceedings of 14th Annual International Conference on Research in Computational Molecular Biology*, Lecture Notes in Computer Science, pages 522–534. Springer, April 2010.

[140] Tak-Lam Wong, Wai Lam, and Bo Chen. Mining employment market via text block detection and adaptive cross-domain information extraction. In *Proceedings of the 32nd International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 283–290. ACM, July 2009.

[141] Dan Wu, Wee Sun Lee, Nan Ye, and Hai Leong Chieu. Domain adaptive bootstrapping for named entity recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1523–1532. ACL, August 2009.

[142] Pengcheng Wu and Thomas G. Dietterich. Improving svm accuracy by training on auxiliary data sources. In *Proceedings of the 21st International Conference on Machine Learning*. ACM, July 2004.

[143] Evan W. Xiang, Bin Cao, Derek H. Hu, and Qiang Yang. Bridging domains using world wide knowledge for transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:770–783, 2010.

[144] Evan W. Xiang, Sinno Jialin Pan, Weike Pan, Jian Su, and Qiang Yang. Source-selection-free transfer learning. In *Proceedings of 22nd International Joint Conference on Artificial Intelligence*, pages 2355–2360. IJCAI/AAAI, July 2011.

[145] Sihong Xie, Wei Fan, Jing Peng, Olivier Verscheure, and Jiangtao Ren. Latent space domain transfer between high dimensional overlapping distributions. In *18th International World Wide Web Conference*, pages 91–100. ACM, April 2009.

[146] Qian Xu, Sinno Jialin Pan, Hannah Hong Xue, and Qiang Yang. Multitask learning for protein subcellular location prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(3):748–759, 2011.

[147] Gui-Rong Xue, Wenyuan Dai, Qiang Yang, and Yong Yu. Topic-bridged plsa for cross-domain text classification. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 627–634. ACM, July 2008.

[148] Rong Yan and Jian Zhang. Transfer learning using task-level features with application to information retrieval. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, pages 1315–1320. Morgan Kaufmann Publishers Inc., July 2009.

[149] Jian-Bo Yang, Qi Mao, Qiaoliang Xiang, Ivor Wai-Hung Tsang, Kian Ming Adam Chai, and Hai Leong Chieu. Domain adaptation for coreference resolution: An adaptive ensemble approach. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 744–753. ACL, July 2012.

[150] Jun Yang, Rong Yan, and Alexander G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *Proceedings of the 15th International Conference on Multimedia*, pages 188–197. ACM, September 2007.

[151] Qiang Yang. Activity recognition: Linking low-level sensors to high-level intelligence. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 20–25. Morgan Kaufmann Publishers Inc., July 2009.

[152] Qiang Yang, Sinno Jialin Pan, and Vincent W. Zheng. Estimating location using Wi-Fi. *IEEE Intelligent Systems*, 23(1):8–13, 2008.

[153] Tianbao Yang, Rong Jin, Anil K. Jain, Yang Zhou, and Wei Tong. Unsupervised transfer classification: application to text categorization. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1159–1168. ACM, July 2010.

[154] Xiaolin Yang, Seyoung Kim, and Eric Xing. Heterogeneous multitask learning with joint sparsity constraints. In *Advances in Neural Information Processing Systems 22*, pages 2151–2159. 2009.

[155] Yi Yao and Gianfranco Doretto. Boosting for transfer learning with multiple sources. In *Preceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*, pages 1855–1862. IEEE, June 2010.

[156] Jie Yin, Qiang Yang, and L.M. Ni. Learning adaptive temporal radio maps for signal-strength-based location estimation. *IEEE Transactions on Mobile Computing*, 7(7):869–883, July 2008.

[157] Xiao-Tong Yuan and Shuicheng Yan. Visual classification with multi-task joint sparse representation. In *Proceedings of the 23rd IEEE Conference on Computer Vision and Pattern Recognition*, pages 3493–3500. IEEE, June 2010.

[158] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 21st International Conference on Machine Learning*. ACM, July 2004.

[159] Kai Zhang, Joe W. Gray, and Bahram Parvin. Sparse multitask regression for identifying common mechanism of response to therapeutic targets. *Bioinformatics*, 26(12):i97–i105, 2010.

[160] Yu Zhang, Bin Cao, and Dit-Yan Yeung. Multi-domain collaborative filtering. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, pages 725–732. AUAI Press, July 2010.

[161] Yu Zhang and Dit-Yan Yeung. A convex formulation for learning task relationships in multi-task learning. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, pages 733–442. AUAI Press, July 2010.

[162] Yu Zhang and Dit-Yan Yeung. Multi-task warped gaussian process for personalized age estimation. In *Proceedings of the 23rd IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2622–2629. IEEE, June 2010.

[163] Lili Zhao, Sinno Jialin Pan, Evan W. Xiang, Erheng Zhong, Zhongqi Lu, and Qiang Yang. Active transfer learning for cross-system recommendation. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*. AAAI Press, July 2013.

[164] Vincent W. Zheng, Derek H. Hu, and Qiang Yang. Cross-domain activity recognition. In *Proceedings of the 11th International Conference on Ubiquitous Computing*, pages 61–70. ACM, September 2009.

[165] Vincent W. Zheng, Sinno Jialin Pan, Qiang Yang, and Jeffrey J. Pan. Transferring multi-device localization models using latent multi-task learning. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 1427–1432. AAAI Press, July 2008.

[166] Vincent W. Zheng, Qiang Yang, Evan W. Xiang, and Dou Shen. Transferring localization models over time. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 1421–1426. AAAI Press, July 2008.

[167] Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

[168] Yin Zhu, Yuqiang Chen, Zhongqi Lu, Sinno Jialin Pan, Gui-Rong Xue, Yong Yu, and Qiang Yang. Heterogeneous transfer learning for image classification. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*. AAAI Press, August 2011.

[169] Hankui Zhuo, Qiang Yang, Derek H. Hu, and Lei Li. Transferring knowledge from another domain for learning action models. In *Proceedings of 10th Pacific Rim International Conference on Artificial Intelligence*, pages 1110–1115. Springer-Verlag, December 2008.

[170] Thomas Zimmermann, Nachiappan Nagappan, Harald Gall, Emanuel Giger, and Brendan Murphy. Cross-project defect prediction: a large scale experiment on data vs. domain vs. process. In *Proceedings of the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, pages 91–100. ACM, August 2009.