Principled Deep Learning Approaches for Learning from Limited Labeled Data through Distribution Matching

Thèse

Changjian Shui

Sous la direction de:

Christian Gagné, directeur de recherche
Boyu Wang, codirecteur de recherche

Résumé

Les réseaux de neurones profonds ont démontré un fort impact dans de nombreuses applications du monde réel et ont atteint des performances prometteuses dans plusieurs domaines de recherche. Cependant, ces gains empiriques sont généralement difficiles à déployer dans les scénarios du monde réel, car ils nécessitent des données étiquetées massives. Pour des raisons de temps et de budget, la collecte d'un tel ensemble de données d'entraînement à grande échelle est irréaliste.

Dans cette thèse, l'objectif est d'utiliser le *distribution matching* pour développer de nouvelles approches d'apprentissage profond pour la prédiction de peu de données étiquetées. En particulier, nous nous concentrons sur les problèmes d'apprentissage multi-tâches, d'apprentissage actif et d'adaptation au domaine, qui sont les scénarios typiques de l'apprentissage à partir de données étiquetées limitées.

La première contribution consiste à développer l'approche principale de l'apprentissage multi-tâches. Concrètement, on propose un point de vue théorique pour comprendre le rôle de la similarité entre les tâches. Basé sur les résultats théoriques, nous re-examinons l'algorithme du *Adversarial Multi-Task Neural Network*, et proposons un algorithme itératif pour estimer le coefficient des relations entre les tâches et les paramètres du réseaux de neurones.

La deuxième contribution consiste à proposer une méthode unifiée pour les requêtes et les entraînements dans l'apprentissage actif profond par lots. Concrètement, nous modélisons la procédure interactive de l'apprentissage actif comme le *distribution matching*. Nous avons ensuite dérivé une nouvelle perte d'entraînement, qui se décompose en deux parties : l'optimisation des paramètres du réseaux de neurones et la sélection des requêtes par lots. En outre, la perte d'entraînement du réseau profond est formulée comme un problème d'optimisation min-max en utilisant les informations des données non étiquetées. La sélection de lots de requêtes proposée indique également un compromis explicite entre incertitude et diversité.

La troisième contribution vise à montrer l'incohérence entre le *domain adversarial training* et sa correspondance théorique supposée, basée sur la $\mathcal{H}$-divergence. Concrètement, nous découvrons que la $\mathcal{H}$-divergence n'est pas équivalente à la divergence de Jensen-Shannon, l'objectif d'optimisation dans les entraînements adversaires de domaine. Pour cela, nous établissons un nouveau modèle théorique en prouvant explicitement les bornes supérieures et inférieures du risque de la cible, basées sur la divergence de Jensen-Shannon. Notre framework présente des flexibilités inhérentes pour différents problèmes d'apprentissage par transfert. D'un point de vue algorithmique, notre théorie fournit une guidance de l'alignement conditionnel sémantique, de l'alignement de la distribution marginale et de la correction du label-shift marginal.

La quatrième contribution consiste à développer de nouvelles approches pour agréger des domaines de sources avec des distributions d'étiquettes différentes, où la plupart des approches récentes de sélection de sources échouent. L'algorithme que nous proposons diffère des approches précédentes sur deux points essentiels : le modèle agrège plusieurs sources principalement par la similarité de la distribution conditionnelle plutôt que par la distribution marginale ; le modèle propose un cadre unifié pour sélectionner les sources pertinentes pour trois scénarios populaires, l'adaptation de domaine avec une étiquette limitée sur le domaine cible, l'adaptation de domaine non supervisée et l'adaptation de domaine non supervisée partielle par étiquette.

Abstract

Deep neural networks have demonstrated a strong impact on a wide range of tasks and achieved promising performances. However, these empirical gains are generally difficult to deploy in real-world scenarios, because they require large-scale hand-labeled datasets. Due to the time and cost budget, collecting such large-scale training sets is usually infeasible in practice.

In this thesis, we develop novel approaches through distribution matching to learn limited labeled data. Specifically, we focus on the problems of multi-task learning, active learning, and domain adaptation, which are the typical scenarios in learning from limited labeled data.

The first contribution is to develop a principled approach in multi-task learning. Specifically, we propose a theoretical viewpoint to understand the importance of task similarity in multi-task learning. Then we revisit the adversarial multi-task neural network and propose an iterative algorithm to estimate the task relation coefficient and neural-network parameters.

The second contribution is to propose a unified and principled method for both querying and training in deep batch active learning. We model the interactive procedure as distribution matching. Then we derive a new principled approach in optimizing neural network parameters and batch query selection. The loss for neural network training is formulated as a min-max optimization through leveraging the unlabeled data. The query loss indicates an explicit uncertainty-diversity trade-off batch-selection.

The third contribution aims at revealing the incoherence between the widely-adopted empirical domain adversarial training and its generally assumed theoretical counterpart based on $\mathcal{H}$-divergence. Concretely, we find that $\mathcal{H}$-divergence is not equivalent to Jensen-Shannon divergence, the optimization objective in domain adversarial training. To this end, we establish a new theoretical framework by directly proving the upper and lower target risk bounds based on the Jensen-Shannon divergence. Our framework exhibits flexibilities for different transfer learning problems. Besides, our theory enables a unified guideline in conditional matching, feature marginal matching, and label marginal shift correction.

The fourth contribution is to design novel approaches for aggregating source domains with different label distributions, where most existing source selection approaches fail. Our proposed algorithm differs from previous approaches in two key ways: the model aggregates multiple sources mainly through the similarity of conditional distribution rather than marginal distribution; the model proposes

a unified framework to select relevant sources for three popular scenarios, i.e., domain adaptation with limited label on the target domain, unsupervised domain adaptation and labels partial unsupervised domain adaption.

Contents

Résumé ii

Abstract iv

Contents vi

List of Tables ix

List of Figures x

Acknowledgment xii

Foreword xiii

Introduction 1

1 Background 6

1.1 Brief Introduction to Deep Multi-Task Learning 6

1.2 Brief Introduction to Deep Active Learning 8

1.3 Brief Introduction to Domain Adaptation 11

1.4 Distribution Matching and Statistical Divergence 14

2 Principled Approach for Learning Task Similarity in MTL 18

2.1 Introduction 19

2.2 Problem Set-up 20

2.3 Theoretical Guarantees in Multi-Task Learning 21

2.4 Adversarial Multitask Neural Network (AMTNN) 23

2.5 Experiments 26

2.6 Conclusion 29

3 Unified and Principled Method for Query and Training in Deep Active Learning 30

3.1 Introduction 31

3.2 Active Learning as Distribution Matching 31

3.3 Practical Deep Batch Active Learning 35

3.4 Experiments 39

3.5 Conclusion 43

4 Domain Adaptation Theory With Jensen-Shannon Divergence 44

4.1 Introduction 45

4.2 $\mathcal{H}$-Divergence based DA Theory 46
4.3 DA Theory with Jensen-Shannon Divergence and Theoretical Insights 48
4.4 Practical Principles for Unsupervised DA 52
4.5 Related Work 55
4.6 Experiments 56
4.7 Conclusion 60

5 Aggregating From Multiple Target-Shifted Sources 61
5.1 Introduction 62
5.2 Related Work 63
5.3 Problem Setup and Theoretical Insights 64
5.4 Practical Algorithm in Deep Learning 67
5.5 Extension to Unsupervised Scenarios 69
5.6 Experiments 71
5.7 Conclusion 76

Conclusion 79

A Details of Chapter 2 83
A.1 Theoretical tools 83
A.2 Proof of Theorem 2.1 85
A.3 Proof of Theorem 2.2 87
A.4 Experiment details 91

B Details of Chapter 3 93
B.1 Theorem 3.1: Proof 93
B.2 Corollary 3.1: Proof 95
B.3 Computing $\mathcal{H}$-divergence and Wasserstein distance 96
B.4 Developing loss in deep batch active learning 99
B.5 Redundancy trick: Computation 100
B.6 Uniform Output Arrives with the Minimal loss 100
B.7 Experiments 100
B.8 Ablation study 101

C Details of Chapter 4 104
C.1 $\mathcal{H}$-Divergence v.s. Jensen-Shannon Divergence 104
C.2 Domain Adaptation: Upper Bound (Theorem 4.1) 105
C.3 Domain Adaptation Theory: Lower Bound (Theorem 4.2) 108
C.4 Joint Jensen-Shannon Divergence Decomposition (Corollary 4.3) 110
C.5 Target Intrinsic Error Upper Bound 110
C.6 Inherent Difficulty for Controlling Label Conditional Shift 111
C.7 Novel Practice 113
C.8 Practical Guidelines 117

D Details of Chapter 5 119
D.1 Additional Related Work 119
D.2 Table of Notation 120
D.3 Proof of Theorem 5.1 120
D.4 Proof of Theorem 5.2 126

D.5 Approximating W_1 distance 128
D.6 Proof of Lemma 5.2 128
D.7 Derive the label ratio loss 129
D.8 Label Partial Multi-source unsupervised DA 130
D.9 Explicit and Implicit conditional learning 130
D.10 Algorithm Descriptions 131
D.11 Dataset Description and Experimental Details 133
D.12 Analysis of Unsupervised DA 137
D.13 Partial multi-source Unsupervised DA 138

Bibliography **145**

List of Tables

1.1 Popular f-divergence 14

2.1 Average test accuracy (in %) of MTL algorithms on the digits datasets. 27
2.2 Average test accuracy (in %) of MTL algorithm in the sentiment dataset. 28

3.1 Relative Average querying time, assuming the query time of WAAL as the unit. . . . 39
3.2 Results of WAAL on SVHN (by accuracy and Macro-F1) 43

4.1 Different DA Theories for classification. 49
4.2 Empirical Methods for Bi-Directional Marginal/Conditional Shifts 52
4.3 Accuracy (%) on Digits Dataset 58
4.4 Accuracy (%) on Office-31 Dataset 59

5.1 Unsupervised DA: results on Source-Shifted Amazon Review 72
5.2 Unsupervised DA: results on Source-Shifted Source-Shifted Digits 72
5.3 Unsupervised DA: results on Office-Home 74
5.4 Multi-source transfer: results on Amazon review 74
5.5 Multi-source Transfer: results on Source-Shifted Digits 76
5.6 Unsupervised Partial DA: results on Office-Home 76

B.1 Dataset descriptions 100
B.2 Hyper-parameter setting 101
B.3 Result of FashionMNIST (Average ± std) 102
B.4 Result of SVHN (Average ± std) 102
B.5 Result of CIFAR10 (Average ± std) 102
B.6 Result of STL10 (Average ± std) 102
B.7 Ablation study of CIFAR10 (Average ± std) 103
B.8 Ablation study of SVHN (Average ± std) 103

D.1 Table of Notations 120

List of Figures

0.1 Task similarity based multi-task learning 2
0.2 Thesis Structure 5

1.1 Structure of Chapter 1 7
1.2 Typical Deep Multi-Task Network Structure 7
1.3 Deep Multi-Task Learning through an auxiliary task 8
1.4 A high-level illustration of Batch Active Learning 8
1.5 Sampling bias in AL 9
1.6 Uncertainty v.s diversity criteria 10
1.7 Illustration of Domain Adaptation 11
1.8 General framework of domain adversarial neural network 13
1.9 Generalized Distribution Matching Network 17

2.1 General framework of Adversarial Multitask Neural Network (AMTNN). 25
2.2 Estimated task relation coefficients matrix 27
2.3 t-SNE in the feature space 28

3.1 $\mathcal{H}$-divergence vs. Wasserstein distance for $\mathcal{D}$-$\mathcal{Q}$ distribution matching 34
3.2 Empirical performance (Accuracy) on Fashion MNIST, SVHN, CIFAR-10 and STL-10 41
3.3 Ablation Study 42

4.1 Incoherence between JS divergence and $\mathcal{H}$-divergence 47
4.2 Label distribution on two Datasets 57
4.3 Analysis of proposed principles. 59
4.4 Ablation Study: Label-drifted DA. 60

5.1 Limitation of merely considering marginal distribution 62
5.2 Illustration of proposed algorithm (WADN) and relation with other scenarios 67
5.3 Label distribution visualization 73
5.4 The evolution of $\boldsymbol{\lambda}$ in different approaches 75
5.5 Ablation study on different scenarios 77
5.6 Analysis on Partial DA of target Product 78

D.1 Network Structure of Proposed Approach 131
D.2 Visualization of Amazon Review dataset 135
D.3 Neural Network Structure in the digits recognition (Ganin et al., 2016) 136
D.4 Label distribution drift in digits dataset 136
D.5 Samples Images From Office-Home dataset 137
D.6 Neural Network Structure in the Office-Home 137

D.7 Different label drift levels on Amazon Dataset 138
D.8 WADN approach: evolution of $\hat{\alpha}_t$ during the training 139
D.9 Digits Dataset. WADN approach: evolution of $\hat{\alpha}_t$ during the training 141
D.10 Multi-source Label Partial DA: Performance with different target selected classes. . . 142
D.11 Multi-source partial DA: visualization of α, class number: 15 143
D.12 Multi-source partial DA: visualization of α, class number: 35 144

Acknowledgment

First of all, I gratefully thank my PhD supervisory panel. I deeply appreciate the countless help and strong support from my primary supervisor Prof. Christian Gagné, who is an excellent AI researcher in the solidity of knowledge, accuracy in communication and uncompromising pursuit of useful research. Christian also provided for his invaluable efforts towards a relaxed atmosphere and respectful culture which I have truly benefited from. Without his consistent encouragement, I could not explore different research directions in machine learning. I would also like to thank my co-supervisor Prof. Boyu Wang for being not only a great advisor but also a kind listener, and for our resonating discussions.

I also would like to express appreciations for my amazing collaborators during my PhD research: Qi Chen, Fan Zhou, Mahdieh Abbasi, Jun Wen, Jiaqi Li, Zijian Li, Kezheng Xu and Wei Wang. They provided me enormous assistance during my Ph.D such as coding, idea validation and paper proof-reading. Clearly, this thesis could not finish without their tremendous support and assistance. They are not only research collaborators but also my friends. Indeed, I would appreciate the help and hope for the future collaborations.

I would like to thank my thesis committee members: Mario Marchand, Thierry Duchesne and Aaron Courville. The detailed feedback and suggestions are indeed valuable to make a consistent better and solid thesis.

Besides, I would like appreciate the wonderful members in computer vision lab (LVSN): Annette Schwerdtfeger, Hugo Siqueira Gomes, Azadeh Sadat Mozafari and Gabriel Leclerc. Their precious discussions and valuable suggestions enable me to pass difficult moments during my research.

Finally I would like to thanks my parents and my family, who provided me sufficient confidence, trust and encouragement for these years. I could not make it without your love.

Foreword

This thesis is mainly written by articles and we briefly describe each of them.

The first part of this thesis (chapter. 2) has been published in the International Joint Conference on Artificial Intelligence (IJCAI) 2019, with an acceptance rate of 17.9%.

A Principled Approach for Learning Task Similarity in Multitask Learning.
Changjian Shui, Mahdieh Abbasi, Louis-Émile Robitaille, Boyu Wang, Christian Gagné.
Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. Main track. Pages 3446-3452. https://doi.org/10.24963/ijcai.2019/478

Changjian Shui is the main contributor to this work. Mahdieh Abbasi, Louis-Émile Robitaille helped Changjian in debugging the code and polishing the paper. Boyu Wang and Christian Gagné are the supervisors of this research. In the thesis, the introduction and related work are rewritten. Additional experiments and analyses are added.

The second part of the thesis (chapter. 3) has been published in International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, with an acceptance of 28.5%.

Deep active learning: Unified and principled method for query and training.
Shui, Changjian, Fan Zhou, Christian Gagné, and Boyu Wang.
Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, PMLR 108:1308-1318, 2020.

Changjian Shui is the main contributor to this work. Fan Zhou helped Changjian in testing code and data preprocessing. Boyu Wang and Christian Gagné are the supervisors of this research. In the thesis, the introduction and related work are rewritten. Additional experiments and analyses are added.

The third part of the thesis (chapter. 4) is in preparation for submission.

A New Domain Adaptation Theory With Jensen-Shannon Divergence
Changjian Shui, Qi Chen, Jun Wen, Fan Zhou, Christian Gagné, and Boyu Wang.

Changjian Shui is the main contributor to this work. Qi Chen, Fan Zhou and Jun Wen help Changjian in code testing, data preprocessing, and paper polishing. Boyu Wang and Christian Gagné are the supervisors of this research. The article has been rewritten in the introduction and related work part. Additional experiments and analysis are added.

The last part of the thesis (chapter. 5) has been published in International Conference on Machine Learning (ICML) 2021, with the acceptance rate 21.5%.

Aggregating From Multiple Target-Shifted Sources
Changjian Shui, Zijian Li, Jiaqi Li, Christian Gagné, Charles X. Ling, Boyu Wang.
Proceedings of the 38th International Conference on Machine Learning, PMLR 139:9638-9648, 2021.

Changjian Shui is the main contributor to this work. Zijian Li, Jiaqi Li helped Changjian in code testing and data preprocessing. Charles X. Ling helped in defining the idea and polishing the paper. Boyu Wang and Christian Gagné are the supervisors of this research. The article has been rewritten in the introduction and related work part. Additional experiments and analysis are added.

Introduction

Modern machine learning systems such as deep neural networks have demonstrated a strong impact on a wide range of real-world applications and achieved state-of-the-art performances for computer vision (Krizhevsky et al., 2012), natural language processing (Bahdanau et al., 2014) and speech recognition (Yu and Deng, 2014). However, these empirical gains are generally difficult to deploy to real-world scenarios, because they usually require *large scale* hand-labeled datasets. Moreover, it is usually not practically feasible to collect such a large training dataset due to the time and cost budget (Roh et al., 2019).

To address this, various research directions have been proposed to address the label annotation scarcity, such as labeling the most informative data for training (*Active Learning*) (Dasgupta, 2011)), learning multiple similar tasks with the limited labels by extracting the common knowledge (*Multi-Task Learning*) (Zhang and Yang, 2017), or using the information from a reference task (i.e. source task) with sufficient data (*Transfer Learning/Domain Adaptation*) (Pan and Yang, 2009). Based on these settings, learning from limited label data becomes theoretically and practically feasible.

In this thesis, we will develop novel and principled representation learning approaches to learn limited-label tasks. In particular, we will introduce the scenarios of multi-task learning, active learning, and domain adaptation, which are the typical scenarios of learning with limited labeled data.

Throughout the thesis, we will use *distribution matching* (Sugiyama and Kawanabe, 2012) as the tool in analysis that has been widely applied in enormous deep learning regimes such as deep generative models (Goodfellow et al., 2014; Kingma and Welling, 2013), transfer learning (Ganin et al., 2016). The core idea is to learn an embedding function (representation) to obtain the invariant features between two distributions.

The rest of the introductory chapter will consist of three parts: (1) a brief introduction to the learning scenarios in this paper, (2) the research objective of this paper, and (3) the structure of the paper.

Scenarios of learning from limited labeled data

Multi-task learning Multitask learning aims at solving a set of related tasks simultaneously by using shared knowledge to improve performance on each task. Under various assumptions, multitasking has been shown to reduce the number of labeled examples required per task to achieve acceptable performance.

Active Learning Active learning has proposed an alternative solution to reduce the data annotation overhead. The learning system seeks the *most informative* data points for labeling from a pool of unlabeled examples to maximize the prediction performance.

Domain Adaptation Domain Adaptation is based on the motivation that learning a new task is easier after learning a similar task. Adopting the *inductive bias* from a set of related source domains and then using the shared knowledge when learning the target domain can significantly improve prediction performance.

Research Objective

This thesis focuses on three specific scenarios with a series of new theoretical and practical contributions. Briefly, the first contribution is to understand multi-task learning through task similarity information. The second contribution is a novel principled training and query strategy in deep active learning. The third contribution is the novel theoretical analysis through Jensen-Shannon Divergence for domain adaptation. The fourth contribution is to understand label shift in multi-source domain adaptation. In the following, the aims of this thesis and research questions will be detailed.

Objective 1. Understanding Similarity-Based Multi-Task Learning. The success of multitask learning approaches lies in assumptions of shared knowledge among different tasks. One possible strategy is to adopt the *task similarity* as an inductive bias to understand multi-task learning, shown in Fig. 0.1.

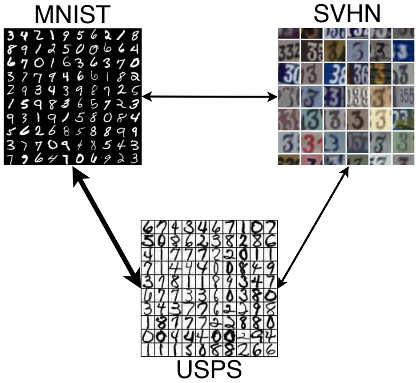


Figure 0.1 – Task similarity based multi-task learning. Compared with dataset SVHN, the MNIST and USPS datasets are semantically similar without noisy background.

Fig. 0.1 illustrates the intuition of using similarity-based multi-task learning. Assuming three tasks in digit recognition: MNIST (LeCun et al., 1998), USPS (Pan and Yang, 2009) and SVHN (Netzer et al., 2011), intuitively, the MNIST and USPS dataset are more similar since SVHN has a relative noisy background and different digit configurations. Therefore, in the algorithm design, MNIST and USPS tasks should be processed in a similar way to ensure good performance.

The work associated with this objective aims at answering the following research questions:

1. What is the theoretical and empirical benefit of considering task similarity information?
2. How to derive practical algorithm through using task similarity information?

Objective 2. Unified Method for Query and Training in Deep Active Learning The key to successful active learning is to properly design the query algorithms. For instance, using the uncertainty principle (e.g., maximum entropy principle) to conduct the query, which can lead to biased sampling in a subset of the dataset. To mitigate this effect, recent works have considered obtaining a diverse set (Sener and Savarese, 2018) of samples to capture the global data distribution information. However, this can be problematic in a small query budget for ensuring a subset that covers the whole data.

The work associated with this objective aims at answering the following research questions:

1. Is there a principled query strategy that simultaneously considers the *uncertainty* and *diversity* criteria?
2. In the context of deep learning, the available large set of *unlabeled samples* may be helpful to construct a good feature representation that would potentially improve performance. In order to further promote better results, can we additionally design a loss for optimizing DNN's weights that would leverage the unlabeled samples during training?

Objective 3. Domain Adaptation Theory With Jensen-Shannon Divergence Domain Adaptation (DA) theory is crucial to the fundamental understanding and practical development of relevant algorithms. Conventionally, such theoretical guarantees were typically based on the notion of $\mathcal{H}$-divergence (Ben-David et al., 2007, 2010a) and its subsequent variants, where it requires a small $\mathcal{H}$-divergence between source-target and small joint risk. In the context of deep learning, $\mathcal{H}$-divergence is minimized via the well-known *domain adversarial training* (Ganin et al., 2016), which is a stimulating topic. Despite domain adversarial training being widely successful in various DA problems such as open set DA (Panareda Busto and Gall, 2017; Cao et al., 2018; You et al., 2019) or label shift (Li et al., 2019b), the generally assumed theoretical counterpart $\mathcal{H}$-divergence itself is rather limited to explain these working principles, which hampers the further practical advancement.

The work associated with this objective aims at answering the following research questions:

1. It has been noted that the inherent principle of domain adversarial training is analog to GANs (Goodfellow et al., 2014), which is equivalent to minimizing the Jensen-Shannon divergence

(Nowozin et al., 2016) of two distributions. Therefore, can we derive a domain adaptation theory directly on the Jensen-Shannon divergence?

2. What are the novel theoretical and practical implications of Jensen-Shannon divergence-based theory?

Objective 4. Multi-Source Domain Adaptation Approach Under Label Shifts One implicit assumption in most transfer learning algorithms is that the label proportions remain unchanged across different domains (Du Plessis and Sugiyama, 2014) (i.e., $\mathcal{S}(y) = \mathcal{T}(y)$). In many real-world applications, the label distributions can vary markedly (i.e., label shift) (Wen et al., 2014; Lipton et al., 2018; Li et al., 2019b). For instance, in disease diagnostics, the portions of different diseases can vary dramatically, whereas existing approaches cannot guarantee a small target generalization error.

Moreover, domain adaptation becomes even more challenging when transferring knowledge from *multiple sources* to build a model for the target domain, as this requires an effective selection and leveraging the most useful source domains when label shift occurs ($\mathcal{S}(y) \neq \mathcal{T}(y)$). This is not only theoretically interesting but also commonly encountered in real-world applications.

For example, (Liu et al., 2004) pointed out the ratio of coronary heart disease (CHD) varies among different provinces in China. If we consider the task of CHD diagnosing in a province without sufficient labeled data or even unlabeled data, how can we leverage the information from other provinces with abundant data to help the diagnosing? Furthermore, in the intelligent health, due to different demographics, the disease proportion generally varies over countries (Geiss et al., 2014).

Motivated by this real-world practice: the work associated with this objective aims at answering the following research question:
Can we derive new theories and practices in multisource domain adaptation when label shift occurs?

Thesis Structure

The outline of the thesis is shown in Fig.0.2. Chapter 1 deals with the foundations and background of the thesis. Chapter 2 introduces the novel approach of similarity-based multi-task learning. Chapter 3 presents the principled approach to query and training in Deep Active Learning. Chapter 4 presents a theory of domain adaptation through Jensen-Shannon divergence with new practical and theoretical insights. Chapter 5 presents a novel approach for handling label shift in multi-source domain adaptation.

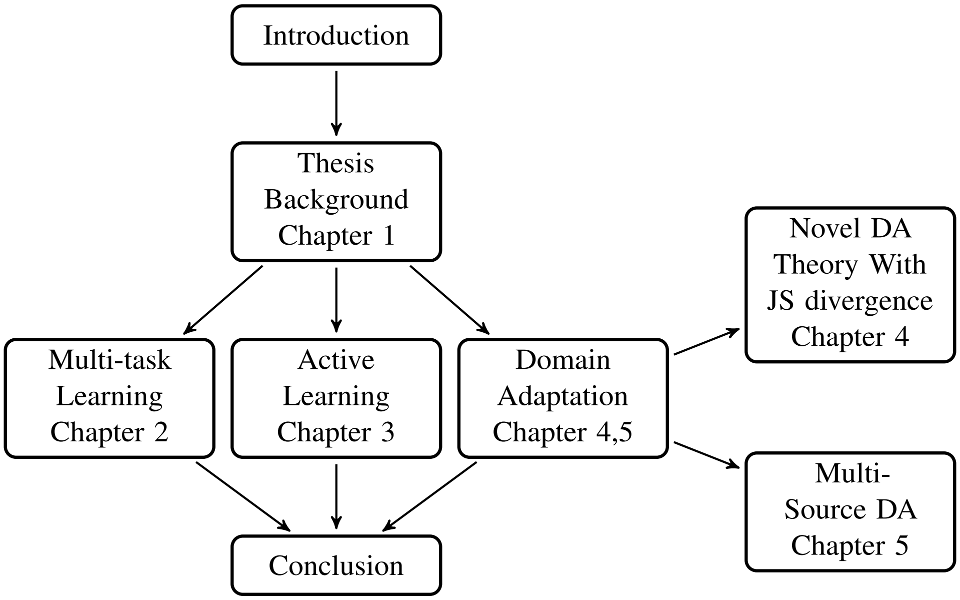

Introduction
Thesis Background Chapter 1
Multi-task Learning Chapter 2
Active Learning Chapter 3
Domain Adaptation Chapter 4,5
Novel DA Theory With JS divergence Chapter 4
Multi-Source DA Chapter 5
Conclusion

Figure 0.2 – Thesis Structure

Chapter 1

Background

This chapter is divided into two parts. In the first part, an overview of different scenarios for learning limited labels is proposed. In the second part, distribution matching is presented as the mathematical tool used in these scenarios.

1.1 Brief Introduction to Deep Multi-Task Learning

Traditional machine learning focused mainly on developing learning algorithms for single problems. While significant progress has been made in applied and theoretical research, in such a context, a large amount of labeled data is still required to achieve a small generalization error. In practice, this can be very costly, for example, in modeling user preferences for products (Murugesan and Carbonell, 2017), classifying multiple objects in computer vision (Long et al., 2017), or analyzing patient data in computational medicine (Wang and Pineau, 2015). In the multi-task scenario (MTL), an agent learns the shared knowledge between *a set of related tasks*. Under various assumptions about task relations, MTL has been shown to reduce the number of labeled examples required per task to achieve acceptable performance.

At the same time, constructing models for the different tasks raises new problems that do not arise in single-task learning. Specifically, *negative transfer* often occurs in the MTL regime because different tasks have different or even conflicting goals. If we only enforce loss minimization for all tasks, the minority tasks are generally neglected by the learning algorithm, which can lead to undesirable social and ethical problems. For this reason, the main goal of the MTL framework is to promote performance for each task while avoiding a negative transfer.

From a practical point of view, several empirical approaches have been proposed to mitigate negative transfer, such as developing specific losses and studying task relationships in optimization, which are presented in the following sections.

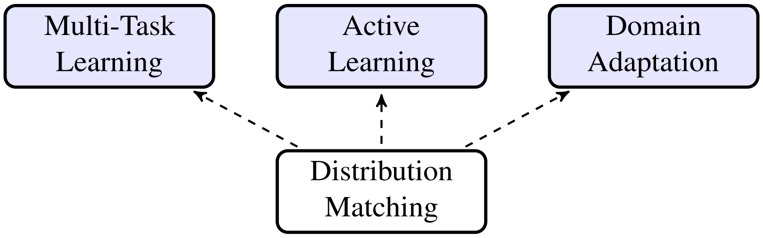


Figure 1.1 – Structure of Chapter 1. We will first present the brief introduction on the scenarios of learning limited label tasks: multi-task learning, active learning and domain adaptation. Then on the level of methodology, we will introduce *distribution matching*, the common tool in analyzing these scenarios.

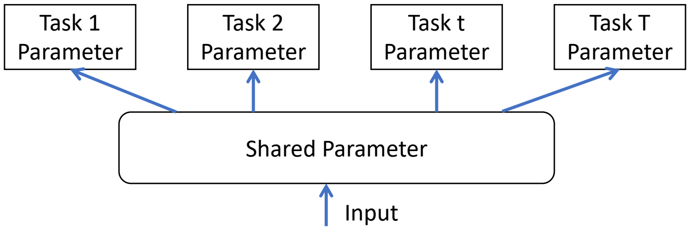


Figure 1.2 – A typical Deep Multi-Task Network Structure. It consists of two main components, the shared parameter θ_{share} and the task-specific parameter θ_t. Specifically, the role of θ_{share} is to extract the common representation among all tasks, while θ_t aims to model specific task information.

1.1.1 MTL Approaches: An optimization perspective

The deep multi-task learning can be broadly viewed as an optimization procedure over the parameters on each task (task parameters) and parameters for all tasks (shared parameters) (Zhang and Yang, 2017), shown in Fig. 1.2.

Based on this structure, we aim to design losses to learn θ_{share} and θ_t for promoting shared information and neglecting negative transfer. A common strategy is to define weighted losses for each task, formulated as:

$$\sum_{t=1}^{T} w_t \mathcal{L}_t(\theta_t, \theta_{\text{share}}),$$

where $w_t > 0$ represents the task weights and $\mathcal{L}_t(\theta_t, \theta_{\text{share}}) = \sum_{(x_t,y_t)\sim\mathcal{D}_t} \ell(\theta_t, \theta_{\text{share}}; (x_t, y_t))$ is the loss for task t. Therefore, the MTL problem is equivalent to determining the importance of each task.

To address this problem, Murugesan and Carbonell (2017); Murugesan et al. (2016) used some heuristics such as the norm of the gradient $w_t \approx \|\nabla \mathcal{L}_t\|$. Later, Sener and Koltun (2018) treated the optimization procedure as a multi-objective optimization that aims to seek *Pareto optimal solutions* by optimizing w_t. However, the main technical challenge of multi-objective optimization-based approaches is the high computational complexity of finding the optimal Pareto value in the deep network, which is still an open question.

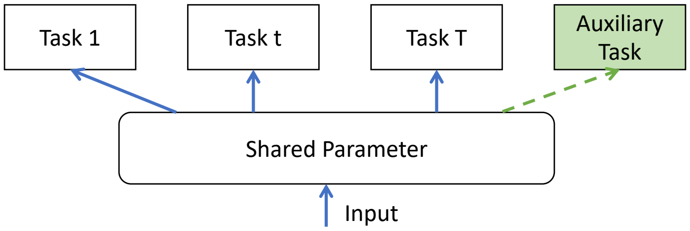


Figure 1.3 – Deep Multi-Task Learning through an auxiliary task. The role of the auxiliary task (e.g, adversarial loss) generally aims at balancing the task loss for ensuring all the tasks being learned.

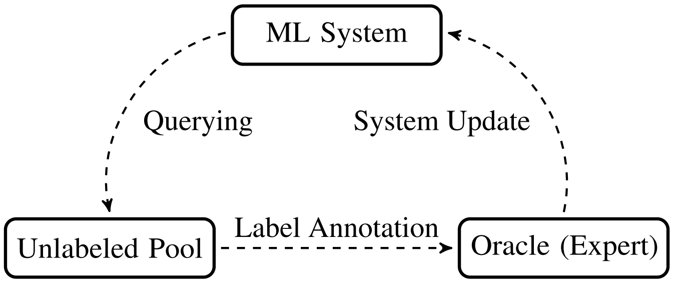


Figure 1.4 – A high-level illustration of Batch Active Learning. It consists of three main components: the machine learning system, unlabeled data pool and oracle.

1.1.2 MTL Approaches: Task relation perspective

An alternative approach to find w_t is to use task relations. Intuitively, we assign similar weights to similar tasks. Following this line of research, Pentina and Lampert (2017); Liu et al. (2017); Li et al. (2018b) measured task similarity and assigned weights through an auxiliary task, as shown in Fig. 1.3.

The auxiliary task can be either adversarial learning (Goodfellow et al., 2014) or a data reconstruction task (Kingma and Welling, 2013), where the general goal is to estimate task relationships. In adversarial training, for example, we have developed a discriminator that evaluates the similarity of tasks based on the difficulty of distinguishing them. One contribution of our work is to derive a novel theoretical analysis of the role of auxiliary tasks in MTL.

1.2 Brief Introduction to Deep Active Learning

An alternative perspective in learning limited labeled data is only to select important samples in the training (i.e., active learning). We showed a general protocol of deep active learning (AL) in Fig 1.4, which aims to query the most informative samples from the unlabeled data pool. Then the Oracle (or expert) annotates for the labeling to maximize the prediction performance.

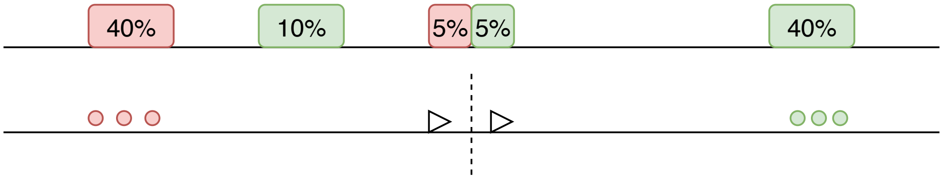


Figure 1.5 – Sampling bias of uncertainty in Active Learning (Dasgupta, 2011). Consider one-dimensional binary classification problem (prediction red/green), the data generation distribution consists of four uniform intervals. The Red/Green dots are the initial observations; dotted line are the estimated decision boundary from the initial samples; The triangles are querying samples according to the uncertainty based strategies w.r.t. current decision boundary. Clearly, the uncertainty based approach falls into a sub-optimal solution with prediction error 10% rather than the global optimal 5%.

1.2.1 Uncertainty and Diversity Query

Uncertainty Since the performance of AL depends on the selected samples, a key question is to search the most informative samples in the context of the deep neural network (DNN). A common solution is to apply the DNN output confidence score as the uncertainty acquisition function (Settles, 2012; Gal et al., 2017; Haussmann et al., 2019). For example, denote the output of the classifier as $S(x, y)$, which is a function containing the probability of different classes y, given input x.[1] Then the uncertainty aims at searching the unsure samples such as maximum entropy w.r.r. $S(x, y)$ or least confident score $\max_y S(x, y)$, showing in Fig 1.6.

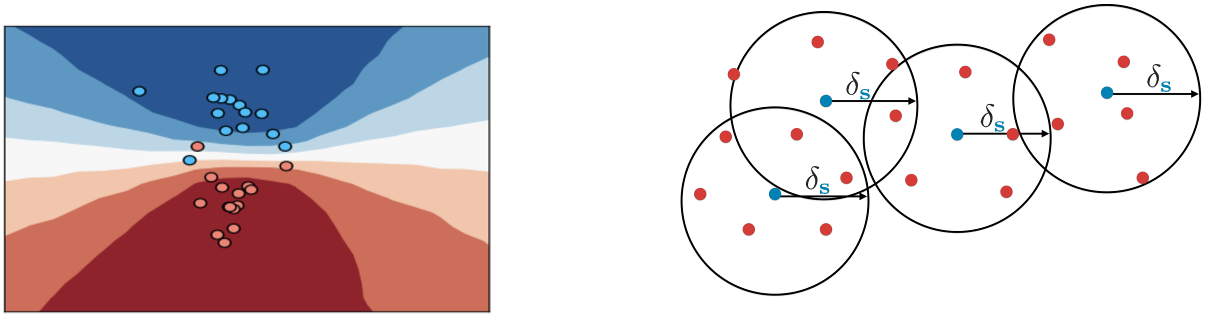
However, a well-known issue for uncertainty-based sampling in AL is the so-called *sampling bias* (Dasgupta, 2011): the current labeled points are not representative of the underlying distribution. For example, as shown in Fig. 1.5, assuming that the few initial labeled samples lie in the two extreme regions. Then based on an uncertainty approach, the queried samples are nearest to the currently estimated decision boundary, which will lead to a final suboptimal risk of 10% instead of the true optimal risk of 5%. This issue will be more severe in high-dimensional and complex datasets, where DNNs are commonly deployed.

Diversity An alternative solution is to query a *diverse* set [2] of samples for training DNN to reduce the sampling bias. I.e, if the queried samples are the most representative of the whole dataset (e.g, the centers of the clusters of the dataset), the queries samples will be informative. In other words, if the the queried dataset is $Q(x)$, then the diversity suggests queried data should be resemble the whole dataset $\mathcal{D}(x)$: $Q(x) \approx \mathcal{D}(x)$, showing in Fig 1.6.

For example, Sener and Savarese (2018) constructed the core-sets through solving the K-center problem. However, searching the core-set is still computationally expensive as it requires constructing a large distance matrix of unlabeled samples. More importantly, it might not be a proper approach,

[1] Clearly we have $\sum_y S(x, y) = 1$.

[2] In some studies, diversity criteria is also referred to as representative criteria, with the same purpose: to query the samples to reflect their underlying distribution.



(a) Uncertainty Criteria (Gal et al., 2017)

(b) Diversity Criteria (Sener and Savarese, 2018)

Figure 1.6 – Uncertainty v.s diversity criteria. (a) The Uncertainty criteria aims to query the most unsure samples near the current decision boundary. However, the current samples are not representative of the whole dataset, which is consistent with Fig.3.2. (b) The Diversity criteria aims to query the most representative samples from the the whole data: the centers of the clusters within the dataset.

particularly for a large-scale unlabeled pool and a *small* query batch, which it is hard to cover the entire data (Ash et al., 2019).

Instead of solely focusing on uncertainty or diversity samples during the query, a hybrid strategy might be more appropriate. For example, Yin et al. (2017) heuristically selected a portion of samples according to the uncertainty score for exploitation and random sampling in the remaining portion for exploration. Ash et al. (2019) collected samples whose gradients span a diverse set of directions by implicitly considering these two criteria. Since such hybrid strategies empirically showed improved performance, an interesting direction is to derive the querying approach with explicitly considering the uncertainty-diversity trade-off from a principled viewpoint.

1.2.2 Training Rules in Deep AL

Besides, in the context of deep AL, the available large set of unlabeled samples may be helpful to construct a good feature representation that would potentially improve performance. To further promote better results, the question is how we can additionally design a loss for optimizing DNN's weights that would leverage from the unlabeled samples during the training.

To address this, a promising line of work is to integrate the training with a *deep generative model* which naturally focuses on the unlabeled data information (Goodfellow et al., 2014; Kingma and Welling, 2013). Only a few works strode in this direction. Notably, Sinha et al. (2019) empirically adopted a β-VAE to construct the latent variables. Then they adopted the intuition from (Gissin and Shalev-Shwartz, 2019), which searched the diverse unlabeled batch for samples that do not look like the labeled samples, through adversarial training. Despite some good performance, this approach still concentrated on empirically designing the training loss and *the formal justifications remain elusive*,

Therefore, in our thesis, the second contribution is to derive a principled approach of both query and training in a deep active learning approach.

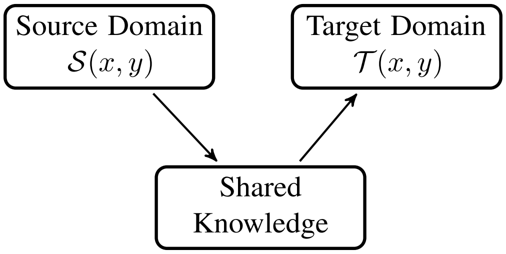


Figure 1.7 – Illustration of Domain Adaptation. The shared knowledge (inductive bias) is learned from the sources, and then transferred to the target.

1.3 Brief Introduction to Domain Adaptation

Domain Adaptation (DA) or Transfer Learning proposes another prospect to learn a target task with limited or even no labels. The success of DA lies in extracting the shared knowledge between two domains, shown in Fig.1.7. To this end, the system can learn the target with limited data.

Unsupervised DA Unsupervised Domain Adaptation is one popular scenario, where we want to learn the source domain $\mathcal{S}(x, y)$ and leverage the knowledge to the *unlabeled* target domain $\mathcal{T}(x)$. Unsupervised DA is impossible if no additional assumption between the source and target is added. Therefore, a core question in unsupervised DA is to explore the working conditions. Ben-David et al. (2006) proposed a series of theoretical works to explore the possibility of a successful DA.

$\mathcal{H}$-divergence and Unsupervised DA theory

Suppose the data is defined in input space $\mathcal{X}$ and label space $\mathcal{Y} = \{-1, 1\}$. The source and target tasks on $\mathcal{X}$ are different, denoting as the source domain $\mathcal{S}$ with probability density $\mathcal{S}(x, y)$ and target domain $\mathcal{T}$ with probability density $\mathcal{T}(x, y)$. [3] In unsupervised DA, we aim to learn a model on the labeled source dataset S that are i.i.d. sampled from $\mathcal{S}$ and unlabeled target dataset T that is i.i.d. sampled from target domain $\mathcal{T}$:

$$S = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n} \sim (\mathcal{S})^n \quad T = \{(x_i)\}_{i=n+1}^{N} \sim (\mathcal{T})^{n'}$$

with $N = n + n'$ being the total number of samples. The goal is to build a classifier $h : \mathcal{X} \to \mathcal{Y}$, $h \in \mathcal{H}$ with a small target risk

$$R_{\mathcal{T}}(h) = \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{T}}(h(\mathbf{x}) \neq y)$$

without label information in the target domain.

To address the domain adaptation, the motivation is to upper bound the target prediction error by the source prediction error plus a term which evaluates the *similarity* between source and target. *Intuitively, if two domains are similar, it is easier to leverage the knowledge from the source domain.* Formally,

[3] Throughout this thesis, $\mathcal{S}(x)$ and $\mathcal{T}(x)$ are denoted as the marginal distribution rather than the specific evaluation at point x.

Ben-David et al. (2006) introduced $\mathcal{H}$-divergence to measure the similarity and derived the following theoretical results.

Definition 1.1 ($\mathcal{H}$-divergence (Ben-David et al., 2006, 2010a)). Consider two distributions $\mathcal{P}(x)$ and $\mathcal{Q}(x)$ over $\mathcal{X}$, and hypothesis set $\mathcal{H}$, which is a set of binary classifier $\eta : \mathcal{X} \rightarrow \{0, 1\}$. Then $\mathcal{H}$-divergence between $\mathcal{P}(x)$ and $\mathcal{Q}(x)$ is defined as:

$$d_{\mathcal{H}}(\mathcal{P}(x), \mathcal{Q}(x)) = 2 \sup_{\eta \in \mathcal{H}} |\mathbb{P}_{x\sim\mathcal{P}}(\eta(x) = 1) - \mathbb{P}_{x\sim\mathcal{Q}}(\eta(x) = 1)|$$

Intuitively, $\mathcal{H}$-divergence depends on the capacity of the hypothesis set $\mathcal{H}$ to distinguish between the samples generated from $\mathcal{P}$ and $\mathcal{Q}$. Ben-David et al. (2006) further pointed out that for a hypothesis class $\mathcal{H}$, we can compute empirical $\mathcal{H}$-divergence between two samples $P \sim (\mathcal{P}(x))^m$ and $Q \sim (\mathcal{Q}(x))^m$ as:

$$\hat{d}_{\mathcal{H}}(P, Q) = 2\left(1 - \min_{\eta\in\mathcal{H}}\left[\frac{1}{m}\sum_{i=1}^{m}\mathbf{1}\{\eta(\mathbf{x}_i) = 0\} + \frac{1}{m}\sum_{i=m+1}^{2m}\mathbf{1}\{\eta(\mathbf{x}_i) = 1\}\right]\right)$$

where $\{\mathbf{x}_i\}_{i=1}^{m}$ is the source samples and $\{\mathbf{x}_i\}_{i=m+1}^{2m}$ are the target samples, $\mathbf{1}\{\cdot\}$ is the indicator function, which equals to 1 if prediction is true and 0 otherwise. Ben-David et al. (2006, 2010a) demonstrated a PAC (Probability Approximately Correct) bound to estimate the gap between the expected and empirical $\mathcal{H}$-divergence.

Proposition 1.1. *Let $\mathcal{H}$ be a hypothesis space on $\mathcal{X}$ with VC dimension d. If P and Q are samples of size m from $\mathcal{P}$ and $\mathcal{Q}$ respectively and $\hat{d}_{\mathcal{H}}(P, Q)$ is the empirical $\mathcal{H}$-divergence between samples, then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$:*

$$d_{\mathcal{H}}(\mathcal{P}(x), \mathcal{Q}(x)) \leq \hat{d}_{\mathcal{H}}(P, Q) + 4\sqrt{\frac{d\log(2m) + \log(\frac{2}{\delta})}{m}}$$

The proposition shows that the empirical $\mathcal{H}$-divergence between two samples from distributions $\mathcal{P}$ and $\mathcal{Q}$ converges uniformly to the true $\mathcal{H}$-divergence for hypothesis class $\mathcal{H}$ of finite VC dimension.

Based on $\mathcal{H}$-divergence, the target source risk can be bounded through Theorem 1.

Theorem 1.1. *Let $\mathcal{H}$ be a hypothesis class with VC dimension d, with high probability over the choice of source samples $S \sim (\mathcal{S}(x))^n$ and unlabeled target samples $T \sim (\mathcal{T}(x))^n$, for $\forall h \in \mathcal{H}$, we have:*

$$R_{\mathcal{T}}(h) \leq \hat{R}_S(h) + \hat{d}_{\mathcal{H}}(S, T) + \sqrt{\frac{4}{n}(d\log(\frac{2en}{d}) + \log(\frac{4}{\delta}))} + 4\sqrt{\frac{1}{n}(d\log(\frac{2n}{d}) + \log(\frac{4}{\delta}))} + \beta,$$

where $\beta = \inf_{h^\star\in\mathcal{H}}[R_{\mathcal{S}}(h^\star) + R_{\mathcal{T}}(h^\star)]$ is the optimal risk on both domain and the empirical source risk $\hat{R}_S(h) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{1}\{h(\mathbf{x}_i) \neq y_i\}$.

Theorem 1.1 shows that the target risk is upper bounded by the source risk, the $\mathcal{H}$ divergence between source and target, as well as the optimal classification error. Besides, to ensure a small upper bound, a

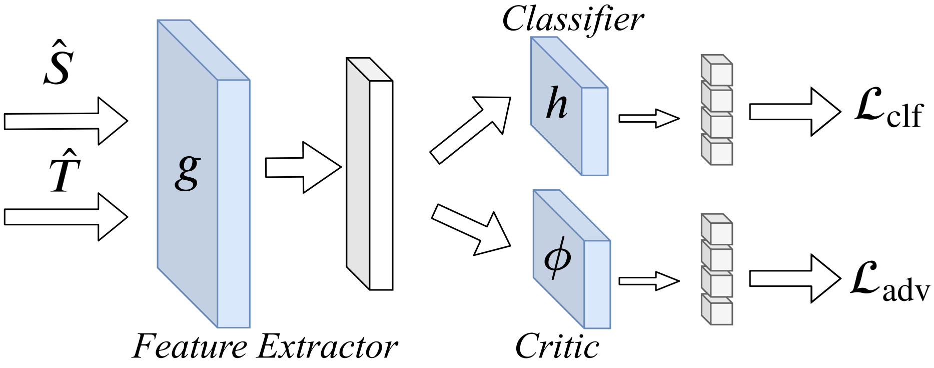


Figure 1.8 – General framework of domain adversarial neural network. The network consists three main components: the feature extractor g, the classifier and the critic function ϕ. The min-max optimization is conducted between ϕ and g, where the critic function ϕ aims to differentiate from source and target and enforces g to learn the invariant representation.

small joint optimal classification error β is necessarily. Otherwise, optimizing the upper bound for an arbitrary large β can not guarantee a small target risk.

It is also worth mentioning that Ben-David et al. (2010a) illustrated that the $\mathcal{H}$-divergence based theory is inaccurate. Actually, the $\mathcal{H}$-divergence should be $\mathcal{H}\Delta\mathcal{H}$ divergence. However, in practice, $\mathcal{H}\Delta\mathcal{H}$ divergence is impossible to estimate from the data. Then Ben-David et al. (2010a) still adopted the $\mathcal{H}$-divergence to approximate $\mathcal{H}\Delta\mathcal{H}$ divergence. Thus, in this thesis, we will treat $\mathcal{H}$ and $\mathcal{H}\Delta\mathcal{H}$ divergence interchangeably throughout the paper.

Given a small β, Ganin et al. (2016) designed a domain adversarial neural network to realize the unsupervised DA, shown in Fig 1.8.

The domain adversarial network consists of three parts: the feature extractor g that maps the data to the latent space $z = g(x)$, the classifier h that predicts the output $y = h(z)$ and the critic (domain discriminator) function ϕ that aims at differentiating the source and target distribution. Therefore, the loss in the deep domain adaptation can be expressed as:

$$\min_{h,g}\max_{\phi} \underbrace{\mathbb{E}_{(\mathbf{x}_s,y_s)\sim S} L_{\text{clf}}(h\circ g(\mathbf{x}_s), y_s)}_{\text{Source Classification loss}} + \lambda \underbrace{\left[\mathbb{E}_{\mathbf{x}_s\sim S(x)} \log(\phi\circ g(x_s)) + \mathbb{E}_{\mathbf{x}_t\sim T(x)} \log(1-\phi\circ g(x_t))\right]}_{\text{Domain Adversarial Loss}}$$

The aforementioned loss consists of two parts: the source classification loss that corresponds to the first term of Theorem 1.1, and the domain adversarial loss that corresponds to the second term of Theorem 1.1. Specifically, the adaptation procedure is trained in an adversarial way, *that the representation learning function g learns features that are domain-invariant, while the critic function ϕ discriminates between features from source and target.* Through this adversarial training, the feature extractor g learns the domain-invariant feature from the source and target.

1.4 Distribution Matching and Statistical Divergence

We have introduced domain adversarial training that aims at learning a domain invariant representation through *distribution matching*, which is a popular tool in deep generative modeling (Kingma and Welling, 2013; Goodfellow et al., 2014) and transfer learning (Ganin et al., 2016).

In general, we have two distribution functions $\mathcal{P}(x)$ and $\mathcal{Q}(x)$ with $x \in \mathcal{X}$, $\mathcal{P}(x) \neq \mathcal{Q}(x)$. The distribution matching aims to learn a transformation function $g(x) : \mathcal{X} \to \mathcal{Z}$ such that maps $x \in \mathcal{X}$ into a new variable $z \in \mathcal{Z}$. Then the goal is to find a good transformation function g to achieve $\mathcal{P}(z) = \mathcal{Q}(z)$ on the latent space, $\forall z \in \mathcal{Z}$.

In practice, we adopt a statistical divergence D to measure the "similarity" of two distribution $D(\mathcal{P}\|\mathcal{Q})$. Generally, the statistical divergence requires:

1. The divergence is non-negative with $D(\mathcal{P}\|\mathcal{Q}) \geq 0$
2. $D(\mathcal{P}\|\mathcal{Q}) = 0$ if and only if identical distributions $\mathcal{P} = \mathcal{Q}$.

We list several popular statistical divergences for machine learning.

1.4.1 f-Divergence

The information theoretical f-divergence is characterized by a *convex* function f with $f(1) = 0$, then f-divergence of distribution $\mathcal{P}$ and $\mathcal{Q}$ (probability density function $p(x)$ and $q(x)$) is defined as:

$$D_f(\mathcal{P}\|\mathcal{Q}) = \int_x f\left(\frac{p(x)}{q(x)}\right) q(x)dx$$

We also propose several common f-divergence, shown in Tab. 1.1.

Table 1.1 – Popular f-divergence

Divergence Name	$f(x)$	$D_f(\mathcal{P}\|\mathcal{Q})$
Kullback-Leibler Divergence	$x\log(x)$	$D_{\text{KL}}(\mathcal{P}\|\mathcal{Q}) = E_p[\log(\frac{p(x)}{q(x)})]$
Total Variation	$\frac{1}{2}\|x-1\|$	$\frac{1}{2}\int_x \|p(x)-q(x)\|$
χ^2 Divergence	$(x-1)^2$	$\int_x \frac{(p(x)-q(x))^2}{q(x)}$
Squared Hellinger distance	$(1-\sqrt{x})^2$	$\int_x(\sqrt{p(x)}-\sqrt{q(x)})^2$
Jensen-Shannon Divergence	$x\log(\frac{2x}{x+1})+\log(\frac{2}{x+1})$	$D_{\text{KL}}(\mathcal{P}\|\frac{\mathcal{P}+\mathcal{Q}}{2})+D_{\text{KL}}(\mathcal{Q}\|\frac{\mathcal{P}+\mathcal{Q}}{2})$

Variational f-divergence One advantage of introducing f-divergence is that it can be efficiently estimated through the variational form. Specifically, if we define the convex conjugates of f as

$$f^{\star}(y) = \sup_x\,[xy - f(x)]$$

Then the variational term of f-divergence can be expressed as:

$$D_f(\mathcal{P}\|\mathcal{Q}) = \sup_{d:\mathcal{X}\to\mathbb{R}} \mathbb{E}_{\mathcal{P}}[d(x)] - \mathbb{E}_{\mathcal{Q}}[f^{\star}(d(x))].$$

Example (TV distance) By using $f(x) = \frac{1}{2}|x-1|$, we have

$$d_{\mathrm{TV}}(\mathcal{P}\|\mathcal{Q}) = \sup_{d} \mathbb{E}_{\mathcal{P}}[d(x)] - \mathbb{E}_{\mathcal{Q}}[d(x)].$$

Therefore, in deep learning approaches, f-divergence can be efficiently estimated by introducing a statistical critic function d to maximize the dual term of f-divergence.

1.4.2 Integral Probability Metrics: IPM

Another popular distribution similarity metric is the Integral Probability Metrics (IPM). Given two probabilities $\mathcal{P}$ and $\mathcal{Q}$, and a real valued function family $f \in \mathcal{F}$, then IPM can be defined as:

$$\mathrm{IPM}(\mathcal{P}\|\mathcal{Q}) = \sup_{f\in\mathcal{F}} |\mathbb{E}_{\mathcal{P}} f(x) - \mathbb{E}_{\mathcal{Q}} f(x)|.$$

We also propose several common IPM metrics on different families $\mathcal{F}$.

- Wasserstein Distance (Villani, 2009). If we set f as 1-Lipschitz function with $|f(x) - f(y)| \leq \|x-y\|_2, \forall x, y$, denoted as $\mathcal{F} := \{f : \|f\|_L \leq 1\}$. Then the IPM is equivalent to the Wasserstein-1 distance.
- Total Variation Distance. If the function is bounded by 1 with $\sup_{x\in\mathcal{X}} |f(x)| \leq 1$, then IPM recovers the Total Variation distance.
- Maximum Mean Discrepancy (MMD) (Gretton et al., 2012). We set $\mathcal{F} = \{f : \|f\|_{\mathcal{H}} \leq 1\}$, where $\mathcal{H}$ represents a reproducing kernel Hilbert space (RKHS). Then IPM recovers the Maximum Mean Discrepancy.

1.4.3 Hypothesis-Based Divergence

Apart from the statistical divergence, an alternative way is to adopt a hypothesis family $\mathcal{H}$ to estimate the divergence between two distributions such as the aforementioned $\mathcal{H}$-divergence. In this part, we will introduce other popular hypothesis-based divergences.

Discrepancy Distance Analogue to $\mathcal{H}$-divergence, the discrepancy distance can be estimated from finite samples with data-dependent bound on Rademacher Complexity. In addition, compared with $\mathcal{H}$-divergence, the discrepancy distance can be used for comparing distributions for more general scenarios, including regression and general loss functions.

Definition 1.2 (Mansour et al. (2009a); Cortes et al. (2019))**.** Let $\mathcal{H}$ be a set of functions mapping the input $x \in \mathcal{X}$ to output $y \in \mathcal{Y}$ with $h \in \mathcal{H}$, and let $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ be the loss function. The discrepancy distance disc_L between two distributions $\mathcal{P}$ and $\mathcal{Q}$ is:

$$\mathrm{disc}_L(\mathcal{P}, \mathcal{Q}) = \max_{h,h^{'}\in\mathcal{H}} |\mathbb{E}_{x\sim\mathcal{P}} L(h(x), h^{'}(x)) - \mathbb{E}_{x\sim\mathcal{Q}} L(h(x), h^{'}(x))|$$

Then we can prove the uniform convergence property of discrepancy distance.

Proposition 1.2 (Mansour et al. (2009a)). *Let $\mathcal{H}$ be a hypothesis set bounded by some $M > 0$ for the loss function is q-quadratic loss with $L_q(h(x), h^{'}(x)) = |h(x) - h^{'}(x)|^q$ and $L_q(h, h^{'}) \leq M$, for all hypothesis pair $h, h \in \mathcal{H}$. Let $\mathcal{P}$ be a distribution over $\mathcal{X}$ and P the corresponding empirical distribution, and let $\mathcal{Q}$ be a distribution over $\mathcal{X}$ and Q the corresponding empirical distribution. Then for any $\delta > 0$, with probability at least $1 - \delta$ over samples $\hat{P}$ of size m drawn according to $\mathcal{P}$ and $\hat{Q}$ of size n drawn according to $\mathcal{Q}$:*

$$\text{disc}_{L_q}(\mathcal{P}, \mathcal{Q}) \leq \text{disc}_{L_q}(\hat{P}, \hat{Q}) + 4q(\hat{R}_P(\mathcal{H}) + \hat{R}_Q(\mathcal{H})) + 3M\left(\sqrt{\frac{\log(4/\delta)}{2m}} + \sqrt{\frac{\log(4/\delta)}{2n}}\right),$$

Where $\hat{R}_P(\mathcal{H})$ represents the Rademacher Complexity w.r.t. the empirical observation $P = (x_1, \ldots, x_m)$, which is defined as $\hat{R}_P(\mathcal{H}) = \frac{2}{m}\mathbb{E}_{\sigma_i}[\sup_{h\in\mathcal{H}} | \sum_{i=1}^m \sigma_i h(x_i)| \, |P = (x_1, \ldots, x_m)]$. Where σ_i is the Rademacher variable that follows Bernoulli distribution with probability 0.5: $\sigma_i \sim \text{Ber}(0.5)$.

1.4.4 Generalized Distribution Matching Network

Apart from introducing the auxiliary (adversarial) task to learn domain invariant features, one can directly learn the domain invariant feature by distribution matching. In fact, the Theoretical result in Theorem 1.1 can be extended to other divergences with the following form:

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + d(\mathcal{S}, \mathcal{T}) + \beta,$$

where $R_{\mathcal{S}}(h)$ represents the source prediction risk. $d(\cdot, \cdot)$ represents several metrics such as Wasserstein distance (Shen et al., 2017), Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) and discrepancy distance (Mansour et al., 2009a). The β term represents the discrepancy of source-target ground-truth labeling function, which can be viewed as a sufficient condition for ensuring a successful DA algorithm. Inspired by this general domain adaptation theory, we can design the generalized distribution matching network, shown in Fig 1.9.

The matching network consists of two parts: the feature extractor g that maps the data into the latent space $z = g(x)$, the classifier h that predicts the output $y = h(z)$. Therefore, the loss for generalized distribution matching can be expressed as:

$$\min_{h,g} \underbrace{\mathbb{E}_{(\mathbf{x}_s,y_s)\sim\mathcal{S}} L_{\text{clf}}(h \circ g(\mathbf{x}_s), y_s)}_{\text{Source Classification loss}} + \lambda \underbrace{d(\mathcal{S}(g(x)) \| \mathcal{T}(g(x)))}_{\text{Distribution Matching Loss}},$$

where d represents distribution distance metrics which measure the similarity of two distributions on the latent space $\mathcal{Z}$, and $\lambda > 0$ is the hyper-parameter. For example,

- If we adopt MMD (Gretton et al., 2012) with a specific kernel $K(\cdot, \cdot)$, then we have

$$\text{MMD}^2(\mathcal{S}\|\mathcal{T}) = E_{x,x'\sim\mathcal{S}}\, K(x, x') + E_{y,y'\sim\mathcal{T}}\, K(y, y') - 2E_{x\sim\mathcal{S},y\sim\mathcal{T}}\, K(x, y),$$

- If we adopt TV distance, then we have:

$$d_{\text{TV}}(\mathcal{S}\|\mathcal{T}) = \|\mathcal{S}(X) - \mathcal{T}(X)\|_1 = \int_x |\mathcal{S}(x) - \mathcal{T}(x)| dx,$$

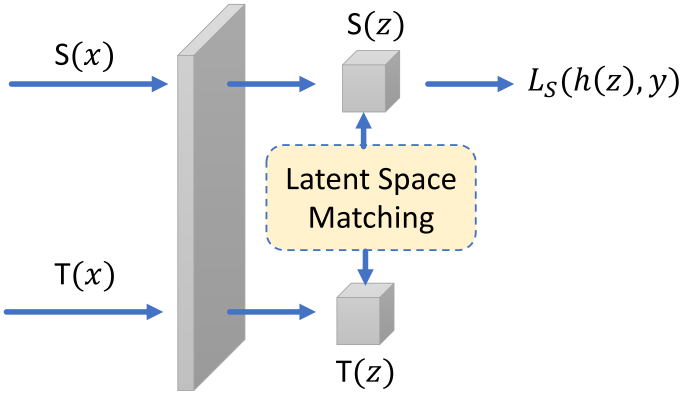


Figure 1.9 – Generalized Distribution Matching Network. The generalized matching aims to match the source and target distribution in the embedding space.

In general, these distribution metrics can be directly estimated from the data. Therefore, the deep learning algorithms aim at align the source and target features to learn the domain-invariant representation.

Chapter 2

Principled Approach for Learning Task Similarity in MTL

Original title of the article: **A Principled Approach for Learning Task Similarity in Multitask Learning**

Résumé

Dans ce chapitre, nous proposons un nouveau point de vue théorique pour comprendre la similarité des tâches dans l'apprentissage multitâche. Nous proposons d'abord une borne supérieure sur l'erreur de généralisation de l'apprentissage multitâche, en montrant l'avantage de la explicite et implicite de la similarité des tâches. Nous dérivons systématiquement les bornes supérieures du risque cible en nous basant sur deux métriques de similarité des tâches : la divergence $\mathcal{H}$ et la distance de wasserstein. À partir des résultats théoriques, nous revisitons le réseau neuronal multi-tâches adversarial, en proposant un nouvel algorithme pour apprendre les coefficients de relation de tâche et les paramètres du réseau de manière itérative. Nous évaluons le nouvel algorithme sur plusieurs benchmarks, ce qui indique non seulement des relations de tâches robustes, mais aussi des performances supérieures à celles des baselines.

Abstract

In this chapter, we propose a novel theoretical view in understanding task similarity in Multi-Task Learning. We first provide an upper bound on the generalization error of multitask learning, showing the benefit of explicit and implicit task similarity knowledge. We systematically derive the target risk upper bounds based on two task similarity metrics: $\mathcal{H}$-divergence and Wasserstein distance. From the theoretical results, we revisit the Adversarial Multi-Task Neural Network, proposing a new training algorithm to learn the task relation coefficients and neural network parameters iteratively. We assess

the new algorithm on several benchmarks, which not only indicates robust task relations, but also outperforms the baselines.

2.1 Introduction

Understanding the theoretical assumptions of the task relationship plays a key role in designing a good MTL algorithm. It determines which *inductive bias* should be involved during the learning. Recently, many successful algorithms relied on task similarity information, which assumes the *Probabilistic Lipschitzness* (PL) condition (Urner and Ben-David, 2013) as the inductive bias. For instance, Murugesan and Carbonell (2017); Murugesan et al. (2016); Pentina and Lampert (2017) minimized a weighted sum of empirical loss in which similar tasks are assigned similar weights. These approaches explicitly estimate the task similarities through a linear model. Since these approaches are estimated in the original input space, it is difficult to handle the large covariate shift problem, e.g source and target have different support. Therefore, many neural network based approaches started to explore implicitly tasks similarities: Liu et al. (2017); Li et al. (2018b) used adversarial loss by feature adaptation, minimizing the distribution distance between the tasks to construct a shared feature space. Then, the hypothesis for the different tasks are learned over this adapted feature space.

The implicit similarity learning approaches are inspired from the idea of Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). However, the fundamental implications of incorporating task similarity information in MTL algorithms are unclear. The two main questions remain elusive: why should we combine explicit and implicit similarity knowledge in the MTL framework and how can we properly handle it.

Previous works either consider explicit or implicit similarity knowledge separately, or heuristically combine these two in some specific applications. In contrast, the main goal in this chapter is to (1) *propose a rigorous analysis of the benefits of task similarities, (2) derive an algorithm which properly uses this information. We start by deriving an upper bound on the generalization error of MTL under different similarity metrics (or adversarial loss).* These theoretical results show the benefits of using adversarial loss in MTL: control the generalization error in MTL. Then we derive a new procedure to update the relationship coefficients from these theoretical guarantees. This procedure allows us to bridge the gap between the explicit and implicit similarities, which have been previously seen as disjointed or treated in an ad hoc manner. We derive a new algorithm to train the Adversarial Multitask Neural Network (AMTNN) and empirically validate on two benchmarks: digit recognition and Amazon sentiment analysis. The results show that our method not only highlights some interesting relations, but also outperforms the previous baselines, reaffirming the benefits of the theory in algorithm design.

2.2 Problem Set-up

Considering a set of T tasks $\{\hat{\mathcal{D}}_t\}_{t=1}^T$, in which the observations are generated by the underlying distribution $\mathcal{D}_t$ over $\mathcal{X}$ and the real target is determined by the underlying labeling functions $f_t : \mathcal{X} \to \mathcal{Y}$ for $\{(\mathcal{D}_t, f_t)\}_{t=1}^T$. Then, the goal of MTL is to find T hypothesis: $h_1, \ldots, h_T$ over the hypothesis space $\mathcal{H}$ to control the average expected error of all tasks:

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h_t),$$

where $R_t(h_t) \equiv R_t(h_t, f_t) = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_t}\ell(h_t(\mathbf{x}), f_t(\mathbf{x}))$ is the expected risk at task t and ℓ is the loss function. Throughout the theoretical part, the loss is $\ell(h_t(\mathbf{x}), f_t(\mathbf{x})) = |h_t(\mathbf{x}) - f_t(\mathbf{x})|$, which is coherent with (Pentina and Lampert, 2017; Li et al., 2018b; Ben-David et al., 2010a; Ganin et al., 2016; Redko et al., 2017). If h, f are the binary mappings with output in $\{-1, 1\}$, it recovers the typical zero-one loss.

We also assume that each task has m_t examples, with $\sum_{t=1}^T m_t = m$ examples in total. Then for each task t, we consider a minimization of weighted empirical loss for each task. That means we define a simplex $\boldsymbol{\alpha}_t \in \Delta^T = \{\boldsymbol{\alpha}_{t,i} \geq 0,\ \sum_{i=1}^T \boldsymbol{\alpha}_{t,i} = 1\}$ for the corresponding weight for task t. Then the weighted empirical error w.r.t. the hypothesis h for task t can be written as:

$$\hat{R}_{\boldsymbol{\alpha}_t}(h) = \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\hat{R}_i(h),$$

where $\hat{R}_i(h) = \frac{1}{m_i}\sum_{j=1}^{m_i}\ell(h(\mathbf{x}_j), \mathbf{y}_j)$ is the average empirical error for task i. Intuitively, in the proposed loss, each task risk is controlled by a weighted sum of other tasks. By leveraging the other tasks, the current task can be learned with limited data.

2.2.1 Similarity Measures

As we illustrated in the previous section, we are interested in task similarity metrics in MTL. Therefore, the first element to determine is how to measure the similarity between two distributions. For this, we introduce two metrics: $\mathcal{H}\Delta\mathcal{H}$-divergence (Ben-David et al., 2010a) and Wasserstein distance (Arjovsky et al., 2017), which are widely used in machine learning.

$\mathcal{H}\Delta\mathcal{H}$-divergence Given an input space $\mathcal{X}$ and two probability distributions $\mathcal{D}_i$ and $\mathcal{D}_j$ over $\mathcal{X}$, let $\mathcal{H}$ be a hypothesis class on $\mathcal{X}$. We define the $\mathcal{H}\Delta\mathcal{H}$-divergence of two distributions as

$$d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j) = \sup_{h,h'\in\mathcal{H}} |R_i(h, h') - R_j(h, h')|.$$

The empirical $\mathcal{H}\Delta\mathcal{H}$-divergence corresponds to:

$$d_{\mathcal{H}\Delta\mathcal{H}}(\hat{\mathcal{D}}_i, \hat{\mathcal{D}}_j) = \sup_{h,h'\in\mathcal{H}} |\hat{R}_i(h, h') - \hat{R}_j(h, h')|.$$

Wasserstein distance We assume $\mathcal{X}$ is the measurable space and denote $\mathcal{P}(\mathcal{X})$ as the set of all probability measures over $\mathcal{X}$. Given two probability measures $\mathcal{D}_i \in \mathcal{P}(\mathcal{X}_1)$ and $\mathcal{D}_j \in \mathcal{P}(\mathcal{X}_2)$, the *optimal transport* (or Monge-Kantorovich) problem can be defined as searching for a probabilistic coupling γ refined as a joint probability measure over $\mathcal{X}_1 \times \mathcal{X}_2$ with marginals $\mathcal{D}_i$ and $\mathcal{D}_j$ for all $\mathbf{x}, \mathbf{y}$ that are minimizing the cost of transport w.r.t. some cost function c:

$$\begin{aligned}&\text{argmin}_{\gamma} \int_{\mathcal{X}_1 \times \mathcal{X}_2} c(\mathbf{x},\mathbf{y})^p d\gamma(\mathbf{x},\mathbf{y}),\\ &\text{s.t.} \quad \mathbf{P}^{\mathcal{X}_1}\#\gamma = \mathcal{D}_i; \quad \mathbf{P}^{\mathcal{X}_2}\#\gamma = \mathcal{D}_j,\end{aligned}$$

where $\mathbf{P}^{\mathcal{X}_1}$ is the projection over $\mathcal{X}_1$ and $\#$ denotes the push-forward measure. The Wasserstein distance of order p between $\mathcal{D}_i$ and $\mathcal{D}_j$ for any $p \geq 1$ is defined as:

$$W_p^p(\mathcal{D}_i, \mathcal{D}_j) = \inf_{\gamma \in \Pi(\mathcal{D}_i, \mathcal{D}_j)} \int_{\mathcal{X}_1 \times \mathcal{X}_2} c(\mathbf{x},\mathbf{y})^p d\gamma(\mathbf{x},\mathbf{y}),$$

where $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is the cost function of transportation of one unit of mass $\mathbf{x}$ to $\mathbf{y}$ and $\Pi(\mathcal{D}_i, \mathcal{D}_j)$ is the collection of all joint probability measures on $\mathcal{X} \times \mathcal{X}$ with marginals $\mathcal{D}_i$ and $\mathcal{D}_j$. Throughout this paper, we only consider the case of $p = 1$, i.e., Wasserstein-1 distance.

2.3 Theoretical Guarantees in Multi-Task Learning

Based on the definition of the distribution similarity metric, we demonstrate that the generalization error in multitask learning can be upper bounded by the following result:

Theorem 2.1. *Let $\mathcal{H}$ be a hypothesis family with VC dimension d. Assume T tasks generated by the underlying distribution and labeling function $\{(\mathcal{D}_1, f_1), \dots, (\mathcal{D}_T, f_T)\}$ with observation numbers $m_1, \dots, m_T$. If we adopt $\mathcal{H}$ divergence as a similarity metric, then for any fixed simplex $\boldsymbol{\alpha}_t \in \mathbb{R}_+^T$, and for $\delta \in (0,1)$, with probability at least $1-\delta$, for $h_1, \dots, h_T \in \mathcal{H}$, we have:*

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T} R_t(h_t) \leq \; & \underbrace{\frac{1}{T}\sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(h_t)}_{\textit{Weighted empirical loss}} \; \underbrace{+ C_1 \sum_{t=1}^{T}\Big(\sqrt{\sum_{i=1}^{T}\frac{\boldsymbol{\alpha}_{t,i}^2}{\beta_i}}\Big)}_{\textit{Coefficient regularization}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t, \mathcal{D}_i)}_{\textit{Empirical distribution distance}}\\ + \quad & \underbrace{C_2 + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\lambda_{t,i}}_{\textit{Complexity term and optimal expected loss}}\end{aligned}$$

Where $\beta_i = \frac{m_i}{m}$, $C_1 = 2\sqrt{\frac{2(d\log(\frac{2em}{d})+\log(\frac{16T}{\delta}))}{m}}$ and $C_2 = 2\min_{i,j}\sqrt{\frac{2d\log(2m_{i,j})+\log(32T/\delta)}{m_{i,j}}}$ with $m_{i,j} = \min\{m_i, m_j\}$, and $\lambda_{i,j} = \inf_{h\in\mathcal{H}}\{R_i(h) + R_j(h)\}$ the joint expected minimal error *w.r.t. $\mathcal{H}$.*

Discussion Theorem 1 illustrates that the upper bound on the generalization error in our MTL settings can be decomposed into the following terms:

1. The empirical loss and empirical distribution similarities control the weights (or task relation coefficient) $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_T$. For instance, for a given task t, if task i has a small empirical distance $d_{\mathcal{H}\Delta\mathcal{H}}(\hat{\mathcal{D}}_t, \hat{\mathcal{D}}_i)$ and hypothesis h_t has a small empirical loss $\hat{R}_i(h_t)$ on task i, it means that task i is very similar to t. Hence, more information should be borrowed from task i when learning t and the corresponding coefficient $\boldsymbol{\alpha}_{t,i}$ should have high values.
2. Simultaneously, the *coefficient regularization term* prevents the relation coefficients locating only on the $\boldsymbol{\alpha}_{t,t}$, in which it will completely recover the independent MTL framework. Then the coefficient regularization term proposed a trade-off between learning the single task and sharing information from the others tasks.
3. The complexity and optimal terms depend on the setting hypothesis family $\mathcal{H}$. Given a fixed hypothesis family such as neural network, the complexity is constant. As for the optimal expected loss, throughout this paper we assume $\lambda_{t,i}$ is *much smaller* than the empirical term, which indicates that the hypothesis family $\mathcal{H}$ can learn the multiple tasks with a small expected risk. This is a natural setting in the MTL problem since we want the predefined hypothesis family to learn well for all of the tasks. While a high expected risk means such a hypothesis set cannot perform well, which contradicts our assumption.

Proof Sketch The complete proof is delegated in Appendix Sec.A.2, which consists of three main steps.

1. *Expected Transfer Risk.* We bound the expected risk in MTL by using the task similarity information.

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h_t) \leq \frac{1}{T}\sum_{t=1}^{T} R_{\boldsymbol{\alpha}_t}(h_t) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(D_t, D_i) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\lambda_{t,i}$$

2. *Generalization bound expected and empirical $\mathcal{H}$-divergence.* In the second step, we upper bound the expected $\mathcal{H}$-divergence, which is related to the empirical $\mathcal{H}$-divergence and a sample complexity term.

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t, \mathcal{D}_i) \leq \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{\alpha}_{t,i} \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t, S_i) + 2\sqrt{\frac{2d\log(2m_\star) + \log(32T/\delta)}{m_\star}}$$

Where $m_\star = \text{argmin}_{m_{i,j}} \sqrt{\frac{2d\log(2m_{i,j})+\log(32T/\delta)}{m_{i,j}}}$

3. *Generalization bound expected and empirical risk.* In the third step, the expected risk can be upper bounded by its empirical counterpart and a sample complexity term.

$$\frac{1}{T}\sum_{t=1}^{T} R_{\boldsymbol{\alpha}_t}(h_t) \leq \frac{1}{T}\sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(h_t) + 2\sqrt{\frac{2(d\log(\frac{2em}{d}) + \log(\frac{16T}{\delta}))}{m}} \sum_{t=1}^{T} \Big(\sqrt{\sum_{j=1}^{N} \frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\Big)$$

Combining the results in steps 1-3, we have the aforementioned theoretical results.

In theorem 2.1, we have derived a bound based on the $\mathcal{H}$ divergence and applied for the classification. Then we proposed another bound based on the Wasserstein distance, which can be applied in the classification and regression problem.

Theorem 2.2. *Let $\mathcal{H}$ be a hypothesis family from $\mathcal{X}$ to $[0,1]$, with pseudo-dimension d and each member $h \in \mathcal{H}$ is K Lipschitz. Consider T tasks generated by the underlying distribution and labeling function $\{(\mathcal{D}_1, f_1), \dots, (\mathcal{D}_T, f_T)\}$ with observation numbers $m_1, \dots, m_T$. If we adopt Wasserstein-1 distance as a similarity metric with cost function $c(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2$, then for any simplex $\boldsymbol{\alpha}_t \in \Delta^T$, and for $\delta \in (0,1)$, with a probability at least $1-\delta$, for $h_1, \dots, h_T \in \mathcal{H}$, we have:*

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h_t) \leq \underbrace{\frac{1}{T}\sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(h_t)}_{\textit{Weighted empirical loss}} + \underbrace{C_1 \sum_{t=1}^{T}\Big(\sqrt{\sum_{j=1}^{T}\frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\Big)}_{\textit{Coefficient regularization}} + \underbrace{\frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i} W_1(\hat{D}_t, \hat{D}_i)}_{\textit{Empirical distribution distance}}$$

$$+ \underbrace{C_2 + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\lambda_{t,i}}_{\textit{Complexity term and optimal expected loss}}$$

where $\beta_i = \frac{m_i}{m}$, $C_1 = 2\sqrt{\frac{2(d\log(\frac{2em}{d})+\log(\frac{16T}{\delta}))}{m}}$, $C_2 = \frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\gamma_{t,i}$ with $\gamma_{t,i} = \mu_t m_t^{-1/s} + \mu_i m_i^{-1/s} + \sqrt{\log(\frac{2T}{\delta})}(\sqrt{\frac{1}{m_t}} + \sqrt{\frac{1}{m_i}})$ and s and $\mu_{\cdot}$ are some specified constants. and $\lambda_{i,j} = \inf_{h\in\mathcal{H}}\{R_i(h) + R_j(h)\}$ the joint expected minimal error.

The proof w.r.t. the Wasserstein-1 distance is analogous to the proof in the $\mathcal{H}$-divergence but under different assumptions. The complete proof is delegated in Appendix Sec. A.3.

Remark The MTL generalization error bound indicates that we should not only minimize the weighted empirical loss, but also minimize the empirical distribution divergence between each task. Based on this, in the neural networks based MTL, these conclusions provide a theoretical support for understanding the role of *adversarial losses*, which exactly minimize the distribution divergence.

2.4 Adversarial Multitask Neural Network (AMTNN)

From the generalization error upper bound, we developed a new training algorithm for the Adversarial Multitask Neural Network (AMTNN). It consists of multiple training steps by iteratively optimizing the parameters in the neural network for a given fixed relation coefficient $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_T$ and estimating the relation coefficient, given fixed neural network weights.

Moreover, we have three types of parameters in AMTNN: $\boldsymbol{\theta}^f$, $\boldsymbol{\theta}^d_{\cdot}$ and $\boldsymbol{\theta}^h_{\cdot}$, corresponding to the parameter for feature extractor, adversarial loss (distribution similarity) and task prediction loss, respectively.

To simplify the problem, we assume that each task has the same number of observations, i.e., $\beta_i = \frac{1}{T}$, and the regularization will recover l_2 norm of $\|\boldsymbol{\alpha}_t\|_2$.

2.4.1 Neural Network Parameters Update

Given a fixed $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_T$, according to the theoretical bound, we want to minimize the weighted empirical error $\frac{1}{T}\sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t)$ and the empirical distribution "distance" $\hat{d}(\mathcal{D}_t, \mathcal{D}_i)$ with $t, i = 1, \dots, T$. Inspired by (Ganin et al., 2016), the minimization of the distribution "distance" is equivalent to the maximization of the adversarial loss $\hat{E}_{t,i}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^d_{t,i})$ (defined below). Overall, we have the following loss function with a trade-off coefficient ρ:

$$\min_{\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_1, \dots, \boldsymbol{\theta}^h_t} \max_{\boldsymbol{\theta}^d_{t,i}} \sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t) + \rho \sum_{i,t=1}^{T} \boldsymbol{\alpha}_{t,i} \hat{E}_{t,i}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^d_{t,i}). \tag{2.1}$$

It should be noted that for a given task t, the sum loss is $\frac{1}{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} \sum_{\mathbf{x} \in \hat{D}_i} \ell((\mathbf{x}, y), \boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t)$, with ℓ being the cross entropy loss. This means that the empirical loss is a weighted sum of all of the task losses, determined by task relation coefficient $\boldsymbol{\alpha}_t$. This is coherent with (Murugesan et al., 2016), which does not provide theoretical explanations.

In addition, the adversarial loss $\hat{E}_{t,i}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^d_{t,i})$ is a symmetric metric for which we need to compute $\hat{E}_{t,i}$ only for $t < i$. Motivated by (Ganin et al., 2016), the neural network will output for a pair of observed *unlabeled* tasks $(\hat{\mathcal{D}}_t, \hat{\mathcal{D}}_i)$ a score in $[0, 1]$ to predict from which distribution it comes. Supposing the output function is $g_{t,i}(\mathbf{x}, (\boldsymbol{\theta}^f, \boldsymbol{\theta}^d_{t,i})) \equiv g_{t,i}(\mathbf{x})$, the adversarial loss will be the following under different distance metrics:

$\mathcal{H}\Delta\mathcal{H}$ divergence We adopt the $\mathcal{H}$-divergence as the approximation of $\mathcal{H}\Delta\mathcal{H}$ divergence:

$$\hat{E}_{t,i} = \sum_{\mathbf{x} \in \hat{\mathcal{D}}_t} \log(g_{t,i}(\mathbf{x})) + \sum_{\mathbf{x} \in \hat{D}_i} \log(1 - g_{t,i}(\mathbf{x}));$$

Wasserstein-1 distance Since the primal form has a high computational complexity, we adopted the same strategy as (Arjovsky et al., 2017) by estimating the empirical Kantorovich-Rubinstein duality of Wasserstein-1 distance, which is equivalent to

$$W_1(\hat{D}_t, \hat{D}_i) = \frac{1}{K} \sup_{\|f\| \leq K} (\mathbb{E}_{\mathbf{x} \in \hat{\mathcal{D}}_t}[g_{t,i}(\mathbf{x})] - \mathbb{E}_{\mathbf{x} \in \hat{\mathcal{D}}_i}[g_{t,i}(\mathbf{x})]).$$

Combining with the result of Theorem 2.2, we can derive

$$\hat{E}_{t,i} = \mathbb{E}_{\mathbf{x} \in \hat{\mathcal{D}}_t}[g_{t,i}(\mathbf{x})] - \mathbb{E}_{\mathbf{x} \in \hat{\mathcal{D}}_i}[g_{t,i}(\mathbf{x})].$$

2.4.2 Task Relation Coefficients Estimation

After updating the neural network parameters, we need to reestimate the coefficients $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_T$ when giving fixed $\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_{\cdot}, \boldsymbol{\theta}^d_{\cdot}$. According to the theoretical guarantees, we need to solve the following convex constraint optimization problem.

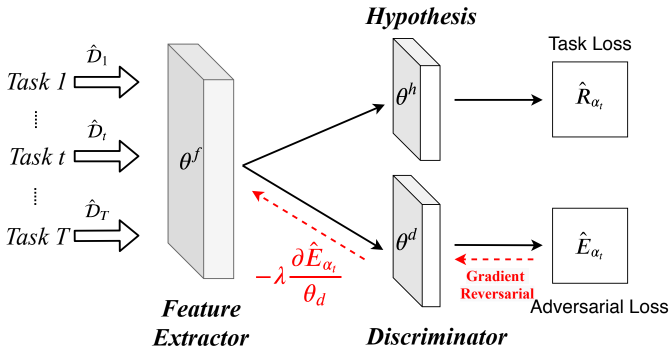


Figure 2.1 – General framework of Adversarial Multitask Neural Network (AMTNN).

$$\begin{aligned}
&\min_{\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_T} \sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t) + \kappa_1 \sum_{t=1}^{T} \|\boldsymbol{\alpha}_t\|_2 + \kappa_2 \sum_{i,t=1}^{T} \boldsymbol{\alpha}_{t,i} \hat{d}_{t,i}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^d_{t,i}), \\
&\text{s.t.} \quad \|\boldsymbol{\alpha}_t\|_1 = 1, \quad \boldsymbol{\alpha}_{t,i} \geq 0 \ \ \forall t, i,
\end{aligned} \tag{2.2}$$

where κ_1 and κ_2 are hyper-parameters and $\hat{d}_{\cdot}$ is the estimated distribution "distance". This distribution "distance" may have different forms according to the similarity metric used:

1. **$\mathcal{H}\Delta\mathcal{H}$-divergence.** According to (Pentina and Lampert, 2017; Ben-David et al., 2010a; Ganin et al., 2016), the distribution "distance" is proportional to the accuracy of the discriminator $\boldsymbol{\theta}^d_{\cdot}$, i.e., we applied $g_{t,i}(\mathbf{x})$ to predict $\mathbf{x}$ coming from distribution t or i. The prediction accuracy reflects the difficulty to distinguish two distributions. Hence, we set $\hat{d}_{t,i}$ as the accuracy of the discriminator $g_{t,i}(\mathbf{x})$; We notice the adversarial training is essentially the approximation of $\mathcal{H}$-divergence, which is slightly different from $\mathcal{H}\Delta\mathcal{H}$ divergence.
2. **Wasserstein-1 distance.** According to (Arjovsky et al., 2017), the approximation $\hat{d}_{t,i} = -\hat{E}_{t,i}$ is adopted.

We also assume $\hat{d}_{t,t} = 0$ since the discriminator cannot distinguish two identical distributions. Moreover, the expected loss $\boldsymbol{\alpha}_t \lambda_{t,\cdot}$ is omitted since we assume that $\lambda_{t,\cdot}$ is much smaller than the empirical term. Then, we only use the empirical parts to reestimate the relationship coefficient.

As it is mentioned in the theoretical part, the L_2 norm regularization aims at preventing all of the relation coefficients from being concentrated on the current task $\boldsymbol{\alpha}_{t,t}$. The theoretical bound proposes an elegant interpretation for training AMTNN, which is shown in Algorithm 1.

2.4.3 Proposed algorithm

The general framework of the neural network is shown in Fig. 2.1. We propose a complete iteration step on how to update the neural network parameters and relation coefficients in Algorithm 1. When

Algorithm 1 AMTNN updating algorithm

```
Require: Samples from different tasks $\{\hat{\mathcal{D}}_t\}_{t=1}^T$, initial coefficients $\{\boldsymbol{\alpha}_t\}_{t=1}^T$ and learning rate $\eta$
Ensure: Neural network parameters $\boldsymbol{\theta}^f$, $\boldsymbol{\theta}^h_{\cdot}$, $\boldsymbol{\theta}^d_{\cdot}$ and relationship coefficient $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_T$
for mini-batch of samples $\{(\mathbf{x}_t^b, \mathbf{y}_t^b)\}$ from $\{\hat{\mathcal{D}}_t\}_{t=1}^T$ do
    For each distribution pair $(t, i)$ with $t < i$, compute the adversarial loss $\hat{E}_{t,i}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^d_{t,i})$.
    For each task $t$, define the empirical loss matrix $\hat{R}_{t,i} = \sum_{(\mathbf{x}_i^b, \mathbf{y}_i^b) \in \hat{D}_i} \ell((\mathbf{x}_i^b, \mathbf{y}_i^b), \boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t)$ and
    compute the label loss:
        $$\hat{R}_{\boldsymbol{\alpha}_t} = \sum_{i=1}^T \boldsymbol{\alpha}_{t,i} \hat{R}_{t,i}$$
    Update $\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t$:
        $$\boldsymbol{\theta}^h_t = \boldsymbol{\theta}^h_t - \eta \frac{\partial \hat{R}_{\boldsymbol{\alpha}_t}}{\partial \boldsymbol{\theta}^h_t}$$
        $$\boldsymbol{\theta}^f = \boldsymbol{\theta}^f - \eta \left( \sum_{t=1}^T \frac{\partial \hat{R}_{\boldsymbol{\alpha}_t}}{\partial \boldsymbol{\theta}^f} + \sum_{t,i:t<i}^T (\boldsymbol{\alpha}_{t,i} + \boldsymbol{\alpha}_{i,t}) \frac{\partial \hat{E}_{t,i}}{\partial \boldsymbol{\theta}^f} \right)$$
    Update $\boldsymbol{\theta}^d_{t,i}$ $(t < i)$: $\boldsymbol{\theta}^d_{t,i} = \boldsymbol{\theta}^d_{t,i} + \eta \Big( (\boldsymbol{\alpha}_{t,i} + \boldsymbol{\alpha}_{i,t}) \frac{\partial \hat{E}_{t,i}}{\partial \boldsymbol{\theta}^d_{t,i}}) \Big)$
end for
Re-estimate $\{\boldsymbol{\alpha}_t\}_{t=1}^T$ by optimizing Eq. (2.2).
```

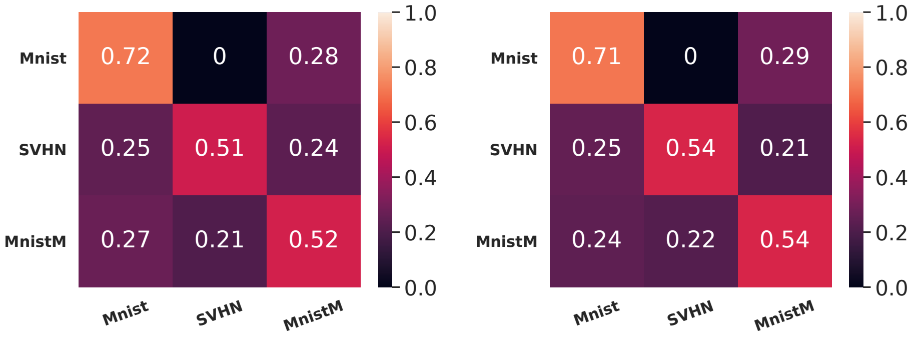
updating the feature extraction parameter $\boldsymbol{\theta}^f$, we applied *gradient reversal* (Ganin et al., 2016) in the training procedure. We also add the *gradient penalty* (Gulrajani et al., 2017) to improve the Lipschitz property when training with the adversarial loss based on Wasserstein distance.

It is worth mentioning that the proposed algorithm can potentially have the task collapsing: if one task is rather different from others, the proposed approach could have the risk of ignoring this task and learning the easier tasks. To this end, we need to assign a strong task coefficient regularization. By imposing a stronger regularization of the uniform weights, the different tasks are enforce to be learned for avoiding the task collapsing.

2.5 Experiments

We evaluate the modified AMTNN method on two benchmarks, that is the digits datasets and the Amazon sentiment dataset. We also consider the following approaches as baseline to make comparisons:

1. MTL_uni: the vanilla MTL framework where $\frac{1}{T}\sum_{t=1}^T \hat{R}_t(\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t)$ is minimized;
2. MTL_weighted: minimizing $\frac{1}{T}\sum_{t=1}^T \hat{R}_{\boldsymbol{\alpha}_t}(\boldsymbol{\theta}^f, \boldsymbol{\theta}^h_t)$, computation of $\boldsymbol{\alpha}_t$ depending on $\hat{R}_{t,i}$ (Murugesan et al., 2016);
3. MTL_disH and MTL_disW: we apply the same type of loss function but with two different adversarial losses ($\mathcal{H}\Delta\mathcal{H}$ divergence and Wasserstein distance) and a general neural network (Liu et al., 2017). It is worth mentioning that the original baselines are designed for the NLP



(a) AMTNN_W

(b) AMTNN_H

Figure 2.2 – Estimated task relation coefficients matrix (ranging from $0 - 1$) from the two proposed algorithms, with 8K samples on training set.

	3K				**5K**				**8K**			
Approach	**MNIST**	**MNIST_M**	**SVHN**	Average	**MNIST**	**MNIST_M**	**SVHN**	Average	**MNIST**	**MNIST_M**	**SVHN**	Average
MTL_uni	93.23	76.85	57.20	75.76	97.41	77.72	67.86	81.00	97.73	83.05	71.19	83.99
MTL_weighted	89.09	73.69	68.63	77.13	91.43	74.07	73.81	79.77	92.01	76.69	73.77	80.82
MTL_disH	89.91	**81.13**	70.31	80.45	91.92	82.68	73.27	82.62	92.96	**85.04**	78.50	85.50
MTL_disW	96.77	80.38	68.40	81.85	95.47	**83.48**	72.66	83.87	98.09	84.13	74.37	85.53
MTL_MOB	**97.54**	76.50	54.51	76.18	**98.22**	80.22	61.22	79.89	**98.48**	82.81	69.92	83.40
AMTNN_H	**97.47**	77.87	71.26	82.20	**97.94**	76.28	76.06	83.43	**98.28**	82.75	76.63	85.89
AMTNN_W	97.20	80.70	**76.93**	**84.95**	97.67	82.50	**76.36**	**85.51**	98.01	82.53	**79.97**	**86.84**

Table 2.1 – Average test accuracy (in $\%$) of MTL algorithms on the digits datasets.

application. I.e, the LSTM and GRU are adopted, we just simply replace it by MLP or CNN.

4. MTL_MOB: (Sener and Koltun, 2018) The deep multitask learning problem as multi-objective optimization;
5. AMTNN_H and AMTNN_W: proposed approaches with two different adversarial losses, $\mathcal{H}$ divergence and Wasserstein distance respectively.

The additional experimental details can be found in Appendix Sec. A.4.

2.5.1 Digit recognition

We first evaluate our algorithm on three benchmark datasets of digit recognition: MNIST, MNIST_M Ganin et al. (2016), and SVHN. The MTL setting is to jointly allow a system to learn to recognize the digits from the three datasets, which can differ significantly. To show the effectiveness of MTL, only a small portion of the original dataset is used for training (i.e., 3K, 5K and 8K for each task).

We use the LeNet-5 architecture and define the feature extractor $\boldsymbol{\theta}^f$ as the two convolutional layers of the network, followed by multiple blocks of two fully connected layers as label prediction parameter $\boldsymbol{\theta}^h_{\cdot}$ and discriminator parameter $\boldsymbol{\theta}^d_{\cdot}$. Five repetitions are conducted for each approach, and the average test accuracy (%) is reported in Table 2.1. We also show the estimated coefficient $\{\boldsymbol{\alpha}_t\}_{t=1}^3$ of AMTNN_H and AMTNN_W, in Fig. 2.2.

	1000 examples					1600 examples				
Approach	**Book**	**DVDs**	**Kitchen**	**Elec**	Average	**Book**	**DVDs**	**Kitchen**	**Elec**	Average
MTL_uni	81.31	78.44	87.07	84.57	82.85	81.35	80.14	86.54	87.50	83.88
MTL_weighted	81.88	79.02	86.91	85.31	83.28	80.72	81.20	87.60	88.12	84.41
MTL_disH	81.23	78.12	87.34	84.82	82.88	**81.92**	79.86	87.79	87.31	84.22
MTL_disW	81.13	78.38	87.11	84.82	82.86	81.88	79.81	87.07	87.69	84.11
MTL_MOB	81.58	78.65	87.18	84.53	82.99	81.05	80.76	87.01	87.59	84.10
AMTNN_H	**82.36**	79.24	**87.42**	85.53	**83.64**	80.82	**81.54**	**88.27**	88.17	**84.70**
AMTNN_W	81.68	**79.38**	87.27	**85.66**	83.50	81.20	80.38	87.69	**88.46**	84.44

Table 2.2 – Average test accuracy (in %) of MTL algorithm in the sentiment dataset.

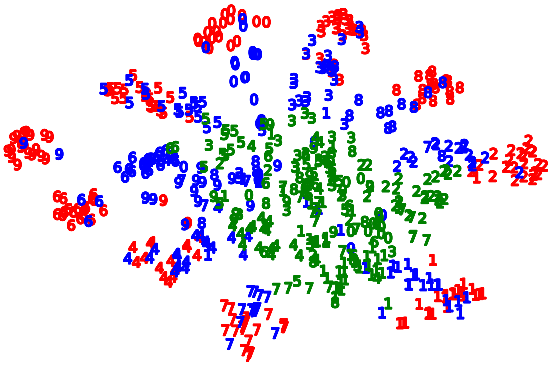


Figure 2.3 – t-SNE in the feature space of task MNIST in AMTNN_W for 8K. Red: MNIST dataset; blue: MNIST_M dataset; green: SVHN dataset.

Discussion Reported results show that the proposed approaches outperform all baselines in the task average and in most single tasks. Particularly for the AMTNN_W, it outperforms the baselines with $1.0\% \sim 2.9\%$ in the test accuracy. The reason can be that the Wasserstein-1 distance is more efficient for measuring the high dimensional distribution, which has been verified theoretically (Redko et al., 2017). Moreover, the $\mathcal{H}\Delta\mathcal{H}$ divergence-based approach (AMTNN_H) outperforms the baselines with increment performance ($< 0.3\%$). The reason may be that the VC-dimension with $\mathcal{H}\Delta\mathcal{H}$ divergence is not a good metric for measuring a high dimensional complex dataset, which is consistent with (Li et al., 2018b).

As for the coefficients $\boldsymbol{\alpha}_t$, the proposed algorithm appears robust in estimating these task relationships with almost identical values under different similarity metrics. Moreover, in contrast to the previous approaches, we obtain a non-symmetric matrix with a better interpretability. For instance, when learning the MNIST dataset, only information from MNIST_M is used, which is reasonable since these two tasks have the same digit configurations with different background, while SVHN is different in most ways (i.e., digits taken from street view house numbers). However, when learning MNIST_M, the information from SVHN is beneficial because it provides some information on the background, which is absent from MNIST but similar to MNIST_M. Therefore, the information of both tasks are involved in training for MNIST_M.

To show the role of the weighted sum, we use t-SNE to visualise in Fig. 2.3 the embedded space of the MNIST task from the training data. Information from SVHN is not relevant for learning MNIST as $\boldsymbol{\alpha}_{1,2} = 0$ (see Fig. 2.2), such that SVHN data is arbitrarily distributed in the embedded space without influence on the final result. At the same time, information from MNIST_M is used for training on the MNIST task ($\boldsymbol{\alpha}_{1,3} = 0.28$), which can be seen by a slight overlap in the embedded space. From that perspective, the role of weighted loss, which helps us to achieve some reasonable modifications of the decision boundary, is trained by the relevant and current tasks jointly. For a small scale task (typically the MTL scenario), during the test procedure, the agent predicts the labels by borrowing its neighbors (relevant task) information. This is coherent with the Probabilistic Lipschitzness condition (Urner and Ben-David, 2013).

2.5.2 Sentiment analysis

We also evaluate the proposed algorithm on *Amazon reviews* datasets. We extract reviews from four product categories: Books, DVD, Electronics and Kitchen appliances. Reviews datasets are preprocessed with the same strategy proposed by Ganin et al. (2016): 10K dimensional input features of uni-gram/bi-gram occurrences and binary output labels $\{0, 1\}$, Label 0 is given if the product is ranked less than 3 stars, otherwise label 1 is given for products above 3 stars. Results are reported for two sizes of labelled training sets, that is 1000 and 1600 examples in each product category.

The output of the first fully connected layers as feature extractor parameters $\boldsymbol{\theta}^f$ and several sets of two fully-connected layers are given as discriminator $\boldsymbol{\theta}_{\cdot}^d$ and label predictor $\boldsymbol{\theta}_{\cdot}^h$, with test accuracy $(\%)$ reported in Table 2.2 as an average over 5 repetitions.

Discussions We found the proposed approaches outperform all baselines in the task average and also in most tasks. Meanwhile, we observed that the role of adversarial loss (MTL_disH, MTL_disW, AMTNN_H and AMTNN_W) is not statistically significant (gains $< 0.25\%$), compared to the results on the digits datasets. The possible reason is that we applied the algorithm on the preprocessed feature instead of the original feature, making the discriminator $\boldsymbol{\theta}_{\cdot}^d$ less powerful in the feature adaptation. On the contrary, adding the weighted loss can improve performance by $0.4\% \sim 0.9\%$, enhancing the importance of the role of explicit similarity, which is coherent with (Murugesan et al., 2016).

2.6 Conclusion

In this chapter, we propose a principle approach for using the task similarity information in the MTL. We first derive an upper bound of the generalization error in the MTL. Inspired by the theoretical results, we design a new training algorithm on the Adversarial Multi-Task Neural Network (AMTNN). Finally, the empirical results on the benchmarks are showing that the proposed algorithm outperforms the baseline, reaffirming the benefits of theoretical insight in the algorithm design.

Chapter 3

Unified and Principled Method for Query and Training in Deep Active Learning

Original title of the article: **Deep active learning: unified and principled method for query and training.**

Résumé

Dans ce chapitre, nous proposons une méthode unifiée sur des principes pour les requêtes et les entraînements dans l'apprentissage actif profond par lots. Nous proposons un point de vue théorique à partir de l'intuition de modéliser la procédure interactive dans l'apprentissage actif comme *distribution matching*, en adoptant la distance de Wasserstein.

Nous avons dérivé une nouvelle perte de l'entraînement. De plus, la perte pour la formation d'un réseau neuronal profond est naturellement formulée comme un problème d'optimisation min-max en exploitant les informations des données non étiquetées. De plus, les principes proposés indiquent également un compromis incertitude-diversité dans la sélection des lots de requêtes. Enfin, nous évaluons notre méthode proposée sur différents benchmarks, qui démontrent des performances empiriques constamment meilleures et une stratégie de requête plus efficace en termes de temps, par rapport aux baselines.

Abstract

In this chapter, we propose a unified and principled method for both querying and training in deep batch active learning. We are providing theoretical insights from the intuition of modeling the interactive procedure in active learning as distribution matching, by adopting the Wasserstein distance.

We derived a new training loss from the theoretical analysis, which is decomposed into optimiz-

ing deep neural network parameters and batch query selection through alternative optimization. In addition, the loss for training a deep neural network is naturally formulated as a min-max optimization problem through leveraging the unlabeled data information. Moreover, the proposed principles also indicate an *explicit* uncertainty-diversity trade-off in the query batch selection. Finally, we evaluate our proposed method on different benchmarks, which demonstrates consistently better empirical performances and the better time-efficient query strategy compared to the baseline.

3.1 Introduction

As we previously discussed the query and training are two key components in deep Active Learning. We proposed a *unified and principled* approach for *both* a fast querying and a better training procedure in deep AL, relying on the use of labeled and unlabeled examples. We derived the theoretical analysis through modeling the interactive procedure in AL as *distribution matching* by adopting the Wasserstein distance. We further analytically reveal that the Wasserstein distance is better at capturing the diversity in AL, compared to the most common $\mathcal{H}$-divergence in the distribution matching. From the theoretical result, we derived the loss from the distribution matching, which is naturally decomposed into two stages: optimization of DNN parameters and query batch selection, through alternative optimization.

For the stage of training DNN, the derived loss indicates a min-max optimization problem by leveraging the unlabeled data. More precisely, this involves a maximization of the critic function to distinguish the labeled and unlabeled empirical distributions based on the Wasserstein distance, while the feature extractor function aims, on the contrary, to confound the distributions (minimization of empirical distribution divergence). In the query stage, the loss for batch selection *explicitly* indicates the uncertainty diversity trade-off. For the uncertainty, we want to find the samples with low prediction confidence over two different interpretations: the highest prediction confidence score and the uniform prediction score (Section 3.3.4). As for the diversity, we want to find the unlabeled batch holding a larger transport cost w.r.t. labeled set under Wasserstein distance (i.e. samples that looks different from the current labeled ones), which has been shown as a good metric for measuring diversity.

We tested our proposed method on different benchmarks, showing a consistently improved performance, particularly in the initial training, and a much faster query strategy compared to the baseline. The results reaffirmed the benefits and potential of deriving unified principles for Deep Active Learning. We also hope it will open up a new avenue for rethinking and designing query efficient and principled Deep Active Learning algorithms in the future.

3.2 Active Learning as Distribution Matching

In supervised learning, observations $\hat{\mathcal{D}}$ are i.i.d. generated by the underlying distribution $\mathcal{D}$ and a ground truth labeling function $h^{\star}$, i.e. $\{(x_i, h^{\star}(x_i))\}_{i=1}^{N}$ with $x_i \sim \mathcal{D}$. However in AL, the querying sample is not an i.i.d. procedure w.r.t. $\mathcal{D}$ — otherwise it will be simple random sampling. Thus we assume in AL that the query procedure is an i.i.d. empirical process from another distribution $\mathcal{Q} \neq \mathcal{D}$. For example, in the disagreement based approach (Balcan et al., 2009), $\mathcal{Q}$ can be regarded as a uniform distribution over the disagreement region. Then the interactive procedure can be viewed as estimating a proper $\mathcal{Q}$ to control the generalization error w.r.t. $(\mathcal{D}, h^\star)$.

3.2.1 Preliminaries

We define the hypothesis $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ over $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \in [0,1]$, and loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$. The expected risk w.r.t. $\mathcal{D}$ is $R_{\mathcal{D}}(h) = \mathbb{E}_{x\sim\mathcal{D}}\ell(h(x), h^\star(x))$ and empirical risk $\hat{R}_{\mathcal{D}}(h) = \frac{1}{N}\sum_{i=1}^{N}\ell(h(x_i), y_i)$. $\mathbb{P}(\mathcal{X})$ is the set of all probability measures over $\mathcal{X}$. We assume that the loss ℓ is symmetric, L-Lipschitz and M-upper bounded and $\forall h \in \mathcal{H}$ is at most H-Lipschitz function.

Wasserstein Distance Given two probability measures $\mathcal{D} \in \mathbb{P}(\mathcal{X})$ and $\mathcal{Q} \in \mathbb{P}(\mathcal{X})$ defined on Ω, the *optimal transport* (or Monge-Kantorovich) problem can be defined as searching for a probabilistic coupling (joint probability distribution) $\gamma \in \mathbb{P}(\Omega \times \Omega)$ for $x_{\mathcal{D}} \sim \mathcal{D}$ and $x_{\mathcal{Q}} \sim \mathcal{Q}$ that are minimizing the cost of transport w.r.t. some cost function c:

$$\begin{aligned}&\text{argmin}_\gamma \int_{\mathcal{X}\times\mathcal{X}} c(x_{\mathcal{D}}, x_{\mathcal{Q}})^p d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}),\\ &\text{s.t.} \quad \mathbf{P}^+\#\gamma = \mathcal{D}; \quad \mathbf{P}^-\#\gamma = \mathcal{Q},\end{aligned}$$

where $\mathbf{P}^+$ and $\mathbf{P}^-$ is the marginal projection over $\Omega \times \Omega$ and $\#$ denotes the push-forward measure. The p-Wasserstein distance between $\mathcal{D}$ and $\mathcal{Q}$ for any $p \geq 1$ is defined as:

$$W_p^p(\mathcal{D}, \mathcal{Q}) = \inf_{\gamma\in\Pi(\mathcal{D},\mathcal{Q})} \int_{\mathcal{X}\times\mathcal{X}} c(x_{\mathcal{D}}, x_{\mathcal{Q}})^p d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}),$$

where $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is the cost function of transportation of one unit of mass x to y and $\Pi(\mathcal{D}, \mathcal{Q})$ is the collection of all joint probability measures on $\mathcal{X} \times \mathcal{X}$ with marginals $\mathcal{D}$ and $\mathcal{Q}$. Throughout this paper, we only consider the case of $p = 1$, i.e. the Wasserstein-1 distance and the cost function as Euclidean (ℓ_2) distance.

Labeling Function Assumption Some theoretical works show that AL cannot improve the sample complexity in the worst case, thus identifying properties of the AL paradigm is beneficial (Urner and Ben-David, 2013). For example, Urner et al. (2013) defined a formal *Probabilistic Lipschitz* condition, in which the Lipschitzness condition is relaxed and formalizes the intuition that *under suitable feature representation, the probability of two close points having different labels is small* (Urner and Ben-David, 2013). We adopt the Joint Probabilistic Lipschitz property, which can be viewed as an extension of (Pentina and Ben-David, 2018) and also coherent with (Courty et al., 2017).

Definition 3.1. Let $\phi : \mathbb{R} \to [0,1]$. We say labeling function $h^\star$ is $\phi(\lambda)$-$(\mathcal{D}, \mathcal{Q})$ Joint Probabilistic Lipschitz if $\text{supp}(\mathcal{Q}) \subseteq \text{supp}(\mathcal{D})$ and for all $\lambda > 0$ and all distribution couplings $\gamma \in \Pi(\mathcal{D}, \mathcal{Q})$:

$$\mathbb{P}_{(x_{\mathcal{D}},x_{\mathcal{Q}})\sim\gamma}[|h^\star(x_{\mathcal{D}}) - h^\star(x_{\mathcal{Q}})| > \lambda\|x_{\mathcal{D}} - x_{\mathcal{Q}}\|_2] \leq \phi(\lambda), \tag{3.1}$$

where $\phi(\lambda)$ reflects the decay property. Urner et al. (2013) showed that the faster the decay of $\phi(\lambda)$ with $\lambda \to 0$, the better the labeling function and the easier it is to learn the task.

3.2.2 Theoretical results related to Querying Distribution

In this part, we will derive the relation between the querying and the data generation distribution.

Theorem 3.1. *Supposing $\mathcal{D}$ is the data generation distribution and $\mathcal{Q}$ is the querying distribution. If the loss ℓ is symmetric, L-Lipschitz and bounded; $\forall h \in \mathcal{H}$ is at most H-Lipschitz and the underlying labeling function $h^\star$ is $\phi(\lambda)$-$(\mathcal{D}, \mathcal{Q})$ Joint Probabilistic Lipschitz; then the expected risk w.r.t. $\mathcal{D}$ can be upper bounded by:*

$$R_{\mathcal{D}}(h) \leq R_{\mathcal{Q}}(h) + L(H+\lambda)W_1(\mathcal{D}, \mathcal{Q}) + L\phi(\lambda). \tag{3.2}$$

From Eq. (3.2), the expected risk of $\mathcal{D}$ is upper bounded by the expected risk w.r.t. the query distribution $\mathcal{Q}$, the Wasserstein distance $W_1(\mathcal{D}, \mathcal{Q})$, and the labeling function property $\phi(\lambda)$. That means a desirable query should hold a small expected risk with a better matching to the original distribution $\mathcal{D}$ (diversity).

Proof Sketch The complete proof is delegated in Appendix Sec. B.1, which mainly consists of two steps.

1. Transfer Risk Upper Bound, we upper bound the gap between the true distribution and query distribution:

$$R_{\mathcal{D}}(h) - R_{\mathcal{Q}}(h) \leq L \int_{\Omega\times\Omega} |h^\star(x_{\mathcal{D}}) - h^\star(x_{\mathcal{Q}})| d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}) + LH \int_{\Omega\times\Omega} ||x_{\mathcal{D}} - x_{\mathcal{Q}}||_2 d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}})$$

2. Using Joint Probabilistic Lipschitz to upper bound $\int_{\Omega\times\Omega} |h^\star(x_{\mathcal{D}}) - h^\star(x_{\mathcal{Q}})| d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}})$, then we have:

$$\int_{\Omega\times\Omega} |h^\star(x_{\mathcal{D}}) - h^\star(x_{\mathcal{Q}})| d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}) \leq L\lambda W_1(\mathcal{D}, \mathcal{Q}) + L\phi(\lambda)$$

   Combining these two intermediate conclusion, we have the aforementioned theoretical result.

Non-Asymptotic Analysis Moreover, we can extend the non-asymptotic analysis of Theorem 1 since we generally have finite observations. The proof adopted the standard uniform convergence to bound the empirical risk and empirical Wasserstein distance.

Corollary 3.1. *Supposing we have the finite observations which are i.i.d. generated from $\mathcal{D}$ and $\mathcal{Q}$: $\hat{D} = \frac{1}{N}\sum_{i=1}^{N} \delta\{x_{\mathcal{D}}^i\}$ and $\hat{Q} = \frac{1}{N_q}\sum_{i=1}^{N_q} \delta\{x_{\mathcal{Q}}^i\}$ with $N_q \leq N$. Then with high probability $\geq 1-\delta$, $\forall h \in \mathcal{H}$, the expected risk w.r.t. $\mathcal{D}$ can be further upper bounded by:*

$$R_{\mathcal{D}}(h) \leq \hat{R}_{\mathcal{Q}}(h) + L(H+\lambda)W_1(\hat{\mathcal{D}}, \hat{\mathcal{Q}}) + L\phi(\lambda) + 2L\text{Rad}_{N_q}(\mathcal{H}) + \kappa(\delta, N, N_q),$$

where $\kappa(\delta, N, N_q) = \mathcal{O}(N^{-1/s_d} + N_q^{-1/s_q} + \sqrt{\frac{\log(1/\delta)}{N}} + \sqrt{\frac{\log(1/\delta)}{N_q}})$ is the sample complexity term under big-$\mathcal{O}$ notation. Specifically, $s_d > 1$ and $s_q > 1$ are some positive constants that are related to

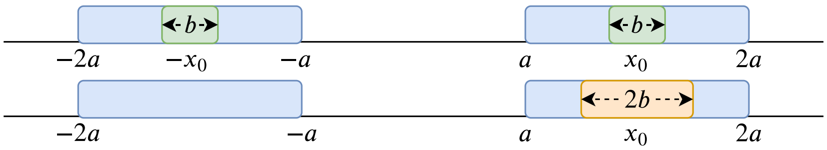


Figure 3.1 – $\mathcal{H}$-divergence vs. Wasserstein distance for $\mathcal{D}$-$\mathcal{Q}$ distribution matching. The desirable query distribution should be more diverse (first row) for avoiding *sampling bias* (second row). The computational result shows that $\mathcal{H}$-divergence is not a proper metric to measure query diversity while Wasserstein is.

the covering number of the hypothesis set $\mathcal{H}$. $\mathrm{Rad}_{N_q}(\mathcal{H}) = \mathbb{E}_{S\sim\mathcal{Q}^{N_q}}\mathbb{E}_{\sigma_1^{N_q}}[\sup_{h\in\mathcal{H}} \frac{1}{N_q}\sum_{i=1}^{N_q}\sigma_i h(x_i)]$ *is the expected Rademacher complexity where* σ_i *is the random variable follows Bernoulli distribution* $\sigma_i \sim$ *Ber*(0.5).

3.2.3 Why Wasserstein Distance

In the context of deep active learning, current work such as (Gissin and Shalev-Shwartz, 2019; Sinha et al., 2019) generally explicitly or implicitly adopted the idea of $\mathcal{H}$-divergence (Ben-David et al., 2010a): $d_{\mathcal{H}}(\mathcal{D}, \mathcal{Q}) = 1 - 2\epsilon$, with ϵ the prediction error when training a binary classifier to *discriminate* the observations sampled from the query and original distribution. Thus a smaller error facilitates the separation of the two distributions with larger $\mathcal{H}$-divergence and vice versa.

However, we notice that in AL, $\mathrm{supp}(\mathcal{Q}) \subseteq \mathrm{supp}(\mathcal{D})$, thus $\mathcal{H}$-divergence may not be a good metric for indicating the diversity property of the querying distribution. On the contrary, Wasserstein distance reflects the optimal transport cost for moving one distribution to another. A smaller transport cost means a better coverage of the distribution $\mathcal{D}$.

For a better understanding of this problem, we give an illustrative example by computing the exact $\mathcal{H}$-divergence and Wasserstein-1 distance in one dimension, shown in Fig. 3.1. Specifically, we have three uniform distributions: $\mathcal{D}_1$ the original data distribution, $\mathcal{D}_2$, $\mathcal{D}_3$ two different query distributions:

$$\begin{aligned}
&\mathcal{D}_1 \sim \mathcal{U}\big([-2a, -a] \cup [a, 2a]\big),\\
&\mathcal{D}_2 \sim \mathcal{U}\big([-x_0 - \frac{b}{2}, -x_0 + \frac{b}{2}] \cup [x_0 - \frac{b}{2}, x_0 + \frac{b}{2}]\big),\\
&\mathcal{D}_3 \sim \mathcal{U}\big([x_0 - b, x_0 + b]\big).
\end{aligned}$$

Where $a > 2b > 0$ and $x_0 \in (a, 2a)$.

In AL, we can further assume $\mathrm{supp}(\mathcal{D}_2) \subseteq \mathrm{supp}(\mathcal{D}_1)$, $\mathrm{supp}(\mathcal{D}_3) \subseteq \mathrm{supp}(\mathcal{D}_1)$ and $a > b > 0$. For $\mathcal{H}$-divergence, we set the classifier as a threshold function $f(x) = \mathbf{1}\{x \geq p\}$. Then we can compute the exact $d_{\mathcal{H}}(\cdot, \cdot)$ and $W_1(\cdot, \cdot)$:

$$\begin{aligned} d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_2) &= d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_3) \\ \min_{x_0} W_1(\mathcal{D}_1, \mathcal{D}_3) &> \max_{x_0} W_1(\mathcal{D}_1, \mathcal{D}_2) \end{aligned} \tag{3.3}$$

From Eq. (3.3), the $\mathcal{H}$-divergence indicates the same divergence result where Wasserstein-1 distance exactly captures the property of diversity: *more diverse query distribution $\mathcal{Q}$ means smaller Wasserstein-1 distance $W_1(\mathcal{D}, \mathcal{Q})$.*

3.3 Practical Deep Batch Active Learning

We have discussed the interactive procedure as the distribution matching and showed that Wasserstein distance is a *proper* metric for measuring the diversity during distribution matching. Based on the aforementioned analysis, in the batch active learning problem, we have labelled data $\hat{L} = \frac{1}{L}\sum_{i=1}^{L} \delta\{x_i^l\}$ and its labels $\{y_i^l\}_{i=1}^{L}$, unlabelled data $\hat{U} = \frac{1}{U}\sum_{i=1}^{U} \delta\{x_i^u\}$ and total distribution $\hat{\mathcal{D}} = \hat{L} \cup \hat{U}$ with partial labels $\{y_i^l\}_{i=1}^{L}$. The goal of AL at each interaction is: 1) find a batch $\hat{B} = \frac{1}{B}\sum_{i=1}^{B} \delta\{x_i^b\}$ with $x_i^b \in \hat{U}$ during the query; 2) find a hypothesis $h \in \mathcal{H}$ such that:

$$\min_{\hat{B}, h} \hat{R}_{\hat{L}\cup\hat{B}}(h) + \mu W_1(\hat{\mathcal{D}}, \hat{L} \cup \hat{B}). \tag{3.4}$$

Where $\mu > 0$ is the hyper-parameter to control the trade-off between uncertainty and diversity.

Eq.(3.4) follows the principles (upper bound) from Theorem 3.1 and Corollary 3.1. Moreover, if we fix the hypothesis h, the sampled batch simultaneously holds two requirements:

1. Minimize the empirical error. We will show later it is related to uncertainty based sampling.
2. Minimize the Wasserstein-1 distance w.r.t. original distribution, which encourages a better distribution matching of $\hat{\mathcal{D}}$.

3.3.1 Min-Max Problem in DNN

Based on Eq.(3.4), we can extend the loss to the deep representation learning scenario, since directly estimating the Wasserstein-1 distance through solving optimal transport for complex and large-scale data is still a challenging and open problem.

Inspired by (Arjovsky et al., 2017), we then adopt the min-max optimizing through training the DNN. Namely, according to Kantorovich-Rubinstein duality, the Wasserstein distance $W_1(P, Q)$ can be expressed as:

$$W_1(P, Q) = \max_{\|g\|_L \leq 1} \mathbb{E}_{x\sim P}[g(x)] - \mathbb{E}_{x\sim Q}[g(x)]$$

Where $\|g\|_L \leq 1$ is the statistic critic function that is restricted within 1-Lipschitz function. The dual term suggests that the estimation of Wasserstein-1 distance can be realized by introducing a statistic critic function g. In the context of deep learning, we denote $\boldsymbol{\theta}^f$, $\boldsymbol{\theta}^h$, $\boldsymbol{\theta}^d$ are parameters corresponding to the *feature extractor*, *task predictor* and *distribution critic*; $\hat{R}$ is the prediction loss (the first term in Eq.(3.4)) and $\hat{E}$ is the adversarial (min-max) loss in estimating Wasserstein-1 distance (the second term in Eq.(3.4)). Then Eq. (3.4) motivates the following loss in deep active learning:

$$\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h,\hat{B}} \max_{\boldsymbol{\theta}^d} \hat{R}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^h) + \mu\hat{E}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^d), \tag{3.5}$$

We further denote the *parametric task prediction* function $h(x,y,(\boldsymbol{\theta}^f,\boldsymbol{\theta}^h)) \equiv h(x,y) : \mathcal{X}\times\mathcal{Y} \to (0,1]$ with $\sum_y h(x,y) = 1$ and the *parametric critic* function $g(x,(\boldsymbol{\theta}^f,\boldsymbol{\theta}^d)) \equiv g(x) : \mathcal{X} \to [0,1]$ with restricting $g(x)$ to the 1-Lipschitz function (Kantorovich-Rubinstein duality). Then each term in Eq. (3.5) can be expressed as:

$$\begin{aligned}\hat{R}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^h) &= \mathbb{E}_{(x,y)\sim\hat{L}\cup\hat{B}}\ell(h(x,y)),\\ \hat{E}(\boldsymbol{\theta}^f,\boldsymbol{\theta}^d) &= \mathbb{E}_{x\sim\hat{\mathcal{D}}}[g(x)] - \mathbb{E}_{x\sim\hat{L}\cup\hat{B}}[g(x)].\end{aligned}$$

3.3.2 Two-stage Optimization

Through some computation, we can decompose Eq. (3.5) into three terms:

$$\begin{aligned}\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h,\hat{B}} \max_{\boldsymbol{\theta}^d} & \underbrace{\frac{1}{L+B}\sum_{(x,y)\in\hat{L}}\ell(h(x,y))}_{\text{Training: Prediction Loss}} + \underbrace{\mu\Big(\frac{1}{L+U}\sum_{x\in\hat{U}}g(x) - \big(\frac{1}{L+B}-\frac{1}{L+U}\big)\sum_{x\in\hat{L}}g(x)\Big)}_{\text{Training: Min-max Loss}}\\ & + \underbrace{\frac{1}{L+B}\sum_{(x,y^?)\in\hat{B}}\ell(h(x,y^?)) - \frac{\mu}{L+B}\sum_{x\in\hat{B}}g(x)}_{\text{Query}},\end{aligned} \tag{3.6}$$

where the critic function $g(x)$ is 1-Lipschitz and L, U, B are the size of labeled, unlabeled, and query data. $y^?$ is called the *agnostic-label*, since it is not available during the query stage. Then from Eq. (3.3.2), each interaction of AL can be naturally decomposed into two stages (optimizing DNN and batch selection), through alternating optimization.

3.3.3 Training DNN

In the training stage, we used all observed data to optimize the neural network parameters:

$$\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h} \max_{\boldsymbol{\theta}^d} \frac{1}{L+B}\sum_{(x,y)\in\hat{L}}\ell(h(x,y)) + \mu\Big(\frac{1}{L+U}\sum_{x\in\hat{U}}g(x) - \big(\frac{1}{L+B}-\frac{1}{L+U}\big)\sum_{x\in\hat{L}}g(x)\Big), \tag{3.7}$$

by restricting $g(x)$ to the 1-Lipschitz function. Instead of only minimizing the prediction error, the proposed approach naturally leverages the unlabeled data information through a min-max training. More intuitively, the critic function g aims to evaluate how probable it is that the sample comes from the labeled or unlabeled parts[1]. According to the loss, given a fixed g, when $g(x) \to 1$ meaning that it is highly probable that the samples come from the unlabeled set $x \in \hat{U}$ and vice versa. [2] Since $B < U$, thus $\frac{1}{L+B} - \frac{1}{L+U} > 0$, indicates that the proposed adversarial loss is always valid.

Based on Eq. (3.7), we call this framework the Wasserstein Adversarial Active Learning (WAAL) in our deep batch AL. The labeled $\hat{L}$ and unlabeled data $\hat{U}$ pass a common feature extractor, then $\hat{L}$ will be used in the prediction and $\hat{L}$, $\hat{U}$ together will be used in the min-max (adversarial) training. In the practical deep learning, we apply the cross entropy loss: $\ell(x, y) = -\log(h(x, y))$[3].

Redundancy Trick One can directly apply gradient descent to optimize Eq. (3.7) on the whole dataset. Actually, we generally apply the mini-batch-based SGD approach in training the DNN[4]. While a practical concern during the adversarial training procedure is the unbalanced label and unlabeled data during the training procedure. Thus we propose the *redundancy trick* to solve this concern. For abuse of notation, we denote the unbalanced ratio $\gamma = \frac{U}{L}$ and the query ratio $\alpha = \frac{B}{L}$, with the adversarial loss simplified as:

$$\mu'\big(\frac{1}{U}\sum_{x\in\hat{U}} g(x) - \frac{1}{\gamma}\frac{\gamma-\alpha}{1+\alpha}\frac{1}{L}\sum_{x\in\hat{L}} g(x)\big),$$

with $\mu' = \frac{\gamma}{1+\gamma}\mu$.

Then following the *redundancy trick* for optimizing the adversarial loss, we keep the same mini-batch size S for labelled and unlabeled observations. Due to the existence of the unbalanced data, we simply conduct a replacement sampling to construct the training batch for the labeled data, then divided by the unbalanced ratio γ. For each training batch, the adversarial loss can be rewritten as:

$$\min_{\boldsymbol{\theta}^f}\max_{\boldsymbol{\theta}^d} \mu'\big(\frac{1}{S}\sum_{x\in\hat{U}_S} g(x) - C_0\frac{1}{S}\sum_{x\in\hat{L}_S} g(x)\big), \tag{3.8}$$

where $\hat{U}_s$ $\hat{L}_s$ are unlabeled and labeled training batch and $C_0 = \frac{1}{\gamma^2}\frac{\gamma-\alpha}{1+\alpha}$ is the "bias coefficient" in deep active adversarial training. For example, if there exist 1K labeled samples, 9K unlabeled samples and a current query batch budget of 1K, then we can compute $C_0 \approx 0.05$ so as to control excessive reusing of the labelled dataset.

[1] We should point out that this is the high-level intuition. More specifically, the critic parameter $\boldsymbol{\theta}^d$ of g tries to maximize and the feature parameter $\boldsymbol{\theta}^f$ of g tries to minimize the adversarial loss according to the Wasserstein metric. Moreover the proposed min-max loss differs from the standard Wasserstein min-max loss since they hold different weights ("bias coefficient")

[2] This can be alternatively explained by assuming g is the discriminator to differentiate labeled and unlabeled dataset. Thus $g(x) = 1$ means the x comes from unlabeled data.

[3] Although the cross entropy loss does not satisfy the exact assumptions in the theoretical analysis, our later experiments suggest that the proposed algorithm is very effective for cross-entropy loss.

[4] We have referred to this as the *training/mini batch* to avoid any confusion with the querying batch mentioned before.

3.3.4 Query Strategy

The second stage over the unlabeled data aims to find a queried batch such that:

$$\text{argmin}_{\hat{B}\subset\hat{U}} \frac{1}{L+B} \sum_{(x,y^?)\in\hat{B}} \ell(h(x,y^?)) - \frac{\mu}{L+B}\sum_{x\in\hat{B}} g(x), \tag{3.9}$$

where $y^?$ is the agnostic label.

Agnostic-label upper bound loss indicates uncertainty Since we do not know $y^?$ during the query, we can instead optimize an *upper bound* of Eq. (3.9). In the classification problem with cross entropy loss, suppose that we have $\{1,\dots,K\}$ possible outputs with $\sum_{y\in\{1,\dots,K\}} h(x,y) = 1$, then we have upper bounds Eq. (3.10) and (3.11), which both reflect the uncertainty measures with different interpretations.

1. Minimizing over the single worst case upper bound indicates the sample with the *highest least prediction confidence score*:

$$\min_x \ell(h(x,y^?)) \leq \min_x \max_{y\in\{1,\dots,K\}} -\log(h(x,y)). \tag{3.10}$$

   For example, we have two samples with a binary decision score $h(x_1,\cdot) = [0.4,\, 0.6]$ and $h(x_2,\cdot) = [0.3,\, 0.7]$. Since $\max_y -\log(h(x_1,y)) < \max_y -\log(h(x_2,y))$, we will choose x_1 as the query since the least prediction label confidence 0.4 is higher. Intuitively such a sample seems uncertain since the least label prediction confidence is high.
   It is worth mentioning that in the binary classification setting, it recovers the least prediction confidence score approach (Baseline 2), which is a common strategy in AL. Notably, it can be problematic in the multi-classification setting. E.g, consider $h(x_1,\cdot) = [0.3,\, 0.3,\, 0.4]$ and $h(x_2,\cdot) = [0.29,\, 0.36,\, 0.35]$, then according to the rule, x_1 will be selected. However, x_2 is more uncertain since the prediction is more uniform, which can lead to potential issues. To this end, we further consider the L_1 norm as the upper bound without this issue as the uncertainly criteria.
2. Minimizing over ℓ_1 norm upper bound indicates the sample with a *uniformly of prediction confidence score*:

$$\min_x \ell(h(x,y^?)) \leq \min_x \sum_{y\in\{1,\dots,K\}} -\log(h(x,y)). \tag{3.11}$$

   Intuitively if the sample's prediction confidence trend is more uniform, the more uncertain the sample will be. We can also show the $\min$ arrives when the output score is uniform, as shown in the supplementary material.

We would like to point out that the upper bounds proposed in Eq. (3.10,3.11) are additive, i.e. we can apply any convex combination of these two losses as the hybrid uncertain query strategy.

Critic output indicates diversity As for the critic function $g(x) : \mathcal{X} \to [0,\, 1]$ from the adversarial loss, if the critic function output trends to $g(x) \to 1$, it means $x \in \hat{U}$ and vice versa. Then according

Method	LeastCon	Margin	Entropy	K-Median	DBAL	Core-set	DeepFool	WAAL
Time	0.94	0.95	0.95	33.98	9.25	45.88	124.46	1

Table 3.1 – Relative Average querying time, assuming the query time of WAAL as the unit.

to the query loss, we want to select the batch with higher critic values $g(x)$, meaning they look more different than the labelled samples under the Wasserstein metric.

If the unlabeled samples look like the labeled ones (small $g(x)$ with $x \in \hat{U}$), then under some proper conditions (such as *Probabilistic Lipschitz Condition* in def. 3.1), such examples can be more easily predicted because we can infer them from their very near neighbours' information.

On the contrary, the unlabeled samples with high $g(x)$ under the current assumption cannot be effectively predicted by the current labeled data (far away data). Moreover, the $g(x)$ is trained through Wasserstein distance-based loss, shown as a proper metric for measuring the diversity. Therefore, the query batch with higher critic value ($g(x)$) means a larger transport cost from the labeled samples, indicating that it is more informative and represents diversity.

Remark The aforementioned two terms in the query strategy indicate an explicit *uncertainty* and *diversity* trade-off. Uncertainty criteria can reduce the future empirical risk while inducing a potential sampling bias. While the diversity criteria can improve the exploration of the distribution, it might be inefficient for a small query batch. Our query approach naturally combines these two for choosing the samples with prediction uncertainty and diversity. Moreover, since Eq. (3.9) is additive, we can easily estimate the query batch through the greedy algorithm.

3.3.5 Proposed Algorithm

Based on the previous analysis, our proposed algorithm includes a training stage [Eq. (3.7,3.8)] and a query stage [Eq. (3.9), (3.10) and (3.11)] for solving Eq. (3.5) or (3.3.2). We only show the learning algorithm for one interaction in Algorithm 2, then the remaining interactions will be repeated accordingly.

Since the discriminator function g should be restricted in 1-Lipschitz, we add the gradient penalty term such as (Gulrajani et al., 2017) to g to restrict the Lipschitz property.

3.4 Experiments

We start our experiments with a small initial labeled pool of the training set. The initial observation size and the budget size range from $1\% - 5\%$ of the training dataset, depending on the task. Following Alg. 2, the selected batch will be annotated and added into the training set. Then the training process for the next iteration will be repeated on the new formed labeled and unlabeled set *from scratch*.

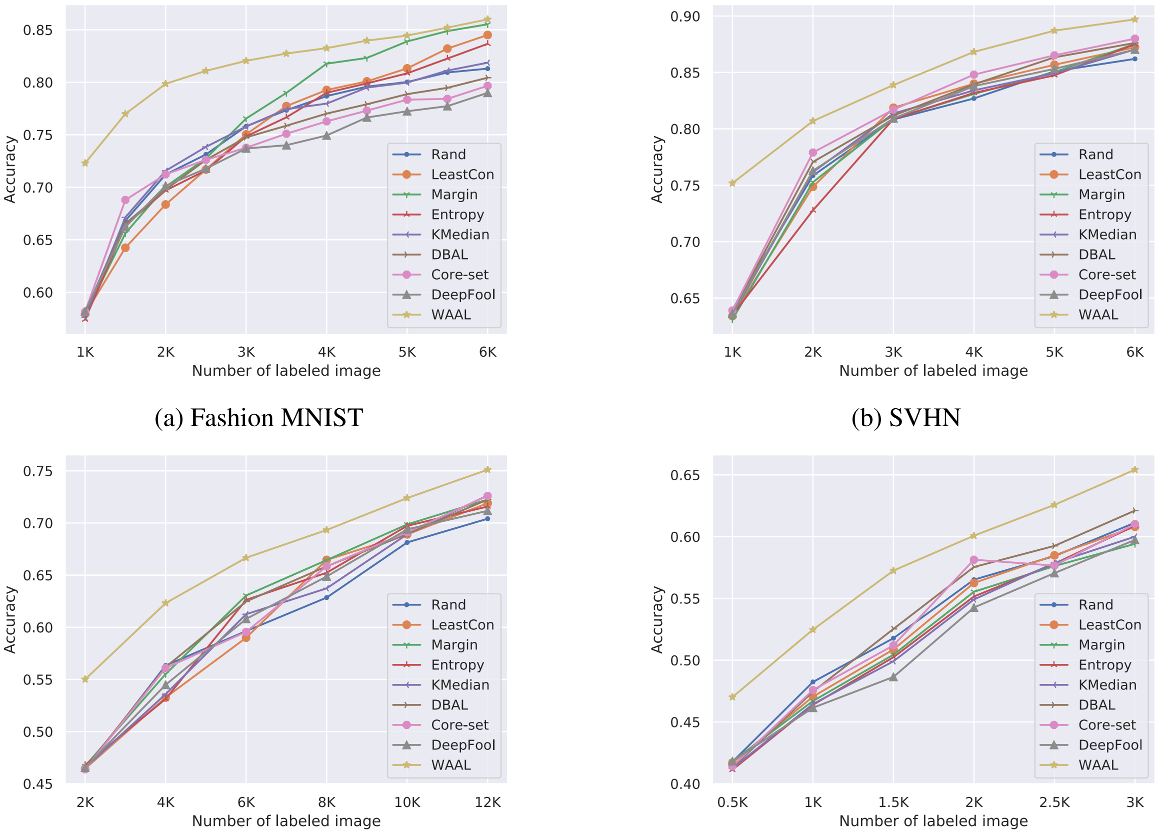
Algorithm 2 WAAL: one interaction

```
Require: Labeled samples $\hat{L}$, unlabeled samples $\hat{U}$, query budget $B$ and hyper-parameters (learning
    rate $\eta$, trade-off rate $\mu$, $\mu'$)
Ensure: Neural network parameters $\boldsymbol{\theta}^f$, $\boldsymbol{\theta}^h$, $\boldsymbol{\theta}^d$
$\triangleright\triangleright\triangleright$ DNN Parameter Training Stage $\triangleleft\triangleleft\triangleleft$
for mini-batch of samples $\{(x_i^u)\}_{i=1}^S$ from $\hat{U}$ do
    Constructing minibatch $\{(x_i^l, y_i^l)\}_{i=1}^S$ from $\hat{L}$ through sampling with replacement (redundancy
    trick).
    Updating $\boldsymbol{\theta}^h$: $\boldsymbol{\theta}^h = \boldsymbol{\theta}^h - \frac{\eta}{S}\sum_{(x^l,y^l)}\frac{\partial\ell(h((x^l,y^l))}{\partial\boldsymbol{\theta}^h}$
    Updating $\boldsymbol{\theta}^f$: $\boldsymbol{\theta}^f = \boldsymbol{\theta}^f - \frac{\eta}{S}\big(\sum_{(x^l,y^l)}\frac{\partial\ell(h((x^l,y^l))}{\partial\boldsymbol{\theta}^f} + \mu'\{\sum_{x^u}\frac{\partial g(x)}{\partial\boldsymbol{\theta}^f} - C_0\sum_{x^l}\frac{\partial g(x)}{\partial\boldsymbol{\theta}^f}\}\big)$
    Updating $\boldsymbol{\theta}^d$: $\boldsymbol{\theta}^d = \boldsymbol{\theta}^d + \frac{\eta\mu'}{S}\{\sum_{x^u}\frac{\partial g(x)}{\partial\boldsymbol{\theta}^d} - C_0\sum_{x^l}\frac{\partial g(x)}{\partial\boldsymbol{\theta}^d}\}$
end for
$\triangleright\triangleright\triangleright$ Querying Stage $\triangleleft\triangleleft\triangleleft$
Applying the convex combination of Eq. (3.10) and (3.11) to compute uncertainly score $\mathcal{U}(x^u)$;
Computing diversity score $g(x^u)$;
Ranking the score $\mathcal{U}(x^u) - \mu g(x^u)$ with $x^u \in \hat{U}$, choosing the smallest $B$ samples, forming
querying batch $\hat{B}$
$\triangleright\triangleright\triangleright$ Updating $\triangleleft\triangleleft\triangleleft$
$\hat{L} = \hat{L} \cup \hat{B}$, $\hat{U} = \hat{U} \setminus \hat{B}$
```

We evaluate our proposed approach on three object recognition tasks, namely Fashion-MNIST (image size: 28 × 28) (Xiao et al., 2017), SVHN (32 × 32) (Netzer et al., 2011), CIFAR-10 (32 × 32) (Krizhevsky et al., 2009) and STL10. For each task, we split the whole data into training, validation, and testing parts. We evaluate the performance of the proposed algorithm for image classification task by computing the prediction accuracy. We repeat all experiments 5 times and report the average value. The details of the experimental settings (dataset description, train/validation/test splitting, detailed implementations, hyper-parameter settings and choices) and additional experimental results are provided in the supplementary material.

Baselines We compare the proposed approach with the following baselines: (1) Random sampling; (2) Least confidence (Culotta and McCallum, 2005); (3) Smallest Margin (Scheffer and Wrobel, 2001); (4) Maximum-Entropy sampling (Settles, 2012); (5) K-Median approach (Sener and Savarese, 2018): choosing the points to be labelled as the cluster centers of K-Median algorithm ; (6) Core-set approach (Sener and Savarese, 2018); (7) Deep Bayesian AL (DBAL) (Gal et al., 2017); and (8) DeepFoolAL (Mayer and Timofte, 2018).

Implementations For the proposed approach, differing from baselines, we train the DNN from labeled and unlabeled data without data-augmentation. For the tasks on SVHN, CIFAR-10 we implement VGG16 (Simonyan and Zisserman, 2014) and for task on Fashion MNIST we implement LeNet5 (LeCun et al., 1998) as the feature extractor. On top of the feature extractor, we implement a two-layer multi-layer perceptron (MLP) as the classifier and critic function. For all tasks, at each interaction we set the maximum training epoch as 80. For each epoch, we feed the network with mini-batch of 64



(a) Fashion MNIST

(b) SVHN

(c) CIFAR-10

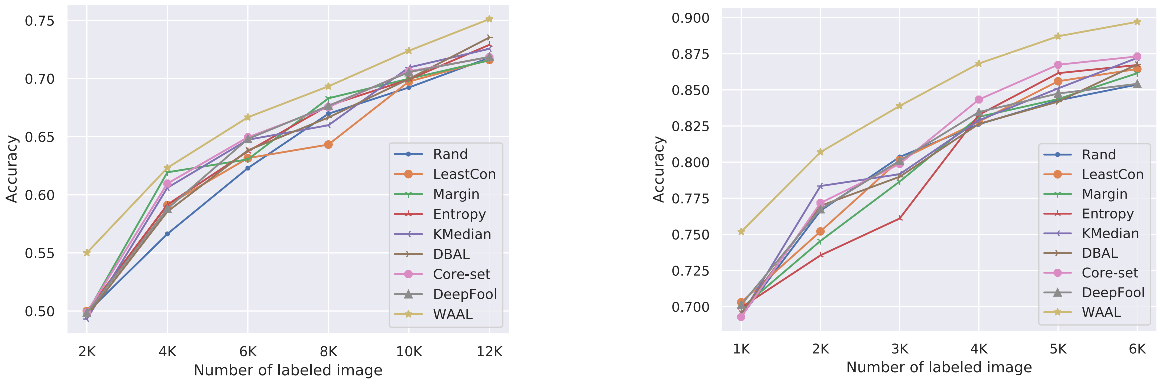
(d) STL-10

Figure 3.2 – Empirical performance on Fashion MNIST, SVHN, CIFAR-10 and STL-10 over five repetitions. All the approaches are implemented without data-augmentation.

samples and adopt SGD with momentum (Sutskever et al., 2013) to optimize the network. We tune the hyper-parameter through grid search. In addition, in order to avoid over-training, we also adopt early stopping (Caruana et al., 2001) techniques during training.

3.4.1 Results

We demonstrate the empirical results in Fig. 3.2. Exact numerical values and standard deviations are reported in the supplementary material. The proposed approach (WAAL) consistently outperforms all of the baselines during the interactions. We noticed that WAAL shows a large improvement ($> 5\%$) in the initial training procedure since it efficiently constructs a good representation through leveraging the unlabeled data information. For the relatively simple input task Fashion MNIST, the simplest uncertainty query (Smallest Margin/Least Confidence) finally achieved almost the same level performance with WAAL under 6K labeled samples. Moreover, we observed that for the small or middle sized queried batch (0.5K-2K) in the relatively complex dataset (SVHN, CIFAR-10), the baselines show similar results in deep AL, which is coherent with previous observations (Gissin and Shalev-Shwartz, 2019; Ash et al., 2019). On the contrary, our proposed approach still shows a good



(a)

(b)

Figure 3.3 – Ablation study (a) CIFAR-10, (b) SVHN, the baselines are all trained by leveraging the unlabeled information through $\mathcal{H}$-divergence. All the approaches are implemented without data-augmentation.

improved empirical result, emphasizing the benefits of properly designing loss by considering the unlabeled data in the context of deep AL.

It is worth mentioning that the reported performances are actually lower than the reported results in (Sener and Savarese, 2018). Because we did not adopt any data augmentation in training, whereas this trick can significantly improve the performance in the limited labeled data such as active learning.

We also report the average query time for the baselines and the proposed approach on SVHN dataset in Tab. 3.1. The result indicates that WAAL holds the same querying time level with the standard uncertainty based strategies since they are all *end-to-end* strategies without knowing the internal information of the DNN. However some diversity-based approaches such as Core-set and K-Median require the computation of the distance in the feature space and finally induce a much longer query time.

3.4.2 Ablation Study: Advantage of Wasserstein Metric

In this part, we empirically show the advantage of considering the Wasserstein distance in the ablation study. Specifically, for the whole baselines we adopt $\mathcal{H}$-divergence based adversarial loss for training DNN. That is, we set a discriminator and we used the binary cross entropy (BCE) adversarial loss to discriminate the labeled and unlabeled data (Gissin and Shalev-Shwartz, 2019). Then in the query we still apply the different baselines strategies to obtain the labels. We tested in the CIFAR-10 dataset and report the performances in Fig. 3.3. We present a brief introduction, exact numerical values, and more results in the appendix.

From the results, we observed that the gap between the initial training procedure has been reduced from about 8% to 5% because of introducing the adversarial based training. However, our proposed approach (WAAL) still consistently outperforms the baseline. The reason might be that the $\mathcal{H}$-divergence based

	1K	2K	3K	4K	5K	6K
Accuracy	72.9%	85.4%	86.7%	91.2%	91.8%	92.8%
Macro-F1	63.2%	77.6%	78.7%	85.3%	86.3%	87.5%

Table 3.2 – Results of WAAL on SVHN (by accuracy and Macro-F1)

adversarial loss is not a good metric for the Deep AL as we formally analyzed before. The results indicate the practical potential of adopting the Wasserstein distance for the Deep Active Learning problem.

3.4.3 Performance under other performance metrics

Apart from the prediction accuracy, we also evaluate WAAL under other metrics, such as Macro-F1 score (Fawcett, 2006), shown in Tab. 3.2. The results indicate similar trends in other performance metrics.

3.5 Conclusion

In this chapter, we proposed a unified and principled method for both querying and training in deep Active Learning. We analyzed the theoretical insights from the intuition of modeling the interactive procedure in AL as distribution matching. Then we derived a new training loss for jointly learning hypothesis and query batch searching. We formulated the loss for DNN as a min-max optimization problem by leveraging the unlabeled data. As for the query for batch selection, it explicitly indicates the uncertainty-diversity trade-off. The results on different benchmarks showed a consistent better accuracy and faster efficient query strategy. The analytical and empirical results reaffirmed the benefits and potentials for reflecting on the unified principles for deep active learning.

Chapter 4

Domain Adaptation Theory With Jensen-Shannon Divergence

Original title of the article: **A New Domain Adaptation Theory With Jensen-Shannon Divergence**

Résumé

Dans ce chapitre, nous révélons l'incohérence entre l'entraînement adversarial de domaine et son équivalent théorique généralement supposé, basé sur la divergence $\mathcal{H}$. Concrètement, nous découvrons que la divergence $\mathcal{H}$ n'est pas équivalente à la divergence de Jensen-Shannon, l'objectif d'optimisation de *domain adversarial training*. Pour cela, nous élaborons un nouveau cadre théorique en prouvant directement les bornes supérieure et inférieure du risque cible basées sur la divergence de Jensen-Shannon distributionnelle conjointe.

Nous dérivons également des bornes supérieures bidirectionnelles pour les transferts marginaux et conditionnels. Notre modèle présente une flexibilité inhérente pour différents problèmes d'apprentissage par transfert, ce qui le rend utilisable dans divers scénarios. D'un point de vue algorithmique, notre théorie permet un guide générique des principes unifiés de l'appariement conditionnel sémantique, de *distribution matching* marginal et de la correction de la distribution marginale des étiquettes. Nous appliquons des algorithmes pour chaque principe et validons empiriquement les avantages de notre cadre sur des jeux de données réels.

Abstract

In this chapter, we reveal the incoherence between the widely adopted empirical domain adversarial training and its generally assumed theoretical counterpart based on $\mathcal{H}$-divergence. Concretely, we find that $\mathcal{H}$-divergence is not equivalent to Jensen-Shannon divergence, the optimization objective in

domain adversarial training. To this end, we establish a new theoretical framework by directly proving the upper and lower target risk bounds based on the joint distributional Jensen-Shannon divergence.

We further derive bidirectional upper bounds for marginal and conditional shifts. Our framework exhibits inherent flexibility for different transfer learning problems, which is usable for various scenarios. From an algorithmic perspective, our theory enables a generic guideline of the unified principles of semantic conditional matching, feature marginal matching, and label marginal shift correction. We employ algorithms for each principle and empirically validate the benefits of our framework on real datasets.

4.1 Introduction

Domain Adaptation (DA) theory is crucial to the fundamental understanding and practical development of relevant algorithms. Conventionally, such theoretical guarantees were typically established based on the notion of $\mathcal{H}$-divergence (Ben-David et al., 2007, 2010a) and its subsequent variants (Redko et al., 2020), where it requires a small $\mathcal{H}$-divergence between source-target and small joint risk. In the context of representation learning, this quantity ($\mathcal{H}$-divergence) is minimized via the well-known *domain adversarial training* (Ganin et al., 2016; Long et al., 2015; Tzeng et al., 2017), which is a stimulating topic in current research.

Domain adversarial training is widely successful in various DA problems such as open set DA (Panareda Busto and Gall, 2017; Cao et al., 2018; You et al., 2019) or conditional shift (Li et al., 2019b), however, the generally assumed theoretical counterpart $\mathcal{H}$-divergence itself is rather limited to explain these working principles, which hampers the further practical advancement. It has been noted that the inherent principle of domain adversarial training is analogous to GANs (Goodfellow et al., 2014), which is equivalent to minimize Jensen-Shannon divergence (Nowozin et al., 2016) between two distributions. Therefore, a DA theory established directly on the Jensen-Shannon divergence would provide a thorough understanding of adversarial training and help overcome the limitations imposed by the use of $\mathcal{H}$-divergence.

In this work, we reveal that $\mathcal{H}$-divergence is **not** consistent with the Jensen-Shannon divergence, indicating the improper adoption of $\mathcal{H}$-divergence theory to explain domain adversarial training practice. Then we build a DA theoretical framework *directly* based on Jensen-Shannon divergence. We establish that the upper bound of the target risk is determined by the source error and the Jensen-Shannon divergence of the two joint distribution (Sec. 4.3.1). Moreover, we derive the upper bounds of bidirectional shifts (Sec. 4.3.2), including (a) Feature Marginal Shift ($\mathcal{T}(x) \neq \mathcal{S}(x)$) and Label Conditional Shift ($\mathcal{T}(y|x) \neq \mathcal{S}(y|x)$); (b) Label Marginal Shift ($\mathcal{T}(y) \neq \mathcal{S}(y)$) and Semantic (Feature) Conditional Shift ($\mathcal{T}(x|y) \neq \mathcal{S}(x|y)$). The theory provides a unified understanding of domain shifts, with covariate shift and label shift being its special cases, which can provide intriguing theoretic insights and effective practice guidelines:

Theoretical Insights Jensen-Shannon divergence enables us to analyze the factors of label space that influence the transfer procedure, which remains elusive in the $\mathcal{H}$-divergence. Specifically, (I) we reveal that the intrinsic error of learning in the target-domain is controlled by the label-space size, the source domain intrinsic error and the similarity of the two domains (Sec. 4.3.3). (II) we also reveal why transfer learning is challenging if the label space of source and target are not identical (a.k.a. open set DA). We formally show that a smaller overlap over the label space leads to a more difficult transfer (Sec. 4.3.3).

Practical Implications Our theory motivates new DA practice for representation learning, which is missing in $\mathcal{H}$-divergence. More concretely, we propose unified principles to control the target risk (Sec. 4.4.2): (I) re-weighted semantic conditional matching, to control the feature conditional shift $D_{\text{JS}}(\mathcal{T}(x|y)\|\mathcal{S}(x|y))$; (II) label marginal shift correction, as the way to eliminate the label marginal shift $D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))$; (III) constraining the feature marginal shift, an approach to prevent poor target pseudo label predictions (i.e. predicted labels), a common phenomena that can lead to negative transfer in semantic conditional matching. The proposed guideline enables us to select existing algorithms for each principle. The empirical results on real datasets verify the benefits of unified principles (Sec. 4.6).

4.2 $\mathcal{H}$-Divergence based DA Theory

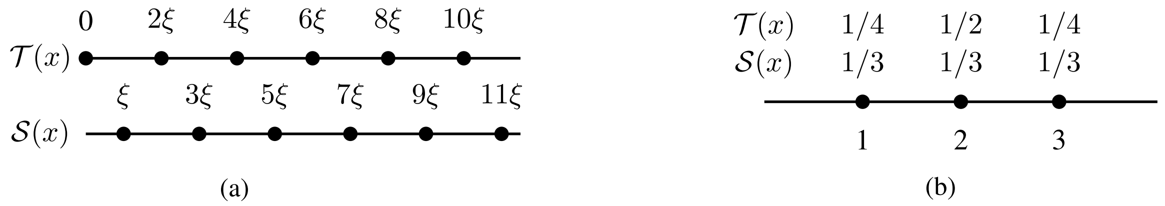
In this chapter, we suppose to have the source distribution $\mathcal{S}$ and target distribution $\mathcal{T}$ over the *joint* input and output space $\mathcal{X} \times \mathcal{Y}$. According to (Ben-David et al., 2007, 2010a), if the data is generated by a marginal distribution and underlying labeling function pair $(\mathcal{D}, h^\star)$, then the upper bound of the target risk error w.r.t. $\forall h \in \mathcal{H}$ is:

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x)) + \beta, \tag{4.1}$$

where $R_{\mathcal{D}}(h) = \mathbb{E}_{x\sim\mathcal{D}}|h(x) - h^\star(x)|$, $d_{\mathcal{H}}$ denotes the $\mathcal{H}$-divergence for measuring the marginal distribution similarities between $\mathcal{S}(x)$ and $\mathcal{T}(x)$ w.r.t. x, β is the optimal joint risk over the two domains. [1]

As pointed out by Ben-David et al. (2007), it is generally impossible to exactly estimate the $\mathcal{H}$-divergence. Hence, this measure is approximated as a binary classification task where we are discriminating the source and the target samples. More specifically, the $\mathcal{H}$-divergence is approximated by distance $d_{\mathcal{A}} = 2(1 - 2\epsilon)$, with ϵ corresponding to the discrimination generalization error. Inspired by this intuition, Ganin et al. (2016) and subsequent approaches empirically adopted adversarial loss

[1] For the sake of convenience, we denote $\mathcal{S}(x)$ as the marginal distribution w.r.t. x. And $\mathcal{S}(y|X = x)$ as the conditional distribution w.r.t. y, given a specified $X = x$. Then dist$(\mathcal{S}(x), \mathcal{T}(x))$ simply denotes the statistical distance w.r.t. variable x. dist$(\mathcal{S}(y|x), \mathcal{T}(y|x))$ is denoted as the statistical distance w.r.t. variable y, which depends on $X = x$ (essentially a function of x).



(a) (b)

Figure 4.1 – $D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))$ cannot be viewed as the approximation of $d_{\mathcal{H}}(\mathcal{S}(x), \mathcal{T}(x))$: (a) for two uniform distributions with different supports, there exists $d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x)) \ll D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))$ if $0 < \xi \ll 1$; (b) while for two distributions with different probability mass, there exists $D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x)) < d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x))$

(Goodfellow et al., 2014) between the domain classifier d and feature extractor function g in the context of representation learning:

$$\begin{aligned} &\min_g \max_d \mathbb{E}_{x\sim\mathcal{S}(x)} \log(d \circ g(x)) + \mathbb{E}_{x\sim\mathcal{T}(x)} \log(1 - d \circ g(x)), \\ &= \min_g D_{\text{JS}}(\mathcal{S}(g(x))\|\mathcal{T}(g(x))), \end{aligned} \tag{4.2}$$

where Eq. (4.2) is the dual term of Jensen-Shannon divergence (Nowozin et al., 2016)

It is worth mentioning that the sup on Jensen-Shannon divergence is defined on the all measurable functions d. In practice, we adopt a predefined family of function $d \in \mathcal{D}$ during the optimization, which is actually lower than the ground truth Jensen-Shannon divergence. i.e,

$$D_{\text{JS}}(\mathcal{S}(g(x))\|\mathcal{T}(g(x)) \geq \max_{d\in\mathcal{D}} \{\mathbb{E}_{x\sim\mathcal{S}(x)} \log(d \circ g(x)) + \mathbb{E}_{x\sim\mathcal{T}(x)} \log(1 - d \circ g(x))\}$$

However, if we further assume the function family has the rich expressive power (e.g, $\mathcal{D}$ is the deep neural network), the maximization will approach the true Jensen-Shannon divergence with a small approximation error.

4.2.1 Jensen-Shannon Divergence is not consistent with $\mathcal{H}$-Divergence

From Eq. (4.2), domain adversarial training is essentially in learning representation to minimize the Jensen-Shannon divergence. However a D_{JS} is not equivalent to $d_{\mathcal{H}}$ in Eq. (4.1). We find these two metrics can be very different and present two counterexamples to illustrate it, shown in Fig. 4.1.

For the sake of simplicity, we design all examples over one dimensional space and use the threshold functions $\mathcal{H} = \{h_t : t \in \mathbb{R}\}$ as the hypothesis class. That is, for any $t \in \mathbb{R}$, the threshold function is defined by $h_t(x) = 1$ for $x < t$ and $h_t(x) = 0$ otherwise.

Counterexample 1 We adopt the example of (Ben-David et al., 2010b), showed in Fig.4.1(a), with a small fixed $\xi \in (0, 1)$. Let the target $\mathcal{T}(x)$ be the uniform distribution over $\{2k\xi : k \in \mathbb{N}, 2k\xi \leq 1\}$ and the source $\mathcal{S}(x)$ be the uniform distribution over $\{(2k+1)\xi : k \in \mathbb{N}, (2k+1)\xi \leq 1\}$. We can compute $d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x)) \approx d_{\mathcal{A}}(\mathcal{T}(x), \mathcal{S}(x)) = \xi$ while $D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x)) = 1$ since the two distributions have *disjoint* supports. Then $d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x)) \ll D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))$ when $\xi \ll 1$, indicating a *small $\mathcal{H}$-divergence can correspond to a very large Jensen-Shannon divergence*.

Counterexample 2 Fig. 4.1 (b) further illustrates that Jensen-Shannon divergence is *not* the upper bound of $\mathcal{H}$-divergence. We assume the source $\mathcal{S}(x)$ be the uniform distribution over $\{1, 2, 3\}$ and let the target $\mathcal{T}(x)$ be the distribution on the same support with different probability mass $\{\mathcal{T}(x=1) = 1/4, \mathcal{T}(x=2) = 1/2, \mathcal{T}(x=3) = 1/4\}$. Then Jensen-Shannon divergence can be even smaller than $\mathcal{H}$-divergence: $D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x)) < d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x))$.

Due to these differences, $\mathcal{H}$-divergence is *not a proper* theoretical tool for analyzing the practice that minimizes the Jensen-Shannon divergence (e.g. domain adversarial training and its variants such as Ganin et al. (2016)).

4.3 DA Theory with Jensen-Shannon Divergence and Theoretical Insights

4.3.1 Upper and Lower Risk Bound

Slightly different from the settings in (Ben-David et al., 2010a), we assume the data $(x, y) \in \mathcal{X} \times \mathcal{Y}$ is generated from a *joint* distribution $\mathcal{D}$ and denote the hypothesis and loss function as $h: \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and $L: \mathbb{R} \to \mathbb{R}$, where the hypothesis $h \in \mathcal{H}$ actually outputs a confidence score of an observation (x, y). We also denote $R_{\mathcal{D}}(h)$ the expected risk w.r.t. distribution $\mathcal{D}$: $R_{\mathcal{D}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}} L(h(x, y))$.

Theorem 4.1 (Upper Bound)**.** *Supposing the prediction loss L is bounded within an interval G: $G = \max(L) - \min(L)$, then for all the hypothesis h the expected risk w.r.t. the target domain can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}}\sqrt{D_{JS}(\mathcal{T}\|\mathcal{S})},$$

where $D_{JS}(\mathcal{T}\|\mathcal{S}) = \frac{1}{2}[D_{KL}(\mathcal{S}\|\mathcal{M}) + D_{KL}(\mathcal{T}\|\mathcal{M})]$ with $\mathcal{M} = \frac{1}{2}(\mathcal{T} + \mathcal{S})$ is the Jensen-Shannon divergence between the joint distribution $\mathcal{S}(x, y)$ and $\mathcal{T}(x, y)$.

The complete proof is demonstrated in Appendix Sec. C.2.

Discussions We notice Ben-David et al. (2010a) analogously proposed the $d_{\mathcal{H}\Delta\mathcal{H}}$ divergence to measure domain discrepancy. However, as it pointed out (in Sec. 7.2 of (Ben-David et al., 2010a)), it is also impossible to exactly estimate this discrepancy. As a consequence, (Ben-David et al., 2010a) still adopted $d_{\mathcal{A}}$ distance to approximate, recovering the same empirical strategy as $\mathcal{H}$ divergence. In addition, it seamlessly connects the well-known assumptions in DA. When the *covariate shift* assumption holds ($\mathcal{T}(y|X = x) = \mathcal{S}(y|X = x)$), the upper bound can be expressed as $R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}}\sqrt{D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))}$. Besides, when the *label shift* assumption holds ($\mathcal{T}(x|Y = y) = \mathcal{S}(x|Y = y)$), the upper bound can be alternatively expressed as $R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}}\sqrt{D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))}$.

Tab 4.1 illustrates the comparisons between the proposed Jensen-Shannon divergence and the $\mathcal{H}$-divergence.

Extension to Unbounded loss The proposed upper bound can be further extended to the *unbounded loss* with sub-Gaussian or sub-Gamma property (Boucheron et al., 2013).

Corollary 4.1 (Sub-Gaussian Upper Bound)**.** *If the loss function satisfies σ-Sub Gaussian w.r.t. distribution P: $\forall\lambda > 0$: $\log \mathbb{E}_P\, e^{[\lambda(\ell(z)-\mathbb{E}_P\ell(z))]} \leq \frac{\lambda^2\sigma^2}{2}$, then the expected risk in the target domain can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \sigma\sqrt{2D_{\text{JS}}(\mathcal{T}\|\mathcal{S})}$$

Corollary 4.2 (Sub-Gamma Upper Bound)**.** *If the loss function satisfies (σ, a)-Sub Gamma property: $\log \mathbb{E}_P\, e^{[\lambda(\ell(z)-\mathbb{E}_P\ell(z))]} \leq \frac{\lambda^2\sigma}{2(1-a|\lambda|)}$, for $0 < |\lambda| < \frac{1}{a}$. Then the expected risk in the target domain can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + (\sigma+1)\sqrt{2D_{JS}(\mathcal{T}\|\mathcal{S})} + 2aD_{JS}(\mathcal{T}\|\mathcal{S})$$

The extended upper bounds can be tighter than the conclusion in Theorem 4.1, particularly when the loss is in a large range with a small variance.

Theorem 4.2 (Lower Bound)**.** *If we assume the loss L as zero-one binary loss, then for any h, we can prove the target risk is lower bounded by:*

$$R_{\mathcal{T}}(h) \geq R_{\mathcal{S}}(h) - \sqrt{D_{JS}(\mathcal{T}\|\mathcal{S})}.$$

The proof is shown in Appendix Sec.C.3. The lower bound provides the insights of the *easy transfer* (Hanneke and Kpotufe, 2019) scenario: learning the target domain can be easier than the source domain, and the gap is controlled (smaller than) by their distribution distance. For example, if we assume $R_{\mathcal{S}}(h) = 0.2$, $D_{\text{JS}}(\mathcal{T}\|\mathcal{S}) = 2 \times 10^{-4}$, then the target risk is also bounded: $R_{\mathcal{T}}(h) \in [0.186, 0.21]$. This indicates $R_{\mathcal{T}}(h)$ can be smaller than $R_{\mathcal{S}}(h)$ but not an arbitrary large gap.

4.3.2 Bi-Directional Marginal/Conditional Shifts

We can decompose the joint Jensen-Shannon divergence into bi-directional marginal and conditional shift upper bounds, according to the information theoretical chain rule (Polyanskiy and Wu, 2019).

Table 4.1 – Different DA Theories for classification.

Divergence	Data Generation	Non-binary Loss
$\mathcal{H}$-divergence (Ben-David et al., 2010a)	$x \sim \mathcal{D}(x), y = h^\star(x)$ $\lvert\mathcal{Y}\rvert = 2$	×
Discrepancy (Mansour et al., 2009a)		√
Jensen-Shannon	$x \sim \mathcal{D}(x), y \sim \mathcal{D}(y\lvert x)$ $\lvert\mathcal{Y}\rvert \geq 2$	√

Corollary 4.3. *The upper bound in Theorem 4.1 can be further decomposed as:*

$$\begin{aligned} R_{\mathcal{T}}(h) \leq \quad & R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}} \underbrace{\sqrt{D_{\text{JS}}(\mathcal{T}(x) \| \mathcal{S}(x))}}_{\textit{Feature Marginal Shift}} + \frac{G}{\sqrt{2}} \underbrace{\sqrt{\mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{JS}}(\mathcal{T}(y|x) \| \mathcal{S}(y|x))}}_{\textit{Label Conditional Shift}} \\ & + \frac{G}{\sqrt{2}} \underbrace{\sqrt{\mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{JS}}(\mathcal{T}(y|x) \| \mathcal{S}(y|x))}}_{\textit{Label Conditional Shift}} \end{aligned} \tag{4.3}$$

$$\begin{aligned} R_{\mathcal{T}}(h) \leq \quad & R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}} \underbrace{\sqrt{D_{\text{JS}}(\mathcal{T}(y) \| \mathcal{S}(y))}}_{\textit{Label Marginal Shift}} + \frac{G}{\sqrt{2}} \underbrace{\sqrt{\mathbb{E}_{y\sim\mathcal{S}(y)} D_{\text{JS}}(\mathcal{T}(x|Y=y) \| \mathcal{S}(x|Y=y))}}_{\textit{Semantic (Feature) Conditional Shift}} \\ & + \frac{G}{\sqrt{2}} \underbrace{\sqrt{\mathbb{E}_{y\sim\mathcal{T}(y)} D_{\text{JS}}(\mathcal{T}(x|Y=y) \| \mathcal{S}(x|Y=y))}}_{\textit{Semantic (Feature) Conditional Shift}} \end{aligned} \tag{4.4}$$

The derivation is delegated in Appendix Sec. C.4. In particular, Eq. (4.4) provides an alternative direction for understanding DA. The target risk bound is alternatively controlled by the label marginal shift and the semantic (feature) conditional distribution shift. Generally, the source and target label marginal distribution, as well as the semantic (feature) conditional distributions are *both different*. For example, in the classification of different digit datasets (e.g., MNIST, USPS), when conditioning on the certain digit $Y = y$, it is clear that $\mathcal{S}(x|Y = y) \neq \mathcal{T}(x|Y = y)$, indicating the necessity of considering semantic information in DA.

4.3.3 Theoretical Applications

One fundamental challenge in DA is to discover the relations and inherent properties of learning tasks, that ensure a successful transfer (Ben-David et al., 2010b). Jensen-Shannon divergence enables us to analyze the factor of label space that influences the transfer procedure, illustrated in two concrete scenarios.

Application I: Target Intrinsic Error In DA

To characterize the inherent difficulty in the learning a task, we adopt the conditional entropy $H(Y_{\mathcal{D}}|X_{\mathcal{D}}) = \mathbb{E}_{x\sim\mathcal{D}(x)} H(Y|X = x)$ as the intrinsic error, an error in predicting the labels given that the underlying data distribution $\mathcal{D}$ is known (Achille and Soatto, 2018; Zhang et al., 2020a). For example, if X does not provide any information for the label Y such that $Y \perp\!\!\!\perp X$, then the conditional entropy arrives its maximum: $H(Y|X) = H(Y)$, indicating the impossibility to guarantee a small prediction error. However, in the context of $\mathcal{H}$-divergence (Ben-David et al., 2010a), this property *can not be analyzed* since the label is determined by a *fixed* labeling function, such that $H(Y|X) = \mathbb{E}_{x\sim\mathcal{D}(x)} H(h^\star(x)|X = x) \equiv 0$.

Target Intrinsic Error: Upper Bound In the context of DA, our goal is to ensure a small target risk, i.e., a small target intrinsic error is necessary. However, we never have the full target distribution $\mathcal{T}(x, y)$, indicating the impossibility to directly estimate target intrinsic error $H(Y_t|X_t)$. In contrast, we can have the information of source distribution, as well as the relations of source and target distribution. Then we can derive the target intrinsic error is controlled by the label space size, as well as the source intrinsic error and Jensen-Shannon divergence of two distributions. This result is also consistent with our intuition and the lower bound derived by Fano's inequality (Polyanskiy and Wu, 2019): a smaller label space $|\mathcal{Y}|$ is generally easier to learn, if the other conditions are identical.

Theorem 4.3. *If we have:*

1. *Small source intrinsic error:* $H(Y_s|X_s) \leq \epsilon$*,*
2. *Marginal distributions defined in Eq. (4.3) are close:* $D_{JS}(\mathcal{S}(x)\|\mathcal{T}(x)) \leq \delta_1$*,*
3. *Conditional distributions defined in Eq. (4.3) are close:* $D_{JS}(\mathcal{S}(y|X = x)\|\mathcal{T}(y|X = x)) \leq \delta_2$*,* $\forall x$*,*

Then the target intrinsic error can be upper bounded by:

$$H(Y_t|X_t) \leq \epsilon + \sqrt{\frac{\delta_2}{2}} + \frac{\sqrt{\delta_1}}{2} \log |\mathcal{Y}|.$$

Application II: Inherent Difficulty in Learning Open Set DA

Our theory also proposes the analysis to understand when and what is difficult to transfer in Open Set DA, i.e., the source and target domain share only a portion of label space (Cao et al., 2018; You et al., 2019; Panareda Busto and Gall, 2017).

The key observation in the Open Set DA is that $\text{supp}\{\mathcal{T}(y)\} \cap \text{supp}\{\mathcal{S}(y)\} \neq \emptyset$. We suppose a small semantic conditional shift ($\forall y$, $D_{\text{JS}}(\mathcal{S}(x|y)\|\mathcal{T}(x|y)) \leq \delta$ for a small $\delta > 0$), and a uniform label distributions over two different label spaces $\mathcal{Y}_1$ and $\mathcal{Y}_2$ such that $\mathcal{S}(y) \sim \text{Unif}(\mathcal{Y}_1)$, $\mathcal{T}(y) \sim \text{Unif}(\mathcal{Y}_2)$, $|\mathcal{Y}_1| = |\mathcal{Y}_2| = N$. We further assume the number of shared classes is $|\mathcal{Y}_1 \cap \mathcal{Y}_2| = \alpha N$, $0 < \alpha < 1$. Then if the loss is binary and based on Theorem 4.2 and Eq. (4.4), the target risk can be bounded:

$$R_{\mathcal{S}}(h) - \left(\sqrt{1-\alpha} + 2\sqrt{\delta}\right) \leq R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \frac{1}{\sqrt{2}}\left(\sqrt{1-\alpha} + 2\sqrt{\delta}\right).$$

When $\alpha \to 1$, $D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y)) \to 0$, the source risk is approaching the target risk from the two sides, then simply minimizing the source risk and further semantic conditional matching (see Sec. 4.4) can effectively control the target risk. On the contrary, if $\alpha \to 0$, the gap between target and source risk is large, indicating that a small source risk and semantic conditional shift no more guarantee a small target risk. From the practical perspective, less label overlapping means that it is harder to transfer the exact corresponding semantic conditional information from the source to the target.

Table 4.2 – Empirical Methods for Bi-Directional Marginal/Conditional Shifts

Corollary 4.3		Source	Marginal Shift	Conditional Shift
Eq.(4.3)	Term	$R_{\mathcal{S}}(h)$	$D_{\text{JS}}(\mathcal{T}(z)\\|\mathcal{S}(z))$	$D_{\text{JS}}(\mathcal{T}(y\|z)\\|\mathcal{S}(y\|z))$
	Method	ERM	Feature Marginal Matching	N/A
Eq.(4.4)	Term	$R_{\mathcal{S}}(h)$	$D_{\text{JS}}(\mathcal{T}(y)\\|\mathcal{S}(y))$	$D_{\text{JS}}(\mathcal{T}(z\|y)\\|\mathcal{S}(z\|y))$
	Method	Label Marginal Shift Correction		Semantic Distribution Matching

4.4 Practical Principles for Unsupervised DA

In this section, we instantiate our theoretical framework with practical principles for designing unsupervised DA algorithms in deep learning. We would like to point out that our theory is based on the labelled data information and the practical principle can be applied in the unsupervised scenario. In addition, our results not only reaffirm the principles induced by $\mathcal{H}$-divergence, but also motivate new DA practice in the representation learning.

We introduce a *feature learning function* $g : \mathcal{X} \to \mathcal{Z}$ and denote latent variable (feature) $z = g(x)$. Our objective is to find a representation function g and classifier h, following the principles in Tab. 4.2. We also denote $\hat{\mathcal{S}}(x, y) = \{(x_s^i, y_s^i)\}_{i=1}^{N_s}$, $\hat{\mathcal{T}}(x) = \{x_t^i\}_{i=1}^{N_t}$ as the observed (empirical) distribution.

4.4.1 Inherent Practical Difficulty for Controlling Label Conditional Shift

Equation (4.3) in Corollary 4.3 recovers the principles induced by $\mathcal{H}$-divergence. Specifically, the domain adversarial training is equivalent to minimize the dual form of Jensen-Shannon divergence (Nowozin et al., 2016) i.e. $\min_g D_{\text{JS}}(\hat{\mathcal{T}}(z)\|\hat{\mathcal{S}}(z))$.

However, domain adversarial training *cannot* guarantee a small upper bound in Corollary 4.3. To this end, we can prove that merely minimizing $D_{\text{JS}}(\hat{\mathcal{T}}(z)\|\hat{\mathcal{S}}(z))$ can lead to an increase in the label conditional shift $D_{\text{JS}}(\hat{\mathcal{T}}(y|z)\|\hat{\mathcal{S}}(y|z))$, which is illustrated as the following:

$$\begin{aligned}
&\mathbb{E}_{z\sim\hat{\mathcal{T}}(z)} D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) + \mathbb{E}_{z\sim\hat{\mathcal{S}}(z)} D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) \\
&\qquad\qquad\qquad\qquad \geq 2\left(\sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))} - \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))}\right)^2
\end{aligned}$$

The aforementioned inequality indicates that the third term in Eq.4.3 is lower bounded by the gap between $D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))$ and $D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))$, then merely minimizing $D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))$ will be problematic if their label distributions are significantly different.

Moreover, controlling the label condition shift is practically difficult. Because it requires two identical *continuous* and high dimensional features such that $z_s = z_t$ with $z_s \in \hat{\mathcal{S}}(z)$, $z_t \in \hat{\mathcal{T}}(z)$, then minimizing $D_{\text{JS}}(\hat{\mathcal{T}}(y|Z = z_s)\|\hat{\mathcal{S}}(y|Z = z_t))$. Generally, it is not trivial to find such feature pairs $z_s = z_t$, only from finite observational samples.

4.4.2 Novel Practice

According to Eq. (4.4) in Corollary 4.3, the target risk can be alternatively bounded by $R_{\mathcal{S}}(h)$, label marginal shift $D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))$, and semantic (feature) conditional shift $D_{\text{JS}}(\mathcal{T}(z|y)\|\mathcal{S}(z|y))$, which enables us to consider new principles in DA.

(I) Semantic Conditional Distribution Matching Different from the controlling the label conditional shift $D_{\text{JS}}(\hat{\mathcal{T}}(y|Z=z)\|\hat{\mathcal{S}}(y|Z=z))$, controlling the semantic (feature) conditional shift $D_{\text{JS}}(\hat{\mathcal{T}}(z|Y=y)\|\hat{\mathcal{S}}(z|Y=y))$ is practically more efficient, since *labels are usually categorical variables with the finite classes*, comparing with continuous latent variable Z. However, there are no ground truth labels on the target domain, inducing the main issue in semantic conditional matching in DA. For addressing this concern, target *pseudo labels* Y_p, estimated from the classifier, are introduced as the approximation of the real target label. Then following insights of the third term in Eq. (4.4), the semantic conditional loss can be expressed as:

$$\sum_y (\hat{\mathcal{S}}(y) + \hat{\mathcal{T}}_p(y)) D_{\text{JS}} \left(\hat{\mathcal{T}}(z|Y_p = y) \| \hat{\mathcal{S}}(z|Y = y) \right), \tag{4.5}$$

where $\hat{\mathcal{T}}_p(y)$ is the target pseudo distribution predicted by the neural network. We notice that Long et al. (2018) alternatively encoding the label prediction information $h \circ g(x)$ as the conditional domain adversarial training, to implicitly minimize the conditional distribution divergence. However, semantic conditional matching requires *relative good pseudo-label prediction*. Otherwise the incorrect semantic (feature) feature alignment will lead to a negative transfer procedure for the target domain during the learning phase.

(II) Label Marginal Shift Correction *Is the semantic conditional matching sufficient to control the target risk?* From Eq. (4.4), the target risk is also controlled by label marginal shift. We can further extend this conclusion in the representation learning: if the semantic conditional distribution is matched, then the target risk is still controlled by the label marginal shift.

Theorem 4.4. *If any classifier h, feature learner g, and label $y \in \mathcal{Y} = \{-1, +1\}$ such that semantic conditional distribution is matched, $D_{JS}(\mathcal{S}(z|y), \mathcal{T}(z|y)) = 0$, then the target risk can be bounded:*

$$R_{\mathcal{S}}(h \circ g) - \sqrt{2D_{JS}(\mathcal{S}(y), \mathcal{T}(y))} \leq R_{\mathcal{T}}(h \circ g) \leq R_{\mathcal{S}}(h \circ g) + \sqrt{2D_{JS}(\mathcal{S}(y), \mathcal{T}(y))},$$

where $R_{\mathcal{S}}(h \circ g) = R_{\mathcal{S}}(h(g(x), y))$ is the expected risk over the classifier h and feature learner g.

The proof is delegated in Appendix Sec. C.7.1. As Theorem 4.4 suggests, we need to control label marginal shift $D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))$. Therefore we adopt the popular label re-weighted loss (Cortes et al., 2010):

$$\hat{R}^{\alpha}_{\mathcal{S}}(h \circ g) = \sum_{(x_s, y_s) \sim \hat{\mathcal{S}}(x,y)} \alpha(y_s) L(h(g(x_s), y_s))$$

with $\alpha(y) = \frac{\mathcal{T}(y)}{\mathcal{S}(y)}$. In addition, we can further prove the empirical re-weighted loss converges to $R_{\mathcal{T}}(h \circ g)$, if $D_{\text{JS}}(\mathcal{S}(z|y), \mathcal{T}(z|y)) = 0$ (see Appendix for details). As for estimating label weight $\hat{\alpha}$

from the data, several approaches have been proposed, e.g. Black Box Shift Learning (BBSL) (Lipton et al., 2018) or Regularized Learning under Label Shift (RLSS) (Azizzadenesheli et al., 2019).

(III) Feature Marginal Matching as the Constraint Although the aforementioned principles are theoretically appealing, we practically use the pseudo label Y_p for the semantic conditional matching $D_{\text{JS}}(\hat{\mathcal{T}}(z|Y_p = y)\|\hat{\mathcal{S}}(z|Y = y))$, which can lead to negative transfer in the training loop if we face poor pseudo label predictions.

Can we derive a principle to recognize poor pseudo label prediction during learning? Theorem 4.5 reveals one consequence of poor target pseudo label prediction: it can lead to a large empirical feature marginal divergence $D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))$ (in Eq. (4.3)), under mild conditions.

Theorem 4.5. *We denote $\hat{\mathcal{S}}_p(y), \hat{\mathcal{T}}_p(y)$ as the prediction output (pseudo-label) distributions. If we have such a "bad" pseudo label prediction such that $D_{JS}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{T}}_p(y)) = P$, small source prediction error $D_{JS}(\hat{\mathcal{S}}(y)\|\hat{\mathcal{S}}_p(y)) \leq \epsilon_1$ and small label ground truth empirical distribution divergence $D_{JS}(\hat{\mathcal{S}}(y)\|\hat{\mathcal{T}}(y)) \leq \epsilon_2$, then the feature marginal divergence on the latent space Z can be lower bounded by:*

$$D_{JS}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z)) \geq (\sqrt{P} - \sqrt{\epsilon_1} - \sqrt{\epsilon_2})^2.$$

The proof is delegated in Appendix Sec. C.7.3. From Theorem 4.5, if $P \to 1$ and ϵ_1, ϵ_2 are small, $D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))$ can be very large. Therefore we add the constraint $\mathcal{D}_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z)) \leq \kappa$ as a broad adaptation step, to prevent the poor pseudo-label prediction (a.k.a. a large P).

Practical Guideline Based on these three principles, we propose a generic and iterative practical framework, where parameter optimization and pseudo-label prediction steps are conducted iteratively.

Moreover, we would like to emphasize that the realization of each principle is flexible. For example, the distribution matching can be done through either adversarial training by introducing the auxiliary domain discriminator d or parametric distribution matching (e.g. statistical moment matching approach). More empirical choices can be found in the Appendix.

Parameter Optimization Step (fixed Pseudo-Labels) classifier h and feature extractor g:

$$\min_{h,g} \quad \underbrace{\hat{R}^{\hat{\alpha}}_{\hat{\mathcal{S}}}(h(g(x),y))}_{\text{(I)}} + \underbrace{\sum\nolimits_y (\hat{\mathcal{S}}(y)+\hat{\mathcal{T}}_p(y)) D_{\text{JS}}\Big(\hat{\mathcal{T}}(g(x)|Y_p=y)\|\hat{\mathcal{S}}(g(x)|Y=y)\Big)}_{\text{(II)}}$$

$$\text{s.t.} \quad \underbrace{D_{\text{JS}}(\hat{\mathcal{T}}(g(x))\|\hat{\mathcal{S}}(g(x))) \leq \kappa}_{\text{(III)}}$$

(I) Label marginal shift correction: $\hat{R}^{\hat{\alpha}}_{\hat{\mathcal{S}}}(h(g(x),y)) = \sum_{(x_s,y_s)\sim\hat{\mathcal{S}}} \hat{\alpha}(y_s) L(h(g(x_s),y_s))$; (II) Semantic conditional matching, aligning the semantic feature; (III) Feature marginal matching as the constraint, a broad adaptation step to prevent a poor initialization of pseudo label prediction.

Pseudo-Label Prediction Step (fixed Parameters): y^p, $\hat{\alpha}$, $\hat{\mathcal{T}}_p(y)$
y^p,$\hat{\mathcal{T}}_p(y)$ are pseudo-labels and distributions on the target domain. $\hat{\alpha}$ is the reweighting coefficient.

4.5 Related Work

DA theory An important aspect in DA is to establish the proper theory to understand how it influences the target risk. The most popular approach is based on $\mathcal{H}$-divergence (Ben-David et al., 2010a), which is set on the deterministic labeling function and binary loss. Then, variants of hypothesis based discrepancy have been proposed such as distribution discrepancy (Cortes et al., 2019), Margin disparity discrepancy (Zhang et al., 2019), etc. However, these theoretical results mainly focus on the relation of feature marginal discrepancy $d(\mathcal{S}(x)\|\mathcal{T}(x))$ and the difficulty to analyze various scenarios such as target shift, open-set DA, etc.

An alternative is to adopt the statistical divergence. Mansour et al. (2009b) proposed Rényi-α divergence to measure the feature marginal discrepancy. Then Germain et al. (2016); Hoffman et al. (2018) analyzed Rényi divergence on the joint source target distribution, with binary and cross entropy loss, respectively. However, they generally focus on covariate shift settings by assuming $\mathcal{S}(y|x)=\mathcal{T}(y|x)$. Moreover, the aforementioned theories did not discuss the inspired practice under the representation learning, which restricts its utility in deep learning. Another popular choice is the Wasserstein distance such as (Shen et al., 2018), but still focus on the feature marginal distance $W_1(\mathcal{S}(x)\|\mathcal{T}(x))$, since the chain rule generally does not hold on Wasserstein distance, then it is difficult to derive the label shift and semantic conditional shift simultaneously.

DA principles for the representation learning Deriving principles for DA problems in the representation learning is crucial for the real-world applications. From the conventional DA theories such as (Ben-David et al., 2010a,b; Ben-David and Urner, 2014; Germain et al., 2013; Johansson et al., 2019), a small joint optimal risk β is important to ensure a small target risk. Therefore, different empirical approaches have speculated various ideas to control a small β. From the theoretical prospective, (Zhao et al., 2019a) adopted Jensen-Shannon divergence to derive the lower bound of β, indicating necessarily of considering the target shift. However, it is still not clear how the algorithms explicitly guarantee a small β. Indeed, our work can further extend this by proving a new theoretical upper bound through target shift and feature conditional shift, which enable the possible practice to explicitly control the target risk.

We also notice that Zhang et al. (2013); Gong et al. (2016) analyzed feature conditional shift from the causal prospective in RKHS space, which is generally difficult to adapt in the large-scale dataset. From empirical aspects, Li et al. (2019b); Tan et al. (2019); Long et al. (2013); Saito et al. (2017); Chen et al. (2019a); Xie et al. (2018); Cai et al. (2019) proposed various strategies for eliminating conditional shift, which speculated one or two principles to improve the prediction performance. We formally demonstrate the unified three principles, as a way to control the target risk. In addition, our D_{JS} analysis provides justifications to explain these empirical success e.g, Cai et al. (2019), which in fact are *not particularly focused on previous theories but already achieved meaningful results for current deep DA problems.*

4.6 Experiments

We validate the proposed guideline by realizing each principle. We aim to show whether applying the unified principles is better than merely considering only one or two of them.

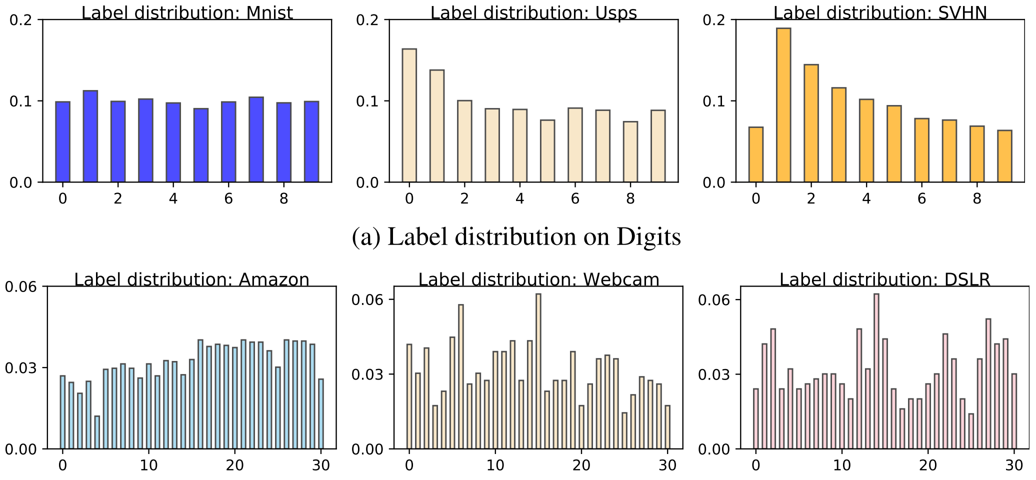
4.6.1 Experimental Settings

We evaluate the proposed framework on digit recognition and the Office-31 dataset.

Digits Recognition. It includes 3 domains: MNIST, SVHN (Netzer et al., 2011) and USPS (Hull, 1994) dataset. MNIST is composed of grey images of size 28×28, USPS contains 16×16 grey digits; and SVHN consists of 32×32 color digit images, which are more challenging and can contain more than one digit in each image. We randomly sample 7K samples for each task. We evaluate our method by using the three typical adaptation tasks: USPS↔MNIST (two tasks) and SVHN→MNIST (one task).

Office-31 dataset (Saenko et al., 2010). It consists of 4,652 images and 31 categories collected from three different domains: Amazon (A) from amazon.com, Webcam (W) and DSLR (D), taken by web camera and digital SLR camera in different environmental settings, respectively.

We further visualize the label distribution of digits and Office-31, showing in Fig. 4.2. We observe the non-uniform label distributions over these two tasks. We implement the digits dataset based on the LeNet5 (LeCun et al., 1998). All digit images are resized to 28×28 for fair comparisons. As for Office-31 task, we implement it on the Pre-trained AlexNet (Krizhevsky et al., 2012). We use the same



(a) Label distribution on Digits

(b) Label distribution on Office-31

Figure 4.2 – Label distribution on two Datasets

hyper-parameter training strategy with DANN (Ganin et al., 2016). We illustrate our practical pipeline in the attached source code. We update the neural network parameters and $\hat{\alpha}$, iteratively (frequency: source batch number $\times 100$). We compare the baselines of merely considering feature marginal (DANN (Ganin et al., 2016)), conditional matching (CDAN (Long et al., 2018)), and our principles. We repeat the experiments five times and report the average and std.

4.6.2 Algorithm Design

The proposed principles consist three components:

Source re-weighting loss We use the cross-entropy function as the prediction loss. We estimate the $\hat{\alpha}$ by BBSL approach (Lipton et al., 2018) with

$$\hat{\alpha} = \hat{C}^{-1}\hat{\mathcal{T}}_p,$$

where $\hat{C}$ is source prediction confusion matrix with $\hat{C}[i,j] = \mathbb{P}(h(g(x_s)) = i, y_s = j)$. In practice, we iteratively update $\hat{\alpha}$ and network parameters.

Semantic Conditional Matching $D_{\text{JS}}(\hat{\mathcal{S}}(z|y)\|\hat{\mathcal{T}}(z|y))$. For each label $Y = y$, we align its first-order statistics (feature mean matching). i.e, we aim at optimizing

$$\left\| \frac{1}{|\#y_s = y|} \sum_{(x_s,y_s)} \delta_{\{y_s=y\}} g(x_s) - \frac{1}{|\#y_t^p = y|} \sum_{(x_t,y_t^p)} \delta_{\{y_t^p=y\}} g(x_t) \right\|_2 ,$$

where $|\#y_s = y|$ is the number of $y_s = y$, which is the approximation of d_{TV}, the upper bound of D_{JS}.

Algorithm 3 Jensen-Shannon Principles in Unsupervised DA

```
Require: Labeled source $\hat{\mathcal{S}}$, Unlabelled Target $\hat{\mathcal{T}}$
Ensure: Label distribution ratio $\hat{\alpha}$. Feature Learner $g$, Classifier $h$, Statistic critic function $d_1, \dots, d_T$,
    class centroid for source $\mathbf{C}_s^y$ and target $\mathbf{C}^y$ ($\forall y \in \mathcal{Y}$).
▷ ▷ ▷ DNN Parameter Training Stage (fixed $\hat{\alpha}$) ◁ ◁ ◁
for mini-batch of samples $(\mathbf{x}_{\mathcal{S}}, \mathbf{y}_{\mathcal{S}}) \sim \hat{\mathcal{S}}$, $(\mathbf{x}_{\mathcal{T}}) \sim \hat{\mathcal{T}}$ do
    Predict target pseudo-label $\bar{\mathbf{y}}_{\mathcal{T}} = \text{argmax}_y h(g(\mathbf{x}_{\mathcal{T}}), y)$
    Compute source confusion matrix for each batch (un-normalized)
        $C_{\hat{\mathcal{S}}} = \#[\text{argmax}_{y'} h(z, y') = y, Y = k]$
    Compute the *batched* class centroid for source $C_s^y$ and target $C^y$.
    Moving Average for update source/target class centroid:
    Source class centroid update
    $\mathbf{C}_s^y = \epsilon_1 \times \mathbf{C}_s^y + (1 - \epsilon_1) \times C_s^y, \forall y \in \{1, \dots, \mathcal{Y}\}$
    Target class centroid update
    $\mathbf{C}^y = \epsilon_1 \times \mathbf{C}^y + (1 - \epsilon_1) \times C^y$
    Computing the approximation semantic conditional loss $\sum_y (\hat{\mathcal{S}}(y) + \hat{\mathcal{T}}_p(y)) \|\mathbf{C}_s^y - \mathbf{C}^y\|_2^2$
    Updating $g, h, d$ (SGD and Gradient Reversal), to minimize the loss.
end for
▷ ▷ ▷ Estimation $\hat{\alpha}$ ◁ ◁ ◁
Compute the global(normalized) source confusion matrix
$C_{\hat{\mathcal{S}}} = \hat{\mathcal{S}}[\text{argmax}_{y'} h(z, y') = y, Y = k]$ $(t = 1, \dots, T)$
Solve $\alpha$ (denoted as $\{\alpha'\}_{t=1}^T$).
Update $\alpha$ by moving average: $\alpha = \epsilon_1 \times \alpha + (1 - \epsilon_1) \times \alpha'$
```

Table 4.3 – Accuracy (%) on Digits Dataset

Method	SVHN → MNIST	MNIST → USPS	USPS → MNIST
Without DA	62.1±1.2	87.1±0.9	78.1±0.6
DANN (Ganin et al., 2016)	73.8±1.8	89.1±0.6	83.0±0.8
CDAN (Long et al., 2018)	86.7±0.8	93.2±0.6	93.0±0.5
(I + III)	76.7±0.8	89.4±0.7	84.6±1.4
(I + II)	87.3±0.6	94.6±0.7	94.7±0.5
(II + III)	88.6±0.9	95.5±0.8	95.5±0.7
(I + II + III)	**89.6**±1.1	**96.5**±0.6	**97.0**±0.6

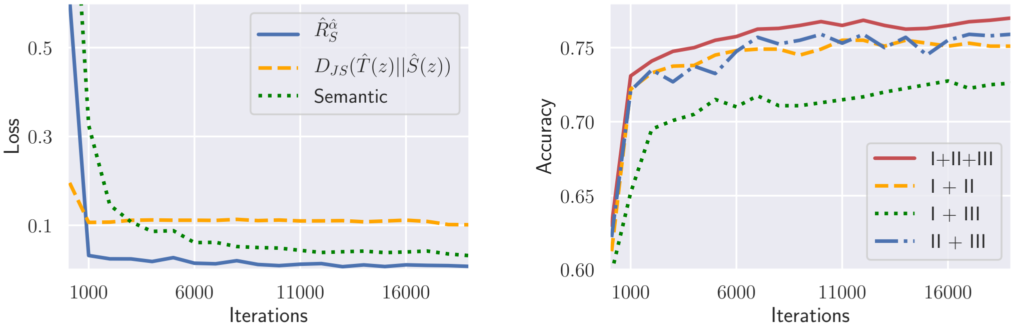
Feature marginal matching $D_{\text{JS}}(\hat{\mathcal{T}}(z) \| \hat{\mathcal{S}}(z))$ as the constraint. We adopt the Lagrangian relaxation for treating the constraint as the regularization, with κ as the hyper-parameter. We use the dual term of D_{JS} in Eq. (4.2) to estimate the JS divergence. Thus this term can be rewritten as $\max_d \kappa[\mathbb{E}_{x_s \sim \mathcal{S}(x)} \log(d \circ g(x_s)) + \mathbb{E}_{x_t \sim \mathcal{T}(x)} \log(1 - d \circ g(x_t))]$. In the experiments, we denote $\kappa = 0.01$.

4.6.3 Results and Analysis

We report the empirical performances in Tab.4.3 and 4.4. The empirical results indicate the improved performance of the unified principles, comparing with merely one or two principles. We observe that

Table 4.4 – Accuracy (%) on Office-31 Dataset

Method	A → D	A → W	D → W	W → D	W → A	D → A	Ave
Without DA	63.8±0.5	61.6±0.5	95.4±0.3	99.0±0.2	49.8±0.4	51.1±0.6	70.1
DANN (Ganin et al., 2016)	72.3±0.3	73.0±0.5	96.4±0.3	99.2±0.3	51.2±0.5	52.4±0.4	74.1
CDAN (Long et al., 2018)	76.3±0.1	78.3±0.2	97.2±0.1	**100.0**±0.0	57.5±0.4	57.3±0.2	77.7
(I + III)	72.6±0.4	73.5±0.4	96.2±0.2	99.3±0.5	51.4±0.2	52.8±0.5	74.3
(I + II)	75.3±0.7	79.4±1.1	97.1±0.5	97.5±0.5	58.2±0.9	61.8±0.8	78.2
(II + III)	75.7±0.1	79.2±0.7	96.8±0.1	99.8±0.1	59.5±0.4	58.7±0.3	78.3
(I + II + III)	**76.7**±0.4	**80.8**±0.4	**97.5**±0.2	99.8±0.1	**59.8**±0.4	**62.3**±0.2	**79.5**



(a) (b)

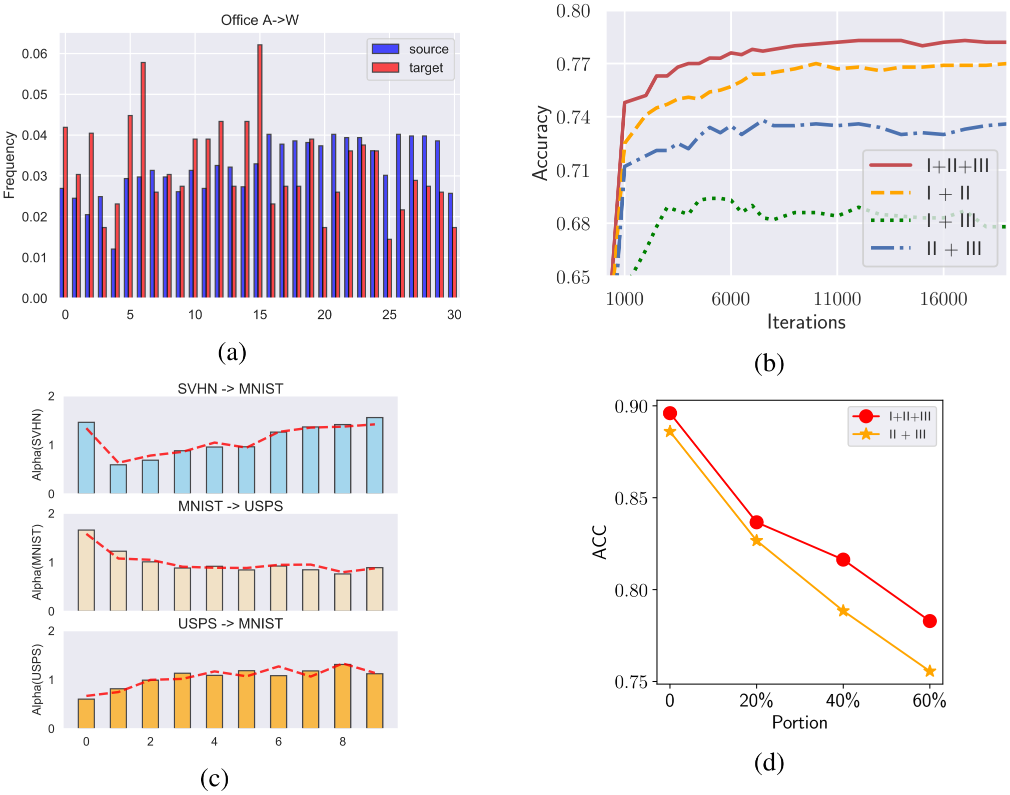
Figure 4.3 – Analysis of proposed principles. (a) Office-31, Domain A→D. Evolution of each loss during the training. (b) Office-31, Domain A→D. Evolution of accuracy during the training.

the empirical benefit of semantic conditional matching (II) is most notable.

Fig. 4.3 further reveals the properties of the proposed principles. Specifically, Fig. 4.3(a) shows the evolution of each principle (loss) during the training, which is coherent with the goals in the guideline. The semantic conditional shift (Principle II) and the weighted source classification error (Principle I) gradually diminish and $D_{\text{JS}}(\hat{\mathcal{T}}(z)\|\hat{\mathcal{S}}(z))$ (Principle III) restricts within a small value. In addition, we trace the target domain prediction accuracy of different principles combinations in Fig. 4.3(b), for demonstrating the impact of each principle. The results indicate the importance of considering semantic (feature) conditional distribution matching (II), with a significant performance influence ($\sim 4.2\%$). On the other hand, the influences of principle (I) and (III) are relatively modest ($\sim 1.3\%$). Fig. 4.4(c) revealed estimated $\hat{\alpha}$ and its ground truth value, which verified the correctness of the proposed principle.

Ablation Study: Label-drifted DA To further elaborate the role of proposed principles, we simulate a significant label drifting in DA.

In Office-31 (A→ W dataset), we randomly drop out 25% samples in the first half of classes within source, and 25% samples in latter half of classes in target. We visualize result in Fig.4.4 (b), which verifies the strong practical benefits of semantic conditional matching (principle II, with improvement $\sim 5.8 - 9.4\%$). Besides, principle (III) empirically offers a coarse adaptation step to improve pseudo-



(a)

(b)

(c)

(d)

Figure 4.4 – Ablation Study: Label-drifted DA. (a) Office-31. Label distribution of drifted A→W. (b) Label drifted A→W. Evolution of accuracy of different principles during the training. (c) In digits dataset, we visualize the estimated $\hat{\alpha}$ (red dot curve) and ground truth value (bar plot). (d) Label drifted Digits, SVHN → MNIST. Evolution of accuracy under different label drifts.

label prediction. E.g., in Fig.4.4 (b), introducing principle (III) improves the prediction performance by $\sim 1.1\%$ In Digits dataset (SVHN $\rightarrow$ MNIST), based on (Li et al., 2019b), we randomly drop out different portion $\%$ of samples in latter half of classes (i.e digits $5-9$) in source domain. To show the role of label shift correction, we visualize the results in Fig.4.4 (c). We observe that in a relative large label drift, the re-weighted loss (principle I) improves $\sim 3\%$ performance.

4.7 Conclusion

We proposed a new theoretical framework based on Jensen-Shannon divergence for analyzing DA problems. Our theory established bi-directional marginal/conditional shifts for the target risk bound. We further demonstrated its flexibility in various theoretical and algorithmic applications. It is worth mentioning that our theoretical framework is not only suitable for DA, but also extendable to analyzing the real shift problems such as fair representation learning (Louizos et al., 2015; Edwards and Storkey, 2015), individual treatment effect estimation (Shalit et al., 2017).

Chapter 5

Aggregating From Multiple Target-Shifted Sources

Original title of the article: **Aggregating From Multiple Target-Shifted Sources**

Résumé

L’adaptation à un domaine multisource vise à exploiter les connaissances de plusieurs tâches pour prédire un domaine cible connexe. Par conséquent un aspect crucial consiste à combiner correctement différentes sources en fonction de leurs relations. Dans cet article, nous avons analysé le problème de l’agrégation de domaines sources avec différentes distributions d’étiquettes, où la plupart des approches récentes de sélection de sources échouent. L’algorithme que nous proposons diffère des approches précédentes sur deux points essentiels: le modèle agrège plusieurs sources principalement par la similarité de la distribution conditionnelle sémantique plutôt que par la distribution marginale; le modèle propose un cadre *unifie* pour sélectionner les sources pertinentes pour trois scénarios populaires, à savoir l’adaptation de domaine avec une étiquette limitée sur le domaine cible, l’adaptation de domaine non supervisée et l’adaptation de domaine non supervisée partielle par étiquette. Nous évaluons la méthode proposée par des expériences. Les résultats empiriques surpassent de manière significative les résultats de base en particulier pour l’adaptation partielle du domaine.

Abstract

Multi-source domain adaptation aims at leveraging the knowledge from multiple tasks for predicting a related target domain. Hence, a crucial aspect is to properly combine different sources based on their relations. In this paper, we analyzed the problem for aggregating source domains with different label distributions, where most recent source selection approaches fail. Our proposed algorithm differs from previous approaches in two key ways: the model aggregates multiple sources mainly

through the similarity of semantic conditional distribution rather than marginal distribution; the model proposes a *unified* framework to select relevant sources for three popular scenarios, i.e., domain adaptation with limited label on target domain, unsupervised domain adaptation and label partial unsupervised domain adaption. We evaluate the proposed method through extensive experiments. The empirical results significantly outperform the baselines, especially for partial domain adaptation.

5.1 Introduction

In various real-world applications, we want to transfer knowledge from *multiple sources* $(\mathcal{S}_1, \dots, \mathcal{S}_T)$ to build a model for the target domain, which requires an effective selection and leveraging the *most useful* sources. Clearly, solely combining all the sources and applying one-to-one single DA algorithm can lead to undesired results, as it can include irrelevant or even untrusted data from certain sources, which can severely influence the performance (Zhao et al., 2020).

To select related sources, most existing works (Zhao et al., 2018; Peng et al., 2019; Li et al., 2018a; Wen et al., 2020) used the marginal distribution similarity $(\mathcal{S}_t(x), \mathcal{T}(x))$ to search the similar tasks. However, this can be problematic if their label distributions are different. As illustrated in Fig. 5.1, in a binary classification, the source-target marginal distributions are identical $(\mathcal{S}_1(x) = \mathcal{S}_2(x) = \mathcal{T}(x))$, however, using $\mathcal{S}_2$ for helping predict target domain $\mathcal{T}$ will lead to a negative transfer since their decision boundaries are rather different. This is not only theoretically interesting but also practically demanding. For example, in medical diagnostics, the disease distribution between the countries can be drastically different (Liu et al., 2004; Geiss et al., 2014). Thus applying existing approaches for leveraging related medical information from other data abundant countries to the destination country can be problematic.

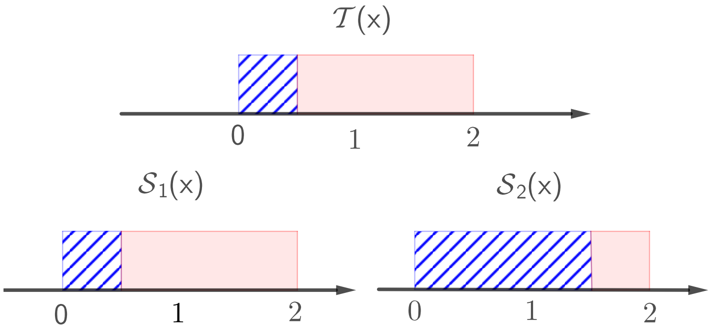


Figure 5.1 – Limitation of merely considering marginal distribution $\mathbb{P}(x)$ in the source selection. Consider the uniform distributions on x, with Unif$[0, 2]$. In binary classification, we have $\mathcal{S}_1(x) = \mathcal{S}_2(x) = \mathcal{T}(x) =$ Unif$[0, 2]$, however adopting $\mathcal{S}_2$ is worse than $\mathcal{S}_1$ for predicting target $\mathcal{T}$ due to different decision boundaries.

In this work, we aim to address multi-source deep DA under different label distributions with $\mathcal{S}_t(y) \neq \mathcal{T}(y), \mathcal{S}_t(x|y) \neq \mathcal{T}(x|y)$, which is more realistic and challenging. In this case, if the label information on $\mathcal{T}$ is absent (unsupervised DA), it is known as a underspecified problem and unsolvable *in the general case* (Ben-David et al., 2010b; Johansson et al., 2019). For example, in Figure 5.1, it is impossible to know the preferable source if there is no label information on the target domain. Therefore, a natural extension is to assume few labeled samples on target domain, which is commonly encountered in practice and a stimulating topic in recent research (Mohri and Medina, 2012; Wang et al., 2019a; Saito et al., 2019; Konstantinov and Lampert, 2019; Mansour et al., 2020). Based on this, we propose a novel DA theory with limited labeled data on $\mathcal{T}$ (Theorem 5.1, 5.2), which motivates a novel source selection strategy by mainly considering the similarity of semantic conditional distribution $\mathbb{P}(x|y)$ and source re-weighted prediction loss.

Moreover, in the *specific case*, the proposed source aggregation strategy can be further extended to the unsupervised scenarios. Concretely, in our algorithm, we assume the problem satisfies the Generalized Label Shifted (GLS) condition (Combes et al., 2020), which is related to the cluster assumption and feasible in many practical applications, as shown in Sec. 5.5. Based on GLS, we simply add a label distribution ratio estimator, to assist the algorithm in selecting related sources in two popular multi-source scenarios: unsupervised DA and unsupervised label partial DA (Cao et al., 2018) with supp$(\mathcal{T}(y)) \subseteq$ supp$(\mathcal{S}_t(y))$ (i.e., inherently label distribution shifted.)

Compared with previous work, the proposed method has the following benefits:

Better Source Aggregation Strategy We overcome the limitation of previous selection approaches when the label distributions are different by significant improvements. Notably, the proposed approach is shown to simultaneously learn meaningful task relations and label distribution ratio.

Unified Method We provide a unified perspective to understand the source selection approach in different scenarios, in which previous approaches regarded them as separate problems. We show their relations in Fig. 5.2.

5.2 Related Work

Multi-Source DA Multi-source DA has been investigated in previous literature with different aspects to aggregate source datasets. In the popular unsupervised DA, Zhao et al. (2018); Li et al. (2018b); Peng et al. (2019); Wen et al. (2020); Hoffman et al. (2018) adopted the marginal distribution $d(\mathcal{S}_t(x), \mathcal{T}(x))$ of $\mathcal{H}$-divergence (Ben-David et al., 2007), discrepancy (Mansour et al., 2009a) and Wasserstein distance (Arjovsky et al., 2017) to estimate domain relations. These works provided theoretical insights through upper bounding the target risk by the source risk, domain discrepancy of $\mathbb{P}(x)$ and an un-observable term η – the optimal risk on all the domains. However, as the counterexample indicates, relying on $\mathbb{P}(x)$ does not necessarily select the most related source. Therefore, Konstantinov and Lampert (2019); Wang et al. (2019a); Mansour et al. (2020) alternatively consider the divergence between two domains with limited target label by using $\mathcal{Y}$-discrepancy, which is commonly faced in practice and less focused in theory. However, we empirically show it is still difficult to handle target-shifted sources.

Target-Shifted DA Target-Shifted DA (Zhang et al., 2013) is a common phenomenon in DA with $\mathcal{S}(y) \neq \mathcal{T}(y)$. Several theoretical analysis has been proposed under label shift assumption with $\mathcal{S}_t(x|y) = \mathcal{T}(x|y)$, e.g. Azizzadenesheli et al. (2019); Garg et al. (2020). Redko et al. (2019) proposed optimal transport strategy for the multiple unsupervised DA by assuming $\mathcal{S}_t(x|y) = \mathcal{T}(x|y)$. However, this assumption is restrictive for many real-world cases, e.g., in digits dataset, the conditional distribution is clearly different between MNIST and SVHN. In addition, the representation learning based approach is *not* considered in their framework. Therefore, Wu et al. (2019); Combes et al. (2020) analyzed DA under different assumptions in the *embedding space* $\mathcal{Z}$ for one-to-one unsupervised deep DA problem but did not provide guidelines of *leveraging different sources* to ensure a reliable transfer, which is our core contribution. Moreover, the aforementioned works focus on one specific scenario, without considering its flexibility for other scenarios such as *partial multi-source unsupervised DA*, where the label space in the target domain is a subset of the source domain (i.e., for some classes $\mathcal{S}_t(y) \neq 0$; $\mathcal{T}(y) = 0$) and class distributions are *inherently* shifted.

5.3 Problem Setup and Theoretical Insights

Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ the output space. We consider the predictor h as a scoring function (Hoffman et al., 2018) with $h : \mathcal{X} \times \mathcal{Y} \to R$ and predicted loss as $\ell : \mathbb{R} \to \mathbb{R}_+$ is positive, L-Lipschitz and upper bound by $L_{\max}$. We also assume that h is K-Lipschitz w.r.t. the feature x (given the same label), i.e. for $\forall y$, $\|h(x_1, y) - h(x_2, y)\|_2 \leq K\|x_1 - x_2\|_2$. We denote the expected risk w.r.t distribution $\mathcal{D}$: $R_{\mathcal{D}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\, \ell(h(x, y))$ and its empirical counterpart (w.r.t. a given dataset $\hat{\mathcal{D}}$) $\hat{R}_{\mathcal{D}}(h) = \sum_{(x,y)\in\hat{\mathcal{D}}} \ell(h(x, y))$.

In this work, we adopt the commonly used Wasserstein distance as the metric to measure domains' similarity, which is theoretically tighter than the previously adopted TV distance Gong et al. (2016) and Jensen-Shnannon divergence. Besides, based on previous work, a common strategy to adjust the imbalanced label portions is to introduce *label-distribution ratio* weighted loss with $R^{\alpha}_{\mathcal{S}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{S}}\, \alpha(y)\ell(h(x, y))$ with $\alpha(y) = \mathcal{T}(y)/\mathcal{S}(y)$. We also denote $\hat{\alpha}(y)$ as its empirical counterpart, estimated from the data.

Besides, in order to measure the task relations, we define a simplex $\boldsymbol{\lambda}$ ($\boldsymbol{\lambda}[t] \geq 0, \sum_{t=1}^{T} \boldsymbol{\lambda}[t] = 1$) as the *task relation coefficient* vector by assigning higher weights to the more related task. I.e, if the source $\mathcal{S}_i$ is more related to target $\mathcal{T}$, then we assign higher $\boldsymbol{\lambda}[i]$. Then we prove Theorem 5.1, which proposes theoretical insights of combining source domains through properly estimating $\boldsymbol{\lambda}$.

Theorem 5.1. *Let* $\{\hat{\mathcal{S}}_t = \{(x_i, y_i)\}_{i=1}^{N_{\mathcal{S}_t}}\}_{t=1}^{T}$ *and* $\hat{\mathcal{T}} = \{(x_i, y_i)\}_{i=1}^{N_{\mathcal{T}}}$*, respectively be* T *source and target i.i.d. samples. For* $\forall h \in \mathcal{H}$ *with* $\mathcal{H}$ *the hypothesis family and* $\forall \boldsymbol{\lambda}$*, with high probability* $\geq 1 - 4\delta$*, the target risk can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq \underbrace{\sum_t \boldsymbol{\lambda}[t]\hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h)}_{\text{(I)}} + \underbrace{L_{\max} d^{\sup}_{\infty}\sqrt{\sum_{t=1}^{T}\frac{\boldsymbol{\lambda}[t]^2}{\beta_t}}\sqrt{\frac{\log(1/\delta)}{2N}}}_{\text{(II)}}$$

$$+ \underbrace{LK\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))}_{\text{(III)}}$$

$$+ \underbrace{L_{\max}\sup_t\|\alpha_t-\hat{\alpha}_t\|_2}_{\text{(IV)}} + \underbrace{\textit{Comp}(N_{\mathcal{S}_1},\ldots,N_{\mathcal{S}_T},N_{\mathcal{T}},\delta)}_{\text{(V)}},$$

where $N = \sum_{t=1}^{T} N_{\mathcal{S}_t}$ and $\beta_t = N_{\mathcal{S}_t}/N$ and $d^{\sup}_{\infty} = \max_{t\in[1,T],y\in[1,\mathcal{Y}]}\alpha_t(y)$ the maximum true label distribution ratio value. $W_1(\cdot\|\cdot)$ is the Wasserstein-1 distance with L_2-distance as the cost function. Comp$(N_{\mathcal{S}_1},\ldots,N_{\mathcal{S}_T},N_{\mathcal{T}},\delta)$ is a function that decreases with larger $N_{\mathcal{S}_1},\ldots,N_{\mathcal{T}}$, given a fixed δ and hypothesis family $\mathcal{H}$. (See Appendix for details)

Proof Sketch

The details of the proof can be found in the Appendix Sec. D.3, and Theorem 1 consists of three main steps.

Step 1: Expected Transfer Risk

Lemma 5.1. *If the prediction loss is assumed as L-Lipschitz and the hypothesis is K-Lipschitz w.r.t. the feature x (given the same label), i.e. for $\forall Y = y$, $\|h(x_1, y) - h(x_2, y)\|_2 \leq K\|x_1 - x_2\|_2$. Then the target risk can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq \sum_t \boldsymbol{\lambda}[t]R^{\alpha_t}_{\mathcal{S}}(h) + LK\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}(x|Y=y))$$

Step2: Bounding Empirical and Expected Risk According to the statistical learning theory, with high probability $\geq 1 - 2\delta$, we have:

$$\sum_t \boldsymbol{\lambda}[t]R^{\alpha_t}_{\mathcal{S}}(h) \leq \sum_t \boldsymbol{\lambda}[t]\hat{R}^{\hat{\alpha}_t}_{\mathcal{S}}(h) + 2\bar{\mathcal{R}}(\ell, h) + L_{\max}d^{\sup}_{\infty}\sqrt{\sum_{t=1}^{T}\frac{\boldsymbol{\lambda}[t]^2}{\beta_t}}\sqrt{\frac{\log(1/\delta)}{2N}} + L_{\max}\sup_t\|\alpha_t - \hat{\alpha}_t\|_2$$

Where $\bar{\mathcal{R}}(\ell, h)$ is the Rademacher Complexity.

Step 3: Bounding the Empirical and Expected Wasserstein Distance

$$\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)) \leq \sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))$$

$$C_{\max}\left(\sqrt{\frac{2\log(2|\mathcal{Y}|)}{N_{\mathcal{T}}}} + \sqrt{\frac{\log(T/\delta)}{2N_{\mathcal{T}}}}\right)$$

Where C_{max} is a positive constant. Then combining the conclusion in three steps, we have the aforementioned results in Theorem 5.1.

Discussions (1) In (I) and (III), the relation coefficient $\boldsymbol{\lambda}$ is decided by $\hat{\alpha}_t$-weighted loss $\hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h)$ and conditional Wasserstein distance $\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))$. Intuitively, a higher $\boldsymbol{\lambda}[t]$ is assigned to the source t with a *smaller weighted prediction loss* and *a smaller weighted semantic conditional Wasserstein distance*. In other words, the source selection depends on the similarity of the conditional distribution $\mathbb{P}(x|y)$ rather than $\mathbb{P}(x)$.

(2) If each source has equal samples ($\beta_t = 1$), then term (II) will become $\|\boldsymbol{\lambda}\|_2$, *a regularization term for the encouragement of uniformly leveraging all sources*. Term (II) is meaningful in the selection, because if several sources are simultaneously similar to the target, then the algorithm tends to select *a set of* related domains rather than only one most related domain (without regularization).

(3) Considering (I,II,III), we derive a novel source selection approach through the trade-off between assigning a higher $\boldsymbol{\lambda}[t]$ to the source t that has a smaller weighted prediction loss and similar semantic distribution with smaller conditional Wasserstein distance, and assigning balanced $\boldsymbol{\lambda}[t]$ for avoiding concentrating on one source.

(4) $\|\hat{\alpha}_t - \alpha_t\|_2$ (IV) indicates the gap between ground-truth and empirical label ratio. Therefore, if we can estimate a good label distribution ratio $\hat{\alpha}_t$, these terms can be small. $\text{Comp}(N_{\mathcal{S}_1},\dots,N_{\mathcal{S}_T},N_{\mathcal{T}},\delta)$ (V) is a function that reflects the convergence behavior, which decreases with larger observation numbers. If we fix $\mathcal{H}, \delta$, N and $N_{\mathcal{T}}$, this term can be viewed as a constant.

Analysis in the Representation Learning Apart from Theorem 5.1, we further drive theoretical analysis in the *representation learning*, which motivates practical guidelines in the deep learning regime. We define a stochastic embedding function g and we denote its conditional distribution w.r.t. latent variable Z (induced by g) as $\mathcal{S}(z|Y=y) = \int_x g(z|x)\mathcal{S}(x|Y=y)dx$. Then we have:

Theorem 5.2. *Assume the loss, the hypothesis families are the same with Theorem 5.1. We further denote the stochastic feature learning function $g : \mathcal{X} \to \mathcal{Z}$, and the hypothesis $h : \mathcal{Z}\times\mathcal{Y} \to \mathbb{R}$. Then $\forall\boldsymbol{\lambda}$, the target risk is upper bounded by:*

$$R_{\mathcal{T}}(h,g) \leq \sum_t \boldsymbol{\lambda}[t] R^{\alpha_t}_{\mathcal{S}_t}(h,g) + LK\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{S}_t(z|Y=y)\|\mathcal{T}(z|Y=y)),$$

where $R_{\mathcal{T}}(h,g) = \mathbb{E}_{(x,y)\sim\mathcal{T}(x,y)}\mathbb{E}_{z\sim g(z|x)}\ell(h(z,y))$ is the expected risk w.r.t. the function g, h.

The proof is delegated in D.4. Theorem 5.2 motivates the practice of deep learning, which requires to learn an embedding function g that minimizes the weighted conditional Wasserstein distance and learn (g,h) that minimizes the weighted source risk $R^{\alpha_t}_{\mathcal{S}_t}$.

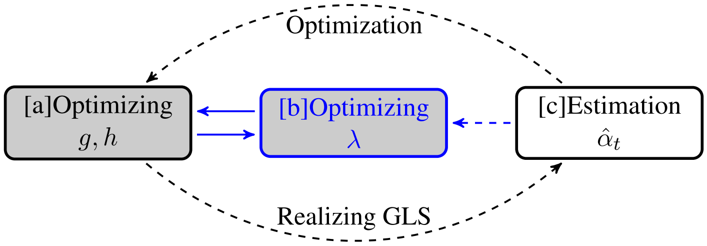


Figure 5.2 – Illustration of proposed algorithm (WADN) and relation with other scenarios. WADN consists of three components: **[a]** learning embedding function g and classifier h; **[b]** source aggregation through properly estimating $\boldsymbol{\lambda}$; **[c]** label distribution ratio ($\hat{\alpha}_t$) estimator. (1) If target labels are available, then WADN only requires **[a,b]** without gradually estimating $\hat{\alpha}_t$ (dashed arrows). (2) In the unsupervised scenarios, if we only have one source, WADN only contains **[a,c]** and recovers the single DA problem with label proportion shift, which can be solved under specific assumptions such as GLS (Li et al., 2019b; Combes et al., 2020) or (Wu et al., 2019). (3) If there are multiple sources in the unsupervised DA, WADN gradually selects the related sources through interacting with other algorithmic components. (shown in blue).

5.4 Practical Algorithm in Deep Learning

From the aforementioned theoretical results, we derive novel source aggregation approaches and training strategies, which can be summarized as follows.

Source Selection Rule Balance the trade-off between assigning a higher $\boldsymbol{\lambda}[t]$ to the source t that has a smaller weighted prediction loss and semantic conditional Wasserstein distance, and assigning balanced $\boldsymbol{\lambda}[t]$.

Training Rules (1) Learning an embedding function g that minimizes the weighted conditional Wasserstein distance, learning classifier h that minimizes the $\hat{\alpha}_t$-weighted source risk; (2) Properly estimate the label distribution ratio $\hat{\alpha}_t$.

Based on these ideas, we proposed Wasserstein Aggregation Domain Network (WADN) to automatically learn the network parameters and select related sources, where the high-level protocol is illustrated in Fig. 5.2.

5.4.1 Training Rules

Based on Theorem 2, given a fixed label ratio $\hat{\alpha}_t$ and fixed $\boldsymbol{\lambda}$, the goal is to find a representation function $g : \mathcal{X} \to \mathcal{Z}$ and a hypothesis function $h : \mathcal{Z} \times \mathcal{Y} \to \mathbb{R}$ such that:

$$\min_{g,h} \sum_t \boldsymbol{\lambda}[t] \hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h, g) + C_0 \sum_t \boldsymbol{\lambda}[t] \mathbb{E}_{y \sim \hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{S}}_t(z|Y = y) \| \hat{\mathcal{T}}(z|Y = y))$$

Explicit Conditional Loss One can *explicitly* solve the conditional optimal transport problem with g and h for a given $Y = y$. However, due to the high computational complexity in solving $T \times |\mathcal{Y}|$

optimal transport problems, the original form is practically intractable. To address this, we can approximate the conditional distribution on latent space Z as Gaussian distribution with identical Covariance matrix such that $\hat{\mathcal{S}}_t(z|Y=y) \approx \mathcal{N}(\mathbf{C}_t^y, \mathbf{\Sigma})$ and $\hat{\mathcal{T}}(z|Y=y) \approx \mathcal{N}(\mathbf{C}^y, \mathbf{\Sigma})$. Then we have $W_1(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y)) \leq \|\mathbf{C}_t^y - \mathbf{C}^y\|_2$. Intuitively, the approximation term is equivalent to the well known *feature mean matching* (Sugiyama and Kawanabe, 2012), which computes the feature centroid of each class (on the latent space Z) and aligns them by minimizing their L_2 distance.

Implicit Conditional Loss Apart from the approximation, we can derive a dual term for facilitating the computation, which is equivalent to the reweighted Wasserstein adversarial loss by the label distribution ratio.

Lemma 5.2. *The weighted conditional Wasserstein distance can be implicitly expressed as:*

$$\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\mathcal{S}_t(z|Y=y)\|\mathcal{T}(z|Y=y)) = \max_{d_1,\cdots,d_T} \sum_t \boldsymbol{\lambda}[t][\mathbb{E}_{z\sim\mathcal{S}_t(z)}\bar{\alpha}_t(z)d_t(z) - \mathbb{E}_{z\sim\mathcal{T}(z)}d_t(z)],$$

where $\bar{\alpha}_t(z) = \mathbf{1}_{\{(z,y)\sim\mathcal{S}_t\}}\alpha_t(Y=y)$*, and* $d_1,\ldots,d_T : \mathcal{Z} \to R_+$ *are the* 1*-Lipschitz domain discriminators (Ganin et al., 2016).*

The proof is delegated in Appendix Sec. D.6. Lemma 5.2 reveals that one can train T domain discriminators with weighted Wasserstein adversarial loss. When the source and target distributions are identical, this loss recovers the conventional Wasserstein adversarial loss (Arjovsky et al., 2017). In practice, we adopt a hybrid approach by linearly combining the explicit and implicit matching, in which empirical results show its effectiveness.

Estimation $\hat{\alpha}$ When the target labels are available, $\hat{\alpha}_t$ can be directly estimated from the data with $\hat{\alpha}_t(y) = \hat{\mathcal{T}}(y)/\hat{\mathcal{S}}(y)$ and $\hat{\alpha}_t \to \alpha_t$ can be proved from asymptotic statistics. As for the unsupervised scenarios, we will discuss in Sec. 5.5.1.

5.4.2 Estimation Relation Coefficient $\boldsymbol{\lambda}$

Inspired by Theorem 1, given a *fixed* $\hat{\alpha}_t$ and (g,h), we estimate $\boldsymbol{\lambda}$ through optimizing the derived upper bound.

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \sum_t \boldsymbol{\lambda}[t]\hat{R}_{\mathcal{S}_t}^{\hat{\alpha}_t}(h,g) + C_1\sqrt{\sum_{t=1}^T \frac{\boldsymbol{\lambda}^2[t]}{\beta_t}} + C_0\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{T}}(z|Y=y)\|\hat{\mathcal{S}}(z|Y=y)) \\ & \text{s.t} \quad \forall t, \boldsymbol{\lambda}[t] \geq 0, \sum_{t=1}^T \boldsymbol{\lambda}[t] = 1 \end{aligned}$$

In practice, $\hat{R}_{\mathcal{S}_t}^{\hat{\alpha}_t}(h,g)$ is the weighted empirical prediction loss and $\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{T}}(z|Y=y)\|\hat{\mathcal{S}}(z|Y=y))$ is approximated by the dynamic form of the critic function from Lemma 5.2. Then, solving $\boldsymbol{\lambda}$ can be viewed as a standard convex optimization problem with linear constraints.

5.5 Extension to Unsupervised Scenarios

In this section, we extend WADN to the unsupervised multi-source DA, which is known as unsolvable if the semantic conditional distribution ($\mathcal{S}_t(x|y) \neq \mathcal{T}(x|y)$) and label distribution ($\mathcal{S}_t(y) \neq \mathcal{T}(y)$) are simultaneously different and no specific conditions are considered (Ben-David et al., 2010b; Johansson et al., 2019).

In the WADN algorithm, this challenge is equivalent to properly estimate conditional Wasserstein distance and label distribution ratio $\hat{\alpha}_t(y)$ to help estimate $\boldsymbol{\lambda}$. According to Lemma 5.2, estimating the conditional Wasserstein distance can be viewed as an $\hat{\alpha}_t$-weighted adversarial loss, thus if we can correctly estimate label distribution ratio such that $\hat{\alpha}_t \to \alpha_t$, then we can properly compute the conditional Wasserstein-distance through the adversarial term.

Therefore, the problem turns to properly estimate the label distribution ratio. To this end, we assume the problem satisfies the Generalized Label Shift (GLS) condition (Combes et al., 2020), which has been theoretically justified and empirically evaluated in the single source unsupervised DA. The GLS condition states that *in unsupervised DA, there exists an optimal embedding function $g^\star \in \mathcal{G}$ that can ultimately achieve $\mathcal{S}_t(z|y) = \mathcal{T}(z|y)$ on the latent space.* Combes et al. (2020) further pointed out that the clustering assumption on $\mathcal{Z}$ is one sufficient condition to reach GLS, which is feasible for many practical applications.

Based on the achievability condition of GLS, the techniques of (Lipton et al., 2018; Garg et al., 2020) can be adopted to gradually estimate $\hat{\alpha}_t$ during learning the embedding function. Following this spirit, we add an distribution ratio estimator for $\{\hat{\alpha}_t\}_{t=1}^T$, shown in Sec. 5.5.1.

5.5.1 Estimation of $\hat{\alpha}_t$

Unsupervised DA We denote $\bar{\mathcal{S}}_t(y)$, $\bar{\mathcal{T}}(y)$ as the predicted t-source/target label distribution through the hypothesis h, and also define $C_{\hat{\mathcal{S}}_t}[y,k] = \hat{\mathcal{S}}_t[\text{argmax}_{y'} h(z,y') = y, Y = k]$ is the t-source *prediction confusion matrix*. According to the GLS condition, we have $\bar{\mathcal{T}}(y) = \bar{\mathcal{T}}_{\hat{\alpha}_t}(y)$, with $\bar{\mathcal{T}}_{\hat{\alpha}_t}(Y = y) = \sum_{k=1}^{\mathcal{Y}} C_{\hat{\mathcal{S}}_t}[y,k]\hat{\alpha}_t(k)$ the constructed target prediction distribution from the t-source information. (See Appendix for justification). Then we can estimate $\hat{\alpha}_t$ through matching these two distributions by minimizing $D_{\text{KL}}(\bar{\mathcal{T}}(y)\|\bar{\mathcal{T}}_{\hat{\alpha}_t}(y))$, which is equivalent to solving the following convex optimization:

$$\begin{aligned} &\min_{\hat{\alpha}_t} \quad -\sum_{y=1}^{|\mathcal{Y}|} \bar{\mathcal{T}}(y) \log(\sum_{k=1}^{|\mathcal{Y}|} C_{\hat{\mathcal{S}}_t}[y,k]\hat{\alpha}_t(k)) \\ &\text{s.t} \quad \forall y \in \mathcal{Y}, \hat{\alpha}_t(y) \geq 0, \quad \sum_{y=1}^{|\mathcal{Y}|} \hat{\alpha}_t(y)\hat{\mathcal{S}}_t(y) = 1 \end{aligned} \tag{5.1}$$

Unsupervised Partial DA If we have $\text{supp}(\mathcal{T}(y)) \subseteq \text{supp}(\mathcal{S}_t(y))$, α_t will be sparse due to the non-overlapped classes. Thus, we impose such prior knowledge by adding a regularizer $\|\hat{\alpha}_t\|_1$ to the objective of Eq. (5.1) to induce the sparsity in $\hat{\alpha}_t$.

Algorithm 4 WADN (unsupervised scenario, one epoch)

```
Ensure: Label ratio $\hat{\alpha}_t$ and task relation $\boldsymbol{\lambda}$. Feature Learner $g$, Classifier $h$, statistic critic function $d_1, \dots, d_T$, class centroid for source $\mathbf{C}_t^y$ and target $\mathbf{C}^y$.($t = 1, \dots, T$), hyper-parameter $\epsilon > 0$.
▷ DNN Parameter Training Stage (fixed $\alpha_t$ and $\boldsymbol{\lambda}$) ◁
for mini-batch of samples $(\mathbf{x}_{\mathcal{S}_1}, \mathbf{y}_{\mathcal{S}_1}) \sim \hat{\mathcal{S}}_1, \dots, (\mathbf{x}_{\mathcal{S}_T}, \mathbf{y}_{\mathcal{S}_T}) \sim \hat{\mathcal{S}}_T$, $(\mathbf{x}_{\mathcal{T}}) \sim \hat{\mathcal{T}}$ do
    Target predicted-label $\bar{\mathbf{y}}_{\mathcal{T}} = \text{argmax}_y h(g(\mathbf{x}_{\mathcal{T}}), y)$
    Compute unnormalized source confusion matrix on current *batch* $C_{\hat{\mathcal{S}}_t}[y, k]$.
    Compute feature centroid for source $C_t^y$ and target $C^y$ on current *batch*; Use moving average to update source and target class centroid $\mathbf{C}_t^y$ and $\mathbf{C}^y$.
    Updating $g, h, d_1, \dots, d_T$, by optimizing:
        $\min_{g,h} \max_{d_1,\dots,d_T} \underbrace{\sum_t \boldsymbol{\lambda}[t]\hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h, g)}_{\text{Classification Loss}} + \epsilon C_0 \underbrace{\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\bar{\mathcal{T}}(y)}\|\mathbf{C}_t^y - \mathbf{C}^y\|_2}_{\text{Explicit Conditional Loss}}$
        $+ (1-\epsilon)C_0 \underbrace{\sum_t \boldsymbol{\lambda}[t][\mathbb{E}_{z\sim\hat{\mathcal{S}}_t(z)}\bar{\alpha}_t(z)d(z) - \mathbb{E}_{z\sim\hat{\mathcal{T}}(z)}d(z)]}_{\text{Implicit Conditional Loss}}$
end for
▷ Estimation $\hat{\alpha}_t$ and $\boldsymbol{\lambda}$ ◁
Compute normalized source confusion matrix; Solve $\{\hat{\alpha}_t\}_{t=1}^T$ w.r.t. current training epoch through Sec.5.5.1 ; Update global $\hat{\alpha}_t$ through moving average.
Solve $\boldsymbol{\lambda}$ through Sec.5.4.2 w.r.t. current training epoch; Update global $\boldsymbol{\lambda}$ through moving average.
```

$$
\begin{aligned}
&\min_{\hat{\alpha}_t} \quad -\sum_{y=1}^{|\mathcal{Y}|} \bar{\mathcal{T}}(y) \log(\sum_{k=1}^{|\mathcal{Y}|} C_{\hat{\mathcal{S}}_t}[y, k]\hat{\alpha}_t(k)) + \|\hat{\alpha}_t\|_1 \\
&\text{s.t} \quad \forall y \in \mathcal{Y}, \hat{\alpha}_t(y) \geq 0, \quad \sum_{y=1}^{|\mathcal{Y}|} \hat{\alpha}_t(y)\hat{\mathcal{S}}_t(y) = 1
\end{aligned}
$$

In training the neural network, the non-overlapped classes will be automatically assigned with a small or zero $\hat{\alpha}_t$, then (g, h) will be less affected by the classes with small $\hat{\alpha}_t$.

5.5.2 Algorithm implementation and discussion

We give an algorithmic description of Fig. 5.2, shown in Algorithm 4. The high-level protocol is analogue to the Expectation-Maximization (EM) algorithm, which *iteratively* optimizes the neural-network parameters to gradually realize GLS condition with $g \to g^\star$ and dynamically update $\boldsymbol{\lambda}$, $\hat{\alpha}_t$ to better estimate conditional distance and aggregate the sources. The GLS assumes the achievability of existing an optimal $g^\star$. Similar to EM algorithm, our proposed algorithm can achieve a stationary solution, but due to the high non-convexity of the deep network, converging to the global optimal is not necessarily guaranteed.

Concretely, we update the $\hat{\alpha}_t$ and $\boldsymbol{\lambda}$ on the fly through a moving averaging strategy. Within one training epoch over the mini-batches, we fix the $\hat{\alpha}_t$ and $\boldsymbol{\lambda}$ and optimize the network parameters g, h. Then at each training epoch, we re-estimate the $\hat{\alpha}_t$ and $\boldsymbol{\lambda}$ by using the proposed estimator. When computing the explicit conditional loss, we empirically adopt the target pseudo label. For reducing the influence of possible initial poor predictions in the early training epochs, we assign a small weight for the explicit loss and gradually increase its weight during training. As for the optimization of $\boldsymbol{\lambda}$ and α_t, it is a standard convex optimization problem and we use package CVXPY.

As for WADN with limited target label, we do not require the label distribution ratio component and directly compute $\hat{\alpha}_t$.

5.6 Experiments

In this section, we compare the proposed approach with several baselines on the popular tasks. For all the scenarios, the following multi-source DA baselines are evaluated:

(I) **Source** method applied with only labeled source data to train the model.

(II) **DANN** (Ganin et al., 2016). We follow the protocol of Wen et al. (2020) to merge all the source dataset as a global source domain.

(III) **MDAN** (Zhao et al., 2018). A baseline that adopts the multiple domain discriminators to estimate task similarity. Then assigning weights for similar tasks.

(IV) **MDMN** (Li et al., 2018b). A baseline that adopts pairwise domain discriminators. Different from MADN, MDMN naturally computes the similarity between the pair of source domains then aggregate the source weights for the target.

(V) **M^{3}SDA** (Peng et al., 2019) adopted maximizing classifier discrepancy (Saito et al., 2018) in the multi-source scenario. Then the task similarity is estimated from the classifier discrepancy.

(VI) **DARN** (Wen et al., 2020). The recent baselines that adopt marginal distribution similarity and source prediction risk to select the relevant sources. This baseline is quite similar to ours, where the key differences lie in estimating the similarity through the lens of conditional distribution.

For the multi-source with limited target label and partial unsupervised multi-source DA, we additionally add specific baselines. All baselines are reimplemented in the same network structure for fair comparisons. The detailed network structures, hyper-parameter settings, and training details are delegated in Appendix Sec. D.11.

We evaluate the performance on three different datasets:

(1) **Amazon Review.** (Blitzer et al., 2007) It contains four domains (Books, DVD, Electronics, and Kitchen) with positive and negative product reviews. We follow the common data preprocessing

Table 5.1 – Unsupervised DA: Accuracy $(\%)$ and standard deviation $(\%)$ over five repeats (with the form Average$_{\pm\text{standard deviation}}$) on Source-Shifted Amazon Review.

Target	Books	DVD	Electronics	Kitchen	Average
Source	68.15$_{\pm 1.37}$	69.51$_{\pm 0.74}$	82.09$_{\pm 0.88}$	75.30$_{\pm 1.29}$	73.81
DANN	65.59$_{\pm 1.35}$	67.23$_{\pm 0.71}$	80.49$_{\pm 1.11}$	74.71$_{\pm 1.53}$	72.00
MDAN	68.77$_{\pm 2.31}$	67.81$_{\pm 2.46}$	80.96$_{\pm 0.77}$	75.67$_{\pm 1.96}$	73.30
MDMN	70.56$_{\pm 1.05}$	69.64$_{\pm 0.73}$	82.71$_{\pm 0.71}$	77.05$_{\pm 0.78}$	74.99
M^{3}SDA	69.09$_{\pm 1.26}$	68.67$_{\pm 1.37}$	81.34$_{\pm 0.66}$	76.10$_{\pm 1.47}$	73.79
DARN	71.21$_{\pm 1.16}$	68.68$_{\pm 1.12}$	81.51$_{\pm 0.81}$	77.71$_{\pm 1.09}$	74.78
WADN	**73.72**$_{\pm 0.63}$	**79.64**$_{\pm 0.34}$	**84.64**$_{\pm 0.48}$	**83.73**$_{\pm 0.50}$	**80.43**

Table 5.2 – Unsupervised DA: Accuracy $(\%)$ and standard deviation $(\%)$ over five repeats (with the form Average$_{\pm\text{standard deviation}}$) on the Source-Shifted Digits.

Target	MNIST	SVHN	SYNTH	USPS	Average
Source	84.93$_{\pm 1.50}$	67.14$_{\pm 1.40}$	78.11$_{\pm 1.31}$	86.02$_{\pm 1.12}$	79.05
DANN	86.99$_{\pm 1.53}$	69.56$_{\pm 2.26}$	78.73$_{\pm 1.30}$	86.81$_{\pm 1.74}$	80.52
MDAN	87.86$_{\pm 2.24}$	69.13$_{\pm 1.56}$	79.77$_{\pm 1.69}$	86.50$_{\pm 1.59}$	80.81
MDMN	87.31$_{\pm 1.88}$	69.84$_{\pm 1.59}$	80.27$_{\pm 0.88}$	86.61$_{\pm 1.41}$	81.00
M^{3}SDA	87.22$_{\pm 1.70}$	68.89$_{\pm 1.93}$	80.01$_{\pm 1.77}$	86.39$_{\pm 1.68}$	80.87
DARN	86.98$_{\pm 1.29}$	68.59$_{\pm 1.79}$	80.68$_{\pm 0.61}$	86.85$_{\pm 1.78}$	80.78
WADN	**89.07**$_{\pm 0.72}$	**71.66**$_{\pm 0.77}$	**82.06**$_{\pm 0.89}$	**90.07**$_{\pm 1.10}$	**83.22**

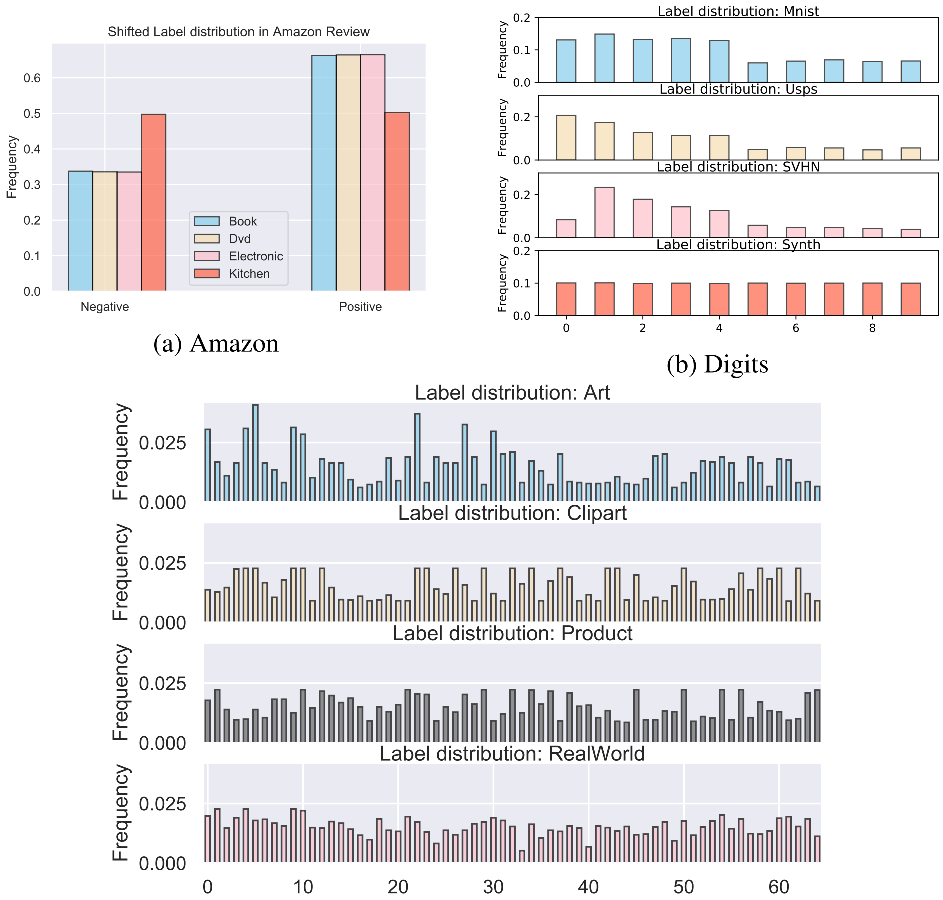
strategies as (Chen et al., 2012) to form a 5000-dimensional bag-of-words feature. Note that the label distribution in the original dataset is uniform. *To show the benefits of the proposed approach, we create a label distribution drifted task by randomly dropping* 50% *of the negative reviews of all the sources while keeping the target unchanged.*

(2) **Digits**. It consists of four digits recognition datasets including MNIST, USPS (Hull, 1994), SVHN (Netzer et al., 2011) and Synth (Ganin et al., 2016). *We also create a label distribution drift for the sources by randomly dropping* 50% *samples on digits 5-9 and keep target domain unchanged.*

(3) **Office-Home Dataset** (Venkateswara et al., 2017). It contains 65 classes for four different domains: Art, Clipart, Product and Real-World. We used the ResNet50 (He et al., 2016) pretrained from ImageNet in PyTorch as the base network for feature learning and put a MLP for the classification. The label distributions in these four domains are different and we did not manually create a label drift.

5.6.1 Unsupervised Multi-Source DA

In the unsupervised multisource DA, we evaluate the proposed approach on all three datasets. We use a similar hyper-parameter selection strategy as in DANN (Ganin et al., 2016). All reported results are averaged over five runs. The detailed experimental settings are illustrated in Appendix. The empirical results are illustrated in Tab. 5.1, 5.2 and 5.3. Since we did not change the target label distribution throughout the whole experiment, we still report the target accuracy as the metric. We report the



(a) Amazon

(b) Digits

(c) Office-Home

Figure 5.3 – Label distribution visualization. (a) One example in Amazon Review dataset with sources: Book, Dvd, Electronic and target: Kitchen. We randomly drop 50% of the negative reviews in all the sources while keeping target label distribution unchanged. (b) One example in Digits dataset with Sources: MNIST, USPS, SVHN and Target Synth. We randomly drop 50% data on digits 5-9 in all sources while keeping target label distribution unchanged. (c) Office-Home dataset. The original label distribution is non-uniform.

means and standard deviations for each approach. The best approaches based on a two-sided Wilcoxon signed-rank test (significance level $p = 0.05$) are shown in bold.

The empirical results reveal a significantly better performance ($\approx 2\% - 6\%$) on different benchmarks. For understanding the aggregation principles of WADN, we visualize the task relations in digits (Fig. 5.4(a)) with demonstrating a *non-uniform* $\boldsymbol{\lambda}$, which highlights the importance of properly choosing the most related source rather than simply merging all the data. For example, when the target domain is SVHN, WADN mainly leverages the information from SYNTH, since they are more semantically similar, and MNIST does not help too much for SVHN, which is also observed by Ganin et al. (2016). Besides, Fig. 5.4(b) visualizes the evolution of $\boldsymbol{\lambda}$ between WADN and recent principled approach DARN (Wen et al., 2020), which utilized the $\mathbb{P}(x)$ information and dynamic updating to find the similar domains. Compared with WADN, $\boldsymbol{\lambda}$ in DARN is *unstable* during updating under drifted label

Table 5.3 – Unsupervised DA: Accuracy $(\%)$ and standard deviation $(\%)$ over five repeats (with the form Average$_{\pm\text{standard deviation}}$) on Office-Home

Target	Art	Clipart	Product	Real-World	Average
Source	$49.25_{\pm 0.60}$	$46.89_{\pm 0.61}$	$66.54_{\pm 1.72}$	$73.64_{\pm 0.91}$	59.08
DANN	$50.32_{\pm 0.32}$	$50.11_{\pm 1.16}$	$68.18_{\pm 1.27}$	$73.71_{\pm 1.63}$	60.58
MDAN	$67.93_{\pm 0.36}$	$66.61_{\pm 1.32}$	$79.24_{\pm 1.52}$	$81.82_{\pm 0.65}$	73.90
MDMN	$68.38_{\pm 0.58}$	$67.42_{\pm 0.53}$	$82.49_{\pm 0.56}$	$83.32_{\pm 1.93}$	75.28
M^{3}SDA	$63.77_{\pm 1.07}$	$62.30_{\pm 0.44}$	$75.85_{\pm 1.24}$	$79.92_{\pm 0.60}$	70.46
DARN	$69.89_{\pm 0.42}$	$68.61_{\pm 0.50}$	$83.37_{\pm 0.62}$	$84.29_{\pm 0.46}$	76.54
WADN	$\mathbf{73.78}_{\pm 0.43}$	$\mathbf{70.18}_{\pm 0.54}$	$\mathbf{86.32}_{\pm 0.38}$	$\mathbf{87.28}_{\pm 0.87}$	**79.39**

Table 5.4 – Multi-source Transfer: Accuracy $(\%)$ and standard deviation $(\%)$ over five repeats (with the form Average$_{\pm\text{standard deviation}}$) on Source-Shifted Amazon Review

Target	Books	DVD	Electronics	Kitchen	Average
Source + Tar	$72.59_{\pm 1.89}$	$73.02_{\pm 1.84}$	$81.59_{\pm 1.58}$	$77.03_{\pm 1.73}$	76.06
DANN	$67.35_{\pm 2.28}$	$66.33_{\pm 2.42}$	$78.03_{\pm 1.72}$	$74.31_{\pm 1.71}$	71.50
MDAN	$68.70_{\pm 2.99}$	$69.30_{\pm 2.21}$	$78.78_{\pm 2.21}$	$74.07_{\pm 1.89}$	72.71
MDMN	$69.19_{\pm 2.09}$	$68.71_{\pm 2.39}$	$81.88_{\pm 1.46}$	$78.51_{\pm 1.91}$	74.57
M^{3}SDA	$69.28_{\pm 1.78}$	$67.40_{\pm 0.46}$	$76.28_{\pm 0.81}$	$76.50_{\pm 1.19}$	72.36
DARN	$68.57_{\pm 1.35}$	$68.77_{\pm 1.81}$	$80.19_{\pm 1.66}$	$77.51_{\pm 1.20}$	73.76
MME	$69.66_{\pm 0.58}$	$71.36_{\pm 0.96}$	$78.88_{\pm 1.51}$	$76.64_{\pm 1.73}$	74.14
RLUS	$71.83_{\pm 1.71}$	$69.64_{\pm 2.39}$	$81.98_{\pm 1.04}$	$78.69_{\pm 1.15}$	75.54
WADN	$\mathbf{74.83}_{\pm 0.84}$	$\mathbf{75.05}_{\pm 0.62}$	$\mathbf{84.23}_{\pm 0.58}$	$\mathbf{81.53}_{\pm 0.90}$	**78.91**

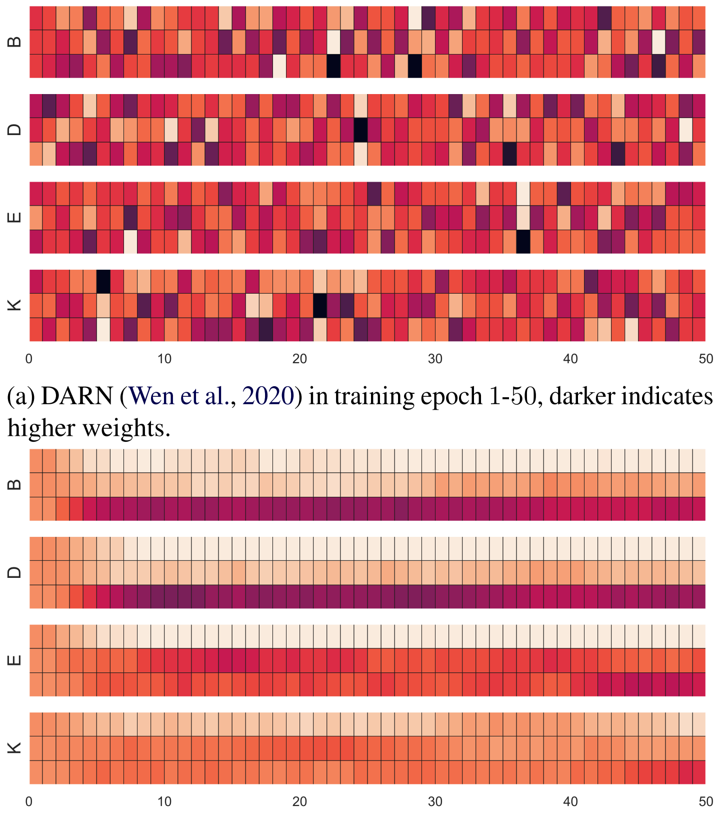
distribution.

Besides, we conduct the ablation study through evaluating the performance under different levels of source label shift in Amazon Review dataset (Fig. 5.5(a)). The results show strong practical benefits for WADN in the larger label shift. The additional analysis and results can be found in Appendix.

5.6.2 Multi-Source DA with Limited Target Labels

We adopt Amazon Review and digits in the multi-source DA with limited target samples, which have been widely used. In the experiments, we still use shifted sources. We randomly sample only 10% labeled samples (w.r.t. target dataset in unsupervised DA) as training set and the rest 90% samples as the unseen target test set. We adopt the same hyper-parameters and training strategies with unsupervised DA. We specifically add two recent baselines RLUS (Konstantinov and Lampert, 2019) and MME (Saito et al., 2019), which also considered DA on the labeled target domain.

The results are reported in Tab. 5.4, which also indicates strong empirical improvement. Interestingly, on the Amazon review dataset, the previous aggregation approach RLUS is unable to select the related source when the label distribution varies. To show the effectiveness of WADN, we tested various portions of labelled samples ($1\% \sim 10\%$) on the target. The results in Fig. 5.5(b) on USPS dataset



(a) DARN (Wen et al., 2020) in training epoch 1-50, darker indicates higher weights.

(b) WADN in training epoch 1-50, darker indicates higher weights.

Figure 5.4 – (a,b) The evolution of $\boldsymbol{\lambda} \in \mathbb{R}^3$ with three sources of Amazon dataset (B=Books, D=DVD, E=Electronics, K=Kitchen) during the training epoch. For instance, B indicates the target is Books and the domain weights for other three sources (D,E and K). We compare with a recent principle approach DARN, which uses $\mathbb{P}(x)$ to measure the similarity and dynamically update the $\boldsymbol{\lambda}$. The results verifies the limitation of DARN under changing label distributions with relative unstable results.

show consistently better than the baseline, even in the few target samples scenarios such as $1 - 3\%$.

5.6.3 Partial Unsupervised Multi-Source DA

In this scenario, we adopt the Office-Home dataset to evaluate our approach, as it contains a large number of classes (65). We do not change the source domains and we randomly choose 35 classes from the target. We evaluate all the baselines on the same selected classes and repeat 5 times. All reported results are averaged from 3 different sub-class selections (15 runs in total), shown in Tab. 5.6. We additionally compare PADA (Cao et al., 2018) approach by merging all sources and use a one-to-one partial DA algorithm. We adopt the same hyper-parameters and training strategies in unsupervised DA scenario.

The reported results are also significantly better than the current multi-source DA or one-to-one partial DA approach, which again emphasizes the benefits of WADN: properly selecting the related sources by using semantic information.

Table 5.5 – Multi-source Transfer: Accuracy $(\%)$ and standard deviation $(\%)$ over five repeats (with the form Average$_{\pm\text{standard deviation}}$) on the Source-Shifted Digits

Target	MNIST	SVHN	SYNTH	USPS	Average
Source + Tar	$79.63_{\pm 1.74}$	$56.48_{\pm 1.90}$	$69.64_{\pm 1.38}$	$86.29_{\pm 1.56}$	73.01
DANN	$86.77_{\pm 1.30}$	$69.13_{\pm 1.09}$	$78.82_{\pm 1.35}$	$86.54_{\pm 1.03}$	80.32
MDAN	$86.93_{\pm 1.05}$	$68.25_{\pm 1.53}$	$79.80_{\pm 1.17}$	$86.23_{\pm 1.41}$	80.30
MDMN	$77.59_{\pm 1.36}$	$69.62_{\pm 1.26}$	$78.93_{\pm 1.64}$	$87.26_{\pm 1.13}$	78.35
M^{3}SDA	$85.88_{\pm 2.06}$	$68.84_{\pm 1.05}$	$76.29_{\pm 0.95}$	$87.15_{\pm 1.10}$	79.54
DARN	$86.58_{\pm 1.46}$	$68.86_{\pm 1.30}$	$80.47_{\pm 0.67}$	$86.80_{\pm 0.89}$	80.68
MME	$87.24_{\pm 0.95}$	$65.20_{\pm 1.35}$	$80.31_{\pm 0.60}$	$87.88_{\pm 0.76}$	80.16
RLUS	$87.61_{\pm 1.08}$	$\mathbf{70.50}_{\pm 0.94}$	$79.52_{\pm 1.30}$	$86.70_{\pm 1.13}$	81.08
WADN	$\mathbf{88.32}_{\pm 1.17}$	$\mathbf{70.64}_{\pm 1.02}$	$\mathbf{81.53}_{\pm 1.11}$	$\mathbf{90.53}_{\pm 0.71}$	**82.75**

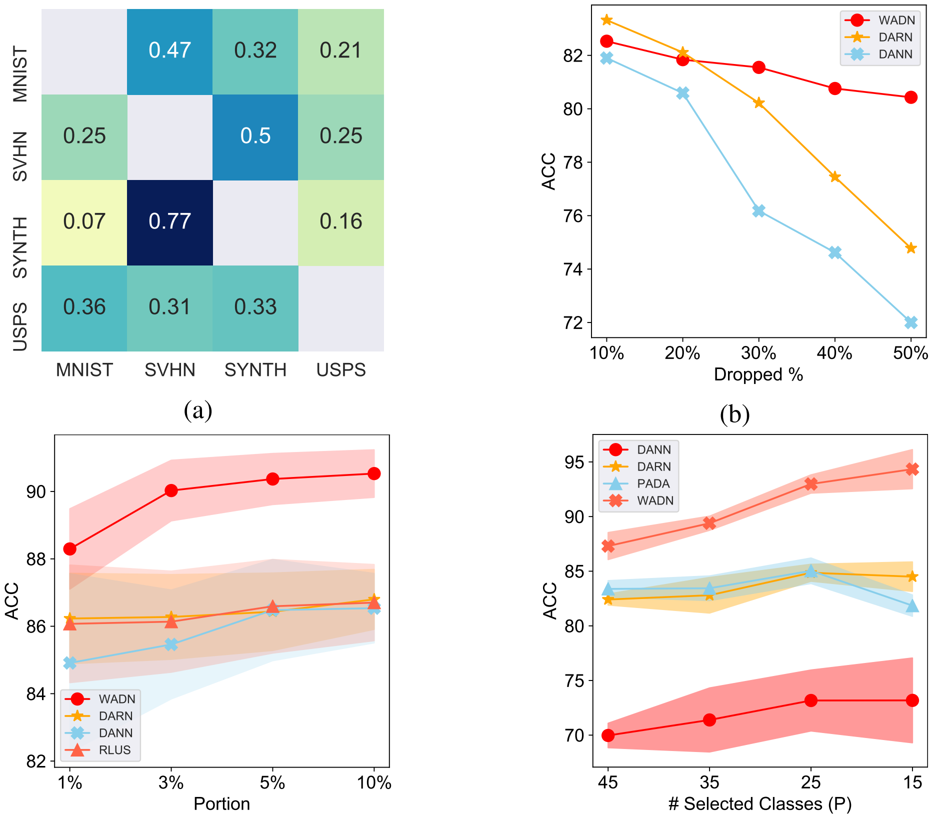
Table 5.6 – Unsupervised Partial DA: Accuracy $(\%)$ and standard deviation $(\%)$ over five repeats (with the form Average$_{\pm\text{standard deviation}}$) on Office-Home (#Source: 65, #Target: 35)

Target	Art	Clipart	Product	Real-World	Average
Source	$50.56_{\pm 1.42}$	$49.79_{\pm 1.14}$	$68.10_{\pm 1.33}$	$78.24_{\pm 0.76}$	61.67
DANN	$53.86_{\pm 2.23}$	$52.71_{\pm 2.20}$	$71.25_{\pm 2.44}$	$76.92_{\pm 1.21}$	63.69
MDAN	$67.56_{\pm 1.39}$	$65.38_{\pm 1.30}$	$81.49_{\pm 1.92}$	$83.44_{\pm 1.01}$	74.47
MDMN	$68.13_{\pm 1.08}$	$65.27_{\pm 1.93}$	$81.33_{\pm 1.29}$	$84.00_{\pm 0.64}$	74.68
M^{3}SDA	$65.10_{\pm 1.97}$	$61.80_{\pm 1.99}$	$76.19_{\pm 2.44}$	$79.14_{\pm 1.51}$	70.56
DARN	$71.53_{\pm 0.63}$	$69.31_{\pm 1.08}$	$82.87_{\pm 1.56}$	$84.76_{\pm 0.57}$	77.12
PADA	$74.37_{\pm 0.84}$	$69.64_{\pm 0.80}$	$83.45_{\pm 1.13}$	$85.64_{\pm 0.39}$	78.28
WADN	$\mathbf{80.06}_{\pm 0.93}$	$\mathbf{75.90}_{\pm 1.06}$	$\mathbf{89.55}_{\pm 0.72}$	$\mathbf{90.40}_{\pm 0.39}$	**83.98**

Besides, if we change the number of selected classes (Fig 5.5(c)), the proposed WADN still indicates consistent better results by a large margin, which indicates the importance of considering $\hat{\alpha}_t$ and $\boldsymbol{\lambda}$. In contrast, DANN shows unstable results on average in selected classes. Beside, WADN shows a good estimation of the label distribution ratio (Fig 5.6) and has correctly detected the non-overlapping classes, which verifies the effectiveness of the label distribution estimator and indicates its good explainability.

5.7 Conclusion

In this paper, we proposed a novel algorithm WADN for multi-source domain adaptation problem under different label proportions. WADN differs from previous approaches in two key prospects: a better source aggregation approach when label distribution changes; a unified empirical framework for three popular DA scenarios. We evaluated the proposed method by extensive experiments and showed its strong empirical results.



(a)

(b)

(c)

(d)

Figure 5.5 – Ablation study on different scenarios. (a) Visualization of $\boldsymbol{\lambda}$ on digits datset, each row corresponds to a target domain, which indicates a *non-uniform* and *non-symmetric* task relations. (b) Unsupervised DA with Amazon Review dataset. Accuracy under different levels of label shifted sources (higher dropping rate means larger label drift). The results are reported on the average of all the domains, see the results for each domain in Appendix. (c) Multi-Source DA with limited target label in digits task with target USPS. The performance (mean ± std) of WADN is consistently better under different target samples (smaller portion indicates fewer target samples). (d) Partial Multi-source DA in office-home dataset with target domain Product. Performance (mean ± std) of different number of selected classes on the target, where WADN shows a consistent better performance under different selected sub-classes.

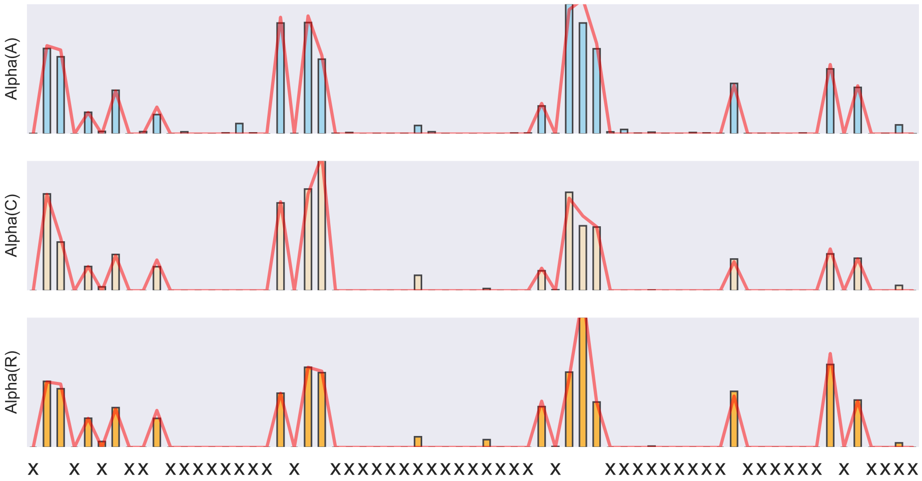


Figure 5.6 – Analysis on Partial DA of target Product. We select 15 classes and visualize estimated $\hat{\alpha}_t$ (the bar plot). The "X" along the x-axis represents the index of *dropped* 50 classes. The red curves are the true label distribution ratio.

Conclusion

In this thesis, we proposed novel and principled approaches through distribution matching for learning from limited labeled data. We particularly focus on three scenarios: deep multitask learning, deep active learning, and domain adaptation. Although we focused on these three scenarios throughout this thesis, the proposed theoretical and practical insights can be nevertheless extended to broader settings, e.g. learning fair and transferable features in algorithmic fairness (Zemel et al., 2013; Louizos et al., 2015; Edwards and Storkey, 2015), and individual treatment effect estimation in causal inference (Shalit et al., 2017). Below we will present the highlights and potential impacts of our proposed theories and practice.

Understanding Similarity-Based Multi-Task Learning

In Chapter 2, we proposed a theoretical analysis of the benefits of considering task similarities in MTL and further derived an algorithm that used task similarity information. Specifically, we have the following interesting insights and impacts from this research:

- **Characterizing and formulating task relations.** Our algorithm proposed a principled approach to characterize the task relations. We adopted simplex vectors α_t to explicitly express the task relations. *Different from previous works (Zhang and Yeung, 2010; Ciliberto et al., 2015), the extracted task relations in our algorithm are* ***non-symmetric*** *and robust w.r.t. different similarity metrics, which better reflects the real-world scenarios.* For instance, in healthcare applications, our proposed algorithm has been applied for estimating the similarity between different Parkinson patients (Zhou et al., 2020a). Geden et al. (2020) used similar ideas in predictive student modeling in educational games.
  Besides, the proposed idea has inspired new approaches in the related fields, for example, Cai et al. (2020) followed the same research line to estimate task similarity in meta-learning; Wu et al. (2020a) extended the similarity measure by using nonlinear task relations and Zhang et al. (2020b) investigated the negative transfer problem through the proposed task relations.

Unified Method for Query and Training in Deep Active Learning

In Chapter 3, we proposed novel query and training algorithms in deep active learning, which is simple and scalable for the large-scale dataset. Specifically, we have the following interesting insights and potential impacts from this research:

- **Novel explicit query strategy.** Compared with recent works such as (Ash et al., 2019; Sener and Savarese, 2018), we proposed a principled query strategy with *explicitly* considering uncertainty and diversity trade-off through optimizing the derived upper bound. This idea has motivated several recent active learning papers, e.g, Cui and Sato (2020) replaced the Wasserstein-1 distance by a general discrepancy distance for obtaining a tighter bound in AL. Besides, a recent survey paper of deep active learning such as (Ren et al., 2020; Liu et al., 2021) highlighted the benefits of our proposed explicit query.
- **Leveraging the unlabeled information in active learning.** Compared with previous work (Gissin and Shalev-Shwartz, 2019), our approach investigated the benefits of leveraging unlabeled information in AL. Moreover, this idea has been recently discussed and extended in other related domains, e.g, Zhou et al. (2020b) followed a similar research line in active learning for domain adaptation, Mundt et al. (2020) associated this idea with continual learning and Popordanoska et al. (2020) discussed this idea in human-machine interactive learning.

Domain Adaptation Theory with Jensen-Shannon Divergence

In Chapter 4, we proposed a novel Domain Adaptation (DA) theory with Jensen-Shannon divergence and illustrated its practical and theoretical benefits. Specifically, we have the following interesting insights from this research:

- **Jensen-Shannon divergence in theory.** We proposed a novel DA theory through information-theoretic Jensen-Shannon divergence. We notice that Jose and Simeone (2020); Azizzadenesheli (2020) extended this idea to transfer learning with weighted ERM rule. Asatryan et al. (2020) analogously adopted Jensen-Shannon divergence for analyzing the distribution shift.
- **Label shift correction and semantic conditional distribution matching.** Our paper also reveals the importance of considering different label distributions $\mathcal{Y}$, which remains elusive from $\mathcal{H}$-divergence. As we mentioned in Chapter 5, this idea can be applied in *distribution shift* problems, e.g., Zhou et al. (2021) extended this principle to the multitask learning problem, which showed significantly better performance when label shift occurs among different tasks. We do think the proposed principles are towards solving practical motivated distribution-shifted problems such as algorithmic fairness, privacy, and transparent machine learning.

Unified Methods for Multi-Source Domain Adaptation under Target-Shift

In Chapter 5, we extend the direction in Chapter 4 by proposing unified approaches in multisource domain adaptation when label shifts occur. Specifically, we have the following interesting insights from this research:

- **Novel algorithm in label shift in multi-source domain adaptation.** In this work, we first proposed a novel theoretical analysis on the label shifts without assuming the identical semantic conditional distribution, i.e., $\mathcal{S}(x|Y=y) \neq \mathcal{T}(x|Y=y)$. We adopted Wasserstein-1 distance, which is more computationally efficient compared to existing related theories such as (Mansour et al., 2020; Wang et al., 2019a), where they adopted $\mathcal{Y}$-discrepancy (Mohri and Medina, 2012) in the theoretical analysis, which is difficult to be estimated for a multiclass classification problem. Besides, we first analyze the principles in representation learning and propose a new guideline in learning the embedding function.

- **Unified practical framework.** Different from the previous practical framework, we focused on the unified approaches for three common transfer scenarios: domain adaptation with limited target label (Pan and Yang, 2009), unsupervised domain adaptation (Ganin et al., 2016), and label partial unsupervised domain adaptation (Cao et al., 2018). In our paper, these scenarios can be treated under the same protocol, we also believe it is an important and interesting direction since it reveals their inherent relations and different assumptions among these scenarios.

Limitations

We have derived several methods for learning limited labeled data through distribution matching. This thesis simultaneously remains several limitations.

Only for classification The thesis mainly focused on the classification setting with accuracy as the only metric. In contrast, a number of real world practice is involved with other metrics such as AUC curve. It is still opening and promising to extend the distribution matching for the regression and multi-label settings.

Understanding the influence of pseudo-label The proposed approaches in unsupervised domain adaptation generally involve with pseudo-labels. i.e, we treat the algorithm's output as the proxy of true label in the conditional distribution matching. Clearly, the conditional distribution matching depends on the pseudo-label. And the conditional distribution will reversely influence the algorithm, which induces a cycle scenario. In this thesis, we proposed several strategies to address the potential poor pseudo-labels, whereas the formal theoretical support still remains unclear.

Future Directions

In this thesis, we focused on the problem of learning the limited labeled dataset through distribution matching approaches. Specifically, we analyzed three common scenarios and showed novel practical and theoretical implications. Simultaneously, there exist various interesting aspects to be explored in the future.

Extending the concept of task similarity. We mainly focused on the task with the same label space (e.g., digit recognition problem in different contexts). It will be highly interesting to consider the theory and practice of two tasks with completely different label spaces. For instance, how to measure the distance between digits and alphabet datasets? In contrast, the proposed theoretical results in this thesis (such as $\mathcal{H}$-divergence) can measure the distance only on the observations (i.e $\mathcal{X}$) without considering the causal information $\mathbb{P}(y|x)$, which may be restrictive for many practical scenarios. Besides, the proposed semantic information $\mathbb{P}(x|y)$ (anti-causal direction) can not handle this new scenario because of the non-overlapped label space. (e.g., in measuring similarity between digits and alphabet dataset, we cannot measure $d(\mathbb{P}_1(x|y)\|\mathbb{P}_2(x|y))$ because $\mathcal{Y}$ are completely different.) Therefore it is essential to develop novel theoretical and practical results in this prospect, which can be potentially adapted for various *distribution shift* problems such as transfer learning, fairness, meta-learning, and continual learning.

Other scenarios of label shift problems. We have explored the label shift problem in transfer learning. A natural question is whether these techniques can be extended to other scenarios. For instance, in active learning, if the label distributions of the initial dataset are highly different from the unlabeled pools, the query strategy may be biased rather than properly capturing the whole data information. Hence, a label shift correction may be necessary. Another example is in continual learning, the imbalanced data will consistently change the decision boundary, which will surely influence the performance. As label shift commonly occurs in many practical scenarios, extending our theoretical and practical results will be beneficial for solving these real-world problems.

Understanding algorithmic fairness through the lens of distribution matching It has been observed that conventional machine learning algorithms exhibit the prediction disparities among different subgroups, due to the distribution shift. To this end, it will be quite interesting to adopt distribution matching for a better understanding the algorithmic fairness.

From a general viewpoint, this thesis focused on the principled approach in learning with limited labeled data. We studied various scenarios with theoretical and practical support, which has motivated a series of recent works in theory and practice. For a broader impact, we think such analysis will be beneficial in designing intelligent systems in the real world with smaller label annotations.

Appendix A

Details of Chapter 2

A.1 Theoretical tools

In this section, we will list the theoretical tools, which will be applied multiple times in the later proof.

A.1.1 Transfer bounds

In this section, we will analyze the relations of the *expected risk* on the distributions.

Lemma A.1. Ben-David et al. (2010a) *Let $\mathcal{H}$ be the hypothesis space with VC dimension d. For two tasks with marginal w.r.t x generation distribution $\mathcal{D}_i$ and $\mathcal{D}_j$, and every $h \in \mathcal{H}$, we have:*

$$R_j(h) \leq R_i(h) + d_{\mathcal{H}}(D_i, D_j) + \lambda_{i,j}, \tag{A.1}$$

where $\lambda_{i,j} = \inf_{h\in\mathcal{H}}\{R_i(h) + R_j(h)\}$.

A.1.2 Concentration bounds between empirical and expected divergence

Lemma A.2. *Let $\mathcal{H}$ be the hypothesis space on $\mathcal{X}$ with VC dimension d. S_i and S_j are the i.i.d samples with size m_i and m_j, respectively. We also define the empirical divergence $\hat{d}_{\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j)$ w.r.t. S_1 and S_2, then for any $\delta \in (0,1)$, with probability at least $1-\delta$, we have:*

$$d_{\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j) \leq \hat{d}_{\mathcal{H}}(\mathcal{D}_i, \mathcal{D}_j) + 2\sqrt{\frac{2d\log(2m_{ij}) + \log(\frac{2}{\delta})}{m_{ij}}}, \tag{A.2}$$

where $m_{ij} = \min\{m_i, m_j\}$.

The bound is slightly different from Ben-David et al. (2010a), because the original paper supposed the equal number of observations between two distributions. However, the proof is also a simple plugging in the conclusion of Kifer et al. (2004) .

Proof. From the Theorem 3.4 of Kifer et al. (2004), we have:

$$P[|d_{\mathcal{H}}(\mathcal{D}_i,\mathcal{D}_j) - \hat{d}_{\mathcal{H}}(\mathcal{D}_i,\mathcal{D}_j)| \geq \epsilon] \leq (2m_i)^d e^{-m_i\epsilon^2/16} + (2m_j)^d e^{-m_j\epsilon^2/16} \tag{A.3}$$

We consider the function $f(x) = (2x)^d \exp(-x\epsilon^2/16)$, then we compute the gradient w.r.t x, we have $f'(x) = (2x)^{d-1}\exp(-x\epsilon^2/16)(2d - 2x\epsilon^2/16) < 0$. When $\epsilon^2 \geq 16d/m_{i,j}$, we have $f(m_j), f(m_j)$ can both be upper bounded by $f(m_{i,j})$.

Hence we can verify the R.H.S. in equation (A.3) can be upper bounded by $2(2m_{i,j})^d e^{-m_{i,j}\epsilon^2/16}$. Then we set this value as δ, we have $\epsilon^2 \geq 16\frac{\log(2/\delta)+d\log(2m_{i,j})}{m_{i,j}}$.

Under this condition, we have the conclusion shown in the Lemma. Moreover, the divergence is defined for the $\mathcal{H}$ hypothesis set, then the VC dimension is $2d$ for such a hypothesis set. □

A.1.3 Concentration bounds between empirical and expected risk

Another useful inequality is to bound the difference between the empirical and the expected error in the weighted loss.

Lemma A.3. *For each task index $j = \{1,\dots,T\}$, let S_j be a labeled sample of size $\beta_j n$ generated from distribution $\mathcal{D}_j$ and labeled according to the function f_j. For any fixed $\boldsymbol{\alpha}$, and all binary classifiers $h \in \mathcal{H} : \mathcal{X} \to \{-1,1\}$ with VC dimension d, with a probability greater than $1-\delta$, we have:*

$$R_{\boldsymbol{\alpha}_t}(h) \leq \hat{R}_{\boldsymbol{\alpha}_t}(h) + 2\sqrt{\sum_{j=1}^{T}\frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\sqrt{\frac{2(d\log(\frac{2en}{d}) + \log(\frac{8}{\delta}))}{n}} \tag{A.4}$$

Proof. The proof is analogue to the proof of the uniform convergence bound. We first apply the symmetrization trick by generating ghost samples. For the notation simplification, we define $\hat{R}'_{\boldsymbol{\alpha}_t}$ is the empirical risk induced by $Z'_1, Z'_2, \dots$ by sampling from the same distribution (but we never know it, so we called *ghost sample*).

From the symmetrization lemma, we have, for $\epsilon \geq \sqrt{2/n}$:

$$\mathbb{P}\big(\sup_{h\in\mathcal{H}}|R_{\boldsymbol{\alpha}_t}(h) - \hat{R}_{\boldsymbol{\alpha}_t}(h)| \geq \epsilon\big) \leq 2\mathbb{P}\big(\sup_{h\in\mathcal{H}}|\hat{R}_{\boldsymbol{\alpha}_t}(h) - \hat{R}'_{\boldsymbol{\alpha}_t}(h)| \geq \frac{\epsilon}{2}\big) \tag{A.5}$$

Then we prove the modified VC-bound, defining $V = \mathcal{H}_{Z_1,\dots,Z_n,Z'_1,\dots,Z_n}$. For any $v \in V$, we can write $\hat{R}_{\boldsymbol{\alpha}_t}(h) - \hat{R}'_{\boldsymbol{\alpha}_t}(h) = \frac{1}{n}\sum_{j=1}^n v_j - \sum_{j=n+1}^{2n} v_j$

$$\mathbb{P}\big(\sup_{h\in\mathcal{H}}|\hat{R}_{\boldsymbol{\alpha}_t}(h) - \hat{R}'_{\boldsymbol{\alpha}_t}(h)| \geq \frac{\epsilon}{2}\big) \leq 2\mathbb{P}\big(\max_{v\in V}|\frac{1}{n}\sum_{j=1}^{n} v_j - \sum_{j=n+1}^{2n} v_j| \geq \frac{\epsilon}{2}\big) \tag{A.6}$$

Then we have:

$$\leq 2\Pi(2n)\mathbb{P}\big(|\frac{1}{n}\sum_{j=1}^{n} v_j - \sum_{j=n+1}^{2n} v_j| \geq \frac{\epsilon}{2}\big) \qquad \text{(Union Bound)}$$

By introducing the Rademacher variable σ_j, we have:

$$\leq 4\Pi(2n)\mathbb{P}\big(|\frac{1}{n}\sum_{j=1}^{n}\sigma_j v_j| \geq \frac{\epsilon}{4}\big) \qquad \text{(Introducing the Rademacher variable)}$$

According the Hoeffding's inequality, we have:

$$\leq 8\Pi(2n)\exp(-\frac{n\epsilon^2}{8\sum_{j=1}^{T}(\frac{\boldsymbol{\alpha}_{t,j}}{\beta_j})^2}) \qquad \text{(Hoeffding's inequality)}$$

Then we have at probability at least $1-\delta$:

$$R_{\boldsymbol{\alpha}_t}(h) \leq \hat{R}_{\boldsymbol{\alpha}_t}(h) + 2\sqrt{\sum_{j=1}^{T}\frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\sqrt{\frac{2(d\log(\frac{2en}{d}) + \log(\frac{8}{\delta}))}{n}} \qquad \text{(Sauer's lemma)}$$

This bound can also be proved with Mcdiarmid inequality and Rademacher complexity with sight loose results. □

A.2 Proof of Theorem 2.1

A.2.1 Three-steps proof

In this section, we make connections between the similarity measuring and the expected risk. For a pair of distributions $(\mathcal{D}_i, \mathcal{D}_j)$, we define $h^{\star}_{i,j} \in \text{argmin}_{h\in\mathcal{H}}\{R_i(h) + R_j(h)\}$ the *joint expected minimal error* for the hypothesis class $\mathcal{H}$.

Step 1: For one task t we have:

$$|R_{\boldsymbol{\alpha}_t}(h) - R_t(h)| = |\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}R_i(h) - R_t(h)| \leq \sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}|R_i(h) - R_t(h)| \tag{A.7}$$

According to the triangle inequality of the loss function, we have

$$\leq \sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\Big(|R_i(h) - R_i(h, h^{\star}_{i,t})| + |R_i(h, h^{\star}_{i,t}) - R_t(h, h^{\star}_{i,t})| + |R_t(h) - R_t(h, h^{\star}_{i,t})|\Big) \tag{A.8}$$

According to the triangle inequality and definition of the distribution discrepancy, we have:

$$|R_i(h) - R_i(h, h^{\star}_{i,t})| \leq R_i(h^{\star}_{i,t})$$

$$|R_i(h, h^{\star}_{i,t}) - R_t(h, h^{\star}_{i,t})| \leq d_{\mathcal{H}}(D_i, D_j)$$

$$|R_t(h) - R_t(h, h^{\star}_{i,t})| \leq R_t(h^{\star}_{i,t})$$

Plugging in (A.8), we have:

$$\begin{aligned}
&\leq \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}(R_i(h^{\star}_{i,t}) + R_t(h^{\star}_{i,t}) + d_{\mathcal{H}\Delta\mathcal{H}}(D_i, D_j)) \\
&= \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}(\lambda_{t,i} + d_{\mathcal{H}\Delta\mathcal{H}}(D_i, D_j))
\end{aligned} \tag{A.9}$$

$$= \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\lambda_{t,i} + \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(D_i, D_j) \tag{A.10}$$

Finally, for $t = 1, \dots, T$ tasks, the expected risk can be upper bounded by:

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h_t) \leq \frac{1}{T}\sum_{t=1}^{T} R_{\boldsymbol{\alpha}_t}(h_t) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(D_t, D_i) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\lambda_{t,i} \tag{A.11}$$

The next step is to find the high probability bound to measure the expected and empirical terms.

Step 2: With probability at least $1 - \delta/2$, the expected discrepancy can be upper bounded by:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t, \mathcal{D}_i) \leq \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{\alpha}_{t,i} \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t, S_i) + 2\sqrt{\frac{2d\log(2m_\star) + \log(32T/\delta)}{m_\star}}, \tag{A.12}$$

where $m_\star = \operatorname{argmin}_{m_{i,j}} \sqrt{\frac{2d\log(2m_{i,j}) + \log(32T/\delta)}{m_{i,j}}}$

Proof. For task t, from Corollary A.2, we have with probability less than δ':

$$d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t, \mathcal{D}_i) \geq \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t, S_i) + 2\sqrt{\frac{2d\log(2m_{ti}) + \log(2/\delta')}{m_{ti}}}$$

Then we have

$$\boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t, \mathcal{D}_i) \geq \boldsymbol{\alpha}_{t,i}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t, S_i) + 2\boldsymbol{\alpha}_{t,i}\sqrt{\frac{2d\log(2m_{ti}) + \log(2/\delta')}{m_{ti}}} \tag{A.13}$$

Then we set $\delta' = \delta/(4T)$, then apply the union bound, we know $\exists h$, such that

$$\begin{aligned}
\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t, \mathcal{D}_i) &\geq \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t, S_i) + \sum_{i=1}^{T} 2\boldsymbol{\alpha}_{t,i}\sqrt{\frac{2d\log(2m_{t\star}) + \log(8T/\delta)}{m_{t\star}}} \qquad \text{(A.14)} \\
&= \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t, S_i) + 2\sqrt{\frac{2d\log(2m_{t\star}) + \log(8T/\delta)}{m_{t\star}}},
\end{aligned}$$

where $m_{t\star} = \operatorname{argmin}_{m_{t,i}} \sqrt{\frac{2d\log(2m_{ti}) + \log(8T/\delta')}{m_{ti}}}$, is the $m_{t,i}$ which has the smallest complexity term.

Then we again apply the union bound over t, finally there exists a hypothesis h with probability smaller than $\delta/2$, holding the following bound:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_t,\mathcal{D}_i) \geq \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{\alpha}_{t,i}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t,S_i)+2\sqrt{\frac{2d\log(2m_\star)+\log(32T/\delta)}{m_\star}} \tag{A.15}$$

Where $m_\star = \operatorname{argmin}_{m_{i,j}}\sqrt{\frac{2d\log(2m_{i,j})+\log(32T/\delta)}{m_{i,j}}}$ is the $m_{i,j}$ which has the smallest complexity term. Finally, we have (A.12) under high probability $1-\delta/2$. □

Step 3: Applying the union bound and we have the high probability at least $1-\frac{\delta}{2}$, we have:

$$\frac{1}{T}\sum_{t=1}^{T}R_{\boldsymbol{\alpha}_t}(h_t) \leq \frac{1}{T}\sum_{t=1}^{T}\hat{R}_{\boldsymbol{\alpha}_t}(h_t)+2\sqrt{\frac{2(d\log(\frac{2em}{d})+\log(\frac{16T}{\delta}))}{m}}\sum_{t=1}^{T}\Big(\sqrt{\sum_{j=1}^{N}\frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\Big) \tag{A.16}$$

Step 4: Combining previous conclusions, we have at high probability at least $1-\delta$:

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}R_t(h_t) \leq & \frac{1}{T}\sum_{t=1}^{T}\hat{R}_{\boldsymbol{\alpha}_t}(h_t)+C_1\sum_{t=1}^{T}\Big(\sqrt{\sum_{i=1}^{T}\frac{\boldsymbol{\alpha}_{t,i}^2}{\beta_i}}\Big)+\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S_t,S_i) \\ & + C_2+\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\lambda_{t,i}\end{aligned}$$

Where $C_1 = 2\sqrt{\frac{2(d\log(\frac{2em}{d})+\log(\frac{16T}{\delta}))}{m}}$, $C_2 = 2\sqrt{\frac{2d\log(2m_\star)+\log(32T/\delta)}{m_\star}}$

A.3 Proof of Theorem 2.2

Theorem A.1. *Let $\mathcal{H}$ be a hypothesis family from $\mathcal{X}$ to $[0,1]$, with pseudo-dimension d and each member $h\in\mathcal{H}$ is K Lipschitz. We have T tasks generated by the underlying distributions and labeling functions $\{(\mathcal{D}_1,f_1),\dots,(\mathcal{D}_T,f_T)\}$ with observation numbers $m_1,\dots,m_T$. We adopt Wasserstein-1* [1] *distance as a similarity metric with cost function $c(\mathbf{x},\mathbf{y})=\|\mathbf{x}-\mathbf{y}\|_2$, then for any fixed simplex $\boldsymbol{\alpha}_t\in\mathbb{R}_+^T$, and for $\delta\in(0,1)$, with probability at least $1-\delta$, for $h_1,\dots,h_T\in\mathcal{H}$, we have:*

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}R_t(h_t) \leq & \underbrace{\frac{1}{T}\sum_{t=1}^{T}\hat{R}_{\boldsymbol{\alpha}_t}(h_t)}_{\textit{Weighted empirical loss}} + \underbrace{C_1\sum_{t=1}^{T}\Big(\sqrt{\sum_{j=1}^{T}\frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\Big)}_{\textit{Coefficient regularization}} + \underbrace{\frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}W_1(\hat{D}_t,\hat{D}_i)}_{\textit{Empirical distribution distance}} \\ & + \underbrace{C_2+\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\boldsymbol{\alpha}_{t,i}\lambda_{t,i}}_{\textit{Complexity term and optimal expected loss}}\end{aligned}$$

[1] This bound can be extended to any Wasserstein $p>1$ distance with restricting the hypothesis satisfies K Hölder condition.

Where $\beta_i = \frac{m_i}{m}$, $C_1 = 2\sqrt{\frac{2(d\log(\frac{2em}{d})+\log(\frac{16T}{\delta}))}{m}}$, $C_2 = \frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\gamma_{t,i}$ *with* $\gamma_{t,i} = \mu_t m_t^{-1/s} + \mu_i m_i^{-1/s} + \sqrt{\log(\frac{2T}{\delta})}(\sqrt{\frac{1}{m_t}} + \sqrt{\frac{1}{m_i}})$ *and* s *and* $\mu_{\cdot}$ *are some specified constants.*

The proof is analogue to the previous proof with different assumptions.

A.3.1 Transfer bounds

The proof extends the work of Redko et al. (2017) where the hypothesis is only restricted in the unit ball of RKHS. We extend this result to any hypothesis with Lipschitz function.

Lemma A.4. *Let* $\mathcal{D}_i$ *and* $\mathcal{D}_j$ *be two probability measures on* $\mathcal{X}$*. Assume that:*

1. *Cost function is the Euclidean distance, with the form* $c(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|$
2. *The hypothesis set* $\mathcal{H}$ *satisfies* K*-Lipschitz continuous:* $\forall h \in \mathcal{H}$*,* h *is* K*-Lipschtiz continuous.*

Then we have the following result:

$$R_j(h,h') \leq R_i(h,h') + 2KW_1(\mathcal{D}_i,\mathcal{D}_j) \tag{A.17}$$

for any hypothesis $h, h' \in \mathcal{H}$

Proof. According to the definition of the expected risk, we have:

$$\begin{aligned} R_j(h,h') &= R_j(h,h') + R_i(h,h') - R_i(h,h') \\ &\leq R_i(h,h') + |R_j(h,h') - R_i(h,h')| \\ &\leq R_i(h,h') + |\mathbb{E}_{y\sim\mathcal{D}_j}|h(y)-h'(y)| - \mathbb{E}_{x\sim\mathcal{D}_i}|h(x)-h'(x)|| \end{aligned}$$

By defining $\phi(x) = |h(x)-h'(x)|$, we have:

$$= R_i(h,h') + |\int_{\mathcal{X}} \phi d(D_j - D_i)|$$

$$= R_i(h,h') + |\int_{\mathcal{X}\times\mathcal{X}} \phi(x) - \phi(y) d\gamma(x,y)|$$

For **any** joint measure $\gamma(x,y)$, we have:

$$\leq R_i(h,h') + \int_{\mathcal{X}\times\mathcal{X}} |\phi(x)-\phi(y)| d\gamma(x,y)$$

Thus it will also satisfy the minimal w.r.t $\gamma(x,y)$:

$$\leq R_i(h,h') + \min_{\gamma(x,y)\in\Pi(\mathcal{D}_i,\mathcal{D}_j)} \int_{\mathcal{X}\times\mathcal{X}} |\phi(x)-\phi(y)| d\gamma(x,y)$$

We also have $|\phi(x)-\phi(y)| = ||h(x)-h'(x)| - |h(y)-h'(y)|| \leq |h(x)-h'(x)-h(y)+h'(y)| \leq 2K\|x-y\|$, plugging in we have:

$$R_j(h,h') \leq R_i(h,h') + 2KW_1(\mathcal{D}_i,\mathcal{D}_j)$$

□

Remark If the function satisfies (C, p)-Hölder condition with $|h(x) - h(y)| \leq C\|x - y\|^p$, then the conclusion can be extended to any p-Wasserstein distance.

A.3.2 Concentration bounds between empirical and expected divergence

There exists several concentration bounds such as Bolley et al. (2007); Weed and Bach (2017), we adopt the conclusion from Weed and Bach (2017) and apply to bound the empirical measures of Wasserstein distance.

Lemma A.5. *Weed and Bach (2017) [Definition 3,4] Given a measure μ on X, the (ϵ, τ)-covering number on a given set $S \subseteq X$ is:*

$$\mathcal{N}_\epsilon(\mu, \tau) := \inf\{\mathcal{N}_\epsilon(S) : \mu(S) \geq 1 - \tau\}$$

and the (ϵ, μ)-dimension is:

$$d_\epsilon(\mu, \tau) := \frac{\log \mathcal{N}_\epsilon(\mu, \epsilon)}{-\log \epsilon}$$

Then the upper Wasserstein dimensions can be defined as:

$$d_p^\star(\mu) = \inf\{s \in (2p, +\infty) : \limsup_{\epsilon \to 0} d_\epsilon(\mu, \epsilon^{-\frac{sp}{s-2p}}) \leq s\}$$

Lemma A.6. *Weed and Bach (2017)[Theorem 1, Proposition 20] For $p \geq 1$ and $s \geq d_p^\star(\mu)$, there exists a positive constant C with probability at least $1 - \delta$, we have:*

$$W_p^p(\mu, \hat{\mu}_n) \leq C n^{-1/s} + \sqrt{\frac{1}{2n} \log(\frac{1}{\delta})}$$

A.3.3 Concentration bounds between empirical and expected risk

In the regression problem, we suppose the hypothesis family $\mathcal{H}$ is a set of continuous mapping with *pseudo-dimension* d. Then we can directly apply the conclusion of (A.4). The procedure is analogue to the proof in $\mathcal{H}$ divergence but under different assumptions.

Step 1: For a pair of distribution $(\mathcal{D}_i, \mathcal{D}_j)$, for the task t we have:

$$|R_{\boldsymbol{\alpha}_t}(h) - R_t(h)| = |\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} R_i(h) - R_t(h)| \leq \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} |R_i(h) - R_t(h)|$$

$$\leq \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} \Big(|R_i(h) - R_i(h, h_{i,t}^\star)| + |R_i(h, h_{i,t}^\star) - R_t(h, h_{i,t}^\star)| + |R_t(h) - R_t(h, h_{i,t}^\star)| \Big)$$

According to the triangle inequality and the previous lemma, we have:

$$|R_i(h) - R_i(h, h_{i,t}^\star)| \leq R_i(h_{i,t}^\star)$$

$$|R_i(h, h_{i,t}^\star) - R_t(h, h_{i,t}^\star)| \leq 2K W_1(\mathcal{D}_i, \mathcal{D}_j)$$

$$|R_t(h) - R_t(h, h^\star_{i,t})| \leq R_t(h^\star_{i,t})$$

Plugging in, we have:

$$\leq \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}(R_i(h^\star_{i,t}) + R_t(h^\star_{i,t}) + 2KW_1(D_i, D_j)) = \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}(\lambda_{t,i} + 2KW_1(D_i, D_j)) \tag{A.18}$$

$$= \sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\lambda_{t,i} + 2K\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} W_1(D_i, D_j) \tag{A.19}$$

Summing over the $t = 1, \dots, T$:

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h_t) \leq \frac{1}{T}\sum_{t=1}^{T} R_{\boldsymbol{\alpha}_t}(h_t) + \frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} W_1(D_t, D_i) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\lambda_{t,i} \tag{A.20}$$

Step 2: The next step is to bound the expected and the empirical Wasserstein distance. According to the triangle inequality of Wasserstein distance, we have:

$$W_1(D_t, D_i) \leq W_1(D_t, \hat{\mathcal{D}}_t) + W_1(\hat{\mathcal{D}}_t, \mathcal{D}_i) \leq W_1(D_t, \hat{\mathcal{D}}_t) + W_1(\hat{\mathcal{D}}_t, \hat{\mathcal{D}}_i) + W_1(\hat{\mathcal{D}}_i, \mathcal{D}_i) \tag{A.21}$$

According to the concentration lemma, we have with probability $1 - \delta'/2$:

$$W_1(D_t, D_i) \leq W_1(\hat{\mathcal{D}}_t, \hat{\mathcal{D}}_i) + C_t m_t^{-1/s} + C_i m_i^{-1/s} + \sqrt{\frac{1}{2}\log(\frac{2}{\delta'})}(\sqrt{\frac{1}{m_t}} + \sqrt{\frac{1}{m_i}} \tag{A.22}$$

Then setting $\delta' = \frac{\delta}{T^2}$ and applying union bound, we have the following with probability at least $1 - \delta/2$:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} W_1(D_t, D_i) \leq \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} W_1(\hat{D}_t, \hat{D}_i) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \gamma_{t,i} \tag{A.23}$$

Where

$$\gamma_{t,i} = C_t N_t^{-1/s} + C_i N_i^{-1/s} + \sqrt{\log(\frac{2T}{\delta})}(\sqrt{\frac{1}{N_t}} + \sqrt{\frac{1}{N_i}})$$

Step 3: Then the next step is to bound the empirical and expected errors. Since here is the regression problem, we suppose the hypothesis family $\mathcal{H}$ is a set of continuous mapping with *pseudo-dimension* d. Then we combine with the previous lemma, with probability at least $1 - \delta$, the expected error can be upper bounded by:

$$\frac{1}{T}\sum_{t=1}^{T} R_t(h_t) \leq \frac{1}{T}\sum_{t=1}^{T} \hat{R}_{\boldsymbol{\alpha}_t}(h_t) + C_1 \sum_{t=1}^{T}\Big(\sqrt{\sum_{j=1}^{T} \frac{\boldsymbol{\alpha}_{t,j}^2}{\beta_j}}\Big) + \frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i} W_1(\hat{D}_t, \hat{D}_i) + C_2 + \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{T} \boldsymbol{\alpha}_{t,i}\lambda_{t,i} \tag{A.24}$$

Where $C_1 = 2\sqrt{\frac{2(d\log(\frac{2en}{d}) + \log(\frac{16T}{\delta}))}{n}}$, $C_2 = \frac{2K}{T}\sum_{t=1}^{T}\sum_{i=1}^{T}\gamma_{t,i}$ with $\gamma_{t,i} = C_t m_t^{-1/s} + C_i m_i^{-1/s} + \sqrt{\log(\frac{2T}{\delta})}(\sqrt{\frac{1}{m_t}} + \sqrt{\frac{1}{m_i}})$

Remark The bound proposed in (A.24) is analogue to the bound in the $\mathcal{H}$ Divergence measure with completely different assumptions. For example, the Wasserstein bound is derived from the real output function, which can be naturally applied in the regression problem.

A.4 Experiment details

A.4.1 Digits recognition

In the digit recognition, we used three different kinds of digits: Mnist (LeCun et al., 1998), MnistM (Arbelaez et al., 2011; Ganin et al., 2016) and SVHN (Netzer et al., 2011). As we described in the the paper, we only sample 3K, 5K and 8K examples for each task. The input image dimension is 28×28.

We used a modified LeNet-5 architecture for training the digit datasets.

- Feature extractor: with 2 convolution layers.
  'layer1': 'conv': [1, 32, 5, 1, 2], 'relu': [], 'maxpool': [3, 2, 0],
  'layer2': 'conv': [32, 64, 5, 1, 2], 'relu': [], 'maxpool': [3, 2, 0]
- Task prediction: with 2 fc layers.
  'layer3': 'fc': [*, 128], 'act_fn': 'elu',
  'layer4': 'fc': [128, 10], 'act_fn': 'softmax'
- Discriminator part: with 2 fc layers.
  reverse_gradient()
  'layer3': 'fc': [*, 128], 'act_fn': 'elu',
  'layer4': 'fc': [128, 10], 'act_fn': 'softmax'

Hyper-parameter setting We set the $\rho = \frac{1}{T}$, with T the number of the task. $\kappa_1 = 1$ and tuning the hyper-parameter κ_2 from 0.2 to 2 through grid search. In the Wasserstein-1 distance-based approach, we set the gradient penalty weight as 1.

As for the configurations for training the neural networks, we used *SGD* optimizer with learning rate 0.01 and momentum 0.9. The maximum training epoch is 100 for the proposed approach and baselines.

A.4.2 Amazon reviews

We also evaluate the proposed algorithm in *Amazon reviews* datasets. We extract reviews from four kinds of product (book, dvd disks, electronics and kitchen appliances). Reviews datasets are preprocessed with the same strategy from Ganin et al. (2016): 10K dimensional input features and binary output labels $\{0, 1\}$, "0" if the product is ranked less equal than 3 stars, and "1" if higher than 3 stars. For each task we have 1000 and 1600 labelled training examples, respectively.

We used the standard MLP architecture for training the preprocessed dataset.

- Feature extractor: with 2 fc layers.

'layer1': 'fc': [10000, 256], 'act_fn': 'elu',
'layer2': 'fc': [256,128], 'act_fn': 'elu',

- Task prediction: with 2 fc layers.
  'layer3': 'fc': [128, 64], 'act_fn': 'elu',
  'layer4': 'fc': [64, 1], 'act_fn': 'sigmoid'
- Discriminator part: with 2 fc layers.
  reverse_gradient()
  'layer3': 'fc': [128, 64], 'act_fn': 'elu',
  'layer4': 'fc': [64, 1], 'act_fn': 'sigmoid'

Hyper-parameter tuning We set the $\rho = \frac{1}{T}$, with T the number of the task. $\kappa_1 = 1$ and tuning the hyper-parameter κ_2 from 0.2 to 1 through grid search. In the Wasserstein-1 distance-based approach, we set the gradient penalty weight as 1.

As for the configurations for training the neural networks, we used *SGD* optimizer with learning rate 0.005 and momentum 0.9. The maximum training epoch is 100 for the proposed approach and baselines.

Appendix B

Details of Chapter 3

B.1 Theorem 3.1: Proof

Theorem B.1. *Supposing $\mathcal{D}$ is the data generation distribution and $\mathcal{Q}$ is the querying distribution, if the loss ℓ is symmetric, L-Lipschitz; $\forall h \in \mathcal{H}$ is at most H-Lipschitz function and underlying labeling function $h^\star$ is $\phi(\lambda)$-$(\mathcal{D}, \mathcal{Q})$ Joint Probabilistic Lipschitz, then the expected risk w.r.t. $\mathcal{D}$ can be upper bounded by:*

$$R_{\mathcal{D}}(h) \leq R_{\mathcal{Q}}(h) + L(H + \lambda)W_1(\mathcal{D}, \mathcal{Q}) + L\phi(\lambda)$$

B.1.1 Notations

We define the hypothesis $h : \mathcal{X} \to \mathcal{Y} = [0, 1]$ and loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$, then the expected risk w.r.t. $\mathcal{D}$ is $R_{\mathcal{D}}(h) = \mathbb{E}_{x\sim\mathcal{D}}\ell(h(x), h^\star(x))$ and empirical risk $\hat{R}_{\mathcal{D}}(f) = \frac{1}{N}\sum_{i=1}^{N} \ell(h(x_i), y_i)$. We assume the loss ℓ is symmetric, L-Lipschitz and bounded by M.

B.1.2 Transfer risk

The first step is to bound the the gap $R_{\mathcal{D}}(h) - R_Q(h)$:

$$\begin{aligned} R_{\mathcal{D}}(h) - R_Q(h) \leq |R_{\mathcal{D}}(h) - R_Q(h)| &= |E_{x\sim\mathcal{D}}\ell(h(x), h^\star(x)) - E_{x\sim\mathcal{Q}}\ell(h(x), h^\star(x))| \\ &= |\int_{x\in\Omega} \ell(h(x), h^\star(x))d(\mathcal{D} - \mathcal{Q})| \end{aligned} \tag{B.1}$$

From the Kantorovich - Rubinstein duality property and combing Eq. (B.1), for **any** distribution coupling $\gamma \in \Pi(\mathcal{D}, \mathcal{Q})$, with $\mathcal{D} \in \Omega$ and $\mathcal{Q} \in \Omega$ we have:

$$\begin{aligned} &= |\int_{\Omega\times\Omega} \big(\ell(h(x_{\mathcal{D}}), h^\star(x_{\mathcal{D}})) - \ell(h(x_{\mathcal{Q}}), h^\star(x_{\mathcal{Q}}))\big)d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}})| \\ &\leq \int_{\Omega\times\Omega} |\ell(h(x_{\mathcal{D}}), h^\star(x_{\mathcal{D}})) - \ell(h(x_{\mathcal{Q}}), h^\star(x_{\mathcal{Q}}))|d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}) \\ &\leq \int_{\Omega\times\Omega} |\ell(h(x_{\mathcal{D}}), h^\star(x_{\mathcal{D}})) - \ell(h(x_{\mathcal{D}}), h^\star(x_{\mathcal{Q}}))| + |\ell(h(x_{\mathcal{D}}), h^\star(x_{\mathcal{Q}})) - \ell(h(x_{\mathcal{Q}}), h^\star(x_{\mathcal{Q}}))|d\gamma(x_{\mathcal{D}}, x_{\mathcal{Q}}) \end{aligned}$$

Since we assume ℓ is symmetric and L-Lipschitz, then we have:

$$\leq L \int_{\Omega\times\Omega} |h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| d\gamma(x_\mathcal{D}, x_\mathcal{Q}) + L \int_{\Omega\times\Omega} |h(x_\mathcal{D}) - h(x_\mathcal{Q})| d\gamma(x_\mathcal{D}, x_\mathcal{Q}) \tag{B.2}$$

From Eq.(B.2) the risk gap is controlled by two terms, the property of labeling function and property of predictor. Moreover, we assume the learner is H-Lipschitz function, then we have:

$$\leq L \int_{\Omega\times\Omega} |h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| d\gamma(x_\mathcal{D}, x_\mathcal{Q}) + LH \int_{\Omega\times\Omega} ||x_\mathcal{D} - x_\mathcal{Q}||_2 d\gamma(x_\mathcal{D}, x_\mathcal{Q})$$

Labeling function assumption As mentioned before, the goodness of the underlying labeling function decides the level of risk. (Urner et al., 2013) formalize the such a property as *Probabilistic Lipschitz* condition in AL, in which relaxes the condition of Lipschitzness and formalizes the intuition that *under suitable feature representation the probability of two close points having different labels is small* (Urner and Ben-David, 2013). We adopt the joint probabilistic Lipschitz property, which is coherent with (Courty et al., 2017).

Definition B.1. The labeling function $h^\star$ satisfies $\phi(\lambda)$-$(\mathcal{D}, \mathcal{Q})$ Joint Probabilistic Lipschitz if $\mathrm{supp}(\mathcal{Q}) \subseteq \mathrm{supp}(\mathcal{D})$ and for all $\lambda > 0$:

$$\mathbb{P}_{(x_\mathcal{D}, x_\mathcal{Q})\sim\gamma}[|h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| > \lambda \|x_\mathcal{D} - x_\mathcal{Q}\|_2] \leq \phi(\lambda) \tag{B.3}$$

Where $\phi(\lambda)$ reflects the decay rate. (Urner et al., 2013) showed that the faster the decay of $\phi(\lambda)$ with $\lambda \to 0$, the nicer the distribution and the easier it is to learn the task.

Combining with Eq.(B.3), the labeling function term can be decomposed and upper bounded by:

$$\begin{aligned}
&\leq L \int_{\Omega\times\Omega} \mathbf{1}\{|h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| \leq \lambda \|x_\mathcal{D} - x_\mathcal{Q}\|_2\} |h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| d\gamma(x_\mathcal{D}, x_\mathcal{Q}) \\
&+ L \int_{\Omega\times\Omega} \mathbf{1}\{|h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q}) > \lambda \|x_\mathcal{D} - x_\mathcal{Q}\|_2\} |h^\star(x_\mathcal{D}) - h^\star(x_\mathcal{Q})| d\gamma(x_\mathcal{D}, x_\mathcal{Q}) \\
&\leq L\lambda \int_{\Omega\times\Omega} \|x_\mathcal{D} - x_\mathcal{Q}\|_2 d\gamma(x_\mathcal{D}, x_\mathcal{Q}) + L\phi(\lambda)
\end{aligned}$$

The first term is upper bounded through the probability of this event at most 1 and second term adopts the definition of Joint Probabilistic Lipschitz with restricting the output space $h^\star(\cdot) \in [0, 1]$. Plugging in the aforementioned results, we have:

$$\leq L(H + \lambda) \int_{\Omega\times\Omega} \|x_\mathcal{D} - x_\mathcal{Q}\|_2 d\gamma(x_\mathcal{D}, x_\mathcal{Q}) + L\phi(\lambda)$$

Since this inequality satisfies with any distribution coupling γ, then it is also satisfies with the optimal coupling, w.r.t. the Wasserstein-1 distance with the cost function ℓ_2 distance: $\| \cdot \|_2$. Then we have:

$$R_\mathcal{D}(h) - R_\mathcal{Q}(h) \leq L(H + \lambda) \inf_\gamma \int_{\Omega\times\Omega} \|x_\mathcal{D} - x_\mathcal{Q}\|_2 d\gamma(x_\mathcal{D}, x_\mathcal{Q}) + L\phi(\lambda)$$

Finally we can derive:

$$R_\mathcal{D}(h) \leq R_\mathcal{Q}(h) + L(H + \lambda) W_1(\mathcal{D}, \mathcal{Q}) + L\phi(\lambda) \tag{B.4}$$

B.2 Corollary 3.1: Proof

B.2.1 Basic statistical learning theory

According to the standard statistical learning theory such as (Mohri et al., 2018), w.h.p. $1-\delta/2$, $\forall h \in \mathcal{H}$ we have:

$$R_{\mathcal{D}}(h) \leq \hat{R}_{\mathcal{D}}(h) + 2L\text{Rad}_N(h) + \kappa_1(\delta, N) \tag{B.5}$$

Where $\text{Rad}_N(h) = \mathbb{E}_{S\sim\mathcal{D}^N}\mathbb{E}_{\sigma_1^N}[\sup_h \frac{1}{N}\sum_{i=1}^N \sigma_i h(x_i)]$ is the expected Rademacher complexity with $\text{Rad}_N(h) = \mathcal{O}(\sqrt{\frac{1}{N}})$, and $\kappa_1(\delta, N) = \mathcal{O}(\sqrt{\frac{M\log(2/\delta)}{N}})$ is the confidence term.
In the Active learning, the goal is to control the generalization error w.r.t. $(\mathcal{D}, h^\star)$, thus from Eq.B.5 we have:

$$R_{\mathcal{D}}(h) \leq (R_{\mathcal{D}}(h) - R_{\mathcal{Q}}(h)) + \hat{R}_{\mathcal{Q}}(h) + 2L\text{Rad}_{N_q}(h) + \kappa_1(\delta, N_q)$$

Combining with Eq.(B.4), we have

$$R_{\mathcal{D}}(h) \leq \hat{R}_{\mathcal{Q}}(h) + L(H+\lambda)W_1(\mathcal{D},\mathcal{Q}) + L\phi(\lambda) + 2L\text{Rad}_{N_q}(h) + \kappa_1(\delta, N_q)$$

In general we have finite observations (suppose we have the sample i.i.d. sampled from the query distribution $\mathcal{Q}$) with Dirac distribution: $\hat{D} = \frac{1}{N}\sum_{i=1}^N \delta\{x_{\mathcal{D}}^i\}$ and $\hat{Q} = \frac{1}{N_q}\sum_{i=1}^{N_q}\delta\{x_{\mathcal{Q}}^i\}$ with $N_q \leq N$. Several recent works show the concentration bound between empirical and expected Wasserstein distance such as (Bolley et al., 2007; Weed and Bach, 2017). We just adopt the conclusion from (Weed and Bach, 2017) and apply to bound empirical measures in Wasserstein-1 distance.

Lemma B.1. *(Weed and Bach, 2017) [Definition 3,4] Given a measure μ on X, the (ϵ,τ)-covering number on a given set $S \subseteq X$ is:*

$$\mathcal{N}_\epsilon(\mu,\tau) := \inf\{\mathcal{N}_\epsilon(S) : \mu(S) \geq 1-\tau\}$$

and the (ϵ,μ)-dimension is:

$$d_\epsilon(\mu,\tau) := \frac{\log \mathcal{N}_\epsilon(\mu,\epsilon)}{-\log\epsilon}$$

Then the upper Wasserstein-1 dimensions can be defined as:

$$d_1^\star(\mu) = \inf\{s \in (2,+\infty) : \limsup_{\epsilon\to 0} d_\epsilon(\mu, \epsilon^{-\frac{s}{s-2}}) \leq s\}$$

Lemma B.2. *(Weed and Bach, 2017)[Theorem 1, Proposition 20] For $p=1$ and $s \geq d_1^\star(\mu)$, there exists a positive constant C with probability at least $1-\delta$, we have:*

$$W_1(\mu,\hat{\mu}_N) \leq CN^{-1/s} + \sqrt{\frac{1}{2N}\log(\frac{1}{\delta})}$$

Since $s > 2$ thus the convergence rate of Wasserstein distance is slower than $\mathcal{O}(N^{-1/2})$, also named as *weak convergence*. Then according to the triangle inequality of Wasserstein-1 distance, we have:

$$W_1(\mathcal{D},\mathcal{Q}) \leq W_1(\mathcal{D},\hat{\mathcal{D}}) + W_1(\hat{\mathcal{D}},\mathcal{Q}) \leq W_1(\mathcal{D},\hat{\mathcal{D}}) + W_1(\hat{\mathcal{D}},\hat{\mathcal{Q}}) + W_1(\hat{\mathcal{Q}},\mathcal{Q}) \tag{B.6}$$

Combing the conclusion with Lemma 2, there exist some constants (C_d, s_d) and (C_q, s_q) we have with probability $1-\delta/2$:

$$W_1(\mathcal{D},\mathcal{Q}) \leq W_1(\hat{\mathcal{D}},\hat{\mathcal{Q}}) + C_d N^{-1/s_d} + C_q N_q^{-1/s_q} + \sqrt{\frac{1}{2}\log(\frac{2}{\delta})}(\sqrt{\frac{1}{N}} + \sqrt{\frac{1}{N_q}}) \tag{B.7}$$

Combining Eq.B.7 and Eq.B.5, we have

$$R_{\mathcal{D}}(h) \leq \hat{R}_{\mathcal{Q}}(h) + L(H+\lambda)W_1(\hat{\mathcal{D}},\hat{\mathcal{Q}}) + L\phi(\lambda) + 2L\mathrm{Rad}_{N_q}(h) + \kappa(\delta, N, N_q)$$

Where $\kappa(\delta, N, N_q) = \mathcal{O}(N^{-1/s_d} + N_q^{-1/s_q} + \sqrt{\frac{\log(1/\delta)}{N}} + \sqrt{\frac{\log(1/\delta)}{N_q}})$

B.3 Computing $\mathcal{H}$-divergence and Wasserstein distance

B.3.1 $\mathcal{H}$ divergence

$d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_3)$ We can discuss the discrepancy with different values of p, since $\mathcal{D}_3 \subseteq \mathcal{D}_1$ then we have $x_0 \in [a+b, 2a-b]$:

1. If $p \leq x_0 - b$, then the area of mis-classification will be $(2a-p)+2b$. If we select $p = x_0 - b$, then the optimal mis-classification area will be $2a+2b-x_0 \geq 2a+2b-2a+b = 3b$
2. If $p \in [x_0-b, x_0+b]$, then the area of mis-classification will be $(2a-p)+(p-(x_0-b)) = 2a+b-x_0 \geq 2a+b-2a+b = 2b$
3. If $p \geq x_0+b$, then the area of mis-classification will be $2b + \max(0, 2a-p)$, if we select $p \geq 2a$, then the optimal mis-classification area will be $2b$.

Then the minimal mis-classification area is $2b$, corresponding the optimal risk $\frac{b}{a+b}$, then $d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_3) = \frac{b}{a+b}$

$d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_2)$ We can discuss the discrepancy with different values of p, since $\mathcal{D}_2 \subseteq \mathcal{D}_1$, then we have $x_0 \in [a+b/2, 2a-b/2]$:

1. If $p \leq -x_0 - b/2$, then the mis-classification area will be $a + \max(0, p+a) + 2b$ with optimal value $2a + b/2 - x_0 + 2b \geq a + 2b$;
2. If $p \in [-x_0-b/2, -x_0+b/2]$, then the mis-classification area will be $p-(-x_0-b/2)+(-a-p)+a+b = x_0+3b/2 \geq a+2b$;
3. If $p \in [-x_0+b/2, x_0-b/2]$, then the mis-classification area will be $b+a$;
4. If $p \in [x_0-b/2, x_0+b/2]$, then the mis-classification area will be $b+p-(x_0-b/2)+(2a-p) = 2a+3b/2-x_0 \geq 2a+b/2-2a+b/2+b = 2b$
5. If $p \geq x_0+b/2$, then the mis-classification area will be $2b+\max(0, 2a-p) \geq 2b$

Then the minimal mis-classification area is $2b$, corresponding the optimal risk $\frac{b}{a+b}$, then $d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_2) = \frac{b}{a+b}$.
From the previous example $d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_2) = d_{\mathcal{H}}(\mathcal{D}_1, \mathcal{D}_3)$, we show the $\mathcal{H}$ divergence is not a good

metric for measuring the representative in the data space. Since we want the query distribution more diverse spread in the space, then $\mathcal{H}$ may not be a good indicator.

B.3.2 Wasserstein-1 distance

We can also estimate the distribution distance through Wasserstein-1 metric. From (Wasserman, 2019) we have:

$$W_1(P, Q) = \int_0^1 |F^{-1}(z) - G^{-1}(z)| dz$$

where $F(z)$ and $G(z)$ is the CDF (cumulative density function) of distribution P and Q, respectively.

CDF of $\mathcal{D}_1$, $\mathcal{D}_2$ and $\mathcal{D}_3$

1.

$$F_1(z) = \begin{cases} \frac{1}{2a}(z + 2a) & -2a \leq z \leq -a \\ \frac{1}{2} & -a \leq z \leq a \\ \frac{1}{2a} z & a \leq z \leq 2a \end{cases}$$

$$F_1^{-1}(z) = \begin{cases} 2a(z-1) & 0 \leq z < 1/2 \\ [-a, a] & z = 1/2 \\ 2az & 1/2 < z \leq 1 \end{cases}$$

2.

$$F_2(z) = \begin{cases} \frac{1}{2b}(z + x_0 + b/2) & -x_0 - b/2 \leq z \leq -x_0 + b/2 \\ \frac{1}{2} & -x_0 + b/2 \leq z \leq x_0 - b/2 \\ \frac{1}{2b}(z - x_0 + 3b/2) & x_0 - b/2 \leq z \leq x_0 + b/2 \end{cases}$$

$$F_2^{-1}(z) = \begin{cases} 2bz - x_0 - b/2 & 0 \leq z < 1/2 \\ [-x_0 + b/2, x_0 - b/2] & z = 1/2 \\ 2bz + x_0 - 3b/2 & 1/2 < z \leq 1 \end{cases}$$

3.

$$F_3(z) = \frac{1}{2b}(z - x_0 + b) \;\; z \in [x_0 - b, x_0 + b]$$

$$F_3^{-1}(z) = 2bz + x_0 - b \;\; z \in [0, 1]$$

Computing $W_1(\mathcal{D}_1, \mathcal{D}_2)$ According to the definition, we can compute

$$W_1(\mathcal{D}_1, \mathcal{D}_2) = \int_0^{1/2} |2a(z-1) - 2bz - x_0 - \frac{b}{2}| dz + \int_{1/2}^1 |2az - 2bz - x_0 + \frac{3}{2} b| dz$$

We firstly compute $\int_0^{1/2}|2a(z-1)-2bz-x_0-\frac{b}{2}|dz$, since $2a(z-1)-2bz-x_0-\frac{b}{2}<0$ for $z\in[0,1/2]$ (since $-a-b-x_0-b/2<0$. Then we have:

$$\begin{aligned}\int_0^{1/2}|2a(z-1)-2bz-x_0-\frac{b}{2}|dz &= \int_0^{1/2}\{-2a(z-1)+2bz+x_0+\frac{b}{2}\}dz\\ &= \frac{3}{4}a+\frac{1}{2}x_0+\frac{1}{2}b\end{aligned}$$

Then we compute the second part:

$$\begin{aligned}&\int_{1/2}^{1}|2az-2bz-x_0+\frac{3}{2}b|dz\\ &=\int_{1/2}^{z_0}\{(x_0-\frac{3}{2}b)-2(a-b)z\}dz+\int_{z_0}^{1}\{2(a-b)z-x_0+\frac{3}{2}b\}dz\\ &=\frac{1}{2(a-b)}(x_0-\frac{3}{2}b)^2-\frac{3}{2}(x_0-\frac{3}{2}b)+\frac{3}{4}(a-b)\end{aligned}$$

with $z_0=\frac{x_0-3b/2}{2(a-b)}$. Therefore, we can compute the wasserstein-1 distance between distribution $\mathcal{D}_1$ and $\mathcal{D}_2$:

$$=\frac{1}{2(a-b)}(x_0-\frac{3}{2}b)^2-x_0+2b+\frac{3}{2}a$$

With $x_0\in[a+b/2,2a-b/2]$. If we take $x_0=2a-b/2$, we can get the maximum:

$$\max_{x_0}W_1(\mathcal{D}_1,\mathcal{D}_2)=\frac{3}{2}a-\frac{b}{2}$$

Computing $W_1(\mathcal{D}_1,\mathcal{D}_3)$ According to definition, we can compute

$$W_1(\mathcal{D}_1,\mathcal{D}_3)=\int_0^{1/2}|2a(z-1)-2bz-x_0+b|dz+\int_{1/2}^{1}|2az-2bz-x_0+b|dz$$

We firstly compute $\int_0^{1/2}|2a(z-1)-2bz-x_0+b|dz$, since $2(a-b)z-2a-x_0+b\le 0$ for $z\in[0,1/2]$. (easy to verify: $2(a-b)z-2a-x_0+b\le(a-b)-2a-x_0+b=-a-x_0<0$), then

$$\begin{aligned}&\int_0^{1/2}|2a(z-1)-2bz-x_0+b|dz=\int_0^{1/2}(x_0+2a-b)-2(a-b)zdz\\ &=\frac{1}{2}(x_0+2a-b)-\frac{1}{4}(a-b)=\frac{3}{4}a+x_0-\frac{1}{4}b\end{aligned}$$

Then we compute the second term $\int_{1/2}^{1}|2az-2bz-x_0+b|dz$, we define $z_0=\frac{x_0-b}{2(a-b)}$ and we can verify that $z_0\in[1/2,1]$, then this term can be decomposed as we can rewrite it as:

$$\int_{1/2}^{z_0}-2(a-b)z+(x_0-b)dz+\int_{z_0}^{1}2(a-b)z-(x_0-b)dz$$

$$=\frac{(x_0-b)^2}{2(a-b)}-\frac{3}{2}(x_0-b)+\frac{5}{4}(a-b)$$

Then $W_1(\mathcal{D}_1, \mathcal{D}_3) = \frac{(x_0-b)^2}{2(a-b)} - \frac{3}{2}(x_0 - b) + \frac{5}{4}(a-b) + \frac{1}{2}x_0 + \frac{3}{4}a - \frac{b}{4} = \frac{1}{2(a-b)}(x_0 - b)^2 - x_0 + 2a = \frac{1}{2(a-b)}(x_0 - b)^2 - x_0 + 2a$ since $x_0 \in [a+b, 2a-b]$, then we have:

$$\min_{x_0} W_1(\mathcal{D}_1, \mathcal{D}_3) = \frac{(x_0 - b)^2}{2(a-b)} - (a+b) + 2a = \frac{a^2}{2(a-b)} + a - b$$

We can verify: $\frac{a^2}{2(a-b)} + a - b > \frac{3}{2}a - \frac{b}{2}$ when $a > b$, then we have:

$$\min_{x_0} W_1(\mathcal{D}_1, \mathcal{D}_3) > \max_{x_0} W_1(\mathcal{D}_1, \mathcal{D}_2)$$

which means in Wasserstein-1 distance metric, the diversity of two distribution can be much better measured.

B.4 Developing loss in deep batch active learning

We have the original loss:

$$\min_{\boldsymbol{\theta}^f, \boldsymbol{\theta}^h, \hat{B}} \max_{\boldsymbol{\theta}^d} \mathbb{E}_{(x,y)\sim\hat{L}\cup\hat{B}} \ell(h(x,y)) + \mu\big(\mathbb{E}_{x\sim\hat{\mathcal{D}}}[g(x)] - \mathbb{E}_{x\sim\hat{L}\cup\hat{B}}[g(x)]\big). \tag{B.8}$$

Since $\hat{L}$, $\hat{B}$ and $\hat{\mathcal{D}}$ are Dirac distributions, then we have:

$$\begin{aligned}
&\frac{1}{L+B} \sum_{(x,y)\in\hat{L}\cup\hat{B}} \ell(h(x,y)) + \mu\big(\frac{1}{L+U}\sum_{x\in\hat{D}} g(x) - \frac{1}{L+B}\sum_{x\in\hat{L}\cup\hat{B}} g(x)\big) \\
&= \frac{1}{L+B}\sum_{(x,y)\in\hat{L}} \ell(h(x,y)) + \frac{1}{L+B}\sum_{(x,y^{?})\in\hat{B}} \ell(h(x,y^{?})) \\
&+ \mu\big(\frac{1}{L+U}\sum_{x\in\hat{L}} g(x) + \frac{1}{L+U}\sum_{x\in\hat{U}} g(x) - \frac{1}{L+B}\sum_{x\in\hat{L}} g(x) - \frac{1}{L+B}\sum_{x\in\hat{B}} g(x)\big) \\
&= \frac{1}{L+B}\sum_{(x,y)\in\hat{L}} \ell(h(x,y)) + \frac{1}{L+B}\sum_{(x,y^{?})\in\hat{B}} \ell(h(x,y^{?})) \\
&+ \mu\big(\frac{1}{L+U}\sum_{x\in\hat{U}} g(x) - (\frac{1}{L+B} - \frac{1}{L+U})\sum_{x\in\hat{L}} g(x)\big) - \frac{\mu}{L+B}\sum_{x\in\hat{B}} g(x) \\
&= \underbrace{\Big(\frac{1}{L+B}\sum_{(x,y)\in\hat{L}} \ell(h(x,y)) + \mu\big(\frac{1}{L+U}\sum_{x\in\hat{U}} g(x) - (\frac{1}{L+B} - \frac{1}{L+U})\sum_{x\in\hat{L}} g(x)\big)\Big)}_{\text{Training Stage}} \\
&+ \underbrace{\Big(\frac{1}{L+B}\sum_{(x,y^{?})\in\hat{B}} \ell(h(x,y^{?})) - \frac{\mu}{L+B}\sum_{x\in\hat{B}} g(x)\Big)}_{\text{Querying Stage}}
\end{aligned}$$

We note that $x \in \hat{D}$ means enumerating all samples from the observations (empirical distribution).

B.5 Redundancy trick: Computation

$$
\begin{aligned}
&\mu\big(\frac{1}{L+U}\sum_{x\in\hat{U}} g(x) - (\frac{1}{L+B} - \frac{1}{L+U})\sum_{x\in\hat{L}} g(x)\big) \\
&= \mu\big(\frac{\gamma}{1+\gamma}\frac{1}{U}\sum_{x\in\hat{U}} g(x) - (\frac{1}{1+\alpha} - \frac{1}{1+\gamma})\frac{1}{L}\sum_{x\in\hat{L}} g(x)\big) \\
&= \mu'\big(\frac{1}{U}\sum_{x\in\hat{U}} g(x) - \frac{1}{\gamma}(\frac{1+\gamma}{1+\alpha} - 1)\frac{1}{L}\sum_{x\in\hat{L}} g(x)\big) \\
&= \mu'\big(\frac{1}{U}\sum_{x\in\hat{U}} g(x) - \frac{1}{\gamma}\frac{\gamma-\alpha}{1+\alpha}\frac{1}{L}\sum_{x\in\hat{L}} g(x)\big)
\end{aligned}
$$

B.6 Uniform Output Arrives with the Minimal loss

For the abuse of notation, we suppose the output of classifier $h(x,\cdot) = [p_1,\ldots,p_K] \equiv \mathbf{p}$ with $p_i > 0$ and $\sum_{i=1}^K p_i = 1$. Then we tried to minimize

$$\min_{\mathbf{p}} \sum_{i=1}^{K} -\log p_i$$

By applying the Lagrange Multiplier approach, we have

$$\min_{\mathbf{p},\lambda>0} \sum_{i=1}^{K} -\log p_i + \lambda(\sum_{i=1}^{K} p_i - 1)$$

Then we do the partial derivative w.r.t. p_i, then we have $\forall i$:

$$\frac{-1}{p_i} + \lambda = 0 \ \rightarrow \ p_i = \frac{1}{\lambda}$$

Given $\sum_{i=1}^K p_i = 1$, then we can compute $p_i = \frac{1}{K}$ arriving at the minimal, i.e the uniform distribution.

B.7 Experiments

B.7.1 Dataset Descriptions

Dataset	#Classes	Train + Validation	Test	Initially labelled	Query size	Image size
Fashion-MNIST (Xiao et al., 2017)	10	40K + 20K	10K	1K	500	28 × 28
SVHN (Netzer et al., 2011)	10	40K + 33K	26K	1K	1K	32 × 32
CIFAR10 (Krizhevsky et al., 2009)	10	45K + 5K	10K	2K	2K	32 × 32
STL10* (Coates et al., 2011)	10	8K + 1K	4K	0.5K	0.5K	96 × 96

Table B.1 – Dataset descriptions

*We used a variant instead of the original STL10 dataset by arranging the training size to 8K (each class 800) and validation 1K and test 4K. We do not use the unlabeled dataset in our training or test procedure.

B.7.2 Implementation details

FashionMNIST For the FashionMNIST dataset, we adopted the LeNet5 as the feature extractor, then we used two-layer MLPs for the classification (320-50-relu-dropout-10) and critic function (320-50-relu-dropout-1-sigmoid).

SVHN, CIFAR10 We adopt the VGG16 with batch normalization as the feature extractor. then we used two-layer MLPs for the classification (512-50-relu-dropout-10) and critic function (512-50-relu-dropout-1-sigmoid).

STL10 We adopt the VGG16 with batch normalization as the feature extractor. then we used two-layer MLPs for the classification (4096-100-relu-dropout-10) and critic function (4096-100-relu-dropout-1-sigmoid).

B.7.3 Hyper-parameter setting

Dataset	lr	Momentum	Mini-Batch size	μ	Selection coefficient	Mixture coefficient**
Fashion-MNIST	0.01*	0.5	64	1e-2	5	0.5
SVHN	0.01*	0.5	64	1e-2	5	0.5
CIFAR10	0.01*	0.3	64	1e-2	10	0.5
STL10	0.01*	0.3	64	1e-3	10	0.5

Table B.2 – Hyper-parameter setting

* We set the initial learning rate as 0.01, then at 50% epoch we decay to 1e-3, after 75% epoch we decay to 1e-4.

** The mixture coefficient means the convex combination coefficient in the two uncertainty-based approaches.

B.7.4 Detailed results with numerical values

We report the accuracy in the form of percentage (%), showing in Tab. B.3, B.4, B.5, B.6.

B.8 Ablation study

In this part, we will conduct $\mathcal{H}$-divergence based adversarial training for the parameters of DNN.

$$\min_{\boldsymbol{\theta}^f,\boldsymbol{\theta}^h} \max_{\boldsymbol{\theta}^d} \sum_{(x,y)\in\hat{L}} \ell(h(x,y)) - \mu\big(\sum_{x\in\hat{U}} \log(g(x)) + \sum_{x\in\hat{L}} \log(1-g(x))\big) \tag{B.9}$$

Where g is defined as the discriminator function [1]. In the adversarial training, the discriminator parameter aims at discriminating the empirical unlabeled and labeled data via the binary classification,

[1] This notation is slightly different from the critic function (Arjovsky et al., 2017)

	Random	LeastCon	Margin	Entropy	KMedian	DBAL	Core-set	DeepFool	WAAL
1K	58.03±2.81	57.93±1.62	57.81±2.19	57.40±1.75	57.62±2.5	58.01±2.75	58.14±2.19	58.19±2.4	72.29±1.16
1.5K	66.81±1.02	64.24±2.49	65.61± 2.5	66.37±0.62	67.13±2.87	66.53±2.5	68.79±1.99	66.21±1.78	76.99±1.05
2K	71.21±2.35	68.36±1.09	70.05±2.77	69.70±0.88	71.57±0.79	69.77±0.93	71.22±1.38	70.14±1.32	79.85±0.49
2.5K	73.12±2.1	71.68±1.67	72.74±1.55	71.60±1.42	73.84±0.98	72.60±0.60	72.61±1.16	71.77±1.49	81.08±0.68
3K	75.80±0.64	75.03±1.56	76.55±1.01	74.84±1.29	75.79±0.44	74.75±1.04	73.77±1.74	73.69±1.21	82.04±0.58
3.5K	77.34±0.67	77.73±1.04	78.99±1.11	76.66±1.26	77.44±0.97	75.86±1.02	75.10±1.11	74.00±0.71	82.74±0.79
4K	78.68±0.41	79.26±0.47	81.77±0.51	79.00±0.24	77.97±0.65	77.02±0.42	76.28 ±0.98	74.93±2.05	83.25±0.62
4.5K	79.58±0.47	80.08±0.82	82.32±0.47	79.89±0.78	79.49±0.7	77.90±0.58	77.30±0.61	76.64±0.97	83.96±0.54
5K	80.02±0.45	81.32±0.64	83.89±0.84	80.85±0.87	79.97±0.59	78.87±0.58	78.34±0.37	77.24±0.69	84.45±0.45
5.5K	80.93±0.33	83.21±0.42	84.87±0.18	82.26±0.77	81.11±0.41	79.47±2.9	78.42±0.66	77.72±0.57	85.20±0.44
6K	81.30±0.25	84.50±0.73	85.52±0.27	83.66±0.98	81.86±0.6	80.43±0.76	79.66±0.34	78.99±0.33	85.99±0.43

Table B.3 – Result of FashionMNIST (Average ± std)

	Random	LeastCon	Margin	Entropy	KMedian	DBAL	Core-set	DeepFool	WAAL
1K	63.97±2.04	63.40±2.16	63.10±2.3	63.49±2.79	63.50±2.53	63.76±0.73	63.90±1.07	63.62±2.34	75.18±1.41
2K	75.85±1.16	74.86±2.44	75.27± 1.7	72.78±3.15	76.17±3.2	77.07±1.57	77.9±1.25	76.29±1.62	80.69±2.00
3K	80.83± 1.04	81.87± 0.64	80.9± 2.22	80.88± 1.26	81.36± 1.29	81.17± 1.72	81.7 ± 0.84	80.92± 0.79	83.89± 2.08
4K	82.70±1.18	84.00±0.88	83.10±1.38	83.19±0.95	83.41±1.58	83.95±1.87	84.81±1.3	83.79±0.64	86.82±1.11
5K	85.10±0.73	85.68±0.94	85.02±1.1	84.75±0.83	84.93±0.94	86.34±1.1	86.52±0.95	85.32±0.58	88.71±1.08
6K	86.20±0.48	87.23±0.97	87.53±0.63	87.51±0.50	87.04±0.45	87.61±0.72	88.00 ±0.44	87.02±0.64	89.71±0.83

Table B.4 – Result of SVHN (Average ± std)

	Random	LeastCon	Margin	Entropy	KMedian	DBAL	Core-set	DeepFool	WAAL
2K	46.33±3.18	46.43±3.17	46.69±3.87	46.79±3.62	46.53±3.39	46.48±3.11	46.38±4.03	46.54±3.77	55.00±0.40
4K	56.33±3.40	53.26±3.84	55.52± 2.69	53.13±2.99	53.58±2.57	56.18±2.37	56.09±3.89	54.48±1.62	62.32±0.36
6K	59.63± 4.17	59.00± 2.19	63.05± 1.78	62.63± 1.29	61.25±1.76	62.48± 1.38	59.56 ± 1.17	60.80± 0.70	66.67± 0.60
8K	62.85±3.37	66.46±1.33	66.44±1.85	65.23±1.89	63.73±1.34	65.84±0.78	65.84±1.27	64.87±1.98	69.33±1.47
10K	68.13±2.53	68.91±1.10	69.86±0.24	69.72±1.53	68.92±2.33	68.94±1.96	69.11±0.80	69.39±0.47	72.39±1.21
12K	70.41±1.02	71.90±1.35	72.25±0.68	71.58±0.77	72.65±0.64	72.25±1.24	72.60 ±0.79	71.17±1.03	75.11±0.49

Table B.5 – Result of CIFAR10 (Average ± std)

	Random	LeastCon	Margin	Entropy	KMedian	DBAL	Core-set	DeepFool	WAAL
0.5K	41.78±2.42	41.69±3.22	41.81±2.27	41.12±1.67	41.24±1.41	41.30±1.45	41.41±2.30	41.82±2.67	47.01±1.09
1K	48.24±1.37	47.05±1.42	46.7±0.85	46.38±2.31	46.45±1.11	47.45±3.71	47.58±2.06	45.15±0.74	52.47±1.62
1.5K	51.78± 2.5	50.87± 1.24	50.44± 2.57	50.24± 1.21	49.91±1.74	52.53± 1.29	51.2 ± 1.63	48.64± 2.43	57.25± 1.78
2K	56.52±1.78	56.25±1.58	55.54±1.09	55.15±2.13	54.92±2.19	57.54±1.70	58.13±1.57	54.26±2.40	60.08±1.63
2.5K	58.42±1.42	58.49±2.05	57.62±1.42	57.81±2.87	57.87±1.51	59.25±2.89	57.66±1.79	57.05±2.53	62.58±1.44
3K	61.13±1.67	60.80±2.64	59.42±1.49	60.88±0.72	60.00±0.65	62.11±1.65	61.02 ±0.48	59.74±1.74	65.42±1.33

Table B.6 – Result of STL10 (Average ± std)

while the feature extractor parameter aims at not being correctly classified. In this manner, the unlabeled dataset will be used for constructing a better feature representation in the adversarial training. As for the query part, we directly used the baseline strategies. The numerical values will show in Tab. B.7. Moreover, we also evaluated the ablation study for the SVHN dataset, shown in Tab. B.8.

	Random	LeastCon	Margin	Entropy	KMedian	DBAL	Core-set	DeepFool	WAAL
2K	49.85±0.32	50.00±1.81	49.8±2.28	49.92±1.08	49.31±2.76	49.58±1.05	49.87±4.03	49.85±1.36	55.00±0.40
4K	56.63±3.27	59.11±0.85	61.93± 2.12	59.15±0.41	60.6±0.72	58.55±1.99	60.97±1.62	58.80±2.59	62.32±0.36
6K	62.30± 2.54	63.15± 2.21	63.04± 1.98	63.74± 0.94	64.73±1.37	63.82± 2.33	64.95 ± 1.66	64.80± 1.4	66.67± 0.60
8K	66.97±0.76	64.32±2.58	68.30±1.02	67.67±1.04	65.98±1.45	66.65±1.00	67.54±2.16	67.65±1.27	69.33±1.47
10K	69.23±1.97	69.74±2.52	69.98±0.25	69.92±1.17	70.95±1.93	69.96±1.74	70.62±0.74	70.55±0.80	72.39±1.21
12K	71.78±1.34	71.60±1.25	71.56±1.53	72.90±1.37	72.56±1.39	73.53±1.71	71.83 ±1.20	71.86±0.33	75.11±0.49

Table B.7 – Ablation study of CIFAR10 (Average ± std)

	Random	LeastCon	Margin	Entropy	KMedian	DBAL	Core-set	DeepFool	WAAL
1K	68.34±0.96	70.3±1.75	68.88±1.19	68.94±1.17	68.38±0.92	68.54±3.65	69.29±0.71	70.14±1.84	75.18±1.41
2K	76.63±3.14	75.21±2.45	74.55± 3.16	73.55±2.49	78.35±1.63	76.97±1.19	77.17±1.8	76.74±2.15	80.69±2.00
3K	80.36± 0.46	80.14±1.66	78.66± 1.54	76.10± 1.46	79.16± 1.47	78.99± 1.57	79.87 ± 0.33	80.10± 1.27	83.89± 2.08
4K	82.62±1.15	82.81±0.66	83.13±1.01	83.27±0.18	82.89±0.73	82.65±1.61	84.33±0.72	83.47±0.74	86.82±1.11
5K	84.27±0.77	85.59±0.74	84.36±0.75	86.15±0.23	85.10±0.57	84.18±0.25	86.74±0.34	84.75±0.57	88.71±1.08
6K	85.36±0.36	86.44±0.93	86.15±0.89	86.72 ± 0.66	87.21±0.52	86.77±1.26	87.31±0.71	85.42 ±0.55	89.71±0.83

Table B.8 – Ablation study of SVHN (Average ± std)

Appendix C

Details of Chapter 4

C.1 $\mathcal{H}$-Divergence v.s. Jensen-Shannon Divergence

C.1.1 Counterexample One

We take the example proposed by (Ben-David et al., 2010b) (Example 6), which has already computed the $d_{\mathcal{H}}(\mathcal{S}(x), \mathcal{T}(x)) = \xi$. However, since $\text{supp}(\mathcal{S}(x)) \cap \text{supp}(\mathcal{T}(x)) = \emptyset$, $D_{\text{JS}}(\mathcal{S}(x) \| \mathcal{T}(x)) = 1$.

C.1.2 Counterexample Two

We have $\mathcal{S} = \text{Unif}\{1, 2, 3\}$ and $\mathcal{T} = \{\mathbb{P}(X = 1) = \frac{1}{4}, \mathbb{P}(X = 2) = \frac{1}{2}, \mathbb{P}(X = 3) = \frac{1}{4}\}$.

Computing $d_{\mathcal{H}}$ It is also related to the optimal classification error.

$$\text{err}(h) = \begin{cases} 1/2 & \text{if} \quad t < 1, t > 3 \\ 11/24 & \text{if} \quad 1 < t < 2 \\ 13/24 & \text{if} \quad 2 < t < 3 \end{cases}$$

Then the $\mathcal{H}$ divergence is $d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x)) = 1 - 2\min_h[\text{err}(h)] = \frac{1}{12} \approx 0.0833$

Computing $D_{\textbf{JS}}(\mathcal{T}(x) \| \mathcal{S}(x))$ Since the two distributions hold the same support, we can compute the mixture distribution $\mathcal{M} = \{\mathbb{P}(X = 1) = \frac{7}{24}, \mathbb{P}(X = 2) = \frac{5}{12}, \mathbb{P}(X = 3) = \frac{7}{24}\}$, We can compute the Jensen-Shannon divergence:

$$D(\mathcal{S} \| \mathcal{M}) = \tfrac{1}{3}\log(\tfrac{1/3}{7/24}) + \tfrac{1}{3}\log(\tfrac{1/3}{5/12}) + \tfrac{1}{3}\log(\tfrac{1/3}{7/24}) \approx 0.02110$$
$$D(\mathcal{T} \| \mathcal{M}) = \tfrac{1}{4}\log(\tfrac{1/4}{7/24}) + \tfrac{1}{2}\log(\tfrac{1/2}{5/12}) + \tfrac{1}{4}\log(\tfrac{1/4}{7/24}) \approx 0.02032$$

Then $D_{\text{JS}}(\mathcal{T}(x) \| \mathcal{S}(x)) = \frac{1}{2}(0.0211 + 0.02032) = 0.0207$. In this scenario, the $D_{\text{JS}}(\mathcal{T}(x) \| \mathcal{S}(x)) < d_{\mathcal{H}}(\mathcal{T}(x), \mathcal{S}(x))$, therefore, the D_{JS} can not be viewed as an upper bound of $d_{\mathcal{H}}$.

C.2 Domain Adaptation: Upper Bound (Theorem 4.1)

We first prove an intermediate lemma:

Lemma C.1. *Let $Z \in \mathcal{Z}$ be the real valued integrable random variable, let P and Q be 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 distributions Q and P, and the equality arrives when $f(z) = \mathbb{E}_P f(z) + \frac{1}{\lambda}\log(\frac{dQ}{dP})$.

Proof. We 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 we can verify that $\mathbb{E}_P(Z_g) = 1$. We 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:

$$\begin{aligned}
\mathbb{E}_Q[\log Z_g] &= \mathbb{E}_Q[\log \frac{q(z)}{p(z)} + \log(Z_g \frac{p(z)}{q(z)})] \\
&= D_{\text{KL}}(Q\|P) + \mathbb{E}_Q[\log(Z_g \frac{p(z)}{q(z)})] \\
&\leq D_{\text{KL}}(Q\|P) + \log \mathbb{E}_Q[\frac{p(z)}{q(z)} Z_g] \\
&= D_{\text{KL}}(Q\|P) + \log \mathbb{E}_P[Z_g]
\end{aligned}$$

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:

$$\mathbb{E}_Q[g(z)] \leq \log \mathbb{E}_P[e^{g(z)}] + D_{\text{KL}}(Q\|P) \tag{C.1}$$

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 $\forall\lambda$ and f we have:

$$\mathbb{E}_Q \lambda(f(z) - \mathbb{E}_P f(z)) \leq D_{\text{KL}}(Q\|P) + \log \mathbb{E}_P[e^{\lambda(f(z)-\mathbb{E}_P f(z)}]$$

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:

$$\lambda(\mathbb{E}_Q f(z) - \mathbb{E}_P f(z)) \leq D_{\text{KL}}(Q\|P) + \log \mathbb{E}_P[e^{\lambda(\mathbb{E}_Q f(z)-\mathbb{E}_P f(z))}]$$

As for the attainment in the equality of Eq.(C.1), 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 $\lambda(f(z) - \mathbb{E}_P f(z)) = \log(\frac{dQ}{dP})$. □

In the classification problem, we define the observation pair $z = (x, y)$. We also define the loss function $\ell(z) = L \circ h(z)$ with deterministic hypothesis h and prediction loss function L. Then for abuse of notation, we simply denote the loss function $\ell(z)$ in this part.

Supposing the prediction loss L is bounded with interval G with $G = \max(L) - \min(L)$, then the expected risk in the target domain can be upper bounded by:

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}}\sqrt{D_{\text{JS}}(\mathcal{T}\|\mathcal{S})}$$

Where $D_{\text{JS}} = \frac{1}{2}\big[D(\mathcal{T}\|\frac{1}{2}(\mathcal{T}+\mathcal{S})) + D(\mathcal{S}\|\frac{1}{2}(\mathcal{T}+\mathcal{S}))\big]$ is the joint Jensen-Shannon divergence.

Proof. According to Lemma 1, $\forall\lambda > 0$ we have:

$$\mathbb{E}_Q f(z) - \mathbb{E}_P f(z) \leq \frac{1}{\lambda}(\log \mathbb{E}_P\, e^{[\lambda(f(z) - \mathbb{E}_P f(z))]} + D_{\text{KL}}(Q\|P)) \tag{C.2}$$

And $\forall\lambda < 0$ we have:

$$\mathbb{E}_Q f(z) - \mathbb{E}_P f(z) \geq \frac{1}{\lambda}(\log \mathbb{E}_P\, e^{[\lambda(f(z) - \mathbb{E}_P f(z))]} + D_{\text{KL}}(Q\|P)) \tag{C.3}$$

Then we introduce an intermediate distribution $\mathcal{M}(z) = \frac{1}{2}(\mathcal{S}(z) + \mathcal{T}(z))$, then $\text{supp}(\mathcal{S}) \subseteq \text{supp}(\mathcal{M})$ and $\text{supp}(\mathcal{T}) \subseteq \text{supp}(\mathcal{M})$, and let $f = \ell$. Since the random variable ℓ is bounded through $G = \max(L) - \min(L)$, then according to (Wainwright, 2019)(Chapter 2.1.2), $\ell - \mathbb{E}_P\ell$ is sub-Gaussian with parameter at most $\sigma = \frac{G}{2}$, then we can apply Sub-Gaussian property to bound the $\log$ moment generation function:

$$\log \mathbb{E}_P\, e^{[\lambda(\ell(z) - \mathbb{E}_P \ell(z))]} \leq \log e^{\frac{\lambda^2\sigma^2}{2}} \leq \frac{\lambda^2 G^2}{8}.$$

In Eq.(C.2), we let $Q = \mathcal{T}$ and $P = \mathcal{M}$, then $\forall\lambda > 0$ we have:

$$\mathbb{E}_{\mathcal{T}}\,\ell(z) - \mathbb{E}_{\mathcal{M}}\,\ell(z) \leq \frac{G^2\lambda}{8} + \frac{1}{\lambda}D_{\text{KL}}(\mathcal{T}\|\mathcal{M}) \tag{C.4}$$

In Eq.(C.3), we let $Q = \mathcal{S}$ and $P = \mathcal{M}$, then $\forall\lambda < 0$ we have:

$$\mathbb{E}_{\mathcal{S}}\,\ell(z) - \mathbb{E}_{\mathcal{M}}\,\ell(z) \geq \frac{G^2\lambda}{8} + \frac{1}{\lambda}D_{\text{KL}}(\mathcal{S}\|\mathcal{M}) \tag{C.5}$$

In Eq.(C.4), we denote $\lambda = \lambda_0 > 0$ and $\lambda = -\lambda_0 < 0$ in Eq.(C.5). Then Eq.(C.4), Eq.(C.5) can be reformulated as:

$$\begin{aligned}\mathbb{E}_{\mathcal{T}}\,\ell(z) - \mathbb{E}_{\mathcal{M}}\,\ell(z) &\leq \frac{G^2\lambda_0}{8} + \frac{1}{\lambda_0}D_{\text{KL}}(\mathcal{T}\|\mathcal{M}) \\ \mathbb{E}_{\mathcal{M}}\,\ell(z) - \mathbb{E}_{\mathcal{S}}\,\ell(z) &\leq \frac{G^2\lambda_0}{8} + \frac{1}{\lambda_0}D_{\text{KL}}(\mathcal{S}\|\mathcal{M})\end{aligned} \tag{C.6}$$

Adding the two inequalities in Eq.(C.6), we therefore have:

$$\mathbb{E}_{\mathcal{T}}\, \ell(z) \leq \mathbb{E}_{\mathcal{S}}\, \ell(z) + \frac{1}{\lambda_0}\big(D_{\text{KL}}(\mathcal{S}\|\mathcal{M}) + D_{\text{KL}}(\mathcal{T}\|\mathcal{M})\big) + \frac{\lambda_0}{4}G^2 \tag{C.7}$$

Since the inequality holds for $\forall\lambda_0$, then by taking $\lambda_0 = \frac{2}{G}\sqrt{D_{\text{KL}}(\mathcal{S}\|\mathcal{M}) + D_{\text{KL}}(\mathcal{T}\|\mathcal{M})}$ we finally have:

$$\mathbb{E}_{\mathcal{T}}\, \ell(z) \leq \mathbb{E}_{\mathcal{S}}\, \ell(z) + \frac{G}{\sqrt{2}}\sqrt{D_{\text{JS}}(\mathcal{T}\|\mathcal{S})} \tag{C.8}$$

□

C.2.1 Extension to the unbounded loss

The proposed theory can be naturally extended to the unbounded loss.

Corollary C.1 (Sub-Gaussian Upper Bound)**.** *If the loss function satisfies σ-Sub Gaussian property: $\log \mathbb{E}_P\, e^{[\lambda(\ell(z)-\mathbb{E}_P\ell(z))]} \leq \frac{\lambda^2\sigma^2}{2}$, then the expected risk in the target domain can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \sigma\sqrt{2D_{\text{JS}}(\mathcal{T}\|\mathcal{S})}$$

Proof. The proof is trivial by simply plugging in the Sub-Gaussian condition in the moment generation function. □

Corollary C.2 (Sub-Gamma Upper Bound)**.** *If the loss function satisfies (σ, a)-Sub Gamma property: $\log \mathbb{E}_P\, e^{[\lambda(\ell(z)-\mathbb{E}_P\ell(z))]} \leq \frac{\lambda^2\sigma}{2(1-a|\lambda|)}$, for $0 < |\lambda| < \frac{1}{a}$. Then the expected risk in the target domain can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + (\sigma+1)\sqrt{2D_{JS}(\mathcal{T}\|\mathcal{S})} + 2aD_{JS}(\mathcal{T}\|\mathcal{S})$$

Proof. For the same step for the moment generation function, by taking $\lambda_0 \in (0, \frac{1}{a})$, then analogously we have:

$$\begin{aligned}
\mathbb{E}_{\mathcal{T}}\, \ell(z) - \mathbb{E}_{\mathcal{M}}\, \ell(z) &\leq \frac{\lambda_0\sigma}{2(1-a\lambda_0)} + \frac{1}{\lambda_0}D_{\text{KL}}(\mathcal{T}\|\mathcal{M}) \\
\mathbb{E}_{\mathcal{M}}\, \ell(z) - \mathbb{E}_{\mathcal{S}}\, \ell(z) &\leq \frac{\lambda_0\sigma}{2(1-a\lambda_0)} + \frac{1}{\lambda_0}D_{\text{KL}}(\mathcal{S}\|\mathcal{M})
\end{aligned}$$

Therefore we have

$$\begin{aligned}
\mathbb{E}_{\mathcal{T}}\, \ell(z) - \mathbb{E}_{\mathcal{S}}\, \ell(z) &\leq \frac{\lambda_0\sigma}{(1-a\lambda_0)} + \frac{1}{\lambda_0}\left(D_{\text{KL}}(\mathcal{T}\|\mathcal{M}) + D_{\text{KL}}(\mathcal{S}\|\mathcal{M})\right) \\
&= \frac{\lambda_0\sigma}{(1-a\lambda_0)} + \frac{1}{\lambda_0}\left(2D_{\text{JS}}(\mathcal{T}\|\mathcal{S})\right)
\end{aligned}$$

We let $\lambda_0 = \frac{\sqrt{2D_{\text{JS}}(\mathcal{T}\|\mathcal{S})}}{\sigma+a\sqrt{2D_{\text{JS}}(\mathcal{T}\|\mathcal{S})}} \in (0, \frac{1}{a})$ and we can simplify the upper bound as:

$$\mathbb{E}_{\mathcal{T}}\, \ell(z) - \mathbb{E}_{\mathcal{S}}\, \ell(z) \leq (\sigma+1)\sqrt{2D_{\text{JS}}(\mathcal{T}\|\mathcal{S})} + 2aD_{\text{JS}}(\mathcal{T}\|\mathcal{S})$$

□

The extended upper bounds can be much tighter than the conclusion in Theorem 1, particularly when the loss is in a large range with a small variance.

C.3 Domain Adaptation Theory: Lower Bound (Theorem 4.2)

We firstly introduce several information theoretical tools:

Lemma C.2 (Pinsker's inequality)**.** *If P and Q are two probability distribution on the measurable space $(\Omega, \mathcal{F})$, then*

$$\text{TV}(P,Q) \leq \sqrt{2D_{KL}(P\|Q)}$$

Where $D_{KL}(P\|Q)$ is the Kullback–Leibler divergence between distribution P and Q and $TV(P\|Q) = \sum_z |P(z) - Q(z)|$

Lemma C.3. *(Polyanskiy and Wu, 2019)[f-divergence data processing inequality] Consider a channel that produces Y given X on the deterministic function g. If P_Y is the distribution of Y when X is generated by P_X and Q_Y is the distribution of Y when X is generated by Q_X, then for any f-divergence $D_f(\cdot\|\cdot)$:*

$$D_f(P_Y\|Q_Y) \leq D_f(P_X\|Q_X)$$

If we restrict the zero-one loss $L \in \{0,1\}$, then we can prove the target risk be lower bounded by:

$$R_{\mathcal{T}}(h) \geq R_{\mathcal{S}}(h) - \sqrt{D_{\text{JS}}(\mathcal{T}\|\mathcal{S})}$$

Proof. Again we denote the observation pair $z = (x, y)$. For abuse of notation, we simply denote the loss function $\ell = L \circ h$ with $\ell \in \{0, 1\}$.

According to f-divergence data processing inequality, if we set the deterministic function g as $g(Z) = \mathbf{1}_E(Z)$ for any event E, then Y is Bernoulli distribution with parameter $P(E)$ or $Q(E)$ and the data processing inequality becomes:

$$D_f(\text{Bern}(P(E))\|\text{Bern}(Q(E))) \leq D_f(P_Z\|Q_Z)$$

If we define the event E as we make an error in the prediction (a.k.a $l(z) = 1$), then $P(E) = P(\text{making an error}) = E_P\mathbf{1}\{\text{making an error}\} = \mathbb{E}_P[\ell(z)]$. Therefore we have:

$$D_f(\text{Bern}(\mathbb{E}_P[\ell(z)])\|\text{Bern}(\mathbb{E}_Q[\ell(z)])) \leq D_f(P_Z\|Q_Z)$$

Again we introduce the intermediate distribution $\mathcal{M} = \frac{1}{2}(\mathcal{S}+\mathcal{T})$. According to the data processing inequality on the expectation of random variables, if we adopt KL divergence by letting $f(t) = t\log(t)$, then we have:

$$\begin{aligned}
&D_{\mathrm{KL}}(\mathrm{Bern}(\mathbb{E}_{\mathcal{T}}[\ell(z)])\|\mathrm{Bern}(\mathbb{E}_{\mathcal{M}}[\ell(z)])) \leq D_{\mathrm{KL}}(\mathcal{T}\|\mathcal{M}) \\
&D_{\mathrm{KL}}(\mathrm{Bern}(\mathbb{E}_{\mathcal{S}}[\ell(z)])\|\mathrm{Bern}(\mathbb{E}_{\mathcal{M}}[\ell(z)])) \leq D_{\mathrm{KL}}(\mathcal{S}\|\mathcal{M})
\end{aligned}$$

We notice $\mathbb{E}_{\mathcal{T}}(\ell(z)) \in [0,1], \mathbb{E}_{\mathcal{S}}(\ell(z)) \in [0,1]$. Then we can adopt Pinsker's inequality by treating the expected value as the Bernoulli distribution parameters. Then we can compute their Total Variation (TV) distance.

$$\mathrm{TV}(\mathrm{Bern}(p), \mathrm{Bern}(q)) = |p-q| + |1-p-1+q| = 2|p-q|$$

Then we have:

$$\begin{aligned}
2|\mathbb{E}_{\mathcal{T}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)]| &= \mathrm{TV}(\mathrm{Bern}(p), \mathrm{Bern}(q)) \\
&\leq \sqrt{2D_{\mathrm{KL}}(\mathrm{Bern}(\mathbb{E}_{\mathcal{T}}[\ell(z)])\|\mathrm{Bern}(\mathbb{E}_{\mathcal{M}}[\ell(z)]))} \\
&\leq \sqrt{2D_{\mathrm{KL}}(\mathcal{T}\|\mathcal{M})}
\end{aligned}$$

Similarity we have $2|\mathbb{E}_{\mathcal{S}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)]| \leq \sqrt{2D_{\mathrm{KL}}(\mathcal{S}\|\mathcal{M})}$. Adding these two item together we have:

$$2|\mathbb{E}_{\mathcal{S}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)]| + 2|\mathbb{E}_{\mathcal{T}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)]| \leq \sqrt{2D_{\mathrm{KL}}(\mathcal{T}\|\mathcal{M})} + \sqrt{2D_{\mathrm{KL}}(\mathcal{S}\|\mathcal{M})}$$

We adopt the inequality $\sqrt{a}+\sqrt{b} \leq \sqrt{2(a+b)}$ with $a \geq 0$ and $b \geq 0$, then we have

$$\sqrt{D_{\mathrm{KL}}(\mathcal{T}\|\mathcal{M})} + \sqrt{D_{\mathrm{KL}}(\mathcal{S}\|\mathcal{M})} \leq 2\sqrt{D_{\mathrm{JS}}(\mathcal{T}\|\mathcal{S})}.$$

We also have

$$\begin{aligned}
&2|\mathbb{E}_{\mathcal{S}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)]| + 2|\mathbb{E}_{\mathcal{T}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)]| \\
&\geq 2|\mathbb{E}_{\mathcal{S}}[\ell(z)] - \mathbb{E}_{\mathcal{M}}[\ell(z)] - \mathbb{E}_{\mathcal{T}}[\ell(z)] + \mathbb{E}_{\mathcal{M}}[\ell(z)]| \\
&= 2|\mathbb{E}_{\mathcal{S}}[\ell(z)] - \mathbb{E}_{\mathcal{T}}[\ell(z)]|
\end{aligned}$$

Given the aforementioned results, we have the following the two side inequality:

$$|\mathbb{E}_{\mathcal{S}}[\ell(z)] - \mathbb{E}_{\mathcal{T}}[\ell(z)]| \leq \sqrt{D_{\mathrm{JS}}(\mathcal{T}\|\mathcal{S})}$$

We have $-\sqrt{D_{\mathrm{JS}}(\mathcal{T}\|\mathcal{S})} \leq \mathbb{E}_{\mathcal{T}}[\ell(z)] - \mathbb{E}_{\mathcal{S}}[\ell(z)] \leq \sqrt{D_{\mathrm{JS}}(\mathcal{T}\|\mathcal{S})}$ and finally we have the lower bound:

$$\mathbb{E}_{\mathcal{T}}[\ell(z)] \geq \mathbb{E}_{\mathcal{S}}[\ell(z)] - \sqrt{D_{\mathrm{JS}}(\mathcal{T}\|\mathcal{S})}$$

Remark We should point out the derived upper bound is looser and more restrictive than that we derived from Theorem 1, with a scale $\frac{1}{\sqrt{2}}$ when we restrict the loss in $\{0,1\}$ and Theorem 1 can be extended to any bounded loss while this proof **cannot**. □

C.4 Joint Jensen-Shannon Divergence Decomposition (Corollary 4.3)

In this section, we will provide an upper bound of the chain rule in Jensen-Shannon divergence. According to the definition of Jensen-Shannon divergence and the chain rule of KL divergence we have:

$$\begin{aligned}
2D_{\text{JS}}(\mathcal{T}(x,y)\|\mathcal{S}(x,y)) &= D_{\text{KL}}(\mathcal{T}(x,y)\|\mathcal{M}(x,y)) + D_{\text{KL}}(\mathcal{S}(x,y)\|\mathcal{M}(x,y)) \\
&= D_{\text{KL}}(\mathcal{T}(x)\|\mathcal{M}(x)) + \mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{KL}}(\mathcal{T}(y|x)\|\mathcal{M}(y|x)) \\
&+ D_{\text{KL}}(\mathcal{S}(x)\|\mathcal{M}(x)) + \mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{KL}}(\mathcal{S}(y|x)\|\mathcal{M}(y|x)) \\
&= 2D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x)) + \mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{KL}}(\mathcal{T}(y|x)\|\mathcal{M}(y|x)) \\
&+ \mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{KL}}(\mathcal{S}(y|x)\|\mathcal{M}(y|x))
\end{aligned}$$

In general, for continuous random variable, the D_{KL} divergence does not exist an exact upper bound. While we can simple upper bound these by adding two complementary terms.

$$\begin{aligned}
\mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{KL}}(\mathcal{T}(y|x)\|\mathcal{M}(y|x)) &\leq \mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{KL}}(\mathcal{T}(y|x)\|\mathcal{M}(y|x)) + \mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{KL}}(\mathcal{S}(y|x)\|\mathcal{M}(y|x)) \\
&= 2\mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{JS}}(\mathcal{T}(y|x)\|\mathcal{S}(y|x))
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{KL}}(\mathcal{T}(y|x)\|\mathcal{M}(y|x)) &\leq \mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{KL}}(\mathcal{T}(y|x)\|\mathcal{M}(y|x)) + \mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{KL}}(\mathcal{S}(y|x)\|\mathcal{M}(y|x)) \\
&= 2\mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{JS}}(\mathcal{T}(y|x)\|\mathcal{S}(y|x))
\end{aligned}$$

Plugging in the results, we have the following conditional upper bound

$$\begin{aligned}
\sqrt{D_{\text{JS}}(\mathcal{T}(x,y)\|\mathcal{S}(x,y))} \leq& \sqrt{D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))} + \sqrt{\mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{JS}}(\mathcal{T}(y|x)\|\mathcal{S}(y|x))} \\
&+ \sqrt{\mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{JS}}(\mathcal{T}(y|x)\|\mathcal{S}(y|x))}
\end{aligned}$$

We can derive the analogue result conditioned on y:

$$\begin{aligned}
\sqrt{D_{\text{JS}}(\mathcal{T}(x,y)\|\mathcal{S}(x,y))} \leq& \sqrt{D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))} + \sqrt{\mathbb{E}_{y\sim\mathcal{T}(y)} D_{\text{JS}}(\mathcal{T}(x|y)\|\mathcal{S}(x|y))} \\
&+ \sqrt{\mathbb{E}_{y\sim\mathcal{S}(y)} D_{\text{JS}}(\mathcal{T}(x|y)\|\mathcal{S}(x|y))}
\end{aligned}$$

C.5 Target Intrinsic Error Upper Bound

If $H(Y_s|X_s) \leq \epsilon$, the source target marginal and conditional distribution are close $D_{\text{JS}}(\mathcal{S}(x)\|\mathcal{T}(x)) \leq \delta_1$, $\forall x$, we have $D_{\text{JS}}(\mathcal{S}(y|x)\|\mathcal{T}(y|x)) \leq \delta_2$. Then the target distribution conditional entropy can be upper bounded by:

$$H(Y_t|X_t) \leq \epsilon + \sqrt{\frac{\delta_2}{2}} + \frac{\sqrt{\delta_1}}{2}\log|\mathcal{Y}|$$

Proof. Since $\frac{1}{2}\text{TV}(P,Q)^2 \leq D_{\text{JS}}(P\|Q) \leq TV(P,Q)$ (Thekumparampil et al., 2018), then for $\forall x$ we have:

$$\|\mathcal{S}(y|x) - \mathcal{T}(y|x)\|_1 \leq \sqrt{2\delta_2}$$

Then for conditional entropy for the target distribution, we have:

$$\begin{aligned} H(Y_t|X_t) &= \mathbb{E}_{x\sim\mathcal{T}(x)} H(Y_t|X_t = x) \\ &= \mathbb{E}_{x\sim\mathcal{T}(x)} H(Y_t|X = x) - \mathbb{E}_{x\sim\mathcal{T}(x)} H(Y_s|X = x) + \mathbb{E}_{x\sim\mathcal{T}(x)} H(Y_s|X = x) \\ &\leq \mathbb{E}_{x\sim\mathcal{T}(x)} |H(Y_t|X = x) - H(Y_s|X = x)| + \mathbb{E}_{x\sim\mathcal{T}(x)} H(Y_s|X = x) \end{aligned}$$

Since the Entropy function is $\frac{1}{2}$ Lipschitz w.r.t. L_1 norm, then we have

$$\mathbb{E}_{x\sim\mathcal{T}(x)} |H(Y_t|X = x) - H(Y_s|X = x)| \leq \mathbb{E}_{x\sim\mathcal{T}(x)} \frac{1}{2} \|\mathcal{T}(y|x) - \mathcal{S}(y|x)\|_1 \leq \sqrt{\frac{\delta_2}{2}}$$

Then we need to bound $\mathbb{E}_{x\sim\mathcal{T}(x)} H(Y_s|X = x)$,

$$\begin{aligned} E_{x\sim\mathcal{T}(x)} H(Y_s|X = x) &= E_{x\sim\mathcal{S}(x)} H(Y_s|X = x) + E_{x\sim\mathcal{T}(x)} H(Y_s|X = x) - E_{x\sim\mathcal{S}(x)} H(Y_s|X = x) \\ &\leq \epsilon + E_{x\sim\mathcal{T}(x)} H(Y_s|X = x) - E_{x\sim\mathcal{S}(x)} H(Y_s|X = x) \end{aligned}$$

We still adopt the conclusion when we proof Theorem 1, i.e the transport inequality of the gaps of same function under different marginal distribution measures by assuming $z = x$. We can compute $G = H(Y_s|X = x) \leq H(Y_s) \leq \log|\mathcal{Y}|$, then we have:

$$\begin{aligned} E_{x\sim\mathcal{T}(x)} H(Y_s|X = x) &\leq \epsilon + \frac{\log|\mathcal{Y}|}{\sqrt{2}} \sqrt{D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))} \\ &\leq \epsilon + \sqrt{\frac{\delta_1}{2}} \log|\mathcal{Y}| \end{aligned}$$

Putting all them together we have the aforementioned conclusion. □

C.6 Inherent Difficulty for Controlling Label Conditional Shift

C.6.1 Extension to the Representation Learning

The upper bound in Theorem 4.1 can be further decomposed as:

$$\begin{aligned} R_{\mathcal{T}}(h) \leq & R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}} \sqrt{D_{\text{JS}}(\mathcal{T}(x)\|\mathcal{S}(x))} \\ & + \frac{G}{\sqrt{2}} \sqrt{\mathbb{E}_{x\sim\mathcal{T}(x)} D_{\text{JS}}(\mathcal{T}(\cdot|x)\|\mathcal{S}(\cdot|x)) + \mathbb{E}_{x\sim\mathcal{S}(x)} D_{\text{JS}}(\mathcal{T}(\cdot|x)\|\mathcal{S}(\cdot|x))} \end{aligned} \quad \text{(C.9)}$$

Inspired by (Johansson et al., 2020), we set the representation function $g : \mathcal{X} \to \mathcal{Z}$ and h the hypothesis defined on the (x, z). Then we consider learning twice-differentiable, invertible representations: $g : \mathcal{X} \to \mathcal{Z}$ where g^{-1} is the inverse representation, such that $g^{-1}(g(x)) = x$ for all x. Then these assumptions for $g(x)$, we have $P(g(X) = z) = P(X = g^{-1}(z))$.

We can therefore extend the result in the representaion learning:

$$\begin{aligned}R_{\mathcal{T}}(h\circ g) \leq & R_{\mathcal{S}}(h\circ g) + \frac{G\sqrt{A_g}}{\sqrt{2}}\sqrt{D_{\text{JS}}(\mathcal{T}(z)\|\mathcal{S}(z))} \\ & + \frac{G}{\sqrt{2}}\sqrt{\mathbb{E}_{x\sim\mathcal{T}(x)}D_{\text{JS}}(\mathcal{T}(\cdot|z)\|\mathcal{S}(\cdot|z)) + \mathbb{E}_{x\sim\mathcal{S}(x)}D_{\text{JS}}(\mathcal{T}(\cdot|z)\|\mathcal{S}(\cdot|z))}\end{aligned}$$

Where $A_g = \sup_z |J_{g^{-1}}(z)|$, is the maximum value of the Jacobian of the representation inverse function g^{-1}.

As we mentioned in this and previous paper (Wu et al., 2020b; Zhao et al., 2019a; Johansson et al., 2019; Wu et al., 2019), only controlling the first two terms by learning a bad representation can lead to the third term much larger.

Proof. According to the definition of f-divergence and define $z = g(x)$, under the aforementioned assumptions, we have:

$$\begin{aligned}D_f(P(x)\|Q(x)) &= \int_x q(x) f(\frac{p(x)}{q(x)})dx \\ &= \int_z q(g(x)) f(\frac{p(g(x))}{q(g(x))})|J_{g^{-1}}(z)|dz \\ &\leq A_g D_f(P(z)\|Q(z)),\end{aligned}$$

where $A_g = \sup_z |J_{g^{-1}}(z)|$, is the maximum value of the Jacobian of the representation inverse function g^{-1}. □

C.6.2 Lower Bound of Label Conditional Shift

We can prove the label-conditional shift can be lower bounded by:

$$\begin{aligned}&\mathbb{E}_{z\sim\hat{\mathcal{T}}(z)}D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) + \mathbb{E}_{z\sim\hat{\mathcal{S}}(z)}D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) \\ &\qquad \geq 2\left(\sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))} - \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))}\right)^2\end{aligned}$$

We notice the square form of Jensen-Shannon divergence is a valid statistical distance. Then we have:

$$\begin{aligned}&\sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))} = \sqrt{D_{\text{JS}}\left(\sum_z \hat{\mathcal{T}}(y|z)\hat{\mathcal{T}}(z)\| \sum_z \hat{\mathcal{S}}(y|z)\hat{\mathcal{S}}(z)\right)} \\ &\leq \sqrt{D_{\text{JS}}\left(\sum_z \hat{\mathcal{T}}(y|z)\hat{\mathcal{S}}(z)\| \sum_z \hat{\mathcal{S}}(y|z)\hat{\mathcal{S}}(z)\right)} + \sqrt{D_{\text{JS}}\left(\sum_z \hat{\mathcal{T}}(y|z)\hat{\mathcal{S}}(z)\| \sum_z \hat{\mathcal{T}}(y|z)\hat{\mathcal{T}}(z)\right)} \\ &\leq \sqrt{\mathbb{E}_{z\sim\hat{\mathcal{S}}(z)}D_{\text{JS}}\left(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)\right)} + \sqrt{D_{\text{JS}}\left(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z)\right)}\end{aligned}$$

We derive the inequality according to (1) Jensen-Shannon distance is a valid statistical metric; (2) The convex property of the Jensen-Shannon divergence w.r.t. the empirical distribution; (3) The f-divergence data-processing inequality.

$$\mathbb{E}_{z\sim\hat{\mathcal{S}}(z)} D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) \geq \left(\sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))} - \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))}\right)^2$$

We can analogously derive:

$$\mathbb{E}_{z\sim\hat{\mathcal{T}}(z)} D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) \geq \left(\sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))} - \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))}\right)^2$$

Finally, the third term can be lower bounded by:

$$\begin{aligned}&\mathbb{E}_{z\sim\hat{\mathcal{T}}(z)} D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z)) + \mathbb{E}_{z\sim\hat{\mathcal{S}}(z)} D_{\text{JS}}(\hat{\mathcal{S}}(y|z)\|\hat{\mathcal{T}}(y|z))\\ &\geq 2\left(\sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{S}}(y))} - \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z))}\right)^2\end{aligned}$$

Which exactly recovers the result of (Zhao et al., 2019a): over-matching the marginal distribution divergence to zero can increase this lower bound of the third term.

C.7 Novel Practice

$$\begin{aligned}R_{\mathcal{T}}(h) \leq & R_{\mathcal{S}}(h) + \frac{G}{\sqrt{2}}\underbrace{\sqrt{D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))}}_{\text{Label Marginal Shift}}\\ & + \frac{G}{\sqrt{2}}\underbrace{\sqrt{\mathbb{E}_{y\sim\mathcal{T}(y)} D_{\text{JS}}(\mathcal{T}(x|y)\|\mathcal{S}(x|y)) + \mathbb{E}_{y\sim\mathcal{S}(y)} D_{\text{JS}}(\mathcal{T}(x|y)\|\mathcal{S}(x|y))}}_{\text{Semantic (Covariate) Conditional Shift}}\end{aligned} \tag{C.10}$$

In this section, we firstly prove the lower bound in the context of conditional distribution matching. We demonstrate that in the presence of conditional distribution matching, we still need to control the label shift term to control a small lower bound.

C.7.1 Necessity of Considering Label Shift (Theorem 4.4)

In this section, we suppose there exists a more general stochastic representation learning function g with a conditional probability distribution $g(z|x)$. [1] Then the marginal distribution and conditional distribution w.r.t. latent variable can be reformulated as:

$$\mathcal{S}(z) = \int_x g(z|x)\mathcal{S}(x)dx \qquad \mathcal{S}(z|y) = \int_x g(z|x)\mathcal{S}(x|Y=y)dx$$

[1] The deterministic representation learning function can be viewed as a special case such that fixed $g(z|x) = z$ for a given x

If $\forall$ classifier h, feature function g, and label $y \in \mathcal{Y} = \{-1, +1\}$ such that semantic conditional distribution is matched: $D_{\text{JS}}(\mathcal{S}(z|y), \mathcal{T}(z|y)) = 0$, then the target risk can be bounded:

$$R_{\mathcal{S}}(h \circ g) - \sqrt{2D_{\text{JS}}(\mathcal{S}(y), \mathcal{T}(y))} \leq R_{\mathcal{T}}(h \circ g) \leq R_{\mathcal{S}}(h \circ g) + \sqrt{2D_{\text{JS}}(\mathcal{S}(y), \mathcal{T}(y))} \quad \text{(C.11)}$$

Where $R_{\mathcal{S}}(h \circ g) = R_{\mathcal{S}}(h(g(x), y))$ the expected risk over the classifier h and feature learner g.

Proof. For simplifying the analysis, we only focus on the binary classification with margin style loss with $L(h(z), y) = L(yh(z))$, including $0 - 1$ loss, hinge loss, logistic loss, etc). Throughout the whole analysis, we will simply adopt the $0 - 1$ loss. We additionally define the following distributions:

$$\begin{aligned}
\mu^{\mathcal{S}}(z) &= \mathcal{S}(Y = 1, Z = z) = \mathcal{S}(Y = 1)\mathcal{S}(Z = z|Y = 1) \\
\pi^{\mathcal{S}}(z) &= \mathcal{S}(Y = -1, Z = z) = \mathcal{S}(Y = -1)\mathcal{S}(Z = z|Y = -1) \\
\mu^{\mathcal{T}}(z) &= \mathcal{T}(Y = 1, Z = z) = \mathcal{T}(Y = 1)\mathcal{T}(Z = z|Y = 1) \\
\pi^{\mathcal{T}}(z) &= \mathcal{T}(Y = -1, Z = z) = \mathcal{T}(Y = -1)\mathcal{T}(Z = z|Y = -1)
\end{aligned}$$

Then in the source distribution and target distribution for the common feature extractor Q and hypothesis h, we have:

$$R_{\mathcal{S}}(h \circ g) = \mathbb{E}_{\mathcal{S}}\mathbf{1}\{yh(z) \leq 0\}$$

$$R_{\mathcal{T}}(h \circ g) = \mathbb{E}_{\mathcal{T}}\mathbf{1}\{yh(z) \leq 0\}$$

According to (Nguyen et al., 2009), the risk can be reformulated as

$$\begin{aligned}
R_{\mathcal{S}}(h \circ g) &= \sum_z \mathbf{1}\{h(z) \leq 0\}\mu^{\mathcal{S}}(z) + \mathbf{1}\{h(z) > 0\}\pi^{\mathcal{S}}(z) \\
R_{\mathcal{T}}(h \circ g) &= \sum_z \mathbf{1}\{h(z) \leq 0\}\mu^{\mathcal{T}}(z) + \mathbf{1}\{h(z) > 0\}\pi^{\mathcal{T}}(z)
\end{aligned}$$

Then we have:

$$\begin{aligned}
R_{\mathcal{T}}(h \circ g) - R_{\mathcal{S}}(h \circ g) &= \sum_z \mathbf{1}\{h(z) \leq 0\}\left(\mu^{\mathcal{T}}(z) - \mu^{\mathcal{S}}(z)\right) + \mathbf{1}\{h(z) > 0\}(\pi^{\mathcal{T}}(z) - \pi^{\mathcal{S}}(z)) \\
&\geq \sum_z \min\{\mu^{\mathcal{T}}(z) - \mu^{\mathcal{S}}(z), \pi^{\mathcal{T}}(z) - \pi^{\mathcal{S}}(z)\}
\end{aligned}$$

If we define the conditional distribution matching as there exists a distribution $\exists g$ such that $\mathcal{S}(z|y) = \mathcal{T}(z|y) = \mathcal{D}(z|y)$, then we can simplify as

$$
\begin{aligned}
&\sum_z \min\{\mu^{\mathcal{T}}(z)-\mu^{\mathcal{S}}(z), \pi^{\mathcal{T}}(z)-\pi^{\mathcal{S}}(z)\}\\
&\geq -|\mathcal{S}(y=1)-\mathcal{T}(y=1)| \sum_z \max\{\mathcal{D}(z|y=1), \mathcal{D}(z|y=-1)\}\\
&= -\frac{1}{2} d_{\text{TV}}(\mathcal{S}(y), \mathcal{T}(y)) \frac{1}{2}(1+d_{\text{TV}}(\mathcal{D}(z|y=1), \mathcal{D}(z|y=-1))\\
&\geq -\frac{1}{2} d_{\text{TV}}(\mathcal{S}(y), \mathcal{T}(y)) \frac{1}{2}(1+1) = -\frac{1}{2} d_{\text{TV}}(\mathcal{S}(y), \mathcal{T}(y)) \geq -\sqrt{2D_{\text{JS}}(\mathcal{S}(y), \mathcal{T}(y))}
\end{aligned}
$$

As for the upper bound, since we have:

$$
\begin{aligned}
R_{\mathcal{T}}(h\circ g) - R_{\mathcal{S}}(h \circ g) &= \sum_z \mathbf{1}\{h(z)\leq 0\}\left(\mu^{\mathcal{T}}(z)-\mu^{\mathcal{S}}(z)\right) + \mathbf{1}\{h(z)>0\}(\pi^{\mathcal{T}}(z)-\pi^{\mathcal{S}}(z))\\
&\leq \sum_z \max\{\mu^{\mathcal{T}}(z)-\mu^{\mathcal{S}}(z), \pi^{\mathcal{T}}(z)-\pi^{\mathcal{S}}(z)\}
\end{aligned}
$$

Given the conditional shift, we have:

$$
\begin{aligned}
&\sum_z \max\{\mu^{\mathcal{T}}(z)-\mu^{\mathcal{S}}(z), \pi^{\mathcal{T}}(z)-\pi^{\mathcal{S}}(z)\}\\
&\leq |\mathcal{S}(y=1)-\mathcal{T}(y=1)| \sum_z \max\{\mathcal{D}(z|y=1), \mathcal{D}(z|y=-1)\}\\
&= \frac{1}{2} d_{\text{TV}}(\mathcal{S}(y), \mathcal{T}(y)) \frac{1}{2}(1+d_{\text{TV}}(\mathcal{D}(z|y=1), \mathcal{D}(z|y=-1)))\\
&\leq \frac{1}{2} d_{\text{TV}}(\mathcal{S}(y), \mathcal{T}(y)) \frac{1}{2}(1+1) = \frac{1}{2} d_{\text{TV}}(\mathcal{S}(y), \mathcal{T}(y)) \leq \sqrt{2D_{\text{JS}}(\mathcal{S}(y), \mathcal{T}(y))}
\end{aligned}
$$

Finally we have the two side bound:

$$
R_{\mathcal{S}}(h\circ g) - \sqrt{2D_{\text{JS}}(\mathcal{S}(y), \mathcal{T}(y))} \leq R_{\mathcal{T}}(h\circ g) \leq R_{\mathcal{S}}(h\circ g) + \sqrt{2D_{\text{JS}}(\mathcal{S}(y), \mathcal{T}(y))}
$$

□

C.7.2 Labeling Shift Correction: Theoretical Result

As our previous theoretical results indicate the necessarily of label shift correction. If the semantic (covariate) conditional distribution is matched $D_{\text{JS}}(\mathcal{T}(z|y)\|\mathcal{S}(z|y)) = 0$, we adopt the popular label re-weighted loss strategy: $\hat{R}^{\alpha}_{\mathcal{S}}(h\circ g) = \sum_{(x_s,y_s)\in\hat{\mathcal{S}}} \alpha(y_s) L(h(g(x_s), y_s))$ with $\alpha(y) = \frac{\mathcal{T}(y)}{\mathcal{S}(y)}$. Then for $\forall h \in \mathcal{H}$, we have:

$$
|\hat{R}^{\alpha}_{\mathcal{S}}(h) - R_{\mathcal{T}}(h)| \leq \mathcal{O}(\sqrt{\frac{D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))}{N_S}})
$$

Proof. According to the Lemma 4 of (Azizzadenesheli et al., 2019), for a given hypothesis class $\mathcal{H}$, under N data points we have:

$$
\sup_{h\in\mathcal{H}} |\hat{R}^{\alpha}_{\mathcal{S}}(h) - R_{\mathcal{T}}(h)| \leq \mathcal{O}(\sqrt{\frac{d_2(\mathcal{T}(y)\|\mathcal{S}(y))\log(2/\delta)}{N}})
$$

with probability at least $1 - \delta$.

Since $d_2(\mathcal{T}(y)\|\mathcal{S}(y)) = 2^{D_2(\mathcal{T}(y)\|\mathcal{S}(y))}$ with $D_2\left(\mathcal{T}(y)\|\mathcal{S}(y)\right)$ is the Rényi-2 divergence. Then according to the (Sason and Verdú, 2015; Thekumparampil et al., 2018), there exists a positive constant C' such that:

$$D_2(\mathcal{T}(y)\|\mathcal{S}(y)) \leq \log(1 + C' d_{TV}(\mathcal{T}(y), \mathcal{S}(y))) \leq \log(1 + C' D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y)))$$

Plugging in the model, we have:

$$\sup_{h\in\mathcal{H}} |\hat{R}^{\alpha}_{\mathcal{S}}(h) - R_{\mathcal{T}}(h)| \leq \mathcal{O}(\sqrt{\frac{D_{\text{JS}}(\mathcal{T}(y)\|\mathcal{S}(y))}{N_S}})$$

□

C.7.3 Detecting Poor Pseudo-Labels

We can prove that if we have a poor pseudo-label, the marginal divergence can be very large. If we assume $D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{T}}_p(y)) = P$, and small source prediction error $D_{\text{JS}}(\hat{\mathcal{S}}(y)\|\hat{\mathcal{S}}_p(y)) \leq \epsilon_1$ and small source target ground truth distribution $D_{\text{JS}}(\hat{\mathcal{S}}(y)\|\hat{\mathcal{T}}(y)) \leq \epsilon_2$, then we can prove

$$D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z)) \geq (\sqrt{P} - \sqrt{\epsilon_1} - \sqrt{\epsilon_2})^2$$

Proof. Since in the DA, we adopt the same classifier h to predict both domains, the empirical label prediction output distribution (pseudo-label distribution) is defined as:

$$\hat{\mathcal{S}}_p(y) = \sum_z h(y|z)\hat{\mathcal{S}}(z) \qquad \hat{\mathcal{T}}_p(y) = \sum_z h(y|z)\hat{\mathcal{T}}(z)$$

According to the f-divergence data-processing inequality, we have:

$$D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z)) \geq D_{\text{JS}}(\hat{\mathcal{S}}_p(y)\|\hat{\mathcal{T}}_p(y))$$

Since Jensen-Shannon distance is a valid statistical distance, then we have:

$$\sqrt{D_{\text{JS}}(\hat{\mathcal{S}}_p(y)\|\hat{\mathcal{T}}_p(y))} + \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}_p(y)\|\hat{\mathcal{S}}(y))} + \sqrt{D_{\text{JS}}(\hat{\mathcal{S}}(y)\|\hat{\mathcal{T}}(y))} \geq \sqrt{D_{\text{JS}}(\hat{\mathcal{T}}(y)\|\hat{\mathcal{T}}_p(y))} = \sqrt{P}$$

Since we have a small source prediction error, a small empirical label shift, then we have:

$$\sqrt{D_{\text{JS}}(\hat{\mathcal{S}}_p(y)\|\hat{\mathcal{T}}_p(y))} \geq \sqrt{P} - \sqrt{\epsilon_1} - \sqrt{\epsilon_2}$$

Combining together we have $D_{\text{JS}}(\hat{\mathcal{S}}(z)\|\hat{\mathcal{T}}(z)) \geq (\sqrt{P} - \sqrt{\epsilon_1} - \sqrt{\epsilon_2})^2$ □

C.8 Practical Guidelines

Parameter Optimization Step (fixed Pseudo-Labels) classifier h and feature extractor g:

$$\min_{h,g} \quad \underbrace{\hat{R}^{\hat{\alpha}}_{\mathcal{S}}(h(g(x_s),y_s))}_{\text{(I)}} + \underbrace{\sum_y (\hat{\mathcal{S}}(y)+\hat{\mathcal{T}}_p(y)) D_{\text{JS}}\left(\hat{\mathcal{T}}(g(x_t)|Y_p=y)\|\hat{\mathcal{S}}(g(x_s)|Y=y)\right)}_{\text{(II)}}$$

$$\text{s.t.} \quad \underbrace{D_{\text{JS}}(\hat{\mathcal{T}}(g(x_t))\|\hat{\mathcal{S}}(g(x_s))) \leq \kappa}_{\text{(III)}}$$

(I) Labeling shift correction:$\hat{R}^{\hat{\alpha}}_{\mathcal{S}}(h(g(x_s),y_s)) = \frac{1}{N_S}\sum_{i=1}^{N_S}\alpha(y_i)L(h(g(x_i),y_i))$; (II) Semantic conditional matching, to align the semantic feature; (III) Covariate marginal distribution matching as the constraint, as the adaptation step to obtain a good initialization pseudo-label prediction.

Pseudo-Label Estimation Step (fixed Parameters): y^p, $\hat{\alpha}$, $\hat{\mathcal{T}}_p(y)$
y^p,$\hat{\mathcal{T}}_p(y)$ are pseudo-labels and distributions on the target domain. $\hat{\alpha}$ label reweighting coefficient.

C.8.1 Semantic Conditional Distribution Matching (Principle II)

As we illustrated in the paper, the first component is to match the covariate conditional distribution divergence. Then we have:

$$\begin{aligned}\sum_y (\hat{\mathcal{S}}(y)+\hat{\mathcal{T}}_p(y)) D_{\text{JS}}(\hat{\mathcal{T}}(\cdot|y)\|\hat{\mathcal{S}}(\cdot|y)) &\leq \sum_y (\hat{\mathcal{S}}(y)+\hat{\mathcal{T}}_p(y)) d_{\text{TV}}(\hat{\mathcal{T}}(\cdot|y)\|\hat{\mathcal{S}}(\cdot|y)) \\ &\leq C\sum_y (\hat{\mathcal{S}}(y)+\hat{\mathcal{T}}_p(y)) \|\hat{\mathcal{T}}(\cdot|y)-\hat{\mathcal{S}}(\cdot|y)\|_2\end{aligned} \tag{C.12}$$

In the representation learning, we simply approximate the empirical distribution as the surrogate of the conditional distribution. We therefore denote:

$$\hat{\mathcal{S}}(g(x_s)|y) \approx \frac{1}{|\#y_s=y|}\sum_{(x_s,y_s)}\delta_{\{y_s=y\}}g(x_s) \qquad \hat{\mathcal{T}}(g(x_t)|y) \approx \frac{1}{|\#y_t^p=y|}\sum_{(x_t,y_t^p)}\delta_{\{y_t^p=y\}}g(x_t)$$

Therefore the conditional matching term can be approximated as:

$$\hat{R}_{\text{cond}}(g) = \sum_y (\hat{\mathcal{S}}(Y=y)+\hat{\mathcal{T}}_p(Y=y))\|\hat{\mathcal{S}}(g(x_s)|Y=y)-\hat{\mathcal{T}}(g(x_t)|Y_p=y)\|_2^2 \tag{C.13}$$

Remark We would like to emphasize that we propose one feasible solution. The covariate conditional distribution matching can be naturally extended to the conditional adversarial training (Long et al., 2018), matching higher statistical moments (Cai et al., 2019) or infinite orders as MMD distance (Long et al., 2015).

C.8.2 Marginal Covariate Distribution Matching as the Constraint (Principle III)

Since a relative accurate pseudo-label estimation is important in the iterative algorithm, we introduce the marginal covariate distribution matching as the training constraint. The main goal is to keep a good pseudo-label initial estimation. We just adopt the most popular Jensen-Shannon domain adversarial training (the dual term of linear shift Jensen-Shannon divergence)

$$\hat{R}_{\text{adv}}(d,g) = \mathbb{E}_{x_s \sim \hat{\mathcal{S}}(x)} \log(d \circ g(x_s)) + \mathbb{E}_{x_t \sim \hat{\mathcal{T}}(x)} \log(1 - d \circ g(x_t)) \tag{C.14}$$

As for the constraints, we adopt Lagrangian relaxation approach and treat the constraint as a small regularization term, where κ is the hyper-parameter.

C.8.3 Labeling Marginal Shift Correction (Principle I)

We adopt the cross entropy as classification loss, then we have:

$$\hat{R}^{\hat{\alpha}}_{\mathcal{S}}(f,g) = -\frac{1}{N_S} \sum_{(x_s,y_s)\sim\hat{\mathcal{S}}} \hat{\alpha}(y_s) \log(h \circ (g(x_s), y_s)) \tag{C.15}$$

Estimation $\hat{\alpha}$ and Target label distribution We follow the popular Black Box Shift Learning (BBSL) estimator. We first construct a source prediction confusion matrix $\hat{C} \in |\mathcal{Y}| \times |\mathcal{Y}|$ with $\hat{C}[i,j] = \mathbb{P}(\text{argmax}_y\ h(g(x_s), y) = i, y_s = j)$. The target pseudo-label y^p and target pseudo-label distribution $\hat{\mathcal{T}}_p$ can be directly estimated from the neural network. Then the label re-weighting coefficient can be estimated as:

$$\hat{\alpha} = \hat{C}^{-1}\hat{\mathcal{T}}_p$$

C.8.4 Practical Loss

We consider the whole aforementioned components and derive the following training strategy.

Parameter Optimization

$$\min_{f,g} \max_{d} \hat{R}(f,d,g) = \hat{R}^{\hat{\alpha}}_{\mathcal{S}}(f,g) + \lambda_0 \hat{R}_{\text{adv}}(d,g) + \lambda_1 \hat{R}_{\text{cond}}(g)$$

Pseudo-Label Estimation

$$\hat{\alpha} = \hat{C}^{-1}\hat{\mathcal{T}}_p$$

The source confusion matrix $\hat{C}$, target pseudo-label y^p and target pseudo-label distribution $\hat{\mathcal{T}}_p$ can be directly estimated from the neural network.

Appendix D

Details of Chapter 5

D.1 Additional Related Work

Multi-source transfer learning Practice has been proposed from various perspective. The key idea is to estimate the importance of different sources and then select the most related ones, to mitigate the influence of negative transfer. In the multi-source unsupervised DA, (Sankaranarayanan et al., 2018; Balaji et al., 2019; Pei et al., 2018; Zhao et al., 2019b; Zhu et al., 2019; Zhao et al., 2020, 2019b; Stojanov et al., 2019; Li et al., 2019b; Wang et al., 2019b; Lin et al., 2020) proposed different practical strategies in the classification, regression and semantic segmentation problems. In the presence of target labels, Hoffman et al. (2012); Tan et al. (2013); Wei et al. (2017); Yao and Doretto (2010); Konstantinov and Lampert (2019) used generalized linear model to learn the target. Christodoulidis et al. (2016); Li et al. (2019a); Chen et al. (2019b) focused on deep learning approaches and Lee et al. (2019) proposed an ad-hoc strategy to combine to sources in the few-shot target domains. These ideas are generally data-driven approaches and do not analyze the why the proposed practice can control the generalization error.

Label-Partial Transfer Learning Label-Partial can be viewed as a special case of the label-shift. [1] Most existing works focus on one-to-one partial transfer learning (Zhang et al., 2018; Chen et al., 2020; Bucci et al., 2019; Cao et al., 2019) by adopting the re-weighting training approach without a formal understanding. In our paper, we first rigorously analyzed this common practice and adopt the label distribution ratio as its weights, which provides a principled approach in this scenario.

D.1.1 Other scenarios related to Multi-Source Transfer Learning

Domain Generalization The domain generalization (DG) resembles multi-source transfer but aims at different goals. A common setting in DG is to learn multiple sources but directly predict on the unseen target domain. The conventional DG approaches generally learn the distribution invariant features (Balaji et al., 2018; Saenko et al., 2010; Motiian et al., 2017; Ilse et al., 2019) or conditional

[1] Since supp($\mathcal{T}(y)$) $\subseteq$ supp($\mathcal{S}_t(y)$) then we naturally have $\mathcal{T}(y) \neq \mathcal{S}_t(y)$.

distribution invariant features (Li et al., 2018c; Akuzawa et al., 2019). However, our theoretical results reveal that in the presence of label shift (i.e $\alpha_t(y) \neq 1$) and outlier tasks then learning conditional or marginal invariant features can not guarantee a small target risk. Our theoretical result enables a formal understanding about the inherent difficulty in DG problems.

Few-Shot Learning The few-shot learning (Finn et al., 2017; Snell et al., 2017; Sung et al., 2018) can be viewed as a very specific scenario of multi-source transfer learning. We would like to point out the differences between the few-shot learning and our paper. (1) Few-shot learning generally involves a **very large set** of source domains $T \gg 1$ and each domain consists of **modest number** of observations $N_{\mathcal{S}_t}$. In our paper, we are interested in the a **modest number** of source domains T but each source domain including a **sufficient large** observations ($N_{\mathcal{S}_t} \gg 1$). (2) In the target domain, the few-shot setting generally used K-samples (K is very small) for each class for the fine-tuning. We would like to point out this setting generally violates our theoretical assumption. In our paper, we assume the target data is i.i.d. sampled from $\mathcal{D}(x, y)$. It is equivalently viewed that we first i.i.d. sample $y \sim \mathcal{D}(y)$, then i.i.d. sample $x \sim \mathcal{D}(x|y)$. Generally the $\mathcal{D}(y)$ is **non-uniform**, thus few-shot settings are generally not applicable in our theoretical assumptions.

Multi-Task Learning The goal of multi-task learning (Zhang and Yang, 2017) is to improve the prediction performance of **all** the tasks. In our paper, we aim at controlling the prediction risk of a specified target domain. We also notice some practical techniques are common such as the shared parameter (Zhang and Yeung, 2012), shared representation (Ruder, 2017), etc.

D.2 Table of Notation

Table D.1 – Table of Notations

$R_{\mathcal{D}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(h(x,y))$	Expected Risk on distribution $\mathcal{D}$ w.r.t. hypothesis h
$\hat{R}_{\mathcal{D}}(h) = \frac{1}{N}\sum_{i=1}^{N}\ell(h(x_i, y_i))$	Empirical Risk on observed data $\{(x_i, y_i)\}_{i=1}^{N}$ that are i.i.d. sampled from $\mathcal{D}$.
α and $\hat{\alpha}_t$	True and empirical label distribution ratio $\alpha(y) = \mathcal{T}(y)/\mathcal{S}(y)$
$\hat{R}_{\mathcal{S}}^{\alpha}(h) = \frac{1}{N}\sum_{i=1}^{N}\alpha(y_i)\ell(h(x_i, y_i))$	Empirical Weighted Risk on observed data $\{(x_i, y_i)\}_{i=1}^{N}$.
$\mathcal{S}(z\|y) = \int_x g(z\|x)S(x\|Y = y)dx$	Conditional distribution w.r.t. latent variable Z that induced by feature learning function g.
$W_1(\mathcal{S}_t(z\|y)\\|\mathcal{T}(z\|y))$	Conditional Wasserstein distance on the latent space Z

D.3 Proof of Theorem 5.1

Proof idea The proof of Theorem 5.1 consists three steps:

Lemma D.1. *If the prediction loss is assumed as L-Lipschitz and the hypothesis is K-Lipschitz w.r.t. the feature x (given the same label), i.e. for $\forall Y = y$, $\|h(x_1, y) - h(x_2, y)\|_2 \leq K\|x_1 - x_2\|_2$. Then the target risk can be upper bounded by:*

$$R_{\mathcal{T}}(h) \leq \sum_t \boldsymbol{\lambda}[t] R_{\mathcal{S}}^{\alpha_t}(h) + LK \sum_t \boldsymbol{\lambda}[t] \mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}(x|Y=y)) \tag{D.1}$$

Proof. The target risk can be expressed as:

$$R_{\mathcal{T}}(h(x,y)) = \mathbb{E}_{(x,y)\sim\mathcal{T}}\ell(h(x,y)) = \mathbb{E}_{y\sim\mathcal{T}(y)}\mathbb{E}_{x\sim\mathcal{T}(x|y)}\ell(h(x,y))$$

By denoting $\alpha(y) = \frac{\mathcal{T}(y)}{\mathcal{S}(y)}$, then we have:

$$\mathbb{E}_{y\sim\mathcal{T}(y)}\mathbb{E}_{y\sim\mathcal{T}(x|y)}\ell(h(x,y)) = \mathbb{E}_{y\sim\mathcal{S}(y)}\alpha(y)\mathbb{E}_{x\sim\mathcal{T}(x|y)}\ell(h(x,y))$$

Then we aim to upper bound $\mathbb{E}_{x\sim\mathcal{T}(x|y)}\ell(h(x,y))$. For any fixed y,

$$\mathbb{E}_{x\sim\mathcal{T}(x|y)}\ell(h(x,y)) - \mathbb{E}_{x\sim\mathcal{S}(x|y)}\ell(h(x,y)) \leq |\int_{x\in\mathcal{X}} \ell(h(x,y))d(\mathcal{T}(x|y) - \mathcal{S}(x|y))|$$

Then according to the Kantorovich-Rubinstein duality, for **any** distribution coupling $\gamma \in \Pi(\mathcal{T}(x|y), \mathcal{S}(x|y))$, then we have:

$$\begin{aligned}
&= \inf_\gamma \left| \int_{\mathcal{X}\times\mathcal{X}} \ell(h(x_p, y)) - \ell(h(x_q, y))d\gamma(x_p, x_q) \right| \\
&\leq \inf_\gamma \int_{\mathcal{X}\times\mathcal{X}} |\ell(h(x_p, y)) - \ell(h(x_q, y))| \, d\gamma(x_p, x_q) \\
&\leq L \inf_\gamma \int_{\mathcal{X}\times\mathcal{X}} |h(x_p, y)) - h(x_q, y)| \, d\gamma(x_p, x_q) \\
&\leq LK \inf_\gamma \int_{\mathcal{X}\times\mathcal{X}} \|x_p - x_q\|_2 d\gamma(x_p, x_q) \\
&= LKW_1(\mathcal{T}(x|Y=y)\|\mathcal{S}(x|Y=y))
\end{aligned}$$

The first inequality is obvious; and the second inequality comes from the assumption that ℓ is L-Lipschitz; the third inequality comes from the hypothesis is K-Lipschitz w.r.t. the feature x (given the same label), i.e. for $\forall Y = y$, $\|h(x_1, y) - h(x_2, y)\|_2 \leq K\|x_1 - x_2\|_2$.

Then we have:

$$\begin{aligned}
R_{\mathcal{T}}(h) &\leq \mathbb{E}_{y\sim\mathcal{S}(y)}\alpha(y)[\mathbb{E}_{x\sim\mathcal{S}(x|y)}\ell(h(x,y)) + LKW_1(\mathcal{T}(x|y)\|\mathcal{S}(x|y))] \\
&= \mathbb{E}_{(x,y)\sim\mathcal{S}}\alpha(y)\ell(h(x,y)) + LK\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}(x|Y=y)) \\
&= R_{\mathcal{S}}^{\alpha}(h) + LK\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}(x|Y=y))
\end{aligned}$$

Supposing each source $\mathcal{S}_t$ we assign the weight $\boldsymbol{\lambda}[t]$ and label distribution ratio $\alpha_t(y) = \frac{\mathcal{T}(y)}{\mathcal{S}_t(y)}$, then by combining this T source target pair, we have:

$$R_{\mathcal{T}}(h) \leq \sum_t \boldsymbol{\lambda}[t] R_{\mathcal{S}_t}^{\alpha_t}(h) + LK \sum_t \boldsymbol{\lambda}[t] \mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}_t(x|Y=y))$$

□

Then we will prove Theorem 1 from this result, we will derive the non-asymptotic bound, estimated from the finite sample observations. Supposing the empirical label ratio value is $\hat{\alpha}_t$, then for any simplex $\boldsymbol{\lambda}$ we can prove the high-probability bound.

D.3.1 Bounding the empirical and expected prediction risk

Proof. We first bound the first term, which can be upper bounded as:

$$\begin{aligned}\sup_h |\sum_t \boldsymbol{\lambda}[t] R^{\alpha_t}_{\mathcal{S}_t}(h) - \sum_t \boldsymbol{\lambda}[t] \hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h)| &\leq \underbrace{\sup_h |\sum_t \boldsymbol{\lambda}[t] R^{\alpha_t}_{\mathcal{S}_t}(h) - \sum_t \boldsymbol{\lambda}[t] \hat{R}^{\alpha_t}_{\mathcal{S}_t}(h)|}_{\text{(I)}} \\ &+ \underbrace{\sup_h |\sum_t \boldsymbol{\lambda}[t] \hat{R}^{\alpha_t}_{\mathcal{S}_t}(h) - \sum_t \boldsymbol{\lambda}[t] \hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h)|}_{\text{(II)}}\end{aligned}$$

Bounding term (I) According to the McDiarmid inequality, each item changes at most $|\frac{2\boldsymbol{\lambda}[t]\alpha_t(y)\ell}{N_{\mathcal{S}_t}}|$. Then we have:

$$P\left((\text{I}) - \mathbb{E}(\text{I}) \geq t\right) \leq \exp\left(\frac{-2t^2}{\sum_{t=1}^T \frac{4}{\beta_t N}\boldsymbol{\lambda}^2[t]\alpha_t(y)^2\ell^2}\right) = \delta$$

By substituting δ, at high probability $1-\delta$ we have:

$$(\text{I}) \leq \mathbb{E}(\text{I}) + L_{\max} d^{\sup}_{\infty}\sqrt{\sum_{t=1}^T \frac{\boldsymbol{\lambda}[t]^2}{\beta_t}}\sqrt{\frac{\log(1/\delta)}{2N}}$$

Where $L_{\max} = \sup_{h\in\mathcal{H}} \ell(h)$ and $N = \sum_{t=1}^T N_{\mathcal{S}_t}$ the total source observations and $\beta_t = \frac{N_{\mathcal{S}_t}}{N}$ the frequency ratio of each source. And $d^{\sup}_{\infty} = \max_{t=1,\dots,T} d_{\infty}(\mathcal{T}(y)\|\mathcal{S}(y)) = \max_{t=1,\dots,T}\max_{y\in[1,\mathcal{Y}]}\alpha_t(y)$, the maximum true label shift value (constant).

Bounding $\mathbb{E}\sup(\text{I})$, the expectation term can be upper bounded as the form of Rademacher Complexity:

$$\begin{aligned}\mathbb{E}(\text{I}) &\leq 2\mathbb{E}_{\sigma}\mathbb{E}_{\hat{\mathcal{S}}_1^T}\sup_h \sum_{t=1}^T \boldsymbol{\lambda}[t] \sum_{(x_t,y_t)\in\hat{\mathcal{S}}_t} \frac{1}{TN}\left(\alpha_t(y)\ell(h(x_t,y_t))\right) \\ &\leq 2\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{\sigma}\mathbb{E}_{\hat{\mathcal{S}}_1^T}\sup_h \sum_{(x_t,y_t)\in\hat{\mathcal{S}}_t} \frac{1}{TN}\left(\alpha_t(y)\ell(h(x_t,y_t))\right) \\ &\leq 2\sup_t \mathbb{E}_{\sigma}\mathbb{E}_{\hat{\mathcal{S}}_t}\sup_h \sum_{(x_t,y_t)\in\hat{\mathcal{S}}_t} \frac{1}{TN}\left[\alpha_t(y)\ell(h(x_t,y_t))\right] \\ &= \sup_t 2\mathcal{R}_t(\ell,\mathcal{H}) = 2\bar{R}(\ell,\mathcal{H})\end{aligned}$$

Where $\bar{R}(\ell,\mathcal{H}) = \sup_t \mathcal{R}_t(\ell,\mathcal{H}) = \sup_t \sup_{h\sim\mathcal{H}} \mathbb{E}_{\hat{\mathcal{S}}_t,\sigma}\sum_{(x_t,y_t)\in\hat{\mathcal{S}}_t}\frac{1}{TN}\left[\alpha_t(y)\ell(h(x_t,y_t))\right]$, represents the Rademacher complexity w.r.t. the prediction loss ℓ, hypothesis h and *true* label distribution ratio α_t.

Therefore with high probability $1-\delta$, we have:

$$\sup_h |\sum_t \boldsymbol{\lambda}[t] R_{\mathcal{S}}^{\alpha_t}(h) - \sum_t \boldsymbol{\lambda}[t] \hat{R}_{\mathcal{S}}^{\alpha_t}(h)| \leq \bar{\mathcal{R}}(\ell, h) + L_{\max} d_{\infty}^{\sup} \sqrt{\sum_{t=1}^{T} \frac{\boldsymbol{\lambda}[t]^2}{\beta_t}} \sqrt{\frac{\log(1/\delta)}{2N}}$$

Bounding Term (II) For all the hypothesis h, we have:

$$\begin{aligned} |\sum_t \boldsymbol{\lambda}[t] \hat{R}_{\mathcal{S}_t}^{\alpha_t}(h) - \sum_t \boldsymbol{\lambda}[t] \hat{R}_{\mathcal{S}_t}^{\hat{\alpha}_t}(h)| &= \left| \sum_t \boldsymbol{\lambda}[t] \frac{1}{N_{\mathcal{S}_t}} \sum_i^{N_{\mathcal{S}_t}} (\alpha(y(i)) - \hat{\alpha}(y(i)))\ell(h) \right| \\ &= \sum_t \boldsymbol{\lambda}[t] \frac{1}{N_{\mathcal{S}_t}} \left| \sum_y^{|\mathcal{Y}|} (\alpha(Y=y) - \hat{\alpha}(Y=y)) \bar{\ell}(Y=y) \right| \end{aligned}$$

Where $\bar{\ell}(Y=y) = \sum_i^{N_{\mathcal{S}_t}} \ell(h(x_i, y_i = y))$, represents the cumulative error, conditioned on a given label $Y=y$. According to the Holder inequality, we have:

$$\begin{aligned} \sum_t \boldsymbol{\lambda}[t] \frac{1}{N_{\mathcal{S}_t}} |\sum_y^{|\mathcal{Y}|} (\alpha_t(Y=y) - \hat{\alpha}_t(Y=y)) \bar{\ell}(Y=y)| &\leq \sum_t \boldsymbol{\lambda}[t] \frac{1}{N_{\mathcal{S}_t}} \|\alpha_t - \hat{\alpha}_t\|_2 \|\bar{\ell}(Y=y)\|_2 \\ &\leq L_{\max} \sum_t \boldsymbol{\lambda}[t] \|\alpha_t - \hat{\alpha}_t\|_2 \\ &\leq L_{\max} \sup_t \|\alpha_t - \hat{\alpha}_t\|_2 \end{aligned}$$

Therefore, $\forall h \in \mathcal{H}$, with high probability $1-\delta$ we have:

$$\sum_t \boldsymbol{\lambda}[t] R_{\mathcal{S}}^{\alpha_t}(h) \leq \sum_t \boldsymbol{\lambda}[t] \hat{R}_{\mathcal{S}}^{\hat{\alpha}_t}(h) + 2\bar{\mathcal{R}}(\ell, h) + L_{\max} d_{\infty}^{\sup} \sqrt{\sum_{t=1}^{T} \frac{\boldsymbol{\lambda}[t]^2}{\beta_t}} \sqrt{\frac{\log(1/\delta)}{2N}} + L_{\max} \sup_t \|\alpha_t - \hat{\alpha}_t\|_2$$

D.3.2 Bounding empirical Wasserstein Distance

Then we need to derive the sample complexity of the empirical and true distributions, which can be decomposed as the following two parts. For any t, we have:

$$\begin{aligned} &\mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\mathcal{T}(x|Y=y) \| \mathcal{S}_t(x|Y=y)) - \mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{T}}(x|Y=y) \| \hat{\mathcal{S}}_t(x|Y=y)) \\ &\underbrace{\leq \mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\mathcal{T}(x|Y=y) \| \mathcal{S}_t(x|Y=y)) - \mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\hat{\mathcal{T}}(x|Y=y) \| \hat{\mathcal{S}}_t(x|Y=y))}_{\text{(I)}} \\ &+ \underbrace{\mathbb{E}_{y\sim\mathcal{T}(y)} W_1(\hat{\mathcal{T}}(x|Y=y) \| \hat{\mathcal{S}}_t(x|Y=y)) - \mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{T}}(x|Y=y) \| \hat{\mathcal{S}}_t(x|Y=y))}_{\text{(II)}} \end{aligned}$$

Bounding (I) We have:

$$\begin{aligned}
&\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}_t(x|Y=y)) - \mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))\\
&=\sum_y \mathcal{T}(y)\Big(W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}_t(x|Y=y)) - W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)\Big)\\
&\leq |\sum_y \mathcal{T}(y)|\sup_y\Big(W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}_t(x|Y=y)) - W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)\Big)\\
&=\sup_y\Big(W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}_t(x|Y=y)) - W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)\Big)\\
&\leq \sup_y\,[W_1(\mathcal{S}_t(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)) + W_1(\hat{\mathcal{S}}_t(x|Y=y)\|\hat{\mathcal{T}}(x|Y=y))\\
&+W_1(\hat{\mathcal{T}}(x|Y=y)\|\mathcal{T}(x|Y=y)) - W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))]\\
&=\sup_y W_1(\mathcal{S}_t(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)) + W_1(\hat{\mathcal{T}}(x|Y=y)\|\mathcal{T}(x|Y=y))
\end{aligned}$$

The first inequality holds because of the Holder inequality. As for the second inequality, we use the triangle inequality of Wasserstein distance. $W_1(P\|Q)\leq W_1(P\|P_1)+W_1(P_1\|P_2)+W_1(P_2\|Q)$.

According to the convergence behavior of Wasserstein distance (Weed et al., 2019), with high probability $\geq 1-2\delta$ we have:

$$W_1(\mathcal{S}_t(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)) + W_1(\hat{\mathcal{T}}(x|Y=y)\|\mathcal{T}(x|Y=y)) \leq \kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}})$$

Where $k(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}}) = C_{t,y}(N^y_{\mathcal{S}_t})^{-s_{t,y}} + C_y(N^y_{\mathcal{T}})^{-s_y} + \sqrt{\frac{1}{2}\log(\frac{2}{\delta})}(\sqrt{\frac{1}{N^y_{\mathcal{S}_t}}}+\sqrt{\frac{1}{N^y_t}})$, where $N^y_{\mathcal{S}_t}$ is the number of $Y=y$ in source t and $N^y_{\mathcal{T}}$ is the number of $Y=y$ in target distribution. $C_{t,y}$, C_y $s_{t,y}>2$, $s_y>2$ are positive constant in the concentration inequality. This indicates the convergence behavior between empirical and true Wasserstein distance.

If we adopt the union bound (over all the labels) by setting $\delta\leftarrow\delta/|\mathcal{Y}|$, then with high probability $\geq 1-2\delta$, we have:

$$\sup_y W_1(\mathcal{S}(x|Y=y)\|\hat{\mathcal{S}}(x|Y=y)) + W_1(\hat{\mathcal{T}}(x|Y=y)\|\mathcal{T}(x|Y=y)) \leq \kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}})$$

where $\kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}}) = C_{t,y}(N^y_{\mathcal{S}_t})^{-s_{t,y}} + C_y(N^y_{\mathcal{T}})^{-s_y} + \sqrt{\frac{1}{2}\log(\frac{2|\mathcal{Y}|}{\delta})}(\sqrt{\frac{1}{N^y_{\mathcal{S}_t}}}+\sqrt{\frac{1}{N^y_{\mathcal{T}}}})$

Again by adopting the union bound (over all the tasks) by setting $\delta\leftarrow\delta/T$, with high probability $\geq 1-2\delta$, we have:

$$\begin{aligned}
&\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{T}(x|Y=y)\|\mathcal{S}(x|Y=y)) - \sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}(x|Y=y))\\
&\leq \sup_t \kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}})
\end{aligned}$$

Where $\kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}}) = C_{t,y}(N^y_{\mathcal{S}_t})^{-s_{t,y}} + C_y(N^y_{\mathcal{T}})^{-s_y} + \sqrt{\frac{1}{2}\log(\frac{2T|\mathcal{Y}|}{\delta})}(\sqrt{\frac{1}{N^y_{\mathcal{S}_t}}}+\sqrt{\frac{1}{N^y_{\mathcal{T}}}})$.

Bounding (II) We can bound the second term:

$$\begin{aligned}
&\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)) - \mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))\\
&\leq \sup_y W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))|\sum_y \mathcal{T}(y)-\hat{\mathcal{T}}(y)|\\
&\leq C^t_{\text{max}}|\sum_y \mathcal{T}(y)-\hat{\mathcal{T}}(y)|
\end{aligned}$$

Where $C^t_{\text{max}} = \sup_y W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}(x|Y=y))$ is a positive and bounded constant. Then we need to bound $|\sum_y \mathcal{T}(y)-\hat{\mathcal{T}}(y)|$, by adopting MicDiarmid's inequality, we have at high probability $1-\delta$:

$$\begin{aligned}
|\sum_y \mathcal{T}(y)-\hat{\mathcal{T}}(y)| &\leq \mathbb{E}_{\hat{\mathcal{T}}}|\sum_y \mathcal{T}(y)-\hat{\mathcal{T}}(y)| + \sqrt{\frac{\log(1/\delta)}{2N_{\mathcal{T}}}}\\
&= 2\mathbb{E}_{\sigma}\mathbb{E}_{\hat{\mathcal{T}}}\sum_y \sigma\hat{\mathcal{T}}(y) + \sqrt{\frac{\log(1/\delta)}{2N_{\mathcal{T}}}}
\end{aligned}$$

Then we bound $\mathbb{E}_{\sigma}\mathbb{E}_{\hat{\mathcal{T}}}\sum_y \sigma\hat{\mathcal{T}}(y)$. We use the properties of Rademacher complexity [Lemma 26.11, (Shalev-Shwartz and Ben-David, 2014)] and notice that $\hat{\mathcal{T}}(y)$ is a probability simplex, then we have:

$$\mathbb{E}_{\sigma}\mathbb{E}_{\hat{\mathcal{T}}}\sum_y \sigma\hat{\mathcal{T}}(y) \leq \sqrt{\frac{2\log(2|\mathcal{Y}|)}{N_{\mathcal{T}}}}$$

Then we have $|\sum_y \mathcal{T}(y)-\hat{\mathcal{T}}(y)| \leq \sqrt{\frac{2\log(2|\mathcal{Y}|)}{N_{\mathcal{T}}}} + \sqrt{\frac{\log(1/\delta)}{2N_{\mathcal{T}}}}$

Then using the union bound and denoting $\delta \leftarrow \delta/T$, with high probability $\geq 1-\delta$ and for any simplex $\boldsymbol{\lambda}$, we have:

$$\begin{aligned}
\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y)) \leq \sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}_t(x|Y=y))\\
C_{\text{max}}(\sqrt{\frac{2\log(2|\mathcal{Y}|)}{N_{\mathcal{T}}}} + \sqrt{\frac{\log(T/\delta)}{2N_{\mathcal{T}}}})
\end{aligned}$$

where $C_{\text{max}} = \sup_t C^t_{\text{max}}$.

Combining together, we can derive the PAC-Learning bound, which is estimated from the finite samples (with high probability $1-4\delta$):

$$\begin{aligned}
R_{\mathcal{T}}(h) \leq &\sum_t \boldsymbol{\lambda}_t \hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h) + LH\sum_t \boldsymbol{\lambda}_t\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}W_1(\hat{\mathcal{T}}(x|Y=y)\|\hat{\mathcal{S}}(x|Y=y))\\
&+ L_{\text{max}}d^{\text{sup}}_{\infty}\sqrt{\sum_{t=1}^{T}\frac{\boldsymbol{\lambda}_t^2}{\beta_t}}\sqrt{\frac{\log(1/\delta)}{2N}} + 2\bar{\mathcal{R}}(\ell,h) + L_{\text{max}}\sup_t\|\alpha_t-\hat{\alpha}_t\|_2\\
&+ \sup_t \kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}}) + C_{\text{max}}(\sqrt{\frac{2\log(2|\mathcal{Y}|)}{N_{\mathcal{T}}}} + \sqrt{\frac{\log(T/\delta)}{2N_{\mathcal{T}}}})
\end{aligned}$$

Then we denote $\text{Comp}(N_{\mathcal{S}_1}, \dots, N_{\mathcal{T}}, \delta) = 2\bar{\mathcal{R}}(\ell, h) + \sup_t \kappa(\delta, N^y_{\mathcal{S}_t}, N^y_{\mathcal{T}}) + C_{\max}(\sqrt{\frac{2\log(2|\mathcal{Y}|)}{N_{\mathcal{T}}}} + \sqrt{\frac{\log(T/\delta)}{2N_{\mathcal{T}}}})$ as the convergence rate function that decreases with larger $N_{\mathcal{S}_1}, \dots, N_{\mathcal{T}}$. Bedsides, $\bar{\mathcal{R}}(\ell, h) = \sup_t \mathcal{R}_t(\ell, \mathcal{H})$ is the re-weighted Rademacher complexity. Given a fixed hypothesis with finite VC dimension, it can be proved $\bar{\mathcal{R}}(\ell, h) = \min_{N_{\mathcal{S}_1}, \dots, N_{\mathcal{S}_T}} \mathcal{O}(\sqrt{\frac{1}{N_{\mathcal{S}_t}}})$ i.e (Shalev-Shwartz and Ben-David, 2014). □

D.4 Proof of Theorem 5.2

We first recall the stochastic feature representation g such that $g : \mathcal{X} \to \mathcal{Z}$ and *scoring hypothesis* h $h : \mathcal{Z} \times \mathcal{Y} \to \mathbb{R}$ and the prediction loss ℓ with $\ell : \mathbb{R} \to \mathbb{R}$. Note this definition is different from the conventional binary classification with binary output, and it is more suitable in the multi-classification scenario and cross entropy loss (Hoffman et al., 2018). For example, if we define $l = -\log(\cdot)$ and $h(z, y) \in (0, 1)$ as a scalar score output. Then $\ell(h(z, y))$ can be viewed as the cross-entropy loss for the neural-network.

Proof. The marginal distribution and conditional distribution w.r.t. latent variable Z that are induced by g, which can be reformulated as:

$$\mathcal{S}(z) = \int_x g(z|x)\mathcal{S}(x)dx \qquad \mathcal{S}(z|y) = \int_x g(z|x)\mathcal{S}(x|Y = y)dx$$

In the multi-class classification problem, we additionally define the following distributions:

$$\mu^k(z) = \mathcal{S}(Y = k, z) = \mathcal{S}(Y = k)\mathcal{S}(z|Y = k)$$
$$\pi^k(z) = \mathcal{T}(Y = k, z) = \mathcal{T}(Y = k)\mathcal{T}(z|Y = k)$$

Based on (Nguyen et al., 2009) and $g(z|x)$ is a stochastic representation learning function, the loss conditioned a fixed point (x, y) w.r.t. h and g is $\mathbb{E}_{z\sim g(z|x)}\ell(h(z, y))$.

An alternative understanding the loss is based on the Markov chain. In this case it is a DAG with

$$Y \xleftarrow{\mathcal{S}(y|x)} X \xrightarrow{g} Z, X \xrightarrow{\mathcal{S}(y|x)} Y \xrightarrow{h} S \xleftarrow{h} Z \xleftarrow{g} X$$

where S is the output of the scoring function. Then the expected loss over the all random variable can be equivalently written as

$$\begin{aligned}\int \mathbb{P}(x, y, z, s)\ \ell(s)\ d(x, y, z, s) &= \int \mathbb{P}(x)\mathbb{P}(y|x)\mathbb{P}(z|x)\mathbb{P}(s|z, y)\ell(s)\ d(x, y, z, s)\\ &= \int \mathbb{P}(x, y)\mathbb{P}(z|x)\mathbb{P}(s|z, y)\ell(s)d(x, y)d(z)d(s)\end{aligned}$$

Since the scoring S is determined by $h(x, y)$, then $\mathbb{P}(s|y, z) = 1$. According to the definition we have $\mathbb{P}(z|x) = g(z|x)$, $\mathbb{P}(x, y) = \mathcal{S}(x, y)$, then the loss can be finally expressed as $\mathbb{E}_{\mathcal{S}(x,y)}\mathbb{E}_{g(z|x)}\ell(h(z, y))$.

Then taking the expectation over the $\mathcal{S}(x,y)$ we have:

$$\begin{aligned}
R_{\mathcal{S}}(h,g) &= \mathbb{E}_{(x,y)\sim\mathcal{S}(x,y)}\mathbb{E}_{z\sim g(z|x)}\ell(h(z,y)) \\
&= \sum_{k=1}^{|\mathcal{Y}|}\mathcal{S}(y=k)\int_x \mathcal{S}(x|Y=k)\int_z g(z|x)\ell(h(z,y=k))dzdx \\
&= \sum_{k=1}^{|\mathcal{Y}|}\mathcal{S}(y=k)\int_z [\int_x \mathcal{S}(x|Y=k)g(z|x)dx]\ell(h(z,y=k))dz \\
&= \sum_{k=1}^{|\mathcal{Y}|}\mathcal{S}(y=k)\int_z \mathcal{S}(z|Y=k)\ell(h(z,y=k))dz \\
&= \sum_{k=1}^{|\mathcal{Y}|}\int_z \mathcal{S}(z,Y=k)\ell(h(z,y=k))dz \\
&= \sum_{k=1}^{|\mathcal{Y}|}\int_z \mu^k(z)\ell(h(z,y=k))dz
\end{aligned}$$

Intuitively, the expected loss w.r.t. the joint distribution $\mathcal{S}$ can be decomposed as the expected loss on the label distribution $\mathcal{S}(y)$ (weighted by the labels) and conditional distribution $\mathcal{S}(\cdot|y)$ (real valued conditional loss).

Then the expected risk on the $\mathcal{S}$ and $\mathcal{T}$ can be expressed as:

$$\begin{aligned}
R_{\mathcal{S}}(h,g) &= \sum_{k=1}^{|\mathcal{Y}|}\int_z \ell(h(z,y=k))\mu^k(z)dz \\
R_{\mathcal{T}}(h,g) &= \sum_{k=1}^{|\mathcal{Y}|}\int_z \ell(h(z,y=k))\pi^k(z)dz
\end{aligned}$$

By denoting $\alpha(y) = \frac{\mathcal{T}(y)}{\mathcal{S}(y)}$, we have the α-weighted loss:

$$\begin{aligned}
R^{\alpha}_{\mathcal{S}}(h,g) =&\mathcal{T}(Y=1)\int_z \ell(h(z,y=1))\mathcal{S}(z|Y=1) + \mathcal{T}(Y=2)\int_z \ell(h(z,y=2))\mathcal{S}(z|Y=2) \\
&+ \cdots + \mathcal{T}(Y=k)\int_z \ell(h(z,y=k))\mathcal{S}(z|Y=k)dz
\end{aligned}$$

Then we have:

$$R_{\mathcal{T}}(h,g) - R^{\alpha}_{\mathcal{S}}(h,g) \leq \sum_k \mathcal{T}(Y=k)\int_z \ell(h(z,y=k))d|\mathcal{S}(z|Y=k) - \mathcal{T}(z|Y=k)|$$

Under the same assumption, we have the loss function $\ell(h(z,Y=k))$ is KL-Lipschitz w.r.t. the cost $\|\cdot\|_2$ (given a fixed k). Therefore by adopting the same proof strategy (Kantorovich-Rubinstein duality) in Lemma 2, we have

$$\begin{aligned}
&\leq KL\mathcal{T}(Y=1)W_1(\mathcal{S}(z|Y=1)\|\mathcal{T}(z|Y=1)) + \cdots + KL\mathcal{T}(Y=k)W_1(\mathcal{S}(z|Y=k)\|\mathcal{T}(z|Y=k)) \\
&= KL\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{S}(z|Y=y)\|\mathcal{T}(z|Y=y))
\end{aligned}$$

Therefore, we have:

$$R_{\mathcal{T}}(h,g) \leq R^{\alpha}_{\mathcal{S}}(h,g) + LK\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{S}(z|Y=y)\|\mathcal{T}(z|Y=y))$$

Based on the aforementioned result, we have $\forall t = 1,\dots,T$ and denote $\mathcal{S}=\mathcal{S}_t$ and $\alpha(y)=\alpha_t(y)=\mathcal{T}(y)/\mathcal{S}_t(y)$:

$$\boldsymbol{\lambda}[t]R_{\mathcal{T}}(h,g) \leq \boldsymbol{\lambda}[t]R^{\alpha_t}_{\mathcal{S}_t}(h,g) + LK\boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{S}_t(z|Y=y)\|\mathcal{T}(z|Y=y))$$

Summing over $t=1,\dots,T$, we have:

$$R_{\mathcal{T}}(h,g) \leq \sum_{t=1}^{T}\boldsymbol{\lambda}[t]R^{\alpha_t}_{\mathcal{S}_t}(h,g) + LK\sum_{t=1}^{T}\boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\mathcal{T}(y)}W_1(\mathcal{S}_t(z|Y=y)\|\mathcal{T}(z|Y=y))$$

□

D.5 Approximating W_1 distance

According to Jensen inequality, we have

$$W_1(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y)) \leq \sqrt{[W_2(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y))]^2}$$

Supposing $\hat{\mathcal{S}}_t(z|Y=y)\approx\mathcal{N}(\mathbf{C}^y_t,\boldsymbol{\Sigma})$ and $\hat{\mathcal{T}}(z|Y=y)\approx\mathcal{N}(\mathbf{C}^y,\boldsymbol{\Sigma})$, then we have:

$$[W_2(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y)]^2 = \|\mathbf{C}^y_t-\mathbf{C}^y\|_2^2 + \text{Trace}(2\boldsymbol{\Sigma}-2(\boldsymbol{\Sigma}\boldsymbol{\Sigma})^{1/2}) = \|\mathbf{C}^y_t-\mathbf{C}^y\|_2^2$$

We would like to point out that assuming the identical covariance matrix is more computationally efficient during the matching. This is advantageous and reasonable in the deep learning regime: we adopted the mini-batch (ranging from 20-128) for the neural network parameter optimization, in each mini-batch the samples of each class are **small**, then we compute the empirical covariance/variance matrix will be surely **biased** to the ground truth variance and induce a much higher complexity to optimize. By the contrary, the empirical mean is **unbiased** and computationally efficient, we can simply use the moving the moving average to efficiently update the estimated mean value (with a unbiased estimator). The empirical results verify the effectiveness of this idea.

D.6 Proof of Lemma 5.2

For each source $\mathcal{S}_t$, by introducing the duality of Wasserstein-1 distance, for $y\in\mathcal{Y}$, we have:

$$\begin{aligned} W_1(\mathcal{S}_t(z|y)\|\mathcal{T}(z|y)) &= \sup_{\|d\|_L\leq 1}\mathbb{E}_{z\sim\mathcal{S}_t(z|y)}d(z) - \mathbb{E}_{z\sim\mathcal{T}(z|y)}d(z) \\ &= \sup_{\|d\|_L\leq 1}\sum_z \mathcal{S}_t(z|y)d(z) - \sum_z \mathcal{T}(z|y)d(z) \\ &= \frac{1}{\mathcal{T}(y)}\sup_{\|d\|_L\leq 1}\frac{\mathcal{T}(y)}{\mathcal{S}_t(y)}\sum_z \mathcal{S}_t(z,y)d(z) - \sum_z \mathcal{T}(z,y)d(z) \end{aligned}$$

Then by defining $\bar{\alpha}_t(z) = \mathbf{1}_{\{(z,y)\sim\mathcal{S}_t\}}\frac{\mathcal{T}(Y=y)}{\mathcal{S}_t(Y=y)} = \mathbf{1}_{\{(z,y)\sim\mathcal{S}_t\}}\alpha_t(Y=y)$, we can see for each pair observation (z,y) sampled from the same distribution, then $\bar{\alpha}_t(Z=z) = \alpha_t(Y=y)$. Then we have:

$$\begin{aligned}\sum_y \mathcal{T}(y)W_1(\mathcal{S}_t(z|y)\|\mathcal{T}(z|y)) &= \sum_y \sup_{\|d\|_L\leq 1}\{\sum_z \alpha_t(y)\mathcal{S}_t(z,y)d(z) - \sum_z \mathcal{T}(z,y)d(z)\}\\ &= \sup_{\|d\|_L\leq 1}\sum_z \bar{\alpha}_t(z)\mathcal{S}_t(z)d(z) - \sum_z \mathcal{T}(z)d(z)\\ &= \sup_{\|d\|_L\leq 1}\mathbb{E}_{z\sim\mathcal{S}_t(z)}\bar{\alpha}_t(z)d(z) - \mathbb{E}_{z\sim\mathcal{T}(z)}d(z)\end{aligned}$$

We propose a simple example to understand $\bar{\alpha}_t$: supposing three samples in $\mathcal{S}_t = \{(z_1, Y=1), (z_2, Y=1), (z_3, Y=0)\}$ then $\bar{\alpha}_t(z_1) = \bar{\alpha}_t(z_2) = \alpha_t(1)$ and $\bar{\alpha}_t(z_3) = \alpha_t(0)$. Therefore, the conditional term is equivalent to the label-weighted Wasserstein adversarial learning. We plug in each source domain as weight $\boldsymbol{\lambda}[t]$ and domain discriminator as d_t, we finally have Lemma 1.

D.7 Derive the label ratio loss

We suppose the representation learning aims at matching the conditional distribution such that $\mathcal{T}(z|y) \approx \mathcal{S}_t(z|y)$,$\forall t$, then we suppose the predicted target distribution as $\bar{\mathcal{T}}(y)$. By simplifying the notation, we define $f(z) = \text{argmax}_y h(z,y)$ the most possible prediction label output, then we have:

$$\begin{aligned}\bar{\mathcal{T}}(y) &= \sum_{k=1}^{\mathcal{Y}} \mathcal{T}(f(z)=y|Y=k)\mathcal{T}(Y=k) = \sum_{k=1}^{\mathcal{Y}} \mathcal{S}_t(f(z)=y|Y=k)\mathcal{T}(Y=k)\\ &= \sum_{k=1}^{\mathcal{Y}} \mathcal{S}_t(f(z)=y, Y=k)\alpha_t(k) = \bar{\mathcal{T}}_{\alpha_t}(y)\end{aligned}$$

The first equality comes from the definition of target label prediction distribution, $\bar{\mathcal{T}}(y) = \mathbb{E}_{\mathcal{T}(z)}\mathbf{1}\{f(z) = y\} = \mathcal{T}(f(z)=y) = \sum_{k=1}^{\mathcal{Y}}\mathcal{T}(f(z)=y, Y=k) = \sum_{k=1}^{\mathcal{Y}}\mathcal{T}(f(z)=y|Y=k)\mathcal{T}(Y=k)$.

The second equality $\mathcal{T}(f(z)=y|Y=k) = \mathcal{S}_t(f(z)=y|Y=k)$ holds since $\forall t$, $\mathcal{T}(z|y) \approx \mathcal{S}_t(z|y)$, then for the shared hypothesis f, we have $\mathcal{T}(f(z)=y|Y=k) = \mathcal{S}_t(f(z)=y|Y=k)$.

The term $\mathcal{S}_t(f(z)=y, Y=k)$ is the (expected) source prediction confusion matrix, and we denote its empirical (observed) version as $\hat{\mathcal{S}}_t(f(z)=y, Y=k)$.

Based on this idea, in practice we want to find a $\hat{\alpha}_t$ to match the two predicted distribution $\bar{\mathcal{T}}$ and $\bar{\mathcal{T}}_{\hat{\alpha}_t}$. If we adopt the KL-divergence as the metric, we have:

$$\begin{aligned}\min_{\hat{\alpha}_t} D_{\text{KL}}(\bar{\mathcal{T}}\|\bar{\mathcal{T}}_{\hat{\alpha}_t}) = \min_{\hat{\alpha}_t}\mathbb{E}_{y\sim\bar{\mathcal{T}}}\log(\frac{\bar{\mathcal{T}}(y)}{\bar{\mathcal{T}}_{\hat{\alpha}_t}(y)}) &= \min_{\hat{\alpha}_t} -\mathbb{E}_{y\sim\bar{\mathcal{T}}}\log(\bar{\mathcal{T}}_{\hat{\alpha}_t}(y))\\ &= \min_{\hat{\alpha}_t} -\sum_y \bar{\mathcal{T}}(y)\log(\sum_{k=1}^{\mathcal{Y}}\mathcal{S}_t(f(z)=y, Y=k)\hat{\alpha}_t(k))\end{aligned}$$

We should notice the nature constraints of label ratio: $\{\hat{\alpha}_t(y) \geq 0, \sum_y \hat{\alpha}_t(y)\hat{\mathcal{S}}_t(y) = 1\}$. Based on this principle, we proposed the optimization problem to estimate each label ratio. We adopt its empirical counterpart, the empirical confusion matrix $C_{\hat{\mathcal{S}}_t}[y,k] = \hat{\mathcal{S}}_t[f(z) = y, Y = k]$, then the optimization loss can be expressed as:

$$\begin{aligned} \min_{\hat{\alpha}_t} \quad & -\sum_{y=1}^{|\mathcal{Y}|} \bar{\mathcal{T}}(y) \log(\sum_{k=1}^{|\mathcal{Y}|} C_{\hat{\mathcal{S}}_t}[y,k]\hat{\alpha}_t(k)) \\ \text{s.t.} \quad & \forall y \in \mathcal{Y}, \hat{\alpha}_t(y) \geq 0, \quad \sum_y \hat{\alpha}_t(y)\hat{\mathcal{S}}_t(y) = 1 \end{aligned}$$

D.8 Label Partial Multi-source unsupervised DA

The key difference between multi-conventional and partial unsupervised DA is the estimation step of $\hat{\alpha}_t$. In fact, we only add a sparse constraint for estimating each $\hat{\alpha}_t$:

$$\begin{aligned} \min_{\hat{\alpha}_t} \quad & -\sum_{y=1}^{|\mathcal{Y}|} \bar{\mathcal{T}}(y) \log(\sum_{k=1}^{|\mathcal{Y}|} C_{\hat{\mathcal{S}}_t}[y,k]\hat{\alpha}_t(k)) + C_2\|\hat{\alpha}_t\|_1 \\ \text{s.t.} \quad & \forall y \in \mathcal{Y}, \hat{\alpha}_t(y) \geq 0, \quad \sum_y \hat{\alpha}_t(y)\hat{\mathcal{S}}_t(y) = 1 \end{aligned} \tag{D.2}$$

Where C_2 is the hyper-parameter to control the level of target label sparsity, to estimate the target label distribution. In the paper, we denote $C_2 = 0.1$.

D.9 Explicit and Implicit conditional learning

Inspired by Theorem 2, we need to learn the functions $g : \mathcal{X} \to \mathcal{Z}$ and $h : \mathcal{Z} \times \mathcal{Y} \to \mathbb{R}$ to minimize:

$$\min_{g,h} \sum_t \boldsymbol{\lambda}[t]\hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h,g) + C_0 \sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y))$$

This can be equivalently expressed as:

$$\begin{aligned} \min_{g,h} \sum_t \boldsymbol{\lambda}[t]\hat{R}^{\alpha_t}_{\mathcal{S}_t}(h,g) &+ \epsilon C_0 \sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y)) \\ &+ (1-\epsilon) C_0 \sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)} W_1(\hat{\mathcal{S}}_t(z|Y=y)\|\hat{\mathcal{T}}(z|Y=y)) \end{aligned}$$

Due to the explicit and implicit approximation of conditional distance, we then optimize an alternative form:

$$\begin{aligned} \min_{g,h} \max_{d_1,\ldots,d_T} \underbrace{\sum_t \boldsymbol{\lambda}[t]\hat{R}^{\hat{\alpha}_t}_{\mathcal{S}_t}(h,g)}_{\text{Classification Loss}} &+ \epsilon C_0 \underbrace{\sum_t \boldsymbol{\lambda}[t]\mathbb{E}_{y\sim\hat{\mathcal{T}}(y)}\|\mathbf{C}_t^y - \mathbf{C}^y\|_2}_{\text{Explicit Conditional Loss}} \\ &+ (1-\epsilon) C_0 \underbrace{\sum_t \boldsymbol{\lambda}[t][\mathbb{E}_{z\sim\hat{\mathcal{S}}_t(z)}\bar{\alpha}^t(z)d(z) - \mathbb{E}_{z\sim\hat{\mathcal{T}}(z)}d(z)]}_{\text{Implicit Conditional Loss}} \end{aligned} \tag{D.3}$$

Where

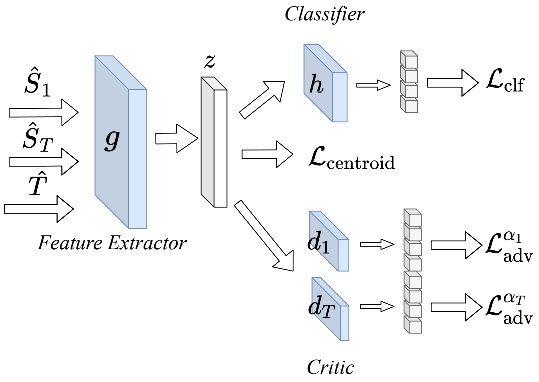

Classifier
$\hat{S}_1$
$\hat{S}_T$
$\hat{T}$
g
Feature Extractor
z
h
$\mathcal{L}_{\text{clf}}$
$\mathcal{L}_{\text{centroid}}$
d_1
d_T
$\mathcal{L}_{\text{adv}}^{\alpha_1}$
$\mathcal{L}_{\text{adv}}^{\alpha_T}$
Critic

Figure D.1 – Network Structure of Proposed Approach. It consists three losses: the weighted Classification losses; the centroid matching for explicit conditional matching; the weighted adversarial loss for implicit conditional matching, showed in Eq. (D.3)

- $\mathbf{C}_t^y = \sum_{(z_t,y_t)\sim\hat{\mathcal{S}}_t} \mathbf{1}_{\{y_t=y\}} z_t$ the centroid of label $Y = y$ in source $\mathcal{S}_t$.
- $\mathbf{C}^y = \sum_{(z_t,y_p)\sim\hat{\mathcal{T}}} \mathbf{1}_{\{y_p=y\}} z_t$ the centroid of pseudo-label $Y = y_p$ in target $\mathcal{S}_t$. (If it is the unsupervised DA scenarios).
- $\bar{\alpha}_t(z) = \mathbf{1}_{\{(z,y)\sim\mathcal{S}_t\}} \hat{\alpha}_t(Y = y)$, namely if each pair observation (z, y) from the distribution, then $\bar{\alpha}_t(Z = z) = \hat{\alpha}_t(Y = y)$.
- $d_1, \cdots, d_T$ are domain discriminator (or critic function) restricted within 1-Lipschitz function.
- $\epsilon \in [0, 1]$ is the adjustment parameter in the trade-off of explicit and implicit learning. Based on the equivalence form, our approach proposed a theoretical principled way to tuning its weights. In the paper, we assume $\epsilon = 0.5$.
- $\hat{\mathcal{T}}(y)$ empirical target label distribution. (In the unsupervised DA scenarios, we approximate it by predicted target label distribution $\bar{\mathcal{T}}(y)$.)

Gradient Penalty In order to enforce the Lipschitz property of the statistic critic function, we adopt the gradient penalty term (Gulrajani et al., 2017). More concretely, given two samples $z_s \sim \mathcal{S}_t(z)$ and $z_t \sim \mathcal{T}(z)$ we generate an interpolated sample $z_{\text{int}} = \xi z_s + (1 - \xi) z_t$ with $\xi \sim \text{Unif}[0, 1]$. Then we add a gradient penalty $\|\nabla d(z_{\text{int}})\|_2^2$ as a regularization term to control the Lipschitz property w.r.t. the discriminator $d_1, \cdots, d_T$.

D.10 Algorithm Descriptions

We propose a detailed pipeline of the proposed algorithm in the following, shown in Algorithm 5 and 6. As for updating $\boldsymbol{\lambda}$ and α_t, we iteratively solve the convex optimization problem after each training epoch and updating them by using the moving average technique.

For solving the $\boldsymbol{\lambda}$ and α_t, we notice that frequently updating these two parameters in the mini-batch level will lead to an instability result during the training. [2] As a consequence, we compute the

[2] In the label distribution shift scenarios, the mini-batch datasets are highly labeled imbalanced. If we evaluate α_t over

accumulated confusion matrix, weighted prediction risk, and conditional Wasserstein distance for the whole training epoch and then solve the optimization problem. We use CVXPY to optimize the two standard convex losses. [3]

Comparison with different time and memory complexities. We discuss the time and memory complexity of our approach.

Time complexity: In computing each batch we need to compute T re-weighted losses, T domain adversarial losses and T explicit conditional losses. Then our computational complexity is still $\mathcal{O}(T)$ during the mini-batch training, which is comparable with recent SOTA such as MDAN and DARN. In addition, after each training epoch we need to estimate α_t and $\boldsymbol{\lambda}$, which can have time complexity $\mathcal{O}(T|\mathcal{Y}|)$ with each epoch. (If we adopt SGD to solve these two convex problems). Therefore, the our proposed algorithm is time complexity $\mathcal{O}(T|\mathcal{Y}|)$. The extra $\mathcal{Y}$ term in time complexity is due to the approach of label shift in the designed algorithm.

Memory Complexity: Our proposed approach requires $\mathcal{O}(T)$ domain discriminators and $\mathcal{O}(T|\mathcal{Y}|)$ class-feature centroids. By the contrary, MDAN and DARN require $\mathcal{O}(T)$ domain discriminator and M3SDA and MDMN require $\mathcal{O}(T^2)$ domain discriminators. Since our class-feature centroids are defined in the latent space (z), then the memory complexity of the class-feature centroids can be much smaller than domain discriminators.

the mini-batch, it can be computationally expensive and unstable.

[3] The optimization problem w.r.t. α_t and $\boldsymbol{\lambda}$ is not large scale, then using the standard convex solver is fast and accurate.

Algorithm 5 Wasserstein Aggregation Domain Network (unsupervised scenarios, one iteration)

```
Require: Labeled source samples $\hat{\mathcal{S}}_1, \dots, \hat{\mathcal{S}}_T$, Target samples $\hat{\mathcal{T}}$
Ensure: Label distribution ratio $\hat{\alpha}_t$ and task relation simplex $\boldsymbol{\lambda}$. Feature Learner $g$, Classifier $h$,
    Statistic critic function $d_1, \dots, d_T$, class centroid for source $\mathbf{C}_t^y$ and target $\mathbf{C}^y$ ($\forall t = [1, T], y \in \mathcal{Y}$).

▷ ▷ ▷ DNN Parameter Training Stage (fixed $\alpha_t$ and $\boldsymbol{\lambda}$) ◁ ◁ ◁
for mini-batch of samples $(\mathbf{x}_{\mathcal{S}_1}, \mathbf{y}_{\mathcal{S}_1}) \sim \hat{\mathcal{S}}_1, \dots, (\mathbf{x}_{\mathcal{S}_T}, \mathbf{y}_{\mathcal{S}_T}) \sim \hat{\mathcal{S}}_T$, $(\mathbf{x}_{\mathcal{T}}) \sim \hat{\mathcal{T}}$ do
    Predict target pseudo-label $\bar{\mathbf{y}}_{\mathcal{T}} = \text{argmax}_y h(g(\mathbf{x}_{\mathcal{T}}), y)$
    Compute source confusion matrix for each batch (un-normalized)
        $C_{\hat{\mathcal{S}}_t} = \#[\text{argmax}_{y'} h(z, y') = y, Y = k]$ $(t = 1, \dots, T)$
    Compute the *batched* class centroid for source $C_t^y$ and target $C^y$.
    Moving Average for update source/target class centroid: (We set $\epsilon_1 = 0.7$)
            Source class centroid update    $\mathbf{C}_t^y = \epsilon_1 \times \mathbf{C}_t^y + (1 - \epsilon_1) \times C_t^y$
            Target class centroid update    $\mathbf{C}^y = \epsilon_1 \times \mathbf{C}^y + (1 - \epsilon_1) \times C^y$
    Updating $g, h, d_1, \dots, d_T$ (SGD and Gradient Reversal), based on Eq.(D.3)
end for
▷ ▷ ▷ Estimation $\hat{\alpha}_t$ and $\boldsymbol{\lambda}$ ◁ ◁ ◁
Compute the global(normalized) source confusion matrix
    $C_{\hat{\mathcal{S}}_t} = \hat{\mathcal{S}}_t[\text{argmax}_{y'} h(z, y') = y, Y = k]$ $(t = 1, \dots, T)$
Solve $\alpha_t$ (denoted as $\{\alpha_t'\}_{t=1}^T$) by Equation (5.1) (Or Eq.(D.2)) in the partial scenario).
Update $\alpha_t$ by moving average: $\alpha_t = \epsilon_1 \times \alpha_t + (1 - \epsilon_1) \times \alpha_t'$
Compute the weighted loss and weighted centroid distance, then solve $\boldsymbol{\lambda}$ (denoted as $\boldsymbol{\lambda}'$) from Sec.
    2.3.
Updating $\boldsymbol{\lambda}$ by moving average: $\boldsymbol{\lambda} = 0.8 \times \boldsymbol{\lambda} + 0.2 \times \boldsymbol{\lambda}'$
```

D.11 Dataset Description and Experimental Details

D.11.1 Amazon Review Dataset

We used the amazon review dataset (Blitzer et al., 2007). It contains four domains (Books, DVD, Electronics, and Kitchen) with positive (label "1") and negative product reviews (label "0"). The data size is 6465 (Books), 5586 (DVD), 7681 (Electronics), and 7945 (Kitchen). We follow the common data pre-processing strategies Chen et al. (2012): use the bag-of-words (BOW) features then extract the top-5000 frequent unigram and bigrams of all the reviews.

We also noticed the original data-set are label balanced $\mathcal{D}(y = 0) = \mathcal{D}(y = 1)$. To enhance the benefits of the proposed approach, we create a new dataset with label distribution drift. Specifically, in the experimental settings, we randomly drop 50% data with label "0" (negative reviews) for all the source data while keeping the target identical, showing in Fig (D.2).

We choose the MLP model with

- feature representation function g: $[5000, 1000]$ units
- Task prediction and domain discriminator function $[1000, 500, 100]$ units,

We choose the dropout rate as 0.7 in the hidden and input layers. The hyper-parameters are chosen

Algorithm 6 Wasserstein Aggregation Domain Network (Limited Target Data, one iteration)

```
Require: Labeled source samples $\hat{\mathcal{S}}_1, \dots, \hat{\mathcal{S}}_T$, Target samples $\hat{\mathcal{T}}$, Label shift ratio $\alpha_t$
Ensure: Task relation simplex $\boldsymbol{\lambda}$. Feature Learner $g$, Classifier $h$, Statistic critic function $d_1, \dots, d_T$,
    class centroid for source $\mathbf{C}_t^y$ and target $\mathbf{C}^y$ ($\forall t = [1, T], y \in \mathcal{Y}$).
▷ ▷ ▷ DNN Parameter Training Stage (fixed $\boldsymbol{\lambda}$) ◁ ◁ ◁
for mini-batch of samples $(\mathbf{x}_{\mathcal{S}_1}, \mathbf{y}_{\mathcal{S}_1}) \sim \hat{\mathcal{S}}_1, \dots, (\mathbf{x}_{\mathcal{S}_T}, \mathbf{y}_{\mathcal{S}_T}) \sim \hat{\mathcal{S}}_T$, $(\mathbf{x}_{\mathcal{T}}) \sim \hat{\mathcal{T}}$ do
    Compute the *batched* class centroid for source $C_t^y$ and target $C^y$.
    Moving Average for update source/target class centroid: (We set $\epsilon_1 = 0.7$)
            Source class centroid update    $\mathbf{C}_t^y = \epsilon_1 \times \mathbf{C}_t^y + (1 - \epsilon_1) \times C_t^y$
            Target class centroid update    $\mathbf{C}^y = \epsilon_1 \times \mathbf{C}^y + (1 - \epsilon_1) \times C^y$
    Updating $g, h, d_1, \dots, d_T$ (SGD and Gradient Reversal), based on Eq.(D.3).
end for
▷ ▷ ▷ Estimation $\boldsymbol{\lambda}$ ◁ ◁ ◁
Solve $\boldsymbol{\lambda}$ by Sec. 2.3. (denoted as $\boldsymbol{\lambda}'$)
Updating $\boldsymbol{\lambda}$ by moving average: $\boldsymbol{\lambda} = \epsilon_1 \times \boldsymbol{\lambda} + (1 - \epsilon_1) \times \boldsymbol{\lambda}'$
```

based on cross-validation. The neural network is trained for 50 epochs and the mini-batch size is 20 per domain. The optimizer is Adadelta with a learning rate of 0.5.

Experimental Setting We use the amazon Review dataset for two transfer learning scenarios (limited target labels and unsupervised DA). We first randomly select 2K samples for each domain. Then we create a drifted distribution of each source, making each source ≈ 1500 and target sample still 2K.

In the unsupervised DA, we use these labeled source tasks and *unlabelled* target task, which aims to predict the labels on the target domain.

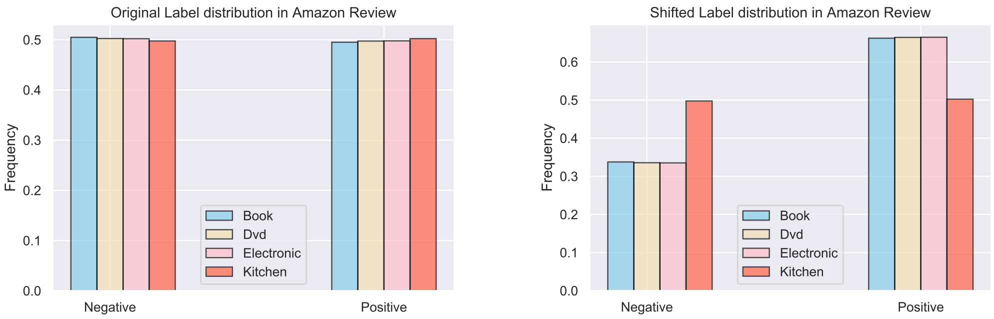
In the conventional transfer learning, we random sample only 10% dataset (≈ 200 samples) as the target training set and the rest 90% samples as the target test set.

We select $C_0 = 0.01$ and $C_1 = 1$ for these two transfer scenarios. In both practical settings, we set the maximum training epoch as 50.

D.11.2 Digit Recognition

We follow the same settings of Ganin et al. (2016) and we use four-digit recognition datasets in the experiments MNIST, USPS, SVHN, and Synth. MNIST and USPS are the standard digits recognition task. Street View House Number (SVHN) Ganin et al. (2016) is the digit recognition dataset from house numbers in Google Street View Images. Synthetic Digits (Synth) Ganin et al. (2016) is a synthetic dataset by transforming the SVHN dataset.

We also visualize the label distribution in these four datasets. The original datasets show an almost uniform label distribution on the MNIST as well as Synth, (showing in Fig. D.4 (a)). In our paper, we generate a label distribution drift on the source datasets for each multi-source transfer learning. Concretely, we drop 50% of the data on digits 5-9 of all the sources while we keep the target label



(a) (b)

Figure D.2 – Amazon Review dataset (a) Original Label Training Distribution; (b) Label-Shifted distribution with sources tasks: Book, Dvd, Electronic, and target task Kitchen. We randomly drop 50% of the negative reviews for all the source distribution while keeping the target label distribution unchanged.

distribution unchanged. (Fig. D.4 (b) illustrated one example with sources: Mnist, USPS, SVHN, and Target Synth. We drop the labels only on the sources.)

MNIST and USPS images are resized to 32×32 and represented as 3-channel color images to match the shape of the other three datasets. Each domain has its own given training and test sets when downloaded. Their respective training sample sizes are 60000, 7219, 73257, 479400, and the respective test sample sizes are 10000, 2017, 26032, 9553.

The model structure is shown in Fig. D.3. There is no dropout and the hyperparameters are chosen based on cross-validation. It is trained for 60 epochs and the mini-batch size is 128 per domain. The optimizer is Adadelta with a learning rate of 1.0. We adopted $\gamma = 0.5$ for MDAN and $\gamma = 0.1$ for DARN in the baseline (Wen et al., 2020).

Experimental Setting We use the Digits dataset for two transfer learning scenarios (limited target labels and unsupervised DA). Notice the USPS data has only 7219 samples and the digits dataset is relatively simple. We first randomly select 7K samples for each domain. We create a drifted distribution of each source, making each source ≈ 5300, and the target sample still 7K.

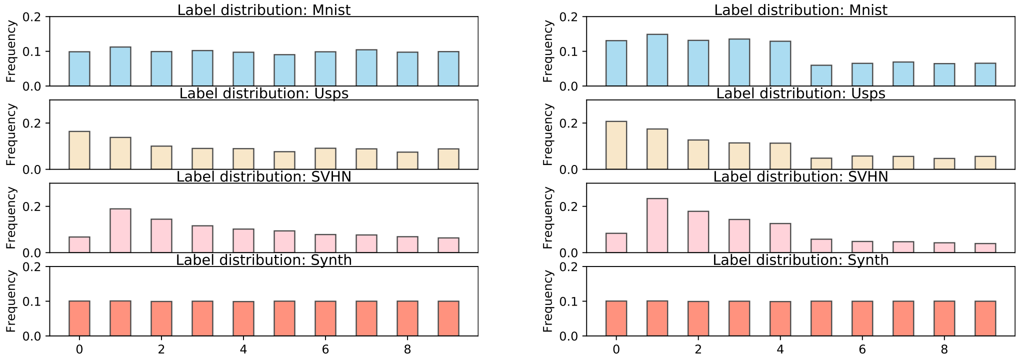
In the unsupervised DA, we use these labeled source tasks and *unlabelled* target task, which aims to predict the labels on the target domain.

In the transfer learning with limited data, we randomly sample only 10% dataset (≈ 700 samples) as the target training set and the rest 90% samples as the target test set.

We select $C_0 = 0.01$ and C_1 as the maximum prediction loss $C_1 = \max_t R^{\alpha_t}(h)$ as the hyper-parameters across these two scenarios. The maximum training epoch is 60.

1. Feature extractor: with 3 convolution layers.
   ’layer1’: ’conv’: [3, 3, 64], ’relu’: [], ’maxpool’: [2, 2, 0],
   ’layer2’: ’conv’: [3, 3, 128], ’relu’: [], ’maxpool’: [2, 2, 0],
   ’layer3’: ’conv’: [3, 3, 256], ’relu’: [], ’maxpool’: [2, 2, 0],
2. Task prediction: with 3 fully connected layers.
   ’layer1’: ’fc’: [*, 512], ’act_fn’: ’relu’,
   ’layer2’: ’fc’: [512, 100], ’act_fn’: ’relu’,
   ’layer3’: ’fc’: [100, 10],
3. Domain Discriminator: with 2 fully connected layers.
   reverse_gradient()
   ’layer1’: ’fc’: [*, 256], ’act_fn’: ’relu’,
   ’layer2’: ’fc’: [256, 1],

Figure D.3 – Neural Network Structure in the digits recognition (Ganin et al., 2016)



(a)

(b)

Figure D.4 – One example in Digits dataset with Sources: MNIST, USPS, SVHN and Target Synth. We randomly drop 50% data on digits 5-9 in all sources while keeping target label distribution unchanged.

D.11.3 Office-Home dataset

To show the dataset in the complex scenarios, we use the challenging Office-Home dataset (Venkateswara et al., 2017). It contains images of 65 objects such as a spoon, sink, mug, and pen from four different domains: Art (paintings, sketches, and/or artistic depictions), Clipart (clipart images), Product (images without background), and Real-World (regular images captured with a camera). One of the four datasets is chosen as an unlabelled target domain and the other three datasets are used as labeled source domains.

The dataset size is 2427 (Art), 4365 (Clipart), 4439 (Product), 4357 (Real-World). We follow the same training/test procedure as (Wen et al., 2020). We did not re-sample the source label distribution to uniform distribution in the data pre-processing step. All the baselines are evaluated under the same setting.

We use the ResNet50 (He et al., 2016) pretrained from the ImageNet in PyTorch as the base network for feature learning and put an MLP with the network structure shown in Fig. D.6.

Experimental Settings We use the original Office-Home dataset for two transfer learning scenarios (unsupervised DA and label-partial unsupervised DA). We use SGD optimizer with learning rate 0.005, momentum 0.9 and weight_decay value 1e-3. It is trained for 100 epochs and the mini-batch size is 32 per domain. As for the baselines, MDAN use $\gamma = 1.0$ while DARN use $\gamma = 0.5$. We select $C_0 = 0.01$ and C_1 as the maximum prediction loss $C_1 = \max_t R^{\alpha_t}(h)$ as the hyper-parameters across these two scenarios.

In the multi-source unsupervised partial DA, we randomly select 35 classes from the target (by repeating 3 samplings), then at each sampling we run 5 times. The final result is based on these $3 \times 5 = 15$ repetitions.

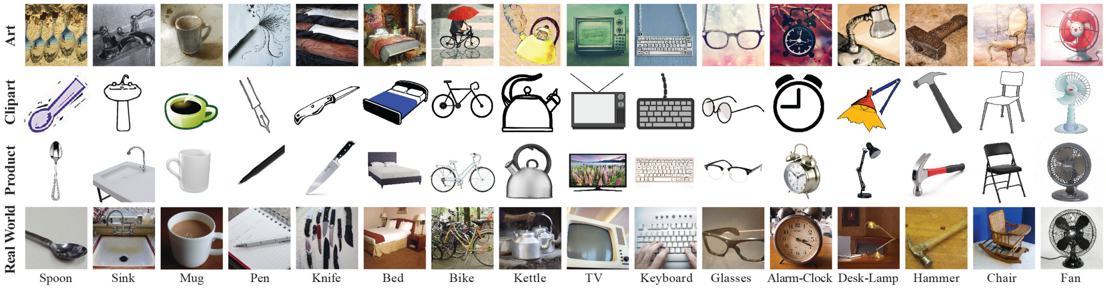


Figure D.5 – Samples Images From Office-Home dataset (Venkateswara et al., 2017), which consists four domains with non-uniform label distribution.

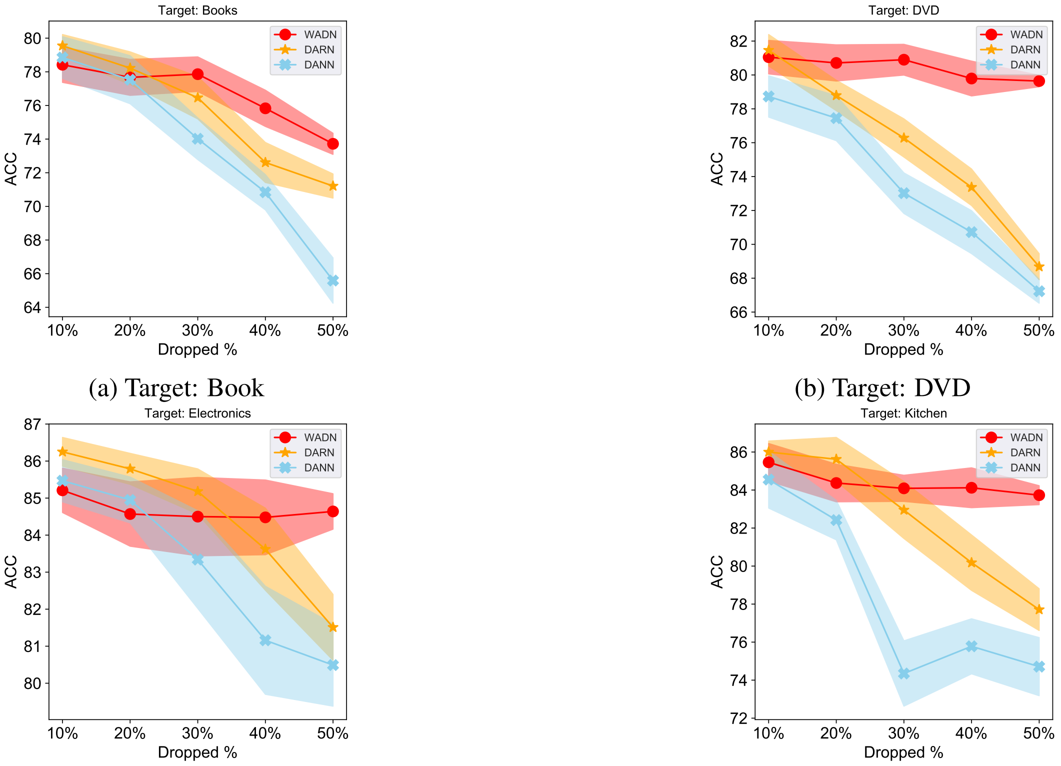
1. Feature extractor: ResNet50 (He et al., 2016),
2. Task prediction: with 3 fully connected layers.
   'layer1': 'fc': [*, 256], 'batch_normalization', 'act_fn': 'Leaky_relu',
   'layer2': 'fc': [256, 256], 'batch_normalization', 'act_fn': 'Leaky_relu',
   'layer3': 'fc': [256, 65],
3. Domain Discriminator: with 3 fully connected layers.
   reverse_gradient()
   'layer1': 'fc': [*, 256], 'batch_normalization', 'act_fn': 'Leaky_relu',
   'layer2': 'fc': [256, 256], 'batch_normalization', 'act_fn': 'Leaky_relu',
   'layer3': 'fc': [256, 1], 'Sigmoid',

Figure D.6 – Neural Network Structure in the Office-Home

D.12 Analysis of Unsupervised DA

D.12.1 Ablation Study: Different Dropping Rate

To show the effectiveness of our proposed approach, we change the drop rate of the source domain, showing in Fig.(D.7). We observe that in task Book, DVD, Electronic, and Kitchen, the results are significantly better under a large label-shift. In the initialization with almost no label shift, the state-of-the-art DARN illustrates a slightly better ($< 1\%$) result.



(a) Target: Book

(b) Target: DVD

(c) Target: Electronics

(d) Target: Kitchen

Figure D.7 – Different label drift levels on Amazon Dataset. Larger dropping rate means higher label shift.

D.12.2 Additional Analysis on Amazon Dataset

We present two additional results to illustrate the working principles of WADN, showing in (D.8).

D.12.3 Additional Analysis on Digits Dataset

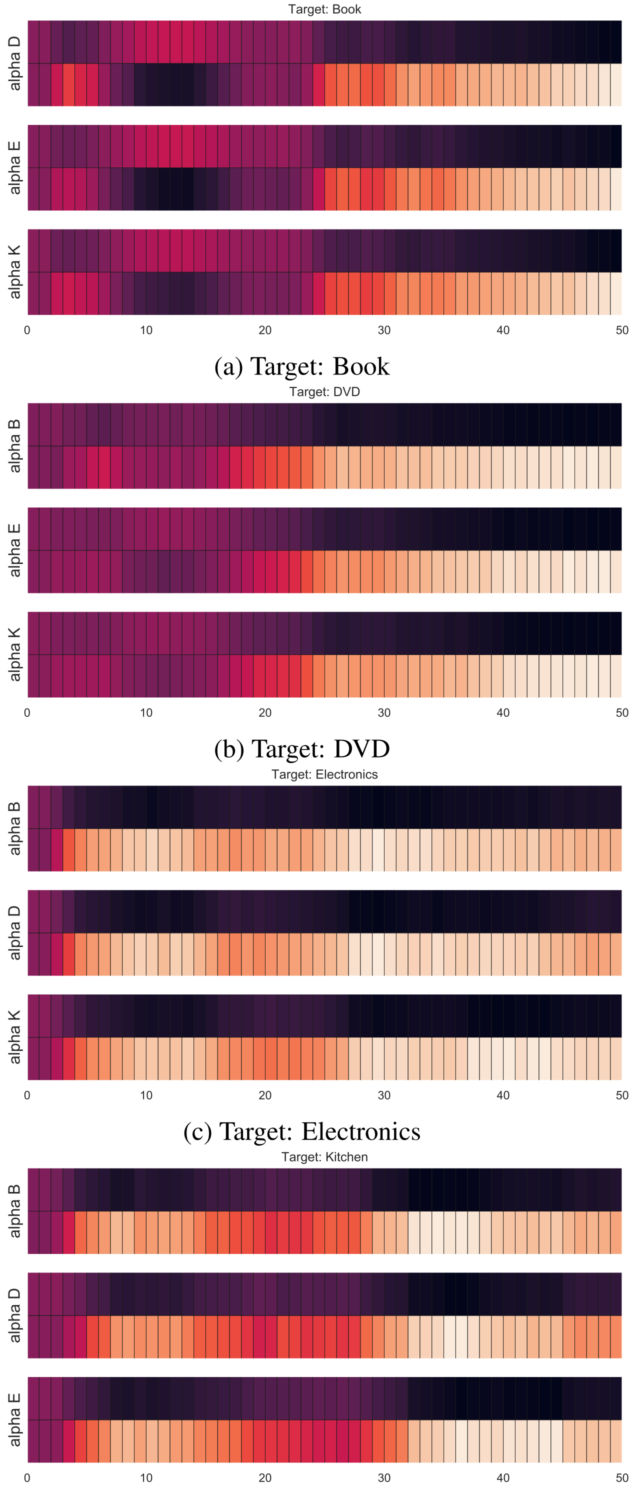
We show the evolution of $\hat{\alpha}_t$ on WADN, which verifies the correctness of our proposed principle. Since we drop digits 5-9 in the source domains, the results in Fig. (D.9) illustrate a higher $\hat{\alpha}_t$ on these digits.

D.13 Partial multi-source Unsupervised DA

From Fig. (D.10), WADN is consistently better than other baselines, given different selected classes.

Besides, when fewer classes are selected, the accuracy in DANN, PADA, and DARN is not drastically dropping but maintaining a relatively stable result. We think the following possible reasons:

- The reported performances are based on the **average of different selected sub-classes rather than one sub-class selection.** From the statistical perspective, if we take a close look at the **variance**, the results in DANN are *much more unstable* (higher std) induced by the different



(a) Target: Book

(b) Target: DVD

(c) Target: Electronics

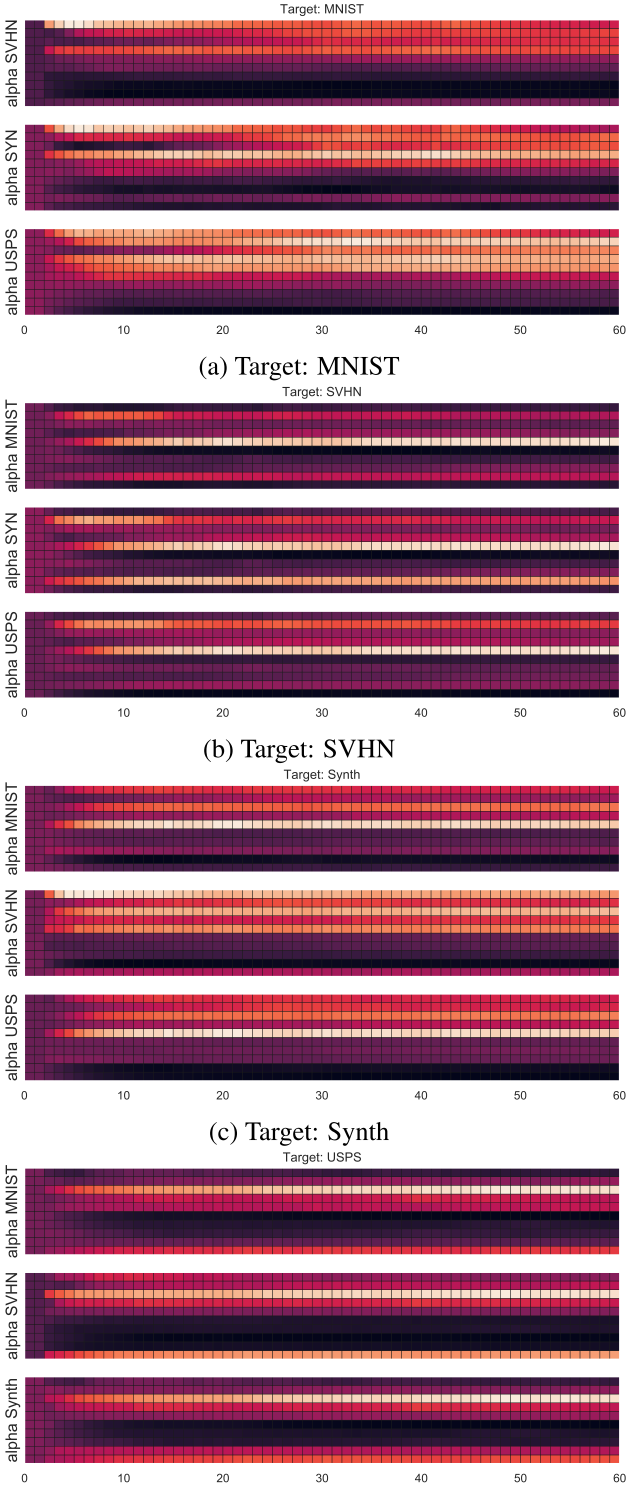
(d) Target: Kitchen

Figure D.8 – Amazon Dataset. WADN approach: evolution of $\hat{\alpha}_t$ during the training. Darker indicates higher Value. Since we drop $y = 0$ in the sources, then the true $\alpha_t(0) > 1$ will be assigned with higher value.

samplings. Therefore, the conventional domain adversarial training is improper for handling the partial transfer since it is not reliable and negative transfer still occurs.

- In multi-source DA, it is equally important to detect the non-overlapping classes and find the most similar sources. Comparing the baselines that only focus on one or two principles shows the importance of unified principles in multi-source partial DA.
- We also observe that in the Real-World dataset, the DANN improves the performance by a relatively large value. This is due to the inherent difficultly of the learning task itself. In fact, the Real-World domain illustrates a much higher performance compared with other domains. According to the Fano lower bound, *a task with smaller classes is generally easy to learn*. It is possible the vanilla approach showed improvement but still with a much higher variance.

Fig (D.11), (D.12) showed the estimated $\hat{\alpha}_t$ with different selected classes. The results validate the correctness of WADN in estimating the label distribution ratio.



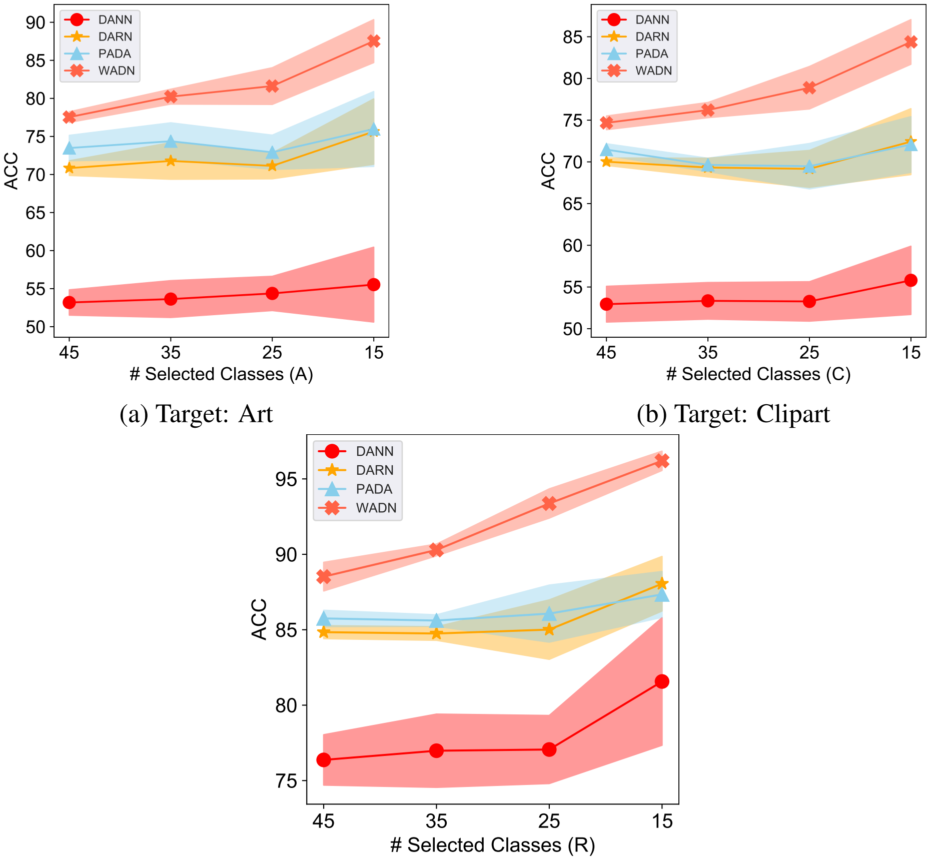
(a) Target: MNIST

(b) Target: SVHN

(c) Target: Synth

(d) Target: USPS

Figure D.9 – Digits Dataset. WADN approach: evolution of $\hat{\alpha}_t$ during the training. Darker indicates higher value. Since we drop digits $5-9$ on source domain, therefore, $\alpha_t(y)$, $y \in [5,9]$ will be assigned with a relative higher value.

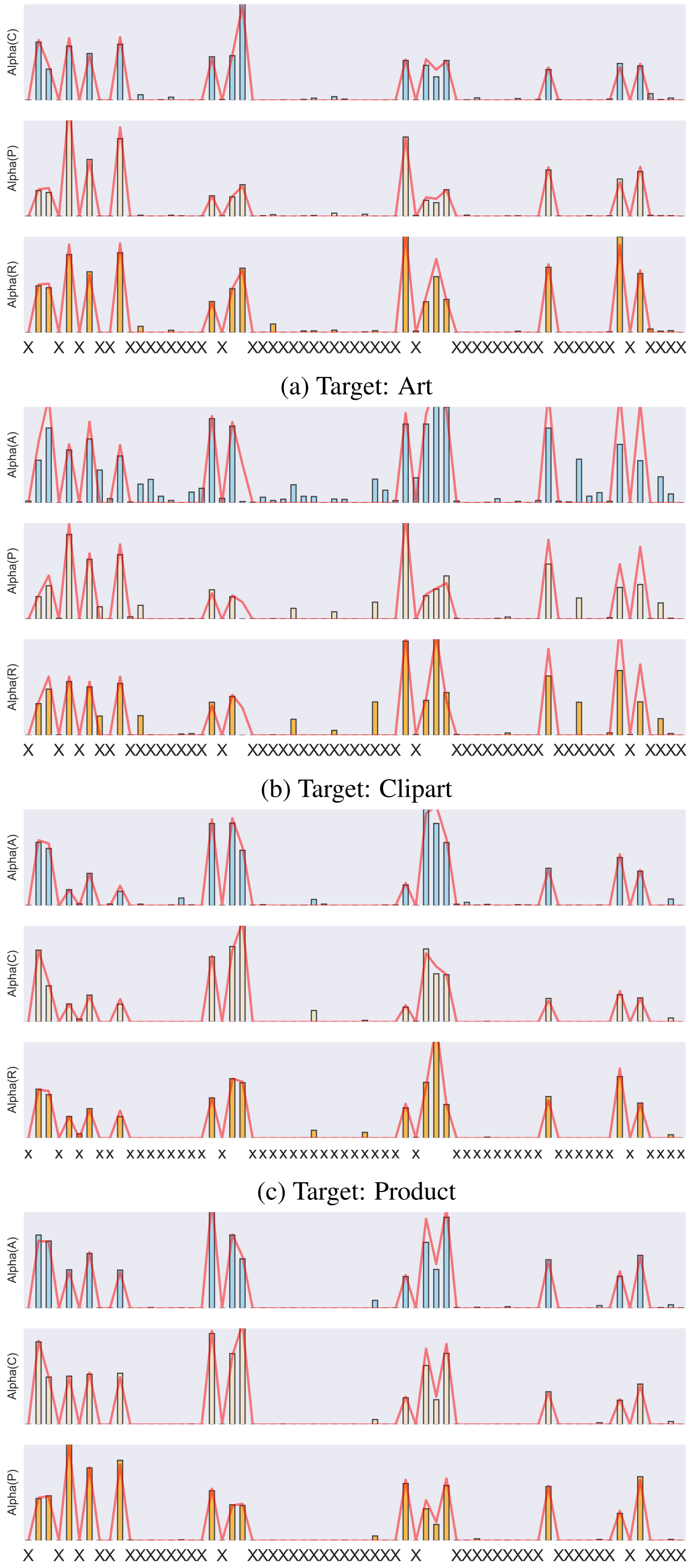


(a) Target: Art

(b) Target: Clipart

(c) Target: Real-World

Figure D.10 – Multi-source Label Partial DA: Performance with different target selected classes.



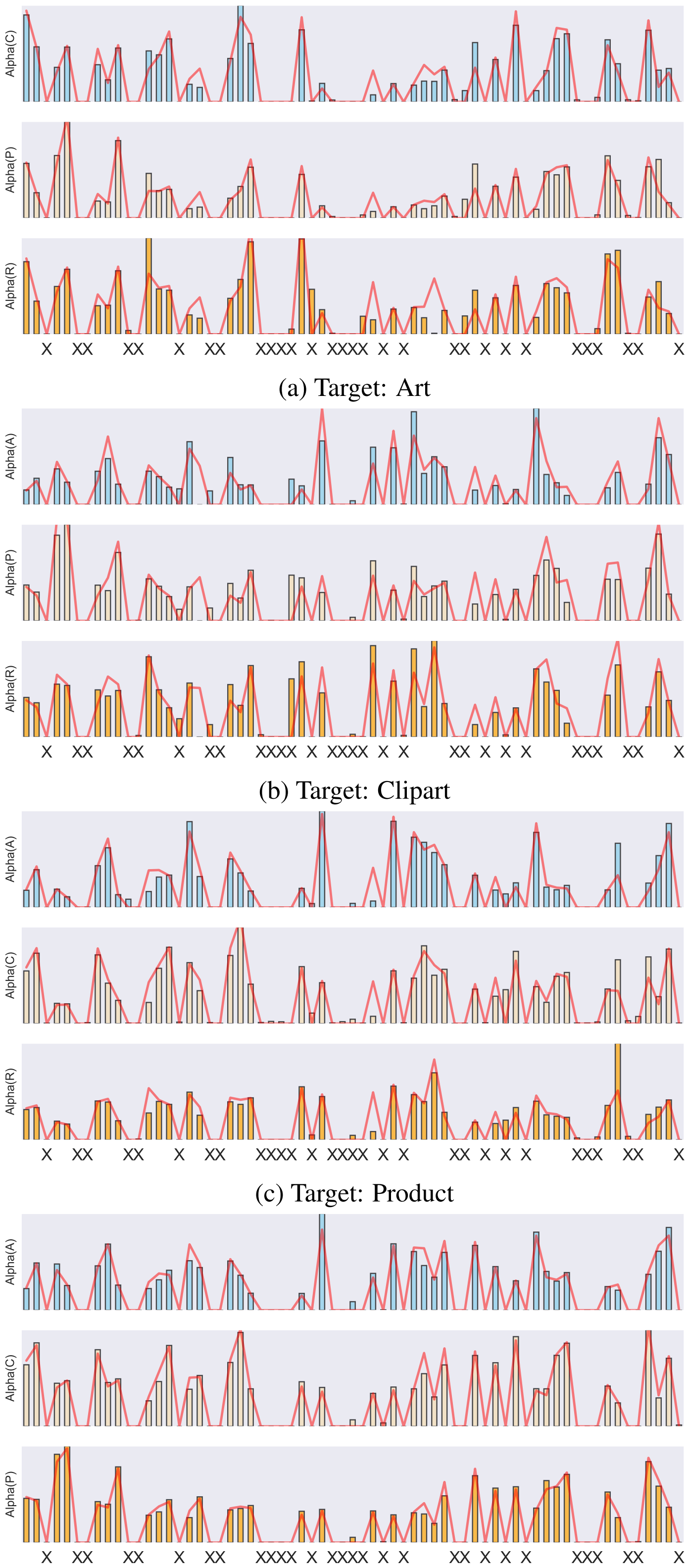
(a) Target: Art

(b) Target: Clipart

(c) Target: Product

(d) Target: Real-World

Figure D.11 – We select 15 classes and visualize estimated $\hat{\alpha}_t$ (the bar plot). The "X" along the x-axis represents the index of **dropped** 50 classes. The red curves are the ground-truth label distribution ratio.



(a) Target: Art

(b) Target: Clipart

(c) Target: Product

(d) Target: Real-World

Figure D.12 – We select 35 classes and visualize estimated $\hat{\alpha}_t$ (the bar plot). The "X" along the x-axis represents the index of **dropped** 30 classes. The red curves are the ground-truth label distribution ratio.

Bibliography

Achille, A. and Soatto, S. (2018). Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1):1947–1980.

Akuzawa, K., Iwasawa, Y., and Matsuo, Y. (2019). Adversarial invariant feature learning with accuracy constraint for domain generalization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 315–331. Springer.

Arbelaez, P., Maire, M., Fowlkes, C., and Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein gan. *arXiv preprint arXiv:1701.07875*.

Asatryan, H., Gottschalk, H., Lippert, M., and Rottmann, M. (2020). A convenient infinite dimensional framework for generative adversarial learning. *ArXiv*, abs/2011.12087.

Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. (2019). Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*.

Azizzadenesheli, K. (2020). Importance weight estimation and generalization in domain adaptation under label shift. *ArXiv*, abs/2011.14251.

Azizzadenesheli, K., Liu, A., Yang, F., and Anandkumar, A. (2019). Regularized learning for domain adaptation under label shifts. In *International Conference on Learning Representations*.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Balaji, Y., Chellappa, R., and Feizi, S. (2019). Normalized wasserstein distance for mixture distributions with applications in adversarial learning and domain adaptation. *arXiv preprint arXiv:1902.00415*.

Balaji, Y., Sankaranarayanan, S., and Chellappa, R. (2018). Metareg: Towards domain generalization using meta-regularization. In *Advances in Neural Information Processing Systems*, pages 998–1008.

Balcan, M.-F., Beygelzimer, A., and Langford, J. (2009). Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89.

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

Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. (2006). Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137–144.

Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. (2007). Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144.

Ben-David, S., Lu, T., Luu, T., and Pál, D. (2010b). Impossibility theorems for domain adaptation. In *International Conference on Artificial Intelligence and Statistics*, pages 129–136.

Ben-David, S. and Urner, R. (2014). Domain adaptation–can quantity compensate for quality? *Annals of Mathematics and Artificial Intelligence*, 70(3):185–202.

Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447.

Bolley, F., Guillin, A., and Villani, C. (2007). Quantitative concentration inequalities for empirical measures on non-compact spaces. *Probability Theory and Related Fields*, 137(3-4):541–593.

Boucheron, S., Lugosi, G., and Massart, P. (2013). *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press.

Bucci, S., D'Innocente, A., and Tommasi, T. (2019). Tackling partial domain adaptation with self-supervision. In *International Conference on Image Analysis and Processing*, pages 70–81. Springer.

Cai, D., Sheth, R., Mackey, L., and Fusi, N. (2020). Weighted meta-learning. *ArXiv*, abs/2003.09465.

Cai, R., Li, Z., Wei, P., Qiao, J., Zhang, K., and Hao, Z. (2019). Learning disentangled semantic representation for domain adaptation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 2060–2066. AAAI Press.

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

Cao, Z., You, K., Long, M., Wang, J., and Yang, Q. (2019). Learning to transfer examples for partial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2985–2994.

Caruana, R., Lawrence, S., and Giles, C. L. (2001). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pages 402–408.

Chen, C., Xie, W., Huang, W., Rong, Y., Ding, X., Huang, Y., Xu, T., and Huang, J. (2019a). Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 627–636.

Chen, M., Xu, Z., Weinberger, K. Q., and Sha, F. (2012). Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1627–1634.

Chen, X., Awadallah, A. H., Hassan, H., Wang, W., and Cardie, C. (2019b). Multi-source cross-lingual model transfer: Learning what to share. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Chen, Z., Chen, C., Cheng, Z., Fang, K., and Jin, X. (2020). Selective transfer with reinforced transfer network for partial domain adaptation. In *AAAI Conference on Artificial Intelligence*.

Christodoulidis, S., Anthimopoulos, M., Ebner, L., Christe, A., and Mougiakakou, S. (2016). Multisource transfer learning with convolutional neural networks for lung pattern analysis. *IEEE journal of biomedical and health informatics*, 21(1):76–84.

Ciliberto, C., Mroueh, Y., Poggio, T., and Rosasco, L. (2015). Convex learning of multiple tasks and their structure. In *International Conference on Machine Learning*, pages 1548–1557.

Coates, A., Ng, A., and Lee, H. (2011). An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223.

Combes, R. T. d., Zhao, H., Wang, Y.-X., and Gordon, G. (2020). Domain adaptation with conditional distribution matching and generalized label shift. *arXiv preprint arXiv:2003.04475*.

Cortes, C., Mansour, Y., and Mohri, M. (2010). Learning bounds for importance weighting. In *Advances in neural information processing systems*, pages 442–450.

Cortes, C., Mohri, M., and Medina, A. M. (2019). Adaptation based on generalized discrepancy. *Journal of Machine Learning Research*, 20(1):1–30.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. (2017). Joint distribution optimal transportation for domain adaptation. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 3730–3739. Curran Associates, Inc.

Cui, Z. and Sato, I. (2020). Active learning using discrepancy.

Culotta, A. and McCallum, A. (2005). Reducing labeling effort for structured prediction tasks. In *AAAI conference on artificial intelligence*.

Dasgupta, S. (2011). Two faces of active learning. *Theoretical computer science*, 412(19):1767–1781.

Du Plessis, M. C. and Sugiyama, M. (2014). Semi-supervised learning of class balance under class-prior change by distribution matching. *Neural Networks*, 50:110–119.

Edwards, H. and Storkey, A. (2015). Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897*.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*.

Gal, Y., Islam, R., and Ghahramani, Z. (2017). Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1183–1192. JMLR. org.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Garg, S., Wu, Y., Balakrishnan, S., and Lipton, Z. C. (2020). A unified view of label shift estimation. *arXiv preprint arXiv:2003.07554*.

Geden, M., Emerson, A., Rowe, J., Azevedo, R., and Lester, J. (2020). Predictive student modeling in educational games with multi-task learning. In *AAAI*.

Geiss, L. S., Wang, J., Cheng, Y. J., Thompson, T. J., Barker, L., Li, Y., Albright, A. L., and Gregg, E. W. (2014). Prevalence and incidence trends for diagnosed diabetes among adults aged 20 to 79 years, united states, 1980-2012. *Jama*, 312(12):1218–1226.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. (2013). A pac-bayesian approach for domain adaptation with specialization to linear classifiers. In *International conference on machine learning*, pages 738–746.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. (2016). A new pac-bayesian perspective on domain adaptation. In *International conference on machine learning*, pages 859–868.

Gissin, D. and Shalev-Shwartz, S. (2019). Discriminative active learning. *CoRR*, abs/1907.06347.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Schölkopf, B. (2016). Domain adaptation with conditional transferable components. In *International conference on machine learning*, pages 2839–2848.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777.

Hanneke, S. and Kpotufe, S. (2019). On the value of target data in transfer learning. In *Advances in Neural Information Processing Systems*, pages 9867–9877.

Haussmann, M., Hamprecht, F., and Kandemir, M. (2019). Deep active learning with adaptive acquisition. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 2470–2476. International Joint Conferences on Artificial Intelligence Organization.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hoffman, J., Kulis, B., Darrell, T., and Saenko, K. (2012). Discovering latent domains for multisource domain adaptation. In *European Conference on Computer Vision*, pages 702–715. Springer.

Hoffman, J., Mohri, M., and Zhang, N. (2018). Algorithms and theory for multiple-source adaptation. In *Advances in Neural Information Processing Systems*, pages 8246–8256.

Hull, J. J. (1994). A database for handwritten text recognition research. *IEEE Transactions on pattern analysis and machine intelligence*, 16(5):550–554.

Ilse, M., Tomczak, J. M., Louizos, C., and Welling, M. (2019). Diva: Domain invariant variational autoencoders. *arXiv preprint arXiv:1905.10427*.

Johansson, F., Sontag, D., and Ranganath, R. (2019). Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 527–536.

Johansson, F. D., Shalit, U., Kallus, N., and Sontag, D. (2020). Generalization bounds and representation learning for estimation of potential outcomes and causal effects. *arXiv preprint arXiv:2001.07426*.

Jose, S. T. and Simeone, O. (2020). Information-theoretic bounds on transfer generalization gap based on jensen-shannon divergence. *ArXiv*, abs/2010.09484.

Kifer, D., Ben-David, S., and Gehrke, J. (2004). Detecting change in data streams. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 180–191. VLDB Endowment.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Konstantinov, N. and Lampert, C. (2019). Robust learning from untrusted sources. *arXiv preprint arXiv:1901.10310*.

Krizhevsky, A. et al. (2009). Learning multiple layers of features from tiny images. Technical report, Citeseer.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*.

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

Lee, J., Sattigeri, P., and Wornell, G. (2019). Learning new tricks from old dogs: Multi-source transfer learning from pre-trained networks. In *Advances in Neural Information Processing Systems*, pages 4370–4380.

Li, H., Jialin Pan, S., Wang, S., and Kot, A. C. (2018a). Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409.

Li, J., Wu, W., Xue, D., and Gao, P. (2019a). Multi-source deep transfer neural network algorithm. *Sensors*, 19(18):3992.

Li, Y., Carlson, D. E., et al. (2018b). Extracting relationships by multi-domain matching. In *Advances in Neural Information Processing Systems*, pages 6799–6810.

Li, Y., Murias, M., Major, S., Dawson, G., and Carlson, D. (2019b). On target shift in adversarial domain adaptation. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 616–625.

Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., and Tao, D. (2018c). Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639.

Lin, C., Zhao, S., Meng, L., and Chua, T.-S. (2020). Multi-source domain adaptation for visual sentiment classification. *arXiv preprint arXiv:2001.03886*.

Lipton, Z., Wang, Y.-X., and Smola, A. (2018). Detecting and correcting for label shift with black box predictors. In *International Conference on Machine Learning*, pages 3122–3130.

Liu, J., Hong, Y., D'Agostino Sr, R. B., Wu, Z., Wang, W., Sun, J., Wilson, P. W., Kannel, W. B., and Zhao, D. (2004). Predictive value for the chinese population of the framingham chd risk assessment tool compared with the chinese multi-provincial cohort study. *Jama*, 291(21):2591–2599.

Liu, P., He, G., and Zhao, L. (2021). From model-driven to data-driven: A survey on active deep learning.

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

Long, M., Cao, Y., Wang, J., and Jordan, M. I. (2015). Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. (2018). Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1640–1650.

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

Long, M., Wang, J., Ding, G., Sun, J., and Yu, P. S. (2013). Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, pages 2200–2207.

Louizos, C., Swersky, K., Li, Y., Welling, M., and Zemel, R. (2015). The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*.

Mansour, Y., Mohri, M., and Rostamizadeh, A. (2009a). Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*.

Mansour, Y., Mohri, M., and Rostamizadeh, A. (2009b). Multiple source adaptation and the rényi divergence. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 367–374. AUAI Press.

Mansour, Y., Mohri, M., Suresh, A. T., and Wu, K. (2020). A theory of multiple-source adaptation with limited target labeled data. *arXiv preprint arXiv:2007.09762*.

Mayer, C. and Timofte, R. (2018). Adversarial sampling for active learning. *arXiv preprint arXiv:1808.06671*.

Mohri, M. and Medina, A. M. (2012). New analysis and algorithm for learning with drifting distributions. In *International Conference on Algorithmic Learning Theory*, pages 124–138. Springer.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of machine learning*. MIT Press.

Motiian, S., Piccirilli, M., Adjeroh, D. A., and Doretto, G. (2017). Unified deep supervised domain adaptation and generalization. In *The IEEE International Conference on Computer Vision (ICCV)*.

Mundt, M., Hong, Y. W., Pliushch, I., and Ramesh, V. (2020). A wholistic view of continual learning with deep neural networks: Forgotten lessons and the bridge to active and open world learning. *ArXiv*, abs/2009.01797.

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

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

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning.

Nguyen, X., Wainwright, M. J., Jordan, M. I., et al. (2009). On surrogate loss functions and f-divergences. *The Annals of Statistics*, 37(2):876–904.

Nowozin, S., Cseke, B., and Tomioka, R. (2016). f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pages 271–279.

Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

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

Pei, Z., Cao, Z., Long, M., and Wang, J. (2018). Multi-adversarial domain adaptation. *arXiv preprint arXiv:1809.02176*.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. (2019). Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415.

Pentina, A. and Ben-David, S. (2018). Multi-task kernel learning based on probabilistic lipschitzness. In *Algorithmic Learning Theory*, pages 682–701.

Pentina, A. and Lampert, C. H. (2017). Multi-task learning with labeled and unlabeled tasks. In *International Conference on Machine Learning*, pages 2807–2816.

Polyanskiy, Y. and Wu, Y. (2019). Lecture notes on information theory.

Popordanoska, T., Kumar, M., and Teso, S. (2020). Toward machine-guided, human-initiated explanatory interactive learning. *ArXiv*, abs/2007.10018.

Redko, I., Courty, N., Flamary, R., and Tuia, D. (2019). Optimal transport for multi-source domain adaptation under target shift. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 849–858. PMLR.

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

Redko, I., Morvant, E., Habrard, A., Sebban, M., and Bennani, Y. (2020). A survey on domain adaptation theory. *arXiv preprint arXiv:2004.11829*.

Ren, P., Xiao, Y., jun Chang, X., Huang, P.-Y., Li, Z., Chen, X., and Wang, X. (2020). A survey of deep active learning. *ArXiv*, abs/2009.00236.

Roh, Y., Heo, G., and Whang, S. E. (2019). A survey on data collection for machine learning: a big data-ai integration perspective. *IEEE Transactions on Knowledge and Data Engineering*.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

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

Saito, K., Kim, D., Sclaroff, S., Darrell, T., and Saenko, K. (2019). Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8050–8058.

Saito, K., Ushiku, Y., and Harada, T. (2017). Asymmetric tri-training for unsupervised domain adaptation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2988–2997. JMLR. org.

Saito, K., Watanabe, K., Ushiku, Y., and Harada, T. (2018). Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3723–3732.

Sankaranarayanan, S., Balaji, Y., Castillo, C. D., and Chellappa, R. (2018). Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8503–8512.

Sason, I. and Verdú, S. (2015). Upper bounds on the relative entropy and rényi divergence as a function of total variation distance for finite alphabets. In *2015 IEEE Information Theory Workshop-Fall (ITW)*, pages 214–218. IEEE.

Scheffer, T. and Wrobel, S. (2001). Active learning of partially hidden markov models. In *In Proceedings of the ECML/PKDD Workshop on Instance Selection*. Citeseer.

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

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*.

Settles, B. (2012). Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Shalit, U., Johansson, F. D., and Sontag, D. (2017). Estimating individual treatment effect: generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3076–3085. JMLR. org.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. (2017). Wasserstein distance guided representation learning for domain adaptation. *arXiv preprint arXiv:1707.01217*.

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

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sinha, S., Ebrahimi, S., and Darrell, T. (2019). Variational adversarial active learning. *arXiv preprint arXiv:1904.00370*.

Snell, J., Swersky, K., and Zemel, R. (2017). Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087.

Stojanov, P., Gong, M., Carbonell, J. G., and Zhang, K. (2019). Data-driven approach to multiple-source domain adaptation. *Proceedings of machine learning research*, 89:3487.

Sugiyama, M. and Kawanabe, M. (2012). *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., and Hospedales, T. M. (2018). Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208.

Sutskever, I., Martens, J., Dahl, G., and Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147.

Tan, B., Zhong, E., Xiang, E. W., and Yang, Q. (2013). Multi-transfer: Transfer learning with multiple views and multiple sources. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 243–251. SIAM.

Tan, S., Peng, X., and Saenko, K. (2019). Generalized domain adaptation with covariate and label shift co-alignment. *arXiv preprint arXiv:1910.10320*.

Thekumparampil, K. K., Khetan, A., Lin, Z., and Oh, S. (2018). Robustness of conditional gans to noisy labels. In *Advances in neural information processing systems*, pages 10271–10282.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017). Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176.

Urner, R. and Ben-David, S. (2013). Probabilistic lipschitzness a niceness assumption for deterministic labels. In *In Learning Faster from Easy Data - Workshop @ NIPS*.

Urner, R., Wulff, S., and Ben-David, S. (2013). Plal: Cluster-based active learning. In *Conference on Learning Theory*, pages 376–397.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. (2017). Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027.

Villani, C. (2009). The wasserstein distances. In *Optimal Transport*, pages 93–111. Springer.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press.

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

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

Wang, H., Yang, W., Lin, Z., and Yu, Y. (2019b). Tmda: Task-specific multi-source domain adaptation via clustering embedded adversarial training. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 1372–1377. IEEE.

Wasserman, L. (2019). Lecture note: Statistical methods for machine learning.

Weed, J. and Bach, F. (2017). Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. *arXiv preprint arXiv:1707.00087*.

Weed, J., Bach, F., et al. (2019). Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. *Bernoulli*, 25(4A):2620–2648.

Wei, P., Sagarna, R., Ke, Y., Ong, Y.-S., and Goh, C.-K. (2017). Source-target similarity modelings for multi-source transfer gaussian process regression. In *International Conference on Machine Learning*, pages 3722–3731.

Wen, J., Greiner, R., and Schuurmans, D. (2020). Domain aggregation networks for multi-source domain adaptation. *Proceedings of the 37th International Conference on Machine Learning*.

Wen, J., Yu, C.-N., and Greiner, R. (2014). Robust learning under uncertain test distributions: Relating covariate shift to model misspecification. In *International Conference on Machine Learning*, pages 631–639.

Wu, S., Zhang, H., and Ré, C. (2020a). Understanding and improving information transfer in multi-task learning. *ArXiv*, abs/2005.00944.

Wu, X., Guo, Y., Chen, J., Liang, Y., Jha, S., and Chalasani, P. (2020b). Representation bayesian risk decompositions and multi-source domain adaptation. *arXiv preprint arXiv:2004.10390*.

Wu, Y., Winston, E., Kaushik, D., and Lipton, Z. (2019). Domain adaptation with asymmetrically-relaxed distribution alignment. In *International Conference on Machine Learning*, pages 6872–6881.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.

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

Yao, Y. and Doretto, G. (2010). Boosting for transfer learning with multiple sources. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1855–1862. IEEE.

Yin, C., Qian, B., Cao, S., Li, X., Wei, J., Zheng, Q., and Davidson, I. (2017). Deep similarity-based batch mode active learning with exploration-exploitation. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 575–584. IEEE.

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

Yu, D. and Deng, L. (2014). *AUTOMATIC SPEECH RECOGNITION*. Springer.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. (2013). Learning fair representations. In *International Conference on Machine Learning*, pages 325–333.

Zhang, J., Ding, Z., Li, W., and Ogunbona, P. (2018). Importance weighted adversarial nets for partial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8156–8164.

Zhang, K., Schölkopf, B., Muandet, K., and Wang, Z. (2013). Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*, pages 819–827.

Zhang, P., Wang, H., Naik, N., Xiong, C., and Socher, R. (2020a). Dime: An information-theoretic difficulty measure for ai datasets.

Zhang, W., Deng, L., and Wu, D. (2020b). Overcoming negative transfer: A survey. *ArXiv*, abs/2009.00909.

Zhang, Y., Liu, T., Long, M., and Jordan, M. (2019). Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413.

Zhang, Y. and Yang, Q. (2017). A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*.

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

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

Zhao, H., Des Combes, R. T., Zhang, K., and Gordon, G. (2019a). On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pages 7523–7532.

Zhao, H., Zhang, S., Wu, G., Moura, J. M., Costeira, J. P., and Gordon, G. J. (2018). Adversarial multiple source domain adaptation. In *Advances in neural information processing systems*, pages 8559–8570.

Zhao, S., Li, B., Xu, P., and Keutzer, K. (2020). Multi-source domain adaptation in the deep learning era: A systematic survey. *arXiv preprint arXiv:2002.12169*.

Zhao, S., Wang, G., Zhang, S., Gu, Y., Li, Y., Song, Z., Xu, P., Hu, R., Chai, H., and Keutzer, K. (2019b). Multi-source distilling domain adaptation. *arXiv preprint arXiv:1911.11554*.

Zhou, F., Chaib-draa, B., and Wang, B. (2021). Multi-task learning by leveraging the semantic information. *arXiv preprint arXiv:2103.02546*.

Zhou, F., C.Shui, M.Abbasi, Robitaille, L.-E., B.Wang, and Gagné, C. (2020a). Task similarity estimation through adversarial multitask neural network. *IEEE Transactions on Neural Networks and Learning Systems*.

Zhou, F., Shui, C., Huang, B., Wang, B., and Chaib-draa, B. (2020b). Discriminative active learning for domain adaptation. *ArXiv*, abs/2005.11653.

Zhu, Y., Zhuang, F., and Wang, D. (2019). Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5989–5996.