Multi-task Learning by Leveraging the Semantic Information

Fan Zhou[1], Brahim Chaib-draa[1], Boyu Wang[2,3]*

[1] Université Laval, Quebec City, QC, G1V 0A6, Canada.
[2] University of Western Ontario, London, ON N6A 5B7, Canada
[3] Vector Institute, Toronto, ON M5G 1M1, Canada
fan.zhou.1@ulaval.ca, chaib@ift.ulaval.ca, bwang@csd.uwo.ca

Abstract

One crucial objective of multi-task learning is to align distributions across tasks so that the information between them can be transferred and shared. However, existing approaches only focused on matching the marginal feature distribution while ignoring the semantic information, which may hinder the learning performance. To address this issue, we propose to leverage the label information in multi-task learning by exploring the semantic conditional relations among tasks. We first theoretically analyze the generalization bound of multi-task learning based on the notion of Jensen-Shannon divergence, which provides new insights into the value of label information in multi-task learning. Our analysis also leads to a concrete algorithm that jointly matches the semantic distribution and controls label distribution divergence. To confirm the effectiveness of the proposed method, we first compare the algorithm with several baselines on some benchmarks and then test the algorithms under label space shift conditions. Empirical results demonstrate that the proposed method could outperform most baselines and achieve state-of-the-art performance, particularly showing the benefits under the label shift conditions.

Introduction

General machine learning paradigms typically focus on learning individual tasks. Even though significant progress has been achieved, recent successes in machine learning, especially in the deep learning area, usually rely on a large amount of labelled data to obtain a small generalization error. In practice, however, acquiring labelled data could be highly prohibitive, *e.g.*, when classifying multiple objects in an image (Long et al. 2017), when analyzing patient data in healthcare data analysis (Wang and Pineau 2015; Zhou et al. 2021b), or when modelling users' products preferences (Murugesan and Carbonell 2017). Data hungry has become a long-term problem for deep learning. Multi-task learning (MTL) aims at addressing this issue by simultaneously learning from multiple tasks and leveraging the shared knowledge across them. Many MTL approaches have been implemented in computer vision (Zhao et al. 2018), natural language processing (Bingel and Søgaard 2017), medical data analysis (Moeskops et al. 2016; Li, Carlson et al. 2018) or brain-computer interaction (Wang et al. 2020) problems etc. It has been shown

with benefits to reduce the amount of annotated data per task to reach the desired performance.

The crucial idea behind MTL is to extract and leverage the knowledge and information shared across the tasks to improve the overall performances (Wang et al. 2019b), which can be achieved by task-invariant feature learning (Maurer, Pontil, and Romera-Paredes 2016; Luo, Tao, and Wen 2017) or task relation learning (Zhang and Yeung 2012; Bingel and Søgaard 2017; Zhou et al. 2021b). One major issue with most of the existing feature learning approaches is that they only align the marginal distributions $\mathbb{P}(x)$ to extract the shared features without taking advantage of label information of the tasks. Consequently, the features can lack discriminative power for supervised learning even if their marginal features have been matched properly (Dou et al. 2019). Furthermore, only aligning $\mathbb{P}(x)$ cannot address the MTL problems when the label space of each task differs from each other, *i.e.*, label shift problem (Redko et al. 2019).

While a few algorithms have been proposed to use semantic matching for MTL (Zhuang et al. 2017; Luo, Tao, and Wen 2017) and have shown improved performances, the theoretical justifications for the value of labels remain elusive. Most theoretical results (Shui et al. 2019; Mao, Liu, and Lin 2020) for MTL derive from the notion of $\mathcal{H}$-divergence (Ben-David et al. 2010) or Wasserstein adversarial training (Redko, Habrard, and Sebban 2017; Shen et al. 2018), which did not take the label information into consideration. As a result, they usually require additional assumptions, *e.g.*, assuming the combined error across tasks is small (Ben-David et al. 2010) to ensure the algorithms succeed, which may not hold in practice.

To this end, we propose the first theoretical analysis for MTL that considers semantic matching. Specifically, our results reveal that the MTL loss can be upper-bounded in terms of the pair-wise discrepancy between the tasks, measured by the Jensen-Shannon divergences of label distribution $\mathbb{P}(y)$ and semantic distribution $\mathbb{P}(x|y)$.

The contributions of our work are trifold: **1.** In contrast to previous theoretical results (Shui et al. 2019; Mao, Liu, and Lin 2020), which only consider the marginal distribution discrepancy (*e.g.*, $\mathcal{H}$-divergence), we build a complete MTL theoretical framework upon the joint distribution discrepancy based on the Jensen-Shannon divergence. Thus, our result provides a deeper understanding of the general problem of

MTL and insights into how to extract and leverage shared knowledge in a more appropriate and principled way by exploiting the label information. **2.** Our analysis also reveals how the label shift problems affects the learning procedure of MTL, which was missing in previous results. **3.** Our theoretical result leads to a novel algorithm, namely Semantic Multi-Task Learning (SMTL) algorithm, which explicitly leverages the label information for MTL.

Specifically, the proposed SMTL algorithm simultaneously learns task-invariant features and task similarities to match the semantic distributions across the tasks and minimizes label distribution divergence via a label re-weighting loss function. In addition, SMTL is based on a novel centroid matching approach for task-invariant feature learning, which is more efficient than other adversarial training based algorithms. To examine the effectiveness of the proposed algorithm, we evaluate SMTL on several benchmarks. The empirical results show that the proposed approach outperforms the baselines achieving state-of-the-art performance. Besides, the experiment results show that our algorithm can be more time-efficient than the adversarial baselines, which confirms the benefits of our proposed method. Furthermore, we also conduct a simulation of the label distribution shift scenario showing that the proposed algorithm could handle the label distribution shift problems that cannot be properly addressed by other baselines.

Related Works

Multi-task Learning

MTL has been prevalent to lots of recent machine learning topics (Li, Liu, and Chan 2014; Wang, Pineau, and Balle 2016; Teh et al. 2017). Our work majorly relates to some feature representation learning-based approaches and task relation based approaches. (Maurer, Pontil, and Romera-Paredes 2016) firstly analyzed generalization error of representation-based approaches. (Murugesan et al. 2016; Pentina and Lampert 2017) approached the online and transductive learning problem by MTL using a weighted summation of the losses. (Wang et al. 2019a) analyzed the algorithmic stability in MTL. For task relations learning, (Zhang and Yeung 2010; Cao et al. 2018a) define a convex optimization problem to measure relationships while (Long et al. 2017; Kendall, Gal, and Cipolla 2018) propose probabilistic models by constructing task covariance matrices or estimate the multi-task likelihood via a deep Bayes model. Latterly, (Shui et al. 2019; Mao, Liu, and Lin 2020) combines the feature representation learning and task relations learning together and analyzed generalization bound under the adversarial training scheme motivated by the domain adaptation problems (Ben-David et al. 2010; Shen et al. 2018; Shui et al. 2020), showing improved performances in vision and language processing applications, respectively. This kind of approach neglected the value of label information, which may impair the learning process. There were also some approaches to investigate the situation where the label space may differ from each other. (Su et al. 2020) investigate the problem where the task samples are confusing, and the model is trained to extract task concepts by discriminating them.

Semantic Transfer

Leveraging semantic information has been prevalent in some machine learning topics (Long et al. 2014), since it is easy to implement when few labels are available. (Dou et al. 2019; Zhou et al. 2021a; Matsuura and Harada 2020) proposed to leverage the semantic information by adopting the metric learning objective under an unsupervised scheme to enforce a class-specific alignment for domain generalization problems. (Zhang et al. 2019) propose to learn the class-specific prototype semantic information by a symmetric network to align semantic features for unsupervised domain adaptation problems. (Xie et al. 2018) theoretically analyzed the semantic transfer method for domain adaptation problems with a pseudo label. (Shui et al. 2020) investigated the value of matching conditional and semantic distribution in domain adaptation problems. In the notion of MTL, leveraging the semantic information was investigated inconspicuously through some matrix decomposition methods in the notion of tensor learning. For example, (Zhuang et al. 2017) proposed a non-negative matrix factorization-based approach to learn a common semantic feature space underlying feature spaces of each task. (Luo, Tao, and Wen 2017) proposed to leveraging high-order statistics among tasks by analyzing the prediction weight covariance tensor of them. However, the theoretical analysis for leveraging the semantic information is still open.

Notations and Preliminaries

Problem Setup

Assuming a set of T tasks $\{\hat{\mathcal{D}}_t\}_{t=1}^T$, each of them is generated by the underlying distribution $\mathcal{D}_t$ over $\mathcal{X}$ and by the underlying labelling functions $f_t : \mathcal{X} \rightarrow \mathcal{Y}$ for $\{(\mathcal{D}_t, f_t)\}_{t=1}^T$. A multi-task (MTL) learner aims to find T hypothesis: $h_1, \ldots, h_T$ over the hypothesis space $\mathcal{H}$ to minimize the average expected error of all the tasks:

$$\arg \min_{h \in \mathcal{H}} \frac{1}{T} \sum_{i=1}^T \epsilon_t(h_t),$$

where $\epsilon_i(h_i) \equiv \epsilon_i(h_i, f_i) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_i} \ell(h_i(\mathbf{x}), f_i(\mathbf{x}))$ is the expected error of task t and ℓ is the loss function. For each task i, assume that there are m_i examples. For each task i, we consider a minimization of weighted empirical loss for each task by defining a simplex $\boldsymbol{\alpha}_j \in \Delta^T = \{\boldsymbol{\alpha}_{i,j} \geq 0, \sum_{j=1}^T \boldsymbol{\alpha}_{i,j} = 1\}$ for the corresponding weight for task j. *It could be viewed as an explicit indicator of the task relations revealing how much information leveraged from other tasks.*

The empirical loss $w.r.t.$ the hypothesis h for task i could be defined as,

$$\hat{\epsilon}_{\boldsymbol{\alpha}_i}(h) = \sum_{j=1}^T \boldsymbol{\alpha}_{i,j} \hat{\epsilon}_j(h), \tag{1}$$

where $\hat{\epsilon}_i(h) = \frac{1}{m_i} \sum_{j=1}^{m_i} \ell(h(x_j), y_j)$ is the average empirical error for task i.

Most of the existing adversarial based MTL approaches, *e.g.* (Mao, Liu, and Lin 2020; Shui et al. 2019), were motivated by the theory of (Ben-David et al. 2010) using the

$\mathcal{H}$ divergence. However, the $\mathcal{H}$-divergence theory itself is limited in many scenarios, *e.g.* when tackling the (semantic) conditional shifts and understanding open set learning problems (Panareda Busto and Gall 2017; Cao et al. 2018b; You et al. 2019). In this work, we adopt the *Jensen-Shannon Divergence* (D_{JS}) to measure the differences of tasks and analyze its potentials for controlling the semantic (covariate) relations *i.e.* measure the divergence between the tasks.

Definition 1 (Jensen-Shannon divergence). *Let $\mathcal{D}_i(x,y)$ and $\mathcal{D}_j(x,y)$ be two distribution over $\mathcal{X} \times \mathcal{Y}$, and let $\mathcal{M} = \frac{1}{2}(\mathcal{D}_i + \mathcal{D}_j)$, then the Jensen-Shannon (JS) divergence between $\mathcal{D}_i$ and $\mathcal{D}_j$ is,*

$$D_{JS}(\mathcal{D}_i \| \mathcal{D}_j) = \frac{1}{2}[D_{KL}(\mathcal{D}_i \| \mathcal{M}) + D_{KL}(\mathcal{D}_j \| \mathcal{M})]$$

where $D_{KL}(\mathcal{D}_i \| \mathcal{D}_j)$ is the Kullback–Leibler divergence.

It has been prevalent in adversarial training based approaches in transfer learning (Dou et al. 2019; Matsuura and Harada 2020; Zhao et al. 2019). In practice, we could compute the *Total Variation* distance (d_{TV}) since it is an upper bound of JS divergence (Lin 1991):

$$d_{TV}(\mathcal{D}_i \| \mathcal{D}_j) = \frac{1}{2}|\mathcal{D}_i - \mathcal{D}_j| \tag{2}$$

Leverage the Semantic and Label Information

As aforementioned, previous MTL advancements (*e.g.* (Mao, Liu, and Lin 2020; Shui et al. 2019)) mostly only matched the marginal distribution while neglecting the labelling information. A successful MTL algorithm should take the semantic (covariate) conditional information into consideration. For example, consider the classification of different digits dataset (e.g. MNIST ($\mathcal{D}_i$) and SVHN ($\mathcal{D}_j$)) using MTL, when conditioning on the certain digit category $Y = y$, it is clear that $\mathcal{D}_i(x|Y=y) \neq \mathcal{D}_j(x|Y=y)$, indicating the necessity of considering semantic information in MTL.

Moreover, a long-neglected issue in existing MTL approaches is that most MTL approaches all implicitly assumed that the label marginal distribution $\mathbb{P}(y)$ are the same. However, this may not hold. For example, in a medical diagnostics problem, if the data are collected from different hospitals with different populations in that area, the label spaces for data can vary from each other. *Label shift* refers to the situation where the source and target distribution have different label distribution (Redko et al. 2019), *i.e.*, $D_{\mathrm{JS}}(\mathcal{D}_i(y), \mathcal{D}_j(y)) \neq 0$. While this issue has been investigated in literature by transfer learning (Panareda Busto and Gall 2017; Geng, Huang, and Chen 2020; Azizzadenesheli et al. 2019), however, the analysis towards label space shift in MTL is still open.

We show, both theoretically and empirically, that the label space shift can impair the MTL performance. Our theoretical and empirical results reveal that a successful multitask learning algorithm should not only match the semantic distribution $\mathcal{D}(x|y)$ among all the tasks via adversarial training with J-S divergence but also measure the label distribution $\mathcal{D}(y)$ under a re-weighting scheme for all tasks. Specifically, we consider the label distribution drift scenario, where the number of classes is the same to each other across the tasks while the number of instances in each class has obvious drift, *i.e.*, imbalanced label distribution for all the tasks.

Methodology and Theoretical Insights

Intuitively, when aligning the distribution of different tasks, features from the same class should be mapped near to each other in the feature space satisfying the semantic conditional relations. We firstly analyze the error bound with the notion of Jensen-Shannon divergence based form to measure the tasks discrepancies. Then, we further extend the results to analyze to control the label space divergence and the semantic conditional distribution divergence. *All the proofs are delegated to the supplementary materials.*

Theorem 1. *Let $\mathcal{H}$ be the hypothesis class $h \in \mathcal{H}$. Suppose we have T tasks generated by the underlying distribution and labelling function $\{(\mathcal{D}_1, f_1), \ldots, (\mathcal{D}_T, f_T)\}$. Assume the loss function ℓ is bounded by L ($\max(\ell) - \min(\ell) \leq L$). Then, with high probability we have*

$$\frac{1}{T}\sum_{t=1}^{T} \epsilon_t(h) \leq \frac{1}{T}\sum_{t=1}^{T} \epsilon_{\alpha_t}(h) + \frac{\lambda_0}{4T}L^2$$
$$+ \frac{2}{\lambda_0 T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} D_{JS}(\mathcal{D}_t(x,y)\|\mathcal{D}_i(x,y))$$

where $\lambda_0 > 0$ is a constant.

Theorem 1 showed that the averaged MTL error is bounded by an averaged summation of all the tasks, the averaged summation of task distribution divergence among all pair of tasks and some constant value.

This bound indicates the joint distribution while we aim to leverage the label ($\mathbb{P}(y)$) and semantic ($\mathbb{P}(x|y)$) information, based on the aforementioned theorem 1, we could then decompose it into the following results,

Corollary 1. *Follow the setting of Theorem 1, we can further bound the overall task error by*

$$\frac{1}{T}\sum_{t=1}^{T} \epsilon_t(h) \leq \frac{1}{T}\sum_{t=1}^{T} \epsilon_{\alpha_t}(h) + \boldsymbol{\lambda} \underbrace{D_{JS}(\mathcal{D}_t(y)\|\mathcal{D}_i(y))}_{\text{Label distribution divergence}}$$
$$+ \boldsymbol{\lambda} \underbrace{\mathbb{E}_{y \sim \mathcal{D}_t(y)} D_{JS}(\mathcal{D}_t(x|y)\|\mathcal{D}_i(x|y))}_{\text{Semantic distribution divergence}}$$
$$+ \boldsymbol{\lambda} \underbrace{\mathbb{E}_{y \sim \mathcal{D}_i(y)} D_{JS}(\mathcal{D}_t(x|y)\|\mathcal{D}_i(x|y))}_{\text{Semantic distribution divergence}} + \frac{\lambda_0}{4T}L^2$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{T \times T}$ is the corresponding matrix whose t-th row and i-th column element is $\frac{2}{\lambda_0 T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}$

Remark: Different from (Shui et al. 2019; Mao, Liu, and Lin 2020), which were motivated by (Ben-David et al. 2010), our theoretical results do not rely on extra assumption of the existence of the optimal hypothesis to achieve a small combined error. Besides, our results also provide new insight by take advantage of label information and semantic conditional relations.

Corollary 1 implies that the averaged error over all tasks is bounded by the summation of task errors (the first term

Algorithm 1 The Global Semantic Matching Method

Input: Training set from each tasks
Parameter: Feature extractor $\boldsymbol{\theta}^f$; decay parameter γ
Output: The semantic loss

1: **for** k=1 to K **do**
2: $\quad C_{\mathcal{D}_i}^k \leftarrow \frac{1}{|\mathcal{D}_i^k|} \sum_{(x_i,y_i)\in\mathcal{D}_i^k} \boldsymbol{\theta}^f(x_i)$
3: $\quad C_{\mathcal{D}_j}^k \leftarrow \frac{1}{|\mathcal{D}_j^k|} \sum_{(x_j,y_j)\in\mathcal{D}_j^k} \boldsymbol{\theta}^f(x_j)$
4: $\quad \mathbf{C}_{\mathcal{D}_i}^k \leftarrow \gamma C_{\mathcal{D}_i}^k + (1-\gamma)C_{\mathcal{D}_i}^k$
5: $\quad \mathbf{C}_{\mathcal{D}_j}^k \leftarrow \gamma C_{\mathcal{D}_j}^k + (1-\gamma)C_{\mathcal{D}_j}^k$
6: $\quad \mathcal{L}_S \leftarrow \mathcal{L}_S + \Phi(C_{\mathcal{D}_i}, C_{\mathcal{D}_j})$
7: **end for**
8: **return** $\mathcal{L}_S$

in *R.H.S.* of Corollary 1), the label distribution divergence (the second term), a constant term (the third term), and the semantic distribution divergence term (the last two terms). The first term could be easily optimized by a general supervised learning loss (*e.g.* the cross-entropy loss). To minimize this bound now is equivalent to match the semantic distribution among the tasks and measure the label divergence. Since the labels of each task samples are available to the learner, we could leverage the label and semantic information directly.

Label Re-weighting Loss

Corollary 1 indicates that the error is also controlled by the label divergence term $D_{\mathrm{JS}}(\mathcal{D}_i(y)\|\mathcal{D}_j(y))$. To reduce the influences caused by the label space shifts, we could adopt a label correction re-weighting loss (Lipton, Wang, and Smola 2018) based on the number of instances in each class,

$$\hat{\epsilon}_{\mathcal{D}_i}^\beta(h) = \sum_{(x_i,y_i)\in\hat{\mathcal{D}}_i} \boldsymbol{\beta}(y_i)\ell(h(x_i), y_i)) \qquad (3)$$

where $\boldsymbol{\beta} \in \mathbb{R}^{K\times 1}$ is weight for each class, and $\boldsymbol{\beta}_j$ is the weight for class y_j. For task i with total m_i instances, the weight of class $k \in K$ (K is the total number of classes) is computed by $\boldsymbol{\beta}_k = \sum \frac{|\#y=y_k|}{m_i}$. This re-weighting scheme guarantees the instances from different classes could have equal probability to be sampled when training the model, which re-weights the loss according to frequency of each class that occurs during training. By doing so, the learner will not neglect those classes who have fewer instances and therefore takes care of label drift.

Note: the coefficient $\boldsymbol{\beta}$ is computed for re-weighting the loss from each task while $\boldsymbol{\alpha}$ is a set of weights indicating the relations between each other, *i.e.* how much information leveraged from other tasks.

When training the model, we maintain the task specific loss $\mathcal{L}_i = \hat{\epsilon}_{\mathcal{D}_i}^\beta(h)$ and compute the total classification loss

$$\mathcal{L}_C = \sum_{i=1}^T \boldsymbol{\alpha}_i \hat{\epsilon}_{\mathcal{D}_i}^\beta(h) \qquad (4)$$
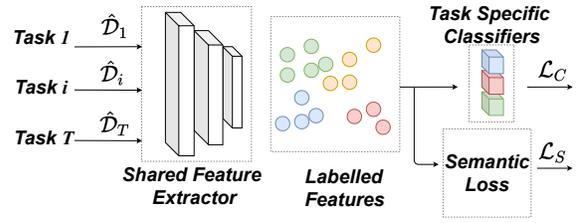
Figure 1: The overall model architecture

Semantic Matching and Task Relation Update

To compute Eq. 4, we still need to estimate the task relation coefficients $\boldsymbol{\alpha}$. As it indicates the relations between tasks, we are not able to measure its value at the beginning. To a better estimation, we need to update the coefficient $\boldsymbol{\alpha}$ automatically during the training process. Through Corollary 1, we could solve the coefficients via an convex optimization as

$$\min_{\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_T} \quad \mathcal{L}_C + \sum_{i,t=1}^T \boldsymbol{\alpha}_{t,i} \sum_y (\hat{\mathcal{D}}_i(y) + \hat{\mathcal{D}}_t(y)))\mathbf{E}_{i,t}$$
$$+ \sum_{t=1}^T \|\boldsymbol{\alpha}_t\|_2 \qquad (5)$$
$$\text{s.t.} \quad \sum_t \boldsymbol{\alpha}_t = 1$$

where $\mathbf{E}_{i,t} = D_{\mathrm{JS}}\left(\hat{\mathcal{D}}_t(x|y)\|\hat{\mathcal{D}}_t(x|y)\right)$ is the empirical semantic distribution divergence.

To align the semantic distribution, we adopt the centroid matching method by computing the Euclidean distance between two centroids in the embedding space. Denote $C_{\mathcal{D}_i}^k$ and $C_{\mathcal{D}_j}^k$ by two feature centroids from class k of distribution $\mathcal{D}_i$ and $\mathcal{D}_j$ respectively, it could be computed by,

$$\Phi(C_{\mathcal{D}_i}^k, C_{\mathcal{D}_j}^k) = \|C_{\mathcal{D}_i}^k - C_{\mathcal{D}_j}^k\|^2 \qquad (6)$$

Algorithm 2 The proposed Semantic Multi-task learning algorithm

Require: Samples from different tasks $\{\hat{\mathcal{D}}_t\}_{t=1}^T$, initial coefficients $\{\boldsymbol{\alpha}_t\}_{t=1}^T$ and learning rate η
Ensure: Neural network $\boldsymbol{\theta}^f$, $\{\boldsymbol{\theta}_t^C\}_{t=1}^T$ and coefficient $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_T$
1: **while** Algorithm Not converge **do**
2: $\quad$ **for** min-batch $\{(\mathbf{x}_t^b, \mathbf{y}_t^b)\}$ from task $\{\hat{\mathcal{D}}_t\}_{t=1}^T$ **do**
3: $\quad\quad$ Compute the classification objective $\mathcal{L}_C$ by Eq. 4
4: $\quad\quad$ Compute the semantic matching objective $\mathcal{L}_S$ via Algorithm 1
5: $\quad\quad$ Update the network parameters $\boldsymbol{\theta}^f, \boldsymbol{\theta}_t^c$ by:
$\quad\quad \boldsymbol{\theta}^f \leftarrow \boldsymbol{\theta}^f - \eta \frac{\partial \mathcal{L}_C + \mathcal{L}_S}{\partial \boldsymbol{\theta}^f}$ and $\boldsymbol{\theta}_t^c \leftarrow \boldsymbol{\theta}_t^c - \eta \frac{\partial \mathcal{L}_C + \mathcal{L}_S}{\partial \boldsymbol{\theta}_t^c}$
6: $\quad$ **end for**
7: $\quad$ Update $\{\boldsymbol{\alpha}_t\}_{t=1}^T$ by optimizing over Eq. (5).
8: **end while**

Approach	3K				5K				8K			
	M	M-M	S	avg.	M	M-M	S	Avg.	M	M-M	S	Avg.
Vanilla	93.9	77.1	57.3	76.1	96.3	79.1	68.0	81.1	97.7	83.7	71.4	84.2
Weighed	89.3	76.4	70.2	78.3	91.8	74.2	73.6	79.8	92.3	76.9	74.1	81.1
Adv.H	90.1	**81.2**	70.8	80.7	91.9	**83.7**	73.6	82.9	94.9	**85.2**	79.1	86.4
Adv.W	96.8	81.3	69.5	82.5	97.5	83.4	72.6	84.5	98.1	84.3	75.4	86.1
Multi-Obj.	**97.5**	76.9	54.8	76.4	**98.2**	80.2	61.2	79.9	**98.5**	82.8	69.9	83.7
AMTNN	96.9	80.8	77.1	84.9	97.7	83.6	78.4	86.6	98.1	83.1	80.2	87.1
Ours	95.4	80.1	**81.5**	**85.7**	95.8	82.4	**83.3**	**87.2**	96.0	83.9	**85.4**	**88.4**

Table 1: The empirical results (in %) on the digits datasets.

Approach	10%					15%					20%				
	A	C	P	S	avg.	A	C	P	S	avg.	A	C	P	S	avg.
Vanilla	78.8	81.8	87.1	83.4	82.8	82.8	86.7	89.3	84.9	85.9	84.1	87.9	90.5	85.8	87.1
Weighted	82.7	86.2	89.7	84.9	85.9	**85.1**	87.9	91.2	87.3	87.9	86.1	89.7	92.1	88.4	89.1
Adv.W	78.8	83.9	87.6	84.0	83.6	83.6	84.8	84.7	84.0	84.2	83.5	89.5	91.4	87.3	87.9
Adv.H	76.8	84.3	88.3	84.0	83.3	82.6	87.8	89.9	86.1	86.6	84.4	87.6	91.5	88.3	87.9
Multi-Obj.	79.4	83.4	87.0	82.9	83.2	82.7	87.5	89.1	86.5	86.4	84.3	88.7	91.0	88.8	88.2
AMTNN	**82.8**	86.7	91.3	81.2	85.5	85.1	88.8	92.9	85.8	88.2	**87.4**	89.9	93.7	87.7	89.7
Ours	80.6	**87.9**	**94.4**	**91.9**	**88.9**	83.4	**89.8**	**94.5**	**93.1**	**90.2**	86.3	**91.6**	**95.5**	**93.8**	**91.8**

Table 2: The empirical results (in %) on PACS dataset with AlexNet as feature extractor.

Method	A	C	D	W	avg.
Vanilla	84.2	80.6	90.8	81.8	84.3
Weighted	88.1	81.5	94.9	94.2	88.6
Adv.H	81.5	73.8	91.4	86.1	83.3
Adv.W	84.9	80.9	94.5	87.5	86.9
Multi-Obj.	82.3	76.7	91.2	86.8	84.3
AMTNN	89.3	84.3	**98.4**	94.1	91.7
Ours	**90.9**	**85.3**	98.1	**94.2**	**92.1**

Table 3: Average test accuracy (in %) of MTL algorithms on Office-Caltech dataset with AlexNet as feature extractor.

Then, we re-visited the *moving average centroid* method by (Xie et al. 2018) where a global centroid matrix was maintained to compute the semantic information between a labeled source and an unlabeled target distribution in DA problem. Unlike (Xie et al. 2018), we could explicitly measure the semantic distribution across all the tasks rather than through assigning pseudo labels to compute them. We illustrate the modified moving average centroid method, namely *The Global Semantic Matching Method* in Algorithm 1.

Remark: In Algorithm 1, $\mathcal{L}_S$ is an approximation of the *total variation distance* (see Eq. 2) of the two centroids, which is an upper bound of $\mathcal{D}_{JS}\left(\mathcal{D}_i(x|y), \mathcal{D}_j(x|y)\right)$. Compared with adversarial training based method, this semantic matching process does not need to train pair-wised discriminators, which may help to reduce the computational costs. For example, for m tasks, (Shui et al. 2019) needs to train $\frac{m(m-1)}{2}$ discriminators. When the number of tasks increase, the training procedure may become time-inefficient.

The Full Objective and Proposed Algorithm

With the key components introduced in previous sections, we could summarize the full method. A general model architecture is provided in Fig. 1. The model learns multiple tasks jointly by a shared feature extractor. For each task, we implement a task-specific classifier. The classifier was trained under a re-weighting loss via measuring label distribution of each task, and we also maintain the semantic loss to match the semantic distribution across tasks to achieve the semantic transfer objective. The proposed Semantic Multi-task learning (SMTL) method is illustrated in Algorithm 2

Experiments and Analysis

We examined the proposed approach comparing with several baselines on **Digits**, **PACS** (Li et al. 2017), **Office-31** (Saenko et al. 2010), **Office-Caltech** (Gong et al. 2012) and **Office-home** (Venkateswara et al. 2017) dataset. For the Digits benchmark, we evaluate the algorithms on *MNIST* (M), *MNIST-M* (M-M) and *SVHN* (S) simultaneously. The PACS dataset contains images from four tasks: *Photo* (P), *Art painting* (A), *Cartoon* (C), Sketch (S), with objects from 7 classes. Office-31 dataset contains three different tasks: *Amazon* (A), *Dslr* (D) and *Webcam* (W); Office-Caltech contains four different tasks: *Amazon* (A), *Dslr* (D), *Webcam* (W) and *Caltech* (C); Office-home contains four different tasks: *Art* (Ar), *Clipart* (Cl), *Product* (Pr) and *Real World* (Rw).

To evaluate the performance of our proposed algorithm, we re-implement and compare our method with the following principled approaches:

- Vanilla MTL: Learning all the tasks simultaneously while optimizing the average summation loss: $\frac{1}{T}\sum_{t=1}^{T}\hat{\epsilon}_t(\boldsymbol{\theta}^f, \boldsymbol{\theta}_t^c)$, *i.e.*, compute the loss uniformly.

Approach	5%				10%				20%			
	A	**D**	**W**	**avg.**	**A**	**D**	**W**	**avg.**	**A**	**D**	**W**	**avg.**
Vanilla	61.3	71.8	72.1	68.3	73.2	80.6	82.1	78.6	79.4	91.2	93.1	87.9
Weighted	63.3	87.4	84.9	78.5	70.6	92.1	88.4	83.7	76.8	**96.6**	95.6	89.7
Adv.W	66.5	71.8	69.9	69.7	74.7	85.9	85.7	82.1	79.3	93.8	92.2	88.4
Adv.H	65.8	73.5	71.4	70.2	71.0	84.1	89.4	81.4	79.7	93.7	93.7	89.1
Multi-Obj.	**68.9**	72.5	72.3	71.3	74.6	86.8	86.9	82.8	79.2	92.1	94.7	88.6
AMTNN	63.3	80.1	85.4	79.3	71.3	**92.8**	89.6	84.6	80.2	94.2	94.4	89.6
Ours	68.5	**87.9**	**86.5**	**80.9**	**75.7**	92.8	90.8	**86.4**	**81.1**	96.5	**96.1**	**91.2**

Table 4: The empirical results (in %) on Office-31 dataset with ResNet-18 as feature extractor.

Approach	5%					10%					20%				
	Ar	**Cl**	**Pr**	**Rw**	**avg.**	**Ar**	**Cl**	**Pr**	**Rw**	**avg.**	**Ar**	**Cl**	**Pr**	**Rw**	**avg.**
Vanilla	26.2	30.1	57.6	47.4	40.3	35.8	43.3	67.1	56.8	50.7	45.5	56.1	74.4	62.6	59.6
Weighted	26.8	31.8	59.2	50.5	42.1	38.2	45.3	69.1	58.3	52.7	47.9	56.7	75.6	64.8	61.2
Adv.W	26.8	32.7	58.3	47.1	41.2	38.5	44.4	67.6	59.5	52.3	47.9	56.7	75.4	65.7	61.3
Adv.H	27.7	32.1	59.6	51.1	42.7	39.0	45.8	69.4	58.8	53.2	46.7	56.5	75.6	65.1	61.0
Multi-Obj.	25.6	31.7	58.7	51.5	41.8	34.6	43.3	66.1	56.8	50.2	46.2	56.6	74.3	62.8	59.8
AMTNN	32.5	34.5	56.3	49.9	43.3	41.1	47.5	68.4	58.9	53.9	48.9	**60.7**	75.4	64.7	62.1
Ours	**38.3**	**40.9**	**62.3**	**55.5**	**49.2**	**43.8**	**50.4**	**71.3**	**62.3**	**57.1**	**51.2**	60.6	**77.9**	**66.1**	**64.3**

Table 5: The empirical results (in %) on Office-home dataset with ResNet-18 as feature extractor.

Method	A	D	W	avg.
Cls. only	79.4	91.2	93.1	87.9
$w.o.$ re-weighting	80.2	94.7	94.1	89.6
$w.o.$ sem. matching	79.8	96.1	95.4	90.4
$w.o.$ cvx opt.	80.7	**96.8**	95.3	90.9
Full method	81.1	96.5	96.1	**91.2**

Table 6: Ablation studies on Office-31 dataset.

- Weighted MTL: Adapted from (Murugesan et al. 2016), learning a weighted summation of losses over different tasks: $\frac{1}{T}\sum_{t=1}^{T}\hat{\epsilon}_{\boldsymbol{\alpha}_t}(\boldsymbol{\theta}^f, \boldsymbol{\theta}_t^c)$

- Adv.H: Adapted from (Liu, Qiu, and Huang 2017) by using the same loss function while training with $\mathcal{H}$-divergence as adversarial objective.

- Adv.W: Replace the adversarial loss of Adv.H by Wasserstein distance based adversarial training method.

- Multi-Obj.: Adapted from (Sener and Koltun 2018), casting the multi-task learning problem as a multi-objective problem

- AMTNN: Adapted from (Shui et al. 2019), a gradient reversal layer with Wasserstein adversarial training method.

Experiments on Benchmark Datasets

We first evaluate the MTL algorithms on Digits dataset. In order to show the effectiveness of MTL methods when dealing with small amount of labelled instances, we follow the evaluation protocol of (Shui et al. 2019) by randomly selecting $3k$, $6k$ and $8k$ instances of the training dataset and choose $1k$ dataset as validation set while testing one the full test set. For the SVHN dataset, we resize the images to $28 \times 28 \times 1$, except for that, we do not apply any data-augmentation towards to digits dataset. A LeNet-5 (LeCun et al. 1998) model is implemented as feature extractor and three 3-layer MLPs are deployed as task-specific classifiers, and extract the semantic feature from the classifier with size 128. We adopt the Adam optimizer (Kingma and Ba 2014) for training the model from scratch. The model is trained for 50 epochs while the initial learning rate is set by 1×10^{-3} and is decayed 5% for every 5 epochs. The results are reported in Table 1.

For the computer vision applications, we then test the SMTL algorithm comparing with the baselines on PACS and Caltech datasests using the AlexNet (Krizhevsky, Sutskever, and Hinton 2012) as feature extractor. For investigating the performance when limited amount labelled instances are available, we evaluate the algorithms on PACS dataset randomly select 10%, 15% and 20% of the total dataset for training, respectively. Since this Office-Caltech dataset is relatively small, we only test the dataset by using 20% of the total images to train the model. We use the pre-trained AlexNet provided by *PyTorch* (Paszke et al. 2019) while removing the last FC layers as feature extractor (out feature size 4096). On top of the feature extractor, we implement several MLPs as task-specific classifiers. The test results are reported in Table 2 and Table 3, respectively. After that, we then evaluate the algorithms on Office-31 and Office-Home dataset by randomly select 5%, 10% and 20% training samples with pre-trained ResNet-18 model of *PyTorch* while removing the last FC layers as feature extractor (out feature size 512). For
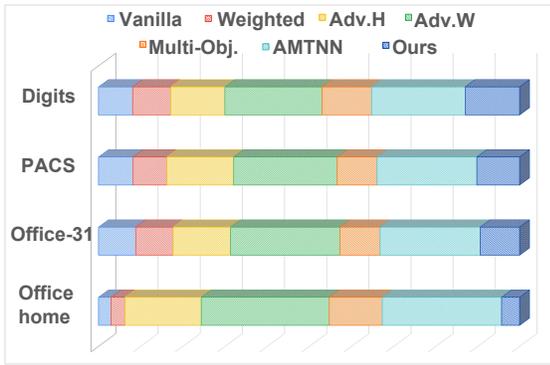
Figure 2: Relative time comparison (one training epoch) of the MTL algorithm on different benchmarks.


Figure 3: Performance comparison under label distribution drift scenario. *Left*: Evaluations on Office-31 dataset with different drift ratio; *Right*: Evaluations on Office-Home dataset with different drift ratio.

these four vision benchmarks we follow the pre-processing and train/val/test protocol by (Long et al. 2017; Cao et al. 2018a; Li et al. 2017). We adopt the Adam optimizer with initial learning rate 2×10^{-4} and decayed 5% every 5 epochs while totally training for 80 epochs. For stable training, we also enable the *weight-decay* in Adam optimizer to enforce a L_2 regularization. The test results are reported in Table 4 and 5, respectively.

From Table. $1 \sim 5$, we could observe that our proposed method could outperform the baselines and improve the benchmark performances with state-of-the-art performances. Particularly, we found that when there are only few of labelled instances (*e.g.* 5% of the total instances), our method could have a large margin of improvements regarding the baselines. *This confirms the effectiveness of our methods when dealing with limited data.*

Further Analysis

Ablation Studies In order to investigate the effectiveness of each component of our method, we conduct ablation studies (Table 6) of the proposed method on Office-31 dataset (20% of total instances) with four ablations, namely 1) *Cls. only*: remove all of the re-weighting scheme, semantic matching and the convex optimization towards updating $\boldsymbol{\alpha}$; 2) *w.o.* re-weighting: removing the re-weighting scheme inside the label weighting loss; 3) *w.o.* sem. matching: omitting the semantic matching; and 4) *w.o.* cvx. opt.: omit the optimization procedure for updating $\boldsymbol{\alpha}$, *i.e,* Eq. (5). The results showed that the label re-weighting scheme is crucial for the algorithm. Besides, we also observe -1.0% drop when omitting the semantic matching procedure and -0.5% once we omit the convex optimization procedure for $\boldsymbol{\alpha}$.

Time Efficiency As our method doesn't rely on adversarial training, it has better time efficiency. We compare the time-efficiency of the MTL algorithms on Digits ($8k$), PACS (20%), Office-31 (20%) and Office-home (20%) datasets, and report the time comparison of one training epoch in a relative percentage bar chart in Fig. 2. The adversarial based training methods (*Adv.H, Adv.W* and *AMTNN*) take longer time for a training epoch, especially on the Office-home dataset.
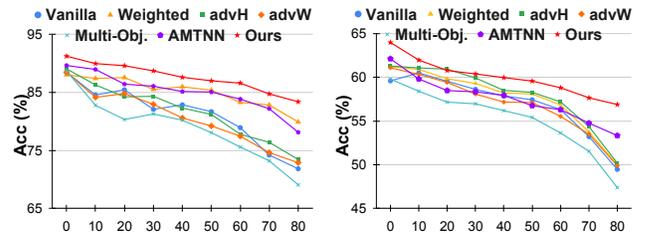
Take the improved performance (Table 1~5) into consideration. Our method could improve that benchmark performance while reduce the time needed for training. *This also demonstrates the benefits of algorithm in terms of time-efficiency.*

Performance under Label Shift We evaluate the MTL algorithms' performance under label shift situation where the label distribution drifts, *i.e.*, the number of classes keeps the same with original one while some classes drift by a certain percentage for a specific task on Office-31 and Office-home dataset. The drift simulation is implemented as keeping all the classes within all the tasks while simulating a significant label distribution drift by randomly drop out some part of the instances of certain tasks. For Office-31 dataset, the task *Amazon*'s class $1 \sim 10$, task *Dslr*'s class $10 \sim 20$ and task *Webcam*'s class $21 \sim 30$ are drifted with different ratios ($10\% \sim 80\%$) while for Office-home dataset, we drift classes $1 \sim 16$ of *Art*, classes $17 \sim 32$ of *Clipart*, classes $33 \sim 48$ of *Product*, and classes $49 \sim 64$ of *Real World* with different ratios ($10\% \sim 80\%$).

We show the performance under label distribution drift ranging from $10\% \sim 80\%$ on Office-31 dataset (left) and on Office-Home dataset (right) in Fig. 3. As we could observe from Fig. 3, when the label space drifts, all the algorithms drop off. Our algorithm could outperform the baselines with a large margin when label space shift. *This demonstrates the benefits of our algorithm for handling label shift problems.*

Conclusion

We propose to leverage the labeling information across different tasks in multi-task learning problems. We first theoretically analyze the generalization bound of multi-task learning based on the notion of Jensen-Shannon divergence, which provides new insights into the value of label information by exploiting the semantic conditional distribution in multi-task learning. Our theoretical results also lead to a concrete algorithm that jointly matches the semantic distribution and controls label distribution divergence. The empirical results demonstrates the effectiveness of our algorithm on improving the benchmark performance with better time efficiency and particularly show the benefits when label distribution shift.

Acknowledgements

References

Azizzadenesheli, K.; Liu, A.; Yang, F.; and Anandkumar, A. 2019. Regularized learning for domain adaptation under label shifts. *arXiv preprint arXiv:1903.09734* .

Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Vaughan, J. W. 2010. A theory of learning from different domains. *Machine learning* 79(1-2): 151–175.

Bingel, J.; and Søgaard, A. 2017. Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 164–169. Valencia, Spain: Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E17-2026.

Cao, W.; Wu, S.; Yu, Z.; and Wong, H.-S. 2018a. Exploring Correlations Among Tasks, Clusters, and Features for Multi-task Clustering. *IEEE transactions on neural networks and learning systems* 30(2): 355–368.

Cao, Z.; Ma, L.; Long, M.; and Wang, J. 2018b. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 135–150.

Dou, Q.; de Castro, D. C.; Kamnitsas, K.; and Glocker, B. 2019. Domain generalization via model-agnostic learning of semantic features. In *Advances in Neural Information Processing Systems*, 6447–6458.

Geng, C.; Huang, S.-j.; and Chen, S. 2020. Recent advances in open set recognition: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2066–2073. IEEE.

Kendall, A.; Gal, Y.; and Cipolla, R. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7482–7491.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.; et al. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11): 2278–2324.

Li, D.; Yang, Y.; Song, Y.-Z.; and Hospedales, T. M. 2017. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, 5542–5550.

Li, S.; Liu, Z.-Q.; and Chan, A. B. 2014. Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 482–489.

Li, Y.; Carlson, D. E.; et al. 2018. Extracting Relationships by Multi-Domain Matching. In *Advances in Neural Information Processing Systems*, 6799–6810.

Lin, J. 1991. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information theory* 37(1): 145–151.

Lipton, Z. C.; Wang, Y.-X.; and Smola, A. 2018. Detecting and correcting for label shift with black box predictors. *arXiv preprint arXiv:1802.03916* .

Liu, P.; Qiu, X.; and Huang, X. 2017. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742* .

Long, M.; Cao, Z.; Wang, J.; and Philip, S. Y. 2017. Learning multiple tasks with multilinear relationship networks. In *Advances in Neural Information Processing Systems*, 1594–1603.

Long, M.; Wang, J.; Ding, G.; Sun, J.; and Yu, P. S. 2014. Transfer joint matching for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1410–1417.

Luo, Y.; Tao, D.; and Wen, Y. 2017. Exploiting High-Order Information in Heterogeneous Multi-Task Feature Learning. In *IJCAI*, 2443–2449.

Mao, Y.; Liu, W.; and Lin, X. 2020. Adaptive Adversarial Multi-task Representation Learning. In *International Conference on Machine Learning*.

Matsuura, T.; and Harada, T. 2020. Domain Generalization Using a Mixture of Multiple Latent Domains. In *AAAI*.

Maurer, A.; Pontil, M.; and Romera-Paredes, B. 2016. The benefit of multitask representation learning. *The Journal of Machine Learning Research* 17(1): 2853–2884.

Moeskops, P.; Wolterink, J. M.; van der Velden, B. H.; Gilhuijs, K. G.; Leiner, T.; Viergever, M. A.; and Išgum, I. 2016. Deep learning for multi-task medical image segmentation in multiple modalities. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 478–486. Springer.

Murugesan, K.; and Carbonell, J. 2017. Active learning from peers. In *Advances in Neural Information Processing Systems*, 7008–7017.

Murugesan, K.; Liu, H.; Carbonell, J.; and Yang, Y. 2016. Adaptive smoothed online multi-task learning. In *Advances in Neural Information Processing Systems*, 4296–4304.

Panareda Busto, P.; and Gall, J. 2017. Open set domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, 754–763.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, 8026–8037.

Pentina, A.; and Lampert, C. H. 2017. Multi-task Learning with Labeled and Unlabeled Tasks. In *International Conference on Machine Learning*, 2807–2816.

Redko, I.; Habrard, A.; and Sebban, M. 2017. Theoretical analysis of domain adaptation with optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 737–753. Springer.

Redko, I.; Morvant, E.; Habrard, A.; Sebban, M.; and Bennani, Y. 2019. *Advances in Domain Adaptation Theory*. Elsevier.

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting visual category models to new domains. In *European conference on computer vision*, 213–226. Springer.

Sener, O.; and Koltun, V. 2018. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, 527–538.

Shen, J.; Qu, Y.; Zhang, W.; and Yu, Y. 2018. Wasserstein distance guided representation learning for domain adaptation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Shui, C.; Abbasi, M.; Robitaille, L.-E.; Wang, B.; and Gagné, C. 2019. A Principled Approach for Learning Task Similarity in Multitask Learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*.

Shui, C.; Chen, Q.; Wen, J.; Zhou, F.; Gagné, C.; and Wang, B. 2020. Beyond $\mathcal{H}$ -Divergence: Domain Adaptation Theory With Jensen-Shannon Divergence. *arXiv preprint arXiv:2007.15567* .

Su, X.; Jiang, Y.; Guo, S.; and Chen, F. 2020. Task Understanding from Confusing Multi-task Data. In *International Conference on Machine Learning*.

Teh, Y.; Bapst, V.; Czarnecki, W. M.; Quan, J.; Kirkpatrick, J.; Hadsell, R.; Heess, N.; and Pascanu, R. 2017. Distral: Robust multitask reinforcement learning. In *Advances in Neural Information Processing Systems*, 4496–4506.

Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep Hashing Network for Unsupervised Domain Adaptation. In *(IEEE) Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wang, B.; Mendez, J.; Cai, M.; and Eaton, E. 2019a. Transfer learning via minimizing the performance gap between domains. In *Advances in Neural Information Processing Systems*, 10645–10655.

Wang, B.; and Pineau, J. 2015. Online boosting algorithms for anytime transfer and multitask learning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Wang, B.; Pineau, J.; and Balle, B. 2016. Multitask Generalized Eigenvalue Program. In *AAAI*, 2115–2121.

Wang, B.; Wong, C. M.; Kang, Z.; Liu, F.; Shui, C.; Wan, F.; and Chen, C. P. 2020. Common Spatial Pattern Reformulated for Regularizations in Brain-Computer Interfaces. *IEEE Trans. Cybern.* doi:10.1109/TCYB.2020.2982901.

Wang, B.; Zhang, H.; Liu, P.; Shen, Z.; and Pineau, J. 2019b. Multitask metric learning: Theory and algorithm. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 3362–3371. PMLR.

Xie, S.; Zheng, Z.; Chen, L.; and Chen, C. 2018. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*, 5423–5432.

You, K.; Long, M.; Cao, Z.; Wang, J.; and Jordan, M. I. 2019. Universal domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2720–2729.

Zhang, Y.; Tang, H.; Jia, K.; and Tan, M. 2019. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5031–5040.

Zhang, Y.; and Yeung, D. Y. 2010. A convex formulation for learning task relationships in multi-task learning. In *26th Conference on Uncertainty in Artificial Intelligence, UAI 2010, Catalina Island, CA, United States, 8-11 July 2010, Code 86680*.

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

Zhao, H.; Des Combes, R. T.; Zhang, K.; and Gordon, G. 2019. On Learning Invariant Representations for Domain Adaptation. In *International Conference on Machine Learning*, 7523–7532.

Zhao, X.; Li, H.; Shen, X.; Liang, X.; and Wu, Y. 2018. A modulation module for multi-task learning with applications in image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 401–416.

Zhou, F.; Jiang, Z.; Shui, C.; Wang, B.; and Chaib-draa, B. 2021a. Domain Generalization via Optimal Transport with Metric Similarity Learning. *Neurocomputing* doi:https://doi.org/10.1016/j.neucom.2020.09.091.

Zhou, F.; Shui, C.; Abbasi, M.; Robitaille, L.-É.; Wang, B.; and Gagné, C. 2021b. Task Similarity Estimation Through Adversarial Multitask Neural Network. *IEEE Transactions on Neural Networks and Learning Systems* 32(2): 466–480. doi:10.1109/TNNLS.2020.3028022.

Zhuang, F.; Li, X.; Jin, X.; Zhang, D.; Qiu, L.; and He, Q. 2017. Semantic feature learning for heterogeneous multitask classification via non-negative matrix factorization. *IEEE transactions on cybernetics* 48(8): 2284–2293.