软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software [doi: 10.13328/j.cnki.jos.006948]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

避免近期偏好的自学习掩码分区增量学习*

姚红革[1], 邬子逸[1], 马姣姣[1], 石 俊[1], 程嗣怡[2], 陈 游[2], 喻 钧[1], 姜 虹[1]

[1](西安工业大学 计算机科学与工程学院, 陕西 西安 710021)

[2](空军工程大学 航空工程学院, 陕西 西安 710051)

通信作者: 邬子逸, E-mail: wuziyi@st.xatu.edu.cn

摘 要: 遗忘是人工神经网络在增量学习中的最大问题, 被称为“灾难性遗忘”. 而人类可以持续地获取新知识, 并能保存大部分经常用到的旧知识. 人类的这种能持续“增量学习”而很少遗忘是与人脑具有分区学习结构和记忆回放能力相关的. 为模拟人脑的这种结构和能力, 提出一种“避免近期偏好的自学习掩码分区增量学习方法”简称ASPIL. 它包含“区域隔离”和“区域集成”两阶段, 二者交替迭代实现持续的增量学习. 首先, 提出“BN稀疏区域隔离”方法, 将新的学习过程与现有知识隔离, 避免干扰现有知识; 对于“区域集成”, 提出自学习掩码(SLM)和双分支融合(GBF)方法. 其中SLM准确提取新知识, 并提高网络对新知识的适应性, 而GBF将新旧知识融合, 以达到建立统一的、高精度的认知的目的; 训练时, 为确保进一步兼顾旧知识, 避免对新知识的偏好, 提出间隔损失正则项来避免“近期偏好”问题. 为评估以上所提出方法的效用, 在增量学习标准数据集CIFAR-100和miniImageNet上系统地进行消融实验, 并与最新的一系列知名方法进行比较. 实验结果表明, 所提方法提高了人工神经网络的记忆能力, 与最新知名方法相比识别率平均提升5.27%以上.

关键词: 增量学习; 灾难性遗忘; 持续学习; 自学习掩码; 近期偏好; 区域隔离

中图法分类号: TP18

中文引用格式: 姚红革, 邬子逸, 马姣姣, 石俊, 程嗣怡, 陈游, 喻钧, 姜虹. 避免近期偏好的自学习掩码分区增量学习. 软件学报. http://www.jos.org.cn/1000-9825/6948.htm

英文引用格式: Yao HG, Wu ZY, Ma JJ, Shi J, Cheng SY, Chen Y, Yu J, Jiang H. Recency Bias-avoiding Self-learning Mask-based Partitioned Incremental Learning. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/6948.htm

Recency Bias-avoiding Self-learning Mask-based Partitioned Incremental Learning

YAO Hong-Ge[1], WU Zi-Yi[1], MA Jiao-Jiao[1], SHI Jun[1], CHENG Si-Yi[2], CHEN You[2], YU Jun[1], JIANG Hong[1]

[1](School of Computer Science and Engineering, Xi'an Technological University, Xi'an 710021, China)

[2](Aeronautics Engineering College, Air Force Engineering University, Xi'an 710051, China)

Abstract: Forgetting is the biggest problem of artificial neural networks in incremental learning and is thus called "catastrophic forgetting". In contrast, humans can continuously acquire new knowledge and retain most of the frequently used old knowledge. This continuous "incremental learning" ability of human without extensive forgetting is related to the partitioned learning structure and memory replay ability of the human brain. To simulate this structure and ability, the study proposes an incremental learning approach of "recency bias-avoiding self-learning mask (SLM)-based partitioned incremental learning", or ASPIL for short. ASPIL involves the two stages of regional isolation and regional integration, which are alternately iterated to accomplish continuous incremental learning. Specifically, this study proposes the "Bayesian network (BN)-based sparse regional isolation" method to isolate the new learning process from the existing knowledge and thereby avoid the interference with the existing knowledge. For regional integration, SLM and dual-branch fusion (GBF) methods are proposed. The SLM method can accurately extracts new knowledge and improves the adaptability of the network to new knowledge, while the GBF method integrates the old and new knowledge to achieve the goal of fostering unified and high-precision cognition. During training, a regularization term for Margin Loss is proposed to avoid the "recency bias", thereby ensuring the further

---

* 作者邬子逸、姚红革对本文同等贡献

收稿时间: 2022-08-31; 修改时间: 2023-01-15; 采用时间: 2023-04-06; jos在线出版时间: 2023-09-13

balance of the old knowledge and the avoidance of the bias towards the new knowledge. To evaluate the effectiveness of the proposed method, this study also presents systematic ablation experiments performed on the standard incremental learning datasets CIFAR-100 and miniImageNet and compares the proposed method with a series of well-known state-of-the-art methods. The experimental results show that the method proposed in this study improves the memory ability of the artificial neural network and outperforms the latest well-known methods by more than 5.27% in average identification rate.

Key words: incremental learning; catastrophic forgetting; continuous learning; self-learning mask; recency bias; regional isolation

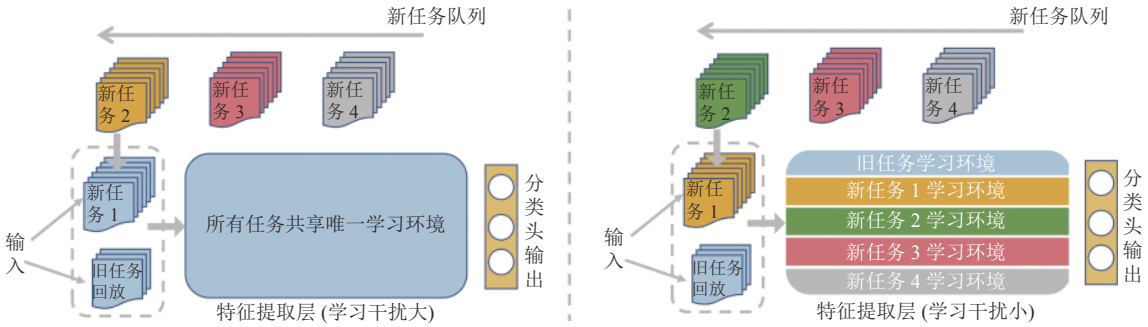
1 引 言

与生物神经网络相比较, 作为人工神经网络代表的深度学习网络存在诸多缺陷, “灾难性遗忘”[1]就是其中主要缺陷之一, 即深度学习网络总是在学习新知识后忘记旧知识. 该缺陷伴随人工神经网络出现至今一直存在, 难以解决[2]. 尤其是在增量学习过程中, 该问题的表现尤其突出.

终身式、增量式的学习能力是人类最重要的能力之一, 人类能够很好地处理“灾难性遗忘”, 不断地吸纳新知识, 整合、优化旧知识而不遗忘[3]. 神经科学发现, 人类这种超凡的增量学习能力与人脑的分区学习结构: 海马系统和新皮质系统有关. 海马系统表现出短期的适应性, 并允许快速学习新知识, 而新学的知识又会随着时间的推移被放回到新皮质系统, 以保持长期记忆[4]. 海马体区和新皮质区的协调配合, 可以在快速学习的进程中最小化知识间的干扰, 以获取持续学习的能力[5]. 另一方面, 人类的增量学习能力还受益于海马系统的记忆重放. 越来越多的证据表明海马体可以在编码和回放模式之间快速切换, 促进新皮层上长期记忆的形成与巩固. 首先, 海马体在清醒时会快速编码记忆, 然后在离线时段内如睡眠, 通过记忆重放重新激活记忆轨迹, 促进新知识向长期存储状态的转移[6].

随着神经科学研究的深入, 人们也不断探索模拟人脑记忆形成过程进行增量学习的可行性. 早期的增量学习探索集中于基于正则化方法的增量学习和基于回放的增量学习. 其中基于正则化的方法, 如 LwF[7]、EWC[8]、SI[9], 通过引入附加损失来修正梯度, 保护模型所学的旧知识, 提供在特定条件下缓解灾难性遗忘的方法. 但是, 这类方法通常需要我们在旧任务和新任务的性能表现上做出权衡, 限制了网络参数灵活性, 优化进程在取得部分推进后提前停止, 而且部分方法的梯度修正训练过程过于复杂[10]. 受人脑记忆重放功能的启发, 回放方法[11]得到了深切的关注, 基于记忆回放的一系列增量学习模型被相继提出[12]. 它通过为系统增加一个存储模块来保存先前的数据, 并定期回放先前知识与新样本的交叉数据[13], 如图 1(a) 所示.

然而, 大部分基于回放结构的方法有一个严重的缺陷是它们需要显式存储旧任务信息, 这将导致较大的工作内存需求, 此外, 仅依靠回放而不考虑新学习对旧知识的覆盖和干扰, 并不能在新、旧知识上均取得良好的认知. 因此近年来的一些方法中, 研究人员开始尝试基于动态结构的增量学习, 将回放技术更多地作为一种辅助手段保留下来[14]. 在新任务到来时, 使用增设的网络架构空间执行新学习, 显式地分离新、旧学习的环境, 不断地扩展神经网络架构来降低新学习对旧知识的覆盖和干扰[15], 如图 1(b) 所示.



(a) 基于回放的增量学习 (b) 基于动态结构的增量学习

图 1 两种增量学习方式对比

但基于动态结构的增量学习研究, 又引起了新的问题: 1) 随着任务数量的增加, 神经网络架构的存储占用也线性增加, 这直接降低模型的可伸缩性[16]; 2) “分类妥协”问题, 即深度网络模型因为保护旧知识而限制了自身的可塑性, 对于新知识的适应性降低[15]; 3) 因为需要集成网络各区域的表达, 就要先对所有知识进行准确提取, 其提取过程存在信息损失[17]; 4) “近期偏好”问题, 是指由于新、旧数据不平衡以及增量学习场景下的 Softmax 分类器能力较弱等原因, 模型无法准确区分新、旧类别的决策边界, 其预测偏向于新类[18].

针对上述问题, 本文模仿人脑的分区学习和记忆重放, 构建了一个基于分区学习和记忆重放的神经网络模型: “避免近期偏好的自学习掩码分区增量学习”, 简称 ASPIL. ASPIL 在新知识编码的同时回放旧知识来模拟人脑的记忆重放, 增加了模型的可伸缩性, 同时避免了“近期偏好”, 巩固了旧知识. 另外, 设计了一个自学习掩码模块, 提高对知识提取的准确度, 便于新旧区域知识表达的集成.

本文总体贡献归纳如下.

(1) 基于人脑分区学习的启发, 提出了分离-集成的两阶段的增量学习机制. 它通过迭代式的区域分离和整合操作给固定容量的网络赋予了可伸缩能力, 在没有带来额外的存储消耗情况下, 实现了所有类增量任务的可训练空间的公平分配.

(2) 第二是模仿海马体, 实现记忆重放功能. 提出一种记忆回放和知识蒸馏方法, 在利用神经网络对新知识编码的同时回放旧知识、施加蒸馏约束, 巩固旧知识.

(3) 设计一个可微分的“自学习掩码模块 (self-learning mask, SLM)”(详见第 3.2.1 节), 它利用掩码将局部神经元与特定任务适配, 实现了准确的知识提取, 提高网络集成后的表征能力.

(4) 针对“近期偏好”问题, 提出间隔损失 Margin Loss 来分离当前类和过去类的决策边界, 使“类内聚集, 类间分离”, 避免新知识对旧知识的覆盖.

最后, 在 CIFAR-100[19]和 miniImageNet[20]数据集上, 与一些领域先进方法进行了比较 (见第 4.2 节), 并系统地进行消融实验以评估 ASPIL 各组成部分的必要性. 在 5 阶段和 10 阶段的增量设置下, 我们的方法始终优于基线, 获得了 5.27% 以上的平均精度提升.

本文第 2 节主要介绍目前已有的增量学习研究现状, 总结并对比了各种方法的优点以及其局限性. 第 3 节先概述本文方法 ASPIL 的总体结构, 之后详细介绍 ASPIL 各组件的工作原理和训练方式, 包括区域隔离, 区域集成, 网络训练流程和损失函数设计. 第 4 节为实验部分, 包含 3 个部分: 实验设置、同领域先进方法的比较以及消融实验. 第 5 节总结回顾. 附录部分是对正文部分细节的补充说明和详细展开, 附录与正文的关联之处已在正文中标识.

2 相关工作

2.1 蒸馏与正则化

通过给新任务的蒸馏损失函数施加正则化约束来保护旧知识不被新知识覆盖[7]. 它利用参数的冗余, 在学习新数据之前对需要保留的历史信息进行正则, 来平衡记忆与更新. Li 等人提出的 LwF[7]将知识蒸馏[21]与微调[22]相结合, 知识蒸馏用来更好地保留旧任务的知识. EBLL[23]则为每个任务创建额外的特征蒸馏编码器, 该编码器重构的特征既接近原始特征, 对原始任务也有很好的可见性. EWC[8]及其改进版本用 Fisher 信息矩阵对网络参数进行约束, 降低模型的遗忘度[24]. 遗憾的是, EWC 方法及其改进版本无法协调新、旧任务争用关键参数[25]. Zenke 等人[9]在 2017 年提出了一种在线计算权重的 SI 方法, 在训练期间, 权重参数值根据它对损失的贡献而动态变化. MAS[26]方法计算的是参数改变对模型输出的影响, 并据此推测权重. 近年来, 更多的作品 (Bic[27], LwM[28], DMC[29]) 使用蒸馏损失来防止遗忘. 如 Dhar 等人[28], 提出了一种基于注意力机制映射的梯度流信息蒸馏方法. 知识蒸馏与增量学习的领域广泛地交汇在一起, 在本文中, 我们用知识蒸馏方法来促进多个增量过程之间的知识转移.

2.2 重放

通过在增量学习中储存旧知识来提醒模型, 是减轻灾难性遗忘的最有效方法之一[15]. 因此许多先进的增量学习方法补充了样本回放, 本文也是如此. 不过, 文献 [18] 的研究工作指出由于样本回放方法中新、旧数据不平衡

以及增量学习场景下的 Softmax 分类器能力较弱等原因, 导致记忆回放的过程中还存在着严重的“近期偏好”问题. 最近的研究方法, 倾向于关注“近期偏好 (recency bias)”问题, 包括 LUCIR[18]、Bic[27]和 E2E[30]. Mai 等人[31]提出的 SCR 同样认为虽然记忆重放已显示出良好的效果, 但常用的 Softmax 分类器在“增量学习”中引起的“近期偏好”仍是一个未解决的挑战. 本文深切关注了这一问题, 首先对分类器中的新、旧任务的权重向量进行了归一化处理, 减少数值大小引起的偏置问题. 其次, 设计了一个间隔损失正则项提升分类能力, 在学习当前任务的特征的同时, 避免偏向最近任务的特征表示. 不同于 ABD[32]通过计算新任务上的局部交叉熵损失来分离新、旧任务, 我们的方法受到 LUCIR[18]中的类间隔损失设计启发, 显式地最大化新、旧任务特征的空间距离. 第 4.2 节的实验结果表明, 由于这些改进, 我们提出的方法的性能优于这两种.

2.3 动态结构方法

渐进式框架作为动态结构方法的代表, 吸引了一些研究者的目光[33]. 这些工作允许在一定程度上扩增网络容量来逐步构建深度增量模型, 同时保持网络结构的紧凑性. 但是, 随着学习任务数量的增加, 网络的内存占用和复杂性也线性增加. 考虑到渐进式框架的结构复杂度困扰, 近年来的部分增量学习作品开始尝试挖掘固定容量网络的潜力, 为未来任务设置一个预设的、持续的扩展. 例如, BNWF[34]在固定容量网络内部按任务量公平分配张量空间, 但其单个任务可训练空间与任务数量呈负相关, 容易欠拟合. LPS[14]综合应用掩码和剪枝方法隔离任务空间、提倡知识共享, 但 LPS 对于新任务的可训练空间是持续递减的. 此外, 按任务量分区的方法使得 LPS 的分区集成难度增加, 充满挑战. S&B[15]同样使用固定容量的动态结构抵抗“灾难性遗忘”, 辅助蒸馏损失迁移旧知识, 同时和 LPS 一样鼓励跨任务的知识共享. 但是, S&B 方法未关注“近期偏好”现象, 且任务隔离机制较为复杂, 不利于训练.

本文方法同样在挖掘固定容量网络的潜力方面做出了努力, 延续了 LPS[14]、S&B[15]、BNWF[34]等类增量学习方法的任务隔离训练机制, 同样鼓励新、旧任务间的知识共享. 不断地等量分配新、旧知识存储区域来隔离开新、旧知识学习环境, 而持续的空间集成过程也仅限于新、旧知识两个分区. 这种两阶段的持续隔离与集成设置, 给固定容量的网络赋予了可伸缩能力, 既降低了学习干扰和存储消耗, 又实现了所有类增量任务的可训练空间的公平分配. 我们的动态结构方法的有效性在第 4.3.1 节通过了实验验证, 且取得了优异于 S&B 的成绩 (比较实验参见第 4.2 节).

2.4 稀疏化

稀疏化技术[35]是我们使用的一个关键技术, 它将网络的学习参数隔离, 以此来抑制“灾难性遗忘”. 其本质上是以神经网络过参数化性质为理论支撑, 利用剪枝技术减少网络的冗余参数, 以实现参数隔离, 达到减少遗忘的目的[15].

CPG[33]使用基于参数 (权重矩阵) 的稀疏性来减少遗忘, 并可以保持模型的紧凑型. 文献 [36] 采用精细的逐层修剪比例, 并基于 ReLU 层激活值的稀疏性在多层感知器上实现了以细粒度的神经元稀疏性为重点的终身学习方法. 逐层的精细修剪比例存在启发性且不利于扩展到深层次的 CNN 网络. 此外, 细粒度的稀疏模式存在不规则和硬件不友好的问题. 因此, 我们建议学习一个跨不同层的特征通道的全局排序以进行分层自适应剪枝. 我们的方法基于 BN 层的权重参数的稀疏性引导跨不同层的特征通道的稀疏性, 这种粗粒度稀疏性带来了更规则的稀疏性模式, 便于硬件加速, 且易于扩展到更深层次的 CNN 网络.

2.5 分离新类和旧类的决策边界

由于旧类样本的缺乏, 以及旧类的特征与权重向量之间的关系无法很好地保留, 导致增量分类器偏向于新类的决策和旧类之间的混淆. 为了避免新、旧类之间的歧义, 引入了一个间隔损失函数 Margin Loss, 来分离当前类和过去类的决策边界. 间隔损失函数的形式多样, 包括 0-1 损失、Hinge Loss、指数损失、交叉熵损失等. 经典的支持向量机算法[37]使用 Hinge Loss 实现多分类求解, 它将样本真实标签和预测标签的乘积作为函数间隔, 通过最大化函数间隔来分离新类和旧类的决策边界, 提高分类可信度. 本文提出的 Margin Loss 针对旧类样本的预测概率, 它将旧类样本预测为新类的概率和旧类样本正确归类的概率之间的差值作为函数间隔, 并选取批量运行的函数间隔中的较大值作为待优化的硬样本. 通过最小化该函数间隔, 来分类新类和旧类的决策边界, 抑制“近期偏好”.

3 ASPIL 网络结构

本文提出了一个可持续隔离与集成的增量学习网络 ASPIL, 如图 2 所示.

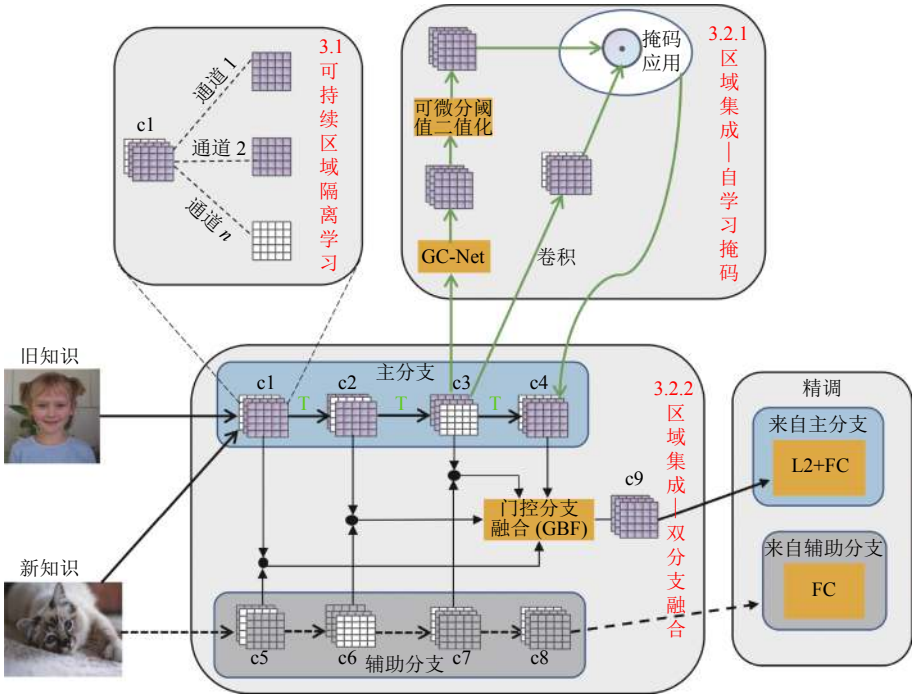


图 2 ASPIL 方法结构图

图 2 中黑实线为主分支线路, 黑虚线为辅助分支线路. 新知识和旧知识一起训练主分支, 仅用新知识训练辅助分支. c1–c9 为通用卷积结构, 构建示意图见图 3. c1–c4 中的紫色块和 c5–c8 中的黑色块为区域分离后用来保留旧知识的通道, 白色块为预留的空闲通道并被用来接收新知识. c1–c4 间的变换 T 可以等效地由子过程 2 (绿色实线) 代替. L2 代表使用 L2 范数归一化处理数据, 目的是消除数据单位影响和加速收敛, FC 是全连接层. “精调”过程冻结除全连接层 (分类头) 之外的所有参数并训练, 通过精调稳固学习效果、提升识别精度.

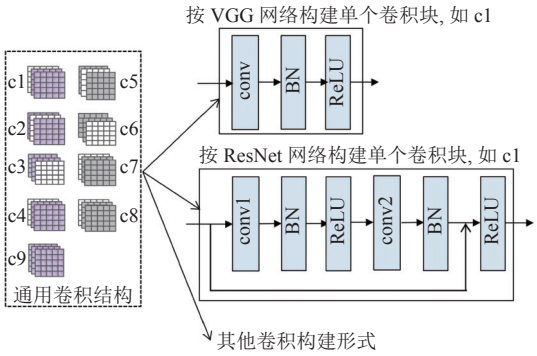


图 3 ASPIL 卷积块构建示意图

图 3 中 c1–c9 都是卷积结构, 可以按照 VGG 网络形式实现该卷积结构, 例如: 先是卷积层, 再是 BN 层, 最后是 ReLU 激活层. 也可以按照 ResNet 网络的残差结构形式实现该卷积结构.

ASPIL 包含“区域隔离”和“区域集成”两阶段, 其中“区域集成”由两个重要子过程构成: 自学习掩码和双分支融合. “可持续区域隔离学习”过程利用通道剪枝技术等量分配新、旧类别任务的训练空间. “区域集成-自学习掩码”利用可微分注意力方式生成掩码来强化特征提取. “区域集成-双分支融合”利用 GBF (gated branch fusion) 模块实现决策融合以提高新知识适应性, 辅助分支单独留出为新知识单独训练所用. 我们将在第 3.1 节和第 3.2 节详细阐述“区域隔离”和“区域集成”.

3.1 可持续区域隔离学习

区域隔离的目的是在保留原有知识的基础上, 尽可能独立地学习新知识, 详见图 4. 它分为“区域隔离”和“整合新知识”. 前者是为新知识留出通道, 避免与旧知识混淆; 后者是为了在留出的新知识通道里学习新知识.

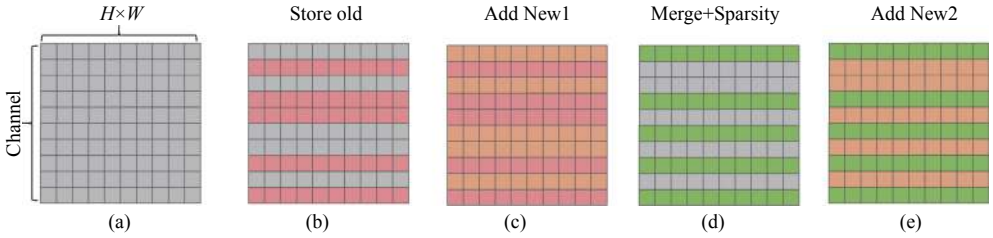


图 4 可持续隔离学习

图 4 中, W、H 分别代表卷积块通道的宽和高, 灰色区域代表可用于权重参数学习的空闲区域. 图 4(a) 是模型初始化状态, 全空闲 (全灰); 用 BN 层权重诱导网络稀疏化方法, 经过一段时间训练, 模型中 50% 的通道用于存储 base(Old) 知识. 图 4(b) 中红色块代表选中的用于存储 base(Old) 知识的通道, 其余 50% 的灰色所示通道为空闲可学习状态; 接下来利用图 4(b) 剩下的 50% 空闲区域学习新知识 New1, New1 存储在橙色所示通道中, 见图 4(c); 接下来在蒸馏损失函数的指导下, 对图 4(c) 中被通道隔离的新、旧知识重新连接的同时, 执行一个同步的稀疏训练 (同图 4(a)–(b) 的过程), 再次得到一个拥有 50% 空闲通道的状态图 4(d); 图 4(d) 再加入橙色新知识 New2 形成图 4(e); 图 4(d)–(e) 与图 4(b)–(c) 过程一致. 即图 4(c)–(d) 过程不断迭代, 进行持续的类增量学习.

3.1.1 BN 稀疏区域隔离

我们希望为新知识不断开辟新的学习区域, 与旧知识隔离, 这样新学习参数更新的过程不会对旧知识的记忆造成影响. 一个自然的想法是从网络中划分出旧知识分区和新学习分区. 为此, 我们采用了最初用于实现高质量模型压缩和构建轻量化网络的结构化通道剪枝[38], 基于通道对输出的重要性执行区域划分. 据调研, 我们将该方法应用到增量学习领域中是首次. 本文以卷积层的通道特征所后接的 BN 层乘法权重因子 γ 代表通道重要性. 它背后的思想是通道特征相乘的权重因子 γ 较小时, 该通道相对于输出的重要性越低, 见公式 (1). 因此, 可以移除具有较低 γ 值的通道, 而保留具有较高 γ 值的通道作为旧知识分区, 这不会影响原始知识的表达. 并且移除的低 γ 值通道可以被赋予新的功能, 即用作新学习分区.

$$\hat{h}=\frac{h^m-\mu_B}{\sqrt{\sigma_B^2+\varepsilon}};\quad h^{m+1}=\gamma\hat{h}+\beta \tag{1}$$

BN 层使用统计特征值 μ_B 和 σ_B^2 对通道输入 h^m 进行归一化, 然后通过在 BN 中学习的权重 γ 和偏置 β 恢复归一化之前的数据特征, 以获得通道输出 h^{m+1}.

本文是通过调节 γ 的分布稀疏度来控制通道的重要性. 基于此思想, 先对 BN 权重因子 γ 进行稀疏训练, 然后让所有卷积层所有通道的 γ 因子参与全局 rank 函数排序. 前 50% 的 γ 值所对应的通道设置为旧知识区域. 后 50% 的 γ 值所对应的通道不再用于旧知识表达, 可用于适应新的学习, 见图 5.

图 5 中第 m 个卷积层的输入通道 h^m 与通道权重因子 γ 相乘得到输出通道 h^{m+1}. 稀疏化训练后, 初始化网络的部分 γ 因子接近于 0, 与之相连的输入通道, 例如 $h^{m,1}$ 和 $h^{m,4}$ 被用于新学习. 其余的橙色通道保存旧知识, 区域隔离完成.

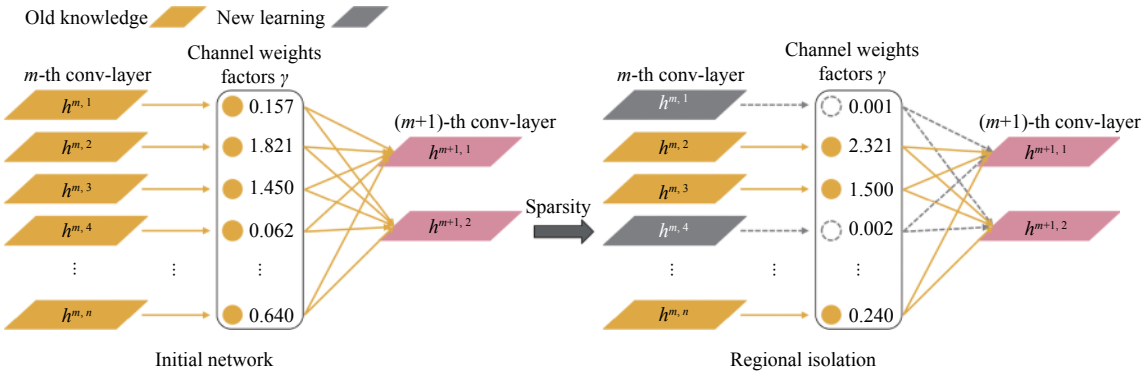


图 5 BN 稀疏区域隔离

3.1.2 整合新知识

整合新知识是利用区域隔离得到的新知识通道来学习新知识. 为了独立学习新知识且不干扰旧知识, 我们通过固定旧知识区域参数的方式来阻止对旧知识的干扰, 同时使用交叉熵损失更新新学习区域的参数.

为了固定旧知识区域的参数, 采用了权重冻结方法. 具体地, 对目标函数施加权重冻结约束. 权重冻结约束将分配给旧知识通道的卷积核权重参数的梯度设置为零. 这意味着梯度置零的通道特征不会参与参数更新, 旧知识区域不受影响, 也就使得模型避免丢失执行旧任务的能力.

因此, 带约束的目标函数被定义为:

$$\begin{cases} \arg\min\limits_{\theta^t} \mathcal{L}_{\text{cls}}(\tilde{y}_i^t, y_i^t; \theta^t), & y_i^t \in C_{\text{new}}^t \\ \text{s.t. } grad(\theta_{c\_\text{old}}^t) = 0, & \theta_{c\_\text{old}}^t \in \theta^t \end{cases} \tag{2}$$

对于增量学习的新会话 t ($t>0$), 将会话 t 时网络可训练的全部参数定义为 θ^t, 其中旧知识通道的卷积核参数为 $\theta_{c\_\text{old}}^t$. 旧知识通道由通道权重因子 γ 的值确定, 见图 5. $\mathcal{L}_{\text{cls}}$ 是交叉熵损失函数, C_{new}^t 是增量学习新会话 t 的类别, y_i^t 是新类样本的 ground truth, $\tilde{y}$ 是新类样本的预测值. $grad(\cdot)=0$ 表示将卷积核权重的梯度置 0. 重复迭代公式 (2), 直到损失不再下降为止. 训练完成后, 新知识被整合进新学习区域.

3.2 区域集成

分离后的空间是对知识的单独表达, 需要集成以形成统一的认知, 并同步执行参数分离来把这些统一的认知存放到旧知识空间中, 作为一种长期记忆保存下来, 这称为区域集成, 与区域隔离交替进行.

区域集成从网络结构和损失函数两方面实现. 在网络结构上, 使用“双分支融合”方式配合着可将神经元与特定任务绑定的“自学习掩码”, 来强化新、旧知识区域的集成表达效果; 在损失函数上, 接受 Margin Loss 在缓解“近期偏好”方面的努力, 并增添蒸馏损失以鼓励稳健的知识融合 (见公式 (9)).

3.2.1 自学习掩码 SLM

区域集成过程中, 解除对旧知识分区的冻结保护, 整个网络同时接受新、旧样本的训练, 来将新、旧知识重新连接成一个整体. 因此, 新旧知识分区的集成效果直接关乎类增量学习网络的整体表达能力. 为此, 我们提出了自学习掩码模块 SLM (见图 6), 它用于在新、旧知识区域集成过程中将局部神经元与特定任务适配. 在训练时 SLM 将不同任务信息绑定不同组合神经元, 在预测时又用掩码定位任务特征, 将任务信息的提取范围限制在一个能尽可能准确地表达任务的局部空间中, 从而提高网络对新、旧任务的整体表达能力.

SLM 首次尝试将轻量化注意力网络 GC-Net[39]和文本分割中提出的二值化方法[40]结合, 在只引入少量计算的情况下, 实现了可微分的注意力掩码生成. SLM 包含 2 个步骤: 自学习掩码生成和掩码应用.

(1) 自学习掩码生成: 依赖于特定的任务特征并经过 GC-Net[39]和可微分二值化处理的掩码, 既实现了端到端的掩码生成训练, 又通过强制性的 0-1 稀疏化的松弛约束, 获取任务更感兴趣的局部神经元位置.

M^{In} 表示原特征图, $M^{\text{In}} \in R^{N\times C\times H\times W}$. 其中 N 代表批量大小, C 代表特征图通道数, H, W 为特征图的高和宽.

M^{In} 经 2 个串联管道加工后得到掩码 M^{Mask}. 首先用注意力网络 GC-Net 管道加工 M^{In}, 得到尺寸不变的处理结果 $M^{\mathrm{GC}} \in R^{N\times C\times H\times W}$; 接着, 用滤波器 $F1 \in R^{O\times C\times H\times W}$ 处理 M^{GC} 输出 $M' \in R^{N\times O\times H\times W}$, 对 M' 进一步执行可微分阈值二值化输出掩码 $M^{\mathrm{Mask}} \in R^{N\times O\times H\times W}$. M^{Mask} 包含着任务更感兴趣的局部神经元位置信息, 其中 O 代表特征图经过 $F1$ 处理后的通道数 (滤波器 $F1$ 的数目). 其中, M^{GC} 和 M' 的计算见公式 (3), 利用可微分阈值二值化生成 M^{Mask} 参见公式 (4).

$$\begin{cases} M^{\mathrm{GC}} = GCNet(M^{\mathrm{In}}) \\ M' = Conv1(M^{\mathrm{GC}}, F1) \end{cases} \tag{3}$$

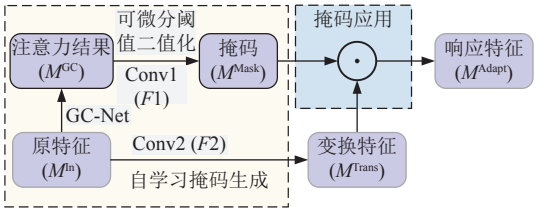


图 6 SLM 模块

阈值二值化对于掩码的生成是至关重要的, 它可以只关注与任务相关的局部神经元. 但是标准的阈值二值化不利于训练的收敛, 本文采用了文献 [40] 提出的基于可微分二值化模块, 实现端到端的学习. 为了保证训练过程的稳定和提高识别精度, 建议使用一个近似的阶跃函数进行二值化, 见公式 (4).

$$M^{\mathrm{Mask}} = \hat{B}_{i,j,k}\left(M'_{i,j,k}\right) = \frac{1}{1+\mathrm{e}^{-s\left(\sigma\left(M'_{i,j,k}\right)-Th_{i,j,k}\right)}} \tag{4}$$

其中, $\hat{B}_{i,j,k}$ 为近似二进制图, i, j, k 分别代表特征图的宽、高和通道坐标. 该式迫使输出的掩码值逼近 0 或者 1, 等价于 0-1 稀疏化的松弛约束. $Th_{i,j,k}$ 为学习到的自适应阈值图, s 为缩放因子, 根据经验设置为 50. 这种近似二值化函数的方法与标准二值化类似, 且是可微的, 因此可以在训练时与神经网络参数同步优化.

(2) 自学习掩码应用: 是将局部神经元位置信息与原任务特征相适配, 掩码值为“1”则是要保留的特征位置, 为“0”的是要舍弃的非必要的特征位置, 所以通过掩码可以进行必要的特征提取和空间压缩.

掩码应用时, 首先用与 $F1$ 同尺寸的卷积核 $F2 \in R^{O*C*H*W}$ 处理原任务特征 M^{In}, 生成一个与掩码 M^{Mask} 同尺寸的变换特征图 $M^{\mathrm{Trans}} \in R^{N\times O\times H\times W}$. 然后将 M^{Mask} 与 M^{Trans} 进行元素维度的点乘, 得到任务适配特征图 $M^{\mathrm{Adapt}} \in R^{N\times O\times H\times W}$, 该任务适配图实现了局部神经元与特定任务绑定的目标, 见公式 (5).

$$\begin{cases} M^{\mathrm{Trans}} = Conv2(M^{\mathrm{In}}, F2) \\ M^{\mathrm{Adapt}} = M^{\mathrm{Trans}} \odot M^{\mathrm{Mask}} \end{cases} \tag{5}$$

3.2.2 整合新知识

双分支融合前的分区学习和自学习掩码, 使主分支对新、旧知识表达能力增强. 但主分支因为保存旧知识记忆而降低了可塑性, 因此不得不降低对新知识的适应性[15], 本文称之为“分类妥协”问题. 为缓解这一问题, 本文将基于门控实现跨层信息融合的方法[41]改造为同时跨层和跨分支的模式以实现双分支信息融合, 用于在新、旧知识分区集成过程中提升网络对新知识的识别能力, 详见图 7.

新知识经辅助分支进行学习, 新、旧信息经主分支流动进行学习, 得到的 $x_1,\ldots,x_k,\ldots,x_m,\ldots,y_n$ $(1,k,m,n<l)$ 作为 GBF 的输入, 执行 GBF 后的特征图经过卷积层和全连接层得到最终的识别结果.

图 7 中深黑色圆点是对双分支网络对应层进行信息聚合, 其具体操作是先将主分支和辅助分支的对应层 ($H\times W\times C$) 进行通道拼接 ($H\times W\times 2C$), 接着经过一层卷积进行信息聚合, 输出 x_i, i 代表层数, $i<l$, l 代表网络的总层数. 标量 a 控制辅助分支的融合参与度, $1–a$ 控制主分支的融合参与度, 具体设计见附录 A. 回形针标注位置表示此处的中间量与 a 相乘来控制分支的融合参与度. Gate 为单层卷积运算, 经激活函数处理实现信息过滤. 最顶部的门控使用 Sigmoid 函数筛选冗余区域; 剩下 3 个门控经 tanh 激活, 决定待融合的分支信息保留与否.

GBF 模块, 是将 $x_1,\ldots,x_k,\ldots,x_m$ 别经过门控 Gate 和 tanh 函数激活后, 执行元素尺度的加法, 详见图 7 中红色

虚线框. 是为了提取辅助分支中新知识的多级上下文信息, 进行信息融合的准备工作. 主分支一方面使用门控 Gate 和 Sigmoid 激活函数, 筛选出高价值信息 G_l; 另一方面用 y_n 筛选后的低价值信息 $1-G_l$ 区域 (可作为冗余区域) 接收来自红色虚线框的新知识, 实现双分支信息融合. 此外, 为减少信息融合的损失, 还将原始信息 y_l 直接保留下来, 与上述两方面的信息执行元素级的加法融合.

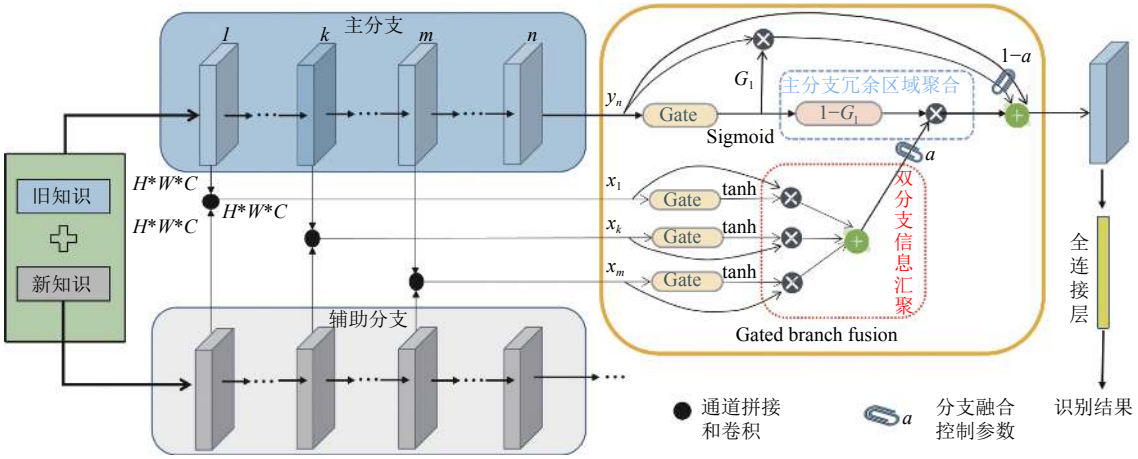


图 7　双分支信息融合

3.3　网络训练

网络训练分为单次“基础训练”和接续的多次“增量训练”, 见图 8 以及附录 C 的算法 C1, 算法 C2.

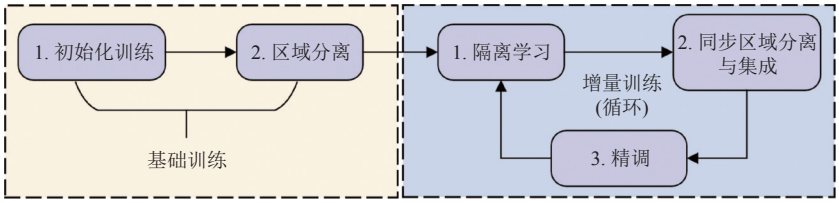


图 8　训练流程图

第一, “基础训练”过程: 包括“初始化训练”和“区域分离”两个阶段.

“初始化训练”目的是将第 1 批接收到的知识整合到网络空间中. 在整个空间中, 接受样本输入, 并利用交叉熵损失函数进行训练, 使网络获得关于基础样本的识别能力.

“区域分离”则压缩首批知识占据的存储空间, 并为后续的学习预留学习空间. 区域分离将完整的网络分裂成两部分: 存储首批知识的空间和独立学习新知识的自由参数空间.

第二, “增量训练”过程: 分 3 个阶段: “隔离学习”“同步区域分离与集成”和“精调”. 增量训练的迭代促成学习所有类增量任务的目的达成.

第 1 个阶段“隔离学习”将学习的新知识放入分区预留空间, 避免对已有知识的干扰. 其关注以下 3 点.

(1) 回放旧知识辅助训练, 缩小集成难度: 利用预留空间学习新知识, 并回放旧知识.

(2) 缓解“灾难性遗忘”: 通过冻结历史任务的权重参数, 避免模型丢失执行旧任务的能力.

(3) 知识共享. 虽然从历史任务中学到的参数被冻结了, 但是允许模型读取先前训练的参数并使用. 旧任务的参数重用自然鼓励回收以前学到的技能, 有助于新任务的学习.

第 2 个阶段“同步区域分离与集成”是将分区知识集成并压缩, 促进模型在同一个域空间内学习到新、旧知识的共同表征, 并为后续学习预留空间. 隔离学习完成后, 执行同步的区域集成和区域分离.

第 3 个阶段“精调”的目的是使原有模型更加适配已习得的知识, 见图 2. 冻结除全连接层 (分类头) 之外的所有参数并训练, 通过“精调”稳固学习效果以及提升识别精度.

3.4 损失函数

ASPIL 损失函数包括 “基础训练”损失函数和“增量训练”损失函数.

3.4.1 基础训练损失函数

“基础训练”包括“初始化训练”和“区域分离”, 其损失函数分别如下.

● 初始化训练: 损失 L^{Init} : 初始化训练使用交叉熵损失直接提取样本信息, 参见公式 (6).

$$L^{\text{Init}} = \sum_{i}^{N} c_i \log(q_i) \tag{6}$$

其中, c_i 表示在第 i 类上的 ground truth 值, $c_i \in \{0,1\}$, 正标签取 1, 负标签取 0. q_i 代表新学习的样本的 Softmax 输出在第 i 类上的值.

● 区域分离: 损失 L^{Split} : 参见公式 (7). 正如第 3.1.1 节的描述, 通过训练 BN 层权重因子 γ, 实现通道的稀疏化以达到区域隔离的目的.

$$L^{\text{Split}} = \sum_{i}^{N} c_i \log(q_i) + \delta g(\gamma) \tag{7}$$

$g(\gamma)$ 是 BN 层权重因子 γ 的 $L1$ 范数, 用来实现通道的稀疏化. δ 用来平衡这两项, 并设置为 1E–4, 这遵循了文献 [38] 的通道稀疏性参数设置.

3.4.2 增量训练损失函数

(1) 增量训练过程的第 1 阶段“隔离学习”实施区域隔离, 发生的是独立进行的新学习. 由于本文通过通道剪枝而非损失函数保护旧知识, 仍用公式 (6) 训练新知识, 见附录 C 的算法 C4.

● 隔离学习: 损失 L^{Iso} 与初始化训练损失 L^{Init} 保持一致, 因为此时只进行新学习:

$$L^{\text{Iso}} = \sum_{i}^{N} c_i \log(q_i) \tag{8}$$

(2) 增量训练过程的“同步区域分离与集成”和“精调阶段”阶段实现区域集成, 自学习掩码和双分支融合即运行在该阶段, 目的是构建一个集成的单分类头模型. 本文利用知识蒸馏转移旧知识到集成模型中, 并设计一个间隔损失函数提升分类头对新、旧知识的区分能力.

● 精调和同步区域分离与集成的损失相同, 用 $L^{\text{S\&I}}$ 表示, 二者的区别只在于参数更新的范围. “精调”和“同步区域分离与集成”的损失函数相同, 是为了保持模型已有的特性, 包括少遗忘、抑制“近期偏好”功能. 不同于“同步区域分离与集成”对 ASPIL 的全部参数进行更新, “精调”只更新 ASPIL 的分类器的参数, 这是为了进一步提升分类层对所有知识 (新知识和旧知识) 的识别和区分能力. 具体的“精调”策略是冻结网络的特征提取层, 然后输入新、旧知识标签, 依据样本标签和公式 (11) 计算损失 $L^{\text{S\&I}}$, 最后依据反向传播算法更新分类层的参数.

$L^{\text{S\&I}}$ 由 3 个部分组成, 分别是 L_{DP} 、 Margin Loss 和 $g(\gamma)$. L_{DP} 的表达见公式 (9), Margin Loss 表达见公式 (10). 同步区域分离与集成是区域分离和区域集成的同步运行, 因此需要正则项 $g(\gamma)$ 实现区域隔离.

(1) L_{DP} 蒸馏过程的目标函数 L_{DP} 由蒸馏损失 L_{soft} 和分类损失 L_{hard} 加权得到. 蒸馏损失为每个类别都分配了概率, 属于正例的类别概率最高, 负例的类别概率可以不为 0, 这是优化的软目标. 分类损失强制正标签取 1, 其余负标签取 0, 目标是使预测值尽可能接近于真实值, 称之为优化的硬目标. L_{DP} 的计算见公式 (9).

$$L_{\text{soft}} = -\sum_{j}^{N} p_j^{Temp} \log(q_j^{Temp}),\ L_{\text{hard}} = -\sum_{j}^{N} c_j \log(q_j^{1}) \tag{9}$$

其中, $p_i^{Temp} = \dfrac{\exp(v_i/Temp)}{\sum_k^N \exp(v_k/Temp)}$, $q_i^{Temp} = \dfrac{\exp(z_i/Temp)}{\sum_k^N \exp(z_k/Temp)}$, $v_i = \theta_{\text{logit}}^{t-1}(x)$, $z_i = \theta_{\text{logit}}^{t}(x)$. 高温蒸馏过程的目标函数 L_{DP}

为 $L_{\mathrm{DP}}=\alpha L_{\mathrm{soft}}+\beta L_{\mathrm{hard}}$, θ 为模型参数集合. $c_j\in\{0,1\}$表示在第 j类上的 ground truth 值, 除了正标签取 1, 其余负标签取 0. v_i 代表输入样本在θ^{t-1}上的 logit 输出, z_i 代表输入样本在θ^t 上的 logit 输出; p_i^{Temp} 和 q_i^{Temp} 表示输入样本 x 分别在模型θ^{t-1}和θ^t 上的蒸馏输出在第 i 类上的值, 蒸馏温度 $Temp$ 一般设为 2. $\alpha=1.0$ 和 $\beta=0.24$ 都是平衡系数.

(2) L_M 表示 Margin Loss, 通过分离新、旧类的决策边界以抑制“近期偏好”现象, 表达式如下:

$$\begin{cases} margin=\max(clafi_{\text{old\_num:new\_num}}((x_i,y_i\in C_{\text{old}});\theta))-\max(clafi_{:\text{old\_num}}((x_i,y_i\in C_{\text{old}});\theta)) \\ L_M=\log(s_0+mean(topK(margin,k))/b),\ k,b\in N_+ \end{cases} \tag{10}$$

其中, x_i 为样本, y_i 为类标签, C_{old} 是旧知识类, $clafi$ 是 Softmax 分类器. s_0 是正数, 确保 log 函数定义域为正数. 向量$clafi_{\text{old_num:new_num}}$ 是分类器将样本归类为新知识类别的概率, 向量$clafi_{:\text{old_num}}$ 是分类器将样本归类为旧知识类别的概率. 因此, 标量 $margin$ 代表将旧类样本混淆地预测为新类样本和正确归类之间的差值. $topK$ 取 margin 值降序排列的前 k 个, k 一般设为 batchsize 大小的 1/5. b 是缩放因子. 另外, 本文预先使用 L2 范数归一化清洗分类层的输入数据.

(3) $L^{\mathrm{S\&I}}$ 表述为公式 (11), $\lambda=0.36$ 和$\delta=1\mathrm{E}-4$ 是平衡系数. 以提升增量识别精度为目标, 利用网格搜索法, 得到参数λ 的最优值是 0.36. 正则项$g(\gamma)$是 BN 层权重因子的 L1 范式, 用来实现通道的稀疏化.

$$L^{\mathrm{S\&I}}=L_{\mathrm{DP}}+\lambda\times L_M+\delta g(\gamma) \tag{11}$$

3.5 ASPIL 度量指标

我们使用两个指标: 平均识别精度和平均遗忘来衡量 ASPIL 在增量学习上的表现.

● 平均识别精度 ($Avg\in[0,1]$) 设$r_{i,j}$ 为模型在任务 i 上进行训练后, 在任务 j 的测试集上的识别精度, 在任务 T 时的识别精度R_T 被定义为公式 (12).

$$R_T=\frac{1}{T}\sum_{j=1}^{T}r_{T,j} \tag{12}$$

平均识别精度 Avg 表示完成迄今为止所有任务 ($1\ldots T$) 后的 T 个识别精度的平均值, 计算见公式 (13).

$$Avg=\frac{1}{T}\sum_{i=1}^{T}R_i \tag{13}$$

● 平均遗忘 ($F\in[0,1]$) 设 f_j^i 为模型在任务 i 上进行训练后对任务 j 的遗忘, 计算见公式 (14).

$$f_j^i=\max_{l\in\{1,2,\ldots,i-1\}}r_{l,j}-r_{i,j} \tag{14}$$

因此, 在任务 T 时的平均遗忘被定义为公式 (15).

$$F_T=\frac{1}{T-1}\sum_{j=1}^{T-1}f_j^T \tag{15}$$

4 实 验

4.1 数据集与测试方案

为了验证 ASPIL 的有效性, 将其应用于 ResNet[42]和 VGG[43]. 实验数据集为 CIFAR-100 和 miniImageNet. 在 12 GB 显存的 GTX-2080 Ti GPU 和 CPU@14.4 GHz 上, 基于 PyTorch 和 NVIDIA CUDA 实现.

● CIFAR-100 数据集: CIFAR-100 是 Alex Krizhevsky 所提供的 8000 万张大小为 32×32 的小图像数据集的子集. 它包含超过 100 个类的 6 万张 RGB 图像, 每个类有 500 张图像用于训练, 另有 100 张图像作为测试集.

● miniImageNet 数据集: miniImageNet 数据集是 ImageNet-1k 的子集. 它包含了 100 个类别中的 6 万张彩色图像. 每个类有 600 张图像, 其中有 500 张用于训练和 100 张供测试. 每张图像尺寸为 84×84. 与 CIFAR-100 相比, miniImageNet 数据集更复杂且适合原型化, 是增量学习领域最常用的基准数据集.

● 实验设置: 所有实验使用 SGD 优化器, 初始学习率为 2.0. 每个训练阶段有 100 个 epochs. 0–47 epochs, 学习率 2.0; 48–61 epochs, 学习率 0.4; 62–79 epochs, 学习率 2/25; 80–99 epochs, 学习率 2/125. 权重衰减恒定为 5E–4, Batchsize 为 128. 使用相同的类随机变换种子, 模型单头输出且能够增加输出的类别, 分类精度均为 TOP-1 精度. 对于 Softmax 知识蒸馏实例, 使用 $Temp$=2 的蒸馏温度. 此外, 为了遵循增量学习的要求, 每个任务只访问一次, 且

不在完整的任务集上调优超参数, 以避免使模型在历史数据上过拟合, 并限制新知识学习的灵活性, 这是违反增量学习原则的.

为了进行公平的比较, 实验条件与 LUCIR[18]和 E2E[30]保持一致, 同样使用 iCaRL[11]的回放样本选取协议, 最大范例存储容量为 2 000, 实施一致的分割方法生成类增量任务批次.

4.2 与相关方法的比较

ASPIL 不仅在平均识别精度方面优于其他所有方法, 还具有低遗忘的优势. 表 1 和表 2 显示了包括第 1 个增量步骤在内的以上所有方法的平均精度 (表示为 *Avg*), 计算见公式 (13). 表 3 显示 ASPIL 在 CIFAR-100 和 miniImageNet 数据集上的 6 阶段和 10 阶段的平均遗忘 (表示为 F), 计算见公式 (15).

表 1 10 阶段识别精度比较表 (%)

Dataset	Method	Number of classes										
		10	20	30	40	50	60	70	80	90	100	*Avg*
CIFAR-100	A-GEM[12]	85.0	58.42	48.3	44.39	43.7	41.4	40.48	37.26	31.74	26.91	45.76
	EMR[44]	82	61.5	54.67	50.74	47.83	44.42	42.71	36.73	34.17	31.78	48.66
	iCaRL[11]	84.9	73.7	69.17	64.75	61.94	60.17	58.3	54.99	53.6	50.83	63.24
	LUCIR[18]	89.1	72.2	63.43	56.17	53	49.87	49.3	46.31	43.81	42.09	56.53
	LwF[7]	85.8	58.8	53.62	48.52	42.02	38.24	35.86	33.16	29.43	25.74	45.12
	EWC[8]	86.1	66.1	60.57	53.75	47.42	43.88	41.07	39.24	35.83	31.33	50.53
	ABD[32]	**91.5**	74.2	70.2	57.8	52.98	46	43.36	38.59	36.52	33.2	54.44
	SCR[31]	86	76.7	74.1	68.7	65.5	63.9	60.03	58.9	54.91	51.08	65.98
	S&B[15]	87.2	81.47	77.52	73.64	69.15	64.66	61.55	59.05	55.31	52.29	68.18
	Upper Bound	87.3	84.57	82.43	81.59	79.74	78.64	78.42	77.11	76.85	76.32	**80.30**
	ASPIL	90.5	**85.5**	**82.52**	**80**	**76.08**	**72.11**	**68.74**	**64.14**	**61.97**	**60.25**	**74.18**
miniImageNet	S&B[15]	89.26	83.77	79.8	75.12	70.46	66.93	63.08	62.35	57.63	54.72	70.31
	LwF[7]	88.1	81.2	72.2	63.57	55.3	49.85	44.86	40.77	37.14	32.3	56.53
	Bic[27]	90.8	80.18	75.53	71.23	67.65	62.58	58.19	54.86	51.54	47.88	66.04
	E2E[30]	90.45	79.68	72.53	67.93	62.65	58.58	54.49	50.86	49.14	43.88	63.02
	ABD[32]	93.3	78.39	73.86	66.54	58.94	53.24	48.01	43.73	39.93	38.03	59.4
	Upper Bound	89.5	86.71	84.13	83.25	81.56	82.14	80.57	79.11	78.88	78.32	**82.42**
	ASPIL	**92.3**	**87.69**	**84.18**	**81.57**	**77.57**	**73.84**	**71.04**	**67.36**	**64.19**	**62.48**	**76.22**

注: *Avg*代表Top-1平均识别精度. 加粗表示各项目中最好的识别结果

表 2 5 阶段识别精度比较表 (miniImageNet 数据集) (%)

Method	Number of classes						
	5	20	40	60	80	100	*Avg*
S&B[15]	93.2	85.05	77.47	69.91	66.29	60.80	75.45
iCaRL[11]	89.24	77.89	69.06	62.37	57.78	50.62	67.83
E2E[30]	88.76	80.21	65.43	58.32	50.21	48.35	65.21
LUCIR[18]	90.03	82.57	74.48	65.21	61.63	55.48	71.57
Upper Bound	93.2	88.81	85.69	83.92	81.46	79.75	**85.47**
ASPIL	**93.9**	**87.24**	**82.2**	**77.45**	**73.40**	**70.14**	**80.72**

注: 在miniImageNet数据集上进行, *Avg*代表Top-1平均识别精度. 加粗表示各项目最优识别结果

图 9 为基于 ResNet18 的 ASPIL 方法与 LwF[7]、EWC[8]、iCaRL[11]、A-GEM[12]、S&B[15]、LUCIR[18]、Bic[27]、E2E[30]、SCR[31]、ABD[32]、EMR[44]的比较结果, 以及本文 ASPIL 方法使用 ResNet50[42]和 VGG16[43]时的泛化.

ASPIL 在所有设置中取得了全过程的领先结果, 这得益于 ASPIL 方法综合运用了区域隔离学习、样本回放和自学习掩码, 并缓解了“近期偏好”和“分类妥协”问题, 将在第 4.3 节消融实验部分进一步讨论.

表 3　平均遗忘比较表 (%)

Method	Forgetting			
	CIFAR-100 (T=6)	CIFAR-100 (T=10)	miniImageNet (T=6)	miniImageNet (T=10)
LwF[7]	36.04	41.72	38.56	44.20
EWC[8]	30.54	37.76	33.06	40.68
ABD[32]	26.62	32.00	28.52	35.40
iCaRL[11]	21.86	28.60	21.30	28.42
S&B[15]	17.23	26.37	20.35	29.96
ASPIL (ours)	**11.81**	**21.72**	**11.24**	**21.37**

注: 度量ASPIL在CIFAR-100和miniImageNet数据集上的6阶段平均遗忘和10阶段平均遗忘, 平均遗忘计算参见第3.5节公式(15). T=6表示6阶段 (1个基础任务和5个增量任务)的平均遗忘, T=10表示10阶段 (1个基础任务和9个增量任务)的平均遗忘. ASPIL遗忘程度最低, 用加粗表示

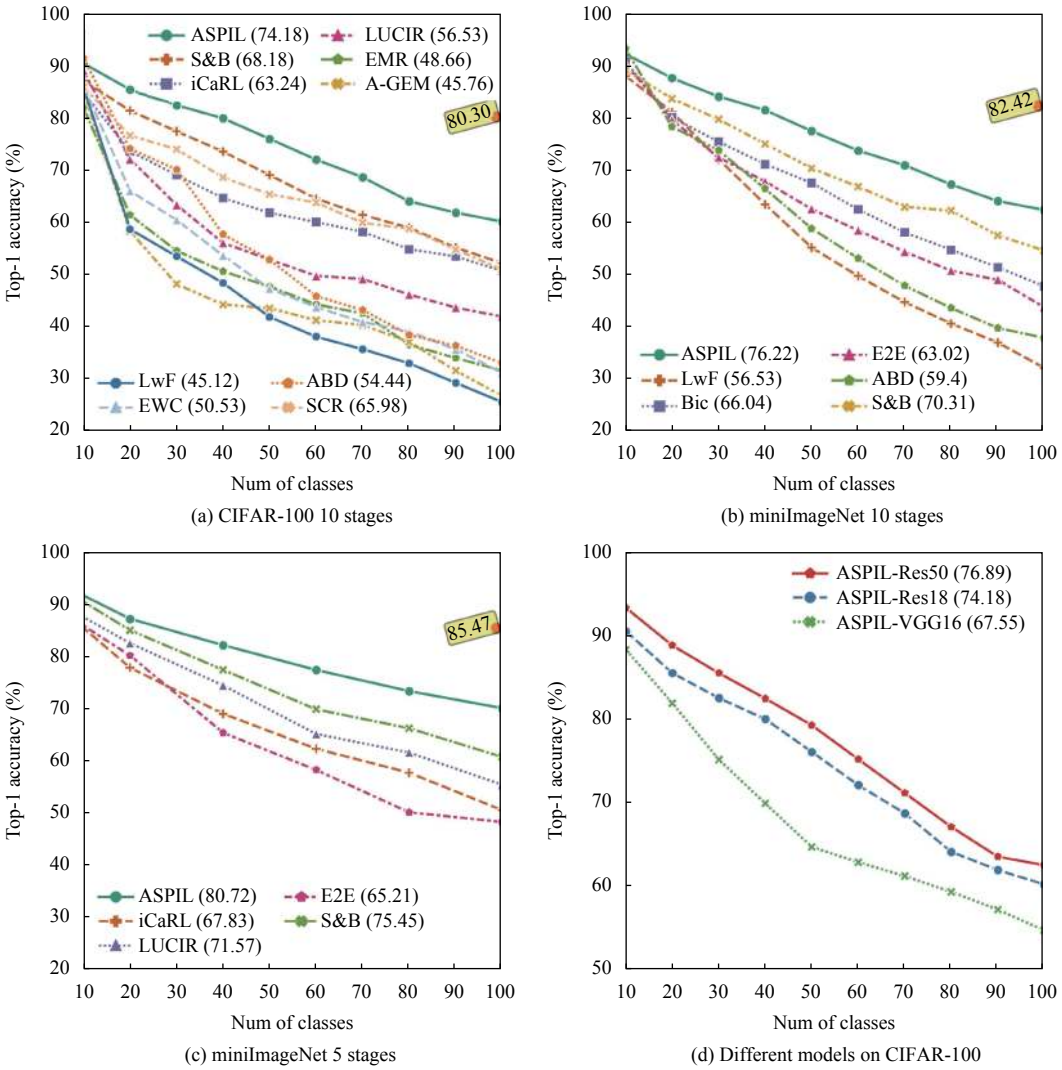


图 9　在 ResNet18 上 ASPIL 与相关方法比较

首先在 CIFAR-100 上进行了 10 阶段 (stages) 的增量训练, 如图 9(a) 所示. 其次在 miniImageNet 数据集上进行了两个系列的实验, 每增量步 10 个类和 20 个类, 见图 9(b)、(c). 最后, 分别使用 ResNet18、ResNet50 和 VGG16 实现了 ASPIL 方法, 其中使用 ResNet50 识别 CIFAR-100 的 10 阶段 Top-1 平均精度高达 76.89%, 胜过 ResNet18 2.71% 和 VGG16 9.34%, 见图 9(d).

第一, CIFAR-100 数据集上 ASPIL (平均精度 74.18%) 不仅较大幅度的领先没有区域隔离的 LwF[7] 29.06%、EWC[8] 23.65% 和 A-GEM[12] 28.42%, 也优于注重区域隔离的 S&B[15]方法 6%, 且全程优胜其他方法. 另 ASPIL 的 74.18% 与上限 80.3% 的差距 6.12% 为最小, 也表明了 ASPIL 的优越性. 由于类增量任务之间受干扰不同, 导致方差较大, 使常用于任务增量的 A-GEM[12]和 EMR[44]方法在本实验表现最差, 而 S&B 是最接近 ASPIL 表现的, 这应归因于对区域隔离方法的运用. LUCIR[18]和 ABD[32]的表现略强于 LwF、EWC 和 A-GEM, 是因为其一定程度解决了“近期偏好”问题. 此外, 与记忆回放方法 iCaRL[11]、SCR[31]相比, ASPIL 也依次取得了 10.94%、8.2% 的优势. 在图 9(a)、(c) 中 iCaRL 方法均取得靠前名次, 可能是因为 iCaRL 的 NCM(近期类均值) 分类器依靠存储范例 (代表性样本) 进行度量分类而不使用全连接层来进行的原因, 由于避免了旧类连接在全连接层中的结构变化, 这对于“灾难性遗忘”是有一定抵抗力的. SCR 表现仅次于 S&B, 比本文低 8.2%, 但优于 EMR 和 LwF.

第二, 在 miniImageNet 上的 10 阶段运行中, 见图 9(b), ASPIL 和 S&B 赢得了和图 9(a) 一致的优异表现, ASPIL 相比于 LwF、Bic、E2E、ABD、S&B, 最后增量阶段依次取得了 30.18%、14.6%、18.6%、24.45%、7.76% 的领先, 与上界平均差距 6.2%. 其中 ABD[32]差于 Bic[27]6.64% 的表现和 ABD[32]的结果一致, Bic 与 E2E 接近且大于 LwF 亦得到 Bic[27]研究的支持. 可以看出, 同样的方法在 miniImageNet 识别精度上普遍略高于 CIFAR-100, 这可能是因为 miniImageNet 有着更丰富的像素细节用于识别. 对本文 ASPIL 来说更丰富的细节更利于掩码对任务绑定, 见第 4.3.2 节实验验证. 考虑到 CIFAR-100 数据集像素低, 实验移除了 ResNet18[42]结构中的第 1 个池化层, 另外 CIFAR-100 和 miniImageNet 使用了不同的归一化因子以提升各自数据集识别精度, 这些都可能带来了轻微影响. 图 9(c) 中 ASPIL 在 miniImageNet 5 阶段增长趋势和图 9(b) 一致, 这是由于增量步骤少所以遗忘少的原因, 相比图 9(a) 平均精度提升 6.54%、相比图 9(b) 提升 4.5%.

图 9 中 Top-1 精度的平均值显示在括号中, 均值来源详见表 1、表 2. 图右边框红色圆点和黄色文本框共同标注联合训练上界均值, 即表 1、表 2 中上限 (upper bound). ASPIL 方法在 CIFAR-100 和 miniImageNet 数据集上均取得了最先进的结果, 见图 9(a)–(c). 为了进行公平的比较, 对比实验的实现遵循 iCaRL[11]的回放样本选取协议和第 4.1 节所声明的实验设置.

4.3 消融实验

为验证 ASPIL 主要构成部分: 可持续区域隔离学习 (第 3.1 节)、自学习掩码 SLM (第 3.2.1 节)、双分支融合 (GBF) (第 3.2.2 节), 这 3 项对 ASPIL 性能的提升程度, 分别对其进行如下消融实验.

4.3.1 可持续区域隔离学习消融实验

为证实在学习一个新任务时, ASPIL 能减轻新、旧任务间的相互干扰, 进行两方面的研究. 一方面通过绘制区域隔离的消融折线图和消融混淆矩阵, 来对分类的质量进行可视化表示, 见图 10. 另一方面, 在不考虑新知识识别能力的情况下, 观察特征提取部分对于旧知识的保留程度, 见表 4, 即旧知识受干扰的程度越小, 旧知识的保留程度就越高.

表 4 的具体实施是: 在隔离学习阶段, 利用之前的区域分离过程提供的空闲自由空间来学习新知识. 然后冻结特征提取部分 (全连接层之前的所有部分), 只在分类器 (全连接层) 上回放旧知识范例进行精调, 观察 ASPIL 的旧知识的识别精度, 以此确定特征提取部分对于旧知识的保留程度.

图 10 验证了区域隔离学习可以提升模型识别精度. 图 11 是对图 10 中 30 和 50 个类这两个中间过程的分类精度细化显示, 它证实区域隔离学习使模型对于旧知识的预测更加准确. 表 4 在此基础上进一步探究, 发现网络的特征提取层确实存储了较为完整的旧知识信息, 因为仅通过分类层的精调, 就能获得近乎上界的分类精度 (例如, 对于 ResNet18 有着低于 1.2% 的衰减). 特别地, 由于表 4 结果是在区域集成之前的, 那时模型尚未形

成对新、旧知识的统一表达能力, 在后续集成表达 (即区域集成训练中) 中仍会产生一定程度的知识损失. 但是, 这至少提供了一个好的融合开始, 是有价值的. 图 10 的结果也验证了这一点, 因为区域隔离学习确实提升了类增量识别精度.

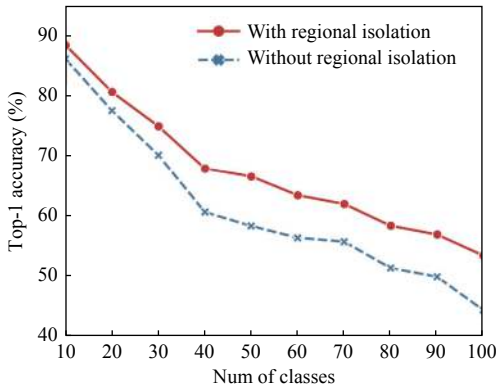
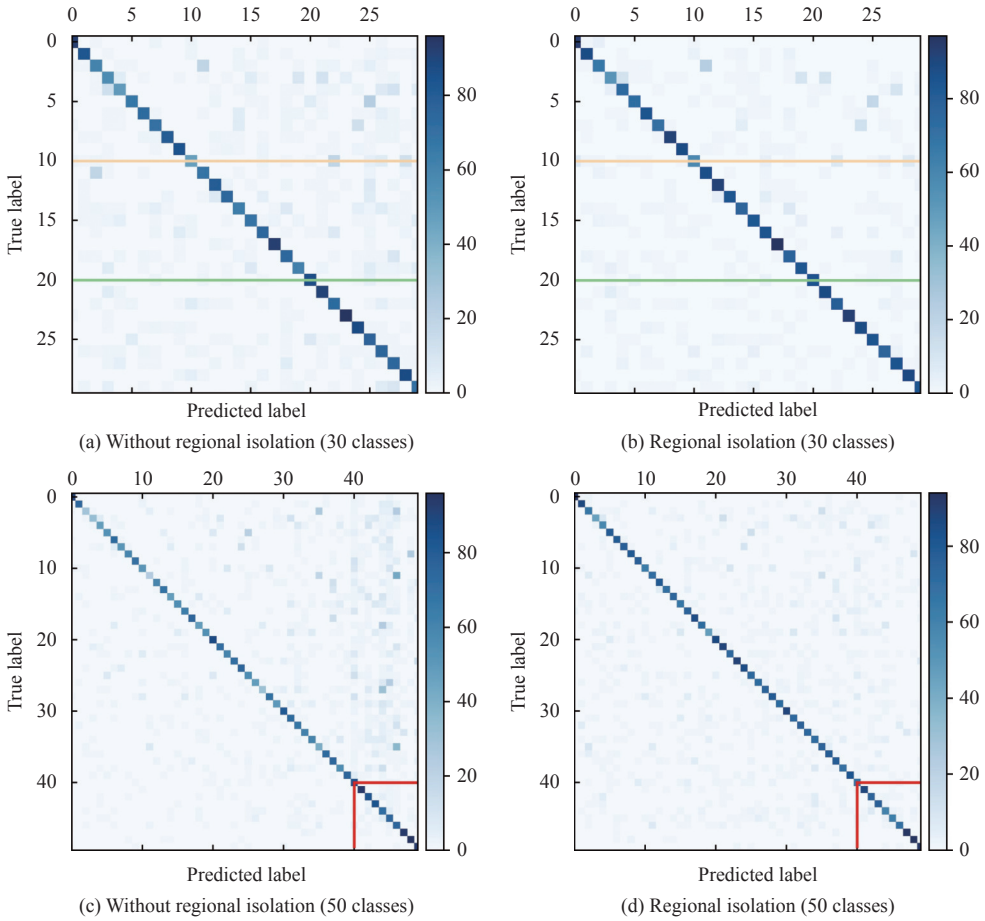


图 10 区域隔离消融实验精度 (CIFAR-100)

表 4 ASPIL 隔离学习阶段后旧知识保留程度 (%)

特征提取网络	旧知识识别精度	
	联合训练	隔离学习后精调恢复
ResNet18[42]	86.4	85.3 (1.1↓)
ResNet34[42]	87.8	86.4 (1.4↓)
VGG16[43]	88.9	87.2 (1.7↓)

注: 网络在终止隔离学习时仍然保留着相对完整的旧知识, 因为基于通道剪枝的区域分离提供的旧知识区域是隔离的



(a) Without regional isolation (30 classes) (b) Regional isolation (30 classes)

(c) Without regional isolation (50 classes) (d) Regional isolation (50 classes)

图 11 ASPIL 区域隔离消融混淆矩阵

图 11 中使用 ResNet18 在 miniImageNet 数据集上进行. 样本回放区尺寸为 2 000. 近期到来的类存在更少的错分现象 (见图 11(a)、(b)); 最后的增量任务分类表现良好 (见图 11(c)、(d)). 相比于图 11(a)、(c), 图 11(b)、(d) 错误分类较少, 因为区域隔离学习隔离了新、旧类的训练环境, 避免新知识与旧知识混淆或覆盖旧的知识, 从而达到降低新、旧任务间的相互干扰的目的.

4.3.2 自学习掩码消融实验

自学习掩码 SLM 模块的消融实验是为检测神经元对特定任务的绑定效果的. 因为 SLM 生成的近似 0-1 掩码和神经元按位点乘实现局部神经元与特定任务的适配, 因此可以通过查看不同分类任务下的掩码输出, 验证 SLM 能够将局部神经元与特定任务绑定, 见图 12.

(a) miniImageNet 掩码可视化 (b) CIFAR-100 掩码可视化

图 12 自学习掩码任务绑定效果

除了关注 SLM 对特定任务的绑定能力外, 本文还通过 SLM 学习效果表展示了 SLM 的优势, 如表 5 所示. 在 CIFAR-100[19]数据集的 10 阶段学习中, 与具有区域隔离学习的基线网络相比, SLM 的平均准确率为 70.51%, 比基线高 3.31%.

表 5 SLM 精度表 (%)

Method	Number of classes (10 stages)										
	10	20	30	40	50	60	70	80	90	100	*Avg*
Baseline	88.4	80.6	74.9	67.85	66.52	63.37	61.92	58.29	56.83	53.33	67.20
SLM(−GC)	90	82.95	78.69	74.24	70.46	66.88	63.86	59.81	58.02	55.9	70.08
SLM	**90.9**	**83.45**	**79.05**	**75**	**70.95**	**67.33**	**64.07**	**60.15**	**58.24**	**56**	**70.51**
Gains	2.5↑	2.85↑	4.15↑	7.15↑	4.43↑	3.96↑	2.15↑	1.86↑	1.41↑	2.67↑	3.31↑

注: 在CIFAR-100数据集上进行, *Avg*代表Top-1平均识别精度, SLM(−GC)表示去除GC的SLM, 加粗表示各项最好的识别结果

图 12 中取自 ResNet18 网络第 9 个特征层对应的掩码, 该特征层和掩码的形状均为 channel=128, high=16, width=16. 图 12(a)、(b) 将 SLM 生成的任务掩码以热力图或者灰度图的形式显示. 图中任务对应的灰度图或者热

力图的局部颜色越深, 代表该局部命中神经元, 因为掩码的位置命中与神经元的位置命中是完全一致的. 在miniImageNet 上, 将掩码绑定到任务的能力优于 CIFAR-100, 因为 miniImageNet 具有更加丰富的像素细节.

表 5 显示 GC-Net 和 SLM 的组合优势, 实现 SLM 需关注以下两点: 1) 裁剪的时候, 依据特征图的尺寸, 对于大尺寸的特征图加大裁剪程度, 反之减小裁剪百分率. 2) 对于特征图的最高层和最底层提供较小的裁剪度, 最高层裁剪度低于 10%. 前者是为了避免未经卷积训练的信息大量丢失, 后者是为了输出信息的丰富.

4.3.3 双分支融合消融实验

双分支融合消融实验主要考察 GBF (门控分支融合) 的效果. 基于区域隔离和 SLM 的 Baseline, 以有无 GBF 来观察网络对新知识的适应性变化. 此外, 我们评估了网络的整体表达能力, 确保在新知识上的适应性提升不会阻碍整体表达. CIFAR-100 数据集用于训练和验证, 网络主干使用 ResNet18 实现.

图 13 显示 GBF 不仅在一定程度上提升了对新知识的容纳能力, 且对旧知识的识别能力也有提高. 新知识得到增强, 因此在下一个增量过程中, 越来越多的新知识将以旧知识的形式积累和保存, 使模型在后续的增量过程中获得持续的收益.

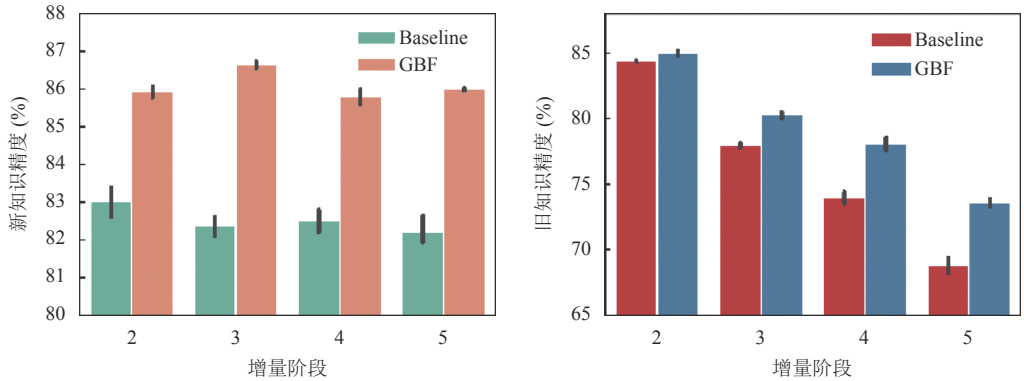


图 13 GBF 新、旧知识识别精度

图 13 也显示 GBF 可以显著提升网络在新知识上的适应性, 从大约 82% 上升到 86%. Baseline 新知识精度的方差大于 GBF, 可能是因为“妥协问题”所引起的新、旧知识对网络空间的争用让新知识的识别处于更大的不确定性中. 此时, 新知识精度尚有很大提升空间.

表 6 显示了 GBF 兼顾新知识的适应性和整体表达的准确性. 这种双分支融合决策机制达到 74.18% 的平均识别精度和 3.67% 的平均精度提升, 且优于单分支融合的 72.15%. GBF 添加至模型后所观察到的性能改善是合理的, 说明跨分支融合使得主分支可以吸收辅助分支学习到的新知识信息, 实现对新知识更准确的拟合. 融合度由控制参数 a (见图 7 标注) 控制辅助分支的融合参与度, 融合控制参数的表达形式及比较见附录 A.

表 6 GBF 精度表 (%)

Method	Number of classes (10 stages)										
	10	20	30	40	50	60	70	80	90	100	*Avg*
Baseline	90.9	83.45	79.05	75	70.95	67.33	64.07	60.15	58.24	56	70.51
GBF	90.5	**85.5**	**82.52**	**80**	**76.08**	**72.11**	**68.74**	**64.14**	**61.97**	**60.25**	**74.18**
Gains	–0.4↓	2.05↑	3.47↑	5.0↑	5.13↑	4.78↑	4.67↑	3.99↑	3.73↑	4.25↑	3.67↑

注: CIFAR-100数据集上进行, *Avg*代表Top-1平均识别精度, 加粗表示各项目中最好的识别结果

另外, 我们还将实验推广到 ResNet50, 验证 GBF 的泛化能力, 参见附录 B.

4.3.4 Margin Loss 消融实验

Margin Loss 消融实验是从 ASPIL 中分离 Margin Loss 正则项作为基线参照, 独立评估该组件的价值. 实验时

取 100 个类的 miniImageNet 数据集等量划分为 5 个学习阶段, 每个学习阶段训练 100 次, 见图 14. 由于初始阶段只有新知识, 不受 Margin Loss 的影响, 因此图中没有绘制 *Stage* 1. 此外, 特别添加的 *Stage* 3 的损失对照反映了 Margin Loss 正则项可以使损失进一步下降, 使模型得到更进一步的训练, 表现出更好的识别能力. 总之, 带有 Margin Loss 的模型, 其能在每个 *Stage* 中的准确率好过没有 Margin Loss 的模型.

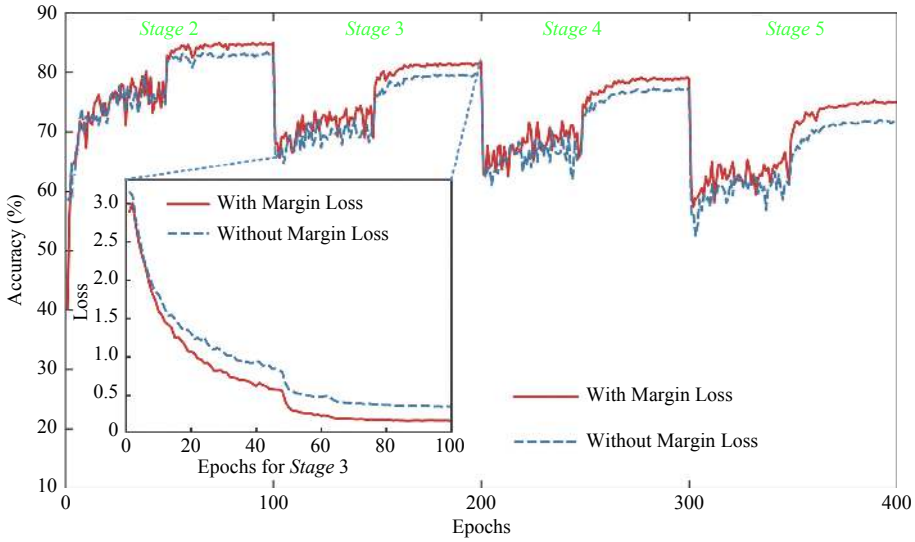


图 14 Margin Loss 5 阶段消融实验

图 15 中 ASPIL 分类器的输入是高维向量, 包含丰富的新、旧知识语义信息. 使用主成分分析 (PCA) 将此高维向量降维成二维, PCA-1 表示 X 轴向, PCA-2 表示 Y 轴向, 据此显示新、旧知识的分布状态. 图 15(a) 是刚结束隔离学习时的新、旧知识初始分布状态 (绿色代表新知识, 橙色代表旧知识, 新旧知识各 10 类); 图 15(b) 是不采用 Margin Loss 约束的区域集成训练后的新、旧知识散布状态; 图 15(c) 是用 Margin Loss 约束训练后的集成效果. 图 15(c) 较图 15(b) 分离良好, 体现了“类内紧簇、类间分离”的效果.

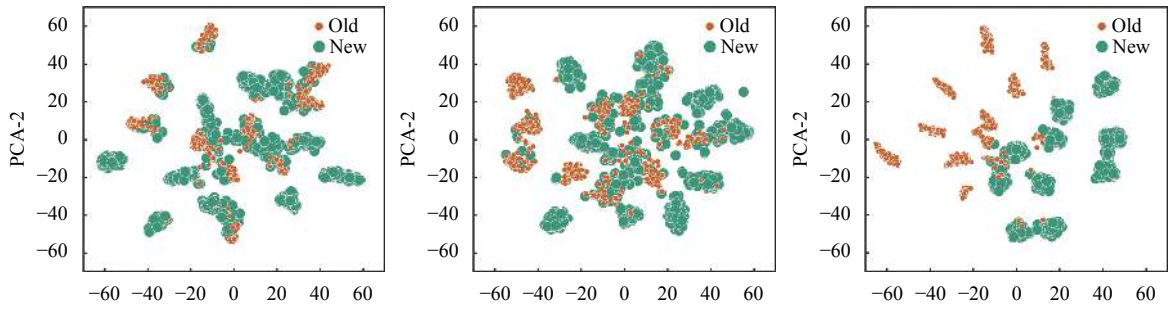


图 15 在 CIFAR-100 上 Margin Loss 避免分类混淆

图 15 显示了“近期偏好”的存在以及 Margin Loss 对其的缓解作用, 这是通过分离新、旧类别的决策边界以形成“类内紧簇、类间分离”的良好分类模型来实现的. 说明 Margin Loss 能够提升识别精度, 因为它能够使模型从偏向于新知识转变为更准确地区分开新知识和旧知识, 即缓解了“近期偏好”.

表 7 显示 Margin Loss 在 CIFAR-100 和 miniImageNet 上的收益, 分别有 3.96% 和 4.93% 的平均性能改善.

表 7　Margin Loss 消融实验精度表 (%)

Dataset	Method	Number of classes (10 stages)										
		10	20	30	40	50	60	70	80	90	100	*Avg*
CIFAR-100	ASPIL($-L_M$)	88.9	82.45	79.15	76.73	72.50	68.21	63.69	59.74	56.27	54.52	70.22
	ASPIL	90.5	**85.5**	**82.52**	**80**	**76.08**	**72.11**	**68.74**	**64.14**	**61.97**	**60.25**	**74.18**
	Gains	1.6↑	3.05↑	3.37↑	3.27↑	3.58↑	3.9↑	5.05↑	4.4↑	5.7↑	5.73↑	3.96↑
miniImageNet	ASPIL($-L_M$)	91.9	83.3	80.53	77.63	73.44	69.15	64.45	60.51	57.1	54.87	71.29
	ASPIL	**92.3**	**87.69**	**84.18**	**81.57**	**77.57**	**73.84**	**71.04**	**67.36**	**64.19**	**62.48**	**76.22**
	Gains	0.4↓	4.39↑	3.65↑	3.94↑	4.13↑	4.69↑	6.59↑	6.85↑	7.09↑	7.61↑	4.93↑

注: ASPIL($-L_M$)代表没有Margin Loss的ASPIL, 在CIFAR-100数据集上进行, 加粗表示各项目中最好的识别结果

4.3.5　各组件随机增减性消融实验比较

为了更好地分析和评价 ASPIL 组件的影响力, 我们还做了可持续区域隔离学习 RI、自学习掩码 SLM、双分支融合 GBF, Margin Loss 损失项 L_M 这 4 个组件的随机增减性消融实验表, 见表 8.

表 8　各组件随机增减性消融实验精度表 (%)

Variations				Number of classes (10 stages)										Avg Acc.	Final Acc.↓
RI	SLM	GBF	L_M	10	20	30	40	50	60	70	80	90	100		
√	√	—	—	88.86	80.75	76.62	74.13	68.85	66.32	63.23	58.98	56.77	54.74	68.93	5.51
√	—	√	—	89.22	81.03	77.26	73.0	68.92	65.81	62.64	57.91	56.25	53.87	68.59	6.38
√	—	—	√	88.4	82.96	77.93	69.85	68.52	65.37	63.92	60.29	57.83	55.33	69.04	4.92
—	√	√	—	88.6	78.60	72.66	67.55	63.94	61.25	58.4	54.36	52.35	49.68	64.74	10.57
—	√	—	√	89.2	79.20	73.25	66.57	64.53	62.88	60.55	56.80	54.11	51.34	65.84	8.91
—		√	√	90.2	79.05	72.97	66.24	64.25	63.0	59.96	56.93	53.81	51.44	65.79	8.81
—	√	√	√	88.5	81.94	76.73	70.74	68.29	66.51	62.33	59.21	55.65	52.43	68.23	7.82
√	—	√	√	90.13	83.42	78.75	74.88	70.94	67.62	64.71	61.61	58.22	55.89	70.62	4.36
√	√	—	√	**90.9**	83.45	79.05	75.0	70.95	67.33	64.07	60.15	58.24	56	70.51	4.25
√	√	√	—	88.9	82.45	79.15	76.73	72.50	68.21	63.69	59.74	56.27	54.52	70.22	5.73
√	√	√	√	90.5	**85.5**	**82.52**	**80**	**76.08**	**72.11**	**68.74**	**64.14**	**61.97**	**60.25**	**74.18**	—

注: 在CIFAR-100数据集上进行, √表示所用组件, RI表示区域隔离学习, L_M 代表Margin Loss损失项, Final Acc.↓代表相比于基线ASPIL的最后阶段识别精度下降量. 加粗标注最好结果, 用红色标注最差结果

分别进行单组件和双组件的消融实验, 以分析各个组件的影响. ① 无论缺失哪些组件都会引起性能下降, 这证实了 ASPIL 各个组件不可或缺. ② 单组件消融实验反映了 RI 和 L_M 正则项的关键性, 具体在于削减 RI 或者 L_M 分别带来 7.82% 和 5.73% 的最终性能下降以及 5.95% 和 3.96% 的平均性能下降. ③ 双组件消融实验中同时保留 RI 和 L_M 的性能衰减最少, 支持了关于单组件消融的分析. 此外, 同时消去 RI 和 L_M, 使得模型承受最严重的损失达 10.57%. 总而言之, ASPIL 各组件都有积极作用, 其中区域隔离是缓解“灾难性遗忘”的重要环节.

4.3.6　旧知识区域比例的影响

(1) 比较方法: 为了进一步了解 ASPIL 中旧知识区域比例对学习过程的作用, 将旧知识区域占模型总体容量的比例固定为 50% 作为基线, 并设置多组不同旧知识区域占比的对照试验, 并在 CIFAR-100 数据集上报告从头开始训练每个增量任务的结果. 运行实验 5 次, 在所有任务完成后报告测试精度的平均值, 见表 9 和表 10.

除了固定旧知识区域占比的方案, 还在 CIFAR-100 数据集上探索了可变旧知识区域占比对增量学习的影响. 具体而言, 首先确定旧知识区域变化空间的最大值和最小值. 在完成基础训练以后, 按照余弦函数形式, 参见公式 (16), 从最小的旧知识区域占比开始, 随增量任务的进行逐渐增加旧知识区域的占比到最大值.

$$ratio_{\text{allocate}} = ratio_{\max} - (ratio_{\max} - ratio_{\min}) \times \cos(\pi/2 \times (t/Stage)) \tag{16}$$

其中, $ratio_{\min}$ 和 $ratio_{\max}$ 分别代表旧知识区域占比变化空间的最小值和最大值. $ratio_{\text{allocate}}$ 代表增量会话 t 时分配

的旧知识区域占比. t 代表增量任务的数量. 例如, 10 阶段训练中包含 1 个基础训练任务和 9 个增量任务, $Stage = 9$. 随着增量任务进行, 任务之间的相互干扰增加, 公式 (16) 能够以更快的速率分配更多的旧知识区域.

表 9　固定旧知识区域占比的 10 阶段识别精度 (%)

Old knowledge ratio (%)	Number of classes										
	10	20	30	40	50	60	70	80	90	100	*Avg*
20	90.44	84.95	82.42	79.90	76.09	71.36	68.55	65.17	61.97	59.46	74.03
30	90.95	84.88	82.32	80.20	75.35	70.76	68.54	65.08	63.30	60.02	74.14
40	90.50	85.50	82.52	80.00	76.08	72.11	68.74	64.14	61.97	60.25	74.18
50	90.55	84.85	81.80	80.26	76.79	72.34	69.60	64.97	62.60	60.32	**74.41**
60	90.75	84.92	81.64	80.13	76.22	72.69	68.83	65.33	62.23	59.79	74.25
70	90.30	84.67	81.70	79.45	75.46	70.28	68.81	64.30	62.17	60.24	73.74
80	90.90	84.50	82.52	80.07	76.07	72.24	69.23	64.90	62.35	59.68	74.25

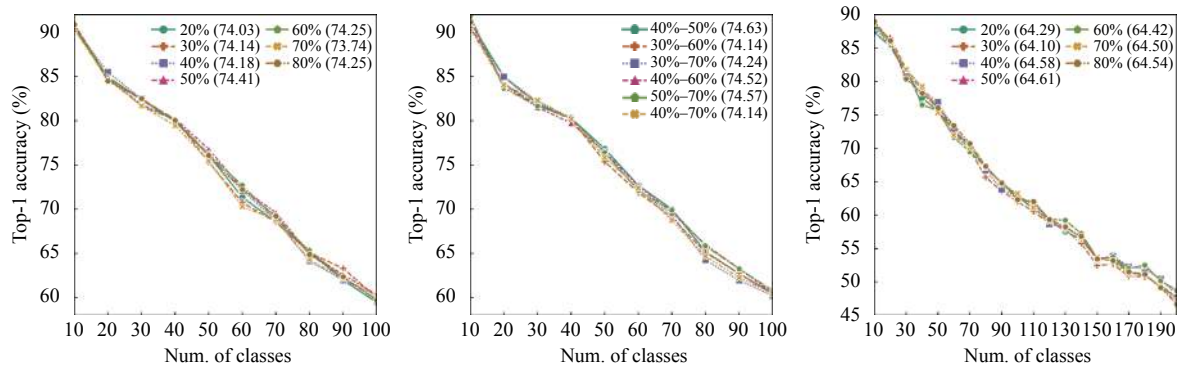
注: 在CIFAR-100数据集上进行, 对照图16查看. *Avg*代表Top-1平均识别精度. 加粗表示最好的平均识别精度

表 10　可变旧知识区域占比的 10 阶段识别精度 (%)

Ratio of old knowledge region (%)	Number of classes										
	10	20	30	40	50	60	70	80	90	100	*Avg*
40–50	91.20	84.95	81.92	80.30	76.84	72.69	70.00	65.18	62.70	60.50	**74.63**
30–60	90.70	84.20	81.75	80.20	75.38	71.89	69.07	65.10	62.70	60.39	74.14
30–70	90.50	85.00	82.19	80.32	76.40	72.19	69.35	64.25	61.93	60.22	74.24
40–60	91.30	84.10	81.52	79.75	76.50	72.64	69.63	65.97	63.29	60.52	74.52
50–70	91.80	83.70	81.65	80.35	76.38	72.09	69.87	65.84	63.25	60.80	74.57
40–70	90.40	84.05	82.32	80.20	75.86	72.39	68.75	64.67	62.26	60.45	74.14

注: 在CIFAR-100数据集上进行, 对照图16查看. *Avg*代表Top-1平均识别精度. 加粗表示最好的平均识别精度

我们还使用了更长的增量任务阶段设置, 将包含 200 类鸟类的 CUB-200[45]数据集平均划分为 20 个学习阶段, 包括 1 次基础训练和 19 次增量训练. 使用了固定旧知识区域占比方案来训练 CUB-200 数据集, 并绘制显示所有 20 个阶段的测试精度和平均值, 见图 16(c).



(a) 10 stages fixed ratio on CIFAR-100　(b) 10 stages variable ratio on CIFAR-100　(c) 20 stages fixed ratio on CUB-200

图 16　旧知识区域比例对模型精度的影响

图 16 中 Top-1 平均精度显示在括号中, 均值来源详见表 9 和表 10. 图 16(a) 和图 16(b) 是 10 阶段增量训练结果, 数据集为 CIFAR-100, 包含固定旧知识占比和可变旧知识占比两种方案. 图 16(c) 是可变旧知识区域占比结果, 在 CUB-200 数据集上进行 20 个阶段的增量训练.

(2) 数据集和实施细节: 我们在两个图像分类数据集 (包括 CIFAR-100[19]和 CUB-200[45]) 上评估了基于 ResNet18 作为骨干网络实现的 ASPIL. CIFAR-100 数据集的介绍见第 4.1 节. CUB-200 数据集是 2010 年由加州理工学院提

出的用于分类识别研究的基准图像数据集. 它包含了 200 个类别的 11 788 张鸟类图像, 每张图像均提供了图像类标记信息, 其中有 5 994 张图像用于训练和 5 794 张供测试.

我们应用第 3.1.1 节的 BN 稀疏区域隔离来调节旧知识区域的占比. 具体地, 先对 BN 权重因子 γ 进行稀疏训练, 然后让所有卷积层所有通道的 γ 因子参与全局 rank 函数排序. 最后从排序后的首个 BN 权重因子开始依次序选取部分 γ 因子, 并将其所对应的通道设置为旧知识区域. 被选取的 γ 因子占因子总数的比例即是旧知识区域占 ASPIL 总体容量的比例.

(3) 实验结果与讨论: 结合图 16、表 9 和表 10, 我们有以下观察结果. 首先, 我们提出的 50% 的旧区域占比方案取得了固定旧知识区域占比中最好的结果, 例如, 在表 9 和表 10 中, 50% 的旧知识区域占比的平均识别精度最高. 第二, 合适的可变旧知识区域占比有着轻微提高 ASPIL 识别精度的潜力. 例如, 40%–50% 的可变旧知识占比达到了所有方案中的最优平均识别精度 74.63%. 40%–60%, 50%–70% 的可变占比也分别取得了 74.52% 和 74.57% 的平均识别精度. 第三, ASPIL 对旧知识区域占比变化并不敏感. 具体而言, 固定旧知识区域占比和可变旧知识区域占比的多阶段识别精度曲线都是接近重合的, 参见图 16. 例如, 表 9 中的平均识别精度变化范围为 0.67% (73.74%–74.41%). 表 10 中的平均识别精度变化范围为 0.49% (74.14%–74.63%).

旧知识区域占比变化并未导致 ASPIL 识别精度的剧烈波动, 可能有以下 3 方面的原因. 首先, ASPIL 的区域分离确实可以很好地保留旧知识. 区域分离是通过调节旧知识区域占比实现的, 为此我们配置了 10% 到 100% 的多个旧知识区域占比, 观察识别精度, 见表 11. 相比于 100% 容量的基线, 30% 以上的旧知识区域都能提升识别精度, 且在仅有 10% 的旧知识区域占比的情况下, 仅有 1.9% 的精度损失. 第二, 基于 BN 稀疏区域隔离的区域分离, 并没有固定各层的剪枝度 (对于深层神经网络而言, 基于网格法来确定所有层的最优剪枝度是比较复杂的), 而是在固定一个模型总体剪枝度的情况下, 使用全局 rank 排序来在所有卷积层的所有通道中选择旧知识区域. 因此, 在使用更小的旧知识区域占比来保留旧知识时, ASPIL 精度更多的受限于模型总体容量, 而较少受到单个卷积层容量的影响. 第三, ASPIL 在区域隔离完成后, 紧接着执行“同步区域分离与集成”, 此时开放全部空间来同时学习新知识和旧知识, 这能在一定程度上抑制由于旧知识区域占比不足而导致的精度衰减.

表 11　多种旧知识区域占比的基础训练精度表 (%)

Old knowledge ratio	100	90	80	70	60	50	40	30	20	10
Top-1 accuracy	89.92	90.85	90.66	90.67	90.83	90.51	90.54	90.4	**89.77**	**88.02**

注: 在CIFAR-100数据集上进行. 使用基于ResNet18实现的ASPIL重复执行10次基础训练(见第3.3节和第3.4.1节的描述), 每次基础训练分别将旧知识区域比例设置为100% (全部容量为旧知识区域的基线模型), 90%, 80%, 70%, 60%, 50%, 40%, 30%, 20%, 10%. 加粗标注低于基线的结果

5 总　结

本文受人脑分区学习和记忆回放巩固知识的过程启发提出 ASPIL 方法, 并通过消融实验证实 ASPIL 组成部分的积极作用, 该方法利用区域隔离学习、自学习掩码、双分支融合和 Margin Loss 正则项来抵抗“灾难性遗忘”. 除精调用于精度恢复以外, ASPIL 主要结构是一个区域隔离和区域集成的两阶段过程. 其中区域隔离独立新学习过程以减少对旧知识的干扰, 区域集成达成统一的、高精度的认知以适配类增量学习的单头输出模型. ASPIL 在实验的全程都领先其他基线方法, 验证了我们的方法的优越性. 目前看来, 使用区域隔离进行增量学习的方法是抑制“灾难性遗忘”的一个很有价值的研究方向, 而从挖掘固定容量网络潜力和节约资源方面来说, 也是符合通用人工智能发展方向的.

References:

[1] Miao YB. Research on image incremental learning based on deep learning [MS. Thesis]. Hangzhou: Zhejiang University of Technology, 2020 (in Chinese with English abstract). [doi: 10.27463/d.cnki.gzgyu.2020.001097]

[2] He L, Han KP, Zhu HX, Liu Y. Deep incremental image classification method based on double-branch iteration. Pattern Recognition and Artificial Intelligence, 2020, 33(2): 150–159 (in Chinese with English abstract). [doi: 10.16451/j.cnki.issn1003-6059.202002007]

[3] Ding SY. Research on augmented class learning [MS. Thesis]. Nanjing: Southeast University, 2019 (in Chinese with English abstract).

[doi: 10.27014/d.cnki.gdnau.2019.001102]

[4] Parisi GI, Kemker R, Part JL, Kanan C, Wermter S. Continual lifelong learning with neural networks: A review. Neural Networks, 2019, 113: 54–71. [doi: 10.1016/j.neunet.2019.01.012]

[5] O'Reilly RC, Bhattacharyya R, Howard MD, Ketz N. Complementary learning systems. Cognitive Science, 2014, 38(6): 1229–1248. [doi: 10.1111/j.1551-6709.2011.01214.x]

[6] O 'Neill J, Pleydell-Bouverie B, Dupret D, Csicsvari J. Play it again: Reactivation of waking experience and memory. Trends in Neurosciences, 2010, 33(5): 220–229. [doi: 10.1016/j.tins.2010.01.006]

[7] Li ZZ, Hoiem D. Learning without forgetting. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2018, 40(12): 2935–2947. [doi: 10.1109/TPAMI.2017.2773081]

[8] Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J, Desjardins G, Rusu AA, Milan K, Quan J, Ramalho T, Grabska-Barwinska A, Hassabis D, Clopath C, Kumaran D, Hadsell R. Overcoming catastrophic forgetting in neural networks. Proc. of the National Academy of Sciences of the United States of America, 2017, 114(13): 3521–3526. [doi: 10.1073/pnas.1611835114]

[9] Zenke F, Poole B, Ganguli S. Continual learning through synaptic intelligence. In: Proc. of the 34th Int'l Conf. on Machine Learning (PMLR). Sydney: JMLR.org, 2017. 3987–3995.

[10] French RM. Catastrophic forgetting in connectionist networks. Trends in Cognitive Sciences, 1999, 3(4): 128–135. [doi: 10.1016/S1364-6613(99)01294-2]

[11] Rebuffi SA, Kolesnikov A, Sperl G, Lampert CH. iCaRL: Incremental classifier and representation learning. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE, 2017. 5533–5542. [doi: 10.1109/CVPR.2017.587]

[12] Chaudhry A, Ranzato MA, Rohrbach M, Elhoseiny M. Efficient lifelong learning with A-GEM. In: Proc. of the 7th Int'l Conf. on Learning Representations (ICLR). New Orleans: OpenReview.net, 2019.

[13] Robins A. Catastrophic forgetting, rehearsal and pseudorehearsal. Connection Science, 1995, 7(2): 123–146. [doi: 10.1080/09540099550039318]

[14] Wang ZF, Jian T, Chowdhury K, Wang YZ, Dy J, Ioannidis S. Learn-prune-share for lifelong learning. In: Proc. of the 2020 IEEE Int'l Conf. on Data Mining (ICDM). Sorrento: IEEE, 2020. 641–650. [doi: 10.1109/ICDM50108.2020.00073]

[15] Kim JY, Choi DW. Split-and-bridge: Adaptable class incremental learning within a single neural network. In: Proc. of the 35th AAAI Conf. on Artificial Intelligence. AAAI, 2021. 8137-8145. [doi: 10.1609/aaai.v35i9.16991]

[16] Han YN, Liu JW, Luo XL. Research progress of continual learning. Journal of Computer Research and Development, 2022, 59(6): 1213–1239 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.20201058]

[17] Liu YY, Schiele B, Sun QR. Adaptive aggregation networks for class-incremental learning. In: Proc. of the 2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). Nashville: IEEE, 2021. 2544–2553. [doi: 10.1109/CVPR46437.2021.00257]

[18] Hou SH, Pan XY, Loy CC, Wang ZL, Lin DH. Learning a unified classifier incrementally via rebalancing. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). Long Beach: IEEE, 2019. 831–839. [doi: 10.1109/CVPR.2019.00092]

[19] Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images. Handbook of Systemic Autoimmune Diseases, 2009, 1(4).

[20] Vinyals O, Blundell C, Lillicrap T, Kavukcuoglu K, Wierstra D. Matching networks for one shot learning. In: Proc. of the 30th Int'l Conf. on Neural Information Processing Systems. Barcelona: Curran Associates Inc., 2016. 3637–3645.

[21] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv:1503.02531, 2015.

[22] Howard J, Ruder S. Universal language model fine-tuning for text classification. In: Proc. of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL, 2018. 328–339. [doi: 10.18653/v1/P18-1031]

[23] Rannen A, Aljundi R, Blaschko MB, Tuytelaars T. Encoder based lifelong learning. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision (ICCV). Venice: IEEE, 2017. 1329–1337. [doi: 10.1109/ICCV.2017.148]

[24] Schwarz J, Czarnecki W, Luketina J, Grabska-Barwinska A, Teh YW, Pascanu R, Hadsell R. Progress & compress: A scalable framework for continual learning. In: Proc. of the 35th Int'l Conf. on Machine Learning. Stockholm: ACM, 2018. 4535–4544.

[25] Liu XL, Masana M, Herranz L, Van De Weijer J, Lopez AM, Bagdanov AD. Rotate your networks: Better weight consolidation and less catastrophic forgetting. In: Proc. of the 24th Int'l Conf. on Pattern Recognition (ICPR). Beijing: IEEE, 2018. 2262–2268. [doi: 10.1109/ICPR.2018.8545895]

[26] Aljundi R, Babiloni F, Elhoseiny M, Rohrbach M, Tuytelaars T. Memory aware synapses: Learning what (not) to forget. In: Proc. of the 15th European Conf. on Computer Vision. Munich: Springer, 2018. 144–161. [doi: 10.1007/978-3-030-01219-9_9]

[27] Wu Y, Chen YP, Wang LJ, Ye YC, Liu ZC, Guo YD, Fu Y. Large scale incremental learning. In: Proc. of the 2019 IEEE/CVF Conf. on

Computer Vision and Pattern Recognition (CVPR). Long Beach: IEEE, 2019. 374–382. [doi: 10.1109/CVPR.2019.00046]

[28] Dhar P, Singh RV, Peng KC, Wu ZY, Chellappa R. Learning without memorizing. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). Long Beach: IEEE, 2019. 5133–5141. [doi: 10.1109/cvpr.2019.00528]

[29] Zhang JT, Zhang J, Ghosh S, Li DW, Tasci S, Heck L, Zhang HM, Kuo CCJ. Class-incremental learning via deep model consolidation. In: Proc. of the 2020 IEEE Winter Conf. on Applications of Computer Vision (WACV). Snowmass: IEEE, 2020. 1120–1129. [doi: 10.1109/WACV45572.2020.9093365]

[30] Castro FM, Marín-Jiménez MJ, Guil N, Schmid C, Alahari K. End-to-end incremental learning. In: Proc. of the 15th European Conf. on Computer Vision. Munich: Springer, 2018. 241–257. [doi: 10.1007/978-3-030-01258-8_15]

[31] Mai ZD, Li RW, Kim H, Sanner S. Supervised contrastive replay: Revisiting the nearest class mean classifier in online class-incremental continual learning. In: Proc. of the 2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW). Nashville: IEEE, 2021. 3584–3594. [doi: 10.1109/CVPRW53098.2021.00398]

[32] Smith J, Hsu YC, Balloch J, Shen YL, Jin HX, Kira Z. Always be dreaming: A new approach for data-free class-incremental learning. In: Proc. of the 2021 IEEE/CVF Int'l Conf. on Computer Vision (ICCV). Montreal: IEEE, 2021. 9354–9364. [doi: 10.1109/ICCV48922.2021.00924]

[33] Hung SCY, Tu CH, Wu CE, Chen CH, Chan YM, Chen CS. Compacting, picking and growing for unforgetting continual learning. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 1225.

[34] Mehta N, Liang KJ, Verma VK, Carin L. Continual learning using a Bayesian nonparametric dictionary of weight factors. arXiv:2004.10098, 2020.

[35] Luo JH, Wu JX, Lin WY. ThiNet: A filter level pruning method for deep neural network compression. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision (ICCV). Venice: IEEE, 2017. 5068–5076. [doi: 10.1109/ICCV.2017.541]

[36] Golkar S, Kagan M, Cho K. Continual learning via neural pruning. arXiv:1903.04476, 2019.

[37] Cortes C, Vapnik V. Support-vector networks. Machine Learning, 1995, 20(3): 273–297. [doi: 10.1023/A:1022627411411]

[38] Liu Z, Li JG, Shen ZQ, Huang G, Yan SM, Zhang CS. Learning efficient convolutional networks through network slimming. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision (ICCV). Venice: IEEE, 2017. 2755–2763. [doi: 10.1109/ICCV.2017.298]

[39] Cao Y, Xu JR, Lin S, Wei FY, Hu H. GCNet: Non-local networks meet squeeze-excitation networks and beyond. In: Proc. of the 2019 IEEE/CVF Int'l Conf. on Computer Vision Workshop (ICCVW). Seoul: IEEE, 2019. 1971–1980. [doi: 10.1109/ICCVW.2019.00246]

[40] Liao MH, Wan ZY, Yao C, Chen K, Bai X. Real-time scene text detection with differentiable binarization. In: Proc. of the 34th AAAI Conf. on Artificial Intelligence. New York: AAAI Press, 2020. 11474–11481. [doi: 10.1609/aaai.v34i07.6812]

[41] Li XT, Zhao HL, Han L, Tong YH, Tan SH, Yang KY. Gated fully fusion for semantic segmentation. In: Proc. of the 34th AAAI Conf. on Artificial Intelligence (AAAI). New York: AAAI Press, 2020. 11418–11425. [doi: 10.1609/aaai.v34i07.6805]

[42] He KM, Zhang XY, Ren SQ, Sun J. Deep residual learning for image recognition. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE, 2016. 770–778. [doi: 10.1109/CVPR.2016.90]

[43] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.

[44] Chaudhry A, Rohrbach M, Elhoseiny M, Ajanthan T, Dokania PK, Torr PHS, Ranzato MA. On tiny episodic memories in continual learning. arXiv:1902.10486, 2019.

[45] Wah C, Branson S, Welinder P, Perona P, Belongie S. The caltech-ucsd birds-200–2011 dataset. 2011. https://resolver.caltech.edu/CaltechAUTHORS:20111026-120541847

附中文参考文献:

[1] 缪永彪. 基于深度学习的图像增量学习研究 [硕士学位论文]. 杭州: 浙江工业大学, 2020. [doi: 10.27463/d.cnki.gzgyu.2020.001097]

[2] 何丽, 韩克平, 朱泓西, 刘颖. 双分支迭代的深度增量图像分类方法. 模式识别与人工智能, 2020, 33(2): 150–159. [doi: 10.16451/j.cnki.issn1003-6059.202002007]

[3] 丁思宇. 增量类学习若干问题研究 [硕士学位论文]. 南京: 东南大学, 2019. [doi: 10.27014/d.cnki.gdnau.2019.001102]

[16] 韩亚楠, 刘建伟, 罗雄麟. 连续学习研究进展. 计算机研究与发展, 2022, 59(6): 1213–1239. [doi: 10.7544/issn1000-1239.20201058]

附录 A. GBF 融合参数形式及比较

融合参数实验的目的是比较不同融合控制参数形式的效果. 融合控制参数控制着双分支的融合参与程度, 这

直接影响着 GBF 的性能高低. 与其他方法相比, GBF 使用一次函数进行分支融合获得了最好的结果. 为了更好地比较, 添加了没有分支融合的 baseline 条目进行对照.

融合控制参数 a 的表达形式见公式 (A1), 其中 E 代表 $epoch$ 最大值, 一般为 100.

$$a=\begin{cases}\max(0.5-0.8\times epoch/2E,0) & \text{GBF}\\ 0.5\times\sin((2\pi/2E)\times epoch) & \text{sin函数参与度}\\ 0.5\times\cos((2\pi/4E)\times epoch) & \text{cos函数参与度}\\ \max(0.5-\phi\times epoch^2,0) & \text{二次函数参与度}\end{cases} \tag{A1}$$

图 A1 中 GBF 融合控制参数 a 被初始化为 0.5, 此时两个分支公平参与, 并在 $0.8E$ 个 epoch 后完全解除对辅助分支的依赖. 最终网络只依靠主分支识别任务, 因此无需对辅助分支实施知识保护, 例如区域隔离学习操作. 特别地, 当 a 恒等于 0 时为主分支跨层融合, 辅助分支的新知识得不到利用. 可以看出, GBF 的分支融合效果最好, cos 和二次函数作为融合控制函数的优化效果低于 GBF, sin 函数则是负优化.

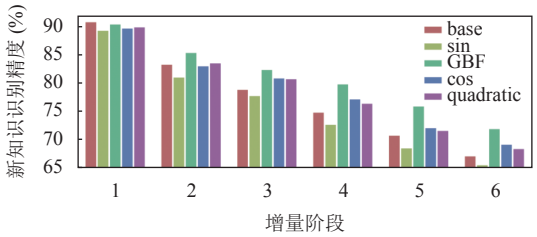


图 A1　GBF 多种融合参数形式的新知识识别精度条形图

a 为 1 时完全参与融合, 0 代表不参与融合. 对于 sin 函数, 辅助分支在 $E/2$ 时达到最大参与度 0.5, E 时衰减为 0; 对于 cos 函数, 辅助分支参与度从 0.5 开始, 经过 E 个 epoch 后, 逐渐衰减为 0; 对于二次函数, 通过 $\varphi=5\text{E}{-4}$ 控制辅助分支的参与度在 E 个 epoch 后逐渐减少为 0. 实验结果见图 A1.

附录 B.　GBF on ResNet50

本文还将 GBF 推广到 ResNet50, 实验遵循第 4.3.3 节的 GBF 融合设定, 观察到和正文表 6 一致的结果.

由于 ResNet50 比 ResNet18 层次更深, 为了加速训练和减小训练难度, 做了如下修改使 GBF 在 ResNet50 上适应良好, 实验结果见表 B1.

(1) 减少门控数量, 降低融合难度. 门控越多, 融合的难度也越大. 对于来自辅助分支某层的信息可以与对应的主分支层直接执行通道维度的拼接, 在拼接后的结果的基础上, 再实施 GBF 融合机制.

(2) 减少融合过程的卷积层添加, 尽量将融合过程适配原有的网络结构, 减小训练负担.

表 B1　带 GBF 的精度表 (%)

Method	Number of classes (10 stages)										
	10	20	30	40	50	60	70	80	90	100	*Avg*
Baseline	90.9	83.45	79.05	75	70.95	67.33	64.07	60.15	58.24	56	70.51
GBF (ResNet18)	90.5	85.5	82.52	80	76.08	72.11	68.74	64.14	61.97	60.25	74.18
GBF (ResNet50)	**92.3**	**87.80**	**84.5**	**81.46**	**78.27**	**74.2**	**70.17**	**66.12**	**62.56**	**61.55**	**75.90**
Gains	1.4↑	4.35↑	5.45↑	6.46↑	7.32↑	6.87↑	6.1↑	5.97↑	4.32↑	5.55↑	5.38↑

注: 实验在ResNet50网络上用CIFAR-100数据集也取得了同样的效果. 相比于Baseline, GBF获得了高达5.38%的平均精度提升

附录 C.　训练算法

训练流程由“基础训练”和“增量训练”组成. “增量训练”流程主要包含 3 个步骤: 依次为“隔离学习”“同步区域分离与集成”“精调”. “基础训练”有 2 个阶段, 依次为“初始化训练”和“区域分离”. 算法 C1、算法 C2 为顶层算法, 显示“基础训练”和“增量训练”的划分. 算法 C3 是对正文第 3.1 节区域隔离学习的展示. 算法 C4 显示各阶段损失函数调用, 算法 C5 向算法 C4 供给辅助分支信息.

算法 C1. Top level algorithm.

输入: $X^s,\ldots,X^t$, $P=\{p^1,\ldots,p^{s-1}\}$; // 训练图片类别 $s,\ldots,t$, 样本集 P, 当 $s=1$ 时, $P=\Phi$
输出: θ^{main}. // θ^{main} 是主分支参数, θ^{aux} 是辅助分支参数

1. **FOR** INCREMENTAL = $0,\ldots,N$ **DO** //N 为增量过程标识符
2. Prepare training set $D=\{X^s,\ldots,X^t\}\cup P$
3. **IF NOT** INCREMENTAL, **THEN** //基础训练
4. **FOR** ITER = 0, 1 **DO** //包括初始化训练和区域分离
5. $\theta^{\text{main}}\leftarrow$TRAIN(INCREMENTAL, ITER, D)
6. **END FOR**
7. **ELSE** //增量训练
8. **FOR** ITER = 0, 1, 2 **DO** //增量训练的 3 个阶段
9. $\theta^{\text{main}}\leftarrow$ Train(INCREMENTAL, ITER, D)
10. **END FOR**
11. **END IF**
12. Update exemplar set P
13. **END FOR**

算法 C2. Train.

输入: EPOCH, LR, BATCHSIZE; //LR 是学习率
INCREMENTAL $\in\{0,\ldots,N\}$, D, ITER $\in\{0, 1, 2\}$ or $\{0, 1\}$; //增量过程标识符 N, 训练集 D
输出: θ^{main}.

1. **FOR** $I=1,\ldots,$ EPOCH **DO** //每个阶段训练次数, EPOCH=100
2. $\theta^{\text{main}}\leftarrow$UpdateMainBranchModel(INCREMENTAL, ITER, D)
3. **END FOR** //剪枝保留通道为旧知识

算法 C3. UpdateMainBranchModel.

输入: INCREMENTAL, D, ITER, BATCHSIZE, LR; //训练阶段标识符 ITER, 基础训练 ITER=2, 增量训练 ITER=3
输出: θ^{main}.

1. **FOR** STEP = $1,\ldots,$ M **DO** //一个 EPOCH 有 M 个 STEP
2. **IF** INCREMENTAL **AND NOT** ITER **AND** EPOCH==1 **AND** STEP==1, **THEN**
3. Record old knowledge channel and store it in list V
4. **END IF**
5. $\theta^{\text{main}}\leftarrow$UpdateModelParameters(INCREMENTAL, ITER, D)
6. **IF** ITER == 1 **THEN** //初始化训练和精调不进入如下处理
7. CHANNEL_SPARSITY(θ^{main}) //通道稀疏达成区域分离
8. **ELSEIF** INCREMENTAL **AND NOT** ITER, **THEN** //增量训练阶段 1: 隔离学习
9. ISOLATED_LEARNING(θ^{main}, V) //此时冻结旧知识区域的权重参数
10. **END IF**
11. **END FOR** //剪枝保留通道为旧知识

算法 C4. UpdateModelParameters.

输入: INCREMENTAL, D, ITER, BATCHSIZE, LR, θ^{aux}; //隔离学习结束后初始化 θ^{aux}
输出: θ^{main}.

1. **IF** INCREMENTAL **AND** ITER, **THEN** //区域集成: 同步区域分离与集成, 精调
2. 来自辅助分支的融合特征, $\theta^{aux} \leftarrow$UpdateAuxiliaryBranchModel(θ^{aux})
3. 主分支和辅助分支信息融合, 前向传播
4. $\theta^{main} \leftarrow$使用公式 (11) 和训练集 D 训练网络
5. **ELSE** //包括: 基础训练和隔离学习
6. $\theta^{main} \leftarrow$使用公式 (8) 和训练集 D 训练网络
7. **END IF**

算法 C5. UpdateAuxiliaryBranchModel.

输入: $X^s,\ldots,X^t$ 和 θ^{aux}; // θ^{aux} 是辅助分支参数
输出: 来自辅助分支的融合特征, θ^{aux+}.

1. 使用最新数据集 $\{X^s,\ldots,X^t\}$ 和公式 (8) 训练辅助分支, 得到更新值 θ^{aux+}
2. 训练后取出辅助分支层信息

姚红革(1968－), 男, 博士, 副教授, 主要研究领域为人工智能, 计算机视觉.

程嗣怡(1980－), 男, 博士, 主要研究领域为机器学习, 电子对抗理论与技术.

邬子逸(1996－), 男, 硕士生, 主要研究领域为元强化学习, 小样本类增量学习.

陈游(1983－), 男, 博士, 副教授, 主要研究领域为雷达信号处理, 信息对抗理论.

马姣姣(1997－), 女, 硕士生, 主要研究领域为机器学习, 计算机视觉.

喻钧(1970－), 女, 硕士, 教授, 主要研究领域为图像处理与模式识别, 计算机网络与信息安全, 无线传感器网络.

石俊(1972－), 男, 博士, 讲师, 主要研究领域为机器学习, 计算机视觉, 无人机控制.

姜虹(1977－), 女, 博士, 副教授, 主要研究领域为软件工程, 图像处理.