[redacted]

# 本科生毕业设计(论文)

学 院(系)：[redacted]

专　　业：[redacted]

学　　生：[redacted]

指导教师：[redacted]

完成日期 2023 年 05 月

[redacted]本科生毕业设计（论文）

# 基于集成学习的半监督目标数据分类算法

## Semi-Supervised Target Data Classification Algorithm Based on Ensemble Learning

总　计：毕业设计(论文)　35 页

表　格：　　9 个

图　片：　　8 个

[redacted] 本 科 毕 业 设 计(论文)

# 基于集成学习的半监督目标数据分类算法

# Semi-Supervised Target Data Classification Algorithm Based on Ensemble Learning

学　　院(系)：[redacted]

专　　业：[redacted]

学 生 姓 名：[redacted]

学　　号：[redacted]

指导教师(职称)：[redacted]

评 阅 教 师：[redacted]

完 成 日 期：2023 年 05 月

[redacted]

[redacted]

# 基于集成学习的半监督目标数据分类算法

[redacted]

[摘 要]在大数据时代，数据量庞大、维度广、更新迅速，在线环境瞬息万变。传统的无监督分类算法性能不足，即使出现了无监督深度算法后也难以保证学习的时效性和经济性。对于监督学习来说，大量的数据虽然提高了深度学习算法的性能，但也存在着标注海量数据的高成本和准确率问题。本文提出了一种新的无监督集成学习算法：无监督目标驱动算法。该算法可以对无标签数据集进行划分，并能以该数据集构建模型完成对同类型数据集的划分。本算法首先通过无监督算法获取数据集的伪标签，其次对数据进行特征提取，再将特征与伪标签对应分割后输入到多个基分类器中，最后将基分类器的结果作为特征输入到赋权分类器中完成对基分类器的赋权过程和划分结果输出。文中还基于该划分算法设计了一套半监督分类子算法，只需极少量的标签便可完成对目标数据的分类。此外，本文将当前常用分类算法与本文提出的半监督分类子算法进行对比实验。实验结果表明，本文算法的分类效果较为优秀。

[关键词]集成学习；无监督学习；半监督学习

# Semi-Supervised Target Data Classification Algorithm Based on Ensemble Learning

[redacted]

**Abstract:** In the era of big data, the data volume is huge, dimensional and rapidly updated, and the online environment is rapidly changing. The performance of traditional unsupervised classification algorithms is inadequate, and even after the emergence of unsupervised deep algorithms, it is difficult to ensure the timeliness and economy of learning. For supervised learning, while large amounts of data improve the performance of deep learning algorithms, there are also problems with the high cost and accuracy of labelling large amounts of data. In this paper, we propose a new unsupervised ensemble learning algorithm: the unsupervised goal-driven algorithm. The algorithm can partition the unlabelled dataset and can build a model with this dataset to complete the partition of the same type of dataset. The algorithm firstly obtains the pseudo-labels of the dataset through the unsupervised algorithm, secondly extracts the features from the data, then inputs the features and pseudo-labels into multiple base classifiers, and finally inputs the results of the base classifier as features into the assignment classifier to complete the assignment process of the base classifier and the output of the classification results. A semi-supervised classification algorithm is also designed based on this segmentation algorithm, which only requires a very small number of labels to complete the classification of the target data. In addition, this paper compares current classification algorithms with the proposed semi-supervised classification sub-algorithm. The experimental results show that the classification effect of this algorithm is better.

**Key words:** ensemble learning; unsupervised learning; semi-supervised learning

# 目录

# 1 绪论

## 1.1 课题背景

在大数据时代，复杂且繁多的数据给机器学习领域带来了一些新的挑战：数据丰富多样，结构复杂，需要不同的机器学习方法来解决具体问题，而传统机器学习算法已经难以满足需求。深度学习拥有对数据的深层特征的有效学习能力，而当前较流行的深度学习算法对数据的先验知识或大量已标注数据有较高的要求。在现实的分类问题中，有时候很难拥有对专业问题的先验知识，如医学图像识别领域。在对数据进行标注的工作中亦可能也需要相应的先验知识，除此之外，大量的数据标注工作相当消耗人力物力财力，在海量的数据标注工作下也无法保证标注工作的准确性且人工标注效率不高，费时费力。机器标注若没有足够的先验知识的注入，准确率不高，这对建立于其上的监督学习算法来说是致命的。虽然无监督学习算法不依赖标注数据，但往往性能强大的无监督学习算法也对先验知识存在依赖。在现实层面，要求数据工作者拥有对不同领域的先验知识是不现实的。而对于不依赖先验知识的无监督算法，在当今快速变动的环境下，精度、效率、时效性都不尽人意。本课题的意义在于寻找到从无标签数据集中划分目标数据并通过少量标注数据完成对同类型数据集分类的方法。

## 1.2 研究目的与意义

在当前时代背景下，获取无标签数据的难度远低于获得有标签数据，这使得无监督学习算法将占据越来越重要的地位。社会的发展则使得数据的丰度、信息密度、数量提升得非常迅速，而随着信息化进入到各行各业，社会各界对好的机器学习算法的需求愈发迫切。对于监督学习算法，数据标注工作的限制一定程度上抑制了其发展，而传统的无监督学习算法已无法应付如今时代复杂的数据环境，性能较强的新型无监督学习算法通常又存在对目标领域先验知识的依赖。本课题的研究目的就是找到一种适应当今环境的划分算法，以此完成对无标签数据集的划分任务，然后通过少量标注数据完成对目标数据的分类任务。本课题的意义在于在分类任务中降低工作者对特定先验知识的要求的前提下寻找到一种行之有效的分类算法。

## 1.3 国内外研究现状

为解决日益膨胀的数据量与机器学习分类算法发展不平衡的矛盾，研究人员提出了使用无监督学习方式。当今主流的无监督划分方式主要有聚类集成、无监督深度学习。聚类集成有时也会与自监督方法结合起来，通过将聚类集成获取的一致性结果（监督信号）作为标签输入到监督模型这一过程完成将无监督学习到监督学习的转化。杜航原等人针对聚类集成中一致性函数设计问题，提出一种深度自监督聚类集成算法[1]。该算法通过聚类集成生成划分结果再计算样本间的相似度矩阵然后将划分结果变换成图表示，

该方法通过改善聚类集成的一致性问题来提升聚类算法的性能。谭茜成在无监督自适应领域提出了基于深度学习的 MRSEAN 算法[2]。该算法通过优化核范数和特征标定机制达到了在无监督领域的自适应算法中较优秀的结果。张浩等人提出了基于深度残差自编码器的无监督聚类方法 ResDAE-KMeans++[3]。该方法将 K-means 聚类应用在度残差自编码器上减轻维度灾难问题，在低维空间中有较好的表现。P.Jay 通过使用聚类算法增强具有堆叠自动编码器的深度集群网络[4]。Nurmaini S 等人在研究心电监测和分类系统中提出了一种新的无监督深度学习架构[5]。该架构以自动编码器作为无监督的特征提取层，深度神经网络作为分类层。在无监督深度学习应用方面，kçakaya 等人在文章中从经典逆向问题的连贯性角度讨论了自监督和生成模型在生物成像方面的应用[6]。

## 1.4 课题的主要内容

本课题主要研究内容有两点，一是将无监督学习与监督学习结合起来，得到一种既不依赖标注数据，又具有相对强大性能的无监督划分算法；二是基于该算法设计一套半监督子算法，通过引入少量标注数据，完成从无监督数据集对目标数据的分类工作。具体工作主要有以下几点：

（1）对机器学习领域的一些概念和方法进行梳理。

（2）设计出了一种新的无监督划分算法：无监督目标驱动算法。

（3）模拟现实场景中的一种分类问题：半监督不平衡数据集的病毒图像分类问题。

（4）根据无监督目标驱动算法的框架和规则设计出一套半监督目标数据分类算法，检验其分类效果，并以此验证无监督目标驱动算法的框架和规则的可行性。

（5）将基于无监督目标驱动算法的半监督目标数据分类算法与其它当今主流算法进行对比，评估半监督目标数据分类算法的性能。

## 1.5 论文组织结构

本文各章节安排如下：

第 1 章为绪论，介绍了本文的课题背景以及研究目的与意义，然后介绍了当下无监督学习的国内外研究现状，最后描述了课题的主要内容与本文的组织结构。

第 2 章为相关技术介绍，介绍了与课题研究内容相关的理论知识以及方法。接着介绍了无监督学习领域的一些常用算法和较新研究。

第 3 章为基于集成学习的无监督目标数据驱动算法，介绍了本文提出的无监督目标驱动算法（UGD）的思路构建、算法描述、框架构成以及算法流程。

第 4 章为基于 UGD 的子算法在半监督分类任务中的应用，介绍了无监督目标驱动算法在现实分类问题中的应用，给出了基于 UGD 算法的半监督分类算法的具体设计和实现过程，并对测试结果进行了分析，提出了改进策略；最后将基于 UGD 算法的半监督分类算法与当今主流分类算法进行了对比评价。

# 2 相关技术介绍

## 2.1 监督学习与无监督学习概述

### 2.1.1 监督学习

监督学习是从已经被标记的训练集中推断一个功能的机器学习任务[7]。

监督学习模型按模型形式以及是否对观测变量建模可分为三类：非概率模型、概率判别模型、生成模型。

在大数据时代，监督学习的分类任务主要为多标签分类。监督学习在多标签分类任务中，有基于二元相关性的分类器链方法[8]，有基于问题转化[9]的标签幂集方法以及随机子标签集成算法等；有基于算法自适应的决策树、K 近邻、支持向量机、神经网络等。

卷积神经网络（Convolutional Neural Networks，CNN）属于监督学习中基于算法自适应的前馈神经网络，主要特点是卷积计算和深度结构，是深度学习中的典型算法，而残差卷积神经网络通过在卷积网络构建中从输入直接引入一个短连接到非线形层的输出上来解决层数过高时网络的退化问题。

支持向量机（Support Vector Machines，SVM）是一种基于统计学习理论和最优化理论的监督学习模型。支持向量机通过算法自适应获得了在多分类问题中的良好的泛化性和全局优化性，支持向量机通过寻找可以将不同类别的数据在特征空间上的数据离划分面间隔最大的最优超平面完成分类和回归任务，是一种线性分类器。

决策树（Decision Tree，DT）是一种树型结构的监督学习分类算法。决策树中的分支节点表示对某个属性的判断，叶节点表示分类结果。当今主流的决策树算法主要有 CART 算法，ID3 算法，以及 C4.5 算法。

### 2.1.2 无监督学习

无监督学习是指从未被标记的数据集中来解决模式识别中的划分或降维问题。

无监督学习在广义上可分成两个子类，一个是强化学习，一个是自组织学习[10]。自组织学习主要有自组织映射，信息论学习模型、统计力学三类。强化学习常与动态规划联系起来，以马尔可夫决策作为数学基础。在分类问题中，监督学习模型因为利用了标记信息，通常效果会更好。

## 2.2 自监督学习与半监督学习概述

### 2.2.1 自监督学习

自监督是一种机器学习范式。自监督算法的大致流程为：先用无监督学习模型获取特征表示，然后用监督学习模型进行训练。

自监督学习解决的是数据的分类或回归问题。根据目标数据是否标注的不同则更像是无监督学习。因此，本文认为自监督学习比起监督与无监督的中间态，更像是一种令

机器学习模型逐渐靠近理想状态（机器无监督自训练）的一种集成学习方式，其性能的有效性解释为输入（先验信息）的减少和输出（有效特征）的提高。

### 2.2.2 半监督学习

半监督学习从含未标记和标记的数据集中学习，通常假定从相同或相似的大分布中抽样。半监督学习从未标记数据的结构中采用不同的方式获得不同的信息。评估半监督学习算法的标准方案为:（1）从一个标准的标记数据集开始；（2）只保留部分标签（比如 10%）；（3）将其余部分视为未标记数据。虽然该方法可能无法反映现实情况，但仍然是标准评估协议[11]。目前许多关于半监督学习的初步成果都基于生成模型的深度神经网络。

## 2.3 集成学习

### 2.3.1 集成学习概述

集成学习使用多个机器学习模型对同一个目标进行学习，并通过不同方法将这些模型组合成一个性能更强的集成模型来完成回归或分类任务。

在分类问题中，Margin 理论能够很好地解释集成算法的有效性[12]。Margin 可以简单解释为预测为某类别的概率与其他类别中的最大概率的间隔。当训练样本的数量与基模型的复杂度不变时，集成模型的泛化误差与 Margin 在训练集中分布的均值成反比，与方差成正比。现有集成学习主要采用启发式方法，从数据的多样性、模型参数的多样性、模型结构的多样性等方面提升基模型之间的多样性[13]。集成学习的理论还需进一步研究。目前集成学习的理论对部分集成模型的有效性进行了阐释，但对集成模型多样性的定义、多样性的度量还没有形成共识[13]。

### 2.3.2 集成学习的主要方法

集成学习的主要方法包括袋装法（Bagging）、提升法（Boosting）、堆栈法（Stacking）、多核学习（Multiple Kernel Learning，MKL）、集成深度学习（Ensemble Deep Learning，EDL）等。

Bagging 通过对数据集重采样用于训练基模型，结果通过投票法等进行集成。Bagging 能够对基模型进行并行训练。基于 Bagging 的典型算法随机森林（Random Forest，RF）在当前的分类以及回归领域中占据了重要地位。Tuysuzoglu 等针对训练子集选择随机性的问题，提出了增强的 Bagging[14]。

Boosting 则是通过对误差权重和传递参数的调整令前一个基模型增强后一个基模型的性能，最终获得一个性能强力的模型。基于 Boosting 的典型算法梯度提升决策树（Gradient Boosting Decision Tree，GBDT）被广泛应用于各类机器学习比赛和工业界。

Stacking 以多个基模型的输出作为后续超特征对元模型进行训练，在对基模型的训练过程中采用 k 折交叉验证方式减少过拟合的发生。与 Bagging 对比，Stacking 基模型

的选择往往是不同质的复杂模型，而元模型通常选择复杂度低的机器学习模型，比如逻辑回归。针对非均衡分类问题，Seng Z 等人提出了一种名为邻域欠采样堆叠集成（NUS-SE）的方法[15]。

多核学习将不同的特征映射到不同的特征空间，对其采用不同的核函数，然后训练每个核的权重，选出最佳核函数组合后进行多核集成，然后使用 SVM 等线性分类器来进行分类。

集成深度学习将深度学习模型作为基模型，使用 Bagging 集成不同结构的神经网络模型以减小整体方差，最后以投票或加权投票等方式完成分类。

## 2.4 聚类集成

聚类算法是一类重要的无监督算法，但在大数据时代，单一的聚类算法的性能不足以应对丰富度与数据量都相当大的样本，这时，聚类集成方法应运而生。聚类集成通过集成多个弱的基聚类结果，得到一个更鲁棒，更稳定的聚类结果[16]。

聚类集成首先通过集成多个基聚类器生成聚类结果，再用共识函数生成一致性结果。对于基聚类器，除非引入规则，否则差异性不大，毕竟聚类算法是建立在数据之上的。

对于基聚类间的集成方式，主流应用的是 Bagging，偶见 Stacking。因此可以说聚类集成的核心就是共识函数。比较经典的共识函数主要有基于投票的、基于图划分的、基于信息理论的、基于共协矩阵的、基于混合模型的[16]。

## 2.5 无监督深度学习

无监督深度学习因为对标记数据量的需求低，在当今时代相当受欢迎，如今的深度学习方法在无监督领域应用主要依赖以下几种方式实现：

（1）无监督领域适配：利用基于已有领域的标记数据生成的深度神经网络模型来解决数据分布不同的其他领域的分类问题[17]。在此过程中的主要工作是寻找模型在新数据集的适配策略。

（2）通过聚类集成和深度神经网络的模型融合：聚类集成提高了聚类算法的精度和鲁棒性，将其稳定结果作为标签训练神经网络，亦可得到还不错的结果。

（3）将先验知识纳入神经网络的训练阶段：在可以使神经网络获取到有意义的特征，如深度信念网络的变体 DBNs[18]、叠加自动编码器的变体 SAEs[19]以及 CAEs[20]。

（4）深度聚类：深度聚类的本质是用神经网络学习一个聚类导向的特征表示，用神经网络拟合数据内蕴的聚类规律[21]。深度聚类模型通过将不同的聚类算法与不同神经网络结合来达到不同的特性和效果。

无监督深度学习算法多是以上几种方式的优化或联用，或者引入半监督信息。此外还有一些特殊的无监督深度学习算法基于生成模型。如生成对抗网络[22]、使用 STDP 机制的深度脉冲神经网络[23]等。

Y. Sun 等人在论文中提出了进化深度网络 EUDNN[24]。EUDNN 通过添加和改进进化策略、优化编码方式、采用局部搜索策略、采用适配性评价等优化方式获得了很好的分类效果。

李嘉恒提出了三种无监督异源遥感图像变化检测方法[25]。他通过耦合自编码器将初始图像的特征映射到高维空间引入先验信息达成无监督深度学习的效果。

Yanpeng Cao 通过利用多光谱数据的互补信息，迭代标记可见光和热通道中的行人实例[26]。通过图像序列对不同的目标进行时间跟踪来生成更可信的标签。

# 3 基于集成学习的无监督目标数据驱动算法

## 3.1 无监督目标驱动算法的构建思路

人类可以从一些简单的数据特征中抽象出蕴含其中的深层信息，比如从照片中获取年代信息。想让机器学会如人脑一般强大的抽象能力是很困难的，深度学习有这方面的潜力，可这依赖于输入的大量的含抽象映射的数据。而给数据做标注，尤其是给抽象数据做标注更是一件不容易的事，这亦是无监督数据划分受到重视的原因。

在大数据时代，信息获取的便捷以及信息丰度的提升大大增加了数据划分的难度。而由于学科分支、行业分支的细化，数据划分算法在专业领域的应用对先验知识的要求更高了。数据类型的复杂度虽然有所提高，但目前并没有发生本质的改变，归根结底都能转换成二进制编码。于是本文针对此现象提出了一种构想：在数据划分任务中，通过人脑的抽象功能获悉数据集的某些特性，然后使用相应的处理方式帮助机器更容易获取到蕴藏在数据集中的关键信息。通过该过程便完成了将划分算法对数据的先验知识的依赖转化为操作者对数据特性的认知。

计算机行业从业者往往有着较高的数据敏感性和相关知识，比如在软件开发中算法工程师会根据问题的规模、类型等选择合适的数据结构与处理方式。在划分问题中，对数据集采用适宜的处理方式，以此完成不同类别的目标数据的划分。这将使专业领域数据划分变得不那么依赖专业知识。

对于如何让机器顺利完成数据划分这一问题，本文选择令机器更容易的产生从特征到类别的映射。对于完成该目标的方式，本文设立了让机器获取足够的与类别相关的特征信息以及排除无用信息的干扰这一目标。基于此目标以及现实环境中的数据划分需求本文提出了无监督目标驱动算法。

## 3.2 无监督目标驱动算法描述

无监督目标驱动算法（Unsupervised goal-driven algorithms，UGD）是一种基于无监督集成学习的目标数据划分算法，通过集成不同的无监督学习算法和监督学习算法解决具体的无监督目标数据划分问题。

该算法的主要特点是通过将无监督学习中对先验知识的依赖转化为操作者对数据类型以及相应处理方式的了解的依赖令从事数据划分工作的人员不必对目标数据有专业认知。UGD 根据工作者对划分任务和数据集的显式信息的了解，如数据集含有的样本量、样本的数据类型、数据结构、编码方式等选择合适的方法对设定目标的逼近。

基于本文对机器完成划分工作的认知，对 UGD 算法设定的最初目标为让机器学习算法模型更轻松的产生从特征到类别的映射。这个最初目标对于计算机来说亦是较难理解的，于是便根据这一目标提出了 UGD 算法更为细致的三大目标：（1）获取足够的深层特征信息；（2）选出有用的深层特征信息；（3）减少无用的信息干扰。在 UGD 算

法的应用时，可通过提出更为具体的目标来达成对三大目标的趋近。

## 3.3 无监督目标驱动算法框架

### 3.3.1 UGD 算法框架层次结构与具体描述

UGD 算法框架大致可分为四层：伪标签获取层、深层特征提取层、误差稀释层，赋权分类层。图 3-1 表示了 UGD 算法各层次的大致构成以及信息的传递方向。

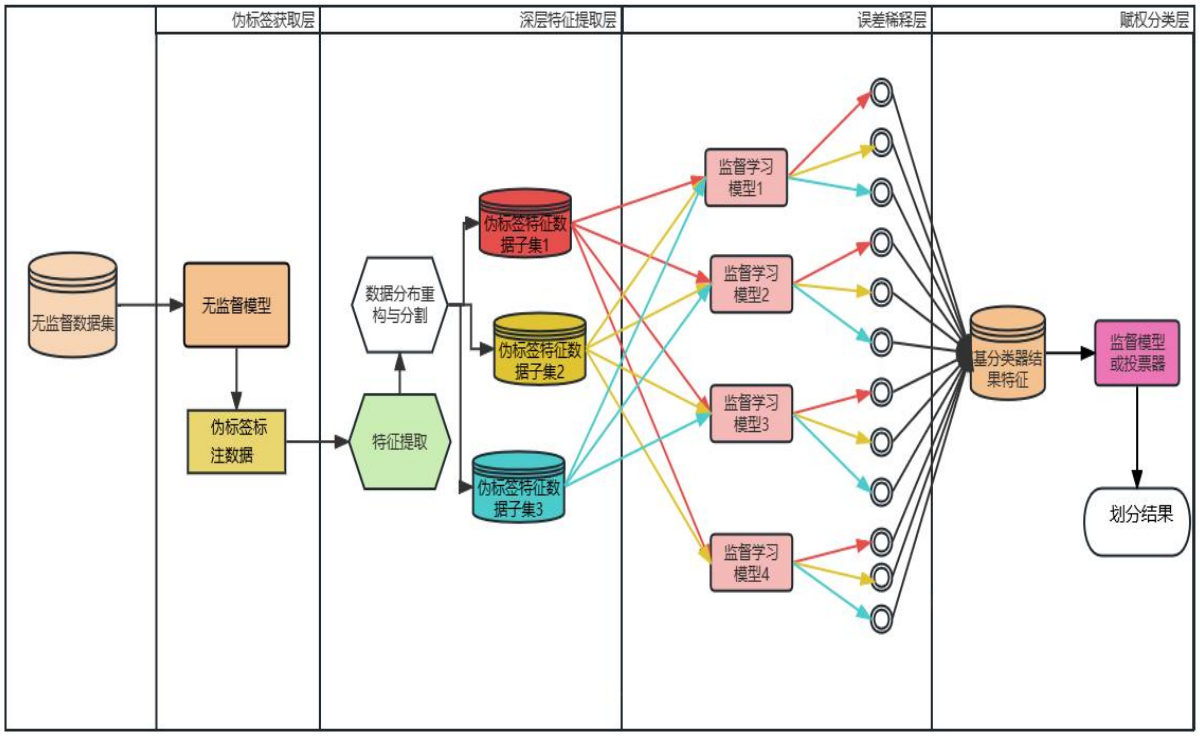


图 3-1 UGD 算法泳道图

为方便后续表达，本文在对各层的具体解释中对操作的数据集以及生成的模型进行了命名。以下为对 UGD 算法框架各层次的具体描述。

（1）伪标签获取层

在学习过程中，该层以无监督学习算法为核心，获取并给无标注样本打上伪标签，将经过伪标签数据集称为伪标签数据集。该层将无监督划分问题转换成了监督学习中的带噪学习问题。该层的输入为无标签数据集 D，该层的输出为经伪标签标注过的数据集，将其称为伪标签数据集 W。该层只在学习过程发挥作用，不参与后续的划分过程。

（2）深层特征提取层

在学习过程中，该层以深度学习算法为核心，给输入数据集中的样本进行重编码、对数据提取出足够的深层特征得到具有伪标签的样本深层特征数据集，将其称为特征数据集 T；还获得了训练好的深度学习算法模型，将其称为特征提取器。然后对 T 中的样本特征数据的分布或者样本标签特征的分布做出调整得到特征重构数据集 t，接着将其进行分割，得到多个特征重构数据集 t 的子集。在学习过程中，对样本数据集的调整和

分割过程应该保留伪标签与样本特征数据的对应性，以将伪标签信息顺利传递到下一层。在此将对应好的伪标签切出，构成特征重构数据集 t 中样本所有对应伪标签的列表 w。该层在学习过程中的输入为伪标签数据集 W，输出为具有样本特征重构数据集 t、t 的多个子集、t 中样本对应的伪标签列表 w，还有训练好的特征提取器。

在划分过程中，该层先对输入的数据集中的样本采取和学习过程一致的重编码方式，然后使用特征提取器对重编码的样本数据完成深层特征的提取得到特征数据集 T，如果在学习过程中采用了调整了数据的特征分布，则采用和学习过程一致的调整方式，得到特征重构数据集 t；若未调整特征分布直接将特征数据集 T 作为特征重构数据集 t。由于分类过程不具有伪标签信息，所以不需要考虑伪标签信息的传递。该层在划分过程的输入为无标签数据集 Y，输出为特征重构数据集 t。

（3）误差稀释层

在学习过程中，该层以集成学习算法为核心，将多个具有伪标签的样本特征重构数据集 t 的子集作为多种监督学习模型的训练集，其中的特征数据作为输入的特征（feature），对应的伪标签作为输入的特征（label）。分别使用 t 的子集作为多种监督学习算法模型的训练集进行训练，得到了子集个数和使用的监督学习模型种数的乘积个的基分类器。然后使用所有基分类器分别将样本特征重构数据集 t 中的特征数据作为输入的特征（feature）进行预测，然后将所有基分类器预测的结果（对训练集中每个样本预测的标签）列表合并，构成预划分数据集 y。该层在学习过程中的输入为样本特征重构数据集 t 以及对应的伪标签列表，还有 t 的多个子集；输出为由各个基分类器对样本特征重构数据集 t 中所有样本的划分预测结果列表构成的二维列表。

在划分过程中，将输入的特征重构数据集输入到多个基分类器中进行预测，然后将所有基分类器预测的结果（对训练集中每个样本预测的标签）列表合并，构成预分类数据集 y。该层在分类过程的输入为特征重构数据集 t，输出为预划分数据集 y。

（4）赋权分类层

该层有两种输出模式：赋权模式以及平均模式。当使用赋权模式时，需要对本层使用的监督学习模型进行训练；训练集的特征（feature）使用预划分数据集 y，标签使用其对应的伪标签列表 w（feature）。当使用赋权模式时，不需要学习过程。

在学习过程中（仅存在于赋权模式），该层将输入的预划分数据集 y 以及对应的伪标签列表 w 训练监督学习模型获得了一个赋权划分器。在赋权划分器的训练过程中，赋权划分器学习到的是基分类器的预测标签与伪标签的对应关系，给误差稀释层中各个分类器完成变相赋权。该层在学习过程的输入为预划分数据集 y 和对应的伪标签列表 w。

在划分过程中，当使用赋权模式时，使用赋权分类器对传入的预划分数据集 y 进行预测，获得最终的划分结果。当使用平均模式时，将预划分数据集 y 中每列数据的值（各个集分类器对该样本的预测结果）进行投票，选出该列得票最高的预测结果作为该样本

最终判定的划分结果，然后将每列的投票结果添加到一个新列表中，作为该无标签数据集 Y 下所有样本的划分结果。亦可将投票决策换成无监督学习模型（如聚类），将无监督分类的结果转化成列表后输出，这种方式对每个基分类器的预测结果的依赖程度也是相同的，所以也属于平均模式。该层在分类过程的输入为预划分数据集 y，输出为 UGD 对无监督数据集上所以样本最终的划分结果列表。

### 3.3.2 UGD 算法框架的构成原理

该UGD算法的核心为设定的目标规则，其算法框架的构建亦基于此。在此重申UGD算法的三个主要目标为：（1）获取足够的深层特征信息；（2）选出有用的深层特征信息；（3）减少无用的信息干扰。为方便后续的表述，分别将其简称为目标 1、目标 2、目标 3。

通过无监督学习算法给无标签数据集打上伪标签可将问题从无监督分类转换成监督学习中的带噪学习问题。伪标签信息给了监督学习模型建立时提供了指导信息，趋近目标 2。因此建立了 UGD 算法框架中的无标签获取层。无监督模型获取的伪标签可能与目标的真实划分有差异，这将使后续的监督学习算法受到噪声干扰，这背离了目标 3。于是在构建该层时，应当选择较好的无监督模型以减少噪声干扰来趋近目标 2 和目标 3。

对输入数据重编码可拓展算法的使用范围，可匹配更多数据类型的输入，这为趋近目标 1 提供了基础。深度学习算法可以从数据提取出更深层次的特征，这趋近了目标 1。对数据或特征的分布做出调整和对数据集进行分割可使算法对特征信息有更多的选择这趋近了目标 2 和目标 3。根据以上分析，构建了以深度学习算法为核心的深层信息提取层。特征信息的深度选择取决于当前划分任务的问题规模以及数据的复杂度。

集成学习算法可通过集成多个分类算法模型提升其分类效果。集成模型的分类性能主要取决于基分类器的数量、性能、集成策略，以及基分类器间的相关性[17]。通过 Bagging 方法集成多个监督学习模型，通过将之前分割过的数据集逐一传入监督学习算法中，可构建更多输入数据不同，但是独立同分布的分类模型。这些模型学习到的数据是不同的，其中错分的伪标签被不同的模型学习到，通过这些模型对相同数据分类，它们的错误划分被整体模型稀释，便达到了降低噪声干扰的作用，趋近了目标 3；同时，以 Bagging 的方式集成这些模型提高了并行计算的效率。通过让 Bagging 构成的基分类器组学习被深度学习模型处理过的特征和对应伪标签达到了纵向集成不同机器学习模型，这和 Boosting 集成方法一致，增强了基分类器的性能，趋近了目标 1。另外，从深度学习算法模型中传出的数据，往往维度较高，在基模型对数据的处理中可以采用不同的核函数，即通过集成学习方法中的多核学习进行处理，提高基分类器间的差异性，可获取到更多的不同信息以趋近目标 1。根据以上分析，构建了以集成学习算法为核心的误差稀释层。

从误差稀释层获得了多个基分类器，要将它们的结果进行输出需要一个合理的方式。集成学习中 Stacking 方法将不同数据训练成的基模型的结果作为超特征用于元模型的训

练这一方式便正好符合这种情况。通过 Stacking 的方式可以挑选出基分类器表现好的部分，抛弃不好的部分，趋近了目标 2 和目标 3，同时，利用完整的含伪标签的数据对监督学习进行训练，可将较完整的伪标签信息引入，给基分类器赋权。有研究人员提出对标签噪声的过度清洗可能会将分类器的性能降低[27]。而该方法可以获取到一些可能被误差稀释层过滤掉的有效信息，还通过对分类器基于与标签数据拟合程度的赋权过滤掉分类效果很差的分类器，趋向了目标 1 和目标 2。但是这种方式重新引入了伪标签信息的干扰，偏离了目标 3。对于该问题，本文提出了另一种输出模式。通过直接投票法或者无监督学习模型对基分类器的预测结果整合归一。这一过程是对每个基分类器赋予的权重都是相同的，不会引入伪标签数据的干扰，但无法完成对基分类器的赋权将使得对基分类器的性能或数量有更高的要求。根据上述分析，构建了赋权输出层的两种输出模式，以监督学习算法为核心的赋权模式和以投票或无监督学习算法为核心的平均模式。除了这两种输出方式之外，如果通过引入半监督信息对赋权模式的赋权分划分器进行训练，便可得到对基分类器的精准赋权，理论上能有效增强整体模型的精度，但本章节研究的为无监督划分，便不再深究。

## 3.4 无监督目标驱动算法流程

表 3-1 和表 3-2 分别以伪代码的形式表示了 UGD 算法的学习过程和 UGD 算法对所需划分的目标无标签数据集的划分过程，其中括号里的为当前层的输入，箭头表示信息的传递，箭头左端为传入信息的来源，右端为传出信息的对象或者该行的输出。

**表 3-1 UGD 算法学习步骤**

| 算法：UGD 算法（学习） |
| --- |
| **输入：**无标签数据集 D（训练集） |
| 1、伪标签获取层（无标签数据集 D）： |
| 2、 D → 无监督模型 → 伪标签数据集 W |
| 3、深层特征提取层（伪标签数据集 W）： |
| 4、 W → 监督学习模型→ 深层特征数据集 T |
| 5、 保存训练好的监督学习模型作为特征提取器 |
| 6、 T → 数据分布处理器/特征分布处理器 → 特征重构数据集 t，伪标签列表 w |
| 7、 T → 数据分割函数 → T 的 n 个子集{t1，t2，t3…} |
| 8、误差稀释层（t，{t1，t2，t3…}） ： |
| 9、 {t1,t2,t3...} → m 个不同的监督学习模型 → m*n 个基分类器 |
| 10、 保存训练好的多个基分类器 |
| 11、 t → m*n 个基分类器 → 预分类数据 y |
| 12、赋权输出层（y，w）： |
| 13、 y，w →监督学习模型 → 赋权分类器 |
| 14、 保存训练好的赋权分类器 |
| **输出：**特征提取器、多个基分类器、赋权分类器 |

**表 3-2　UGD 算法划分步骤**

| 算法：UGD 算法（预测） |
| --- |
| **输入：**无标签数据集 Y |
| 1、载入特征提取器、多个基分类器、赋权分类器 |
| 2、深层特征提取层（无标签数据集 Y）： |
| 3、　　Y → 特征提取器 → 特征数据集 T |
| 4、　　if 学习时使用了特征分布处理器： |
| 5、　　T → 特征分布处理器（与训练过程相同的） → 特征重构数据集 t |
| 6、　　else: t = T |
| 7、误差稀释层（t） |
| 8、　　t → 多个基分类器 → 预分类数据集 y |
| 9、赋权分类层（y） |
| 10、if 输出模式为赋权模式： |
| 11、　　y → 赋权分类器 → 对无标签数据集 Y 中所有样本的分类标签； |
| 12、else : |
| 13、　　y → 投票机制/无监督模型（与训练过程相同的）→ 对无标签数据集 Y 中所有样本的分类标签 |
| **输出：**对无标签数据集 Y 中所有样本的分类标签 |

# 4 基于 UGD 的子算法在半监督分类任务中的应用

## 4.1 任务描述

在实际任务场景中，数据的类别的往往是不平衡的，这是工业应用亟待解决的关键问题[28]。而伴随于此的问题则是标注数据获取的困难。这两种问题出现的场景重叠的概率是很高的，比如在医学图像识别领域。

本文提出的 UGD 算法可以对无标签数据进行划分。基于 UGD 算法构建的分类子算法只要使用极少的标记数据完成对伪标签的重命名，将伪标签与真实标签对应起来，便可以完成对少标签数据的分类任务。于是本文尝试通过模拟“从少标签不平衡数据集中划分出目标数据并完成分类”这一任务，用于检验 UGD 算法的可行性和基于 UGD 算法构建的子算法在该分类任务中的表现。

## 4.2 数据获取以及数据集构建

目标数据来源于 NIAID 发布的电镜下的 covid19 病毒的显微图像以及 CDC 发布的 SASR 病毒显微图像，共 2 类，每类各 5 张，获取方式为官网截图。

干扰数据来源于 https://image.baidu.com，共 40 类，含动物、车辆、日用品、风景、动漫角色等共 820 张，获取方式为爬虫程序。

数据处理：将目标图像通过数据增强方法进行扩充。通过添加高斯噪声、椒盐噪声、旋转、昏暗等方式扩充到了 180 张。

数据集构建：首先将 covid19 病毒图像、SARS 病毒图像、干扰图像标注，得到一个干扰数据为目标数据四倍多的不均衡数据集，总数为 1000 张。然后将初始数据集等比例分割，将其中 20%的数据作为测试集，不参与任何训练过程。

将初始数据集剩下的 80%数据作为和监督学习算法进行对比评价时监督学习算法的训练集。然后将该训练集中的图片数据复制到同目录的同文件夹下混合打乱后重命名构成无标签不均衡数据集。最后在无标签不均衡数据集中标注目标类别的图片各 5 张构成少标签不均衡数据集作为 UGD 子算法的训练样本。该少标签训练集中标注数据的比例为 1.25%（5*2/1000*0.8）低于半监督学习的一般标准（10%）。

标注数据采用的方式：将同目录下的不同文件夹作为不同的类别，文件夹命名为类名，读取数据时通过读入文件的地址便可获取到其中的标注信息。将同文件夹下的文件视为同一类，少标签数据集为这种，读取数据时无法通过文件地址得知无标签样本的标注信息。标注少标签数据集中的标注数据的方式为将标注数据的图片名重命名为类名。

## 4.3 基于 UGD 的子算法设计

首先分析数据集和目标任务，然后根据 UGD 算法三大目标的指导向 UGD 算法框架中填充适配目标任务和数据集的相应算法模型，得到应对具体问题的 UGD 子算法。

以下为 UGD 算法在应对当前分类任务时的设计过程：

目标任务为从少标签不均衡数据集中划分出目标数据（两类病毒显微图像），并完成对目标数据的分类。根据该任务的要求，本文考虑通过将少量标注数据与无标签数据集一同引入到 UGD 算法框架的伪标签提取层，通过将标注数据被划分的位置将伪标签与真实标签进行对应，以完成 UGD 子算法从目标数据划分算法到分类算法的转变。在赋权分类层中，使用标注数据对赋权分类器进行训练可完成对各个基分类器的精准赋权，理论上能提高整体分类的准确度，鉴于任务给出的标记数据过少，便不进行该步骤。

伪标签提取层：无监督深度学习可以准确高效地实现数据的整合与分析[29]。将无监督模型结合深度学习往往能降低伪标签的错分率，这可以令伪标签获取层更接近目标 3。考虑到输入的数据为图像数据，本文选择将图像数据转化为像素矩阵再通过卷积神经网络的卷积层完成对图像数据进行特征抽取，然后使用聚类算法完成划分。通过在这一阶段获取标注数据被划分到的簇，将对应簇名更改为标注数据的类名，以此完成伪标签和真实标签的对应。若有同类的标记数据被划分到不同的簇，则应遍历各个簇，统计其中含该类别标记数据最多的簇，将其命名为该类别的类名。若不同的簇中含有相同数量的同类标记数据，则应该更换性能更好的聚类算法，或者在赋权分类完成之后再对真实标签进行对应。

深层特征提取层：本文基于目标 1 选择适合图像数据的 CNN 作为特征提取的核心。使用含两层卷积含两层池化的卷积神经网络，优化方式使用随机梯度下降，损失使用多类别的交叉熵损失函数，将 CNN 全连接层的输出作为图像提取的特征，输出层做为一个基分类器。结合数据集的规模不大以及样本类别不均衡的情况以及目标 3，数据的重分布选择了 shuffle，过滤掉无用的样本间关联信息；数据集的样本量不大，伪标签获取层使用了较复杂的模型伪标签的错分率预计不会太高，所以误差稀释层不需要过于复杂的分类结构于是便决定只将混淆后的经伪标签标注的样本特征数据集均分为两部分输入到误差稀释层。

误差层稀释层：本文基于目标 3 选择具有较大差别的监督学习模型。差异性是分类器集成具有高泛化能力的必要条件[30]。同时从 CNN 提取的样本特征信息有着较高的维度，而数据集规模却不大，因此基于目标 1 和目标 3 选择使用 3 种基于不同核函数的支持向量机作为误差稀释层的监督学习模型目标，支持向量机对样本数量的要求并不高，决定其决策超平面的是决策边界附近的数据，对数据的分布特征有着更高的要求。

赋权分类层：本文对 UGD 算法框架的两种输出模式都做了尝试，赋权输出模式以决策树作为终端分类器，平均输出模式以直接投票法将基分类器的结果输出。

最后对选择的机器学习算法模型以 UGD 的三个目标进行优化，得到了基于 UGD 算法的子算法，该算法不仅能完成目标数据的分类任务，还可在完成学习后对含目标数据的无标签数据集进行分类。图 4-1 给出了 UGD 子算法两种输出模式下的框架图。

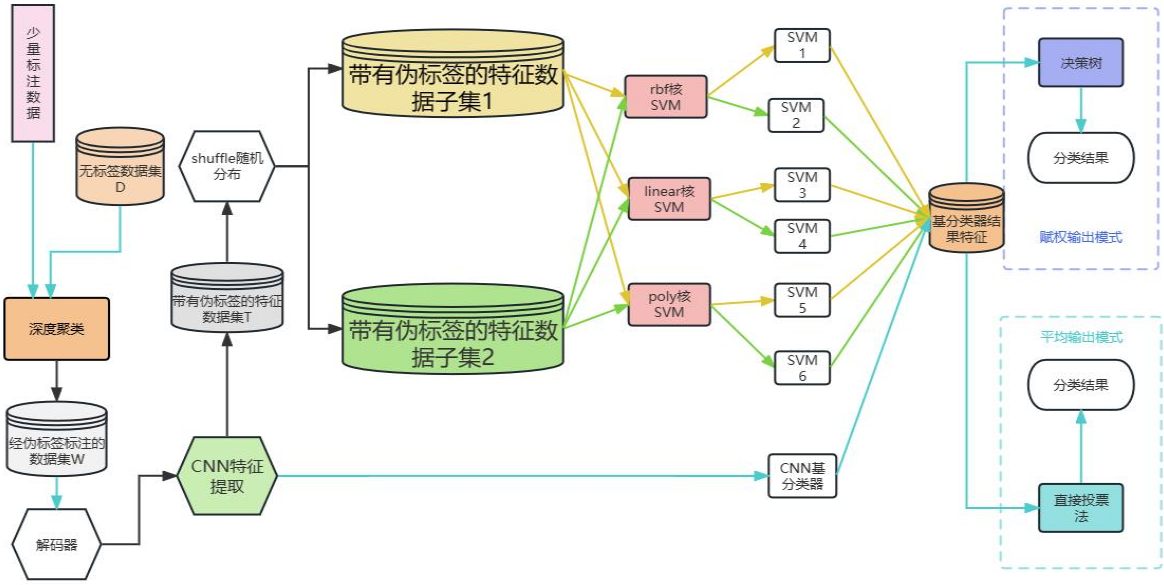


图 4-1　UGD 子算法框架图

## 4.4　UGD 子算法的模型结构

### 4.4.1　伪标签获取层的模型构成与相关参数说明

在算法设计中，本文决定使用通过将深度学习中的卷积神经网络 CNN 结合聚类 K-means 算法的方式构成深度聚类算法作为伪标签获取层的核心。然而通过这种方式与无监督算法结合将使网络的复杂度很大，反而容易产生退化问题。残差神经网络采用跳跃式结构，使得深度残差网络可以越过中间几层直接将参数传递给后面的层可降低网络的复杂度提升性能[31]。本层使用的深度聚类算法模型以残差卷积网络模型 resnet50 以及聚类算法模型 K-means 作为工作主体。其中 K-means 是一种基于欧式距离的数据划分算法。resnet50 网络是一种通过将输入跨层传递来解决神经网络中退化问题的神经网络模型。具体的工作流程为先将输入的图像数据通过使用 resnet50 进行预训练，得到原始数据经过非线性映射到潜在特征空间的特征。然后对得到的特征用 K-means 算法进行网络初始化，得到初始聚类中心。再使用相对熵迭代，微调网络，直至收敛。本文使用的 resnet50 模型来自 pytorch 库中的 torchvision.models 包，相关参数的设置使用 pretrained=True 自动下载模型所对应权重并加载到模型中。由于目标数据有两类，故在 K-means 算法中将 k 值设定为 3，k 为 K-means 算法聚类后生成的簇的个数。除此之外，该层在读入标签时保存了样本的所在地址，在深度聚类完成后将各个簇内的样本复制后保存在同一目录的不同文件夹下，将对应簇名更改为含标注数样本最多的类名，文件夹以簇名命名。如此便可通过样本所在文件夹的地址作为伪标签，且完成伪标签和真实标签的对应。若不同的簇中含有相同数量的同类标记数据，则应该更换性能更好的聚类算

法，或者在赋权分类完成之后再对真实标签进行对应。

### 4.4.2 深层特征提取层的模型构成与相关参数说明

在算法设计中，本文决定使用 CNN 提取图像数据的深层特征，在此过程使用伪标签作为训练过程中的特征对应标签。

表 4-1 CNN 的模型结构

| Layer (type) | Output Shape | Param |
|---|---|---|
| rescaling (Rescaling) | 224， 224， 3 | 0 |
| conv2d (Conv2D) | 222，222，32 | 896 |
| max_pooling2d(MaxPooling2D) | 111，111，32 | 0 |
| conv2d_1 (Conv2D | 109，109，64 | 18496 |
| max_pooling2d_1(MaxPooling2D) | 54，54，64 | 0 |
| flatten (Flatten) | 186624 | 0 |
| dense (Dense) | 128 | 23888000 |
| dense_1 (Dense) | 0 | 387 |

表 4-1 为 CNN 的模型结构，其中 Layer（type）表示每层网络的类型，Output Shape 表示每层输出数据的维度，其中第三个参数表示的是该层卷积核的数量或下层的通道数，Param 表示每层神经元参数的个数，Param 的计算方式如下公式：

$$\text{Param} = (\text{卷积核长} \times \text{卷积核宽} \times \text{通道数} + 1) \times \text{卷积核个数} \quad (4\text{-}1)$$

该 CNN 模型首先对模型做归一化的处理，将 0-255 之间的数字统一处理到 0 到 1。第一个卷积层的输出为 32 个通道，卷积核的大小是 3*3，激活函数为 ReLU，然后是第一个池化层池化的 Kernel 大小是 2*2。第二个卷积层，输出为 64 个通道，卷积核大小为 3*3，激活函数为 ReLU，接着是第二个池化层，最大池化，对 2*2 的区域进行池化操作。然后是全连接层，此时的输出为 128 维，最后是输出层，激活函数为 Softmax 输出的是对应样本每个标签的概率。本文在该层获取的便是全连接层的 128 维特征数据，以及在训练过程中得到的 CNN 模型。

模型的构建中采用了随机梯度下降（sgd）优化器，其更新公式如下：

$$\text{Loop}\{\ \text{for i=1 to m},\{\theta_j := \theta_j + \alpha\left(y^{(i)} - h_\theta\left(x^{(i)}\right)\right)x_j^{(i)}\ \ (\text{for every j})\ \}\ \} \quad (4\text{-}2)$$

对于模型的损失函数将其设置为交叉熵损失函数（Categorical_Crossentropy），其计算公式如下：

$$Loss = -\sum_{i=1}^{n} y_i \cdot log\hat{y}_i \quad (4\text{-}3)$$

卷积层采用的激活函数为 ReLU，其公式如下：

$$f(x) = max(0, x) \quad (4\text{-}4)$$

输出层采用的激活函数为 Softmax，其公式如下：

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_{c=1}^{C} e^{z_c}} \quad (4\text{-}5)$$

### 4.4.3 误差稀释层的模型构成与相关参数说明

在算法设计中，本文决定在该层采用支持向量机（SVM）作为基分类模型。在将数据传入之前，本文先对上层输出的带有伪标签的特征数据集进行数据分布的重构以及数据集的分割。数据分布重构使用了 sklearn 库中的 shuffle 函数，该函数可将输入的样本数据和伪标签数据对应打乱，并可设定随机种子。然后对打乱的数据集进行对半切片。

在该层，本文使用了基于三种不同核函数的 SVM，分别是线性核函数 linear、径向基核函数 RBF，以及多项式核函数 poly。它们的计算公式如下：

$$\text{linear:} \quad k(x,y) = x^T y + c \quad (4\text{-}6)$$

$$\text{poly:} \quad k(x,y) = (\alpha x^T y + c)^d \quad (4\text{-}7)$$

$$\text{RBF:} \quad k(x,y) = exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right) = exp(-\gamma\|x-y\|^2) \quad (4\text{-}8)$$

支持向量机的 c 值表现的是在线性不可分情况下对分类错误的惩罚程度。目标数据集为无标签数据，通过网格查找等 c 值的选择方式得到的最优 c 值不过是对伪标签的拟合程度，作用效果相对有限，还可能引入伪标签的噪声污染。在此便直接将三种不同核函数的 SVM 的 c 值统一设定为 0. 8，不进行最佳调优。用两个切片数据对三种 SVM 进行训练，便得到了 6 个 SVM 基分类器。

### 4.4.4 赋权输出层的模型构成与相关参数说明

在算法设计中，本文决定在该层对两种输出模式都进行尝试。在赋权模式中，使用 CART 决策树作为赋权分类器；在平均模式中，使用直接投票法进行输出。

决策数的不纯度计算采用基尼指数，不采用信息熵的原因是目标数据集规模不大，算法复杂度规模也相对较高，再采用对不纯度更为敏感的信息熵作为不纯度指标，很容出现过拟合现象。最大深度选择所有基分类器的数量，即通过误差稀释层获取的 6 个 SVM 基分类器以及深层特征提取层获取的 CNN 基分类器共 7 个。

## 4.5 实验过程

首先使用深度聚类模型完成对少标签数据集的分类，并根据生成的簇将其各个簇中数据复制到同一个文件夹下，遍历各个簇统计各个数中含两类标注数据的数量，将含同种标注数据最多的文件夹名重命名为标注数据的对应类名。图 4-2 为生成的文件夹，图 4-3，图 4-4，图 4-5 为各个簇内部分图像的展示。


图 4-2 深度聚类生成的文件夹

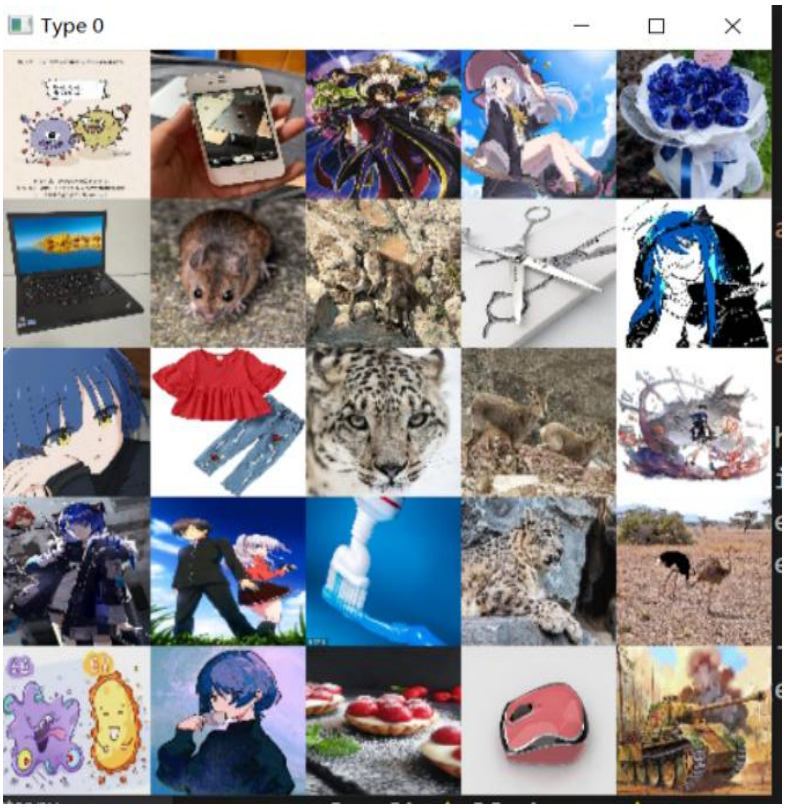


图 4-3　簇 0 包含的部分图像展示

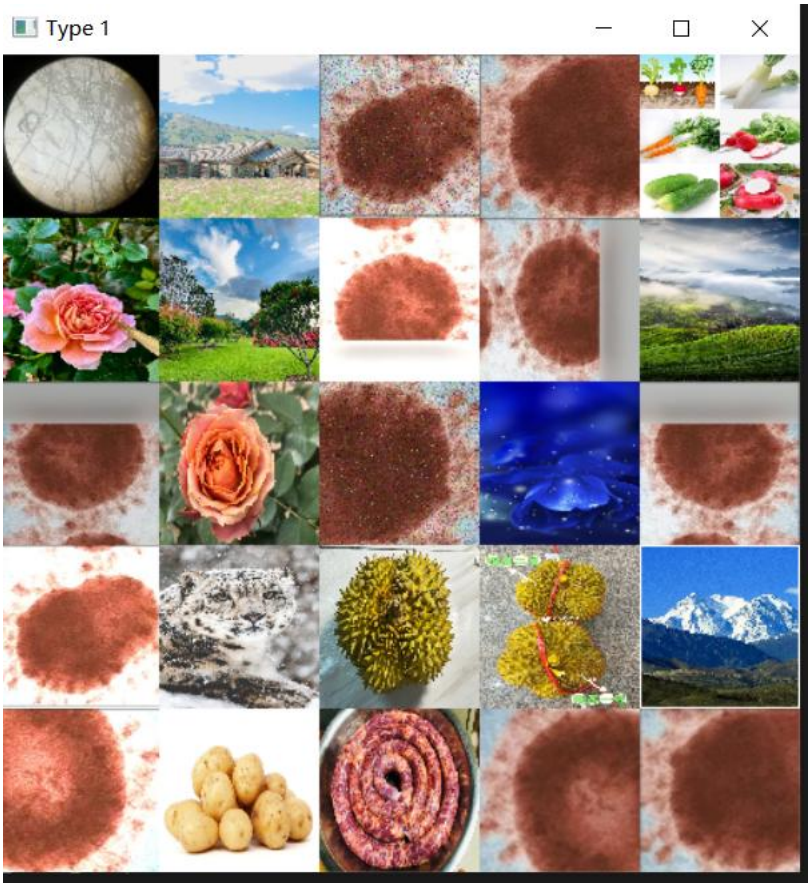


图 4-4　簇 1 包含的部分图像展示

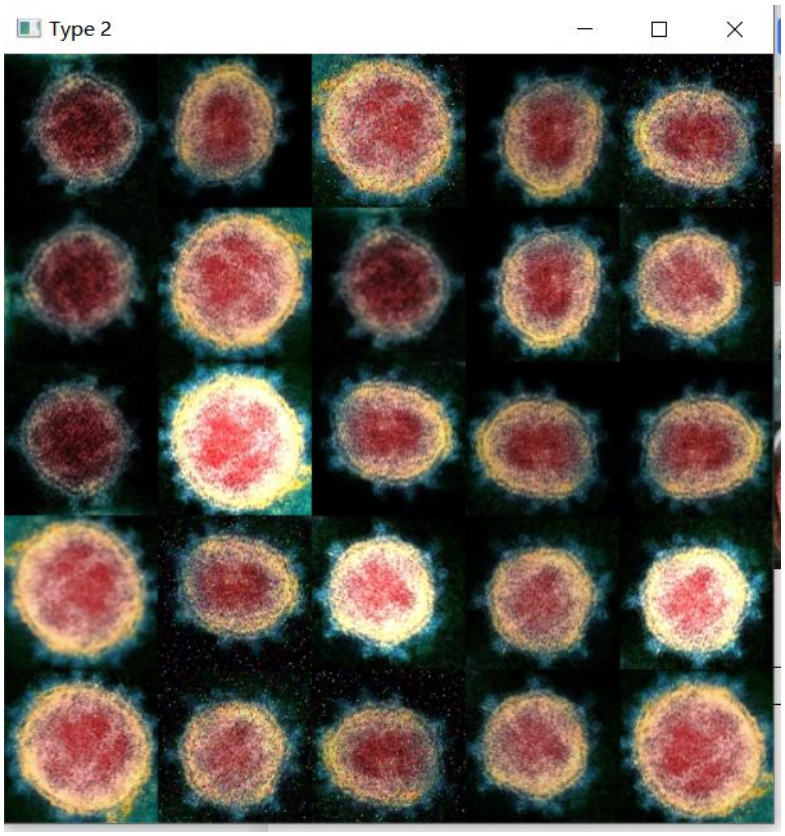


图 4-5　簇 2 包含的部分图像展示

其次使用 CNN 对深度聚类模型生成的经伪标签标注的数据集进行学习，得到了一个卷积神经网络模型，将其保存作为一个 CNN 基分类器。然后对 CNN 基分类器进行重载，将其输出设定为 CNN 的全连接层。将重载的模型对经伪标签标注的数据集进行预测，得到了特征数据集，将其保存。再将特征数据集对应的伪标签列表也进行保存。图 4-6 为迭代次数 30 次时，CNN 对经伪标签标注的数据集的拟合程度。

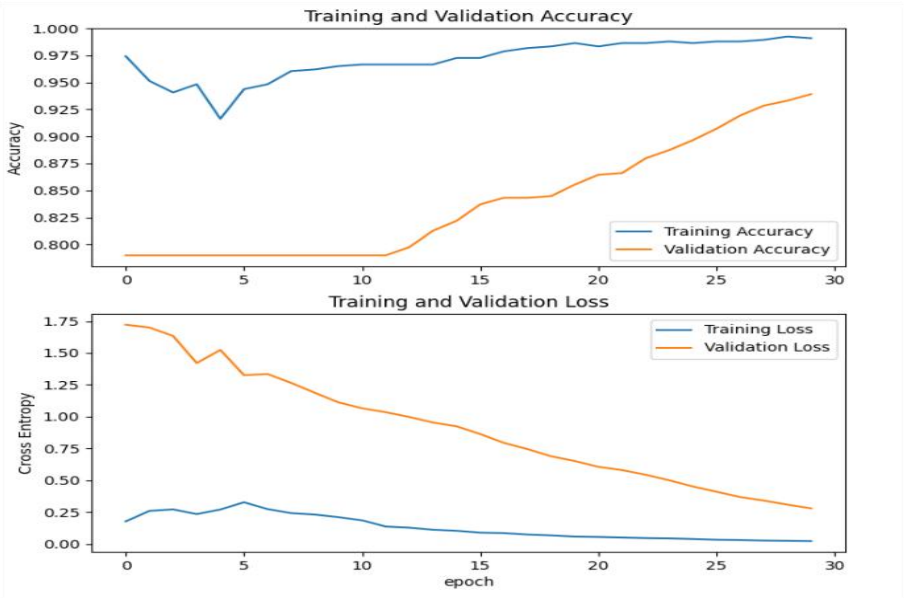


图 4-6　CNN 训练过程的精度与损失

接着读入保存的特征数据集和对应伪标签数据，对文件完成使用 sklearn 库中的 shuffle 函数完成对特征数据和对应伪标签的分布重构，再将其对半切片后分别输入到 3 个使用不同核函数的支持向量机中，训练出共 6 个基分类器，将它们保存。

将未分割的数据输入到 6 个 SVM 基分类器中，将它们的输出结果与 CNN 基分类器对所有样本做的标签预测合并到一个二维列表上，将这个二维列表每列上的 7 个行值做为该样本的 7 个特征数据，其对应伪标签作为该样本的标签。将这个二维列表和之前保存的特征数据集中特征数据对应的伪标签列表对决策树模型进行训练并保存生成的决策树模型作为赋权分类器。到此，UGD 的训练全部完成。表 4-2 以伪代码的形式为 UGD 子算法的学习步骤。

**表 4-2　UGD 子算法学习步骤**

| 算法：UGD 子算法 （学习过程） |
| --- |
| **输入：**半监督数据集 D，目标数据类别数 k<br>1、伪标签获取层（半监督数据集 D，目标数据类别数 k）：<br>2、　将半监督数据集 D 和要生成的簇的数量（k+1）输入到深度聚类算法中<br>3、　遍历生成的各个簇，统计各个簇下各类标注数据的数量<br>4、　将各个簇的簇名改为含某类标注数据最多的类名<br>5、　将各个簇下的数据以簇名打上对应的伪标签获得了伪标签数据集 W<br>6、深层特征提取层（伪标签数据集 W）：<br>7、　将第 5 步获得的伪标签数据集 W 输入到卷积神经网络（CNN）中<br>8、　训练卷积神经网络，导出 CNN 模型作为 CNN 基分类器<br>9、　重载 CNN 基分类器创建一个新模型命名为 TCNN，设定其输入为 CNN 模型的输入，输出为 CNN 模型的全连接层<br>10、　将伪标签数据集输入到 TCNN 模型中，获取其全连接层的特征数据<br>11、　获取第 10 步特征数据对应的伪标签，将其与特征数据一同构成特征数据集 T<br>12、　将特征数据集 T 用 shuffle 函数打乱其样本分布顺序其分布顺序构成重分布特征数据集<br>13、　将第 12 步中的重分布特征数据集均分成两个子集，命名为 T1 和 T2<br>14、误差稀释层（重分布特征数据集的两个子集 T1 和 T2）：<br>15、　构建三个基于不同核函数的支持向量机<br>16、　用第 13 步得到的 T1 与 T2 分别训练 3 个支持向量机得到 6 个 SVM 基分类器<br>17、赋权分类层（伪标签数据集 W，特征数据集 T）：<br>18、　将 11 步中得到的特征数据集 T 输入到 6 个 SVM 基分类器中<br>19、　将第 5 步中获得的伪标签数据集 W 输入到第 6 步得到的 CNN 基分类器中<br>20、　将第 18，19 步的输出作为特征，第 11 步中获得的对应伪标签列表作为标签构成赋权数据集 P<br>21、　构建决策树模型，用第 20 步中的赋权数据集 P 进行训练得到赋权分类器<br>**输出：**UGD 算法分类预测过程需要的 6 个 SVM 基分类器、1 个 CNN 基分类器、以及赋权分类器。 |

接下来是使用 UGD 算法的两种输出模式完成对测试集的分类预测。

首先读入测试集，重载 CNN 基分类器创建一个新模型命名为 TCNN，设定其输入为 CNN 基分类器的输入，输出为 CNN 基分类器的全连接层，将测试集输入到 TCNN 中，输出测试集上样本的特征数据。将特征数据分别输入到 6 个 SVM 基分类器完成分

类预测，再将测试集输入到 CNN 基分类器中完成分类预测，将 6 个 SVM 基分类器与 CNN 基分类器的预测结果合并为预分类数据集。

然后使用赋权输出模式（UGD-t）完成分类预测：将预分类数据集输入到由赋权分类器中进行最终的分类预测于结果输出。

最后使用平均输出模式（UGD-v）完成分类预测：使用投票器完成对预分类数据集的投票与输出，具体方式为输出预分类数据集上每列中出现次数最多的行值。自此，OGD 算法的预测工作完成。表 4-3 以伪代码的形式为 UGD 子算法的预测步骤。

**表 4-3　UGD 子算法预测步骤**

| **算法：UGD 算法（预测）** |
| --- |
| **输入：**无标签数据集 Y |
| 1、载入 CNN 基分类器、6 个 SVM 基分类器、赋权分类器 |
| 2、重载 CNN 基分类器创建一个新模型命名为 TCNN，设定其输入为 CNN 基分类器的输入，输出为 CNN 基分类器的全连接层 |
| 3、将无标签数据集 Y 输入到 TCNN 模型中得到特征列表 T |
| 4、将第 3 步得到的特征列表 T 分别输入到 6 个 SVM 基分类器中，将无标签数据集输入到 CNN 基分类器中 |
| 5、将第 4 步得到的 7 个分类结果列表拼接得到预分类列表 Y1 |
| 6、（UGD-v）输出第 5 步得到的预分类列表 Y1 中的每列中出现次数最多的元素 |
| 7、（UGD-t）将第 5 步得到的预分类列表 Y1 输入到 DCT 模型中完成分类 |
| **输出：**UGD 子算法两种输出模式下对无标签数据集 Y 中每个样本的预测标签 |

## 4.6　实验结果分析与对比

### 4.6.1　评价指标

由于半监督分类的难度较大，所使用的标注数据的比例也有较大差别，因此缺少一个统一的量化标准。考虑到 UGD 子算法的最终输出形式与监督学习算法一致，于是本文决定通过使用监督学习的评价指标来完成对基于 UGD 的半监督子算法分类性能的量化。所以在测试过程中使用了有标签的数据集完成对各种算法的分数评价。UGD 在学习过程中使用的是使用的标注数据的比例（1.25%）远低于当前半监督标准（10%）的少标签数据集，这是符合半监督学习标准的，在测试时使用有标签测试集只是为了量化算法的分类性能，在测试时 UGD 子算法未读入其标签信息，其标签信息仅用于与 UGD 输出预测标签进行对比，这和在实际分类场景中一致。同时，在训练过程中，本文只使用了少量标注数据完成伪标签获取层中聚类簇的命名，并未将标注信息用于模型训练过程，因此该评价方式也可体现出 UGD 算法对无监督数据集的划分性能。

采取的评价指标为准确率（Accuracy）、精准率（Precision）、召回率（Recall）以及 F1 Score，将各个类别结果综合起来的方式使用微平均（Micro-average）。该度量指标中，前三者取值在 0 和 1 之间，数值越接近 1 效果越好。F1 Score 为 Precision 与 Recall

的调和平均数。评价指标计算公式如下：

$$\text{Accuracy} = \frac{\sum_{i=1}^{k} TP_i}{\text{Total}} \tag{4-9}$$

$$\text{Precision} = \frac{\sum_{i=1}^{k} TP_i}{\sum_{i=1}^{k} (TP_i + FP_i)} = \frac{\sum_{i=1}^{k} TP_i}{\text{Total}} \tag{4-10}$$

$$\text{Recall} = \frac{\sum_{i=1}^{k} TP_i}{\sum_{i=1}^{k} (TP_i + FN_i)} = \frac{\sum_{i=1}^{k} TP_i}{\text{Total}} \tag{4-11}$$

$$\text{F1 score} = (1 + \beta^2) \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precison} + \text{Recall}} \tag{4-12}$$

其中 k 为类别数，Total 为样本总数，β用于度量 Precision 与 Recall 间的关系。若β＞1，Recall 有更大影响；β＜1，Precision 有更大影响。本次实验将β设为默认的 1，Precision 与 Recall 有相同的影响力。对某类别而言，TP 为正确的正例数，TN 为正确的反例数，FP 为错误的正例数，FN 为错误的反例数。

### 4.6.2 实验结果分析

表 4-4 UGD-v 和 UGD-t 在测试集上的表现

| | Accuracy | Precision | Recall | F1 score |
|---|---|---|---|---|
| UGD-v | 0.9898 | 0.9898 | 0.9898 | 0.9898 |
| UGD-t | 0.9898 | 0.9898 | 0.9898 | 0.9898 |

由表 4-4 可以看出，UGD 子算法的两种输出模式 UGD-v 和 UGD-t 在测试集上都有着优秀的表现，这不仅展现出了基于 UGD 划分算法的分类子算法在面对特定目标任务时有着较好的性能。同时，由于在训练过程中本文只将标注数据完成聚类簇的命名，并未将标注信息用于后续模型的训练过程，因此 UGD 子算法在测试集上的表现也可体现 UGD 算法构建的子算法在无监督数据集中亦可有着较好的划分性能。

UGD 子算法在测试集的良好表现初步体现出了 UGD 划分算法的设计理念和整体架构是行之有效的，能够凭此构建出将不同类别无标签数据有效区分的划分算法，还可依此完成对含少量标注数据的半监督数据集完成分类。在完成测试后，使用了带有标签数据的测试集来检验 UGD 子算法模型中各层次分类器的分类效果。

表 4-5 基分类器各项分数对比

| | Accuracy | Precision | Recall | F1 score |
|---|---|---|---|---|
| 深度聚类 | 0.7796 | 0.7796 | 0.7796 | 0.7796 |
| SVM1 | 0.9746 | 0.9746 | 0.9746 | 0.9746 |
| SVM2 | 0.9746 | 0.9746 | 0.9746 | 0.9746 |
| SVM3 | 0.9848 | 0.9848 | 0.9848 | 0.9848 |
| SVM5 | 0.9594 | 0.9594 | 0.9594 | 0.9594 |
| SVM6 | 0.9797 | 0.9797 | 0.9797 | 0.9797 |
| CNN | 0.9340 | 0.9340 | 0.9340 | 0.9340 |

由表 4-5 可以看出，伪标签获取层中使用的深度聚类算法在测试集的表现并不算优秀，而 UGD 子算法通过后续层次的处理达到了较为优秀的性能。

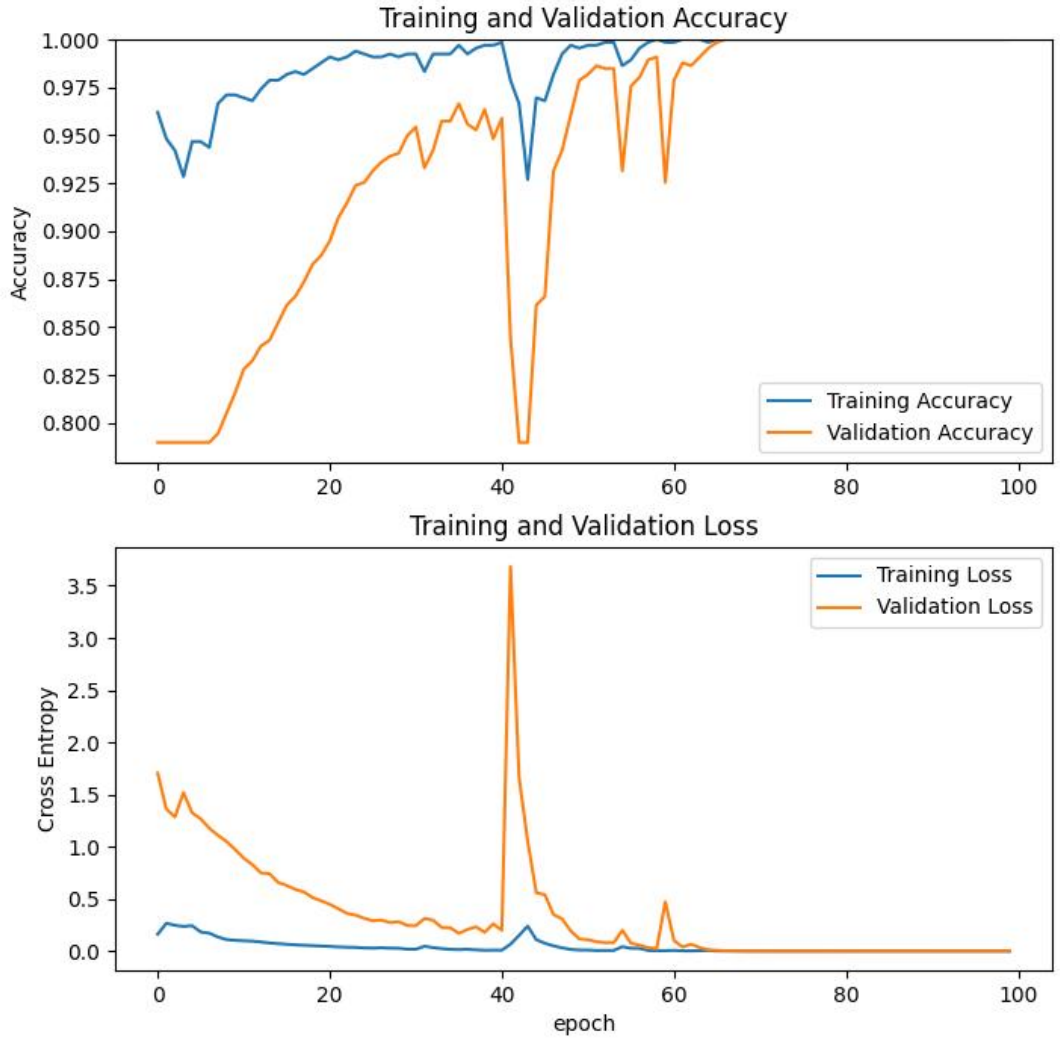


图 4-7　CNN 模型迭代次数为 100 下的精度与损失折线图

图 4-7 体现了 CNN 模型在不同迭代次数下提取的特征与伪标签的拟合程度，借此可以知道当迭代次数到达多少时，提取的特征数据与伪标签处于欠拟合、收敛、过拟合状态，方便后续测试 UGD 算法的鲁棒性。

表 4-6　UGD-v 与 UGD-t 在不同特征迭代次数下的精度

| epochs | UGD-v | UGD-t |
|---|---|---|
| 1 | 0.9645 | 0.9746 |
| 5 | 0.9442 | 0.9543 |
| 30 | 0.9848 | 0.9797 |
| 32 | 0.9898 | 0.9898 |
| 35 | 0.9898 | 0.9746 |
| 40 | 0.9898 | 0.9848 |
| 45 | 0.9898 | 0.9797 |
| 50 | 0.9898 | 0.9846 |
| 100 | 0.9898 | 0.9848 |

表 4-6 为 UGD 子算法两种输出模式在不同特征迭代次数下的精度。由表 4-6 和图 4-6 可以看出，在当前数据集下，UGD-v 的精度随特征提取层中 CNN 提取的特征与伪标签的拟合程度提高而提高，直到 CNN 收敛时涨停；而 UGD-t 的精度在模型收敛时达到峰值，在收敛前后震荡，精度曲线类似“W”形。

对于对伪标签学习的过拟合现象，两种模型都不受影响；对于欠拟合现象，UGD-t 表现得更优秀；对于一般情况，UGD-v 表现更好。仅提高对噪声标签的鲁棒性并不足以获得良好的性能，因为它还受到欠拟合的影响[32]。由表 4-4 可以看出，面对欠拟合现象，UGD 模型依旧有着不错的性能。综合来看对于无监督不均衡数据分类问题，UGD 算法具有较好的精度和鲁棒性。

对于这两者精度峰值都为 0.9898 的问题，本文认为可能是获取的特征信息深度不够的原因。对本算法的优化可以考虑通过增强 UGD 算法的划分性能，基于目标 1 的方式：增加卷积层的层数获取更多维度的特征，或者在误差稀释层加入其他有较大差异的分类算法；还可以考虑基于目标 2 和 3 的方式，通过增加支误差稀释层中基分类器的数量，比如将特征集切割成更多份然后使用 k 折交叉验证。当然，现实场景中还有一个有效的办法便是获取更多一些的标注数据，对赋权分类器进行训练，完成对各个分类器的精准赋权，排除劣质分类器的干扰。

### 4.6.3 与现有算法的比较

在对比实验中，本文分别对比了当今主流的几类分类算法。

本文使用了与 UGD 子算法相同，但经过人工标注的有标签训练集，对比了基于 Bagging 的随机森林算法（RF）、基于 Boosting 的梯度提升决策树算法（GBDT）以及常用于图像识别领域的 CNN。为保证算法评价的公平性，在实验过程中统一了特征编码的维度数，并对其他监督学习算法的参数进行了最佳调优。

**表 4-7 不同分类算法各项分数对比**

| | Accuracy | Precision | Recall | F1 score |
|---|---|---|---|---|
| UGD-v | 0.9898 | 0.9898 | 0.9898 | 0.9898 |
| UGD-t | 0.9898 | 0.9898 | 0.9898 | 0.9898 |
| RF | 0.9329 | 0.9329 | 0.9329 | 0.9329 |
| GBDT | 0.9340 | 0.9340 | 0.9340 | 0.9340 |
| CNN | 0.9645 | 0.9645 | 0.9645 | 0.9645 |

由表 4-7 可以看出，基于 UGD 的半监督分类算法相比当今主流分类算法拥有着较好的分类性能，通常半监督学习算法在分类性能上要逊色于监督学习算法，而通过本文提出的 UGD 划分算法，得到了在选定任务上分类性能强于监督学习算法的 UGD 子算法。即便面对少标签不平衡数据集这种较困难的分类目标，基于 UGD 框架获得的相应算法亦有着不错的分类效果。

# 总结与展望

本文针对大数据时代背景下的无监督划分问题展开了研究，通过研读相应理论知识，积极进行实验验证，勤于思考与提问，提出了一种新的无监督划分算法：无监督目标驱动算法。该算法在无标签不平衡数据集上表现出了强大的划分性能，显著提升了半监督学习算法在目标数据分类问题上的精度和鲁棒性，给无监督学习带来了更多可能性。

本文的主要工作如下：

（1）介绍了机器学习领域关于分类问题的现实背景与国内外研究现状。

（2）介绍了无监督划分问题的相关技术。

（3）介绍了本文提出的无监督目标驱动算法的构筑思想和算法框架。

（4）制作了一个无标签不均衡图像数据集用于对现实分类任务的模拟。

（5）根据目标任务和数据集设计并测试了基于无监督目标驱动算法的子算法。

（6）将本文提出的算法与当今主流分类算法进行对比分析并进行评价。

本文的主要创新点：

（1）通过集成学习的手段将无监督学习转化为了对带噪学习的优化问题。

（2）将分类算法对先验知识的依赖转化为工作者对计算机技术了解的依赖。

（3）提出了目标驱动算法在分类问题中的集成学习框架。

本文的不足之处：

（1）未对目标驱动算法的有效性做出数学上的解释与证明。

（2）受限于硬件设备与时间周期，未对无监督目标驱动算法做出更多尝试和优化。

（3）论文写作过程不够干脆，效果也不尽人意。

本文提出的目标驱动算法虽在实践过程中得到了检验，达成了预期效果但依然有许多需要改进或值得尝试的地方，希望在以后的研究中能够完成：

（1）设计一个目标驱动算法与 AI 大模型相连的接口，降低该算法对工作者计算机知识的要求。

（2）尝试将半监督信息引入目标驱动算法的赋权分类层，了解其在半监督领域的效果。

（3）尝试将目标驱动算法运用在更多不同的分类场景中，分析其在不同的数据集下的表现。

（4）尝试将目标驱动算法与置信学习结合起来。

# 参考文献

[1] 杜航原,张晶,王文剑. 一种深度自监督聚类集成算法[J]. 智能系统学报,2020,15(6):1113-1120.

[2] 谭茜成. 基于深度学习的无监督领域自适应图像分类算法研究[D]. 成都:四川师范大学,2022.

[3] 张浩,陆彦辉. 基于深度残差自编码器的无监督聚类算法[J]. 计算机仿真,2023,40(1):405-409.

[4] Ajay P, Nagaraj B, Kumar R A, et al. Unsupervised hyperspectral microscopic image segmentation using deep embedded clustering algorithm[J]. Scanning, 2022: 1200860

[5] Nurmaini S, Umi Partan R, Caesarendra W, et al. An automated ECG beat classification system using deep neural networks with an unsupervised feature extraction technique[J]. Applied sciences, 2019, 9(14): 2921.

[6] Akçakaya M, Yaman B, Chung H, et al. Unsupervised deep learning methods for biological image reconstruction and enhancement: an overview from a signal processing perspective[J]. IEEE Signal Processing Magazine, 2022, 39(2): 28-44.

[7] Li P, Wang H, Böhm C, et al. Online semi-supervised multi-label classification with label compression and local smooth regression[C]//Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence. 2021: 1359-1365.

[8] 李校林,陆佳丽,王韩林. 基于分类器链的多标签分类算法[J]. 计算机仿真,2022,39(6):380-385.

[9] 武红鑫,韩萌,陈志强,等. 监督和半监督学习下的多标签分类综述[J]. 计算机科学,2022,49(8):12-25.

[10] 甘井中,杨秀兰,吕洁,等. 人工智能中无监督学习算法综述[J]. 海峡科技与产业,2019(1):134-135.

[11] Zhai X, Oliver A, Kolesnikov A, et al. S4l: Self-supervised semi-supervised learning[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 1476-1485.

[12] Gao W, Zhou Z H. On the doubt about margin explanation of boosting[J]. Artificial Intelligence, 2013, 203: 1-18.

[13] 罗常伟,王双双,尹峻松,等. 集成学习研究现状及展望[J]. 指挥与控制学报,2023,9(1):1-8.

[14] Tüysüzoğlu G, Birant D. Enhanced bagging (eBagging): A novel approach for ensemble learning[J]. International Arab Journal of Information Technology, 2020, 17(4): 515-528.

[15] Seng Z, Kareem S A, Varathan K D. A neighborhood undersampling stacked ensemble (NUS-SE) in imbalanced classification[J]. Expert Systems with Applications, 2021, 168: 114246.

[16] 汪霞. 可靠聚类集成算法研究[D]. 合肥:安徽大学,2022.

[17] 钱辉. 基于深度卷积神经网络的无监督领域适配算法研究[D]. 保定:河北大学,2020.

[18] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. science, 2006, 313(5786): 504-507.

[19] Lee H, Ekanadham C, Ng A. Sparse deep belief net model for visual area V2[J]. Advances in neural information processing systems, 2007, 1416-1423.

[20] Rifai S, Vincent P, Muller X, et al. Contractive auto-encoders: Explicit invariance during feature extraction[C]//Proceedings of the 28th international conference on international conference on machine learning. 2011: 833-840.

[21] 邓祥,俞璐. 深度聚类算法综述[J]. 通信技术,2021,54(8):1807-1814.

[22] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks[J]. Communications of the ACM, 2020, 63(11): 139-144.

[23] Roy K, Jaiswal A, Panda P. Towards spike-based machine intelligence with neuromorphic co

mputing[J]. Nature, 2019, 575(7784): 607-617.
[24] Sun Y, Yen G G, Yi Z. Evolving unsupervised deep neural networks for learning meaningful representations[J]. IEEE Transactions on Evolutionary Computation, 2018, 23(1): 89-103.
[25] 李嘉恒. 基于深度神经网络的无监督异源遥感图像变化检测[D]. 西安:西安电子科技大学,2020.
[26] Cao Y, Guan D, Huang W, et al. Pedestrian detection with unsupervised multispectral feature learning using deep neural networks[J]. information fusion, 2019, 46: 206-217.
[27] 王晓莉, 薛丽. 标签噪声学习算法综述[J]. 计算机系统应用, 2020, 30(1): 10-18.
[28] 华南理工大学. 一种适用于无标签不平衡数据流的在线主动学习方法:CN201910001840.2[P]. 2019-05-24.
[29] 许耀奎. 基于无监督深度学习的单细胞 RNA-seq 数据分析[D]. 青岛:青岛科技大学,2022.
[30] Yang C, Yin X, Hao H, et al. Classifier ensemble with diversity: Effectiveness analysis and ensemble optimization[J]. Acta Automatica Sinica, 2014, 40(4): 660-674.
[31] 刘春容,宁芊,雷印杰,等. 改进残差神经网络在遥感图像分类中的应用[J]. 科学技术与工程,2021, 21(31):13421-13429.
[32] Wang L, Zhu T, Kumar N, et al. Attentive-Adaptive Network for Hyperspectral Images Classification With Noisy Labels[J]. IEEE Transactions on Geoscience and Remote Sensing, 2023, 61: 1-14.

# 致　谢

四年时光如白驹过隙，若不是在敲着这篇论文，我或许还察觉不到即将毕业的事实。

忆往昔岁月，感受着时间在指尖流淌，不由感慨万千，原来在这四年的时光中有这么多给予我帮助与感动的人。在此，我诚挚地向你们表示感谢。

感谢我的导师[redacted]老师。[redacted]老师在课堂上认真负责，课堂下亲切温和，对我提出问题总是耐心地解答。[redacted]老师的认真负责与平易近人深深得打动了我，可以说正是[redacted]老师激发了我对机器学习领域的兴趣。在毕业设计和论文写作的过程中，[redacted]老师耐心的解答我的疑问，指引我的方向，指出我的错误，包容我的愚蠢，给了我巨大的帮助，再次感谢[redacted]老师！

感谢[redacted]教授。第一次见到[redacted]教授是在申请补修课程的时候，还记得那天[redacted]教授早早就来到了课堂等待着同学们进来上课，在等待的过程中在背着单词，我情不自禁也打开单词软件背了起来，直到快上课我才向[redacted]老师提交了跟班申请；后来我才发现，[redacted]老师一向如此，逐渐我也养成了这一习惯。感谢[redacted]老师。

感谢学业导师[redacted]老师。自转专业到这个新班级，我便感受到了她对我们的关爱，[redacted]老师总是在积极的影响我们，关注我们的学习和生活，给我们提供信息与资源，鼓励我们追寻自己想要的人生。[redacted]老师的引导令我更明悟了内心所想，感谢[redacted]老师。

感谢[redacted]老师以及[redacted]老师。[redacted]老师和[redacted]老师传授了我许多数据挖掘与处理的技巧，也很乐于给同学们答疑，在接近论文答辩的时间，老师们都非常忙，[redacted]老师还抽出时间解答我的疑惑，感谢[redacted]老师和[redacted]老师。

感谢我的室友们支持与陪伴，特别感谢[redacted]同学和[redacted]同学对我论文格式上的指导和帮助。

最后，感谢我的家人一直以来对我的支持与信任，感谢我的朋友们对我的关心和照顾，感谢师长们对我的鼓励与教导，感谢一切美好的事物，赞美太阳！