本科生毕业论文（设计）

题目：基于原型导向条件传输的无监督域适应算法

姓　　名　胡恩

学　　号　18343043

院　　系　数学学院

专　　业　统计学

指导教师　任传贤 (副教授)

2022 年 7 月 12 日

基于原型导向条件传输的

无监督域适应算法

Unsupervised Domain Adaptation based on

Prototype-Oriented Conditional Transport

姓　　名	胡 恩
学　　号	18343043
院　　系	数学学院
专　　业	统计学
指导教师	任传贤 (副教授)

2022 年 7 月 12 日

表一　毕业论文（设计）开题报告

Form 1: Research Proposal of Graduation Thesis (Design)

论文 (设计) 题目: 基于原型导向条件传输的无监督域适应算法

Thesis (Design) Title: Unsupervised Domain Adaptation based on Prototype-Oriented Conditional Transport

(简述选题的目的、思路、方法、相关支持条件及进度安排等)

(Please briefly state the research objective, research methodology, research procedure and research schedule in this part.)

国内外关于本选题的研究现状、水平和发展趋势，选题研究的目的和意义:

无监督域适应 (unsupervised domain adaptation)，以下简称 UDA，是利用在 source 域（带标签的样本数据集）上训练完成的模型的信息，在相似的 target 域（不带标签的与 source 域数据集相似的样本数据集，如机打阿拉伯数字与手写阿拉伯数字）里通过无监督学习训练模型，以完成目标任务（如图像分类等）的算法。目前的研究大部分基于深度神经网络，并用于图像分类和图像分割等目标任务。研究者们相信，在经过特征编码器的提取后，相似域的特征分布具有一定的相似性，而这种相似性需要对特征经过一定的变换后对齐 (align) 两个域的特征分布才能显现。为此，研究者们各自提出了不同的对齐方法并实验得到了一些不错的成果，比如经典的通过分布间的 MMD(Maximum Mean Discrepancy) 距离来定义损失函数，从优化损失函数的思路确定 target 域的参数从而对齐 source 域和 target 域的特征分布，也有直接定义关于 domain-invariant 的损失函数，通过优化损失函数的方法直接确定 target 域的参数的方法。

EM 算法是统计学中经典的迭代性的参数估计方法，具有严格扎实的理论收敛定理证明。它通过假设隐藏变量存在的方法，迭代性地计算隐藏变量的期望并通过计算值估计参数，从而一步步逼近参数的真实值。在 UDA 的图像分类任务中，target 域的标签是天然的隐藏变量，通过设置 target 域的标签作为隐藏变量应用 EM 算法，能够估计 UDA 模型中的参数。事实上，在 UDA 领域的一些研究中，EM 算法的应用已经取得了一些成功。

本选题的意义在于将统计学经典的 EM 算法应用到崭新的 UDA 领域中，以提供一种新的 UDA 算法框架和思路，并检验其效果。

选题研究的可行性论述:

在 UDA 的图像分类任务中，target 域的标签是天然的隐藏变量，在一定的假设下，设置 target 域的标签作为隐藏变量应用 EM 算法，经过一些推导，初步确定了算法的理论可行性，另外，EM 算法的收敛性证明也是完善的。

毕业论文（设计）撰写提纲:

第一部分，Introduction，介绍 UDA 问题的背景，并简略提及过往经典的算法及现今 state-of-art 的算法的思路，其次介绍本研究算法的思路。

第二，Related Works，简要介绍并指引 UDA 的经典及流行方法

第三，算法名称，介绍算法的思路和推导过程，以及算法的执行步骤

第四，Experiments，介绍实验的设定，展示实验结果

第五，Conclusions，说明本研究的意义，并指出以后可能的改进方向

进度安排:

相关论文阅读及算法推导：12 月——2 月

实验算法与论文撰写：3 月

预留时间：4 月

Student Signature: Date:

指导教师意见:

Comments from Supervisor:

1. 同意开题 ( ) 2. 修改后开题 ( ) 3. 重新开题 ( )

1. Approved 2. Approved after Revision 3. Disapproved

Supervisor Signature: Date:

表二　毕业论文（设计）过程检查记录表

Form 2: Process Check-up Form

指导教师分阶段检查论文的进展情况（要求过程检查记录不少于 3 次）
The supervisor should check up the working process for the thesis（design）and fill up the following check-up log. At least three times of the check-up should be done and kept on the log.

第一次检查（First Check-up）：

学生总结
Student Self-Summary:

阅读指导教师提供的和自己查找的大量无监督域适应的相关文献，初步确定研究思路。

指导教师意见
Comments of Supervisor:

第二次检查（Second Check-up）：

学生总结
Student Self-Summary:

学习深度学习相关知识，学习深度学习的实现代码，尝试实现算法实验。

指导教师意见
Comments of Supervisor:

第三次检查（Third Check-up）：

学生总结

Student Self-Summary:

撰写论文初稿，实验提出的算法。

指导教师意见

Comments of Supervisor:

学生签名（Student Signature）：　　日期（Date）：

指导教师签名（Supervisor Signature）：　　日期（Date）：

总体完成情况 **(Overall Assessment)**	指导教师意见 Comments from Supervisor: 1、按计划完成，完成情况优（Excellent）：（　） 2、按计划完成，完成情况良（Good）：（　） 3、基本按计划完成，完成情况合格（Fair）：（　） 4、完成情况不合格（Poor）：（　） 指导教师签名（Supervisor Signature）： 日期（Date）：

表三　毕业论文（设计）答辩情况登记表

Form 3: Thesis Defense Performance Form

答辩人 **Student Name**	胡 恩 En Hu	**专业** Major	统计学 Statistics
论文 (设计) 题目 **Thesis（Design）Title**	基于原型导向条件传输的无监督域适应算法 Unsupervised Domain Adaptation based on Prototype-Oriented Conditional Transport		
答辩小组成员 **Committee Members**			

答辩记录

Records of Defense Performance:

记录人签名（Clerk Signature）：　　日期（Date）：

学术诚信声明

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文（设计）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本论文（设计）的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文（设计）的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日　　期：　　年　　月　　日

【摘　要】

在海量标注数据的支持下，深度卷积神经网络已经在各种计算机视觉任务上取得了巨大的成功，然而在实际应用中，未标注的数据更多。为了在一个特定的任务上利用这些在靶 (target) 域上的未标注的数据，一些无监督域适应方法，迁移在相同任务下相似的源 (source) 域数据上训练的模型。近期提出的 PCT(原型导向的条件传输) 通过双向传输原型和样本特征尝试实现这一目标。基于 PCT，我们通过完善理论改进了 PCT。具体地说，我们重新推导了从特征到原型的和从原型到特征的传输损失函数，讨论了传输代价函数的定义并使用 MLE 而非 EM 算法来估计靶域上的各类别比例。改进的方法 ours-cos 在数据集 Office31 上的大部分任务中取得了比 PCT 更好的实验结果。

关键词：　无监督域适应，原型，条件传输

[ABSTRACT]

Deep convolutional neural network has achieved a great success in various computer vision tasks with the support of massive labeled data, though there are more unlabeled data in practice. To make use of unlabeled data in target domain for a specific task, several unsupervised domain adaptation approaches, which transfer a model trained on a similar source domain for the same task to target domain, are proposed. A recently proposed method called PCT(Prototype-oriented Conditional Transport) tried to accomplish this mission by transporting prototypes, which is constructed by components of parameter of Softmax classifier, and features bi-directionally. Based on PCT, our work improves PCT by complementing the theory. Specifically, we re-derived the loss of transporting from features to prototypes, discuss the definition of transport cost function and use MLE instead of EM algorithm to estimate class proportions in target domain. Proposed method, ours-cos, outperforms PCT in most of tasks on dataset Office31.

Keywords: unsupervised domain adaptation, prototype, conditional transport

目录

1 绪论

1.1 问题背景

深度卷积神经网络 (deep Convolutional Neural Network) 已经在诸多计算机视觉任务，如图像分类、目标检测、语义分割，上取得了巨大的成就，这不但归功于模型和算法本身的优越性和机器计算能力的飞跃，海量带标签的图像数据也是必不可少的。然而，当应用到现实情况时，由于数据分布的偏移 (*shift*)[1]，良好的带标签的数据集 A 上训练的效果理想的模型，在另一处在不同环境，例如光照强度不同、图像角度不同、图像背景不同等的数据集 B 下完成同一任务时的误差则很大，且误差随着数据集 A 与 B 的数据分布的差距增大而增大[2] [3]。为了克服这一问题，域适应 (domain adaptation) 这一概念被提出[4]，通常，数据集 A 服从的分布被称为 source 域，B 服从的分布被称为 target 域，我们希望设计一种算法，在 source 域上训练的模型能够适应 target 域上的任务。更深入地，根据 target 域上数据是否带标签，域适应问题进一步被分为有监督域适应 (supervised domain adaptation)，半监督域适应 (semi-supervised domain adaptation) 和无监督域适应 (unsupervised domain adaptation) 问题，我们的模型和算法针对无监督域适应问题，即 target 域上数据为无标签数据时的域适应问题。

1.2 相关工作

在无监督域适应问题上，基于 MMD(Maximum Mean Discrepancy)[5] 及其近似估计[6]的算法和成果是丰富的[7] [8] [9] [10]。这些方法假设 target 域上的数据的特征 (features) 分布实际上是由 source 域上的特征分布经过一定变换得到的，因此，他们提出将变换后的 target 域上的特征分布和 source 域上的特征分布的 MMD 作为损失函数，通过学习该变换的参数来将 target 域上的特征分布与 source 域上的特征分布对齐 (align)。最近的工作[11]将图像的各类别比例纳入考虑，提出了加权 MMD(Weighted MMD)，实验效果超越了过去基于 MMD 的方法。更晚提出的与 MMD 无关的对抗式 (adversarial) 的域适应算法[12] [13] [14]通过加入一个域判别器判断样本来自哪个域，来学习不随域的改变而改变 (domain-invariant) 的特征，其中在优化特征提取器 (feature extractor)、类别分类器和域判别器时使用了一些技巧来加速反向传播 (Backpropogation)。基于数据特征的二阶统计量的方法[15]通过最小化

source 域数据特征和线性变换后的 target 域特征的协方差矩阵 (covariance) 之差的 Frobenius 范数来对齐两个域的特征分布。利用注意机制 (attention mechanism) 的方法[16]首先通过循环生成式对抗神经网络 (CycleGAN) 生成 source 域和 target 域的成对数据，并对齐每一对数据隐藏特征 (hidden feature) 的注意 (attention) 而非仅对齐最后一层的特征。

另一些方法[17] [18]以各类别数据特征平均作为类别原型 (prototype)，并将数据特征传输 (transport) 或者说分配到类别原型，与之相似的，还有方法[19]将分类器参数的各分量作为各类别原型。我们的工作建立在 PCT[20](Prototypical-oriented Conditional Transport) 的基础上，PCT 除了将分类器参数的各分量作为各类别原型外，还考虑了由类别原型到数据特征的传输，我们在其基础上完善了理论推导并提出了改进的算法。

Office31[21], Office-Caltech-10[22], ImageCLEF, VisDA 等为域适应的实验的常用数据集提供了支持。

1.3 论文结构

我们在2介绍我们的模型和算法，其中，2.1建立模型，2.2定义分类原型 (prototype) 和学习分类原型的损失函数，2.3和2.4分别定义从样本特征 (features) 到分类原型的传输损失函数和从分类原型到样本特征的传输损失函数，其中在2.4.1我们用 MLE(Maximum Likelihood Estimate) 的方法估计各分类原型的比例，目标损失函数在最后2.5给出。我们在3介绍我们初步实验的细节和结果。在4我们总结全文。

2 原型导向的条件传输 (Prototype-Oriented Conditional Transport)

在这一节，我们介绍在 PCT(Prototype-oriented Conditional Transport)[20] 基础上改进的算法，改进简要概括为以下三点。

首先，我们认为 PCT[20] 对后验概率和 likelihood 函数的定义可能存在误解，复杂化了从 target 域样本的特征 (feature) 到类别原型 (prototype) 的传输 (transport) 概率。因此我们在2.3中重新推导了此概率的表达式，结果是简单的，且无需用到 target 域上的类别比例。除此之外，在2.4中，我们推导的从类别原型到 target 域样本特征的传输概率也与 PCT 不同。

其次，PCT 提出将从 target 域样本的特征到类别原型的传输代价函数定义为负余弦相似度，即 1 - cosine similarity，在本文中我们简称其为余弦定义。然后 PCT 简要介绍了当定义代价函数为传输概率的负对数时，存在最小化 Loss 实际上等价于最小化传输概率的熵 (entropy) 的问题，我们简称这种定义为对数定义。实际上，在 PCT 参考的有关类别原型的文献[17] [18] 中，类别原型一般被定义为属于该类别的样本特征的平均 (average)，所以类别原型和特征是处于同一空间上同一语义下的，那么用任意衡量向量之间相似度的度量都是合理的，比如余弦相似度和欧式距离[17]。然而在 PCT 及本文中，类别原型被定义为分类器预测该类别概率的函数的参数，此时类别原型和特征在不同语义下，也不一定在同一空间，使用余弦相似度的方法便存疑。实验上，从 PCT 在 Office-31 数据集上的六个任务的实验结果来看，除余弦定义在四个任务上取得最好效果外，另外两个任务对数定义取得最好效果，说明后者方法在一定情况下不逊于前者。另外，在最初定义类别原型为分类器预测该类别概率的函数参数的文章中，代价函数同样定义为了传输概率的负对数[19]。综上所述，由于无法确定哪种定义更适合改进后的算法，在本文中，我们分别在2.3中讨论、在3.2中实验并比较了这两种定义的实验效果。

最后，在计算从类别原型到 target 域样本特征的传输代价期望的过程中，需要估计 target 域上的类别比例，我们在2.4.1中推导与 PCT 不同的 likelihood 函数。除此之外，与 PCT 中非必要地使用 EM 算法估计类别比例不同，我们在2.4.1中直接使用 MLE(Maximum Likelihood Estimate) 估计。经过推导，在大部分情况下，该 MLE 仅在单个样本时有唯一的实现值，即传输概率。我们通过一个技巧将这一 MLE 的结果应用到了批量样本 (a batch sample) 损失函数的计算过程中。

2.1 模型建立

模型部分，我们没有对 PCT 做任何修改。

在无监督域适应 (Unsupervised Domain Adaptation) 领域，通常我们拥有的数据集为：其一，source 域上的带标签 (labeled) 数据，我们记其为 $(\boldsymbol{x}_i^s, y_i^s)$，记此数据服从概率分布 $(\boldsymbol{x}_i^s, y_i^s) \sim \mathcal{P}_s$，实验时，记给定样本大小为 n_s 的数据集为 $\{(\boldsymbol{x}_i^s, y_i^s)\}_{i=1}^{n_s}$；其二，target 域上的无标签 (unlabeled) 数据，我们记其为 $\boldsymbol{x}_j^t$，记此数据服从概率分布 $\boldsymbol{x}_j^t \sim \mathcal{P}_t^{\boldsymbol{x}}$，实验时，记给定样本大小为 n_t 的数据集为 $\{\boldsymbol{x}_j^t\}_{j=1}^{n_t}$。

我们的模型由两部分组成，其一为特征提取器 (features extractor)，其二为分类器。我们记特征提取器函数为 F，其参数为 $\boldsymbol{\theta}$，对输入数据 $\boldsymbol{x}_i^s$(或 $\boldsymbol{x}_j^t$)，记提取后的特征 $f_i^s = F_{\boldsymbol{\theta}}(\boldsymbol{x}_i^s), f_i^s \in \mathbb{R}^{d_f}$(或 f_j^t)，特征提取器通常为深度特征提取器，如 ResNet。我们采用 Softmax 作为分类器，记分类器函数为 C，其参数为 $\boldsymbol{\mu} = [\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K], \boldsymbol{\mu}_k \in \mathbb{R}^{d_f}$，其中，$K$ 表示 K 分类任务，对输入特征 $\boldsymbol{f}_i^s$(或 $\boldsymbol{f}_j^t$)，记对各分类的预测概率为 $p_i^s = [p_{i1}^s, ..., p_{iK}^s]^T = C_{\boldsymbol{\mu}}(\boldsymbol{f}_i^s) = \frac{\exp(\boldsymbol{\mu}^T \boldsymbol{f}_i^s)}{\mathbf{1}^T \exp(\boldsymbol{\mu}^T \boldsymbol{f}_i^s)}$(或 $p_j^s = [p_{j1}^s, ..., p_{jK}^s]^T$)，其中 $\mathbf{1}$ 为全 1 向量。

2.2 分类原型 (Prototype) 及其学习 (learning)

在分类原型的定义和学习算法上，本文与 PCT 完全相同。

不同于一些文献[17] [18]定义分类原型为该分类下样本特征的平均，PCT 和我们定义分类原型为分类器参数的分量[19]。从 Softmax 分类器函数我们可以得知，对特征 f_i^s，第 k 类的预测概率 $p_{ik}^s \sim \exp(\boldsymbol{\mu}_k^T \boldsymbol{f}_i^s)$，可以看出 $\boldsymbol{\mu}_k$ 在所有分类器参数中是影响预测为 k 类的概率的主要部分，因此我们定义 Softmax 的参数矩阵 $\boldsymbol{\mu}$ 的各列向量 $\boldsymbol{\mu}_k, k = 1, ...K$ 为各类别原型。

相比特征平均的定义，这种定义有三个明显的优势：首先这种定义不会像取特征平均一样明显地受到 outlier 的影响；其二，在实验或是实际应用中，我们通常用批量的数据训练模型，而一批数据往往不会太大，从而导致一批数据中可能不含有属于某些分类的样本，那么按分类取特征平均的定义便变得十分困难，我们的定义则绕开了这个问题；其三，当我们无法获得样本特征的具体值时，例如在受限于数据隐私的情况下，我们就无法获得特征的平均，而我们的定义则绕开了这个问题。

这种定义的不足则体现在它只能够应用在 Softmax 分类器这种函数表达式和参数格式性质良好的分类器上，另一个例子是线性分类器，即 $C_{\boldsymbol{\mu}}(\boldsymbol{f}_i^s) = \frac{\boldsymbol{\mu}^T \boldsymbol{f}_i^s}{\mathbf{1}^T \boldsymbol{\mu}^T \boldsymbol{f}_i^s}$，对于决策树这种非参数分类器和多分类支持向量机 (Support Vector Machine) 这种

没有针对一个类别的预测概率的函数表达式的分类器，这种定义是困难的。

我们通过 source 域数据来学习类别原型 $\boldsymbol{\mu}_k$ 的具体值。和分类任务通常的做法一样，我们定义损失函数为交叉熵损失 (Cross Entropy Loss)，

$$\begin{aligned} L_{cls}(\boldsymbol{\theta}, \boldsymbol{\mu}) =& \mathbf{E}_{(\boldsymbol{x}_i^s, y_i^s)\sim\mathcal{P}_s} \sum_{k=1}^{K} -\mathbf{1}_{\{y_i^s=k\}} \log p_{ik}^s(\boldsymbol{\theta}, \boldsymbol{\mu}) \\ =& \mathbf{E}_{(\boldsymbol{x}_i^s, y_i^s)\sim\mathcal{P}_s} \sum_{k=1}^{K} -\mathbf{1}_{\{y_i^s=k\}} \log \frac{\exp\left(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_i^s)\right)}{\mathbf{1}^T \exp\left(\boldsymbol{\mu}^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_i^s)\right)}. \end{aligned}$$

2.3 Target 域样本特征 (feature) 到分类原型的传输 (transport) 损失函数

在无监督域适应问题上，通常的做法是根据不同的对齐 (align) 规则，通过变换使 target 域上样本特征的分布对齐 source 域上样本特征的分布，从而能够将 source 域上训练的模型迁移到 target 域上应用。与 PCT 相同，我们提出的对齐方法是以各分类原型为标准，最小化 target 域上样本特征传输到分类原型的代价的期望，但在具体细节，即 target 域样本特征到类别原型的传输概率 $p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t)$ 的推导上，我们的做法与 PCT 不同。

为了最小化传输代价的期望，我们需要先定义代价函数，再将代价函数的期望作为损失函数。前文已经提到，对数定义将代价函数定义为传输概率的负对数，即

$$c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t) = -\log p_{jk}^t(\boldsymbol{\theta}) = -\log \frac{\exp\left(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}{\mathbf{1}^T \exp\left(\boldsymbol{\mu}^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}.$$

余弦定义将代价函数定义为负余弦相似度，即

$$c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t) = 1 - \frac{\boldsymbol{\mu}_k^T \boldsymbol{f}_j^t}{||\boldsymbol{\mu}_k||_2 ||\boldsymbol{f}_j^t||_2}.$$

于是我们的损失函数即为传输代价期望，

$$\begin{aligned} L_{t\to\boldsymbol{\mu}}(\boldsymbol{\theta}) =& \mathbf{E}_{\boldsymbol{f}_j^t} \mathbf{E}_{\boldsymbol{\mu}_k\sim p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t)} c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t) \\ =& \mathbf{E}_{\boldsymbol{x}_j^t\sim\mathcal{P}_t^{\boldsymbol{x}}} \mathbf{E}_{\boldsymbol{\mu}_k\sim p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t(\boldsymbol{x}_j^t))} c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t(\boldsymbol{x}_j^t)), \end{aligned}$$

其中 $\mathcal{P}_t^{\boldsymbol{x}}$ 可以由 target 域数据集来估计。

对于 $p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t)$，我们认为 PCT[20]对 likelihood 函数存在误解。在 PCT 中，

$$\begin{aligned} p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t) \sim & p(\boldsymbol{\mu}_k)L(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t), \\ L(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t) \sim & \exp(\boldsymbol{\mu}_k^T\boldsymbol{f}_j^t) \sim p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t) \end{aligned}$$

我们推导与 PCT 不同

$$p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t) = p(y_j^t = k|\boldsymbol{f}_j^t) = p_{jk}^t(\boldsymbol{\theta}) = \frac{\exp\left(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}{\mathbf{1}^T \exp\left(\boldsymbol{\mu}^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}. \tag{2.1}$$

于是我们得到损失函数

$$L_{t\to\boldsymbol{\mu}}(\boldsymbol{\theta}) = \mathbf{E}_{\boldsymbol{x}_j^t\sim\mathcal{P}_t^{\boldsymbol{x}}} \sum_{k=1}^{K} p_{jk}^t(\boldsymbol{\theta}) c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t).$$

实际上，最小化 $L_{t\to\boldsymbol{\mu}}(\boldsymbol{\theta})$ 等价于最小化分布 p_{jk}^t 的熵 (entropy)，而这有时会导致的一个问题是有些分类原型没有被任何样本特征传输，例如一个极端的例子是所有样本特征都传输到一个分类原型上[23] [24]。

2.4 分类原型到 Target 域样本特征的传输损失函数

和 PCT 相同，为了避免存在分类原型没有被任何样本特征传输的问题，我们从相反方向定义传输代价期望

$$\begin{aligned} L_{\boldsymbol{\mu}\to t}(\boldsymbol{\theta}) =& \mathbf{E}_{\boldsymbol{\mu}_k}\mathbf{E}_{\boldsymbol{f}_j^t\sim p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k)} c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t) \\ =& \mathbf{E}_{\boldsymbol{x}_{j_1}^t\sim\mathcal{P}_t^{\boldsymbol{x}}}\mathbf{E}_{\boldsymbol{\mu}_k\sim p(\boldsymbol{\mu}_k|\boldsymbol{x}_{j_1}^t)}\mathbf{E}_{\boldsymbol{f}_j^t\sim p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k)} c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t). \end{aligned}$$

但在类别原型到 target 域样本特征的传输概率上，我们的推导与 PCT 不同。设 target 域一批样本的批大小 (batch size) 为 B，由 Bayes 公式，有

$$\begin{aligned} p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k) =& \frac{p(\boldsymbol{f}_j^t)p(\boldsymbol{\mu}_k|\boldsymbol{f}_j^t)}{\sum_{j'=1}^{B} p(\boldsymbol{f}_{j'}^t)p(\boldsymbol{\mu}_k|\boldsymbol{f}_{j'}^t)} \\ =& \frac{p(\boldsymbol{f}_j^t)p_{jk}^t(\boldsymbol{\theta})}{\sum_{j'=1}^{B} p(\boldsymbol{f}_{j'}^t)p_{j'k}^t(\boldsymbol{\theta})}. \end{aligned}$$

于是，

$$p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k) = \frac{p(\boldsymbol{f}_j^t)\exp(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t))}{\sum_{j'=1}^{B} p(\boldsymbol{f}_{j'}^t)\exp(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_{j'}^t))} \tag{2.2}$$

PCT 紧接着推导出

$$p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k) = \frac{\exp(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t))}{\sum_{j'=1}^{B} \exp(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_{j'}^t))}, \tag{2.3}$$

而这个结论仅在对任意 j，$p(\boldsymbol{f}_j^t) = \frac{1}{B}$ 的情况下才是正确的。

对每个类别原型 $\boldsymbol{\mu}_k$，总存在 j 使得 $p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k) > 0$，因此每个分类原型都将传输到一些样本特征。

2.4.1 估计 target 域上的类别比例 $p(\boldsymbol{\mu}_k|\boldsymbol{x}_{j_1}^t)$

我们估计类别比例的方法与 PCT 完全不同。

在 PCT 中，$p(\boldsymbol{\mu}_k|\boldsymbol{x}_j^t)$ 被理想化地替换成了 $p(\boldsymbol{\mu}_k)$，并且由于2.1处对 likelihood 函数的误解，估计 $p(\boldsymbol{\mu}_k)$ 的 EM 算法中 E 步的推导出现了问题。实际上，我们并不需要通过最大化 $\{\boldsymbol{\mu}_k\}_{k=1}^K$ 的联合 likelihood 函数来估计每一个 $\boldsymbol{\mu}_k$，我们只需最大化 $\boldsymbol{\mu}_k$ 的 likelihood 函数，这样寻求极值的方法就不再复杂。

似然函数

$$L(p(\boldsymbol{\mu}_k)|\boldsymbol{X}^t) = p(\boldsymbol{X}^t|p(\boldsymbol{\mu}_k)) = \prod_{j=1}^{B} p(\boldsymbol{x}_j^t|p(\boldsymbol{\mu}_k)).$$

考虑

$$p(\boldsymbol{x}_j^t|p(\boldsymbol{\mu}_k)) \sim p(\boldsymbol{x}_j^t)p(p(\boldsymbol{\mu}_k)|\boldsymbol{x}_j^t) = p(\boldsymbol{x}_j^t)\mathbf{1}_{\{p(\boldsymbol{\mu}_k)=p_{jk}^t\}},$$

即在给定 $\boldsymbol{x}_j^t$ 的条件下，$p(\boldsymbol{\mu}_k)$ 服从单点分布。于是似然函数

$$L(p(\boldsymbol{\mu}_k)|\boldsymbol{X}^t) = p(\boldsymbol{X}^t|p(\boldsymbol{\mu}_k)) \sim \prod_{j=1}^{B} p(\boldsymbol{x}_j^t)\mathbf{1}_{\{p(\boldsymbol{\mu}_k)=p_{jk}^t\}}.$$

通过 $\mathbf{1}_{\{p(\boldsymbol{\mu}_k)=p_{jk}^t\}}$ 这一因子我们可以发现，当且仅当对于任意 j，p_{jk}^t 全都相等，且 $p(\boldsymbol{\mu}_k) = p_{jk}^t$ 时 likelihood 函数有唯一的大于 0 的也是最大的值，其余情况函数值皆为 0，这在批大小大于 1，即 $B > 1$ 的情况下几乎是不可能的，也就是说 MLE 几乎不

存在唯一解。因此我们考虑批大小为 1，即 $B=1$，此时 $L(p(\boldsymbol{\mu}_k)|\boldsymbol{X}^t)=L(p(\boldsymbol{\mu}_k)|\boldsymbol{x}_j^t)$，则 MLE 即为

$$\hat{p}(\boldsymbol{\mu}_k|\boldsymbol{x}_j^t)=p_{jk}^t,$$

并且，传输代价的期望则为条件期望的期望，即 $\mathbf{E}_{\boldsymbol{x}_j^t\sim\mathcal{P}_t^{\boldsymbol{x}}}\mathbf{E}_{\boldsymbol{\mu}_k\sim p(\boldsymbol{\mu}_k|\boldsymbol{x}_j^t)}$。

结论是符合直觉的，给定一个输入 x 数据集，我们希望估计该数据集的各类别 y 的比例，那么对每个输入 x_j，对每个分类 $y=k$ 的比例的最优估计就是对该分类的预测概率，然后每个 x_j 用他们的估计结果进行投票，从而得到由整个数据集对类别比例的估计。

现在我们考虑传输代价期望 $L_{\boldsymbol{\mu}\to t}(\boldsymbol{\theta})$，

$$\begin{aligned}L_{\boldsymbol{\mu}\to t}(\boldsymbol{\theta})=&\mathbf{E}_{\boldsymbol{x}_{j_1}^t\sim\mathcal{P}_t^{\boldsymbol{x}}}\mathbf{E}_{\boldsymbol{\mu}_k\sim p(\boldsymbol{\mu}_k|\boldsymbol{x}_{j_1}^t)}\mathbf{E}_{\boldsymbol{f}_j^t\sim p(\boldsymbol{f}_j^t|\boldsymbol{\mu}_k)}c(\boldsymbol{\mu}_k,\boldsymbol{f}_j^t)\\=&\mathbf{E}_{\boldsymbol{x}_{j_1}^t\sim\mathcal{P}_t^{\boldsymbol{x}}}\sum_{k=1}^{K}p_{j_1k}^t(\boldsymbol{\theta})\sum_{j=1}^{B}\frac{p(\boldsymbol{f}_j^t)p_{jk}^t(\boldsymbol{\theta})c(\boldsymbol{\mu}_k,\boldsymbol{f}_j^t)}{\sum_{j'=1}^{B}p(\boldsymbol{f}_{j'}^t)p_{j'k}^t(\boldsymbol{\theta})}.\end{aligned}$$

2.5 损失函数及优化

经过一系列修改，现在我们定义我们的损失函数为

$$L(\boldsymbol{\theta},\boldsymbol{\mu})=L_{cls}(\boldsymbol{\theta},\boldsymbol{\mu})+\lambda_t(\lambda_bL_{t\to\boldsymbol{\mu}}(\boldsymbol{\theta})+(1-\lambda_b)L_{\boldsymbol{\mu}\to t}(\boldsymbol{\theta})).$$

如果在批大小分别为 B_s 和 B_t 的训练数据内每个样本仅出现一次，则可以简化各损失函数为

$$
\begin{aligned}
L_{cls}(\boldsymbol{\theta}, \boldsymbol{\mu}) =& \sum_{i=1}^{B_s} \frac{1}{B_s} \sum_{k=1}^{K} -\mathbf{1}_{\{y_i^s=k\}} \log p_{ik}^s(\boldsymbol{\theta}, \boldsymbol{\mu}), \\
\text{where } p_{ik}^s(\boldsymbol{\theta}, \boldsymbol{\mu}) =& \frac{\exp\left(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_i^s)\right)}{\mathbf{1}^T \exp\left(\boldsymbol{\mu}^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_i^s)\right)}, \\
L_{t\to\boldsymbol{\mu}}(\boldsymbol{\theta}) =& \sum_{j=1}^{B_t} \frac{1}{B_t} \sum_{k=1}^{K} p_{jk}^t(\boldsymbol{\theta}) c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t), \\
L_{\boldsymbol{\mu}\to t}(\boldsymbol{\theta}) =& \sum_{j_1=1}^{B_t} \frac{1}{B_t} \sum_{k=1}^{K} p_{j_1k}^t(\boldsymbol{\theta}) \sum_{j=1}^{B} \frac{p_{jk}^t(\boldsymbol{\theta}) c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t)}{\sum_{j'=1}^{B} p_{j'k}^t(\boldsymbol{\theta})}, \\
=& \sum_{k=1}^{K} \Big(\sum_{j_1=1}^{B_t} \frac{1}{B_t} p_{j_1k}^t(\boldsymbol{\theta})\Big)\Big(\sum_{j=1}^{B} \frac{p_{jk}^t(\boldsymbol{\theta}) c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t)}{\sum_{j'=1}^{B} p_{j'k}^t(\boldsymbol{\theta})}\Big) \\
\text{where } p_{j_k}^t(\boldsymbol{\theta}) =& \frac{\exp\left(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}{\mathbf{1}^T \exp\left(\boldsymbol{\mu}^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}.
\end{aligned}
$$

对于传输代价 $c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t)$，若使用对数定义，则

$$
c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t) = -\log p_{jk}^t(\boldsymbol{\theta}) = -\log \frac{\exp\left(\boldsymbol{\mu}_k^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}{\mathbf{1}^T \exp\left(\boldsymbol{\mu}^T F_{\boldsymbol{\theta}}(\boldsymbol{x}_j^t)\right)}.
$$

若使用余弦定义，则

$$
c(\boldsymbol{\mu}_k, \boldsymbol{f}_j^t) = 1 - \frac{\boldsymbol{\mu}_k^T \boldsymbol{f}_j^t}{||\boldsymbol{\mu}_k||_2 ||\boldsymbol{f}_j^t||_2}.
$$

我们的算法寻找损失函数的最小值点，而两种传输代价函数定义代表了两种算法，其中，我们记使用对数定义的算法为算法 ours，使用余弦定义的算法为算法 ours-cos。优化算法可选择 SGD(Stochastic Gradient Descent 随机梯度下降) 算法或 Adam(Adaptive Movement Estimation algorithm) 算法。

3 实验

由于时间和计算资源的限制，我们对 PCT、算法 ours、算法 ours-cos 进行了初步实验，初步意为我们没有通过调整各超参数来使每个算法在数据集上达到最好效果，而是统一使用默认的超参数，以及使用 ResNet18 而非 ResNet50 作为特征提取器，且在每个任务上使用每种方法执行 3 次实验。可从 https://github.com/Grape20/ThesisPCT 获取实验 Pytorch 代码。

3.1 实验设定和细节

我们使用算法 PCT、算法 ours、算法 ours-cos，在数据集 Office31[21] 上进行图像分类实验。Office31 是一个带标签的 (labeled) 图片数据集，数据来自不同的三个域 Amazon、DSLR、Webcam，以下简称 A、D、W，每个域上的数据被分为相同的 31 个分类，如背包、笔记本等。实验按 source 域和 target 域的不同分为 6 个任务，即 A→D、A→W、D→A、D→W、W→A、W→D。我们使用 ResNet18 全连接一层瓶颈层 (bottleneck) 作为特征提取器，使用 SGD 作为优化器 (optimizer)。

超参数方面，设定 $\lambda_t = 1, \lambda_b = 0.5$。在算法 PCT 中，与 PCT 原文相同，设定 EM 算法估计类别比例中的步进率 $\beta = 0.001$。每个任务执行 32 个 epoch，每个 epoch 执行 10 个批次。我们设置自动批大小，以实现在每个 epoch 中尽可能多的样本被用到一次，少部分样本未被用到，另外，实验发现当 target 域上训练集大小大于 source 域上时具有较高准确率，因此我们设置 target 域上的批次大小为 source 域上批次大小的 3 倍。

我们使用 source 域的所有数据和 target 域上 80% 的数据进行训练，使用 target 域上 10% 的数据在每个 epoch 测试模型的准确率。

3.2 结果

三种方法在测试过程中的最高准确率如表3.1所示，测试过程中准确率的变化如图3.1所示。

表 3.1 最高测试准确率

Method	A→D	A→W	D→A	D→W	W→A	W→D
ours	72.464	79.583	61.210	97.917	58.600	**100.000**
ours-cos	**74.638**	**81.250**	62.515	**98.750**	62.515	**100.000**
PCT	73.188	80.833	**65.718**	96.667	**62.989**	98.551

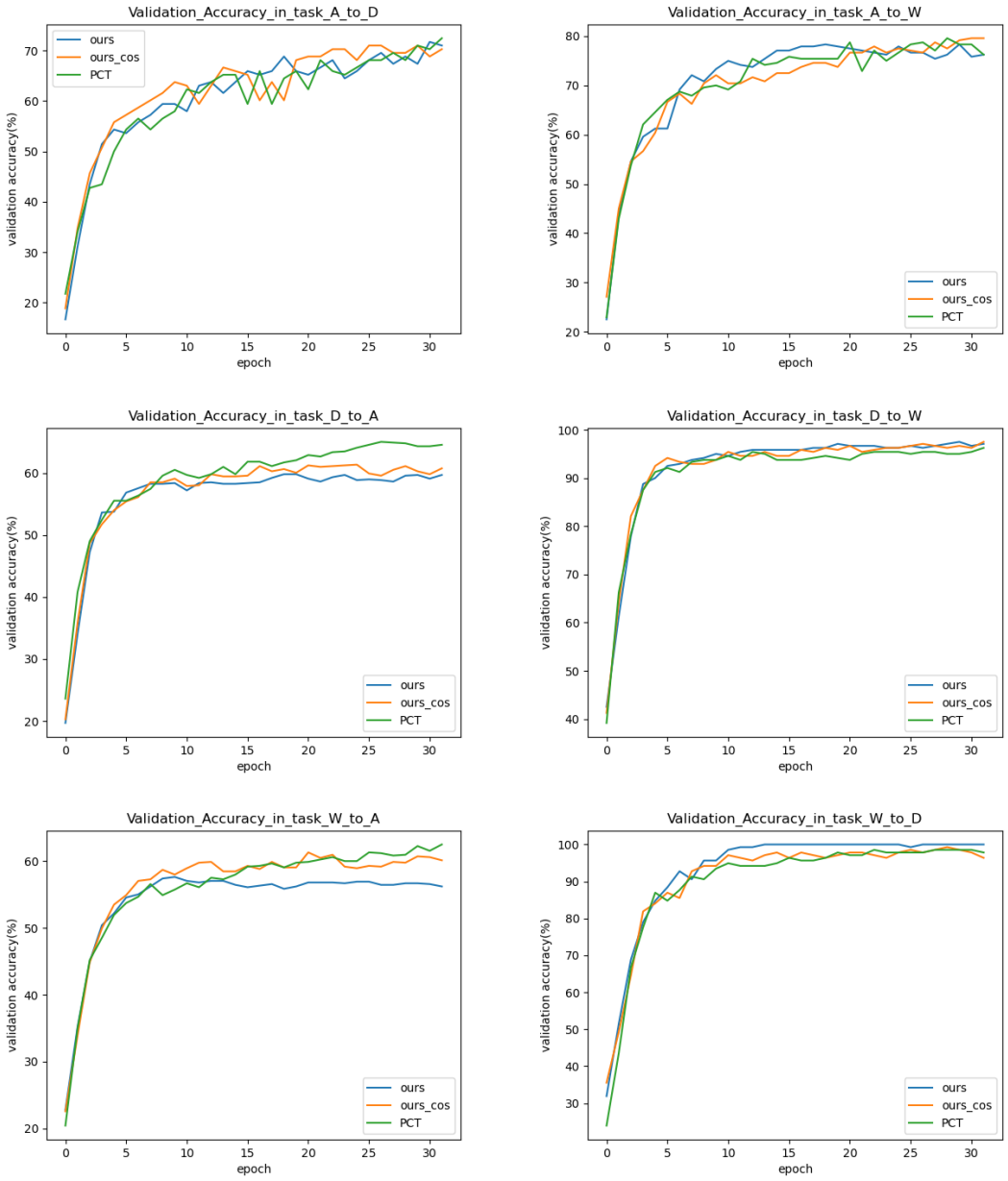

图 3.1 测试准确率折线图

3.3 分析

初步实验结果由表3.1可以看出，在 6 个任务中，算法 ours 在 2 个任务上取得比原 PCT 更好的分类效果，在 4 个任务上算法 ours-cos 取得了比 PCT 更好的分类效果，在 5 个任务上算法 ours-cos 取得了比算法 ours 更好的分类效果。分析实验结果可以得出结论，经过对各传输概率和类别比例估计的改进，算法 ours-cos 在实验上取得了比 PCT 更好的效果，但在传输代价函数的定义上，用对数定义替代余弦定义的算法 ours 在实验上表现不佳。

4 结论

本文在PCT[20]的基础上通过完善理论推导提出了改进的算法ours和算法ours-cos，主要包括：重新推导了从样本特征到类别原型和从类别原型到样本特征的传输概率，讨论了传输代价函数的定义，使用了MLE而非复杂的EM算法估计类别比例。我们建立了无监督域适应领域常用的特征提取器加分类器的模型，并推导出了改进算法使用的损失函数。我们在Office31[21]上初步实验了两个改进算法和算法PCT，实验结果显示经过对各传输概率和类别比例估计的改进，算法ours-cos在实验上取得了比PCT更好的效果，但在传输代价函数的定义上，用对数定义替代余弦定义的算法ours在实验上表现不佳。

参考文献

[1] SHIMODAIRA H. Improving predictive inference under covariate shift by weighting the log-likelihood function[J]. Journal of statistical planning and inference, 2000, 90(2): 227-244.

[2] BEN-DAVID S, BLITZER J, CRAMMER K, et al. Analysis of representations for domain adaptation[J]. Advances in neural information processing systems, 2006, 19.

[3] BLITZER J, DREDZE M, PEREIRA F. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification[C]//Proceedings of the 45th annual meeting of the association of computational linguistics. 2007: 440-447.

[4] PAN S J, YANG Q. A survey on transfer learning[J]. IEEE Transactions on knowledge and data engineering, 2009, 22(10): 1345-1359.

[5] BORGWARDT K M, GRETTON A, RASCH M J, et al. Integrating structured biological data by kernel maximum mean discrepancy[J]. Bioinformatics, 2006, 22(14): e49-e57.

[6] GRETTON A, BORGWARDT K M, RASCH M J, et al. A kernel two-sample test[J]. The Journal of Machine Learning Research, 2012, 13(1): 723-773.

[7] LONG M, WANG J, DING G, et al. Transfer feature learning with joint distribution adaptation [C]//Proceedings of the IEEE international conference on computer vision. 2013: 2200-2207.

[8] PAN S J, TSANG I W, KWOK J T, et al. Domain adaptation via transfer component analysis[J]. IEEE transactions on neural networks, 2010, 22(2): 199-210.

[9] BAKTASHMOTLAGH M, HARANDI M, SALZMANN M. Distribution-matching embedding for visual domain adaptation[J]. Journal of Machine Learning Research, 2016, 17: Article-number.

[10] ZHONG E, FAN W, PENG J, et al. Cross domain distribution adaptation via kernel mapping [C]//Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. 2009: 1027-1036.

[11] YAN H, DING Y, LI P, et al. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2272-2281.

[12] GANIN Y, LEMPITSKY V. Unsupervised domain adaptation by backpropagation[C]// International conference on machine learning. PMLR, 2015: 1180-1189.

[13] CHEN X, WANG S, LONG M, et al. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation[C]//International conference on machine learning. PMLR, 2019: 1081-1090.

[14] TANG H, JIA K. Discriminative adversarial domain adaptation[C]//Proceedings of the AAAI Conference on Artificial Intelligence: volume 34. 2020: 5940-5947.

[15] SUN B, FENG J, SAENKO K. Return of frustratingly easy domain adaptation[C]//Proceedings of the AAAI Conference on Artificial Intelligence: volume 30. 2016.

[16] KANG G, ZHENG L, YAN Y, et al. Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 401-416.

[17] SNELL J, SWERSKY K, ZEMEL R. Prototypical networks for few-shot learning[J]. Advances in neural information processing systems, 2017, 30.

[18] PAN Y, YAO T, LI Y, et al. Transferrable prototypical networks for unsupervised domain adaptation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 2239-2247.

[19] SAITO K, KIM D, SCLAROFF S, et al. Semi-supervised domain adaptation via minimax entropy [C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 8050-8058.

[20] TANWISUTH K, FAN X, ZHENG H, et al. A prototype-oriented framework for unsupervised domain adaptation[J]. Advances in Neural Information Processing Systems, 2021, 34.

[21] SAENKO K, KULIS B, FRITZ M, et al. Adapting visual category models to new domains [C]//DANIILIDIS K, MARAGOS P, PARAGIOS N. Computer Vision – ECCV 2010. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010: 213-226.

[22] GONG B, SHI Y, SHA F, et al. Geodesic flow kernel for unsupervised domain adaptation[C]// 2012 IEEE conference on computer vision and pattern recognition. IEEE, 2012: 2066-2073.

[23] MORERIO P, CAVAZZA J, MURINO V. Minimal-entropy correlation alignment for unsupervised deep domain adaptation[J]. arXiv preprint arXiv:1711.10288, 2017.

[24] WU X, ZHANG S, ZHOU Q, et al. Entropy minimization versus diversity maximization for domain adaptation[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021.

致谢

能够完成这篇毕业论文，首先我要感谢我的指导老师任传贤老师，在他的耐心指导下我对 UDA 这一领域有了大概的认识并完成了我的毕业论文。其次我要感谢罗又维学长，他对我在代码上的帮助和与我的讨论让我少走了一些弯路。

胡恩

2022 年 7 月 12 日

毕业论文 (设计) 成绩评定记录

Grading Sheet of the Graduation Thesis (Design)

指导教师评语 Comments of Supervisor: 成绩评定 Grade: 指导教师签名 Supervisor Signature: Date:
答辩小组或专业负责人意见 Comments of the Defense Committee: 成绩评定 Grade: 签名: Date: Signatures of Committee Members
院系负责人意见 Comments of the Academic Chief of School: 成绩评定 Grade: 签名 院系盖章 Signature: Stamp: Date: