

华中科技大学

# 本科生毕业论文

## 基于深度迁移学习的 轴承故障诊断方法研究

院 系	机械科学与工程学院
专业班级	工业工程 1701 班
姓 名	柯审强
学 号	U201710628
指导教师	高亮

2021 年 5 月 14 日

## 学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的  
研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集  
体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名：年 月 日

## 学位论文版权使用授权书

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留  
并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅  
和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编  
入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论  
文。

本学位论文属于 1、保密口，在 年解密后适用本授权书

2、不保密口。

（请在以上相应方框内打“√”）

作者签名：年 月 日

导师签名：年 月 日

## 摘 要

作为机械系统的组成部分之一,轴承长时间在复杂多变的环境中工作。轴承故障的发生会给工厂带来损失,同时也可能会对工人造成伤害。本文旨在通过深度迁移学习的技术,建立效率高、准确率高、鲁棒性好的轴承故障诊断模型。主要工作如下所示:

对于含有标签的目标域而言,本文选择在 ResNet18 网络模型的基础上,利用 Finetune 的方法建立轴承故障诊断模型,通过对卷积层和全连接层不同的学习率设置,利用 Adam 优化器进行网络模型的训练。该模型在凯斯西储大学公开轴承数据集表现良好,诊断准确率保持在 99.678%左右,且在迭代 5 代左右就能快速收敛,模型性能显著高于传统的深度神经网络和卷积神经网络。

对于不含标签,但数据分布与源域相似的目标域而言,本文选择采取基于 MDA 的迁移学习方法,以最大均值差异作为自适应度量准则,减小目标域和源域之间的边缘分布差异,并根据网络前向传播为不含标签的样本生成伪标签,并随迭代的进行而更新。最后所建立的诊断模型在在条件分布差异不大时,该迁移模型具有良好的迁移效果,且能够很快地达到收敛的效果,大大降低了模型的运算时间。但当条件分布差异较大时,该迁移模型虽然能够很快收敛,但很明显权重和偏置的更新陷入局部最优解,模型的诊断效果不佳。

对于不含标签,且数据分布与源域差异较大的目标域而言,本文选择采取 JDA 的迁移学习方法,不仅考虑两领域之间的边缘分布差异,同时还考虑两领域之间的条件分布差异,用 MMD 度量源域和目标域的条件分布的不匹配程度,建立诊断模型。模型在各种迁移任务均表现良好,且能快速收敛,同时伪标签的生成效果也十分优秀。

最后,对全文进行总结,并提出未来可能继续深入的研究方向。

**关键词:** 轴承故障诊断; 深度迁移学习; 微调; 边缘适配分布; 联合适配分布

## Abstract

As part of a mechanical system, bearings operate in complex and varied environments for long periods of time. Bearing failures can cause damage to the plant and can also cause harm to workers. This paper's purpose is to establish a high efficiency, high accuracy, and high robust bearing fault diagnosis model through the technology of deep transfer learning. The main work is as follows:

For the target domain with label, this paper chooses to build the bearing fault diagnosis model by Finetune on the basis of the ResNet18 network model and train the network model by using Adam optimizer through different learning rate settings of convolution layer and full-connection layer. The model performed well in the open bearing data set at Case Western Reserve University, the diagnostic accuracy was maintained at about 99.678 percent, and it was able to converge quickly in the iteration about 5 generations, and the model performance was significantly higher than that of the traditional deep neural network and convolutional neural network.

For the target domain that do not contain labels, but whose data distribution is like the source domain, this paper chooses to adopt an deep transfer learning approach as Marginal Distribution Adaptation, using the maximum mean difference as an adaptive measure, reducing the edge distribution difference between the target domain and the source domain, and generating pseudo-labels based on network forward propagation as unlabeled samples, updated as iteration progresses. Finally, the diagnostic model is established when the distribution of conditions is not different, the transfer model has a good transfer effect, and can quickly achieve the convergence effect, greatly reducing the operation time of the model. However, when the distribution of conditions is different, although the transfer model can converge quickly, it is obvious that the update of weight and bias is caught in the local optimal solution, and the diagnostic effect of the model is not good.

For the target domain without labels and with large differences in data distribution and source domain, this paper chooses to adopt a transfer learning method which name is

Joint Distribution Adaptation, which not only considers the difference of edge distribution between the two domains, but also considers the difference of conditional distribution between the two domains, measures the mismatch between the condition distribution of the source domain and the target domain with MMD, and establishes a diagnostic model. The model performs well in various transfer tasks and converges quickly, while the pseudo-labeling is also very good.

Finally, the paper summarizes the full text and puts forward the direction of further research which may be further studied in the future.

**Key Words:** Bearing Fault Diagnosis; Deep Transfer Learning; Finetune;  
Marginal Distribution Adaptation; Joint Distribution Adaptation

# 目 录

摘 要.....	I
Abstract .....	II
<b>1 绪论.....</b>	<b>1</b>
1.1 课题来源.....	1
1.2 课题背景及意义.....	1
1.2.1 课题背景.....	1
1.2.2 课题意义 .....	2
1.3 国内外研究趋势.....	3
1.3.1 轴承故障诊断问题描述 .....	3
1.3.2 轴承故障诊断研究趋势 .....	4
1.3.3 研究趋势小结 .....	7
1.4 本文的主要工作与结构 .....	8
1.4.1 主要工作 .....	8
1.4.2 全文结构 .....	8
<b>2 关键理论及技术.....</b>	<b>11</b>
2.1 信号处理.....	11
2.2 深度学习.....	12
2.2.1 深度学习介绍 .....	12
2.2.2 卷积神经网络 .....	13
2.3 迁移学习.....	17
2.3.1 迁移学习分类 .....	18
2.3.2 迁移学习方法 .....	19
2.4 本章小结.....	23
<b>3 基于 Finetune 的轴承故障诊断方法.....</b>	<b>24</b>
3.1 微调 Finetune.....	24
3.2 基于 ResNet 网络的微调 .....	26
3.3 轴承数据集介绍.....	27

3.3.1 数据切片 .....	28
3.3.2 数据处理 .....	30
3.4 实验验证 .....	33
3.4.1 诊断流程 .....	33
3.4.2 诊断结果 .....	36
3.5 本章小结 .....	39
<b>4 基于 MDA 和 JDA 的轴承故障诊断方法 .....</b>	<b>40</b>
4.1 基于 ResNet 网络的 MDA .....	40
4.1.1 最大均值差异 .....	41
4.1.2 真实标签和伪标签 .....	41
4.1.3 模型设计 .....	42
4.2 基于 ResNet 网络的 JDA .....	43
4.2.1 JDA 正则化项 .....	44
4.2.2 模型设计 .....	45
4.3 实验验证 .....	46
4.3.1 迁移任务 .....	46
4.3.2 诊断流程 .....	47
4.3.3 诊断结果 .....	49
4.4 本章小结 .....	54
<b>5 总结与展望 .....</b>	<b>55</b>
5.1 全文总结 .....	55
5.2 工作展望 .....	56
致谢 .....	57
参考文献 .....	59

# 1 绪论

## 1.1 课题来源

本课题来源于国基金重点项目(编号为 51435009), 课题名称为: 大数据驱动的智能车间运行分析与决策方法研究。

## 1.2 课题背景及意义

### 1.2.1 课题背景

自第一次工业革命以来, 机械设备在提高生产率、降低人力资源需求等方面做出了不可替代的贡献, 并且已经广泛应用在了如机械制造和冶金等各大领域中<sup>[1]</sup>。现代工业中, 机器设备正在朝着高精度、高效率以及更加自动化更加复杂的方向发展, 最为明显的变化就是机器设备的体积越来越大、结构越来越精密, 这也就导致了倘若机器设备一旦发生故障其后果也变得愈发严重。从某种意义上讲, 智能监控和故障诊断系统一直是提高工业设备安全性和可靠性的关键<sup>[2]</sup>。

作为机械系统的组成部分之一, 轴承长时间在复杂多变的环境中工作。轴承故障的出现不仅会影响机器正常运转, 耽误工厂的生产进度, 同时也可能会威胁到工人的生命安全。然而, 轴承故障作为最常见的机械故障, 几乎是不可避免的<sup>[3]</sup>(如表 1-1 所示)。

表 1-1 轴承故障占比

机械设备	轴承故障占比
旋转机械	10%
感应电机	40%
齿轮箱	20%

在轴承故障诊断的应用中, 目前共有三种常见的诊断方法<sup>[4]</sup>(如图 1-1 所示)。其中, 基于定性知识经验的方法要求较高水平的专家技术支撑, 适用于难以或者不能建立机理模型的情况下。基于分析模型的方法则是通过各种方法建立一个过程较为精确的定量数学模型。随着传感器技术的飞速发展, 工厂在机器设备上设置监测点, 从而通过传感器获取海量的数据。在这种情况下, 基于定性知识经验和分析



模型的方法使用成本极高，二者都需要对数据进行精确的分析估计才能建立较为准确的模型。在这种情况下，基于数据驱动的方法则表现出其优越的性能，其能利用这些海量数据，在不知道精确过程的前提下建立模型完成故障诊断<sup>[5]</sup>。

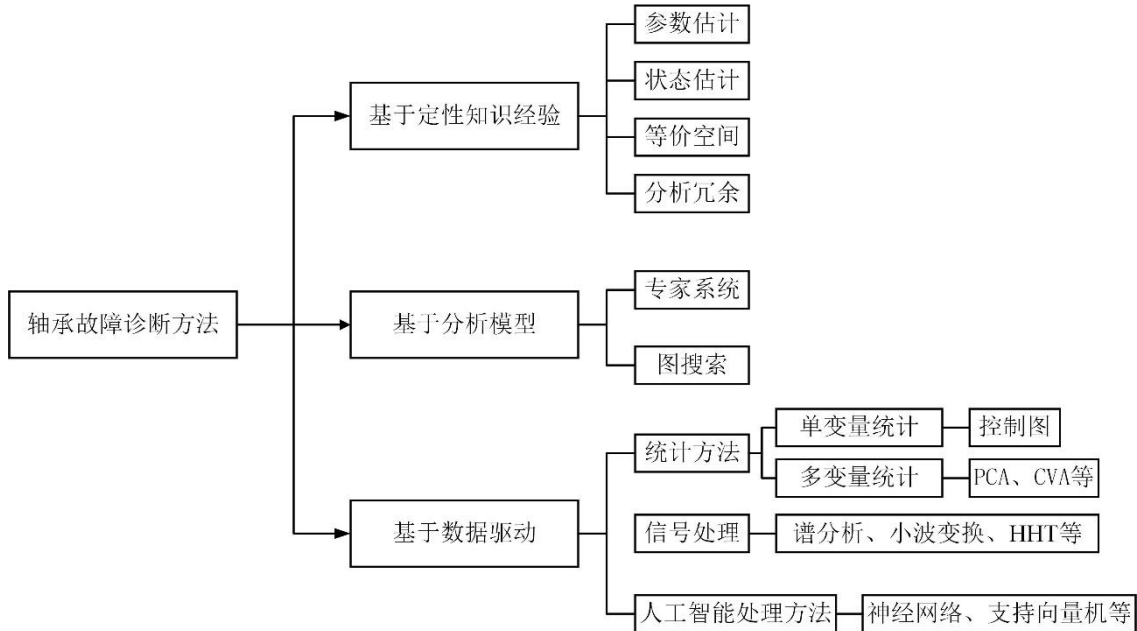


图 1-1 轴承故障诊断方法

然而，传统的数据驱动方法需要人工进行信号处理和特征提取，这不仅步骤繁琐，而且对技术人员专业水平要求较大，同时诊断的结果在很大的程度上依赖于特征提取的优劣<sup>[6]</sup>，难以保证较好的诊断性能。随着机器学习的飞速发展以及 GPU 技术的大力推进，深度学习已经被引入到轴承故障的诊断工作中。深度学习方法不需要人工地进行特征提取，同时还能保证较好的诊断结果。然而应用深度学习进行诊断有一个前提：训练数据集和测试数据集必须来自相同的特征空间，并处于独立同分布的状态下<sup>[7]</sup>。在实际工业应用中，由于不同的载荷条件、工作环境以及轴承的故障程度，实际情况下训练数据集和测试数据集之间的分布是不同的，这就导致在标准数据集上训练好的模型难以直接应用在实际生产中（这种现象被称为域偏移现象<sup>[8]</sup>）。迁移学习便能很好地解决这一问题，它是指在不同的领域之间进行知识的转移过程，利用现有知识来解决不同但相关领域中的问题，从而减轻对数据特性的要求<sup>[9]</sup>。

## 1.2.2 课题意义

在实际应用中，目前存在的最大问题便是，通常消耗大量资源在实验室数据集或某一特定数据集上建立并训练的轴承故障诊断模型在不同的其它数据集上的诊

断效果难以达到要求。因此,从工程意义方面来看,本文基于深度迁移学习的研究能大大提高模型的鲁棒性,使得模型能够在不同的任务不同的数据集完成迁移,提高诊断效率,能较好地达到实际应用的目标。

本文的研究意义在于将深度学习和迁移学习进行有机结合,所提出的模型既不受人工提取的特征所干扰,表现出较强的诊断能力,同时该模型也具备迁移学习的性质,能够有效避免域偏移现象造成的巨大影响。

因此,本文工作具有较高的工程意义和科研意义,值得研究。

### 1.3 国内外研究趋势

为了延长机器的使用寿命并提高正常运行时间,工厂需要进行状态监测来监视机器的状态并发现故障成分。旋转机器中最容易发生故障的部分大多都是滚动轴承,如风力涡轮机等<sup>[10]</sup>。大量国内外研究学者针对轴承故障的诊断工作展开了深入研究。

作为一项监测机器组成部分的状态的重要指标,振动信号不仅会在正常的工作条件下产生,同时它存在于轴承的各种故障状态(如滚动球体故障等)中。因此,在过往的研究当中,大部分研究学者都是通过轴承的振动信号来判别其工作状态。本文接下来将从轴承故障诊断问题的描述以及研究两大部分展开分析。

#### 1.3.1 轴承故障诊断问题描述

轴承故障诊断是通过传感器收集轴承在工作时的振动信号,运用已经训练好的模型对其进行特征提取与模式识别,判断其工作状态,从而起到对故障的轴承进行快速响应的目的<sup>[11]</sup>。

在轴承故障的诊断工作中,核心工作在于数据收集、信号特征提取和模式识别这三个步骤<sup>[12]</sup>。随着计算机技术和传感器技术的飞速发展,工厂在大型设备上设置监测点,通过各类传感器获取设备信息。目前,已经应用在轴承信号收集的传感器有:加速度计、红外温度计、声发射传感器等。这些设备可以监测机器运行状态并获取海量相关数据,如振动信号等<sup>[13]</sup>,研究人员通过对所收集的数据进行处理与分析以进行故障识别。然而在工业设备精度和可靠性不断提高的时代,轴承出现故障的概率在逐渐变小,这就导致收集到的数据的类别标签不平衡<sup>[14]</sup>。同时,每条数据来源于不同的机械设备、不同的工况条件,传感器的物理位置也有所不同,这

就导致数据的共同特征难以挖掘,从而增大轴承故障诊断的难度。在这一情况下,从广泛且不平衡的数据中提取特征变得尤为重要。特征提取是指利用信号处理方法对大量数据进行分析,提取出有关轴承状态的主要特征<sup>[15]</sup>。常用的特征提取方法有快速傅里叶变换 FFT<sup>[16]</sup>、经验模式分解 EMD<sup>[17]</sup>、时频信号处理<sup>[18]</sup>、短时傅里叶变换 STFT 以及信号的统计学特征<sup>[19]</sup>。在完成信号特征提取,得到低维的特征向量后,便可以设计如支持向量机<sup>[20]</sup>、感知机<sup>[21]</sup>等分类器进行模式识别,完成轴承故障的诊断任务。

### 1.3.2 轴承故障诊断研究趋势

本文接下来将从早期学习、机器学习、深度学习和迁移学习四方面展开对轴承故障诊断问题研究近况和发展趋势的综述。

#### I) 早期轴承故障诊断研究

在上个世纪,研究学者都是通过对振动信号的频谱分析来判断故障的存在与否。Dyer and Stewart(1978)<sup>[22]</sup>最早提出了使用峰度作为轴承故障监测的评价指标。他指出,对于正常工作的轴承来说,其振动信号的峰度约为 3,而在发生故障时,该峰度会明显上升,并随着故障的恶化峰度会降低。Tandon 和 Nakra(1993)<sup>[23]</sup>通过对从原始的振动信号计算相关时间波形指标(如峰值、平均值、均方根值)来进行轴承状态的分析,取得了一定的成绩。McInerny(2003)<sup>[24]</sup>则对振动信号采用快速傅里叶变换(FFT),并结合海明窗以及 50%重叠处理技术进行频谱分析,发现在频率为 40Hz 的条件下故障轴承的频率及谐波可以明显被区分开来,从而为轴承故障诊断作出贡献。Yiakopoulos(2005)<sup>[25]</sup>在轴承的滚动元件滑动时的振动信号中引入背景噪声。当增大背景噪声时,具有内圈缺陷的轴承的特征缺陷频率明显降低。为了克服快速傅里叶变换 FFT 的局限性,Wadhwani 和 Gupta(2005)<sup>[26]</sup>选择采用短时傅里叶变换和小波变换将给定的不平稳信号分解为多个平稳信号,进而再对信号进行模式识别,完成故障诊断。

#### II) 基于机器学习的轴承故障诊断研究

在这一阶段,传统的机器学习方法和模型被用于轴承故障的诊断研究中。其常用的诊断流程如下:首先,通过被监测机器上的速度传感器或加速器计等传感器获取如振动信号等原始数据,然后针对原始数据进行预处理以及人工特征的提取,最后基于这些人工提取的特征,设计分类器进行模式识别,完成故障诊断任务。

Geramifard(2010)<sup>[27]</sup>使用遗传算法从 16 维特征向量空间中进行数据降维,并选择出 3 个代表性特征,然后分别结合 K-NN、SOM、MLP 和 BNC 进行分类和预测。Miao 和 Pecht(2010)<sup>[28]</sup>采用经验模态分解(EMD)来进行可适应信号分析,首先通过希尔伯特变换进行特征提取,根据贝叶斯规则采用隐马尔科夫模型(HMM)针对现有的数据进行状态概率评估,实现了在线健康评估,并可将其用于早期故障监测,对轴承检测起预警作用。Ghapanchi 和 Esmalifalak(2014)<sup>[29]</sup>提出了一种基于机器学习的数据挖掘方法——异常检测,该方法选择峰度和非高斯性分数(Non-Gaussianity Score, NGS)作为特征,采用分类算法进行故障样本的区分。该方法能够比 SVM 更早探测出轴承故障状态,并具有较高的准确率。Samanta(2001)<sup>[19]</sup>以轴承的时域振动信号的特征(如均方根、方差等)为输入,构建人工神经网络(ANN),运用前向传播和反向传播对网络参数(各层权重和偏置)进行训练,结果显示了人工神经网络在轴承故障诊断应用的有效性。Kankar 和 Sharma(2011)<sup>[30]</sup>对比了支持向量机(SVM)和人工神经网络(ANN)这两种机器学习方法在轴承故障诊断中的应用效果。作者首先通过信号的统计学方法针对轴承振动的时域振动信号提取出等特征,然后分别运用 SVM 和 ANN 方法进行故障分类,结果发现在有限的数据下 SVM 的分类准确率要明显高于 ANN。Kateis 和 Moshou(2013)<sup>[31]</sup>基于贝叶斯框架建立了自动关联确定的多重感知机(MLP-ARD),结合峰度和线性积分两个信号指标,并综合考虑水平方向和竖直方向的信号特征,对于不同故障类别的分类准确率高达 99%。

### III) 基于深度学习的轴承故障诊断研究

这一阶段,研究学者不再需要进行人工的特征提取以作为智能诊断算法的输入。通过输入原始数据或预处理后的数据,深度学习经过多层神经网络自动进行特征提取,避免了人工提取特征的局限性以及高要求,从而保证了诊断结果不受人为因素的干扰。

Khan 和 Yairi(2013)<sup>[32]</sup>提出了一种基于多传感器健康诊断的深度置信网络(DBN),用来进行状态分类。该网络具有多层受限玻尔兹曼机和一层反向传递层。不同于深度玻尔兹曼机,该网络的中间层和相邻层的连接是单向的,该网络的分类准确率达到 90.99%。Lee 和 Dean(2016)<sup>[33]</sup>针对原始的轴承振动信号,以双通道的数据作为输入建立卷积神经网络模型,自动提取特征并进行训练,大大减少了模型

训练对于庞大数据的依赖。Ahmed 和 Nandi(2018)<sup>[34]</sup>基于压缩感知理论,提出了一种针对高度压缩测量的轴承故障的分类方法,采用无监督特征学习算法稀疏自编码器(SAE)来进行特征表示的学习,大大提高了算法的精度,并缩短了计算时间。为了提高自编码器的诊断性能,Lu 和 Wang(2016)<sup>[35]</sup>选择堆叠降噪自编码器(SDA)来进行轴承故障的分类工作,该模型在各种工作条件 and 环境噪声的影响中都具有较高的鲁棒性能。

#### IV) 基于迁移学习的轴承故障诊断研究

在实际生产中,工厂搜集的轴承信号数据与标准轴承信号数据库的分布具有差异性,这就导致基于标准数据库建立并训练好的分类模型在实际生产中的应用效果不佳。为了增加训练模型的鲁棒性,研究学者将目光投入到了迁移学习领域中。迁移学习能够很好地解决源域和目标域之间数据分布差异较大的问题,也就是能够将模型在不同的训练集上均表现良好。

迁移学习的主要方法有转移成分分析(TCA)<sup>[36]</sup>、联合分布适应(JDA)<sup>[37]</sup>、深度适应网络(DANs)<sup>[38]</sup>以及对抗域适应(ADA)<sup>[39]</sup>等方法。

Long 和 Cao(2015)<sup>[38]</sup>提出了一种新的深度适应网络 DAN 架构,该体系结构将深度卷积神经网络推广到领域自适应场景。在 DANs 中,所有特定于任务的层的隐藏表现形式都内嵌入再生核希尔伯特空间 RKHS 中。在该空间中,可以实现精确匹配不同域分布的嵌入。作者使用用于均值嵌入匹配的最佳多内核选择方法,进一步减少域差异。Shen 和 Chen(2016)<sup>[40]</sup>将迁移学习应用在轴承故障诊断中的特征提取上,从而提高不同工作情况下故障诊断的性能。其主要工作是调整了被选择的辅助数据和辅助目标域数据在分类时的权重,避免了负迁移的影响。Wen 和 Gao(2017)<sup>[41]</sup>通过将深度神经网络引入到迁移学习中,提出了一种深度迁移学习模型。其采用一个三层的稀疏自编码器来提取数据的原始特征。为了降低误差损失,作者采用了目标域和源域之间数据的最大均值误差来衡量。该模型在凯斯西储大学的公开数据集上具有很好的特征提取能力,并表现出很好的诊断性能。Cao 和 Zhang(2017)<sup>[42]</sup>基于卷积神经网络提出了一种迁移学习模型,采用预训练深度神经网络自动提取输入的实验数据的特征,并大大减少了训练集的数据量。Hasan 和 Kim(2018)<sup>[43]</sup>首先采用离散正交斯托克韦尔变换(Discrete Orthonormal Stockwell Transform, DOST)将一维轴承信号数据转换成二维图像数据,并提出了一种深度迁



移卷积神经网络模型。相较于传统的卷积神经网络,该模型不仅在准确率方面有显著的提升,同时还显著降低了运行时间,加快了收敛效率。Wang 和 Youren(2018)<sup>[44]</sup>将生成对抗网络 GAN 与堆叠式稀疏自编码器 SDAE 相结合用于行星齿轮箱的故障诊断工作中。该模型利用 GAN 的生成器生成与标签样本分布相似的新样本,并采用 SDAE 进行自动特征提取。实验结果证明该 GAN-SDAE 模型相较于其他传统模型或 SDAE 模型均表现出更好的诊断效果。此外, Li 和 Zhang(2019)<sup>[45]</sup>在深度生成对抗网络的基础上,结合域自适应的方法,提出了一种基于深度迁移学习的跨域诊断模型,来实现滚动体轴承的故障诊断。该模型的优势在于不需要较多的样本标签,通过人工生成用于域自适应的伪样本。当机器故障条件下的测试数据不可用于训练时,该方法能够提供可靠的跨域诊断结果。Guo 和 Wu(2019)<sup>[36]</sup>为了减少模型训练对于含有标签数据的依赖,选择采用迁移支持向量机 TSVM 的方法来完成对 Delta 3D 打印机的故障诊断,提出转移成分分析 TCA,以通过从源域中标记较少的数据和目标域中大量未标记的数据进行跨域特征提取来捕获用于表示源域和目标域的共享特征。Han 和 Liu(2020)<sup>[37]</sup>基于深度迁移学习模型领域适应方案,提出了一种新型的智能故障诊断框架,通过将边际分布适应扩展为联合分布适应,利用与源域中的标记数据相关的判别结构来适应未标记的目标数据的条件分布,从而保证更为精确的分布匹配。

### 1.3.3 研究趋势小结

随着计算机技术和 GPU 技术的飞速发展,越来越多的研究学者将目光投向了机器学习和迁移学习领域,致力于将这些领域中的模型与方法应用在轴承故障诊断领域上。如下图 1-2 所示,蓝色表现浅层学习方法的应用,橙色表示深度学习方法的应用,而绿色代表迁移学习的应用。

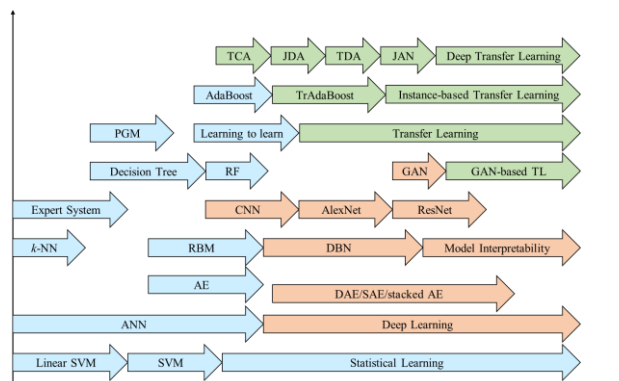


图 1-2 轴承故障诊断中的研究进展

然而, 尽管深度学习模型及其相关衍生模型已经在轴承故障诊断领域中取得了较好的诊断结果, 但仍然存在一些未攻克的矛盾。对于复杂的深度学习模型来说, 超参数优化这部分具有较大的工作量, 可能需要花费较多的计算时间以及运行成本, 这便给实时监控带来了一定的困难。在实际生产中, 获取足够的系统状态样本是一件极其困难的事, 并且实际数据和实验室数据可能不满足同一分布的条件, 这就导致了基于实验室数据建立训练的模型很难用于实际工厂场景中。

高效率的诊断过程和高准确率诊断结果、模型训练所需的大量数据和实际生产收集的较少数据、基于实验室数据建立的诊断模型和实际生产中诊断模型的需求, 这三项矛盾都直接影响到了诊断模型在实际场景中的推广普及。

因此, 本文回顾了深度迁移学习的研究进展, 并基于深度迁移学习讨论了三种不同情况下的轴承故障诊断模型, 旨在在不同的样本条件下均能实现高效率、高准确率、高鲁棒性的轴承故障诊断。

## 1.4 本文的主要工作与结构

### 1.4.1 主要工作

本文在对信号处理理论、深度学习理论以及迁移学习理论等理论方法进行学习和研究的基础上, 以轴承故障诊断的效率与准确率作为研究目标, 结合深度学习和迁移学习相关模型, 以传统的卷积神经网络 CNN 作为出发点, 研究轴承故障诊断的高效算法。主要工作总结如下:

- (1) 以 ResNet18 网络作为基本模型, 结合迁移学习展开研究;
- (2) 研究了目标域数据含有标签情况下的有监督故障诊断问题;
- (3) 研究了目标域数据不含标签情况下的无监督故障诊断问题;
- (4) 基于深度迁移学习, 建立了三种不同的轴承故障诊断模型;
- (5) 将本文所提出的模型与传统的深度学习模型等进行了对比。

### 1.4.2 全文结构

全文的主要组织结构内容如下:

第一章: 绪论。主要介绍了本课题的来源、目的以及研究意义, 从早期学习、机器学习、深度学习和迁移学习四方面整理总结了轴承故障诊断问题的研究趋势和进展, 并且系统地介绍了本文的主要研究内容。

第二章：关键理论及技术。在这一部分，本文详细介绍了本课题所涉及到的三大方面的理论知识——信号处理、深度学习和迁移学习，并介绍了如传统卷积神经网络的网络结构和参数学习过程。

第三章：基于 Finetune 的轴承故障诊断方法。在这一章节中，本文以传统的深度残差网络 ResNet18 网络作为预训练模型，采用微调的深度迁移学习方法，建立并训练故障诊断模型。并以凯斯西储大学轴承公开数据集作为研究对象，根据数据切片和短时傅里叶变换等知识对振动信号数据集进行预处理，得到参与模型训练的训练集和测试集，通过 TensorFlow2.0 环境进行求解，并对诊断结果进行分析对比。

第四章：基于 MDA 和 JDA 的轴承故障诊断方法。在这一章节中，本文继续研究深度迁移学习方法在故障诊断方法的应用。通过引入自适应度量准则来完成对源域和目标域之间距离减小的操作，比较不同方法和不同正则化项对模型求解效果的影响，并对诊断结果进行分析。

第五章：总结与展望。在这一部分，本文对所构建的三种轴承故障诊断模型进行总结，得出结论，并根据模型表现以及分析结果，展望未来研究方向。

本文的整体框架如图 1-3 所示：



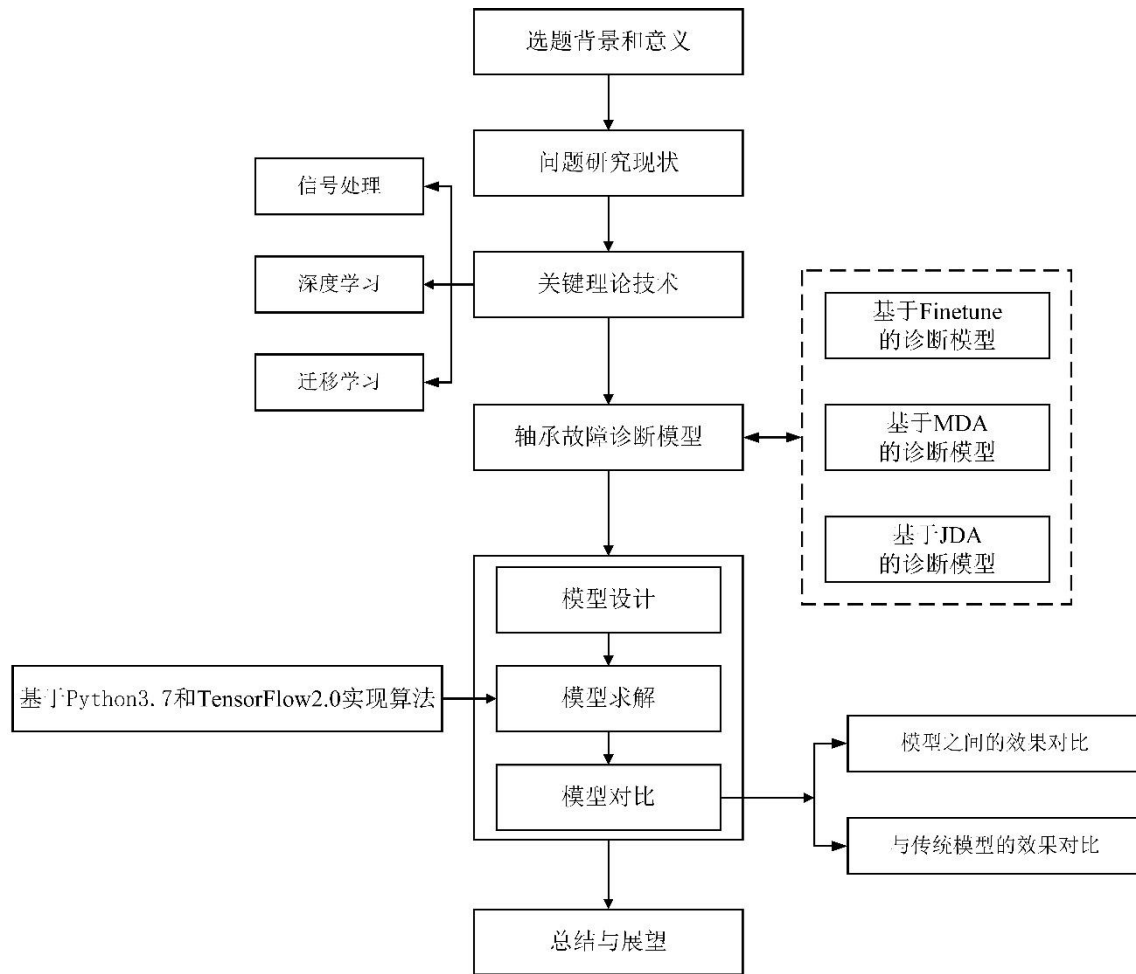


图 1-3 全文整体框架

## 2 关键理论及技术

本文所采用的关键理论共涉及三大方面：信号处理、深度学习以及迁移学习。这三大方面理论同时包含了若干项主要技术。接下来本文将在这一章节展开详细介绍。

### 2.1 信号处理

本文拟采用的轴承信号数据集为凯斯西储大学(CWRU)的轴承公开数据集，该数据集是通过加速度计传感器收集实验设备台在特定环境下的振动信号（具体介绍见下文第3章）。根据 Jayaswal 等人<sup>[46]</sup>的工作，我们可以知道：在收集信号的过程中会有很多因素，如：轴承组成成分（内圈轴承、外圈轴承、球轴承等）、监测环境以及监测状态等，会对轴承信号进行干预。由于原始数据信号包含许多噪声和干扰，这些因素会影响信号成分从而影响信号的分析过程并可能造成错误的分析结果。因此，对原始信号进行信号处理是重要且必须的。

目前，在信号处理领域中，共有三类处理方法：时域信号处理、频域信号处理以及时频域信号处理<sup>[47]</sup>（如图2-1所示）。

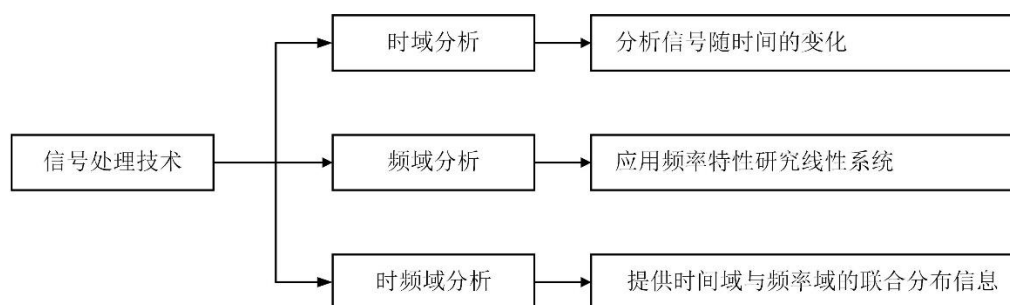


图 2-1 信号处理方法分类

时域信号处理主要是针对具有振幅-时间函数的信号轴的展示。时域信号处理的结果是绘制一张时域图，其展示了随着时间变化信号所发生的变化。时域分析不能进行很好地降噪和去干扰，其分析结果往往包含较多的信息，而无法对关键信息进行判别。频域信号处理主要是针对具有振幅-频率函数的信号轴的展示。频域信号处理的结果是绘制一张频域图，其展示了在一定频率范围内每个给定频段内的信号数量。频域分析的优势在于，振动信号中重复出现的信号会被清晰地在线谱图的峰值中展现，这就意味着偶然发生的噪声以及干扰不会导致整个信号频谱图发

生重大变化。时频域信号处理则是使用各种表示方法同时对时域信号和频域信号进行研究的的技术。时频域信号处理能够较好地分析非线性和非平稳信号,这是时域分析和频域分析所不具备的。

作为最为经典的时频域分析方法之一,短时傅里叶变换(Short Time Fourier Transform, STFT)<sup>[48]</sup>采用滑动窗口的形式来切分完整信号,进而生成时频图。简而言之,短时傅里叶变换通过滑动窗口将完整的信号切分成若干个信号,并假设这若干个信号在很短的时间内是平稳的。同时,为了保证被切分信号的平稳性,被分割信号的宽度需要与滑动窗口的宽度一致。接着,对每个分割后的信号采用傅里叶变换。最后,将完整信号切割后的所有信号傅里叶变换的结果进行拼接,即得到最后的时频分析结果。与传统的快速傅里叶变换相比较,该方法可以有效地处理非平稳信号。

## 2.2 深度学习

### 2.2.1 深度学习介绍

作为机器学习领域的子分支,深度学习涉及受大脑结构和功能启发的算法。起工作模式可以简单定义为:模仿人脑在处理数据和创建用于决策的模式方面的工作,具有能够从非结构化或者说是未标记的数据中不受监督地学习的网络。

2006年,Geoffrey Hinton 首次提出深度学习的概念<sup>[49]</sup>。常见的深度学习模型有:卷积神经网络 CNN、循环神经网络 RNN 以及生成对抗网络 GAN 等。一般来讲,神经网络与深度学习的差异并不大,广义上的深度学习指的便是基于深层神经网络(网络隐藏层数一般大于3)实现的模型或者算法。深度学习的发展时间线如下图 2-2 所示:

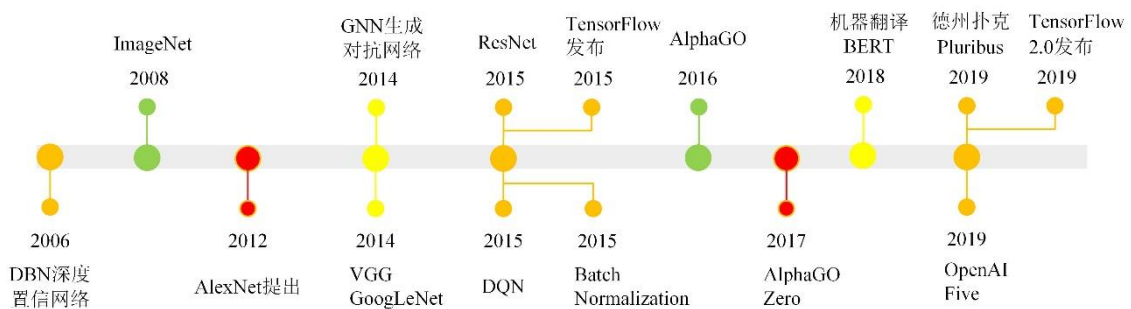


图 2-2 深度学习发展时间线

相较于传统的机器学习算法（如：决策树、支持向量机等），神经网络能够在不需要先验知识的前提下自动地进行特征提取，这在很大程度上降低了研究人员的工作要求，并避免了建模过程中由于人工参与而导致的主观性。然而，浅层神经网络（网络层数小于 3）的特征提取能力一般不能达到理想要求，其难以发觉数据的深层特性，从而导致应用效果在一些领域中并不良好。深度学习中所采用的是深层神经网络，其通过多个隐藏层的叠加对数据的特征进行深度提取，实现端对端式的特征提取。

深度学习算法与基于规则的系统算法、传统机器学习算法以及浅层神经网络算法之间比较如下图 2-3 所示。

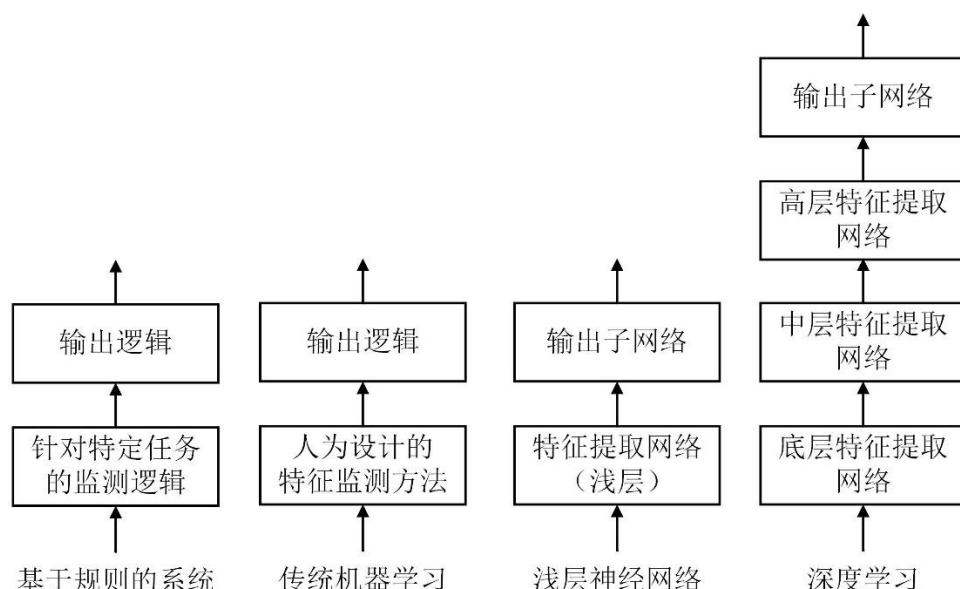


图 2-3 深度学习与其他算法的比较

### 2.2.2 卷积神经网络

卷积神经网络作为基础的深度学习网络模型之一，在各个领域中都有着至关重要的作用。相较于传统的深度神经网络 DNN，卷积神经网络利用局部相关性和权值共享这两大思想，将网络的参数量级大大缩小，减小了网络训练的运算量和运算存储空间。

如下图 2-4 所示，典型的卷积神经网络 CNN 主要由输入层、卷积层、下采样层（池化层）、全连接层和输出层组成。以 2D 图像数据为例，卷积层接受高、宽分别为  $h$  和  $w$ ，通道数为  $c_{in}$  的输入特征图  $X$ ，在  $c_{out}$  个高宽均为  $k$ ，通道数为  $c_{in}$  的卷积核的作用下，生成高、宽分别为  $h'$ 、 $w'$ ，通道数为  $c_{out}$  的特征图输出。

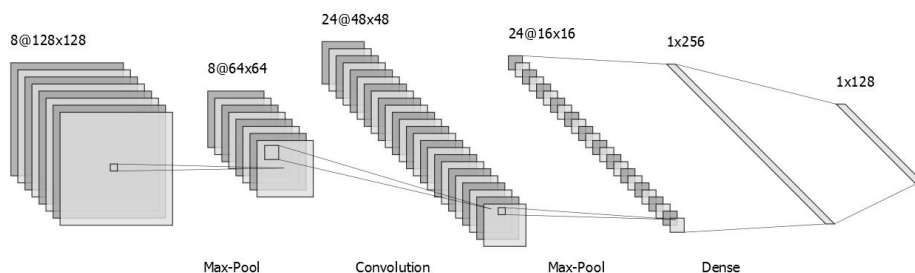


图 2-4 卷积神经网络示意图

### 2.2.2.1 网络结构

本文采用  $H_i$  表示卷积神经网络第  $i$  层的特征图，且有  $H_0 = \mathbf{X}$ 。其中， $\mathbf{X}$  表示网络的输入图像。

当卷积神经网络第  $i$  层为卷积层时，卷积层利用权值共享的思想，通过卷积核对特征图以固定的步长进行遍历运算。



图 2-5 权值共享矩阵示意图

以上图 2-5 为例，权值共享的思想便是，无论对任何输出节点，均使用相同的权值矩阵  $\mathbf{W}$ 。当计算左上位置的输出像素时，采用权值矩阵：

$$\mathbf{W} = \begin{bmatrix} w_{00} & w_{01} & w_{02} \\ w_{10} & w_{11} & w_{12} \\ w_{20} & w_{21} & w_{22} \end{bmatrix} \quad (2-1)$$

与感受野内部对应位置的像素进行卷积运算操作。当平移感受野位置时，每次都共享权值参数  $\mathbf{W}$  进行卷积运算操作，得到该感受野对应的输出值。这样，每一层共用相同的权值矩阵，也称为卷积核，大大缩小了网络参数量。

然而在实际操作过程中，倘若使用卷积操作，神经网络在反向传播训练过程中需要翻转卷积核，运算量较大且难以操作，因此本文采用相关运算来进行卷积操作

运算<sup>[51]</sup>，具体运算如下：

$$H_i = H_{i-1} \otimes W_i = \sum_m \sum_n H_{i-1}(m, n) \cdot W_i(m - k, n - k) \quad (2-2)$$

其中， $H_{i-1}$ 为卷积神经网络第 $i - 1$ 层的输出特征图， $W_i$ 为卷积神经网络第 $i$ 层的卷积核，其高和宽均为 $k$ 。每进行一次卷积层的操作运算，都相当于一次特征提取，从而得到一个特征映射。

然而，经过卷积层之后的特征图是线性不可分的，因此需要激活层对多维特征进行非线性变换，从而增强特征的线性可分性。

在神经网络中，常用的激活函数共有三种：Sigmoid 函数、Tanh 函数以及 ReLU 函数。三种激活函数的曲线图如下图 2-6 所示：

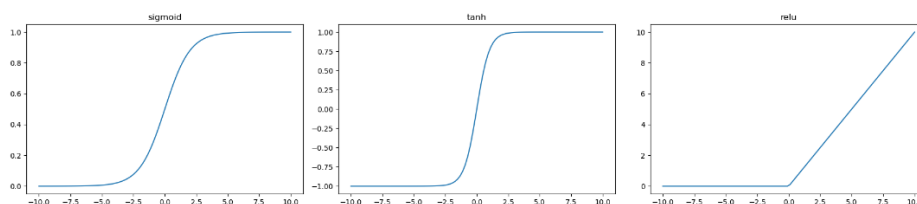


图 2-6 常用的激活函数

### (1) Sigmoid 函数

Sigmoid 函数的表达式如下所示：

$$f(x) = \frac{1}{1+e^{-x}} \quad (2-3)$$

Sigmoid 函数的输出范围是 $[0,1]$ ，因此可以将该函数运算看作为归一化。使用 Sigmoid 函数可以有效避免梯度爆炸，但不能避免梯度消失。当输入值的绝对值较大时，其梯度为 0。

### (2) Tanh 函数

Tanh 函数的表达式如下所示：

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2-4)$$

Tanh 函数的输出范围是 $[-1,1]$ ，曲线与 Sigmoid 函数相似。在该激活函数中，负输入将被强映射为负值，正输入被强映射为正值，整个函数以 0 为中心，权重更新效率相较于 Sigmoid 函数较高。

### (3) ReLU 函数

ReLU 函数的表达式如下所示：

$$f(x) = \max(0, x) \quad (2-5)$$



相较于 Sigmoid 函数和 Tanh 函数相比, ReLU 函数不存在梯度弥散和梯度爆炸的现象。这是由于:当输入的绝对值较大时, Sigmoid 函数和 Tanh 函数的导数值均接近于 0,这会严重影响到反向传播的训练过程,使得底层网络权重训练难以进行,而 ReLU 函数在输入大于 0 时始终保持着导数值为 1,不存在梯度饱和的问题,从而避免了梯度弥散和梯度爆炸现象。

经过卷积层和激活层后,特征的维度会增加,从而可能会造成过拟合现象的发生,因此需要在卷积层和激活层之后添加一个下采样层(也称池化层)。池化层的作用是将高维的特征通过池化的方式进行降维操作,从而减少网络模型训练过程中的参数量。

常见的池化函数有两种:均值池化函数和最大值池化函数。最大值池化函数是将局部感受野内中最大的元素值作为返回值输出,而均值池化函数则是将局部感受野内所有元素的均值作为返回值输出。

值得注意的是,在卷积神经网络中,池化层并没有参数参与网络的训练过程中,其仅是对输入进行固定的功能操作。

卷积层、激活层和池化层结合构成一次完整的特征提取,经过多个卷积层、激活层和池化层结合的交替传递,卷积神经网络依靠全连接网络对针对提取的特征进行分类,最后得到基于输入的分类概率分布:

$$Y(i) = P(L = l_i | H_0; (W, b)) \quad (2-6)$$

其中,  $l_i$  表示该输入对应的预测结果为第  $i$  类的标签类别。

#### 2.2.2.2 参数学习

卷积神经网络的训练过程共分为两步:前向传播以及反向传播。

在前向传播的过程中,输入依次从输入层开始,向前传播并进行相应的运算直至输出层,最终输出该输入值对应的输出值。

在反向传播的过程中,根据输出值与实际值之间的差异,计算梯度,并由输出层依次反向逐层进行传播,计算所有层的梯度,采用梯度下降法等优化算法来进行网络可训练参数的更新。

对于前向传播,输入依次经过各层,并完成相应运算,最后通过全连接层的运算得到概率分布。倘若全连接层上一层的输出并非是一维特征向量,则全连接层的输入并非上一层的输出,而是上一层输出铺展得到的一维特征向量。全连接层的前

向传播表达式为:

$$H_i = W_i \cdot H_{i-1} + b_i \quad (2-7)$$

其中,  $W_i$ 和 $b_i$ 分别表示卷积神经网络第 $i$ 层的权重和偏置。最后一层全连接层采用的激活函数为 Softmax 函数。Softmax 函数可以将输入映射到和为 1 的概率分布, 满足分类问题的需求。

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{d_{out}} e^{z_j}} \quad (2-8)$$

理论上, 输入在经过前向传播后得到的输出值应当与其目标值一致, 评价这种一致性的函数被称为目标函数(在神经网络中也可称为损失函数)。在分类问题中, 常用的损失函数为交叉熵损失函数, 其表达式为:

$$L = -\frac{1}{m} \sum_{k=1}^m \sum_j p_k^j \log q_k^j \quad (2-9)$$

其中,  $m$ 为输入的批量大小,  $q$ 为经过 Softmax 函数的输出值,  $p$ 为目标值的 one-hot 编码向量, 满足下式:

$$p^j = \begin{cases} 1 & \text{目标函数为 } j \\ 0 & \text{目标函数不为 } j \end{cases} \quad (2-10)$$

## 2.3 迁移学习

尽管传统的机器学习或深度学习已经取得了巨大的成功, 并且已经成功应用到了许多实际应用中, 但其仍然存在一定的局限性。机器学习的理想方案是对大量含有标签的样本数据进行训练和测试, 且训练数据和测试数据处于相同的分布条件。但是, 通常情况下, 收集含有标签的数据的代价是昂贵的, 且训练大规模神经网络的时间成本和运算成本也难以忽略。在这种情况下, 迁移学习仅专注于跨领域转移知识, 是解决上述问题和矛盾的一种行之有效的方法。

所谓迁移学习, 也就是指在不同领域问题之间进行知识的转移过程, 利用现有知识来解决不同但相关领域中的问题, 从而减轻对数据特性的要求。在迁移学习中, 领域(Domain)和任务(Task)是两个关键性要素。领域 $D$ 是由特征空间 $X$ 和边缘概率分布 $P(x)$ 这两部分组成的。对于任一特定的领域 $D$ , 任务 $T = \{y, f(\cdot)\}$ 由标签空间 $y$ 和函数 $f(\cdot)$ 组成。函数 $f(\cdot)$ 是预测函数, 能够对未知的样本 $\{x^*\}$ 进行标签预测。从概率的角度看,  $f(x) = P(y|x)$ 。



迁移学习的任务便是使用在源域以及对应的目标任务中学习到的知识去解决目标域中的问题和目标任务。

### 2.3.1 迁移学习分类

迁移学习可以根据源域和目标域的标签以及源域任务和目标任务的区别,进行如下图 2-7 所示的分类:

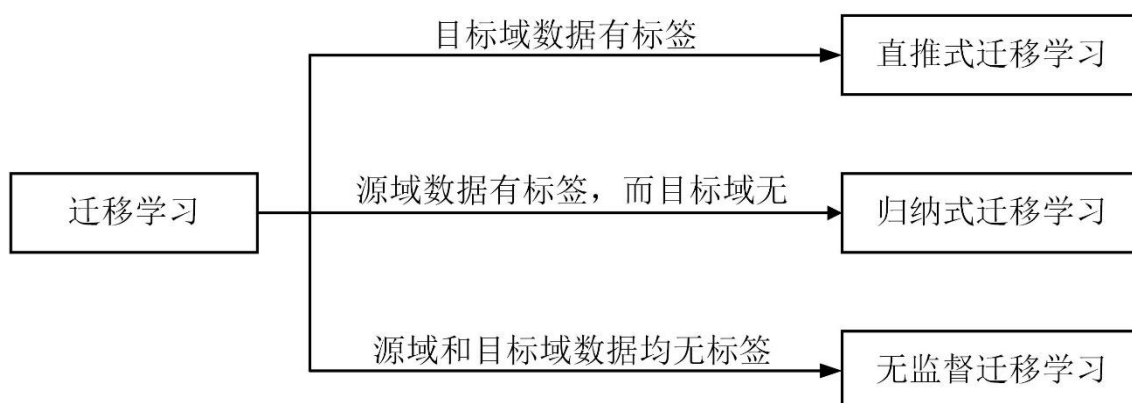


图 2-7 迁移学习的分类

#### 2.3.1.1 归纳式迁移学习

在归纳式迁移学习中,源域和目标域中的数据分布是相同的,源任务和目标任务一定是不同的,但又一定是相关的。

当源域数据不含有标签时,该学习模式与自学习(Self-taught Learning)是相似的,区别在于源域数据没有可用的具有标签的数据,这样便很难通过源任务的条件概率分布 $P_s^{Y|X}$ 构造目标任务的条件概率分布 $P_t^{Y|X}$ 。

当源域数据含有标签时,该学习模式与多任务学习(Multi-task Learning)是相似的,区别在于迁移学习的目标是提升目标域的运算效率与性能,而多任务学习是为了提升所有领域的运算效率与性能。为了同时利用源域和目标域中的有标签数据为目标域训练模型,最为常见的方法便是将损失函数分为两大部分,一个用于源域有标签数据,另一个用于目标域有标签数据,并引入协调函数平衡两种损失的影响。

#### 2.3.1.2 直推式迁移学习

在直推式迁移学习中,源任务和目标任务一定是相同的。对于标签数据,源域数据是含有标签的,但目标域数据是不含有标签的,二者需要具有一定的相关性。

一般情况下,对于直推式迁移学习,源域含有大量具有标签的数据,辅助目标域数据完成同一任务( $T_s = T_t$ )。常见的两种方式<sup>[52]</sup>是:

(1) 通过更改源域实例的权重。在进行目标域模型的训练过程中, 采用极小化期望风险的方法来辅助完成目标任务 $T_t$ :

$$\begin{aligned}\theta^* &= \arg \min_{\theta} \sum_{(x,y) \in D_T} P_{D_T}(x,y) l(x,y,\theta) \\ \theta^* &= \arg \min_{\theta} \sum_{(x,y) \in D_T} \frac{P_{D_T}(x,y)}{P_{D_S}(x,y)} P_{D_S}(x,y) l(x,y,\theta) \\ \theta^* &= \arg \min_{\theta} \sum_{(x_i^s, y_i^s) \in D_T} \frac{P_{D_T}(x_i^s, y_i^s)}{P_{D_S}(x_i^s, y_i^s)} P_{D_S}(x_i^s, y_i^s) l(x_i^s, y_i^s, \theta)\end{aligned}\quad (2-11)$$

上式中,  $l(x,y,\theta)$  为关于  $\theta$  的损失函数, 其中  $\theta$  表示目标领域模型的参数。由全概率公式

$$P(x_i^s, y_i^s) = P(y_i^s | x_i^s) P(x_i^s) \quad (2-12)$$

因此, 权重可以表达为:

$$w(x_i^s) = \frac{P_{D_T}(x_i^s)}{P_{D_S}(x_i^s)} \quad (2-13)$$

当不断调整源域数据的权重时, 则可以通过学习对目标域模型进行训练, 从而完成目标域的任务 Task。

(2) 通过源域和目标域之间共同特征进行迁移。常见的一种方法便是域自适应, 即学习一对映射函数 $\{\varphi_s(\cdot), \varphi_t(\cdot)\}$ , 将来自源域和目标域的数据映射到共同的特征空间, 从而使域之间的差异性减少。然后使用映射之后的源域和目标域数据在新的特征空间上训练目标分类器。为了测试目标域上的未见数据, 首先需要将新的数据映射到新的特征空间上, 然后执行训练好的目标分类器。

然而, 这一问题的重点在于, 如何学习良好的映射函数 $\{\varphi_s(\cdot), \varphi_t(\cdot)\}$ 使得不用域的样本映射到由域不变特征组成的公共空间。目前, 最大均值差异, 作为常见的度量标准, 被引入以测量学习特征的域不变性。

### 2.3.1.3 无监督迁移学习

在无监督迁移学习中, 目标域中只有无标签的数据可以参与模型的训练中, 且源域和目标域、源任务和目标任务均各不相同( $D_s \neq D_t$  和  $T_s \neq T_t$ )。

### 2.3.2 迁移学习方法

迁移学习常见的方法有三种: 数据分布自适应、特征选择法以及子空间学习法(如图 2-8 所示)。数据分布自适应法的操作思想是通过变换操作, 将概率分布不同的源域数据和目标域数据之间的分布距离拉近。特征选择法的核心在于通过机

器学习等方法寻找源域和目标域这两个领域之间的公共特征，并基于这些公共特征构建模型。子空间学习学习法则是将概率分布不同的源域和目标域变换至子空间中，二者在子空间中具有相似的分布，从而可以在子空间中构建模型。

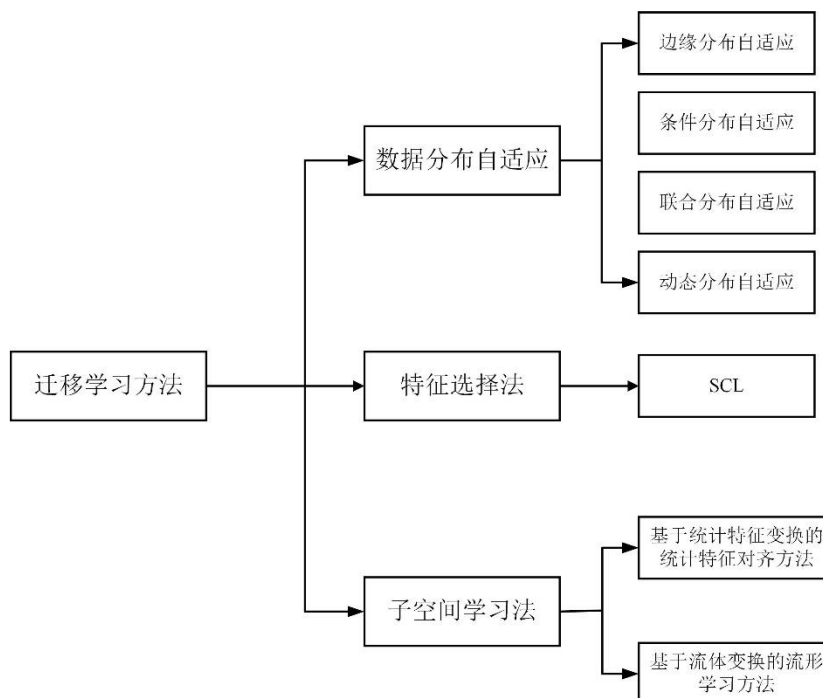


图 2-8 迁移学习方法分类

本文所采用的基于 MDA 和基于 JDA 的轴承故障诊断方法研究均是基于数据分布自适应法。本文接下来将对边缘分布自适应、条件分布自适应以及联合分布自适应展开介绍。

### 2.3.2.1 边缘分布自适应

如图 2-9 所示，源域数据和目标域数据二者边缘分布不同，也就是其整体分布不相似。边缘分布自适应的目标在于减小二者的边缘概率分布的距离，从而达到迁移学习的目的。

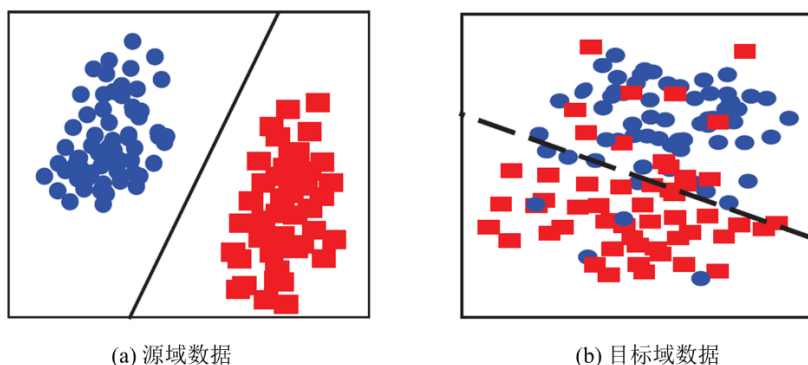


图 2-9 边缘分布差异

边缘分布自适应是通过减小目标域和源域之间的概率分布 $P(x_s)$ 和 $P(x_t)$ 来近似减小源域和目标域之间的差异，即：

$$Dist(\mathcal{D}_s, \mathcal{D}_t) \approx \|P(x_s) - P(x_t)\| \quad (2-14)$$

上式中， $Dist(\mathcal{D}_s, \mathcal{D}_t)$ 表示源域 $\mathcal{D}_s$ 和目标域 $\mathcal{D}_t$ 之间的距离。由于 $P(x_s) \neq P(x_t)$ ，因此我们不能直接将二者之间的距离简单相减。目前，最为常见的做法是寻找某个特征映射 $\phi$ ，使得映射后源域和目标域的分布满足：

$$P(\phi(x_s)) \approx P(\phi(x_t)) \quad (2-15)$$

并假设二者的条件分布也满足：

$$P(y_s|\phi(x_s)) \approx P(y_t|\phi(x_t)) \quad (2-16)$$

这样，问题的核心便转变为寻找映射函数 $\phi$ 。通常情况下，我们引入最大均值差异来衡量目标域和源域之间的距离：

$$Dist(x_s, x_t) = \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_i^s) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_j^t) \right\|_{\mathcal{H}}^2 \quad (2-17)$$

在这里，我们引入一个核矩阵 $K$ 以及一个MMD矩阵 $L$ ：

$$K = \begin{bmatrix} K_{s,s} & K_{s,t} \\ K_{t,s} & K_{t,t} \end{bmatrix} \quad (2-18)$$

$$L = [L_{ij}] = \begin{cases} \frac{1}{n_1^2}, x_i, x_j \in X_s \\ \frac{1}{n_2^2}, x_i, x_j \in X_j \\ -\frac{1}{n_1 n_2}, others \end{cases} \quad (2-19)$$

为了更为简单的构造结果，我们引入一个更低维度的矩阵 $W$ 来进行降维：

$$\tilde{K} = \left( K K^{-\frac{1}{2}} \tilde{W} \right) \left( \tilde{W}^T K^{-\frac{1}{2}} K \right) = K W W^T K \quad (2-20)$$

于是，问题的优化目标便转化为了：

$$\min_W tr(W^T K L K^T M) \quad (2-21)$$

$$s. t. W^T K H K M = I_m$$

其中， $H$ 是一个中心矩阵：

$$H = I_{n_1+n_2} - \frac{1}{n_1+n_2} \mathbf{1}\mathbf{1}^T \quad (2-22)$$

### 2.3.2.2 条件分布自适应

如图 2-10 所示，源域数据和目标域数据二者条件分布不同，也就是其整体分布相似，但具体到每个类里其分布不相似。条件分布自适应的目标在于减小二者的

条件概率分布的距离，从而达到迁移学习的目的。

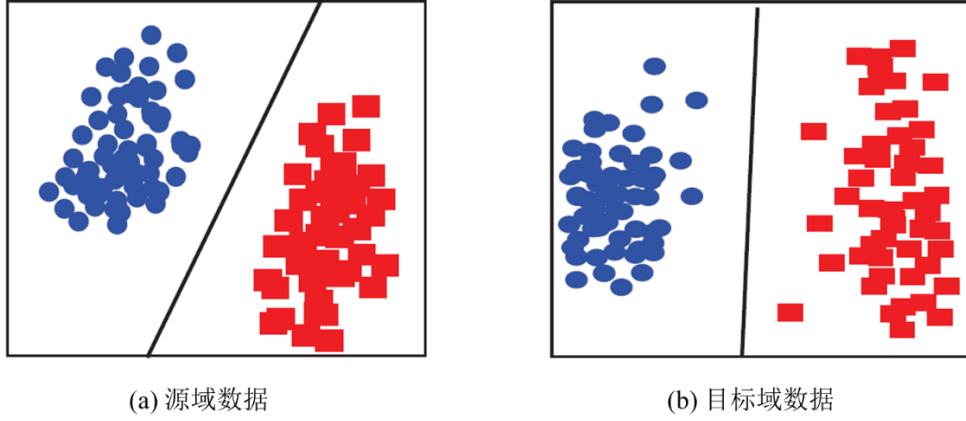


图 2-10 条件分布差异

条件分布自适应是通过减小目标域和源域之间的概率分布 $P(y_s|x_s)$ 和 $P(y_t|x_t)$ 来近似减小源域和目标域之间的差异，即：

$$Dist(\mathcal{D}_s, \mathcal{D}_t) \approx \|P(y_s|x_s) - P(y_t|x_t)\| \quad (2-23)$$

目前，常用的方法是 Wang 等人提出的 STL(Stratified Transfer Learning)方法。其基本思路（如图 2-11 所示）是：首先，通过各种方法（作者采用的是大多数投票思想），将不含标签的数据进行伪标签处理；在再生核希尔伯特空间内，通过类内相关性完成空间降维的操作；最后，对不含标签的数据进行二次标定，增大伪标签的准确率。

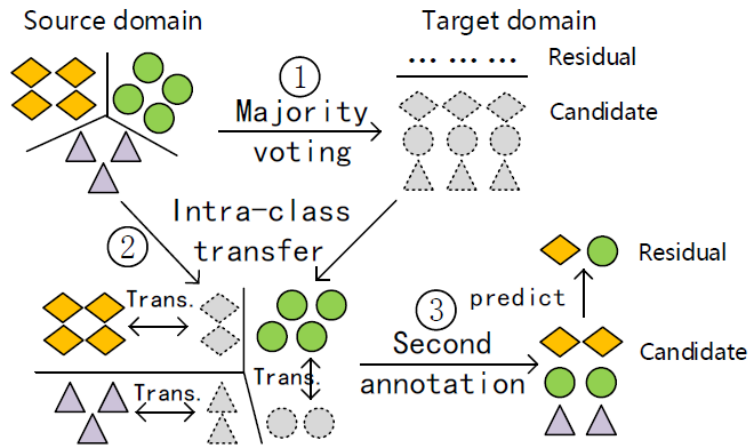


图 2-11 SCL 基本思路

实际操作中，我们通过伪标签完成类内迁移（详细介绍见后文）：

$$Dist(\mathcal{D}_s, \mathcal{D}_t) \approx \sum_{c=1}^C \left\| \frac{1}{n_s^{(c)}} \sum_{x_i^s \in \mathcal{D}_s^{(c)}} \phi(x_i^s) - \frac{1}{n_t^{(c)}} \sum_{x_j^t \in \mathcal{D}_t^{(c)}} \phi(x_j^t) \right\| \quad (2-24)$$

其中,  $c \in \{1, 2, \dots, C\}$  表示类别标记。我们采用与边缘分布自适应相同的核方法, 得到优化目标:

$$\begin{aligned} \min_W \sum_{c=1}^C \text{tr}(W^T K L_c K M) \\ \text{s.t. } W^T K H K M = I_m \end{aligned} \quad (2-25)$$

联合分布自适应指的是减小源域和目标域的联合概率分布的距离, 即:

$$\text{Dist}(\mathcal{D}_s, \mathcal{D}_t) \approx \|P(x_s) - P(x_t)\| + \|P(y_s|x_s) - P(y_t|x_t)\| \quad (2-26)$$

## 2.4 本章小结

本章详细介绍了本文构建、学习、训练轴承故障诊断模型所需要的理论知识, 首先介绍了信号处理方法的三种常用方法, 接着以卷积神经网络的网络结构和参数学习为例介绍了深度学习的相关知识, 最后详细地介绍了迁移学习, 包括基于源域和目标域的标签以及源域任务和目标域任务的差别的分类以及迁移学习的三种常见方法, 最后通过边缘分布自适应和条件分布自适应对数据自适应法进行系统介绍。

### 3 基于 Finetune 的轴承故障诊断方法

对于含有标签的目标域数据而言，根据数据集特征选择合适的模型加以改进并从头开始训练，最后在训练好的模型进行故障诊断，这样固然能够得到较好的诊断效果，但其时间成本和运算成本有时是难以接受的。因此，在这一章节，本文选择采用深度迁移学习方法中最基本的方法——微调 Finetune 建立并训练轴承故障诊断模型，希望通过深度迁移方法在保证诊断准确率的前提下减少模型的时间成本和运算成本。

#### 3.1 微调 Finetune

实际应用中，我们在处理新的数据集和任务时，通常会遇到以下两大问题：

(1) 数据集量级不够，难以在有限的数据集上训练泛化能力强、准确率高的模型；

(2) 对于一个新的模型，我们从网络参数初始化开始进行训练的话，训练的时间成本和运算成本有时是难以承受的。

然而，倘若直接利用已经训练好的模型来完成任务，其有时并非能够完全适用于我们自己的任务。此时，便需要对已经训练好的网络模型进行微调 Finetune。简而言之，Finetune 就是利用别人已经训练好的模型，针对自己的数据集和任务，进行合理的参数和结构调整，在此基础上继续训练，从而减少模型训练的时间成本和运算成本。其基本思想如下图 3-1 所示：

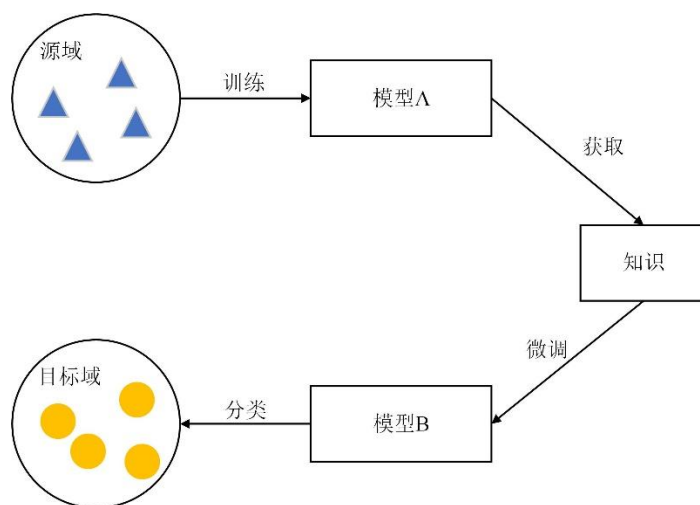


图 3-1 Finetune 示意图

本文拟采用卷积神经网络进行微调，其一般流程为：

- 1) 先利用相关大型数据集（如 ImageNet 数据集）对网络参数进行训练，或直接利用 TensorFlow 或 Pytorch 等环境调用已经训练完备的模型；
- 2) 利用训练好的卷积神经网络，针对特定的应用领域数据进行特征提取；
- 3) 利用提取后的特征，针对数据训练卷积神经网络或其他分类器。

该模型框架如下图 3-2 所示：

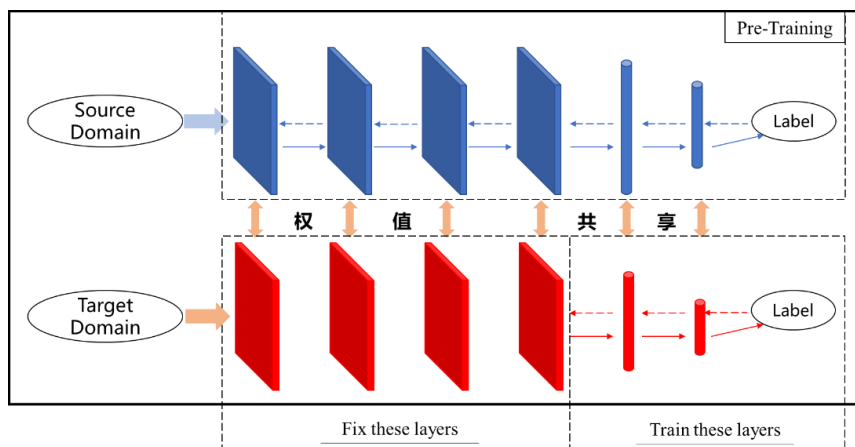


图 3-2 基于 Finetune 的轴承故障诊断模型框架

然而，当源域与目标域的分布差异过大的时候，Finetune 的准确率会明显下降。这是因为随着网络层数的增加其提取的特征也由基础几何特征等通用特征逐渐变成直接反应任务的特殊特征，这些高层特征并不能在分布差异较大的情况下直接共享。

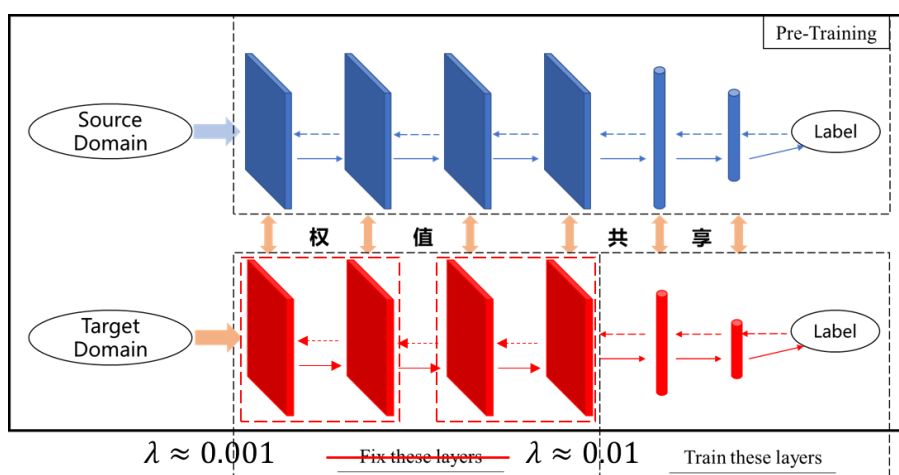


图 3-3 改进 Finetune 模型框架

本文思路就是在目标域网络中，将原本固定的网络层也加入训练中，对各层设置不同的学习率。对于网络底层来说，其仅提取通用特征，学习率设置得稍微小一



点,而网络高层的学习率设置得稍微大一点。意思就相当于,将原本训练好的模型参数作为目标域网络模型的初始参数,使得网络模型能够更快更好地进行收敛(如上图 3-3 所示)。

## 3.2 基于 ResNet 网络的微调

深度残差网络 ResNet 是由微软研究院的何恺明、张祥雨<sup>[53]</sup>等人于 2015 年提出的,在当年 ILSVRC (ImageNet Large Scale Visual Recognition Challenge) 中取得了冠军。作为特征提取器,ResNet 网络已被用于多项图像识别的任务中,取得较好的效果。

传统深度神经网络在网络到达一定深度后,当增加网络层数时,即使选择 ReLU 函数作为激活函数,在反向传播的过程中浅层网络同样会出现梯度消失或梯度爆炸的情况,从而导致网络训练结果准确率下降。深度残差网络通过引入残差块,结合“跳远连接”解决梯度消失和梯度爆炸的问题。

传统的残差块(如下图 3-4 所示)的表达式为:

$$x_{l+1} = h(x_l) + \mathcal{F}(x_l, \varpi_l) \quad (3-1)$$

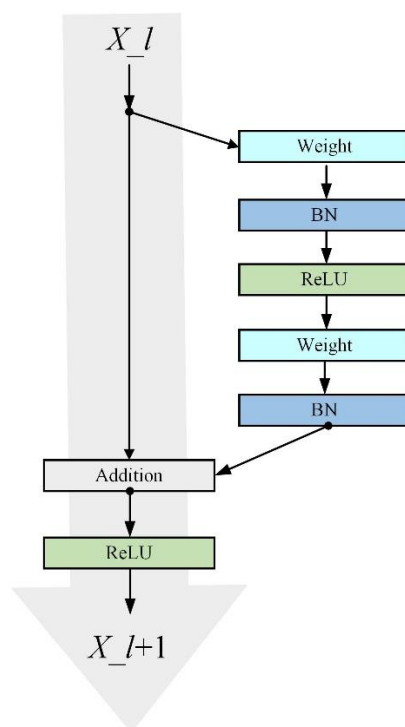


图 3-4 残差块结构图

上图中,Weight 层在卷积网络中代表卷积操作,Addition 层表示单位加操作。

BN 代表批标准化层, ReLU 代表激活层(也可以称为线性修正单元)。

残差块共分为两部分:图 4.3 左侧部分的直接映射部分以及右侧部分的残差部分。直接映射部分将信息跳过一层或者多层直接传递至神经网络的深层,让网络学习恒等函数变得较为容易。

由于本文选择的轴承训练集规模并不庞大,因此本文选择采用 ResNet18 网络作为模型的预训练网络。ResNet18 网络是由残差块堆叠而成的网络,其由 1 个卷积层、4 个残差块(每个残差块共含有 2 个卷积层)以及一个全连接层组成。该网络的内部参数见表 3-1 所示:

表 3-1 ResNet18 网络内部参数

层名称	输出尺寸	层参数量
Conv1	112×112	7×7,64
Max Pool	56×56	3×3,64
Conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
Conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$
Conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$
Conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$
FC	1×1	—
FLOPs		$1.8 \times 10^9$

在本文的模型中,除去最后一层全连接层,其余各层的网络参数(权重和偏置)的初始值均不是随机生成的,而是已经训练好的 ResNet18 网络的参数值,这在很大的程度上减少了模型训练学习的时间,并保证了模型的准确率。

### 3.3 轴承数据集介绍

本文参与模型训练的数据来源于美国凯斯西储大学(CWRU,全称为 Case Western Reserve University)公开轴承数据集。该数据集是通过加速度计传感器,模

拟各种工况条件,进而分别收集实验台在各类故障情况下的振动信号。CWRU 数据集为全球通用的标准数据集,被国内外广大学者引用。

如下图 3-5 所示,该实验平台由一个 1.5KW (2 马力)的电动机、一个扭矩传感器/译码器、一个功率测试计以及电子控制器组成。对于待测轴承,其支撑着电动机的转轴。驱动端的轴承型号为 6205-2RS JEM SKF 深沟球轴承,采样频率为 12kHz 和 48kHz。风扇端的轴承型号为 6203-2RS JEM SKF 深沟球轴承,采样频率为 12kHz。

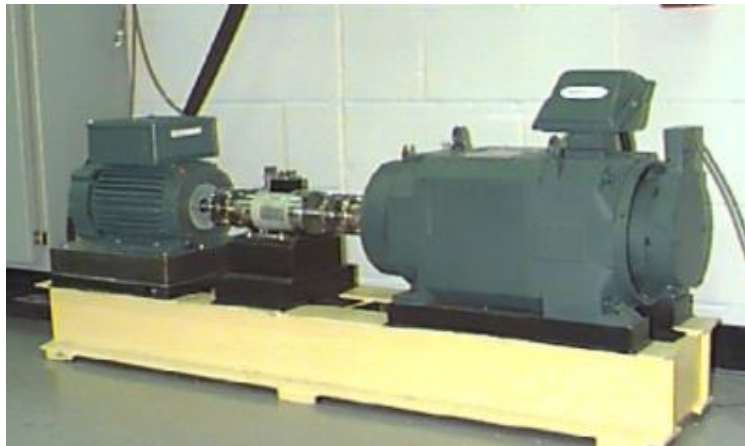


图 3-5 CWRU 实验平台图

实验台采用下列方式来模拟真实工况下的故障发生:

- (1) 轴承的损伤是由电火花加工的单点损伤;
- (2) SKF 轴承是用来检测直径为 0.1778、0.3556、0.5334 毫米的损伤;
- (3) NTN 轴承是用来检测直径为 0.7112、1.016 毫米的损伤。

电动机风扇端和驱动端的轴承座上方各放置一个加速度计传感器用来采集故障轴承的振动加速度信号。其中,振动信号由 16 通道数据记录仪采集而来,功率和转速通过扭矩传感器/译码器测得。收集到的振动信号通过软件 Matlab 进行处理,并以.mat 格式保存。

对于轴承故障类型来说,实验台共模拟了三种轴承部位的故障:内圈故障、外圈故障以及球故障,并结合不同的轴承尺寸以及负载环境生成了四十八种故障类型,同时该数据集还包含了十二种健康轴承的振动信号。

### 3.3.1 数据切片

在 CWRU 公开轴承数据集中,轴承振动信号的收集工作可能来源于不同的传

感器物理位置、不同的负载条件、不同的采样频率等(如下表 3-2 所示)。在这里,本文首先以负载情况为 0 马力、采样频率为 12 kHz 的情况下驱动端轴承的相关数据为例,对数据进行相关预处理。其他数据在进行模型训练前的处理操作与该数据集相同。

表 3-2 轴承数据的不同来源

	收集条件	备注
1	传感器的物理位置	基座、风扇端、驱动端
2	负载条件	0 马力、1 马力、2 马力、3 马力
3	采样频率	12kHz、48kHz

在此前提下,本文共选取了如下表 3-3 所示的十种故障类型,并依次将其标号为 0,1,2,...,9:

表 3-3 故障类型及标签

故障标签	故障部位	尺寸（英寸）	电机载荷	采样频率
0	滚动体故障	0.007	0hp	12kHz
1	内圈故障			
2	外圈故障			
3	滚动体故障	0.014		
4	内圈故障			
5	外圈故障			
6	滚动体故障	0.021		
7	内圈故障			
8	外圈故障			
9	正常	——		

在负载情况为 0 马力时,电动机的近似转速约为 1797 转每分钟,于是电动机转动一圈会收集到  $\frac{60}{1797} \times 12000 = 400.98$  个采样点。因此,本文近似估计电动机每转动 1 圈,加速度计传感器便会收集到 400 个采样点。

正常基座的轴承振动信号可用如下时序图所表示:

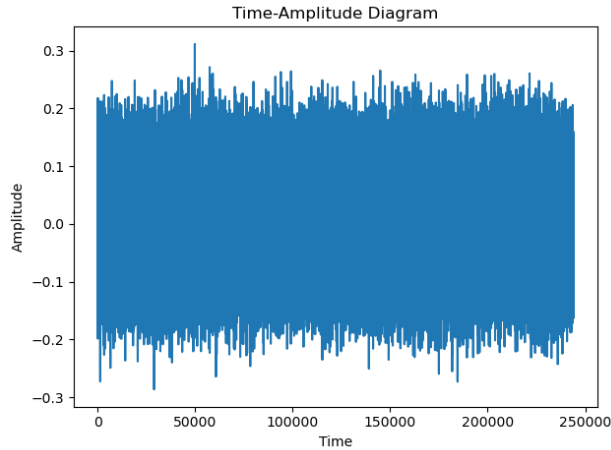


图 3-6 正常基座的轴承振动信号时序图

在上图 3-6 中，一方面，由于采样点数量规模较为庞大难以发现其内在规律与特征。另一方面，深度学习的前提是大量的样本参与训练，因此倘若不对数据进行切片划分的话，后文建立的模型难以进行学习和训练。

于是，本文根据轴承信号的周期性，将每类故障类型进行切片，以 1200 个采样点（电动机转动 3 圈）作为一个样本。以正常基座的轴承信号第一个样本为例，该样本的时序图如下图 3-7 所示：

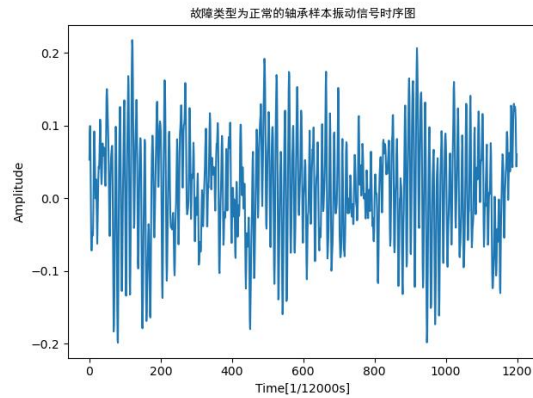


图 3-7 正常基座某样本的振动信号时序图

为了保证网络训练过程中各类标签样本数量的平衡性，本文选择在切分好后的各类故障类型中分别随机选择 100 个样本，共 1000 个样本参与模型的学习训练和测试中，其中随机选择 800 个样本作为测试集，200 个样本作为训练集。

### 3.3.2 数据处理

前文中，我们共获得了 1000 个样本，每个样本共含有 1200 个采样点数据。如果不对样本进行处理变换，直接将其作为卷积神经网络的输入，特征提取较为复杂，且提取的特征可能会包含无用的信息，因此本文决定进行时频域信号处理，将样本

的采样点数据转化为时频图，作为卷积神经网络的输入。

对于平稳信号，傅里叶变换将其分解为频率分量，并确定其相对强度。傅里叶变换的表达式为：

$$F(w) = \int_{-\infty}^{\infty} f(t)e^{-jw\tau}dt \leftrightarrow \quad (3-2)$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(w)e^{jw\tau}dw \quad (3-3)$$

然而，当信号不稳定时，在处理时我们需要考虑其时变因素。本文采取的方法是采用短时傅里叶变换。短时傅里叶变换借用窗函数将一段完整的时变信号分解为若干段信号（如图 3-8 所示），并假设其在该窗口是平稳的。

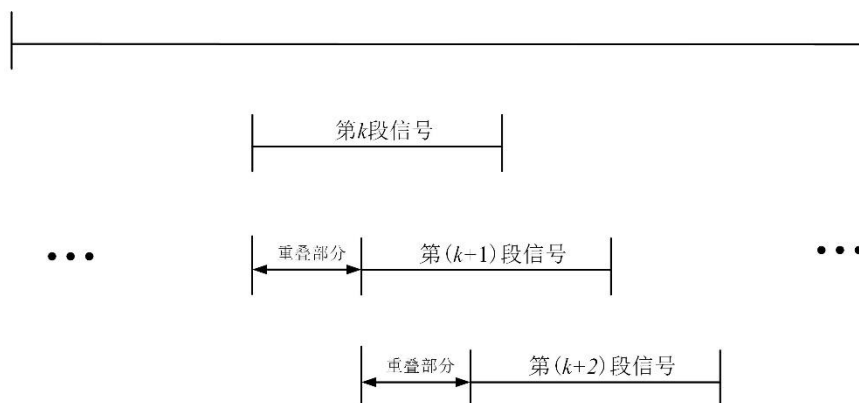


图 3-8 信号分解过程示意图

对于每段被分解后的平稳信号，运用傅里叶变换进行处理：

$$F(w, \tau) = \int_{-\infty}^{\infty} f(t)\Psi^*(t - \tau)e^{-jw\tau}dt \quad (3-4)$$

其中， $\Psi(t)$ 函数表示短时傅里叶变换的窗函数，其有两个参数： $w$ 和 $\tau$ 分别表示窗函数的调制和转换，完成窗函数的基本功能。常用的窗函数有矩形窗和海明窗。

对于大小为  $N$  的矩形窗而言，其函数表达式为：

$$w(n) = \begin{cases} 1, & 0 \leq n \leq (N - 1) \\ 0, & \text{others} \end{cases} \quad (3-5)$$

对于大小为  $N$  的海明窗而言，其函数表达式为：

$$w(n) = \begin{cases} 0.5(1 - \cos \frac{2\pi n}{N-1}), & 0 \leq n \leq (N - 1) \\ 0, & \text{others} \end{cases} \quad (3-6)$$

矩形窗口不符合过高侧叶的要求，因为侧叶会泄漏更多的能量。因此本文选择采用海明窗作为窗函数进行短时傅里叶变换。

短时傅里叶变换的流程如下图 3-9 所示：

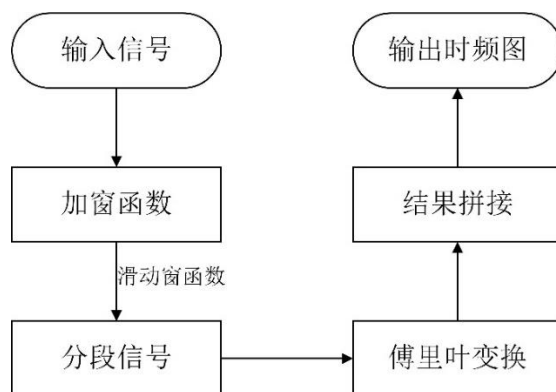


图 3-9 短时傅里叶变换示意图

本文以表 3-所示的参数对各样本进行短时傅里叶变换操作，将所有样本均转化为二维的时频图，以方便后续卷积神经网络的输入和训练。

表 3-4 短时傅里叶变换的参数设置

参数名称	参数值
窗函数	海明窗 Hamming
窗大小	256
步长间隔	40

在经过短时傅里叶变换后，正常基座的轴承振动信号的第一个样本的时频图如下图 3-10 所示，图 3-11 分别表示了某个内圈故障样本和某个外圈故障样本的时频图。从图 3-10 和图 3-11 可以看出，在经过短时傅里叶变换后，不同故障类型的时频图表现均明显有较大的差异。

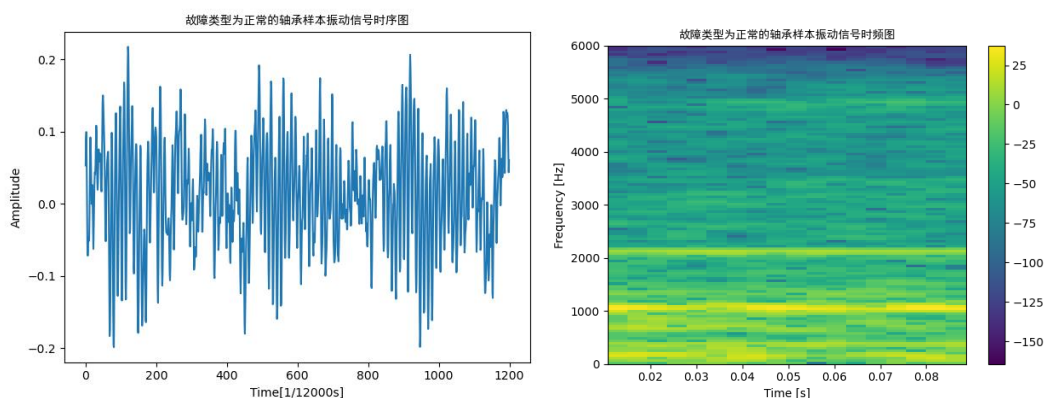
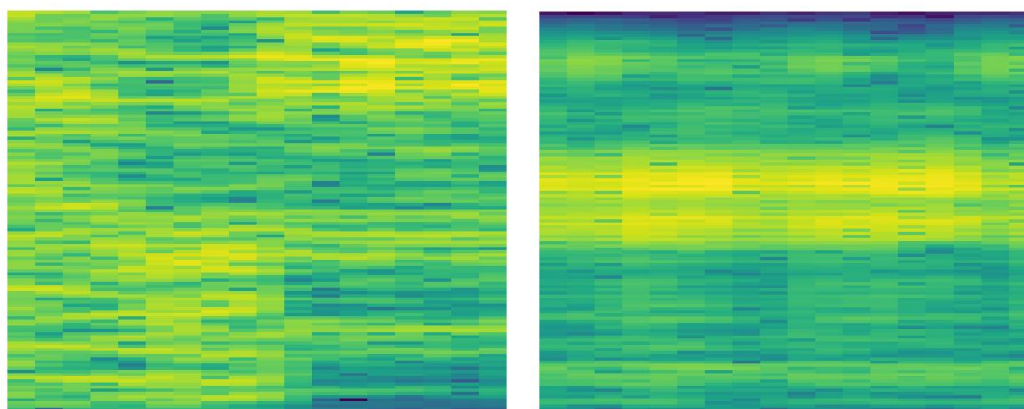


图 3-10 正常基座某样本的振动信号时序图和对应时频图





(a) 内圈故障

(b) 外圈故障

图 3-11 内圈故障和外圈故障的样本时频图示意图

### 3.4 实验验证

在本小节中, 本文基于上述预处理后的 CWRU 数据集, 对基于微调 Finetune 的轴承故障诊断模型进行训练和测试, 来验证所提算法的有效性和优越性。

在对网络模型进行训练时, 本文所采用的计算机配置为 i5-8300H CPU、GTX 1060 Ti 显卡, 24 GB 内存。本文基于深度学习, 所选择的框架为 Google 公司的 TensorFlow2.0, 选择的计算机编程语言为 Python 3.8。

#### 3.4.1 诊断流程

首先, 本文将 CWRU 数据集进行数据预处理, 将数据集切分为 1200 采样点/样本, 并通过短时傅里叶变换将样本转换为二维时频图, 进一步根据一定比例(8:2)将所有数据划分为训练集和测试集。训练集参与模型的反向传播训练中, 而测试集则作为评价数据用来验证模型训练结果的好坏。

由于 ResNet18 网络已经在 ImageNet 网络训练完毕, 且保存为模型文件, 故本文直接选择导入 ResNet18 模型文件, 对其参数进行设置再训练, 完成迁移学习。

为了满足 ResNet 网络的输入尺寸, 本文将样本的二维时频图的图片大小调整为  $3 \times 244 \times 244$ 。此外, 由于本文训练集的故障标签种类与原网络全连接层的最后一层不匹配, 故本文对该网络模型的最后一层进行调整, 网络层节点数改为 10。并对该层的可训练参数(权重和偏置)进行随机初始化。

因为卷积层负责的是提取图像的抽象特征, 其工作是较为通用的分类任务, 因此本文在训练时将这些层的学习率设置得较小, 为 0.001。由于全连接层负责的是



轴承故障诊断的工作,且其网络参数并未参加预训练,因此将这些层的学习率设置为 0.01。

本文所设计的基于微调 Finetune 的轴承故障诊断流程如图 3-12 所示。

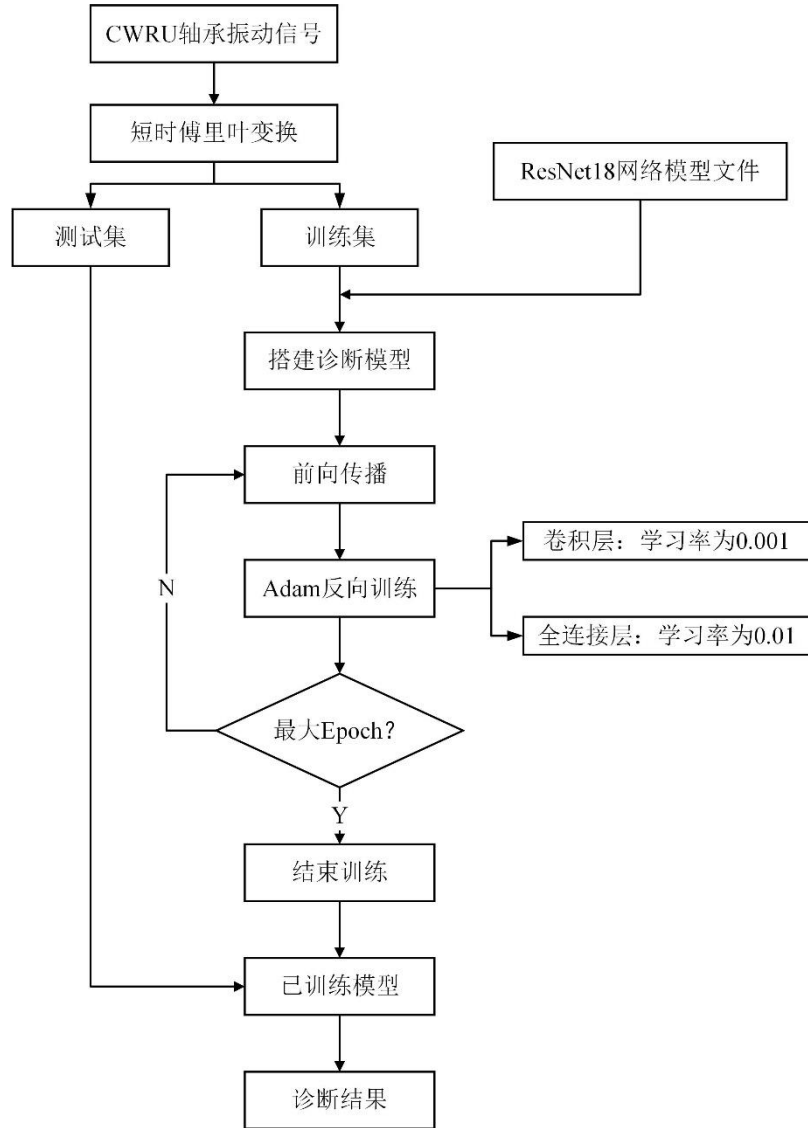


图 3-12 基于 Finetune 的模型诊断流程图

本文基于 Finetune 方法和 ResNet18 网络设计并建立了针对目标域数据含有标签的情况下的轴承故障诊断模型,该模型在 TensorFlow2.0 环境下进行训练和测试。

对于模型反向传播时所采用的梯度下降法,常用的有:随机梯度下降法 SGD、批量梯度下降法 BGD、Momentum 算法以及 Adam 算法。随机梯度下降法和批量梯度下降法能够较为快速地完成收敛,但当网络层数逐渐加深时这两种方法均可能会陷入局部最优。而 Momentum 算法和 Adam 算法作为梯度下降法的改进方法可以有效降低这一问题带来的影响。

由于本文所选择的基本模型为 ResNet18 网络，该网络涉及到的参数和超参数均较为庞大，因此本文选择 Adam 算法作为优化算法进行反向传播训练。

传统的梯度下降法在训练过程中始终保持单一的学习率 $\alpha$ 来进行参数（权重和偏置）的更新， $\alpha$ 在训练过程中并不会随着迭代次数而发生变化。Adam 算法则采用的是学习率自适应法，通过计算梯度的一阶矩估计和二阶矩估计，对于不同的参数自适应地设计特定的学习率，使得算法能够表现出较优的性能。Adam 算法的具体流程如下表 3-5 所示：

表 3-5 Adam 算法流程

算法 1: Adam 算法

输入：步长 $\epsilon$ （默认： $\epsilon = 0.001$ ）

输入：矩估计指数衰减速率 $\rho_1 (= 0.9)$ 和 $\rho_2 (= 0.999)$

输入：常数 $\delta = 1 \times 10^{-8}$ ，用于数值未稳定

输入：初始参数 $\theta$

初始化：一阶矩变量 $\mathbf{s} = \mathbf{0}$

初始化：二阶矩变量 $\mathbf{r} = \mathbf{0}$

初始化：时间步 $t = 0$

while 没有达到终止条件

do

从训练集中选择一个批量样本 $\{x^{(1)}, x^{(2)}, \dots, x^{(n)}\}$ 及对应目标 $y^{(i)}$

梯度计算： $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(x^{(i)}; \theta), y^{(i)})$

更新： $\mathbf{s} \leftarrow \rho_1 \mathbf{s} + (1 - \rho_1) \mathbf{g}$

更新： $\mathbf{r} \leftarrow \rho_2 \mathbf{r} + (1 - \rho_2) \mathbf{g} \odot \mathbf{g}$

修正： $\hat{\mathbf{s}} \leftarrow \frac{\mathbf{s}}{1 - \rho_1^t}$

修正： $\hat{\mathbf{r}} \leftarrow \frac{\mathbf{r}}{1 - \rho_2^t}$

更新权重： $\theta \leftarrow \theta - \epsilon \frac{\hat{\mathbf{s}}}{\sqrt{\hat{\mathbf{r}} + \delta}}$

end

除此之外，在模型的训练过程中，其他参数设置如下表 3-6 所示：

表 3-6 模型训练参数设置

参数名称	参数值
epoch	50
batch size	64
learning rate	0.001
优化器	Adam 优化器

本文所采用的基于 Finetune 的轴承故障诊断模型的求解流程如下表 3-7 所示:

表 3-7 基于 Finetune 的模型求解流程

算法 2: 基于 Finetune 的诊断算法

输入: 迭代次数 epoch(= 50)

输入: 小批量大小 batch size(= 64)

输入: 学习率 learning rate( $\lambda_1 = 0.001, \lambda_2 = 0.01$ )

初始化: 导入 ResNet18 网络的模型文件

初始化: 将网络最后一层全连接层节点数改为 10

for step in epoch:

    前向传播: 卷积层→激活层→池化层→...→全连接层

    反向传播: 计算梯度  $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(x^{(i)}; \theta), y^{(i)})$

    梯度更新:  $\theta \leftarrow \theta - \lambda \frac{\partial L}{\partial \theta}$

    step  $\leftarrow$  step +1

end

测试集上进行模型测试

### 3.4.2 诊断结果

基于上述诊断流程和训练参数, 本文利用搭建好的模型对所选取的数据集(负载情况为 0 马力、采样频率为 12 kHz 的情况下驱动端轴承的相关数据)进行训练。

训练结果如下图 3-13 所示:

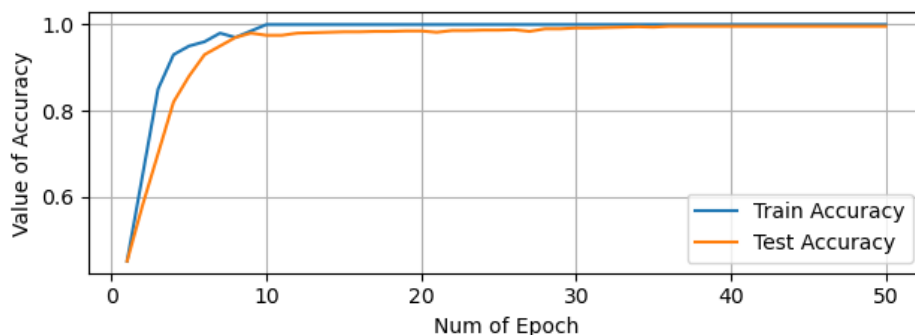


图 3-13 基于 Finetune 的模型训练结果

由图 3-13 可知，该模型在训练至 10 代左右便开始收敛，该次训练下模型最终在测试集的轴承诊断准确率高达 99.72%。由此可以说明，基于 Finetune 的轴承故障诊断模型不仅可以大大减少训练时间，并且能够保证较高的准确率。

此外，本文进行重复实验（结果如下图 3-14 所示），发现基于 Finetune 的模型诊断的结果准确率均保持在 99.50% 以上，能够很好地完成轴承故障识别的任务。

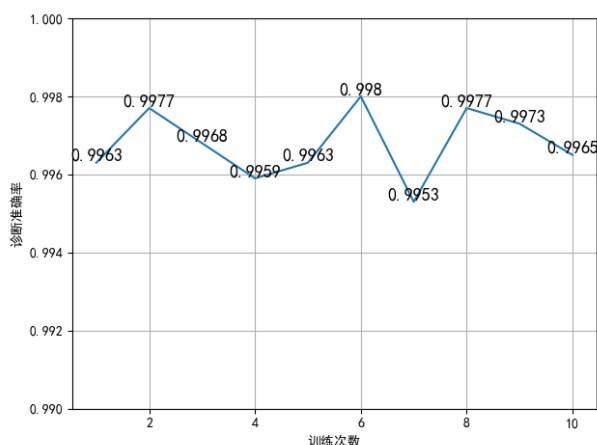


图 3-14 模型重复训练的准确率

除此之外，为了比较本文所提算法的优越性，本文还将基于 Finetune 的轴承故障诊断模型结果与基于传统深度神经网络和卷积神经网络的诊断模型结果进行对比，所选方法如下表 3-8 所示，对比结果如下表 3-9 所示：

表 3-8 验证方法对比

所选方法	输入数据	数据量
本文方法	时频图	800
卷积神经网络	时频图	800
深度神经网络	原始信号	800

表 3-9 多种模型效果对比

	Finetune	DNN	CNN
实验 1	99.72%	98.86%	99.23%
实验 2	99.77%	98.69%	99.18%
实验 3	99.68%	99.20%	99.31%
实验 4	99.59%	98.42%	99.34%
实验 5	99.63%	98.79%	98.56%
平均值	99.678%	98.792%	99.124%

由上表 3-9 以及图 3-15，我们可以得知：在三种模型的对比中，基于 Finetune 的轴承故障诊断模型的准确率稳定保持在最高的水平，效果相较于 DNN 和 CNN 均表现良好。

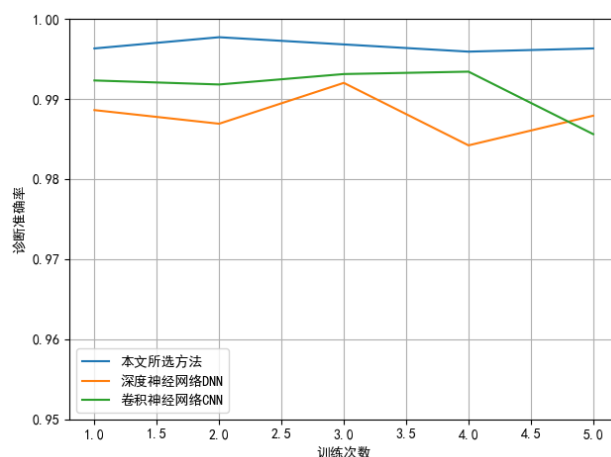


图 3-15 多种模型效果对比图

此外，由下图 3-16 所示，对于传统的深度神经网络，网络需要经过约 1000 代的迭代过程才会趋于收敛，因此基于 Finetune 的诊断模型的另一优势便是：网络训练迭代次数显著降低，直接表现为训练时间相较于 DNN 和 CNN 大幅度下降。

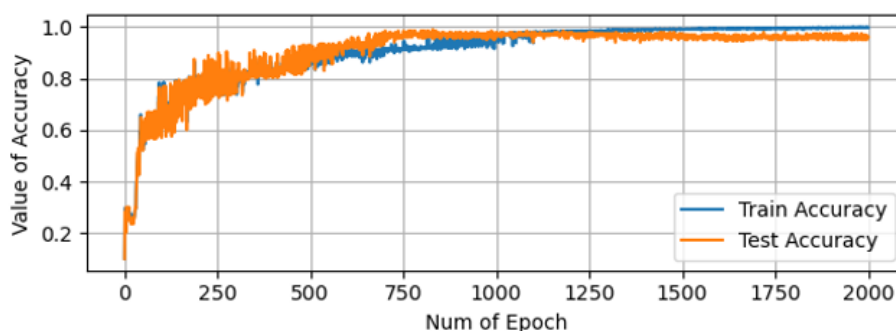


图 3-16 DNN 模型的运算结果

### 3.5 本章小结

本章在 ResNet18 网络模型的基础上, 利用 Finetune 的方法建立轴承故障诊断模型, 通过对卷积层和全连接层不同的学习率设置, 利用 Adam 优化器进行网络模型的训练。

在这一章节中, 本文还详细介绍了模型训练时所采用的数据集——凯斯西储大学 CWRU 轴承公开数据集, 并以负载情况为 0 马力、采样频率为 12 kHz 的情况下驱动端轴承的相关数据为例, 对其进行时域信号分析和时频域信号分析, 将数据进行切片划分。

最后将该模型的诊断结果与传统的深度神经网络和卷积神经网络模型进行效果对比, 分析对比三种模型方法的诊断准确率以及收敛速度, 最后得出结论: 基于 Finetune 的轴承故障诊断模型表现出较高的诊断准确率, 并且能够在很快的时间内完成收敛, 模型训练的时间成本大大减小。

## 4 基于 MDA 和 JDA 的轴承故障诊断方法

实际工作情况下,收集带有标签的信号往往比收集不含标签的信号成本更高,操作更难。因此,在实际应用中,大部分情况都只能收集到不含标签的数据。在这一章节,本文选择采用 MDA (Deep Domain Confusion)以及联合分布适配 JDA (Joint Distribution Adaptation)方法来建立并训练轴承故障诊断模型,对不含标签的样本数据集进行故障诊断

### 4.1 基于 ResNet 网络的 MDA

DDC (Deep Domain Confusion)方法最早是由加州大学伯克利分校 Tzeng 等人于 2014 年提出<sup>[54]</sup>,用以解决深度神经网络的域自适应问题。传统的 DDC 采用了在 ImageNet 数据集上训练好的 AlexNet 网络进行自适应学习。DDC 的思路就是:先让源域的网络进行训练,同时在源域和目标域的网络中加入一个共同的自适应层,从而进行特征对齐。对其之后,目标域上的网络就获得了和源域网络相似的特征。对获得的特征进行微调,就可以得到针对目标域任务训练完毕的网络。当然,也可以将目标域中的标记数据加入源域中一起训练。

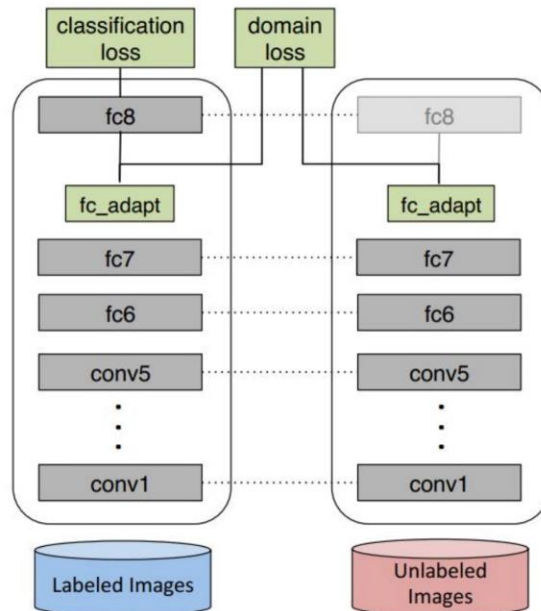


图 4-1 DDC 的模型框架

其模型框架如上图 4-1 所示,DDC 固定了 AlexNet 的前 7 层,在第 8 层(分类器前一层)上加入了自适应的度量。网络中共包含两个流向,第一条流向的输入



为源域数据，即带有标签的数据；另一条流向的输入为目标域数据，即未带有标签的数据。两个流向的卷积神经网络共享权值。和以往不同的是，该网络在两个流向的特征层之间增加了一个适应层，并通过适应层的输出计算出一个 domain loss，也就是前面提到的自适应度量。一般而言，自适应度量采用的是源域数据和目标域数据特征之间的 MMD（最大平均差异）距离作为 domain loss。

由于目标域和源域的数据分布有差异，因此直接进行网络微调可能会导致模型诊断率下降，因此需要引入自适应度量准则，减小源域和目标域的距离。

#### 4.1.1 最大均值差异

前文中提到，迁移学习的任务是为了利用源域及其学习任务的知识来辅助学习目标域中的预测函数 $f(\cdot)$ 。然而为了更好地学习 $f(\cdot)$ ，需要缩小源域 $\mathcal{D}_s$ 和目标域 $\mathcal{D}_t$ 的分布差异，于是本文引入传统的度量准则——最大均值差异 MMD。

$$MMD_{\mathcal{H}}(P_s, P_t) = \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_i^s) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_j^t) \right\|_{\mathcal{H}}^2 \quad (4-1)$$

上式中， $P_s$ 表示源域的概率分布， $P_t$ 表示目标域的概率分布， $n_s$ 和 $n_t$ 分别表示源域和目标域的样本数量， $\phi: X \rightarrow H$ 表示再生希尔伯特空间中的非线性映射函数。

在计算完源域和目标域特征之间的 MMD 距离后，将其加入到损失函数中，参与网络反向传播的训练中，这样便可以在训练的过程中减小目标域和源域之间的距离，也就是减小目标域和源域分布之间的差异，从而实现域适应（如下图 4-2 所示）。

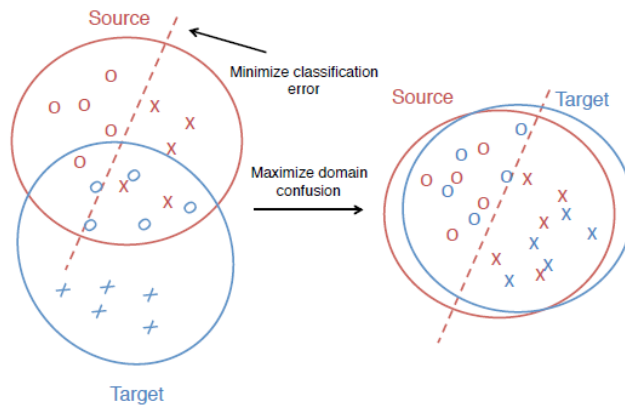


图 4-2 域自适应示意图

#### 4.1.2 真实标签和伪标签

由于源域和目标域之间的数据分布存在差异，因此在网络训练时需要同时使用源域和目标域的数据。

对于源域中的数据,其含有标签,选择采用交叉熵损失函数,来最小化该类样本的分类误差,表达式如下:

$$\mathcal{L}_{ce}(\theta) = -\frac{1}{m} \sum_{i=1}^m \sum_j^c 1(y_i = j) \log(P(y_i = j|x_i, \theta)) \quad (4-2)$$

$$\mathcal{L}_{ce}(\theta) = -\frac{1}{m} \sum_{i=1}^m \sum_j^c 1(y_i = j) \log \frac{e^{\theta_j^T x_i}}{\sum_{l=1}^c e^{\theta_l^T x_i}} \quad (4-3)$$

上式中,  $m$  为样本数量,  $P(y_i = j|x_i, \theta)$  表示样本  $i$  经过神经网络后得到的输出,  $y_i$  表示样本  $i$  的真实标签为第  $j$  类故障类型,。

对于目标域中的数据,其不含标签,因此需要首先对其进行伪标签的学习和训练。将目标域的数据输入神经网络,最后通过 softmax 分类器输出不同类别的概率值:

$$P(y_i|x_i) = \frac{e^{\theta_j^T x_i}}{\sum_{l=1}^c e^{\theta_l^T x_i}} \quad (4-4)$$

上式中,  $P(y_i|x_i)$  表示样本  $i$  属于故障类别  $j$  的预测概率值。对于特定的输入  $x_i$ , 其输出可以表示为:

$$y_i = [P(y_i = 0|x_i), P(y_i = 1|x_i), \dots, P(y_i = 9|x_i)] \quad (4-5)$$

根据上式,可以得到该输入的伪标签:

$$\hat{y}_i = [\hat{y}_0, \hat{y}_1, \dots, \hat{y}_j, \dots, \hat{y}_9] \quad (4-6)$$

$$\hat{y}_j = \begin{cases} 1, j = \arg \max_j \hat{y}_i \\ 0, otherwise \end{cases}$$

其中,  $\hat{y}_i$  表示样本  $i$  的伪标签。尽管在网络训练初期,该伪标签可能与真实标签相差较大,但随着网络训练的迭代进行,伪标签也会随之迭代更新,以获得当前学习条件下的最优预测精度。

伪标签的损失函数也可以用交叉熵损失函数来计算:

$$\mathcal{L}_{Pseudo}(\theta) = -\frac{1}{m} \sum_{i=1}^m \sum_j^c 1(\hat{y}_i = j) \log \frac{e^{\theta_j^T x_i}}{\sum_{l=1}^c e^{\theta_l^T x_i}} \quad (4-7)$$

#### 4.1.3 模型设计

针对工况条件不同的轴承故障诊断问题,本文所设计的模型的结构示意图如下图 4-3 所示:

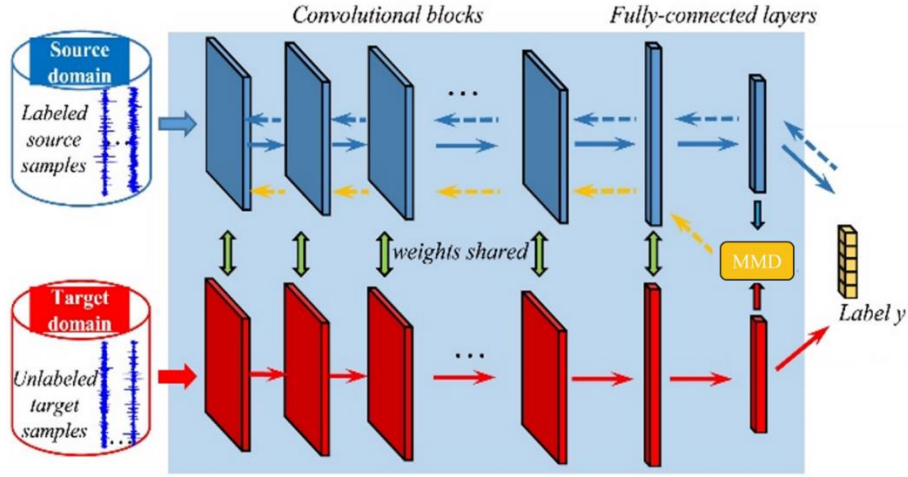


图 4-3 基于 MDA 的轴承故障诊断模型框架

本文所设计的基于 MDA 的轴承故障诊断模型框架如上图 4-3 所示，采用域共享的卷积神经网络 ResNet18 网络提取源域数据和目标域数据的信号特征。也就是说，卷积层和全连接层的结构和权重在源域和目标域中保持一致。源域和部分目标域的数据一同参与模型的学习和训练过程中，该模型的损失函数可以表示为：

$$\mathcal{L}(\theta) = \mathcal{L}_{ce}(\theta) + \lambda_1 \mathcal{L}_{pseudo}(\theta) + \lambda_2 MMD_{\mathcal{H}}^2(P_s, P_t) \quad (4-8)$$

其中， $\lambda_1$  和  $\lambda_2$  均为权衡系数。模型通过前向传播，可以计算出式(4-8)中的三项损失函数值，即传统的交叉熵损失函数、目标域数据伪标签的交叉熵损失函数以及 MDA 的正则化项。在反向传播过程中，模型利用 Adam 算法进行网络优化。

正则化项  $\lambda_2 MMD_{\mathcal{H}}^2(P_s, P_t)$  中的映射函数  $\phi: X \rightarrow H$  的选择会在一定程度上影响模型网络参数的训练结果。因此，本文选择不人为设置函数  $\phi$ ，选择将其与网络模型的上一层相结合，即

$$\phi(x) = H_{i-1}(x) \quad (4-9)$$

上式中， $H_{i-1}(x)$  表示网络在第  $i-1$  层经过非线性特征变换（激活函数）后得到的特征图。通过这种方法，便可以从数据中自适应地学习映射函数  $\phi$ ，以避免手动设置参数化的内核函数。

## 4.2 基于 ResNet 网络的 JDA

基于 DDC 的轴承故障诊断方法固然能够减小目标域和源域之间边际分布的差异，但是有时候对于不同的领域而言，其判别的超平面可能都会有所不同（如下图 4-4 所示）。因此，对于源域和目标域中数据分布差异较大的情况下，我们不仅需要

对齐源域和目标域的分布中心, 同时还需要匹配二者的区别结构, 从而完成迁移学习的任务。

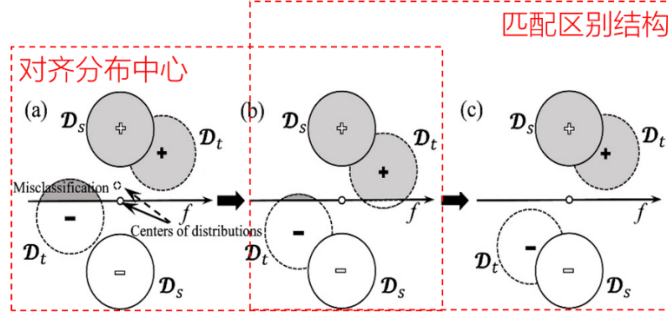


图 4-4 边际分布和条件分布自适应

#### 4.2.1 JDA 正则化项

联合分布自适应方法(Joint Distribution Adaptation, JDA)的目标是减小源域和目标域的联合概率分布的距离, 从而完成迁移学习。该距离应该同时最小化边际分布和条件分布之间的差异。边际分布差异本文用 MMD 距离来代替, 而条件分布差异可以表示为:

$$\min D(Q_s(Y_s|\phi(X_s)), Q_t(Y_t|\phi(X_t))) \quad (4-11)$$

在没有分类依据的情况下, 上式是很难处理的。我们将其重写为:

$$\min D\left(\frac{Q_s(\phi(X_s)|Y_s) \cdot P_s(Y_s)}{P_s(\phi(X_s))}, \frac{Q_t(\phi(X_t)|Y_t) \cdot P_t(Y_t)}{P_t(\phi(X_t))}\right) \quad (4-12)$$

而当边际分布最小化成立, 且假定目标域和源域之间具有相同的分布, 可以将上式简化为:

$$\min D(Q_s(\phi(X_s)|Y_s), Q_t(\phi(X_t)|Y_t)) \quad (4-13)$$

对于上式, 我们可以利用目标数据的伪标签来处理该问题。借助于带有标签的源域的预训练模型, 可以预先提供目标数据的伪标签。距离指标 MMD 可以定义为测量类别  $c(c \in \{0, 1, \dots, 9\})$  的条件分布  $Q_s(x_s|y_s = c)$  和  $Q_t(x_t|y_t = c)$  之间的不匹配程度:

$$MMD_{\mathcal{H}}(Q_s^{(c)}, Q_t^{(c)}) = \left\| \frac{1}{n_s^{(c)}} \sum_{x_i^s \in \mathcal{D}_s^{(c)}} \phi(x_i^s) - \frac{1}{n_t^{(c)}} \sum_{x_j^t \in \mathcal{D}_t^{(c)}} \phi(x_j^t) \right\|_{\mathcal{H}}^2 \quad (4-14)$$

其中,  $\mathcal{D}_s^{(c)} = \{x_i: x_i \in \mathcal{D}_s \wedge y(x_i) = c\}$ ,  $y(x_i)$  是真实标签, 并且  $n_s^{(c)} = |\mathcal{D}_s^{(c)}|$ 。

$\mathcal{D}_t^{(c)} = \{x_j: x_j \in \mathcal{D}_t \wedge y(x_j) = c\}$ ,  $y(x_i)$  是伪标签, 并且  $n_t^{(c)} = |\mathcal{D}_t^{(c)}|$ 。

因此, 通过整合上述两项分布差异, 基于 JDA 的轴承故障诊断方法网络中的

JDA 正则化项可以写为:

$$D_{\mathcal{K}}(J_s, J_t) = MMD_{\mathcal{K}}^2(P_s, P_t) + \sum_{c=1}^C MMD_{\mathcal{K}}^2(Q_s^{(c)}, Q_t^{(c)}) \quad (4-15)$$

#### 4.2.2 模型设计

针对工况条件不同的轴承故障诊断问题, 本文所设计的模型的结构示意图如下图 4-5 所示:

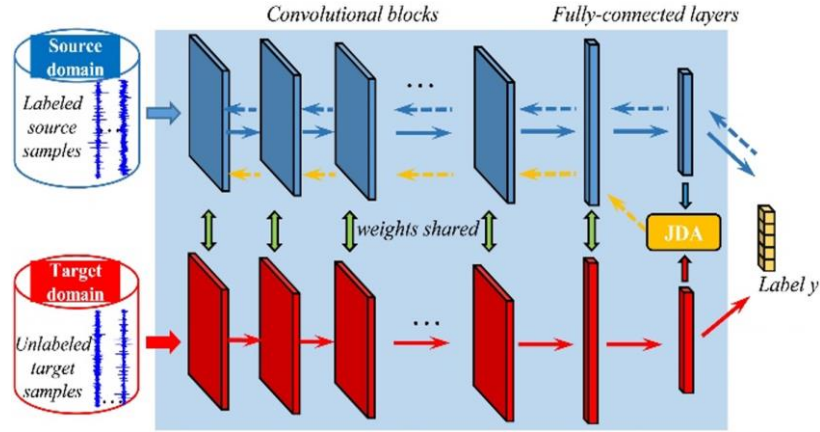


图 4-5 基于 JDA 的轴承故障诊断模型框架

该网络模型的目标函数可以表示为:

$$\mathcal{L}(\theta) = \mathcal{L}_{ce}(\theta) + \lambda_1 \mathcal{L}_{Pseudo}(\theta) + \lambda_2 D_{\mathcal{K}}(J_s, J_t) \quad (4-16)$$

目标函数相对于网络参数的梯度为:

$$\nabla_{\theta^l} = \frac{\partial \mathcal{L}_{ce}(\theta)}{\partial \theta^l} + \lambda_1 \frac{\partial \mathcal{L}_{Pseudo}(\theta)}{\partial \theta^l} + \lambda_2 (\nabla D_{\mathcal{K}}(J_s, J_t))^T \left( \frac{\partial \phi(x)}{\partial \theta^l} \right) \quad (4-17)$$

其中  $\frac{\partial \phi(x)}{\partial \theta^l}$  是含网络参数的第  $l-1$  层输出的偏导。 $\nabla D_{\mathcal{K}}(J_s, J_t)$  的详细方程如下所示:

$$\nabla D_{\mathcal{K}}(J_s, J_t) = \nabla MMD_{\mathcal{K}}^2(P_s, P_t) + \sum_{c=1}^C \nabla MMD_{\mathcal{K}}^2(Q_s^{(c)}, Q_t^{(c)}) \quad (4-18)$$

其中,  $\nabla MMD_{\mathcal{K}}^2(P_s, P_t)$  和  $\nabla MMD_{\mathcal{K}}^2(Q_s^{(c)}, Q_t^{(c)})$  可表示为:

$$\nabla MMD_{\mathcal{K}}^2(P_s, P_t) = \begin{cases} \frac{2}{n_s} \left( \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_i^s) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_j^t) \right), x \in \mathcal{D}_s \\ \frac{2}{n_t} \left( \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_j^t) - \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_i^s) \right), x \in \mathcal{D}_t \end{cases} \quad (4-19)$$

$$\nabla MMD_{\mathcal{K}}^2(Q_s^{(c)}, Q_t^{(c)}) = \begin{cases} \frac{2}{n_s^{(c)}} \left( \frac{1}{n_s^{(c)}} \sum_{x_i^s \in \mathcal{D}_s^{(c)}} \phi(x_i^s) - \frac{1}{n_t^{(c)}} \sum_{x_j^t \in \mathcal{D}_t^{(c)}} \phi(x_j^t) \right), x \in \mathcal{D}_s \\ \frac{2}{n_t^{(c)}} \left( \frac{1}{n_t^{(c)}} \sum_{x_j^t \in \mathcal{D}_t^{(c)}} \phi(x_j^t) - \frac{1}{n_s^{(c)}} \sum_{x_i^s \in \mathcal{D}_s^{(c)}} \phi(x_i^s) \right), x \in \mathcal{D}_t \end{cases} \quad (4-20)$$

## 4.3 实验验证

在这一小节中,本文基于上述设计的MDA和JDA的轴承故障诊断模型,通过设置多组迁移学习任务来比较两种模型的效果性能。

### 4.3.1 迁移任务

由于在这一章节我们考虑的是目标域数据不含标签的无监督任务,因此我们需要在模型的训练过程中让含有标签的源域数据和不含标签的目标域数据对网络进行网络适应。然而,为了保证网络适应的稳定,源域的样本数量应该和目标域的样本数量保持一致。如果跨域的数据大小不同,网络适应可能难以进行。

本文将90%的目标域数据参与到迁移网络的训练过程中,并将10%的目标域数据进行网络性能的测试。(如下图4-6所示)。

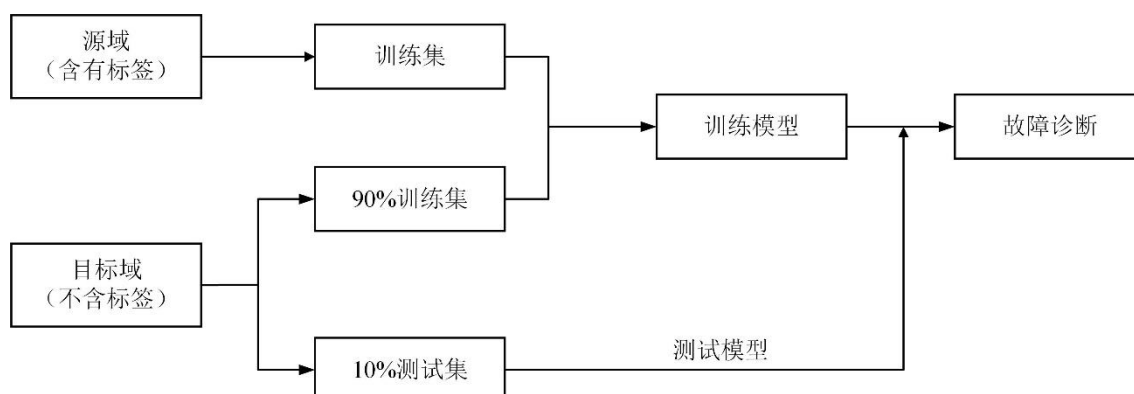


图 4-6 迁移任务示意图

同时,为了验证所设计的算法的优越性,本文设置了多组任务来进行实验验证,各项任务如下表4-1所示:

表 4-1 迁移任务的数据集

数据集	
A	1 马力、采样频率为 12kHz, 传感器位于驱动端
B	0 马力、采样频率为 12kHz, 传感器位于驱动端
C	1 马力、采样频率为 12kHz, 传感器位于风扇端
D	0 马力、采样频率为 12kHz, 传感器位于风扇端

上表4-1中,以轴承数据集A为例,数据集A表示选择负载情况为1马力、采样频率为12kHz的情况下驱动端轴承的情况。

在此基础上,迁移任务 $A \rightarrow B$ 的含义表示为从源域数据集A到目标域数据集B



的迁移任务。为了评估模型性能，我们共设计了六组迁移学习任务：任务 $A \rightarrow B$ 、任务 $B \rightarrow A$ 、任务 $C \rightarrow B$ 、任务 $B \rightarrow C$ 、任务 $D \rightarrow B$ 以及任务 $B \rightarrow D$ 。在各项迁移任务中，目标域中参与模型训练的数据默认不含有标签，参与测试的数据含有标签，验证模型的准确率。

#### 4.3.2 诊断流程

本章节中所设计的两种方法——基于 MDA 的轴承故障诊断模型和基于 JDA 的轴承故障诊断模型的差别仅在于其模型网络中的损失函数项表达式不同，如式 4-8 和式 4-16 所示：

$$\mathcal{L}(\theta) = \mathcal{L}_{ce}(\theta) + \lambda_1 \mathcal{L}_{pseudo}(\theta) + \lambda_2 MMD_{\mathcal{H}}^2(P_s, P_t) \quad (4-8)$$

$$\mathcal{L}(\theta) = \mathcal{L}_{ce}(\theta) + \lambda_1 \mathcal{L}_{pseudo}(\theta) + \lambda_2 D_{\mathcal{K}}(J_s, J_t) \quad (4-16)$$

因此，本文这里仅以基于 MDA 的轴承故障诊断模型为例展开诊断流程的介绍。

该模型的诊断流程主要包括两个步骤：

- (1) 根据含有标签的样本数据预训练网络模型；
- (2) 通过输入含有标签的源域数据以及不含标签的目标域数据对目标域进行网络域自适应，完成迁移学习的任务。

在 3.4 章节中，本文已经完成了对基于 ResNet18 网络和微调方法的轴承故障诊断模型的训练，于是在这一章节本文直接选择采用该网络模型的全部参数作为基于 MDA 的轴承故障诊断模型网络的初始参数。除此之外，网络模型的其他超参数如下表 4-2 所示：

表 4-2 基于 MDA 的模型训练参数表

参数名称	参数值
epoch	50
batch size	64
learning rate	0.001
优化器	Adam 优化器
$\lambda_1$	0.5
$\lambda_2$	0.5

上式中， $\lambda_1$ 和 $\lambda_2$ 均为权衡系数。基于上述超参数，本文设计并构建了基于 MDA



的轴承故障诊断模型，该模型诊断流程图如下图所示：

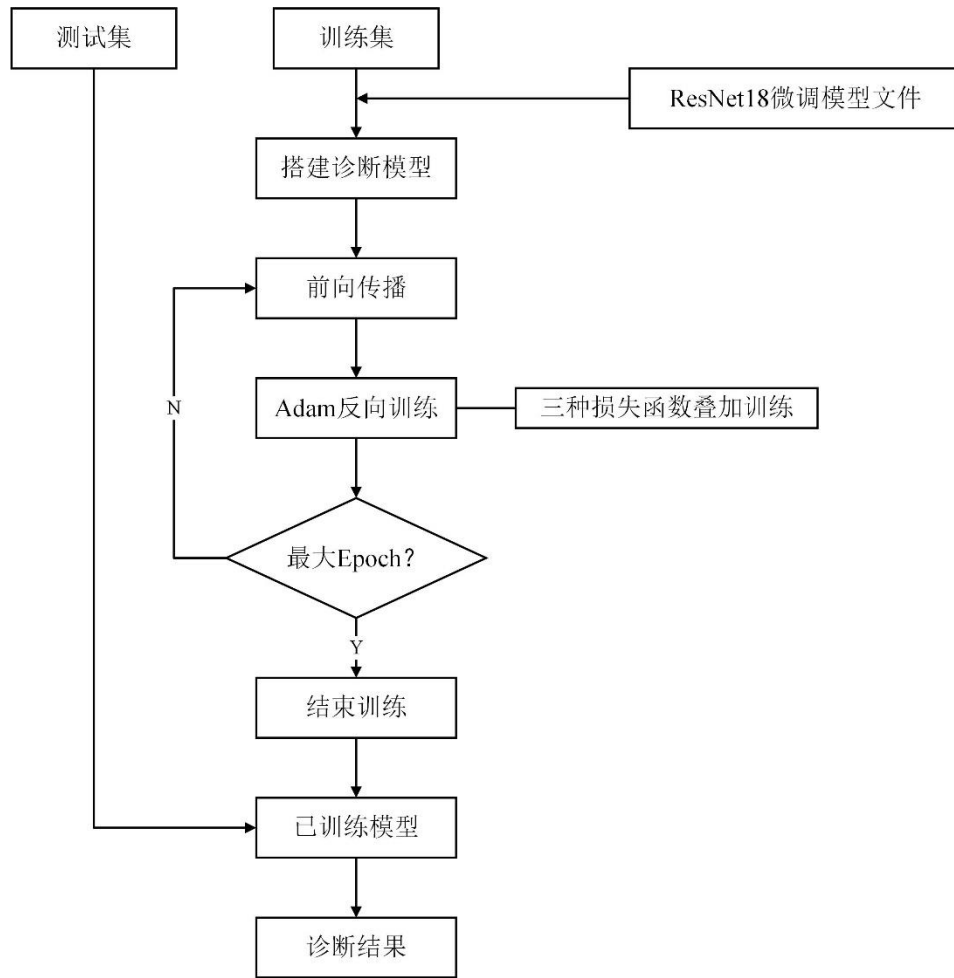


图 4-7 基于 MDA 的模型诊断流程图

一般情况下，模型在进行预训练的过程中所采用的数据集为迁移任务中的源域数据集，但本文考虑到尽管不同数据集收集于不同的负载条件和不同的传感器位置，但其仍是来自于相同的实验台，因此本文直接将第 3 章中基于负载情况为 0 马力、采样频率为 12 kHz 的情况下驱动端轴承的数据训练得到的 Finetune 故障诊断模型作为迁移任务的预训练模型，直接在该模型的基础上进行网络域自适应。

本文所采用的基于 MDA 的轴承故障诊断模型的求解流程如下表 4-3 所示，基于 JDA 的轴承故障诊断模型的求解流程也是相同的。

表 4-3 基于改进的模型求解流程

算法 3：基于 Finetune 的诊断算法

输入：迭代次数 epoch(= 50)

输入：小批量大小 batch size(= 64)

算法 3: 基于 Finetune 的诊断算法

输入: 学习率 learning rate( $\alpha = 0.001$ )

输入: 权衡系数 $\lambda_1 = \lambda_2 = 0.5$

初始化: 导入基于 Finetune 的诊断模型的文件

前向传播: 为不含标签的数据生成伪标签 $\hat{y}^{(i)}$

for step in epoch:

if step != 1:

前向传播: 随着迭代的进行完成伪标签 $\hat{y}^{(i)}$ 的更新

损失计算:  $\mathcal{L}(\theta) = \mathcal{L}_{ce}(\theta) + \lambda_1 \mathcal{L}_{Pseudo}(\theta) + \lambda_2 MMD_{\mathcal{H}}^2(P_s, P_t)$

反向传播: 计算梯度 $\mathbf{g} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L$

梯度更新:  $\theta \leftarrow \theta - \alpha \frac{\partial L}{\partial \theta}$

step  $\leftarrow$  step +1

end

测试集上进行模型测试

### 4.3.3 诊断结果

#### 4.3.3.1 MDA 模型的诊断结果

基于上述超参数设置和诊断流程, 本文在 TensorFlow2.0 环境下搭建网络模型, 并对网络模型进行学习和训练, 训练结果如下表 4-4 所示:

表 4-4 基于 MDA 的模型训练结果

跨域任务	实验 1	实验 2	实验 3	实验 4	实验 5
$A \rightarrow B$	96.2%	96.7%	95.6%	95.9%	94.3%
$B \rightarrow A$	92.5%	91.1%	91.7%	92.5%	91.4%
$C \rightarrow B$	90.7%	90.7%	89.4%	90.3%	91.5%
$B \rightarrow C$	86.8%	87.4%	88.0%	87.1%	86.3%
$D \rightarrow B$	91.8%	91.5%	92.0%	91.1%	92.6%
$B \rightarrow D$	88.6%	89.3%	89.7%	89.0%	87.7%

根据上表 4-4 的结果, 我们可以知道: 任务 $A \rightarrow B$ 和任务 $B \rightarrow A$ 的诊断准确率

明显高于任务 $C \rightarrow B$ 、任务 $B \rightarrow C$ 、任务 $D \rightarrow B$ 以及任务 $B \rightarrow D$ ，这就说明在条件分布差异不大时，该迁移模型具有良好的迁移效果，且能够很快地达到收敛的效果，大大降低了模型的运算时间。但当条件分布差异较大时，该迁移模型虽然能够很快收敛，但很明显权重和偏置的更新陷入局部最优解，模型的诊断效果不佳。

跨域任务 $A \rightarrow B$ 明显较跨域任务 $B \rightarrow A$ 表现良好，这是因为：模型网络的初始参数是在任务 $B$ 中基于 ResNet 网络进行微调训练得来的，其在重新开始训练时已经初步具有较优的预测性能。对于目标域 $B$ 中不含标签的数据，该网络在训练初期便能表现出较准确的伪标签预测能力。本文引入召回率作为评价指标。召回率是指在原本样本总量中，每类标签对应的样本中被预测正确的样本数量所占比例：

$$P_j = \frac{TP_j}{TP_j + FN_j} \quad (4-21)$$

其中， $P_j$ 表示故障类别为 $j$ 的样本的召回率， $TP_j$ 表示故障类别为 $j$ 的样本被预测为故障类别 $j$ 的数量， $FN_j$ 则表示故障类别为 $j$ 的样本未被预测为故障类别 $j$ 的数量。

在经过 50 次迭代后，得到跨域任务 $A \rightarrow B$ 以及 $B \rightarrow A$ 的混淆矩阵（如下图 4-8 和图 4-9 所示），可以看出，跨域任务 $A \rightarrow B$ 模型对各种故障类别的预测准确率以及召回率均高于跨域任务 $B \rightarrow A$ 模型。

Ball-7	90	1	20	0	0	7	0	0	0	0	0.76
Ball-14	8	99	34	0	0	0	0	0	0	0	0.7
Ball-21	0	0	46	0	0	0	0	0	0	0	1
IR-7	0	0	0	100	0	0	0	8	0	0	0.93
IR-14	0	0	0	0	100	6	0	0	0	0	0.94
IR-21	0	0	0	0	0	87	0	0	0	0	1
OR-7	1	0	0	0	0	0	100	0	0	0	0.99
OR-14	0	0	0	0	0	0	0	90	0	0	1
OR-21	0	0	0	0	0	0	0	2	100	0	0.98
Health	1	0	0	0	0	0	0	0	0	100	0.99
Recall	0.9	0.99	0.46	1	1	0.87	1	0.9	1	1	
	Ball-7	Ball-14	Ball-21	IR-7	IR-14	IR-21	OR-7	OR-14	OR-21	Health	Acc

图 4-8 跨域任务 $A \rightarrow B$ 的混淆矩阵

Ball-7	92	1	12	0	0	12	0	0	0	0	0.79
Ball-14	7	99	22	0	0	0	0	0	0	0	0.77
Ball-21	0	0	47	0	0	0	0	0	0	0	1
IR-7	0	0	0	98	0	0	0	0	1	0	0.99
IR-14	0	0	0	0	99	15	0	0	0	0	0.87
IR-21	0	0	0	0	0	73	0	6	0	0	0.92
OR-7	0	0	19	0	1	0	100	0	0	0	0.83
OR-14	0	0	0	0	0	0	0	93	0	0	1
OR-21	0	0	0	2	0	0	0	1	99	0	0.97
Health	1	0	0	0	0	0	0	0	0	100	0.99
Recall	0.92	0.99	0.47	0.98	0.99	0.73	1	0.93	0.99	1	
	Ball-7	Ball-14	Ball-21	IR-7	IR-14	IR-21	OR-7	OR-14	OR-21	Health	Acc

图 4-9 跨域任务 $B \rightarrow A$ 的混淆矩阵

#### 4.3.3.2 JDA 模型的诊断结果

本文所构建的基于 JDA 的轴承故障诊断模型的训练结果如下表 4-5 所示：

表 4-5 基于 JDA 的模型训练结果

跨域任务	实验 1	实验 2	实验 3	实验 4	实验 5
$A \rightarrow B$	96.4%	96.9%	97.0%	96.7%	97.2%
$B \rightarrow A$	92.5%	93.1%	93.6%	93.3%	93.0%
$C \rightarrow B$	95.2%	95.3%	96.6%	95.1%	94.8%
$B \rightarrow C$	93.1%	92.7%	92.8%	91.5%	92.6%
$D \rightarrow B$	96.1%	95.7%	95.9%	95.7%	96.3%
$B \rightarrow D$	93.6%	94.1%	94.2%	93.9%	93.8%

根据上表 4-5 的结果，我们可以知道：基于 JDA 的轴承故障诊断模型的诊断效果均表现良好，无论是条件分布差异不大的任务 $A \rightarrow B$ 和任务 $B \rightarrow A$ ，还是条件分布差异较大的任务 $C \rightarrow B$ 、任务 $B \rightarrow C$ 、任务 $D \rightarrow B$ 以及任务 $B \rightarrow D$ 。该模型在这六组迁移任务中均表现出较高的诊断准确率，同时模型也能较快地完成收敛。

同时，为了衡量模型对于伪标签的生成能力，本文仍然考虑跨域任务 $C \rightarrow B$ 以及任务 $B \rightarrow C$ 的混淆矩阵，如下图 4-10 和图 4-11 所示：

Ball-7	99	0	1	0	0	0	0	0	0	0	0.99
Ball-14	0	97	0	0	0	0	0	0	0	0	1
Ball-21	1	0	96	0	0	0	0	0	0	0	0.99
IR-7	0	0	0	99	0	0	0	0	0	0	1
IR-14	0	0	0	0	100	1	0	0	0	0	0.99
IR-21	0	0	2	0	0	99	0	0	0	0	0.98
OR-7	0	0	0	0	0	0	100	0	0	0	1
OR-14	0	1	0	1	0	0	0	100	0	0	0.98
OR-21	0	0	0	0	0	0	0	0	100	0	1
Health	0	2	1	0	0	0	0	0	0	100	0.97
Recall	0.99	0.97	0.96	0.99	1	0.99	1	1	1	1	
	Ball-7	Ball-14	Ball-21	IR-7	IR-14	IR-21	OR-7	OR-14	OR-21	Health	Acc

图 4-10 跨域任务  $C \rightarrow B$  的混淆矩阵

Ball-7	66	0	19	0	0	0	0	0	0	0	0.99
Ball-14	22	96	0	0	0	0	0	0	0	0	1
Ball-21	1	0	62	0	0	0	0	0	0	0	0.99
IR-7	0	0	0	99	2	0	0	0	0	0	1
IR-14	0	0	0	0	98	31	0	0	0	0	0.99
IR-21	0	0	19	0	0	57	0	0	0	0	0.98
OR-7	0	0	0	0	0	0	100	0	0	0	1
OR-14	11	2	0	1	0	12	0	94	0	0	0.98
OR-21	0	0	0	0	0	0	0	6	100	0	1
Health	0	2	0	0	0	0	0	0	0	100	0.97
Recall	0.99	0.97	0.96	0.99	1	0.99	1	1	1	1	
	Ball-7	Ball-14	Ball-21	IR-7	IR-14	IR-21	OR-7	OR-14	OR-21	Health	Acc

图 4-11 跨域任务  $B \rightarrow C$  的混淆矩阵

同样可以看出,跨域任务 $A \rightarrow B$ 模型对各种故障类别的预测准确率以及召回率均高于跨域任务 $B \rightarrow A$ 模型。且整体而言,基于 JDA 的诊断模型对各种故障类别的预测准确率以及召回率相较于基于 MDA 的诊断模型均表现得更优。

#### 4.3.3.3 模型对比

为了验证本文所设计的两种算法的优越性,本文将基于 MDA 和 JDA 的轴承故障诊断结果与传统的深度神经网络和卷积神经网络进行对比,对比结果如下表 4-6 所示:

表 4-6 多种模型效果对比

跨域任务	MDA	JDA	DNN	CNN
$A \rightarrow B$	95.74%	96.84%	82.16%	91.83%
$B \rightarrow A$	91.84%	93.10%	81.73%	86.84%
$C \rightarrow B$	90.52%	95.40%	79.55%	88.75%
$B \rightarrow C$	87.12%	92.54%	72.13%	84.29%
$D \rightarrow B$	91.80%	95.94%	80.40%	84.73%
$B \rightarrow D$	88.86%	93.92%	64.51%	82.52%

由上表 4-6 以及图 4-12,我们可以得知:在四种模型的对比中,基于 JDA 的轴承故障诊断模型的准确率稳定保持在最高的水平,效果相较于 MDA、DNN 和 CNN 均表现良好。

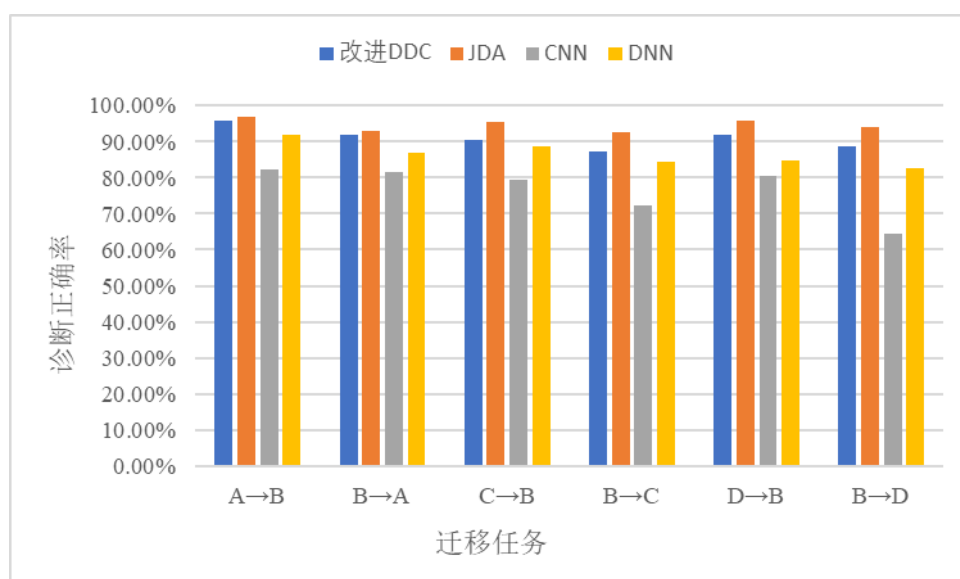


图 4-12 多种模型效果对比

## 4.4 本章小结

本章在源域数据含有标签而目标域的数据不含标签的基础上,继续采用深度迁移学习的方法建立并训练轴承故障诊断模型。

当源域和目标域的数据分布相似时,我们只考虑适配二者的边缘分布差异,以最大均值差异 MMD 作为自适应度量加入到网络损失函数的训练中,建立并训练基于 MDA 的轴承故障诊断模型。

当源域和目标域的数据分布差异较大时,倘若仅对其二者的分配中心而不匹配区别结构时,源域和目标域的判别超平面不同,会严重影响到诊断准确率。因此在这种情况下,本文不仅考虑了二者的边缘分布差异,还考虑了二者的条件分布差异,用 MMD 度量源域和目标域的条件分布的不匹配程度,建立并训练基于 JDA 的轴承故障诊断模型。



## 5 总结与展望

### 5.1 全文总结

本文的主要研究内容是轴承故障诊断的方法研究,通过深度迁移学习网络的知识构建相关模型,对于含有标签的数据以及不含标签的数据分别进行诊断识别,期望能够在保证诊断准确率的前提下大大减少模型求解的运算时间,并增加模型的鲁棒性。

首先,本文介绍了全球通用轴承数据集——CWRU 轴承数据集,并根据短时傅里叶变换将振动信号转换为二维的时频图,方便后续模型网络的输入。

接着,本文通过对 ResNet18 网络的微调,进行有标签的轴承数据的故障诊断。由于网络的可迁移性,本文在已经训练好的 ResNet18 网络模型中,将全连接层最后一层的节点数改为 10 个节点,并初始化该层权重和偏置。对于卷积层,将参数学习率设置为 0.001;对于全连接层,将参数学习率设置为 0.01。采用 Adam 优化器对该模型进行训练求解,最后在测试集上表现良好,诊断准确率保持在 99.50% 以上。

由于实际工作情况下很难收集完全含有标签的数据集,因此本文引用 DDC 方法,结合最大均值差异 MMD,进行无标签的轴承数据的故障诊断。本文将通过在全连接层最后一层之前设置自适应度量标准,减小目标域和源域数据的边缘分布差异,并将两领域之间最大均值差异 MMD 引入损失函数中,参与网络反向传播的训练。为了将无标签的数据也参与到网络的训练中,本文将这些数据输入网络,通过前向传播进行标签预测,并采用交叉熵损失函数计算预测标签的损失值,参与反向训练,不断优化伪标签,最终伪标签的预测结果会与真实标签高度重合。通过对基于 DDC 的模型的求解,本文得出结论:该模型下,将在负载条件为 0hp 下建立的模型迁移至其他条件不变,负载条件为 1hp 模型时,通过 MMD 度量标准,诊断结果准确率保持在 94.00% 以上。

引入 MMD 度量准则只能匹配源域和目标域的边缘分布差异。当源域和目标域的结构差异较大时,模型的迁移效果会显著降低。为了解决这一问题,本文提出了基于 JDA 的轴承故障诊断模型,不仅考虑两领域之间的边缘分布差异,同时还

考虑两领域之间的条件分布差异。具体方法是改变自适应度量准则,并改变反向传播训练中的损失函数。通过对基于 JDA 的模型的求解,本文得出结论:该模型下,对于来自不同位置传感器收集而来的不同负载条件的两类数据,模型迁移的效果良好,在测试集的诊断准确率保持在 94.5%左右。

本文实现了对轴承故障诊断方法的算法研究,可以得到如下结论:

(1) 对于含有标签的轴承振动信号数据,本文采用 Finetune 的方法,对已经训练好的模型进行微调,在保证较好的诊断效果的同时大大减少模型训练的时间。

(2) 对于不含标签的轴承振动信号数据,但有与其分布相似的其他数据集时,本文采用基于域自适应的方法,采用自适应度量准则 MMD,进行源域和目标域的网络训练。

(3) 对于不含标签的轴承振动信号数据,且无与其分布相似的其他数据集时,这时进行模型迁移时不仅需要考虑边缘分布差异,还需要考虑条件分布差异。本文采用联合分配适应的方法,进行源域和目标域的网络训练。

## 5.2 工作展望

在本文的工作基础上,还有许多可以改进的地方,具体如下几点:

(1) 对于含有多种故障类型的信号数据,本文所采用的网络模型无法完全识别出来,且识别单一故障的效果可能不佳;

(2) 在实际操作过程中,倘若出现其他部位的故障类型(非本文选择的十种故障类型),模型的诊断效果可能会受到影响;

(3) 对于基于 DDC 和 JDA 的轴承故障诊断方法,本文训练后的模型诊断准确率相较于成熟的算法的准确率还有一定的差距。

在以后的工作中,可以有以下几点展望:

(1) 通过引入多核 MMD 度量(MK-MMD)代替 DDC 方法中的单一核,适配网络的多层结构从而提高网络模型的诊断效果;

(2) 尝试将本文设计的基于深度迁移学习的轴承故障诊断模型,应用在齿轮等零件的故障诊断中;

(3) 可以尝试结合长短期记忆网络等方法,实时对运转中收集到的轴承振动信号进行分析,从而起到对故障的预测作用。

## 致谢

时光匆匆,眨眼大学生活悄然过去。在这四年间,我在华中科技大学这所美丽的学校收获了很多,特借此机会对所有帮助过我的老师朋友致以谢意。

首先,我要感谢我的毕设指导老师——高亮老师。从选题的开始,到主要内容的确定,再到最终论文的定稿,高老师悉心的指导和解疑总能拨云见日,让我看清前行的方向。高老师身上严谨的科研态度和对学术的真诚与热忱之心也潜移默化地影响着我。在今后的学习中,我一定会认真对待科研,争取向老师看齐。

同时,我还要感谢文龙老师长期以来对我的指导和帮助。倘若没有文龙老师的帮助,我不可能以这样的水平完成本次毕业设计的工作。文龙老师细心的指导和严格的要求一直推动着我,使我变得更加优秀。

此外,我要感谢大学四年所有指导过我的各位老师。每位老师都身体力行地用自己的方式让我们领悟到知识之美、科研之美。在他们身上,我感受到了华中大老师对学生的无私关怀以及对自己的严格要求。

感谢两位已经毕业了学长——赵岳虎和胡恒一。自相识以来,赵岳虎学长身上的淡然气质和处万变而不惊的能力,胡恒一学长对知识的渴求和对生活的向往,一直都深深地吸引着我,改变着我。我很庆幸疫情和升学并没有将我们之间的距离拉远,也很感激能在人生这一重要阶段结识两位如此优秀的 *soulmates*,愿我们的友谊不仅有今朝,更比万年长。

感谢两位电信学院的好友——夏元倩和王正林,我们三人从小学,到初中,到高中,再到如今大学毕业,求学十六载,与你们相识相交也有十六年。在这十六年里,我们一起分享喜悦,一起共克难关,一起在知识的海洋、人生的道路中相互鼓励,互帮互助,希望未来我们能够始终保持这份友谊,让友谊之花在我们之间常盛。

感谢我曾最好的知心朋友,在我处于低谷的时候陪伴着我,与我一起成长,祝往后学业有成,万事如意。感谢李舒雅和赵宇航两位好友,大大小小的各种比赛我们一起努力、并肩作战的时光我会永远铭记于心。感谢好友高宇良,虽相交不早,但和你一起讨论学习,分享日常,是我大学很珍贵的回忆。同时还要感谢机械学院所有的朋友同学,能在这段美好时光遇见你们,是我最大的幸运。往后若志同道合,愿我们平分秋色;若各奔前程,望我们各有千秋。

最后,我要将最诚挚的感激和敬意献给我的父母及家人。感谢你们尊重和支持我做的所有决定,感谢你们永远无私地为我提供温暖的港湾,感谢你们始终如明灯一般,以最温柔的光芒指引着我前行。

同时, 特别感谢百忙中抽出时间对论文评审的各位老师, 老师们辛苦了。

柯审强

2021 年 5 月

写于华中大韵苑学生公寓

## 参考文献

- [1] WEI Z, WANG Y, HE S, 等. A Novel Intelligent Method for Bearing Fault Diagnosis Based on Affinity Propagation Clustering and Adaptive Feature Selection[J]. Knowledge-Based Systems, 2017, 116: 1–12. DOI:10.1016/j.knosys.2016.10.022.
- [2] JANSSENS O, SLAVKOVÍK J V, VERVISCH B, 等. Convolutional Neural Network Based Fault Detection for Rotating Machinery[J]. Journal of Sound and Vibration, 2016, 377: 331–345. DOI:10.1016/j.jsv.2016.05.027.
- [3] 马萍, 张宏立, 范文慧. 基于局部与全局结构保持算法的滚动轴承故障诊断[J]. 机械工程学报, 2017, 53(02): 20–25.
- [4] 李晗, 萧德云. 基于数据驱动的故障诊断方法综述[J]. 控制与决策, 2011, 26(01): 1-9+16.
- [5] 周东华, 胡艳艳. 动态系统的故障诊断技术[J]. 自动化学报, 2009, 35(06): 748–758.
- [6] LI C, SANCHEZ R-V, ZURITA G, 等. Multimodal Deep Support Vector Classification with Homologous Features and Its Application to Gearbox Fault Diagnosis[J]. Neurocomputing, 2015, 168: 119–127. DOI:10.1016/j.neucom.2015.06.008.
- [7] LIU H, ZHOU J, XU Y, 等. Unsupervised Fault Diagnosis of Rolling Bearings Using a Deep Neural Network Based on Generative Adversarial Networks[J]. Neurocomputing, 2018, 315: 412–424. DOI:10.1016/j.neucom.2018.07.034.
- [8] LIA X, SUNC Q D J-Q. Multi-Layer Domain Adaptation Method for Rolling Bearing Fault Diagnosis[J]. [日期不详]. .
- [9] DAY O, KHOSHGOFTAAR T M. A survey on heterogeneous transfer learning[J]. Journal of Big Data, 2017, 4(1): 1–42.
- [10] IRASARI P. Experiment and Analysis of Car Alternator for Wind Turbine Application[J]. Journal of Mechatronics, Electrical Power, and Vehicular Technology, 2012, 2(1): 1–10.
- [11] 刘长良, 武英杰, 甄成刚. 基于变分模态分解和模糊 C 均值聚类的滚动轴承故障诊断[J]. 中国电机工程学报, 2015, 35(13): 3358–3365.
- [12] LEI Y, HE Z, ZI Y. EEMD method and WNN for fault diagnosis of locomotive roller bearings[J]. Expert Systems with Applications, 2011, 38(6): 7334–7341.
- [13] JIN X, ZHAO M, CHOW T W S, 等. Motor Bearing Fault Diagnosis Using Trace Ratio Linear Discriminant Analysis[J]. IEEE Transactions on Industrial Electronics, 2014, 61(5): 2441–2451. DOI:10.1109/TIE.2013.2273471.
- [14] HASAN M J, ISLAM M M M, KIM J-M. Acoustic Spectral Imaging and Transfer Learning for Reliable Bearing Fault Diagnosis under Variable Speed Conditions[J]. Measurement, 2019, 138: 620–631. DOI:10.1016/j.measurement.2019.02.075.
- [15] SOHAIB M, KIM C-H, KIM J-M. A Hybrid Feature Model and Deep-Learning-Based Bearing Fault Diagnosis: 12[J]. Sensors, 2017, 17(12): 2876. DOI:10.3390/s17122876.
- [16] RAI V K, MOHANTY A R. Bearing fault diagnosis using FFT of intrinsic mode functions in Hilbert–Huang transform[J]. Mechanical Systems and Signal Processing,

- 2007, 21(6): 2607–2615.
- [17] YU Y, JUNSHENG C. A roller bearing fault diagnosis method based on EMD energy entropy and ANN[J]. Journal of sound and vibration, 2006, 294(1–2): 269–277.
- [18] LOU X, LOPARO K A. Bearing Fault Diagnosis Based on Wavelet Transform and Fuzzy Inference[J]. Mechanical Systems and Signal Processing, 2004, 18(5): 1077–1095. DOI:10.1016/S0888-3270(03)00077-3.
- [19] SAMANTA B, AL-BALUSHI K R. Artificial neural network based fault diagnostics of rolling element bearings using time-domain features[J]. Mechanical systems and signal processing, 2003, 17(2): 317–328.
- [20] FERNÁNDEZ-FRANCOS D, MARTÍNEZ-REGO D, FONTENLA-ROMERO O, 等. Automatic Bearing Fault Diagnosis Based on One-Class v-SVM[J]. Computers & Industrial Engineering, 2013, 64(1): 357–365. DOI:10.1016/j.cie.2012.10.013.
- [21] SHI J, WU X, ZHOU J, 等. BP neural network based bearing fault diagnosis with differential evolution EEMD denoise[C]//2017 9th International Conference on Modelling, Identification and Control (ICMIC). . DOI:10.1109/ICMIC.2017.8321610.
- [22] DYER D, STEWART R M. Detection of rolling element bearing damage by statistical vibration analysis[J]. 1978. .
- [23] TANDON N, NAKRA B C. Detection of defects in rolling element bearings by vibration monitoring[J]. Indian Journal of Mechanical Engineering Division, 1993, 73: 271–282.
- [24] MCINERNY S A, DAI Y. Basic vibration signal processing for bearing fault detection[J]. IEEE Transactions on Education, 2003, 46(1): 149–156. DOI:10.1109/TE.2002.808234.
- [25] YIAKOPOULOS C, GRYLLIAS K, ANTONIADIS I. Spectral effects of the slip motion in defective rolling element bearings[J]. Advances in Vibration Engineering, 2005, 4: 199–211.
- [26] WADHWANI S, GUPTA S P, KUMAR V. Wavelet based vibration monitoring for detection of faults in ball bearings of rotating machines[J]. Journal of the Institution of Engineers (India): Electrical Engineering Division, 2005.
- [27] GERAMIFARD O, XU J-X, PANG C K, 等. Data-driven approaches in health condition monitoring—A comparative study[C]//IEEE ICCA 2010. IEEE, 2010: 1618–1622.
- [28] MIAO Q, WANG D, PECHT M. A probabilistic description scheme for rotating machinery health evaluation[J]. Journal of Mechanical Science and Technology, 2010, 24(12): 2421–2430.
- [29] PURARJOMANDLANGRUDI A, GHAPANCHI A H, ESMALIFALAK M. A data mining approach for fault diagnosis: An application of anomaly detection algorithm[J]. Measurement, 2014, 55: 343–352.
- [30] KANKAR P K, SHARMA S C, HARSHA S P. Fault diagnosis of ball bearings using machine learning methods[J]. Expert Systems with applications, 2011, 38(3): 1876–1886.
- [31] KATERIS D, MOSHOU D, PANTAZI X-E, 等. A machine learning approach for



- the condition monitoring of rotating machinery[J]. Journal of Mechanical Science and Technology, 2014, 28(1): 61–71.
- [32]TAMILSELVAN P, WANG P. Failure diagnosis using deep belief learning based health state classification[J]. Reliability Engineering & System Safety, 2013, 115: 124–135.
- [33]LEE D, SIU V, CRUZ R, 等. Convolutional neural net and bearing fault analysis[C]//Proceedings of the International Conference on Data Science (ICDATA). The Steering Committee of The World Congress in Computer Science, Computer ..., 2016: 194.
- [34]AHMED H O A, WONG M L D, NANDI A K. Intelligent condition monitoring method for bearing faults from highly compressed measurements using sparse over-complete features[J]. Mechanical Systems and Signal Processing, 2018, 99: 459–477.
- [35]LU C, WANG Z-Y, QIN W-L, 等. Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification[J]. Signal Processing, 2017, 130: 377–388.
- [36]GUO J, WU J, SUN Z, 等. Fault Diagnosis of Delta 3D Printers Using Transfer Support Vector Machine With Attitude Signals[J]. IEEE Access, 2019, 7: 40359–40368. DOI:10.1109/ACCESS.2019.2905264.
- [37]HAN T, LIU C, YANG W, 等. Deep Transfer Network with Joint Distribution Adaptation: A New Intelligent Fault Diagnosis Framework for Industry Application[J]. ISA Transactions, 2020, 97: 269–281. DOI:10.1016/j.isatra.2019.08.012.
- [38]LONG M, CAO Y, WANG J, 等. Learning transferable features with deep adaptation networks[C]. .
- [39]GANIN Y, USTINOVA E, AJAKAN H, 等. Domain-Adversarial Training of Neural Networks[M/OL]. CSURKA G, 编 //Domain Adaptation in Computer Vision Applications. Cham: Springer International Publishing, 2017: 189–209[2021–05–12]. [https://doi.org/10.1007/978-3-319-58347-1\\_10](https://doi.org/10.1007/978-3-319-58347-1_10). DOI:10.1007/978-3-319-58347-1\_10.
- [40]SHEN F, CHEN C, YAN R, 等. Bearing fault diagnosis based on SVD feature extraction and transfer learning classification[C]//2015 Prognostics and System Health Management Conference (PHM). IEEE, 2015: 1–6.
- [41]WEN L, GAO L, LI X. A new deep transfer learning based on sparse auto-encoder for fault diagnosis[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017, 49(1): 136–144.
- [42]CAO P, ZHANG S, TANG J. Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning[J]. Ieee Access, 2018, 6: 26241–26253.
- [43]HASAN M J, KIM J-M. Bearing fault diagnosis under variable rotational speeds using stockwell transform-based vibration imaging and transfer learning[J]. Applied Sciences, 2018, 8(12): 2357.
- [44]WANG Z, WANG J, WANG Y. An Intelligent Diagnosis Scheme Based on Generative Adversarial Learning Deep Neural Networks and Its Application to



- Planetary Gearbox Fault Pattern Recognition[J]. *Neurocomputing*, 2018, 310: 213–222. DOI:10.1016/j.neucom.2018.05.024.
- [45] LI X, ZHANG W, DING Q. Cross-Domain Fault Diagnosis of Rolling Element Bearings Using Deep Generative Neural Networks[J]. *IEEE Transactions on Industrial Electronics*, 2019, 66(7): 5525–5534. DOI:10.1109/TIE.2018.2868023.
- [46] JAYASWAL P, VERMA S N, WADHWANI A K. Development of EBP-Artificial neural network expert system for rolling element bearing fault diagnosis[J]. *Journal of Vibration and Control*, 2011, 17(8): 1131–1148.
- [47] SAUFI M, AHMAD Z A, LIM M H, 等. A review on signal processing techniques for bearing diagnostics[J]. *Int. J. Mech. Eng. Technol*, 2017, 8(6): 327–337.
- [48] TAO H, WANG P, CHEN Y, 等. An unsupervised fault diagnosis method for rolling bearing using STFT and generative neural networks[J]. *Journal of the Franklin Institute*, 2020, 357(11): 7286–7307.
- [49] HINTON G E, OSINDERO S, TEH Y-W. A fast learning algorithm for deep belief nets[J]. *Neural computation*, 2006, 18(7): 1527–1554.
- [50] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. *Advances in neural information processing systems*, 2012, 25: 1097–1105.
- [51] GOODFELLOW I, BENGIO Y, COURVILLE A, 等. *Deep learning*[M]. MIT press Cambridge, 2016.
- [52] 佚名. 直推式迁移学习及其应用研究--《哈尔滨工业大学》2012 年硕士论文 [EB/OL]([日期不详])[2021-04-28]. <http://cdmd.cnki.com.cn/Article/CDMD-10213-1013035954.htm>.
- [53] TARG S, ALMEIDA D, LYMAN K. Resnet in resnet: Generalizing residual architectures[J]. *arXiv preprint arXiv:1603.08029*, 2016.
- [54] TZENG E, HOFFMAN J, ZHANG N, 等. Deep Domain Confusion: Maximizing for Domain Invariance[J/OL]. *arXiv:1412.3474 [cs]*, 2014[2021-01-24]. <http://arxiv.org/abs/1412.3474>.