Overcoming Negative Transfer: A Survey

Wen Zhang, Lingfei Deng, Lei Zhang, *Senior Member, IEEE*, Dongrui Wu, *Senior Member, IEEE*

Abstract—Transfer learning (TL) tries to utilize data or knowledge from one or more source domains to facilitate the learning in a target domain. It is particularly useful when the target domain has few or no labeled data, due to annotation expense, privacy concerns, etc. Unfortunately, the effectiveness of TL is not always guaranteed. Negative transfer (NT), i.e., the source domain data/knowledge cause reduced learning performance in the target domain, has been a long-standing and challenging problem in TL. Various approaches to overcome NT have been proposed in the literature. However, there has not been a systematic survey on overcoming NT. This paper fills the gap, by categorizing and reviewing near 100 approaches for combating NT, from four perspectives: source data quality, target data quality, domain divergence, and integrated algorithms. NT in related fields, e.g., multi-task learning, multilingual models, and lifelong learning, is also discussed.

Index Terms—Transfer learning, domain adaptation, negative transfer, domain divergence, transferability

1 Introduction

A Basic assumption in traditional machine learning is that the training and the test data are drawn from the same distribution. However, this assumption does not hold in many real-world applications. For example, two image datasets may be taken using cameras with different resolutions under different light conditions; different people may demonstrate strong individual differences in brain-computer interfaces [1]. Therefore, the resulting machine learning model may generalize poorly.

A conventional approach to mitigate this problem is to re-collect a large amount of labeled or partly labeled data, which have the same distribution as the test data, and then train a machine learning model from them. However, many factors may prevent easy access to such data, e.g., high annotation cost, privacy concerns, etc.

A better solution to the above problem is transfer learning (TL) [2], or domain adaptation (DA) [3], which tries to utilize data or knowledge from related domains (called source domains) to facilitate the learning in a new domain (called target domain). TL was first studied in educational psychology to enhance human's ability of learning new tasks and solving novel problems [4]. In machine learning, TL is mainly used to improve a model's generalization performance in the target domain, which usually has zero or a very small number of labeled data. Many different TL approaches have been proposed, e.g., traditional (statistical) TL [5]–[8], deep TL [9], [10], adversarial TL [11], [12], etc.

Unfortunately, the effectiveness of TL is not always guaranteed, unless its basic assumptions are satisfied: 1) the learning tasks in the two domains are related/similar; 2) the source domain and target domain data distributions are not too different; and, 3) a suitable model can be applied to both domains. Violation of these assumptions may result in **negative transfer (NT)**, i.e., the source domain data/knowledge cause reduced learning performance in the target domain, as illustrated in Fig. 1. NT is a long-standing and important problem in TL [2], [13], [14].

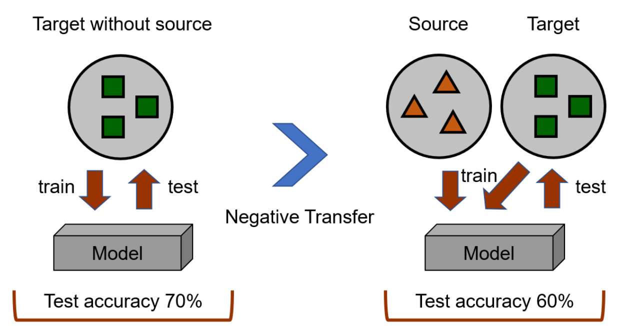

Fig. 1. Illustration of NT: learning in the target domain only works better than TL.

Three fundamental problems need to be considered for reliable TL [2]: 1) what to transfer; 2) how to transfer; and, 3) when to transfer. Most TL research [3], [15] focuses only on the first two, whereas all three should be taken into consideration in avoiding NT. To our knowledge, NT was first studied in 2005 [13], and received rapidly increasing attention recently [14], [16], [17]. Various ideas, e.g., finding similar parts of domains, evaluating the transferability of different tasks/models/features, etc., have been explored.

Though very important, there does not exist a comprehensive survey on NT. This paper aims to fill this gap. We systematically summarize nearly 100 representative approaches to cope with NT, according to four factors:

- *Source Data Quality.* Usually, the amount of source domain data in TL is much larger than the target domain, and hence source data quality is important to the TL learning performance. To overcome NT, it is necessary to discover and exploit shared underly-

- *Wen Zhang, Lingfei Deng and Dongrui Wu are with the Key Laboratory of the Ministry of Education for Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China. (e-mail: {wenz, lfdeng, drwu}@hust.edu.cn).*
- *Lei Zhang is with the School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China (e-mail: leizhang@cqu.edu.cn).*
- *Dongrui Wu is the corresponding author.*

ing structures and avoid unrelated patterns, such as unrelated domains, instances, features, or classes.

- *Target Data Quality.* The learned machine learning model is eventually applied to the target domain, and the target domain data are also critical in determining what to transfer. Hence, target data quality is also important to the TL performance.
- *Domain Divergence.* Arguably, divergence between the two domains is the root of NT. Many approaches have been proposed to measure and reduce it.
- *Integrated Algorithms.* Many integrated algorithms, e.g., secure transfer, robust transfer, etc., have also been proposed to overcome NT. Some of them have theoretical guarantees to avoid NT.

The remainder of this paper is organized as follows: Section 2 introduces some background knowledge on TL and NT. Sections 3-6 review approaches for overcoming NT, according to the four factors above, respectively. Section 7 discusses NT in several related machine learning fields. Finally, Section 8 draws conclusions.

2 Background Knowledge

This section introduces some background knowledge, including the notations and definitions, four factors causing NT, and the connections between transferability and generalization.

2.1 Notations

Assume the source domain $\mathcal{S}$ consists of n_s labeled samples, i.e., $\mathcal{S} = \{(\boldsymbol{x}_s^i, y_s^i)\}_{i=1}^{n_s}$, or has a pre-trained model with parameters $\boldsymbol{\theta}$. Assume the target domain has samples $\mathcal{T} = (\mathcal{T}_l, \mathcal{T}_u)$, where $\mathcal{T}_l = \{(\boldsymbol{x}_l^j, y_l^j)\}_{j=1}^{n_l}$ consists of n_l labeled samples, and $\mathcal{T}_u = \{\boldsymbol{x}_u^k\}_{k=1}^{n_u}$ consists of n_u unlabeled samples. Usually $n_s \gg n_l$. For unsupervised TL, $n_l = 0$, i.e., $\mathcal{T} = \mathcal{T}_u$.

We use $P_s(X,Y)$ and $P_t(X,Y)$, respectively, to denote the joint probability distribution in the source and target domains. $P_s(X)$ and $P_t(X)$ are their corresponding marginal probability distributions. $A(\mathcal{S},\mathcal{T})$ is a transfer model between the two domains, and $d(P_s(X,Y), P_t(X,Y))$ a domain divergence measure.

2.2 Transfer Learning (TL)

Under the above notation, TL aims to design a model $A(\mathcal{S},\mathcal{T})$, which utilizes data/information in both domains to output a hypothesis $h = A(\mathcal{S},\mathcal{T})$ for the target domain, with a small expected loss:

$$\epsilon_{\mathcal{T}}(h) = \mathbb{E}_{\boldsymbol{x},y\sim P_t(X,Y)}[\ell(h(\boldsymbol{x}), y)], \tag{1}$$

where ℓ is a target domain loss function.

2.3 Negative Transfer (NT)

Intuitively, NT happens when transferring data/knowledge from the source domain has a negative impact on the target domain learner [2], [14]. Although it has been widely studied in the literature, few publications gave NT a formal definition.

Wang et al. [14] defined NT as $\epsilon_{\mathcal{T}}(A(\mathcal{S},\mathcal{T})) > \epsilon_{\mathcal{T}}(A(\emptyset,\mathcal{T}))$, where $\emptyset$ means the source domain data/information are not used by the target domain learner at all. More precisely, the negative transfer gap (NTG) is:

$$NTG = \epsilon_{\mathcal{T}}(A(\mathcal{S},\mathcal{T})) - \epsilon_{\mathcal{T}}(A(\emptyset,\mathcal{T})). \tag{2}$$

NT occurs if NTG is positive.

Inspired by NTG, Wu and He [18] proposed a transfer signature (TS) to measure the transferability between two domains:

$$TS(\mathcal{T}\|\mathcal{S}) = \inf_{A\in\mathcal{H}} \left(\epsilon_{\mathcal{T}}(A(\mathcal{S},\mathcal{T})) - \epsilon_{\mathcal{T}}(A(\emptyset,\mathcal{T}))\right), \tag{3}$$

where $\mathcal{H}$ is the set of all learning algorithms. Clearly, NT occurs when TS is positive.

2.4 Factors of NT

The above definitions of NT can reveal some underlying factors of NT. For example, Wang et al. [14] pointed out that the algorithm, the domain divergence, and the size of labeled target domain data are three such factors.

Next, we give a more comprehensive analysis of NT factors, using the Ben-David theory [19], a widely used theoretical bound of TL.

Let $\mathcal{H}$ be the hypothesis space, and $\epsilon_{\mathcal{S}}$ and $\epsilon_{\mathcal{T}}$ be the generalization error of a classifier $h \in \mathcal{H}$ on the source domain feature space X_s and the target domain feature space X_t, respectively. Then, for any classifier $h \in \mathcal{H}$,

$$\epsilon_{\mathcal{T}}(h) \le \epsilon_{\mathcal{S}}(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(X_s, X_t) + \lambda, \tag{4}$$

where

$$\begin{aligned} d_{\mathcal{H}\Delta\mathcal{H}}(X_s, X_t) = 2 \sup_{h,h'\in\mathcal{H}} &|\mathbb{E}_{\boldsymbol{x}\sim X_s}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})] \\ &- \mathbb{E}_{\boldsymbol{x}\sim X_t}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})]| \end{aligned} \tag{5}$$

is the $\mathcal{H}\Delta\mathcal{H}$-divergence[1] between X_s and X_t, and $\lambda = \epsilon_{\mathcal{S}}(h^*) + \epsilon_{\mathcal{T}}(h^*)$ is the error of an ideal joint hypothesis h^* in both domains, in which $h^* = \arg\min_{h\in\mathcal{H}}\ \epsilon_{\mathcal{S}}(h) + \epsilon_{\mathcal{T}}(h)$.

To reduce $\epsilon_{\mathcal{T}}(h)$, we need to reduce all three terms on the right hand side of (4). To reduce the first term, $\epsilon_{\mathcal{S}}(h)$, the source domain $\mathcal{S}$ must have good data quality. To reduce the second term, $d_{\mathcal{H}\Delta\mathcal{H}}(X_s, X_t)$, the domain divergence between the source and target domains must be small. To reduce the third term, λ, the data quality in both domains should be good, and an appropriate hypothesis h^* should be chosen.

In summary, (4) suggests that the source data quality, the target data quality, the domain divergence, and the chosen machine learning algorithm are all important considerations in overcoming NT. Representativeness approaches to account for these four factors are summarized in Fig. 2 and introduced in details in the next four sections.

1. $\mathcal{H}\Delta\mathcal{H}$ is called the symmetric difference hypothesis space [19]. Every hypothesis $h \in \mathcal{H}\Delta\mathcal{H}$ is the set of disagreements between two hypotheses in $\mathcal{H}$.

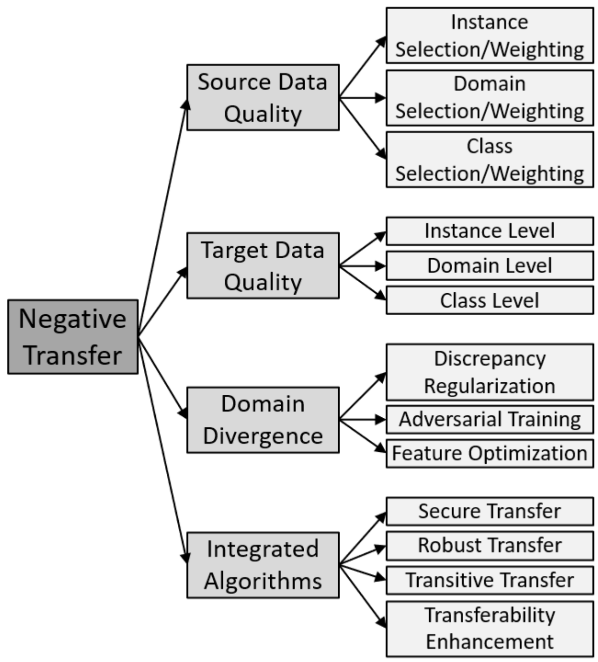

Fig. 2. Factors/remedies to NT.

2.5 Transferability versus Generalization

Transferability [20] indicates the ability of a TL approach to handle the discrepancy across domains. Model generalization measures the performance of a learned model when applied to a test set. Their differences are explained next.

Let $\mathcal{S}_{\text{tr}}$ be a source training set with n_{tr} samples drawn independently from the source distribution $P_s(X,Y)$. The true source generalization error (sometimes called risk) is $\epsilon_{\mathcal{S}}(h) = \mathbb{E}_{(\boldsymbol{x},y)\sim P_s(X,Y)}[h(\boldsymbol{x}) \neq y]$, and the empirical source risk over $\mathcal{S}_{\text{tr}}$ is $\hat{\epsilon}_{\mathcal{S}}(h) = \mathbb{E}_{(\boldsymbol{x},y)\in \mathcal{S}_{\text{tr}}}[h(\boldsymbol{x}) \neq y]$. The generalization error bound [21]

$$\epsilon_{\mathcal{S}}(h) \leq \hat{\epsilon}_{\mathcal{S}}(h) + \sqrt{\frac{\ln|\mathcal{H}| + \ln(1/\delta)}{2n_{\text{tr}}}}. \quad (6)$$

holds with probability at least $1-\delta$, for $\delta \in (0,1)$. Clearly, $\epsilon_{\mathcal{S}}(h)$ approaches $\hat{\epsilon}_{\mathcal{S}}(h)$ when the size of the source domain training set increases.

Equation (6) considers the source domain risk only, because it assumes the unknown test set is drawn independently from the same distribution as $\mathcal{S}_{\text{tr}}$. Clearly, this does not hold in TL, which considers different training and test distributions.

In summary, generalization usually means a model's performance in the same domain, whereas transferability measures a model's performance in a domain different from what it is trained.

3 Source Data Quality

This section introduces some representative approaches to evaluate or improve source data quality to overcome NT, as summarized in Table 1. The key is to discover and exploit similar information and avoid unrelated information in various forms, including domains, instances, or labels. While using the source domain(s) as a whole may result in NT, transferring part of the source information may still be beneficial.

TABLE 1
Approaches to evaluate or improve source data quality.

Category	Approaches	References
Instance Selection/ Weighting	Classification error based	[5], [22]–[24]
	Classification confidence based	[25], [26]
	Manifold based	[27]–[30]
	Adversarial network based	[14]
Domain Selection/ Weighting	Test performance based	[22], [31], [32]
	Similarity measure based	[18], [33]–[40]
	Bayesian Theorem based	[13], [41], [42]
Class Selection/ Weighting	Partial DA	[43]–[47]

3.1 Source Instance Selection/Weighting

Source instance selection/weighting can bring the source domain data distribution closer to the target domain distribution, and hence may help overcome NT. There are four categories of approaches: 1) classification error based, 2) classification confidence based, 3) manifold based, and 4) adversarial network based.

3.1.1 *Classification Error Based Approaches*

TrAdaBoost [5] is a classical instance level boosting-based TL approach to weight the source domain data, and two instance level boosting extensions introduced below. In each iteration, it combines the weighted samples in both domains to train a classifier, computes its error in the target domain, and then uses traditional AdaBoost to adjust the target domain sample weights (i.e., increase the weight if the corresponding sample is misclassified, and vice versa). An opposite strategy is used to adjust the source domain sample weights: the weight is increased if the corresponding sample is correctly classified, and vice versa. In this way, source domain samples more similar to the target domain samples are emphasized.

Yao and Doretto [22] proposed MultiSourceTrAdaBoost, which uses M iterations to train M weak classifiers from N source domains ($M \leq N$) and then combines them to overcome NT. In the mth iteration, the weighted instances from each source domain and the target domain are combined to train a classifier, whose error on the weighted target domain instances is recorded. Then, the classifier with the smallest error is chosen as the mth weak classifier. The instance weights in each source domain and the target domain are also updated accordingly.

Lin et al. [23] proposed a source domain sample selection based TL approach that integrates TrAdaBoost and TrBagg [48]. It first draws several bootstrap replicates from the target domain, and trains an ensemble of classifier from them. The source samples are tested using the ensemble, and the correctly classified ones are selected as potential useful source data. Then, it draws several bootstrap replicates from the selected source domain samples, each combined with the target data to train a classifier. The TrBagg criterion is next applied to these classifiers to select a smaller ensemble for the target task.

Yi et al. [24] proposed multi-component transfer metric learning. It first partitions the source domain into several components using a distance-based clustering method such as spectral clustering, k-means clustering, etc. For each component, the number of samples in each class is made equal by duplicating existing samples from the same class. It then performs transfer metric learning for each component to determine its sample weights and the Mahalanobis distance metric. The label of a test input is determined from its nearest source component only.

3.1.2 Classification Confidence Based Approaches

Many machine learning models also output a classification confidence for each input, where a larger confidence represents a larger chance of being classified correctly. Therefore, classification confidence may be an indicator of classification error, and hence can also be used to guide instance selection.

Seah et al. [25] proposed a predictive distribution matching (PDM) regularizer to remove irrelevant source domain data so that the target and source data are maximally aligned. In each iteration, it infers the pseudo-labels of the unlabeled target domain data and retain the highly confident ones, whose distribution is then used to remove source data that do not align with it. The process iterates until convergence. Finally, an SVM or logistic regression classifier is trained using the remaining source domain data and the pseudo-labeled target domain data.

Li et al. [26] believed that source domain instances that are classified with low confidence, e.g., support vectors of a support vector machine (SVM) classifier trained in the source domain, are less reliable, and hence may cause NT. They removed these support vectors from consideration in TL.

3.1.3 Manifold Based Approaches

The manifold assumption says that high-dimensional data lie on an embedded low-dimensional manifold, and data points on the same manifold have the same label [49]. It can also be used for instance selection/weighting.

Instance selection keeps only part of the instances which are close to each other on a local manifold. Peng et al. [27] proposed active transfer learning (ATL) to actively select appropriate source samples that are class balanced and highly similar to those in the target domain. ATL simultaneously minimizes maximum mean discrepancy (MMD) [6] and eliminates NT. It also uses the local geometric structure information of the source samples to find their informative and discriminative subsets. Peng et al. [28] also developed a source domain sample selection strategy to handle negative samples, by considering both the local geometry structure and the source label information.

Instance weighting assigns large weights to source regions whose predictions are consistent with the target labels. Ge et al. [29] developed a supervised local weight (SLW) scheme, based on a supervised manifold assumption: *"If predictions from a particular source domain are smooth and consistent with true labels on a manifold, the source domain will be assigned a high weight on this manifold."* An error bound is deducted to guarantee that the performance of SLW is always better than using the target training data alone. Moon and Carbonell [30] proposed attentional heterogeneous transfer, which pre-clusters the source data into several subsets and uses attention mechanism to learn a weight vector over the discrete subsets of data, corresponding to its relative importance or relevance in transfer.

3.1.4 Adversarial Network Based Approaches

Wang et al. [14] developed a discriminator gate to achieve both adversarial adaptation and class-level weighting of source samples. They used the output of a discriminator to estimate the distribution density ratio of two domains at each specific feature point by:

$$\frac{P_t(\boldsymbol{x}, y)}{P_s(\boldsymbol{x}, y)} = \frac{D(\boldsymbol{x}, y)}{1 - D(\boldsymbol{x}, y)}, \tag{7}$$

where $D(\boldsymbol{x}, y)$ represents the output of the discriminator when the input is the concatenation of the feature representation $\boldsymbol{x}$ and its predicted label y. The supervised learning loss is:

$$\begin{aligned}\mathcal{L}(C, F) = &\mathbb{E}_{\boldsymbol{x}_j, y_j \sim \mathcal{T}_l}[\ell(C(F(\boldsymbol{x}_j)), y_j)] \\ &+ \lambda \mathbb{E}_{\boldsymbol{x}_i, y_i \sim \mathcal{S}}[w(\boldsymbol{x}_i, y_i)\ell(C(F(\boldsymbol{x}_i)), y_i)],\end{aligned} \tag{8}$$

where C and F represent the classifier and feature extractor, respectively, and $w(\boldsymbol{x}_i, y_i) = D(\boldsymbol{x}, y)/(1 - D(\boldsymbol{x}, y))$ is the weight of each source sample.

3.2 Source Domain Selection/Weighting

Source instance selection/weighting, introduced in the previous subsection, mainly focuses on only one source domain. When there are multiple source domains, utilizing a subset of them which have high similarity to the target domain may achieve better TL performance than using all of them [23], [31], [50]. Therefore, source domain selection/weighting can also be used to overcome NT. There are three different categories of approaches: test performance based, similarity measure based, and Bayesian Theorem based.

3.2.1 Test Performance Based Approaches

If a classifier is built from each source domain, then test performance based approaches can perform source domain selection/weighting by evaluating their test errors on a few labeled target data and only choosing the classifiers whose combination can minimize the test error [22].

Boosting is a frequently used framework for the above purpose. Yao and Doretto [22] proposed TaskTrAdaBoost, which first uses AdaBoost to train M weak classifiers in each of the N source domains, then sequentially selects M best classifiers from these $N \cdot M$ weak classifiers. The selection criterion is the classification error on the weighted target domain samples, whose weights are also updated in each iteration, as in traditional AdaBoost, to ensure more difficult samples are emphasized by the next selected weak classifier. Eaton et al. [31] proposed TransferBoost, which trains a weak classifier in each iteration from the union of weighted samples from all domains. Each source instance weight is then increased or decreased according to whether its domain shows positive or negative transferability to the target task, and each target domain instance weight is increased or decreased according to whether the instance is misclassified or not.

Test performance based source domain weighting can also be used to overcome NT in multi-task learning. In this scenario, one group of related tasks may dominate the training process and NT may occur simultaneously on tasks outside the dominant group. In order to balance the performance on all tasks, Liu et al. [32] proposed an iterative method called loss-balanced task weighting, which uses the training loss of each task to indicate whether it is well trained or not, and decreases the relative weights of the well trained tasks.

3.2.2 Similarity Measure Based Approaches

Similarity measure based approaches aim to select or weight the source domains according to their similarities to the target domain. The similarity can be computed from Kullback-Leibler (KL) divergence, correlation, or by using a discriminator or MMD.

(1) KL-divergence based similarity measures. KL-divergence [51] is a non-symmetric measure of the divergence between two probability distributions. Gong et al. [33] proposed a rank of domain (ROD) approach to rank the similarities of the source domains to the target domain, by computing the symmetrized KL divergence weighted average of principal angles. It can be used to automatically select the optimal source domains to adapt and avoid less desirable ones. Azab et al. [34] computed the similarity weight α_s between the target domain feature set d_t and the source domain feature set d_s as:

$$\alpha_s = \frac{1/\left(\overline{KL}[d_t, d_s] + \epsilon\right)^4}{\sum_{i=1}^{m}\left(1/\left(\overline{KL}[d_t, d_i] + \epsilon\right)^4\right)}, \tag{9}$$

where $\overline{KL}$ represents the average per-class KL-divergence, m is the number of source domains, and $\epsilon = 0.0001$ is used to ensure the stability of calculation.

(2) Correlation based similarity measures. The correlation between two high-dimensional random variables from different distributions can also be used to evaluate the distribution discrepancy. Lin and Jung [35] calculated the inter-subject similarity in EEG-based emotion classification as the correlation coefficient of feature representations from two different subjects. Zhang and Wu [36] developed a domain transferability estimation (DTE) index to evaluate the transferability between a source domain $\mathcal{S}$ and the target domain $\mathcal{T}$:

$$DTE(\mathcal{S}, \mathcal{T}) = \frac{\|S_b^{\mathcal{S}}\|_1}{\|S_b^{\mathcal{S},\mathcal{T}}\|_1}, \tag{10}$$

where $S_b^{\mathcal{S}}$ is the between-class scatter matrix in the source domain, $S_b^{\mathcal{S},\mathcal{T}}$ is the between-domain scatter matrix, and $\|\cdot\|_1$ is the L_1 norm. DTE has low computational cost and is insensitive to the sample size.

In order to avoid NT caused by concept drift [52], Xie et al. [37] proposed selective transfer incremental learning (STIL) to remove less relevant historical models, which may be viewed as models trained from multiple source domains. STIL computes the following Q-statistic to represent the correlation between a historical model and the newly trained target model:

$$Q_{f_i,f_j} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}}, \tag{11}$$

where f_i and f_j are two classifiers. $N^{y_i y_j}$ is the number of examples for which the classification result is y_i by f_i ($y_i = 1$ if f_i classifies the example correctly; otherwise $y_i = 0$), and y_j by f_j. STIL then removes the less transferable historical models, whose Q-statistics are close to 0. In this way, it can avoid NT. This strategy was also used in [53].

In multi-task TL, Bao et al. [38] proposed an H-score to characterize the asymptotic error probability of using zero-mean transferred feature representation $f(X)$ to estimate the label Y in the hypothesis testing context:

$$H(f) = \mathrm{tr}(\mathrm{cov}(f(X))^{-1}\mathrm{cov}(\mathbb{E}_{P(X|Y)}[f(X)|Y])). \tag{12}$$

The task transferability can then be computed as $H_{\mathcal{T}}(f_{\mathcal{S}})/H_{\mathcal{T}}(f_{\mathcal{T}_{\text{opt}}})$, where subscripts $\mathcal{S}$ and $\mathcal{T}$ denote variables for the source and the target tasks, respectively, and $f_{\mathcal{T}_{\text{opt}}}$ is the minimum error probability feature of the target task.

(3) Discriminator based similarity measures. These approaches train a classifier to discriminate the two domains and then define a similarity measure from the classification error [19]. Ben-David et al. [39] proposed an unsupervised $\mathcal{A}$-distance to find the minimum-error classifier

$$d_{\mathcal{A}}(\boldsymbol{\mu}_S, \boldsymbol{\mu}_T) = 2\left(1 - 2\min_{h\in\mathcal{H}} \epsilon(h)\right), \tag{13}$$

where $\mathcal{H}$ is the hypothesis space, h a domain classifier, and $\epsilon(h)$ the domain classification error. The $\mathcal{A}$-distance should be small for good transferability.

Unfortunately, computing $d_{\mathcal{A}}(\boldsymbol{\mu}_S, \boldsymbol{\mu}_T)$ is NP-hard. To reduce the computational cost, they trained a linear classifier to determine which domain the data come from, and utilized its error to approximate the optimal classifier.

Based on the $\mathcal{A}$-distance, recently Wu and He [18] proposed a novel label-informed divergence between the source and the target domains when the target domain is time evolving. This divergence can measure the shift of data distributions as well as to identify potential NT.

(4) MMD based similarity measures. These approaches use MMD (maybe also in combination with other measures) to measure the proximity between the source and the target domains. Wang et al. [40] proposed a peer-weighted TL approach for multi-source transfer. It first re-weights the samples in each source domain so that the means of the source and target domains after mapping onto a Reproducing Kernel Hilbert Space (RKHS) is minimized. A classifier is then trained in each source domain and used in a weighted ensemble to classify a test input, where the weight is a combination of the source domain's MMD-based proximity to the target domain and its reliability (transferability) to other source domains. In the special case that the confidence of a source domain classifier is low, its own classification is not used; instead, it queries its peers on this specific test example, where each peer is weighted by its transferability to the current source domain.

In summary, similarity measure based source domain selection/weighting has attracted increasing attention, for it can help understand the relationship between domains, selects groups of highly transferable domains for joint training, or chooses good source models for a given target task.

3.2.3 Bayesian Theorem Based Approaches

Bayesian Theorem provides a basis for analyzing domain discrepancy via joint probability distribution.

One approach is to fit source and target data with a hyperprior distribution and estimate its parameters, which may reflect the domain difference, from posterior probabilities. For example, Rosenstein et al. [13] used hierarchical naive Bayes to construct two coupled Bayesian models on the source and target domains, whose parameters are encouraged to be similar to facilitate TL.

Cao et al. [41] proposed an adaptive TL algorithm based on Gaussian process, to automatically estimate the similarity between domains. It assumes source and target data obey Gaussian distribution with a semi-parametric transfer kernel $\boldsymbol{K}$,

$$\boldsymbol{K}_{nm} \sim k(\boldsymbol{x}_n, \boldsymbol{x}_m)(2e^{-\varsigma(\boldsymbol{x}_n, \boldsymbol{x}_m)\rho} - 1), \tag{14}$$

where k is a valid kernel function. $\varsigma(\boldsymbol{x}_n, \boldsymbol{x}_m) = 0$ if $\boldsymbol{x}_n$ and $\boldsymbol{x}_m$ are from the same domain, otherwise $\varsigma(\boldsymbol{x}_n, \boldsymbol{x}_m) = 1$. The parameter ρ represents the dissimilarity between source and target domains. By assuming ρ is from a Gamma distribution $\Gamma(b, \mu)$, the transfer kernel has its Bayesian form,

$$\widetilde{\boldsymbol{K}}_{nm} = \begin{cases} \lambda k(\boldsymbol{x}_n, \boldsymbol{x}_m), & \varsigma(\boldsymbol{x}_n, \boldsymbol{x}_m) = 1 \\ k(\boldsymbol{x}_n, \boldsymbol{x}_m), & \text{otherwise} \end{cases}, \tag{15}$$

where $\lambda = 2\left(\frac{1}{1+\mu}\right)^b - 1$, in which b and μ are hyperparameters of the Gamma distribution. λ, which can be estimated from labeled data in both domains, determines the similarity between the domains, and what can be transferred.

Supervised pre-training and model fine-tuning [54], [55] are frequently used in deep TL. To reduce the computational cost and avoid NT, a small amount of target domain data can be used to select the most related tasks, where Bayesian Theorem based statistical indices can be computed to estimate the task transferability. Tran et al. [42] developed negative conditional entropy (NCE), which measures the amount of information from a source task to the target task, to evaluate the task transferability. Nguyen et al. [17] proposed log expected empirical prediction (LEEP), which can be computed from a source model θ and n_l labeled target data, by running the target data through the model only once:

$$T(\theta, \mathcal{D}) = \frac{1}{n_l} \sum_{i=1}^{n_l} \log\left(\sum_{z \in Z} \hat{P}(y_i|z)\theta(x_i)_z\right), \tag{16}$$

where $\hat{P}(y_i|z)$ is the empirical conditional distribution of the real target label y_i given the dummy target label z predicted by model θ. $T(\theta, \mathcal{D})$ represents the transferability of the pre-trained model θ to the target domain D, and is an upper bound of the NCE measure.

3.3 Source Class Selection/Weighting

In partial DA, the target classes may only be a subset of the source classes. Extra source classes could cause NT. Two solutions are source class selection and weighting.

Source class selection can be achieved by aligning each target instance to one or multiple most relevant source classes, and then removing the irrelevant classes. Cao et al. [43] proposed selective adversarial network (SAN) for this purpose. SAN consists of multiple class-wise domain discriminators, each matching the source and target data for a specific class. It uses an adaptive classifier to estimate the probabilities of each target instance belonging to different source classes, which also characterize the probabilities of assigning that instance to the corresponding domain discriminators. By maximizing a probability-weighted domain discriminator loss, outlier source classes are identified and removed.

Zhang et al. [44] proposed an importance weighted adversarial nets-based method to identify the source samples that are potentially from the outlier classes and simultaneously to reduce the shift of shared classes between domains. The main intuition is that if a source sample is predicted by a domain discriminator with high confidence, then very likely it belongs to an outlier source class, because the region covering that sample has few or no target samples at all.

Once the outlier source classes are identified, the contribution of the corresponding source samples to the source classifier, which transfers to the target domain, should be reduced. Cao et al. [45] proposed example transfer network (ETN), which achieves this goal by adding the source domain sample weights [43] to the source classifier loss function. Zhang et al. [46] proposed a relation-gate (R-Gate) mechanism, which averages the label predictions on all target data from the source classifier as class weights, to identify outlier source classes and reduce their negative impact.

When there are multiple source domains, a single source domain may not cover all labels of the target domain. Ding et al. [47] proposed bi-directional low-rank transfer (BLRT) to cope with this problem. It uses a cross-source regularizer to couple the same labels from multiple incomplete source domains, so that missing data from other source domains can be compensated.

4 Target Data Quality

As the TL model will eventually be used in the target domain, target data quality also has important impact on NT. Interestingly, compared with the rich literature on source data quality, target data quality is much less frequently studied. These limited studies are summarized in Table 2.

4.1 Instance Level

When there are enough labeled samples in the target domain, transferring from a source domain may not be necessary, especially when the two are different. TL is more necessary when the target domain has very limited or no labeled data. In this case, a frequently used used strategy is to assign pseudo-labels to the unlabeled samples and then iteratively refine them [6]–[8]. However, NT can easily

TABLE 2
Approaches to evaluate or improve target data quality.

CATEGORY	APPROACHES	REFERENCES
Instance Level	Error trade-off, active learning, active class selection	[19], [56]–[58]
Domain Level	Multi-target DA, open compound DA	[59], [60]
Class Level	Zero/few shot learning, open set DA, universal DA	[61]–[64]

occur, especially when the initial pseudo-labels are very inaccurate.

Given some labeled instances in the target domain, a natural TL setting is to trade-off minimizing the source and the target empirical errors:

$$\hat{\epsilon}_\alpha(h) = \alpha\epsilon_{\mathcal{T}}(h) + (1-\alpha)\epsilon_{\mathcal{S}}(h), \tag{17}$$

The weight $\alpha \in [0,1]$ depends on the amount of labeled samples in the target domain. Ben-David et al. [19] showed that if there is no labeled target data, then the most appropriate choice is to use the source domain labeled data directly, i.e., $\alpha = 0$. On the other hand, if there are enough labeled target data, i.e., $n_T^l \geq VC(\mathcal{H})/d^2$, where d is the total divergence between the two domains, then no source data are required for efficient learning, i.e., $\alpha = 1$. For other cases, α takes values in $(0,1)$.

If the unlabeled target domain samples can be queried for their labels, then active learning [65] may be integrated with TL to improve the overall learning performance. Since TL and active learning are independent and complementary, Wu et al. [56], [57] employed TL to make use of information from the source domains, and active learning to select the most beneficial unlabeled target domain samples to label to maximally increase the target domain data quality. Active class selection [66], which selects the most beneficial class to query an example, instead of selecting an unlabeled sample to query for its label in active learning, can also be used in a similar manner [58].

4.2 Domain Level

Conventional TL usually assumes that there is only one target domain; however, this may not always hold. For example, when transferring an autonomous driving system from simulations to the real world, there are many different conditions (target domains), some of which could be more complex than, or even not included in, the source domains.

To handle one source domain with labeled instances and multiple target domains with unlabeled instances, Gholami et al. [59] proposed a multi-target DA-information-theoretic-approach, which finds a shared latent space common to all domains, and simultaneously accounts for the remaining domain-specific private factors. For a given input $\boldsymbol{x}$, it learns an encoder $E_s(\boldsymbol{x})$ to capture the feature representations ($\boldsymbol{z}_s$) shared by all domains, and another encoder $E_p(\boldsymbol{x})$ to capture domain–specific private features ($\boldsymbol{z}_p$). A shared decoder $F(\boldsymbol{z}_p, \boldsymbol{z}_s)$ learns to reconstruct $\boldsymbol{x}$ by using both $\boldsymbol{z}_s$ and $\boldsymbol{z}_p$. A domain classifier $D(\boldsymbol{z}_s, \boldsymbol{z}_p)$ and a class classifier $C(\boldsymbol{z}_s)$ learns to predict the domain label and class label of $\boldsymbol{x}$, respectively. $C(\boldsymbol{z}_s)$ is used as the final classifier.

Liu et al. [60] studied open compound domain adaptation (OCDA), i.e., the target is a compound of multiple homogeneous domains without domain labels. It first uses a class-confusion algorithm to learn a class encoder $E_{class}(\boldsymbol{x})$ from the source domain labeled samples, and a domain encoder $E_{domain}(\boldsymbol{x})$ by alternating between the following two problems:

$$\min_{E_{domain}} -\sum_i z^i_{random} \log D(E_{domain}(\boldsymbol{x}^i)), \tag{18}$$

$$\min_{D} -\sum_i y^i \log D(E_{domain}(\boldsymbol{x}^i)), \tag{19}$$

where D is a discriminator $E_{domain}(\boldsymbol{x})$ tries to confuse, and z^i_{random} a random label. To learn in the open target domain, OCDA uses a memory module to increase the model's agility towards novel domains:

$$\boldsymbol{v}_{transfer} = \boldsymbol{v}_{direct} + \boldsymbol{e}_{domain} \otimes \boldsymbol{v}_{enhance}, \tag{20}$$

where $\boldsymbol{v}_{direct}$ is the direct representation of an input, $\boldsymbol{v}_{enhance}$ is the enhanced representation augmented with knowledge in the memory about the source domain, $\boldsymbol{e}_{domain}$ is a domain indicator, and $\otimes$ denotes element-wise multiplication. The above feature representation balances between the input features and the enhanced features, and the weights (domain indicator) depend on how close the input features are to the source domains.

4.3 Class Level

Most existing TL approaches [7], [11] assume a shared label set between the source and the target domains. However, unseen open classes could exist in the target domain. Different schemes have been developed to cope with this problem: zero/few shot learning [61], [62], open set DA [63], and universal DA [64].

Zero shot learning (ZSL) tries to recognize the samples of unseen categories that have not appeared in the training data, i.e., there is no overlap between the seen categories in the training data and the unseen categories in the test data. An extension of ZSL is one/few shot learning, where one or very few labeled examples in each unseen target class are obtained during training [67].

Busto and Gall [63] explored open set DA, where both source and target domains contain samples that do not belong to the classes of interest, and the target domain also contains samples not related to any sample in the source domain, and vice versa. Their approach solves two problems alternatively: 1) labelling the target samples, i.e., associating a subset of the target samples to the known source domain classes; and, 2) computing a mapping from the source domain to the target domain by minimizing the distances of the assignments. The transformed source samples are then used in the next iteration to refine the assignments and update the mapping.

You et al. [64] considered another realistic scenario that the source and the target label spaces share some common classes, but each also has its own private classes. They proposed a universal DA network, which quantifies instance-level transferability to discover the common and private label sets, and promotes the adaptation in the common label set.

5 Domain Divergence

Arguably, the divergence between different domains is the root of NT. As a result, many approaches have been proposed to measure and reduce it, as summarized in Table 3.

TABLE 3
Approaches to measure or reduce domain divergence.

Category	Approaches	References
Discrepancy Measures and Regularization	HSIC, Bregman divergence, MMD, moment matching, optimal transport, Wasserstein distance	[6], [7], [10], [68]–[78]
Adversarial Training	Domain adversarial neural network	[11]
Feature Optimization	Feature selection/weighting, latent feature space learning, feature transferability enhancement	[79]–[84]

5.1 Discrepancy Measures and Regularization

Many domain discrepancy measures have been proposed for TL, which can be used as regularizers to reduce the domain divergence.

5.1.1 Traditional (Statistical) TL

Hilbert-Schmidt independence criterion (HSIC) [68] and Bregman divergence [69] were early measures of domain discrepancy in traditional TL. HSIC is a nonparametric measure of the dependency between two sets, but it is not scale invariant and not designed to compare data in different feature spaces [85]. The Bregman divergence requires density estimation, which limits its applicability.

MMD [6]–[8], [86] may be the most popular discrepancy measure in traditional TL, due to its simplicity and effectiveness. It is a nonparametric measure, and can be computed directly from the feature means.

Let $\mathcal{F} = \{f : \|f\|_{\mathcal{H}} \leq 1\}$ be a class of real-valued bounded measurable functions, where $\mathcal{H}$ is an RKHS space. Then, the MMD is computed as:

$$d_{\mathrm{MMD}}(\boldsymbol{\mu}_S, \boldsymbol{\mu}_T) = \sup_{\|f\|_{\mathcal{H}} \leq 1} [\mathbb{E}_{\boldsymbol{\mu}_S} f(X_S) - \mathbb{E}_{\boldsymbol{\mu}_T} f(X_T)], \quad (21)$$

There are also some extensions of the classical MMD. Gretton et al. [70] proposed a multi-kernel MMD. Long et al. [7] developed a joint MMD to measure simultaneously the marginal and conditional distribution discrepancies. Zhang and Wu [71] proposed a discriminative joint probability MMD (DJP-MMD) to directly measure the domain divergence by the joint probability distribution.

5.1.2 Deep TL

Deep TL is very promising in discovering domain invariant factors [79]. Many studies have included MMD [9], [10], [72], [73], moment matching [74], [75], optimal transport [76], [77], or Wasserstein distance [78] in their loss/objective function.

The reason why MMD is also popular in deep TL is that it is based on a sufficiently rich RKHS space, which provides more flexibility when incorporating prior knowledge into TL [20], [87]. However, MMD needs a large amount of data to evaluate the feature means, which may not be suitable for mini-batch based optimization. Moment matching seeks to bridge different distributions by matching also their second- or all-order statistics, in addition to the first-order discrepancy.

Another metric that has attracted increasing interest in recent years is the Wasserstein distance. It comes from the optimal transportation theory, which measures how far one needs to move the mass of one distribution to match another. The Monge version [88] of the optimal transport distance seeks for a map $T^* : \Omega_S \to \Omega_T$ that pushes $\boldsymbol{\mu}_S$ towards $\boldsymbol{\mu}_T$, defined as:

$$T^* = \arg\min_T \int \|\boldsymbol{x} - T(\boldsymbol{x})\|^p d\boldsymbol{\mu}_S(\boldsymbol{x}), \quad (22)$$

where $\boldsymbol{x}$ is an instance from a metric space with finite p-th moment. When T^* exists, it is called an optimal transport map. However, for an indivisible input $\boldsymbol{x}$, the minimizer might not exist. A relaxation of this assumption is the Kantorovich formulation [89], which allows the mass at $\boldsymbol{x}$ to be split and moved to more than one locations. This leads to the Wasserstein distance, defined as:

$$d_{\mathrm{W}_p}(\boldsymbol{\mu}_S, \boldsymbol{\mu}_T) = \left(\inf_{\gamma \sim \prod(\boldsymbol{\mu}_S, \boldsymbol{\mu}_T)} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\gamma}[||\boldsymbol{x} - \boldsymbol{y}||^p] \right)^{1/p}, \quad (23)$$

where $\prod(\boldsymbol{\mu}_S, \boldsymbol{\mu}_T)$ is a space of all joint distributions γ with marginals $\boldsymbol{\mu}_S$ and $\boldsymbol{\mu}_T$, and $\boldsymbol{x}$ and $\boldsymbol{y}$ are two instances sampled from a joint distribution.

Wasserstein distance has many nice properties in representing domain divergence, such as allowing geometrical information to be taken into account, providing a meaningful and smooth representation even when two distributions do not overlap, etc. However, the original Wasserstein distance is hard to compute directly. Shen et al. [78] provided an approximation of the empirical Wasserstein distance by maximizing a domain critic loss.

5.2 Adversarial Training

Adversarial training has also become popular in deep TL. Motivated by generative adversarial networks (GAN) [90], and the theory [19], [39] that a good cross-domain representation should contain no domain-discriminative information, many adversarial TL approaches [11], [78], [91] tried to learn more transferable feature representations by confusing a domain discriminator. For example, Ganin et al. [11] proposed a domain adversarial neural network with a binomial cross-entropy loss,

$$\begin{aligned}\mathcal{L}(F, D) = \mathbb{E}_{\boldsymbol{x}_u \sim P_T(\boldsymbol{x})}[\log(D(F(\boldsymbol{x}_u))] \\ + \mathbb{E}_{\boldsymbol{x}_s \sim P_S(\boldsymbol{x})}[\log(1 - D(F(\boldsymbol{x}_s))],\end{aligned} \quad (24)$$

where F is a feature extractor, and D a domain discriminator.

5.3 Feature Optimization

A basic assumption in feature based TL is that the discrepancy of different domains can be minimized in a common latent space, which may be obtained through selection/weighting of the original features, or dedicated learning. This latent space can also be improved through transferability enhancement.

5.3.1 Feature Selection/Weighting

Feature selection in TL aims to choose a subset of the original features as the common knowledge shared by different domains. Feature weighting, which assigns different weights to the original features, may be viewed as an extension of feature selection.

Sun et al. [92] proposed multi-source part-based TL to combat NT. It first splits the source and target feature spaces into several parts (subsets), and trains a classifier for every part of each source task. Then, parameter TL is used to transfer their parameters to the corresponding part of the target task. Finally, it learns several better classifiers based on these parameters and combines them into a final one by a weighted average.

5.3.2 Latent Feature Space Learning

Long et al. [93] proposed dual TL to distinguish between the common and domain-specific latent factors automatically. Its main idea is to find a latent feature space that can maximally help the classification in the target domain, formulated as an optimization problem of non-negative matrix tri-factorizations:

$$\min_{U_0,U_S,H,V_S\geq 0} \quad \left\| X_S - [U_0, U_S] H V_S^\top \right\|, \tag{25}$$

where X_S is the source domain feature matrix, U_0 and U_S are common feature clusters and domain specific feature clusters, respectively, V_S is a sample cluster assignment matrix, and H is the association matrix. (25) minimizes the marginal distribution discrepancy between different domains to enable optimal knowledge transfer.

Similar ideas have also been investigated in different application fields. Rajesh et al. [94] proposed a dual TL approach for emotion recognition, which seeks a latent feature space of the source and target data by considering the duality between the marginal and conditional distributions. Shi et al. [95] proposed a twin bridge transfer approach for recommender systems, which can transfer similar latent factors between domains without negative effects. It uses latent factor decomposition of users/items and similarity graph transfer to facilitate knowledge transfer across domains and to reduce NT.

5.3.3 Feature Transferability Enhancement

Yosinski et al. [79] defined feature transferability based on its specificity to the domain in which it is trained and its generality. Chen et al. [80] proposed a feature transferability index (FTI):

$$FTI = \alpha \cdot FGI - (1-\alpha) \cdot FSI, \tag{26}$$

where $\alpha \in [0,1]$ is a weight, FGI is the feature generality index, defined as the performance gain of a transfer model on the target data, and FSI is the feature specificity index, defined as the performance drop of a transfer model on target data. The more the transfer benefits the target domain learning, the higher transferability the feature has.

Several approaches have been proposed to enhance the feature transferability.

Chen et al. [81] found that features with small singular values have low transferability in deep network fine-tuning. They proposed a regularization term to reduce NT, by suppressing the small singular values of the feature matrices.

Unfortunately, only focusing on improving feature transferability may lead to poor discriminability. It is necessary to consider both feature transferability and discriminability to overcome NT. Chen et al. [82] proposed to enhance the feature transferability with guaranteed acceptable discriminability by using batch spectral penalization regularization on the largest k singular values. Chen et al. [83] also found that learning discriminative features in the shared feature space can significantly boost the performance of deep DA approaches. Chen et al. [84] developed a hierarchical transferability calibration network, which hierarchically calibrates the transferability of feature representations to balance the transferability and discriminability, by exploring different local regions, images, and instances.

6 Integrated TL Algorithms

Various integrated TL algorithms have been proposed to facilitate knowledge transfer. However, most of them do not provide a lower bound of their performance. A few approaches with guaranteed TL performance are introduced next, including: secure transfer to explicitly avoid NT in the objective function, robust transfer to minimize the noise during the learning process, transitive transfer to handle dramatically different tasks, and model transferability enhancement to design more transferable networks.

TABLE 4
Integrated TL algorithms to overcome NT.

Category	Approaches	References
Secure Transfer	Hypothesis TL, positive TL, face detector adaptation, safe weakly supervised learning	[16], [96]–[98]
Robust Transfer	Feature noise, class noise	[99]–[102]
Transitive Transfer	Distance domain TL	[103], [104]
Transferability Enhancement	TransNorm, adversarial robustness	[105]–[108]

6.1 Secure Transfer

Secure transfer explicitly alleviates NT in the objective function of TL, i.e., the TL algorithm should perform better than the one without transfer.

Kuzborskij and Orabona [96] introduced the hypothesis TL problem and analyzed a class of RLS algorithms with biased regularization to avoid NT. The original RLS algorithm solves the following optimization problem:

$$\min_{\boldsymbol{w}} \frac{1}{m} \sum_{i=1}^{m} (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i)^2 + \lambda \|\boldsymbol{w}\|^2. \tag{27}$$

RLS based TL, which modifies the training set to $\{(\boldsymbol{x}_i, y_i - f'(\boldsymbol{x}_i))\}_{i=1}^m$, generates a hypothesis

$$f_S^{htl'}(\boldsymbol{x}) = T_C(\boldsymbol{x}^\top \hat{\boldsymbol{w}}_S) + f'(\boldsymbol{x}), \tag{28}$$

where $T_C(\hat{y}) = \min(\max(\hat{y}, -C), C)$ is a truncation function to limit the output to $[-C, C]$, and

$$\hat{\boldsymbol{w}}_S = \arg\min_{\boldsymbol{u}} \frac{1}{m} \sum_{i=1}^m (\boldsymbol{w}^\top \boldsymbol{x}_i - y_i + f'(\boldsymbol{x}_i))^2 + \lambda \|\boldsymbol{w}\|^2. \tag{29}$$

They showed that the proposed approach is equivalent to RLS trained solely on the target domain when the source domains are unrelated to the target domain.

Yoon and Li [97] proposed a positive TL (PTL) approach, based on the regularized least squares (RLS) algorithm [109]. It assumes the source parameters follow a normal distribution, and optimizes the following loss function:

$$\min_{\boldsymbol{w}} \ \ell_{\mathcal{T}_l}(\boldsymbol{w}; b) + \beta \mathcal{R}(\boldsymbol{w}) + \lambda N(\boldsymbol{w}; \boldsymbol{\mu}_{\boldsymbol{w}}, \Sigma_{\boldsymbol{w}}), \tag{30}$$

where $\boldsymbol{w}$ denotes model coefficients, $\mathcal{R}(\boldsymbol{w})$ is a regularization term to control the model complexity, and $N(\boldsymbol{w}; \boldsymbol{\mu}_{\boldsymbol{w}}, \Sigma_{\boldsymbol{w}})$ is a regularization term to control the $\boldsymbol{w}$ space, with respect to mean $\boldsymbol{\mu}_{\boldsymbol{w}}$ and variance $\Sigma_{\boldsymbol{w}}$ of the source parameters. They showed that NT arises when λ is too large, thus proposed a one-standard-error rule to select the weight λ and eliminate unhelpful source domains.

Jamal et al. [98] proposed a deep face detector adaptation approach to avoid NT and catastrophic forgetting, by minimizing the following loss function:

$$\min_{\boldsymbol{u}, \tilde{\boldsymbol{\theta}}} \left[\frac{\lambda}{2} \|\boldsymbol{u}\|_2^2 + \mathbb{E}_t \max_{y_t \in \{0,1\}} RES_t(\boldsymbol{w} + \boldsymbol{u}, \tilde{\boldsymbol{\theta}}) \right], \tag{31}$$

where $\boldsymbol{w} + \boldsymbol{u}$ and $\boldsymbol{w}$ are the classifier weights of the target detector and the source detector, respectively, $\boldsymbol{u}$ is the offset weights to constrain the target face detector around the source detector, $\tilde{\boldsymbol{\theta}}$ denotes the parameters of the target feature extractor, and $\mathbb{E}_t$ is the mean average. RES_t is the relative performance loss of the learned target detector over the pre-trained source face detector, which is non-positive after optimization. Hence, the obtained target detector is always no worse than the source detector, i.e., NT is avoided.

Li et al. [16] developed a safe weakly supervised learning (SAFEW) scheme that can be used in semi-supervised learning, DA, etc. Let $h^* = \sum_{i=1}^b \alpha_i h_i$ be the unknown ground-truth label assignment function, where $\{h_i\}_{i=1}^b$ are the base learners and $\boldsymbol{\alpha} = [\alpha_1; \alpha_2; ...; \alpha_b] \geq \boldsymbol{0}$ their weights in a convex set $\mathcal{M}$. The weights satisfy the constraint $\sum_{i=1}^b \alpha_i = 1$. The goal is to learn a prediction h that maximizes the performance gain against the baseline h_0, which is trained from the labeled target data only, by optimizing the following objective function:

$$\max_h \min_{\boldsymbol{\alpha} \in \mathcal{M}} \ \ell(h_0, \sum_{i=1}^b \alpha_i h_i) - \ell(h, \sum_{i=1}^b \alpha_i h_i), \tag{32}$$

i.e., SAFEW optimizes the worst-case performance gain.

6.2 Robust Transfer

Robust Transfer regards the discrepancy between source and target domains as noise; so, it reduces the impact of NT via enhancing the robustness and generalization of the TL models [110]. Pseudo labels and embedded features are two common unstable variables during transfer, thus designing a TL model robust to feature noise and/or class noise in the learning process is of great significance.

To decrease the negative impact of noise in the learned feature spaces, Xu et al. [99] introduced a sparse matrix in unsupervised TL to model the feature noise. The loss function with noise minimization is:

$$\begin{aligned} \min_{P,Z,E} \ & \frac{1}{2}\phi(P, Y, X_S) + \|Z\|_* + \alpha \|Z\|_1 + \beta \|E\|_1 \\ & \text{s.t. } P^\top X_t = P^\top X_s Z + E, \end{aligned} \tag{33}$$

where P, Z and E represent the transformation matrix, reconstruction matrix and noise matrix, respectively, $\phi(P, Y, X_S)$ is a discriminant subspace learning function, and $\|\cdot\|_*$ is the nuclear norm of a matrix. The goal is to align the source and target domains in a common low-rank sparse space with noise suppression.

Robust transfer against class noise regards the differences between the predicted target labels and the real ones as noise. Fang et al. [100] proposed a multi-source TL approach, utilizing a multi-label shared subspace. A base classifier C_i is trained for each source domain $\mathcal{S}_i$. Each target sample $\boldsymbol{x}_j$ has a label set $\hat{y}_j = (y_j, b_1, \cdots, b_k)$, where y_j is the real label, and $b_i = C_i(\boldsymbol{x}_j)$. Based on the class noise assumption, each predicted label set Y_l is composed of a shared label space and domain-specific noise. So, the predictive function is:

$$f_l(\boldsymbol{x}) = \boldsymbol{w}_l^\top \boldsymbol{x} + \boldsymbol{v}_l^\top \Theta \boldsymbol{x}, \tag{34}$$

where $\boldsymbol{w}_l$ and $\boldsymbol{v}_l$ are prediction vectors, and Θ is a mapping matrix of the label subspace. $\boldsymbol{w}_l^\top \boldsymbol{x}$ is the mapping from features to labels on the original input samples, and $\boldsymbol{v}_l^\top \Theta \boldsymbol{x}$ is the shared label predictor. The prediction vectors and mapping matrix can be solved by optimizing the following loss function:

$$\min_{\boldsymbol{w}_l, \boldsymbol{v}_l, \Theta} \ \sum_{l=1}^{n+1} \left(\ell(f_l(\boldsymbol{x}), y^l) + \alpha \|\boldsymbol{w}_l\|^2 + \beta \|\boldsymbol{w}_l + \Theta^\top \boldsymbol{v}_l\|^2 \right), \tag{35}$$

where y^l is the l-th label of $\boldsymbol{x}$, and $\|\boldsymbol{w}_l + \Theta^\top \boldsymbol{v}_l\|^2$ is a label shared subspace in the prediction model. In this way, a shared label subspace representing common and stable knowledge from all source domains is transferred to the target domain to eliminate class noise.

Unsupervised TL usually needs to train a source model to annotate some unlabeled target instances, and misclassification of the target samples introduces class noise to the pseudo labels. The class noise accumulates with the training iterations, and causes NT. To cope with this, Gui et al. [101], [102] developed an approach to predict when NT would occur. They identified and removed the noisy samples in the target domain to reduce class noise accumulation in future training iterations.

6.3 Transitive Transfer

Transitive TL [103] bridges dramatically different source and target domains through one or more intermediate domains to reduce NT.

Tan et al. [104] introduced an instance selection mechanism to identify useful source data, and constructs multiple intermediate domains. They learned a pair of encoding function $f_e(\cdot)$ and decoding function $f_d(\cdot)$ to minimize the reconstruction errors on the selected instance in the intermediate domains, and on all instances in the target domain simultaneously:

$$\begin{aligned}\mathcal{L}(f_e, f_d, v_S, v_I) = R(v_S, v_I) + \frac{1}{n_S}\sum_{i=1}^{n_s} v_S^i \|\hat{\boldsymbol{x}}_S^i - \boldsymbol{x}_S^i\|_2^2 \\ \frac{1}{n_I}\sum_{i=1}^{n_I} v_I^i \|\hat{\boldsymbol{x}}_I^i - \boldsymbol{x}_I^i\|_2^2 + \frac{1}{n_T}\sum_{i=1}^{n_t} \|\hat{\boldsymbol{x}}_T^i - \boldsymbol{x}_T^i\|_2^2,\end{aligned} \tag{36}$$

where $\hat{\boldsymbol{x}}_S^i$, $\hat{\boldsymbol{x}}_T^i$ and $\hat{\boldsymbol{x}}_I^i$ are reconstructions of $\boldsymbol{x}_S^i$, $\boldsymbol{x}_T^i$ and $\boldsymbol{x}_I^i$ from an auto-encoder, v_S and v_I are selection indicators, and $R(v_S, v_I)$ is a regularization term. They also incorporated side information, such as predictions in the intermediate domains, to help the model learn more task-related feature representations.

6.4 Model Transferability Enhancement

Model transferability enhancement can be achieved through transferable normalization, adversarial training, etc.

Transferable normalization (TransNorm) [105] can reduce domain shift in batch normalization [111], and hence improves its performance. Let the mean and variance of the source domain be $\boldsymbol{u}_s$ and $\boldsymbol{\sigma}_s$, and the target domain be $\boldsymbol{u}_t$ and $\boldsymbol{\sigma}_t$. TransNorm quantifies the domain distance as

$$\boldsymbol{d}^{(j)} = \left\| \frac{\boldsymbol{u}_s^{(j)}}{\boldsymbol{\sigma}_s^{2(j)} + \epsilon} - \frac{\boldsymbol{u}_t^{(j)}}{\boldsymbol{\sigma}_t^{2(j)} + \epsilon} \right\|, \tag{37}$$

where j denotes the j-channel in a layer that TransNorm applies to. Then, it uses distance-based probability $\boldsymbol{\alpha}$ to adapt each channel according to its transferability,

$$\boldsymbol{\alpha}^{(j)} = \frac{c(1+\boldsymbol{d}^{(j)})^{-1}}{\sum_{k=1}^{c}(1+\boldsymbol{d}^{(k)})^{-1}}. \tag{38}$$

TransNorm is usually applied after the convolutional layer to enhance the model transferability.

Another way to enhance the model transferability is to improve its robustness to adversarial examples. Adversarial examples are slightly perturbed inputs aiming to fool a machine learning model. A model that is resilient to such adversarial examples is referred to as "adversarially robust", which can be achieved by replacing the standard empirical risk minimization loss with a robust optimization loss [106]:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x},y)\sim D}\left[\max_{\|\boldsymbol{\delta}\|_2 \le \varepsilon} \ell(\boldsymbol{x}+\boldsymbol{\delta}, y; \boldsymbol{\theta})\right], \tag{39}$$

where $\boldsymbol{\delta}$ is a small perturbation, ε is a hyper-parameter to control the perturbation magnitude, and $\boldsymbol{\theta}$ is the set of model parameters.

Several recent studies found that adversarially robust models have better transferability. Salman et al. [107] empirically verified that adversarially robust networks obtained higher transfer accuracies than standard ImageNet models, and increasing the width of a robust network may increase its transfer performance gain. Liang et al. [108] found strong positive correlation between adversarial transferability and knowledge transferability; thus, adversarial transferability may be used as a surrogate to indicate knowledge transferability.

7 NT IN RELATED FIELDS

NT has also been detected and studied in several related fields, including multi-task learning [112], multilingual models [113], and lifelong learning [114].

7.1 Multi-Task Learning

Multi-task learning solves multiple learning tasks jointly, by exploiting commonalities and differences across them. Similar to TL, it needs to facilitate positive transfer among tasks to improve the overall learning performance on all tasks.

Previous studies [115], [116] have observed that conflicting gradients among different tasks may induce NT (also known as negative interference). Various techniques have been explored to remedy negative interference, such as altering the gradients directly [117], [118], learning task relatedness [119], [120], routing networks [121], [122], and searching for Pareto solutions [123], [124], etc.

7.2 Multilingual Models

As a concrete example of multi-task learning, multilingual models have demonstrated success in processing tens or even hundreds of languages simultaneously [125]–[127]. However, not all languages can benefit from this training paradigm. Studies [128] have revealed NT in multilingual models, especially for high-resource languages [127]. Possible remedies include parameter soft-sharing [129], meta-learning [128], and gradient vaccine [116].

7.3 Lifelong Learning

Lifelong learning learns a series of tasks in a sequential order, without revisiting previously seen data. While the goal is to master all tasks in a single model, there are two key challenges, which may lead to NT. First, the model may forget earlier knowledge when trained on new tasks, known as catastrophic forgetting [130]. Second, transferring from early tasks may hurt the performance in later tasks. Existing literature mainly studies how to mitigate catastrophic forgetting using regularization [131], [132] and memory replay [133]–[135], whereas forward NT in lifelong learning is less investigated [136].

8 CONCLUSIONS

Negative transfer is undesirable in TL, and has been attracting increasing research interest recently. This paper systematically reviews recent progresses on overcoming NT, from four perspectives: source data quality, target data quality, domain divergence, and integrated algorithms. To our knowledge, this is the first comprehensive survey on NT.

Although most studies so far only considered one of these four aspects, more than one aspects may also be considered together. We recommend the following procedure to overcome NT in a generic multi-source TL problem:

1) Estimate the transferability of each source domain, and remove or down-weight those with low transferability.
2) For each remaining source domain, select or up-weight the most discriminative, informative and transferrable instances.
3) If possible, also select or weight the target domain instances to further increase the transferability.
4) Select an appropriate TL algorithm, e.g., one with positive transfer guarantees. More considerations and regularization, such as domain divergence reduction, may also be added to the loss function to enhance its performance.

Acknowledgment

This work was supported by the Hubei Province Funds for Distinguished Young Scholars under Grant 2020CFA050, the Technology Innovation Project of Hubei Province of China under Grant 2019AEA171, the National Natural Science Foundation of China under Grants 61873321 and U1913207, and the International Science and Technology Cooperation Program of China under Grant 2017YFE0128300. The authors would also like to thank Mr. Zirui Wang of the Carnegie Mellon University for insightful discussions.

References

[1] D. Wu, Y. Xu, and B.-L. Lu, "Transfer learning for EEG-based brain-computer interfaces: A review of progress made since 2016," *IEEE Trans. on Cognitive and Developmental Systems*, 2020, in press.
[2] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.
[3] L. Zhang, "Transfer adaptation learning: A decade survey," *arXiv preprint arXiv:1903.04687*, 2019.
[4] Z. Chen and M. W. Daehler, "Positive and negative transfer in analogical problem solving by 6-year-old children," *Cognitive Development*, vol. 4, no. 4, pp. 327–344, 1989.
[5] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. 24th Int'l Conf. on Machine learning*, Corvallis, OR, Jun. 2007, pp. 193–200.
[6] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.
[7] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int'l Conf. on Computer Vision*, Sydney, Australia, Dec. 2013, pp. 2200–2207.
[8] J. Zhang, W. Li, and P. Ogunbona, "Joint geometrical and statistical alignment for visual domain adaptation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, Jul. 2017, pp. 1859–1867.
[9] M. Ghifary, W. B. Kleijn, and M. Zhang, "Domain adaptive neural networks for object recognition," in *Proc. Pacific Rim Int'l Conf. on Artificial Intelligence*, Queensland, Australia, Jun. 2014, pp. 898–904.
[10] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. 32nd Int'l Conf. on Machine Learning*, Lille, France, Jul. 2015, pp. 97–105.
[11] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.
[12] H. Tang and K. Jia, "Discriminative adversarial domain adaptation." in *Proc. 34th AAAI Conf. on Artificial Intelligence*, New York, NY, Feb. 2020, pp. 5940–5947.
[13] M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich, "To transfer or not to transfer," in *Proc. NIPS 2005 Workshop on Transfer Learning*, vol. 898, Vancouver, Canada, May 2005, pp. 1–4.
[14] Z. Wang, Z. Dai, B. Póczos, and J. Carbonell, "Characterizing and avoiding negative transfer," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Long Beach, CA, Jun. 2019, pp. 11 293–11 302.
[15] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int'l Conf. on Artificial Neural Networks*, Rhodes, Greece, Oct. 2018, pp. 270–279.
[16] Y.-F. Li, L.-Z. Guo, and Z.-H. Zhou, "Towards safe weakly supervised learning," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2019, in press.
[17] C. Nguyen, T. Hassner, C. Archambeau, and M. Seeger, "LEEP: A new measure to evaluate transferability of learned representations," in *Proc. 37th Int'l Conf. on Machine Learning*, Vienna, Austria, Jul. 2020, pp. 5640–5651.
[18] J. Wu and J. He, "Continuous transfer learning with label-informed distribution alignment," *arXiv preprint arXiv:2006.03230*, 2020.
[19] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, no. 1-2, pp. 151–175, 2010.
[20] I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani, "A survey on domain adaptation theory: Learning bounds and theoretical guarantees," *arXiv preprint arXiv:2004.11829*, 2020.
[21] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. Cambridge, MA: MIT Press, 2018.
[22] Y. Yao and G. Doretto, "Boosting for transfer learning with multiple sources," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, San Francisco, CA, Jun. 2010, pp. 1855–1862.
[23] D. Lin, X. An, and J. Zhang, "Double-bootstrapping source data selection for instance-based transfer learning," *Pattern Recognition Letters*, vol. 34, no. 11, pp. 1279–1285, 2013.
[24] Y. Xu, H. Yu, Y. Yan, Y. Liu *et al.*, "Multi-component transfer metric learning for handling unrelated source domain samples," *Knowledge-Based Systems*, p. 106132, 2020.
[25] C.-W. Seah, Y.-S. Ong, and I. W. Tsang, "Combating negative transfer from predictive distribution differences," *IEEE Trans. on Cybernetics*, vol. 43, no. 4, pp. 1153–1165, 2012.
[26] J. Li, S. Qiu, Y.-Y. Shen, C.-L. Liu, and H. He, "Multisource transfer learning for cross-subject EEG emotion recognition," *IEEE Trans. on Cybernetics*, 2019.
[27] Z. Peng, W. Zhang, N. Han, X. Fang, P. Kang, and L. Teng, "Active transfer learning," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 30, no. 4, pp. 1022–1036, 2020.
[28] Z. Peng, Y. Jia, and J. Hou, "Non-negative transfer learning with consistent inter-domain distribution," *IEEE Signal Processing Letters*, vol. 27, pp. 1720–1724, 2020.
[29] L. Ge, J. Gao, H. Ngo, K. Li, and A. Zhang, "On handling negative transfer and imbalanced distributions in multiple source transfer learning," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 7, no. 4, pp. 254–271, 2014.
[30] S. Moon and J. G. Carbonell, "Completely heterogeneous transfer learning with attention-what and what not to transfer," in *Proc. Int'l Joint Conf. on Artificial Intelligence*, vol. 1, no. 1, Melbourne, Australia, Aug. 2017, pp. 1–2.
[31] E. Eaton *et al.*, "Selective transfer between learning tasks using task-based boosting," in *Proc. 25th AAAI Conf. on Artificial Intelligence*, San Francisco, CA, Aug. 2011, pp. 337–342.
[32] S. Liu, Y. Liang, and A. Gitter, "Loss-balanced task weighting to reduce negative transfer in multi-task learning," in *Proc. 33rd AAAI Conf. on Artificial Intelligence*, vol. 33, Honolulu, HI, Jan. 2019, pp. 9977–9978.
[33] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Providence, RI, Jun. 2012, pp. 2066–2073.
[34] A. M. Azab, L. Mihaylova, K. K. Ang, and M. Arvaneh, "Weighted transfer learning for improving motor imagery-based brain-computer interface," *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 7, pp. 1352–1359, 2019.

[35] Y.-P. Lin and T.-P. Jung, "Improving EEG-based emotion classification using conditional transfer learning," *Frontiers in Human Neuroscience*, vol. 11, p. 334, 2017.
[36] W. Zhang and D. Wu, "Manifold embedded knowledge transfer for brain-computer interfaces," *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 5, pp. 1117–1127, 2020.
[37] G. Xie, Y. Sun, M. Lin, and K. Tang, "A selective transfer learning method for concept drift adaptation," in *Proc. Int'l Symposium on Neural Networks*, Hokkaido, Japan, Jun. 2017, pp. 353–361.
[38] Y. Bao, Y. Li, S.-L. Huang, L. Zhang, L. Zheng, A. Zamir, and L. Guibas, "An information-theoretic approach to transferability in task transfer learning," in *Proc. IEEE Int'l Conf. on Image Processing*, Taiwan, China, 2019, pp. 2309–2313.
[39] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2007, pp. 137–144.
[40] Z. Wang and J. Carbonell, "Towards more reliable transfer learning," in *Proc. Joint European Conf. on Machine Learning and Knowledge Discovery in Databases*, Dublin, Ireland, 2018, pp. 794–810.
[41] B. Cao, S. J. Pan, Y. Zhang, D.-Y. Yeung, and Q. Yang, "Adaptive transfer learning," in *Proc. 24th AAAI Conf. on Artificial Intelligence*, vol. 2, no. 5, Atlanta, GA, Jul. 2010, p. 7.
[42] A. T. Tran, C. V. Nguyen, and T. Hassner, "Transferability and hardness of supervised classification tasks," in *Proc. IEEE Int'l Conf. on Computer Vision*, Seoul, Korea, Nov. 2019, pp. 1395–1405.
[43] Z. Cao, M. Long, J. Wang, and M. I. Jordan, "Partial transfer learning with selective adversarial networks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, Jun. 2018, pp. 2724–2732.
[44] J. Zhang, Z. Ding, W. Li, and P. Ogunbona, "Importance weighted adversarial nets for partial domain adaptation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, Jun. 2018, pp. 8156–8164.
[45] Z. Cao, K. You, M. Long, J. Wang, and Q. Yang, "Learning to transfer examples for partial domain adaptation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Long Beach, CA, Jun. 2019, pp. 2985–2994.
[46] N. Zhang, S. Deng, Z. Sun, J. Chen, W. Zhang, and H. Chen, "Transfer learning for relation extraction via relation-gated adversarial learning," *arXiv preprint arXiv:1908.08507*, 2019.
[47] Z. Ding, M. Shao, and Y. Fu, "Transfer learning for image classification with incomplete multiple sources," in *Proc. Int'l Joint Conf. on Neural Networks*, Vancouver, Canada, Jul. 2016, pp. 2188–2195.
[48] T. Kamishima, M. Hamasaki, and S. Akaho, "TrBagg: A simple transfer learning method and its application to personalization in collaborative tagging," in *Proc. 9th IEEE Int'l Conf. on Data Mining*, 2009, pp. 219–228.
[49] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.
[50] Y.-L. Yu and C. Szepesvári, "Analysis of kernel mean matching under covariate shift," in *Proc. 29th Int'l Conf. on Machine Learning*, Edinburgh, Scotland, Jun. 2012, pp. 1147–1154.
[51] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
[52] L. L. Minku, "Transfer learning in non-stationary environments," in *Learning from Data Streams in Evolving Environments*. Cham, Switzerland: Springer, 2019, pp. 13–37.
[53] Y. Sun, K. Tang, Z. Zhu, and X. Yao, "Concept drift adaptation by exploiting historical knowledge," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4822–4832, 2018.
[54] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. 31st Int'l Conf. on Machine Learning*, Beijing, China, Jun. 2014, pp. 647–655.
[55] P. Agrawal, R. Girshick, and J. Malik, "Analyzing the performance of multilayer neural networks for object recognition," in *Proc. European Conf. on Computer Vision*, Zurich, Switzerland, 2014, pp. 329–344.
[56] D. Wu, "Active semi-supervised transfer learning (ASTL) for offline BCI calibration," in *Proc. IEEE Int'l. Conf. on Systems, Man and Cybernetics*, Banff, Canada, October 2017.
[57] D. Wu, V. J. Lawhern, W. D. Hairston, and B. J. Lance, "Switching EEG headsets made easy: Reducing offline calibration effort using active wighted adaptation regularization," *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 11, pp. 1125–1137, 2016.
[58] D. Wu, B. J. Lance, and T. D. Parsons, "Collaborative filtering for brain-computer interaction using transfer learning and active class selection," *PLoS ONE*, 2013.
[59] B. Gholami, P. Sahu, O. Rudovic, K. Bousmalis, and V. Pavlovic, "Unsupervised multi-target domain adaptation: An information theoretic approach," *IEEE Trans. on Image Processing*, vol. 29, pp. 3993–4002, 2020.
[60] Z. Liu, Z. Miao, X. Pan, X. Zhan, D. Lin, S. X. Yu, and B. Gong, "Open compound domain adaptation," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, Jun. 2020, pp. 12 406–12 415.
[61] M. Rohrbach, M. Stark, and B. Schiele, "Evaluating knowledge transfer and zero-shot learning in a large-scale setting," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Colorado Springs, CO, Jun. 2011, pp. 1641–1648.
[62] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2013.
[63] P. Panareda Busto and J. Gall, "Open set domain adaptation," in *Proc. IEEE Int'l Conf. on Computer Vision*, Venice, Italy, Oct. 2017, pp. 754–763.
[64] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Long Beach, CA, Jun. 2019, pp. 2720–2729.
[65] D. Wu, "Pool-based sequential active learning for regression," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 30, no. 5, pp. 1348–1359, 2019.
[66] R. Lomasky, C. E. Brodley, M. Aernecke, D. Walt, and M. Friedl, "Active class selection," in *Proc. 18th European Conf. on Machine Learning*, Warsaw, Poland, September 2007, pp. 640–647.
[67] S. Rahman, S. Khan, and F. Porikli, "A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning," *IEEE Trans. on Image Processing*, vol. 27, no. 11, pp. 5652–5667, 2018.
[68] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf, "Measuring statistical dependence with Hilbert-Schmidt norms," in *Proc. Int'l Conf. on Algorithmic Learning Theory*, Padova, Italy, Oct. 2005, pp. 63–77.
[69] S. Si, D. Tao, and B. Geng, "Bregman divergence-based regularization for transfer subspace learning," *IEEE Trans. on Knowledge and Data Engineering*, vol. 22, no. 7, pp. 929–942, 2009.
[70] A. Gretton, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, K. Fukumizu, and B. K. Sriperumbudur, "Optimal kernel choice for large-scale two-sample tests," in *Proc. Advances in Neural Information Processing Systems*, Lake Tahoe, NV, Dec. 2012, pp. 1205–1213.
[71] W. Zhang and D. Wu, "Discriminative joint probability maximum mean discrepancy (DJP-MMD) for domain adaptation," in *Proc. Int'l Joint Conf. on Neural Networks*, Glasgow, UK, Jul. 2020.
[72] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. 34th Int'l Conf. on Machine Learning*, Sydney, Australia, Aug. 2017, pp. 2208–2217.
[73] C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Póczos, "MMD GAN: Towards deeper understanding of moment matching network," in *Proc. Advances in Neural Information Processing Systems*, Long Beach, CA, Dec. 2017, pp. 2203–2213.
[74] B. Sun and K. Saenko, "Deep CORAL: Correlation alignment for deep domain adaptation," in *Proc. European Conf. on Computer Vision*, Amsterdam, The Netherlands, Oct. 2016, pp. 443–450.
[75] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *Proc. IEEE Int'l Conf. on Computer Vision*, Seoul, Korea, Nov. 2019, pp. 1406–1415.
[76] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint distribution optimal transportation for domain adaptation," in *Proc. Advances in Neural Information Processing Systems*, Long Beach, CA, Dec. 2017, pp. 3730–3739.
[77] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy, "Optimal transport for domain adaptation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1853–1865, 2017.
[78] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proc. 32nd AAAI Conf. on Artificial Intelligence*, New Orleans, LA, Feb. 2018, pp. 4058–4065.

[79] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Advances in Neural Information Processing Systems*, Montréal, Canada, Dec. 2014, pp. 3320–3328.

[80] J. Chen, F. Lécué, J. Z. Pan, I. Horrocks, and H. Chen, "Knowledge-based transfer learning explanation," in *Proc. 16th Int'l Conf. on Principles of Knowledge Representation and Reasoning*, Tempe, AZ, Oct. 2018, pp. 349–358.

[81] X. Chen, S. Wang, B. Fu, M. Long, and J. Wang, "Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning," in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, 2019, pp. 1908–1918.

[82] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *Proc. 36th Int'l Conf. on Machine Learning*, Long Beach, CA, Jun. 2019, pp. 1081–1090.

[83] C. Chen, Z. Chen, B. Jiang, and X. Jin, "Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation," in *Proc. 33rd AAAI Conf. on Artificial Intelligence*, vol. 33, Honolulu, HI, Jan. 2019, pp. 3296–3303.

[84] C. Chen, Z. Zheng, X. Ding, Y. Huang, and Q. Dou, "Harmonizing transferability and discriminability for adapting object detectors," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, Jun. 2020, pp. 8869–8878.

[85] A. Bakry, M. Elhoseiny, T. El-Gaaly, and A. Elgammal, "Digging deep into the layers of CNNs: In search of how CNNs achieve view invariance," in *Proc. Int'l Conf.on Learning Representations*, San Juan, Puerto Rico, May 2016.

[86] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. 3, pp. 723–773, 2012.

[87] M. Arbel, A. Korba, A. Salim, and A. Gretton, "Maximum mean discrepancy gradient flow," in *Prooc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019, pp. 6484–6494.

[88] G. Monge, "Mémoire sur la théorie des déblais et des remblais," *Histoire de l'Académie Royale des Sciences de Paris*, 1781.

[89] L. V. Kantorovich, "On the translocation of masses," *Journal of Mathematical Sciences*, vol. 133, no. 4, pp. 1381–1382, 2006.

[90] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Advances in Neural Information Processing Systems*, Montréal, Canada, Dec. 2014, pp. 2672–2680.

[91] S. Motiian, Q. Jones, S. Iranmanesh, and G. Doretto, "Few-shot adversarial domain adaptation," in *Proc. Advances in Neural Information Processing Systems*, Long Beach, CA, Dec. 2017, pp. 6670–6680.

[92] S. Sun, Z. Xu, and M. Yang, "Transfer learning with part-based ensembles," in *Proc. Int'l Workshop on Multiple Classifier Systems*, Nanjing, China, May 2013, pp. 271–282.

[93] M. Long, J. Wang, G. Ding, W. Cheng, X. Zhang, and W. Wang, "Dual transfer learning," in *Proc. 2012 SIAM Int'l Conf. on Data Mining*, Brussels, Belgium, Dec. 2012, pp. 540–551.

[94] M. Rajesh and J. Gnanasekar, "Annoyed realm outlook taxonomy using twin transfer learning," *Int'l Journal of Pure and Applied Mathematics*, vol. 116, no. 21, pp. 549–558, 2017.

[95] J. Shi, M. Long, Q. Liu, G. Ding, and J. Wang, "Twin bridge transfer learning for sparse collaborative filtering," in *Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, Gold Coast, Australia, Apr. 2013, pp. 496–507.

[96] I. Kuzborskij and F. Orabona, "Stability and hypothesis transfer learning," in *Proc. 30th Int'l Conf. on Machine Learning*, Atlanta, GA, Jun. 2013, pp. 942–950.

[97] H. Yoon and J. Li, "A novel positive transfer learning approach for telemonitoring of Parkinson's disease," *IEEE Trans. on Automation Science and Engineering*, vol. 16, no. 1, pp. 180–191, 2018.

[98] M. Abdullah Jamal, H. Li, and B. Gong, "Deep face detector adaptation without negative transfer or catastrophic forgetting," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, Jun. 2018, pp. 5608–5618.

[99] Y. Xu, X. Fang, J. Wu, X. Li, and D. Zhang, "Discriminative transfer subspace learning via low-rank and sparse representation," *IEEE Trans. on Image Processing*, vol. 25, no. 2, pp. 850–863, 2015.

[100] M. Fang, Y. Guo, X. Zhang, and X. Li, "Multi-source transfer learning based on label shared subspace," *Pattern Recognition Letters*, vol. 51, pp. 101–106, 2015.

[101] L. Gui, R. Xu, Q. Lu, J. Xu, J. Xu, B. Liu, and X. Wang, "Cross-lingual opinion analysis via negative transfer detection," in *Proc. 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, MD, Jun. 2014, pp. 860–865.

[102] L. Gui, R. Xu, Q. Lu, J. Du, and Y. Zhou, "Negative transfer detection in transductive transfer learning," *Int'l Journal of Machine Learning and Cybernetics*, vol. 9, no. 2, pp. 185–197, 2018.

[103] B. Tan, Y. Song, E. Zhong, and Q. Yang, "Transitive transfer learning," in *Proc. 21st ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, Sydney, Australia, Aug. 2015, pp. 1155–1164.

[104] B. Tan, Y. Zhang, S. J. Pan, and Q. Yang, "Distant domain transfer learning," in *Proc. 31st AAAI Conf. on Artificial Intelligence*, San Francisco, CA, Feb. 2017, pp. 2604–2610.

[105] X. Wang, Y. Jin, M. Long, J. Wang, and M. I. Jordan, "Transferable normalization: Towards improving transferability of deep neural networks," in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019, pp. 1953–1963.

[106] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int'l Conf. on Learning Representations*, Vancouver, Canada, Apr. 2018.

[107] K. Liang, J. Y. Zhang, O. Koyejo, and B. Li, "Does adversarial transferability indicate knowledge transferability?" *arXiv preprint arXiv:2006.14512*, 2020.

[108] H. Salman, A. Ilyas, L. Engstrom, A. Kapoor, and A. Madry, "Do adversarially robust ImageNet models transfer better?" *arXiv preprint arXiv:2007.08489*, 2020.

[109] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2006.

[110] X. Zhu and X. Wu, "Class noise vs. attribute noise: A quantitative study," *Artificial Intelligence Review*, vol. 22, no. 3, pp. 177–210, 2004.

[111] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int'l Conf. on Machine Learning*, Lille, France, Jul. 2015, pp. 448–456.

[112] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint arXiv:1706.05098*, 2017.

[113] G. Lample and A. Conneau, "Cross-lingual language model pretraining," *arXiv preprint arXiv:1901.07291*, 2019.

[114] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[115] T. Yu, S. Kumar, A. Gupta, S. Levine, K. Hausman, and C. Finn, "Gradient surgery for multi-task learning," *arXiv preprint arXiv:2001.06782*, 2020.

[116] Z. Wang, Y. Tsvetkov, O. Firat, and Y. Cao, "Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models," *arXiv preprint arXiv:2010.05874*, 2020.

[117] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proc. Int'l Conf. on Machine Learning*, Stockholm, Sweden, Jul. 2018, pp. 794–803.

[118] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, Jun. 2018, pp. 7482–7491.

[119] Y. Zhang and D.-Y. Yeung, "A convex formulation for learning task relationships in multi-task learning," *arXiv preprint arXiv:1203.3536*, 2012.

[120] C. Shui, M. Abbasi, L.-É. Robitaille, B. Wang, and C. Gagné, "A principled approach for learning task similarity in multitask learning," in *Proc. 28th Int'l Joint Conf. on Artificial Intelligence*, Macao, China, Aug. 2019, pp. 3446–3452.

[121] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[122] C. Rosenbaum, I. Cases, M. Riemer, and T. Klinger, "Routing networks and the challenges of modular and compositional computation," *arXiv preprint arXiv:1904.12774*, 2019.

[123] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in *Proc. Advances in Neural Information Processing Systems*, Montréal, Canada, Dec. 2018, pp. 527–538.

[124] X. Lin, H.-L. Zhen, Z. Li, Q.-F. Zhang, and S. Kwong, "Pareto multi-task learning," in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019, pp. 12 060–12 070.

[125] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
[126] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, “Unsupervised cross-lingual representation learning at scale,” *arXiv preprint arXiv:1911.02116*, 2019.
[127] N. Arivazhagan, A. Bapna, O. Firat, D. Lepikhin, M. Johnson, M. Krikun, M. X. Chen, Y. Cao, G. Foster, C. Cherry *et al.*, “Massively multilingual neural machine translation in the wild: Findings and challenges,” *arXiv preprint arXiv:1907.05019*, 2019.
[128] Z. Wang, Z. C. Lipton, and Y. Tsvetkov, “On negative interference in multilingual models: Findings and a meta-learning treatment,” *arXiv preprint arXiv:2010.03017*, 2020.
[129] J. Guo, D. J. Shah, and R. Barzilay, “Multi-source domain adaptation with mixture of experts,” in *Proc. 2018 Conf. on Empirical Methods in Natural Language Processing*, Brussels, Belgium, nov 2018, pp. 4694–4703.
[130] M. McCloskey and N. J. Cohen, “Catastrophic interference in connectionist networks: The sequential learning problem,” in *Psychology of Learning and Motivation*, 1989, vol. 24, pp. 109–165.
[131] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, “Efficient lifelong learning with A-GEM,” in *Proc. Int’l Conf. on Learning Representations*, New Orleans, LA, May 2019.
[132] P. Sprechmann, S. Jayakumar, J. Rae, A. Pritzel, A. P. Badia, B. Uria, O. Vinyals, D. Hassabis, R. Pascanu, and C. Blundell, “Memory-based parameter adaptation,” in *Proc. Int’ Conf. on Learning Representations*, Vancouver, Canada, May 2018.
[133] D. Lopez-Paz and M. Ranzato, “Gradient episodic memory for continual learning,” in *Proc. Advances in Neural Information Processing Systems*, Long Beach, CA, Dec. 2017, pp. 6467–6476.
[134] D. Rolnick, A. Ahuja, J. Schwarz, T. Lillicrap, and G. Wayne, “Experience replay for continual learning,” in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019, pp. 348–358.
[135] C. d. M. d’Autume, S. Ruder, L. Kong, and D. Yogatama, “Episodic memory in lifelong language learning,” in *Proc. Advances in Neural Information Processing Systems*, Vancouver, Canada, Dec. 2019.
[136] Z. Wang, S. V. Mehta, B. Póczos, and J. Carbonell, “Efficient meta lifelong-learning with limited memory,” *arXiv preprint arXiv:2010.02500*, 2020.