

分类号:

单位代码: 10560

密 级:

汕 頭 大 學
碩 士 學 位 論 文



中文题名: 广义领域自适应算法研究

英文题名: Research on Generalized Domain
Adaptation Algorithms

姓 名 温力胜 学 号 152109092

所在学院 数学与计算机学院 导 师 陈森涛

专 业 计算机技术 合作导师

入学日期 2021 年 10 月 9 日 答辩日期 2024 年 5 月 27 日

论文提交日期 二〇二四年四月二十八日



硕士学位论文

论文中文题目：广义领域自适应算法研究

论文英文题名：Research on Generalized Domain Adaptation
Algorithms

指 导 教 师： 陈森涛

申 请 人： 温力胜

论文答辩委员会成员

摘要

传统的监督学习假设用于训练和测试神经网络模型的数据是独立同分布的，因此从训练数据中得到的网络模型能在测试数据上表现出良好性能。然而，在现实场景中，这个假设往往不成立，导致训练出来的网络模型在测试数据上的性能不佳。近年来，大量的领域自适应研究工作致力于解决这一问题，使得训练（源域）模型能够在非同分布的测试（目标域）数据上保持良好的性能。需要注意的是，这些领域自适应工作中也存在假设，例如假设源域和目标域具有相同的标签空间，即模型在源域训练时所见过的标签也出现在目标域中。然而，在实际情况下，源域和目标域可能具有不同的标签空间，而标签空间的不同可能会降低模型在目标域上的性能，这就需要针对这种情况提出适用算法。因此，本文的研究重点在于放宽领域自适应的约束条件，探索在这些假设不被满足的情况下如何解决相对应的广义领域自适应问题，其中，本文主要研究的广义领域自适应问题包括多源领域自适应、部分领域自适应和开集领域自适应。

本文的主要工作包括：

1) 研究多源领域自适应（Multi-Source Domain Adaptation, MSDA）问题。该问题的主要挑战在于训练数据集来自多个相关但不同的联合分布。为了解决这一挑战，我们提出了一种名为互信息估计和最小化（Mutual Information Estimation and Minimization, MIEM）的算法。该算法在网络特征空间中同时对齐所有源域和目标域的联合分布。值得注意的是，在 MIEM 算法中，互信息的估计问题被转化为凸优化问题，这使得我们可以轻松地找到全局最优解。此外，我们在几个公共数据集上进行了实验，结果表明 MIEM 算法在统计学意义上优于其他对比方法。

2) 研究部分领域自适应（Partial Domain Adaptation, PDA）问题。该问题的主要挑战包括：(i) 源域与目标域的联合分布相关但存在差异；(ii) 源域离群数据的标签不属于目标域标签空间，这些数据对网络模型的学习产生了负面影响。为了解决这些挑战，我们提出了一种名为极大似然权重估计（Maximum Likelihood Weight Estimation, MLWE）的算法，用于估计一个权重函数。该权重函数将源域联合分布的相关部分对齐到目标域联合分布，同时降低源域离群

数据的负面影响。值得注意的是，我们通过最大化似然函数来估计权重函数，这可以导出一个良好的、具有全局最优解的凸优化问题。在实验中，所提出的 MLWE 方法在流行的基准数据集上表现出了优越的性能。

3) 研究开集领域自适应 (Open-Set Domain Adaptation, OSDA) 问题。该问题的目标是将源域模型泛化至包含新类别的目标域。在这个问题中，我们将源域已有的类别称为已知类，而目标域中新出现的类别称为未知类。我们从模型在目标域上的泛化误差出发，展示该误差可由源域分类损失、源域联合分布和目标域已知类联合分布的卡方散度，以及目标域未知类概率损失所控制。基于控制误差的上界，我们提出了联合对齐与未知类概率损失 (Joint Alignment with Unknown Probability Loss, JAUPL) 算法，通过最小化误差上界的经验估计来优化模型。特别地，在算法中，我们使用卡方散度来度量源域联合分布与目标域已知类联合分布之间的差异，并构造出一个带有解析解的凸优化问题。实验结果证明，我们提出的算法取得了优越的效果。

关键词：领域自适应；联合分布对齐；凸优化；极大似然估计；互信息；卡方散度

Abstract

Traditional supervised learning assumes that the data for training and testing neural networks are independent and identically distributed (i.i.d.), the trained networks can have a good performance on the testing data. However, in real-world scenarios, this assumption is not easy to hold, which degrades the performance of the networks during the testing phase. In recent years, lots of Domain Adaptation (DA) works have been conducted to address this problem, enabling the trained networks to perform well on the testing data under such non-identically-distributed setting. However, these DA works based on certain assumptions. For instance, they often assume that the training (source) and test (target) data share an identical class label space, which is restricted and can be violated in practical scenarios. Such deviation may weaken the model's generalization capability in the target domain, necessitating the development of tailored algorithms to handle this scenario. In this paper, we relax the constraints of DA, and investigate Generalized Domain Adaptation problems, including Multi-Source Domain Adaptation, Partial Domain Adaptation, and Open-Set Domain Adaptation.

The main contributions of this paper are as follows:

1) We investigate the Multi-Source Domain Adaptation (MSDA) problem. The main challenge in this problem is that the datasets are generated by relevant but different joint distributions. To address this challenge, we propose an algorithm called Mutual Information Estimation and Minimization (MIEM). To be specific, MIEM aligns the source joint distributions and target joint distribution simultaneously in the network feature space. In MIEM, the estimation of the mutual information is formulated into a convex optimization problem, such that the global optimal solution can be easily found. We conduct experiments on several public datasets, and show that our algorithm statistically outperforms its competitors.

2) We investigate the Partial Domain Adaptation (PDA) problem. There are two main challenges in PDA that weaken the model's performance in the target domain: (i)

the joint distribution of the source domain is related but distinct from that of the target domain, and (ii) the source outlier data, whose labels do not belong to the target label space, have a negative impact on learning the target model. To tackle these challenges, we propose an algorithm called Maximum Likelihood Weight Estimation (MLWE) to estimate a weight function. The weight function aligns the source joint distribution of the relevant part to the target joint distribution, and reduces the negative impact of the source outlier data. To be specific, MLWE maximizes the likelihood function to estimate the weight function, resulting in a nice convex optimization problem with a global optimal solution. Experimental results on popular benchmark datasets demonstrate the superior performance of the proposed MLWE method.

3) We investigate the Open-Set Domain Adaptation (OSDA) problem. The goal of OSDA is to generalize a model from a labeled source domain to an unlabeled target domain, where the target label space contains new classes. Here, the classes present in the source label space are termed known classes, and the newly introduced classes in the target label space are termed unknown classes. We start from the target generalization error, and show that the target error is upper bounded by the source error, the chi-square divergence between the source joint distribution and target known joint distribution, and the target unknown probability loss. Leveraging the error bound, we propose an algorithm called Joint Alignment with Unknown Probability Loss (JAUPL), which trains the model by minimizing the empirical risk of the error bound. Specifically, we employ the chi-square divergence to compare the source joint distributions and target known joint distribution, and construct a convex optimization problem with analytic solution. Experimental results demonstrate the effectiveness of our proposed algorithm.

Keywords: Domain adaptation; Joint distribution alignment; Convex optimization; Maximum likelihood estimation; Mutual information; Chi-square divergence

目录

摘要	1
第 1 章 绪论	1
1.1 研究背景及意义	1
1.2 研究现状	2
1.2.1 领域自适应	2
1.2.2 广义领域自适应	3
1.3 本文内容	5
第二章 多源领域自适应	6
2.1 引言	6
2.2 相关工作	8
2.2.1 领域自适应	8
2.2.2 多源领域自适应	9
2.3 互信息估计和最小化	10
2.3.1 问题定义	10
2.3.2 解决思路	10
2.3.3 互信息估计	12
2.3.4 互信息最小化	14
2.3.5 联合分布对齐	15
2.4 实验	16
2.4.1 实验数据	16
2.4.2 实验设置	17
2.4.3 实验结果	18
2.4.4 统计检验	21
2.4.5 分析实验	22
2.5 本章总结	24
第三章 部分领域自适应	26
3.1 引言	26

3.2 相关工作	29
3.2.1 领域自适应	29
3.2.2 部分领域自适应	30
3.3 极大似然权重估计	31
3.3.1 问题定义	31
3.3.2 解决思路	31
3.3.3 极大似然权重估计	32
3.3.4 模型训练	34
3.4 实验	35
3.4.1 数据集	36
3.4.2 实验细节	36
3.4.3 实验结果	37
3.4.4 实验分析	38
3.5 本章总结	43
第四章 开集领域自适应	45
4.1 引言	45
4.2 相关工作	48
4.2.1 领域自适应	48
4.2.2 开集领域自适应	48
4.3 算法设计	50
4.3.1 问题定义	50
4.3.2 算法设计	50
4.3.3 算法实现	54
4.4 实验	56
4.4.1 实验数据	56
4.4.2 实验设置	57
4.4.3 实验结果	57
4.4.4 统计检验	60
4.5 本章总结	60

第五章 总结与展望	61
5.1 工作总结	61
5.2 未来展望	62
参考文献	65
攻读学位期间主要研究成果	72
一、学术论文	73
致谢	74

第 1 章 绪论

1.1 研究背景及意义

随着深度神经网络架构的进步(例如, ResNet^[1], Transformer^[2]), 机器学习算法在目标分类、语义分割和自然语言处理等诸多应用领域取得了巨大成功。然而, 这些算法的有效性常常建立在一个假设之上, 即用于训练和测试神经网络模型的数据是独立同分布的, 从而在训练数据上得到的网络模型才能在测试数据上也有着良好的性能。然而, 在现实场景中, 训练(源域)数据和测试(目标域)数据通常来自不同的联合分布, 这显著降低了源域模型在目标域上的泛化性能^{[3][4]}。而领域自适应解决了这一问题, 使得模型在非同分布场景下依旧有着良好的泛化性能。

此外, 值得注意的是, 机器学习模型的有效性, 尤其是深度神经网络模型的有效性, 往往依赖于大量的带标签数据。然而, 收集这些带标签数据往往是一项昂贵且耗时的任务, 有时在新的目标领域中是不可行的。领域自适应的重要价值之一就是能够将模型泛化到无标签的领域上, 从而大大减轻了机器学习对于大量带标签数据的依赖程度。

在领域自适应中, 领域被定义为联合分布 $P(x, y)$, 其中, $x \in X$ 为输入特征, $y \in Y$ 为类别标签。在这里, 我们通常会假设我们有一个带标签的源域数据集和一个无标签的目标域数据集, 其中, 带标签的源域数据集由源域联合分布 $P^s(x, y)$ 采样得到, 无标签的目标域数据集由目标域的边际分布 $P^t(x) = \int P^t(x, y)dy$ 采样得到, 而源域与目标域的联合分布之间存在差异, 即 $P^s(x, y) \neq P^t(x, y)$ 。我们希望用这些数据训练一个网络模型, 使得该模型能很好地预测出目标域数据的标签。这个表述构成了领域自适应问题的基础定义, 其中我们进行了一系列假设, 例如假设源域和目标域的特征和标签空间是相同的, 假设源域和目标域的数据集只有一个, 并且数据同时给出^{[5][6][7]}。然而, 在实际情况下, 由于现实场景的复杂性和多样性, 这些假设往往不被完全满足^{[8][9][10]}。在这些情况下, 由于未能解决假设不满足而产生的新问题, 传统的领域自适应方法表现不佳。本文的研究重点在于放宽领域自适应的约束条件, 探索在这些假设不被

满足的情况下如何解决对应的广义领域自适应问题。具体来说，我们研究了以下情况下的广义领域自适应问题：

多源领域自适应^[11]：不再假设源域数据集数量为一，而是考虑多个源域数据集的情况；

部分领域自适应^[8]：不再假设源域与目标域共享相同的标签空间，而是允许源域标签空间包含目标域标签空间的情况；

开集领域自适应^[10]：同样不再假设源域与目标域共享相同的标签空间，但允许源域标签空间是目标域标签空间的子集的情况。

通过研究这三种情况下的广义领域自适应问题，我们旨在为解决实际应用中更加复杂和多样化的情况提供有效的算法。

1.2 研究现状

1.2.1 领域自适应

领域自适应问题的本质在于源域与目标域的联合分布存在差异，这种差异会导致模型在目标域上的性能下降。为了解决这个问题，研究者提出了各种方法来缓解联合分布之间的差异。从概率论的角度来看^[12]，这些方法可以大致划分为三类：1. 边缘分布对齐方法，2. 类条件分布对齐方法，3. 联合分布对齐方法。

早期的领域自适应研究主要集中在基于边缘分布对齐的方法上。这些方法旨在对齐源域和目标域的边缘分布，即 $P^s(x) = P^t(x)$ 。例如，Long 等人^[13]提出了一种方法，将所有特定任务层的隐藏表示嵌入到一个可再现核希尔伯特空间中，并利用最大均值差异（Maximum Mean Discrepancy, MMD）来显式地对齐不同域之间边缘分布的平均嵌入。另外，Ganin 等人^[5]引入了域鉴别器，并使用梯度反转层来最小最大化域鉴别器的适应损失，从而实现了在 \mathcal{H} -散度下对齐两个域之间的边缘分布。Nguyen 等人^[14]则从模型在目标域上的泛化误差出发，推导出一个理论上界，并通过最小化该上界来得到模型，实质上是在 KL 散度下对齐源域边缘表征分布和目标域边缘表征分布。然而，从概率论的角度来看^[12]，一个联合分布可以被拆分为边缘分布与类后验分布的乘积 $P(x, y) = P(x)P(y|x)$ 。因此，仅基于边缘分布对齐的方法，要想完全消除源域与目标域联合分布之间

的差异，还需要假设类后验分布在跨域之间是稳定的，或者对齐域间的类后验分布，即 $P^s(y|x) = P^t(y|x)$ 。否则，在假设不被满足或类后验分布未进行对齐的情况下，这些边际分布对齐方法并不能完全消除源域与目标域联合分布之间的差异。

随后，一些研究工作提出了固定类别标签，对齐该标签下源域与目标域的边际分布的方法，这实现了更加细粒度的领域对齐，即 $P^s(x|y) = P^t(x|y)$ 。例如，Zhu 等人^[15]基于局部最大均值差异，将数据按类别划分为子域，并对齐不同类别之间的跨域相关子域分布。另外，一些工作提出了同时对齐边际分布和类条件分布的方法。例如，Long 等人^[13]在 MMD 度量下，不仅对齐源域和目标域之间的边际分布，还对齐了两个域之间的类条件分布。然而，从概率论的角度来看^[12]，对于同时对齐边际分布和类条件分布的方法来说，由于一个联合分布并不能拆分为边际分布和类条件分布的乘积，即 $P(x, y) \neq P(x)P(x|y)$ 。因此，这些方法并没有真正地对齐源域与目标域之间的联合分布。对于仅对齐类条件分布的方法来说，对齐类条件分布的逻辑似乎与联合分布存在差异的本质问题并不相关，类条件分布的对齐并不意味着联合分布的对齐。虽然这些方法在实践中被证明是有效的，但从理论角度来看，它们并没有很好地解决源域与目标域的联合分布存在差异的本质问题。

一些研究工作基于对齐源域与目标域的联合分布来解决领域自适应问题。例如，Chen 等人^{[16][7][6]}通过散度或距离来衡量两个域之间联合分布的差异，并通过优化投影矩阵或神经网络的特征提取器来最小化这些度量值，从而实现源域与目标域的联合分布对齐。这些基于联合分布对齐的方法在理论上直接解决了领域自适应的核心问题，并且在实验阶段取得了令人满意的性能结果。

1.2.2 广义领域自适应

在放宽领域自适应的约束条件的情况下，涌现了大量新颖的广义领域自适应问题。举例来说，不再假设源域数据集数量为一个，而是考虑多个源域数据集的情况，这便引出了多源领域自适应问题^[11]。另外，不再假设目标域数据集只存在无标签的数据，而是允许存在部分带标签的目标域数据，这就是半监督领域自适应问题^[7]。再者，不再假设源域与目标域共享相同的标签空间，而是允许源域标签空间包含目标域标签空间，这导致了部分领域自适应问题的出现

[8]。另外，同样不再假设源域与目标域共享相同的标签空间，但允许目标域标签空间包含源域标签空间，那么就出现了开集领域自适应问题^[10]。除此之外，还有一些其他的广义领域自适应问题，如异构领域自适应^{[17][18]}、领域泛化^[19]，通用领域自适应^[20]等等。这些新的广义领域自适应问题在实际应用中具有重要意义，对于解决实际场景中的复杂问题有着积极的作用。

但，需要注意的是，放宽领域自适应的约束条件，代表着问题的复杂化，因为除了要解决原本的联合分布差异问题外，还需要处理放宽约束所带来的新问题。

(1) 以多源领域自适应问题为例，多个源域数据集代表着多个不同的源域联合分布，如何处理多个源域联合分布与目标域联合分布之间的差异成为新的问题。为了解决这个问题，Ren 等人^[21]构建了一个伪目标域，并将源域与该伪目标域进行对齐，用于提取多个源域之间的信息，从而提高模型在真实目标域上的性能。Wang 等人^[22]提出聚合从多个源域学习到的知识，以提升对不确定样本的预测，并对齐源域与目标域之间的类条件分布。

(2) 在部分领域自适应问题中，存在与目标域无关的源域离群数据，这些数据可能会降低模型在目标域上的性能。为此，Cao 等人^[8]利用目标域数据标签的预测概率，为源域数据赋予权重，并将这些权重结合域鉴别器来对齐源域与目标域的边际分布。特别地，他们为源域离群数据赋予较小的权重，以确保它们不会对模型的训练产生过大的负面影响。而 Chen 等人^[23]设计了源域的重构误差来选择源域相关数据，并使用这些源域相关数据结合强化学习来训练目标域模型。

(3) 对于开集领域自适应问题，主要挑战在于，模型既要目标域已知类数据进行分类，也要识别目标域未知类数据。为了解决开集领域自适应问题，Satio 等人^[10]利用域对抗模型结合一个固定的阈值来区分已知/未知类的目标样本，并对齐领域之间已知类部分的边际分布。Liu 等人^[24]训练多个二元分类器来评估目标域数据和已知类之间的相似度，为目标域数据赋予权重，并通过对抗性训练对齐源域与加权目标域之间的边际分布。

总而言之，对于不同的广义领域自适应问题，需要具体问题具体分析，提出相应的算法来解决。这需要深入研究每个问题的特点和挑战，并设计出针对性的解决方案，以确保模型在实际应用中的有效性和可靠性。

1.3 本文内容

本文第一章阐述了广义领域自适应算法研究的必要性，并介绍了目前解决领域自适应问题的一些主流方法。在这些方法中，联合分布对齐方法在解决领域自适应的本质问题上有着清晰的逻辑，并展现了强有力的实验结果。

基于此，在接下来的第二章至第四章，我们以联合分布对齐为主要思想，根据当前章节研究的具体问题，进行具体分析，并提出算法来解决这些问题。每一章都会针对不同的广义领域自适应问题进行深入研究，并设计相应的算法。具体而言，第二章将探讨多源领域自适应问题，第三章将研究部分领域自适应问题，而第四章则将专注于开集领域自适应问题。通过这样的章节安排，我们将逐步深入探讨不同类型的广义领域自适应问题，提出相应的解决方法，并通过实验验证所提方法的有效性。

第二章 多源领域自适应

在领域自适应的基础上，本章放松对源域数据集数量的约束，研究广义领域自适应中的多源领域自适应问题。在该问题设定下，我们使用多个带标签的源域数据集和一个无标签的目标域数据集来训练网络模型，并期望训练好的网络模型能够很好地对无标签的目标域数据进行正确分类。这个问题的主要挑战在于训练的数据集来源于多个相关但不同的联合分布。为了应对这一挑战，本章提出互信息估计和最小化（Mutual Information Estimation and Minimization, MIEM）算法。该算法可以在网络特征空间中同时对齐所有源域和目标域的联合分布。值得注意的是，我们在算法中将互信息的估计问题被转化为凸优化问题，这使得我们可以轻松地找到全局最优解。同时，在几个公共数据集上进行了实验，结果表明 MIEM 算法在统计学意义上优于其他对比方法。

2.1 引言

随着深度神经网络架构的进步(例如, ResNet^[1], Transformer^[2]), 机器学习算法在许多应用中取得了巨大的成功, 如目标分类, 语义分割和自然语言处理。假设训练和测试网络模型的数据是独立同分布的, 训练好的网络模型可以在测试数据上表现出良好的泛化性能。然而, 在许多现实场景下, 训练数据和测试数据可能来自不同的分布^{[3][4]}, 这降低了网络模型在测试期间的性能。因此, 近年来, 开发能够在非同分布场景下依旧有着良好泛化性能的算法, 成为机器学习中的一个重要研究课题^{[6][14][21]}。

多源领域自适应 (Multi-Source Domain Adaptation, MSDA) ^[87]正是基于这种非同分布的设置, 其中“领域”代表输入特征 x 和标签类别 y 的联合分布 $P(x, y)$, 或代表由该联合分布生成的数据集。正式地说, 在多源领域自适应中, 训练数据由 $n(n \geq 2)$ 个带标签的源域数据集以及一个无标签的目标域数据集组成, 这些数据集分别来自于 n 个源域联合分布 $P^1(x, y), \dots, P^n(x, y)$ 和目标域的边际分布 $P^t(x)$ ($P^t(x) = \int P^t(x, y) dy$)。源域和目标域的联合分布之间存在联系但也存在区别。多源领域自适应的目标是训练一个能够很好地预测目标域数据标

签的网络模型。根据问题描述，我们可以发现多源领域自适应应与领域自适应（Domain Adaptation, DA）^{[16][25]}和领域泛化（Domain Generalization, DG）^{[26][27]}相关。然而，与领域自适应相比，多源领域自适应的训练数据包含多个带标签的源域数据集，而与领域泛化相比，多源领域自适应的训练数据还包含无标签的目标域数据集。

从统计学角度来看，多源领域自适应的主要挑战在于源域数据集和目标域数据集是从不同的联合分布中采样的，即： $P^1(x, y) \neq \dots \neq P^n(x, y) \neq P^t(x, y)$ 。为了解决这一挑战，大多数现有工作^{[28][29][11][30][18]}提出对齐源域边际分布 $P^1(x), \dots, P^n(x)$ 和目标域边际分布 $P^t(x)$ ，或进一步对齐源域类条件分布 $P^1(x|y), \dots, P^n(x|y)$ 和目标域类条件分布 $P^t(x|y)$ 。由于存在两个以上的边际分布或类条件分布，这些工作提出以成对的方式对齐它们。具体地，这些工作通过矩匹配或对抗性训练对多对分布中的每一对进行对齐。然而，这些工作可能会受到以下限制。首先，由概率的角度出发^[12]，一个联合分布可以被拆分为边际分布和类后验分布的乘积，也就是， $P(x, y) = P(x)P(y|x)$ 。由于对应的源域类后验分布 $P^1(y|x), \dots, P^n(y|x)$ 和目标域类后验分布 $P^t(y|x)$ 没有进行对齐，因此，对齐边际分布并不等同于对齐联合分布。当进一步对齐类条件分布时，联合分布依旧可能没有被对齐。因为一般来说，联合分布不能拆分为边际分布和类条件分布的乘积，即： $P(x, y) \neq P(x)P(x|y)$ ^[6]。其次，以成对的方式对齐分布可能是次优的，特别是对于一些基于对抗性训练的工作^{[30][31]}。因为这些工作需要为多对分布引入带有优化参数的域鉴别器网络，然后经历一个困难的对抗性训练过程^{[32][14]}。

本章通过在网络特征空间中估计和最小化互信息来克服以上限制并解决联合分布的差异问题。我们将算法命名为 MIEM，即互信息估计和最小化(Mutual Information Estimation and Minimization)。互信息是衡量变量之间统计依赖关系的重要指标，并已成功应用于多个机器学习问题，包括聚类，特征选择和无监督表征学习^{[6][29][36]}。在 MIEM 算法中，互信息用于衡量联合变量 $(h(x), y)$ 和领域标签变量 $l \in \{1, \dots, n, t\}$ 之间的依赖关系，其中 h 是生成网络特征空间的网络特征提取器。为了估计互信息，我们将互信息表达为一个泛函的最大值，用样本均值代替泛函中的数学期望，然后用一个线性参数函数作为泛函的输入，并求

解一个易于得到全局最优解的凸问题。请注意，在这里，我们使用 Schölkopf 和 Smola 书中的术语^[36]，将把函数作为输入的函数称为泛函。最小化互信息估计值会使得联合变量独立于领域标签变量。这种独立性直接导致源域和目标域的联合分布在网络特征空间中同时被对齐，并且避免了多对分布的对齐。为了验证算法的有效性，本章在多个图像分类数据集上进行了实验。实验结果表明，该算法的性能在统计意义上优于对比算法。

本章的贡献可以概括如下：

- 通过在网络特征空间中估计和最小化互信息来解决多源领域自适应问题。
- 将互信息的估计问题转化为一个易于找到全局最优解的凸优化问题。
- 在多个图像分类数据集进行了实验及统计检验，验证了算法的有效性。

本章的其余部分组织如下。首先，第 2.2 节介绍相关工作。第 2.3 介绍提出的 MIEM 算法。第 2.4 节设计实验并与其他方法进行对比。第 2.5 节对本章工作进行总结。

2.2 相关工作

2.2.1 领域自适应

领域自适应 (Domain Adaptation, DA) 与多源领域自适应密切相关。在过去的十年间，已经进行了大量的工作来解决这个问题^{[17][40][56][75]}。例如 Ganin 等人^[5]提出在 \mathcal{H} -散度下对齐网络特征空间中的源域边际分布和目标域边际分布。

Acuna 等人^[40]最小化源域平均样本误差以及源域和目标域边际分布之间的 $D_{\mathcal{H}}^{\phi}$ 散度。Rakotomamonjy 等人^[41]最小化加权源域误差和加权边际分布间的 Wasserstein 距离，其中权重基于目标域标签分布。Jing 等人^[42]显式地从原始图像中学习额外的网络特征，以减少领域之间的分布差距，并最小化条件熵损失以避免语义混淆。Ren 等人^[39]通过最小化条件核度量来减少源域类条件分布和目标域类条件分布之间的差异，并保留了源域和目标域的内在结构。Ge 等人^[32]通过条件最大均值差异来对齐源域类条件分布和目标域类条件分布，并通过最大化目标样本与预测标签之间的互信息来提取目标域判别信息。Xu 等人^[43]通过最大化网络特征与类标签之间的互信息来增强特征的可辨别性，并通过最小

化网络特征与领域标签之间的互信息来增强特征的可转移性。**Chen** 等人^[44]优化了深度特征提取器，在相对卡方散度下对齐网络特征空间中的源域联合分布和目标域联合分布，并最小化交叉熵损失，以学习目标模型。

2.2.2 多源领域自适应

近年来，多源领域自适应在机器学习中受到了广泛关注^{[45][29][30]}。为了解决多源领域自适应问题，**Redko** 等人^[45]提出在最优运输框架中学习目标域标签分布，并对齐多个概率分布。**Wen** 等人^[30]证明了目标误差由加权散度控制，并通过优化领域权值和网络模型来最小化散度估计值。每个散度用于衡量对应的源域边际分布和目标域边际分布之间的差异，并被表示为一个新添加的子网络的逻辑损失。**Park** 等人^[29]优化了神经网络模型来最小化源域误差和互信息估计值，其中互信息用于衡量网络特征和领域标签之间的依赖关系。互信息最小化迫使网络特征独立于领域标签，从而同时对齐源域和目标域的边际分布。**Liu** 等人^[46]利用对抗性学习，对多个源域边际分布和目标域边际分布进行对齐，通过最小化类别原型和未标记目标域样本之间的距离，减少了领域之间的类别级差异，并设计了一种实例加权策略来对源域样本进行加权。**Yao** 等人^[18]量化了不同源域的重要性，并在类条件最大均值差异下对齐了源域类条件分布和目标域类条件分布。此外，**Zhao** 等人^[47]为每个源域预训练特征提取器和分类器，进行对抗判别自适应和源域知识蒸馏，并对源域分类器的预测进行加权。

我们的工作与上述大多数多源领域自适应工作不同，我们的工作旨在解决源域联合分布和目标域联合分布之间的差异，而不是源域边际分布和目标域边际分布之间的差异（例如 **Park**，**Wen** 等人的工作^{[29][30]}），也不是源域类条件分布和目标域类条件分布之间的差异（例如 **Yao** 等人的工作^[18]）。我们认为联合分布之间的差异是影响网络模型在目标域性能的关键。为了应对这一挑战，本章提出 **MIEM** 算法。在 **MIEM** 中，我们对互信息的估计被转化为一个很好的凸优化问题，并且我们通过最小化互信息估计值来使得源域联合分布和目标域联合分布同时被对齐。当然，我们注意到 **Park** 等人^[29]的工作也提出了最小化多源领域自适应中的互信息。然而在 **Park** 等人的工作中，最小化互信息是为了对齐边际分布，而不是联合分布。此外，**Park** 等人工作中的互信息估计可能不是一个凸优化问题。我们还注意到，类条件分布对齐方法（例如，**Ge**，**Yao**，**Zhu**

等人的工作^{[32][18][48]}）在实践中已被证明是有效的，并可以产生良好的实验结果。然而，在多源领域自适应问题中，当源域数量和类别数量都增加时，这些工作可能需要去对齐更多的类条件分布组合，每个组合包含更多的分布。在某种程度上，这是具有挑战性的。相比之下，我们最小化互信息使得联合分布对齐的方法更有利，因为无论存在多少源域和类别，我们仅仅需要对齐两个分布，即：在 KL 散度下对齐联合分布 $P(h(x), y, l)$ 和乘积分布 $P(h(x), y)P(l)$ （见等式(2-4)中互信息的定义）。对齐这两个分布可以促使联合变量 $(h(x), y)$ 和领域标签变量 l 之间相互独立，从而使得多个源域联合分布和目标域联合分布被对齐。

2.3 互信息估计和最小化

2.3.1 问题定义

设 x 为输入特征， y 为类别标签。将一个领域定义为一个联合分布 $P(x, y)$ ，或者定义为一个从 $P(x, y)$ 中采样得到的数据集 $D = \{(x_i, y_i)\}_{i=1}^m$ 。在多源领域自适应问题中，训练数据包括 $n(n \geq 2)$ 个带标签的数据集 $\mathcal{D}^1 = \{(x_i^1, y_i^1)\}_{i=1}^{m_1}, \dots, \mathcal{D}^n = \{(x_i^n, y_i^n)\}_{i=1}^{m_n}$ 和一个无标签的数据集 $\mathcal{D}^t = \{x_i^t\}_{i=1}^{m_t}$ 。其中，带标签的数据集分别从 n 个源域联合分布 $P^1(x, y), \dots, P^n(x, y)$ 中采样得到，无标签的数据集从目标域的边际分布 $P^t(x) = \int P^t(x, y)dy$ 中采样得到。源域与目标域的联合分布之间存在联系但不完全相同，即： $P^1(x, y) \neq \dots \neq P^n(x, y) \neq P^t(x, y)$ 。多源领域自适应问题的目标是训练一个能正确地预测目标域数据标签的网络模型 f 。换句话说，训练好的网络模型 f 在目标域上应该有着较小的泛化误差 $\mathbb{E}_{(x,y) \sim P^t(x,y)}[\mathcal{L}(f(x), y)] = \int P^t(x, y)\mathcal{L}(f(x), y)dxdy$ ，其中 \mathcal{L} 是衡量模型预测值 $f(x)$ 和真实标签值 y 之间误差的损失函数。

2.3.2 解决思路

我们从多源领域自适应的目标开始，展示我们是如何解决这个问题的。遵循先前工作中的常见做法^{[44][49][29]}，我们首先将网络模型拆分为特征提取器 h 和特征分类器 g ，即： $f = g \circ h$ 。其中，特征提取器 h 生成网络特征空间。由于源域联合分布和目标域联合分布之间是存在联系的，因此我们可以训练一个特征

提取器 h ，使得这些联合分布在网络特征空间中被对齐，即： $P^1(h(x), y) \approx \dots \approx P^n(h(x), y) \approx P^t(h(x), y)$ 。

因此，我们可以得出以下推导：

$$\begin{aligned} & \mathbb{E}_{(x,y) \sim P^t(x,y)}[\mathcal{L}(f(x), y)] \\ &= \mathbb{E}_{(h(x),y) \sim P^t(h(x),y)}[\mathcal{L}(g(h(x)), y)] \end{aligned} \quad (2-1)$$

$$\approx \frac{1}{n} \sum_{s=1}^n \mathbb{E}_{(h(x),y) \sim P^s(h(x),y)}[\mathcal{L}(g(h(x)), y)] \quad (2-2)$$

$$\approx \frac{1}{n} \sum_{s=1}^n \frac{1}{m_s} \sum_{i=1}^{m_s} \mathcal{L}(g(h(x_i^s)), y_i^s), \quad (2-3)$$

其中等式(2-3)通过样本均值近似等式(2-2)中的数学期望。显然，上述推导表明，当源域联合分布和目标域联合分布在网络特征空间中被对齐时，训练一个在目标域上有着较小泛化误差的网络模型的目标，可以近似转换为训练一个在源域数据集上平均样本误差小的网络模型的实际操作。

现在的困难在于我们如何训练一个特征提取器 h ，使源域和目标域的联合分布在网络特征空间中被对齐，也就是， $P^1(h(x), y) \approx \dots \approx P^n(h(x), y) \approx P^t(h(x), y)$ 。为了解决这个困难，我们引入领域标签 $l \in \{1, \dots, n, t\}$ 的概念，并将源域联合分布分别重新定义为 $P^1(h(x), y) = P(h(x), y|l=1), \dots, P^n(h(x), y) = P(h(x), y|l=n)$ ，将目标域联合分布重新定义为 $P^t(h(x), y) = P(h(x), y|l=t)$ 。在引入领域标签 l 的概念后，我们的目标是训练特征提取器 h ，使得 $P(h(x), y|l=1) \approx \dots \approx P(h(x), y|l=n) \approx P(h(x), y|l=t)$ 。根据概率论知识^[85]，这意味着联合变量 $(h(x), y)$ 和领域标签变量 l 之间相互独立。在这里，我们使用互信息来衡量这两个变量之间的独立关系^{[6][29]}。具体来说，我们将互信息定义为

$$\text{MI}(h(X), Y; L) = \int P(h(x), y, l) \log \left(\frac{P(h(x), y, l)}{P(h(x), y)P(l)} \right) dh(x) dy. \quad (2-4)$$

此时，互信息可以看作是联合分布 $P(h(x), y, l)$ 与乘积分布 $P(h(x), y)P(l)$ 之间的 Kullback-Leibler (KL) 散度。值得注意的是，互信息是非负的，当且仅当联合变量 $(h(x), y)$ 和领域标签变量 l 相互独立时，互信息的值为零。基于这一性质，我们可以从数据样本中估计出互信息，然后训练特征提取器 h 使互信息估计值最

小化，以使得联合变量 $(h(x), y)$ 和领域标签变量 l 相互独立，从而得到 $P(h(x), y|l=1) \approx \dots \approx P(h(x), y|l=n) \approx P(h(x), y|l=t)$ ，即 $P^1(h(x), y) \approx \dots \approx P^n(h(x), y) \approx P^t(h(x), y)$ 。在接下来的两个小节中，我们将介绍互信息估计和互信息最小化的实现细节。

2.3.3 互信息估计

为了估计互信息，我们首先将其转化为：

$$\begin{aligned} \text{MI}(h(X), Y; L) = \max_r \left(\int r(h(x), y, l) P(h(x), y, l) dh(x) dy dl \right. \\ \left. - \int \exp(r(h(x), y, l) - 1) P(h(x), y) P(l) dh(x) dy dl \right). \end{aligned} \quad (2-5)$$

其中，由于互信息 $\text{MI}(h(X), Y; L)$ 是联合分布 $P(h(x), y, l)$ 和乘积分布 $P(h(x), y)P(l)$ 之间的 KL 散度^{[40][50]}，我们可以将互信息转化为 KL 散度的变分表示形式（即等式(2-5)）。当输入函数 $r(h(x), y, l) = 1 + \log(\frac{P(h(x), y, l)}{P(h(x), y)P(l)})$ 时，等式(2-5)中的函数将达到其最大值，即等式(2-4)。

在转化互信息表达式之后，我们发现等式(2-5)中分布的数学期望呈线性。因此，我们可以将数学期望替换为样本均值，并将互信息近似为

$$\begin{aligned} \text{MI}(h(X), Y; L) \approx \max_r \left(\frac{1}{m} \sum_{i=1}^m r(h(x_i), y_i, l_i) \right. \\ \left. - \frac{1}{m^2} \sum_{i,j=1}^m \exp(r(h(x_i), y_i, l_j) - 1) \right). \end{aligned} \quad (2-6)$$

等式(2-6)使用由分布 $P(h(x), y, l)$ 生成的数据集 $\mathcal{D}^{xyl} = \{(h(x_i), y_i, l_i)\}_{i=1}^m$ ，样本数量 $m = m_1 + \dots + m_n + m_t$ 。该数据集由带标签的源数据集 $\mathcal{D}^1, \dots, \mathcal{D}^n$ ，带标签的目标数据集 $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{m_t}$ ，以及领域标签 $l \in \{1, \dots, n, t\}$ 组成。

其定义如下：

$$\mathcal{D}^{xyl} = \{(h(x_i), y_i, l_i)\}_{i=1}^m = (\cup_{s=1}^n \{(h(x_i^s), y_i^s, s)\}_{i=1}^{m_s}) \cup \{(h(x_i^t), y_i^t, t)\}_{i=1}^{m_t}. \quad (2-7)$$

这是因为在 2.3.2 小节，源域联合分布和目标域联合分布已经用领域标签重新定义了。请注意，在这里，我们假设带标签的目标域数据集 \mathcal{D}^t 是可用的，以便我们在这一小节中可以专注于估计互信息。在下一小节中，我们将解释如何从无标签的目标域数据集 $\mathcal{D}^u = \{x_i^t\}_{i=1}^{m_t}$ 构建带标签的目标域数据集 \mathcal{D}^t 。

接下来，我们采用线性参数函数作为输入函数 $r(h(x), y, l)$ 。即，

$$r(h(x), y, l; \theta) = \sum_{i=1}^m \theta_i k(h(x), h(x_i)) \delta(y, y_i) \delta(l, l_i), \quad (2-8)$$

其中 $\theta = (\theta_1, \dots, \theta_m)^T$ 是函数参数， $k(h(x), h(x_i)) = \exp(-\|h(x) - h(x_i)\|^2 / \sigma)$ 是高斯核函数，核宽为 $\sigma (> 0)$ ¹。此外， $\delta(y, y_i)$ 是狄拉克核函数，当 $y = y_i$ 时值为1，否则值为0。 $\delta(l, l_i)$ 也是狄拉克核函数。显然，函数 $r(h(x), y, l; \theta)$ 对其输入变量是非线性的，对参数 θ 中是线性的。非线性使它能够更好地逼近最优函数 $1 + \log(\frac{P(h(x), y, l)}{P(h(x), y)P(l)})$ ，线性可以将等式(2-6)变成一个凸问题，我们将在下面展示这一点。

将等式(2-8)中的函数代入等式(2-6)中，我们可以得到互信息估计值：

$$\begin{aligned} \widehat{MI}(h(X), Y; L) \\ = \max_{\theta} \left(\frac{1}{m} \sum_{i=1}^m r(h(x_i), y_i, l_i; \theta) - \frac{1}{m^2} \sum_{i,j=1}^m \exp(r(h(x_i), y_i, l_j; \theta) - 1) \right) \end{aligned} \quad (2-9)$$

$$= -\min_{\theta} \left(\frac{1}{m^2} \sum_{i,j=1}^m \exp(r(h(x_i), y_i, l_j; \theta) - 1) - \frac{1}{m} \sum_{i=1}^m r(h(x_i), y_i, l_i; \theta) \right) \quad (2-10)$$

$$= \frac{1}{m} \sum_{i=1}^m r(h(x_i), y_i, l_i; \hat{\theta}) - \frac{1}{m^2} \sum_{i,j=1}^m \exp(r(h(x_i), y_i, l_j; \hat{\theta}) - 1). \quad (2-11)$$

其中，等式(2-10)涉及到一个凸优化问题。为了证明这一点，我们需要证明目标函数对 θ 是凸的。由于两个函数 $\exp(r(h(x), y, l; \theta) - 1)$ 和 $-r(h(x), y, l; \theta)$ 对 θ 都为凸，根据文献^[51]，二者之和，即目标函数，对 θ 也是凸的。在等式(2-11)中， $\hat{\theta}$ 是凸问题的全局最优解²。这个全局最优解通过运行 Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS)算法^[52]得到，该算法是解决凸问题的主流算法。为了清晰起见，我们在算法 2-1 中总结了互信息估计过程。

¹ 在实验中，我们将核宽 σ 设置为源域和目标域数据上成对平方距离的中位数。

² 在实验中，我们在等式(2-10)中的目标函数中添加了一个惩罚项 $0.001\|\theta\|^2$ 来避免过拟合。

Algorithm 2-1 互信息估计

Input: 等式(2-7)中定义的数据集 \mathcal{D}^{xy^l} 。

Output: 互信息估计值 $\widehat{\text{MI}}(h(X), Y; L)$ 。

1. 构造等式(2-10)中的凸问题。
 2. 通过 L-BFGS 算法解决凸问题并得到解 $\hat{\theta}$ 。
 3. 通过等式(2-11)得到 $\widehat{\text{MI}}(h(X), Y; L)$ 。
-

2.3.4 互信息最小化

结合在 2.3.2 小节提出的源域平均样本误差，我们可以得到 MIEM 算法的优化问题：

$$\min_{g,h} \frac{1}{n} \sum_{s=1}^n \frac{1}{m_s} \sum_{i=1}^{m_s} \mathcal{L}(g(h(x_i^s)), y_i^s) + \lambda \widehat{\text{MI}}(h(X), Y; L), \quad (2-12)$$

其中 \mathcal{L} 是交叉熵损失， $\lambda(>0)$ 是平衡两种损失的权衡参数。显然，我们的算法训练神经网络 $f = g \circ h$ 来最小化源域平均样本误差和互信息估计值。在 2.3.3 小节中曾提到，互信息的估计涉及到带标签的目标数据集 \mathcal{D}^t 。在这里，我们通过对无标签的目标数据集 $\mathcal{D}^u = \{x_i^t\}_{i=1}^{m_t}$ 运用伪标签技术^{[7][44][39]}来构建带标签的目标数据集 \mathcal{D}^t 。具体来说，我们用训练好的源域网络模型得到预测的目标域数据标签 y_i^t ，将对应的预测标签（伪标签）分配给 \mathcal{D}^u ，得到带标签的目标域数据集 $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{m_t}$ 。为了清晰起见，我们在图 2-1 中展示我们算法中的输入数据(来自 PACS 数据集^[53]的图像)、网络模型和损失项。

为了解决等式(2-12)中的最小化问题，我们采用了 mini-batch Stochastic Gradient Descent (SGD)算法。在 SGD 算法中的每一次小批量迭代中，我们用源域数据集和目标域数据集的小批量数据来计算目标函数。需要注意的是，由于目标域伪标签在一开始并不准确，因此在迭代过程中需要对目标域伪标签进行更新，使其更加准确。为了清晰起见，我们在算法 2-2 中总结了上述过程。

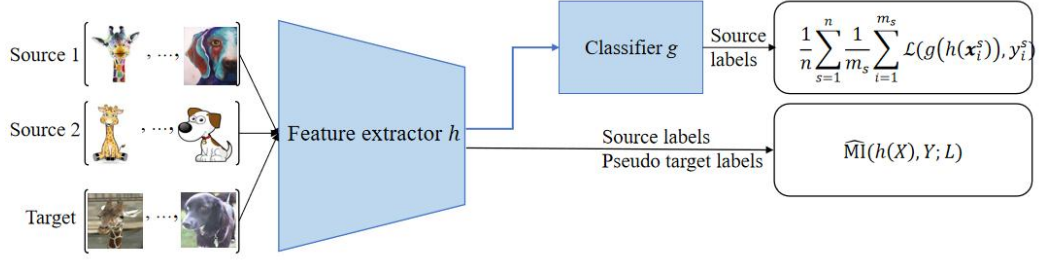


图 2-1 输入数据 (来自 PACS 数据集的图像), 网络模型, 以及我们 MIEM 算法中的损失项。

Algorithm 2-2 互信息估计值最小化

1. Input: 带标签的源域数据集 $\mathcal{D}^1, \dots, \mathcal{D}^n$, 无标签的目标域数据集 \mathcal{D}^u 。
2. Output: 训练好的网络模型 $f = g \circ h$ 。
3. 用 $\mathcal{D}^1, \dots, \mathcal{D}^n$ 训练一个网络模型。
4. 用训练好的网络模型给 \mathcal{D}^u 。打上伪标签, 得到带标签的目标域数据集 \mathcal{D}^t 。
5. while training does not end do
6. for k in 1:K do
7. 分别从数据集 $\mathcal{D}^1, \dots, \mathcal{D}^n, \mathcal{D}^t$ 中抽取 $n + 1$ 个小批量样本。
8. 使用这些样本计算等式(2-12)的目标函数, 其中函数的第二项是通过运行算法 2-1 得到的。
9. 采取梯度下降更新特征提取器 h 和分类器 g 的参数。
10. end for
11. 使用当前网络模型更新 \mathcal{D}^t 中的伪标签。
12. end while

2.3.5 联合分布对齐

除了第 2.3.2 节中“解决思路”外, 这里加入了更多的讨论, 以强化我们在多源领域自适应中的联合分布对齐的思想。

从 Vapnik 的监督学习理论出发^[54], 我们知道, 如果训练联合分布 $P^{tr}(x, y)$ 和测试联合分布 $P^{te}(x, y)$ 是相同的, 即: $P^{tr}(x, y) = P^{te}(x, y)$, 那么一个模型 f 的测试误差 $E_{(x, y) \sim P^{te}(x, y)}[\mathcal{L}(f(x), y)]$ 可以用它在训练数据集 $\mathcal{D}^{tr} = \{(x_i^{tr}, y_i^{tr})\}_{i=1}^{m_{tr}}$ 上的估计误差 $\frac{1}{m_{tr}} \sum_{i=1}^{m_{tr}} \mathcal{L}(f(x_i^{tr}), y_i^{tr})$ 来近似 (该数据集的样本是由训练联合分布 ($P^{tr}(x, y)$) 生成的)。因此, 训练具有较小测试误差的模型的目标可以通过最

最小化模型的训练估计误差(经验风险)来实现。这通常被称为经验风险最小化原则^[54]。相反,如果训练联合分布和测试联合分布不同,即: $P^{tr}(x, y) \neq P^{te}(x, y)$, 那么模型的测试误差不能与模型的训练估计误差联系起来,最小化训练估计误差可能不会导致小的测试误差。

在多源领域自适应中,由于源域和目标域的联合分布不同,因此我们建议将它们对齐,使目标域泛化误差可以近似为源域泛化误差的样本平均估计值。在此基础上,最小化模型的源域泛化误差以降低其目标泛化误差,达到多源领域自适应的目标。在相关的领域自适应文献中^{[55][44]},联合分布对齐从理论和算法的角度被证实是一个根本性的解决方案。我们认为,相比于边际分布对齐或类条件分布对齐,联合分布对齐更好地与 Vapnik 的理论^[54]相连接,在解决问题上的逻辑更清晰。当然,当类先验分布 $P(y)$ 在不同域之间差异很大时,仅通过变换输入特征 x 可能无法很好地对齐联合分布。例如,在部分领域自适应^[7]中的极端情况,其中源域类分布 $P^s(y) > 0$, 而目标域中某些类标签的类分布 $P^t(y) = 0$ 。在这种情况下,除了特征变换,还需要进一步利用加权的思想来增强联合分布对齐。我们将在下一章节展示这一点,将联合分布对齐与加权思想结合以解决部分领域自适应问题。

2.4 实验

2.4.1 实验数据

在实验中,我们使用了 3 个图像分类数据集,这些数据集在以往的工作^{[19][29][11][22][30]}中经常使用。我们将在下文中介绍这些数据集,并在图 2-2 中展示它们的示例图像。

PACS 数据集^[53]包含 7 个类别和 4 个领域: ArtPainting, Cartoon, Photo 和 Sketch。这 4 个领域分别包含 2048、2344、1670 和 3929 张图像。

Office 数据集^[56]包含 31 个类别和 3 个领域: Amazon、DSLR 和 Webcam。这 3 个领域分别包含 2817、795 和 498 张图像。

Office-Home 数据集^[57]包含 65 类别和 4 个领域: Art, Clipart, Product 和 RealWorld。这 4 个领域分别包含 2421、4379、4428 和 4357 张图像。

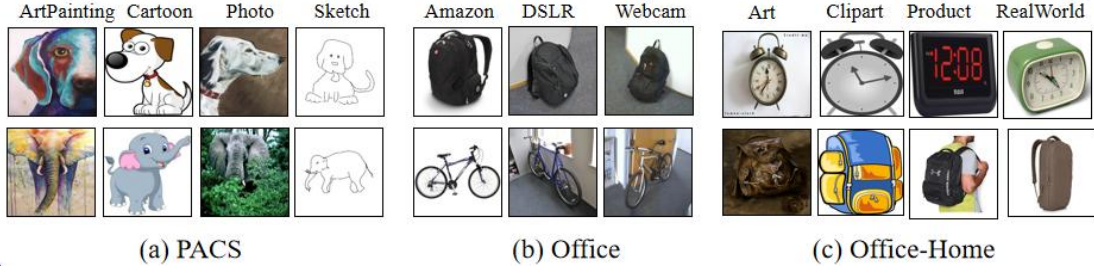


图 2-2 来自 PACS, Office, Office-Home 数据集的样本图像

2.4.2 实验设置

对比算法：遵循以往工作^{[68][69][84]}中采用的实验协议，我们将所提出的 MIEM 算法与基线算法、领域自适应和多源领域自适应算法进行了比较。其中，基线算法将所有源域数据集合并在一起，并在合并后的数据集上训练网络模型，没有使用任何领域自适应操作。领域自适应算法包括 DAN^[38]、DAN^[5]和 DSAN^[15]。这些算法也是在合并后的源域数据集上运行。此外，我们还与最相关的多源领域自适应算法进行了实验对比，这些多源领域自适应算法包括 MDAN^[31]，M3SDA^[11]，MFSAN^[25]，LiC-MSDA^[22]，DARN^[30]和 MIAN^[29]。

实验协议：我们通过以下实验协议来构建多源领域自适应任务，这些协议在以往工作^{[11][29]}中是通用的。即，对于具有多个领域的数据集(例如，PACS)，我们选择其中一个领域作为目标域，其他领域作为源域。一个任务可以被表示为“ \rightarrow ArtPainting”，其中 ArtPainting 表示所选择的目标域。我们将算法在目标域的分类准确率(%)作为其性能指标。在每个任务中，由于不同的随机数种子可能会导致略有不同的分类精度，因此我们重复 3 次实验，并报告平均分类精度。

实验细节：我们在 Pytorch 上进行了实验，神经网络模型框架基于 ResNet18 和 ResNet50^[1]模型。请注意，这些模型已在 ImageNet 上进行了预训练，是多源领域自适应图像分类任务上的常用模型^{[11][22][29]}。在每个图像数据集上，我们使用预训练好的网络模型作为特征提取器，并添加一个特征分类器，其中特征分类器的输出维度与该数据集中的类别数量相同(也就是说，PACS 为 7, Office 为 31, Office- home 为 65)。我们通过 mini-batch SGD 算法优化网络模型(包括特征提取器和特征分类器)参数。其中，特征提取器和特征分类器的学习率分别为 0.001 和 0.01。此外，我们遵循 Ganin 等人^[5]工作中的操作，使用公式 $\lambda_p =$

$\frac{2}{1-e^{-10p}} - 1$ 渐进式地调整权衡参数 λ , 其中 p 表示训练进度, 从 0 到 1 线性变化。
该操作已在相当多的工作验证了有效性^{[19][26][46]}。

2.4.3 实验结果

我们在表 2-1 和表 2-2 中报告了 PACS 上的分类结果(使用 ResNet18 和 ResNet50)。在表 2-3 和表 2-4 报告了在 Office 上的分类结果(ResNet18 和 ResNet50), Office-Home 上的结果(使用 ResNet18 和 ResNet50)报告在表 2-5 和表 2-6。在这些表格中, 因为我们的实验设置与一些工作^{[11][25][29]}是一致的, 因此我们引用了这些对比算法在原始论文中报告的结果。由于部分对比算法在一些数据集上没有报告分类精度结果, 因此, 我们运行文章公开的源代码, 并报告它们在该数据集上的最佳结果。在给定对比算法在所有数据集上的完整结果后, 我们可以将所提的 MIEM 算法与这些对比算法进行综合比较。为了便于比较, 在表中的每一列, 我们将最好的结果以**粗体**突出显示, 并给次好的结果添加下划线。

表 2-1 多源领域自适应任务在 PACS (7 类, ResNet18) 上的分类准确率(%)

Algorithm	→ArtPainting	→Cartoon	→Photo	→Sketch	Avg
Baseline	79.90	74.67	96.53	66.79	79.47
DAN	82.08	77.70	96.83	70.22	81.71
DANN	80.06	80.08	95.57	77.24	83.24
DSAN	83.77	83.76	97.32	77.02	85.47
MDAN	83.76	81.55	96.31	76.68	84.58
M3SDA	85.85	80.11	96.83	77.41	85.05
MFSAN	88.00	85.82	95.93	76.46	86.55
DARN	83.63	78.23	96.67	75.13	83.42
LtC-MSDA	90.19	<u>90.47</u>	97.23	81.53	<u>89.86</u>
MIAN	91.18	85.11	<u>97.30</u>	<u>83.67</u>	89.32
MIEM	<u>90.74</u>	90.67	97.01	90.81	92.31

表 2-2 多源领域自适应任务在 PACS(7 类, ResNet50) 上的分类准确率(%)

Algorithm	→ArtPainting	→Cartoon	→Photo	→Sketch	Avg
Baseline	85.16	76.78	97.94	71.22	82.77
DAN	87.35	83.92	98.32	77.07	86.67
DANN	87.53	84.21	97.64	78.44	86.96
DSAN	92.77	89.38	98.58	85.42	91.54
MDAN	88.14	86.49	98.23	78.92	87.95
M3SDA	91.07	88.13	98.56	<u>86.28</u>	91.01
MFSAN	90.45	88.52	97.17	76.82	88.24
DARN	8.23	84.94	98.17	79.98	87.83
LtC-MSDA	92.85	90.04	97.34	82.72	90.74
MIAN	<u>94.32</u>	<u>90.42</u>	<u>98.71</u>	84.23	<u>91.92</u>
MIEM	94.50	90.96	99.30	89.63	93.60

表 2-3 多源领域自适应任务在 Office(31 类, ResNet18) 上的分类准确率(%)

Algorithm	→Amazon	→DSLr	→Webcam	Avg
Baseline	60.85	98.66	93.58	84.36
DAN	62.01	98.66	95.07	85.25
DANN	61.97	99.06	94.18	85.07
DSAN	63.81	98.52	95.11	85.82
MDAN	62.17	<u>99.79</u>	97.28	86.41
M3SDA	62.52	<u>99.79</u>	97.61	86.64
MFSAN	<u>70.89</u>	99.20	98.28	<u>89.46</u>
DARN	61.93	99.86	97.85	86.55
LtC-MSDA	60.75	98.26	99.80	86.27
MIAN	70.38	99.24	97.77	89.13
MIEM	71.85	99.73	<u>98.49</u>	90.02

表 2-4 多源领域自适应任务在 Office(31 类, ResNet50) 上的分类准确率(%)

Algorithm	→Amazon	→DSLR	→Webcam	Avg
Baseline	66.21	99.59	97.48	87.76
DAN	66.68	99.53	96.98	87.73
DANN	65.77	98.86	95.58	86.73
DSAN	68.82	99.19	97.11	88.37
MDAN	66.02	99.60	97.83	87.81
M3SDA	68.56	100.0	99.06	89.21
MFSAN	72.70	99.50	98.50	90.23
DARN	66.31	<u>99.87</u>	98.64	88.27
LtC-MSDA	68.95	99.62	<u>99.53</u>	89.37
MIAN	<u>74.65</u>	99.48	98.49	<u>90.87</u>
MIEM	75.89	99.84	99.70	91.81

表 2-5 多源领域自适应任务在 Office-Home(65 类, ResNet18) 上的分类准确率(%)

Algorithm	→Art	→Clipart	→Product	→RealWorld	Avg
Baseline	60.90	46.94	74.59	77.39	64.96
DAN	61.74	49.37	74.92	77.34	65.84
DANN	60.26	48.97	73.56	75.74	64.63
DSAN	62.22	50.23	72.59	75.73	65.19
MDAN	60.99	50.68	74.47	77.02	65.79
M3SDA	61.36	49.11	73.94	77.24	65.41
MFSAN	62.57	<u>57.40</u>	<u>77.21</u>	<u>77.58</u>	<u>68.69</u>
DARN	59.97	48.27	74.78	75.78	64.70
LtC-MSDA	<u>62.76</u>	50.08	75.74	77.60	66.55
MIAN	58.48	55.67	74.43	75.25	65.96
MIEM	64.26	59.76	77.46	77.50	69.75

表 2-6 多源领域自适应任务在 Office-Home (65 类, ResNet50) 上的分类准确率(%)

Algorithm	→Art	→Clipart	→Product	→RealWorld	Avg
Baseline	69.23	53.47	79.82	81.73	71.06
DAN	69.32	56.82	78.32	81.32	71.44
DANN	68.94	55.13	79.58	81.37	71.26
DSAN	71.56	59.08	79.86	81.86	73.09
MDAN	69.73	55.06	79.92	81.36	71.52
M3SDA	69.78	57.93	79.12	81.28	72.03
MFSAN	<u>72.10</u>	62.00	80.30	81.80	<u>74.05</u>
DARN	67.88	54.32	79.83	81.43	70.87
LtC-MSDA	71.08	58.21	<u>81.90</u>	<u>82.84</u>	73.51
MIAN	69.39	<u>63.05</u>	79.62	80.44	73.13
MIEM	73.56	65.89	83.22	83.14	76.45

从表 2-1 到表 2-6, 我们发现 MIEM 算法在大多数任务上优于基线、领域自适应和其他多源领域自适应算法。例如, 在表 2-1 中, 我们的算法在平均分类精度方面比基线算法提高了 10% 以上(92.31% 与 79.47%), 比第二好的算法 LtC-MSDA 提高了 2% 以上(92.31% 与 89.86%)。实验结果表明, 在不同类别数 (7、31、65) 的 PACS、Office 和 Office-Home 数据集上, 所提算法在 ResNet18 和 ResNet50 模型上的分类效果均优于对比算法。我们还发现, 将互信息作为正则项的 MIAN 算法取得了很好的分类效果。然而, MIAN 并没有解决联合分布差异 (我们认为这是影响网络模型在目标域性能的关键(请参阅 2.3.2 节中提出的理论动机))。此外, MIAN 可能会遭受一个困难的对抗性训练过程。我们推测这些限制在一定程度上导致了此算法的性能不佳。实验结果表明, MIEM 算法在解决多源领域自适应问题上比这些对比算法具有更大的优势。

2.4.4 统计检验

我们进行了 Holms step-down 测试^[58], 该测试可以验证我们的算法在表 2-1 到表 2-6 的分类准确率方面, 是否在统计意义上优于其他算法。该测试使用一

个统计量 $z = (R_i - R_j) / \sqrt{\frac{k(k+1)}{6N}}$ 来比较第 i 个和第 j 个算法，其中 R_i 是第 i 个算法的平均排名， k 是算法的数量， N 是任务的数量。具体来说，在每个任务中，性能最好的算法排名为 1，性能次好的算法排名为 2，以此类推。我们计算所有算法的平均排名，并在表 2-7 中报告这些平均排名。然后根据统计量 z 的值从正态分布表中搜索概率 $prob$ ，计算 p 的值为 $p = 2(1 - prob)$ 。最后，我们按照 p 值对假设进行升序排序，并在表 2-8 中报告排序结果。从表 2-8 中，我们发现所有的零假设都可以被拒绝，因为它们的 p 值小于 $\alpha/(k - i)$ ，其中 $\alpha = 0.05$ 是一个显著水平。根据文献^[58]，这表明我们的算法在统计意义上优于这些对比算法。

表 2-7 多源领域自适应算法的平均排名

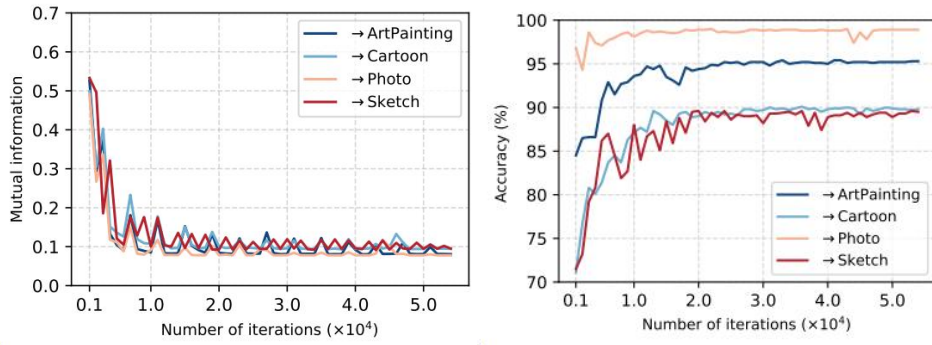
Algorithm	Baseline	DAN	DANN	DSAN	MDAN	M3SDA
Avg Rank	8.86	7.77	9.05	5.59	6.68	5.55
Algorithm	MFSAN	DARN	LtC-MSDA	MIAN	MIEM	--
Avg Rank	4.77	7.05	4.09	5.00	1.57	--

表 2-8 Holms step-down 测试排序结果

i	Algorithm	z	p	$\alpha/(k - i)$	Reject
1	DANN	7.50	0.000	0.005	Y
2	Baseline	7.32	0.000	0.006	Y
3	DAN	6.23	0.000	0.006	Y
4	DARN	5.50	0.000	0.007	Y
5	MDAN	5.14	0.000	0.008	Y
6	DSAN	4.05	0.000	0.010	Y
7	M3SDA	4.00	0.000	0.013	Y
8	MIAN	3.45	0.001	0.017	Y
9	MFSAN	3.23	0.001	0.025	Y
10	LtC-MSDA	2.55	0.010	0.050	Y

2.4.5 分析实验

互信息与分类精度：我们记录了我们的算法（基于 ResNet50）在 PACS 任务迭代过程中的互信息和分类精度值，并在图 2-3(a)和图 2-3(b)上绘制了相应的变化曲线。从图 2-3(a)和图 2-3(b)中，我们观察到，在所有的任务上，随着迭代的进行，互信息的值趋向于越来越小，而分类准确率的值趋向于越来越高。经过足够的迭代次数后，互信息和分类精度都收敛至一个值并趋于稳定。这表明在我们的算法中，互信息最小化对目标分类精度是相关和重要的。



(a) 互信息变化曲线。

(b) 分类精度变化曲线。

图 2-3 在 PACS 任务集上的互信息和分类精度变化曲线(ResNet50)。

伪标签更新：我们在图 2-4 中的柱状图中展示了 MIEM 及其简化版本 MIEM-r 在 Office-Home 上的分类精度，其中 MIEM-r 删除了伪标签更新过程。这两种算法都使用 ResNet50 模型。显然，图 2-4 中的结果表明，MIEM 在所有任务上的性能都比简化版好得多。这表明，更新伪标签对于提高算法的性能是非常必要的。

特征可视化：我们在图 2-5(a) -图 2-5(d)中绘制了由 Baseline, DANN, MIAN 和 MIEM 在任务 “→ Art” (Office-Home)上网络特征的 t-SNE 可视化^[59]结果。所有算法都使用 ResNet50 模型。从图 2-5(a) -图 2-5(d)中，我们观察到我们的 MIEM 算法比这些对比算法更好地减少了源域和目标域之间的差异，并且很好地对齐了源域和目标域。在一定程度上，这有效地解释了我们的算法在分类任务中表现更好的原因。

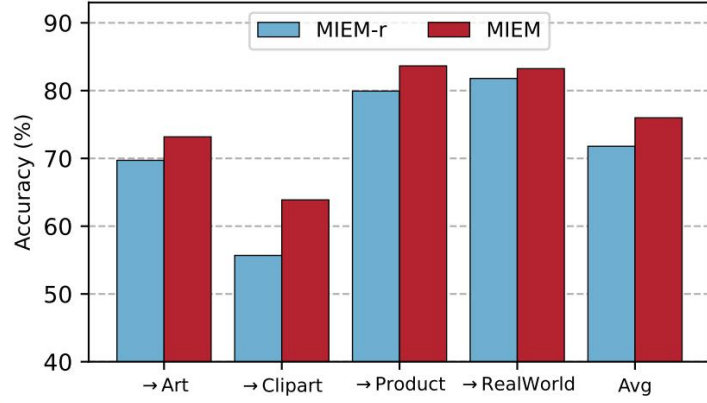


图 2-4 MIEM-r 及其简化版本 MIEM-r 在 Office-Home 任务上的分类精度。其中 MIEM-r 消除了伪标签更新过程。这两种算法都使用 ResNet50 模型。

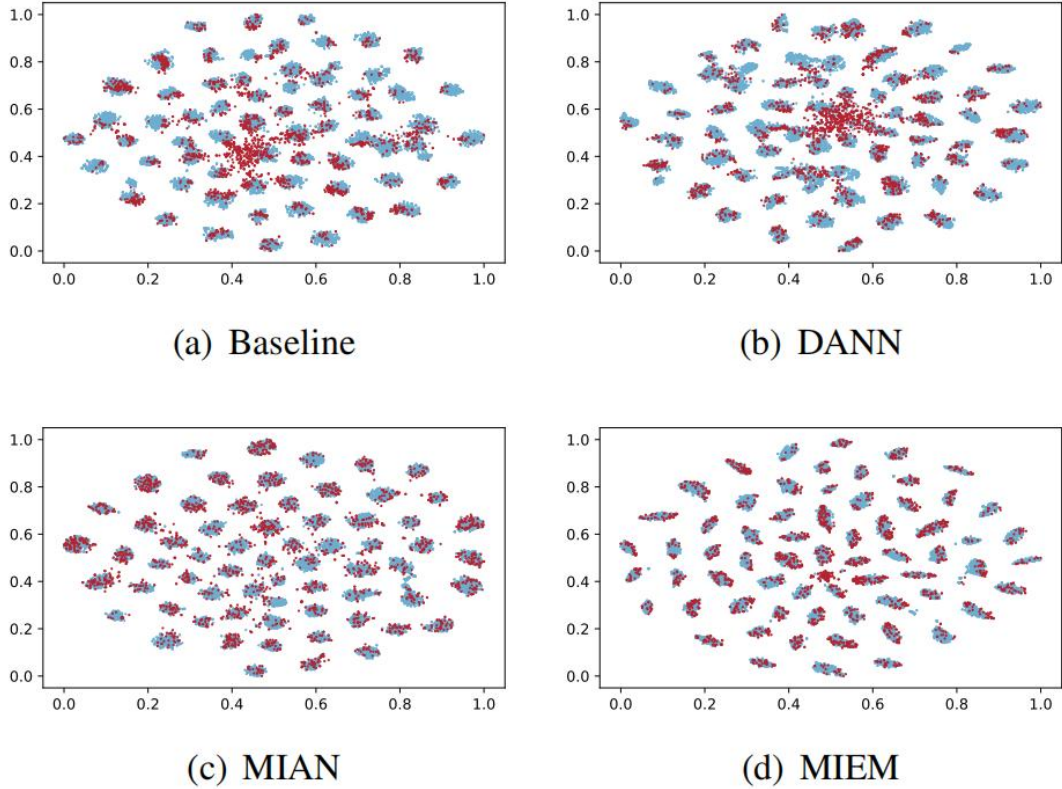


图 2-5 Baseline、DANN、MIAN 和 MIEM 生成的网络特征的 T-SNE 可视化结果。源域是 Clipart、Product 和 RealWorld，颜色为蓝色，目标域是 Art，颜色为红色。所有算法都使用 ResNet50 模型。

2.5 本章总结

本章放松对源域数据集数量的约束，研究多源领域自适应问题。针对多源领域自适应中的联合分布差异挑战，提出了一种互信息估计与最小化 (Mutual

Information Estimation and Minimization, MIEM) 算法。该算法训练多源领域自适应网络以最小化互信息估计值和源域平均样本误差。特别地, 我们将互信息的估计问题转化为一个凸优化问题, 并通过求解该问题得到互信息的估计值。在多个图像分类数据集上的实验表明, 在 ResNet18 和 ResNet50 模型下, MIEM 算法在统计意义上优于其他对比算法。

第三章 部分领域自适应

在领域自适应的基础上，本章放松对标签空间的约束，研究广义领域自适应中的部分领域自适应（Partial Domain Adaptation, PDA）问题。在这里，我们假设源域标签空间包含目标域标签空间。部分领域自适应存在两个主要挑战：(i)源域与目标域的联合分布相关但存在不同；(ii)源域离群数据的标签不属于目标域标签空间，这些数据对分类模型的学习产生了负面影响。为应对这些挑战，本章提出一种极大似然权重估计（Maximum Likelihood Weight Estimation, MLWE）算法来估计一个权重函数。该权重函数将源域联合分布的相关部分对齐到目标域联合分布，降低源域离群数据的负面影响。具体地，我们通过最大化似然函数来估计权重函数，该估计导致一个良好的，具有全局最优解的凸优化问题。在实验中，MLWE 方法在流行的基准数据集上表现出了优越的性能。

3.1 引言

近年来，机器学习在各种实际应用中取得了显著的成功。一般来说，机器学习模型的有效性，特别是深度神经网络模型的有效性，依赖于大量可用的带标签数据。然而，这些带标签数据的收集过程往往是昂贵的，耗时的，在一些新的目标领域上是不可行的。为了减轻对大量带标签数据的依赖，领域自适应（Domain Adaptation, DA）^[60]将分类模型（如深度神经网络）从有标签的源域（源域联合分布 $P^s(x, y)$ ）泛化到无标签的目标领域（目标域联合分布 $P^t(x, y)$ ），其中 x 是输入特征， y 是类标签。源域联合分布 $P^s(x, y)$ 和目标域联合分布 $P^t(x, y)$ 之间的差异是领域自适应中的一个根本问题，它削弱了模型在目标域上的泛化能力^[6]。为了解决这个基本问题，一系列的领域自适应工作^{[58][44][37]}利用源域与目标域联合分布之间的关系，设计各种方法来对齐联合分布。这种联合分布的对齐有助于源域训练模型在目标域上的良好泛化，从而有效地解决领域自适应问题。然而，这些领域自适应工作假设源域和目标域共享相同的标签空间，在实际场景中不一定满足^{[8][61]}。为了放宽源域和目标域共享相同标签空间的假设，近期提出了一种更符合实际情况的领域自适应设定，称为部分领域自适应（Partial Domain Adaptation, PDA）^{[8][61]}。

部分领域自适应的目标是将分类模型从有标签的源域泛化到无标签的目标域，其中源域标签空间包含目标域标签空间。我们将带有不属于目标域标签空间标签的源域数据称为源域离群数据，这些数据与目标域不存在联系。在部分领域自适应中，我们不仅需要缓解领域间的联合分布差异（这是领域自适应中的一个基本问题），而且需要排除这些源域离群数据。这是因为源域离群数据的存在会导致源域联合分布的不相关部分与目标域联合分布错误地进行对齐，从而对目标分类模型的学习造成负面影响。为了解决这个问题，一种策略是在模型训练过程中降低源域离群数据的权重，以减少它们的负面影响。遵循这一策略，现有的部分领域自适应方法^{[8][61][63][66][65][64]}提出先估计源域数据权重，为源域离群数据分配较小的权重，对齐加权的源域边际分布和目标域边际分布，为目标域训练分类模型。例如，Zhang 等人^[67]提出了重要性加权对抗网络（Importance Weighted Adversarial Nets, IWAN），通过对网络模型的特征提取器和鉴别器网络的极大-极小优化来对齐边际分布，并根据另一个鉴别器网络的输出启发式地学习源域数据权重。Li 等人^[68]提出了双对齐（Dual Alignment）算法，引入了一种权重估计方案，用于为源域和目标域数据分配实例级权重和类别级权重，并对齐边际分布。此外，Gu 等人^[66]提出了对抗性重加权（Adversarial Reweighting, AR）算法，通过最小化 Wasserstein 距离来学习实例级权重，对齐加权的源域边际分布与目标域边际分布，并训练加权神经网络对目标域数据分类。有关部分领域自适应方法的更多讨论，请参阅第 3.2.2 小节“相关工作”。

然而，现有的部分领域自适应方法存在两个主要的局限。第一个局限在于分布的对齐。一些部分领域自适应方法通过对齐源域边际分布 $P^s(x)$ 和目标域边际分布 $P^t(x)$ 来解决联合分布差异^{[63][66][69][67]}。由于联合分布 $P(x, y)$ 是边际分布 $P(x)$ 和类后验分布 $P(y|x)$ 的乘积，即， $P(x, y) = P(x)P(y|x)$ ，并且类后验分布 $P(y|x)$ 在跨域之间不一定稳定，因此这些边际分布对齐方法存在联合分布不完全对齐的可能，是次优的^{[19][6][70]}。第二个局限在于源域数据权重的学习方面。在部分领域自适应中，源域数据权重对于降低源域离群数据的负面影响起着至关重要的作用。然而，一些方法以启发式方式估计权重：利用目标域数据的输

出来估计源域数据权重^{[61][68]}。由于目标域数据输出和源域数据权重之间没有明确的联系，这种启发式策略也会导致部分领域自适应的次优结果。

本章提出了一种适用于部分领域自适应的极大似然权重估计（Maximum Likelihood Weight Estimation, MLWE）方法，克服了上述局限。MLWE 引入了一个权重函数来将源域联合分布的相关部分对齐到目标域联合分布。并且，权重函数可以生成合适的权重来排除源域异常数据（参阅 3.4.4 小节“权重可视化与分析”）。为了估计权重函数，我们首先将其设计为一个带参数的指数模型。然后，将加权的源域联合分布近似为目标域联合分布，并通过最大化似然函数来估计权重函数的未知参数。该估计导出一个良好的，具有全局最优解的凸优化问题。最后，通过求解凸问题获得最优解并确定权重函数的未知参数。利用估计的权重函数，我们将源域联合分布的相关部分对齐到目标域联合分布，并通过给源域离群数据分配较小的权重来排除它们。值得一提的是，我们的 MLWE 方法与著名的极大似然估计（Maximum Likelihood Estimation, MLE）方法^[85]存在联系，因为我们的 MLWE 方法和 MLE 方法都将参数估计的似然函数最大化。但是，MLWE 方法是为估计权重函数的参数而设计的，而 MLE 方法是为估计概率分布的参数而设计的。与上述部分领域自适应方法^{[8][63][61][66][65][64]}相比，我们的 MLWE 方法解决了部分领域自适应中的联合分布差异，而不是边际分布差异。并且，我们的 MLWE 方法通过严谨的数学推导而不是启发式策略来估计源域数据权重。

本章的贡献可以概括如下：

- 提出极大似然权重估计（Maximum Likelihood Weight Estimation, MLWE）方法来解决部分领域自适应问题，该方法通过学习一个权重函数，以(i)将源域联合分布的相关部分对齐到目标域联合分布，(ii)减少源域离群数据的负面影响。
- 通过设计指数模型和最大化似然函数来学习权重函数，通过求解一个具有全局最优解的凸优化问题来确定权重函数的最优参数。
- 进行了大量的对比和分析实验，证明了所提出方法的优越性和有效性。在 3.4.3 小节中的对比实验结果表明，该方法在流行的图像分类数据集上优于其他部分领域自适应方法，特别在具有挑战性的大规模数据集 DomainNet

上。在 3.4.4 小节上的分析实验结果表明,该方法在特征学习、权重学习、大规模任务应用等方面都是有效的。

本章的其余部分组织如下。首先,第 3.2 节介绍相关工作。第 3.3 介绍提出的极大似然权重估计方法。第 3.4 节进行对比和分析实验,以验证方法的有效性。第 3.5 节对本章工作进行总结。

3.2 相关工作

3.2.1 领域自适应

领域自适应 (Domain Adaptation, DA) 在假设源域和目标域的标签空间相同的前提下解决了分布差异问题^{[49][37][14]}。这些领域自适应工作通常通过对齐不同领域的分布来减少差异。例如, Ganin 等人^[5]在 \mathcal{H} -散度下对齐两个域之间的边际分布,并被表示为域鉴别器网络的逻辑斯蒂损失。Nguyen 等人^[14]推导了目标域泛化误差的上界,并最小化源域边际表征分布和目标域边际表征分布之间的 KL 散度。Zhang 等人^[71]通过特征正则化和归一化来解决领域自适应问题,提出了可转移的正则化和归一化来对齐特征分布。Ge 等人^[66]对齐跨域之间的类条件分布,通过互信息最大化提取判别信息。Chen 和其他作者^{[16][19][26]}进行了一系列工作来对齐领域自适应和相关的领域泛化(Domain Generalization, DG)问题中的联合分布。这些工作通过使用投影矩阵或神经网络的特征提取器,在 f -散度或 L^p -距离下对齐两个或多个联合分布。该系列工作为解决领域自适应和领域泛化问题提供了清晰的解决逻辑、直接的算法和强有力的实验结果。此外,这些工作还提供了各种基于核的方法来估计源域和目标域数据的 f -散度和 L^p -距离,如此得出的散度/距离估计值可以作为联合分布对齐损失。

我们在部分领域自适应的 MLWE 工作与领域自适应的联合分布对齐工作^{[16][19][26]}相关。然而,本章部分领域自适应的工作不同于这些领域自适应工作,本章工作提出使用一个权重函数,而不是用投影矩阵或神经网络的特征提取器来对齐联合分布。对于目标域标签空间只是源域标签空间的一个子集的部分领域自适应问题,通过权重函数对齐(特征和标签的)联合分布比通过投影矩阵或特征提取器实现对齐要好。事实上,对于标签空间 $Y^t \subset Y^s$ 的部分领域自适应

问题, 优化投影矩阵或特征提取器来对齐联合分布的方法可能是存在问题的, 例如, 优化特征提取器 h 来对齐联合分布 $P^s(x, y)$ 和 $P^t(x, y)$, 即 $P^s(h(x), y) \approx P^t(h(x), y)$ 。在等式的两边对 $h(x)$ 进行积分, 可以得到 $\int P^s(h(x), y) dh(x) \approx \int P^t(h(x), y) dh(x)$, 即 $P^s(y) \approx P^t(y)$, 我们发现这样的优化解决方案导致了一个悖论: 部分领域自适应问题中不存在 $y \in Y^s/Y^t$, 与 $P^s(y) \approx P^t(y)$ 冲突。

3.2.2 部分领域自适应

部分领域自适应专注于源域标签空间包含目标域标签空间的领域自适应场景。因此, 除了解决分布差异问题外, 部分领域自适应工作还需要处理源域离群数据。大量工作^{[7][8][9][10][32][34]}致力于解决部分领域自适应问题。例如, Cao 等人^[8]利用目标域数据标签的预测概率对源域离群数据进行加权, 并将这些权重与源域模型和域鉴别器相结合来训练目标域模型。Chen 等人^[23]先设计了源域的重构误差来选择源域相关数据(其标签属于目标域标签空间的源域数据), 然后用强化学习来对齐不同域之间的边际分布。Li 等人^[73]设计了一种加权损失来识别源域相关数据, 并插入一个残差块来缓解领域间的差异, 减少源域离群数据带来的负迁移问题。Zhang 等人^[74]引入自注意力机制, 增强了以往方法在学习细粒度特征方面的能力, 缓解了源域离群数据的负面影响。He 等人^[72]构建了独立的样本权重机制, 通过流形判别和对抗性学习对齐两个领域之间的数据分布。Cao 等人^[63]估计标签和数据的可迁移概率, 通过标签选择缓解源域离群数据的负面影响, 通过数据选择提升源域相关数据的正面影响。Yang 等人^[69]结合目标域信息学习源域数据权重, 利用对比学习实现跨域分布对齐。Li 等人^[75]设计了对齐两个领域的歧义分数, 并提出了一种加权机制来识别相关类别。

与上述部分领域自适应工作不同, 我们的 MLWE 工作通过学习权重函数 $w(h(x), y)$ 解决问题。该权重函数将源域联合分布的相关部分对齐到目标域联合分布, 为源域数据生成合适的权重, 以排除源域离群数据。特别地, 通过设计指数模型和最大化似然函数来学习权重函数, 从而得到一个具有全局最优解的凸问题。我们相信我们的联合分布对齐和基于严谨的数学推导得到的权重函数是解决部分领域自适应问题的关键。

3.3 极大似然权重估计

3.3.1 问题定义

设定 X 为输入特征空间， Y 为标签空间。根据相关工作^{[44][76][77]}定义，我们将领域定义为联合分布 $P(x, y)$ ，其中 $x \in X, y \in Y$ 。在部分领域自适应中，我们有一个带标签的源域数据集 $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ 和一个无标签的目标域数据集 $\mathcal{D}^u = \{x_i^t\}_{i=1}^{n_t}$ 。这两个数据集分别从源域联合分布 $P^s(x, y)$ 和目标域边际分布 $P^t(x) = \int P^t(x, y)dy$ 采样得到。部分领域自适应的目标是学习一个分类模型 f ，在给定目标域数据输入特征 x^t 的情况下，该分类模型可以准确预测目标域数据的类别标签 y^t 。换句话说，模型 f 在目标域上的泛化误差应该达到最小，即： $\min_f \int P^t(x, y) \mathcal{L}(f(x), y) dx dy$ ，其中 \mathcal{L} 是误差函数。

在部分领域自适应问题中，需要解决两个主要挑战：

- (i)源域联合分布与目标域联合分布存在联系但也存在差异，即： $P^s(x, y) \neq P^t(x, y)$ 。
- (ii)源域标签空间 Y^s 包含目标域标签空间 Y^t ，即： $Y^t \subset Y^s$ ，并且，源域离群数据的标签不属于目标域标签空间，这些数据对学习目标分类模型 f 有负面影响。

3.3.2 解决思路

首先，我们将分类模型 f 实现为一个由特征提取器 h 和特征分类器 g 组成的深度神经网络，即： $f = g \circ h$ 。特征提取器 h 生成网络特征空间。作为 MLWE 方法的关键，我们在网络特征空间中学习一个权重函数 $w(h(x), y)$ ，以解决上述挑战。具体来说，由于源域联合分布与目标域联合分布存在差异，但也存在联系，我们需要缩小这两个联合分布之间的差异并找到联系。为了实现这一点，我们利用权重函数 $w(h(x), y)$ 在网络特征空间中对齐源域联合分布和目标域联合分布，即 $w(h(x), y)P^s(h(x), y) \approx P^t(h(x), y)$ 。此外，对于与目标域联合分布无关的源域离群数据，我们利用权重函数 $w(h(x), y)$ 为其分配较小的非负权重。这些较小的非负权重可以使源域联合分布更接近目标域联合分布。

因此，对于实现部分领域自适应的目标，我们可以将目标域上的泛化误差表示为：

$$\begin{aligned} & \int P^t(x, y) \mathcal{L}(f(x), y) dx dy \\ &= \int P^t(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \end{aligned} \quad (3-1)$$

$$\approx \int w(h(x), y) P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \quad (3-2)$$

$$\approx \frac{1}{n_s} \sum_{i=1}^{n_s} w(h(x_i^s), y_i^s) \mathcal{L}(g(h(x_i^s)), y_i^s). \quad (3-3)$$

等式(3-3)将等式(3-2)中的数学期望替换为样本均值， $w(h(x_i^s), y_i^s)$ 是权重函数在源域数据 (x_i^s, y_i^s) 上的权重值。

根据上述公式，我们将目标域上的泛化误差转换为等式(3-3)中的加权源域分类损失。这种转换使我们能够通过最小化加权源域分类损失来训练目标域模型 $f = g \circ h$ 。由于源数据集 \mathcal{D}^s 是可用的，那么剩下的问题是如何学习和估计权重函数 $w(h(x), y)$ 。在下一小节中，我们将详细说明我们的 MLWE 方法如何估计 $w(h(x), y)$ 。

3.3.3 极大似然权重估计

为了估计权重函数 $w(h(x), y)$ ，我们首先将其设计为以下指数模型：

$$w(h(x), y; \theta) = \frac{\exp(\theta^\top \phi(h(x), y))}{\frac{1}{n_s} \sum_{i=1}^{n_s} \exp(\theta^\top \phi(h(x_i^s), y_i^s))}. \quad (3-4)$$

其中 $\theta = (\theta_1, \dots, \theta_n)^\top$ 是需要学习的模型参数， $\phi(h(x), y) = (k(h(x), h(x_1))\delta(y, y_1), k(h(x), h(x_2))\delta(y, y_2), \dots, k(h(x), h(x_n))\delta(y, y_n))$ 是乘积核向量。第 i 个核中心 $(x_i, y_i) \in \mathcal{D}^s \cup \mathcal{D}^t = \{(x_1^s, y_1^s), \dots, (x_{n_t}^t, y_{n_t}^t)\} = \{(x_i, y_i)\}_{i=1}^n$ ，其中核中心的数量 $n = n_s + n_t$ 。我们在这里假设带标签的目标数据集 $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$ 是可用的，并将在下一小节中解释如何获得它。 $k(h(x), h(x_i)) = \exp(-\|h(x) - h(x_i)\|^2 / \sigma)$ 是核宽度为 σ 的高斯核函数， $\delta(y, y_i)$ 是狄拉克核函数，当且仅当 $y = y_i$ 时值为1，否则为0。请注意，我们在等式(3-4)中设计的权重函数是基于以下考虑。(i)权重函数应该是非负的。(ii)由于 $w(h(x), y) P^s(h(x), y) \approx P^t(h(x), y)$ ，加权源域联合分布 $w(h(x), y) P^s(h(x), y)$ 应该是一个积分为1的概

率分布，也就是说： $1 = \int w(h(x), y) P^s(h(x), y) dh(x) dy \approx \frac{1}{n_s} \sum_{i=1}^{n_s} w(h(x_i^s), y_i^s; \theta)$ 。

(iii)等式(3-4)中的模型能够让我们通过求解凸问题获得 θ 的最优解，我们会在下面展示这一点。

我们通过最大化似然函数来确定权重函数 $w(h(x), y; \theta)$ 的最优参数 $\hat{\theta}$ ，这与著名的极大似然方法 MLE 方法^[85]有关。具体来说，我们解决如下优化问题：

$$\hat{\theta} = \operatorname{argmax}_{\theta} \prod_{i=1}^{n_t} w(h(x_i^t), y_i^t; \theta) P^s(h(x_i^t), y_i^t) \quad (3-5)$$

$$= \operatorname{argmax}_{\theta} \log \prod_{i=1}^{n_t} w(h(x_i^t), y_i^t; \theta) P^s(h(x_i^t), y_i^t) \quad (3-6)$$

$$= \operatorname{argmax}_{\theta} \sum_{i=1}^{n_t} \log (w(h(x_i^t), y_i^t; \theta) P^s(h(x_i^t), y_i^t)) \quad (3-7)$$

$$= \operatorname{argmax}_{\theta} \sum_{i=1}^{n_t} \log w(h(x_i^t), y_i^t; \theta) + \sum_{i=1}^{n_t} \log P^s(h(x_i^t), y_i^t) \quad (3-8)$$

$$= \operatorname{argmax}_{\theta} \sum_{i=1}^{n_t} \log \left(\frac{\exp(\theta^\top \phi(h(x_i^t), y_i^t))}{\frac{1}{n_s} \sum_{j=1}^{n_s} \exp(\theta^\top \phi(h(x_j^s), y_j^s))} \right) \quad (3-9)$$

$$= \operatorname{argmax}_{\theta} \frac{1}{n_t} \sum_{i=1}^{n_t} \theta^\top \phi(h(x_i^t), y_i^t) - \log \left(\frac{1}{n_s} \sum_{i=1}^{n_s} \exp(\theta^\top \phi(h(x_i^s), y_i^s)) \right) \quad (3-10)$$

$$= \operatorname{argmin}_{\theta} \log \left(\frac{1}{n_s} \sum_{i=1}^{n_s} \exp(\theta^\top \phi(h(x_i^s), y_i^s)) \right) - \frac{1}{n_t} \sum_{i=1}^{n_t} \theta^\top \phi(h(x_i^t), y_i^t). \quad (3-11)$$

在等式(3-5)中， $\prod_{i=1}^{n_t} w(h(x_i^t), y_i^t; \theta) P^s(h(x_i^t), y_i^t)$ 是似然函数，代表带标签目标数据集 \mathcal{D}^t 的概率。由于 \log 函数是连续且单调递增的，我们可以直接将其添加到等式(3-5)中，得到等式(3-6)。等式(3-7)和等式(3-8)是通过简单的数学计算得到的，其中等式(3-8)的第二项独立于 θ ，因此可以在优化过程中忽略。等式(3-9)是通过将等式(3-4)中权重函数的指数模型代入等式(3-8)得到的。等式(3-11)将等式(3-10)中的最大化问题转换为最小化问题，这是因为二者的最优解是相同的。观察到等式(3-11)中的目标函数包含指数函数和仿射函数，它们对于 θ 都是凸的，

根据文献^[51]，二者之和，即目标函数，对 θ 也是凸的，等式(3-11)中的最小化问题是凸问题。由于 Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) 算法^[52]广泛用于求解凸问题，本章利用它来获得最优解 $\hat{\theta}$ 。最后，将最优参数 $\hat{\theta}$ 代入等式(3-4)，我们得到权重函数的估计 $w(h(x), y; \hat{\theta})$ 。

为了清晰起见，我们在算法 3-1 中概述了极大似然权重估计过程。

Algorithm 3-1 极大似然权重估计

Input: 带标签的源域数据集 \mathcal{D}^s 和带标签的目标域数据集 \mathcal{D}^t 。

Output: 权重函数的估计 $w(h(x), y; \hat{\theta})$ 。

1. 构造等式(3-11)中的凸问题。
 2. 通过运行 L-BFGS 算法得到最优解 $\hat{\theta}$ 。
 3. 通过等式(3-4)得到 $w(h(x), y; \hat{\theta})$ 。
-

3.3.4 模型训练

在本小节中，我们将介绍 MLWE 算法训练神经网络模型 $f = g \circ h$ 的过程。

我们首先解释上一小节中使用的带标签的目标域数据集 \mathcal{D}^t 是如何获取的。我们采用流行并广泛运用的伪标签技术^{[78][44][77]}为无标签的目标域数据集 $\mathcal{D}^u = \{x_i^t\}_{i=1}^{n_t}$ 分配伪标签 $\{y_i^t\}_{i=1}^{n_t}$ ，得到带标签的目标域数据集 $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$ 。在这里， y_i^t 是通过网络模型 f 得到的预测标签。利用 \mathcal{D}^s 和 \mathcal{D}^t ，我们通过运行算法 3-1 得到估计的权重函数 $w(h(x), y; \hat{\theta})$ 。然后，通过最小化加权源域分类损失和目标条件熵损失来训练模型 $f = g \circ h$ (包含特征提取器 h 和特征分类器 g)。

目标函数表示为：

$$\min_{g,h} \frac{1}{n_s} \sum_{i=1}^{n_s} w(h(x_i^s), y_i^s; \hat{\theta}) \mathcal{L}(g(h(x_i^s)), y_i^s) + \frac{\lambda}{n_t} \sum_{i=1}^{n_t} H(g(h(x_i^t))). \quad (3-12)$$

其中 \mathcal{L} 是交叉熵损失， H 是条件熵损失， $\lambda(>0)$ 是一个权衡参数， $w(h(x_i^s), y_i^s; \hat{\theta})$ 是估计的权重函数在源域数据 (x_i^s, y_i^s) 的权重值。在等式(3-12)中，第一项优化得到跨域预测目标域数据标签的网络模型。第二项常见于一些部分领域自适应工作^{[8][66][64]}，用于促使目标域中标签之间的低密度分离。我们使用

mini-batch SGD 算法对模型进行优化。在 mini-batch SGD 算法的每次迭代过程中，我们对带标签的源数域数据集 \mathcal{D}^s 和无标签的目标域数据集 \mathcal{D}^u 进行小批量采样，并使用这些小批量样本数据计算等式(3-12)中的目标函数，其中每个源域数据 (x_i^s, y_i^s) 与权重值 $w(h(x_i^s), y_i^s; \hat{\theta})$ 相关联。注意到，初始的伪标签可能不那么准确，因此权重函数产生的源域数据权重可能存在噪声。为了解决这个问题，在 Gu 等人工作^{[44][66][77]}的启发下，我们在迭代过程中更新伪标签和权重函数，以提高伪标签和产生的源域数据权重的质量。为清晰起见，我们在算法 3-2 中总结了 MLWE 方法的模型训练过程。

Algorithm 3-2 MLWE 算法的模型训练步骤

Input: 带标签的源域数据集 \mathcal{D}^s ，无标签的目标域数据集 \mathcal{D}^u 。

Output: 训练好的网络模型 $f = g \circ h$ 。

1. 通过设置 $\hat{\theta} = 0$ ，将权重函数 $w(h(x), y; \hat{\theta})$ 产生的权重值初始化为 1。
 2. while training does not end do
 3. for k in 1:K do
 4. 从数据集 $\mathcal{D}^s, \mathcal{D}^u$ 中抽取小批量样本 $\mathcal{D}_k^s, \mathcal{D}_k^u$ 。
 5. 使用这些样本计算等式(3-12)的目标函数，其中 \mathcal{D}_k^s 中的每个数据 (x_i^s, y_i^s) 与权重 $w(h(x_i^s), y_i^s; \hat{\theta})$ 相关联。
 6. 采取梯度下降更新特征提取器 h 和分类器 g 的参数。
 7. end for
 8. 通过当前模型 f 获取带标签的目标域数据集 \mathcal{D}^t 。
 9. 通过运行算法 3-1 更新 $w(h(x), y; \hat{\theta})$ 。
 10. end while
-

3.4 实验

我们在部分领域自适应工作中常用的三个流行的图像分类数据集上评估了 MLWE 算法^{[63][66]}，以证明所算法的优越性能。在接下来的小节中，我们将介绍数据集，描述实验细节，展示实验结果，并进行实验分析。

3.4.1 数据集

Office 数据集^[56]包括 31 个类别，来自 3 个领域：Amazon (A)，DSLR (D)和 Webcam (W)。3 个领域的图像数量分别为 2817、795 和 498。然后，我们从 Office 数据集和 Caltech-256 数据集中共享的 10 个类别中选择图像来创建目标域。一些图像示例展示在图 3-1 中。

Office-Home 数据集^[57]包括 65 个物体的图像，来自 4 个领域：Art(A)，Clipart(C)，Product(P) 和 RealWorld(R)。4 个领域的图像数量分别为 2421,4379,4428 和 4357。然后，我们按字母顺序选择前 25 个类来创建目标域。一些图像示例已展示在图 3-1 中。

DomainNet 数据集^[11]包括 6 个领域和 345 个类。由于一些领域和类中存在噪声，遵循 Gu 等人^[66]的工作，我们选择了 4 个领域和 126 个类。选择的 4 个领域是 Clipart(C)，Painting(P)，Real(R)和 Sketch(S)。4 个领域的图像数量分别为 18703、31502、70358 和 24582。然后，再次遵循 Gu 等人^[66]工作的操作，我们按字母顺序选择前 40 个类来创建目标域。一些图像示例已展示在图 3-1 中。



图 3-1 来自 Office, Office-Home, DomainNet 数据集的样本图像

3.4.2 实验细节

为了估计权重函数，我们将等式(3-11)中的核宽 σ 设置为源域和目标域数据上成对平方距离的中位数，并在等式(3-11)中添加正则化项 $\gamma \|\theta\|^2$ 以避免过拟合。对于小规模任务(例如，来自 Office 数据集的任务)，由于这些任务容易出现过拟合现象，我们将 γ 设置为 0.01。对于其他任务，我们将 γ 设置为 0.001。为了减少在大规模任务上的计算成本(例如，来自 DomainNet 数据集的任务)，我们从合并的源域和目标域数据集中采样一个子集，并使用子集中的数据作为权重函

数的核中心。该子集包含 10000 个数据点。在第 3.4.4 小节“大规模任务的策略消融”中，我们验证了这些策略的可行性。

在模型训练过程中，我们使用 PyTorch 实现我们的 MLWE 方法。具体来说，我们采用 ImageNet 预训练的 ResNet50 网络^[35]作为特征提取器 h ，在特征提取器之后添加特征分类器 g 。特征分类器的输出维度对应于数据集中的类别数量(也就是说, Office 为 31, Office-home 为 65, DomainNet 为 126)。我们使用动量为 0.9 的 mini-batch SGD 算法优化模型的参数。遵循 Ganin 等人^[5]工作的操作，我们通过公式 $\eta_p = \frac{\eta_0}{(1+\alpha p)^\beta}$ 渐进式调整特征提取器 h 的学习率，其中 $\eta_0=0.001, \alpha=10, \beta=0.75$ ，其中 p 表示训练进度，从 0 到 1 线性变化。由于 h 是从 ImageNet 预训练好的， g 是从头开始训练的，所以我们将 g 的学习率设置为 h 的 10 倍。此外，权衡参数 λ 使用公式： $\lambda_p = \frac{2}{1+e^{-10p}} - 1$ 渐进式增调整，其中 p 也表示训练进度。

3.4.3 实验结果

我们将 MLWE 方法与其他部分领域自适应方法进行比较，并在表 4.1，4.2 和 4.3 中报告目标分类精度(%)结果。这些对比方法包括：ResNet50^[1], PADA^[61], DRCN^[73], SAN^[8], AR^[66], ETN^[62], RTNet^[23], DARL^[79], DMP+ent^[80], TSCDA^[81], MSAN+SAN^[74], CSDN^[75], BA3US^[64], SAN++^[63], TRN+SAN^[71], MDPDA^[72], DANN^[5], 和 CDAN+E^[82]。由于我们的实验设置与一些工作一致^{[63][72][66]}，因此我们直接引用了这些工作报告的结果。为清晰起见，我们将部分领域自适应任务表示为 $S \rightarrow T$ ，其中 S 表示源域， T 表示目标域。在表格的每一列中，我们用**粗体**突出显示最好的结果，并在次好的结果下面加下划线。

从表 3-1, 3-2, 3-3 中，我们发现 MLWE 方法在大多数任务上优于对比方法，并在三个数据集上的平均分类准确率达到最高，其中，在 Office 上达到了 98.32%，在 Office-home 上达到了 79.81%，在 DomainNet 上达到了 71.29%。特别是，在具有挑战性的大规模数据集 DomainNet 上，我们的 MLWE 方法在 12 个任务中的 10 个任务上优于次优方法 AR，在平均分类精度上的差距甚至超过 5.5%。此外，我们观察到现有的部分领域自适应方法如 SAN, BA3US 和 TSCDA 在目标域数据集上取得了很好的分类结果。然而，正如在第 3.1 小节“引言”的第三段中提到的，这些方法(i)对齐边际分布而不是联合分布，(ii)通

过一些启发式方法而不是严谨的数学推导来估计源域数据权重，存在局限性。而我们提出的 MLWE 算法解决了重要的联合分布差异问题，并以严谨的数学推导来估计权重函数。因此，MLWE 方法的结果优于其他部分领域自适应方法的结果。总之，实验结果有效地展示了 MLWE 方法在解决部分领域自适应问题上的优势。

表 3-1 部分领域自适应任务在 Office (31 类→10 类) 上的准确率 (%)

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg
ResNet50	83.44	75.59	83.92	96.27	84.97	98.09	87.05
PADA	82.17	86.54	92.69	99.32	95.41	100.0	92.69
DRCN	86.00	88.50	95.60	100.0	95.80	100.0	94.32
SAN	94.27	93.90	94.15	99.32	88.73	99.36	94.96
AR	91.72	97.63	95.62	100.0	95.30	100.0	96.71
ETN	95.03	94.52	96.21	100.0	94.64	100.0	96.73
RTNet	96.20	97.60	92.30	100.0	95.40	100.0	96.92
DARL	98.73	94.58	94.57	99.66	94.26	100.0	96.97
DMP+ent	96.40	96.60	95.10	100.0	95.40	100.0	97.25
TSCDA	98.09	96.84	94.75	100.0	<u>96.00</u>	100.0	97.61
MSAN+SAN	100.0	95.26	95.45	100.0	95.69	100.0	97.73
CSDN	98.73	98.93	94.26	100.0	94.63	100.0	97.76
BA3US	99.36	<u>98.98</u>	94.82	100.0	94.99	98.73	97.81
SAN++	98.09	99.66	94.05	100.0	95.51	100.0	<u>97.89</u>
MLWE	100.0	97.63	<u>96.03</u>	100.0	96.24	100.0	98.32

3.4.4 实验分析

特征可视化：我们在图 3-2 中使用 t-SNE^[54]可视化了由 ResNet50 和我们的 MLWE 在三个任务(Office 上的 Amazon→DSLR, Office-home 上的 Product→RealWorld, DomainNet 上的 Clipart→Sketch)上的网络特征。图中，源域相关数据用红色表示，源域离群数据用蓝色表示，目标域数据用紫色表示。

图 3-2(a)-图 3-2(c)展示了 ResNet50 的结果。我们可以看到源域和目标域并没有很好地对齐，一些目标域数据难以分类。此外，图 3.2(d)-图 3.2(f)展示了我们 MLWE 的结果。可以看出，MLWE 很好地对齐了源域和目标域。通过比较源域相关数据(红色)与源域离群数据(蓝色)和目标域数据(紫色)，可以发现我们的 MLWE 很好地将源域的相关部分与目标域对齐，并排除了源域离群数据。

表 3-2 部分领域自适应任务在 Office-Home (65 类→25 类) 上的准确率(%)

Method	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
ResNet50	46.33	67.51	75.87	59.14	59.94	62.73	58.22	41.79	74.88	67.40	48.18	74.17	61.35
PADA	51.95	67.00	78.74	52.16	53.78	59.03	52.61	43.22	78.79	73.73	56.60	77.09	62.06
SAN	44.42	68.68	74.60	67.49	64.99	77.80	59.78	44.72	80.07	72.18	50.21	78.66	65.30
DRCN	54.00	76.40	83.00	62.10	64.50	71.00	70.80	49.80	80.50	77.50	59.10	79.90	69.05
TRN+SAN	53.34	71.49	81.23	64.46	65.83	75.58	68.87	52.50	81.67	76.22	59.82	79.78	69.23
ETN	59.24	77.03	79.54	62.92	65.73	75.01	68.29	55.37	84.37	75.72	57.66	84.54	70.45
MSAN+SAN	59.28	77.59	82.50	64.00	68.24	75.48	68.87	51.10	83.27	76.78	59.82	82.80	70.81
DARL	55.31	80.73	86.36	67.93	66.16	78.52	68.74	50.93	87.74	79.45	57.19	85.60	72.06
RTNet	63.20	80.10	80.70	66.70	69.30	77.20	71.60	53.90	84.60	77.40	57.90	85.50	72.34
MDPDA	55.82	82.97	86.03	74.20	68.96	80.78	68.22	55.70	83.71	77.04	57.85	83.26	72.86
DMP+ent	59.00	81.20	86.30	68.10	72.80	78.80	71.20	57.60	84.90	77.30	61.50	82.90	73.47
CSDN	57.31	78.10	87.02	70.98	70.08	79.02	75.76	54.93	86.03	79.61	61.25	84.65	73.73
SAN++	61.25	81.57	88.57	72.82	<u>76.41</u>	81.94	74.47	57.73	87.24	79.71	63.76	86.05	75.96
BA3US	60.62	83.16	88.39	71.75	72.79	83.40	75.45	61.59	86.53	79.25	62.80	86.05	75.98
TSCDA	63.64	82.46	89.64	73.74	73.93	81.43	75.36	61.61	87.87	83.56	<u>67.19</u>	88.80	77.44
AR	67.40	<u>85.32</u>	<u>90.00</u>	77.31	70.59	85.15	<u>78.97</u>	<u>64.78</u>	<u>89.51</u>	80.44	66.21	86.44	<u>78.51</u>
MLWE	<u>66.99</u>	86.55	91.11	<u>77.04</u>	79.5	<u>84.82</u>	80.26	65.31	90.12	<u>80.90</u>	67.58	<u>87.51</u>	79.81

表 3-3 部分领域自适应任务在 DomainNet (126 类→40 类) 上的准确率(%)

Method	C→P	C→R	C→S	P→C	P→R	P→S	R→C	R→P	R→S	S→C	S→P	S→R	Avg
DANN	27.83	36.64	29.91	31.79	41.98	36.58	47.64	46.81	40.85	25.82	29.54	32.72	35.68
PADA	22.49	32.85	29.95	25.71	56.47	30.45	65.28	63.35	54.17	17.45	23.89	26.91	37.41
CDAN+E	37.46	48.26	46.61	45.50	60.96	52.63	62.01	60.63	54.74	35.37	38.50	43.63	48.86
ResNet50	41.21	60.01	42.13	54.52	70.80	48.32	63.10	58.63	50.26	45.43	39.30	49.75	51.96
SAN	34.35	51.62	46.23	57.13	70.21	58.25	69.61	67.49	67.88	41.69	41.15	48.44	54.50
BA3US	42.87	54.72	53.79	64.03	76.39	64.69	<u>79.99</u>	<u>74.31</u>	74.02	50.36	42.69	49.65	60.63
AR	<u>52.66</u>	<u>68.24</u>	<u>58.29</u>	<u>66.78</u>	<u>77.53</u>	74.38	76.70	71.77	<u>70.48</u>	<u>53.66</u>	<u>53.60</u>	<u>61.57</u>	<u>65.47</u>
MLWE	60.86	75.51	67.54	72.29	86.45	<u>70.01</u>	81.49	77.10	68.85	70.31	57.46	67.63	71.29

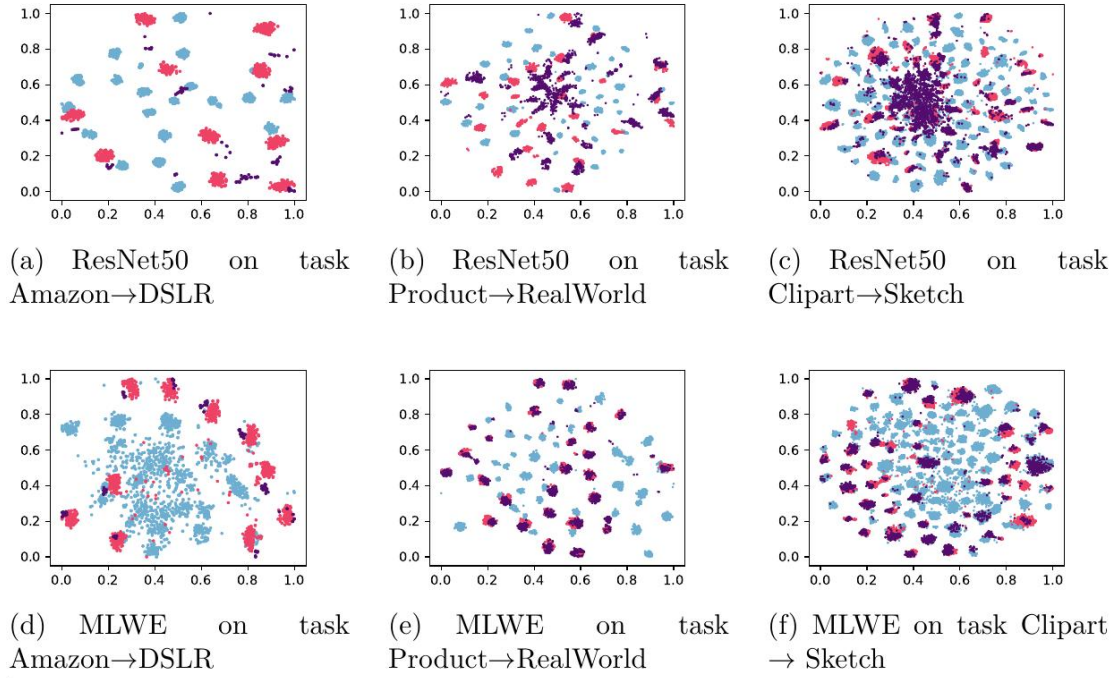


图 3-2 :由 ResNet50 和 MLWE 生成的网络特征的可视化结果。红色的, 蓝色和紫色的点分别表示源域相关数据、源域离群数据和目标域数据。

权重可视化及分析：我们记录了源域数据权重在同一类别标签内的总和，并在图 3-3 中将任务 Art→Product (Office-Home)上的和值进行可视化。源域相关数据权重的和值用红色表示，源域离群数据权重的和值用蓝色表示。显然，源域相关数据被赋予了比源域离群数据更高的权重，表明我们的 MLWE 方法通过为源域离群数据分配小权重(≈ 0)来有效地排除源域离群数据。此外，我们还在图 3-4(a)和图 3-4(b)中分别可视化了迭代过程中的权重比例和分类精度。这里的权重比例被定义为源域离群数据权重之和与所有源域数据权重之和的比例。显然，随着迭代的进行，权重比例趋于下降，分类精度逐渐提高。在 6000 轮迭代后，权重比例和分类精度均趋于收敛。这在一定程度上表明(i)我们的权重函数有效地降低了源域离群数据的权值;(ii)权重比例与分类精度有关，即权值比例越小，分类精度越高。

大规模任务的策略消融：为了降低在大规模任务中权重函数估计的计算成本，我们验证了可用策略在 MLWE 上的可行性。由于权重函数估计的计算成本依赖于核中心的数量，我们用三种可用的，减少了核中心数量的策略来实现 MLWE，包括(i) MLWE-subset:从源域和目标域数据集的组合中采样一个子集，

并使用该子集作为权重函数的核中心，(ii) MLWE-S:使用源域数据作为核中心，以及(iii) MLWE-T:使用目标域数据作为核中心。然后我们在表 3-4 中报告 Office-Home 上的分类精度结果。为了检查这些方法在分类精度结果上是否等效，我们利用分类精度结果执行 Wilcoxon 符号秩测试^[76]，并在表 3-5 中报告测试结果。可见，三种执行不同策略的 MLWE 相对于不使用策略的 MLWE 的 T 值都超过了 13，其中，13 是 Wilcoxon 检验在 12 个任务，置信水平 $\alpha=0.05$ 上的临界值。这表明我们的 MLWE 方法在使用这些策略时，在分类精度方面没有产生统计学上的显著差异。因此，在大规模任务中，将这些策略运用在 MLWE 是可行的，并在统计意义上能产生等效的性能。

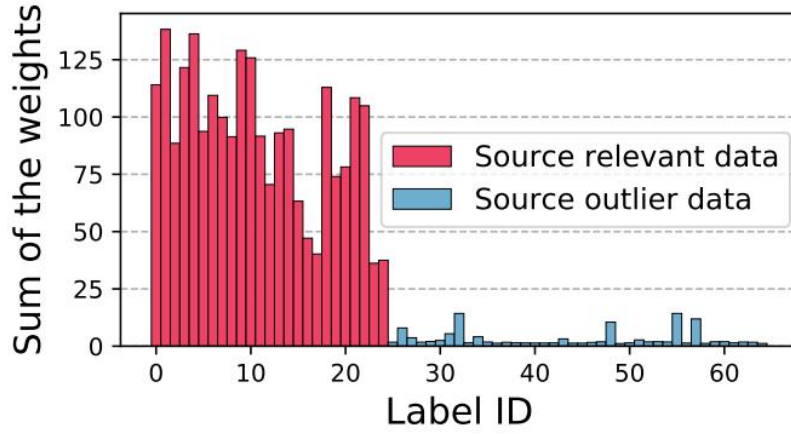
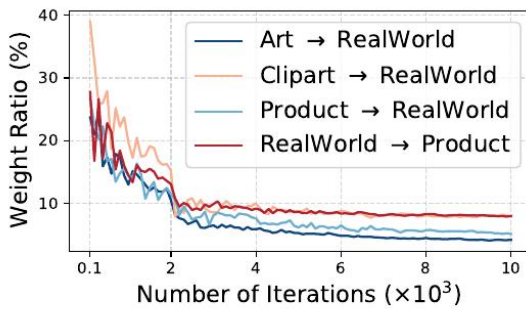
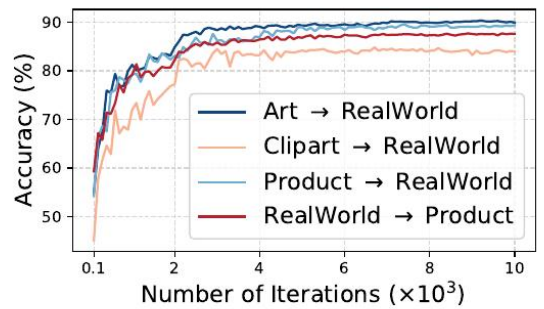


图 3-3 :权重可视化



(a) Variation of weight ratio



(b) Variation of classification accuracy

图 3-4 : (a) Office-Home 任务上的权重比例变化曲线。 (b) Office-Home 任务上的分类精度变化曲线。

表 3-4 Office-Home 上策略消融结果 (65 类→25 类)

Strategy	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
MLWE	66.99	86.55	91.11	77.04	79.55	84.82	80.26	65.31	90.12	80.90	67.58	87.51	79.81
MLWE-subset	64.60	87.06	89.73	77.32	81.40	84.87	78.88	64.48	89.67	81.36	68.24	87.68	79.61
MLWE-S	67.34	87.39	90.01	77.41	81.23	84.04	77.50	67.04	90.12	82.37	68.66	86.16	79.94
MLWE-T	67.04	86.05	90.89	77.78	80.73	83.55	79.16	65.55	89.45	80.44	67.28	87.51	79.62

表 3-5 三个 MLWE 变体与 MLWE 的 Wilcoxon 符号秩检验结果。

Strategy	MLWE-subset	MLWE-S	MLWE-T
T value	35.00	31.50	26.50

方法消融：我们验证了联合分布对齐和基于严谨数学推导的权重估计的贡献。本章将 MLWE 与两种 MLWE 变体进行比较：(i) MLWE-Marginal 进行边际分布对齐而不是联合分布对齐，以及(ii) MLWE-Heuristic 进行启发式权重估计而不是基于严谨数学推导的权重估计。这里，启发式权重估计通过目标域数据来获取源域数据权重，即 $w(y^s) \propto \frac{1}{n_t} \sum_{i=1}^{n_t} g(h(x_i^t))$ ，这常见于之前的部分领域自适应工作[8][68][73]。我们在表 3-6 中报告了 Office 上的比较结果。观察表明，两种 MLWE 变体的性能均优于基线 ResNet50，但我们所提出的 MLWE 比两种 MLWE 变体的性能更好。这表明，在解决部分领域自适应问题时，联合分布对齐和基于严谨数学推导的权值估计比边际分布对齐和启发式权值估计具有更大的优势。

表 3-6 Office 上的方法消融结果 (31 类→10 类)

Method	A→D	A→W	D→A	D→W	W→A	W→D	Avg
ResNet50	83.44	75.59	83.92	96.27	84.97	98.09	87.05
MLWE-Heuristic	96.18	92.54	88.52	99.66	91.75	99.36	94.67
MLWE-Marginal	92.99	91.86	94.78	100.0	95.20	100.0	95.81
MLWE	100.0	97.63	96.03	100.0	96.24	100.0	98.32

性能提升：我们将 MLWE 与标签平滑(Label Smoothing, LS)^[83]相结合以提高性能。标签平滑技术的优点在于，它鼓励数据分布在紧密且均匀的簇中。我们在 Office-Home 上运行 MLWE+LS，并在图 3-5 中将结果可视化。图 3-5 表明，当与其他技术(例如，标签平滑)相结合时，我们的 MLWE 的性能可以进一步提高。

领域自适应：我们展示了 MLWE 方法在领域自适应上能取得与之前的工作相当的表现。表 3-7 给出了 Office-Home 上的分类精度结果。显然，所提出的 MLWE 方法适用于领域自适应问题，并产生了可观的分类结果。我们推测这是因为我们的 MLWE (i)解决了联合分布差异问题，这在领域自适应中是至关重要的，(ii)学习了一个权重函数来排除源域离群数据，这可能也有利于解决领域自适应问题。

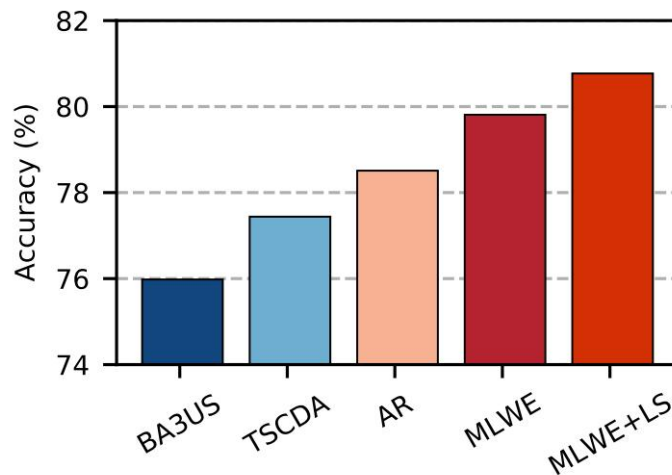


图 3-5:Office-Home 上性能提升的结果

3.5 本章总结

本章提出极大似然权重估计(Maximum Likelihood Weight Estimation, MLWE)算法来解决部分领域自适应问题。该问题放松了对标签空间的约束，并假设源域标签空间包含目标域标签空间，困难在于除了域间联合分布存在差异外，源域离群数据的存在降低了模型的性能。为了解决这些问题，MLWE 通过学习一个权重函数来(i)将源域联合分布的相关部分对齐到目标域联合分布，(ii)排除源域离群数据，以减少其对目标域分类模型的负面影响。特别地，通过设计指数

表 3-7 Office-Home 上领域自适应结果

Method	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
ResNet50 ^[11]	34.90	50.00	58.00	37.40	41.90	46.20	38.50	31.20	60.40	53.90	41.20	59.90	46.10
DANN ^[5]	45.60	59.30	70.10	47.00	58.50	60.90	46.10	43.70	68.50	63.20	51.80	76.80	57.60
JAN ^[13]	45.90	61.20	68.90	50.40	59.70	61.00	45.80	43.40	70.30	63.90	52.40	76.80	58.30
AUDAF ^[84]	47.30	69.20	73.90	51.60	67.20	68.00	53.30	45.90	75.20	61.60	52.70	76.10	61.80
DIAA ^[70]	<u>53.95</u>	76.23	<u>79.11</u>	57.02	71.48	71.44	57.30	50.66	78.73	65.01	56.68	80.90	66.54
DSAN ^[15]	54.40	70.80	75.40	60.40	67.80	68.00	<u>62.60</u>	55.90	78.50	73.80	60.60	<u>83.10</u>	67.60
SBPA ^[85]	53.30	73.80	79.10	65.90	74.10	75.10	65.60	<u>50.80</u>	<u>80.20</u>	73.10	<u>57.40</u>	83.20	69.40
MLWE	50.65	<u>75.20</u>	79.27	<u>64.32</u>	<u>72.64</u>	<u>74.79</u>	61.68	49.44	80.26	<u>73.79</u>	55.30	83.04	<u>68.37</u>

模型和最大化似然函数来估计权重函数，从而得到一个具有全局最优解的凸优化问题。在流行的基准数据集上进行了大量对比和分析实验，结果表明了所提出方法的优越性和有效性。

第四章 开集领域自适应

在领域自适应的基础上，本章放松对标签空间的约束，研究广义领域自适应中的开集领域自适应（Open-Set Domain Adaptation, OSDA）问题。与上一章的部分领域自适应不同，在这里，我们假设目标域标签空间包含源域标签空间。在这个问题中，我们的目标是将源域模型泛化至包含新类别的目标域。我们将源域中已有的类别称为已知类，将目标域中新出现的类别称为未知类。模型既要正确预测目标域中的已知类数据，也要准确识别出目标域中的未知类数据。为了解决开集领域自适应问题，本章从模型在目标域上的泛化误差出发，展示该误差可由源域分类损失、源域联合分布和目标域已知类联合分布的卡方散度、以及目标域未知类概率损失所控制。根据控制误差的上界，我们提出了联合对齐与未知类概率损失（Joint Alignment with Unknown Probability Loss, JAUPL）算法。该算法通过最小化误差上界的经验估计来优化模型，在这个过程中，源域分类损失的最小化可以得到一个在源域上性能良好的模型，卡方散度的最小化对齐了源域和目标域已知类的联合分布，使得源域模型能够泛化至目标域的已知类部分，而目标域未知类概率损失的最小化则保证了模型在未知类数据上的识别能力。特别地，在算法中，我们使用卡方散度来度量源域联合分布与目标域已知类联合分布之间的差异，并构造出一个带有解析解的凸优化问题。通过对比实验和统计检验，证明了所提方法的有效性。

4.1 引言

基于训练数据与测试数据独立同分布的假设，传统的监督学习可以通过经验风险最小化从训练数据中得到一个在测试数据上也有良好性能模型。然而，在实际应用中，训练数据与测试数据往往来自不同的概率分布，这种情况下模型的性能会显著下降^{[3][4]}。为了解决这个问题，出现了许多领域自适应工作^{[6][14][21]}，其目的是在非同分布的情况下将从训练（源域 $P^s(x, y)$ ）数据上得到的模型有效地泛化到测试（目标域 $P^t(x, y)$ ）数据上，其中 x 是图像特征， y 是类别标签。为了应对更加复杂且多样化的现实应用场景需求，近些年来，出现了一

种被称为开集领域自适应（Open-Set Domain Adaptation, OSDA）^[10]的新颖的领域自适应问题。

在开集领域自适应中，源域模型需要泛化至包含新的标签类别的目标域，这些新标签类别在源域标签空间中是未见过的，因此我们称其为未知类。此时，我们将源域中已有的类别称为已知类。在开集领域自适应问题中，模型的任务是在目标域中对已知类数据进行正确预测，并将所有未知类数据统一识别为单一未知类^{[10][36][86]}。值得注意的是，源域和目标域的联合分布存在差异，即： $P^s(x, y) \neq P^t(x, y)$ ，这是领域自适应中的一个基本问题。为了解决这个问题，一系列领域自适应方法^{[7][76][77]}通过对齐领域之间的联合分布来优化模型。然而，在开集领域自适应中，这些方法错误地将目标域未知类联合分布与源域联合分布对齐，从而降低了模型在目标域上的泛化性能。因此，这些领域自适应方法并不能直接应用于开集领域自适应问题，它们还需要处理目标域中的未知类数据。

开集领域自适应的主要挑战在于如何有效地识别目标域中的未知类数据，并解决源域和目标域联合分布之间的差异。为了应对这些挑战，一些研究工作提出了结合开集识别（Open Set Recognition, OSR）^{[89][88]}技术来识别目标域中的未知类数据，并在浅层网络中对齐源域和目标域已知类的分布。近些年来，深度神经网络由于其能够学习到更具有可转移性的特征，因此一些研究工作将深度神经网络引入到开集领域自适应问题的解决方案中。这些方法结合熵值^[90]、对抗性训练^{[91][10]}、前门调整^[92]和旋转识别^[93]等技术，以阈值界限^[10]或者权重赋予^{[90][24]}的方式对已知/未知类数据进行划分，并且对齐源域和目标域已知类的分布。例如，Liu 等人^[88]优化浅层网络以对齐源域与目标域已知类的分布，增加类内的紧凑性，区分不同的已知类和区分未知类。并且，使用了 Open Set Nearest Neighbor（一种 OSR 技术）以获取初始已知/未知类信息。Satio 等人^[10]提出训练特征提取器，使目标域数据被分类为未知类的概率偏离预定义的阈值界限，并以对抗性训练对齐源域与目标域之间的分布。Jang 等人^[90]利用熵值结合后验推理计算目标域数据权值（权值代表该数据属于未知类的概率），通过域鉴别器对齐源域分布和目标域已知类的分布，并同时分离目标域未知类的分布。有关开集领域自适应方法的更多讨论，请参阅第 4.2.2 小节“相关工作”。

虽然这些工作已被证实是实际有效的，但这些工作存在三个缺陷。第一，这些工作提出对齐源域与目标域已知类的边际分布或类条件分布^{[24][88][10]}。对于边际分布对齐，从概率论^[12]的角度来说，我们知道一个联合分布可以被拆分为边际分布与类后验分布的乘积，即： $P(x, y) = P(x)P(y|x)$ ，由于类后验分布 $P(y|x)$ 在跨域之间不一定稳定，因此这些边际分布对齐方法存在联合分布不完全对齐的可能。(ii)对于类条件分布对齐，可以理解为将类别标签 y 固定，对齐该标签下源域与目标域的边际分布。虽然一系列基于类条件分布对齐思想的工作取得了不错的实验结果，但对齐类条件分布的逻辑似乎与联合分布存在差异的本质问题并不相关，类条件分布的对齐并不意味着联合分布之间的差异被消除。第二，一些工作引入了域鉴别器，以对抗性训练对齐领域间的分布，这可能会经历一个具有挑战性的极大-极小优化过程^{[30][31]}。第三，已存在工作^{[24][10]}大多以启发式方法划分未知/已知类数据，但缺乏对这些方法进行理论分析的支持，这可能导致一个次优的结果。

在这项工作中，我们克服以上缺陷，并从理论到算法上解决了开集领域自适应问题。我们从模型在目标域上的泛化误差出发，展示了该误差可由：源域分类损失，源域联合分布和目标域已知类联合分布之间的卡方散度，以及目标域未知类概率损失，这三项所控制。因此，我们基于控制误差的上界提出联合对齐与未知类概率损失（Joint Alignment with Unknown Probability Loss, JAUPL）算法，通过最小化误差上界的经验估计来优化模型，使该模型在目标域上有着较小的泛化误差，得到期望的目标模型。值得注意的是，在误差上界最小化中，源域分类损失的最小化保证了模型在源域上的良好性能。卡方散度的最小化对齐了源域联合分布和目标域已知类联合分布，使得源域模型能够有效地泛化到目标域已知类部分，对目标域已知类数据正确分类。目标域未知类概率损失的最小化使得模型能够有效地识别未知类数据。值得注意的是，我们使用卡方散度来比较两个联合分布，并将散度估计问题转化为带有解析解的凸问题。我们在两个基准数据集上进行了对比实验及统计检验，实验结果证明了我们的算法的有效性。

本章的贡献可以概括如下：

- 证明了开集领域自适应问题的目标域泛化误差可由源域分类损失，源域联

合分布和目标域已知类联合分布之间的卡方散度，以及目标域未知类概率损失所控制。

- 根据误差上界，提出算法，通过最小化误差上界的经验估计来优化模型，解决开集领域自适应问题。
- 将卡方散度的估计问题转化为一个带有解析解的凸问题，并通过最小化该估计值对齐源域联合分布和目标域已知类联合分布。
- 进行对比实验和统计检验，在统计学意义上证明了所提方法的有效性。

4.2 相关工作

4.2.1 领域自适应

在领域自适应 (Domain Adaptation, DA) 问题中，源域和目标域共享相同的标签空间，问题的难点在于源域与目标域服从于不同的联合分布。近年来出现了大量的领域自适应工作^{[6][5][37][38]}。一部分工作通过对齐领域间的边际分布来解决问题，例如 Ganin 等人^[5]提出在 \mathcal{H} -散度下，利用域鉴别器对齐网络特征空间中源域和目标域的边际分布。Long 等人^[58]在多个任务特定层下，最小化源域和目标域边际分布之间的多核 MMD (MK-MMD)。Nguyen 等人^[14]推导出目标域上泛化误差的上界，在 KL 散度下对齐源域边际表征分布和目标域边际表征分布。一部分工作对齐了领域间的类条件分布，例如 Zhu 等人^[15]基于局部最大均值差异 (LMMD)，按类别划分子域，并对齐跨域间的相关子域分布。一部分工作对齐了联合分布，例如 Chen 和其他作者^{[16][19][26]}在 f -散度或 L^p -距离下对齐联合分布。这部分联合分布对齐的工作提供了直接的算法来解决领域自适应的难点——源域联合分布和目标域的联合分布之间存在差异，并得到了强有力的实验结果。

4.2.2 开集领域自适应

在开集领域自适应中，目标域存在未知类，并且这些未知类不存在于源域。因此，在开集领域自适应中，不仅需要解决领域间的联合分布差异问题，还需要对未知类数据进行处理。大量工作^{[93][79][36][24][74]}致力于解决开集领域自适应问题。例如，Satio 等人^[10]利用域对抗模型结合一个固定的阈值来区分已知/未知

类的目标样本，并对齐领域间已知类部分的边际分布。Shermin 等人^[36]扩展了上述的对抗模型，引入加权模块，根据域特征来为目标域数据分配权重。Liu 等人^[24]训练多个二元分类器来评估目标域数据和已知类之间的相似度，为目标域数据赋予权重，并通过对抗性训练对齐源域与加权目标域之间的边际分布。Chang 等人^[79]训练样本分离网络以分离已知类和未知类的样本，训练分布对齐网络以增强源域和目标域之间的域混淆，并结合相互学习模块，使两个网络训练模块相互积极受益。Bucci 等人^[93]利用旋转不变性，分阶段地分离已知/未知类数据以及缓解领域间的分布差异。Jang 等人^[90]以训练过程中产生的熵值作为开集识别的指标，结合后验推理计算目标域数据属于未知类的概率，以对抗性方法训练网络模型来对齐源域和目标域已知类部分的边际分布，同时对目标域未知类部分的分布进行分离。

注意到，Fang 等人^[89]引入开放集差分以衡量目标域模型对未知类的风险，为开集领域自适应提供了理论上界，通过最小化上界的经验估计，可以训练一个浅层网络模型来实现目标。然而，将模型扩展为深度神经网络时，开放集差分将收敛于负值，从而导致大多数目标域数据被识别为未知类。因此，Zhong 等人^[86]为开放集差分加入了阈值界限，放松了误差上界，避免了开放集差分过小的问题，并通过对抗性训练对齐源域类条件分布与目标域已知类部分的类条件分布，使得模型能够正确分类目标域已知类数据并识别目标域未知类数据。

我们的工作与上述大多数开集领域自适应工作不同，我们的工作从目标域泛化误差出发，提出一个新的理论上界，该上界包含源域分类损失、卡方散度以及目标域未知类概率损失三项。从而能够让我们通过最小化误差上界的经验估计来训练模型，从理论到算法上解决开集领域自适应问题。值得注意的是，不同于上述边际分布对齐或类条件分布对齐的工作，我们方法对齐的是联合分布，这与领域自适应中的本质问题（源域与目标域之间的联合分布存在差异）有着更为直接的联系。与大多基于对抗性训练来对齐分布的工作对比，我们在对齐分布这块将卡方散度的估计问题转化为一个带解析解的凸问题，而无需引入额外的域鉴别器以及无需经历一个困难的极大-极小的对抗训练过程，我们的方法因此可能会取得更优的效果。同时，与同样提出理论上界解决开集领域自适应的工作相比，Zhong 等人^[86]对齐了类条件分布而不是联合分布，引入了域

鉴别器带来了更多需要优化的网络参数，从而导致网络模型更加复杂化，并经历了一个困难的极大-极小的对抗训练过程。

4.3 算法设计

4.3.1 问题定义

将一个领域定义为一个联合分布 $P(x, y)$ ，其中， $x \in X$ 为输入特征， $y \in Y$ 为类别标签。在开集领域自适应问题中，我们将源域标签空间 Y^s 中的类别称为已知类，将目标域标签空间 Y^t 中新出现的类别 $Y^t \setminus Y^s$ 归为单一的未知类 unk 。我们有一个带标签的源域数据集 $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ 和一个无标签的目标域数据集 $\mathcal{D}^u = \{x_i^t\}_{i=1}^{n_t}$ 。其中，带标签的源域数据集由源域联合分布 $P^s(x, y)$ 采样得到，无标签的目标域数据集由目标域的边际分布 $P^t(x) = \int P^t(x, y)dy$ 采样得到。开集领域自适应的目标是用这些数据训练一个网络模型 f ，并期望 f 在目标域上能够(i)正确分类已知类数据，(ii)准确识别未知类数据。即：训练好的网络模型 f 在目标域（包含已知类和未知类）上有着较小的泛化误差 $\int P^t(x, y)\mathcal{L}(f(x), y)dxdy$ ，其中 \mathcal{L} 是衡量模型预测值和真实值误差的损失函数。

在开集领域自适应问题中，我们需要解决两个主要挑战：(i)源域与目标域的联合分布之间存在联系但不完全相同，即： $P^s(x, y) \neq P^t(x, y)$ 。(ii)如何保证模型在未知类数据上的识别能力。

4.3.2 算法设计

根据目标域数据所属类别，我们可以将标签空间划分为已知类标签空间 Y^s ，和未知类标签空间 unk 。因此，对于模型在目标域上的泛化误差 $\int P^t(x, y)\mathcal{L}(f(x), y)dxdy$ ，可以将其拆分为(i)当目标域数据为已知类 Y^s 时，模型的泛化误差，即目标域已知类泛化误差 $\int_{X \times Y^s} P^t(x, y)\mathcal{L}(f(x), y)dxdy$ ，(ii)当目标域数据为未知类 unk 时，模型的泛化误差，即目标域未知类泛化误差 $\int_{X \times unk} P^t(x, y)\mathcal{L}(f(x), y)dxdy$ 。也就是说：

$$\int P^t(x, y)\mathcal{L}(f(x), y)dxdy$$

$$= \int_{X \times Y^s} P^t(x, y) \mathcal{L}(f(x), y) dx dy + \int_{X \times \text{unk}} P^t(x, y) \mathcal{L}(f(x), y) dx dy. \quad (4-1)$$

在接下来的子节，我们将用联合对齐与未知类概率损失分别对这两项进行约束及优化。

4.3.2.1 联合对齐

我们首先将模型 f 拆分为特征提取器 h 和特征分类器 g ，即： $f = g \circ h$ 。其中，特征提取器 h 将输入特征 x 映射到网络特征空间，得到 $h(x)$ 。作为 JAUPL 算法的关键之一，我们在网络特征空间中对齐源域联合分布与目标域已知类联合分布，使得训练好的网络模型在目标域已知类部分上表现出良好的性能。这是因为，我们在定理 4.1 中证明了在温和的假设条件下，目标域已知类泛化误差受源域分类损失、源域联合分布和目标域已知类部分联合分布之间的卡方散度控制。

定理 4.1：假设存在 $M > 0$ ，使得 $\mathcal{L}(\cdot, \cdot) \leq M$ ，存在 $\epsilon > 0$ ，使得卡方散度 $\chi_{X \times Y^s}^2(P^s(h(x), y), P^t(h(x), y)) \leq \epsilon$ ，那么，对于任意 f ，都有：

$$\begin{aligned} & \int_{X \times Y^s} P^t(x, y) \mathcal{L}(f(x), y) dx dy \\ & \leq \int_{X \times Y^s} P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy + \frac{\sqrt{2}M}{\sqrt{\epsilon}} \chi_{X \times Y^s}^2(P^s(h(x), y), P^t(h(x), y)). \end{aligned} \quad (4-2)$$

证明：请参考附录 1。

定理 4.1 表明，为了使目标域已知类泛化误差小，我们可以最小化(i)源域分类损失，(ii)源域联合分布和目标域已知类联合分布之间的卡方散度。其中，源域分类损失的最小化可以得到在源域上有着良好泛化性能的模型，卡方散度的最小化在网络特征空间中对齐了源域联合分布与目标域已知类部分的联合分布，使得源域模型在目标域已知类部分上也能有着良好的泛化性能。因此，为了得到一个能对正确分类目标域已知类数据的模型，我们可以最小化源域分类损失和卡方散度的经验估计。

其中，源域分类损失的经验估计为：

$$\int_{X \times Y^s} P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \approx \frac{1}{n_s} \sum_{i=1}^{n_s} \mathcal{L}(g(h(x_i^s)), y_i^s) \quad (4-3)$$

为了得到卡方散度的经验估计，我们可以将其表述为：

$$\begin{aligned} & \chi_{X \times Y^s}^2(P^s(h(x), y), P^t(h(x), y)) \\ &= \int_{X \times Y^s} \left[\left(\frac{P^s(h(x), y)}{P^t(h(x), y)} \right)^2 - 1 \right] P^t(h(x), y) dh(x) dy \end{aligned} \quad (4-4)$$

$$= \max_r \int_{X \times Y^s} \left(2 \frac{P^s(h(x), y)}{P^t(h(x), y)} r(h(x), y) - r(h(x), y)^2 - 1 \right) P^t(h(x), y) dh(x) dy \quad (4-5)$$

$$= \max_r \int_{X \times Y^s} (2P^s(h(x), y)r(h(x), y) - P^t(h(x), y)r(h(x), y)^2 - P^t(h(x), y)) dh(x) dy \quad (4-6)$$

$$\approx \max_r \left(\frac{2}{n_s} \sum_{i=1}^{n_s} r(h(x_i^s), y_i^s) - \frac{1}{n_t} \sum_{i=1}^{n_t} r(h(x_i^t), y_i^t)^2 - 1 \right). \quad (4-7)$$

等式(4-4)是卡方散度的原始表示形式。基于等式 $p^2 = \max_r(2pr - r^2)$ ，将 $\frac{P^s(h(x), y)}{P^t(h(x), y)}$ 视为 p ， $r(h(x), y)$ 视为 r ，我们可将卡方散度表达为一个泛函（以函数为输入的函数）的最大值，即等式(4-5)。当输入函数 $r(h(x), y) = \frac{P^s(h(x), y)}{P^t(h(x), y)}$ 时，等式(4-5)中的函数将达到其最大值，即等式(4-4)。通过简单的数学计算可以推导出等式(4-6)。使用带标签的源域数据集 $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ 和带标签的目标域数据集 $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$ ，我们可以把等式(4-6)中数学期望替换为样本均值，得到等式(4-7)。在这里，我们假设 \mathcal{D}^t 是可用的，并在第4.3.3小节“算法实现”小节中说明我们是如何获得可用的 \mathcal{D}^t 。

接下来，我们采用线性参数函数作为输入函数 $r(h(x), y)$ 。即，

$$r(h(x), y; \theta) = \sum_{i=1}^n \theta_i k(h(x), h(x_i)) \delta(y, y_i), \quad (4-8)$$

其中 $\theta = (\theta_1, \dots, \theta_m)^T$ 是函数参数， $k(h(x), h(x_i)) = \exp(-\|h(x) - h(x_i)\|^2 / \sigma)$ 是核宽为 $\sigma(>0)$ 的高斯核函数， $\delta(y, y_i)$ 是狄拉克核函数，当 $y = y_i$ 时值为1，否则值为0。第 i 个核中心 $(x_i, y_i) \in \mathcal{D}^s \cup \mathcal{D}^t = \{(x_1^s, y_1^s), \dots, (x_{n_t}^t, y_{n_t}^t)\} = \{(x_i, y_i)\}_{i=1}^n$ ，核中心的数量 $n = n_s + n_t$ 。注意到，函数 $r(h(x), y; \theta)$ 对其输入变量 $(h(x), y)$ 是非

线性的，对函数参数 θ 是线性的。非线性使它能够很好地逼近最优函数 $\frac{P^s(h(x), y)}{P^t(h(x), y)}$ ，

线性可以将等式(4-7)变成一个凸问题，我们将在下面展示这一点。

将等式(4-8)中的线性参数函数代入等式(4-7)中，我们可以得到：

$$\widehat{\chi_{X \times Y^s}^2}(P^s(h(x), y), P^t(h(x), y)) \approx \max_{\theta} \left(\frac{2}{n_s} \sum_{i=1}^{n_s} r(h(x_i^s), y_i^s; \theta) - \frac{1}{n_t} \sum_{i=1}^{n_t} r(h(x_i^t), y_i^t; \theta)^2 - 1 \right) \quad (4-9)$$

$$= \max_{\theta} \left(\frac{2}{n_s} \mathbf{1}^\top G^s \theta - \frac{1}{n_t} \theta^\top G^t{}^\top G^t \theta - 1 \right) \quad (4-10)$$

$$= \max_{\theta} (2b^\top \theta - \theta^\top H \theta - 1) \quad (4-11)$$

$$= 2b^\top \hat{\theta} - \hat{\theta}^\top H \hat{\theta} - 1. \quad (4-12)$$

其中，代入的线性参数函数将等式(4-7)中一个泛函的最大化问题转换为等式(4-9)中线性函数的最大化问题。在等式(4-10)中， $\mathbf{1}^\top$ 是一个元素均为1的 n_s 维行向量， G^s 是一个 $n_s * n_s$ 维的矩阵，其第 (i, j) 个元素 $g_{ij}^s = k(h(x_i^s), h(x_j^s)) \delta(y_i^s, y_j^s)$ ， G^t 是一个 $n_t * n_t$ 维的矩阵，其第 (i, j) 个元素 $g_{ij}^t = k(h(x_i^t), h(x_j^t)) \delta(y_i^t, y_j^t)$ 。等式(4-11)引入了两种符号 $b = \frac{1}{n_s} G^s \mathbf{1}$ ， $H = \frac{1}{n_t} G^t{}^\top G^t$ 。通过求解等式(4-11)中的凸问题，我们可以得到一个解析解 $\hat{\theta} = (H + \epsilon I)^{-1} b$ 。注意，为了确保矩阵 H 的数值在计算中始终是可逆的，我们在解析解中添加了一个对角矩阵 ϵI ，其中 ϵ 是一个很小的正值， I 是单位矩阵。

Algorithm 4-1 卡方散度的经验估计

Input: 带标签的源域数据集 \mathcal{D}^s 和带标签的目标域数据集 \mathcal{D}^t 。

Output: 卡方散度估计值 $\widehat{\chi_{X \times Y^s}^2}(P^s(h(x), y), P^t(h(x), y))$ 。

1. 从 \mathcal{D}^s 和 \mathcal{D}^t 中构造核矩阵 G^s ， G^t
 2. 计算得到 b ， H 和 $\hat{\theta}$ 。
 3. 通过等式(4-12)得到卡方散度的估计值 $\widehat{\chi_{X \times Y^s}^2}(P^s(h(x), y), P^t(h(x), y))$ 。
-

为了清晰起见，我们在算法 4-1 中概述了联合对齐中卡方散度的经验估计过程。

4.3.2.2 未知类概率损失

对于目标域未知类泛化误差，我们有以下推导：

$$\begin{aligned} & \int_{X \times \text{unk}} P^t(x, y) \mathcal{L}(f(x), y) dx dy \\ &= \int_X P^t(x, \text{unk}) \mathcal{L}(f(x), \text{unk}) dx \end{aligned} \quad (4-13)$$

$$= \int_X P^t(x) \mathcal{L}(f(x), \text{unk}) P^t(\text{unk}|x) dx \quad (4-14)$$

$$\approx \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}(f(x_i^t), \text{unk}) P^t(\text{unk}|x_i^t). \quad (4-15)$$

其中，等式(4-13)是在目标域未知类泛化误差上类别标签 y 积分的结果。根据概率论知识^[12]，我们可以将目标域未知类联合分布 $P^t(x, \text{unk})$ 拆分为边际分布 $P^t(x)$ 与类后验分布 $P^t(\text{unk}|x)$ 的乘积，得到等式(4-14)中的目标域未知类概率损失。将等式(4-14)中的数学期望替换为样本均值，得到目标域未知类概率损失的经验估计，即等式(4-15)。

注意到目标域未知类概率损失的经验估计，即等式(4-15)，涉及到目标域数据属于未知类的概率值 $P^t(\text{unk}|x_i^t)$ 。在这里，根据模型在 softmax 层的输出，我们将输出中对应未知类的维度数值作为 $P^t(\text{unk}|x_i^t)$ 。具体地说，假设已知类包含 K 个类别，在我们的算法中，模型包含 softmax 层，其输出维度被定为 $K+1$ ，数据 x 在 softmax 层输出的第 k 维的值可看作该数据属于第 k 类的概率值 $P^t(y = k|x) = \frac{\exp(l_k)}{\sum_{k=1}^{K+1} \exp(l_k)}$ ，其中 l 表示模型 softmax 层输入的对数向量， l_k 为向量第 k 维的值。

4.3.3 算法实现

在本小节中，我们将介绍 JAUPS 算法训练神经网络模型的过程。

我们首先解释上一小节中使用的带标签的目标域数据集 \mathcal{D}^t 是如何获取的。我们采用了一种流行且广泛应用的伪标签策略^{[18][20][75]}，为无标签的目标数据集

$\mathcal{D}^u = \{x_i^t\}_{i=1}^{n_t}$ 分配了伪标签 $\{y_i^t\}_{i=1}^{n_t}$ ，从而得到了带标签的目标数据集 $\mathcal{D}^t = \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$ 。在这里，伪标签是指网络模型 f 的预测标签。借助于这种策略，我们通过运行算法 4-1 获得了源域联合分布与目标域已知类联合分布之间的卡方散度估计值。然后，通过最小化源域分类损失的经验估计、卡方散度估计值以及目标域未知概率损失的经验估计来训练模型。

目标函数表示为：

$$\begin{aligned} \min_{g,h} \frac{1}{n_s} \sum_{i=1}^{n_s} \mathcal{L}(g(h(x_i^s)), y_i^s) + \lambda_1 \widehat{\chi_{X^s \times Y^s}^2}(P^s(h(x), y), P^t(h(x), y)) \\ + \frac{\lambda_2}{n_t} \sum_{i=1}^{n_t} \mathcal{L}(g(h(x_i^t)), unk) P^t(unk|x_i^t). \end{aligned} \quad (4-16)$$

其中 \mathcal{L} 是交叉熵损失， $\lambda_1(>0)$ 和 $\lambda_2(>0)$ 是两个权衡参数。其中，第一项源域分类损失最小化可以得到在源域上有着良好性能的网络模型，第二项卡方散度最小化对齐了源域联合分布与目标域已知类联合分布，使得源域模型能够有效地泛化到目标域已知类部分，对目标域已知类数据正确分类，第三项目标域未知类概率损失最小化使得模型能够有效地识别未知类数据。

在模型训练的初始阶段，我们通过源域分类损失结合熵值判别对模型进行预训练，以获取初始的已知/未知类信息来启动算法。这是由于在缺乏未知类数据的前验信息时，模型对于未知类输出的概率值为零。具体而言，我们根据熵值来识别目标域中的未知类数据，并将这部分数据与源域数据一起用于网络模型的预训练。在这里，熵值代表着模型输出的不确定性。对于目标域数据而言，较高的熵值通常是由于不确定的分类情况造成的；而由于未知类中缺乏先验的训练样本，未知类数据往往会导致更高的熵值^[90]。因此，在模型预训练阶段，我们将熵值大于源域数据最大熵值的目标域数据视为未知类数据，并让其参与模型的预训练，以获取初始的未知类信息。尽管这种划分方式并不准确，但能够获取到部分未知类信息，足以让我们结合源域分类损失最小化训练出一个具备一定分类能力的网络模型。在算法的后续正式训练阶段，模型将逐步迭代优化已知/未知类信息。

为清晰起见，我们在算法 4-2 中总结了 JAUPL 算法的模型训练过程。

Algorithm 4-2 JAUPL 算法的模型训练过程

Input: 带标签的源域数据集 \mathcal{D}^s , 无标签的目标域数据集 \mathcal{D}^u

Output: 训练好的网络模型 $f = g \circ h$ 。

1. 利用熵值判别未知类数据, 结合源域分类损失得到预训练模型。
 2. while training does not end do
 3. 通过当前模型 f 获取带标签的目标域数据集 \mathcal{D}^t 。
 4. for k in 1:K do
 5. 从数据集 $\mathcal{D}^s, \mathcal{D}^t$ 中抽取小批量样本 $\mathcal{D}_k^s, \mathcal{D}_k^t$ 。
 6. 使用这些样本, 运行算法 4-1 得到卡方散度估计值, 通过等式(4-3)(4-15)得到源域分类损失和目标域未知类概率损失的经验估计。
 7. 组合得到等式(4-15)的目标函数。
 8. 采取梯度下降更新特征提取器 h 和特征分类器 g 的参数。
 9. end for
 10. end while
-

4.4 实验

我们在两个流行的图像分类数据集上评估了 JAUPL 算法, 以证明所提算法的优越性能。在接下来的小节中, 我们将介绍数据集, 描述实验设置, 展示实验结果, 进行统计检验。

4.4.1 实验数据

Office-Home 数据集^[57] 包含 65 类别和 4 个领域: Art(A), Clipart(C), Product(P)和 RealWorld(R)。这 4 个领域分别包含 2421、4379、4428 和 4357 张图像。遵循标准实验协议, 我们按字母顺序选择前 25 类作为目标域的已知类, 26-65 类作为未知类。

DomainNet 数据集^[11] 包括 6 个领域和 345 个类。由于一些领域和类中存在噪声, 遵循 Gu 等人^[66]的工作, 我们选择了 4 个领域和 126 个类。选择的 4 个领域是 Clipart(C), Painting(P), Real(R)和 Sketch(S)。4 个领域的图像数量分别为 18703、31502、70358 和 24582。仿造 Gu 等人^[66]在部分领域自适应工作中的

实验协议，我们按字母顺序选择前 40 类作为目标域的已知类，41-126 类作为未知类。

4.4.2 实验设置

对比算法：遵循以往工作中采用的实验协议，我们将所提出的 JAUPL 算法与开集领域自适应算法进行了比较。这些方法包括 OSBP^[10]、STA^[24]、UAN^[20]、MTS^[79]、UADAL^[90]、Anna^[92]。由于我们和一些工作^{[90][92]}使用了相同的实验设置，因此我们引用了这些对比算法在原始论文中报告的结果。由于部分对比算法在一些数据集上没有报告结果，因此，我们使用作者提供的官方代码运行实验，实验参数遵循论文描述，并报告它们在该数据集上的最佳结果。

实验指标：我们通过以下实验指标来对上述开集领域自适应任务进行性能对比，这些指标在以往工作^{[10][92][93]}中是通用的。具体来说，我们以 OS * 和 UNK 的调和平均值 HOS 作为对比指标，其中 OS * 是模型在已知类上的类平均精度，而 UNK 是模型在未知类的精度。

实验细节：我们在 Pytorch 上进行了实验，使用 ResNet50 模型^[35]作为特征提取器 h ，并在 h 后添加一个特征分类器 g 。其中，特征分类器包含 softmax 层，其输出维度为源域数据集中的类别数量加一，其中最后一个输出维度对应未知类(也就是说, Office-home 为 26，前 25 维对应已知类，最后一维对应未知类, DomainNet 为 41，前 40 维对应已知类，最后一维对应未知类)。我们通过 mini-batch SGD 算法优化网络模型参数。我们遵循 Ganin 等人^[5]工作中的操作，使用公式 $\lambda_p = \frac{2}{1-e^{-10p}} - 1$ 渐进式地调整权衡参数 λ_1 和 λ_2 ，其中 p 表示训练进度，从 0 到 1 线性变化。此外，再次遵循 Ganin 等人^[30]工作中的操作，我们通过公式 $\eta_p = \frac{\eta_0}{(1+\alpha p)^\beta}$ 渐进式调整特征提取器 h 的学习率，其中 $\eta_0=0.001, \alpha=10, \beta=0.75$ ，其中 p 也是训练进度。由于 h 是从 ImageNet 预训练好的， g 是从头开始训练的，所以我们将 g 的学习率设置为 h 的 10 倍。

4.4.3 实验结果

我们在表 4-1 和表 4-2 中分别报告了 Office-Home 和 DomainNet 上的分类结果。为了便于比较，在表中的 HOS 指标一列中，我们将最好的结果以**粗体**突出显示，并给次好的结果加上下划线。

表 4-1 开集领域自适应任务在 Office-Home (25→65 类) 上的 OS*, UNK, HOS (%)

A→C				A→P			A→R		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	50.20	61.10	55.10	71.80	59.80	65.20	79.30	67.50	72.90
STA	46.00	72.30	55.80	68.00	48.40	54.00	78.60	60.40	68.30
UAN	62.40	0.00	0.00	81.10	0.00	0.00	88.20	0.00	0.00
MTS	58.62	62.10	60.31	63.99	57.39	60.51	67.39	53.09	59.40
UADAL	54.90	74.70	63.20	69.10	72.50	70.80	81.30	73.70	77.40
Anna	61.40	78.70	69.00	68.30	79.90	73.70	74.10	79.70	<u>76.80</u>
JAUPL	61.17	70.26	<u>65.40</u>	78.59	65.64	<u>71.53</u>	84.82	68.19	75.60
C→A				C→P			C→R		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	59.40	70.30	64.30	67.00	62.70	64.70	72.00	69.20	70.60
STA	51.40	65.00	57.40	61.80	59.10	60.40	67.00	66.70	66.80
UAN	70.50	0.00	0.00	74.00	0.00	0.00	80.60	0.10	0.20
MTS	56.51	53.29	54.85	50.95	57.74	54.13	74.85	50.63	60.40
UADAL	53.50	80.50	64.20	62.10	78.80	<u>69.50</u>	69.10	78.30	<u>73.40</u>
Anna	58.00	73.10	<u>64.70</u>	64.20	73.60	68.60	66.90	80.20	73.00
JAUPL	60.37	71.45	65.44	69.90	74.11	71.95	77.20	72.62	74.84
P→A				P→C			P→R		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	59.10	68.10	63.20	44.50	66.30	53.20	76.20	71.70	73.90
STA	54.20	72.40	61.90	44.20	67.10	53.20	76.20	64.30	69.50
UAN	73.70	0.00	0.00	59.10	0.00	0.00	84.00	0.10	0.20
MTS	80.31	48.67	60.61	82.22	53.42	64.76	57.05	61.40	59.15
UADAL	50.50	83.70	63.00	43.40	81.50	56.60	71.60	83.10	<u>76.90</u>
Anna	63.00	70.30	66.50	54.60	74.80	63.10	74.30	78.90	76.60
JAUPL	56.55	78.33	<u>65.68</u>	46.10	75.43	<u>57.22</u>	77.76	80.09	78.91
R→A				R→C			R→P		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	66.10	67.30	66.70	48.00	63.00	54.50	76.30	68.60	72.30
STA	67.50	66.70	67.10	49.90	61.10	54.50	77.10	55.40	64.50
UAN	77.50	0.10	0.20	66.20	0.00	0.00	85.00	0.10	0.10
MTS	72.88	52.51	61.04	81.80	43.25	56.58	70.89	51.95	59.96
UADAL	66.70	78.60	72.10	51.10	74.50	60.60	77.40	76.20	<u>76.80</u>
Anna	66.10	77.30	<u>71.30</u>	59.70	73.10	65.70	76.40	81.00	78.70
JAUPL	68.11	68.31	68.21	57.17	70.71	<u>63.22</u>	81.84	69.10	74.94
Avg									
Method	OS*	UNK	HOS						
OSBP	64.16	66.30	64.72						
STA	61.83	63.24	61.12						
UAN	75.19	0.03	0.06						
MTS	68.12	53.79	59.31						
UADAL	62.56	78.01	68.71						
Anna	65.58	76.72	70.64						
JAUPL	68.30	72.02	<u>69.41</u>						

表 4-2 开集领域自适应任务在 DomainNet (40→126 类) 上的 OS*, UNK, HOS (%)

C→P				C→R			C→S		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	51.78	55.71	53.68	67.52	59.44	63.23	43.25	58.25	49.64
STA	53.56	46.24	49.63	40.46	52.25	45.61	46.86	72.91	57.05
UAN	58.29	10.64	17.99	67.62	10.85	18.70	56.55	10.75	18.06
MTS	40.19	62.90	49.05	43.81	63.49	51.84	47.74	62.13	53.99
UADAL	60.20	65.90	<u>62.90</u>	72.10	75.30	<u>73.70</u>	62.10	62.90	<u>62.50</u>
Anna	49.45	69.19	57.68	60.79	71.79	65.83	52.36	65.53	58.21
JAUPL	62.63	71.09	66.59	75.28	74.13	74.70	58.61	71.20	64.30
P→C				P→R			P→S		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	52.06	63.09	57.05	74.21	60.67	66.76	46.48	61.63	53.00
STA	42.29	50.22	45.91	32.24	55.64	40.82	41.18	41.34	41.26
UAN	53.69	7.51	13.17	64.80	8.76	15.44	58.83	6.44	11.61
MTS	47.26	56.62	51.52	47.38	58.44	52.33	58.86	56.68	57.75
UADAL	60.70	71.40	<u>65.60</u>	77.40	73.80	75.60	53.60	61.50	<u>57.30</u>
Anna	56.47	69.30	62.23	69.73	68.46	69.09	51.53	64.03	57.11
JAUPL	65.54	73.40	69.25	78.17	69.05	<u>73.33</u>	56.41	80.61	66.37
R→C				R→P			R→S		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	53.93	70.56	61.13	57.53	66.42	61.65	40.97	69.64	51.59
STA	58.71	34.56	43.51	39.28	61.97	48.08	40.88	44.60	42.66
UAN	44.45	25.50	32.41	44.42	24.91	31.92	44.48	22.34	29.74
MTS	67.02	45.86	54.46	47.98	60.30	53.44	45.63	64.31	53.38
UADAL	52.40	40.10	45.40	31.10	91.30	46.30	45.60	41.70	43.60
Anna	65.68	74.22	<u>69.69</u>	62.19	65.67	<u>63.88</u>	51.77	64.86	57.58
JAUPL	66.34	74.68	70.26	60.15	74.53	66.57	42.54	66.36	51.84
S→C				S→P			S→R		
Method	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	51.88	53.75	52.80	46.60	57.31	51.41	65.22	57.18	60.94
STA	49.68	70.36	58.24	50.45	51.38	50.91	37.53	75.91	50.23
UAN	44.59	19.67	27.29	44.18	28.55	34.69	44.45	25.50	32.41
MTS	53.59	70.81	61.01	57.59	53.23	55.32	60.70	50.50	55.13
UADAL	63.10	72.20	<u>67.30</u>	59.00	62.10	<u>60.50</u>	66.70	76.50	71.30
Anna	58.09	71.03	63.91	53.10	63.26	57.74	63.80	65.68	64.72
JAUPL	67.43	67.63	67.53	61.86	73.76	67.29	71.26	68.66	<u>69.93</u>
Avg									
Method	OS*	UNK	HOS						
OSBP	54.29	61.14	56.91						
STA	44.43	54.78	47.83						
UAN	52.20	16.78	23.62						
MTS	51.48	58.77	54.10						
UADAL	58.67	66.23	61.00						
Anna	57.91	67.75	<u>62.31</u>						
JAUPL	63.85	72.09	67.33						

4.4.4 统计检验

我们使用表 4-1 和表 4-2 的结果进行了 Wilcoxon 符号秩检验^[76]，以验证我们的 JAUPL 方法是否显著优于其他比较方法。该检验使用统计量 T 来比较两种方法在 N 个任务上的性能。具体来说，我们使用 HOS 作为方法的性能度量，将 JAUPL 方法作为对照方法，与其他方法进行了比较。我们将结果报告在表 4-3 中。从表 4-3 中可以看出，在 6 对检验中，有 5 对检验的 T 值都低于 81。其中 81 是在置信水平 $\alpha=0.05$ 和任务数 $N=24$ 时，Wilcoxon 符号秩检验的临界值。根据文献^[76]，这表明 JAUPL 方法在统计意义上优于 STA、OSBP、UAN、MTS 和 UADAL 方法，并且与当前最优方法 Anna 取得了相当的性能。

表 4-3 对比方法与 JAUPL 的 Wilcoxon 符号秩检验的统计量 T 值。

Strategy	OSBP	STA	UAN	MTS	UADAL	Anna
T value	0	0	0	8	57	92

4.5 本章总结

与第三章的部分领域自适应问题不同，本章研究的是目标域标签空间包含源域标签空间的开集领域自适应问题。在这里，我们根据开集领域自适应的目标推导出了一个目标域泛化误差上界，并利用误差上界来学习目标模型。我们根据误差上界提出了联合对齐与未知类概率损失（Joint Alignment with Unknown Probability Loss, JAUPL）算法。在算法中，我们通过最小化卡方散度的经验估计对齐了源域与目标域已知类的联合分布，此过程中涉及到一个带有解析解的凸问题的求解。我们通过实验证明了所提出方法的有效性。

第五章 总结与展望

5.1 工作总结

本文以联合分布对齐为主要思想，深入研究在领域自适应的约束条件被放宽，所做假设不被满足的情况下的广义领域自适应问题，其中，本文主要研究的广义领域自适应问题包括多源领域自适应、部分领域自适应和开集领域自适应。

在第二章中，我们研究多源领域自适应问题。该问题放松了对源域数据集数量的约束，困难在于每个源域联合分布与目标域联合分布都是不同的。为了解决多源领域自适应问题，我们提出了一种名为互信息估计和最小化的算法，通过在网络特征空间中最小化互信息的估计值，实现了所有源域联合分布和目标域联合分布的对齐。特别地，我们将互信息估计问题转化为凸优化问题，这使得我们可以很容易得到全局最优解。实验证明所提算法在统计学意义上优于其他方法。

在第三章中，我们研究部分领域自适应问题。该问题放松了对标签空间的约束，并假设源域标签空间包含目标域标签空间，困难在于除了域间联合分布的不同降低了模型在目标域上的性能外，源域离群数据也降低了模型在目标域上的性能。为了解决这些问题，我们提出了一种名为极大似然权重估计的算法，通过最大化似然函数来估计一个权重函数，将源域联合分布的相关部分对齐到目标域联合分布，同时降低源域离群数据的负面影响。在算法中，我们通过设计指数模型和最大化似然函数来估计权重函数，从而得到一个具有全局最优解的凸优化问题。实验证明所提算法在多个基准数据集上取得了优越的性能。

在第四章中，我们研究开集领域自适应问题，与第三章部分领域自适应相似，该问题同样放松了对标签空间的约束，但不同之处在于开集领域自适应假设源域标签空间是目标域标签空间的子集。为了解决开集领域自适应问题，我们提出了一种名为联合对齐与未知类概率损失的算法。在算法中，我们推导出模型在目标域上的泛化误差上界，并通过最小化误差上界的经验估计来优化模型。特别地，误差上界包含源域联合分布与目标域已知类部分联合分布的卡方

散度，在这里，我们将卡方散度的估计问题转化为一个带有解析解的凸问题。实验证明所提算法取得了优越的性能。

综上所述，本文主要研究多源领域自适应、部分领域自适应、以及开集领域自适应这三种广义领域自适应问题，为具体的问题提出对应的算法，并通过实验证明了所提算法的有效性，为解决实际应用中更加复杂和多样化的领域自适应问题提供了有效的解决方案。

5.2 未来展望

虽然本文在广义领域自适应问题中取得了一定成果，但随着现实应用场景的不断演变和复杂化，仍然存在许多需要深入研究的问题。未来，我们的研究工作将集中在以下两个方面：

1) 研究其他场景下的广义领域自适应问题：我们将探索在不同的实际应用场景中出现的广义领域自适应问题，并针对这些广义领域自适应问题提出适用的算法。这些广义领域自适应问题可能涉及到不同类型的数据（如文本、图像、视频等）以及特定领域的需求（如数据渐进式提供，无目标域数据等）。例如，源域和目标域具有不同特征空间的异构领域自适应问题、无目标域数据的领域泛化问题等。此外，本文研究的三个问题可能相互交叉，产生多源开集领域自适应问题、通用领域自适应问题等。通过深入研究和算法创新，我们可以为这些广义领域自适应问题提供更有效的解决方案，推动领域自适应技术在实际中的应用。

2) 将所提算法落地在实际场景中：我们将致力于将已提出的算法应用到实际场景中，例如物体检测、医疗图像处理等领域。通过与相关领域的专业人士合作，我们可以更好地理解实际需求，并根据实际情况对算法进行调整和优化，以确保其在真实环境中的有效性和可靠性。这将促进领域自适应技术在实际应用中的落地，并为解决现实生活中的实际问题提供有力支持。

附录

附录 1 定理 4.1 的证明

定理 4.1: 假设存在 $M > 0$, 使得 $\mathcal{L}(\cdot, \cdot) \leq M$, 存在 $\epsilon > 0$, 使得卡方散度

$\chi_{X \times Y^S}^2(P^S(h(x), y), P^t(h(x), y)) \leq \epsilon$, 那么, 对于任意 h , 都有:

$$\begin{aligned} & \int_{X \times Y^S} P^t(x, y) \mathcal{L}(f(x), y) dx dy \\ & \leq \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy + \frac{\sqrt{2}M}{\sqrt{\epsilon}} \chi_{X \times Y^S}^2(P^S(h(x), y), P^t(h(x), y)). \end{aligned} \quad (5-1)$$

证明:

$$\begin{aligned} & \int_{X \times Y^S} P^t(x, y) \mathcal{L}(f(x), y) dx dy \\ & = \int_{X \times Y^S} P^t(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \end{aligned} \quad (5-2)$$

$$\begin{aligned} & = \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\ & \quad + \int_{X \times Y^S} P^t(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\ & \quad - \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \end{aligned} \quad (5-3)$$

$$\begin{aligned} & \leq \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\ & \quad + \left| \int_{X \times Y^S} P^t(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \right. \\ & \quad \left. - \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \right| \end{aligned} \quad (5-4)$$

$$\begin{aligned} & = \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\ & \quad + \int_{X \times Y^S} \mathcal{L}(g(h(x)), y) |P^t(h(x), y) - P^S(h(x), y)| dh(x) dy \end{aligned} \quad (5-5)$$

$$\begin{aligned} & \leq \int_{X \times Y^S} P^S(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\ & \quad + M \int_{X \times Y^S} |P^S(h(x), y) - P^t(h(x), y)| dh(x) dy \end{aligned} \quad (5-6)$$

$$\begin{aligned}
 &\leq \int_{X \times Y^s} P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\
 &\quad + \sqrt{2}M \sqrt{\int_{X \times Y^s} P^s(h(x), y) \log \frac{P^s(h(x), y)}{P^t(h(x), y)} dh(x) dy} \quad (5-7)
 \end{aligned}$$

$$\begin{aligned}
 &\leq \int_{X \times Y^s} P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\
 &\quad + \sqrt{2}M \sqrt{\int_{X \times Y^s} P^s(h(x), y) \frac{P^s(h(x), y)}{P^t(h(x), y)} dh(x) dy} - 1 \quad (5-8)
 \end{aligned}$$

$$\begin{aligned}
 &= \int_{X \times Y^s} P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy + \sqrt{2}M \sqrt{\chi_{X \times Y^s}^2(P^s(h(x), y), P^t(h(x), y))} \\
 &\quad (5-9)
 \end{aligned}$$

$$\begin{aligned}
 &\leq \int_{X \times Y^s} P^s(h(x), y) \mathcal{L}(g(h(x)), y) dh(x) dy \\
 &\quad + \frac{\sqrt{2}M}{\sqrt{\epsilon}} \chi_{X \times Y^s}^2(P^s(h(x), y), P^t(h(x), y)). \quad (5-10)
 \end{aligned}$$

等式(5-6)是基于积分的性质和 M 为损失函数上界的假设。不等式(5-7)由 Pinsker 不等式推导可得。当 $x > 0$ 时，基于不等式 $\log x \leq x - 1$ ，可推导得出等式(5-8)。等式(5-10)基于 ϵ 为卡方散度上界的假设。

参考文献

- [1] He, K., Zhang, X., Ren, S., Sun, J. Deep residual learning for image recognition[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2016. pp. 770 – 778.
- [2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. Attention is all you need[A]. In: Advances in Neural Information Processing Systems 30[C], 2017.
- [3] Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N.D. Dataset shift in machine learning[M]. MIT Press, 2008.
- [4] Sugiyama, M., Kawanabe, M. Machine Learning in Nonstationary Environments: Introduction to Covariate Shift Adaptation[M]. MIT Press, 2012.
- [5] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., Lempitsky, V. Domain-adversarial training of neural networks[J]. Journal of Machine Learning Research, 2016, 17: 1 – 35.
- [6] Chen, S., Harandi, M., Jin, X., Yang, X. Domain adaptation by joint distribution invariant projections[J]. IEEE Transactions on Image Processing, 2020, 29: 8264 – 8277.
- [7] Chen, S., Harandi, M., Jin, X., Yang, X. Semi-supervised domain adaptation via asymmetric joint distribution matching[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32(12): 5708 – 5722.
- [8] Cao, Z., Long, M., Wang, J., Jordan, M.I. Partial transfer learning with selective adversarial networks[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2018. pp. 2724 – 2732.
- [9] Chen, S. Decomposed adversarial domain generalization[J]. Knowledge-Based Systems, 2023, 263: 110300.
- [10] Saito, K., Yamamoto, S., Ushiku, Y., Harada, T. Open set domain adaptation by backpropagation[A]. In: Proceedings of the European Conference on Computer Vision[C], 2018. pp. 153-168.
- [11] Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B. Moment matching for multi-source domain adaptation[A]. In: IEEE International Conference on Computer Vision[C], 2019. pp. 1406-1415.

- [12] Wasserman, L. All of statistics: a concise course in statistical inference[M]. Springer, 2004.
- [13] Long, M., Zhu, H., Wang, J., Jordan, M. Deep transfer learning with joint adaptation networks[A]. In: International Conference on Machine Learning[C], 2017. pp. 2208-2217.
- [14] Nguyen, A.T., Tran, T., Gal, Y., Torr, P.H., Baydin, A.G. KL guided domain adaptation[A]. In: International Conference on Learning Representations[C], 2022. pp. 1-12.
- [15] Zhu, Y., Zhuang, F., Wang, J., Ke, G., Chen, J., Bian, J., et al. Deep subdomain adaptation network for image classification[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32(4), 1713 – 1722.
- [16] Chen, S., Han, L., Liu, X., He, Z., Yang, X. Subspace distribution adaptation frameworks for domain adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(12): 5204 – 5218.
- [17] Liu, F., Zhang, G., Lu, J. Heterogeneous domain adaptation: An unsupervised approach[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(12): 5588-5602.
- [18] Yao, Y., Li, X., Zhang, Y., Ye, Y. Multi-source heterogeneous domain adaptation with conditional weighting adversarial network[J]. IEEE Transactions on Neural Networks and Learning Systems, 2023, 34(4): 2079-2092.
- [19] Chen, S., Chen, L. Joint-product representation learning for domain generalization in classification and regression[J]. Neural Computing and Applications, 2023, 35: 16509 – 16526.
- [20] You, K., Long, M., Cao, Z., Wang, J., Jordan, M. I. Universal domain adaptation[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2019. pp. 2715-2724.
- [21] Ren, C.-X., Liu, Y.-H., Zhang, X.-W., Huang, K.-K. Multi-source unsupervised domain adaptation via pseudo target domain[J]. IEEE Transactions on Image Processing, 2022, 31(5): 2122-2135.
- [22] Wang, H., Xu, M., Ni, B., Zhang, W. Learning to combine: Knowledge aggregation for multi-source domain adaptation[A]. In: European Conference on Computer Vision[C], 2020. pp. 727-744.
- [23] Chen, Z., Chen, C., Cheng, Z., Jiang, B., Fang, K., Jin, X. Selective transfer with reinforced transfer network for partial domain adaptation[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2020. pp. 12706-12714.

- [24] Liu, H., Cao, Z., Long, M., Wang, J., Yang, Q. Separate to adapt: Open set domain adaptation via progressive separation[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2019. pp. 2927-2936.
- [25] Zhuang, F., Zhu, H., Li, Z., Liu, Y., Wen, Y., Xu, Z., & Han, J. Explicit self-ensemble for partial domain adaptation[J]. IEEE Transactions on Image Processing, 2022, 31, 601 – 613.
- [26] Chen, S., Hong, Z. Domain generalization by distribution estimation[J]. International Journal of Machine Learning and Cybernetics, 2023, 14: 3457 – 3470.
- [27] Zhou, F., Chen, Y., Yang, S., Wang, B., & Chaib-Draa, B. On the value of label and semantic information in domain generalization[J]. Neural Networks, 2023, 163, 244 – 255.
- [28] Li, Y., MURIAS, M., DAWSON, G., CARLSON, D.E. Extracting relationships by multi-domain matching[A]. In: Advances in Neural Information Processing Systems[C], Vol. 31, 2018.
- [29] Park, G.Y., Lee, S.W. Information-theoretic regularization for multi-source domain adaptation[A]. In: IEEE International Conference on Computer Vision[C], 2021. pp. 9214-9223.
- [30] Wen, J., Greiner, R., Schuurmans, D. Domain aggregation networks for multi-source domain adaptation[A]. In: International Conference on Machine Learning[C], 2020. pp. 10214-10224.
- [31] Zhao, H., Zhang, S., Wu, G., Moura, J.M.F., Costeira, J.P., Gordon, G.J. Adversarial multiple source domain adaptation[A], in: Advances in Neural Information Processing Systems[C], 2018, Vol. 31.
- [32] Ge, P., Ren, C.-X., Xu, X.-L., Yan, H. Unsupervised domain adaptation via deep conditional adaptation network[J]. Pattern Recognition, 2023, 134: 109088.
- [33] Calandriello, D., Niu, G., Sugiyama, M. Semi-supervised information-maximization clustering[J]. Neural Networks, 2014, 57: 103 – 111.
- [34] Frénay, B., Doquire, G., Verleysen, M. Is mutual information adequate for feature selection in regression?[J]. Neural Networks, 2013, 48: 1 – 7.
- [35] Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y. Learning deep representations by mutual information estimation and maximization[A]. In: International Conference on Learning Representations[C], 2019. pp. 1 – 14.

- [36] Shermin, T., Lu, G., Teng, S.W., Murshed, M., Sohel, F. Adversarial network with multiple classifiers for open set domain adaptation[J]. IEEE Transactions on Multimedia, 2020, 23: 2732-2744.
- [37] Jin, X., Yang, X., Fu, B., Chen, S. Joint distribution matching embedding for unsupervised domain adaptation[J]. Neurocomputing, 2020, 412: 115 – 128.
- [38] Long, M., Cao, Y., Cao, Z., Wang, J., Jordan, M.I. Transferable representation learning with deep adaptation networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(9): 3071-3085.
- [39] Ren, C.-X., Luo, Y.-W., Dai, D.-Q. BuresNet: Conditional Bures metric for transferable representation learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(4): 4198-4213.
- [40] Acuna, D., Zhang, G., Law, M.T., Fidler, S. f-domain adversarial learning: Theory and algorithms[A]. In: International Conference on Machine Learning[C], 2021. pp. 66 – 75.
- [41] Rakotomamonjy, A., Flamary, R., Gasso, G., Alaya, M.E., Berar, M., Courty, N. Optimal transport for conditional domain matching and label shift[J]. Machine Learning, 2022, 1-20.
- [42] Jing, M., Li, J., Lu, K., Zhu, L., Yang, Y. Learning explicitly transferable representations for domain adaptation[J]. Neural Networks, 2020, 130: 39-48.
- [43] Xu, X.-L., Xu, G.-X., Ren, C.-X., Dai, D.-Q., Yan, H. Conditional independence induced unsupervised domain adaptation[J]. Pattern Recognition, 2023, 143: 109787.
- [44] Chen, S., Wang, L., Hong, Z., Yang, X. Domain generalization by joint-product distribution alignment[J]. Pattern Recognition, 2023, 134: 109086.
- [45] Redko, I., Courty, N., Flamary, R., Tuia, D. Optimal transport for multi-source domain adaptation under target shift[A]. In: International Conference on Artificial Intelligence and Statistics[C], 2019. pp. 849-858.
- [46] Liu, Y.-H., Ren, C.-X. A two-way alignment approach for unsupervised multi-source domain adaptation[J]. Pattern Recognition, 2022, 124: 108430.
- [47] Zhao, S., Wang, G., Zhang, S., Gu, Y., Li, Y., Song, Z., Xu, P., Hu, R., Chai, H., Keutzer, K. Multi-source distilling domain adaptation[A], in: AAAI Conference on Artificial Intelligence[C], 2020, pp. 12975 – 12983.

- [48] Zhu, Y., Zhuang, F., & Wang, D. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources[A]. In AAAI conference on artificial intelligence[C]. 2019, Vol. 33. No. 01 pp. 5989 – 5996.
- [49] Chen, S., Hong, Z., Harandi, M., Yang, X. Domain neural adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022, 1 – 12.
- [50] Nguyen, X., Wainwright, M.J., Jordan, M.I. Estimating divergence functionals and the likelihood ratio by convex risk minimization[J]. IEEE Transactions on Information Theory, 2010, 56(11): 5847-5861.
- [51] Boyd, S., Vandenberghe, L. Convex optimization[M]. Cambridge: Cambridge University Press, 2004.
- [52] Nocedal, J., Wright, S.J. Numerical Optimization[M]. Springer, 1999.
- [53] Li, D., Yang, Y., Song, Y.Z., Hospedales, T.M. Deeper, broader and artier domain generalization[A]. In: IEEE International Conference on Computer Vision[C], 2017. pp. 5542-5550.
- [54] Vapnik, V. N. Statistical learning theory[M]. Wiley-Interscience, 1998.
- [55] Bhushan Damodaran, B., Kellenberger, B., Flamary, R., Tuia, D., Courty, N. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation[A]. In: European Conference on Computer Vision[C], 2018. 447 – 463.
- [56] Saenko, K., Kulis, B., Fritz, M., Darrell, T. Adapting visual category models to new domains[A]. In: European Conference on Computer Vision[C], 2010. pp. 213-226.
- [57] Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S. Deep hashing network for unsupervised domain adaptation[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2017. pp. 5018-5027.
- [58] Demšar, J. Statistical comparisons of classifiers over multiple data sets[J]. Journal of Machine Learning Research, 2006, 7: 1 – 30.
- [59] Maaten, L.v.d., Hinton, G. Visualizing data using t-SNE[J]. Journal of Machine Learning Research, 2008, 9(2579-2605): 85.
- [60] Jiang, J. A literature survey on domain adaptation of statistical classifiers[R]. 2008, 3: 1-12.
- [61] Cao, Z., Ma, L., Long, M., Wang, J. Partial adversarial domain adaptation[A]. In: European Conference on Computer Vision[C], 2018. pp. 135-150.

- [62] Cao, Z., You, K., Long, M., Wang, J., Yang, Q. Learning to transfer examples for partial domain adaptation[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2019. pp. 2985-2994.
- [63] Cao, Z., You, K., Zhang, Z., Wang, J., Long, M. From big to small: Adaptive learning to partial-set domains[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(5): 1766-1780.
- [64] Liang, J., Wang, Y., Hu, D., He, R., Feng, J. A balanced and uncertainty-aware approach for partial domain adaptation[A]. In: European Conference on Computer Vision[C], 2020. pp. 123-140.
- [65] Hu, J., Tuo, H., Wang, C., Qiao, L., Zhong, H., Yan, J., Jing, Z., Leung, H. Discriminative partial domain adversarial network[A]. In: European Conference on Computer Vision[C], 2020. pp. 632-648.
- [66] Gu, X., Yu, X., Yang, Y., Sun, J., Xu, Z. Adversarial reweighting for partial domain adaptation[A]. In: Advances in Neural Information Processing Systems[C], 2021. pp. 14860-14872.
- [67] Zhang, J., Ding, Z., Li, W., Ogunbona, P. Importance weighted adversarial nets for partial domain adaptation[A], in: IEEE Conference on Computer Vision and Pattern Recognition[C], 2018, pp. 8156-8164.
- [68] Li, L., Wan, Z., He, H. Dual alignment for partial domain adaptation[J]. IEEE Transactions on Cybernetics, 2021, 51(6): 3404-3416.
- [69] Yang, C., Cheung, Y. M., Ding, J., Tan, K. C., Xue, B., Zhang, M. Contrastive learning assisted-alignment for partial domain adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2023, 34: 7621-7634.
- [70] Chen, S., Wu, H., Liu, C. Domain invariant and agnostic adaptation[J]. Knowledge-Based Systems, 2021, 227: 107192.
- [71] Zhang, C., Zhang, J. Transferable regularization and normalization: Towards transferable feature learning for unsupervised domain adaptation[J]. Information Sciences, 2022, 609: 595-604.
- [72] He, C., Zheng, L., Tan, T., Fan, X., Ye, Z. Manifold discrimination partial adversarial domain adaptation[J]. Knowledge-Based Systems, 2022, 252: 109320.

- [73] Li, S., Liu, C.H., Lin, Q., Wen, Q., Su, L., Huang, G., Ding, Z. Deep residual correction network for partial domain adaptation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 43(8): 2329-2344.
- [74] Zhang, C., Zhao, Q. Attention guided for partial domain adaptation[J]. Information Sciences, 2021, 547, 860-869.
- [75] Li, S., Gong, K., Xie, B., Liu, C.H., Cao, W., Tian, S. Critical classes and samples discovering for partial domain adaptation[J]. IEEE Transactions on Cybernetics, 2023, 53(6): 5641-5654.
- [76] Chen, S., Zheng, L., Wu, H. Riemannian representation learning for multi-source domain adaptation[J]. Pattern Recognition, 2023, 137: 109271.
- [77] Wen, L., Chen, S., Xie, M., Liu, C., Zheng, L. Training multi-source domain adaptation network by mutual information estimation and minimization[J]. Neural Networks, 2024, 171: 353-361.
- [78] Chen, S. Multi-source domain adaptation with mixture of joint distributions[J]. Pattern Recognition, 2024, 149: 110295.
- [79] Chang, D., Sain, A., Ma, Z., Song, Y.Z., Wang, R., Guo, J. Mind the gap: open set domain adaptation via mutual-to-separate framework[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2023, pp. 1 - 1.
- [80] Luo, Y., Ren, C., Dai, D., Yan, H. Unsupervised domain adaptation via discriminative manifold propagation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(4): 1653-1669.
- [81] Ren, C., Ge, P., Yang, P., Yan, S. Learning target-domain-specific classifier for partial domain adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 32(6): 1989-2001.
- [82] Long, M., Cao, Z., Wang, J., Jordan, M. Conditional adversarial domain adaptation[A]. In: Advances in Neural Information Processing Systems[C], 2018. pp. 1640-1650.
- [83] Müller, R., Kornblith, S., Hinton, G.E. When does label smoothing help?[A]. In: Advances in Neural Information Processing Systems[C], 2019. pp. 4694-4703.

- [84] Mishra, S., Sanodiya, R.K. A novel angular based unsupervised domain adaptation framework for image classification[J]. IEEE Transactions on Artificial Intelligence, 2023, 1(1): 1-13.
- [85] Li, J., Lu, S., Li, Z. Unsupervised domain adaptation via softmax-based prototype construction and adaptation[J]. Information Sciences, 2022, 609: 257-275.
- [86] Zhong, L., Fang, Z., Liu, F., Yuan, B., Zhang, G., Lu, J. Bridging the theoretical bound and deep algorithms for open set domain adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 34(8), 3859 – 3873.
- [87] Chen, J., Wu, X., Duan, L., Gao, S. Domain adversarial reinforcement learning for partial domain adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022, 539-553.
- [88] Liu, J., He, H., Liu, M., Li, J., Lu, K. Manifold regularized joint transfer for open set domain adaptation[J]. IEEE Transactions on Multimedia, 2023, 25: 9356-9369.
- [89] Fang, Z., Lu, J., Liu, F., Xuan, J., Zhang, G. Open set domain adaptation: Theoretical bound and algorithm[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 32(10): 4309 – 4322.
- [90] Jang, J., Na, B., Shin, D. H., Ji, M., Song, K., Moon, I. C. Unknown-aware domain adversarial learning for open-set domain adaptation[A]. In: Advances in Neural Information Processing Systems[C], 2022, 35: 16755 – 16767.
- [91] Liu, J., Guo, X., Yuan, Y. Unknown-oriented learning for open set domain adaptation[A]. In: European Conference on Computer Vision[C], 2022. pp. 334-350.
- [92] Li, W., Liu, J., Han, B., Yuan, Y. Adjustment and alignment for unbiased open set domain adaptation[A]. In: IEEE Conference on Computer Vision and Pattern Recognition[C], 2023. pp. 24110-24119.;
- [93] Bucci, S., Loghmani, M.R., Tommasi, T. On the effectiveness of image rotation for open set domain adaptation[A]. In: European Conference on Computer Vision[C], 2020. 422 – 438.

攻读学位期间主要研究成果

一、学术论文

- [1] **Wen, L.**, Chen, S., Xie, M., Liu, C., Zheng, L. Training multi-source domain adaptation network by mutual information estimation and minimization[J]. Neural Networks, 2024,171, 353-361. (中科院一区 Top, CCF-B, IF:7.8)
- [2] **Wen, L.**, Chen, S., Hong, Z., Zheng, L. Maximum likelihood weight estimation for partial domain adaptation[J], Information Sciences. (Minor Revision, 中科院一区 Top, CCF-B, IF:8.1)
- [3] **Wen, L.**, Chen, S., Open-set domain adaptation by joint alignment with unknown probability loss. (In Preparation)

致谢

在汕头大学度过了三年的研究生生活，不知不觉间，我从一个对科研一无所知的学生，成长为了能够做出一些科研成果的研究生。在这段宝贵的时光里，我要特别感谢我的指导老师陈森涛老师。老师不厌其烦地给予我指导，在自己科研繁忙期间还特意腾出时间与我见面讨论交流。在研一期间，老师耐心地教我学习数学、机器学习和领域自适应的基础知识，给了我一整年的时间来打好基础。在研二期间，基于研一的积累，我开始有了一些自己的想法，老师站在我的角度，为我解惑，帮助我理清思路。在我科研过程中遇到困难时，他及时提供帮助，纠正我的方向，指导我前进。在研三期间，我也开始取得了一些成果。回首硕士三年，我经历了从对科研的畏惧，迷茫到笃定，自信的过程，而这不易的旅途，让我收获到前所未有的喜悦。

我要感谢在这三年里一直陪伴在我身边的家人，感谢爸妈对我的支持，总是不假思索地站在我身边给予我理解和鼓励，陪伴着我给予我关怀和支持。他们的支持是我坚强前行的力量源泉。

感谢我的挚友——火火，感谢你从考研至读研期间的支持和帮助，是你陪伴我走过痛苦的考研旅途，并在我科研陷入低谷时给予我鼓励与陪伴。你的陪伴是我人生道路上最珍贵的财富。

最后，我感谢各位评审老师，在百忙之中抽出时间审阅和评议本论文，给予我宝贵的意见和建议。

温力胜