# 调研二

北邮2016级-计算机学院-网络工程-张婷旖

对于主动学习，增量学习，强化学习，迁移学习，其实经过上一次的资料查找，我的个人观感是，迁移学习是目前和未来一段时间内机器学习的一个发展趋势，所以我准备收集一些“迁移学习”方向的当前进展。

## 迁移学习概述

### 概念

从以前的任务当中去学习知识（knowledge）或经验，并应用于新的任务当中。换句话说，迁移学习目的是从一个或多个源任务（source tasks）中抽取知识、经验，然后应用于一个不同但是相关的目标领域（target domain）当中去。（比如说，人类在学会骑自行车后，再骑摩托车就很容易了，这其实是一种适应的能力）

### 难点

找出不变量（在一些学习任务中有一些特征是个体所特有的，这些特征不可以迁移。而有些特征是在所有的个体中具有贡献的，这些可以进行迁移。）

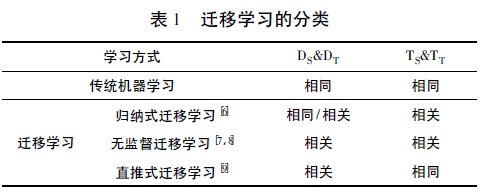
### 使用场景

资源不足（数据或者运算力的不足）的 AI 项目

### 基本思路

利用预训练模型，即已经通过现成的数据集训练好的模型，在其中找到能够输出可复用特征的层次，然后利用该层次的输出作为输入特征来训练那些需要参数较少的规模更小的神经网络。

### 分类



### 研究目标

1. 解决小数据问题．
   * 假设训练数据与测试数据服从相同的数据分布。当训练数据过少时，经典监督学习会出现严重过拟合问题，而迁移学习可从源域的小数据中抽取并迁移知识，用来完成新的学习任务．
2. 解决个性化问题
   * 当需要专注于某个目标领域时，源领域范围太广却不够具体．利用迁移学习可以将源领域上的预训练模型特征迁移到目标域，实现个性化．

### 应用领域

1. 自然语言处理:
   * 自然语言领域标注和内容数据稀缺．可以利用源域( 例如英语) 中标注的样本集来对目标域( 例如法语) 中的样本进行处理．
   * 迁移学习能够从长文本中迁移标注和内容知识，帮助处理短文本语言的分析与处理．
2. 计算机视觉:
   * 标注数据与未标注数据具有不同的数据属性和统计分布，用传统机器学习无法满足要求．迁移学习算法能够将领域适配，进而达到训练效果，提升准确率．
3. 医疗健康和生物信息学:
4. 从模拟中学习:
   * 风险较小
   * 源数据域和目标域的特征空间是一样的，但是模拟和现实世界的边缘概率分布是不一样的，即模拟和目标域中的物体看上去是不同的．模拟环境和现实世界的条件概率分布可能是不一样的，不会完全模仿现实世界中的物体交互．
5. 用户评价:
   * 在评价用于对某服装品牌的情感分类任务中，我们无法收集到非常全面的用户评价的数据．因此当我们直接通过之前训练好的模型进行情感识别时，效果必然会受到影响．
   * 迁移学习可以将少量与测试数据相似的数据作为训练集进行训练，能达到较好的分类效果，并且节省大量的时间和精力．
6. 推荐系统:
   * 在源领域训练好一个推荐系统，然后应用在稀疏的、新的目标领域．
   * 比如已经成熟完善的电影推荐系统就可以应用在书籍推荐系统中
7. 个性化对话:
   * 训练一个通用型的对话系统，该系统可能是闲聊型，也可能是任务型的．
   * 我们可以利用迁移学习训练特定领域的小数据集，使这个对话系统适应不同任务．

## 迁移学习算法

### 归纳式迁移学习

#### TrAdaBoost 迁移学习算法

传统的机器学习，有一个假设: 训练集和测试集分布相同。但如果来了一批新的数据（new data），分布与之前的训练的旧数据（old data）不一样，那么我们的算法的准确率就会下降很多。使用新数据，完全抛弃旧数据，有两个问题：

1. 新数据可能量不够；
2. 旧数据可能还有利用价值，完全抛弃太浪费。

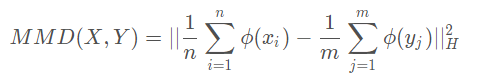
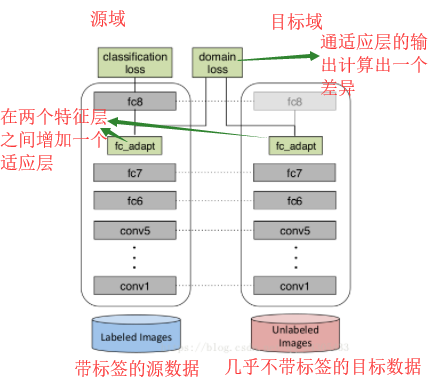
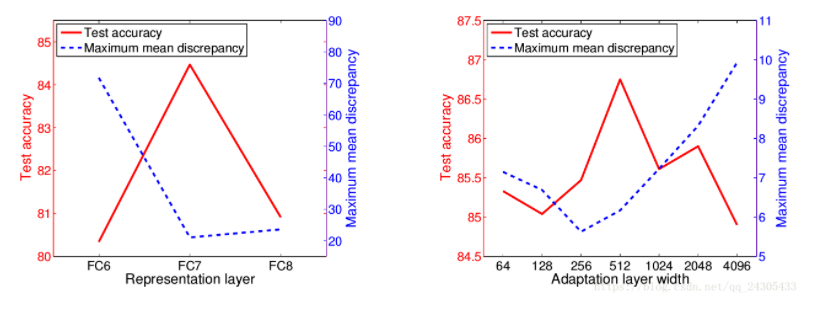
所以才有了迁移学习，从旧数据从提取信息，用于新的模型训练。

TrAdaBoost算法：从old data中提取实例，将一部分能用的old labeled data，结合new labeled data（可能是少量），构建出比单纯使用new labeled data训练更精确的模型。

利用AdaBoost算法的思想原理：

* 当目标域中的样本被错误地分类之后，可以认为这个样本是很难分类的，增大这个样本在分类器中所占的比重．
* 如果源域中的一个样本被错误地分类了，可以认为这个样本对于目标数据是不同的，降低这个样本在分类器中所占的比重．

为了解决迁移学习中的领域适配问题，基于 DDC( Deep Domain Confusion， <u>深度领域适配</u>) 提出了<u>深度适配网络</u>DAN

* DDC：
  + 在源域与目标域之间添加了一层适应层及添加域混淆损失函数，来让网络在学习如何分类的同时，来减小源域及目标域之间的分布差异，从而实现域的自适应
  + 方法：
    - 在两个流向的网络的特征层之间增加一个适应层，并通过适应层的输出计算出一个domain loss（以利用源域及目标域特征之间的MMD【最大均值差异】来作为domain loss为例，通过最小化MMD来减小源于与目标域之间的差异）
      * MMD ：把源域和目标域用一个相同的映射， 映射到一个再生核希尔伯特空间( RKHS) 中， 然后求映射后两部分数据的均值差异， 就当作是两部分数据的差异．MMD中核是固定的， 可以选择是高斯核或者线性核．
      * 
      * 本质上是要找到一个变换函数，使得变换后的源域数据和目标域数据的距离是最小的。所以这其中就要涉及如何度量两个域中数据分布差异的问题，因此也就用到了MMD
    - 
    - 最终，网络的损失函数fig:
    - fig:是分类损失函数
    - fig:衡量了源域与目标于之间的距离。
    - 适应层（adaptation layer）位置的选取及大小的设定
      * 在适应层位置的选取上，通过逐层的计算源数据与目标数据之间的MMD距离，来选择最小的MMD距离的层的位置为适应层的位置。
      * 确定适应层的位置后，同样通过尝试不同的尺寸的适应层，选择能使MMD距离最小的尺寸来作为适应层的尺寸。
      * eg：fc7层，尺寸为256
      * 
* DAN
  + 深度迁移学习方法， 它适配高层网络 DDC，并加入了多核的 MK-MMD，用多个核去构造总的核， 这样效果比一个核更好．
  + 解决了 DDC 的两个问题：
    - DDC 只适配了一层网络， 而 DAN 适配最后三层  
      ( 6～8 层) ， 网络的迁移能力在最后三层开始就会变得专化( specific) ， 所以要重点适配这三层．
    - DDC 是用了单一核的 MMD， 单一固定的核可能  
      不是最优的核．DAN 用了多核的 MMD( MK－MMD) ，效果比 DDC 更好．

#### 部分迁移学习”( Partial Transfer Learning)

通过筛选，选出源域中与目标域的标签空间对应的那部分数据进行迁移，通过 SAN 【选择对抗网络】来处理部分迁移问题，他可以很好地学习领域不变特征 ，从而在迁移学习中能发挥很大作用．（通常情况下是一个大的域（源域）向一个小的域（目标域）的部分迁移问题，源域的数据量比目标域大得多，源域的标签空间比目标域大，而目标域则是源域标签空间的一个子集）

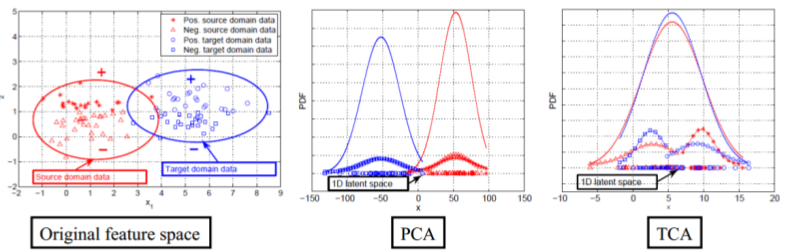
### 无监督迁移学习

#### 迁移成分分析 TCA

数据维度降维技术，基于特征的迁移学习方法。

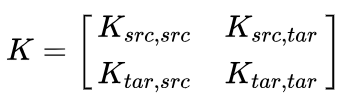
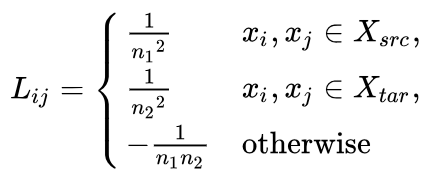
和PCA主成分分析法很像：PCA是大矩阵进去，小矩阵出来。TCA是是两个大矩阵进去，两个小矩阵出来。

TCA 针对域适配问题中，源域和目标域处于不同数据分布时，将两个领域的数据一起映射到一个高维的再生核希尔伯特空间。在此空间中，最小化源和目标的数据距离，同时最大程度地保留它们各自的内部属性。



TCA 的假设源域和目标域的边缘分布是不一样的。但，TCA 假设存在一个特征映射 ，使得映射后数据的分布

迁移学习的本质是最小化源域和目标域的距离，TCA利用的距离是最大均值差异MMD，求映射后源域和目标域的均值之差

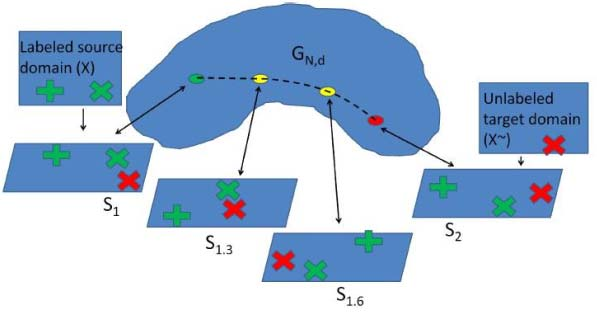
* 首先输入两个特征矩阵， 计算 L 和 H 矩阵
  + 
  + 
* 然后选择常用的核函数进行映射( 比如线性核、 高斯核) 计算 K
* 接着求的前 m个特征值，得到源域和目标域的降维后的数据
* 最后就可以在降维后的数据上使用传统机器学习方法了

实现简单，没有太多的限制。

绕开了半定规划问题的求解， 却需要花费很多计算时间在大矩阵伪逆的求解以及特征值分解．

#### SGF

把源域和目标域分别看成高维空间中的两个点， 在这两个点的测地线上取 有限个中间点， 依次连接起来．然后由源域和目标域就构成了一条测地线流．找到每一步的变换， 就能从源域变换到目标域．

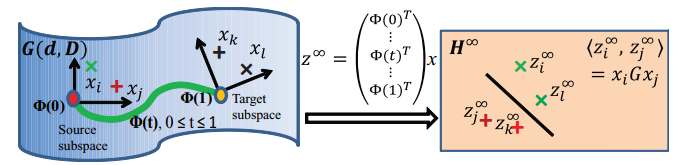


#### GFK

通过一个特征映射， 把源域和目标域变换到一个距离最小  
( 相似度最高) 的公共空间上．

GFK 方法的实施步骤为:

* 继承了SGF方法,采样无穷个点
* 转化成Grassmann流形中的核学习,构建了GFK
* 计算测地线流式核， 构建分类器．



#### 视觉领域自适应(VDA)

利用联合转移学习和领域适应来处理分布差异较大的转换问题， 特别是视觉数据集。 以无监督的方式，在测试集中没有可用标签的情况下减少跨域的联合边际和条件分布。对于不同的任务，边缘分布和条件分布并不是同等重要。

* 数据整体差异性大 (相似度较低),边缘分布更重要
* 数据整体差异性小 (协方差漂移),条件分布更重要

这里边缘分布更占优，应该优先适配。

此外， VDA 构造了嵌入表示中的凝聚域不变集群， 以将各个域与类转移分开， 使用细化的伪目标标签来迭代收敛至最终解决方案．采用迭代过程以及新颖的优化问题为跨领域的适应创建一个稳健而有效的表示．

### 直推式迁移学习

#### 联合分布适配方法JDA

解决迁移学习中的领域适配问题， 用源数据域来标定目标数据域，JDA 假设 DS和 DT边缘分布不同， DS和 DT 条件分布不同，适配联合概率恰好能解决这个问题．具体步骤是:

* 首先用TCA 来适配边缘分布，用 MMD 适配 DS 和 DT的条件概率分布
* 然后通过弱分类器迭代， 将上一轮得到的标签作伪标签， 迭代多次以达到更高的精度．
* 

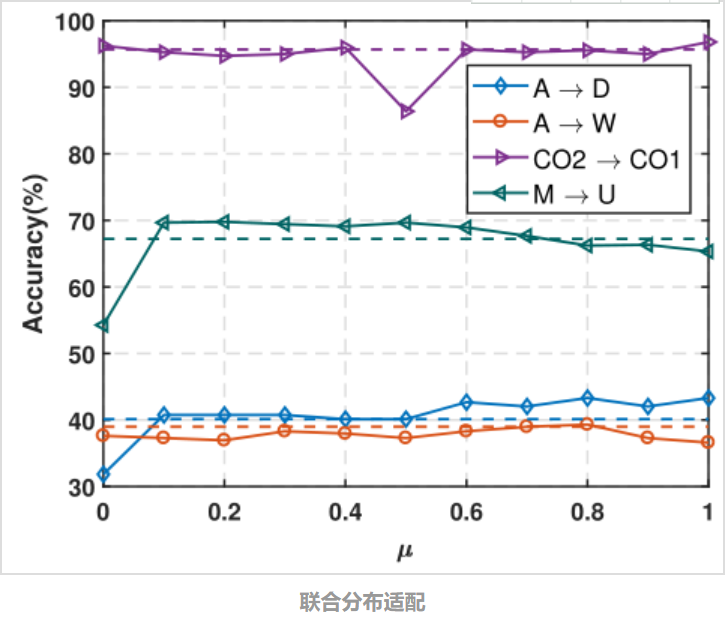
JDA与 TCA 的区别有两点:

* TCA 是无监督的， 即边缘分布适配不需要标签， JDA 需要源域有标签;
* TCA 不需要迭代， JDA 需要迭代．

还有一类非常重要的，联合分布适配BDA，使用**平衡因子**来动态衡量两种分布的重要性fig:

* 当μ→0，表示边缘分布更占优，应该优先适配
* 当μ→1，表示条件分布更占优，应该优先适配

平衡因子的重要性在于：对于不同的任务，边缘分布和条件分布并不是同等重要，因此，BDA方法可以有效衡量这两个分布的权重，从而达到最好的结果。



#### 开放集迁移学习(OpenSet Domain Adaptation)

利用源域和目标域的关系， 给目标域的样本打上标签，并将源域转换到和目标域同一个空间中， 让学习标签和学习映射进行交替， 直到收敛或者目标值小于某一值即可．

### 其他算法

现有的域适配方法主要针对向量， 这种表示所带来的问题是， 当把这些数据应用于高维度表示( 如卷积) 时， 数据首先要经过向量化．此时， 无法精准完备地保留一些统计属性或者重要结构．

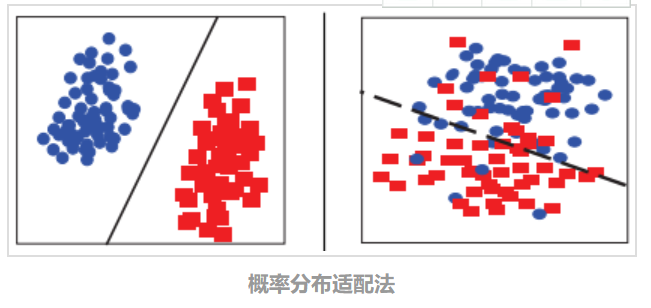
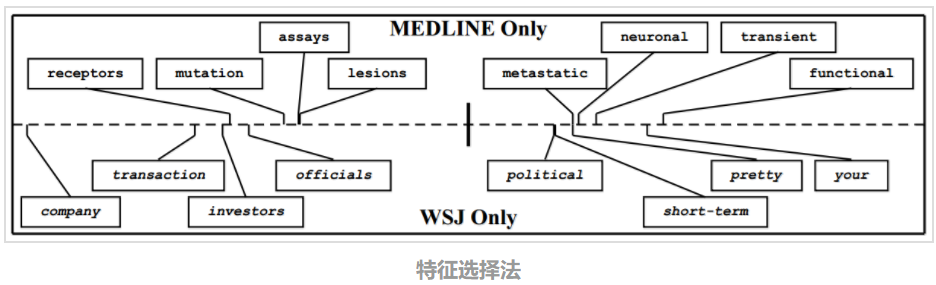
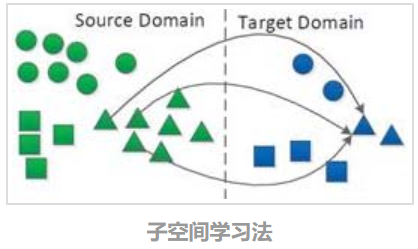
Lu 等人基于张量的 Tucker 分解，提出了一个称为 Naive Tensor Subspace Learning 的迁移学习算法．这个算法的出发点， 是假设源域和目标域共享了一部分子空间， 而这只在它们差异非常小时才有效．在更一般的条件下， 这种共享变量要通过一个线性变换来实现．

Fernando 等人提出了一个加强版的算法 ─Tensor － Aligned Invariant Subspace Learning( TAISL) ， 这个算法是对 ICCV－13 的那个子空间校准的扩展版

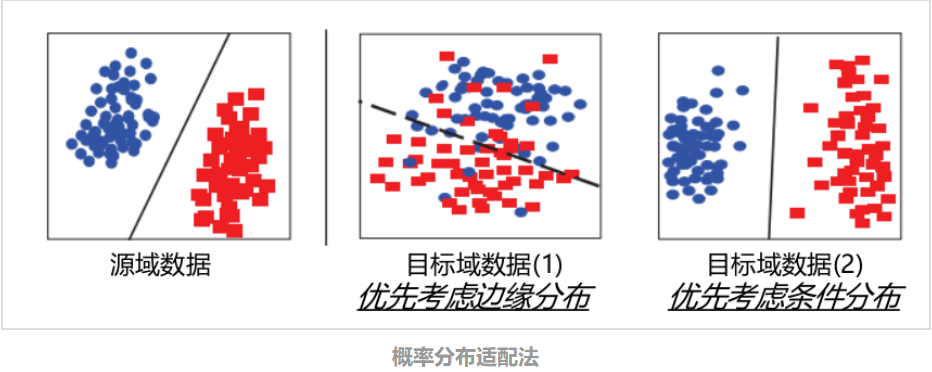
Feuz 和 Cook 等人提出了新颖的异构传输学习技术， 特征空间重映射( FSR) ， 它在具有不同特征空间的域之间传输知识， 构建元特征来将不同特征空间中的特征相关联．这些技术利用多个源数据集来构建进一步提高性能的集成学习器， FSR 应用于活动识别问题和文档分类问题．集合技术能够胜过所有其他基线， 甚至比在目标域中使用大量标记数据进行训练的分类器执行得更好．

### Domain Adaptation (领域自适应问题)

基本假设：

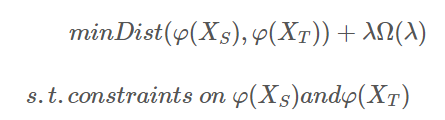
* 数据分布的角度：源域和目标域的**概率分布相似**（最小化概率分布距离）
* 
* 特征选择的角度：源域和目标域**共享某些特征**（选择出公共特征）
* 
* 特征变化的角度：源域和目标域**共享某些子空间**（把两个域变换到相同的子空间）
* 

#### 概率分布适配

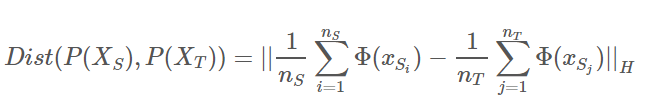


##### 边缘分布适配

TCA

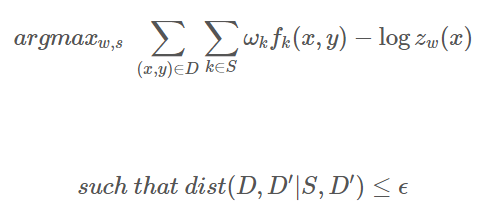
* 优化目标为
* 

MMD

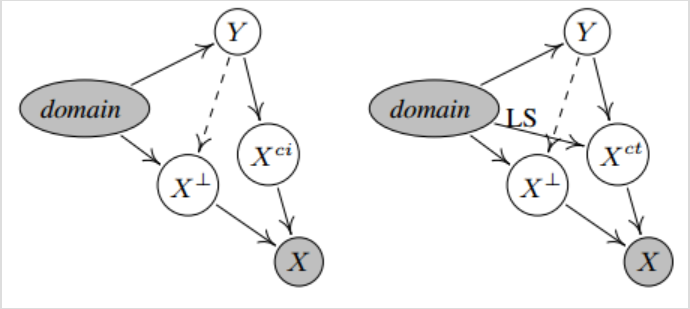
* 

##### 条件分布适配

优化目标为：

* 

定义条件转移成分,对其进行建模



##### 联合分布适配

直接继承于TCA，但是加入了条件分布适配，其优化目标为

* 

使用一个充分统计量，也就是类条件概率近似条件概率，也就是用一个弱分类器生成目标域的初始软标签

##### BDA

使用**平衡因子**来动态衡量两种分布的重要性，也就是

* 
  + 当μ→0，表示边缘分布更占优，应该优先适配
  + 当μ→1，表示条件分布更占优，应该优先适配

BDA方法可以有效衡量这两个分布的权重，从而达到最好的结果。

平衡因子的求解和估计来说，目前没有精确估计方法，我们一般使用**A距离**来进行估计

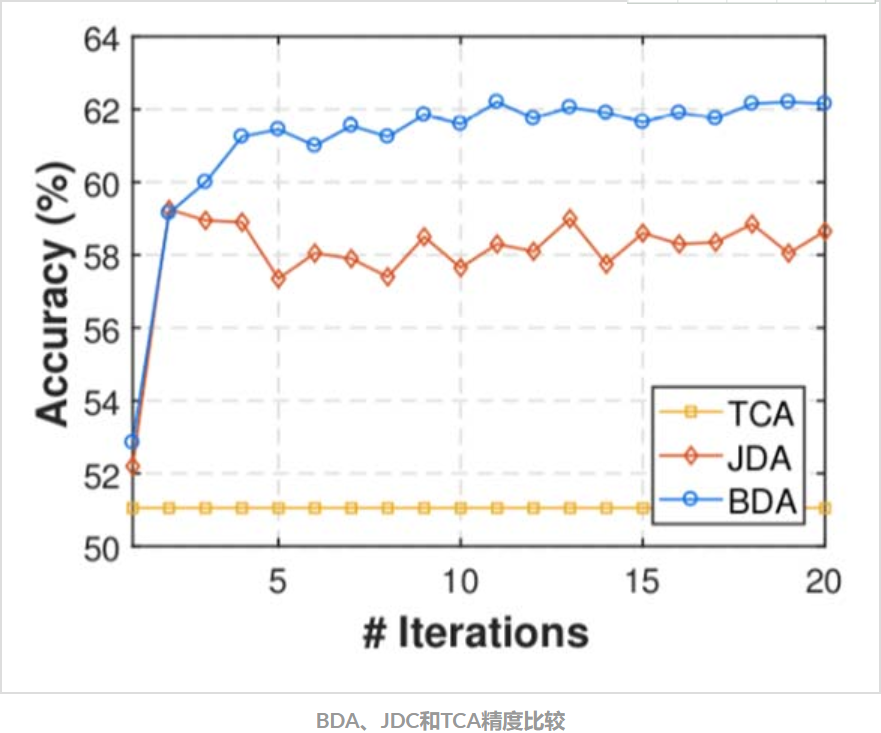
* 求解源域和目标域整体的A距离
* 对目标域聚类，计算源域和目标域每个类的A距离
* 计算上述两个距离的比值，也就是平衡因子

##### 总结

大部分方法都是基于MMD距离进行优化求解，然后可以分别进行边缘/条件/联合概率适配，在效果上，平衡 (BDA) > 联合 (JDA) > 边缘 (TCA) > 条件。

* 数据整体差异性大 (相似度较低),边缘分布更重要
* 数据整体差异性小 (协方差漂移),条件分布更重要

使用深度学习和分布适配往往有更好的效果。



#### 特征选择法

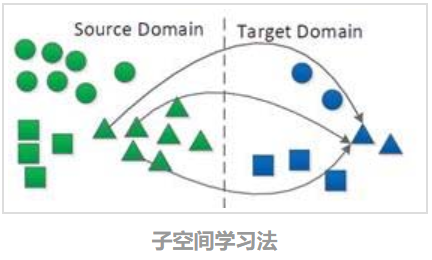
从源域和目标域中选择提取共享的特征,建立统一模型，寻找Pivot feature,将源域和目标域进行对齐。

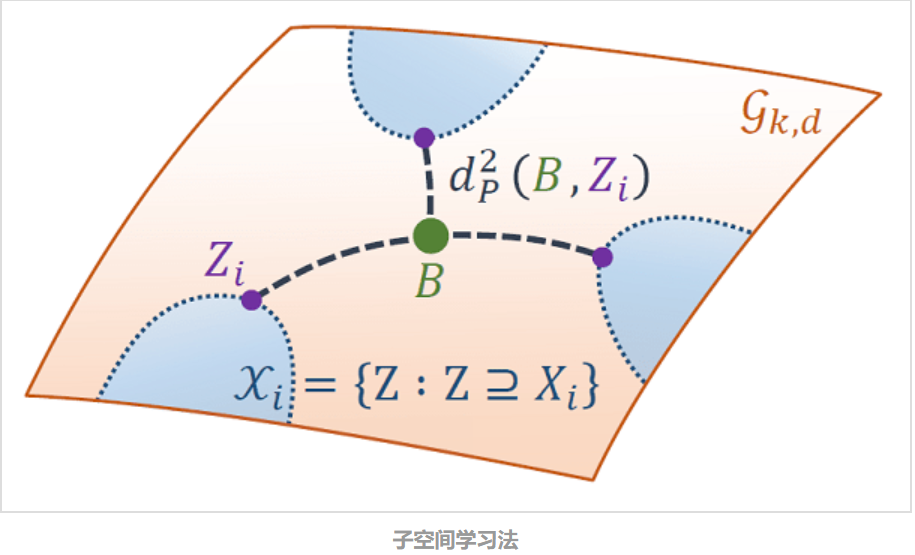
特点：

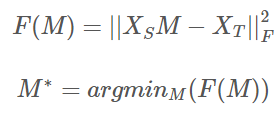
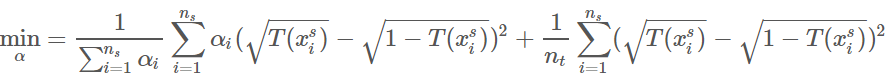
* 从源域和目标域中选择提取共享的特征,建立统一模型
* 通常与分布适配进行结合
* 选择特征通常利用稀疏矩阵

#### 子空间学习法

将源域和目标域变换到相同的子空间,然后建立统一的模型





* 统计特征变换 (Statistical Feature Transformation)
  + 将源域和目标域的一些统计特征进行变换对齐
  + 子空间对齐法 (Subspace Alignment, SA)
    - 直接寻求一个线性变换,把source变换到target空间中
    - 优化目标:
    - 
  + 关联对齐法 (CORrelation Alignment, CORAL)
    - 最小化源域和目标域的二阶统计特征，其优化目标为：
    - 
    - 形式简单，求解高效
  + 深度关联对齐 (Deep-CORAL)
    - 深度学习之中加入CORAL
    - 
* 流形学习 (Manifold Learning)
  + 在流形空间中进行子空间变换
  + 采样测地线留方法（Sample Geodesic Flow）
    - 把领域自适应的问题看成一个增量式“行走”问题
    - 从源域走到目标域就完成了一个自适应过程
    - 在流形空间中采样有限个点,构建一个测地线流
  + 测地线流式核方法 (Geodesic Flow Kernel, GFK)
    - 继承了SGF方法,采样无穷个点
    - 转化成Grassmann流形中的核学习,构建了GFK
    - 优化目标:
    - 
  + 域不变映射 (Domain-Invariant Projection, DIP) [Baktashmotlagh,CVPR-13]
    - 直接度量分布距离是不好的:原始空间特征扭曲
    - 仅作流形子空间学习:无法刻画分布距离
    - 解决方案:流形映射+分布度量
  + 统计流形法 (Statistical Manifold) [Baktashmotlagh, CVPR-14]
    - 在统计流形(黎曼流形)上进行分布度量
    - 用Fisher-Rao distance (Hellinger distance)进行度量
* 总结
  + 主要包括统计特征对齐和流形学习方法两大类
  + 和分布适配结合效果更好
  + 趋势:与神经网络结合

## 最新研究举例

### Google PlaNet

利用迁移学习去改进强化学习

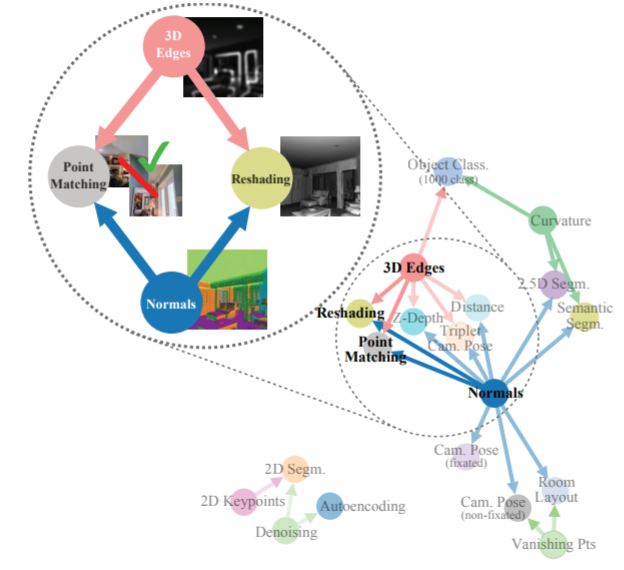
许多人依然认为强化学习是实现人工智能的最可行方法，但强化学习存在其局限性，其技术和研究主要集中在掌握个人任务上，我们使用迁移学习，有助于强化学习研究达到普遍性，对于多个任务，我们无需训练多个模型来实现任务的可靠性，这意味着数据高效和普遍强化学习新时代的到来。

### Taskonomy: Disentangling Task Transfer Learning

迁移学习在计算机视觉领域的使用。

论文通过做大量的实验，来揭示任务迁移学习中的一些现象。

Taskonomy是一个计算图，它定义了任务之间的可迁移性。节点表示任务，节点之间的边表示迁移性，边的权重表示从一个任务迁移到另一个任务的可能表现。

* 

Taskonomy一共由4 个步骤构成：

* 对不同任务进行建模
* 让它们两两之间进行迁移并获取迁移的表现
* 为了构建一个统一的字典，我们对这些迁移结果进行归一化
* 最后，构建可迁移图

针对目标建模：

* 监督学习，使用编码-解码机制，让神经网络针对特定的任务提取到强表征力的特征。

迁移学习建模，迁移的几种情况：

* 用于迁移的网络不能太复杂，不然特征不具有可迁移性，会使问题难以继续。使用一个浅层的卷积网络，用少量的数据去训练。
* 高阶的迁移，不同于简单地对一个任务的一种特征表达的迁移，而是多个不同任务共同进行迁移。
* 可传递式的迁移

任务相似性标准化：

* 构建一个迁移学习的相似度矩阵，从中我们可以很清楚地知道哪两个任务迁移效果最好
* 使用归一化方法，把所有的损失函数结果归一化到 [0,1] 之间
* 使用一种基于序列的方法，使得训练的表现和损失函数的值呈正相关。对于目标任务t，用矩阵Wt来表示可迁移到t的源域任务的表现。矩阵中的元素wij就表示：在同一个分出来的测试集上，源域si迁移到t，比sj迁移到t的表现好的百分比。（tournament matrix（锦标赛矩阵））

计算可迁移图

得到任务相似度矩阵后，我们需要在这个矩阵里找一条路，选择从多个源域到t的这条路上，迁移表现最好。在这个图中，任务是节点，迁移性和迁移效果是边。