Learning Transferable Features From Different Domains

Thèse

Fan Zhou

Doctorat en informatique
Philosophiæ doctor (Ph. D.)

Québec, Canada

Learning Transferable Features From Different Domains

Thèse

Fan Zhou

Sous la direction de:

Brahim Chaib-draa, directeur de recherche

Résumé

Les progrès récents en matière d'apprentissage automatique supposent généralement que les données d'apprentissage et de test proviennent de la même distribution de données. Cependant, dans la pratique, les données peuvent être collectées séparément comme des ensembles de données différents.

Apprendre à partir de données provenant de plusieurs domaines sources et les généraliser à un autre domaine est un problème crucial de l'apprentissage automatique. Nous abordons ce type de problème dans le contexte de l'apprentissage par transfert (TL), notamment l'adaptation de domaine (DA), la généralisation de domaine (DG) et l'apprentissage multi-tâches (MTL), et ce dans le but de transférer les caractéristiques invariantes communes à de nouveaux domaines. Nous avons étudié ce type d'apprentissage par transfert sous différents aspects, y compris les problèmes liés au décalage conditionnel dans l'adaptation de domaine, les problèmes de désalignement sémantique et de décalage d'étiquettes dans la généralisation de domaine et l'apprentissage multi-tâches en parvenant à plusieurs résultats.

Concrètement, nous explorons d'abord les problèmes de décalage conditionnel (DA) avec une stratégie d'apprentissage actif pour interroger les instances les plus informatives dans le domaine cible afin de faire migrer le terme de désaccord entre les fonctions d'étiquetage des domaines source et cible. Nous explorons ensuite les similitudes de catégories dans les problèmes liés à la généralisation de domaine (DG) via l'entraînement adversarial basé sur le transport optimal avec un objectif d'apprentissage de similarité métrique afin d'améliorer la correspondance au niveau du domaine et de la classe pour les problèmes DG. Nous étudions ensuite, plus en détail les relations entre les étiquettes et la sémantique dans le MTL, où nous fournissons une compréhension théorique de la manière de contrôler les divergences entre les étiquettes et la distribution sémantique. Enfin, nous étendons l'analyse théorique sur la façon d'exploiter les étiquettes et l'information sémantique dans la généralisation de domaine (DG), en fournissant une première analyse pour comprendre les propriétés de généralisation dans le contrôle des divergences de distribution des étiquettes et de la sémantique.

Pour chaque travail reflété dans cette thèse, nous menons des expériences approfondies afin de démontrer l'efficacité et les objectifs d'apprentissage. Les résultats expérimentaux confirment que nos méthodes parviennent aux performances souhaitées et indiquées par les principes d'analyse et d'apprentissage, ce qui valide les contributions de cette thèse.

Abstract

Recent machine learning progresses usually assume the data for training and testing are from the same data distribution. However, in practice, the data might be gathered separately as different datasets. To learn data from several source domains and generalize to another domain, is a crucial problem in machine learning.

We tackle this kind of problem in the context of *Transfer Learning* (TL), including Domain Adaptation (DA), Domain Generalization (DG) and Multi-task Learning (MTL), with the sake of transferring the common invariant features to new domains. We have investigated this kind of transfer learning method in several different aspects, including the conditional shift problems in domain adaptation, semantic misalignment and label shift problems in domain generalization and multi-task learning problems with several accomplishments.

Concretely, we first explore the conditional shift problems DA with an active learning strategy to query the most informative instances in the target domain to migrate the disagreement term between the source and target domain labelling functions. We then explore the category similarities in the DG problems via the optimal transport-based adversarial training with a metric similarity learning objective to enhance both the domain-level and class-level matching for DG problems. After that, we further investigate the label and semantic relations in MTL, where we provide the first theoretical understanding of how to control the label and semantic distribution divergences. Lastly, we extend the theoretical analysis on how to leverage the label and semantic information in DG, providing the first analysis to understand the generalization properties on controlling the label and semantic distribution divergences.

For each work reflected in this thesis, we also conduct intensive experiments to demonstrate the effectiveness and learning objectives. The experimental results confirm that our methods achieve the desired performance indicated by the analysis and learning principles, which confirms the contributions of this thesis.

Table des matières

Liste des tableaux

Liste des figures

Liste des sigles et abréviations

Abbreviation	Meaning
AcDA	Active Discriminative Domain Adaptaion
AL	Active Learning
CORAL	Correlation Alignment
CV	Computer Vision
DA	Domain Adaptation
DANN	Domain Adversarial Neural Network
DG	Domain Generalization
DL	Deep Learning
DNN	Deep Neural Network
GAN	Generative Adversarial Net
IPM	Integral Probability Metrics
JAN	Joint Adaptation Network
JS	Jensen-Shannon Divergence
KL	Kullback-Leibler Divergence
LHS	Left Hand Side
MAML	Model Agnostic Meta-Learning
ML	Machine Learning
MLP	Multi-layer Perceptron
MMD	Mean Max Discrepancy
MTL	Multi-task Learning
NLP	Natural Learning Processing
NN	Neural Network
OT	Optimal Transport
PCA	Principle Component Analysis
RHKS	Reproduced Hibert Kernel Space
RHS	Right Hand Side
SDA	Stacked Denoising Autoencoders
SMDG	Semantic Domain Generalization
SMTL	Semantic Multi-task Learning
SVM	Support Vector Machine
TL	Transfer Learning
UDA	Unsupervised Domain Adaptation
WADG	Wasserstein Adversarial Domain Generalization
WDGRL	Wasserstein Distance Guided Representation Learning

Liste des notations

Notation	Meaning
$F(\cdot)$	Feature extractor function
$C(\cdot)$	Classifier function
$\phi(\cdot)$	Discriminator function
$\boldsymbol{\theta}_f$	Feature extractor model parameters
$\boldsymbol{\theta}_c$	Classifier model parameters
$\boldsymbol{\theta}_\phi$	Discriminator model parameters
$\mathcal{X}, \mathcal{Y}$	The input space and output space
$\mathbf{x}, y$	An instance and its corresponding label
z	The extracted features ($z = F(\mathbf{x})$)
$\mathcal{D}(\mathbf{x}, y)$	Data distirbution defined on $\mathcal{X} \times \mathcal{Y}$
$\mathcal{D}(\mathbf{x})$	The feature marginal distribution
$\mathcal{D}(y)$	The label marginal distribution
$\mathcal{D}(\mathbf{x}\|y)$	The semantic distribution
$\hat{\mathcal{D}}(\cdot)$	Empirical data distribution
$\mathbb{P}(\cdot)$	Probability Distribution
$\ell(\cdot)$	Error function
$\mathcal{L}(\cdot)$	Learning objective function
$W_1(\mathcal{D}_i, \mathcal{D}_j)$	Wasserstein-1 distance between $\mathcal{D}_i$ and $\mathcal{D}_j$
$D_{JS}(\mathcal{D}_i \Vert \mathcal{D}_j)$	Jensen-Shannon divergence between $\mathcal{D}_i$ and $\mathcal{D}_j$
$d_{TV}(\mathcal{D}_i, \mathcal{D}_j)$	Total variation distance between $\mathcal{D}_i$ and $\mathcal{D}_j$
N_i	Number of instances of a distribution i
$\mathcal{H}$	Hypothesis class
$R(h), \hat{R}(h)$	The expected risk and empirical risk of hypothesis h, respectively
$\mathcal{S}, \mathcal{T}$	The source and target domain, respectively
$\alpha, \beta, \epsilon, \kappa, \lambda, \gamma$	Constants
$\boldsymbol{\alpha}, \boldsymbol{\beta}, \ldots$	Vectors
$\mathbf{A}, \mathbf{B}, \mathbf{C}, \ldots$	Matrices
$I, J, K \ldots$	Upper indices
$i, j, k \ldots$	Running indices
$\Vert A \Vert_1, \Vert A \Vert_2$	L1 and L2 norm of A, respectively
$\langle \cdot, \cdot \rangle$	Inner product operator

To my parents

Remerciements

Looking back to my journey at Laval University, I found myself very lucky to have so many kind professors and friends helping me to conquer the difficulties.

Firstly, I am massively grateful to my supervisor, Prof. Brahim Chaib-draa who has been always accompanying and supervising me. He always encourages me to explore new ideas, to keep trying and never giving up. These years with him and the research experiences in DAMAS lab will be a lifelong treasure for me.

It is highly appreciated that the China Scholarship Council (CSC) supported my studies in Laval University.

Many thanks to Prof. Boyu Wang, Prof. Audrey Durand and Prof. Pascal Germain for their advice and help of my researches. I also would like to thank Prof. Mario Marchand, Prof. Claude-Guy Quimper, Prof. Philippe Giguère, Prof. Pascal Tensson and Prof. Christian Gagné for their wonderful courses and valuable help.

I owe my gratitude to Changjian Shui and Amar Ali-bey, who helped me a lot with valuable suggestions and sharing me with experiences at early stage of my research as well as helping me solve life-related problems.

Besides, my colleagues in Association des Étudiants et Chercheurs Chinois à l'Université Laval, and my friends : Chongyang Wu, Meihong Shi, Nan Jia, Qi Jin, Meng Liu, Yiding Wang, Qi Chen, Ying Zhang, Yang Zhao, Kang Gao, Wenfa Zhang, Lei Yang, Yao Dou, Xu Chen, Sheng Xiang, Jean-Philippe Mercier, Philippe Dandurand, Philippe Babin, Mathieu Alain, Dominic Baril, Alexandre Lemire Paquin, Mathieu Pagé Fortin, were always with me, and we enjoyed wonderful years so that I could feel like home.

Last but not least, I would like to thank my parents and my girl friend who were always with me to share my happiness and to encourage me when I have difficulties. Their love were the most firm backing during my journey to Ph.D.

Avant-propos

The introduction chapter explains the motivation of the research project of this thesis.

Chapter 1 presents the background knowledge on the learning problems explored in this thesis.

Chapter 2 presents the related works and state-of-the-art works, which summarizes the progresses in the general transfer learning, domain adaptation, domain generalization and multi-task learning. We merge the related works parts of the following four articles to show the most related works while also summarize some broader topics as a literature review.

The core parts of this thesis consists of four chapters (chapter 3–6) in the form of articles as follows :

Chapter 3 : Article 1 : Discriminative Active Learning for Domain Adaptation, In *Knowledge-Based Systems* (published, first author).

Chapter 4 : Article 2 : Domain Generalization via Optimal Transport with Metric similarity learning, In *Neurocomputing* (published, first author).

Chapter 5 : Article 3 : Multi-task Learning by Leveraging the Semantic Information, In *Proc. of the Thirty-Fifth AAAI Conference on Artificial Intelligence* (published, first author).

Chapter 6 : Article 4 : On the Value of Label and Semantic Distributions in Domain Generalization, submitted to IEEE Transactions on Neural Networks and Learning Systems, first author.

The conclusion chapter will be a summary of the contributions of this thesis and provide some further discussions for the future researches.

Introduction

In this chapter, we provide a brief overview of our Ph.D. researches. We first introduce the motivations of our research and then give a high-level discussion on our work. We then provide the organization of our thesis to show the contents of each chapter.

Recent years have witnessed the rapid developments of machine learning and deep learning with impressive performances in different application areas including computer vision (He et al., 2016a; Strubell et al., 2020; Høye et al., 2021; Hassaballah and Awad, 2020), natural language processing (Young et al., 2018; Otter et al., 2020) and medical applications (Wang et al., 2021a; Zou et al., 2020) etc. The performance achieved by deep models with end-to-end training has largely outperformed the traditional hand-engineered methods.

Although the deep learning-based approaches have shown improved performances, there still exists some open problems. In this thesis, we aim to figure out how to learn a model by transferring the knowledge from different data distributions to learn from fewer data. We can introduce the problems we tackled in the next section.

Problem Statement

Most of the modern deep learning models usually require a large amount of training data (Bhavsar and Ganatra, 2012). When designing a learning system to have a good generalization ability, we should usually provide it with a large dataset. In practice, however, acquiring labelled data could be highly prohibitive, *e.g.*, when classifying multiple objects in an image (Long et al., 2017a), when analyzing patient data in healthcare data analysis (Wang and Pineau, 2015), or when modelling users' products preferences (Murugesan and Carbonell, 2017). Data hungry has become a long-term problem for deep learning (Marcus, 2018).

Besides, most of the progress of recent machine learning, especially the deep learning approaches, are typically based on the traditional learning paradigm, which assumes that datasets, on which the hypothesis trained and tested, are *i.i.d.* from the same distribution. However, in many learning scenarios, such assumptions may not hold. For example, appearance shifts caused by illumination, seasonal, and weather changes are hard to tackle for learning-based vision systems (Kawano and Yanai, 2014). If a vision system was trained on one dataset (source

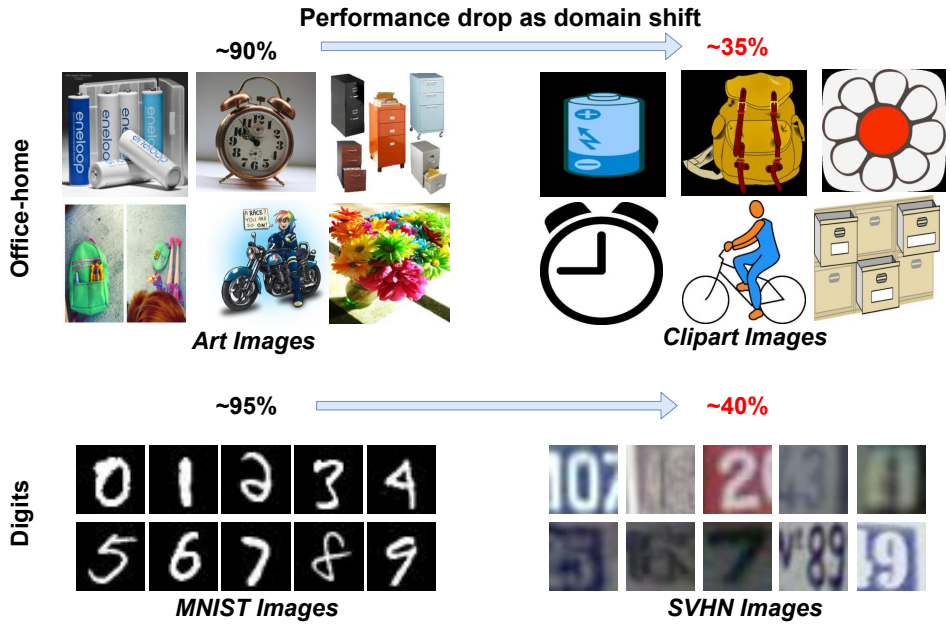


FIGURE 0.1 – Performance drop as domain shift. Typically a model can have high prediction accuracy on a single domain. However, if we deploy the model directly on the target domain, we could observe an obvious performance drop-off due to the domain shift. Data images captured from Office-home (Venkateswara et al., 2017), MNIST (LeCun et al., 1998) as well as SVHN (Netzer et al., 2011) dataset.

domain) but then tested on another (target domain), the performance may diverge (Li et al., 2016a). In many learning problems, we are not able to expect to have access to a large-scaled dataset with plenty of labelled data. If we train the model on one dataset while training on another, the learning performance may drop rapidly. This kind of image changes in terms of style, color or illumination etc., is also known as *domain shift* as discussed in (Ganin et al., 2016) and references therein. We illustrate in Fig. 0.1 to show that a model trained on one dataset can drop off rapidly when tested on another. A general case in many practical scenarios is that we have multiple datasets collected from different positions or angles using different cameras or sensors, leading the data to have similar but different illuminations, styles and poses etc., and our goal will be how to build a model that learns from those datasets.

Motivations

Human beings can learn by knowledge transfer. For example, consider a navigation task in a maze environment using a visual recognition model illustrated in Fig 0.2 an agent is trying to locate and catch the fruit near to it while it suffers from learning the knowledge due to the color and illumination changes in the maze. On the contrary, we humans can easily handle this kind of problem to identify the fruit's location despite the color or position changes. Besides,

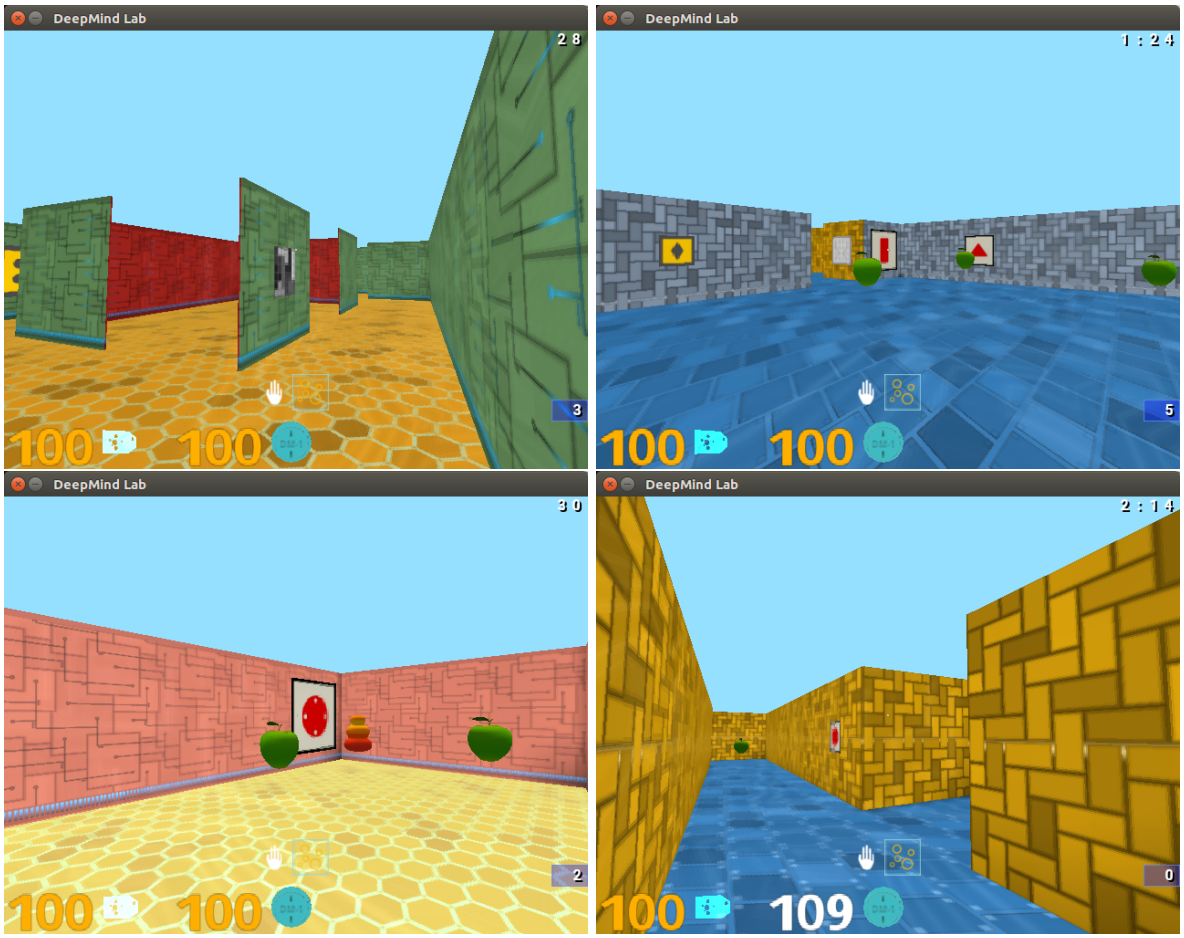


FIGURE 0.2 – Example of visual recognition tasks in a maze environment. Images captured from DeepMind Lab (Beattie et al., 2016) simulation environment.

in daily life visual recognition, humans can easily identify the dogs from images no matter the color, illumination or sitting pose. However, this kind of domain shift (*e.g.*, whether a cat with orange color is standing on a bed or a cat with black color is lying on the ground) would hinder the performances of a learning algorithm. To achieve a successful knowledge transfer, we need to design an algorithm that can handle the data distribution shift.

Many previous approaches have been proposed to solve this kind of domain shift problem in the context of domain adaptation, domain generalization and multi-task learning frameworks. In terms of methodologies, a large number of methods fall into the method of domain-invariant learning, which is typically achieved by the distribution alignment method using statistical distance minimization or the adversarial training methods (Ganin et al., 2016).

Although this kind of invariant feature learning method has shown improved performances in transferring the knowledge for predicting on the new dataset, there are still many open issues that can prevent the model from a successful knowledge transfer. In order to find out the

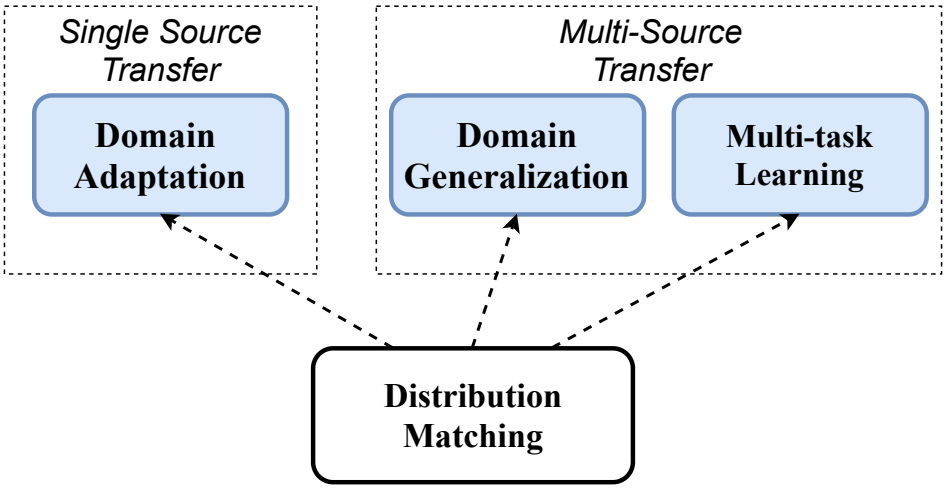


FIGURE 0.3 – A brief structure of our work. We have conduct the distribution matching methods to explore both the single source transfer problems (domain adaptation) as well as the multi-source transfer problems (domain generalization and multi-task learning).

guarantees for a good transfer learning algorithm, we explore in this thesis the transferable feature from different domains with several accomplished works. We illustrate a brief structure of our research in Fig. 0.3. The core methodology of our work is to match the data distributions so that we can align the features from all the domains (will be introduced in section 1.5). In the problem setting aspects, we have explored both the single source transfer (domain adaptation) and the multi-source transfer (domain generalization and multi-task learning) problems.

Thesis Organizations

With our work and accomplishments, the rest parts of the thesis could be summarized as follows,

— In Chapter 1, we introduce the background knowledge of general transfer learning with specific attention to domain adaptation, domain generalization and multi-task learning.

— In Chapter 2, we summarize the related works that mostly inspire our work and a state-of-the-art literature review to show the progresses in this area.

— In Chapter 3, we introduce our work (Zhou et al., 2021c) in which we focus on the conditional shift problems in DA via querying target instances with uncertainty and diversity criteria by exploiting the optimal transport and active learning methodologies.

— In Chapter 4, we introduce our work (Zhou et al., 2021b) where we tackle the domain generalization problems via exploring the feature level alignment using optimal transport with Wasserstein adversarial training, as well as the exploring the category level alignment with metric similarity learning.

— In Chapter 5, we leverage the label and semantic distributions to handle the semantic misalignment and label shift problems in MTL problems. We provide in (Zhou et al., 2021a) the first theoretical results on matching the semantic distributions for MTL problems.

— In Chapter 6, we further explore the semantic relations in the domain generalization problems and provide the theoretical generalization bounds to better understand the DG problems via controlling the label and semantic distribution divergences.

— In the Conclusion chapter, we conclude our contributions towards the thesis and also discuss the limitations in our works as well as the potential directions for future work.

— In the Appendix chapters, we first provide some basic models and approaches involved in our research, then we provide the proofs of the theoretical results of our work.

Summary

In this chapter, we briefly introduce the motivations of our research work and show the thesis organization. In the next chapter, we will formally define the learning problems involved in our work, including domain adaptation, domain generalization and multi-task learning problems.

Chapitre 1

Background and Preliminaries

In this chapter, we briefly introduce the general knowledge of transfer learning including the single-source transfer (domain adaptation) and multi-source (domain generalization and multi-task) learning paradigms, which are essential for our work.

1.1 Notations and Problem Setup

In this thesis, we focus on how to transfer the knowledge from different domains in the context of Transfer Learning (TL) (Torrey and Shavlik, 2010) including Domain Generalization (DG) (Dou et al., 2019), Domain Adaptation (DA) (Pan et al., 2010) and Multi-task Learning (MTL) (Yang and Gao, 2013) with the sake that a learner could leverage the knowledge learned from one (or many) domain(s) or task(s) to generalize to another different but related domains or tasks. We could first give the definition of the notion of *domain* and *task* that involved in our work,

Definition 1.1. ***Domain*** *A domain $\mathcal{D}$ is defined as a yet unknown distribution over $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$ are the input and output space, respectively.*

Similarly, we can also consider a task as a yet unknown data distribution over the $\mathcal{X} \times \mathcal{Y}$. For the DA and DG problems, we aim to find out the transferable features from the source domain and generalize to the target domain (see section 1.3.1 and section 1.4.1 for detailed problem setup). For the MTL problems, we aim to learn several tasks together to improve the learning model for each task by using the knowledge contained in other tasks (see section 1.4.2 for detailed problem setup).

All these three learning scenarios (DA, DG and MTL) focus on learning and transfer knowledge from other domains. We tackle the transfer learning problems by measuring and matching the data distributions (Ben-David et al., 2006; Ben-David et al., 2010a; Redko et al., 2017; Shen et al., 2018) to handle different learning paradigms with different data distribution settings.

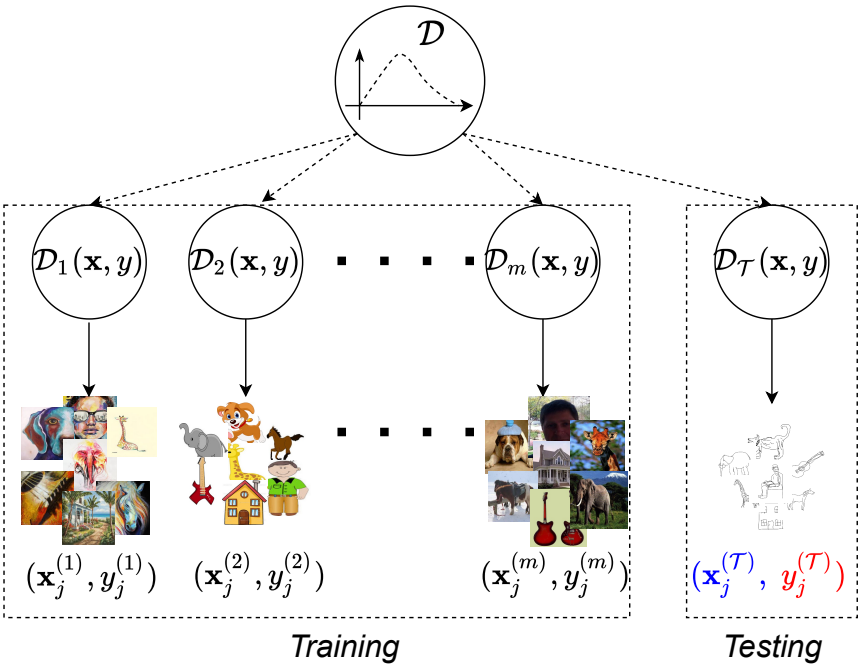


FIGURE 1.1 – Problem set up of the transfer learning paradigms considered in our work. Dataset from different domains are sampled from different data distribution $\mathcal{D}_i(\mathbf{x}, y)$, which are different with each other while are considered generated from a compound probability distribution $\mathcal{D}$. During the training phase, we may have data from multiple sources while during the testing phase, we may deploy the learning algorithm on a target domain which is partially known or totally unknown during the training phase. Depending on the different learning paradigms, we may have different but similar problem setups (see also in Table 1.1).

We present an illustration of general distribution and datasets generation setups in Fig. 1.1. The respective domain shifts are based on different but related data distributions. The learner is trained to generalize to a target domain generated by target distributions. We may have different problem setups *w.r.t.* different learning paradigms.

In this thesis, we have made efforts to the domain adaptation, domain generalization and multi-task learning problems. In a domain adaptation setting, the learner may have access to the instances in the target domain but have no label information (unsupervised DA) or part of the label information (semi-supervised DA) in hands. For the domain generalization tasks, the learner only has access to source domains but have no instances nor labels about the target domain(s). For the multi-task learning tasks, we consider the situation under which the learner is trained on all the source domains with limited data while deployed on the same domains (Long et al., 2017a).

We illustrate a general case of this learning process in Fig. 1.2. Typically, the learner is provided with one or some labelled source domains or tasks and has an objective to generalize its prediction ability to a target domain. In Fig. 1.3, we present a brief visualized comparison on the three learning paradigms involved in our work. Now we start to introduce the problems

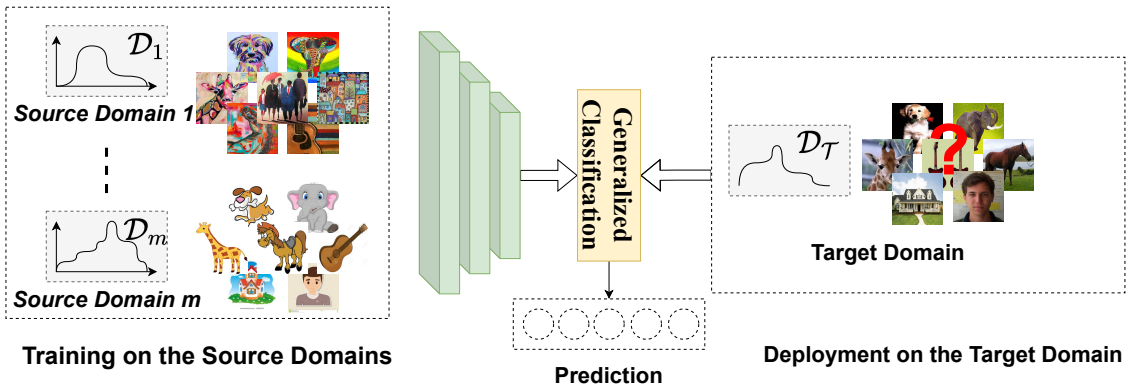


FIGURE 1.2 – General case of paradigms involved in our research : A learner has a set of labelled data from some source domains, and it aims at extracting invariant features across the seen source domains and then learn to generalize to a new domain. Based on the *manifold assumption* (Goldberg et al., 2009), each domain i is supported by distribution $\mathcal{D}_i$. The learner could measure the distance between the source and target domain by measuring the source and target distributions. With the measurement of distribution, the learner could apply some domain confusion training method (*e.g.* adversarial training) to achieve a domain invariant feature space. When a new domain comes, the learner could generalize the knowledge from the invariant feature space to a new domain. In most DA cases, the learner will have access to the instances and labels in the source domain but has only instances with no labels (unsupervised DA) or only parts of labels (semi-supervised DA) in the target domain. For the MTL problems, the learner is trained on all the task distributions with limited data and test on the same distributions. For the DG task, the learner has only access to the source domain features and labels, but has no features nor labels of the target domain.

Learning task	Training			Testing
		Data	Labelled ?	Data
Domain Adaptation (DA)	Source : Target :	$\mathcal{D}_\mathcal{S}(\mathbf{x}, y)$ $\mathcal{D}_\mathcal{T}(\mathbf{x})$	Source : ✓ Target : ✗	$\mathcal{D}_\mathcal{T}(\mathbf{x}, y)$
Domain Generalization (DG)	Source : Target :	$\mathcal{D}_1(\mathbf{x}, y) \dots \mathcal{D}_m(\mathbf{x}, y)$ N/A	Source : ✓ Target : N/A	$\mathcal{D}_\mathcal{T}(\mathbf{x}, y)$ (unseen)
Multi-task Learning (MTL)	Train | | Test	$\mathcal{D}_1(\mathbf{x}, y) \dots \mathcal{D}_T(\mathbf{x}, y)$	Source : ✓ Target : ✓	$\mathcal{D}_1(\mathbf{x}, y) \dots \mathcal{D}_T(\mathbf{x}, y)$

TABLE 1.1 – Different problem setups involved in our work. For the domain adaptation problems, the learner is trained on labelled source domains as well as unlabeled target features. For the domain generalization problems, the learner only has access to the source domain data $\mathcal{D}_1(\mathbf{x}, y) \dots \mathcal{D}_m(\mathbf{x}, y)$ while hope to generalize to the unseen target domain. For the multi-task learning problems, the learner has access to the data from all the tasks $\mathcal{D}_1(\mathbf{x}, y) \dots \mathcal{D}_T(\mathbf{x}, y)$.

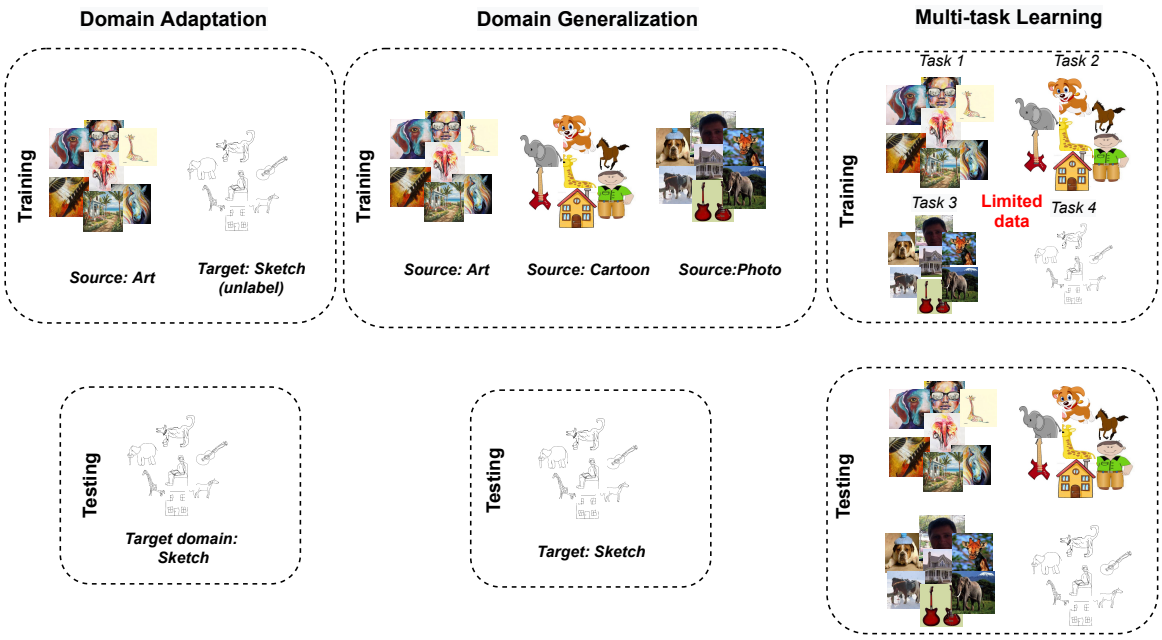


FIGURE 1.3 – Brief comparison of the three learning paradigms involved in this thesis. Dataset images are drawn from PACS dataset (Li et al., 2017b). For DA problems, we learner is trained with a source domain and unlabelled target domain and tested on the target domain. For DG problems, the learner is trained on several source domains and then tested on the unseen target domain. For the MTL tasks, we train the learner on several tasks with limited data while test on the same tasks.

we focused on in the next part.

Before we introduce the core contribution chapters, we first go through the basic knowledge on general transfer learning problems in section 1.2, then the background information of DA, DG and MTL in section 1.3, 1.4.1 and 1.4.2, respectively. After that, we discuss the statistical divergences and distances involved in our work in section 1.5.

1.2 Preliminary of Transfer Learning

As stated before, acquiring labelled data is usually expensive for deep learning. To address that, transfer learning was introduced to leverage the previously learned tasks and transfer the knowledge to another new task. In transfer learning, the goal is to train the hypothesis (learner) from one task (distribution) and then re-use it as a starting point for another task. For domain adaptation, which is a subset area of transfer learning, we aim to transfer the learned knowledge from the source domain to the target domain.

Typically, the learner aims to predict on the target domain $\mathcal{D}_{\mathcal{T}}$, and aims to leverage from the knowledge gained from the source domain. Denote by $\mathcal{D}_{\mathcal{S}}$ and $\mathcal{D}_{\mathcal{T}}$ the source and target domain distributions, we can give the definition of transfer learning as follows,

Definition 1.2. *Transfer Learning (Pan and Yang, 2009) : Transfer Learning aims to improve the performance of predictive function $f_{\mathcal{T}}(\cdot)$ by discovering and transferring latent knowledge from $\mathcal{D}_{\mathcal{S}}$ and $\mathcal{D}_{\mathcal{T}}$ where $\mathcal{D}_{\mathcal{S}} \neq \mathcal{D}_{\mathcal{T}}$. In addition, and in most cases, the size of $\mathcal{D}_{\mathcal{S}}$ is much larger than the size of $\mathcal{D}_{\mathcal{T}}$.*

Typically, there could be different terminology to categorize transfer learning methods. Follow the general transfer learning literature (Pan and Yang, 2009; Weiss et al., 2016; Zhuang et al., 2020) we can categorize the transfer learning methods into three-folds :

— Instance-weighting based methods.

— Mapping-based methods.

— Network-based methods.

Notice that those kind of approaches can also be extended to the other transfer learning topics including domain adaptation, domain generalization or multi-task learning. We first briefly introduce here in the context of general transfer learning.

1.2.1 Instance-weighting based transfer learning

The instance-weighting based transfer learning approach aims at leveraging the knowledge of each instance, which inspires our work (Zhou et al., 2021a,d) as part of the learning algorithm (see the re-weighting scheme presented section 5.3.1 and section 6.3.2, respectively).

Considering a transfer learning scenario that we have labelled source domain data and we only have limited labelled target domain data *i.e.*, large amount of unlabelled data. In this case we can only estimate the marginal distribution $\mathcal{D}(\mathbf{x})$ of the target domain. In this context, we can assume that although the source domain knowledge couldn't be reused directly on the target domain, some parts of the source data could still be reused with few labelled target instances (Zhuang et al., 2020). In order to align the source and target data distributions, the source-domain instances are assigned with weights to reduce the marginal distribution difference. Such weights are usually calculated based on the similarity between source and target instances. This kind of approach needs to assign for each instance, so it is less attractive for modern machine learning approaches as we usually have to deal with large-scaled data. However, the idea of assigning weights to some instances can still inspire our work introduced in Chapter 3, Chapter 5 and Chapter 6, where we leverage the weighting scheme of instances as a part of our methodologies. We will briefly summarize some instance based transfer learning approach in section 2.1.1 and introduce how this kind of weighting scheme used in our work in Chapter 3, Chapter 5 and Chapter 6, respectively.

1.2.2 Mapping-based transfer learning

Mapping based transfer learning approaches were very popular in the early literature of transfer learning as well as domain adaptation as it enables the model to match the data distributions in the feature space. This kind of method refers to map the instances from both the source and target domain into a new data space (typically the reproduced-Hilbert kernel space) with the assumption that "*Although there are differences between the source and target domain, they can be more similar in an elaborate new data space*" (Weiss et al., 2016). Before the era of deep learning, many *principle component analysis* (PCA)-based (Schölkopf et al., 1997) and *transfer component analysis* (Pan et al., 2010) based methods were introduced for such kind of mapping-based transfer learning approaches. The core part of these kinds of approach are based on MMD mapping. For example. Tzeng et al. (2014) implemented MMD in comparing the distributions inside a neural network together with a domain confusion loss to get a domain invariant feature space. Redko et al. (2017) provided a Wasserstein Distance-based approach to align the kernel space for domain adaptation. We will also summarize some other statistical distances in section 1.5.1. Besides, the mapping based approaches were then adopted in the domain adaptation and domain generalization approaches, which we presented in section 2.2.2 and section 2.3.1, respectively.

The early mapping based approaches usually mapped the features into a kernel space, which is somehow limited in especially deep neural network-based approaches. It may not be suitable for more advanced deep learning-based researches (Wang et al., 2018). With the development of neural networks and deep learning, many network-based transfer learning approaches have been proposed. We now introduce that in next section.

1.2.3 Parameter based Transfer Learning

Transfer learning using neural networks (NN) aims to transfer some parts of the NN parameters (learned on a source domain) to a new model (deployed on a target domain). For example, in many domain adaptation approaches, we usually take a pre-trained model (*e.g.* AlexNet by Krizhevsky et al. (2012), VGG-Net by Liu and Deng (2015) or ResNet by He et al. (2016b)) as a feature extractor. Such a pre-trained model contains the pre-trained model parameters (weights and biases) for some general computer vision tasks (usually on ImageNet by Deng et al. (2009)), and we can transfer such parameters for new a new task. This kind of model parameters transfer is also known as fine-tuning (He et al., 2016a). For a large-scale transfer learning problem, the fine-tuning method usually requires a large amount of target labels. Besides, the general fine-tuning method usually needs to re-train the whole model when adapting to the target dataset, and this is somehow inefficient. To this end, one recent remarkable work has been proposed to use such a network-based approach for meta transfer learning where a base learner adopts a pre-trained backbone network as a base feature extractor (Sun et al., 2019). Then, for each task, a specific classifier network will be used for a particular task. Each

classifier is not trained from scratch but uses the network parameter trained on the previous task and then applies a linear transformation based on the trained parameters. By following such a cycle, the model can leverage the learned knowledge using the network parameters.

Now, with the briefly introduced background knowledge on transfer learning, we can introduce the preliminaries of domain adaptation, domain generalization and multi-task learning in the following parts.

1.3 Preliminary of Domain Adaptation

In this section, we present a general introduction to domain adaptation. After that, we show the preliminaries on multi-source transfer learning, including domain generalization and multi-task learning, in section 1.4.

1.3.1 Background of Domain Adaptation

As stated in the previous sections, in general learning paradigms, we usually have a basic assumption that all the training and testing data come from the same distribution. However, such an assumption may not hold if the datasets on which the learning algorithm was trained are different from the one it was deployed. If the testing set and training set are from different distributions, the test performance may diverge. Fig. 1.4 explains a binary classification task that one usually can easily learn a classifier for the given domain (left of Fig. 1.4). What if we have a new classification task for a new domain (middle of Fig. 1.4) ? Could we still use the classifier trained on the source domain ? We can see from the right part of Fig. 1.4 that if we directly deploy the classifier trained on the source domain to a new target domain, this could fail. Then, one question came to us that could we train a classifier that can work on both the source and target domains ? This problem has been widely investigated in the context of Domain Adaptation (DA) (Pan et al., 2010). Typically in DA, we aim to train a discriminative classifier that can handle shift between training and test domains (distributions). During training, the learner takes lots of *labelled* data in the *source* domain (*e.g.* synthetic images) in hand. Then, it learns to predict *unlabelled* data in the *target* domain (*e.g.* real images). The goal is described as training the hypothesis (typically Neural Network) on source domain and can also perform well (with a high accuracy) on the target domain (Ben-David et al., 2010a).

We follow (Ben-David et al., 2010a; Redko et al., 2019) to give a formal definition of domain adaptation. Consider classification tasks where $\mathcal{X}$ is the input space and $\mathcal{Y} = \{0, 1, \ldots, K-1\}$ is the set of K possible labels. Denote by $\mathbf{x}$ and y by the instance feature and the corresponding labels, and $\mathcal{X} \times \mathcal{Y}$ by the input feature and label space, respectively. There are two different (but related) distributions over input-output space $\mathcal{X} \times \mathcal{Y}$, called the source domain $\mathcal{D}_{\mathcal{S}}$ and the target domain $\mathcal{D}_{\mathcal{T}}$. An *unsupervised domain adaptation* learning algorithm tries to learn on labeled source samples drawn *i.i.d.* from $\mathcal{D}_{\mathcal{S}}(\mathbf{x}, y)$, and an unlabeled target samples drawn *i.i.d.*

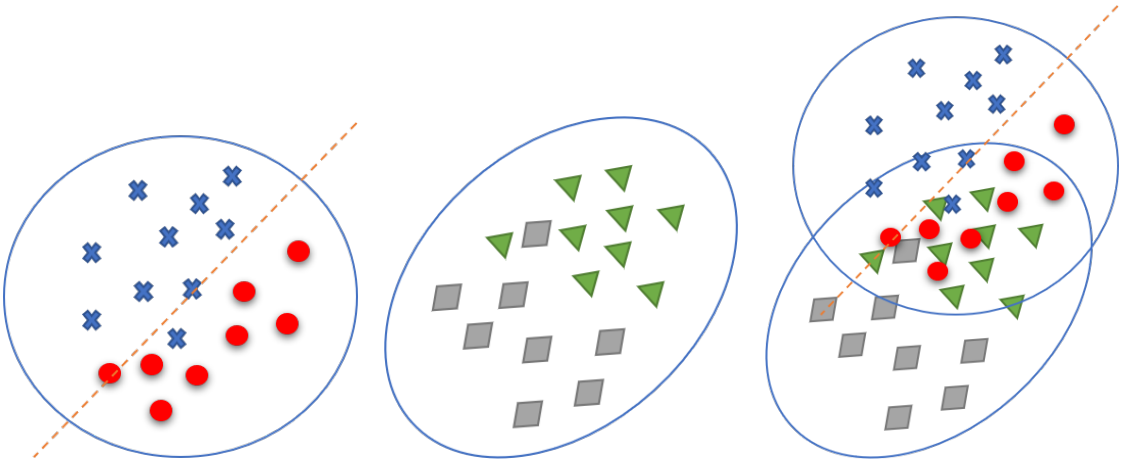


FIGURE 1.4 – For one single source domain, we can easily train a classifier (left), but for a new domain (middle) if we just adopt the classifier trained on the source domain, then the classification process fails (right).

from $\mathcal{D}_{\mathcal{T}}(\mathbf{x})$, where $\mathcal{D}_{\mathcal{T}}(\mathbf{x})$ is the marginal distribution of $\mathcal{D}_{\mathcal{T}}$ over $\mathcal{X}$. In this thesis, we refer the notion *domain adaptation* to the general single source unsupervised domain adaptation problems.

Define the expected source and target risk of $h \in \mathcal{H}$ over $\mathcal{S}$ (respectively, $\mathcal{T}$) as the probabilities that h errs on the entire distribution $\mathcal{D}_{\mathcal{S}}$ (respectively, $\mathcal{D}_{\mathcal{T}}$) : $R_{\mathcal{S}}(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_{\mathcal{S}}}\ell(h(\mathbf{x}, y))$ and $R_{\mathcal{T}}(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_{\mathcal{T}}}\ell(h(\mathbf{x}, y))$, where $\ell(\cdot)$ is the loss function. The goal of DA is to build a classifier $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ training on source domain with a low *target risk* $R_{\mathcal{T}}(h)$. The goal of DA process is to learn $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ training on source domain can also perform well on the target domain (with a low *target risk* $R_{\mathcal{T}}(h)$).

Previously, lots of efforts have been addressed towards domain adaptation, both theoretically (Ben-David et al., 2006; Ben-David et al., 2010a; Redko et al., 2017) and empirically (Ganin et al., 2016; Wen et al., 2020; D'Innocente and Caputo, 2018). There are many kinds of terminology to separate DA methods in different categories, and we categorize recent DA techniques in three primary folds : 1) discrepancy based, 2) adversarial based and 3) reconstruction based domain adaptation method. For the discrepancy based approach, it typically achieves domain invariant features by minimizing the domain shift while fine-tuning the neural network. For adversarial based ones, a domain discriminator is trained using an adversarial objective to learn domain confusion features. For reconstruction based methods, typically, an AutoEncoder is established to reconstruct the data as an auxiliary loss function to ensure the feature extractor (encoder) can have invariant features. In this thesis, we mainly focus on the adversarial based methods, which are widely used in recent domain adaptation researches (Wang et al., 2018).

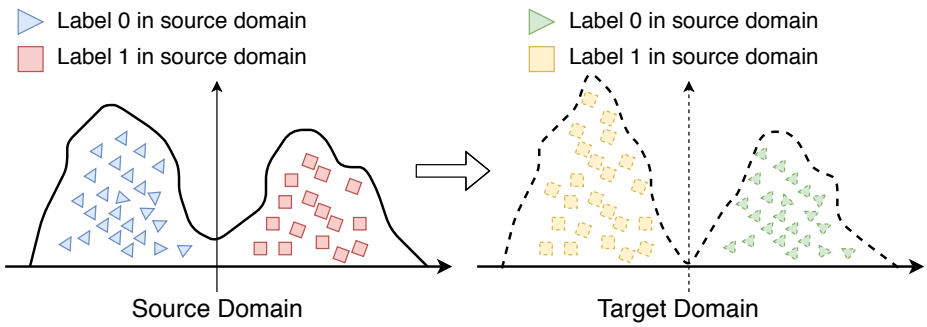


FIGURE 1.5 – The conditional relationship shifts during the adaptation process : The left is source domain distribution with the binary labels and right is the target domain distribution with binary labels. The marginal distribution looks similar while the conditional distribution is different.

1.3.2 Basic Assumptions in Domain Adaptation

In this section, we will discuss the well-known assumptions that stand behind the domain adaptation tasks, namely covariate shift, conditional shift and label shift.

1. Covariate Shift (Zhang et al., 2013) : Most recent DA advancements are mostly based on the basic *Covariate Shift* assumption that the marginal distributions of source and target domain change ($\mathcal{D}_{\mathcal{S}}(\mathbf{x}) \neq \mathcal{D}_{\mathcal{T}}(\mathbf{x})$) while the conditional distribution (predictive relation) is preserved ($\mathcal{D}_{\mathcal{S}}(y|\mathbf{x}) = \mathcal{D}_{\mathcal{T}}(y|\mathbf{x})$) during the adaptation process.

2. Conditional Shift (Zhao et al., 2019a) : As stated before, most of the previous domain adaptation process assumes that the conditional probability relations remain unchanged during the adaptation process. Fig. 1.5 illustrates the conditional shift problem : the marginal distribution of the source and target domain are similar, but the conditional distribution is different ($\mathcal{D}_{\mathcal{S}}(y|\mathbf{x}) \neq \mathcal{D}_{\mathcal{T}}(y|\mathbf{x})$). For real-world images, typically the conditional shift is minor *i.e.* $\mathcal{D}_{\mathcal{T}}(y|\mathbf{x}) \approx \mathcal{D}_{\mathcal{S}}(y|\mathbf{x})$ but there might be some implicit conditional shifts. For example, hand-written digits from different people may have different labels.

3. Label Shift (Zhang et al., 2013) : In most cases of DA (e.g. Ganin et al., 2016; Wen et al., 2020), it's assumed that the number of label spaces in the source and target domain is the same ($\mathcal{D}_{\mathcal{S}}(y) = \mathcal{D}_{\mathcal{T}}(y)$). Label shifts refer to the case that the label space of source and target domains are different ($\mathcal{D}_{\mathcal{S}}(y) \neq \mathcal{D}_{\mathcal{T}}(y)$).

1.4 Preliminary of Multi-source Transfer

In this section, we introduce some background knowledge of the multi-source transfer learning paradigms studied in our research, *i.e.*, the domain generalization and multi-task learning (MTL) problems. As aforementioned, these two learning paradigms share a common setting that the data are issued from several related but different distributions, while they differ from

	Domain Generalization	Multi-task Learning
Training	All m source domains : $\mathcal{D}_1, \ldots, \mathcal{D}_m$	All T tasks : $\mathcal{D}_1, \ldots, \mathcal{D}_T$ (Learn each task with limited data)
Testing	Target domain $\mathcal{D}_\mathcal{T}$ (unseen during training)	All T tasks : $\mathcal{D}_1, \ldots, \mathcal{D}_T$
Distribution relation	Uniformly	Weighted relations
Learning objective	Minimize risks on source domains : $\min \sum_{i=1}^{m} R_i(h)$	Minimize risks on all tasks with weighted summation : $\min \boldsymbol{\alpha}_t \sum_{t=1}^{T} R_t(h)$

TABLE 1.2 – Comparison of two multi-source transfer learning problems tackled in our work. For the domain generalization problems, we have no access to the target data during training, so we have no idea about which source domain can contribute more for a successful transfer process, which leads to a uniformly relations for all the source domains during training. For the multi-task learning problems, we have (limited) data from all the tasks, so we can measure the task similarity $\boldsymbol{\alpha}_t$ during the training process, which leads to an optimization of the weighted summation of all task risks.

each other in terms of learning objective and the data availability. We show in Table 1.2 a brief comparison of these two learning paradigms. We then briefly give the preliminaries of these two learning paradigms in the following sections 1.4.1 and 1.4.2, while we will further elaborate the problem settings of domain generalization (DG) in Chapter 4 and 6, as well as the settings of MTL in Chapter 5.

1.4.1 Preliminary on Domain Generalization

As aforementioned, domain generalization (DG) shares some common assumptions with domain adaptation (DA) while we have neither the instances nor the labels in the target domain. We illustrate a general workflow of DG in Fig. 1.6.

Let $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ be a training example drawn from some unknown distribution $\mathcal{D}$, where $\mathbf{x}$ is the data point, and y is its label. A hypothesis is a function $h \in \mathcal{H}$ that maps $\mathcal{X}$ to the set $\mathcal{Y}'$ sometimes different from $\mathcal{Y}$, where $\mathcal{H}$ is a hypothesis class. For a non-negative loss function $\ell : \mathcal{Y}' \times \mathcal{Y} \mapsto \mathbb{R}_+$, we denote by $\ell(h(x), y)$ the loss of hypothesis h at (x, y). Let $S = \{(\mathbf{x}_j, y_j)\}_{j=1}^{N}$ be a set of N training examples drawn independently from $\mathcal{D}$. The empirical loss of h on S and its generalization loss over $\mathcal{D}$ are defined, respectively, by $\hat{R}(h) = \frac{1}{N} \sum_{j=1}^{N} \ell(h(\mathbf{x}_j, y_j))$, and $R(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \ell(h(\mathbf{x}), y)$.

In the context of DG, we are given m source tasks $\{S_i\}_{i=1}^{m}$, where $S_i = \{(\mathbf{x}_j^{(i)}, y_j^{(i)})\}_{j=1}^{N_i}$ is drawn from a distribution $\mathcal{D}_i$. The objective of a DG algorithm is to learn a feature representation

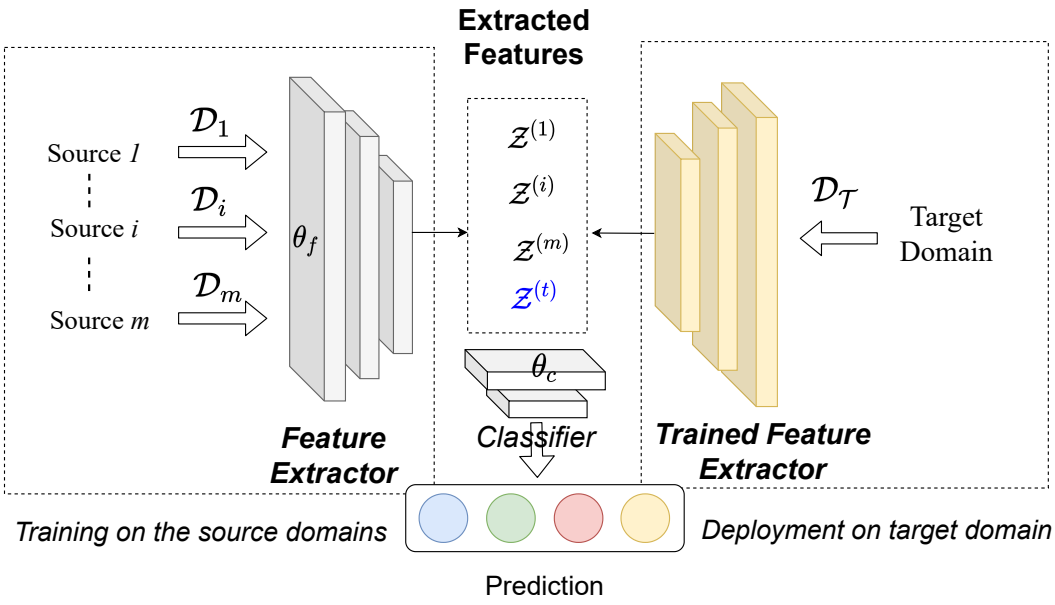


FIGURE 1.6 – General workflow of domain generalization (DG) : The model is trained on several source domains ($\mathcal{D}_1, \mathcal{D}_2, \ldots \mathcal{D}_m$) while deployed on an unseen target domain. During the training phase, the source data are mixed as input and feed into the model, during which both the source feature $\mathcal{D}(\mathbf{x})$ and label $\mathcal{D}(y)$ are available to the learner. At the deployment phase, the model is frozen and test on the target domain $\mathcal{D}_\mathcal{T}$, which is not accessible to the model during the training phase.

that extracts the knowledge that can be shared across all the known source domains so that it can also generalize well to an unseen target domain distribution $\mathcal{D}_\mathcal{T}$.

1.4.2 Preliminary on Multi-task Learning

Another multi-source transfer learning paradigm we have investigated is the multi-task learning problems. MTL aims to learn multiple tasks simultaneously and improves learning efficiency by leveraging the shared features across tasks. It has been prevalent in lots of recent machine learning topics (Li et al., 2014; Wang et al., 2016; Teh et al., 2017). Typically, there could be several kinds of terminology of MTL. We illustrate a general scheme of the MTL approaches in Fig. 1.7. Our work (Zhou et al., 2021a) is most similar to the learning framework in Fig. 1.7(a), where we have multiple tasks as input and hope to make predictions on a same set of output. Then, in Fig. 1.7(b) we show a learning scenario where we have single-input task and hope to make multiple output predictions. This kind of framework is widely used in multi-label learning (Gibson et al., 2019; He et al., 2020) or multi-objective (Zhou et al., 2017; Sener and Koltun, 2018) learning problems. Finally, Fig. 1.7(c) refers to the learning scenario that multiple tasks data are mapped to the same multiple sets of output target. This kind of framework is widely used in the multi-view learning (Jin et al., 2014; Zhang et al., 2019b) and multi-modality learning (Liu et al., 2018; Zhou et al., 2020a) problems. In this thesis, we will

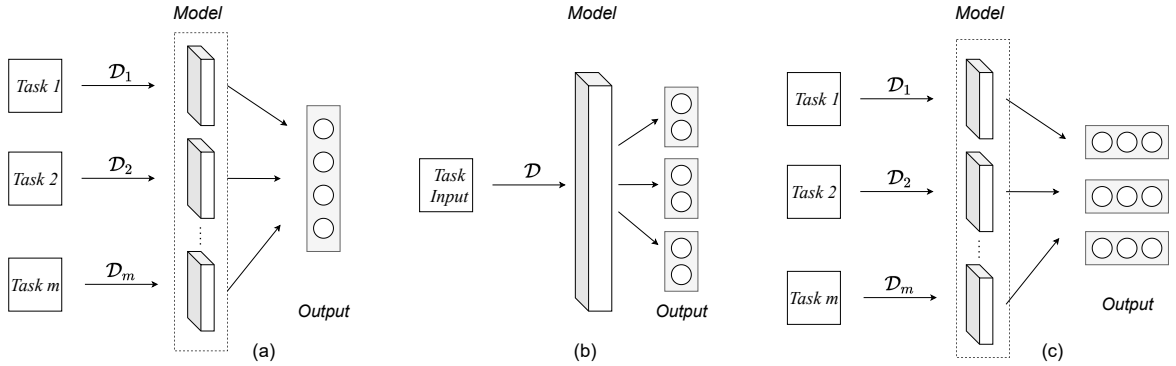


FIGURE 1.7 – Different framework terminology of MTL. (a) : MTL framework where we have multiple input task data and hope to map to a same set of output. Our work (Zhou et al., 2021a) is tackled in this kind of framework. (b) : MTL framework where we have single-input task and hope to make multiple output predictions. (c) MTL framework where we have multiple sets of input task data and hope to map the data to the same multiple sets of output target data. Figure referred from Thung and Wee (2018).

mainly focus on the learning framework in Fig. 1.7(a), which is tackled in our work (Zhou et al., 2021a).

Fig. 1.8 shows the typical model structure of the MTL problems in our work. 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 $f_t : \mathcal{X} \to \mathcal{Y}$ 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 :

$$\arg\min_{h \in \mathcal{H}} \frac{1}{T} \sum_{i=1}^{T} R_t(h_t), \tag{1.1}$$

where $R_i(h_i) \equiv R_i(h_i, f_i) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_i} \ell(h_i(\mathbf{x}), f_i(\mathbf{x}))$ 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 (risk) *w.r.t.* the hypothesis h for task i could be defined as

$$\hat{R}_{\boldsymbol{\alpha}_i}(h) = \sum_{j=1}^{T} \boldsymbol{\alpha}_{i,j} \hat{R}_j(h), \tag{1.2}$$

where $\hat{R}_i(h) = \frac{1}{m_i} \sum_{j=1}^{m_i} \ell(h(\mathbf{x}_j), y_j)$ is the average empirical risk for task i. The learning goal is to learn multiple related tasks simultaneously by minimizing the averaged risk over all the tasks, and improves learning efficiency by leveraging the shared features across tasks.

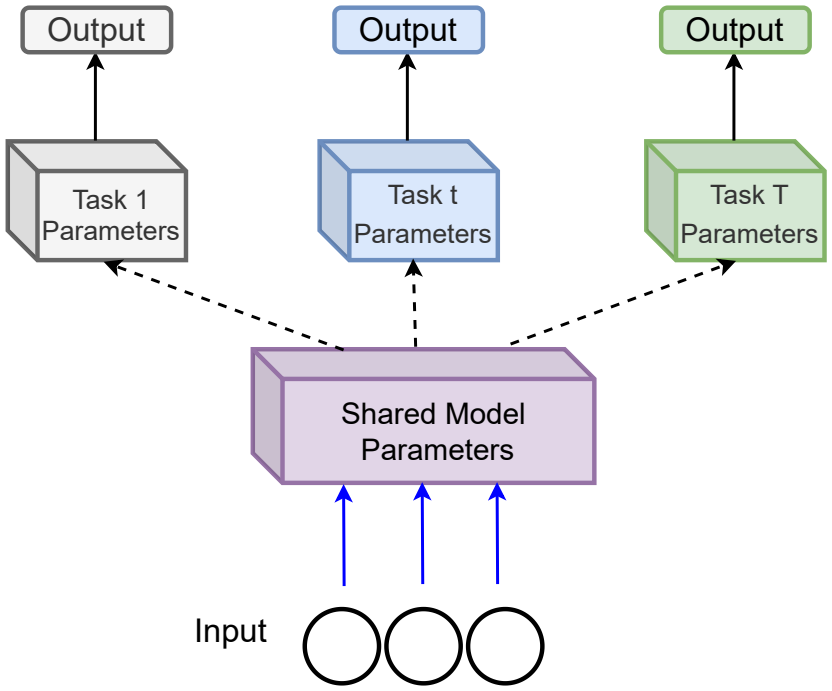


FIGURE 1.8 – Typical model structure involved in our work. The model shares some parts of parameters $\boldsymbol{\theta}_{\text{share}}$ while for a certain task t we construct the specific classifier $\boldsymbol{\theta}_t$ for it. For training, we mixed all the instances from each task to the shared model. Then, for each specific task, we feed the extracted features to each specific classifier $\boldsymbol{\theta}_t$.

1.5 Distribution Divergences for Similarity Measuring

As previously noticed, the idea of transfer learning is to transfer knowledge from one domain to another. A basic assumption behind this process is that the related domains should not be too far from each other. And if the distance between two domains is arbitrarily large, the transfer process may fail (Ben-David et al., 2010a). Thus, we need to measure distance (divergence) between the different data distributions. In this context, we usually have a generalization bound to measure the target domain error so that we can implement learning techniques to minimize that error bound to achieve successful transfer (Zhuang et al., 2020). Generally, the generalization bound of transferring knowledge from source domain $\mathcal{D}_{\mathcal{S}}$ to a target domain $\mathcal{D}_{\mathcal{T}}$ could be summarized with the following form

$$R_{\mathcal{T}}(h) \leq R_S + D_{div}(\mathcal{D}_{\mathcal{S}} \| \mathcal{D}_{\mathcal{T}}) + \lambda \tag{1.3}$$

where $R_{\mathcal{T}}$ is the risk on the target domain, $R_{\mathcal{S}}$ is the risk on the source domain and $D_{div}(\mathcal{D}_{\mathcal{S}} \| \mathcal{D}_{\mathcal{T}})$ represents some divergences or metrics (*e.g.*, Wasserstein distance (Courty et al., 2016), Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) or $\mathcal{H}$-divergence (Ben-David et al., 2010a) etc.) to measure the distance between the source and target domain. The last term λ represents the discrepancy between source and target labelling function. Depending on different divergence or metric, Eq. 1.3 could have different forms, which will be further elaborated in

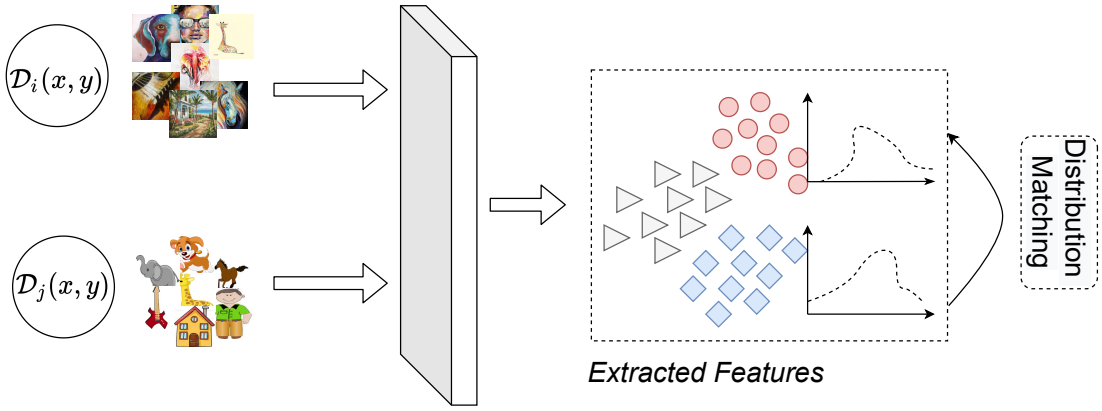


FIGURE 1.9 – Typical distribution matching process : typically the model extracts the feature from different data distributions and get the extracted features. The distribution matching process is then applied in the extracted feature space by matching the extracted features.

section 1.5.2. In general, the divergence or metric between domains can be estimated from the input data. To minimize this bound we can minimizing the source error and distance between source and target domains. Therefore, we can learn the domain-invariant features by matching the distributions. We present an illustration on the general distribution matching process in Fig. 1.9. Typically, we have a feature model to extract the features from different domains. Then, in the feature space, we can match the feature distributions by minimizing the distance or adversarial training etc. so that the model can learn a domain-invariant feature representation which can be generalized to a new domain.

Generally, a valid statistical divergence $D_{div}(\cdot\|\cdot)$ satisfy the following properties :

— The divergence between distributions $\mathcal{D}_i$ and $\mathcal{D}_j$ is non-negative : $D_{div}(\mathcal{D}_i||\mathcal{D}_j) \geq 0$

— The divergence attains zero **iff.** the two distributions are identical to each other, *i.e.*, $\mathcal{D}_i = \mathcal{D}_j$.

Depending on specific conditions, we may have different family of divergences. In this section, we briefly summarize some popular statistical divergences in the literature and then show the ones most related to our work.

1.5.1 Brief Summary of Different Distribution Divergences

We briefly summarize the Integral Probability Metrics, f-Divergence and Hypothesis-based Divergences.

Integral Probability Metrics

A family of popular distribution distance measure is the *Integral Probability Metrics* (IPM) (Sriperumbudur et al., 2009) : For two distributions $\mathcal{D}_i$ and $\mathcal{D}_j$, and for a real valued function

Divergence	$f(x)$	$D_f(\mathcal{D}_i \| \mathcal{D}_j)$
Kullback-Leibler (KL) Divergence	$x\log(x)$	$D_{KL}(\mathcal{D}_i \| \mathcal{D}_j) = \mathbb{E}_{p_i}[\log(\frac{p_i(x)}{p_j(x)})]$
Total Variation Distance	$\frac{1}{2}\lvert x-1\rvert$	$\frac{1}{2}\int_x \lvert p_i(x) - p_j(x)\rvert$
Jensen-Shannon (J-S) Divergence	$x\log(\frac{2x}{x+1} + \log(\frac{2}{x+1}))$	$D_{KL}(\mathcal{D}_i \| \frac{\mathcal{D}_i+\mathcal{D}_j}{2}) + D_{KL}(\mathcal{D}_j \| \frac{\mathcal{D}_i+\mathcal{D}_j}{2})$

TABLE 1.3 – The f-divergences used in our work.

family $f \in \mathcal{F}$, then the IPM metric could be computed as,

$$\mathcal{IPM}(\mathcal{D}_i, \mathcal{D}_j) = \sup_{f \in \mathcal{F}} |\mathbb{E}_{\mathcal{D}_i} f(x) - \mathbb{E}_{\mathcal{D}_j} f(x)| \tag{1.4}$$

In the literature, the following divergences are used,

— Wasserstein Distance (Villani, 2009) : Let $f(x)$ be 1-Lipschitz such that $f(x) - f(x') \leq ||x - x'||_2$ and $\mathcal{F} := \{f : ||f||_L \leq 1\}$, then we have the Wasserstein-1 distance :

$$W_1(\mathcal{D}_i, \mathcal{D}_j) = \sup_{||f||_L < 1} \mathbb{E}_{x \in \mathcal{D}_i} f(x) - \mathbb{E}_{x' \in \mathcal{D}_j} f(x') \tag{1.5}$$

— Total Variation Distance [1] (Lin, 1991) : Let $\sup_x |f(x)| \leq 1$, then we have the Total Variation Distance :

$$d_{TV}(\mathcal{D}_i, \mathcal{D}_j) = \frac{1}{2}|\mathcal{D}_i - \mathcal{D}_j| \tag{1.6}$$

— Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) : Denote $\mathcal{H}$ by a reproducing kernel Hilbert space and let $\mathcal{F} = \{f : \|f\|_{\mathcal{H}} \leq 1\}$, then the $\mathcal{IPM}$ recovers the MMD :

$$D_{MMD}(\mathcal{D}_S, \mathcal{D}_T) = \| \frac{1}{|X_S|} \sum_{x_i^s \in X_S \sim \mathcal{D}_S} \psi(x_i^s) - \frac{1}{|X_T|} \sum_{x_i^t \in X_T \sim \mathcal{D}_T} \psi(x_i^t) \| \tag{1.7}$$

The Information f-Divergence

f-divergence family (Sason and Verdú, 2016) is characterized by a *convex* function f *s.t.* $f(1) = 0$, then the f-divergence between two distributions $\mathcal{D}_i$ and $\mathcal{D}_j$, with which the probability density functions $p_i(x)$ and $p_j(x)$, respectively, can be defined as,

$$D_f(\mathcal{D}_i \| \mathcal{D}_j) = \int_x f(\frac{p_i(x)}{p_j(x)}) p_j(x) dx \tag{1.8}$$

To define different function $f(x)$, we can have different kind of divergences widely used in the literature. We summarize some f-divergence used in our work (Chapter 5 and Chapter 6) in Table 1.3.

1. It can also be viewed as a kind of f-Divergence.

One property of f-divergence is that it can be efficiently estimated via its variations form (Wan et al., 2020) by define a convex conjugate of f as,

$$f^{\star}(x') = \sup_{x}[xx' - f(x)] \tag{1.9}$$

Let $d(x)$ be a function such that $d : \mathcal{X} \to \mathbb{R}$, then the variational term of f-divergence is estimated by

$$D_f(\mathcal{D}_i \| \mathcal{D}_j) = \sup_{d:\mathcal{X}\to\mathbb{R}} \mathbb{E}_{\mathcal{D}_i}[d(x)] - \mathbb{E}_{\mathcal{D}_j}[f'(d(x))] \tag{1.10}$$

Then, by adopting different $f(x)$, we can have different dual term of the f-divergence. For example, let $f(x) = \frac{1}{2}|x-1|$, we have the total variation distance as,

$$d_{TV}(\mathcal{D}_i, \mathcal{D}_j) = \sup_{d:\mathcal{X}\to\mathbb{R}} \mathbb{E}_{\mathcal{D}_i}[d(x)] - \mathbb{E}_{\mathcal{D}_j}[d(x)] \tag{1.11}$$

To solve Eq. 1.11, we can introduce a critic function $d(x)$ to maximize the dual term of f-divergence. In the deep learning setting, this then inspires us with an adversarial training scheme. We will further elaborate this in Chapter 3, Chapter 4 as well as Appendix A.1.

Hypothesis-based Divergences

Another kind of divergence to measure the divergences between different distributions is the hypothesis-based divergences. For example, $\mathcal{H}$-divergence is a widely used measure for domain adaptation (Ben-David et al., 2010a; Zhao et al., 2019a) and multi-task learning approaches (Mao et al., 2020).

Definition 1.3. *$\mathcal{H}$ **divergence** (Ben-David et al., 2010a) : Given two domain distributions $\mathcal{D}_i(x)$ and $\mathcal{D}_j(x)$ over $\mathcal{X}$, and a hypothesis class $\mathcal{H}$, the $\mathcal{H}$ divergence between $\mathcal{D}_i(x)$ and $\mathcal{D}_j(x)$ is*

$$d_{\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j) = 2 \sup_{h\in\mathcal{H}} |\mathbb{P}_{\mathbf{x}\sim\mathcal{D}_i}[h(\mathbf{x}) = 1] - \mathbb{P}_{\mathbf{x}\sim\mathcal{D}_j}[h(\mathbf{x}) = 1]| \tag{1.12}$$

From Eq. 1.12, we note that $\mathcal{H}$-divergence depends on the richness of the hypothesis class $\mathcal{H}$ (Shalev-Shwartz and Ben-David, 2014) to discriminate the samples from $\mathcal{D}_i$ and $\mathcal{D}_j$. The $\mathcal{H}$-divergence then inspires many theoretical analysis in the transfer learning methodologies in domain adaptation (Ganin et al., 2016; Zhang et al., 2019c; Saito et al., 2018) and multi-task learning (Mao et al., 2020). Apart from the $\mathcal{H}$-divergence, another distance adopted in the literature is the *Discrepancy Distance*,

Definition 1.4. ***Discrepancy Distance** (Mansour et al., 2009a) : Let $\mathcal{H}$ be a hypothesis set of functions $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$, and denote $\ell : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$ by a loss function over $\mathcal{Y}$. The discrepancy distance $disc_L$ between two distributions $\mathcal{D}_i$ and $\mathcal{D}_j$ is computed by,*

$$disc_L(\mathcal{D}_i, \mathcal{D}_j) = \max_{h,h'\in\mathcal{H}} |\mathbb{E}_{x\in\mathcal{D}_i}\ell(h(\mathbf{x}), h'(\mathbf{x})) - \mathbb{E}_{\mathbf{x}\in\mathcal{D}_j}\ell(h(\mathbf{x}), h'(x))| \tag{1.13}$$

The discrepancy distance then inspires many transfer learning approaches (Mansour et al., 2009a; Kuroki et al., 2019; de Mathelin et al., 2021). Unlike the $\mathcal{H}$-divergence, The discrepancy distance is suitable to a more general scenarios, including regression, and general loss functions.

In chapter 3 and 4, we exploit the Wasserstein-1 distance since it could constrain labelled source samples of the same class to remain close during the transportation process (Courty et al., 2016), which quietly fit to the problem setting in those two chapters. Moreover, some information theoretical metrics such as KL divergence is not capable to measure the inherent geometric relations among the different domains (Arjovsky et al., 2017). In contrast, the Wasserstein distance can exactly measure their corresponding geometry properties. Besides, compared with (Ben-David et al., 2010a), Wasserstein distance has gradient property (Arjovsky et al., 2017) and the promising generalization bound (Redko et al., 2017). The empirical studies (Gulrajani et al., 2017; Shen et al., 2018) also demonstrated the effectiveness of the Wasserstein adversarial training for extracting the invariant features to align the marginal distributions of different domains (see also in Appendix A). In Chapter 5 and Chapter 6, we exploit the family of f-divergence especially with the Jensen-Shannon Divergence and Total Variation Distance, which were widely adopted in the GAN style adversarial training.

The MMD metric is usually implemented in kernel space, which is not sufficient for large-scaled applications, and KL divergence is unbounded, which is also insufficient for a successful measuring domain shift (Zhao et al., 2019a). Note that the Wasserstein-1 distance introduced in Eq. 1.5 is not the original form of the optimal transport problems but the *Kantorovich-Rubinstein duality* form, which is introduced in Appendix A.2.1.

With the background knowledge of the statistical distances in hand, we can introduce the basic generalization bound explored in the literature that related to our work in the following section 1.5.2.

1.5.2 Basic Generalization Bounds

The essential component of recent transfer learning, especially for the domain adaptation (DA) approaches aims to measure the divergence between the source and target domain. Previous works have proposed hypothesis based metrics such as $\mathcal{H}$-divergence (Ben-David et al., 2010a), distribution discrepancy (Cortes et al., 2019), margin disparity discrepancy (Zhang et al., 2019c) and $\mathcal{X}$-discrepancy (Kuroki et al., 2019), which only focus on the covariate marginal distribution similarity.

An alternative way is to directly derive the statistical divergence based theory : information theoretic Rényi divergence (Mansour et al., 2009c; Germain et al., 2016) and Wasserstein distance (Redko et al., 2017), which can capture the joint distribution similarity. Those previous works have reported lots of concentration inequalities in domain adaptation-related researches. Such concentration bounds play an essential role in recent DA related works. In the following

section, we introduce the representative ones using $\mathcal{H}$-divergence, which then inspires to many adversarial training (Ganin et al., 2016) based approaches, and the generalization bound using Wasserstein distance, which inspires our work in Chapter 3 and Chapter 4.

Generalization Bound over Source and Target Distribution with $\mathcal{H}$-divergence

$\mathcal{H}$-divergence was firstly analyzed in Ben-David et al. (2006) for analyzing the concentration equality between source and target domains. Then, Ben-David et al. (2010a) provided the theoretical framework on learning from different domains using $\mathcal{H}$ divergence.

Theorem 1.1. *(Ben-David et al., 2010a) Let $\mathcal{H}$ be the hypothesis space with VC dimension d. For two distribution $\mathcal{D}_{\mathcal{S}}$ and $\mathcal{D}_{\mathcal{T}}$. For every $h \in \mathcal{H}$:*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + d_{\mathcal{H}}(\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}) + \lambda_{s,t} \tag{1.14}$$

where $\lambda_{s,t} = \inf_{h \in \mathcal{H}}\{R_{\mathcal{S}}(h) + R_{\mathcal{T}}(h)\}$ and $R_{\mathcal{S}}(h), R_{\mathcal{T}}(h)$ are the source and target risks.

Here the term $\lambda_{s,t} = \inf_{h \in \mathcal{H}}\{R_{\mathcal{S}}(h) + R_{\mathcal{T}}(h)\}$ is not observable in the unsupervised domain adaptation tasks since the learner has no access to the labels in the target domain. Zhao et al. (2019a) analyzed the conditional shift problem using the $\hat{\mathcal{H}}$ divergence and show that risk on the target task can be decided by the source risk, the marginal distribution divergence, and disagreement between the *two labelling distribution* :

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + d_{\hat{\mathcal{H}}}(\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}) + \min\{\mathbb{E}_{\mathcal{D}_{\mathcal{S}}}\left[|f_{\mathcal{S}} - f_{\mathcal{T}}|\right], \mathbb{E}_{\mathcal{D}_{\mathcal{T}}}\left[|f_{\mathcal{S}} - f_{\mathcal{T}}|\right]\} \tag{1.15}$$

Here $R_{\mathcal{T}}(h)$, $R_{\mathcal{S}}(h)$ and f refers to target risk, source risk and labeling function, respectively. We can see the term $\min\{\mathbb{E}_{\mathcal{D}_{\mathcal{S}}}\left[|f_{\mathcal{S}} - f_{\mathcal{T}}|\right], \mathbb{E}_{\mathcal{D}_{\mathcal{T}}}\left[|f_{\mathcal{S}} - f_{\mathcal{T}}|\right]\}$ refers to the labelling function and is also an non-observable term in the unsupervised domain adaptation. We further investigate this problem using active learning and the results are illustrated in Chapter 3.

Many DA approaches widely investigated the concentration bound by Ben-David et al. (2010a) reflected by Eq. 1.14 (e.g. Ganin et al., 2016), which works as a principled neural network-based architecture in DA. However, the $\mathcal{H}$-divergence can not be directly minimized, and typically it should be computed by approximation. For example, in the literature, $J - S$ divergence is widely used for approximating the $\mathcal{H}$-divergence (e.g. Pu et al., 2017; Xu et al., 2018). However, such approximations don't have many theoretical insights and is not a good choice.

We adopt the example by Ben-David et al. (2010b) and Shui et al. (2020a) illustrated in Fig. 1.10 to show the difference between $\mathcal{H}$-divergence and the J-S divergence. First fix a small $\xi \in (0, 1)$. Let the target $\mathcal{D}_{\mathcal{T}}(x)$ be the uniform distribution over $\{2k\xi : k \in \mathbb{N}, 2k\xi \leq 1\}$ and let the source distribution $\mathcal{D}_{\mathcal{S}}(x)$ be the uniform distribution over $\{(2k+1)\xi : k \in \mathbb{N}, (2k+1)\xi \leq 1\}$. We can compute $d_{\mathcal{H}}(\mathcal{D}_{\mathcal{T}}, \mathcal{D}_{\mathcal{S}}) = \xi$ while $D_{\mathrm{JS}}(\mathcal{D}_{\mathcal{T}} \| \mathcal{D}_{\mathcal{S}}) = 1$ since the two distributions have *disjoint* support. Then we have $d_{\mathcal{H}}(\mathcal{D}_{\mathcal{T}}, \mathcal{D}_{\mathcal{S}}) \ll D_{\mathrm{JS}}(\mathcal{D}_{\mathcal{T}} \| \mathcal{D}_{\mathcal{S}})$ when $\xi \ll 1$, indicating those

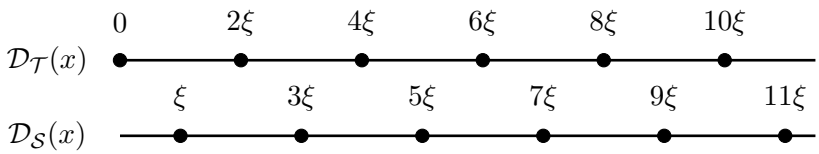


FIGURE 1.10 – Counter Example to show the differences between the J-S divergence and $\mathcal{H}$-divergence. Figure referred from Ben-David et al. (2010b).

metrics can be very different. This aforementioned example indicates the different properties between these two metrics. Since in practice we broadly applied Jensen-Shannon divergence based loss for designing the algorithm (e.g. Li et al., 2018c; Wu et al., 2019; Wen et al., 2020), thus adopting the $\mathcal{H}$-divergence based theory for explaining the theoretical insights generally is not adequate.

Recently, Redko et al. (2017) and Shen et al. (2018) implemented Wasserstein Distance as a measure in adversarial training for DA. We now report the generalization bound *w.r.t* Wasserstein Distance.

Generalization Bound over Source and Target Distribution with Wasserstein Distance

As stated before, Wasserstein Distance, together with Optimal Transport theory, has played an important role in recent computer vision applications (*e.g.* Wasserstein GAN by Arjovsky et al. (2017)). We follow the strategy of Redko et al. (2017) and Shen et al. (2018) to analyze the domain discrepancy with Wasserstein Distance with Lipshciz-continuous functions.

Theorem 1.2. *(Shen et al., 2018) Let $\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}$ be two probability measures, assume $\forall h \in \mathcal{H}$ are* k-Lipschitz continuous *functions and cost function for OT is the Euclidean distance $\rho(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$. Then we have that :*

$$R_{\mathcal{T}}(h, h') \leq R_{\mathcal{S}}(h, h') + 2kW_1(\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}) + \lambda$$

for every hypothesis $h, h' \in \mathcal{H}$ and λ is the combined error of the ideal hypothesis $h^\star$ that minimizes the combined error $R_{\mathcal{S}}(h) + R_{\mathcal{T}}(h)$.

The main idea of the proof is symmetric to Lemma 1 of Redko et al. (2017), while they restricted all the hypothesis within an unit ball in the Reproduced Hilbert Space, following the work of Shen et al. (2018), it can be extended to $k - Lipschiz$ continuous functions. This Wassertein Distance based bound inspires some practical methods and showed a good practical performance, where they proposed *Wasserstein Distance Guided Representation Learning* (WDGRL) algorithm (details are displayed in Appendix A.3). This algorithm is also a fundamental tool of our work in Chapter 3 and 4.

Theoretical Analysis	Type of $\ell_S(h)$	Divergence $D(\mathcal{D}_\mathcal{T}(x)\|\mathcal{D}_\mathcal{S}(x))$	Non-observable term
Ben-David et al. (2010a)	Binary	$\mathcal{H}$-divergence	$\inf_{h\in\mathcal{H}}\{R_S(h)+R_T(h)\}$
Shen et al. (2018)	Binary	Wasserstein-1	$\inf_{h\in\mathcal{H}}\{R_\mathcal{S}(h)+R_\mathcal{T}(h)\}$
Zhao et al. (2019a)	Binary	$\tilde{\mathcal{H}}$-divergence	$\min\{\mathbb{E}_\mathcal{S}[\|f_S-f_T\|],\mathbb{E}_\mathcal{T}[\|f_S-f_T\|]\}$
Peng et al. (2019b)	Binary	Moment Matching Distance	$\min_{h\in\mathcal{H}}\{R_T(h)+R_j(h)\}$

TABLE 1.4 – Brief comparisons of different unsupervised domain adaptation. *Type of* $\ell_S(h)$: the type assumptions on the loss ; $D(\mathcal{D}_\mathcal{T}(x)\|\mathcal{D}_\mathcal{S}(x))$ metrics for measuring marginal distribution similarities ; *Non-observable term* : the underlying assumption for ensuring a success transfer. No matter applying which theoretical divergence, there is always a non-observable term.

Contributions	Problem setup	Source data	Target data	Objective
Zhou et al. (2021c)	Single source Transfer (Domain Adaptation)	$\mathcal{D}_\mathcal{S}(\mathbf{x},y)$	$\mathcal{D}_\mathcal{T}(\mathbf{x})$	1. OT Feature Alignment 2. AL for Conditional Shift
Zhou et al. (2021b)	Multi-source Transfer (Domain Generalization)	$\{\mathcal{D}_i(\mathbf{x},y)\}_{i=1}^m$	N/A	1. OT Feature Alignment 2. Metric Learning Class-level Alignment
Zhou et al. (2021a)	Multi-source Transfer (Multi-task Learning)	$\{\mathcal{D}_i(\mathbf{x},y)\}_{i=1}^m$	$\{\mathcal{D}_i(\mathbf{x},y)\}_{i=1}^m$	1. Semantic Shift 2. Label Shift 3. Limited data
Zhou et al. (2021d)	Multi-source Transfer (Domain Generalization)	$\{\mathcal{D}_i(\mathbf{x},y)\}_{i=1}^m$	N/A	1. Semantic Shift 2. Label Shift

TABLE 1.5 – A brief summary of our contributions towards the thesis.

These theoretical results above are usually based on an underlying assumption that there exists a hypothesis that can minimize both the source and target risk. However, the target risk is not observable (Zhao et al., 2019a) since we don't have the target labels under the unsupervised domain adaptation setting. Table 1.4 shows compression of recent domain adaptation theory where there is especially focus on the non-observable term which may lead to the conditional shift problem (Zhao et al., 2019a). From this table, we can see that, no matter applying which theoretical divergence, there is always a non-observable term, which may lead to the conditional shift problem. We will discuss this issue in detail in Chapter 3.

1.6 Overview of Objective and Methodology

Before introducing the related works and our main contributions in details, it is helpful to take an overview of the research objectives and methodologies involved in this thesis.

As aforementioned, we have devoted to leveraging the transferable features from different domains by focusing on some specific learning scenarios to solve the conditional shift problems, semantic matching and label distribution shift problems in transfer learning with different settings. We illustrate a brief summary of our contributions in Table 1.5 to show the main research objectives of each contribution. In this thesis, we aim to figure out how to extract the transferable features from some source domains and then implementing some advanced

techniques to enhance the transfer process. One key idea in such a transferring process is to learn transferable and domain invariant features then adapting them to another domain.

Previous works have proposed a large number of approaches to align the marginal feature distribution ($\mathcal{D}(\mathbf{x})$) and showed an impressive performance for real-world applications (Tzeng et al., 2014; Isola et al., 2017; Saito et al., 2018), especially with the help of adversarial training techniques (Ganin et al., 2016). Such approaches usually ignored the conditional distribution shifting and assumed that the conditional distribution ($\mathcal{D}(y|\mathbf{x})$) remains unchanged during the transferring process. However, such assumptions may not hold in some cases (Zhao et al., 2019a), and the conditional or label distribution may shift during the transfer process, which may hinder the learning performance (Zhao et al., 2019a). To alleviate this difficulty, in Zhou et al. (2021c) we investigated the conditional shift problem in DA and theoretically analyzed the conditional shift problems with Wasserstein distance and thereafter the discriminative active learning for addressing this issue, which will be elaborated in Chapter 3.

Another issue involved in the marginal-only training method (Ganin et al., 2016) is that the labelling and category information will be neglected. Due to the similar problem setting between DA and DG, many previous works have directly applied the adversarial training method of DA for DG. However, this may still lead to the semantic misalignment problems in DG (Dou et al., 2019). The semantic misalignment problems usually refers to the situation that the semantic distributions ($\mathcal{D}(\mathbf{x}|y)$) were not correctly aligned during the transfer learning process. To address this issue, in Zhou et al. (2021b), we investigate the value of category relations in domain generalization problems via adopting the metric learning objective to leverage the category similarity relations of the instances to encourage a clear decision boundary, and achieve the state-of-the-art performances for domain generalization problems. We show the detailed results of this work in Chapter 4.

Then, we focused on the semantic matching problems in Zhou et al. (2021a) by bounding the task error with semantic distributions $\mathcal{D}(\mathbf{x}|y)$ divergence across all the tasks, where we provide the first theoretical results for controlling the semantic divergence in MTL. This theoretical result then inspired us the *Semantic Multi-task Learning* algorithm, which can help to alleviate the label shift problems and showing state-of-the-art performances. Details are introduced in Chapter 5.

Lastly, in Zhou et al. (2021d), we extend the learning objectives in another multi-source transfer learning task, *i.e.*, explore the value of label and semantic information in the domain generalization problems, in which we provide a concrete theoretical framework to understand the generalization property by bounding the target domain risk with the label and semantic divergences. Our results reveal that to control the target risk, one should jointly control the source errors that are weighted according to label information and align the semantic conditional distributions between different source domains. The proposed theoretical analysis

then leads to an efficient algorithm to control the label distributions while matching the semantic conditional distributions. We also consider evaluating the algorithm performances under label distribution shift problems in domain generalization learning paradigms. The results showed that our method could outperform most of the baselines, and could handle the label shift problems which could not be solved by several principled approaches. We introduce this work in Chapter 6.

In terms of methodologies, we have studied the adversarial training with distribution matching methods from different aspects. In Chapter 3, we explore the conditional shift problems in domain adaptation with the Wasserstein adversarial training and active learning objectives. In Chapter 4, we explore the category relations in domain generalization with the Wasserstein adversarial training together with the metric similarity learning objective. In Chapter 5, we theoretically studied the semantic information in multi-task learning with a concrete algorithm to match the semantic distributions together with the label distributions for each task, which is lastly extended to the domain generalization problems in Chapter 6.

1.7 Discussion and Conclusion

In this chapter, we quickly introduced the preliminaries of the learning paradigms tackled in this thesis, namely, domain adaptation, domain generalization and multi-task learning problems. The core methodology in our work is to match the data distributions among different domains. We present the comparison of some related distribution measures and also discuss the benefits and disadvantages of these kind of measures, which then inspire our work using the Wasserstein distance in Chapter 3 and Chapter 4, and the family of f-divergence used in Chapter 5 and Chapter 6. Now, based on these preliminaries of the transfer learning problems involved in this thesis, we survey the state-of-the-art works in the next Chapter 2.

Chapitre 2

Literature Review

This thesis is devoted to learning transferable features from different domains. In this chapter. we start by summarizing the transfer learning methods, then the single source transfer problems, i.e., domain adaptation (DA) methods, and lastly, the multi-source transfer learning methods, including domain generalization (DG) and multi-task learning (MTL) methods related to our work.

We present in Fig. 2.1 a basic structure of this literature review chapter to show the contents and connections of each section. More specifically, we first show the related works of the general transfer learning problems in section 2.1, which are the early-stage approaches in transfer learning and are essential to the following works for DA, DG and MTL. Then, we present the state-of-the-art works on DA in section 2.2, which is mostly relative to our contribution (Zhou et al., 2021c) introduced in Chapter 3.

After that, we present the literature review on the multi-source transfer learning problems, including the MTL problems, which is mostly connected to our contribution (Zhou et al., 2021a) introduced in Chapter 5, as well as the literature of DG problems, which is most related to our contributions (Zhou et al., 2021b,d).

Below we start by introducing the general transfer learning approaches.

2.1 Reviews on Traditional Transfer Learning Approaches

Transfer Learning aims to transfer the knowledge learned from a source domain to a new target domain (Pan and Yang, 2009). We focus here on traditional transfer learning, of which the brief summary is represented in Table 2.1. Specifically, as proposed by Weiss et al. (2016), we can categorize the traditional transfer learning approaches into several groups including the instance-based (section 2.1.1) and mapping-based approaches (section 2.1.2). As a new trend, model-based methods for meta-transfer learning have been investigated in recent years, and we also present a quick summary on this area in section 2.1.3.

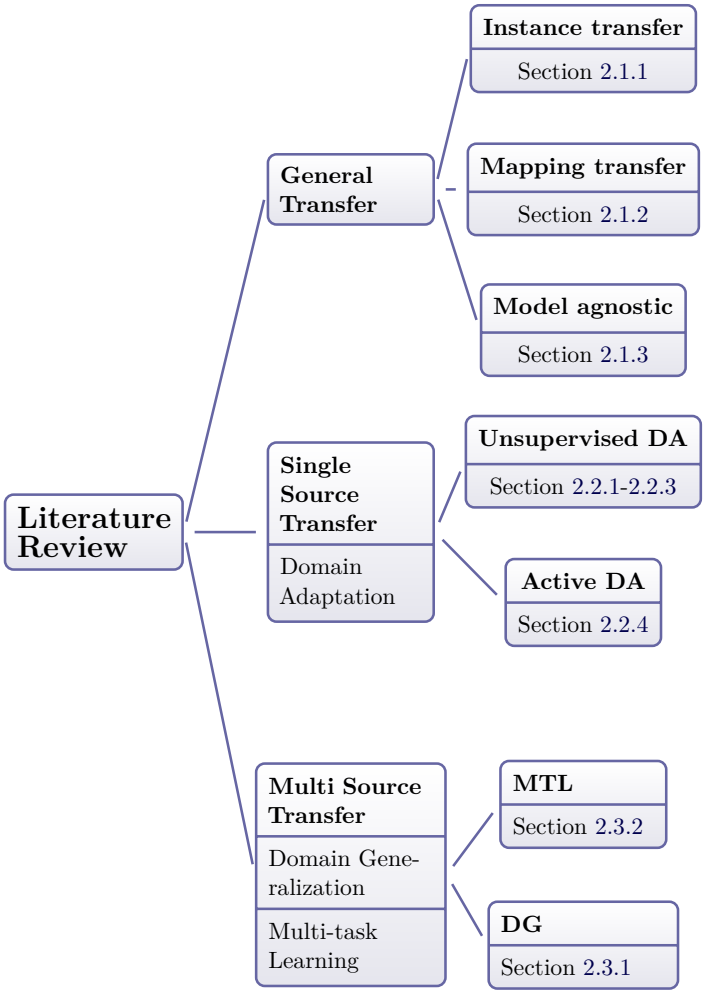


FIGURE 2.1 – The Structure of the Literature Review chapter.

Then, we summarize in section 2.2 the related works on single-source transfer learning, *a.k.a.* the domain adaptation problems, and we present in section 2.3 the state-of-the-art works on multi-source transfer learning problems.

2.1.1 Transferring the knowledge of instances

The main idea of such an instance-based method is to assign an instance-specific weight to compute the objective loss function. This kind of approaches are based on the assumption that although the source domain knowledge couldn't be reused directly on the target domain, some parts of the source data could still be reused with few labelled target instances (Pan and Yang, 2009).

As introduced in section 1.2.1, the instance-weighting-based transfer learning approach played an important role at the early stage of transfer learning (e.g. Dai et al., 2007; Ma et al., 2012) and also domain adaptation (e.g. Long et al., 2014; Li et al., 2016b). This kind of method was

Category	Description	Example
Instance-based	assign a weight to the source domain instances	Boosting based Methods : TrAdaBoost (Dai et al., 2007), Online Boosting Wang and Pineau (2015), GapBoost (Wang et al., 2019a)
Mapping-based	mapping data from different domains into a latent space and then apply the classification on this extracted space	Mean-Max Discrepancy (Tzeng et al., 2014)
Network-based	use the network (pre-trained on source domain) then fine-tuning on the target domain	Fine-tuning on the trained models (Sun et al., 2019)

TABLE 2.1 – A brief categorization of transfer learning. Transfer Learning approaches usually could be categorized into 1) instance-based, 2) mapping-based, 3) network-based and 4) adversarial based transfer learning. Categorization methodology borrowed from Weiss et al. (2016).

firstly adopted in natural language processing problems (e.g. Jiang and Zhai, 2007; Alyafeai et al., 2020) and then for computer vision problems (e.g. Wang et al., 2019a; Soleimani and Nazerfard, 2021; Niu et al., 2020).

The instance based transfer method was usually combined with boosting method, of which the key idea is also to assign weights for different instances. When applying the boosting method into transfer learning, one can assume that even though the distributions of source and target are different, they still satisfy the basic assumption that some of the source instances may be useful in transferring the knowledge to the target domain while the other instances may be harmful.

For example, TrAdaBoost, proposed by Dai et al. (2007), is an AdaBoost based method for transfer learning, which took advantage of the nature of Boosting methods which could help to reduce the effect of 'useless' instances in the source domain and encourage the effect of *useful* instances in the source domain to transfer the knowledge to the target domain.

Jiang and Zhai (2007) proposed to heuristically remove the 'misleading' source instances by evaluation the conditional probabilities between source $\mathcal{D}_s(y|\mathbf{x})$ and target $\mathcal{D}_t(y|\mathbf{x})$. As the conditional distribution stands for the labelling function f in different domains, this kind of evaluation also can be used for addressing the conditional shift problem in unsupervised domain adaptation problems (see our contribution (Zhou et al., 2021c) presented in Chapter 3).

Recently, Wang et al. (2019a) also considered the situation where some of the target labels are available in the target domain and adopt the $\mathcal{Y}$-Discrepancy (Mehryar et al., 2018) for measuring the discrepancies between the source and target domain in case where some target

labels are given. With the target labels in hand, the learner can measure the weights of the source instances and apply the boosting method for transferring the knowledge. The notion of performance gap proposed by Wang et al. (2019a) enables us to measure the divergence between domains in transfer learning by exploiting the label information in the target domain. The performance gap then encouraging some theoretical analysis for instance weighting. It could be used to analyze other forms of transfer learning like a parameter or feature transfer (Wang and Pineau, 2015). It could also have some connections with knowledge transfer strategies for other learning paradigms such as meta-learning (Sun et al., 2019; Franceschi et al., 2018) or lifelong learning (Ruvolo and Eaton, 2013).

2.1.2 Transferring feature using mapping-based methods

The mapping-based method refers to the approaches mapping the source and target information to a new space (Pan and Yang, 2009), and aligning the source and target features in that space. As introduced in section 1.2.2, this kind of method usually adopts a distribution distance measure, *e.g.* the Mean Max Discrepancy (MMD) metric introduced the in section 1.5, to align the source and target features (see also in Fig. 1.9 in section 1.5.) in the latent space. The mapping-based transfer learning methods were widely used in SVM or shallow model-based approaches before the era of deep learning (Yang and Gao, 2013; Muandet et al., 2013).

A remarkable work of mapping-based transfer learning approach is the *Transfer Component Analysis* (TCA) by Pan et al. (2010). TCA focuses on finding a transferable invariant feature representation in the Reproducing Kernel Hilbert Space (RKHS) with Maximum Mean Discrepancy. It assumes that the distributions and the transferable components (*i.e.*, the underlying transferable features) are close to each other *w.r.t.* MMD. With the development of deep learning, the traditional SVM or shallow model were extended to a deep neural network model while also extending the MMD to compare the distributions in a deep model with the help of an extra domain confusion loss. For example, the discrepancy based domain adaptation method (see section 2.2.2) was based on this kind of transfer learning approach.

In the same context, Muandet et al. (2013) implemented MMD as a distribution regularizer and proposed the kernel-based *domain invariant component analysis* (DICA) algorithm. DICA minimizes the distribution mismatch by minimizing MMD regularizer across domains. Yang and Gao (2013) proposed a model based on canonical correlation analysis with MMD to regularize the differences among domains for generalization to a new domain.

These mapping-based transfer learning approaches then inspired many DA and DG approaches. We will present them in section 2.2.2 and section 2.3.1.

2.1.3 Model Agnostic Meta Learning-based Approaches

In previous sections, we focus on the task setting where the source domain and target domain information are provided to the learner at the same time. However, this situation may not hold in many practical scenarios. For example, consider a robot vision system used for navigation ; it's not possible to train algorithm with all the navigation environments at one time. A more practical setting is to train the model in an online setting (Mancini et al., 2018; Dou et al., 2019; Li and Hospedales, 2020), a sequential setting (Balaji et al., 2018) or the continual learning setting (Parisi et al., 2019), from which the model could fast adapt to task $N+1$ based on the trained task N (Finn et al., 2017). Recent works have proposed to use a meta-learning based transfer learning setting (Sun et al., 2019, 2020).

Meta-learning (*a.k.a.* learning to learn) is a long-standing topic exploring the training of a meta-learner that learns how to train particular models (Aiolli, 2012; Sahoo et al., 2018). Most of the recent meta-learning approaches were rooted in the fundamental Meta-Learning paradigm, namely, Model Agnostic Meta-Learning (MAML by Finn et al. (2017)). MAML was proposed to learn an internal feature that is broadly applicable to all tasks in a task distribution, rather than a single task (Sun et al., 2019). In the same context, Soh et al. (2020) proposed to leverage the parameters for internal learning then exploit the external training loop inside the meta-transfer training method to handle the super-resolution for few-shot learning. Park et al. (2020) proposed to learn the transferable features from one class to another with a meta-training framework.

The meta-learning method is also applied in the domain generalization problems by leveraging the meta-train and meta-test framework, as introduced in section 2.3.1. In our contributions (Zhou et al., 2021b,a), we compared our method and this kind of meta-learning approach to confirm the effectiveness of our contribution.

In next section, we introduce the domain adaptation (DA) approaches, which are most related to our contribution (Zhou et al., 2021c).

2.2 Related Works on Single Source Transfer Learning

In this section, we summarize the single source transfer learning problem involved in our work, *i.e.*, the domain adaptation problem. Generally, the DA approach can be splited into several parts, including :

1. Reconstruction based approach, which usually reconstructs the data of the source or target domain using an auxiliary task to learn the shared features (Ghifary et al., 2015b).
2. Discrepancy based approach, which usually focuses on fine-tune the model with the unlabelled target data to overcome the domain shift (Zhuang et al., 2015).

2. Adversarial based approach, which leverage the power of the adversarial train scheme to achieve the distribution alignment (Ganin et al., 2016).

We present a brief structure of the DA approaches in Fig. 2.2. The early-stage approaches on DA majorly leveraged the reconstruction and discrepancy based approaches. Latterly, the adversarial training based approaches have attracted more attentions and showed state-of-the-art performances (Wang et al., 2018; Zhao et al., 2020b).

In this section, we briefly introduce the reconstruction and discrepancy based approaches in section 2.2.1 and section 2.2.2, respectively. After that, we focus on the adversarial based approaches in section 2.2.3. Note that our contribution (Zhou et al., 2021c) and (Zhou et al., 2021b) were most inspired by the adversarial based method. Finally, we will present the work with semi-supervised DA related to our work, especially the active learning method enhanced DA approaches in section 2.2.4. Furthermore, we also notice some more recent works that combine some new methods for DA including the contrastive training (Thota and Leontidis, 2021; Song et al., 2021b; Wang et al., 2021b), image generation (Yang and Soatto, 2020) for domain randomization training (Volpi et al., 2021) and self-training (Yang et al., 2021) etc. Even though they are not directly related to the methodologies involved in this thesis, we summarize some interesting works in section 2.2.5 to show some new trends in this area.

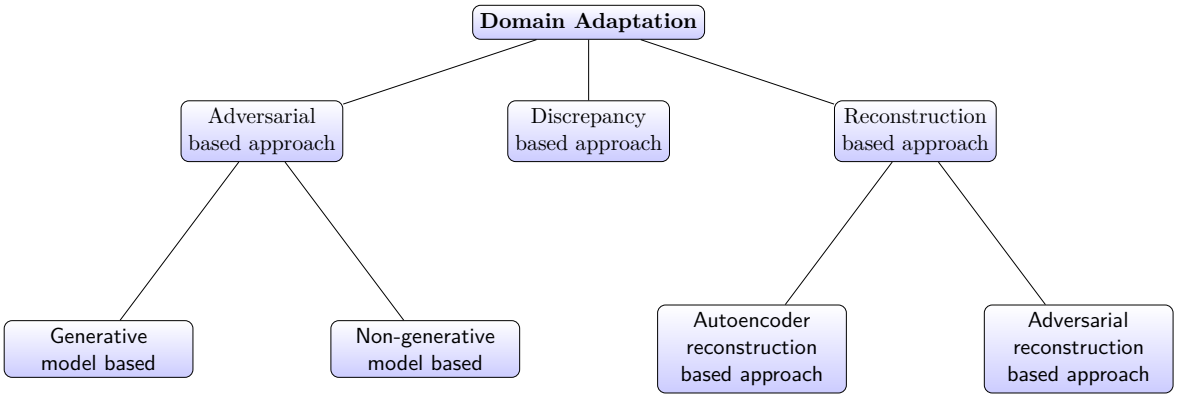


FIGURE 2.2 – Domain Adaptation Taxonomy.

2.2.1 Reconstruction based approaches

The reconstruction based DA approach usually reconstructs the data of the source or target domain using an auxiliary task to learn the shared features (Wang et al., 2018). This kind of approach typically adopts the autoencoder reconstruction or adversarial reconstruction. The autoencoder model uses an encoder-decoder framework (Bengio, 2009) and explores the power of reconstruction loss to learn the feature relation (Ghifary et al., 2016). And the adversarial reconstruction approaches usually estimate the reconstruction loss by a GAN model (*e.g.*, the dual GAN (Yi et al., 2017), cycle GAN (Zhu et al., 2017) etc.) Notice that the adversarial

reconstruction methods also have connections to the adversarial based approaches introduced in section 2.2.3.

Under the DA settings, the encoder is trained to learn a shared encoding function that maps the inputs from source and target to a shared feature space while keeping the domain-specific features by the reconstruction loss between the source and target, while the decoder is trained for data reconstruction (Ghifary et al., 2015a).

A recent work by Glorot et al. (2011) adopted the denoising autoencoder (SDA) to reconstruct the data from different domains with the same network, which can learn the high-level representations for both the source and target domain. Together with SDA process, a linear classifier is then trained to make predictions for both the source and target domain. Notice that the framework by Glorot et al. (2011) required large amount of computational costs and is limited by the scale of the high-dimensional features. This is why Chen et al. (2012) proposed the marginal SDA approach to marginalize the noise for the model, which then improved the computational efficiencies.

After that, Ghifary et al. (2015b) proposed an autoencoder model to jointly learn the self-reconstruction for the data from source domain and between-domain reconstruction for cross-domain features. Another stream of reconstruction based approach is the adversarial reconstruction based approach, which usually jointly trains an adversarial objective with a reconstruction objective (Wang et al., 2018). This stream of research is closely connected to the general adversarial based approaches presented in section 2.2.3.

2.2.2 Discrepancy based approaches

The discrepancy based approach usually focuses on fine-tuning the model with the unlabelled target data to overcome the domain shift. Recall that the domain shift problems usually refer to the scenario where the data distribution change between the training sets (see Fig. 0.1). This stream of approach played an important role in the early stage DA researches, which usually focused on transferring the features via minimizing the discrepancies or learn the kernel mappings of the representations (Ben-David et al., 2006). The discrepancy based approach usually focuses on aligning the statistical discrepancy between source and target by minimizing the statistical distribution measure introduced in section 1.5. For example, by diminishing the domain distribution shift using the MMD (Tzeng et al., 2014; Long et al., 2015a), KL divergence (Zhuang et al., 2015) or discrepancy distance (Mansour et al., 2009b) between domains.

Typically, a statistical distribution distance (*e.g.*, the ones introduced in section 1.5.1) is adopted to measure the divergence between source and target domain. We present a general framework of this kind of approach in Fig. 2.3. The source and target data are processed by the source and target feature extractors, respectively, which typically share the model parameters

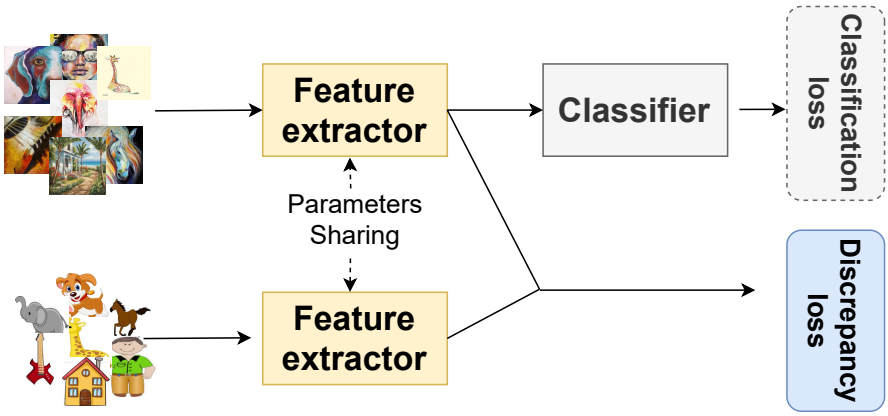


FIGURE 2.3 – General scheme of the discrepancy based domain adaptation approach. The source and target feature extractors typically share the model parameters with each other. The classifier computes the classification loss. The discrepancy loss is computed with the features from both source and target domain using the statistical distance (see section 1.5.1). The invariant feature is learned by minimizing the discrepancy loss. Dataset images are drawn from PACS dataset (Li et al., 2017b).

with each other. Then, there follows the classifier, which computes the classification loss. The discrepancy loss is computed with the features from both source and target domain using the statistical distance. The invariant feature is learned by minimizing the discrepancy loss.

In the early stage of DA researches, MMD has been used as a metric for computing the distance between two datasets. For instance, Ghifary et al. (2014) proposed a single hidden layer model using the MMD metric to compute the representations from source and target to minimize the domain distance. This work was limited by the single hidden layer network and was not able to handle with complex features or raw image inputs. Then, Tzeng et al. (2014) and Long et al. (2015b) extended the method of Ghifary et al. (2014) with a deep CNN model to extract the input features. After that, Long et al. (2017b) adopted the joint MMD metric in the joint adaptation network (JAN) to align the shift in the joint distributions of both features and the corresponding labels.

As discussed in section 1.5.1, the methods with MMD metrics usually need to map the input to the kernel space, which will limit the performance of the model. Recently, Wang et al. (2021c) proposed to rethink the value of the MMD metric in DA and propose a novel discriminative MMD metric to restrain the degradation of feature discriminability. Besides, this discriminative MMD can also help to control the expansion of intra-class distance, which cannot be handled by the original MMD metric.

Another kind of method adopted in the discrepancy based approach is the correlation alignment (CORAL) method (Sun et al., 2017), which aims to measure the covariance of the source and target domain features. For example, Zhang et al. (2018) adopted the general Euclidean distance for covaraince of source and target domains. Recently, Chen et al. (2020a) adopted

a monument matching distance (Peng et al., 2019a) to measure the high-order statistics of the source and target features. In fact, this kind of approach usually relies on computing the high-order statistics of the source and target features, which will increase the computational costs. This made this kind of approach less attractive.

Except the MMD metric measures or the covariance discrepancies between the source and target domains, other approaches were proposed to to directly minimize the discrepancies by aligning the statistics of certain network layers. For example, Carlucci et al. (2017) proposed to automatically choose the layers to align based on the statistics of model layers trained on the source and target data. Then, Li et al. (2018g) proposed to align the mean and variance of the classifier's batch normalization layers with the estimated mean and variance from the unlabelled target data. This kind of method explicitly manipulates the neural network layers, so it will be more time-efficient than those methods which need to compute the MMD or other discrepancy distances. However, to directly change the statistics of certain layers cannot guarantee the invariant features since the feature distributions are usually aligned by both the feature extractor and classifier rather than some certain layers.

To summarize, the performance of the discrepancy based DA approaches are usually limited since they are constrained to map the features to a kernel space or only to align part of the model layer statistics. In recent years, lots of adversarial training based approaches have been proposed. The adversarial based method is connected to our work (Zhou et al., 2021c) introduced in Chapter 3. We introduce them in the next part.

2.2.3 Adversarial based approach

The adversarial training method of domain adaptation was first inspired by the theoretical work of Ben-David et al. (2010a). As we introduced in section 1.5.2 (Eq. 1.14), the results of Ben-David et al. (2010a) demonstrated that the target domain error is bounded by the source domain error, the divergence between source and target domain, as well as the joint prediction error between source and target, which is assumed as a small constant.

Based on this theoretical result, Ganin et al. (2016) proposed the domain adversarial training network (DANN), which jointly learns to minimize the classification error and achieve a domain invariant feature learning via a gradient reversal layer in the discriminator. This framework then inspired some following adversarial training based works in domain adaptation. We provide a detailed introduction of DANN model in the Appendix A.1.

To the best of our knowledge, most recent approaches inspired by DANN typically could be summarized as represented in Fig. 2.4. The model could contain three main parts : feature extractor, discriminator and classifier. Many methods differ from each other based on different objective functions. Generally, the objective functions are trained adversarial by the gradient reversal or GAN style losses. In the following, we elaborate more on these kinds of approach

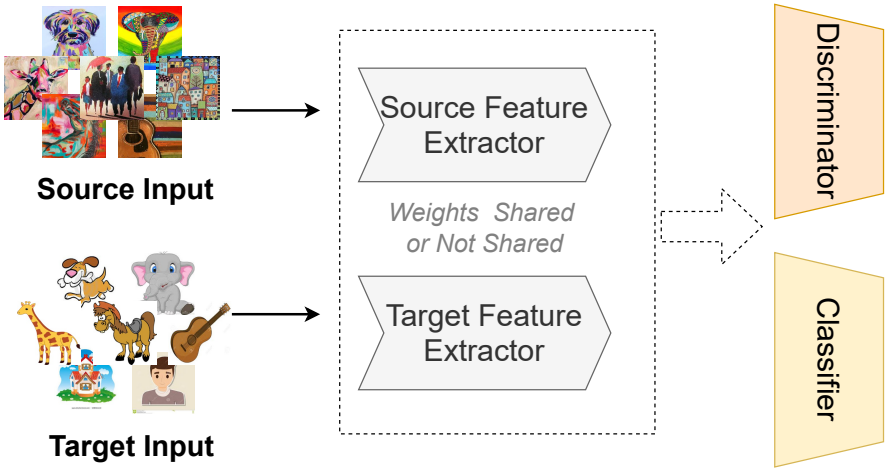


FIGURE 2.4 – General scheme of recent adversarial training based DA methods. Usually there are source feature extractor and target feature extractor to take the inputs from both source and target domain. The source and target feature extractors typically share the model parameters. Depending on specific task, discriminator is trained with different objective functions (see section 2.2.3 for detailed discussion). The classifier is trained with labelled source domains under a supervised mode. Figure modified from Tzeng et al. (2017). Dataset images are drawn from PACS dataset (Li et al., 2017b).

since they are closely connected to our work (Zhou et al., 2021c,b).

The discriminative adversarial method usually relies on a discriminator to distinguish the extracted feature if it is from the source or target domain. For example, the discriminator model of DANN is trained to predict a binary output on discriminating the data from the source and target domain. Then, Tzeng et al. (2017) combined the DANN method with the GAN training method to aligning the marginal distribution ($\mathcal{D}(\mathbf{x})$) between the source and target domain. After that, Long et al. (2017c) proposed to aligning the conditional distribution ($\mathcal{D}(y|\mathbf{x})$) across domains to leverage the cross-covariance between features and classifier predictions. Recently, Liu et al. (2019b) proposed to not only aligning the features between source and target domain but also to generate some transferable examples so that the classifier could learn a better classification boundary.

There are also recent approaches that implemented consistency loss (Hoffman et al., 2018) to ensure the generative model's ability to capture the pixel level features of the inputs (Wu et al., 2019; Wang and Deng, 2021). This can be viewed as a stream of combination with the reconstruction loss based approach (see section 2.2.1) since the input data are reconstructed during the adversarial training process by aligning the image pixels, *i.e.*, pixel-level alignment. Hoffman et al. (2016) first proposed to align not only the domain level features but also the image pixel-level information. The pixel-level alignment is achieved by a pixel adversarial loss to apply the category-specific constrains, *e.g.*, the pixel percentage histograms in the source and target domain data. Then, Bousmalis et al. (2017) proposed an unsupervised

DA method and focuses on pixel-level alignment with the generative models. The intuition behind that is to achieve invariant features via a content similarity loss to make the source domain feature appears as those drawn from the target domain.

In the same vein, the content similarity loss is extended to the content consistent loss (Zhu et al., 2017) to ensure the image pixel-level alignment. The intuition of Zhu et al. (2017) is that an image from the source domain is still identical to itself after a cycle of the adaptation process. Based on this content consistent loss, Hoffman et al. (2018) proposed the cycle consistent adversarial domain adaptation method mixing both image-pixel information and the domain features. Similar to the reconstruction based approach, this consistency loss based adversarial training approach is limited due to the computational costs. For example, the training procedure of the cycle consistency loss requires an aligned pair of images, which made the adaptation become less effective.

Some Optimal Transport (OT) (Courty et al., 2016; Redko et al., 2019) based approaches (Redko et al., 2017; Shen et al., 2018; Stan and Rostami, 2021) with Wasserstein adversarial training (see section A.2) have also been proposed for transfer learning especially for domain adaptation (Xu et al., 2020; Dhouib et al., 2020). As explained in section 1.5.1, the optimal transport technique with Wasserstein adversarial training could capture the geometry properties of the data distributions. Besides, compared with the adversarial training methods promoted by Ganin et al. (2016); Tzeng et al. (2017); Wang and Deng (2021) and benefits from the advantages of Wasserstein distance by its gradient property (Arjovsky et al., 2017) and the promising generalization bound (Redko et al., 2019). The empirical studies (Gulrajani et al., 2017; Shen et al., 2018) also demonstrated the effectiveness of OT for extracting the invariant features to align the marginal distributions of different domains.

In this thesis, we also adopt the Wasserstein adversarial training method (Zhou et al., 2021c,b) when finding the invariant features. In Zhou et al. (2021c), we implement the Wasserstein adversarial learning method to train a critic model for measuring the diversity of the target instances (see Chapter 3), and in Zhou et al. (2021b), we implement similar kinds of pair-wise adversarial training methods for all the source domain instances to achieve domain invariant features (see Chapter 4).

In the following part, we will briefly summarize the active learning methods for domain adaptation, which are closely connected to our contributions in Zhou et al. (2021c).

2.2.4 Active Learning for Domain Adaptation

We also noticed that some DA methods are equipped with Active Learning (AL) for enhancing the adaptation process, which is closely related to our work (Zhou et al., 2021c). Persello and Bruzzone (2012) proposed a two-direction AL algorithm for DA : query the most informative from the target domain and remove the most strange features out of the source domain. Wang

et al. (2014) proposed the active transfer technique for the model shift problem while assuming the shifts are smooth. Based on this assumption, Wang et al. (2014) implemented conditional distribution matching algorithm and off-set algorithm to modelling the source and target tasks by comparing the Gaussian Distributions. Zhang et al. (2013) proposed a distribution correction algorithm over kernel embeddings to handle the target shift.

The last two methods held on the assumption that there existed an affine transformation of conditional distribution from the source to the target. In fact, these methods were based on an SVM classifier or kernel matching technique, which are somehow difficult to implement for large-scale applications. Li et al. (2021a) proposed a semi-supervised domain adaptation approach that randomly query some target domain instances. This random selection method can not control the uncertainty and diversity of the target domain instances.

The most similar approach related to our work is the active adversarial domain adaptation (AADA) (Su et al., 2020) that proposed an active learning method using $\mathcal{H}$ divergence and the importance sampling technique to query the target instances. However, the importance sampling, query strategy they adopted assumed that the data support of target domain is a subset of the data support of the source domain ($\text{supp}(\mathcal{T}) \subseteq \text{supp}(\mathcal{S})$), which may not hold in many DA settings (Prabhu et al., 2020) while our work (Zhou et al., 2021c) directly estimates the uncertainty and diversity using the model's output, which doesn't require such kind assumption (see section 3.3.2). Furthermore, as discussed in section A.2 and Appendix A, the Wasserstein adversarial training method we adopt could predict a constant critic score. On the contrary, in the AADA method, the discriminator was trained under the domain adversarial training method (Ganin et al., 2016) where the domain label is trained under a binary classification model to distinguish the instances from source or target domain, which restricts the power of active training. In our contribution (Zhou et al., 2021c), we exhaustively investigate the work on active learning-based domain adaptation with both theoretical analysis and extensive empirical demonstrations.

2.2.5 Some other approaches

Besides the approaches summarized above, we also notice some new methods have been proposed in recent years. Most of them were combined with some new techniques *e.g.*, domain randomization with meta training (Volpi et al., 2021), Fourier image generation (Yang and Soatto, 2020; Issar et al., 2021), constrastive training (Thota and Leontidis, 2021; Song et al., 2021b; Wang et al., 2021b) etc. Those methods appear new to the community and are hard to be categorized to a certain principled approach mentioned in section 2.2.1 to section 2.2.3. Besides, they are not directly closed to our method, so we summarize some of them here to show some potential new trends in this area.

One approach is to implement the meta learning method which splits the training data into

meta-train and meta-test subset to simulate the domain shift(see also 2.1.3). Volpi et al. (2021) adopted the meta-learning objective to generate some intermediate meta-domains with the randomized image manipulations. Yue et al. (2021) proposed a self-supervised learning framework for the few-shot DA problem. This work not only aligns the cross-domain features but also captures the category-wise semantic structure of the source and target domain features through the self-supervised learning process.

Similar to our work (Zhou et al., 2021b), learning the class-level information is also attractive in the recent DA approaches. In this context Kang et al. (2019) proposed the first constrastive learning method for domain adaptation, where they proposed to minimize the intra-class discrepancy and maximize the inter-class discrepancy. Similarly, Thota and Leontidis (2021) proposed the contrastive domain adaptation method in which several recent contrastive learning frameworks were studied to tackle the DA problems. More recently, Wang et al. (2021b) proposed a method where for an anchor image from a domain, minimizing the distance with samples from the same class no matter which domain they come from. Since the target domain label is not accessible during training, this method assigns a pseudo label to the target domain instances. The contrastive learning methods have become more and more popular in recent years, and it might be a trend to implement some new contrastive approaches in all areas of transfer learning.

Another interesting stream of work aims to implement a Fourier transform to generate new images for DA method. Yang and Soatto (2020); Issar et al. (2021) adopt the Fourier image generation method to solve the image segmentation problems. The Fourier image generation method reduces the domain alignment, which is not relying on learning the transferable features but only use the Fourier transform and its inverse to generate some new data for improving the adaptation performance. This kind of approach has nothing with knowledge transfer, and consequently is less related to our work.

Recent DA works also show promising performance on some real-world applications. For example, Li et al. (2021a) implemented the DA method for animal pose estimation tasks. Jiang et al. (2021) tackles the keypoint detection problems using a regressive adaptation network. Zhang et al. (2021) adopted the DA methods together with the scale and range information of images for handling a 3D objection detection problem. Chen et al. (2021) implemented a semi-supervised DA approach together with a double-level domain mixing method to solve the semantic segmentation problems. Bai et al. (2021) adopted the unsupervised DA method for person re-identification tasks. Yi et al. (2021) implemented an adversarial DA method for handling the sparse point cloud LiDAR labelling problems. Finally, Song et al. (2021a) proposed the spatio-temporal contrastive DA method to the video recognition problems. Those recent DA applications have shown promising performances. They could drive to more applied works in the future.

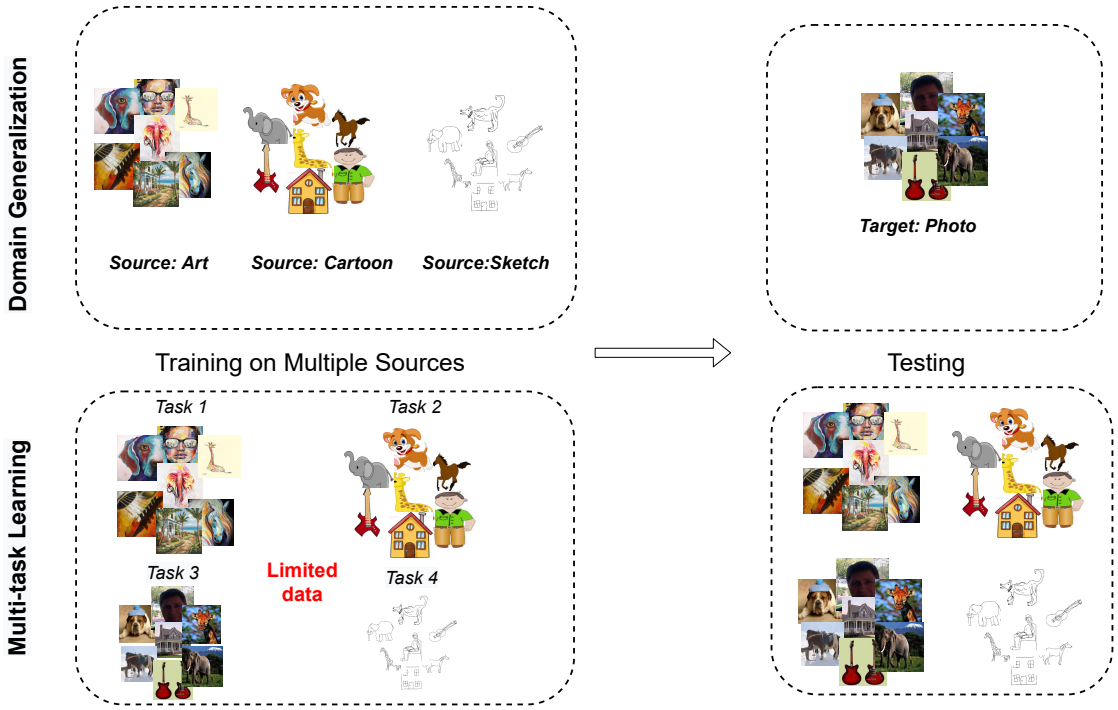


FIGURE 2.5 – Comparison of multi-source transfer problems involved in our work. We consider the Domain Generalization (DG) and Multi-task Learning (MTL) problems. For the DG problems, we train the model on several source domains and test it on the unseen target domain. For the MTL problems, we train the model with limited data on several different tasks and test it on the same tasks. Dataset images are drawn from PACS dataset (Li et al., 2017b).

Now we start to survey the multi-source transfer learning problems, which is prevalent in recent researches, as well as relating to our contributions (Zhou et al., 2021a,b,d)

2.3 Related Works on Multi-source Transfer Learning

In previous sections, we mainly focused on the situation where we have a single source domain and aim to adapt it to a single target domain. When the amount of data in one source is not sufficient, we could not expect a successful adaptation performance (Peng et al., 2019b). For example, consider the adaptation task in Office-31 dataset **D** (Dslr) $\rightarrow$ **A** (Amazon). The total size in domain **D** is 498, and the total number of images in domain **A** is 2817, indicating that the target domain has richer information than the source domain. Besides, we also observe that the images in **A** are more diverse than those in **D**, *i.e.*, the data support of the source domain is a subset of the data support of the target domain ($\text{supp}(\mathcal{S}(\mathbf{x})) \subseteq \text{supp}(\mathcal{T}(\mathbf{x}))$). From our preliminary results, we notify that the unsupervised single-to-single adaptation process is not very useful (low accuracy on the target domain), confirming the problem mentioned above.

When the data from one source dataset is not enough , we may expect to leverage several

different source datasets. In the previous Office-31 dataset example, the images in domain **D** is not enough for successful adaptation to domain **A** In this case, we may combine the data from domain **D** with domain **W** to learn more enough features for transferring the knowledge.

Multi-source transfer learning is a powerful extension where the labelled data may come from multiple sources domains with different distributions (Zhao et al., 2019b). Many of the recent multi-source transfer learning researches have been focusing on leveraging the knowledge from multiple source domains and generalizing to a target domain. As stated in table 1.1, we have investigated this kind of problem in several different aspects, including the domain generalization problems and multi-task learning problems. In the following sections, we will briefly summarize the works most related to our research in the context of domain generalization and multi-task learning.

2.3.1 Domain Generalization (DG)

One learning paradigm of multi-source transfer learning we may start by the domain generalization (DG). The general framework of domain generalization methods usually focuses on extracting the transferable knowledge shared across domain distributions, which is committed to generalize well on the unseen target domain. Similar to domain adaptation problems, the underlying assumption of domain generalization is that there exists an invariant feature distribution across all the domains, which consequently generalize well to an unseen target domain (see also in Figure 1.1). Typically, there are several kinds of DG approaches. We could summarize the recent literature into threefold :

1. Invariant representation learning approaches, which focus on adopts some discrepancy minimization or adversarial training method to learn the invariant features for generalization (Li et al., 2018c).
2. Episodic Training-based (meta-learning-based) approaches, which simulate the domain shift by split the data into meta-train and meta-test subgroups (Balaji et al., 2018).
3. Augmentation-based approaches, which generate new data for learning diverse domain features (Schmidt, 2021).

We firstly present the brief categorization of those DG approaches in Table 2.2.

The invariant representation learning approaches usually adopt the distribution matching methods, which were majorly motivated by the domain adaptation theory (Ben-David et al., 2010a; Redko et al., 2017). The domain distributions were aligned via some distribution matching, distribution distance minimization or adversarial training methods to leverage the shared knowledge. For example, maximum mean discrepancy (MMD) was implemented by Li et al. (2018c) as a distribution regularizer together with the adversarial autoencoder (AAE) to learn the invariant features. From this side, Muandet et al. (2013) proposed the kernel-based *Domain Invariant Component Analysis* (DICA) algorithm, where a kernel-based optimization

Category	Methodology	Remarks	Examples
Invariant Representation Learning	Kernel Matching	Implement kernel mapping functions to transform the original data into the kernel space and then matching the features in the kernel space.	(Blanchard et al., 2011) (Blanchard et al., 2021) (Muandet et al., 2013) (Li et al., 2018d) (Erfani et al., 2016)
	Feature Alignment	Align the features across domains by distribution matching or feature normalization. Our work Zhou et al. (2021d) fits this kind of approach	(Motiian et al., 2017c) (Ghifary et al., 2015b) (Jin et al., 2020) (Jin et al., 2021)
	Adversarial Training	Implement adversarial training technique to learn domain invariant features. Our work Zhou et al. (2021b) fits this kind of approach	(Li et al., 2018c) (Jia et al., 2020) (Li et al., 2018e) (Sicilia et al., 2021) (Rahman et al., 2020)
	Invariant Risk Minimization	Train an optimal classifier on top of the representation space to be the same across domains.	(Arjovsky et al., 2019) (Rosenfeld et al., 2021)
Model Agnostic Training	Meta training	Split the data into meta-train and meta-test to simulate the data distribution shift.	(Dou et al., 2019) (Balaji et al., 2018) (Li et al., 2018a)
Data Augmentation	Domain Randomization	Generate new data to simulate complex environments based on the limited training samples to increase the diversity of data.	(Tobin et al., 2017) (Khirodkar et al., 2019)
	Data Generation	Implement some generative models *e.g.* VAEs or GANs to generate new images	(Zhou et al., 2020c) (Rahman et al., 2019) (Wang et al., 2020b)

TABLE 2.2 – A brief categorization of the DG problems. We could summarize the recent DG progresses into three major categories, including : 1. Domain Invariant Representation Learning, 2. Model Agnostic Training and 3. Data Augmentation methods. For each category, we summarize the major methodologies inside this kind of approach. For example, the category of Domain Invariant Representation Learning could have four kinds of major methodologies, including kernel matching, feature alignment, adversarial training and invariant risk minimization. We briefly introduce the methodology in column *Remarks* and present the representative works in that category in the last column. Discussions of those approaches are presented in section 2.3.1.

algorithm was implemented to learn a domain-invariant transformation by minimizing the dissimilarities. Ghifary et al. (2015b) proposed to implement adversarial training techniques to extract the domain-invariant features under a multi-task learning style setting. Li et al. (2018f) proposed a DG approach by leveraging deep neural networks for domain-invariant representation learning. Motiian et al. (2017b) proposed to minimize the semantic alignment loss as well as the separation loss based on deep learning models.

Recently, there have been some approaches to cast the domain generalization problems into a meta-learning manner via the episodic training paradigm. The general *meta-train* and *meta-test* notions in meta-learning are used to simulate the distribution shift during each training iteration on the source domain dataset. Specifically, MetaReg (Balaji et al., 2018) was proposed as a regularization term with weighted L_1 loss for the top classification layers. Meta Agnostic Meta-Learning (MAML) (Finn et al., 2017) was adopted by Li et al. (2018b) to back-propagate the gradient of the losses of the meta-test tasks (Dou et al., 2019). Du et al. (2020) proposed to model the shared classifier model parameters as a probabilistic meta-learning model. Sharifi-Noghabi et al. (2020) also adopted meta-learning to simulate the domain shift and implemented an entropy-based loss to give pseudo-labels together with class-level centroids to ensure the semantic properties. Gong et al. (2021) introduced a setting where the target domain is assumed as a compound of several unknown domains, which is treated as a sub-target domain. Then a meta-learning algorithm is implemented to fuse the sub-target domain together with the MAML algorithm for handling the generalization process.

We also notice some recent work (Zhou et al., 2020b; Schmidt, 2021) start to implement some data augmentation methods to generate new images for training. For example, Qiao et al. (2020) adopts the Wasserstein Auto-Encoder for domain augmentation to handle the worst-case generalization problems. This kind of work typically boosts the performance by relying more on the new data rather than transferring the knowledge. In our contributions Zhou et al. (2021b) and Zhou et al. (2021d), we compared our method with those other principled approaches and showed state-of-the-art performances.

As introduced in Table 2.2, our work Zhou et al. (2021b) and Zhou et al. (2021d) were focusing on learning the invariant representation features. The invariant representation based DG approaches share the similar assumption with invariant representation based DA approaches, where we assume there exists an underlying common feature space that can generalize from source domains to target domain. To leveraging the invariant features, one can use the the discrepancy based approach or adversarial training based approach. Similar with the invariant representation based DA approaches, this kind of method in DA may neglect the label information and lead to some feature misalignment problem (Dou et al., 2019). That's why we proposed to leverage the label similarities in Zhou et al. (2021b) (Chapter 4) and explore the semantic relations in Zhou et al. (2021d) (Chapter 6). The meta-learning based approaches usually simulate the domain shift process by the meta-train and meta-test split so

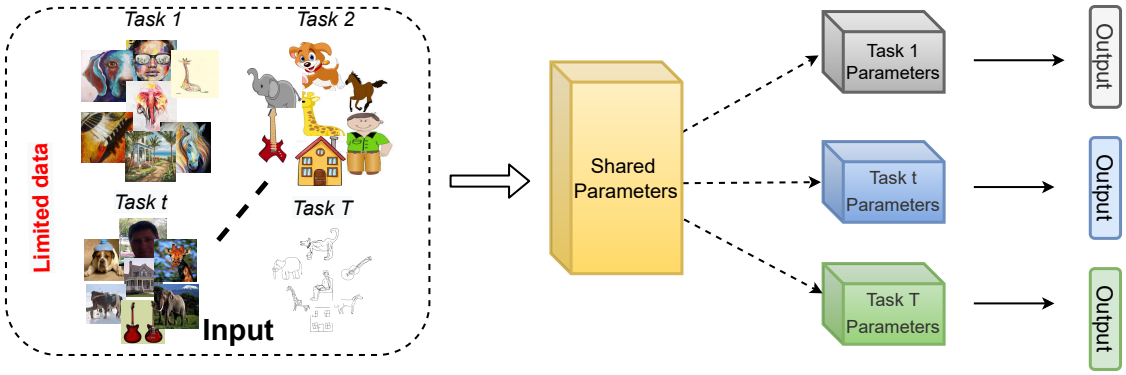


FIGURE 2.6 – The MTL problems involved in our work. Dataset images are drawn from PACS dataset (Li et al., 2017b).

that the model can learn and generalize the features. As pointed by (Dou et al., 2019), the meta learning based approaches can capture the feature diversities. That's also why this kind of method was adopted into the randomization methods (Volpi et al., 2021) to generate some sub-domain features for adaptation or generalization. However, the meta learning method usually relies on the bi-level optimization (Finn et al., 2017), which may be time-inefficient. The time efficiency is also verified in our work Zhou et al. (2021d) showing the meta-learning based approach as less time-efficient (see Fig. 6.7 in section 6.4.3).

2.3.2 Multi-task Learning

Another important multi-source transfer learning paradigm is the multi-task learning (MTL). MTL aims to learn multiple tasks simultaneously and improves learning efficiency by leveraging the shared features across the multiple tasks. It has been prevalent in lots of recent machine learning topics (Li et al., 2014; Wang et al., 2016; Teh et al., 2017). Notice that our contribution (Zhou et al., 2021a) in Chapter 5 is most related to the generalization theoretical results of representation learning based approaches and the task relation approach in multi-task learning.

In the context of theoretical contributions, several results have been proposed. For example, Baxter (2000) utilize the notion of VC-dimension and covering numbers to investigate the generalization bound of MTL, assuming that there exists a common hypothesis space for different tasks. Latterly, Ando and Zhang (2005) provided a more general analysis of structural learning from multiple tasks by leveraging both the labelled and unlabelled data based on the Rademacher complexity. Then, with the Rademacher complexity, some other improved theoretical results have been proposed based on the assumption that the tasks are related with each other and they share a common linear operator chosen to the pre-processed data (Maurer, 2006) or the linear model parameters in a low-dimensional subspace (Maurer et al., 2013, 2016). Besides, the latter work (Maurer et al., 2016) firstly analyzed the generalization error of representation-based approaches. From this side, Zhang (2015) and Liu et al. (2017b) focused

on the algorithm-dependent bounds under the algorithmic stability framework.

In the context of representation learning, Chen et al. (2018) proposed to balance the joint training of multiple tasks to ensure that all tasks are trained at approximately the same rate. Latterly, Wu et al. (2020) investigated the theory for deep multi-task learning, in which they proposed that whether or not tasks' data are well-aligned can significantly affect the performance of multi-task learning. Standley et al. (2020) investigated a task group learning method to dynamically determine which tasks should be learned together while discarded those that should not be learned together, which showed efficiency when learning a large number of tasks. In the context of online MTL, Herbster et al. (2020) studied learning bound under a non-stationary environment. Mao et al. (2020) investigated the theoretical guarantee of MTL under an adversarial training scheme.

All those methods introduced above were mainly focused on how to learn the marginal distribution $\mathcal{D}(\mathbf{x})$ for each task of MTL problems. The labelling information in each task is usually neglected. Furthermore, those works also ignored the label shift problems (Garg et al., 2020), which can hinder the learning performances. In our work (Zhou et al., 2021a), we studied the generalization bounds of MTL on how to control the semantic and label divergence, which demonstrated better performances, especially when label shift problem occurs.

In terms of the task relations aspects, some other approaches also leverage the relations of data distributions of each task to improve the performance of MTL. Aurelio et al. (2019) studied a weighted entropy function for different imbalanced datasets. 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) and Cao et al. (2018a) defined a convex optimization problem to measure relationships while Long et al. (2017a); Kendall et al. (2018) investigated probabilistic models by constructing task covariance matrices or estimate the multi-task likelihood via a deep Bayes model. Latterly, Mao et al. (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., 2010a; Shen et al., 2018), showing improved performances in vision and language processing applications, respectively. In our work (Zhou et al., 2021a), we propose to dynamically estimate the task relations during training which enables the model to learn an optimal task combination that can minimize the task error.

Now, we summarize the semantic matching work for transfer learning, which is connected to our work (Zhou et al., 2021a).

Semantic Matching for Transfer Learning

Another aspect involved in our work is the semantic distribution matching methods in the transfer learning problems. To learn and leverage the semantic distribution $\mathcal{D}(\mathbf{x}|y)$ is an important aspect in machine learning, which has been prevalent in some different topics such as few-shot learning (Motiian et al., 2017a; Luo et al., 2017b), transfer learning (Long et al., 2014) etc. In the context of domain adaptation, Zhang et al. (2019a) 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.

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 et al. (2017a) leveraged the high-order statistics among tasks by analyzing the prediction weight covariance tensor of them.

As many DG approaches were motivated by the adversarial training method in domain adaptation, the semantic misalignment problems could hinder the generalization performance. Aiming to solve this issue, Dou et al. (2019) adopted the triplet loss as an auxiliary learning objective on top of the meta learning-based DG approach (Li et al., 2018b). Matsuura and Harada (2020) proposed to adopt an unsupervised learning objective to explore the class-level similarities to enhance the semantic separation. Our contribution Zhou et al. (2021b) implemented the Wasserstein adversarial training (Shen et al., 2018) to achieve the domain level alignment while exploring the class-level similarities to constrain the instances from the same class to stay close and let the instances from the different category far from each other, *i.e.*, achieving the semantic separation with a metric learning objective.

2.4 Summary

In this chapter, we summarized the related works on the transfer learning problems involved in our work, including the domain adaptation, domain generalization and multi-task learning problems, which are essential to our researches. With these related works in hand, we could show our contributions in the following chapters.

Chapitre 3

Discriminative Active Learning for Domain Adaptation

In this chapter, we show the main results of our work (Zhou et al., 2021c) where we explore the conditional shift problems using a novel active query strategy that controls the diversity and uncertainty principles to find out the most informative features in the target domain.

Résumé

L'adaptation de domaine qui vise à apprendre une caractéristique transférable entre des domaines différents mais connexes a bien été étudiée et a démontré d'excellentes performances empiriques. Les travaux précédents se sont principalement concentrés sur la mise en correspondance des distributions marginales des caractéristiques à l'aide des méthodes d'apprentissage adversarial tout en supposant que les relations conditionnelles entre le domaine source et le domaine cible restent inchangées, c'est-à-dire en ignorant le problème du décalage conditionnel. Des travaux récents ont cependant montré qu'un tel problème de décalage conditionnel existe et peut entraver le processus d'adaptation. Pour résoudre ce problème, il est nécessaire d'exploiter les données étiquetées du domaine cible, mais la collecte de données étiquetées peut être coûteuse et peut prendre du temps. À cette fin, nous introduisons une approche d'apprentissage actif discriminant pour l'adaptation de domaine permettant de réduire les efforts d'annotation de données. Plus précisément, nous proposons un apprentissage actif adversarial de réseaux de neurones en trois étapes : apprentissage invariant de l'espace de caractéristiques (première étape), critères d'incertitude et de diversité et leur compromis pour la stratégie d'interrogation (deuxième étape) et réentraînement avec les étiquettes cibles interrogées (troisième étape). Des comparaisons empiriques avec les méthodes d'adaptation de domaine existantes sur quatre ensembles de données de référence démontrent l'efficacité de l'approche proposée. De plus, en comparant différentes stratégies d'interrogation, nous avons pu démontrer les avantages de notre méthode.

Abstract

Domain Adaptation aiming to learn a transferable feature between different but related domains has been well investigated and has shown excellent empirical performances. Previous works mainly focused on matching the marginal feature distributions using the adversarial training methods while assuming the conditional relations between the source and target domain remained unchanged, *i.e.*, ignoring the conditional shift problem. However, recent works have shown that such a conditional shift problem exists and can hinder the adaptation process. To address this issue, we have to leverage labelled data from the target domain, but collecting labelled data can be quite expensive and time-consuming. To this end, we introduce a discriminative active learning approach for domain adaptation to reduce the efforts of data annotation. Specifically, we propose three-stage active adversarial training of neural networks : invariant feature space learning (first stage), uncertainty and diversity criteria and their trade-off for query strategy (second stage) and re-training with queried target labels (third stage). Empirical comparisons with existing domain adaptation methods using four benchmark datasets demonstrate the effectiveness of the proposed approach. Furthermore, by comparing different query strategies, we demonstrate the benefits of our method.

3.1 Introduction

As introduced in Chapter 1, this thesis aims at exploring the transferable features from the different domains. One approach is the *Domain Adaptation* (DA), which aims to improve the learning performance of a target domain by leveraging the unlabeled data in the target domain as well as the labeled data from a different but related domain (source domain). As introduced in section 1.5.1, most recent DA advancements (Ganin et al., 2016; Redko et al., 2017; Shen et al., 2018) are mostly based on the basic *Covariate Shift* assumption (Redko et al., 2019) that the marginal distributions of source and target domain change ($\mathcal{D}_{\mathcal{S}}(\mathbf{x}) \neq \mathcal{D}_{\mathcal{T}}(\mathbf{x})$) while the conditional distribution (predictive relation) is preserved ($\mathcal{D}_{\mathcal{S}}(y|\mathbf{x}) = \mathcal{D}_{\mathcal{T}}(y|\mathbf{x})$) during the adaptation process, where y refers to the corresponding label.

However, some recent works have revealed that this assumption may not hold, and in this case, one may still need some labeled data from the target domain in order to successfully transfer information from one domain to another. Specifically, Zhao et al. (2019a) discussed the conditional shift problem showing that such a problem exists and can hinder the adaptation process. Recall that conditional shift refers to the situation that ($\mathcal{D}_{\mathcal{S}}(y|\mathbf{x}) \neq \mathcal{D}_{\mathcal{T}}(y|\mathbf{x})$). Zhao et al. (2019a) proved that the risk on target domain is controlled by the source risk, the marginal distribution divergence, and disagreement between the *two labeling distributions* :

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + d_{\hat{\mathcal{H}}}(\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}) + \underbrace{\min\{\mathbb{E}_{\mathcal{D}_{\mathcal{S}}}[|f_{\mathcal{S}} - f_{\mathcal{T}}|], \mathbb{E}_{\mathcal{D}_{\mathcal{T}}}[|f_{\mathcal{S}} - f_{\mathcal{T}}|]\}}_{\text{Impossible to measure in unsupervised DA}} \tag{3.1}$$

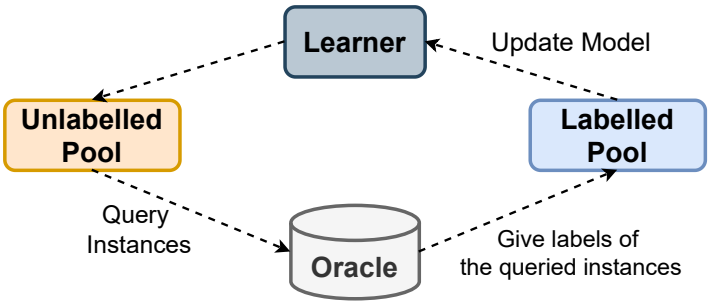


FIGURE 3.1 – General workflow of active learning : typically the learner is trained with the labelled pool and unlabelled pool. In the domain adaptation setting, we regard the labelled pool as source domain while the unlabelled pool as the target domain. During training, the learner can query some instances (a limited budget) from the oracle, which is regarded as an expert who can give the labels of the queried unlabelled instances. After the oracle return the labels of the queried instances, those data are pushed into the labelled pool for training and updating the model again. Figure referred from (Ren et al., 2020).

Here $R_{\mathcal{T}}(h)$, $R_{\mathcal{S}}(h)$ and f refer to target risk, source risk and labeling function, respectively. We will formally define the notations in section 3.2. In a typical *unsupervised DA* setting, it is not possible to measure the third term in Eq. 3.1. Besides, as summarized in Table 1.4, no matter which distribution divergence we used, there is always a non-observable term that relies on the labelling function between the source and target domain.

One possible way to measure this term is to query some data labels from target domain so that the learner can learn the conditional relations in the target domain. However, the label annotations usually is expensive. Notice that the convergence rate at the disagreement term would generally be $\mathcal{O}(1/\sqrt{N_t})$ (Mohri et al., 2018) with *slow* convergence behaviour if the label is *i.i.d.* sampled from the target set with size N_t, which is far sufficient to minimize the last term.

To alleviate such difficulties, one can use *Active Learning* (AL) (Settles, 2009) technique for DA so that the learner can reduce the cost of acquiring labels by requesting labeling from the oracle. We show a general active learning process in Fig. 3.1. AL only tries to query the labels of the most informative examples, and has been shown, in some optimal cases, to achieve *exponentially-lower label-complexity* (number of queried labels) than passive learning (Cohn et al., 1994). From this perspective, we tried to break the general *i.i.d.* sampling with limited information in the target domain (*a.k.a* semi-supervised domain adaptation approach).

Aiming to address all the aforementioned issues, we proposed a three-stage discriminative active domain adaptation algorithm, which aims to actively query the most informative instances in the target domain to minimize the labeling disagreement term, under the same and small querying label budget.

In the first stage, we adopted the Wasserstein Distance-based adversarial training tech-

nique (Redko et al., 2017; Shen et al., 2018) for unsupervised DA through training a critic function for learning the domain invariant feature. The critic is also used to discriminate the target domain features for active querying. In the second stage, we derived a sample-efficient and straightforward active query strategy based on the network structure, for sampling the most *informative* samples in the target domain by controlling *uncertainty* and *diversity* for selecting the target instances. Finally in the third stage, we deployed a re-weighting technique based on the prediction uncertainty for determining the importance of queried samples to retrain the network.

To summarize, our contributions in this chapter are two-folds :

1. We theoretically analyzed the conditional shift problem in domain adaptation using the Wasserstein distance and provide an active query strategy to migrate the disagreement term between the source and target domain
2. Based on the previous theoretical analysis, we then proposed the active query strategy based on the Wasserstein critic and model classifier without requiring extra computations.

We then implemented extensive experiments on four benchmark datasets. The empirical results showed that our proposed algorithm improves the classification accuracy with a small query budget. When the query budget is small, the proposed approach can have better performance than its *i.i.d* (random) selection counterparts (reported in Table 3.5). Furthermore, the comparison with other query strategy based DA baselines also demonstrates the effectiveness of our algorithm.

3.2 Problem Setup

For the notations and basic definitions, we follow the notations introduced in section 1.3.1. More specifically, we consider a classification task and denote $\mathcal{X}$ and $\mathcal{Y}$ as the input and output space. A learning algorithm is then provided with a *labeled source dataset* $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_s}$ consisting of N_s examples drawn *i.i.d.* from $\mathcal{D}_{\mathcal{S}}(\mathbf{x}, y)$ and an unlabeled target dataset $T = \{\mathbf{x}_j\}_{j=1}^{N_t}$ consisting of N_t examples drawn *i.i.d.* from $\mathcal{D}_{\mathcal{T}}(\mathbf{x})$, where $\mathcal{D}_s(\mathbf{x}, y)$ is the joint distribution on $\mathbf{x} \times y$ and $\mathcal{D}_{\mathcal{T}}(\mathbf{x})$ is the marginal target distribution on $\mathbf{x}$, respectively. The expected source and target risk of $h \in \mathcal{H}$ over $\mathcal{D}_{\mathcal{S}}$ (respectively, $\mathcal{D}_{\mathcal{T}}$), are the probabilities that h errs on the entire distribution $\mathcal{D}_{\mathcal{S}}$ (respectively, $\mathcal{D}_{\mathcal{T}}$) : $R_{\mathcal{S}}(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_{\mathcal{S}}}\ell(h(\mathbf{x}), y)$ and $R_{\mathcal{T}}(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}\ell(h(\mathbf{x}), y)$, where $\ell(\cdot)$ is the loss function. The goal of DA is to build a classifier $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ training on source domain with a low *target risk* $R_{\mathcal{T}}(h)$.

3.2.1 Optimal Transport and Wasserstein Distance

As introduced in section 1.5.1, we adopted the OT and Wasserstein adversarial training method, which is implemented to align the feature distribution for the unsupervised domain adaptation stage (first stage). OT can constrain labelled source samples from the same category to keep

close with each other during the transportation process (Courty et al., 2016), which is helpful to alleviate the semantic misalignment problem during the adversarial training process (Zhou et al., 2021b). Besides, compared with some other information theoretical metrics, such as KL divergence, which is not capable to measure the inherent geometric relations among the different domains (Arjovsky et al., 2017), OT is capable to exactly measure their corresponding geometry properties of each domain. Furthermore, compared with $\mathcal{H}$ divergence (Ben-David et al., 2010a), Wasserstein distance has better gradient property. As discussed in section 1.5.1 and Appendix A.2, $\mathcal{H}$ divergence is dependent on the hypothesis class and has a high VC dimension, which made it is difficult to optimize using neural network. On the contrary, the OT and Wasserstein distance can be efficiently computed using the Kantorovich-Rubinstein duality. Furthermore, $\mathcal{H}$ divergence can only deal with the marginal distribution $\mathcal{D}(x)$ while the OT and Wasserstein distance can handle the label information $\mathcal{D}(y)$ (Arjovsky et al., 2017), thus it can deal with the joint distribution $\mathcal{D}(x, y)$, which then leads to the promising generalization bound (Redko et al., 2017).

For the OT and Wasserstein adversarial training method, we follow Redko et al. (2017) and define $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ as the cost function for transporting one unit of mass $\mathbf{x}$ to $\mathbf{x}'$, then Wasserstein Distance is computed by

$$W_p^p(\mathcal{D}_i, \mathcal{D}_j) = \inf_{\gamma \in \Pi(\mathcal{D}_i, \mathcal{D}_j)} \int_{\mathcal{X} \times \mathcal{X}} c(\mathbf{x}, \mathbf{x}')^p d\gamma(\mathbf{x}, \mathbf{x}')$$

where $\Pi(\mathcal{D}_i, \mathcal{D}_j)$ is joint probability measures on $\mathcal{X} \times \mathcal{X}$ with marginals $\mathcal{D}_i$ and $\mathcal{D}_j$ referring to all the possible coupling functions. Throughout this paper, we shall use Wasserstein-1 distance only ($p = 1$). According to *Kantorovich-Rubinstein* theorem (see Appendix A.2), let f be a Lipschiz-continous function $||f||_L < 1$, we have

$$W_1(\mathcal{D}_i, \mathcal{D}_j) = \sup_{||f||_L < 1} \mathbb{E}_{\mathbf{x} \in \mathcal{D}_i} f(\mathbf{x}) - \mathbb{E}_{\mathbf{x}' \in \mathcal{D}_j} f(\mathbf{x}') \tag{3.2}$$

In practice, we can implement a deep neural network to approximate function f. Then, computing the sup in Eq. 3.2 is to the maximum of Wasserstein distance by arg max operator through the general deep neural network optimizer (*e.g.* SGD (Bottou, 2010) or Adam (Kingma and Ba, 2015) optimizer). This allows us to compute the Wasserstein distance efficiently and the complexity $w.r.t$ $f(x)$ is only $\mathcal{O}(n + m)$.

3.2.2 Conditional Shift and Error Bound

As stated before, traditional DA researches (Ben-David et al., 2010a; Ganin et al., 2016; Tzeng et al., 2017) typically assumed that the conditional relationships remain unchanged during the adaptation process. From a probabilistic perspective, the general learning process of most previous DA approaches is to learn the joint distribution of the target domain $\mathcal{D}_\mathcal{T}(\mathbf{x}, y)$ through source domain joint distribution $\mathcal{D}_\mathcal{S}(\mathbf{x}, y)$. Note that $\mathcal{D}_\mathcal{T}(\mathbf{x}, y) = \mathcal{D}_\mathcal{T}(\mathbf{x}|y)\mathcal{D}_\mathcal{T}(\mathbf{x})$, to

guarantee a successful transfer from source domain S to target domain T, the underlying assumption is $\mathcal{D}_{\mathcal{S}}(y|\mathbf{x}) \neq \mathcal{D}_{\mathcal{T}}(y|\mathbf{x})$.

For the conditional shift situation, $\mathcal{D}_{\mathcal{S}}(y|\mathbf{x}) \neq \mathcal{D}_{\mathcal{T}}(y|\mathbf{x})$, Zhao et al. (2019a) theoretically showed that such a conditional shift problem exists in many situations and that typically if we only try to minimize the source error together with the domain distances, the target error might increase, which shall hinder the adaptation process. Their analysis was based on $\hat{\mathcal{H}}$ divergence, which is somehow hard to compute in deep learning based methods. In order to be coherent with our work, we now present the error bound using Wasserstein Distance with the following Theorem 3.1.

Theorem 3.1. *Let $\langle \mathcal{D}_{\mathcal{S}}, f_s \rangle$ and $\langle \mathcal{D}_{\mathcal{T}}, f_t \rangle$ be the source and target distributions and corresponding labeling function, if the hypothesis h is 1-Lipschtiz and the loss function is $0-1$ loss, then we have*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + 2W_1(\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}) + \mathbb{E}_{\mathcal{D}_{\mathcal{S}}}\big[|f_s - f_t|\big] \tag{3.3}$$

The proof is illustrated in the Appendix B.1. This theorem showed that error on the target domain is bounded by source domain error, Wasserstein Distance between source and target, and the conditional distribution on both source and target domains. Here the third term is not measurable in the unsupervised domain adaptation setting. If the conditional distribution changes during the adaptation process, then the target error may diverge (Zhao et al., 2019a). One direct approach to reduce the disagreement between f_s and f_t is to partially acquire the labeling function f_t, *i.e.*, the labels in the target domain.

Besides, the Wasserstein distance between the source and target distribution (second term in Eq. 3.3), is measured by total transportation cost between the source and target domain. Denote $\mathcal{D}_U$ and $\mathcal{D}_L$ by the corresponding distributions of unlabeled and labeled datasets, then the Wasserstein distance is denoted by :

$$W_1(\mathcal{D}_U, \mathcal{D}_L) = \inf_{\gamma \in \Pi(\mathcal{U}, \mathcal{L})} \int_{\mathcal{X} \times \mathcal{X}} c(\mathbf{x}_l, \mathbf{x}_u) d\gamma(\mathbf{x}_l, \mathbf{x}_u)$$

Intuitively, if we can query some instances in the target domain $\mathcal{T}$ ($\mathcal{D}_U$) and move them from target into the source domain $\mathcal{S}$ ($\mathcal{D}_L$), we can reduce the total transportation cost between the two domains, *i.e.*, the Wasserstein distance between the two domains.

Based on this, minimizing the RHS of Eq. 3.3 is equivalent to train a learner $h \in \mathcal{H}$ that :

1. Minimize the source error ;
2. Train a critic to estimate the empirical Wasserstein Distaince between the source and target domain and approximately find a feature extractor that can minimize the total transportation cost between the source and target domain in an adversarial way with the critic ;

3. Can query the labeling information in the target domain so that to minimize the disagreement of labeling function between the source and target domain *i.e.*,the third term of Eq. 3.3.

To train this kind of learner, we argue that if it can actively query labeling information in the target domain, then, it can partially get the conditional information in the target domain. With the minority of labeled target instances in hand, it can learn to jointly minimize the error both on the source and target domain. Furthermore, to *i.i.d.* query the label is somehow slow. In order to reduce the annotation expense, we may expect the learner to query some informative instances using an active learning strategy. Also, if the queried instances in the target domain are informative enough, they will have a better representative property on the target domain. Then, the learner can have better generalization performance on the target domain.

We then can bound the error after such active selection as referred by the following theorem,

Corollary 3.1. *Assume the learner has a budget β of total target samples to query the oracle for ground truth label. Let $\mathcal{X}_s$ and $\mathcal{X}_t$ be two sample sets with size N_s and N_t drawn i.i.d. from $\mathcal{D}_\mathcal{S}$ and $\mathcal{D}_\mathcal{T}$ respectively. Let $\hat{\mathcal{D}}_\mathcal{S} = \frac{1}{N_s}\sum_{i=1}^{N_s}\Delta_{x_i}^s$ and $\hat{\mathcal{D}_\mathcal{T}} = \frac{1}{N_t}\sum_{i=1}^{N_t}\Delta_{x_i}^t$ be the associated empirical measure. Then $\forall d' \geq d$ and $\lambda' < \lambda$ there exists some constant N_0 depending on d' such that for any $\delta > 0$ and $\min(N_s, N_t) \geq N_0 \max(\delta^{-(d'+2),1})$ with probability at least $1-\delta$ for all hypothesis $h \in \mathcal{H}$, then, the following holds,*

$$\begin{aligned} R_\mathcal{T}(h) \leq & R_\mathcal{S}(h) + 2W_1(\hat{\mathcal{D}}_\mathcal{S}, \hat{\mathcal{D}}_\mathcal{T}) + \mathbb{E}_{\mathcal{D}_\mathcal{S}}\left[|f_s - f_t|\right] \\ & + 2\sqrt{2\log(\frac{1}{\delta})/\lambda'}\Big(\sqrt{\frac{1}{N_s+\beta N_t}} + \sqrt{\frac{1}{N_t-\beta N_t}}\Big) \end{aligned} \tag{3.4}$$

Note that N_s and N_t are constant since they are just depend on the source and target domains. Corollary 3.1 extend Theorem 3.1 to the empirical errors and is free of the joint optimal errors between the source and target domain. We also notice a latter work (Li et al., 2021b) further investigates this kind of empirical error bound in the context of invariant risk and feature learning, which also indicates the necessity to query the target labels for empirical minimization.

Take those above into consideration, we can formally present the discriminative active domain adaptation method.

3.3 Active Discriminative Domain Adaptation

The learning process mainly consists of three main stages. The three-stage scheme is designed to firstly implement adversarial training to learn domain invariant features through OT. The second stage is to actively query the most informative instances on the invariant feature space.

Finally, those informative instances are used for retraining the network to reinforcing the importance of the target features. We now introduce the three-stages in details.

3.3.1 Stage 1 : Domain Adversarial Training via Optimal Transport

For the first stage, we adopt *Wasserstein Distance Guided Representation Learning* (Shen et al., 2018) method for adversarial training. The network receives a pair of instances from the source and target domain. Denoted by F and C the feature extractor and classifier functions, parameterized by $\boldsymbol{\theta}_f$ and by $\boldsymbol{\theta}_c$, respectively. The feature extractor is trained to learn invariant features, and the classifier is expected to learn the conditional prediction relations $\mathbb{P}(Y|\mathbf{X})$ for predicting the instances from both source and target domain correctly. For the classification loss, we employ the traditional cross-entropy loss : $\mathcal{L}_{cls} = -\sum_{i=1}^{m} y_i \log(\mathbb{P}(C(F(\mathbf{x}_i))))$.

Then, there follows the domain critic network ϕ, parameterized by θ_ϕ. It estimates the empirical Wasserstein Distance between the source and target domain through a pair of batched instances $\mathbf{X}_\mathcal{S}$ and $\mathbf{X}_\mathcal{T}$,

$$W_1(\mathbf{X}_\mathcal{S}, \mathbf{X}_\mathcal{T}) = \frac{1}{n_s} \sum_{\mathbf{x}_s \in \mathbf{X}_\mathcal{S}} \phi(F(\mathbf{x}_s)) - \frac{1}{n_t} \sum_{\mathbf{x}_t \in \mathbf{X}_\mathcal{T}} \phi(F(\mathbf{x}_t)) \tag{3.5}$$

The feature extractor F is then trained to minimize the estimated Wasserstein Distance in an adversarial manner with the critic ϕ. Then, goal of first stage training is described by

$$\min_{\theta_f, \theta_c} \max_{\theta_d} \mathcal{L}_{cls} + \lambda_w (W_1(\mathbf{X}_\mathcal{S}, \mathbf{X}_\mathcal{T}) - \mathcal{L}_{grad}) \tag{3.6}$$

where λ_w is a trade-off coefficient and $\mathcal{L}_{grad}$ is the gradient penalty term suggested by Gulrajani et al. (2017). When computing the gradient of such loss function, we use the gradient penalty method suggested in Gulrajani et al. (2017) which can help to prevent gradient vanishing or exploding problems caused by weight clipping.

$$\mathcal{L}_{grad} = (||\nabla_{F(X)} \phi(F(\mathbf{x}))||_2 - 1)^2 \tag{3.7}$$

Note that the 'min-max' computations in Eq. 3.6 is achieved by the adversarial training process (see Appendix A.3). The source and target features (marginal distributions) can be aligned via such an adversarial training process (Eq. 3.6). Then, based on this aligned marginal distribution, we can implement the active strategy to query the most informative target instances

3.3.2 Stage 2 : Active Query with Wasserstein Critic

For the second stage, we aim to have an active learner that can find out the most informative features among the unlabeled target so that it can leverage from the labeling information of

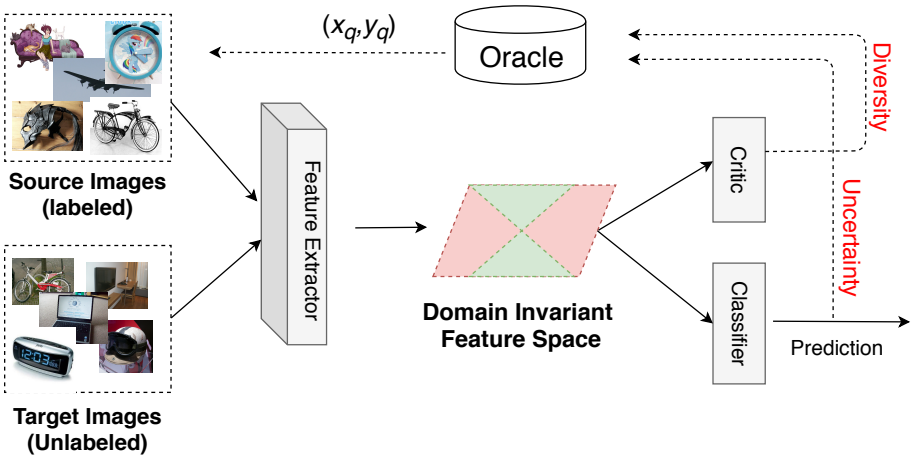


FIGURE 3.2 – Ac-DA workflow : feature extractor are trained to learn a domain invariant feature space together with the critic. The learner selects the informative instances by measuring *uncertainty* and *diversity* based on critic and classifier outputs.

the target domain. The informative features, intuitively, are the *ones that are most different from what the learner has already dealt with*. The hardest instances to adapt are those with least confidence (*i.e.* the most uncertain ones) to predict based on current classifier. As pointed out in previous work (Dasgupta, 2011), only focusing on the uncertainty might lead to the *sampling bias*. In order to reduce this bias, the active learner shall also search a diversity in target samples. We therefore find the most informative target samples holding both uncertainty and diversity properties.

Prediction Uncertainty The conditional prediction $\mathbb{P}_T(Y|\mathbf{X})$ is learned by the classification network. To measure the uncertainty, we can borrow the idea from Long et al. (2018) to adopt entropy measure to quantify the uncertain of the classifier. The uncertainty entropy measure over an instance $\mathbf{x}_t$ is denoted by

$$\mathcal{U}(y_t|\mathbf{x}_t) = \mathcal{H}(\hat{\mathbb{P}}(y_t|\mathbf{x}_t)) \tag{3.8}$$

where $\mathcal{H}(\cdot)$ is the information entropy measure, $\hat{\mathbb{P}}(y_t|\mathbf{x}_t)$ is the output of classification network $\hat{\mathbb{P}}(y_t|\mathbf{x}_t) = C(F(\mathbf{x}_t))$.

Diversity by Critic Function If some instances, in terms of distribution distance measures, are very far from the unknown labeled ones, then they should contain most informative and diverse features from the known labelled ones. Recall that in the first stage, we match the marginal distribution between the source and target domain to achieve a domain invariant feature space with Wasserstein Distance. Then, for the target domain instances, the one with highest critic score is the one that have the highest transportation cost.

Sinha et al. (2019) and Shui et al. (2020b) showed that such critic term $\phi(F(\cdot)) : \mathcal{X} \to [0, 1]$ indicates the diversity in the query process. Then, we can leverage from the trained Wasserstein

Algorithm 1 The Active Discriminative Domain Adaptation (AcDA) Algorithm.

Input : Source and target domain input S, T ; Query budget β
Parameter : Feature extractor θ_f ; Classifier θ_c ; Critic θ_d.
Output : Optimized $\theta_f^\star$, $\theta_c^\star$, $\theta_d^\star$.

while Domain level adaptation not finish **do**
  Sample batches $(\mathbf{x}_s, y_s) \sim S$, $\mathbf{x}_t \sim T$;
  Train the network based on Eq. 3.6 until converge ;
end while ;
if Query budget is not empty **then**
  Select the target instances $\{\mathbf{x}_1^q, \ldots, \mathbf{x}_{N_q}^q\}$ according to Eq. 3.9 and query the label $\{y_1^q, \ldots, y_{N_q}^q\}$ from oracle ;
else
  Update the dataset $Q = \{(\mathbf{x}_1^q, y_1^q), \ldots, (\mathbf{x}_{N_q}^q, y_{N_q}^q)\}$, $S' = S \cup Q$, $T' = T/Q$;
end if
Compute the uncertainty vector $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_C]_{j=1}^C$ with Eq. 3.10
Train the network on new labeled and unlabeled dataset via domain adaptation techniques with Eq. 3.11.
return solution

Critic network to evaluate and find out the most informative (diverse) target features on the invariant feature space. That is, measuring the diversity of target instances via critic score. Consider the critic output of a target instance $\mathbf{x}_t$, if $\phi(F(\mathbf{x}_t)) \to 1$, then $\mathbf{x}_t$ is far, *w.r.t.* Wasserstein Distance, from the source domain images and if $\phi(F(\mathbf{x}_t)) \to 0$, then $\mathbf{x}_t$ is near to the source images.

Based on those above, if we hope to find out the most informative (uncertain and diverse) instances in the target domain, then we should query by controlling two terms :

— uncertainty score $\mathcal{U} = \mathcal{H}(\hat{\mathbb{P}}(y_t|\mathbf{x}_t))$ defined by Eq. 3.8, which is indicates the uncertainty of the classifier to predict a label $y_t^?$ given the instance $\mathbf{x}_t$ in the target domain

— critic score $\phi(F(\mathbf{x}_t))$ by the the Wasserstein critic function, which indicates the diversity of the unlabeled target instance compared with the source labeled ones.

Then, we can have the following objective

$$\text{argmax}_{\mathbf{x}_t \in \mathcal{X}_t} \mathcal{U}(y_t^?|\mathbf{x}_t) - \lambda_{div}\phi(F(\mathbf{x}_t)) \tag{3.9}$$

where λ_{div} is a coefficient to regularize the Wasserstein critic term. So, for a query budget β and N_t of target set instances, the query process can be described as : *looking for $N_q = \beta N_t$ instances by solving Eq. 3.9 and query the labels of those N_q instance from the oracle.* Denote the queried set by $Q = \{(\mathbf{x}_1^q, y_1^q), \ldots, (\mathbf{x}_{N_q}^q, y_{N_q}^q)\}$. Then, uniting such small batch instances with the source domain and removing them from the target domain. The source and target datasets shall be updated as : $S' = S \cup Q$, $T' = T/Q$. We illustrate a general query workflow in Fig. 3.2.

3.3.3 Stage 3 : DA training with new dataset

The goal of our method is to leverage the most informative instances in the target domain to reinforce the adaptation process. General adversarial training methods (see section 2.2.3) for domain adaptation usually assign each instance with the same importance weight. In order to enforce the uncertainty information to the classifier, we hope to give higher weights to the instances with higher uncertainty scores during the supervised classification process.

Denote by a set of N_q queried instances $\{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{N_q}$, we shall re-weight the importance of each instance class based on its uncertainty score. Denote by uncertainty vector $\boldsymbol{\alpha} = [\alpha_1 \dots \alpha_j \dots \alpha_C]_{j=1}^{C}$ over all C classes. For each class j, the weight is computed by,

$$\alpha_j = \frac{N_j \cdot \mathcal{U}(y_j|\mathbf{x})}{\sum_{i=1}^{N_q} \mathcal{U}(y^{(i)}|\mathbf{x})} \tag{3.10}$$

where N_j is the number of instances with label y_j and $\mathcal{U}(\cdot)$ is the uncertainty score defined in Eq.3.8. For a batch of queried instances, the weighted cross-entropy loss is computed by

$$\mathcal{L}_w^q = \alpha_j(-\sum_{j=1}^{C} y_j \times \log \mathbb{P}(y_j|\mathbf{x}))$$

Then, objective function for the third stage is,

$$\min_{\boldsymbol{\theta}_f, \boldsymbol{\theta}_c} \max_{\boldsymbol{\theta}_\phi} \mathcal{L}_w^q + \mathcal{L}_{cls} + \lambda_w(W_1(\mathcal{X}'_{\mathcal{S}}, \mathcal{X}'_{\mathcal{T}}) - \mathcal{L}_{grad}) \tag{3.11}$$

where $\mathcal{X}'_{\mathcal{S}}$ and $\mathcal{X}'_{\mathcal{T}}$ are sampled from the updated source and target datasets, $\mathcal{L}_{cls}$ is the classification loss on the original source set and $\mathcal{L}_w^q$ is the weighted loss for the query set. Finally, we illustrate our Active Discriminative Domain Adaptation (Ac-DA) algorithm in Algorithm 1

3.4 Experiments and Results

In the experiments part, we aim to demonstrate the following aspects :

— Firstly, we would like to show that even randomly select a few amount of labelled target data can improve the performance compared with the unsupervised counterparts.

— Secondly, we show that compared with the semi-supervised method, *i.e.*, the random (*i.i.d.*) selection, the active learning query strategy can have more benefits.

— Lastly, confirming the effectiveness of our method comparing with another active domain adaptation method, and also other query strategies with certain query budget for further analysis.

For the comparison with unsupervised DA methods, we evaluate the performance of the proposed algorithm on four benchmark datasets and compared with some other approaches :

Method	M → MM	M → U	U → M	avg.
LeNet5	56.1	67.4	65.3	60.3
DANN	74.2	77.1	73.2	74.6
WDGRL	80.3	81.1	74.2	76.2
ADDA	78.9	83.5	82.3	81.5
Rand.	92.4	**95.7**	95.8	94.7
Ac-DA	**95.4**	95.5	**96.5**	**95.6**

TABLE 3.1 – Classification accuracy (%) on **digits datasets** with different adaptation tasks. The last two line are our method, Random refers to randomly query some instance while Ac-DA is the proposed approach. Both two methods are restrict to 10% query budget.

Method	A → W	A → D	D → A	W → A	avg.
ResNet50	68.6	69.3	61.1	60.7	64.9
DAN	80.5	78.6	63.6	60.7	62.7
DANN	81.3	79.2	68.2	67.4	74.0
WGDRL	79.2	80.2	69.3	69.1	74.5
Rand.	86.1	85.6	76.3	78.1	81.6
Ac-DA	**86.6**	**87.7**	**78.5**	**80.2**	**83.3**

TABLE 3.2 – Classification accuracy (%) on **Office-31** dataset with different adaptation settings with 10% query budget.

Wasserstein Guided Domain Adaptation (WDGRL (Shen et al., 2018)), Domain Adversarial Neural Networks (DANN (Ganin et al., 2016)), Adversarial Discriminative Domain Adaptation (ADDA (Tzeng et al., 2017)) and Conditional Adversarial Domain Adaptation (CDAN (Long et al., 2018)). In order to show the benefits of active query method, we also compare the results with random selection process (Rand.) when the query budget is the same. All experiments are programmed by *PyTorch*.

3.4.1 Datasets and Implementations

We test our proposed algorithm on four benchmark datasets.

Digits Datasets : We test our algorithm on digits datasets with the experiments setting : USPS (U)↔ MNIST (M) and MNIST → MNIST-M (MM). For USPS we resize the images to size 28×28. We train the network using training sets with size : MNIST/MNIST-M($60k$), USPS($7,291$) and testing sets with size : MNIST/MNIST-M ($10k$), USPS($2,007$).

Method	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	avg.
ResNet50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DANN	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
WGDRL	42.6	57.9	69.3	47.3	59.5	63.4	46.2	41.3	67.4	62.4	52.8	74.9	57.1
CDAN	49.0	69.2	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
Rand.	56.9	76.4	76.3	**61.7**	78.1	73.3	57.8	**56.9**	74.2	68.5	60.3	83.2	68.6
Ac-DA	56.8	80.3	80.8	67.2	80.0	78.4	64.8	57.5	80.1	75.9	62.8	88.7	**72.7**

TABLE 3.3 – Classification accuracy (%) on **Office Home** dataset with different adaptation settings with query budget 10%.

Method	C → I	C→P	I → P	I → C	P → C	P → I	avg.
ResNet50	76.4	62.5	73.2	89.3	90.3	79.8	78.5
DANN	84.8	72.6	73.8	92.8	91.5	81.9	82.9
WDGRL	82.3	70.8	73.9	90.7	91.3	85.4	82.4
Rand.	89.8	75.0	78.2	94.4	**94.9**	89.9	87.1
Ac-DA	**91.1**	**76.3**	**80.8**	**96.7**	94.7	**94.2**	**88.9**

TABLE 3.4 – Classification accuracy (%) on **Image-CLEF dataset** with different adaptation tasks under 10% query budget.

Office-31 dataset is a standard benchmark for domain adaptation evaluations. It contains three different domains : Amazon (A), Dslr (D) and WebCam (W), with 31 categories in each domain. We report the average results in Table 3.2.
Office Home dataset : is more challenging than Office-31, contains four different domains : *Art* (Ar), *Clipart* (Cl), *Prodcut* (Pr) and *Real World* (Rw), with 65 categories in each domain. We report the average results in Table 3.3.
Image-CLEF 2014 dataset contains three domains, which are *Caltech-256*(C), *ILSVRC-2012*(I), and PascalVOC-2012(P), with 12 common shared catagories. We report the average results in Table. 3.4

For digits datasets, we do not apply any data-augmentation. For Office-31, Office-Home and Image-CLEF datasets, we follow the previous DA approaches (Long et al., 2018; You et al., 2019), and apply the following pre-processing pipline : 1) for training set, firstly resize the image to 256×256 then, apply *RandomCrop* downgrade the size to 224×224, after that, apply the same random flipping strategy of You et al. (2019) ; 2) for testing set, resize the images to 256×256 then use *CenterCrop* to size 224×224.

CNN Architecture and Implementations

For digits experiments, similar to the previous DA works (Wen et al., 2019b, 2020), we adopt *LeNet-5* (LeCun et al., 1998) as feature extractor and trained from scratch. For the rest three real-world datasets, we implement ImageNet pretrained *ResNet-50* as feature extractor. For the digits experiments, we train the network with mini-batch size 64 and for the rest three datasets with mini-batch size 16. We adopt Adam optimizer for training the network. For the digits dataset, we empirically set the learning rate as 10^{-3} and also enable the *weight-decay* in Adam optimizer. For the Office-31, Office-home and Image-CLEF datasets, we also enable the *weight-decay* in Adam optimizer. For stable training, we set $\lambda_w = \frac{2}{1+\exp(-10 \cdot p)} - 1$, and p is the training progress. This setting of λ_w has been used in many previous adversarial training approaches (Wen et al., 2019b; Dou et al., 2019; Long et al., 2018), and has been shown can stablize the adversarial training process. Also, we empirically set $\lambda_{div} = 10$ which can demonstrate good performances. To avoid over-training, we also adopt early-stopping technique. All hyper-parameters are validated and fine-tuned by grid search.

3.4.2 Results and Analysis

We use t-SNE visualizations (Van der Maaten and Hinton, 2008) in Fig. 3.3 since it allows us to check the feature alignment performance of the adaptation method. We show the t-SNE visualizations comparison of non-adaptation setting, and our proposed approach Ac-DA in Fig. 3.3. We can observe that our method (right of Fig. 3.3) has a good alignment performance. We report the average results of our algorithm and baselines using our data pre-processing pipeline on Digits, Office-31, Office-Home and Image-CLEF datasets in Table 3.1, 3.2, 3.3 and

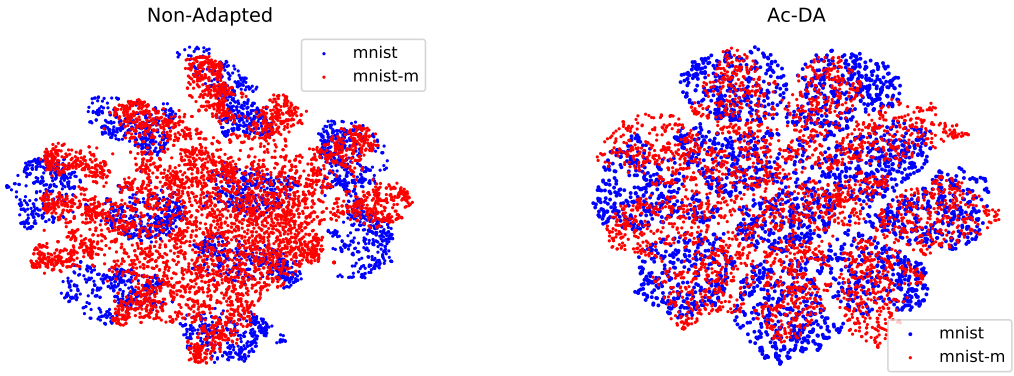


FIGURE 3.3 – t-SNE visualization between our proposed Active Discriminative Domain Adaptation (right, with 5% query budget) and non-adapted setting (left) for MNIST → MNIST-M adaptation task.

	Digits		Office-Home		Image-CLEF	
budget	Rand.	Ac-DA	Rand.	Ac-DA	Rand.	Ac-DA
5%	91.6	92.9(+1.3)	62.4	65.6(+3.2)	82.2	84.9(+2.7)
10%	94.7	95.6(+0.9)	68.6	72.7(+4.1)	87.1	88.9(+1.8)
15%	96.2	96.9(+0.7)	73.9	75.8(+1.9)	89.8	90.4(+0.6)

TABLE 3.5 – Comparison of different query budgets (5%, 10%, 15%) on three datasets. For each query budget, we report the improvements by applying the active query strategy comparing with the random query strategy in the parentheses.

3.4, respectively. In order to show the effectiveness of active query strategy, for a given budget, we also implemented *random (i.i.d.) selction* method to query the labels for comparsion. The name of such implementations are denoted by *Rand.* and *Ac-DA* in each table. In Table 3.5, we also compared the performances under different budgets.

Value of Target Labels

From the tests results on the four benchmark datasets, we observe that to randomly select some instances in the target domain is beneficial to the classification performance on the target domain. Our method is rooted in WDGRL, comparing accuracy performance between the *random selection* with WDGRL we observe improvements on those benchmarks, which confirms the usefulness of label information for adaptation. Also, for each adaptation task on every dataset, we can observe that the proposed Ac-DA algorithm outperforms the random selection method in almost all the tasks. This also confirms that active query can outperform *i.i.d.* selection.

Effectiveness of Active Query

We then compared the performance between active query and random selection. We implement the experiments with different query budgets (with 5%, 10% and 15%), the average on different dataset is reported in Table 3.5. we can observe that the accuracy will increase as the query budget increases. Also, with same query budget, we compare the accuracy of active query and random selection. We can observe that active query method can outperform the random query method with query budget 5% and 10%. That is, *with smaller query budget, the active query strategy can have better performance than random selection.* This confirms the effectiveness of active query strategy. When the query budget goes to 15%, the differences between the random selection and active query become smaller. One interpolation is that as the query budget increase, the more instances in the target domain will be labeled and those most informative ones will be covered with high probability. When the query budget is relatively small, the active strategy can exactly look for the most informative instances rather than uniformly (random) selecting some instances.

Comparison with Different Query Strategies

In order to evaluate the effectiveness of our method, we compare our method with the active domain adaptation baselines.

To the best of our knowledge, Adversarial Active Domain Adaptation (AADA) (Su et al., 2020) is the only similar baseline to our proposed method during we develop our algorithm. AADA is based on the DANN (Ganin et al., 2016) as the adversarial training mode, where the invariant features are learned by fooling the domain discriminator $\phi(\cdot)$ by a binary classification to predict the instances are from source or target domain. Upon the submission of our article (Zhou et al., 2021c) reflected in this chapter, the official codes release of AADA has not yet published. We reproduced the baseline by rigorously following the original implementation while made some adaptations to our setting for a fairer comparison. The original AADA implementation selected certain target instances and retrain the model under a few-shot mode with several query rounds. For a fairer comparison with our proposed method, we reproduced the AADA with the similar setting with ours by selecting certain ratio of instances when the first stage of adversarial training is stable.

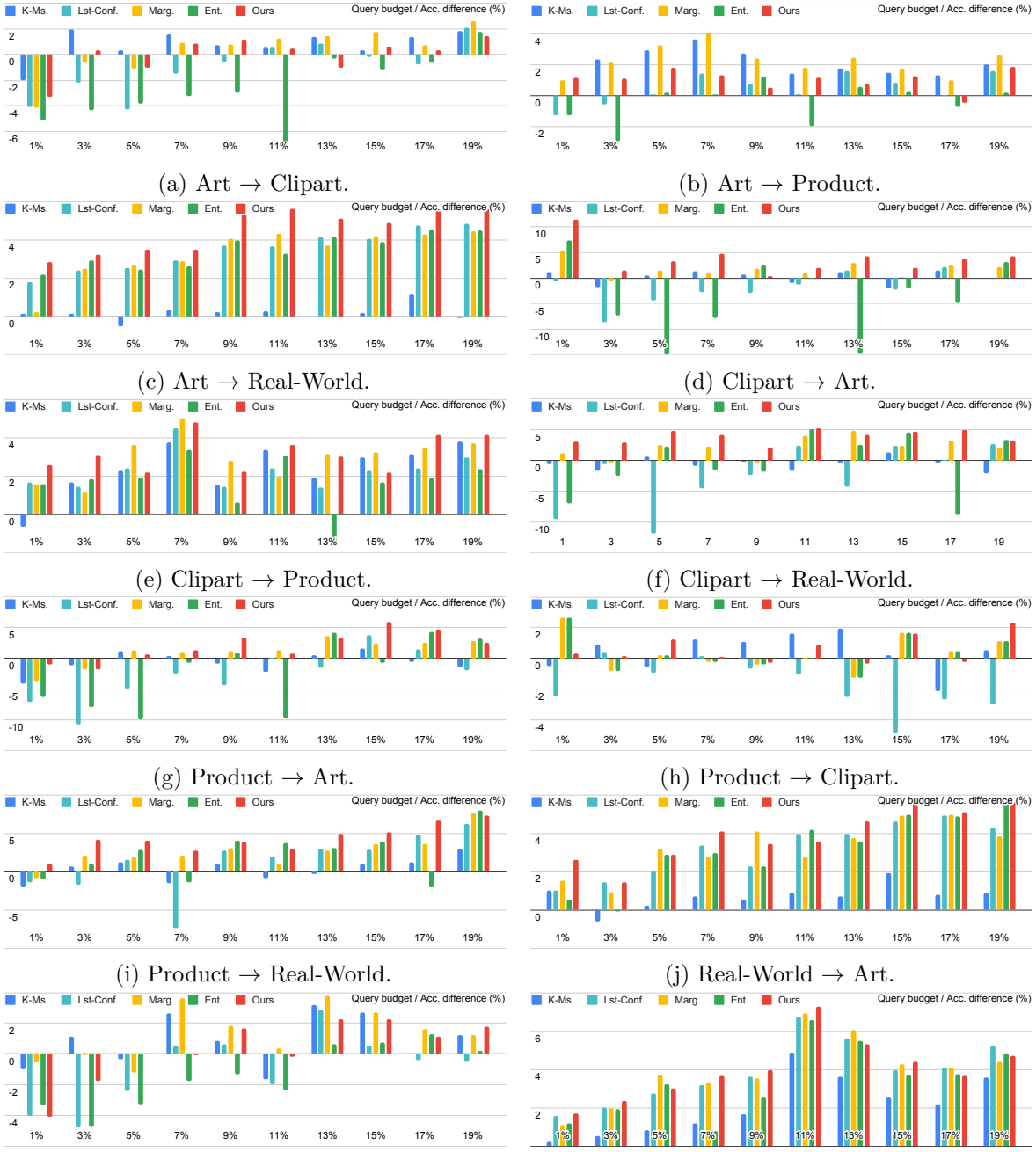
The reproduced AADA model was based on the DANN implementations(see Appendix A.1). We follow Su et al. (2020) to construct the uncertainty cue and diversity cue implementation by scoring the instances with query strategy : $score(x) = \mathcal{H}(\hat{y})w(z)$, where $\hat{y} = C(F(x))$ is the model prediction and $\mathcal{H}(\cdot)$ is the information entropy, while $w(x) = (1 - \phi(z))/\phi(z)$, where $z = F(x)$ is the extracted feature, involving the diversity cue. We compare the performance of re-implemented AADA and our method on Office-home dataset with different query budget and illustrate the performance of each adaptation task average accuracy in Table 3.6. As we can see from Table 3.6, our method outperforms AADA with different query budget.

budget	methods	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	avg.
5%	AADA	41.9	70.9	75.7	58.3	72.6	67.1	56.7	51.5	77.0	71.3	55.8	80.7	65.0
	Ours	40.0	71.5	76.3	62.0	72.8	68.0	56.7	52.1	77.6	72.2	56.2	80.5	**65.5**
10%	AADA	57.8	78.9	78.9	65.9	78.1	77.2	64.1	56.9	79.7	74.7	62.2	88.4	71.9
	Ours	56.8	80.3	80.8	67.2	80.0	78.4	64.8	57.5	80.1	76.0	62.8	88.7	**72.7**
15%	AADA	65.8	81.9	83.5	71.9	82.9	80.9	71.2	65.3	84.3	79.4	69.7	91.1	77.3
	Ours	66.0	84.1	84.8	71.1	83.1	81.8	71.8	64.8	84.9	80.1	69.2	91.3	**77.8**
20%	AADA	68.2	87.6	87.1	73.8	86.1	81.9	72.6	68.2	86.5	82.5	71.8	92.8	79.9
	Ours	68.9	87.4	87.4	74.7	87.2	83.0	73.4	69.1	87.0	83.1	72.6	93.4	**80.6**

TABLE 3.6 – Comparison of our method and the re-implemented AADA with different query budget

	1%	3%	5%	7%	9%	11%	13%	15%	17%	19%
Rand.	57.76	62.61	63.75	67.39	69.99	71.67	72.81	74.42	75.80	76.68
K-Ms.	57.06	62.94	66.13	68.59	70.81	72.13	74.10	75.60	76.60	77.78
Lst-Conf.	55.70	60.80	63.95	67.18	70.33	73.13	74.18	75.92	77.52	78.71
Marg.	58.20	63.14	67.22	69.78	72.05	73.89	75.90	77.17	78.50	79.91
Ent.	56.92	60.68	63.75	66.03	70.50	72.00	72.97	74.85	75.97	77.69
Ours	**59.27**	**64.01**	**67.61**	**69.98**	**72.29**	**74.43**	**75.84**	**77.80**	**79.06**	**80.40**

TABLE 3.7 – Averaged performance of different query strategies on Office home dataset with different query budget (from 1% to 19% of the total instances)



(a) Art → Clipart. (b) Art → Product.

(c) Art → Real-World. (d) Clipart → Art.

(e) Clipart → Product. (f) Clipart → Real-World.

(g) Product → Art. (h) Product → Clipart.

(i) Product → Real-World. (j) Real-World → Art.

(k) Real-World → Clipart. (l) Real-World → Product.

FIGURE 3.4 – Comparison of different query strategies. We take random selection (set the baseline accuracy to 0) as the baseline and report the relative accuracy difference with different query strategies (1% ∼ 19%)

In order to evaluate the effectiveness of our query strategy, we compare the performance of the active domain adaptation algorithm with different query strategies,*i.e.*, we implement some baselines by replacing the query strategy in Eq. 3.9 with the following query strategies :

— Random sampling (Rand.) : randomly select potential instances from the target domain.

— Least confidence (Lst. Conf.) (Culotta and McCallum, 2005) : select the instances with least confidence over the classifier.

— Smallest Margin (Marg.) (Scheffer and Wrobel, 2001) : select the instances via a defined margin.

— Maximum-Entropy sampling (Ent.) (Settles, 2012) : selecting the instances with the maximum entropy, *i.e.*, the most uncertain ones.

— K-Median (K-Ms.) (Sener and Savarese, 2018) : choosing the points to be labelled as the cluster centers of K-Median algorithm

We evaluate the empirical results of the Wasserstein adversarial training with different query strategies and report the overall average of all tasks in Table 3.7. We then report the empirical results on different adaptation task in Fig. 3.4 by choosing random selection as baseline (set as 0) and show the differences. From the empirical results, we observe that our method always outperforms the baseline query strategies under different query budgets *w.r.t.* the averaged accuracy. The most diverse performance occurs on the task Ar $\leftrightarrow$ Cl, this may due to the features look similar with each other, when querying some diverse and uncertain features, it may find some uncommon features which may hurt the learning performance. Besides, we also notice that the performance of entropy sampling diverse a lot. Since entropy sampling means the learner only select the instances with most uncertainty, this may lead the learner to find some strange features, which may hurt the learning performance. Generally, our method has a better averaged performance over all the query strategies under different query budgets.

3.5 Discussion and Conclusion

In this chapter, we presented a three-stage discrimative active algorithm to improve the domain adaptation performance. The first stage adopts a general domain adversarial training. The second stage proposed an end-to-end query strategy combining *uncertainty* and *diversity* criteria to find out the most informative features in the target domain. Finally, the third stage develops a re-weighting technique based on the prediction uncertainty for determining the importance of the queried samples to retrain the network. The empirical results of this three-stage method confirmed the effectiveness of our active domain adaptation algorithm especially when the query budget is small. To this end, we exploited the benefits of optimal transport with Wasserstein distance based adversarial training to achieve the feature alignment as well as the power of the Wasserstein critic to measure the diversity score of the target instances. We then adopt this kind of adversarial training method in the multi-source transfer learning problems, which is introduced in the next chapter.

Chapitre 4

Domain Generalization via Optimal Transport with Metric Similarity Learning

In this chapter, we introduce our work (Zhou et al., 2021b) in which we explored the class similarities across domains for the domain generalization (DG) problem, where the learner has access to labelled data from several source domains but has no target data during training. We adopted the optimal transport based adversarial training methodology for learning invariant features as well as the metric similarity learning objective to explore the class similarities to improve the generalization performance.

Résumé

La généralisation des connaissances à des domaines non vus, où les données et les étiquettes ne sont pas disponibles, est cruciale pour l'apprentissage automatique. Nous abordons ici le problème de la généralisation de domaine pour apprendre de plusieurs domaines sources et généraliser à un domaine cible avec des statistiques inconnues. L'idée consiste à extraire les caractéristiques invariantes sous-jacentes dans tous les domaines. Les approches de généralisation de domaine existantes se sont principalement concentrées sur l'apprentissage de caractéristiques invariantes et sur l'empilement des caractéristiques apprises de chaque domaine source pour généraliser à un nouveau domaine cible tout en ignorant les informations en lien avec les étiquettes. Cela mène généralement à des caractéristiques indiscernables avec une frontière de classification ambiguë. Une solution possible est de contraindre la similarité des étiquettes lors de l'extraction des caractéristiques invariantes et de tirer profit des similarités des étiquettes pour assurer la cohésion entre les classes et la séparation des caractéristiques entre les domaines. Dès lors, nous adoptons ici le transport optimal utilisant la distance de Wasserstein, qui pour pourrait contraindre la similarité des étiquettes des classes, pour l'en-

traînement adversarial. Nous déployons également un objectif d'apprentissage métrique pour exploiter les informations en lien avec les étiquettes afin d'obtenir une frontière de classification distincte. Nos résultats empiriques montrent que la méthode proposée est plus performante que la plupart des méthodes de référence. De plus, des études d'ablation démontrent également l'efficacité de chaque composante de notre méthode.

Abstract

Generalizing knowledge to unseen domains, where data and labels are unavailable, is crucial for machine learning. We tackle in this chapter the domain generalization problem to learn from multiple source domains and generalize to a target domain with unknown statistics. The crucial idea is to extract the underlying invariant features across all the domains. Previous domain generalization approaches mainly focused on learning invariant features and stacking the learned features from each source domain to generalize to a new target domain while ignoring the label information, this generally leads to indistinguishable features with an ambiguous classification boundary. One possible solution is to constrain the label-similarity when extracting the invariant features and take advantage of the label similarities for class-specific cohesion and separation of features across domains. We adopt here the optimal transport with Wasserstein distance, which could constrain the class label similarity, for adversarial training. We also deploy a metric learning objective to leverage the label information for achieving distinguishable classification boundary. Our empirical results show that our proposed method could outperform most of the baselines. Furthermore, ablation studies also demonstrate the effectiveness of each component of our method.

4.1 Introduction

As discussed in Chapter 1, this thesis focus on leveraging the transferable features from different domains. In the previous Chapter 3, we explored transfer learning problems in the context of domain adaptation where the learner has access to the source domain (data and corresponding labels) as well as the target domain labels. In this chapter, we tackle the transfer learning problem under the domain generalization (DG) paradigm, where the learner has access to many source domains (data and corresponding labels), and aims at generalizing to the new (target) domain, where neither data nor labels is accessible during training.

As introduced in section 1.4.1, the goal of DG aims to learn a prediction model on training data from the seen source domains so that such model can generalize well on the unseen target domain. An underlying assumption of DG is that there exists a common feature space underlying the multiple known source domains and unseen target domain (Li et al., 2017a). We have introduced the general process of DG in section 1.4.1, here we recall the DG process with the an example in Fig.4.1.

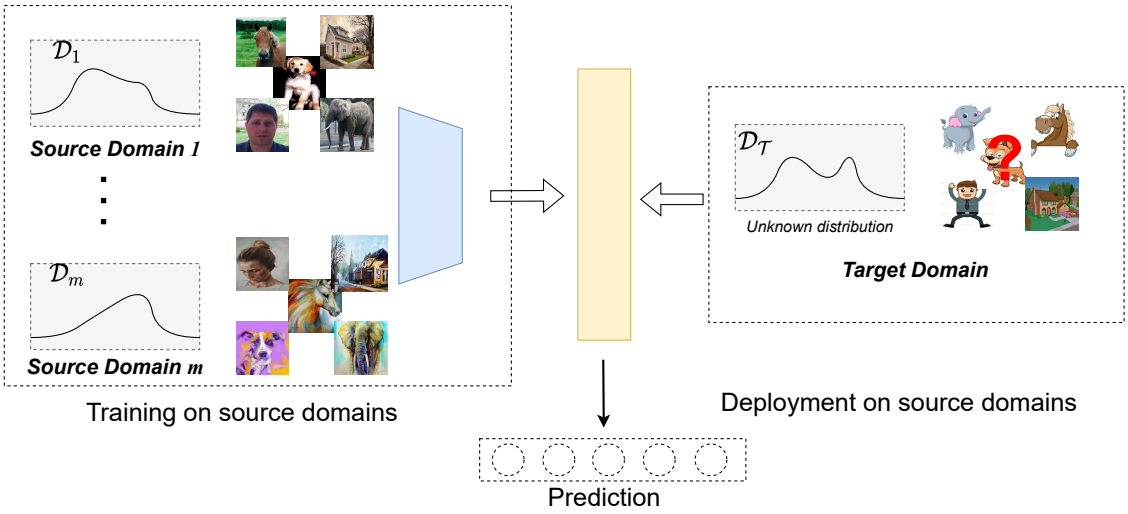


FIGURE 4.1 – Domain Generalization (DG) : A learner faces a set of labelled data from several source domains, of which aims at extracting invariant features across them and learn to generalize to an unseen domain. Data of each domain i is generated by a distribution $\mathcal{D}_i$. The learner can measure the distance between the source domain distributions and thus finding a common feature space, but has no information on the unseen target distribution. After training on the source domains, the model is then deployed to a new domain $\mathcal{D}_\mathcal{T}$ for prediction. Dataset images are drawn from PACS dataset (Li et al., 2017b).

A critical problem in DG and DA involves aligning the domain distributions, which typically are achieved by extracting feature representations of each domain. As summarized in section 2.2, previous DA works usually tried to minimize the domain discrepancies, such as Maximum Mean Discrepancy (MMD) (Li et al., 2018c), or using adversarial training (Zhao et al., 2019a; Chen et al., 2020b), to achieve domain distribution alignments (see also in section 1.5.1, Appendix A.1 and Appendix A.3). Due to the similar problem setting between DA and DG, some DA training methods were directly adopted to DG approaches. For example, the MMD metric was adopted by Li et al. (2018c) as a cross-domain regularizer, and the KL divergence was used to measure the domain shift by Li et al. (2017a) for domain generalization problems. As we discussed in section 1.5.1, the MMD metric is usually implemented in the kernel space, and is restrictive for large-scaled applications Similarly, KL divergence is unbounded, and thus is also insufficient for a successful measuring domain shift (Zhao et al., 2019a).

Besides, previous domain generalization approaches (Ilse et al., 2020; Ghifary et al., 2015b; Li et al., 2018f; D'Innocente and Caputo, 2018; Volpi et al., 2018) mainly focused on applying similar DA technique to extract the invariant features and how to stack the learned features from each domain for generalizing to a new domain. As previous noticed, these methods usually ignore the label similarities in the source domains, which will sometimes make the features became indistinguishable with ambiguous classification boundaries (Deng et al., 2020).

Based on the observations above, a successful generalization process should guide the learner

to not only to align the feature distributions between each domain but also to discriminate the samples using their similarities. That is, during the feature alignment process, samples from the same class could lie close to each other while samples from different classes could stay apart from each other, *a.k.a.* feature compactness (Kamnitsas et al., 2018).

As continuum to DG problems, we adopt the Optimal Transport (OT) with Wasserstein distance (see section A.2 and Appendix A.3) to align the feature distribution for domain generalization since it could constrain labelled source samples of the same class to remain close during the transportation process (Courty et al., 2016). Besides, as discussed in section 1.5, compared with (Ben-David et al., 2010a), OT benefits from the advantages of Wasserstein distance by its gradient property (Arjovsky et al., 2017) and the promising generalization bound (Redko et al., 2017; Shen et al., 2018). The empirical studies (Gulrajani et al., 2017; Shen et al., 2018) also demonstrated the effectiveness of OT for aligning the marginal distributions of different domains and then extracting the invariant features.

Notice that, although the optimal transport process could constrain the labelled samples of the same class to stay close to each other, our preliminary results showed that just implementing optimal transport for domain generalization is not sufficient for a cohesion and separable classification boundary (see in section 1.4.1 and also in Fig. 4.3c). The model could still suffer from indistinguishable features. In order to train the model to predict well on all the domains, this separable classification boundary should also be achieved under a domain-agnostic manner. That is, for a pair of instances, no matter which domain they come from, they should stay close to each other if they are in the same class and away from each other if they are not from the same class. To this end, we further promote metric learning as an auxiliary objective for leveraging the label similarities in the source domains for a domain-independent distinguishable classification boundary.

To sum up, we deployed the optimal transport technique with Wasserstein distance for domain generalization for extracting the domain invariant features. To avoid ambiguous classification boundary, we proposed to implement metric learning strategies to achieve a distinguishable feature space. In this context, we proposed the Wasserstein Adversarial Domain Generalization ($WADG$) algorithm.

In order to check the effectiveness of the WADG algorithm, we tested it on several benchmarks and then compare its performance with some recent domain generalization baselines. Our experiment results showed that our proposed algorithm outperform most of the baselines, achieving the state-of-the-art performance, and this confirms the effectiveness of our proposed algorithm. Furthermore, the ablation studies and the corresponding T-SNE visualizations also demonstrated the contributions of each learning objective of our algorithm.

4.2 Problem Setup

We first recall the problem settings and preliminaries of DG.

Notations and Definitions

Following the notations in section 1.4.1, suppose we have m known source domains $\{\mathcal{D}_i\}_{i=1}^m$, and i^{th} domain contains N_i labeled instances in total, denoted by $\{(\mathbf{x}_j^{(i)}, y_j^{(i)})\}_{j=1}^{N_i}$, where $\mathbf{x}_j^{(i)} \in \mathbb{R}^n$ is the j^{th} instance feature from the i^{th} domain and $y_j^{(i)} \in \{1, \ldots, K\}$ are the corresponding labels. For a hypothesis class $\mathcal{H}$, the expected source and target risk of a hypothesis $h \in \mathcal{H}$ over domain distribution $\mathcal{D}_i$ is the probabilities that h wrongly predicts on the entire distribution $\mathcal{D}_i$: $R_i(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}\ell(h(\mathbf{x}, y))$, where $\ell(\cdot)$ is the error function. The empirical loss is also defined by : $\hat{R}_i(h) = \frac{1}{N_i}\sum_{j=1}^{N_i}\ell(h(\mathbf{x}_j, y_j))$.

In the setting of DG, we only have the access to the seen source domains $\mathcal{D}_i$ but have no information about the target domain. The learner is expected to extract the underlying invariant feature space across the source domains and generalize to a new target domain.

Optimal Transport and Wasserstein Distance

We have introduced the general OT theory in section A.2. In this chapter, we continue to use OT and Wasserstein distance based adversarial training method for extracting the features. To learn domain invariant features, OT technique is implemented to achieve domain alignments for extracting invariant features (details of this process are provided in Appendix A.3). Similar to the previous Chapter 3, here we adopt Wasserstein-1 distance. Then, according to the Kantorovich-Rubinstein duality (see Appendix A.2), let f be a Lipschiz-continous function $\|f\| < 1$, we have,

$$W_1(\mathcal{D}_i, \mathcal{D}_j) = \sup_{\|f\|_L<1} \mathbb{E}_{\mathbf{x}\in\mathcal{D}_i} f(\mathbf{x}) - \mathbb{E}_{\mathbf{x}'\in\mathcal{D}_j} f(\mathbf{x}') \tag{4.1}$$

Since our goal is to not only align the marginal distribution via optimal transport but also to leverage the label similarity information across domains for enhancing class-specific cohesion and separation. We then introduce some basics of metric learning including pair similarity and weighting techniques in the next section.

Metric Learning

For a pair of instances $(\mathbf{x}_i, y_i)$ and $(\mathbf{x}_j, y_j)$, the notion of *positive pairs* usually refers to the condition where pair i, j have same labels ($y_i = y_j$), while the negative pairs usually refers to the condition $y_i \neq y_j$. The central idea of metric learning is to encourage a pair of instances who have the same labels to be closer, and push negative pairs to be apart from each other (Manmatha et al., 2017).

We adopt the general pair-weighting process of metric learning as proposed by Wang et al. (2019c). Assuming the feature extractor f parameterized by $\boldsymbol{\theta}_f$ projects the instance $\mathbf{x} \in \mathbb{R}^n$ to a d-dimensional normalized space : $f(\mathbf{x}; \boldsymbol{\theta}_f) : \mathbb{R}^n \to [0, 1]^d$. Then, for two samples $\mathbf{x}_i$ and $\mathbf{x}_j$, the *similarity* between them could be defined as the inner product of the corresponding feature vector :

$$S_{i,j} := \langle f(\mathbf{x}_i; \boldsymbol{\theta}_f), f(\mathbf{x}_j; \boldsymbol{\theta}_f) \rangle \tag{4.2}$$

where $\langle \cdot, \cdot \rangle$ refers to the inner product. We will further explain the similarity $S_{i,j}$ in section 4.3.

To leverage the across-domain class similarity information can encourage the learner to extract the classification boundary that regardless of domains, which is an useful auxiliary information for the learner. In order to discriminate the instances from all the domains correctly, the feature classification boundary should be independent of the domains, *i.e.*, learning invariant features with *domain-agnostic* classification boundary such that for a pair of instance, no matter which domain they come from, they should be mapped close with each other in the feature space if they are from the same class while away from each other if they are not the same class. We further elaborate this process in section 4.3.

4.3 Methodology

The overall idea of our WADG algorithm is to learn a domain-invariant feature space and domain-agnostic classification boundary. Firstly, we align the marginal distribution of different source domains using optimal transport by minimizing the Wasserstein distance to achieve the domain-invariant feature space. And then, we adopt metric learning objective to guide the learner to leverage the class similarities for a better classification boundary. A general workflow of our method is illustrated in Fig. 4.2a. The model contains three major parts : a feature extractor, a classifier and a critic function.

The feature extractor function F, parameterized by $\boldsymbol{\theta}_f$, extracts the features from different source domain. For set of instances $\mathbf{X}^{(i)} = \{\mathbf{x}_j^{(i)}\}_{j=1}^{N_i}$ from domain $\mathcal{D}_i$, we can then denote the extracted feature from domain i as $\mathbf{Z}^{(i)} = F(\mathbf{X}^{(i)})$. The classifier function C, parameterized by $\boldsymbol{\theta}_c$, is expected to learn to predict labels of instances from all the domains correctly, *i.e.*, for an input $\mathbf{x}$, we compute $C(F(\mathbf{x}; \theta_f); \theta_c) \to y$ for making the predictions. For simplicity, we omit the model parameters when displaying the formulas, then the model prediction is computed by $C(F(\mathbf{x}))$. The critic function ϕ, parameterized by $\boldsymbol{\theta}_\phi$, aims to measure the empirical Wasserstein distance between features from a pair of source domains. For the target domain, all the instances and labels are absent during the training time.

The WADG algorithm aims to learn the domain-agnostic features with distinguishable classification boundary. During each train round, the network receives the labelled data from all domains and train the classifier under a supervised mode with the classification loss $\mathcal{L}_C$. For

the classification process, we use the typical cross-entropy loss for all m source domains :

$$\mathcal{L}_C = -\sum_{i=1}^{m}\sum_{j=1}^{N_i} y_j \log(\mathbb{P}(C(F(\mathbf{x}_j^{(i)})))) \tag{4.3}$$

The feature extractor F is then trained to minimize the estimated Wasserstein Distance in an adversarial manner with the critic ϕ with an objective $\mathcal{L}_D$. We then adopt a metric learning objective (namely, $\mathcal{L}_{MS}$) for leveraging the similarities for a better classification boundary. Our full method then solve the joint loss function,

$$\mathcal{L} = \arg\min_{\theta_f,\theta_c}\max_{\theta_d} \mathcal{L}_C + \mathcal{L}_D + \mathcal{L}_{MS}, \tag{4.4}$$

where $\mathcal{L}_D$ is the adversarial objective function, and $\mathcal{L}_{MS}$ is the metric learning objective function. Note that the 'min-max' process in Eq. 4.4 is achieved by the adversarial training process that introduced in section 1.4.1 and Appendix A.3.

In Eq. 4.4, we need to compute three learning objectives $\mathcal{L}_C$, $\mathcal{L}_D$ and $\mathcal{L}_{MS}$. The classification objective $\mathcal{L}_C$ is the general cross-entropy loss introduced in Eq. 4.3. In the following sections, we explain the last two objectives $\mathcal{L}_D$ and $\mathcal{L}_{MS}$ with more details.

Adversarial Domain Generalization using Optimal Transport

As previously introduced, we deploy the optimal transport technique with Wasserstein distance (Redko et al., 2017; Shen et al., 2018) for aligning the marginal feature distribution over all the source domains. A brief workflow of the optimal transport for a pair of source domains is illustrated in Fig. 4.2b.

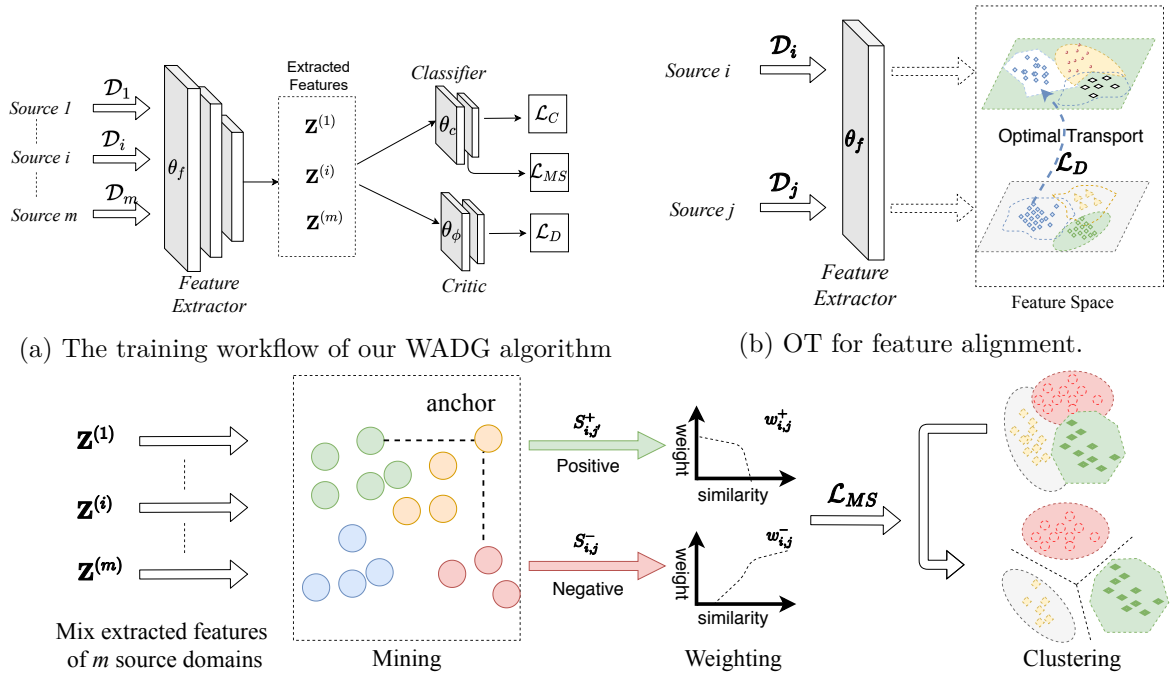
The critic function ϕ estimates the empirical Wasserstein Distance between each source domain through a pair of instances from the empirical sets $\mathbf{x}^{(i)} \in \mathbf{X}^{(i)}$ and $\mathbf{x}^{(j)} \in \mathbf{X}^{(j)}$. Follow Shen et al. (2018), the dual term Eq. 4.1 of Wasserstein distance is computed by,

$$W_1(\mathbf{X}^{(i)}, \mathbf{X}^{(j)}) = \max\Big(\frac{1}{N_i}\sum_{\mathbf{x}^{(i)}\in\mathbf{X}^{(i)}} \phi(F(\mathbf{x}^{(i)})) - \frac{1}{N_j}\sum_{\mathbf{x}^{(j)}\in\mathbf{X}^{(j)}} \phi(F(\mathbf{x}^{(j)}))\Big) \tag{4.5}$$

As in domain generalization setting, there usually exists several source domains, we can sum all the empirical Wasserstein distance between each pair of source domains,

$$\mathcal{L}_D = \sum_{i=1}^{m}\sum_{j=i+1}^{m}\Big[\frac{1}{N_i}\sum_{\mathbf{x}^{(i)}\in\mathbf{X}^{(i)}} \phi(F(\mathbf{x}^{(i)})) - \frac{1}{N_j}\sum_{\mathbf{x}^{(j)}\in\mathbf{X}^{(j)}} \phi(F(\mathbf{x}^{(j)}))\Big] \tag{4.6}$$

By optimizing $\mathcal{L}_D$, the learner could extract a domain-invariant feature space. We then propose to apply metric learning approaches to leverage the class label similarity for domain independent clustering feature extraction. We then introduce the metric learning for domain agnostic clustering in the next section.



(a) The training workflow of our WADG algorithm

(b) OT for feature alignment.

(c) Metric learning for clustering process

FIGURE 4.2 – The proposed WADG method. (a) : The model mainly consists of three parts, the feature extractor, the classifier and the critic function. During training, the model receives all the source domains, and the feature extractor is trained to learn invariant features together with the critic function in an adversarial manner. (b) : For each pair of source domains $\mathcal{D}_i$ and $\mathcal{D}_j$, optimal transport process for aligning the features from different domains. (c) : For a batch of all source domain instances, we first roughly mining the positive and negative pairs by Eq. 4.7. Finally, the corresponding weights are computed using Eq. 4.11 and Eq. 4.12 compute $\mathcal{L}_{MS}$ so that the learner could leverage the metric learning objective.

Metric Learning for Domain Agnostic Classification Boundary

As aforementioned in section 4.1, only aligning the marginal features by adversarial training (see section 1.5.2 and section 2.2) is not sufficient for DG since the label information is neglected and the which can lead to the semantic misalignment problems (Dou et al., 2019). When predicting on the target domain, the learner may still suffer from this ambiguous decision boundary. To solve this issue, we propose to implement the metric learning techniques to help cluster the instances and promote a better prediction boundary for better generalization.

To this end, except to the supervised source classification and alignment of the marginal distribution across domains with the Wasserstein adversarial training defined above, we then further encourage robust domain-independent local clustering by leveraging label information using the metric learning objective. The brief workflow is illustrated in Fig. 4.2c. Specifically, we adopt the metric learning objective (Wang et al., 2019c) to require the images regardless of their domains to follow the two aspects :

1 Instances from the same class are semantically similar, thereby should be mapped nearby in the embedding space (semantic clustering).

2 Instances from different classes should be mapped apart from each other in embedding space

Since goal of domain generalization aims to learn to hypothesis could predict well on all the domains, the clustering should also be achieved under a domain-agnostic manner. That is, for a pair of instances, no matter which domain they come from, they should stay close to each other if they are in the same class and stay away from each other if they are not in the same class.

To this end, we mix the instances from all the source domains together and encourage the clustering for domain agnostic features using the metric learning techniques to achieve a domain-independent clustering decision boundary. For this, during each training iteration, for a batch of instances $\{\mathbf{x}_1^{(i)}, y_1^{(i)}, \ldots, \mathbf{x}_b^{(i)}, y_b^{(i)}\}_{i=1}^{m}$ from m source domains with batch size b, we mix all the instances from each domain and denoted by $\{(\mathbf{x}_i^B, y_i^B)\}_{i=1}^{m'}$ with total size m'. Follow (Wang et al., 2019c), we first measure the relative similarity between the negative and positive pairs, *i.e.*, the pair similarity mining process, which is introduced in the next part.

Pair Similarity Mining : Assume $\mathbf{x}_i^B$ is an anchor, a negative pair $\{\mathbf{x}_i^B, \mathbf{x}_j^B\}$ and a positive pair $\{\mathbf{x}_i^B, \mathbf{x}_{j'}^B\}$ are selected if S_{ij} and $S_{i,j'}$ satisfy the negative condition $S_{i,j}^-$ and the positive condition $S_{i,j}^+$, respectively :

$$S_{i,j}^- \geq \min_{y_i=y_k} S_{i,k} - \epsilon, \quad S_{i,j'}^+ \leq \min_{y_i \neq y_k} S_{i,k} + \epsilon \tag{4.7}$$

where ϵ is a given margin. Through Eq. 4.7 and specific margin ϵ, we will have a set of negative pairs $\mathcal{N}$ and a set of positive pairs $\mathcal{P}$. This process (Eq. 4.7) could roughly cluster the instances with each anchor by selecting informative pairs (inside of the margin, *i.e.*, $S_{i,j'}^+ \leq \min_{y_i \neq y_k} S_{i,k} + \epsilon$), and discard the less informative ones (outside of the margin, *i.e.*, $S_{i,j}^- \geq \min_{y_i=y_k} S_{i,k} - \epsilon$).

With such roughly selected informative pairs $\mathcal{N}$ and $\mathcal{P}$, we then assign the instance with different weights. Intuitively, if a instance has higher similarity with an anchor, then it should stay closer with the anchor and vice-versa. We introduce the weighting process in the next section.

Pair Weighting : For instances of positive pairs, if they are more similar with the anchor, then it should have higher weights while give the negative pairs with lower weights if they are more dissimilar, no matter which domain they come from. Through this process, we can push the instances into several groups by measuring their similarities.

For N instances, computing the similarity between each pair could result in a similarity matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$. Let $S_{i,j}$ be the i^{th} row, j^{th} column element of matrix $\mathbf{S}$, then $S_{i,j}$ refers to the

similarities between instance i and j. According to (Wang et al., 2019c), a loss function based on pair similarity, it can usually be defined by $\mathcal{F}(\mathbf{S}, y)$. Then, the gradient of $\mathcal{F}(\mathbf{S}, y)$ $w.r.t$ the feature extractor $\boldsymbol{\theta}_f$ could be computed by,

$$\frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial \boldsymbol{\theta}_f} = \frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial \mathbf{S}} \frac{\partial \mathbf{S}}{\partial \boldsymbol{\theta}_f} = \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial S_{i,j}} \frac{\partial S_{i,j}}{\partial \boldsymbol{\theta}_f} \tag{4.8}$$

Eq. 4.8 could be reformulated into a new loss function $\mathcal{L}_{MS}$ as,

$$\mathcal{L}_{MS} = \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial S_{i,j}} S_{i,j} \tag{4.9}$$

Eq.4.9 is achieved by the integration of RHS of Eq. 4.8 with $\boldsymbol{\theta}_f$, which defined the metric learning objective $w.r.t$ similarity matrix $\mathbf{S}$ and label y. The term $\frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial S_{i,j}}$ in Eq. 4.9 could be treated as an constant scalar since it doesn't contain the gradient of $\mathcal{L}_{MS}$ $w.r.t$ $\boldsymbol{\theta}_f$. When training the model using back-propagation, we just need to compute the gradient term $\frac{\partial S_{i,j}}{\partial \boldsymbol{\theta}_f}$ for both the positive and negative pairs. Thus, Eq. 4.9 is transformed by the summation over all the positive pair ($y_i = y_j$) and negative pairs ($y_i \neq y_j$),

$$\begin{aligned}
\mathcal{L}_{MS} &= \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial S_{i,j}} S_{i,j} \\
&= \sum_{i=1}^{N} \left(\sum_{j=1, y_j \neq y_i}^{N} \frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial S_{i,j}} S_{i,j} + \sum_{j=1, y_j = y_i}^{N} \frac{\partial \mathcal{F}(\mathbf{S}, y)}{\partial S_{i,j}} S_{i,j} \right) \\
&= \sum_{i=1}^{N} \left(\sum_{j=1, y_j \neq y_i}^{N} w_{i,j} S_{i,j} + \sum_{j=1, y_j = y_i}^{N} (-w_{i,j}) S_{i,j} \right)
\end{aligned} \tag{4.10}$$

where $w_{i,j} = \left| \frac{\partial S_{i,j}}{\partial \boldsymbol{\theta}_f} \right|$ is regarded as the weight for similarity $S_{i,j}$. Since our goal is to encourage the positive pairs to be closer, then we can assume the weight for positive pairs is smaller than 0. Conversely, for a negative pair, we can assume the weight is larger than 0. The intuition is that for a negative pair of instances, let the weight be positive, we can give it a higher loss value. Then, the learner can learn to distinguish them. On the contrary, we can assign the negative weights towards the positive pairs, which will guide the learner to not separate them apart.

For each pair of instances i, j, we could assign different weights according to their similarities $S_{i,j}$. Then, we can denote $w^+_{i,j}$ and $w^-_{i,j}$ as the weight of a positive or negative pairs' similarity, respectively. Previously, Yi et al. (2014) and Wang et al. (2019c) applied a soft function for measuring the similarity. We then consider the similarity of the pair itself (*i.e.* self-similarity), the negative similarity and the positive similarity. The weight of self-similarity could be measured by $\exp(S_{i,j} - \lambda)$ with a small threshold λ. For a selected negative pair $\{\mathbf{x}^B_i, \mathbf{x}^B_j\} \in \mathcal{N}$

the corresponding weight (see Eq. 4.10) could be defined by the soft function of self-similarity together with the negative similarity :

$$\begin{aligned} w_{i,j}^{-} &= \frac{1}{\exp(\beta(\lambda - S_{ij})) + \sum_{k\in\mathcal{N}} \exp(\beta(S_{i,k} - \lambda))} \\ &= \frac{\exp(\beta(S_{ij} - \lambda))}{1 + \sum_{k\in\mathcal{N}} \exp(\beta(S_{ik} - \lambda))} \end{aligned} \tag{4.11}$$

Similarly, the weight of a positive pair $\{\mathbf{x}_i^B, \mathbf{x}_j^B\} \in \mathcal{P}$ is defined by,

$$w_{i,j}^{+} = \frac{1}{\exp(-\alpha(\lambda - S_{i,j})) + \sum_{k\in\mathcal{P}} \exp(-\alpha(S_{i,k} - S_{i,j}))} \tag{4.12}$$

Then, take Eq. 4.11 and Eq. 4.12 into Eq. 4.10, and integrate Eq. 4.10 with the similarity mining $S_{i,j}$, we have the objective function for clustering,

$$\mathcal{L}_{MS} = \frac{1}{m}\sum_{i=1}^{m}\{\frac{1}{\alpha}\log[1 + \sum_{k\in\mathcal{P}_i}\exp(-\alpha(S_{ik} - \lambda))] + \frac{1}{\beta}\log[1 + \sum_{k\in\mathcal{N}_i}\exp(\beta(S_{ik} - \lambda))]\} \tag{4.13}$$

where λ, α and β are fixed hyper-parameters, we determine them in the empirical setting section 4.4.2. Then, the whole objective of our proposed method is,

$$\mathcal{L} = \arg\min_{\boldsymbol{\theta}_f, \boldsymbol{\theta}_c} \max_{\boldsymbol{\theta}_\phi} \mathcal{L}_C + \lambda_d \mathcal{L}_D + \lambda_s \mathcal{L}_{MS} \tag{4.14}$$

where λ_d and λ_s are coefficients to regularize $\mathcal{L}_d$ and $\mathcal{L}_{MS}$ respectively, we show the setting of them also in section 4.4.2.

Based on these components above, we propose the Wasserstein Adversarial Domain Generalization (WADG) algorithm in Algorithm 2.

With the WADG algorithm in hand, we show the empirical results in the next section.

4.4 Experiments and Results

4.4.1 Datasets

In order to evaluate our proposed approach, follow the evaluation protocol of (Dou et al., 2019; Matsuura and Harada, 2020; Li et al., 2017b), we implement experiments on three benchmarks : **VLCS** (Torralba and Efros, 2011), **PACS** (Li et al., 2017a) and **Office-home** (Venkateswara et al., 2017) dataset. The VLCS dataset contains images from 4 different domains : PASCAL VOC2007 (V), LabelMe (L), Caltech (C), and SUN09 (S), where each domain includes five classes : *bird, car, chair, dog* and *person*. PACS dataset is a recent benchmark dataset for domain generalization. It consists of four domains : Photo (P), Art painting (A), Cartoon (C), Sketch (S), with objects from seven classes : dog, elephant, giraffe, guitar, house, horse, person.

Algorithm 2 The Wasserstein Adversarial Domain Generalization (WADG) algorithm

Input: Samples from different source domains $\{\mathcal{D}_i\}_{i=1}^{M}$;
Output: Neural network parameters $\boldsymbol{\theta}_f$, $\boldsymbol{\theta}_c$, $\boldsymbol{\theta}_\phi$

1: **for** mini-batch of samples $\{(\mathbf{x}_s^{(i)}, y_s^{(i)})\}$ from source domains ; **do**
2: Compute the classification loss $\mathcal{L}_C$ over all the domains according to Eq. 4.3 ;
3: Compute the Wasserstein distance $\mathcal{L}_D$ between each pair of source domains according to Eq. 4.6 ;
4: Mix the pairs from different domains and compute the similarity by Eq. 4.2 ;
5: Roughly select the positive and negative pairs by solving Eq. 4.7 ;
6: Compute similarity loss $\mathcal{L}_{MS}$ on all the source instances by Eq. 4.13 ;
7: Update θ_f, θ_c and 1_d by solving Eq. 4.14 with learning rate η :

$$\boldsymbol{\theta}_f \leftarrow \boldsymbol{\theta}_f - \eta \frac{\partial(\mathcal{L}_C + \lambda_d \mathcal{L}_D + \lambda_s \mathcal{L}_{MS})}{\partial \boldsymbol{\theta}_f},$$
$$\boldsymbol{\theta}_c \leftarrow \boldsymbol{\theta}_c - \eta \frac{\partial(\mathcal{L}_C + \lambda_d \mathcal{L}_D + \lambda_s \mathcal{L}_{MS})}{\partial \boldsymbol{\theta}_c},$$
$$\boldsymbol{\theta}_\phi \leftarrow \boldsymbol{\theta}_\phi + \eta \frac{\partial \mathcal{L}_D}{\partial \boldsymbol{\theta}_\phi}.$$

8: **end for**
9: Return the optimal parameters $\boldsymbol{\theta}_f^\star$, $\boldsymbol{\theta}_c^\star$ and $\boldsymbol{\theta}_\phi^\star$.

Office-Home is a more challenging dataset, which contains four different domains : *Art* (Ar), *Clipart* (Cl), *Product* (Pr) and *Real World* (Rw), with 65 categories in each domain.

Previous work showed that matter the adversarial model is trained under supervised (Long et al., 2017a), semi-supervised (Zhou et al., 2021c) or unsupervised (Long et al., 2018) way, the model will suffer from learning the diverse feature. To test our domain generalization model on this dataset could also help to affirm the effectiveness of our approach.

4.4.2 Baselines and Implementation details

To show the effectiveness of our proposed approach, we compared our algorithm on the benchmark datasets with the following recent domain generalization methods.

— ***Deep All*** : We follow the standard evaluation protocol of Domain Generalization to set up the pre-trained model fine-tuned on the aggregation of all source domains with only the classification loss.

— **TF** (Li et al., 2017b) : This method adopt a low-rank parameterized Convolution Neural Network model which aims to reduce the total number of model parameters for an end-to-end Domain Generalization training.

— **CIDDG** (Li et al., 2018f) : This work matches the conditional distribution by change the class prior.

— **MLDG** (Li et al., 2018b) : This one implements the meta-learning approach for domain generalization. It runs the meta-optimization on simulated meta-train/ meta-test sets with domain shift

— **CCSA** (Motiian et al., 2017b) : This work adopts the contrastive semantic alignment loss for both the domain adaptation and domain generalization problem.

— **MMD-AAE** (Li et al., 2018c) : This approach adopts the Adversarial Autoencoder model was adopted together with the Mean-Max Discrepancy to extract a domain invariant feature for generalization.

— **D-SAM** (D'Innocente and Caputo, 2018) : This work aggregates the domain-specific modules from each domain, and also stacks general and specific information from each domain for generalization.

— **JiGen** (Carlucci et al., 2019) : It achieves domain generalization by solving the Jigsaw puzzle through the unsupervised task.

— **MASF** (Dou et al., 2019) : A meta-learning style method which based on MLDG and combined with Consitrastive Loss/ Triplet Loss to encourage domain-independent semantic feature space.

— **MMLD** (Matsuura and Harada, 2020) : An approach that mixes all the source domains by assigning a pseudo domain label for extract domain-independent cluster feature space.

To compare with the recent baselines, we following the general evaluation protocol of general DG works (e.g. Dou et al., 2019; Matsuura and Harada, 2020) and first test our algorithm on by using AlexNet (Krizhevsky et al., 2012) backbone of which the last layer was removed so that we can use it as feature extractor. For preparing the dataset, we follow the train/validation/test dataset split and the data pre-processing protocol of Matsuura and Harada (2020). As for the classifier, follow the work of (You et al., 2019) we initialize a three-layers MLP whose input has the same number of inputs as the feature extractor's output and to have the same number of outputs as the number of object categories. For the metric learning objective, we use the output of the second layer of classifier network for computing the similarity.

We adopt the ADAM (Kingma and Ba, 2015) optimizer for training with learning rate ranging from 5×10^{-4} to 5×10^{-5} for the whole model. For stable training, when optimizing the whole learning objective Eq. 4.14 we set coefficient $\lambda_d = \frac{2}{1+\exp(-10p)} - 1$, where p is the training progress. This regularization scheme λ_d has been used in adversarial training based domain adaptation and generalization setting (e.g. Long et al., 2017a; Wen et al., 2019b; Matsuura and Harada, 2020) and has been proved to help to stabilize the training process. For the value of λ_s in Eq. 4.14, we follow the setting of (Dou et al., 2019) and set the value to 10^{-4}. In our preliminary validation results, the performance is not sensitive with $\lambda_d \in [0, 1]$. We also tried to range λ_s from 10^{-3} to 10^{-6} by reverse validation and didn't observe obvious differences.

Method	Art	Cartoon	Sketch	Photo	Avg.
Deep All	63.30	63.13	54.07	87.70	67.05
TF(Li et al., 2017b)	62.86	66.97	57.51	89.50	59.21
CIDDG(Li et al., 2018f)	62.70	69.73	64.45	78.65	68.88
MLDG (Li et al., 2018b)	66.23	66.88	58.96	88.00	70.01
D-SAM(D'Innocente and Caputo, 2018)	63.87	70.70	64.66	85.55	71.20
JiGen(Carlucci et al., 2019)	67.63	71.71	65.18	89.00	73.38
MASF(Dou et al., 2019)	**70.35**	72.46	67.33	90.68	75.21
MMLD(Matsuura and Harada, 2020)	69.27	**72.83**	66.44	88.98	74.38
Ours	70.21	72.51	**70.32**	**89.81**	**75.71**

TABLE 4.1 – Empirical Results (accuracy %) on PACS dataset with pre-trained AlexNet as feature extractor. For each column, we refer the generalization taks as the target domain name. For example, the third column 'Cartoon' refers to the generalization tasks where domain *Cartoon* is the target domain while the model is trained on the rest three domains (*Art, Sketch, Photo*).

For the hyper-parameters in $\mathcal{L}_{MS}$ (see Eq. 4.7 and Eq. 4.13), we empirically set $\lambda = 1.0$, $\epsilon = 0.1$, $\alpha = 2.0$, $\beta = 40.0$. Notice that ϵ is for roughly selecting the positive and negative pairs and λ is a small margin parameters. Our validation results showed that when $\lambda \in [0.5, 2.0]$, the algorithm can have good performance. Besides, α and β are two parameters used for positive and negative mining referred by Ustinova and Lempitsky (2016) in which α was set to 2.0 and β was set to 50.0. In our validation results, we found the setting of $\beta \in [30.0, 45.0]$ could guarantee stable performance in our domain generalization problems rather 50.0 in the original (Ustinova and Lempitsky, 2016). Notice that $\alpha \in [1.0, 5.0]$ could also have good performance. Based on those findings we report the empirical results with $\alpha = 2.0$ and $\beta = 40.0$.

Then, we examined our algorithm on the office-home benchmark, which is more challenging than the previous PACS and VLCS datasets. We follow the setting of (Carlucci et al., 2019), which is the most recent work who also evaluated on office-home dataset, to have a fair comparison. For this Office-home dataset, we also used reverse validation to set the learning rate as $2e-4$ for the whole model. For the remaining hyper-parameters, we keep the same with PACS and VLCS experiments. To avoid over-training, we also adopt the early stopping technique. All the experiments are programmed with *PyTorch* (Paszke et al., 2019).

4.4.3 Experiments Results

We first reported the empirical results on PACS and VLCS dataset using AlexNet as feature extractor in Table 4.1 and Table 4.2, respectively. For each generalization task, we train the model on all the source domains and test on the target domain and report the average of top 5 accuracy values. The empirical results refers to the average accuracy about training on source domains while testing on the target domain.

Method	Caltech	LabelMe	Pascal	Sun	Avg.
Deep All	92.86	63.10	68.67	64.11	72.19
D-MATE (Ghifary et al., 2015b)	89.05	60.13	63.90	61.33	68.60
CIDDG (Li et al., 2018f)	88.83	63.06	64.38	62.10	69.59
CCSA (Motiian et al., 2017b)	92.30	62.10	67.10	59.10	70.15
SLRC (Ding and Fu, 2017)	92.76	62.34	65.25	63.54	70.97
TF (Li et al., 2017b)	93.63	63.49	69.99	61.32	72.11
MMD-AAE (Li et al., 2018c)	94.40	62.60	67.70	64.40	72.28
D-SAM (D'Innocente and Caputo, 2018)	91.75	56.95	58.95	60.84	67.03
MLDG (Li et al., 2018b)	94.4	61.3	67.7	65.9	73.30
JiGen (Carlucci et al., 2019)	96.93	60.90	70.62	64.30	73.19
MASF (Dou et al., 2019)	94.78	**64.90**	69.14	67.64	74.11
MMLD (Matsuura and Harada, 2020)	96.66	58.77	**71.96**	**68.13**	73.88
Ours	**96.68**	64.26	71.47	66.62	**74.76**

TABLE 4.2 – Empirical Results (accuracy %) on VLCS dataset with pre-trained AlexNet as feature extractor.

	Art	Clipart	Product	Real-World	Avg.
Deep All	52.15	45.86	70.86	73.15	60.51
D-SAM(D'Innocente and Caputo, 2018)	**58.03**	44.37	69.22	71.45	60.77
JiGen(Carlucci et al., 2019)	53.04	**47.51**	71.47	72.79	61.20
Ours	55.34	44.82	**72.03**	**73.55**	**61.44**

TABLE 4.3 – Empirical Results (accuracy %) on Office-home dataset with pre-trained ResNet-18 as feature extractor.

From the empirical results, we can observe our method outperforms the baselines both on the PACS and VLCS dataset, indicating an improvement on benchmark performances. This showed the effectiveness of our method. Then, we report the empirical results on Office-home dataset in Table 4.3. As stated before, Office-home is a more larger and challenging dataset contains more diverse features from 65 different classes. To evaluate the performance on this dataset requires large amount of computational resources. Due to the limits, we follow the evaluation protocol of Carlucci et al. (2019) to report the empirical results. From those results, we could observe that our algorithm outperforms the previous Domain Generalization method, this also confirm the effectiveness of our proposed method.

4.4.4 Further Analysis

To further show the effectiveness of our algorithm especially on more deep models, as suggested by Dou et al. (2019), we also test our algorithm by using ResNet-18 backbone on PACS dataset. The results are reported in Table 4.4.

Method	Art	Cartoon	Sketch	Photo	Avg.
Deep All	77.87	75.89	69.27	95.19	79.55
D-SAM(D'Innocente and Caputo, 2018)	77.33	72.43	77.83	95.30	80.72
JiGen(Carlucci et al., 2019)	79.42	75.25	71.35	96.03	80.51
MASF(Dou et al., 2019)	80.29	77.17	71.69	94.99	81.04
MMLD(Matsuura and Harada, 2020)	81.28	77.16	72.29	**96.09**	81.83
Ours	**81.56**	**78.02**	**78.43**	95.82	**83.45**

TABLE 4.4 – Empirical Results (accuracy %) on PACS dataset with pre-trained ResNet-18 as feature extractor .

	AlexNet					ResNet-18				
Ablation	Art	Carton	Sketch	Photo	Avg.	Art	Carton	Sketch	Photo	Avg.
Deep All	63.30	63.13	54.07	87.70	67.05	77.87	75.89	69.27	95.19	79.55
No $\mathcal{L}_D$	65.80	69.64	63.91	89.53	72.22	74.62	73.02	68.67	94.86	77.79
No $\mathcal{L}_{MS}$	66.78	71.47	68.12	88.87	73.65	78.25	76.27	73.42	95.68	80.91
$\mathcal{L}_{MS}$ w.o. w^+	66.31	70.86	67.11	88.97	73.31	80.58	77.95	75.13	95.63	82.32
$\mathcal{L}_{MS}$ w.o. w^-	66.41	70.95	68.73	87.38	73.37	79.98	77.65	77.89	95.21	82.68
WADG-All	**70.21**	**72.51**	**70.32**	**89.81**	**75.71**	**81.56**	**78.02**	**78.43**	**95.82**	**83.45**

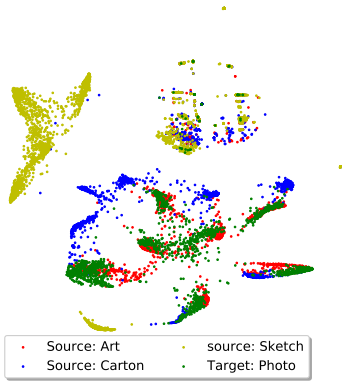
TABLE 4.5 – Ablation Studies on PACS dataset on all components of our proposed method using AlexNet and ResNet-18 backbone

From Table 4.4, we observe that our method outperforms the baselines on most generalization tasks with +1.6% accuracy improvement, achieving state-of-the-art performance.

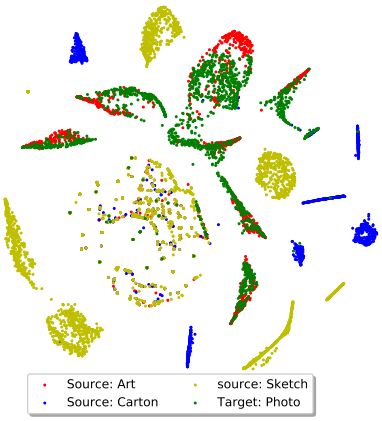
Besides, we also conducted ablation studies on each component of our algorithm. We report the empirical results of ablation studies in Table 4.5, where we test the ablation studies on both the AlexNet backbone and ResNet-18 backbone. We conducted the ablations by :

1. *Deep All* : Train the model using feature extractor on source domain datasets with classification loss only, that is, neither optimal transport nor metric learning techniques is adopted ;
2. *No* $\mathcal{L}_D$: Train the model with classification loss and metric learning loss but without adversarial training component ;
3. $\mathcal{L}_{MS}$ *w.o.* w^+ : omit the positive weighting scheme in $\mathcal{L}_{MS}$;
4. $\mathcal{L}_{MS}$ *w.o.* w^- : omit the positive weighting scheme in $\mathcal{L}_{MS}$;
5. *No* $\mathcal{L}_{MS}$: Train the model with classification loss and adversarial loss but without metric learning component ;
6. *WADG-All* : Train the model with full objective Eq. 4.14.

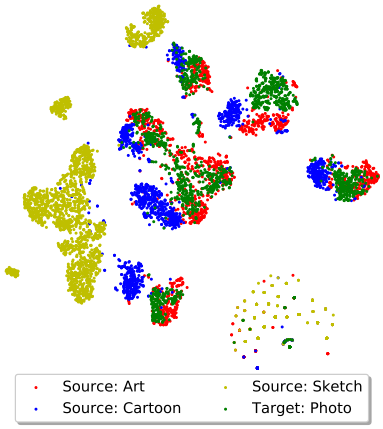
From the ablation results in Table 4.5, we could observe that once we omit the adversarial training, the accuracy would drop off rapidly ($\sim$ 3.5% with AlexNet backbone and $\sim$ 5.8% with ResNet-18 backbone). Comparing the ablations $\mathcal{L}_{MS}$ *w.o.* w^+ and $\mathcal{L}_{MS}$ *w.o.* w^-, we



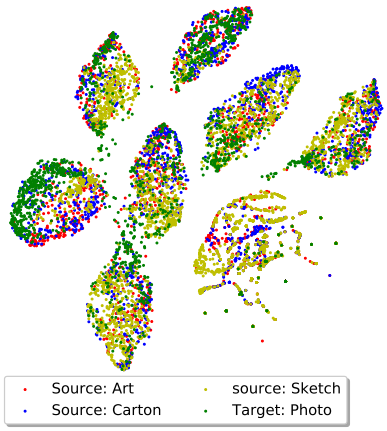
(a) Deep All



(b) No $\mathcal{L}_D$

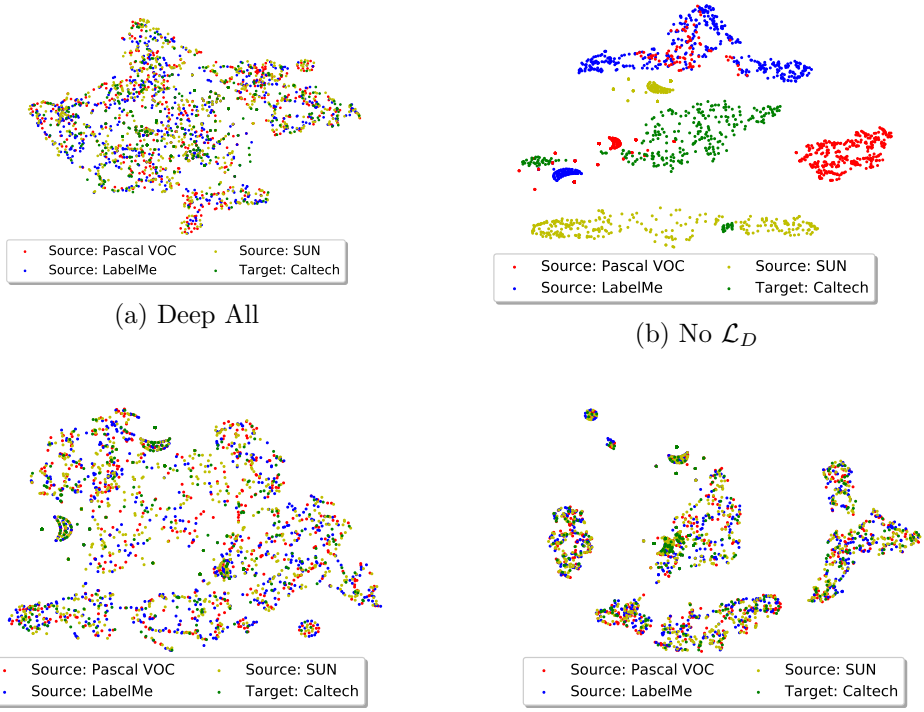


(c) No $\mathcal{L}_{MS}$



(d) WADG-All

FIGURE 4.3 – t-SNE visualization of ablation studies on PACS dataset for target domain as *Photo*. Detailed analysis is presented in section 4.4.4.



(a) Deep All

(b) No $\mathcal{L}_D$

(c) No $\mathcal{L}_{MS}$

(d) WADG-All

FIGURE 4.4 – t-SNE visualization of ablation studies on VLCS dataset for target domain as *Caltech*. Detailed analysis is presented in section 4.4.4.

could observe almost similar accuracy. This indicates that the positive and negative weighting scheme of the metric learning objective have equivalent contributions. Once we reduce the metric learning objecitve, the performance will drop $\sim 2.1\%$ and $\sim 2.5\%$ with AlexNet and ResNet-18 backbone, respectively.

In order to better understand the contribution of each component of our algorithm, the T-SNE visualization of the ablation studies of each components on PACS and VLCS dataset are represented in Fig. 4.3 for the generalization task of target domain *Photo*; and Fig. 4.4 for the generalization task of target domain *Caltech*, respectively. Since our goal is to not only align the feature distribution but also encourage a cohesion and separable boundary, in order to show the alignment and clustering performance, we report the t-SNE features of all the source domains and target domain to show the feature alignment and clustering across domains.

For PASC dataset, as we can see in Fig. 4.3, the t-SNE features by *Deep All* (Fig. 4.3a) are almost randomly displayed, which means the learner could neither project the instances from different domains to align with each other nor to cluster the features into groups. Notice that in the *Deep All* ablation, the learner neither align the marginal distribution nor leveraging the similarities, so the features are not aligned at all, which is consistent with Fig. 4.3a. The

t-SNE features by *No* $\mathcal{L}_D$ (Fig. 4.3b) showed the metric learning loss could to some extent to cluster the features, but without the adversarial training, the features could not be aligned well, which confirms the necessity of adversarial training to align the features. The t-SNE features by *No* $\mathcal{L}_{MS}$ (Fig. 4.3c) showed that the adversarial training could help to align the features from different domains but could not have a good clustering performance, which confirms the effectiveness of metric learning objective on encouraging the classification boundary. Lastly, the T-SNE features by *WADG-All*, of which the learner has both the Wasserstein adversarial training and metric learning objective, showed that the full objective could help to not only align the features from different domains but also could cluster the features from different domains into several cluster groups, which confirms the effectiveness of our algorithm.

As for the VLCS dataset, as shown in Fig. 4.4, we could observe similar performance of the T-SNE on the VLCS dataset while the features are somehow overlap with each other. This is due to the features in Caltech domain is somehow easy to learn and predict. As also analyzed in (Li et al., 2017a), a supervised model on Caltech domain could achieved $\sim 100\%$ accuracy, which also confirms that the features in Caltech domain is easy to learn indicating the features might be more likely overlapping with each other. As we can see from Fig.4.4d, the WADG method could help to separate the features with each other, which again confirms the effectiveness of our proposed method.

4.5 Discussions and Conclusion

In this chapter, we proposed the Wasserstein Adversarial Domain Generalization algorithm, in which we first adopted optimal transport with Wasserstein distance for aligning the marginal distribution and then adopted the metric learning method to encourage a domain-independent distinguishable feature space for a clear classification boundary. Our experimental results showed that our approach outperforms most of the baseline methods on standard benchmark datasets. Furthermore, the ablation studies and visualization of the t-SNE features also confirmed the effectiveness of our algorithm. Besides, we also notice that the metric learning objective involved in this chapter shares some common idea with the supervised contrastive learning (Khosla et al., 2020), which can be an interesting direction in the future. The methodology in this chapter started to investigate the class similarities across different data distributions for multi-source transfer learning problems. In the next chapter, we further investigate the label and semantic information in the multi-task learning problems.

Chapitre 5

Multi-task Learning by Leveraging the Semantic Information

In this chapter, we present our work (Zhou et al., 2021a), where we explored the semantic and task relations in multi-task learning (MTL). We provide a complete theoretical framework to understand the semantic and label distribution divergence in MTL and propose a concrete algorithm showing state-of-the-art empirical performances on several benchmarks, especially when dealing with label distribution shift.

Résumé

L'apprentissage multi-tâches vise à résoudre simultanément les tâches connexes en exploitant les connaissances partagées pour améliorer les performances des tâches individuelles. Un objectif crucial de l'apprentissage multi-tâche est d'aligner les distributions entre les tâches afin que les informations entre elles puissent être transférées et partagées. Cependant, les approches existantes se concentrent uniquement sur la correspondance de la distribution marginale des caractéristiques tout en ignorant les informations sémantiques, ce qui peut entraver les résultats de l'apprentissage. Pour résoudre ce problème, nous proposons d'exploiter les informations des étiquettes dans l'apprentissage multi-tâches en explorant les relations conditionnelles sémantiques entre les tâches. Tout d'abord, nous analysons théoriquement la limite de généralisation de l'apprentissage multi-tâches en nous basant sur la notion de divergence de Jensen-Shannon, ce qui apporte un nouvel aperçu de l'importance des informations des étiquettes dans l'apprentissage multi-tâches. Notre analyse conduit à un algorithme concret qui, conjointement, fait correspondre la distribution sémantique et contrôle la divergence de la distribution des étiquettes. Pour confirmer l'efficacité de la méthode proposée, nous comparons d'abord l'algorithme avec plusieurs baselines sur quelques benchmarks, puis nous les testons dans des conditions de décalage dans l'espace des étiquettes. Les résultats empiriques démontrent que la méthode proposée peut surpasser la plupart des baselines et

atteindre des performances de pointe, en montrant particulièrement les avantages sous les conditions de décalage d'étiquette.

Abstract

Multi-task learning (MTL) aims to solve the related tasks simultaneously by exploiting shared knowledge to improve individual task performance. One crucial objective of MTL 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 our 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 our method could outperform most baselines and achieve state-of-the-art performance, particularly showing the benefits under the label shift conditions.

5.1 Introduction

As we discussed in section 1.4.2, 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., 2017a), when analyzing patient data in healthcare data analysis (Wang and Pineau, 2015), or when modelling users' products preferences Murugesan and Carbonell (2017). Data hungry has become a long-term problem for deep learning (Aggarwal et al., 2018).

Multi-task learning (MTL) aims at addressing this issue by simultaneously learning from multiple related 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 et al., 2018h), brain-computer interaction (Wang et al., 2020a) or cross-modality (Nguyen and Okatani, 2019) learning problems. It has been shown with benefits that with MTL, one can reduce the amount of annotated data per task to reach the desired performance.

In the previous Chapter 4, we explored the domain generalization (DG) problems, where we train the model on several source data distributions and then test it on an unseen target domain. In the MTL problems, we train the model on several tasks using limited data and test it on the same tasks with full data. A general framework of the MTL problems involved in this chapter can be recalled with Fig. 1.8 in section 1.4.2.

The crucial idea behind MTL is to extract and leverage the knowledge and information shared across the tasks to improve the overall performance (Wang et al., 2019b), and this can be achieved by task-invariant feature learning (Maurer et al., 2016; Luo et al., 2017a) or task relation learning (Zhang and Yeung, 2010; Bingel and Søgaard, 2017). As summarized in section 2.3.2, one major issue with most of the existing feature learning approaches is that they only align the marginal distributions $\mathcal{D}(\mathbf{x})$ to extract the shared features without taking advantage of label information $\mathcal{D}(y)$ of the tasks, knowing that $(\mathbf{x}, y)$ reflects data and label, respectively. 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 $\mathcal{D}(\mathbf{x})$ 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).

As discussed in section 2.3.2, while a few algorithms have been proposed to use semantic matching, *i.e.* alignment the semantic distributions $\mathcal{D}(\mathbf{x}|y)$ over all the tasks, for MTL by Zhuang et al. (2017) and Luo et al. (2017a), and have shown improved performances, the theoretical justifications for the value of labels remain elusive. Most existing theoretical results (Shui et al., 2019; Mao et al., 2020) for MTL were derived from the notion of $\mathcal{H}$-divergence (Ben-David et al., 2010a) or Wasserstein adversarial training (Redko et al., 2017; Shen et al., 2018), and both did not take the label information into consideration. As a result, Shui et al. (2019) and Mao et al. (2020)they usually require additional assumptions, *e.g.*, assuming the combined error across tasks is small (Ben-David et al., 2010a) (see section 1.5.2) to ensure the algorithms succeed, which may not hold in practice.

As we see semantic matching is important and we proposed a theoretical analysis for MTL that considers it. Specifically, our results revealed 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 $\mathcal{D}(y)$ and semantic distribution $\mathcal{D}(\mathbf{x}|y)$.

The contributions of our work in this chapter are trifold :

1. In contrast to previous theoretical results (Shui et al., 2019; Mao et al., 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 insight into how to extract and leverage shared knowledge in a more appropriate and principled way by exploiting the label information $\mathcal{D}(y)$.

2. Our analysis also reveals how the label shift 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 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 our algorithm, we evaluated SMTL on several benchmarks. The empirical results show that our 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 method. Furthermore, we also conduct a simulation of the label distribution shift scenario showing that our algorithm could handle the label distribution shift problems that cannot be properly addressed by other baselines.

5.2 Preliminaries

In this section, we first recall some necessary notations and preliminary problem setup, which we have presented in section 1.4.2.

5.2.1 Problem setup

In the previous Chapter 4, we tackled the domain generalization problems where we trained the model on several source domains and test it on the target domain. Here, in the MTL problems, we train the model on several tasks with limited data on some data distributions and then test this model on the same task data distributions.

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 $f_t : \mathcal{X} \to \mathcal{Y}$ for $\{(\mathcal{D}_t, f_t)\}_{t=1}^T$. A MTL learner aims to find T hypothesis : $h_1, \ldots, h_T$ over the hypothesis space $\mathcal{H}$ to minimize the average expected risk of all the tasks :

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

where $R_i(h_i) \equiv R_i(h_i, f_i) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_i} \ell(h_i(\mathbf{x}), f_i(\mathbf{x}))$ 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,

$$\hat{R}_{\boldsymbol{\alpha}_i}(h) = \sum_{j=1}^{T} \boldsymbol{\alpha}_{i,j} \hat{R}_j(h), \tag{5.1}$$

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

Most of existing adversarial based MTL approaches, (e.g. Mao et al., 2020; Shui et al., 2019), were motivated by the theory of Ben-David et al. (2010a) using the $\mathcal{H}$ divergence. However, the $\mathcal{H}$-divergence theory itself is limited in many scenarios. As we discussed in section 1.5, $\mathcal{H}$-divergence deals with the marginal feature distribution $\mathcal{D}(\mathbf{x})$. It will be prohibitive to use $\mathcal{H}$-divergence in some learning scenarios, *e.g.*, when tackling the (semantic) conditional shifts and understanding open set learning problems (Busto and Gall, 2017; Cao et al., 2018b; You et al., 2019). In this work, we adopt the *Jensen-Shannon Divergence* (D_{JS}, introduced in section 1.5) to measure the differences of tasks and analyze its potentials for controlling the semantic (covariate) relations $\mathcal{D}(\mathbf{x}|y)$ *i.e.* measure the divergence between the tasks.

Definition 5.1 (Jensen-Shannon divergence)**.** *Let $\mathcal{D}_i(\mathbf{x}, y)$ and $\mathcal{D}_j(\mathbf{x}, y)$ 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.

As summarized in section 1.5.1, Jensen-Shannon divergence has been implemented in adversarial training based approaches in transfer learning (Dou et al., 2019; Matsuura and Harada, 2020; Zhao et al., 2019a). In practice, we could compute the *Total Variation* distance (d_{TV}) since it is an upper bound of JS divergence (Lin, 1991) :

$$d_{TV}(\mathcal{D}_i, \mathcal{D}_j) = \frac{1}{2}|\mathcal{D}_i - \mathcal{D}_j| \tag{5.2}$$

5.2.2 Leverage the semantic and label information

As aforementioned, previous MTL approaches (e.g. Mao et al., 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 $Y = y$, it is clear that $\mathcal{D}_i(\mathbf{x}|Y = y) \neq \mathcal{D}_j(\mathbf{x}|Y = y)$, indicating the necessity of considering semantic information in MTL problems.

Moreover, a long-neglected issue in existing MTL approaches is that most MTL approaches all implicitly assumed that the label marginal distribution ($\mathcal{D}(y)$) 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. As introduced in section 1.3.2, *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 (Busto and Gall, 2017; Geng et al., 2020; Azizzadenesheli et al., 2019), the analysis towards label space shift in MTL is, however, still open.

We show, in next section, 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 $\mathcal{D}(\mathbf{x}|y)$ 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 shift.

5.3 Methodology and 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 analyzed 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 of the theoretical results in this chapter are presented in Appendix B.2.

Theorem 5.1. *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*

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h) \leq \frac{1}{T}\sum_{t=1}^{T} R_{\alpha_t}(h) + \frac{\lambda_0}{4T}L^2 + \frac{2}{\lambda_0 T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} D_{JS}(\mathcal{D}_t(\mathbf{x}, y)\|\mathcal{D}_i(\mathbf{x}, y)) \tag{5.3}$$

where $\lambda_0 > 0$ is a constant.

Theorem 5.1 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. Minimizing this bound is equivalent to minimizing the supervised classification loss and aligning the distribution of all the tasks so that the divergence term is also minimized.

Note that the bound in Theorem 5.1 indicates the joint distribution while we aim to leverage the label ($\mathcal{D}(y)$) and semantic ($\mathcal{D}(\mathbf{x}|y)$) information. To do that, we can then decompose the aforementioned Theorem 5.1 into the following result,

Corollary 5.1. *Follow the setting of Theorem 5.1, we can further bound the overall task risk by*

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} R_t(h) \leq \frac{1}{T}\sum_{t=1}^{T} R_{\alpha_t}(h) + \mathbf{A} \underbrace{D_{JS}(\mathcal{D}_t(y)\|\mathcal{D}_i(y))}_{\textit{Label distribution divergence}} \\ + \mathbf{A}\underbrace{\mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{JS}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{D}_i(\mathbf{x}|y))}_{\textit{Semantic distribution divergence}} \\ + \mathbf{A}\underbrace{\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{JS}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{D}_i(\mathbf{x}|y))}_{\textit{Semantic distribution divergence}} + \frac{\lambda_0}{4T}L^2\end{aligned} \tag{5.4}$$

where $\mathbf{A} \in \mathbb{R}^{T\times T}$ *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 previous theoretical results (Shui et al., 2019; Mao et al., 2020), which were motivated by Ben-David et al. (2010a) and Redko et al. (2017), our theoretical results reflected by Eq. 5.4 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 taking advantage of label information and semantic conditional relations.

Corollary 5.1 implies that the averaged error over all tasks is bounded by the summation of task errors (the first term in *R.H.S.* of Corollary 5.1), 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 $\frac{1}{T}\sum_{t=1}^{T} R_{\alpha_t}(h)$ can be easily optimized by a general supervised learning loss (*e.g.* the cross-entropy loss). To minimize the remaining terms now is equivalent to match the semantic distribution among the tasks by measuring the label divergence. Since the labels of each task samples are available to the learner, we can leverage the label and semantic information directly.

5.3.1 Label Re-weighting Loss

Corollary 5.1 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 et al., 2018) based on the number of instances in each class,

$$\hat{R}^{\beta}_{\mathcal{D}_i}(h) = \sum_{(\mathbf{x}_i,y_i)\in\hat{\mathcal{D}}_i} \boldsymbol{\beta}(y_i)\ell(h(\mathbf{x}_i), y_i)) \tag{5.5}$$

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 N_i instances, the weight of class $k \in K$ (K is the total number of classes) is computed by $\boldsymbol{\beta}_k = \sum \frac{|\#y=y_k|}{N_i}$. This re-weighting scheme guarantee the instances from different classes could have equal probability to be sampled when training the model (see Fig. 5.1), which re-weights the loss according to frequency of each class that occurs during training. By doing

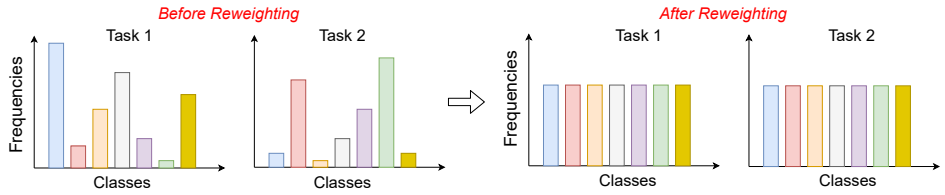


FIGURE 5.1 – Example of re-weighting scheme. Left : The number of instances (frequencies) in each class between task 1 and task 2 are different. Right : After the re-weighting process, the frequencies of each class becomes the same.

so, the learner will not neglect those classes who have fewer instances and therefore takes care of label shift.

Note : the coefficient $\boldsymbol{\beta}$ is computed for re-weighting the loss from each task (as defined in section 5.2.1) 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 $\mathcal{L}_i = \hat{R}^{\beta}_{\mathcal{D}_i}(h)$ and compute the total classification loss

$$\mathcal{L}_C = \sum_{i=1}^{T} \boldsymbol{\alpha}_i \hat{R}^{\beta}_{\mathcal{D}_i}(h) \tag{5.6}$$

Semantic matching and task relation update

To compute Eq. 5.6, 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 5.1, we could solve the coefficients via an convex optimization as

$$\begin{aligned} \min_{\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_T} \quad & \mathcal{L}_C + \sum_{i,t=1}^{T} \boldsymbol{\alpha}_{t,i} \sum\nolimits_y (\hat{\mathcal{D}}_i(y) + \hat{\mathcal{D}}_t(y))) \mathbf{E}_{i,t} + \sum_{t=1}^{T} ||\boldsymbol{\alpha}_t||_2 \\ \text{s.t.} \quad & \sum_t \boldsymbol{\alpha}_t = 1 \end{aligned} \tag{5.7}$$

where $\mathbf{E}_{i,t} = D_{\text{JS}}\left(\hat{\mathcal{D}}_t(\mathbf{x}|y) \| \hat{\mathcal{D}}_t(\mathbf{x}|y)\right)$ 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 $z^k_{\mathcal{D}_i}$ and $z^k_{\mathcal{D}_j}$ by two feature centroids from class k of distribution $\mathcal{D}_i$ and $\mathcal{D}_j$ respectively, it could be computed by,

$$\Phi(z^k_{\mathcal{D}_i}, z^k_{\mathcal{D}_j}) = ||z^k_{\mathcal{D}_i} - z^k_{\mathcal{D}_j}||^2 \tag{5.8}$$

Our goal is to match the semantic distribution across tasks. For this, we re-visited the *moving average centroid* method proposed by Xie et al. (2018) where a global centroid matrix was maintained to compute the semantic information between a labeled source and an unlabeled

Algorithm 3 The Global Semantic Matching Method.

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

for k=1 to K **do**
  $z^k_{\mathcal{D}_i} \leftarrow \frac{1}{|\mathcal{D}^k_i|} \sum_{(\mathbf{x}_i, y_i) \in \mathcal{D}^k_i} F(\mathbf{x}_i)$;
  $z^k_{\mathcal{D}_j} \leftarrow \frac{1}{|\mathcal{D}^k_j|} \sum_{(\mathbf{x}_j, y_j) \in \mathcal{D}^k_j} F(\mathbf{x}_j)$;
  $\mathcal{Z}^k_{\mathcal{D}_i} \leftarrow \gamma z^k_{\mathcal{D}_i} + (1-\gamma) z^k_{\mathcal{D}_i}$;
  $\mathcal{Z}^k_{\mathcal{D}_j} \leftarrow \gamma z^k_{\mathcal{D}_j} + (1-\gamma) z^k_{\mathcal{D}_j}$;
  $\mathcal{L}_S \leftarrow \mathcal{L}_S + \Phi(\mathcal{Z}_{\mathcal{D}_i}, \mathcal{Z}_{\mathcal{D}_j})$;
end for
return $\mathcal{L}_S$.

target distribution for domain adaptation problem. Unlike Xie et al. (2018), however, we 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. 3.

Remark : Through Algorithm 3, the semantic loss $\mathcal{L}_S$ is an approximation of the *total variation distance* (see Eq. 5.2) of the two centroids, which is an upper bound of $\mathcal{D}_{\text{JS}}\left(\mathcal{D}_i(\mathbf{x}|y), \mathcal{D}_j(\mathbf{x}|y)\right)$. 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.

Algorithm 4 The Semantic Multi-task Learning (SMTL) Algorithm.

Input: Samples from different tasks $\{\hat{\mathcal{D}}_t\}_{t=1}^T$, initial coefficients $\{\boldsymbol{\alpha}_t\}_{t=1}^T$ and learning rate η.
Output: Neural network $\boldsymbol{\theta}_f$, $\{\boldsymbol{\theta}^t_c\}_{t=1}^T$ and coefficient $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_T$.

while Algorithm Not converge **do**
  for min-batch $\{(\mathbf{x}^b_t, y^b_t)\}$ from task $\{\hat{\mathcal{D}}_t\}_{t=1}^T$ **do**
    Compute the classification objective $\mathcal{L}_C$ by Eq. 5.6 ;
    Compute the semantic matching objective $\mathcal{L}_S$ using Algorithm 3 ;
    Update the network parameters $\boldsymbol{\theta}_f, \boldsymbol{\theta}^t_c$ by :
    $\boldsymbol{\theta}_f \leftarrow \boldsymbol{\theta}_f - \eta \frac{\partial \mathcal{L}_C + \mathcal{L}_S}{\partial \boldsymbol{\theta}_f}$ and $\boldsymbol{\theta}^t_c \leftarrow \boldsymbol{\theta}^t_c - \eta \frac{\partial \mathcal{L}_C + \mathcal{L}_S}{\partial \boldsymbol{\theta}^t_c}$;
  end for
  Update $\{\boldsymbol{\alpha}_t\}_{t=1}^T$ by optimizing over Eq. (5.7).
end while

5.3.2 The full objective and algorithm

A general model architecture is provided in Fig. 5.2. 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

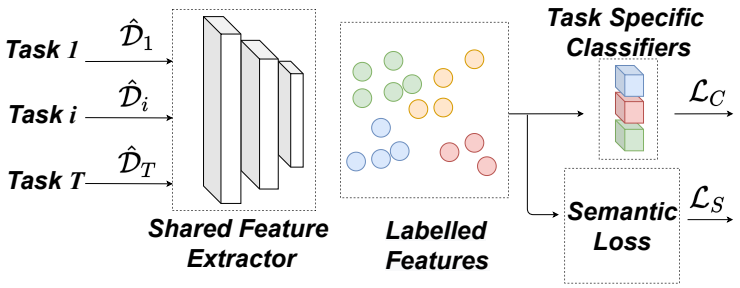


FIGURE 5.2 – The overall model architecture of the SMTL method.

semantic transfer objective. The Semantic Multi-task Learning (SMTL) method is illustrated in Algorithm 4

5.4 Experiments and Analysis

We examined our approach comparing with several baselines on **Digits**, **PACS** (Li et al., 2017b), **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 our algorithms on *MNIST*, *MNIST-M* and *SVHN* simultaneously. The PACS dataset, which was widely used in recent transfer learning researches, consists of images from four tasks : *Photo* (P), *Art painting* (A), *Cartoon* (C), *Sketch* (S), with objects from 7 classes. Office-31 dataset is a vision benchmark widely used in transfer learning related problems which consists of three different tasks : *Amazon*, *Dslr* and *Webcam* ; Office-Caltech contains the 10 shared categories between the Office-31 dataset and Caltech256 dataset, including four different tasks : *Amazon*, *Dslr*, *Webcam* and *Caltech* ; Office-home is a more challenging benchmark, which contains four different tasks : *Art*, *Clipart*, *Product* and *Real World*, with 65 categories in each task.

To evaluate the performance of our 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 : $\frac{1}{T}\sum_{t=1}^{T}\hat{R}_t(\boldsymbol{\theta}_f, \boldsymbol{\theta}_c^t)$, *i.e.*, compute the loss uniformly.

— **Weighted MTL** : Learning a weighted summation of losses over different tasks : $\frac{1}{T}\sum_{t=1}^{T}\hat{R}_{\boldsymbol{\alpha}_t}(\boldsymbol{\theta}_f, \boldsymbol{\theta}_c^t)$; an approach adapted from Murugesan et al. (2016).

— **Adv.H** : Using the same loss function of (Liu et al., 2017a) while training with $\mathcal{H}$-divergence as adversarial objective ; an approach adapted from Liu et al. (2017a).

— **Adv.W** : Replacing the adversarial loss of Adv.H by Wasserstein distance based adversarial training method.

— **Multi-Obj.** : Casting the multi-task learning problem as a multi-objective problem ; an approach adapted from Sener and Koltun (2018).

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

	3K				5K				8K			
Approach	**MNIST**	**MNIST-M**	**SVHN**	**Avg.**	**MNIST**	**MNIST-M**	**SVHN**	**Avg.**	**MNIST**	**MNIST-M**	**SVHN**	**Avg.**
Vanilla	93.9 ± 3.2	77.1 ± 2.6	57.3 ± 0.4	76.1	96.3 ± 1.2	79.1 ± 3.1	68.0 ± 2.9	81.1	97.7 ± 0.5	83.7 ± 2.2	71.4 ± 0.9	84.2
Weighed	89.3 ± 3.3	76.4 ± 3.1	70.2 ± 1.8	78.3	91.8 ± 2.7	74.2 ± 0.9	73.6 ± 3.1	79.8	92.3 ± 2.6	76.9 ± 3.1	74.1 ± 1.6	81.1
Adv.H	90.1 ± 1.2	$\mathbf{81.2} \pm 1.3$	70.8 ± 0.5	80.7	91.9 ± 2.6	$\mathbf{83.7} \pm 1.4$	73.6 ± 1.6	82.9	94.9 ± 1.6	$\mathbf{85.2} \pm 0.3$	79.1 ± 0.3	86.4
Adv.W	96.8 ± 0.6	81.3 ± 0.7	69.5 ± 1.1	82.5	97.5 ± 0.2	83.4 ± 0.4	72.6 ± 1.2	84.5	98.1 ± 0.3	84.3 ± 0.4	75.4 ± 1.1	86.1
Multi-Obj.	$\mathbf{97.5} \pm 0.3$	76.9 ± 0.5	54.8 ± 0.3	76.4	$\mathbf{98.2} \pm 0.2$	80.2 ± 0.7	61.2 ± 0.8	79.9	$\mathbf{98.5} \pm 0.3$	82.8 ± 0.5	69.9 ± 0.9	83.7
AMTNN	96.9 ± 0.2	80.8 ± 1.5	77.1 ± 0.9	84.9	97.7 ± 0.1	83.6 ± 1.1	78.4 ± 0.8	86.6	98.1 ± 0.2	83.1 ± 2.1	80.2 ± 1.3	87.1
Ours	95.4 ± 0.3	80.1 ± 0.5	$\mathbf{81.5} \pm 0.6$	**85.7**	95.8 ± 0.3	82.4 ± 0.4	$\mathbf{83.3} \pm 0.3$	**87.2**	96.0 ± 0.3	83.9 ± 0.4	$\mathbf{85.4} \pm 0.2$	**88.4**

TABLE 5.1 – The empirical results (in %) on the digits datasets.

	10%					15%					20%				
Approach	**A**	**C**	**P**	**S**	**avg.**	**A**	**C**	**P**	**S**	**avg.**	**A**	**C**	**P**	**S**	**avg.**
Vanilla MTL	78.8 ± 1.1	81.8 ± 1.4	87.1 ± 0.4	83.4 ± 0.6	82.8	82.8 ± 0.8	86.7 ± 0.9	89.3 ± 0.7	84.9 ± 0.7	85.9	84.1 ± 0.9	87.9 ± 0.8	90.5 ± 0.6	85.8 ± 1.2	87.1
Weighted	82.7 ± 1.1	86.2 ± 0.6	89.7 ± 1.1	84.9 ± 0.7	85.9	$\mathbf{85.1} \pm 0.9$	87.9 ± 0.7	91.2 ± 1.0	87.3 ± 0.7	87.9	86.1 ± 0.6	89.7 ± 0.4	92.1 ± 1.2	88.4 ± 0.9	89.1
Adv.W	78.8 ± 1.1	83.9 ± 1.2	87.6 ± 1.4	84.0 ± 0.9	83.6	83.6 ± 1.4	84.8 ± 0.5	84.7 ± 0.7	84.0 ± 0.6	84.2	83.5 ± 0.5	89.5 ± 0.7	91.4 ± 0.5	87.3 ± 0.6	87.9
Adv.H	76.8 ± 1.6	84.3 ± 0.4	88.3 ± 0.6	84.0 ± 0.6	83.3	82.6 ± 0.8	87.8 ± 0.7	89.9 ± 0.7	86.1 ± 0.4	86.6	84.4 ± 0.7	87.6 ± 0.3	91.5 ± 0.4	88.3 ± 0.4	87.9
Multi-Obj.	79.4 ± 1.8	83.4 ± 1.3	87.0 ± 0.5	82.9 ± 1.1	83.2	82.7 ± 0.5	87.5 ± 0.4	89.1 ± 0.6	86.5 ± 0.5	86.4	84.3 ± 0.9	88.7 ± 0.4	91.0 ± 0.3	88.8 ± 0.7	88.2
AMTNN	$\mathbf{82.8} \pm 0.4$	86.7 ± 0.4	91.3 ± 0.9	81.2 ± 0.8	85.5	85.1 ± 0.3	88.8 ± 0.3	92.9 ± 0.4	85.8 ± 0.5	88.2	$\mathbf{87.4} \pm 0.2$	89.9 ± 0.6	93.7 ± 0.4	87.7 ± 0.1	89.7
Ours	80.6 ± 0.4	$\mathbf{87.9} \pm 0.4$	$\mathbf{94.4} \pm 0.5$	$\mathbf{91.9} \pm 0.5$	**88.9**	83.4 ± 0.4	$\mathbf{89.8} \pm 0.6$	$\mathbf{94.5} \pm 0.4$	$\mathbf{93.1} \pm 0.4$	**90.2**	86.3 ± 0.3	$\mathbf{91.6} \pm 0.4$	$\mathbf{95.5} \pm 0.4$	$\mathbf{93.8} \pm 0.4$	**91.8**

TABLE 5.2 – The empirical results (in %) on PACS dataset with AlexNet as feature extractor.

5.4.1 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 randomly selecting $3k$, $6k$ and $8k$ instances of the training dataset and choose $1k$ dataset as validation set while testing on the full test set. For the SVHN dataset, whose original image size is 32×32, we resize the images to $28 \times 28 \times 1$, except for that, we do not apply any data-augmentation towards to digits dataset. We used LeNet-5 model (LeCun et al., 1998) as the feature extractor and deployed three 3-layer MLPs as task-specific classifiers. We extract the semantic feature from the second layer of the classifier with size 128. We adopted the Adam optimizer Kingma and Ba (2015) for training the model from scratch. The model is trained for 50 epochs while the initial learning rate is set to 1×10^{-3} and is decayed 5% for every 5 epochs. The results are reported in Table 5.1.

For computer vision applications, we then test the SMTL algorithm comparing with the baselines on PACS and Caltech datasests using the AlexNet by (Krizhevsky et al., 2012) as feature extractor. For investigating the performance when limited amount labelled instances are available, we evaluated our algorithms on PACS dataset randomly selecting 10%, 15% and 20% of the total dataset for training, respectively. Since the 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. On top of the feature extractor, we implemented several MLPs as task-specific classifiers. Our test results are reported in Table 5.2 and Table 5.3, respectively.

After that, we evaluated our algorithm on Office-31 and Office-Home dataset by randomly selecting 5%, 10% and 20% training samples with pre-trained ResNet-18 model of *PyTorch* while removing the last FC layers as feature extractor. For these four vision benchmarks we follow the pre-processing and train/val/test protocol proposed by Long et al. (2017a); Cao et al. (2018a) and Li et al. (2017b). 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 5.4 and 5.5, respectively.

Discussion on our experimental results

From Table. 5.1$\sim$ 5.5, we observe that our method outperforms the baselines and improve the benchmark performances. Specifically, on the Digits dataset (Table 5.1), we observe that the baseline some baselines have better performance on a specific task with a minor gap, for example, the baseline *Adv.H* has better performances on MNIST-M but has poorly performance on SVHN (almost 10% worse than ours). And we see from Table 5.1 that our method has better-averaged performance across all the tasks, which is coherent to the goal of MTL to improve the overall performance. The reason for the difference on MNIST-M and SVHN

Method	Amazon	Caltech	Dslr	WebCam	Average
Vanilla	84.2 ± 1.1	80.6 ± 0.8	90.8 ± 2.3	81.8 ± 0.9	84.3
Weighted	88.1 ± 0.2	81.5 ± 0.9	94.9 ± 0.2	94.2 ± 0.5	88.6
Adv.H	81.5 ± 0.5	73.8 ± 1.8	91.4 ± 2.1	86.1 ± 1.4	83.3
Adv.W	84.9 ± 0.4	80.9 ± 0.9	94.5 ± 2.2	87.5 ± 1.5	86.9
Multi-Obj.	82.3 ± 0.7	76.7 ± 2.4	91.2 ± 1.7	86.8 ± 0.9	84.3
AMTNN	89.3 ± 0.9	84.3 ± 0.6	$\mathbf{98.4} \pm 1.3$	94.1 ± 0.7	91.7
Ours	$\mathbf{90.9} \pm 0.4$	$\mathbf{85.3} \pm 0.5$	98.1 ± 0.8	$\mathbf{94.2} \pm 0.6$	**92.1**

TABLE 5.3 – Average test accuracy (in %) of MTL algorithms on Office-Caltech dataset with pre-trained AlexNet as feature extractor.

between our methods with the baslines is that MNIST-M shares some commonalities with MNIST (*e.g.* digit style) while having apparent semantic differences from SVHN, which leads to the baselines' poor performance on SVHN. Our method can learn $\mathcal{D}(y)$ and $\mathcal{D}(\mathbf{x}|y)$ across tasks, which helps to prevent overfitting on MNIST-M and have balanced performance. As for the performance on the rest four computer vision benchmarks, we can observe that our methods always outperform the baselines.

Effectiveness with Limited Data : Particularly, from Table. 5.1$\sim$ 5.5, we found that when there are only few of labelled instances (*e.g.* 5% or 10% of the total instances), our method could have a large margin of improvements regarding the baselines. For example, when we only train the model with 5% of the total instances, we have higher accuracy compared with the baselines performance. *These improvements confirm the effectiveness of our methods when dealing with limited data.*

5.4.2 Further Analysis

Ablation Studies We conduct ablation studies of our 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 ;
4. *w.o.* cvx. opt. : omit the optimization procedure for updating $\boldsymbol{\alpha}$, *i.e,* Eq. (5.7).

The results of ablation studies are reported in Table 5.6. The ablations showed that the label re-weighting scheme is crucial for the algorithm since the performance can drop with $\sim 2\%$ if we remove the re-weighting scheme. We also observe -1.0% drop when omitting the semantic matching procedure, this also indicates the importance of controlling the semantic divergence. As for the task relations update scheme (Eq. 5.7), we observe -0.5% performance drop once we omit the convex optimization procedure for $\boldsymbol{\alpha}$.

Approach	5%				10%				20%			
	Amazon	**Dslr**	**Webcam**	**avg.**	**Amazon**	**Dslr**	**Webcam**	**avg.**	**Amazon**	**Dslr**	**Webcam**	**avg.**
Vanilla	61.3 ± 1.3	71.8 ± 2.1	72.1 ± 1.1	68.3	73.2 ± 0.5	80.6 ± 1.4	82.1 ± 0.9	78.6	79.4 ± 0.8	91.2 ± 1.0	93.1 ± 0.8	87.9
Weighted	63.3 ± 0.2	87.4 ± 2.3	84.9 ± 0.6	78.5	70.6 ± 1.2	92.1 ± 0.9	88.4 ± 1.3	83.7	76.8 ± 0.9	$\mathbf{96.6} \pm 0.7$	95.6 ± 0.5	89.7
Adv.W	66.5 ± 1.9	71.8 ± 1.1	69.9 ± 0.9	69.7	74.7 ± 1.1	85.9 ± 0.8	85.7 ± 0.8	82.1	79.3 ± 0.6	93.8 ± 0.4	92.2 ± 0.9	88.4
Adv.H	65.8 ± 1.1	73.5 ± 0.8	71.4 ± 0.7	70.2	71.0 ± 0.9	84.1 ± 0.9	89.4 ± 0.1	81.4	79.7 ± 0.5	93.7 ± 0.7	93.7 ± 0.6	89.1
Multi-Obj.	$\mathbf{68.9} \pm 1.2$	72.5 ± 1.4	72.3 ± 0.4	71.3	74.6 ± 0.9	86.8 ± 1.1	86.9 ± 0.8	82.8	79.2 ± 0.8	92.1 ± 0.6	94.7 ± 0.6	88.6
AMTNN	63.3 ± 0.6	80.1 ± 1.6	85.4 ± 0.3	79.3	71.3 ± 1.2	$\mathbf{92.8} \pm 0.9$	89.6 ± 1.2	84.6	80.2 ± 0.9	94.2 ± 1.2	94.4 ± 0.9	89.6
Ours	68.5 ± 0.6	$\mathbf{87.9} \pm 0.8$	$\mathbf{86.5} \pm 0.5$	**80.9**	$\mathbf{75.7} \pm 0.2$	$\mathbf{92.8} \pm 0.2$	$\mathbf{90.8} \pm 0.3$	**86.4**	$\mathbf{81.1} \pm 0.2$	96.5 ± 0.1	$\mathbf{96.1} \pm 0.2$	**91.2**

TABLE 5.4 – The empirical results (in %) on Office-31 dataset with pre-trained ResNet-18 as feature extractor.

Approach	5%					10%					20%				
	Art	**Clipart**	**Product**	**Real-world**	**avg.**	**Art**	**Clipart**	**Product**	**Real-world**	**avg.**	**Art**	**Clipart**	**Product**	**Real-world**	**avg.**
Vanilla	26.2 ± 0.3	30.1 ± 0.2	57.6 ± 0.1	47.4 ± 1.1	40.3	35.8 ± 0.7	43.3 ± 0.6	67.1 ± 0.4	56.8 ± 1.3	50.7	45.5 ± 0.8	56.1 ± 0.6	74.4 ± 0.7	62.6 ± 0.6	59.6
Weighted	26.8 ± 1.6	31.8 ± 1.8	59.2 ± 0.4	50.5 ± 1.2	42.1	38.2 ± 1.0	45.3 ± 1.6	69.1 ± 0.2	58.3 ± 0.8	52.7	47.9 ± 0.1	56.7 ± 0.9	75.6 ± 0.6	64.8 ± 0.9	61.2
Adv.W	26.8 ± 0.8	32.7 ± 0.5	58.3 ± 0.9	47.1 ± 0.4	41.2	38.5 ± 0.8	44.4 ± 0.7	67.6 ± 0.7	59.5 ± 0.9	52.3	47.9 ± 0.5	56.7 ± 0.6	75.4 ± 1.1	65.7 ± 0.8	61.3
Adv.H	27.7 ± 1.4	32.1 ± 1.5	59.6 ± 0.7	51.1 ± 0.9	42.7	39.0 ± 0.9	45.8 ± 1.8	69.4 ± 0.4	58.8 ± 0.6	53.2	46.7 ± 0.5	56.5 ± 1.1	75.6 ± 0.4	65.1 ± 0.7	61.0
Multi-Obj.	25.6 ± 1.5	31.7 ± 1.7	58.7 ± 1.3	51.5 ± 0.9	41.8	34.6 ± 0.9	43.3 ± 1.4	66.1 ± 1.5	56.8 ± 0.7	50.2	46.2 ± 0.8	56.6 ± 0.5	74.3 ± 0.7	62.8 ± 0.6	59.8
AMTNN	32.5 ± 1.3	34.5 ± 0.9	56.3 ± 0.8	49.9 ± 1.8	43.3	41.1 ± 1.0	47.5 ± 0.8	68.4 ± 0.7	58.9 ± 0.9	53.9	48.9 ± 0.5	$\mathbf{60.7} \pm 0.4$	75.4 ± 0.4	64.7 ± 0.4	62.1
Ours	$\mathbf{38.3} \pm 0.9$	$\mathbf{40.9} \pm 0.9$	$\mathbf{62.3} \pm 0.8$	$\mathbf{55.5} \pm 0.6$	**49.2**	$\mathbf{43.8} \pm 0.6$	$\mathbf{50.4} \pm 0.8$	$\mathbf{71.3} \pm 0.9$	$\mathbf{62.3} \pm 0.6$	**57.1**	$\mathbf{51.2} \pm 0.7$	60.6 ± 0.8	$\mathbf{77.9} \pm 0.4$	$\mathbf{66.1} \pm 0.6$	**64.3**

TABLE 5.5 – The empirical results (in %) on Office-home dataset with pre-trained ResNet-18 as feature extractor.

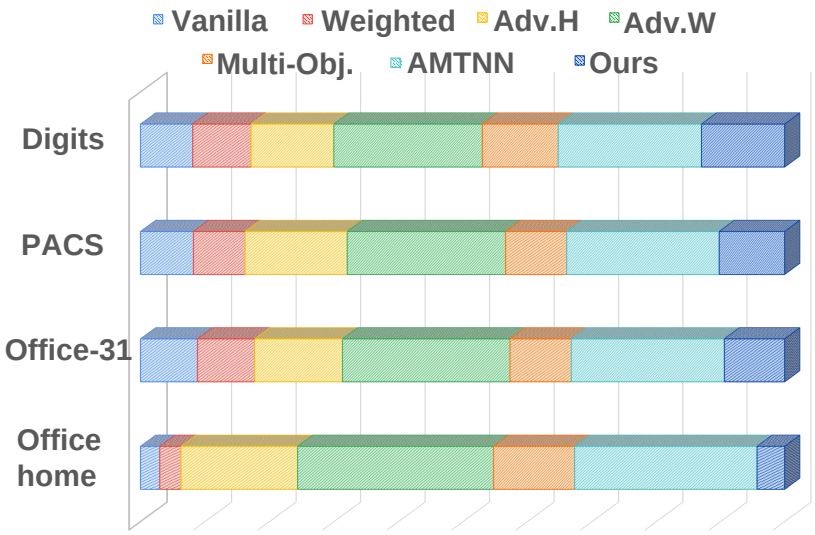


FIGURE 5.3 – Relative time comparison (one training epoch) of the MTL algorithm on different benchmarks.

Time Efficiency As our method doesn't rely on adversarial training, it has better time efficiency. We compare the time-efficiency of both our algorithm and baselines 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. 5.3. 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 5.1∼5.5) into consideration, we see that our method improves the benchmark accuracy performance while reducing the time needed for training. *This also demonstrates the benefits of algorithm in terms of time-efficiency.*

Performance under Label Shift We evaluated the performance of our MTL algorithm and baselines 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

Method	**Amazon**	**Dslr**	**WebCam**	**Average**
Cls. only	79.4 ± 0.8	91.2 ± 1.0	93.1 ± 0.8	87.9
w.o. re-weighting	80.2 ± 0.7	94.7 ± 1.3	94.1 ± 0.8	89.6
w.o. sem. matching	79.8 ± 0.5	96.1 ± 0.3	95.4 ± 0.3	90.4
w.o. cvx opt.	80.7 ± 0.3	$\mathbf{96.8} \pm 0.5$	95.3 ± 0.4	90.9
Full method	$\mathbf{81.1} \pm 0.2$	96.5 ± 0.1	$\mathbf{96.1} \pm 0.2$	**91.2**

TABLE 5.6 – Ablation studies on Office-31 dataset.

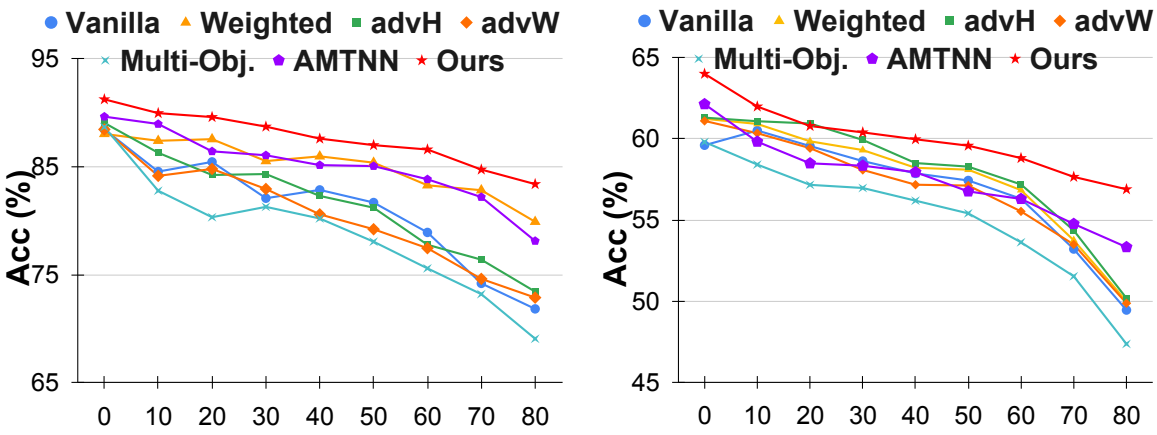


FIGURE 5.4 – Performance comparison under label distribution drift scenario. *Left* : Evaluations on Office-31 dataset with different drift ratio ; *Right* : Evaluations on Office-Home dataset with different drift ratio.

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 ($10\% \sim 80\%$).

In Fig. 5.4, we show the performance under label distribution drift ranging from $10\% \sim 80\%$ on Office-31 dataset (left) and on Office-Home dataset (right) . As we observe from Fig. 5.4, when the label space drifts, all the algorithms drop off. As we see, our algorithm outperforms the baselines with a large margin when label space shift. *This demonstrates the benefits of our algorithm for handling label shift problems.*

5.5 Discussion and Conclusion

In this chapter, we proposed to leverage the labeling information across different tasks in multi-task learning. 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. Our empirical results demonstrates the effectiveness of our algorithm by improving the benchmark performance with better time efficiency and particularly show the benefits when label distribution shift. In the next chapter, we re-visit the domain generalization problems to investigate the theoretical insights on controlling the semantic conditional distribution divergence and label shift problems.

Chapitre 6

On the value of Label and Semantic Distributions in Domain Generalization

In this chapter, we present our work referred by Zhou et al. (2021d) in which we explored the semantic and label distribution shift in domain generalization. We provide a theoretical guarantee of successful transfer with a concrete algorithm. By doing so, we achieve the state-of-the-art performance and showing better performance to handle with label shift problems.

Résumé

Apprendre des connaissances à partir de plusieurs domaines sources et les généraliser à un domaine inconnu mais différent est un problème important dans l'apprentissage automatique lié au monde réel. Dans ce travail (Zhou et al., 2021d), nous abordons les problèmes de généralisation de domaine (DG) visant à apprendre un prédicteur universel sur plusieurs domaines sources et à le déployer sur un domaine cible non déjà vu. De nombreuses approches de généralisation de domaine ont été principalement motivées par les techniques d'adaptation de domaine, qui n'alignent que la distribution marginale des caractéristiques mais ignorent les relations conditionnelles et les informations des étiquettes dans les domaines sources. Bien que certaines avancées récentes aient commencé à tirer parti des distributions sémantiques conditionnelles, des justifications théoriques manquaient encore. Dans le présent travail, nous étudions la garantie théorique d'un processus de généralisation réussi focalisant sur la manière de contrôler l'erreur du domaine cible. Nos résultats révèlent que pour contrôler le risque de la cible, il faut contrôler conjointement les erreurs de la source, qui sont pondérées en fonction des informations du label, tout en alignant les distributions sémantiques conditionnelles entre les différents domaines source. L'analyse théorique proposée conduit à un algorithme efficace pour contrôler les distributions d'étiquettes ainsi que l'alignement des distributions conditionnelles

sémantiques. Pour vérifier l'efficacité de notre méthode, nous l'évaluons par rapport à des baselines récents sur plusieurs benchmarks. Nos résultats empiriques montrent que la méthode proposée surpasse la plupart des baselines et présente des performances de pointe.

Abstract

Learning knowledge from multiple source domains and generalizing it to an unseen but different domain is an important problem in real-world machine learning. In this work reflected by Zhou et al. (2021d), we tackle the domain generalization (DG) problems aiming to learn a universal predictor on several source domains and deploy it on an unseen target domain. Many existing DG approaches were mainly motivated by the domain adaptation techniques, which only aligned the marginal feature distribution but ignored conditional relations and label information in the source domains. Although some recent advances started to take advantage of conditional semantic distributions, theoretical justifications were still missing. In this work, we investigate the theoretical guarantee for a successful generalization process by focusing on how to control the target domain error. Our results reveal that to control the target risk, one should jointly control the source errors that are weighted according to label information and align the semantic conditional distributions between different source domains. The proposed theoretical analysis then leads to an efficient algorithm to control the label distributions while matching the semantic conditional distributions. To verify the effectiveness of our method, we evaluate it against recent baseline algorithms on several benchmarks. Our empirical results show that our method outperforms most of the baseline methods and shows state-of-the-art performances.

6.1 Introduction

In the previous chapters, we have explored the domain adaptation (DA), domain generalization (DG) and multi-task learning (MTL) problems from different aspects. Specifically, we leverage the label similarities for DG problems in Chapter 4 and investigate the semantic and label shift problems in MTL. Even though we have explored the label similarities in DG in Chapter 4 with a metric learning objective, however, the theoretical understanding on the guarantee of controlling the label and semantic distribution shift is still elusive. To contribute to this theoretical understanding, we re-visit the DG problems, which aims to extract the knowledge from source domains that generalizes well to an unseen target (test) domain. The general learning process of DG has been borrowed in section 1.4.1.

As discussed before, due to the similar problem settings with DA, many DA methodologies, especially the adversarial training (Ganin et al., 2016) based approaches (Li et al., 2017a; Dou et al., 2019), were borrowed for DG. However, these approaches only align the feature distribution $\mathcal{D}(\mathbf{x})$ and rely on the theoretical results under the assumption that the combined

error between the source and target domain is small (Ben-David et al., 2010a), and this is not held in practice.

Zhao et al. (2019a) showed that conditional shift problems can degrade the prediction performance. Besides, if we only align the feature distribution $\mathcal{D}(\mathbf{x})$ while ignoring the conditional semantic $\mathcal{D}(\mathbf{x}|y)$ and labelling $\mathcal{D}(y)$ distribution, the class information for each category among different domains can be lost, which leads to indiscriminative features, *a.k.a.* semantic misalignment problem (Dou et al., 2019; Zhou et al., 2021b). As a consequence, the model may suffer from ambiguous classification boundaries (Dou et al., 2019), which hinders the generalization performance. To address this issue, some recent studies, Dou et al. (2019) and Zhou et al. (2021b) have leveraged the label information to explore the semantic relation for the DG. However, the theoretical justifications for the benefits of semantic alignment remain elusive. Existing theoretical results (Zhao et al., 2020a; Li et al., 2018f) only focused on minimizing the conditional distribution divergences from an optimization perspective, while the analysis for the generalization properties are still missing.

In this chapter, we aim to develop theoretical insights into how to ensure a successful generalization process by investigating the test error on the target domain. Our results reveal the necessity of controlling the semantic conditional distributions as well as the label distribution divergence across all the source domains. The contributions in this chapter are three-fold :

1. We build a theoretical analysis framework to understand the domain generalization process upon bounding the test error on the target domain with total variation distance, which provides a deeper understanding of the role of semantic alignment for general DG problems.
2. Our analysis also reveals the importance of controlling the label distribution divergence for each domain to minimize the generalization error.
3. On the algorithmic side, our theoretical results inspire a novel DG algorithm that jointly minimizes the source errors as well as semantic distribution matching for all the source domains.

Specifically, our method simultaneously matches the semantic distributions via minimizing the centroid statistics across distributions and controlling the label distribution losses. We conduct extensive experiments and the results show that our algorithm outperforms various strong baselines, especially when label shift occurs.

6.2 Preliminaries

We start by some preliminaries with notations and definitions, which we have introduced both in section 1.4.1 and section 4.2. Then we analyze the importance of leveraging the label and semantic distribution. After that, we show the harm of label and semantic distribution shift in domain generalization.

6.2.1 Notations and Definitions

Following the notations in section 1.4.1, let $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ be a training example drawn from some distribution $\mathcal{D}$, where $\mathbf{x}$ is the data point, and y is its label. A hypothesis is a function $h \in \mathcal{H}$ that maps $\mathcal{X}$ to the set $\mathcal{Y}'$ sometimes different from $\mathcal{Y}$, where $\mathcal{H}$ is a hypothesis class. For a non-negative loss function $\ell : \mathcal{Y}' \times \mathcal{Y} \mapsto \mathbb{R}_+$, we denote by $\ell(h(\mathbf{x}), y)$ the loss of hypothesis h at $(\mathbf{x}, y)$. Let $S = \{(\mathbf{x}_j, y_j)\}_{j=1}^N$ be a set of N training examples drawn independently from $\mathcal{D}$. The empirical loss of h on S and its generalization loss over $\mathcal{D}$ are defined, respectively, by $\hat{R}(h) = \frac{1}{N}\sum_{j=1}^N \ell(h(\mathbf{x}_j, y_j))$, and $R(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\ell(h(\mathbf{x}), y)$.

In the context of DG, we are given m source tasks $\{S_i\}_{i=1}^m$, where $S_i = \{(\mathbf{x}_j^{(i)}, y_j^{(i)})\}_{j=1}^{N_i}$ is drawn from a distribution $\mathcal{D}_i$. The objective of a DG algorithm is to learn a feature representation that extracts the knowledge that can be shared across all the known source domains so that it can also generalize well to an unseen target domain distribution $\mathcal{D}_\mathcal{T}$.

6.2.2 Distribution Distance Measure

To measure the marginal and conditional distributions, we need a tool to measure the distribution distances, and finding such a tool is crucial in domain adaptation or generalization methodologies. In this chapter, we continue to adopt the Jensen-Shannon divergence in our analysis. This divergence has been studied in recent literature in transfer learning with adversarial training (Dou et al., 2019; Matsuura and Harada, 2020; Zhao et al., 2019a). Its definition can be referred in section 1.5.1.

6.2.3 The Value of Label and Semantic Information

In the context of DG, a learner can only access the data from the source domains (seen), and no target data is available during the training phase (unseen). As summarized in section 2.3.1, early approaches (e.g. Li et al., 2018b,c; Carlucci et al., 2019) usually only focused on aligning the feature distribution $\mathcal{D}(\mathbf{x})$ while ignoring the labeling $\mathcal{D}(y)$ and semantic $\mathcal{D}(\mathbf{x}|y)$ distributions. Previously, Dou et al. (2019) pointed out that only aligning the feature distribution via distribution matching or adversarial training can lead to the semantic misalignment problems (Zhou et al., 2021b,a). Besides, our work (Zhou et al., 2021b) in Chapter 4 also demonstrates the necessity of leverage the label information in DG. Though some recent works (Dou et al., 2019; Matsuura and Harada, 2020) start to consider the label and semantic relations, their theoretical justifications remain elusive. Our work provides a complete framework to understand the generalization properties of DG which then enables us to design an efficient semantic conditional matching algorithm.

Many of the current DG approaches assumed that the label distribution across all the domains are the same ($D_{\text{JS}}(\mathcal{D}_i(y)\|\mathcal{D}_j(y)) = 0, \forall i, j$). However, this assumption is not necessarily true in practice. A long-neglected issue is the *label shift* problem, which has been explored in the

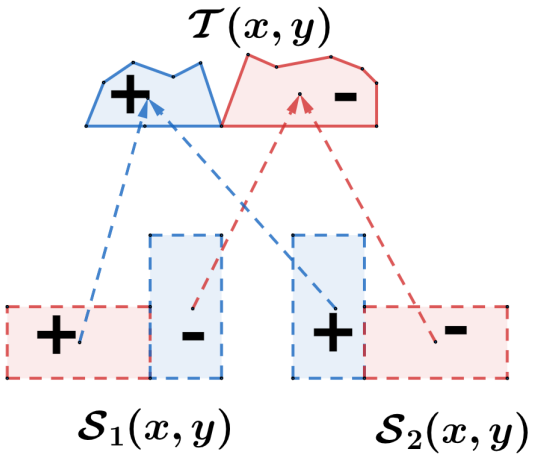


FIGURE 6.1 – A example in semantic shift. The dashed arrow lines indicate the matching process. Let the feature marginal distribution as the color of the instance while the label distribution as positive (+) or negative (-), *i.e.*, $X = \{\text{red}, \text{blue}\}$, $Y = \{+, -\}$. For the source distribution $\mathcal{S}_1$, $\mathbb{P}_{\mathcal{S}_1}(X = \text{red}|Y = +) = 1$ while for the source distribution $\mathcal{S}_2$, $\mathbb{P}_{\mathcal{S}_2}(X = \text{red}|Y = +) = 0$. We can see that if we only design the alignment process with marginal features, the source and target distributions cannot matched correctly.

literature of multi-task learning and domain adaptation (Busto and Gall, 2017; Geng et al., 2020; Azizzadenesheli et al., 2019) but missed in domain generalization. More formally, the label shift between two domains $\mathcal{D}_i$ and $\mathcal{D}_j$ indicates $D_{\text{JS}}(\mathcal{D}_i(y)\|\mathcal{D}_j(y)) \neq 0$ (Redko et al., 2019; Zhou et al., 2021a).

We present an example to show the necessity of controlling semantic divergence in Fig. 6.1. Suppose we have two source distributions $\mathcal{S}_1(\mathbf{x}, y)$ and $\mathcal{S}_2(\mathbf{x}, y)$, and hope to match to the target distribution $\mathcal{T}(\mathbf{x}, y)$. The feature marginal distribution is represented by the color of the region while the label distribution is indicated by positive (+) or negative (-), *i.e.*, $X = \{\text{red}, \text{blue}\}$, $Y = \{+, -\}$. For the source distribution $\mathcal{S}_1$, $\mathbb{P}_{\mathcal{S}_1}(X = \text{red}|Y = +) = 1$ while for the source distribution $\mathcal{S}_2$, $\mathbb{P}_{\mathcal{S}_2}(X = \text{red}|Y = +) = 0$. In this case, if we only use the general adversarial training or MMD based approaches to align the marginal distribution (see the examples summarized in Table 2.2 in section 2.3.1), it will be difficult to fix the semantic shift problem. We should also consider matching the semantic distributions for each domain. Another practical example can be the multi-source generalization problems on the digits problems. Let MNIST, which is a grey-scaled digits dataset, be $\mathcal{D}_i$, and let SVHN dataset, which consists of colorful images of street numbers, be $\mathcal{D}_j$. If we consider a specific class $Y = y_k$, we can easily see that $\mathcal{D}_i(\mathbf{x}|y) \neq \mathcal{D}_j(\mathbf{x}|y)$ since the color and digits styles are obviously different from each other.

On the other hand, the label shift problem may also hurt the generalization performance. For

example, for a health diagnostic learning task using DG (Liu et al., 2021), when collecting data from different hospitals, the labels may vary from each other between different datasets. Since the ultimate goal of DG is to align $\mathcal{D}(\mathbf{x}, y) = \mathcal{D}(\mathbf{x}|y)\mathcal{D}(y)$ between domains, if $\mathcal{D}(y)$ changes, even if we can match $\mathcal{D}(\mathbf{x}|y)$ properly for all the domains, the prediction of the classifier can still diverge since the label distribution is not necessarily aligned during either the supervised classification process or the semantic matching process.

All these examples indicate that we need to consider both the label and semantic distribution alignments when designing DG algorithms. In the next section, we develop the theoretical justifications for controlling the conditional semantic and label distributions. Moreover, our results also lead to an efficient algorithm for DG problems.

6.3 Theoretical Analysis and Methodology

6.3.1 Theoretical Analysis

As introduced in section 1.4.1, one fundamental assumption of DG is that all the domains are not far from each other in terms of distribution distances. More formally, among all the source domains, let $\mathcal{D}^\star$ be the nearest one to the target domain, *i.e.*, $\epsilon^\star \triangleq d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T}) \leq d_{TV}(\mathcal{D}_i, \mathcal{D}_\mathcal{T}), \forall i$, where $d_{TV}(\cdot)$ is the total variation distance (see section 1.5.1). Then, it is reasonable to assume that $\epsilon^\star$ is small for DG problems since if the distance between the source and target is arbitrarily large, the learner will fail to generalize to the target domain. We can also assume that $\mathcal{D}^\star$ and $\mathcal{D}_\mathcal{T}$ should satisfy a semantic conditional distance, *i.e.*, $d_{TV}(\mathcal{D}^\star(\mathbf{x}|y), \mathcal{D}_\mathcal{T}(\mathbf{x}|y)) \leq \kappa^\star$ where $\kappa^\star$ is a constant that is not arbitrarily large. We show a generalization process in Fig. 6.2 where we have several source domains, and the target domain is unseen but assumed not to be far away from the source domains. Then, we can bound the learning risk on the target domain $R_{\mathcal{D}_\mathcal{T}}(h)$ as shown in Theorem 6.1. The proof of the theoretical results in this chapter is presented in Appendix B.3.

Theorem 6.1. *Suppose we have m source domains $\mathcal{D}_1, \dots \mathcal{D}_m$, and $\mathcal{D}^\star$ is the nearest source domain to the target $\mathcal{D}_\mathcal{T}$, and $\epsilon^\star \triangleq d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T})$.*

Then the target domain risk is bounded by,

$$R_{\mathcal{D}_\mathcal{T}}(h) \leq \frac{1}{m}\sum_{i=1}^{m} R_{\mathcal{D}_i}(h) + \epsilon^\star + \frac{1}{m}\sum_{i} d_{TV}(\mathcal{D}^\star(\mathbf{x}, y), \mathcal{D}_i(\mathbf{x}, y)) \tag{6.1}$$

Remark : The first term in Eq. 6.1 is the averaged source error which can be approximated by the empirical risk minimization. The second term is a small constant. The third term can also not be estimated directly since we don't know which source domain is the nearest one to the target. However, this term can be minimized by pair-wised distribution matching between all source domains.

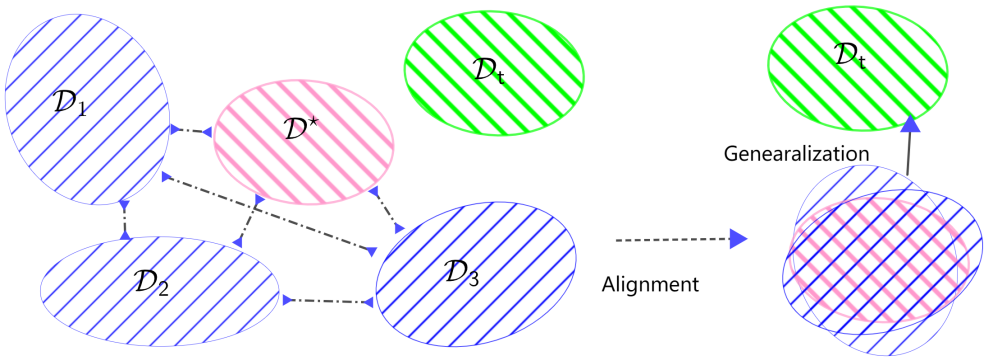


FIGURE 6.2 – The domain generalization process where there are several source domains and a target domain $\mathcal{D}_\mathcal{T}$. In case we have limited number of source domains, there exists a source domain $\mathcal{D}^\star$ that is the nearest to the target domain. At the training phase, we implement the alignment process for all the source domains to learn the transferable features to generalize to the target domain.

Theorem 6.1 bounds the target generalization error in terms of the joint distributions between source domains. To motivate a more concrete DG algorithm that leverages the label ($\mathcal{D}(y)$) and semantic conditional ($\mathcal{D}(\mathbf{x}|y)$) information, we derive the following Corollary.

Corollary 6.1. *Following the assumptions of Theorem 6.1, then the target domain risk could be bounded by,*

$$
\begin{aligned}
R_{\mathcal{D}_\mathcal{T}}(h) \leq \frac{1}{m}\sum_{i=1}^{m} R_{\mathcal{D}_i}(h) + \epsilon^\star \\
+ \frac{1}{m}\sum_{i} [\underbrace{\sqrt{D_{JS}(\mathcal{D}^\star(y)||\mathcal{D}_i(y))}}_{\mathbf{I}} \\
+ \underbrace{\sqrt{\mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{JS}(\mathcal{D}^\star(\mathbf{x}|y)||\mathcal{D}_i(\mathbf{x}|y))}}_{\mathbf{II}} \\
+ \underbrace{\sqrt{\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{JS}(\mathcal{D}^\star(\mathbf{x}|y)||\mathcal{D}_i(\mathbf{x}|y))}}_{\mathbf{III}}]
\end{aligned}
\tag{6.2}
$$

In order to minimize Eq. 6.2, except for minimizing the source domain risks $\frac{1}{m}\sum_{i=1}^{m} R_{\mathcal{D}_i}(h)$ we need to consider the last three terms **I** : J-S distance between the label distribution $\mathcal{D}^\star(y)$ and $\mathcal{D}_i(y)$, as well as **II** and **III**, which are the J-S distance between the semantic distributions.

For **I** in Eq. 6.2, we could adopt a reweighted loss $\hat{\mathcal{L}}^{\boldsymbol{\alpha}}_{\mathcal{D}_i}$ (will be introduced in Eq. 6.6) to balance the label distribution for each pair of source domains so that $D_{JS}(\mathcal{D}_i(y)||\mathcal{D}_j(y)) = 0$ for all the domain pairs i, j. In this case, term **II** and **III** will be identical to each other and we can bound the generalization risk on the target domain as follows.

Corollary 6.2. *Following the assumptions of Theorem 6.1 and assume the semantic distribution between the nearest source domain to the target domain is a constant,* i.e., $d_{TV}(\mathcal{D}^\star(\mathbf{x}|Y =$

$k), \mathcal{D}_{\mathcal{T}}(\mathbf{x}|Y=k) \leq \kappa^{\star}$. *Let* $\hat{\mathcal{L}}^{\boldsymbol{\alpha}}_{\mathcal{D}_i}(h)$ *be the reweighted loss and the prediction loss function is bounded by* $[0,1]$*, then the target domain risk could be bounded by,*

$$
\begin{aligned}
R_{\mathcal{D}_{\mathcal{T}}}(h) \leq & \underbrace{\frac{1}{m}\sum_{i=1}^{m}\mathbb{E}_{\mathbf{x}\sim\mathcal{D}_i}\hat{\mathcal{L}}^{\boldsymbol{\alpha}}_{\mathcal{D}_i}(h)}_{\textit{Re-weighted source risks}} + \underbrace{\kappa^{\star}}_{\textit{Constant}} \\
& + \frac{1}{K}\sum_{k=1}^{K}\Big[\underbrace{\frac{1}{m}\sum_{i=1}^{m} d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k), \mathcal{D}_i(\mathbf{x}|Y=k))}_{\textit{Achieved by pair-wise semantic matching}}\Big]
\end{aligned}
\tag{6.3}
$$

Remark : The first term is the balanced source errors that can help to handle the label distributions shift. The second term is a small constant. The third term could be minimized by a pair-wised semantic matching scheme, and we will elaborate on it in the next section.

Now, we show that by aligning the semantic conditional distributions $\mathcal{D}(\mathbf{x}|y)$, we could also align the marginal distributions $\mathcal{D}(\mathbf{x})$. We notice that, for a pair of source domain distributions $\mathcal{D}_i$ and $\mathcal{D}_j$

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{x}}|\mathcal{D}_i(\mathbf{x}) - \mathcal{D}_j(\mathbf{x})| = \mathbb{E}_{\mathbf{x}}|\mathcal{D}_i(y)\mathcal{D}_i(\mathbf{x}|y) - \mathcal{D}_j(y)\mathcal{D}_j(\mathbf{x}|y)| \\
&= \sum_{k=1}^{K}|\mathcal{D}_i(Y=k)\mathcal{D}_i(\mathbf{x}|Y=k) - \mathcal{D}_j(Y=k)\mathcal{D}_j(\mathbf{x}|Y=k)| \\
&= \frac{1}{K}\mathbb{E}_x|\sum_{y}(\mathcal{D}_i(\mathbf{x}|y) - \mathcal{D}_j(\mathbf{x}|y))| \\
&\leq \frac{1}{K}\sum_{y}\mathbb{E}_{\mathbf{x}}|\mathcal{D}_i(\mathbf{x}|y) - \mathcal{D}_j(\mathbf{x}|y)| \\
&= \frac{2}{K}\sum_{y} d_{\mathrm{TV}}(\mathcal{D}_i(\mathbf{x}|y), \mathcal{D}_j(\mathbf{x}|y))
\end{aligned}
\tag{6.4}
$$

Eq. 6.4 shows that by minimizing the total variation distance between the two semantic conditional distributions $\mathcal{D}_i(\mathbf{x}|y)$ and $\mathcal{D}_j(\mathbf{x}|y)$, we could also take care of the marginal distribution of these two domains $\mathcal{D}_i(\mathbf{x})$ and $\mathcal{D}_j(\mathbf{x})$. That is, *when matching the semantic conditional distributions, we could also align the marginal features simultaneously.*

Now, based on the analysis above, we could summarize that to minimize the target risk, we need to follow the two following principles :

— Minimizing the weighted source risks (will be introduced in Eq. 6.6).

— Matching the semantic divergences between each source domains (will be introduced in Eq. 6.12).

These two principles are to come, before presenting these two principles, we can first introduce our methodology in the next section.

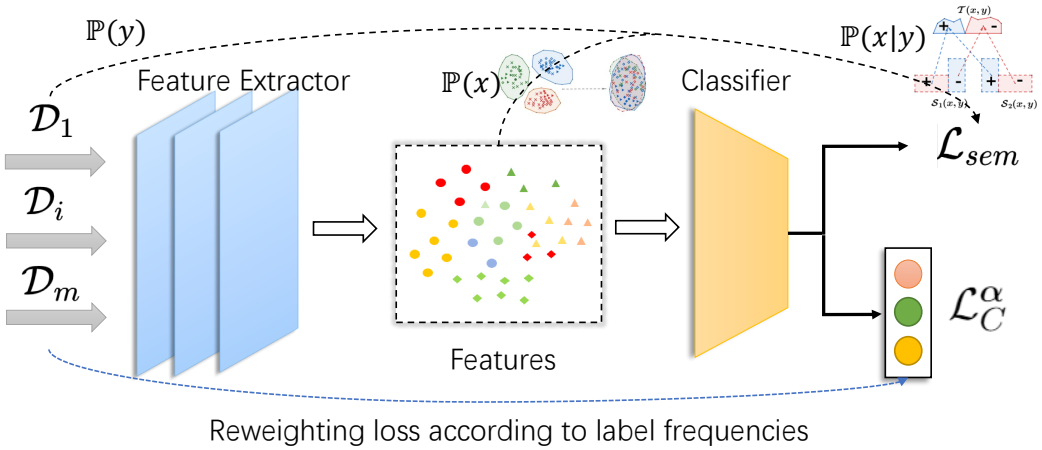


FIGURE 6.3 – The overall model architecture. The feature extractor is trained to find out the shared features, and the extracted features are also used for computing the centroids of the semantic distributions to compute the semantic objective. The classifier is trained using the source domain data and is committed to performing well on the unseen target domain. For all the domains, the model parameters are shared with each other and trainable on all the source domains.

6.3.2 Methodology

The overview of our model

The model architecture of our DG algorithm is presented in Fig. 6.3. It consists of two parts : feature extractor and classifier. The feature extractor function F, parameterized by $\boldsymbol{\theta}_f$, is trained to extract both feature and semantic information that is shared across the sources domains. Once the domains are aligned properly, the classifier function C, parameterized by $\boldsymbol{\theta}_c$, is trained to make universal predictions for all the domains. When training the model, the model predicts $C(F(\mathbf{x};\boldsymbol{\theta}_f);\boldsymbol{\theta}_c) \rightarrow y$. For simplicity, we omit the model parameters when displaying the model functions. For classification, we adopt the cross-entropy loss,

$$\ell = -\sum_{i=1}^{m}\sum_{j=1}^{N_i} y_j^{(i)} \log(\mathbb{P}(C(F(\mathbf{x}_j^{(i)})))) \tag{6.5}$$

As analyzed before, to minimize the risk of the prediction on the target domain, we should both control the semantic conditional distance and the label distribution divergence. In case of the label distributions differ from each other, some minor classes may be regarded as noise, and the minor classes will be neglected (Zhou et al., 2021a). In order to alleviate the impacts of the source domains' label space shifts, similar with the reweighting scheme introduced in section 5.3.1, we can re-weigh the importance of each class to correct the loss based the total instance number in that category (Lipton et al., 2018),

$$\hat{\mathcal{L}}^{\boldsymbol{\alpha}}_{\mathcal{D}_i}(h) = \sum_{(\mathbf{x}_i, y_i) \in \hat{\mathcal{D}}_i} \alpha(y_i) \ell(h(\mathbf{x}_i), y_i)) \tag{6.6}$$

where $\boldsymbol{\alpha} = [\alpha_1, \dots \alpha_k, \dots, \alpha_K]^T$ is the weighting vector for all K classes in each domain. For a certain class k, suppose we have N_k instances in that category, we could compute the weight by,

$$\alpha_k = \sum \frac{|\mathbb{1}[y = y_k]|}{N_k} \tag{6.7}$$

Through Eq. 6.7, the cross-entropy loss is reweighted via the frequency of the number of instances from a specific class, and this ensures the data from different classes among all the domains to have the same probability to be sampled during training (see also in Fig. 5.1 for an example). By this process, the learner will be guided to pay attention to the classes with few instances, and this in general help to handle the label distribution drift. Then, the classification objective could be computed as

$$\mathcal{L}^{\boldsymbol{\alpha}}_{C} = \sum_{i=1}^{m} \hat{\mathcal{L}}^{\boldsymbol{\alpha}}_{\mathcal{D}_i} = \sum_{i=1}^{m} \sum_{(\mathbf{x}_i, y_i) \in \hat{\mathcal{D}}_i} \alpha(y_i) \ell(h(\mathbf{x}_i), y_i)) \tag{6.8}$$

Except for the reweighted loss, we also need to guide the learner to leverage the semantic distributions $\mathcal{D}(\mathbf{x}|y)$ across the domains. To this end, we adopt the extracted features z_i from domain i, to condition on the semantic distributions $\mathcal{D}(z|y)$.

To align the semantic conditional distributions, *i.e.*, minimizing $D_{JS}(\mathcal{D}_i(\mathbf{x}|y) \| \mathcal{D}_j(\mathbf{x}|y))$ for all domains pairs i, j, several solutions (e.g. conditional GAN training, moment matching, etc.) are possible. We adopted an alternative yet popular approach : class-level feature mean matching method, which is prevalent in the general machine learning literature (Dou et al., 2019; Chopra et al., 2005; Xie et al., 2018).

Notice that the semantic minimization objective is computed across all the source domains. We can take out the extracted features and compute the corresponding semantic centroids. And we do that similarly with Dou et al. (2019), we condition the extracted features on each class k to measure the semantic conditional distributions for instances from source domains $S_i = \{(\mathbf{x}_j, y_j)\}_{j=1}^{N_i}$ from all the categories $k \in \{1 \dots K\}$. Then, the empirical semantic centroid is estimated by

$$\begin{aligned} \hat{z}^k_{c_i} &= \frac{1}{|\mathcal{D}^k_i|} \sum_{\mathbf{x}_i \in \mathcal{D}^k_i} z^k_i = \frac{1}{|\mathcal{D}^k_i|} \sum_{\mathbf{x}_i \in \mathcal{D}^k_i} F(\mathbf{x}_j) \\ &\approx \mathbb{E}_{\mathcal{D}_i}[F(\mathbf{x}_i)|Y = k] \end{aligned} \tag{6.9}$$

Through this process, we compute the feature centroids. We then follow the strategy of (Xie et al., 2018; Zhou et al., 2021a) to maintain a global matrix $\mathcal{Z}_{\mathcal{D}_i}$ for each source domain to

Algorithm 5 The Semantic Matching Domain Generalization (SMDG) algorithm

Input: Samples from different source domains $\{\mathcal{D}_i\}_{i=1}^m$;
Output: Neural network parameters $\boldsymbol{\theta}_f$, $\boldsymbol{\theta}_c$

1: **for** mini-batch of samples $\{(\mathbf{x}_s^{(i)}, y_s^{(i)})\}$ from source domains **do**
2: Compute the classification loss $\mathcal{L}_C^{\boldsymbol{\alpha}}$ over all the domains according to Eq. 6.8 ;
3: Mix the instances and compute the semantic matching objective $\mathcal{L}_{Sem}$ via Eq. 6.12 ;
4: Update θ^f, θ^c by solving Eq. 6.13 with learning rate η :

$$\boldsymbol{\theta}_f \leftarrow \boldsymbol{\theta}_f - \eta \frac{\partial(\mathcal{L}_C^{\boldsymbol{\alpha}} + \lambda_s \mathcal{L}_{Sem})}{\partial \boldsymbol{\theta}_f},$$
$$\boldsymbol{\theta}_c \leftarrow \boldsymbol{\theta}_c - \eta \frac{\partial(\mathcal{L}_C^{\boldsymbol{\alpha}} + \lambda_s \mathcal{L}_{Sem})}{\partial \boldsymbol{\theta}_c};$$

5: **end for**
6: Return the optimal model parameters $\boldsymbol{\theta}_f$ and $\boldsymbol{\theta}_c$.

maintain the semantic centroids

$$\mathcal{Z}_{\mathcal{D}_i}^k \leftarrow \gamma \hat{z}_{c_i}^k + (1-\gamma)\hat{z}_{c_i}^k \tag{6.10}$$

Eq.6.10 defines a moving averaging method for the batch training of $\mathcal{Z}$, where γ is a coefficient to control the moving average process. Then, we could maintain a matrix $\mathcal{Z}_i = [\mathcal{Z}_{\mathcal{D}_i}^1, \ldots, \mathcal{Z}_{\mathcal{D}_i}^K]^T$ to trace the semantic relations between domains, through which we could match the semantic distributions via minimize the Euclidean distance $\Phi(\mathcal{Z}_{\mathcal{D}_i}^k, \mathcal{Z}_{\mathcal{D}_j}^k)$ between two centroids in the embedding space, which is computed as

$$\Phi(\mathcal{Z}_{\mathcal{D}_i}^k, \mathcal{Z}_{\mathcal{D}_j}^k) = ||\mathcal{Z}_{\mathcal{D}_i}^k - \mathcal{Z}_{\mathcal{D}_j}^k||^2 \tag{6.11}$$

Here the function $\Phi(\mathcal{Z}_{\mathcal{D}_i}^k, \mathcal{Z}_{\mathcal{D}_j}^k)$ is the approximation of the total variation $D_{TV}(\mathcal{Z}_{\mathcal{D}_i}^k, \mathcal{Z}_{\mathcal{D}_j}^k)$, which is the upper bound of $D_{JS}(\mathcal{Z}_{\mathcal{D}_i}^k || \mathcal{Z}_{\mathcal{D}_j}^k)$ (Lin, 1991). Then for each training epoch, the semantic loss $\mathcal{L}_{Sem}$ is updated by,

$$\mathcal{L}_{Sem} \leftarrow \mathcal{L}_{Sem} + \Phi(\mathcal{Z}_{\mathcal{D}_i}, \mathcal{Z}_{\mathcal{D}_j}) \tag{6.12}$$

By minimizing the semantic objectives of all the domains, we could achieve semantic invariant features.

Now, with the components described above, we could summarize the learning objective of our method as

$$\mathcal{L} = \mathcal{L}_C^{\boldsymbol{\alpha}} + \lambda_s \mathcal{L}_{Sem} \tag{6.13}$$

where $\mathcal{L}_C^{\boldsymbol{\alpha}}$ is the modified classification objective defined in Eq. 6.8, $\mathcal{L}_{Sem}$ is the semantic learning objective defined in Eq. 6.12 and λ_s is a coefficient to regularize the semantic learning

objective. We show the whole learning process in Algorithm 5. The algorithm mainly consists several parts : first to measure the label distributions and compute the reweighted classification objective to enforce the class-level alignment, second to enforce the domain-level semantic alignment for all the domains. We then evaluate the effectiveness of our method in the next part.

6.4 Experiments and Results

We verify the effectiveness of our approach on several benchmarks, including the PACS, VLCS and Office-home dataset, comparing with several baselines. We first evaluate the results comparing with baselines showing the state-of-the-art performance on benchmarks. To further understand the method, we then conduct the ablation studies, evaluations under label distributions shift as well as time efficiency evaluations to confirm the effectiveness of our method.

6.4.1 Datasets and Preparation

We compare our method with some baseline methods on VLCS, PACS and Office-home dataset. The VLCS dataset (Torralba and Efros, 2011) consists of four domains of images from *LabelMe*(L), *PASCAL-VOC2007*(V), *SUN-09*(S) and *Caltech-101*(C) with total five categories in each domain. Unlike some previous work (Li et al., 2018c; Dou et al., 2019), which adopt the *DeCAF* model (Donahue et al., 2014) features (DeCAF6 features), we use the original dataset with images so that the model can explore the semantic features. The PACS dataset (Li et al., 2017a) is a recent standard benckmark for DG which consists of images from four domains : *Art* (A), *Cartoon* (C), *Photo* (P) and *Sketch* (S). Office-Home (Venkateswara et al., 2017) is a more challenging dataset, which has been widely investigated in recent DA and DG researches (Wen et al., 2019b; Dou et al., 2019; Zhou et al., 2021c,b). This dataset contains images from four different domains : *Art* (Ar), *Clipart* (Cl), *Product* (Pr) and *Real World* (Rw). Images from all the domains have 65 categories. Previous work on supervised multi-domain learning (Long et al., 2017a), semi-supervised learning (Zhou et al., 2021c) or unsupervised learning (Long et al., 2018) have shown that the model can suffer from the diverse features present in the different domains. Evaluating the algorithms on those benchmarks generally affirm the effectiveness of our method.

6.4.2 Baselines and Implementation Details

We tested our algorithm on the benchmark datasets with the following principled domain generalization approaches. Specifically, we considered several principled approaches : 1) matching-based approaches, 2) meta-learning-based approaches and 3) conditional alignment approaches. Specially, we compared the following baselines on the benchmarks :

Method	Art	Cartoon	Sketch	Photo	Avg.
Deep All	63.30	63.13	54.07	87.70	67.05
CDANN	62.70	69.73	64.45	78.65	68.88
MLDG	66.23	66.88	58.96	88.00	70.01
D-SAM	63.87	70.70	64.66	85.55	71.20
JiGen	67.63	71.71	65.18	89.00	73.38
MMLD	**69.27**	**72.83**	66.44	88.98	74.38
Ours	67.87	72.14	**70.16**	90.45	**75.16**

TABLE 6.1 – Empirical results (accuracy %) on each target domain on PACS dataset.

— ***Deep All*** : This model trains the model on source domains only. We implement the pre-trained AlexNet or ResNet-18 as the feature extractor and aggregate the classification loss of all source domains as the learning objective ;

— ***CDANN*** (Li et al., 2018f) : This studies conditional alignment method by extracting the conditional-invariant feature and varying the class prior so that the conditional distributions among domains could be matched ;

— ***MLDG*** (Li et al., 2018b) : MLDG is a meta-learning based domain generalization method. It simulates the domain shift by split the source data into *meta-train* and *meta-test* datasets to learn the invariant features for generalization ;

— ***D-SAM*** (D'Innocente and Caputo, 2018) : It is a method that aggregates several domain-specific modules, which allows the model to merge general and specific information from all the domains to generalize to a new domain ;

— ***MMD-AAE*** (Li et al., 2018c) : This method is a Mean-Max Discrepancy (MMD) approach mapping the latent features to kernel space for the MMD minimization. The method is combined with the Adversarial AutoEncoder (AAE) model with shallow layers. Later in this chapter, we adopt this MMD mapping with a deep model while relaxing the reconstruction objective ;

— ***MixUp*** (Yan et al., 2020) : This method leverages the feature level consistency to facilitate the inter-domain regularization.

— ***JiGen*** (Carlucci et al., 2019) : This one leverages the Jigsaw puzzle under an unsupervised task to achieve domain invariant features for generalization.

— ***MASF*** (Dou et al., 2019) : MASF is also a meta-learning-based approach that combines the MLDG with the constrictive loss and triplet loss to encourage the class-level alignment.

— ***WADG*** (Zhou et al., 2021b) : This one is our work (Zhou et al., 2021b) introduced in Chapter 4. It combines the Wasserstein adversarial training with a metric similarity learning objective to achieve both the domain-level and class-level alignment.

— ***MMLD*** (Matsuura and Harada, 2020) : MMLD is an approach mixing all the source features together with an unsupervised objective to extract domain-independent feature space.

Method	Caltech	LabelMe	Pascal	Sun	Avg.
Deep All	92.86	63.10	68.67	64.11	72.19
D-MATE	89.05	60.13	63.90	61.33	68.60
CDANN	88.83	63.06	64.38	62.10	69.59
TF	93.63	**63.49**	69.99	61.32	72.11
MMD-AAE	94.40	62.60	67.70	64.40	72.28
D-SAM	91.75	56.95	58.95	60.84	67.03
MLDG	94.4	61.3	67.7	65.9	73.30
JiGen	96.93	60.90	70.62	64.30	73.19
MMLD	96.66	58.77	**71.96**	**68.13**	73.88
Ours	**97.54**	63.41	69.36	65.63	**73.98**

TABLE 6.2 – Empirical results (accuracy %) on VLCS dataset with pre-trained AlexNet as feature extractor.

— ***DGER*** (Zhao et al., 2020a) : DGER is an approach focusing on minimizing the prediction entropy.

We first adopt the pre-trained AlexNet model as the feature extractor to evaluate the algorithms on the PACS and VLCS datasets. The model is trained with Adam optimizer with a learning rate 2×10^{-4} for a total 180 epochs. The results on PACS and VLCS benchmarks with AlexNet are represented in Table. 6.1 and Table. 6.2, respectively. We refer to the results of the baseline using the original value reported in their manuscripts. From the results, we could see that our method could outperform the baselines on these two benchmarks by achieving state-of-the-art performance. Besides, we also observe that on the VLCS dataset (see Table 6.2) that our method has a small improvement comparing with the baseline MMLD (Matsuura and Harada, 2020), which is a recent method that exploit the semantic alignment using an unsupervised learning objective. Since the number of images in VLCS datasets is relatively small and the features are somehow simple, the improvement may become small. However, when it comes to the PACS dataset (see Table 6.1 and Table 6.4), we observe that our method can have better improvements than the baselines.

We then follow the evaluation protocols of Zhou et al. (2021b); Dou et al. (2019); Matsuura and Harada (2020) to implement the experiments on PACS dataset and Office-home with deeper backbones as the feature extractor to show the benefits of our method. We adopted the pre-trained ResNet-18 model as the feature extractor and trained the model with mini-batch size 64 and test batch size 16. The model is optimized with Adam optimizer with a learning rate ranging from 2×10^{-4} to 5×10^{-5} on PACS and VLCS dataset while 3×10^{-3} on the Office-home dataset.

The test results on PACS and Office-Home benchmarks with ResNet-18 feature extractor are reported in table 6.4 and Table 6.3, respectively. From the test result, we could observe

	Art	Clipart	Product	Real-World	Avg.
Deep All	52.15	45.86	70.86	73.15	60.51
D-SAM	58.03	44.37	69.22	71.45	60.77
JiGen	53.04	**47.51**	71.47	72.79	61.20
WADG	55.34	44.82	72.03	73.55	61.44
Ours	**58.76**	45.49	**72.46**	**75.21**	**62.98**

TABLE 6.3 – Empirical results (accuracy %) on Office-home dataset with pre-trained ResNet-18 as feature extractor

Method	Art	Cartoon	Sketch	Photo	Avg.
Deep All	77.87	75.89	69.27	95.19	79.55
D-SAM	77.33	72.43	77.83	95.30	80.72
JiGen	79.42	75.25	71.35	96.03	80.51
MASF	80.29	77.17	71.69	94.99	81.04
MMLD	81.28	77.16	72.29	96.09	81.83
DGER	80.70	76.40	7.177	**96.65**	81.38
WADG	**81.56**	78.02	78.42	95.82	83.45
Ours	81.10	**79.66**	**78.92**	95.87	**83.89**

TABLE 6.4 – Empirical results (accuracy %) on PACS dataset with pre-trained ResNet-18 as feature extractor.

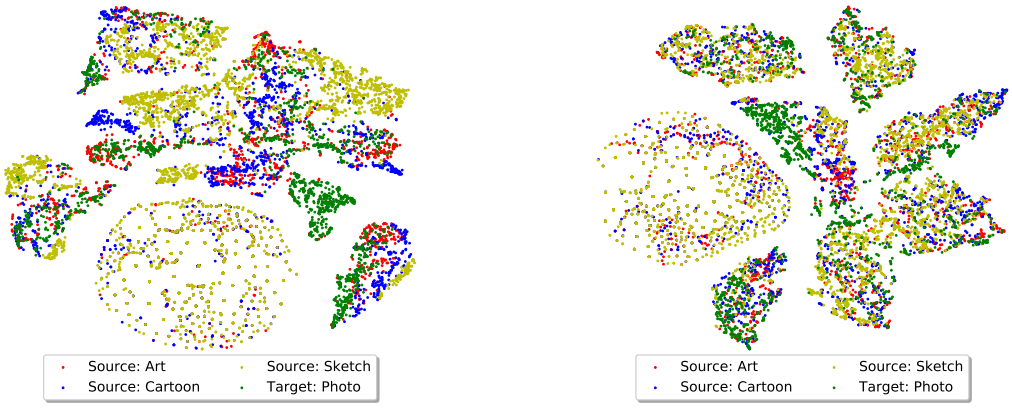
Benchmark	PACS-AlexNet					PACS-ResNet18					Office-Home				
Ablation	A	P	C	S	Avg.	A	P	C	S	Avg.	Ar	Cl	Pr	Rw	Avg.
Deep-All	63.30	87.7	63.13	54.07	67.05	77.87	95.19	75.89	69.27	79.55	52.15	45.86	70.86	73.15	60.51
No-sem.	64.40	87.37	67.55	65.36	71.71	80.08	94.68	79.26	76.75	82.69	57.37	43.37	71.51	73.93	61.54
No re-weight	64.55	86.55	68.33	68.70	72.03	79.17	94.91	78.85	76.71	82.41	58.35	45.06	72.21	75.05	62.67
Full	67.87	90.45	72.14	70.16	75.16	81.10	79.66	78.92	95.87	83.89	72.46	58.76	45.49	75.21	62.98

TABLE 6.5 – The ablation studies on PACS and Office-Home datasets.

an obvious improvement on the benchmarks performances achieving the state-of-the-art performances. Furthermore, here we would like to note that, compared with the methods based on the metric learning objectives (e.g. Dou et al., 2019), our method doesn't require a large batch size for the triplet property to achieve better performances. For example, on the Office-Home benchmark, to ensure the triplet property, one needs a batch size of at least 195. When we adopt some deeper backbones (*e.g.*, ResNet-50 or even more deeper) as feature extractors, the computational cost will be prohibitive. This also confirms the effectiveness of our method.

6.4.3 Further Analysis

Except for the standard benchmark evaluations, we then further investigate our method in several aspects, including the t-SNE visualizations, ablation studies, performance under label



(a) Deep All (b) Our method

FIGURE 6.4 – t-SNE visualizations of our method on PACS dataset

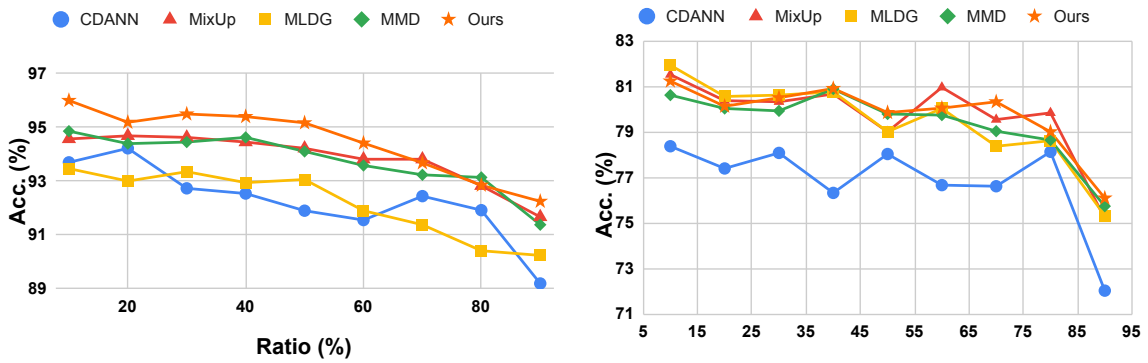
shift and time efficiencies.

t-SNE Visualization We first show the t-SNE visualization of our method to show the alignment performance on the PACS dataset comparing the source-only training only and full method. The results on PACS is illustrated on Fig. 6.4. The results show that our method could well align the features, which confirms the effectiveness of our method on domain level alignment.

Ablation studies We also conducted ablation studies on each part of our algorithm. For that, we implemented the following :

1. *Cls. only* : we only train the model on the source domains using the classification objectives without the re-weighting technique ;
2. *No Sem.* We omit the semantic alignment objective while keeping the classification objective with the re-weighting technique ;
3. *No Re-weighting* : We omit the re-weighting technique in the classification objective while keeping the semantic matching and original cross-entropy classification objective.

In order to better evaluate the effectiveness of our method with depth understanding, we implemented the ablations on PACS dataset using AlexNet and als ResNet-18 as feature extractors. We also conduct ablation studies on Office-Home dataset with ResNet-18 as the feature extractor. The results of ablation studies are presented in Table 6.5. As we could observe from the ablation results, semantic domain alignment is crucial to our method. If we omit the semantic alignment objective, there could be a rapid drop-off in the performance. Besides, the label correction objective also helps to improve the performance compared with the original cross-entropy learning objective.



(a) Target : Photo

(b) Target : Art

(c) Target : Cartoon

(d) Target : Sketch

FIGURE 6.5 – Performance comparison under label shift situation on PACS dataset with respect to the four target domains.

Performance under label distribution shift As our theoretical analysis presented in section 6.3.1 demonstrates the necessity of controlling the label shift. Our algorithm is inspired from this analysis and is committed to handling label distribution shift. To confirm the effectiveness of overcoming label shift problems, we conducted the experiments to check the DG algorithms' performance under label shift scenarios where the label distributions from all the domains drift from each other, *i.e.*, we randomly remove a certain percentage of instances from each domain. We implement the label drift process on PACS and Office-Home datasets. Our method is compared with the following four principled methods :

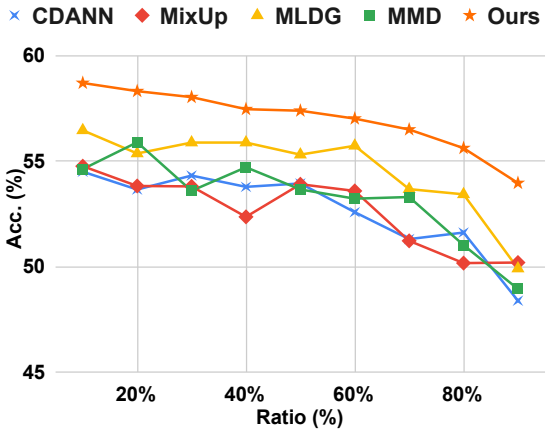
— The conditional alignment method, namely the CDANN method (Li et al., 2018f) ;

— The meta learning-based method, namely the MLDG method (Li et al., 2018b) ;

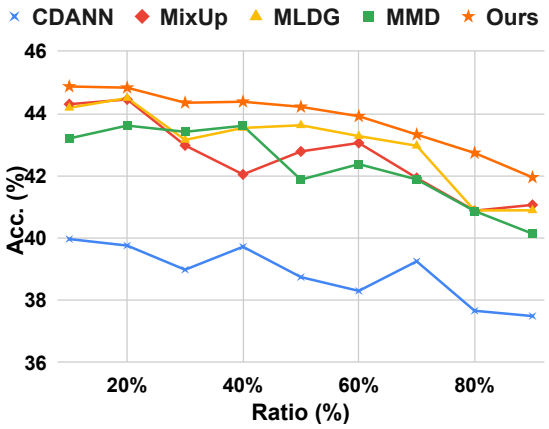
— The Mean-Max Discrepancy (MMD) minimization based method (Li et al., 2017a) ;

— The MixUp method (Yan et al., 2020).

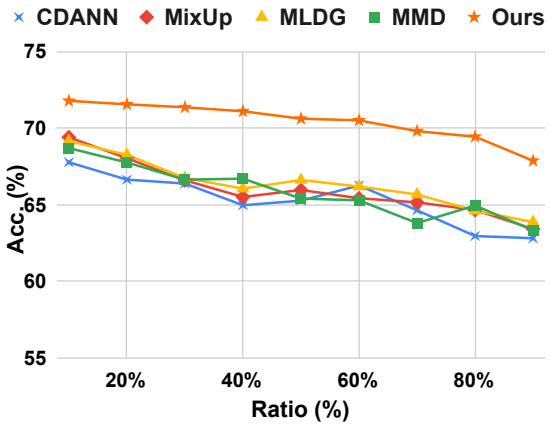
For the label shift simulations on PACS dataset, we removed a certain ratio ($10\% \sim 90\%$) of instances from 2 classes from each source domain. For the Office-Home dataset, for each source domain, we remove a certain ratio ($10\% \sim 90\%$) of instances randomly from 15 categories.



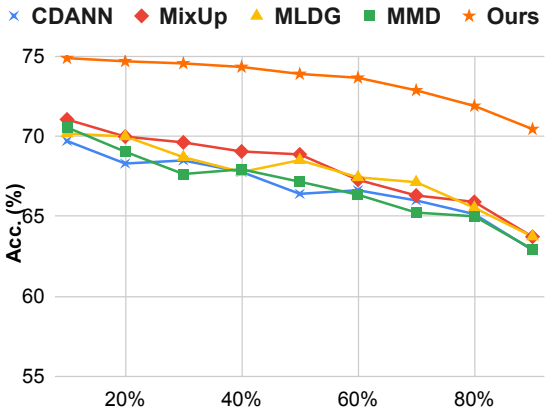
(a) Target : Ar.



(b) Target : Cl.

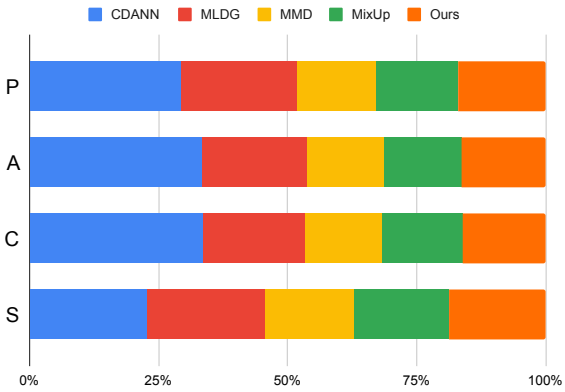


(c) Target : Pr.

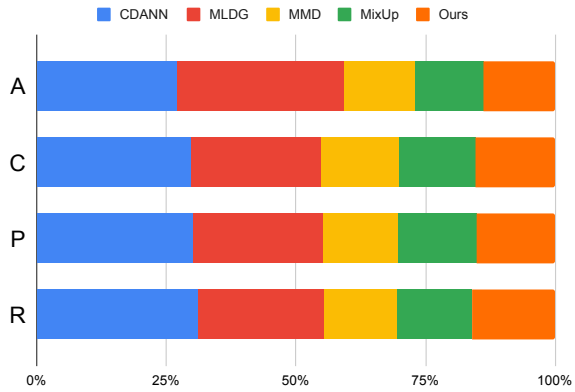


(d) Target : Rw.

FIGURE 6.6 – Performance comparison under label shift situation on Office-Home dataset with respect to the four target domains for each generalization task.



(a) PACS dataset



(b) Office-Home dataset

FIGURE 6.7 – Relative time comparison on PACS and Office-Home dataset.

The compared results curves on PACS and Office-Home datasets with different target domains are illustrated in Fig. 6.5 and Fig. 6.6, respectively.

From the results in Fig. 6.5 and Fig. 6.6, we observe that our method outperforms the baselines under all the shift ratios. Specifically, on the PACS dataset, we see that the MLDG method and MixUp method have a similar performance comparing with ours under certain shift ratios when choosing *Art* and *Sketch* as the target domain. However, on the Office-Home dataset, our method has obvious improvements compared with all the baselines, which confirmed its effectiveness. Since the number of classes of the PACS dataset (7) is obviously smaller than the number of classes of Office-Home dataset (65), the simulated label shift does not have obvious changes to the data distribution, which may lead to similar performances on the shift on PACS dataset. In addition, the number of instances in each domain of PACS dataset is relatively bigger than the number of instances in each domain of the Office-Home dataset. Thus, the baseline methods are more sensitive to label shift on the Office-Home benchmark than PACS benchmark. This also confirms the effectiveness of our method when handling a minor number of instances when the label shift problem occurs.

Time efficiency We then evaluate the time efficiency of our method, comparing it with the four principled baselines on both the PACS and Office-Home benchmark to demonstrate the effectiveness of our method. We demonstrate the time efficiency by comparing the relative average time, setting our time as the unit time for one training round. The results are presented as a relative percentage bar chart by setting the time costs of our method as a unit in Fig. 6.7. From the results, we observe that our method has similar time efficiency with MMD and MixUp methods while has better time efficiency than CDANN and MLDG. Considering the improving performances on the benchmarks and the performances demonstrated under the label shift situations, we conclude that our method reflects a better prediction accuracy with better time efficiency, which confirms its effectiveness of our methods.

6.5 Discussion and Conclusion

In this chapter, we considered the generalization property in DG problems by exploring the value of the label and semantic information across domains, which were mostly neglected by the previous work. We investigated the theoretical guarantee for a successful generalization process by focusing on how to control the target domain error. Our theoretical results revealed that to control the target risk, we should jointly control the source errors that are weighted according to label information while aligning the semantic conditional distributions between different source domains. The theoretical analysis then inspired us for an efficient algorithm to control the label distributions and match the semantic conditional distributions. The empirical results showed that our method outperformed most of the baselines, achieving state-of-the-art performances on the benchmarks. Furthermore, the time efficiency of our method showed

that it achieve better benchmark performances with better time efficiencies. Besides, our method also showed better performances under the label shift situations, which is generally not perfectly handled by the baselines.

Conclusion

In this chapter, we summarize our thesis with the main contributions and limitations as well as an overview of the future works.

In this thesis, we worked on the problems of leveraging the knowledge from different domains with different learning scenarios including domain adaptation (DA), domain generalization (DG) and multi-task learning (MTL) in different aspects, we now summarize this thesis and propose some future works.

Summary of Our Contributions

Explaining a Novel Method for Active Domain Adaptation

In Chapter 3, we focused on the conditional shift problem in domain adaptation. We theoretically analyzed the conditional shift problem in domain adaptation using the Wasserstein distance for indicating an active query strategy to migrate the disagreement term between the source and target domain. Based on this theoretical analysis, we then proposed an active query strategy to control both the uncertainty and diversity of the data instances of the target domain. The diversity score is measured from the critic model while the uncertainty score is measured through the classifier. Our empirical results showed that our algorithm improves the classification accuracy with a small query budget. When the query budget is small, our approach can have better performance than its *i.i.d.* (random) selection counterparts. Furthermore, the comparison with other query strategy based DA baselines also demonstrates the effectiveness of our algorithm.

Exploring the Category Relations for Domain Generalization

In Chapter 4, we explored the category similarities in domain generalization (DG) to learn from multiple source domains and then provide a generalization to a target domain with unknown data distributions. As summarized in section 2.3.1, we found that previous DG approaches mainly focused on learning marginal features and stacking the learned features from each source domain to generalize to a new target domain. By doing so, they ignore the

label information, and this leads to indistinguishable features with an ambiguous classification boundary.

Therefore, we started to constrain the label-similarity when extracting the invariant features for taking advantage of the label similarities for class-specific cohesion and separation of features across domains. To this end, we adopt optimal transport with Wasserstein distance, for constraining the class label similarity, for adversarial training. We also further deploy a metric learning objective to leverage the label information for achieving distinguishable classification boundary. Empirical results show that our proposed method outperforms most of the baselines with achieving the state-of-the-art performances. Furthermore, ablation studies and feature visualizations in section 4.4.4 also demonstrate the effectiveness of each component of our method.

Using a Multi-task Learning Framework to Characterize Semantic and Task Similarities

In Chapter 5, we explored a long-neglected problem in MTL, *i.e.*, the absence of label and semantic information. We provided the first theoretical analysis for MTL that considers semantic matching. In this context, our results revealed 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 $\mathcal{D}(y)$ and semantic distribution $\mathcal{D}(\mathbf{x}|y)$.

In contrast to previous theoretical results (Shui et al., 2019; Mao et al., 2020), which only consider the marginal distribution discrepancy (*e.g.*, $\mathcal{H}$-divergence), we built a complete MTL theoretical framework upon the joint distribution discrepancy. Our result provides a deeper understanding of the general problem of MTL and insights on how to extract and leverage shared knowledge in a more appropriate and principled way by exploiting the label information.

Understanding Domain Generalization through the Control of Label and Semantic Relations

In Chapter 6, we focused on developing theoretical insights into how to ensure a successful generalization process by investigating the test error on the target domain. Our results reveal the necessity of controlling the semantic conditional distributions as well as the label distribution divergence across all the source domains. Specifically, the theoretical side, we provide a framework to understand the domain generalization process upon bounding the test error on the target domain with total variation distance, thus providing a deeper understanding of the role of semantic alignment for general DG problems.

On the algorithmic side, our theoretical results inspired a novel DG algorithm that jointly minimizes the source errors as well as semantic distribution matching for all the source domains. Precisely, we proposed to simultaneously match the semantic distributions via minimizing

the centroid statistics across distributions and controlling the label distribution losses. The empirical results demonstrate that the our algorithm outperforms various strong baselines, especially when label shift occurs.

Discussions

In thesis, we have conducted both theoretically and empirically investigations on how to effectively learn transferable features from different domains. From the theoretical aspects, we implemented the statistical generalization analysis framework to explore the learning guarantees for DA, DG and MTL with especially attention to the (semantic) conditional and label distribution relations in different domains. The empirical evaluations have shown improved performances which also confirm the benefits of our theoretical analysis.

Despite from the contributions, the accomplishments in this thesis still have some limitations. As mentioned in Chapter 5 and Chapter 6, the theoretical analysis assume the number of labels in each domain is same to each other, while in many practical scenarios, the number of total labels in each domain may differ from each other, *i.e.*, the *open set* learning problems. Our preliminary results have shown that the proposed algorithms in this thesis can show better performances than the baselines under the open set situations, however, the theoretical results still open. Besides, our work introduced in Chapter 3 and Chapter 4 exploit the Wasserstein adversarial training method, which we found may have oscillated performances. Similar with the discussions in (Arjovsky et al., 2017; Gulrajani et al., 2017) we found the Wasserstein adversarial training sometimes hard to converge. However, this kind of training can show improved classification accuracy.

Possible Future Directions

In this thesis, we've focused on the learning problems of how to transfer the knowledge from different domains through the distribution matching approaches. Specifically, we focused on the DA, DG and MTL learning scenarios and showed novel theoretical insights as well as the practical implications. Through the current accomplishments, we notice some further directions that deserve to be explored :

Explore the Contrastive Learning for Transfer Learning

As a trend, we notice some contrastive learning techniques have been widely investigated for exploring the similarities via both the unsupervised (Xie et al., 2021) or supervised (Khosla et al., 2020) mode. Previous work (Chen et al., 2020c) has shown the potentials to apply contrastive learning in transfer learning. As discussed in Chapter 4, our work (Zhou et al., 2021b) shares some common idea with contrastive learning when leveraging the similarities. This kind of contrastive learning work can either be implemented as an extra regularization

objective to enhance the feature compactness. As a future direction, we plan to further explore the contrastive learning method into multi-source transfer including domain generalization and multi-task learning method.

Online or Sequential Learning Scenarios

Currently, we focused on the situation where the data are static to the learner, *i.e.*, the offline learning scenario. In the practical aspects, we can expect the data come to the learner one by one under an online learning framework. For example, if we consider a life-long learning problems in which the data is always feed to the learner time-step by time-step, then how to figure out the most transferable knowledge between each time specific data and how long should we keep the previous knowledge is interesting. Our current methodologies on distribution matching can be transformed to this kind of online or life-long learning problems. However, how to find out the useful information for future and how to ignore the useless information in the past time-steps are desirable to explore.

Besides, it will be high interesting if the model can gradually learn from the sequential data. For example, for a self-driving car system, we may pre-train the model with the perception and decision abilities before it can be deployed in the real road driving. When operating, we expect this model can learn on the fly to improve the learning performances as the car is moving on the road. We may try to implement the accomplishments in this thesis to some intelligent vehicle systems in the future.

Open Set Learning Problems

Another interesting directions will be the open set learning problems (Shu et al., 2021) involved in the multi-source transfer learning problems. As discussed before, in Chapter 5 and Chapter 6, we have explored the label shift problems in both MTL and DG, where we assume the number of classes are equal to each other across domains. The open set transfer learning problem aims at learning transferable features under the scenario that the number of labels from each domain can differ with each other. As previously noticed, some practical approaches have been devoted to the open set learning problems (Liu et al., 2019a; Shu et al., 2021), and theoretical justifications (Fang et al., 2021a,b) for the open set domain adaptation. When dealing with multi-source transfer learning problems, we need to find out how to ensure a successful generalization process. The complete theoretical analysis and practical frameworks still deserve further exploration. As a future direction, we plan to explore the contrastive learning or some other unsupervised learning methods to enhance the feature alignment so that we can reduce the problem of negative transfer in the open set learning problems.

Label Shift Problems in Other Learning Scenarios

In this thesis, we explored the label shift problems in transfer learning problems. However, this kind of label distribution shift can also exists in other learning scenarios. For example, it will be interesting to explore the label shift problems in active learning, where the query (target) pool label distribution shift clearly from the current learning data (source). It will also interesting to explore this kind of work in the similarity-based learning approaches. For example, under the metric learning or constrastive learning framework, it will be interesting to see how to design the distance so that we can neglect the abnormal labelled data to ensure the triplet or constrastive property is an interesting area in the literature. We can explore this kind approaches in the future.

To summarize, this thesis explored the distribution matching methodologies in learning the transferable feature and knowledge from different data distributions. We've studied different transfer learning scenarios from both theoretical and empirical aspects, and see how they can be applied to some other learning scenarios. For a future perspective, we believe such kind of accomplishments will be beneficial to the design of real-world learning systems.

Annexe A

Preliminary Methods Involved in Our Work

In this appendix chapter, we provide the background knowledge of Domain Adversarial Neural Network and the Wasserstein Distance Guided Representation Learning, which are two previous approaches involved in our work. We introduce them by referring their original publications to show a background knowledge for these two methods.

A.1 Introduction to Domain Adversarial Neural Network

We firstly start by introducing the *Domain Adversarial Network* (DANN) (Ganin and Lempitsky, 2015; Ganin et al., 2016), which is a well-known architecture of modern adversarial domain adaptation approaches. For this, we mainly focus on re-producing the training process of this proposed method. Based on that, we can then discuss the following substantial work proposed by others.

As illustrated in Fig. A.1, the DANN contains three main parts, a feature extractor (FE) is trained with a classification network on the source dataset under a supervised setting. Then, a domain discriminator learns to distinguish if the images are from the source or from the target domain with the help of a gradient-reversal layer. The gradient reversal layer copies data without change at the forward step and then it multiplies the gradient by $-\lambda$ at back-propagation.

Now, we explain how this network is trained. Let's consider a shallow Neural Network with input $X \in \mathbb{R}^n$. As shown in Fig. A.2. The hidden layer G_f learns a function $G_f : X \to \mathbb{R}^D$, parameterized by a weight matrix $\mathbf{W}_1$ and bias b_1. The prediction layer learns a mapping $G_y : \mathbb{R}^D \to [0,1]^L$ parameterized by $\mathbf{W}_2$ and b_2 to predict the probability to one of the total L classes.

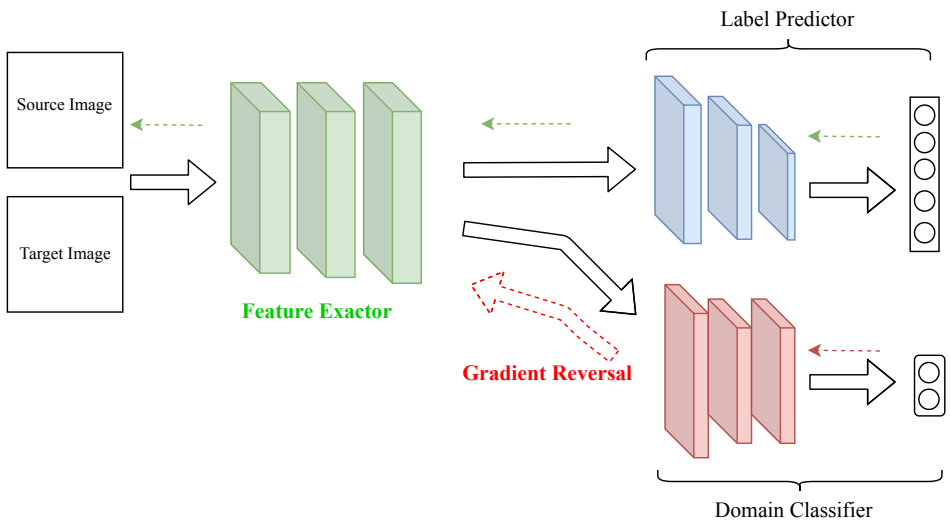


FIGURE A.1 – The framework of DANN mainly contains 3 network parts (feature extractor, label classifier and domain classifier) and a grad-reversal layer. The training process could be summarized as : 1. train feature extractor + class predictor on source data. 2. train feature extractor + domain classifier on source and target data (with grad-reversal layer). 3. use feature extractor + class predictor at test time. (Architecture of DANN is re-drawed from Ganin et al. (2016)).

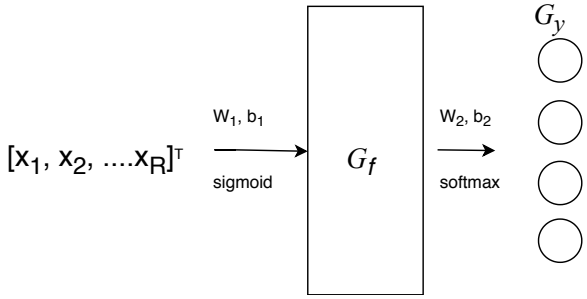


FIGURE A.2 – A shallow MLP example for DANN

Then, the prediction layer outputs will be,

$$G_y(G_f(x); W_2, b_2) = \text{softmax}(W_2 G_f(x) + b_2)$$

The classification loss function is negative log-probability of the correct label :

$$\mathcal{L}_y(G_y(G_f(x_i)), y_i) = \log \frac{1}{G_y(G_f(x))_{y_i}}$$

With this classification loss, a generalized optimization objective regularized by term $R(\mathbf{W}, \mathbf{b})$ is introduced by,

$$\min_{\mathbf{W}_1, \mathbf{W}_2, b_1, b_2} [\frac{1}{n} \sum_{i=1}^{n} \mathcal{L}_y^i(\mathbf{W}_1, \mathbf{W}_2, b_1, b_2) + \lambda \cdot R(\mathbf{W}, \mathbf{b})] \tag{A.1}$$

where $\mathbf{W}$ and $\mathbf{b}$ are the weights and biases of the whole model, respectively.

The heart of DANN is to design the *domain regularizer* derived from the domain divergence measurements (here $\mathcal{H}$- divergence). According to Ben-David et al. (2010a), the empirical $\mathcal{H}$-divergence of a hypothesis class $\mathcal{H}$ between samples $S(G_f)$ and $T(G_f)$ could be defined as :

$$\hat{d}_{\mathcal{H}}(S(G_f), T(G_f)) = 2(1 - \min_{h \in \mathcal{H}}[\frac{1}{n}\sum_{i=1}^{n}\mathbf{I}[h(G_f(\mathbf{x}_i)) = 0] + \frac{1}{n'}\sum_{i=n+1}^{N}\mathbf{I}[h(G_f(\mathbf{x}_i)) = 1]])$$

where $S(G_f) = \{G_f(\mathbf{x})|\mathbf{x} \in S\}$; $T(G_f) = \{G_f(\mathbf{x})|\mathbf{x} \in T\}$ are the source and target sample representations. The 'min' part is hard to exactly compute, Ganin et al. (2016) proposed to estimate the min part by a *domain classification layer* G_d that learns a logistic regression $G_d : \mathbb{R}^D \to [0, 1]$ parameterized by a vector-scalar pair $(\mathbf{u}, z) \in \mathbb{R}^D \times \mathbb{R}$, which measures the probability of a given input is from the source domain or the target domain :

$$G_d(G_f(\mathbf{x}; \mathbf{u}, z)) = \text{sigm}(\mathbf{u}^T G_f(\mathbf{x}) + z) \tag{A.2}$$

Then the authors defined the loss function by :

$$\mathcal{L}_d(G_d(G_f(x_i)), d_i) = d_i \log \frac{1}{G_d(G_f(x_i))} + (1 - d_i) \log \frac{1}{1 - G_d(G_f(x_i))} \tag{A.3}$$

where d_i is a binary variable (domain label). If $x_i \sim \mathcal{D}_S^X$,then $d_i = 0$, if $x_i \sim \mathcal{D}_T^X$, then $d_i = 1$. It's a surrogated loss to original $\mathcal{H}$-divergence. Then, add the domain adaptation term into the regularizer $R(\mathbf{W}, \mathbf{b})$:

$$R(\mathbf{W}, \mathbf{b}) = \max_{\mathbf{u},z}[-\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) - \frac{1}{n'}\sum_{i=n+1}^{N}\mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z)]$$

Then, the full objective function is defined by :

$$E(\mathbf{W_1}, \mathbf{W_2}, b_1, b_2, \mathbf{u}, z) = \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_y^i(\mathbf{W}_1, \mathbf{W}_2, b_1, b_2)$$
$$-\lambda(\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) + \frac{1}{n'}\sum_{i=n+1}^{N}\mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z))$$

With the general gradient descent optimizer, the network parameters could have the saddle points :

$$(\hat{\mathbf{W_1}}, \hat{\mathbf{W_2}}, \hat{b_1}, \hat{b_2}) = \arg\min_{\mathbf{W_1}, \mathbf{W_2}, b_1, b_2} E(\mathbf{W_1}, \mathbf{W_2}, b_1, b_2, \hat{\mathbf{u}}, \hat{z})$$
$$(\hat{\mathbf{u}}, \hat{z}) = \arg\max_{\mathbf{u},z} E(\hat{\mathbf{W_1}}, \hat{\mathbf{W_2}}, \hat{b_1}, \hat{b_2}, \mathbf{u}, z)$$

Here, we notice that there is an arg max term in the second equation, this could be achieved by the Domain Adversarial training method, which will be introduced in next part.

Now, we show how this procedure could be implemented in the deep neural network-based approach. We re-illustrate the main analysis process of Ganin et al. (2016) in the following parts. To be consistent with the original work, we follow their notations. For a deep neural network based adaptation task, typically we implement a feature extractor $G_f(\cdot;\theta_f)$ and source label predictor $G_y(\cdot;\theta_y)$ in supervised mode. $G_d(\cdot;\theta_d)$ is the domain classifier. Then the label prediction loss ($\mathcal{L}_y^i$) and domain prediction loss $\mathcal{L}_d$ could be defined by :

$$\mathcal{L}_y^i(\theta_f,\theta_y) = \mathcal{L}_y(G_y(G_f(\mathbf{x}_i;\theta_f);\theta_y), y_i)$$
$$\mathcal{L}_d^i(\theta_f,\theta_y) = \mathcal{L}_d(G_y(G_f(\mathbf{x}_i;\theta_\mathbf{f});\theta_d), d_i)$$

The objective function for the neural network is :

$$E(\theta_f,\theta_y,\theta_d) = \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_y^i(\theta_f,\theta_y) - \lambda(\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}_d^i(\theta_f,\theta_d) + \frac{1}{n'}\sum_{i=1}^{n}\mathcal{L}_d^i(\theta_f,\theta_d))$$

The network could be optimized by :

$$(\hat{\theta}_f,\hat{\theta}_y) = \arg\min_{\theta_f,\theta_y} E(\theta_f,\theta_y,\hat{\theta}_d)$$

$$(\hat{\theta}_d) = \arg\max_{\theta_d} E(\hat{\theta}_f,\hat{\theta}_y,\theta_d)$$

Apply gradient descent, the updating rule will be :

$$\begin{aligned}
\theta_f &\longleftarrow \theta_f - \mu(\frac{\partial\mathcal{L}_y^i}{\partial\theta_f} - \lambda\frac{\partial\mathcal{L}_d^i}{\partial\theta_f}) \\
\theta_y &\longleftarrow \theta_y - \mu\frac{\partial\mathcal{L}_y^i}{\partial\theta_y} \\
\theta_d &\longleftarrow \theta_d - \mu\frac{\partial\mathcal{L}_d^i}{\partial\theta_d}
\end{aligned} \tag{A.4}$$

The term $-\lambda\frac{\partial\mathcal{L}_d^i}{\partial\theta_f}$ is achieved by the gradient reversal layer. Such gradient reversal layer architecture is a powerful tool to achieve the adversarial training. It has inspired some recent adversarial approaches for domain adaptation (e.g. Cao et al., 2018b, 2019; Wen et al., 2019a,b). As stated before, the VC-dimension of $\mathcal{H}$-divergence is high, which makes it hard to compute in neural network-based approaches (Arjovsky et al., 2017). Recent work of the Wasserstein Distance-based approach has been proposed by academia. We highlight this in the next section.

A.2 Optimal Transport and Wasserstein Metrics

Optimal Transport (OT) theory was firstly introduced by Monge (1781) to study the problem of resource allocation with concern to transport the resources from one point to another with

minimized transportation cost. It was then developed into a statistical learning task that tries to minimize the distances between distributions (*e.g.* (Redko et al., 2017)).

As introduced in section 1.5.1, Wasserstein-1 distance is a kind of IPM metric, which can be computed by,

$$W_1(\mathcal{D}_i, \mathcal{D}_j) = \sup_{||f||_L<1} \mathbb{E}_{\mathbf{x}\in\mathcal{D}_i} f(\mathbf{x}) - \mathbb{E}_{\mathbf{x}'\in\mathcal{D}_j} f(\mathbf{x}') \tag{A.5}$$

Eq. A.5 actually is known as the *Kantorovich-Rubinstein duality* of Wasserstein Distance, we now show how we can derive this duality from the original OT theory.

We begin by illustrating the definition of Lipschitz Continuous functions and then introduce the Wasserstein Distances as a distribution measure.

Definition A.1. ***Lipschiz Continuous Function*** *(Shalev-Shwartz and Ben-David, 2014) : Given a metric space $(\mathcal{X}, \rho)$, A function $f : \mathcal{X} \to \mathbb{R}$ is K-Lipschitz with respect to the metric ρ if for every $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$, we have that $|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq K\rho(\mathbf{x}_1 - \mathbf{x}_2)$*

We then use $||f||_{Lip}$ denote by the smallest K to satisfy the inequality above. Then, we will have the definition of Wasserstein Distance as,

Definition A.2. ***Wasserstein Distance*** *(Wainwright, 2019) : Given two probability distribution $\mathcal{D}_i$ and $\mathcal{D}_j$ on $\mathcal{X}$, the Wasserstein Distance is measured by*

$$W_p(\mathcal{D}_i, \mathcal{D}_j) = \sup_{||f||_{Lip}\leq 1} \Big[\int f d\mathcal{D}_i - \int f d\mathcal{D}_j \Big] \tag{A.6}$$

As we can see in Eq. A.6, the 'sup' of the integration is not able to directly. Recent works (Courty et al., 2016; Redko et al., 2017; Shen et al., 2018) adopt the Kantorovich-Rubinstein duality of Wasserstein Distance to achieve the adversarial objective, we introduce them in the next section.

A.2.1 Kantorovich-Rubinstein duality of Wasserstein Distance

The duality theory by Hanin (1992) showed that *any Wasserstein Distance has an equivalent coupling-function based distance*. In this section, we introduce the well-known Kantorovich-Rubinstein duality of Wasserstein Distance. Firstly, the coupling function is usually defined by,

Definition A.3. ***Coupling function*** *(Hanin, 1992) A distribution Π on $\mathcal{X} \times \mathcal{X}$ is a* coupling *of pair $(\mathcal{D}_i, \mathcal{D}_j)$, if the marginal distributions are coincided with $\mathcal{D}_i$ and $\mathcal{D}_j$.*

We follow the demonstration of Wainwright (2019) to show its relations to Wasserstein Distance. Let $f : \mathcal{X} \to \mathbb{R}$ be any $1 - Lipschiz$ function, denote Π by any coupling function, then,

$$\begin{aligned} \int \rho(\mathbf{x}, \mathbf{x}') d\Pi(\mathbf{x}, \mathbf{x}') &\geq \int (f(\mathbf{x}) - f(\mathbf{x}')) d\Pi(\mathbf{x}, \mathbf{x}') \\ &= \int f(d\mathcal{D}_i - d\mathcal{D}_j) \end{aligned} \tag{A.7}$$

According to the *Kantorovich-Rubinstein duality*, if we minimize the Wasserstein Distance over all the possible couplings, then we will have the following equivalence :

$$\sup_{||f||_{Lip} \leq 1} [\int f d\mathcal{D}_i - \int f d\mathcal{D}_j] = \inf_{\Pi} \int_{\mathcal{X} \times \mathcal{X}} \rho(\mathbf{x}, \mathbf{x}') d\Pi(\mathbf{x}, \mathbf{x}') = \inf_{\Pi} \mathbb{E}[\rho(\mathcal{X}, \mathcal{X}')] \tag{A.8}$$

where $\Pi(\mathcal{D}_i, \mathcal{D}_j)$ is a collection of all joint probability measures on $\mathcal{X} \times \mathcal{X}$ with marginals $\mathcal{D}_i$ and $\mathcal{D}_j$. This duality function reflected by Eq.A.8 plays a fundamental role in many recent works on Wasserstein Distance (e.g. Redko et al., 2017; Shen et al., 2018). The transportation process can be described as assuming there are two resources of data that come from two distributions $\mathcal{D}_i$ and $\mathcal{D}_j$. The goal of OT aims to move (push) the data with mass p_i from $\mathcal{D}_i$ to $\mathcal{D}_j$ with a minimized cost.

Consider two probability distributions $\mathcal{D}_i$ and $\mathcal{D}_j$ having densities p_i and p_j respectively. The p_i is typically considered as the initial density on source distribution $\mathcal{D}_i$ over space $\mathcal{X}$. And the p_j is the desired density of math in the target distribution. The term *transportation cost* could be measured by moving a small incremental mass $\mathrm{d}\mathbf{x}$ to $\mathrm{d}\mathbf{x}'$. Its quantity could be computed by $\rho(\mathbf{x}, \mathbf{x}')\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{x}'$.

Denote $\pi(\mathbf{x}, \mathbf{x}')$ the joint distribution which is also known as *Transportation Plan*. It stands for shifting the mass that p_i is moved (push-forward) to p_j.

Now, take those parts above into consideration, we can formulate the transportation cost as :

$$\int_{\mathcal{X} \times \mathcal{X}} \rho(\mathbf{x}, x')\pi(\mathbf{x}, x')\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{x}' \tag{A.9}$$

Then, by minimizing Eq. A.9, we have :

$$\begin{aligned} &\mathrm{argmin}_{\gamma} \int_{\mathcal{X}_1 \times \mathcal{X}_2} \rho(\mathbf{x}, \mathbf{x}')^p d\gamma(\mathbf{x}, \mathbf{x}') \\ &\text{s.t.} \mathbf{P}^{\mathcal{X}_1} \# \gamma = \mathbf{D}_i; \mathbf{P}^{\mathcal{X}_2} \# \gamma = \mathbf{D}_j \end{aligned}$$

Where $\mathbf{P}^{\mathcal{X}_1}$ is projection over $\mathcal{X}_1$ and $\#$ denotes the push-forward measure. This problem admits an unique solution γ_0 which allows to define the Wasserstein distance of order p between $\mathbf{D}_i$ and $\mathbf{D}_j$ for any $p \geq 1$:

$$W_p^p(\mathbf{D}_i, \mathbf{D}_j) = \inf_{\gamma \in \Pi(\mathbf{D}_i, \mathbf{D}_j)} \int_{\mathcal{X} \times \mathcal{X}} \rho(\mathbf{x}, \mathbf{x}')^p d\gamma(\mathbf{x}, \mathbf{x}')$$

where $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is the cost function for transporting one unit of mass $\mathbf{x}$ to $\mathbf{x}'$. In practice, we usually use the Wasserstein-1 distance, which is defined by,

$$W_1(\mathcal{D}_i, \mathcal{D}_j) = \sup_{||f||_L<1} \mathbb{E}_{\mathbf{x}\in\mathcal{D}_i} f(\mathbf{x}) - \mathbb{E}_{\mathbf{x}'\in\mathcal{D}_j} f(\mathbf{x}') \tag{A.10}$$

This kind of OT theory with Wasserstein distance then inspires many transfer learning methods (Arjovsky et al., 2017; Redko et al., 2017). In our Chapter 3 and Chapter 4, we adopt the adversarial training method, namely *Wasserstein Distance Guided Representation Learning* (WDGRL), proposed by Shen et al. (2018). We further elaborate the method in section 1.5.2 as well as Appendix A.3.

A.3 Domain Adversarial Training with Wasserstein Distance

Optimal Transport technique with Wasserstein adversarial training (Shen et al., 2018) has been investigated in domain adaptation (Courty et al., 2016; Redko et al., 2019). We leveraged the Wasserstein adversarial training method in our work (Zhou et al., 2021c,b), which were represented in Chapter 3 and Chapter 4, respectively.

In this section, we present a brief introduction on Optimal Transport for Domain Adaptation with Wasserstein Distance. We highlight the optimal transport based analysis by Courty et al. (2016), which systemically introduced the optimal transport for domain adaptation and then Redko et al. (2017) theoretically provide the concentration bounds using Wasserstein distance to bound the transfer errors between the source and target domain. The theoretical analysis of this work is based on kernel functions restricted in Reproduced Kernel Hilbert Space. This work has been extended by Shen et al. (2018) extend the work to Lipschitz-continuous functions.

Theorem A.1. *(Shen et al., 2018) Let $\mathcal{D}_\mathcal{S}, \mathcal{D}_\mathcal{T}$ be two probability measures, assume $\forall h \in \mathcal{H}$ are* K-Lipschitz continuous *functions and cost function for OT is the Euclidean distance $\rho(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$. Then we have that :*

$$R_\mathcal{T}(h, h') \leq \mathbb{R}_\mathcal{S}(h, h') + 2KW_1(\mathcal{D}_\mathcal{S}, \mathcal{D}_\mathcal{T}) + \lambda$$

for every hypothesis $h, h' \in \mathcal{H}$ and λ is the combined error of the ideal hypothesis $h^\star$ that minimizes the combined error $R_\mathcal{S}(h) + R_\mathcal{T}(h)$.

Based on this theoretical results, the authors proposed an adversarial training method for domain adaptation, namely Wasserstein Distance Guided Representation Learning (WDGRL) algorithm. Fig. A.3 illustrates the model workflow of WDGRL. The network receives a pair of instances from the source and target domain. The feature extractor and classifier, parameterized by θ_f and by θ_c are denoted by F and C respectively. The feature extractor is trained to learn invariant features, and the classifier is expected to learn the conditional prediction relations $\mathbb{P}(y|\mathbf{x})$ for predicting the instances from both source and target domain correctly.

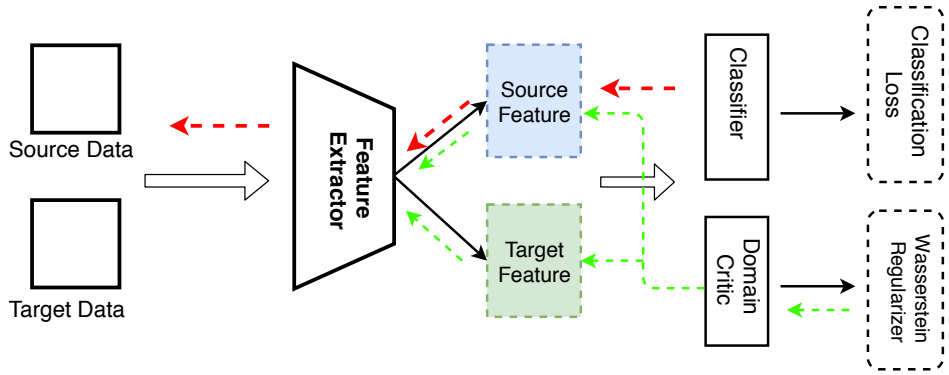


FIGURE A.3 – Workflow of Wasserstein Distance Guided Representation Learning (WDGRL) method. The model consists of three major parts : feature extractor, classifier and domain critic. The feature extractor could extract the features both from source and target domains. The feature extractor and classifier are trained with labelled source instances while the domain critic is trained to measure the empirical Wasserstein distance between the source and target domain. The red dashed arrows stands for the back-propagation paths for the classification loss while the green dashed lines stands for the back-propagation path for the critic loss.

Then, there follows the domain critic network ϕ, parameterized by θ_ϕ. This network takes both the source domain and the target domain features $\mathcal{Z}_S, \mathcal{Z}_T$ as input and estimates the empirical Wasserstein Distance between the source and target domain through a pair of batched instances $\mathcal{X}_\mathcal{S}$ and $\mathcal{X}_\mathcal{T}$,

$$W_1(\mathcal{X}_\mathcal{S}, \mathcal{X}_\mathcal{T}) = \frac{1}{n_s} \sum_{\mathbf{x}_s \in \mathcal{X}_\mathcal{S}} \phi(F(\mathbf{x}_s)) - \frac{1}{n_t} \sum_{\mathbf{x}_t \in \mathcal{X}_\mathcal{T}} \phi(F(\mathbf{x}_t)) \tag{A.11}$$

Since Wasserstein Distance is continuous and is differential almost everywhere, we can optimize the parameters of the critic network together with the classifier. The feature extractor F is then trained to minimize the estimated Wasserstein Distance in an adversarial manner with the critic ϕ. Then, the goal of the adversarial training is described by

$$\min_{\theta_f, \theta_c} \max_{\theta_d} \mathcal{L}_{cls} + \lambda_w (W_1(\mathcal{X}_\mathcal{S}, \mathcal{X}_\mathcal{T}) - \mathcal{L}_{grad}) \tag{A.12}$$

where λ_w is a parameter to regularize the critic loss and $\mathcal{L}_{grad}$ is gradient penalty method suggested by Gulrajani et al. (2017) which can help to prevent gradient vanishing or exploding problems caused by weight clipping.

$$\mathcal{L}_{grad}(h)(\mathcal{Z}) = (||\nabla_h f(h)||_2 - 1)^2 \tag{A.13}$$

In previous works, we found that this approach is useful and can align the marginal feature distribution perfectly. We use this method as a starting point for our proposed approach (Zhou et al., 2021c) as well as (Zhou et al., 2021b).

Annexe B

Proofs to Theoretical Results

In this chapter, we provide the proofs to the theoretical results proposed in our work.

B.1 Proof to Theoretical Results in Chapter 3

Proof to Theorem 3.1 The proof is based on Lemma 1 of Shen et al. (2018) and is symmetric to the proof of Theorem 3 of Zhao et al. (2019a). We first give the Lemma 1 as follows,

Lemma B.1. *(Lemma 1 of Shen et al. (2018)) Let $\mathcal{D}_\mathcal{S}, \mathcal{D}_\mathcal{T}$ be two probability measures, assuming that all the hypothesis $\forall h \in \mathcal{H}$ are* 1-Lipschitz continuous *functions. Then we have that :*

$$R_\mathcal{T}(h, h') \leq R_\mathcal{S}(h, h') + 2W_1(\mathcal{D}_\mathcal{S}, \mathcal{D}_\mathcal{T})$$

for every hypothesis $h, h' \in \mathcal{H}$.

We then could finish the proof of Theorem 3.1 by,

Proof. Based on Lemma B.1, let $h' = f_t$, we have

$$R_\mathcal{T}(h, f_t) \leq R_\mathcal{S}(h, f_t) + 2W_1(\mathcal{D}_\mathcal{S}, \mathcal{D}_\mathcal{T})$$

We noticed that

$$\begin{aligned} R_\mathcal{S}(h, f_t) &= \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_\mathcal{S}}|h(\mathbf{x}) - f_t(\mathbf{x})| \\ &\leq \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_\mathcal{S}}|h(\mathbf{x}) - h_s(\mathbf{x})| + \mathbb{E}_{x\sim\mathcal{D}_\mathcal{S}}|h_\mathcal{S}(\mathbf{x}) - f_\mathcal{T}(\mathbf{x})| \\ &= R_\mathcal{S}(h) + \mathbb{E}_{\sim\mathcal{D}_\mathcal{S}}|h_s(\mathbf{x}) - f_t(\mathbf{x})| \end{aligned}$$

Plugging in we have the result. □

Proof to Corollary 3.1 To complete the proof of this corollary, first show the following lemma.

Lemma B.2. *(Theorem 2.1 of Bolley et al. (2007)) Let $\mathcal{D}$ be a probability measure in $\mathbb{R}^d$ and satisfies the $T_1(\lambda)$ inequality. Denote $\hat{\mathcal{D}} = \frac{1}{N}\sum_{i=1}^{N}\Delta_{x_i}$ by the associated empirical defined on a sample of independent variables $\{x_i\}_{i=1}^{N}$ drawn i.i.d from $\mathcal{D}$. Then, for any $d' \geq d$ and $\lambda' \leq \min(R_s + R_t)$ there exists some constant N_0 depending on d' and some square exponential moment of $\mathcal{D}$ s.t. for any $\epsilon > 0$ and $N \geq N_0 \max(\epsilon^{-(d+2)}, 1)$, we have*

$$\mathbb{P}[W_1(\mathcal{D}, \hat{\mathcal{D}}) > \epsilon] \leq \exp(-\frac{\lambda'}{2}N\epsilon^2) \tag{B.1}$$

We shall refer the reader directly to Bolley et al. (2007) for the proof.

Now, we could give the proof of Theorem 2 in the following.

Proof. The proof of this corollary is similar to Theorem 4 in Redko et al. (2017). We borrow the idea from them and extend it to our setting. Assume the learner have access to β of total target samples. Then,

$$\begin{aligned}
R_{\mathcal{T}}(h) &\leq R_{\mathcal{S}}(h) + 2W_1(\mathcal{D}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}}) + \mathbb{E}_{\mathcal{D}_{\mathcal{S}}}[|f_{\mathcal{S}} - f_{\mathcal{T}}|] \\
&\leq R_{\mathcal{S}}(h) + \mathbb{E}_{\mathcal{D}_{\mathcal{S}}}[|f_{\mathcal{S}} - f_{\mathcal{T}}|] + 2[W_1(\mathcal{D}_{\mathcal{S}}, \hat{\mathcal{D}}_{\mathcal{S}}) + W_1(\hat{\mathcal{D}}_{\mathcal{S}}, \mathcal{D}_{\mathcal{T}})] \\
&\leq R_{\mathcal{S}}(h) + \mathbb{E}_{\mathcal{D}_{\mathcal{S}}}[|f_{\mathcal{S}} - f_{\mathcal{T}}|] + 2[\sqrt{2\log(\frac{1}{\delta})/(N_s + \beta N_t)\lambda'} + W_1(\hat{\mathcal{D}}_{\mathcal{S}}, \hat{\mathcal{D}}_{\mathcal{T}}) + W_1(\hat{\mathcal{D}}_{\mathcal{T}}, \mathcal{D}_{\mathcal{T}})] \\
&\leq R_{\mathcal{S}}(h) + 2W_1(\hat{\mathcal{D}}_{\mathcal{S}}, \hat{\mathcal{D}}_{\mathcal{T}}) + \mathbb{E}_{\mathcal{D}_{\mathcal{S}}}[|f_{\mathcal{S}} - f_{\mathcal{T}}|] + 2\sqrt{2\log(\frac{1}{\delta})/\lambda'}(\sqrt{\frac{1}{N_s + \beta N_t}} + \sqrt{\frac{1}{N_t - \beta N_t}})
\end{aligned} \tag{B.2}$$

The first line is directly from Theorem 1, the second inequality is based from the triangle inequality of Wasserstein distance, third and fourth inequalities are based on Theorem 2.1 of Bolley et al. (2007) (Lemma B.2 above) and Theorem 2.1 of Courty et al. (2017). Besides, the second and third inequality based on the assumption that the query samples drawn from target set would not change the two distribution too much. This assumption can hold since if the active selected samples can largely influence the distribution, then the labeling can cost too much, which contradict to the goal of active learning. Then, we conclude the proof. □

We shall notice that once the set X_s, X_t and the hypothesis class has been chosen, the last three terms of Eq.B.2 are fixed. This theorem showed that the target error can be bounded by the summation of Wasserstein distances between source domains and target domain, plus some fixed term. So the learner shall get smaller target error by minimizing the Wasserstein distance and source error.

B.2 Proof to Theoretical Results in Chapter 5

To conclude our main results, we first show a lemma involved in the proof.

Lemma B.3. *[Lemma of Shui et al. (2020a)] Let $Z \in \mathcal{Z}$ be the real valued integrable random variable, let P and Q are two distributions on a common space $\mathcal{Z}$ such that Q is absolutely continuous w.r.t. P. If for any function f and $\lambda \in \mathbb{R}$ such that $\mathbb{E}_P[e^{\lambda(f(z)-\mathbb{E}_P(f(z))}] < \infty$, then we have :*

$$\lambda(\mathbb{E}_Q f(z) - \mathbb{E}_P f(z)) \leq D_{KL}(Q\|P) + \log \mathbb{E}_P[e^{\lambda(f(z)-\mathbb{E}_P(f(z))}]$$

Where $D_{KL}(Q\|P)$ is the Kullback–Leibler divergence between distribution Q and P, and the equality arrives when $f(z) = \mathbb{E}_P f(z) + \frac{1}{\lambda}\log(\frac{dQ}{dP})$.

Proof to Lemma B.3

Proof. Let g be **any** function such that $\mathbb{E}_P[e^{g(z)}] < \infty$, then we define a random variable $Z_g(z) = \frac{e^{g(z)}}{\mathbb{E}_P[e^{g(z)}]}$, then, $\mathbb{E}_P(Z_g) = 1$. Assume another distribution Q such that Q (with distribution density $q(z)$) is absolutely continuous w.r.t. P (with distribution density $p(z)$), then we have :

$$\begin{aligned}
\mathbb{E}_Q[\log Z_g] &= \mathbb{E}_Q[\log \frac{q(z)}{p(z)} + \log(Z_g \frac{p(z)}{q(z)})] \\
&= D_{\mathrm{KL}}(Q\|P) + \mathbb{E}_Q[\log(Z_g \frac{p(z)}{q(z)})] \\
&\leq D_{\mathrm{KL}}(Q\|P) + \log \mathbb{E}_Q[\frac{p(z)}{q(z)} Z_g] \\
&= D_{\mathrm{KL}}(Q\|P) + \log \mathbb{E}_P[Z_g]
\end{aligned}$$

Since $\mathbb{E}_P[Z_g] = 1$ and according to the definition we have $\mathbb{E}_Q[\log Z_g] = \mathbb{E}_Q[g(z)] - \mathbb{E}_Q \log \mathbb{E}_P[e^{g(z)}] = \mathbb{E}_Q[g(z)] - \log \mathbb{E}_P[e^{g(z)}]$ (since $\mathbb{E}_P[e^{g(z)}]$ is a constant w.r.t. Q) and we therefore have :

$$\mathbb{E}_Q[g(z)] \leq \log \mathbb{E}_P[e^{g(z)}] + D_{\mathrm{KL}}(Q\|P) \tag{B.3}$$

Since this inequality holds for any function g with finite moment generation function, then we let $g(z) = \lambda(f(z) - \mathbb{E}_P f(z))$ such that $\mathbb{E}_P[e^{f(z)-\mathbb{E}_P f(z)}] < \infty$. Therefore we have $\forall\lambda$ and f we have :

$$\mathbb{E}_Q \lambda(f(z) - \mathbb{E}_P f(z)) \leq D_{\mathrm{KL}}(Q\|P) + \log \mathbb{E}_P[e^{\lambda(f(z)-\mathbb{E}_P f(z)}]$$

Since we have $\mathbb{E}_Q \lambda(f(z) - \mathbb{E}_P f(z)) = \lambda \mathbb{E}_Q(f(z) - \mathbb{E}_P f(z))) = \lambda(\mathbb{E}_Q f(z) - \mathbb{E}_P f(z))$, therefore we have :

$$\lambda(\mathbb{E}_Q f(z) - \mathbb{E}_P f(z)) \leq D_{\mathrm{KL}}(Q\|P) + \log \mathbb{E}_P[e^{\lambda(\mathbb{E}_Q f(z)-\mathbb{E}_P f(z))}]$$

As for the attainment in the equality of Eq.(B.3), we can simply set $g(z) = \log(\frac{q(z)}{p(z)})$, then we can compute $\mathbb{E}_P[e^{g(z)}] = 1$ and the equality arrives. Therefore in Lemma 1, the equality reaches when $\lambda(f(z) - \mathbb{E}_P f(z)) = \log(\frac{dQ}{dP})$. □

Proof to Theorem 5.1 Now, we could prove the main results of Theorem 5.1. For task t we have :

$$\begin{aligned} |R_t(h) - R_{\boldsymbol{\alpha}_t}(h)| &= |R_t(h) - \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} R_i(h)| \\ &\leq \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} |R_t(h) - R_i(h)| \\ &= \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} |\mathbb{E}_{\mathcal{D}_t} \ell(h) - \mathbb{E}_{\mathcal{D}_i} \ell(h)| \end{aligned} \tag{B.4}$$

Now we bound $\mathbb{E}_{\mathcal{D}_t} \ell(h) - \mathbb{E}_{\mathcal{D}_i} \ell(h)$. According to Lemma B.3, for two distribution P and Q, we have

$$\lambda(\mathbb{E}_Q f(z) - \mathbb{E}_P f(z)) \leq D_{\mathrm{KL}}(Q\|P) + \log \mathbb{E}_P[e^{\lambda(f(z) - \mathbb{E}_P(f(z))}]$$

$\forall \lambda > 0$ we have :

$$\mathbb{E}_Q f(z) - \mathbb{E}_P f(z) \leq \frac{1}{\lambda}(\log \mathbb{E}_P \ e^{[\lambda(f(z) - \mathbb{E}_P f(z))]} + D_{\mathrm{KL}}(Q\|P)) \tag{B.5}$$

And $\forall \lambda < 0$ we have :

$$\mathbb{E}_Q f(z) - \mathbb{E}_P f(z) \geq \frac{1}{\lambda}(\log \mathbb{E}_P \ e^{[\lambda(f(z) - \mathbb{E}_P f(z))]} + D_{\mathrm{KL}}(Q\|P)) \tag{B.6}$$

Then we introduce an intermediate distribution $\mathcal{M}(z) = \frac{1}{2}(\mathcal{D}_i(z) + \mathcal{D}_t(z))$, then $\mathrm{supp}(\mathcal{D}_i) \subseteq \mathrm{supp}(\mathcal{M})$ and $\mathrm{supp}(\mathcal{D}_t) \subseteq \mathrm{supp}(\mathcal{M})$, and let $f = \ell$.

Since ℓ is bounded through L, then $\ell - \mathbb{E}_P \ell$, according to Wainwright (2019)(Chapter 2.1.2), is sub-Gaussian with parameter at most $\sigma = \frac{L}{2}$, then we can apply Sub-Gaussian property to bound the log moment generation function :

$$\log \mathbb{E}_P \ e^{[\lambda(\ell(z) - \mathbb{E}_P \ell(z))]} \leq \log e^{\frac{\lambda^2 \sigma^2}{2}} \leq \frac{\lambda^2 L^2}{8}.$$

In Eq.(B.5), we let $Q = \mathcal{D}_t$ and $P = \mathcal{M}$, then $\forall \lambda > 0$ we have :

$$\mathbb{E}_{\mathcal{D}_t} \ \ell(z) - \mathbb{E}_{\mathcal{M}} \ \ell(z) \leq \frac{L^2 \lambda}{8} + \frac{1}{\lambda} D_{\mathrm{KL}}(\mathcal{D}_t\|\mathcal{M}) \tag{B.7}$$

In Eq.(B.6), we let $Q = \mathcal{D}_i$ and $P = \mathcal{M}$, then $\forall \lambda < 0$ we have :

$$\mathbb{E}_{\mathcal{D}_i} \ \ell(z) - \mathbb{E}_{\mathcal{M}} \ \ell(z) \geq \frac{L^2 \lambda}{8} + \frac{1}{\lambda} D_{\mathrm{KL}}(\mathcal{D}_i\|\mathcal{M}) \tag{B.8}$$

In Eq.(B.7), we denote $\lambda = \lambda_0 > 0$ and $\lambda = -\lambda_0 < 0$ in Eq.(B.8). Then Eq.(B.7), Eq.(B.8) can be reformulated as :

$$\begin{aligned} \mathbb{E}_{\mathcal{D}_t} \ \ell(z) - \mathbb{E}_{\mathcal{M}} \ \ell(z) &\leq \frac{L^2 \lambda_0}{8} + \frac{1}{\lambda_0} D_{\mathrm{KL}}(\mathcal{D}_t\|\mathcal{M}) \\ \mathbb{E}_{\mathcal{M}} \ \ell(z) - \mathbb{E}_{\mathcal{D}_i} \ \ell(z) &\leq \frac{L^2 \lambda_0}{8} + \frac{1}{\lambda_0} D_{\mathrm{KL}}(\mathcal{D}_i\|\mathcal{M}) \end{aligned} \tag{B.9}$$

Adding the two inequalities in Eq.(B.9), we therefore have :

$$\mathbb{E}_{\mathcal{D}_t}\ \ell(z) - \mathbb{E}_{\mathcal{D}_i}\ \ell(z) \leq \frac{1}{\lambda_0}(D_{\mathrm{KL}}(\mathcal{D}_i\|\mathcal{M}) + D_{\mathrm{KL}}(\mathcal{D}_t\|\mathcal{M})) + \frac{\lambda_0}{4}L^2 \tag{B.10}$$

Then, plug Eq. (B.10) into Eq. (B.4), sum over $t = 1, \ldots, T$ and compute the average, and noting $D_{\mathrm{JS}}(\mathcal{D}_i\|\mathcal{D}_j) = \frac{1}{2}[D_{\mathrm{KL}}(\mathcal{D}_i\|\mathcal{M}) + D_{\mathrm{KL}}(\mathcal{D}_j\|\mathcal{M})]$ we have

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

Proof to Corollary 5.1

Proof. We notice that,

$$\begin{aligned}
&2D_{\mathrm{JS}}(\mathcal{D}_t(\mathbf{x}, y)\|\mathcal{D}_i(\mathbf{x}, y)) = D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}, y)\|\mathcal{M}(\mathbf{x}, y)) + D_{\mathrm{KL}}(\mathcal{D}_i(\mathbf{x}, y)\|\mathcal{M}(\mathbf{x}, y)) \\
&= D_{\mathrm{KL}}(\mathcal{D}_t(y)\|\mathcal{M}(y)) + \mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) \\
&+ D_{\mathrm{KL}}(\mathcal{D}_i(y)\|\mathcal{M}(y)) + \mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{KL}}(\mathcal{D}_i(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) \\
&= 2D_{\mathrm{JS}}(\mathcal{D}_t(y)\|\mathcal{D}_i(y)) + \mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) + \mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{KL}}(\mathcal{D}_i(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y))
\end{aligned}$$

Then, we provide two bounded term for the last two KL divergence based terms,

$$\begin{aligned}
&\mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) \\
&\leq \mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) + \mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{KL}}(\mathcal{D}_i(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) \\
&= 2\mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{JS}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{D}_i(\mathbf{x}|y))
\end{aligned}$$

$$\begin{aligned}
&\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) \leq \mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{KL}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) + \mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{KL}}(\mathcal{D}_i(\mathbf{x}|y)\|\mathcal{M}(\mathbf{x}|y)) \\
&= 2\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{JS}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{D}_i(\mathbf{x}|y))
\end{aligned}$$

$$D_{\mathrm{JS}}(\mathcal{D}_t(\mathbf{x}, y)\|\mathcal{D}_i(\mathbf{x}, y)) \leq D_{\mathrm{JS}}(\mathcal{D}_t(y)\|\mathcal{D}_i(y)) + \mathbb{E}_{y\sim\mathcal{D}_t(y)} D_{\mathrm{JS}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{D}_i(\mathbf{x}|y)) + \mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\mathrm{JS}}(\mathcal{D}_t(\mathbf{x}|y)\|\mathcal{D}_i(\mathbf{x}|y))$$

Plug into Theorem 5.1, we conclude the proof for Corollary 5.1. □

B.3 Proof to Theoretical Results in Chapter 6

In this section we provide the proof to Theorem 6.1, Corollary 6.1 and Corollary 6.2.

B.3.1 Proof to Theorem 6.1

Proof. Let's first consider the risk on the target domain *w.r.t* to the nearest source domain $\mathcal{D}^\star$,

$$R_{\mathcal{D}_\mathcal{T}}(h) \leq \min_{\mathcal{D}_1,\dots,\mathcal{D}_m} R_{\mathcal{D}_i} + d_{TV}(\mathcal{D}_\mathcal{T}, \mathcal{D}_i) \tag{B.11}$$

In the context of DG, the learner has no access to the target domain, so we have no idea about which source domain is the nearest one to the target. In this case, we try to find out the minimization over all the source domains,

$$\begin{aligned}
R_{\mathcal{D}_\mathcal{T}} \leq R_{\mathcal{D}_1}(h) + d_{TV}(\mathcal{D}_\mathcal{T}, \mathcal{D}_1) \leq R_{\mathcal{D}_1} + d_{TV}(\mathcal{D}^\star, \mathcal{D}_1) \\
+ d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T}) \\
\dots \\
R_{\mathcal{D}_\mathcal{T}} \leq R_{\mathcal{D}_i}(h) + d_{TV}(\mathcal{D}_\mathcal{T}, \mathcal{D}_i) \leq R_{\mathcal{D}_i} + d_{TV}(\mathcal{D}^\star, \mathcal{D}_i) \\
+ d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T}) \\
\dots \\
R_{\mathcal{D}_\mathcal{T}} \leq R_{\mathcal{D}_m}(h) + d_{TV}(\mathcal{D}_\mathcal{T}, \mathcal{D}_m) \leq R_{\mathcal{D}_m} + d_{TV}(\mathcal{D}^\star, \mathcal{D}_m) \\
+ d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T})
\end{aligned} \tag{B.12}$$

Sum over all the source domain i, we have,

$$\begin{aligned}
m \cdot R_{\mathcal{D}_\mathcal{T}}(h) \leq & R_{\mathcal{D}_1,\dots,\mathcal{D}_m}(h) + m \cdot d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T}) \\
& + \sum_i d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T})
\end{aligned} \tag{B.13}$$

Then, we have,

$$\begin{aligned}
R_{\mathcal{D}_\mathcal{T}}(h) \leq & \frac{1}{m} R_{\mathcal{D}_1,\dots,\mathcal{D}_m}(h) + d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T}) \\
& + \frac{1}{m} \sum_i d_{TV}(\mathcal{D}^\star, \mathcal{D}_\mathcal{T})
\end{aligned} \tag{B.14}$$

Note $d_{TV}(\mathcal{D}^\star, \mathcal{D}_t) = \epsilon^\star$ and $\frac{1}{m} R_{\mathcal{D}_1,\dots,\mathcal{D}_m}(h)$ is the averaged source errors, we conclude the proof. □

B.3.2 Proof to Corollary 6.1

Since Theorem 6.1 is represented by the joint distribution, in order to show the insights that can motivate the benefits on controlling the semantic and label distribution, we can further provide the proof of Corollary 6.1.

Proof. Since $d_{TV}(\mathcal{D}^\star, \mathcal{D}_i) \leq 2\sqrt{D_{JS}(\mathcal{D}^\star||\mathcal{D}_i)}$, plug into Eq. 6.1, we have

$$\begin{aligned} R_{\mathcal{D}_\mathcal{T}}(h) &\leq \frac{1}{m}\sum_{i=1}^{m} R_{\mathcal{D}_i}(h) + \epsilon^\star + \frac{1}{m}\sum_{i}^{m} d_{TV}(\mathcal{D}^\star(\mathbf{x}, y), \mathcal{D}_i(\mathbf{x}, y)) \\ &\leq \frac{1}{m}\sum_{i=1}^{m} R_{\mathcal{D}_i}(h) + \epsilon^\star + \frac{2}{m}\sum_{i}^{m} [\sqrt{D_{JS}(\mathcal{D}^\star(\mathbf{x}, y) \| \mathcal{D}_i(\mathbf{x}, y))}] \end{aligned} \tag{B.15}$$

Now we need to bound the third term of Eq. B.15. Similar with the proof to Corollary 5.1 (Zhou et al., 2021a), we can introduce an intermediate distribution $\mathcal{M}(\mathbf{x}) = \frac{1}{2}(\mathcal{D}^\star(\mathbf{x}) + \mathcal{D}_i(\mathbf{x}))$, then $\text{supp}(\mathcal{D}_i) \subseteq \text{supp}(\mathcal{M})$ we notice that,

$$\begin{aligned} &2D_{\text{JS}}(\mathcal{D}^\star(\mathbf{x}, y) \| \mathcal{D}_i(\mathbf{x}, y)) = D_{\text{KL}}(\mathcal{D}^\star(\mathbf{x}, y) \| \mathcal{M}(\mathbf{x}, y)) \\ &+ D_{\text{KL}}(\mathcal{D}_i(\mathbf{x}, y) \| \mathcal{M}(\mathbf{x}, y)) \\ &= D_{\text{KL}}(\mathcal{D}^\star(y) \| \mathcal{M}(y)) + \mathbb{E}_{x\sim\mathcal{D}^\star(y)} D_{\text{KL}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \\ &+ D_{\text{KL}}(\mathcal{D}_i(y) \| \mathcal{M}(y)) + \mathbb{E}_{x\sim\mathcal{D}_i(y)} D_{\text{KL}}(\mathcal{D}_i(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \\ &= 2D_{\text{JS}}(\mathcal{D}^\star(y) \| \mathcal{D}_i(y)) + \mathbb{E}_{x\sim\mathcal{D}^\star(y)} D_{\text{KL}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \\ &+ \mathbb{E}_{x\sim\mathcal{D}_i(y)} D_{\text{KL}}(\mathcal{D}_i(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \end{aligned} \tag{B.16}$$

Then, we provide two bounded term for the last two KL divergence based terms,

$$\begin{aligned} &\mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{\text{KL}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \\ &\leq \mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{\text{KL}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) + \mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{\text{KL}}(\mathcal{D}_i(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \\ &= 2\mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{\text{JS}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{D}_i(\mathbf{x}|y)) \end{aligned}$$

Similarly, we could also have,

$$\begin{aligned} &\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\text{KL}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{M}(\mathbf{x}|y)) \\ &\qquad \leq 2\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\text{JS}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{D}_i(\mathbf{x}|y)) \end{aligned}$$

Plug these two terms into Eq B.16, we have

$$\begin{aligned} D_{\text{JS}}(\mathcal{D}^\star(\mathbf{x}, y) \| \mathcal{D}_i(\mathbf{x}, y)) \leq\ & D_{\text{JS}}(\mathcal{D}^\star(y) \| \mathcal{D}_i(y)) \\ &+ \mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{\text{JS}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{D}_i(\mathbf{x}|y)) \\ &+ \mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{\text{JS}}(\mathcal{D}^\star(\mathbf{x}|y) \| \mathcal{D}_i(\mathbf{x}|y)) \end{aligned} \tag{B.17}$$

Now we have

$$\begin{aligned} \sqrt{\text{R.H.S. of Eq. B.17}} \leq\ & \sqrt{D_{JS}(\mathcal{D}^\star(y)||\mathcal{D}_i(y))} \\ &+ \sqrt{\mathbb{E}_{y\sim\mathcal{D}^\star(y)} D_{JS}(\mathcal{D}^\star(\mathbf{x}|y)||\mathcal{D}_i(\mathbf{x}|y))} \\ &+ \sqrt{\mathbb{E}_{y\sim\mathcal{D}_i(y)} D_{JS}(\mathcal{D}^\star(\mathbf{x}|y)||\mathcal{D}_i(\mathbf{x}|y))} \end{aligned} \tag{B.18}$$

Plug Eq. B.18 into Eq. B.15, we conclude the proof. □

B.3.3 Proof to Corollary 6.2

Now we show the proof to Corollary 6.2.

Proof. First consider the risk in the testing phase, *i.e.*, the prediction loss on the target domain,

$$\begin{aligned} R_{\mathcal{D}_{\mathcal{T}}}(h) &= \frac{1}{K}\sum_{k=1}^{K}\int_{x}\mathcal{D}_{\mathcal{T}}(\mathbf{x}|Y=k)\mathcal{L}(h(\mathbf{x}),y) \\ &\leq \frac{1}{K}\sum_{k=1}\big[\mathbb{E}_{x\sim\mathcal{D}^{\star}}(\mathbf{x}|Y=k)\mathcal{L}(h(\mathbf{x}),y) \\ &+ d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_{\mathcal{T}}(\mathbf{x}|Y=k))\big] \end{aligned} \tag{B.19}$$

Similar with the proof of Theorem 6.1, we could bound the two items in Eq. B.19,

$$\begin{aligned} \mathbb{E}_{x\sim\mathcal{D}^{\star}}\mathcal{L}(h(\mathbf{x}),y) \leq& \mathbb{E}_{x\sim\mathcal{D}_1}\mathcal{L}(h(\mathbf{x}),y) \\ &+ d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_1(\mathbf{x}|Y=k)) \\ &\cdots \\ \mathbb{E}_{x\sim\mathcal{D}^{\star}}\mathcal{L}(h(\mathbf{x}),y) \leq& \mathbb{E}_{x\sim\mathcal{D}_i}\mathcal{L}(h(\mathbf{x}),y) \\ &+ d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_i(\mathbf{x}|Y=k)) \\ &\cdots \\ \mathbb{E}_{x\sim\mathcal{D}^{\star}}\mathcal{L}(h(\mathbf{x}),y) \leq& \mathbb{E}_{x\sim\mathcal{D}_m}\mathcal{L}(h(\mathbf{x}),y) \\ &+ d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_m(\mathbf{x}|Y=k)) \end{aligned} \tag{B.20}$$

Sum this Eq. B.20 and plug into Eq. B.19, we have,

$$\begin{aligned} R_{\mathcal{D}_{\mathcal{T}}}(h) \leq& \frac{1}{K}\sum_{k=1}^{K}[\frac{1}{m}\sum_{i=1}^{m}\mathbb{E}_{x\sim\mathcal{D}_i}\mathcal{L}(h) \\ &+\frac{1}{m}\sum_{i=1}^{m}d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_i(\mathbf{x}|Y=k)) \\ &+\frac{1}{m}\sum_{i=1}^{m}d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_{\mathcal{T}}(\mathbf{x})|Y=k)] \\ &\leq \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}_{x\sim\mathcal{D}_i}\hat{\mathcal{L}}^{\boldsymbol{\alpha}}_{\mathcal{D}_i}(h)+\kappa^{\star} \\ &+\frac{1}{K}\sum_{k=1}^{K}[\frac{1}{m}\sum_{i=1}^{m}d_{TV}(\mathcal{D}^{\star}(\mathbf{x}|Y=k),\mathcal{D}_i(\mathbf{x}|Y=k))] \end{aligned} \tag{B.21}$$

Then we could conclude the proof. □

Bibliographie

Charu C Aggarwal et al. Neural networks and deep learning. *Springer*, 10 :978–3, 2018.

Fabio Aiolli. Transfer learning by kernel meta-learning. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, volume 27 of *Proceedings of Machine Learning Research*, pages 81–95, Bellevue, Washington, USA, 02 Jul 2012. PMLR. URL https://proceedings.mlr.press/v27/aiolli12a.html.

Zaid Alyafeai, Maged Saeed AlShaibani, and Irfan Ahmad. A survey on transfer learning in natural language processing. *CoRR*, abs/2007.04239, 2020. URL https://arxiv.org/abs/2007.04239.

Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *J. Mach. Learn. Res.*, 6 :1817–1853, 2005. URL http://jmlr.org/papers/v6/ando05a.html.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 06–11 Aug 2017. URL http://proceedings.mlr.press/v70/arjovsky17a.html.

Martín Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *CoRR*, abs/1907.02893, 2019. URL http://arxiv.org/abs/1907.02893.

Yuri Sousa Aurelio, Gustavo Matheus de Almeida, Cristiano Leite de Castro, and Antonio Padua Braga. Learning from imbalanced data sets with weighted cross-entropy function. *Neural Processing Letters*, pages 1–13, 2019.

Kamyar Azizzadenesheli, Anqi Liu, Fanny Yang, and Animashree Anandkumar. Regularized learning for domain adaptation under label shifts. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=rJl0r3R9KX.

Zechen Bai, Zhigang Wang, Jian Wang, Di Hu, and Errui Ding. Unsupervised multi-source domain adaptation for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12914–12923, June 2021.

Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg : Towards domain generalization using meta-regularization. In *Advances in Neural Information Processing Systems 31 : Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1006–1016, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/647bba344396e7c8170902bcf2e15551-Abstract.html.

Jonathan Baxter. A model of inductive bias learning. *J. Artif. Intell. Res.*, 12 :149–198, 2000. doi : 10.1613/jair.731. URL https://doi.org/10.1613/jair.731.

Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Valdés, Amir Sadik, et al. Deepmind lab. *arXiv preprint arXiv :1612.03801*, 2016.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, pages 137–144. MIT Press, 2006. URL https://proceedings.neurips.cc/paper/2006/hash/b1b0432ceafb0ce714426e9114852ac7-Abstract.html.

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

Shai Ben-David, Tyler Lu, Teresa Luu, and Dávid Pál. Impossibility theorems for domain adaptation. In *International Conference on Artificial Intelligence and Statistics*, pages 129–136, 2010b.

Yoshua Bengio. *Learning deep architectures for AI*. Now Publishers Inc, 2009.

Hetal Bhavsar and Amit Ganatra. A comparative study of training algorithms for supervised machine learning. *International Journal of Soft Computing and Engineering (IJSCE)*, 2(4) : 2231–2307, 2012.

Joachim Bingel and Anders Søgaard. 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*, pages 164–169, Valencia, Spain, 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/E17-2026.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. In *Advances in Neural Information Processing Systems 24 : 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*,

pages 2178–2186, 2011. URL https://proceedings.neurips.cc/paper/2011/hash/b571ecea16a9824023ee1af16897a582-Abstract.html.

Gilles Blanchard, Aniket Anand Deshmukh, Ürün Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *J. Mach. Learn. Res.*, 22 :2–1, 2021.

François Bolley, Arnaud Guillin, and Cédric Villani. Quantitative concentration inequalities for empirical measures on non-compact spaces. *Probability Theory and Related Fields*, 137 (3-4) :541–593, 2007.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 95–104. IEEE Computer Society, 2017. doi : 10.1109/CVPR.2017.18. URL https://doi.org/10.1109/CVPR.2017.18.

Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 754–763. IEEE Computer Society, 2017. doi : 10.1109/ICCV.2017.88. URL https://doi.org/10.1109/ICCV.2017.88.

Wenming Cao, Si Wu, Zhiwen Yu, and Hau-San Wong. Exploring correlations among tasks, clusters, and features for multitask clustering. *IEEE transactions on neural networks and learning systems*, 30(2) :355–368, 2018a.

Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 135–150, 2018b.

Zhangjie Cao, Kaichao You, Mingsheng Long, Jianmin Wang, and Qiang Yang. Learning to transfer examples for partial domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2985–2994. Computer Vision Foundation / IEEE, 2019. doi : 10.1109/CVPR.2019.00310. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Cao_Learning_to_Transfer_Examples_for_Partial_Domain_Adaptation_CVPR_2019_paper.html.

Fabio Maria Carlucci, Lorenzo Porzi, Barbara Caputo, Elisa Ricci, and Samuel Rota Bulo. Autodial : Automatic domain alignment layers. In *2017 IEEE international conference on computer vision (ICCV)*, pages 5077–5085. IEEE, 2017.

Fabio Maria Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2229–2238. Computer Vision Foundation / IEEE, 2019. doi : 10.1109/CVPR.2019.00233. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Carlucci_Domain_Generalization_by_Solving_Jigsaw_Puzzles_CVPR_2019_paper.html.

Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. Homm : Higher-order moment matching for unsupervised domain adaptation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 3422–3429. AAAI Press, 2020a. URL https://aaai.org/ojs/index.php/AAAI/article/view/5745.

Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. Adversarial-learned loss for domain adaptation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04) :3521–3528, Apr. 2020b. doi : 10.1609/aaai.v34i04.5757. URL https://ojs.aaai.org/index.php/AAAI/article/view/5757.

Minmin Chen, Zhixiang Eddie Xu, Kilian Q. Weinberger, and Fei Sha. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012. URL http://icml.cc/2012/papers/416.pdf.

Shuaijun Chen, Xu Jia, Jianzhong He, Yongjie Shi, and Jianzhuang Liu. Semi-supervised domain adaptation based on dual-level domain mixing for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11018–11027, June 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020c.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm : Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning*, pages 794–803. PMLR, 2018.

S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546 vol. 1, 2005. doi : 10.1109/CVPR.2005.202.

David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15(2) :201–221, 1994.

Corinna Cortes, Mehryar Mohri, and Andrés Munoz Medina. Adaptation based on generalized discrepancy. *The Journal of Machine Learning Research*, 20(1) :1–30, 2019.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9) : 1853–1865, 2016.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 3730–3739, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/0070d23b06b1486a538c0eaa45dd167a-Abstract.html.

Aron Culotta and Andrew McCallum. Reducing labeling effort for structured prediction tasks. In *AAAI conference on artificial intelligence*, 2005.

Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. Boosting for transfer learning. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*, volume 227 of *ACM International Conference Proceeding Series*, pages 193–200. ACM, 2007. doi : 10.1145/1273496.1273521. URL https://doi.org/10.1145/1273496.1273521.

Sanjoy Dasgupta. Two faces of active learning. *Theoretical computer science*, 412(19) : 1767–1781, 2011.

Antoine de Mathelin, Mathilde Mougeot, and Nicolas Vayatis. Discrepancy-based active learning for domain adaptation. *arXiv preprint arXiv :2103.03757*, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet : A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. doi : 10.1109/CVPR.2009.5206848. URL https://doi.org/10.1109/CVPR.2009.5206848.

W. Deng, L. Zheng, Y. Sun, and J. Jiao. Rethinking triplet loss for domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2020.

Sofien Dhouib, Ievgen Redko, and Carole Lartizien. Margin-aware adversarial domain adaptation with optimal transport. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 2514–2524. PMLR, 2020. URL http://proceedings.mlr.press/v119/dhouib20b.html.

Zhengming Ding and Yun Fu. Deep domain generalization with structured low-rank constraint. *IEEE Transactions on Image Processing*, 27(1) :304–313, 2017.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf : A deep convolutional activation feature for generic visual recognition. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 647–655. JMLR.org, 2014. URL http://proceedings.mlr.press/v32/donahue14.html.

Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *Advances in Neural Information Processing Systems 32 : Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 6447–6458, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/2974788b53f73e7950e8aa49f3a306db-Abstract.html.

Yingjun Du, Jun Xu, Huan Xiong, Qiang Qiu, Xiantong Zhen, Cees GM Snoek, and Ling Shao. Learning to learn with variational information bottleneck for domain generalization. In *European Conference on Computer Vision*, pages 200–216. Springer, 2020.

Antonio D'Innocente and Barbara Caputo. Domain generalization with domain-specific aggregation modules. In *German Conference on Pattern Recognition*, pages 187–198. Springer, 2018.

Sarah Erfani, Mahsa Baktashmotlagh, Masud Moshtaghi, Xuan Nguyen, Christopher Leckie, James Bailey, and Rao Kotagiri. Robust domain generalisation by enforcing distribution invariance. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, pages 1455–1461. AAAI Press, 2016.

Zhen Fang, Jie Lu, Anjin Liu, Feng Liu, and Guangquan Zhang. Learning bounds for open-set learning. In *International Conference on Machine Learning*, pages 3122–3132. PMLR, 2021a.

Zhen Fang, Jie Lu, Feng Liu, Junyu Xuan, and Guangquan Zhang. Open set domain adaptation : Theoretical bound and algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, 32(10) :4309–4322, 2021b. doi : 10.1109/TNNLS.2020.3017213.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017. URL http://proceedings.mlr.press/v70/finn17a.html.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *Proceedings*

of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, pages 1563–1572. PMLR, 2018. URL http://proceedings.mlr.press/v80/franceschi18a.html.

Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1180–1189. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/ganin15.html.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1) :2096–2030, 2016.

Saurabh Garg, Yifan Wu, Sivaraman Balakrishnan, and Zachary C. Lipton. A unified view of label shift estimation. In *Advances in Neural Information Processing Systems 33 : Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/219e052492f4008818b8adb6366c7ed6-Abstract.html.

Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition : A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. A new pac-bayesian perspective on domain adaptation. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 859–868. JMLR.org, 2016. URL http://proceedings.mlr.press/v48/germain16.html.

Muhammad Ghifary, W. Bastiaan Kleijn, and Mengjie Zhang. Domain adaptive neural networks for object recognition. In *PRICAI 2014 : Trends in Artificial Intelligence - 13th Pacific Rim International Conference on Artificial Intelligence, Gold Coast, QLD, Australia, December 1-5, 2014. Proceedings*, volume 8862 of *Lecture Notes in Computer Science*, pages 898–904. Springer, 2014. doi : 10.1007/978-3-319-13560-1_76. URL https://doi.org/10.1007/978-3-319-13560-1_76.

Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2551–2559. IEEE Computer Society, 2015a. doi : 10.1109/ICCV.2015.293. URL https://doi.org/10.1109/ICCV.2015.293.

Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2551–2559. IEEE Computer Society, 2015b. doi : 10.1109/ICCV.2015.293. URL https://doi.org/10.1109/ICCV.2015.293.

Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016.

James Gibson, David Atkins, Torrey Creed, Zac Imel, Panayiotis Georgiou, and Shrikanth Narayanan. Multi-label multi-task deep learning for behavioral coding. *IEEE Transactions on Affective Computing*, 2019.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification : A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 513–520. Omnipress, 2011. URL https://icml.cc/2011/papers/342_icmlpaper.pdf.

Andrew Goldberg, Xiaojin Zhu, Aarti Singh, Zhiting Xu, and Robert Nowak. Multi-manifold semi-supervised learning. In *Artificial Intelligence and Statistics*, pages 169–176, 2009.

Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 2066–2073. IEEE Computer Society, 2012. doi : 10.1109/CVPR.2012.6247911. URL https://doi.org/10.1109/CVPR.2012.6247911.

Rui Gong, Yuhua Chen, Danda Pani Paudel, Yawei Li, Ajad Chhatkuli, Wen Li, Dengxin Dai, and Luc Van Gool. Cluster, split, fuse, and update : Meta-learning for open compound domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8344–8354, 2021.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25) :723–773, 2012. URL http://jmlr.org/papers/v13/gretton12a.html.

Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5767–5777, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/892c3b1c6dccd52936e27cbd0ff683d6-Abstract.html.

Leonid G Hanin. Kantorovich-rubinstein norm and its application in the theory of lipschitz spaces. *Proceedings of the American Mathematical Society*, 115(2) :345–352, 1992.

Mahmoud Hassaballah and Ali Ismail Awad. *Deep learning in computer vision : principles and applications*. CRC Press, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016a. doi : 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016b. doi : 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

Tao He, Junjie Hu, Ying Song, Jixiang Guo, and Zhang Yi. Multi-task learning for the segmentation of organs at risk with label dependence. *Medical image analysis*, 61 :101666, 2020.

Mark Herbster, Stephen Pasteris, and Lisa Tse. Online multitask learning with long-term memory. In *Advances in Neural Information Processing Systems 33 : Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/cdfa4c42f465a5a66871587c69fcfa34-Abstract.html.

Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild : Pixel-level adversarial and constraint-based adaptation. *CoRR*, abs/1612.02649, 2016. URL http://arxiv.org/abs/1612.02649.

Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada : Cycle-consistent adversarial domain adaptation. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 1994–2003. PMLR, 2018. URL http://proceedings.mlr.press/v80/hoffman18a.html.

Toke T Høye, Johanna Ärje, Kim Bjerge, Oskar LP Hansen, Alexandros Iosifidis, Florian Leese, Hjalte MR Mann, Kristian Meissner, Claus Melvad, and Jenni Raitoharju. Deep learning and computer vision will transform entomology. *Proceedings of the National Academy of Sciences*, 118(2), 2021.

Maximilian Ilse, Jakub M Tomczak, Christos Louizos, and Max Welling. Diva : Domain invariant variational autoencoders. In *Medical Imaging with Deep Learning*, pages 322–348. PMLR, 2020.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5967–5976. IEEE Computer Society, 2017. doi : 10.1109/CVPR.2017.632. URL https://doi.org/10.1109/CVPR.2017.632.

Arnesh Kumar Issar, Kirtan Mali, Aryan Mehta, Karan Uppal, Saurabh Mishra, and Debashish Chakravarty. Reproducibility of "fda : Fourier domain adaptation forsemantic segmentation. *CoRR*, abs/2104.14749, 2021. URL https://arxiv.org/abs/2104.14749.

Yunpei Jia, Jie Zhang, Shiguang Shan, and Xilin Chen. Single-side domain generalization for face anti-spoofing. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8481–8490. IEEE, 2020. doi : 10.1109/CVPR42600.2020.00851. URL https://doi.org/10.1109/CVPR42600.2020.00851.

Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 264–271, Prague, Czech Republic, 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P07-1034.

Junguang Jiang, Yifei Ji, Ximei Wang, Yufeng Liu, Jianmin Wang, and Mingsheng Long. Regressive domain adaptation for unsupervised keypoint detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6780–6789, June 2021.

Xin Jin, Fuzhen Zhuang, Hui Xiong, Changying Du, Ping Luo, and Qing He. Multi-task multi-view learning for heterogeneous tasks. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pages 441–450. ACM, 2014. doi : 10.1145/2661829.2662054. URL https://doi.org/10.1145/2661829.2662054.

Xin Jin, Cuiling Lan, Wenjun Zeng, Zhibo Chen, and Li Zhang. Style normalization and restitution for generalizable person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3143–3152, 2020.

Xin Jin, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Style normalization and restitution for domaingeneralization and adaptation. *arXiv preprint arXiv :2101.00588*, 2021.

Konstantinos Kamnitsas, Daniel Coelho de Castro, Loïc Le Folgoc, Ian Walker, Ryutaro Tanno, Daniel Rueckert, Ben Glocker, Antonio Criminisi, and Aditya V. Nori. Semi-supervised learning via compact latent space clustering. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2464–2473. PMLR, 2018. URL http://proceedings.mlr.press/v80/kamnitsas18a.html.

Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G. Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4893–4902. Computer Vision Foundation / IEEE, 2019. doi : 10.1109/CVPR.2019.00503. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Kang_Contrastive_Adaptation_Network_for_Unsupervised_Domain_Adaptation_CVPR_2019_paper.html.

Yoshiyuki Kawano and Keiji Yanai. Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In *European Conference on Computer Vision*, pages 3–17. Springer, 2014.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7482–7491. IEEE Computer Society, 2018. doi : 10.1109/CVPR.2018.00781. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Kendall_Multi-Task_Learning_Using_CVPR_2018_paper.html.

Rawal Khirodkar, Donghyun Yoo, and Kris M. Kitani. Domain randomization for scene-specific car detection and pose estimation. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019*, pages 1932–1940. IEEE, 2019. ISBN 978-1-7281-1975-5. doi : 10.1109/WACV.2019.00210. URL https://doi.org/10.1109/WACV.2019.00210.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33 : Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html.

Diederik P. Kingma and Jimmy Ba. Adam : A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25 : 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114, 2012. URL https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

Seiichi Kuroki, Nontawat Charoenphakdee, Han Bao, Junya Honda, Issei Sato, and Masashi Sugiyama. Unsupervised domain adaptation based on source-guided discrepancy. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 4122–4129. AAAI Press, 2019. doi : 10.1609/aaai.v33i01.33014122. URL https://doi.org/10.1609/aaai.v33i01.33014122.

Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.

Bo Li, Yezhen Wang, Shanghang Zhang, Dongsheng Li, Kurt Keutzer, Trevor Darrell, and Han Zhao. Learning invariant representations and risks for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1104–1113, June 2021a.

Bo Li, Yezhen Wang, Shanghang Zhang, Dongsheng Li, Kurt Keutzer, Trevor Darrell, and Han Zhao. Learning invariant representations and risks for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1104–1113, 2021b.

Da Li and Timothy Hospedales. Online meta-learning for multi-source and semi-supervised domain adaptation. *arXiv preprint arXiv :2004.04398*, 2020.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5543–5551. IEEE Computer Society, 2017a. doi : 10.1109/ICCV.2017.591. URL https://doi.org/10.1109/ICCV.2017.591.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5543–5551. IEEE Computer Society, 2017b. doi : 10.1109/ICCV.2017.591. URL https://doi.org/10.1109/ICCV.2017.591.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize : Meta-learning for domain generalization. In *Proceedings of the Thirty-Second AAAI Conference on*

Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pages 3490–3497. AAAI Press, 2018a. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16067.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize : Meta-learning for domain generalization. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. AAAI Press, 2018b. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16067.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot. Domain generalization with adversarial feature learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5400–5409. IEEE Computer Society, 2018c. doi : 10.1109/CVPR.2018.00566. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Li_Domain_Generalization_With_CVPR_2018_paper.html.

Jingjing Li, Jidong Zhao, and Ke Lu. Joint feature selection and structure preservation for domain adaptation. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 1697–1703. IJCAI/AAAI Press, 2016a. URL http://www.ijcai.org/Abstract/16/243.

Shuang Li, Shiji Song, and Gao Huang. Prediction reweighting for domain adaptation. *IEEE transactions on neural networks and learning systems*, 28(7) :1682–1695, 2016b.

Sijin Li, Zhi-Qiang Liu, and Antoni B Chan. 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*, pages 482–489, 2014.

Ya Li, Mingming Gong, Xinmei Tian, Tongliang Liu, and Dacheng Tao. Domain generalization via conditional invariant representations. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 3579–3587. AAAI Press, 2018d. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16595.

Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XV*, volume 11219 of *Lecture Notes in Computer Science*, pages 647–663.

Springer, 2018e. doi : 10.1007/978-3-030-01267-0_38. URL https://doi.org/10.1007/978-3-030-01267-0_38.

Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639, 2018f.

Yanghao Li, Naiyan Wang, Jianping Shi, Xiaodi Hou, and Jiaying Liu. Adaptive batch normalization for practical domain adaptation. *Pattern Recognit.*, 80 :109–117, 2018g. doi : 10.1016/j.patcog.2018.03.005. URL https://doi.org/10.1016/j.patcog.2018.03.005.

Yitong Li, Michael Murias, Geraldine Dawson, and David E. Carlson. Extracting relationships by multi-domain matching. In *Advances in Neural Information Processing Systems 31 : Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6799–6810, 2018h. URL https://proceedings.neurips.cc/paper/2018/hash/2fd0fd3efa7c4cfb034317b21f3c2d93-Abstract.html.

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

Zachary C. Lipton, Yu-Xiang Wang, and Alexander J. Smola. Detecting and correcting for label shift with black box predictors. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3128–3136. PMLR, 2018. URL http://proceedings.mlr.press/v80/lipton18a.html.

Hong Liu, Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Qiang Yang. Separate to adapt : Open set domain adaptation via progressive separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2927–2936, 2019a.

Hong Liu, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Transferable adversarial training : A general approach to adapting deep classifiers. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 4013–4022. PMLR, 2019b. URL http://proceedings.mlr.press/v97/liu19b.html.

Jiaying Liu, Yanghao Li, Sijie Song, Junliang Xing, Cuiling Lan, and Wenjun Zeng. Multi-modality multi-task recurrent neural network for online action detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9) :2667–2682, 2018.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1–10, Vancouver, Canada, 2017a. Association

for Computational Linguistics. doi : 10.18653/v1/P17-1001. URL https://www.aclweb.org/anthology/P17-1001.

Quande Liu, Cheng Chen, Jing Qin, Qi Dou, and Pheng-Ann Heng. Feddg : Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Shuying Liu and Weihong Deng. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 730–734, 2015. doi : 10.1109/ACPR.2015.7486599.

Tongliang Liu, Dacheng Tao, Mingli Song, and Stephen J. Maybank. Algorithm-dependent generalization bounds for multi-task learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(2) :227–241, 2017b. doi : 10.1109/TPAMI.2016.2544314. URL https://doi.org/10.1109/TPAMI.2016.2544314.

Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S. Yu. Transfer joint matching for unsupervised domain adaptation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 1410–1417. IEEE Computer Society, 2014. doi : 10.1109/CVPR.2014.183. URL https://doi.org/10.1109/CVPR.2014.183.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 97–105. JMLR.org, 2015a. URL http://proceedings.mlr.press/v37/long15.html.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 97–105. JMLR.org, 2015b. URL http://proceedings.mlr.press/v37/long15.html.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Philip S. Yu. Learning multiple tasks with multilinear relationship networks. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1594–1603, 2017a. URL https://proceedings.neurips.cc/paper/2017/hash/03e0704b5690a2dee1861dc3ad3316c9-Abstract.html.

Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I. Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on*

Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, volume 70 of *Proceedings of Machine Learning Research*, pages 2208–2217. PMLR, 2017b. URL http://proceedings.mlr.press/v70/long17a.html.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems 31 : Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1647–1657, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/ab88b15733f543179858600245108dd8-Abstract.html.

Pinxin Long, Wenxi Liu, and Jia Pan. Deep-learned collision avoidance policy for distributed multiagent navigation. *IEEE Robotics and Automation Letters*, 2(2) :656–663, 2017c.

Yong Luo, Dacheng Tao, and Yonggang Wen. Exploiting high-order information in heterogeneous multi-task feature learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 2443–2449. ijcai.org, 2017a. doi : 10.24963/ijcai.2017/340. URL https://doi.org/10.24963/ijcai.2017/340.

Zelun Luo, Yuliang Zou, Judy Hoffman, and Fei-Fei Li. Label efficient learning of transferable representations acrosss domains and tasks. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 165–177, 2017b. URL https://proceedings.neurips.cc/paper/2017/hash/a8baa56554f96369ab93e4f3bb068c22-Abstract.html.

Ying Ma, Guangchun Luo, Xue Zeng, and Aiguo Chen. Transfer learning for cross-company software defect prediction. *Information and Software Technology*, 54(3) :248–256, 2012.

Massimiliano Mancini, Hakan Karaoguz, Elisa Ricci, Patric Jensfelt, and Barbara Caputo. Kitting in the wild through online domain adaptation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1103–1109. IEEE, 2018.

R. Manmatha, Chao-Yuan Wu, Alexander J. Smola, and Philipp Krähenbühl. Sampling matters in deep embedding learning. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2859–2867. IEEE Computer Society, 2017. doi : 10.1109/ICCV.2017.309. URL https://doi.org/10.1109/ICCV.2017.309.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation : Learning bounds and algorithms. In *COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009*, 2009a. URL http://www.cs.mcgill.ca/~colt2009/papers/003.pdf#page=1.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation : Learning bounds and algorithms. In *COLT 2009 - The 22nd Conference on Learning Theory,*

Montreal, Quebec, Canada, June 18-21, 2009, 2009b. URL http://www.cs.mcgill.ca/~colt2009/papers/003.pdf#page=1.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Multiple source adaptation and the rényi divergence. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 367–374. AUAI Press, 2009c.

Yuren Mao, Weiwei Liu, and Xuemin Lin. Adaptive adversarial multi-task representation learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6724–6733. PMLR, 2020. URL http://proceedings.mlr.press/v119/mao20a.html.

Gary Marcus. Deep learning : A critical appraisal. *arXiv preprint arXiv :1801.00631*, 2018.

Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11749–11756, 2020.

Andreas Maurer. Bounds for linear multi-task learning. *Journal of Machine Learning Research*, 7 :117–139, 2006.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. Sparse coding for multitask and transfer learning. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 343–351. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/maurer13.html.

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

Mohri Mehryar, Rostamizadeh Afshin, and Ameet Talwalkar. *Foundations of Machine Learning (Second Edition)*. MIT Press, Cambridge, Massachusetts, 2018.

Pim Moeskops, Jelmer M Wolterink, Bas HM van der Velden, Kenneth GA Gilhuijs, Tim Leiner, Max A Viergever, and Ivana Ivsgum. Deep learning for multi-task medical image segmentation in multiple modalities. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 478–486. Springer, 2016.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences de Paris*, 1781.

Saeid Motiian, Quinn Jones, Seyed Mehdi Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6670–6680, 2017a. URL https://proceedings.neurips.cc/paper/2017/hash/21c5bba1dd6aed9ab48c2b34c1a0adde-Abstract.html.

Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5716–5726. IEEE Computer Society, 2017b. doi : 10.1109/ICCV.2017.609. URL https://doi.org/10.1109/ICCV.2017.609.

Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5716–5726. IEEE Computer Society, 2017c. doi : 10.1109/ICCV.2017.609. URL https://doi.org/10.1109/ICCV.2017.609.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 10–18. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/muandet13.html.

Keerthiram Murugesan and Jaime G. Carbonell. Active learning from peers. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 7008–7017, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/b87470782489389f344c4fa4ceb5260c-Abstract.html.

Keerthiram Murugesan, Hanxiao Liu, Jaime G. Carbonell, and Yiming Yang. Adaptive smoothed online multi-task learning. In *Advances in Neural Information Processing Systems 29 : Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4296–4304, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/a869ccbcbd9568808b8497e28275c7c8-Abstract.html.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Duy-Kien Nguyen and Takayuki Okatani. Multi-task learning of hierarchical vision-language representation. In *IEEE Conference on Computer Vision and Pattern Recognition,*

CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, pages 10492–10501. Computer Vision Foundation / IEEE, 2019. doi : 10.1109/CVPR.2019.01074. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Nguyen_Multi-Task_Learning_of_Hierarchical_Vision-Language_Representation_CVPR_2019_paper.html.

Shuteng Niu, Meryl Liu, Yongxin Liu, Jian Wang, and Houbing Song. Distant domain transfer learning for medical imaging. *CoRR*, abs/2012.06346, 2020. URL https://arxiv.org/abs/2012.06346.

Daniel W Otter, Julian R Medina, and Jugal K Kalita. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2) :604–624, 2020.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10) :1345–1359, 2009.

Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2) :199–210, 2010.

German Ignacio Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks : A review. *Neural Networks*, 113 :54–71, 2019. doi : 10.1016/j.neunet.2019.01.012. URL https://doi.org/10.1016/j.neunet.2019.01.012.

Seong-Jin Park, Seungju Han, Ji-Won Baek, Insoo Kim, Juhwan Song, Haebeom Lee, Jae-Joon Han, and Sung Ju Hwang. Meta variance transfer : Learning to augment from the others. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 7510–7520. PMLR, 2020. URL http://proceedings.mlr.press/v119/park20b.html.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch : An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32 : Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*,

pages 1406–1415. IEEE, 2019a. doi : 10.1109/ICCV.2019.00149. URL https://doi.org/10.1109/ICCV.2019.00149.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 1406–1415. IEEE, 2019b. doi : 10.1109/ICCV.2019.00149. URL https://doi.org/10.1109/ICCV.2019.00149.

Anastasia Pentina and Christoph H. Lampert. Multi-task learning with labeled and unlabeled tasks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 2807–2816. PMLR, 2017. URL http://proceedings.mlr.press/v70/pentina17a.html.

Claudio Persello and Lorenzo Bruzzone. Active learning for domain adaptation in the supervised classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 50(11) :4468–4483, 2012.

Viraj Prabhu, Arjun Chandrasekaran, Kate Saenko, and Judy Hoffman. Active domain adaptation via clustering uncertainty-weighted embeddings. *CoRR*, abs/2010.08666, 2020. URL https://arxiv.org/abs/2010.08666.

Yunchen Pu, Weiyao Wang, Ricardo Henao, Liqun Chen, Zhe Gan, Chunyuan Li, and Lawrence Carin. Adversarial symmetric variational autoencoder. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4330–4339, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/4cb811134b9d39fc3104bd06ce75abad-Abstract.html.

Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12553–12562. IEEE, 2020. doi : 10.1109/CVPR42600.2020.01257. URL https://doi.org/10.1109/CVPR42600.2020.01257.

Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. Multi-component image translation for deep domain generalization. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019*, pages 579–588. IEEE, 2019. doi : 10.1109/WACV.2019.00067. URL https://doi.org/10.1109/WACV.2019.00067.

Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. Correlation-aware adversarial domain adaptation and generalization. *Pattern Recognition*, 100 :107124, 2020.

Ievgen Redko, Amaury Habrard, and Marc Sebban. Theoretical analysis of domain adaptation with optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 737–753. Springer, 2017.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. *Advances in Domain Adaptation Theory*. Elsevier, 2019.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *CoRR*, abs/2009.00236, 2020. URL https://arxiv.org/abs/2009.00236.

Elan Rosenfeld, Pradeep Kumar Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=BbNIbVPJ-42.

Paul Ruvolo and Eric Eaton. ELLA : an efficient lifelong learning algorithm. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 507–515. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/ruvolo13.html.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

Doyen Sahoo, Hung Le, Chenghao Liu, and Steven CH Hoi. Meta-learning with domain adaptation for few-shot learning under domain shift. *Openreview.net*, 2018.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3723–3732. IEEE Computer Society, 2018. doi : 10.1109/CVPR.2018.00392. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Saito_Maximum_Classifier_Discrepancy_CVPR_2018_paper.html.

Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11) :5973–6006, 2016.

Tobias Scheffer and Stefan Wrobel. Active learning of partially hidden markov models. In *In Proceedings of the ECML/PKDD Workshop on Instance Selection*. Citeseer, 2001.

Robin M Schmidt. Explainability-aided domain generalization for image classification. *arXiv preprint arXiv :2104.01742*, 2021.

Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *International conference on artificial neural networks*, pages 583–588. Springer, 1997.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems 31 : Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 525–536, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/432aca3a1e345e339f35a30c8f65edce-Abstract.html.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks : A core-set approach. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=H1aIuk-RW.

Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1) :1–114, 2012.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning : From theory to algorithms*. Cambridge university press, 2014.

Hossein Sharifi-Noghabi, Hossein Asghari, Nazanin Mehrasa, and Martin Ester. Domain generalization via semi-supervised meta learning. *arXiv preprint arXiv :2009.12658*, 2020.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4058–4065. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17155.

Yang Shu, Zhangjie Cao, Chenyu Wang, Jianmin Wang, and Mingsheng Long. Open domain generalization with domain-augmented meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9624–9633, 2021.

Changjian Shui, Mahdieh Abbasi, Louis-Émile Robitaille, Boyu Wang, and Christian Gagné. A principled approach for learning task similarity in multitask learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 3446–3452. ijcai.org, 2019. doi : 10.24963/ijcai.2019/478. URL https://doi.org/10.24963/ijcai.2019/478.

Changjian Shui, Qi Chen, Jun Wen, Fan Zhou, Christian Gagné, and Boyu Wang. Beyond h-divergence : Domain adaptation theory with jensen-shannon divergence. *CoRR*, abs/2007.15567, 2020a. URL https://arxiv.org/abs/2007.15567.

Changjian Shui, Fan Zhou, Christian Gagné, and Boyu Wang. Deep active learning : Unified and principled method for query and training. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 1308–1318. PMLR, 2020b. URL http://proceedings.mlr.press/v108/shui20a.html.

Anthony Sicilia, Xingchen Zhao, and Seong Jae Hwang. Domain adversarial neural networks for domain generalization : When it works and how to improve. *arXiv preprint arXiv :2102.03924*, 2021.

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 5971–5980. IEEE, 2019. doi : 10.1109/ICCV.2019.00607. URL https://doi.org/10.1109/ICCV.2019.00607.

Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho. Meta-transfer learning for zero-shot super-resolution. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3513–3522. IEEE, 2020. doi : 10.1109/CVPR42600.2020.00357. URL https://doi.org/10.1109/CVPR42600.2020.00357.

Elnaz Soleimani and Ehsan Nazerfard. Cross-subject transfer learning in human activity recognition systems using generative adversarial networks. *Neurocomputing*, 426 :26–34, 2021. doi : 10.1016/j.neucom.2020.10.056. URL https://doi.org/10.1016/j.neucom.2020.10.056.

Xiaolin Song, Sicheng Zhao, Jingyu Yang, Huanjing Yue, Pengfei Xu, Runbo Hu, and Hua Chai. Spatio-temporal contrastive domain adaptation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9787–9795, June 2021a.

Xiaolin Song, Sicheng Zhao, Jingyu Yang, Huanjing Yue, Pengfei Xu, Runbo Hu, and Hua Chai. Spatio-temporal contrastive domain adaptation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9787–9795, 2021b.

Bharath K. Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, and Gert R. G. Lanckriet. On integral probability metrics,ϕ-divergences and binary classification, 2009.

Serban Stan and Mohammad Rostami. Privacy preserving domain adaptation for semantic segmentation of medical images. *arXiv preprint arXiv :2101.00522*, 2021.

Trevor Standley, Amir Roshan Zamir, Dawn Chen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning ? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9120–9132. PMLR, 2020. URL http://proceedings.mlr.press/v119/standley20a.html.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for modern deep learning research. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(09) :13693–13696, Apr. 2020. doi : 10.1609/aaai.v34i09.7123. URL https://ojs.aaai.org/index.php/AAAI/article/view/7123.

Jong-Chyi Su, Yi-Hsuan Tsai, Kihyuk Sohn, Buyu Liu, Subhransu Maji, and Manmohan Chandraker. Active adversarial domain adaptation. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 739–748, 2020.

Baochen Sun, Jiashi Feng, and Kate Saenko. Correlation alignment for unsupervised domain adaptation. In *Domain Adaptation in Computer Vision Applications*, Advances in Computer Vision and Pattern Recognition, pages 153–171. Springer, 2017. doi : 10.1007/978-3-319-58347-1_8. URL https://doi.org/10.1007/978-3-319-58347-1_8.

Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 403–412. Computer Vision Foundation / IEEE, 2019. doi : 10.1109/CVPR.2019.00049. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Sun_Meta-Transfer_Learning_for_Few-Shot_Learning_CVPR_2019_paper.html.

Qianru Sun, Yaoyao Liu, Zhaozheng Chen, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning through hard tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

Yee Whye Teh, Victor Bapst, Wojciech M. Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral : Robust multitask reinforcement learning. In *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4496–4506, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/0abdc563a06105aee3c6136871c9f4d1-Abstract.html.

Mamatha Thota and Georgios Leontidis. Contrastive domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2209–2218, June 2021.

Kim-Han Thung and Chong-Yaw Wee. A brief review on multi-task learning. *Multimedia Tools and Applications*, 77(22) :29705–29725, 2018.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017.

Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 1521–1528. IEEE Computer Society, 2011. doi : 10.1109/CVPR.2011.5995347. URL https://doi.org/10.1109/CVPR.2011.5995347.

Lisa Torrey and Jude Shavlik. Transfer learning. In *Handbook of research on machine learning applications and trends : algorithms, methods, and techniques*, pages 242–264. IGI Global, 2010.

Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion : Maximizing for domain invariance. *CoRR*, abs/1412.3474, 2014. URL http://arxiv.org/abs/1412.3474.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2962–2971. IEEE Computer Society, 2017. doi : 10.1109/CVPR.2017.316. URL https://doi.org/10.1109/CVPR.2017.316.

Evgeniya Ustinova and Victor S. Lempitsky. Learning deep embeddings with histogram loss. In *Advances in Neural Information Processing Systems 29 : Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4170–4178, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/325995af77a0e8b06d1204a171010b3a-Abstract.html.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5385–5394. IEEE Computer Society, 2017. doi : 10.1109/CVPR.2017.572. URL https://doi.org/10.1109/CVPR.2017.572.

Cédric Villani. The wasserstein distances. In *Optimal Transport*, pages 93–111. Springer, 2009.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C. Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems 31 : Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 5339–5349, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/1d94108e907bb8311d8802b48fd54b4a-Abstract.html.

Riccardo Volpi, Diane Larlus, and Grégory Rogez. Continual adaptation of visual representations via domain randomization and meta-learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 4443–4453. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Volpi_Continual_Adaptation_of_Visual_Representations_via_Domain_Randomization_and_Meta-Learning_CVPR_2021_paper.html.

Martin J Wainwright. *High-dimensional statistics : A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Neng Wan, Dapeng Li, and Naira Hovakimyan. f-divergence variational inference. *Advances in Neural Information Processing Systems*, 33, 2020.

Boyu Wang and Joelle Pineau. Online boosting algorithms for anytime transfer and multitask learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 3038–3044. AAAI Press, 2015. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9740.

Boyu Wang, Joelle Pineau, and Borja Balle. Multitask generalized eigenvalue program. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2115–2121. AAAI Press, 2016. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11999.

Boyu Wang, Jorge A. Mendez, Mingbo Cai, and Eric Eaton. Transfer learning via minimizing the performance gap between domains. In *Advances in Neural Information Processing Systems 32 : Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 10644–10654, 2019a. URL https://proceedings.neurips.cc/paper/2019/hash/c66dd00e5fc44ba8de89d7713fedcd50-Abstract.html.

Boyu Wang, Hejia Zhang, Peng Liu, Zebang Shen, and Joelle Pineau. Multitask metric learning : Theory and algorithm. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 3362–3371. PMLR, 2019b. URL http://proceedings.mlr.press/v89/wang19f.html.

Boyu Wang, Chi Man Wong, Zhao Kang, Feng Liu, Changjian Shui, Feng Wan, and CL Philip Chen. Common spatial pattern reformulated for regularizations in brain-computer interfaces. *IEEE Trans. Cybern.*, 2020a. doi : 10.1109/TCYB.2020.2982901.

Jian Wang, Hengde Zhu, Shui-Hua Wang, and Yu-Dong Zhang. A review of deep learning on medical image analysis. *Mobile Networks and Applications*, 26(1) :351–380, 2021a.

Jindong Wang, Wenjie Feng, Yiqiang Chen, Han Yu, Meiyu Huang, and Philip S. Yu. Visual domain adaptation with manifold embedded distribution alignment. In *2018 ACM Multimedia Conference on Multimedia Conference, MM 2018, Seoul, Republic of Korea, October 22-26, 2018*, pages 402–410, 2018. doi : 10.1145/3240508.3240512. URL https://doi.org/10.1145/3240508.3240512.

Mei Wang and Weihong Deng. Cycle label-consistent networks for unsupervised domain adaptation. *Neurocomputing*, 422 :186–199, 2021.

Rui Wang, Zuxuan Wu, Zejia Weng, Jingjing Chen, Guo-Jun Qi, and Yu-Gang Jiang. Cross-domain contrastive learning for unsupervised domain adaptation. *CoRR*, abs/2106.05528, 2021b. URL https://arxiv.org/abs/2106.05528.

Wei Wang, Haojie Li, Zhengming Ding, Feiping Nie, Junyang Chen, Xiao Dong, and Zhihui Wang. Rethinking maximum mean discrepancy for visual domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 2021c.

Xuezhi Wang, Tzu-Kuo Huang, and Jeff G. Schneider. Active transfer learning under model shift. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1305–1313. JMLR.org, 2014. URL http://proceedings.mlr.press/v32/wangi14.html.

Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R. Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 5022–5030. Computer Vision Foundation / IEEE, 2019c. doi : 10.1109/CVPR.2019.00516. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Wang_Multi-Similarity_Loss_With_General_Pair_Weighting_for_Deep_Metric_Learning_CVPR_2019_paper.html.

Yufei Wang, Haoliang Li, and Alex C Kot. Heterogeneous domain generalization via domain mixup. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3622–3626. IEEE, 2020b.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1) :9, 2016.

Jun Wen, Risheng Liu, Nenggan Zheng, Qian Zheng, Zhefeng Gong, and Junsong Yuan. Exploiting local feature patterns for unsupervised domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5401–5408, 2019a.

Jun Wen, Nenggan Zheng, Junsong Yuan, Zhefeng Gong, and Changyou Chen. Bayesian uncertainty matching for unsupervised domain adaptation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 3849–3855. ijcai.org, 2019b. doi : 10.24963/ijcai.2019/534. URL https://doi.org/10.24963/ijcai.2019/534.

Junfeng Wen, Russell Greiner, and Dale Schuurmans. Domain aggregation networks for multi-source domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 10214–10224. PMLR, 2020. URL http://proceedings.mlr.press/v119/wen20b.html.

Sen Wu, Hongyang R Zhang, and Christopher Ré. Understanding and improving information transfer in multi-task learning. *arXiv preprint arXiv :2005.00944*, 2020.

Yifan Wu, Ezra Winston, Divyansh Kaushik, and Zachary C. Lipton. Domain adaptation with asymmetrically-relaxed distribution alignment. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6872–6881. PMLR, 2019. URL http://proceedings.mlr.press/v97/wu19f.html.

Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco : Unsupervised contrastive learning for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8392–8401, 2021.

Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5419–5428. PMLR, 2018. URL http://proceedings.mlr.press/v80/xie18c.html.

Renjun Xu, Pelen Liu, Liyan Wang, Chao Chen, and Jindong Wang. Reliable weighted optimal transport for unsupervised domain adaptation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 4393–4402. IEEE, 2020. doi : 10.1109/CVPR42600.2020.00445. URL https://doi.org/10.1109/CVPR42600.2020.00445.

Ruijia Xu, Ziliang Chen, Wangmeng Zuo, Junjie Yan, and Liang Lin. Deep cocktail network : Multi-source unsupervised domain adaptation with category shift. In *2018 IEEE Conference*

on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, pages 3964–3973. IEEE Computer Society, 2018. doi : 10.1109/CVPR.2018.00417. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Xu_Deep_Cocktail_Network_CVPR_2018_paper.html.

Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *arXiv preprint arXiv :2001.00677*, 2020.

Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d : Self-training for unsupervised domain adaptation on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10368–10378, 2021.

Pei Yang and Wei Gao. Multi-view discriminant transfer learning. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, pages 1848–1854. IJCAI/AAAI, 2013. URL http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6545.

Yanchao Yang and Stefano Soatto. FDA : fourier domain adaptation for semantic segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 4084–4094. Computer Vision Foundation / IEEE, 2020. doi : 10.1109/CVPR42600.2020.00414. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Yang_FDA_Fourier_Domain_Adaptation_for_Semantic_Segmentation_CVPR_2020_paper.html.

Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Deep metric learning for person re-identification. In *2014 22nd International Conference on Pattern Recognition*, pages 34–39. IEEE, 2014.

Li Yi, Boqing Gong, and Thomas Funkhouser. Complete & label : A domain adaptation approach to semantic segmentation of lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15363–15373, June 2021.

Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan : Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857, 2017.

Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Universal domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2720–2729. Computer Vision Foundation / IEEE, 2019. doi : 10.1109/CVPR.2019.00283. URL http://openaccess.thecvf.com/content_CVPR_2019/html/You_Universal_Domain_Adaptation_CVPR_2019_paper.html.

Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13 (3) :55–75, 2018.

Xiangyu Yue, Zangwei Zheng, Shanghang Zhang, Yang Gao, Trevor Darrell, Kurt Keutzer, and Alberto Sangiovanni Vincentelli. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13834–13844, June 2021.

Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 819–827. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/zhang13d.html.

Weichen Zhang, Wen Li, and Dong Xu. Srdan : Scale-aware and range-aware domain adaptation network for cross-dataset 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6769–6779, June 2021.

Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 5031–5040. Computer Vision Foundation / IEEE, 2019a. doi : 10.1109/CVPR.2019.00517. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Zhang_Domain-Symmetric_Networks_for_Adversarial_Domain_Adaptation_CVPR_2019_paper.html.

Yiling Zhang, Yan Yang, Tianrui Li, and Hamido Fujita. A multitask multiview clustering algorithm in heterogeneous situations based on lle and le. *Knowledge-Based Systems*, 163 : 776–786, 2019b.

Yu Zhang. Multi-task learning and algorithmic stability. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 3181–3187. AAAI Press, 2015. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9411.

Yu Zhang and Dit-Yan Yeung. A convex formulation for learning task relationships in multi-task learning. In *UAI 2010, Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, July 8-11, 2010*, pages 733–442. AUAI Press, 2010. URL https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=2117&proceeding_id=26.

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael I. Jordan. Bridging theory and algorithm for domain adaptation. In *Proceedings of the 36th International Conference on*

Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of *Proceedings of Machine Learning Research*, pages 7404–7413. PMLR, 2019c. URL http://proceedings.mlr.press/v97/zhang19i.html.

Yun Zhang, Nianbin Wang, Shaobin Cai, and Lei Song. Unsupervised domain adaptation by mapped correlation alignment. *IEEE Access*, 6 :44698–44706, 2018. doi : 10.1109/ACCESS.2018.2865249. URL https://doi.org/10.1109/ACCESS.2018.2865249.

Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J. Gordon. On learning invariant representations for domain adaptation. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7523–7532. PMLR, 2019a. URL http://proceedings.mlr.press/v97/zhao19a.html.

Shanshan Zhao, Mingming Gong, Tongliang Liu, Huan Fu, and Dacheng Tao. Domain generalization via entropy regularization. *Advances in Neural Information Processing Systems*, 33, 2020a.

Sicheng Zhao, Bo Li, Xiangyu Yue, Yang Gu, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source domain adaptation for semantic segmentation. In *Advances in Neural Information Processing Systems 32 : Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7285–7298, 2019b. URL https://proceedings.neurips.cc/paper/2019/hash/db9ad56c71619aeed9723314d1456037-Abstract.html.

Sicheng Zhao, Xiangyu Yue, Shanghang Zhang, Bo Li, Han Zhao, Bichen Wu, Ravi Krishna, Joseph E Gonzalez, Alberto L Sangiovanni-Vincentelli, Sanjit A Seshia, et al. A review of single-source deep unsupervised visual domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 2020b.

Xiangyun Zhao, Haoxiang Li, Xiaohui Shen, Xiaodan Liang, and Ying Wu. A modulation module for multi-task learning with applications in image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 401–416, 2018.

Dawei Zhou, Lecheng Zheng, Yada Zhu, Jianbo Li, and Jingrui He. Domain adaptive multi-modality neural attention network for financial forecasting. In *WWW '20 : The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2230–2240. ACM / IW3C2, 2020a. doi : 10.1145/3366423.3380288. URL https://doi.org/10.1145/3366423.3380288.

Di Zhou, Jun Wang, Bin Jiang, Hua Guo, and Yajun Li. Multi-task multi-view learning based on cooperative multi-objective optimization. *IEEE Access*, 6 :19465–19477, 2017.

Fan Zhou, Brahim Chaib-draa, and Boyu Wang. Multi-task learning by leveraging the semantic information. *Thirty-Fifth AAAI Conference on Artificial Intelligence*, 2021a.

Fan Zhou, Zhuqing Jiang, Changjian Shui, Boyu Wang, and Brahim Chaib-draa. Domain generalization via optimal transport with metric similarity learning. *Neurocomputing*, 456 : 469–480, 2021b. ISSN 0925-2312. doi : https://doi.org/10.1016/j.neucom.2020.09.091. URL https://www.sciencedirect.com/science/article/pii/S0925231221002009.

Fan Zhou, Changjian Shui, Shichun Yang, Bincheng Huang, Boyu Wang, and Brahim Chaib-draa. Discriminative active learning for domain adaptation. *Knowledge-based Systems*, 2021c.

Fan Zhou, Shichun Yang, Boyu Wang, and Brahim Chaib-draa. On the value of label and semantic information in domain generalization. *IEEE Transcations on Neural Networks and Learning Systems, under review*, 2021d.

Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European Conference on Computer Vision*, pages 561–578. Springer, 2020b.

Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 13025–13032. AAAI Press, 2020c. URL https://aaai.org/ojs/index.php/AAAI/article/view/7003.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2242–2251. IEEE Computer Society, 2017. doi : 10.1109/ICCV.2017.244. URL https://doi.org/10.1109/ICCV.2017.244.

Fuzhen Zhuang, Xiaohu Cheng, Ping Luo, Sinno Jialin Pan, and Qing He. Supervised representation learning : Transfer learning with deep autoencoders. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

Fuzhen Zhuang, Xuebing Li, Xin Jin, Dapeng Zhang, Lirong Qiu, and Qing He. Semantic feature learning for heterogeneous multitask classification via non-negative matrix factorization. *IEEE transactions on cybernetics*, 48(8) :2284–2293, 2017.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109 (1) :43–76, 2020.

Danbing Zou, Qikui Zhu, and Pingkun Yan. Unsupervised domain adaptation with dual-scheme fusion network for medical image segmentation. In *IJCAI*, pages 3291–3298, 2020.