

持续学习的研究进展与趋势

李文斌¹ 熊亚锟¹ 范祉辰¹ 邓波² 曹付元^{3,4} 高阳¹

¹(计算机软件新技术国家重点实验室(南京大学) 南京 210023)

²(中国科学院空天信息创新研究院 北京 100094)

³(山西大学计算机与信息技术学院(大数据学院) 太原 030006)

⁴(山西大学大数据学院 太原 030006)

(liwenbin@nju.edu.cn)

Advances and Trends of Continual Learning

Li Wenbin¹, Xiong Yakun¹, Fan Zhichen¹, Deng Bo², Cao Fuyuan^{3,4}, and Gao Yang¹

¹(State Key Laboratory for Novel Software Technology (Nanjing University), Nanjing 210023)

²(Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094)

³(School of Computer and Information Technology, Shanxi University, Taiyuan 030006)

⁴(School of Big Data, Shanxi University, Taiyuan 030006)

Abstract With the development and successful application of deep learning, continual learning has attracted increasing attention and has been a hot topic in the field of machine learning, especially in the resource-limited and data-security scenarios with the increasing requirements of quickly learning sequential tasks and data. Unlike humans who enjoy the ability of continually learning and transferring knowledge, the existing deep learning models are prone to easily suffering from a catastrophic forgetting problem in a sequential learning process. Therefore, how to continually learn new knowledge and retain old knowledge at the same time on dynamic and non-stationary sequential task and streaming data, is the core of continual learning. Firstly, through the investigation and summary of the related work of continual learning at home and abroad in recent years, continual learning methods can be roughly divided into three categories: replay-based, constraint-based, and architecture-based. We further subdivide these three types of methods. Specifically, the replay-based methods are subdivided into three categories: sample replay, generation replay, and pseudo-sample replay, according to the sample's sources used; the constraint-based methods are subdivided into parameter constraints, gradient constraints, and data constraints, according to the constraint's sources; the architecture-based methods are subdivided into two categories: parameter isolation and model expansion, according to how the model structure is used. By comparing the innovation points of related work, the advantages and disadvantages of various methods are summarized. Secondly, the research progress at home and abroad is analyzed. Finally, the future development direction of continual learning combined with other fields is simply prospected.

Key words deep learning; knowledge transfer; continual learning; catastrophic forgetting; sequential task

摘要 随着深度学习技术的发展与应用,特别是资源受限场景和数据安全场景对序列任务和数据进行快速学习需求的增多,持续学习逐渐成为机器学习领域关注的一个新热点。不同于人类所具备的持续学

收稿日期: 2022-09-23; 修回日期: 2023-09-11

基金项目: 国家自然科学基金项目(62192783, 62106100); 江苏省自然科学基金项目(BK20221441); 江苏省双创博士计划项目(JSSCBS20210021)

This work was supported by the National Natural Science Foundation of China (62192783, 62106100), the Natural Science Foundation of Jiangsu Province (BK20221441), and the Jiangsu Provincial Double-Innovation Doctor Program (JSSCBS20210021).

通信作者: 高阳(gaoy@nju.edu.cn)

习和迁移知识的能力,现有深度学习模型在序列学习过程中容易遭受灾难性遗忘的问题。因此,如何在动态、非平稳的序列任务及流式数据中不断学习新知识、同时保留旧知识是持续学习研究的核心。首先,通过对近年来持续学习国内外相关工作的调研与总结,将持续学习方法分为基于回放、基于约束、基于结构三大类,并对这3类方法做进一步的细分。具体而言,根据所使用的样本来源将基于回放的方法细分为采样回放、生成回放、伪样本回放3类;根据训练约束的来源将基于约束的方法细分为参数约束、梯度约束、数据约束3类;根据对于模型结构的使用方式将基于结构的方法细分为参数隔离、模型拓展2类。通过对比相关工作的创新点,对各类方法的优缺点进行总结。其次,对国内外研究现状进行分析。最后,针对持续学习与其他领域相结合的未来发展方向进行展望。

关键词 深度学习;知识迁移;持续学习;灾难性遗忘;序列任务

中图法分类号 TP391

目前,深度学习算法和模型凭借其优异的性能在众多任务和场景中取得巨大的成功,特别是在图像识别^[1-3]、自然语言处理^[4-6]和视频分析^[7-9]等领域,甚至在具体图像分类、实例分割任务上超越了人类的水平^[10]。然而,当面临新任务,特别是分布差异大的序列任务学习时,深度学习模型的表现就难以达到预期,往往会出现迅速破坏之前学到的旧任务知识的现象,即灾难性遗忘(catastrophic forgetting)。造成深度学习模型这2种截然不同表现的原因在于,在传统离线非序列任务场景下,深度学习模型可以不断地重复交错访问和学习任务数据中的知识直到模型饱和,然而在真实序列或在线场景下,旧的任务和数据无法被交错访问和读取,导致旧任务的知识及其关联的模型参数逐渐被新的任务和数据擦除和覆盖。

不同于深度学习模型,人类等高级动物虽然也会逐渐忘记以前学到的信息,但很少有新信息的学习会对巩固的知识造成灾难性的干扰,即人类天生具备序列学习和抵抗灾难性遗忘的能力,并且能够利用旧任务的知识帮助新任务的学习,同时将新任务知识整合进现有知识体系^[11-12]。例如,人类在小时候学会了骑自行车、游泳等技能,即便多年没再实践过,长大后在需要时仍然能够轻易上手,进一步还能帮助其快速学习骑摩托车和其他水上运动。换言之,人类具有终生学习的能力,这是目前机器学习和深度学习算法所不具备的。根据目前生物学和神经科学的发现,人类之所以能够进行持续学习,是因为大脑中存在一套丰富的神经生理学机制能够很好地调控不同脑区的稳定性-可塑性的平衡(stability-plasticity balance)^[13-15],在早期发育的关键时期,大脑的可塑性特别强,随着生物系统在一系列特定的发育阶段稳定下来,可塑性变得不那么突出,在较小规模上保持

一定程度的可塑性,以快速适应环境和重组知识。此外,互补学习系统理论(complementary learning systems theory)认为智能主体必须拥有2个学习系统,即在哺乳动物的新皮质和在海马中实例化。大脑海马体系统具有短期适应性,可以快速学习新的信息和知识,而大脑新皮层系统具有长期保持性,二者互补的特性使得人类认知系统可实现短期学习和长期记忆^[16-19]。因此,受人类认知系统的启发,在机器学习领域,如何有效克服灾难性遗忘,在记忆旧知识的同时通过迁移加快新任务的学习,使得深度学习模型在序列任务数据中持续不断地学习,即持续学习(continual learning),其得到广泛关注和研究,具有重要科学研究价值和应用价值。

1 研究背景

1.1 持续学习的背景介绍

持续学习的主要目标是从持续到来的不同任务中学习并积累知识,逐渐扩充模型的能力。由此可见,持续学习面对的是一个任务顺序到来的场景,“顺序”是指当前任务的数据在后续任务中不会再出现,即所有数据以流的形式存在。持续学习的关键是在序列任务下如何平衡模型的可塑性(plasticity)与稳定性(stability)^[15,20],既要求模型能从新任务中学到所需的知识,即可塑性,又使得模型能尽量保留完成旧任务的必备经验,即稳定性。然而,如何有效平衡模型的可塑性和稳定性仍然是一个非常具有挑战性的问题,也是当前持续学习方法研究和设计的核心。

持续学习与现有机器学习多个研究方向关系紧密却又不同。具体表现为:1)经典机器/深度学习^[21]往往只关注单个任务、单一目标的最优表现,假设数据离线存储并且可以反复访问。而持续学习关注流式

序列任务学习, 假设任务流式访问并且只能访问1次, 其核心目标在于取得所有任务平均最优表现. 2) 迁移学习^[22]往往只在单个源任务上进行学习, 并且只关注目标任务上的性能, 持续学习则希望所有学过的任务都能表现最好. 3) 多任务学习^[23]并行地学习多个不同类型的相关任务, 让这些任务在学习过程中共享知识, 并利用多个任务之间的相关性来改进模型在每个任务上的性能和泛化能力, 并且假设数据是离线的、不同任务使用相同数据. 与多任务学习不同, 持续学习假设数据是流式的, 串行地学习多个同类型任务, 不同任务使用的数据是完全不同的. 4) 元学习^[24]的目的在于学习大量被构造的辅助任务后, 能够在少量样本的情况下加快在目标任务上的学习和泛化, 但是对于辅助数据的假设仍然是离线的, 并且训练任务间存在交叉和重叠. 不同于元学习更加关注目标任务的性能, 持续学习则更多关注所有任务的平均性能. 5) 在线学习^[25]与持续学习关系密切, 但是在线学习假设数据是独立同分布的, 并且其核心还是在于单个任务的学习, 而持续学习中数据的分布往往存在差异甚至领域发生变化, 并且持续学习更多关注多个任务的学习和记忆.

1.2 持续学习的设定

在广义的持续学习定义中, 任务之间完全不同, 比如前一个任务是图像分类, 后一个任务可能就变成了实例分割等其他计算机视觉任务, 甚至可以是自然语言处理领域的相关任务^[20]. 显然, 这种广义的持续学习设定在现阶段展开研究还存在较大困难, 为了能逐步实现目标, 在目前大多数工作中, 研究人员适当限制了任务种类和难度, 重点关注持续学习中灾难性遗忘的问题. 目前持续学习设置了3种常见的场景: 任务增量学习(task-incremental learning)、类别增量学习(class-incremental learning)和域增量学习(domain-incremental learning).

任务增量学习场景是指不同时刻到来的数据分属于不同的任务, 在一个时间段内我们可以获得当前任务的全部数据, 并且假设测试推理阶段任务的编号(task-ID)是可知的. 因为测试阶段任务编号可知, 所以训练阶段不同任务的输出相互独立, 即最终学到的模型是多头的(multi-head), 为每个任务提供单独的输出层. 因此, 针对任务增量场景, 模型在训练和测试阶段均会提供数据对应的任务编号. 类别增量学习场景是指不同时刻到来的数据属于同一类型任务的不同类别, 随着训练的进行逐步增加输出类别. 不同于任务增量学习场景, 类别增量学习场景在

训练和测试阶段不知道任务的编号, 因此比任务增量场景更具挑战性. 域增量学习场景是指不同时刻到来的数据属于不同任务, 但是任务之间的类别空间是相同的, 只是数据的分布/域发生了变化, 前后2个任务的数据不再满足独立同分布假设.

持续学习目前主流领域重点关注图像分类任务以及监督学习范式下的应用, 常用的数据集包括CIFAR-100^[26], Tiny-ImageNet^[27], mini-ImageNet^[28]. 在模型训练和测试时通常会根据数据集的特点进行相应的任务划分. 以CIFAR-100数据集为例, 其类别数为100, 目前的划分方式主要有3种: 1) 5个任务, 每个任务20个类; 2) 20个任务, 每个任务5个类; 3) 11个任务, 第1个任务50个类, 其他每个任务5个类.

在持续学习模型评估时, 按照上述其中一种划分方式将数据集划分为 T 个任务, 持续学习算法在这 T 个任务上依次进行训练与测试, 得到一个 $T \times T$ 大小的精度矩阵 \mathbf{R} , 其中 $R_{i,j}$ 表示模型学过任务 i 后在任务 j 上的测试分类精度. 显而易见, 对于某个测试分类精度 $R_{i,j}$, 若 $i > j$, 则刻画的是经过任务 i 后任务 j 被遗忘的程度, 反之则反映了模型将任务 i 的知识迁移到任务 j 上的能力. 持续学习中通常定义了3种评价指标, 即平均精度(average accuracy, ACC)、反向迁移精度(backward transfer accuracy, BWT)、前向迁移精度(forward transfer accuracy, FWT):

$$ACC = \frac{1}{T} \sum_{i=1}^T R_{T,i}, \quad (1)$$

$$BWT = \frac{1}{T-1} \sum_{i=1}^{T-1} R_{T,i} - R_{i,i}, \quad (2)$$

$$FWT = \frac{1}{T-1} \sum_{i=2}^T R_{i-1,i} - R_{0,i}, \quad (3)$$

其中 $R_{0,i}$ 表示随机初始化后直接在任务 i 上进行测试的精度. 平均精度衡量了模型学过所有任务后的平均性能, 反向迁移精度衡量了模型在学习过程中对之前任务的遗忘程度, 而前向迁移精度衡量的是之前的知识对后续学习过程可能发生的影响.

此外, 根据对当前任务数据集的利用程度, 持续学习的设定有在线(online)和离线(offline)之分. 在任务顺序到来的基础上, 在线设置下, 持续学习只允许模型学习一轮当前任务的数据集, 要求学习算法能快速适应任务. 而离线设置下, 持续学习对当前任务数据的学习次数没有限制, 在下一个任务到来之前, 模型可以在当前任务的数据集上训练多次以充分挖掘任务相关知识.

1.3 持续学习方法分类

由于持续学习的应用场景非常广泛, 本文主要关注持续学习方法在图像分类任务场景中的应用. 本文按照持续学习方法避免灾难性遗忘所采取的策略, 将持续学习方法进行分类并介绍, 如图 1 所示. 具体而言, 本文将持续学习方法分为基于回放(replay-based)、基于约束(constraint-based)、基于结构(architecture-based)三大类, 其中基于回放的方法使用各种方式保留之前任务的知识, 基于约束的方

法使用了正则项、知识蒸馏等技术来缓解灾难性遗忘问题, 基于结构的方法通过动态地修改模型结构来实现持续学习. 本文进一步对这 3 类方法做了细分, 根据所使用的样本来源将基于回放的方法细分为采样回放、生成回放、伪样本回放 3 类; 根据训练约束的来源将基于约束的方法细分为参数约束、梯度约束、数据约束 3 类; 根据对于模型结构的使用方式将基于结构的方法细分为参数隔离、模型拓展 2 类.

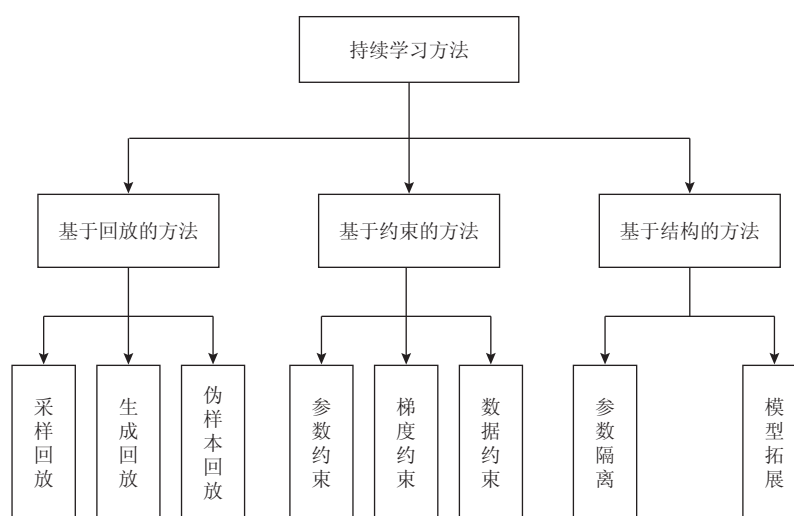


Fig. 1 Illustration of the categories of continual learning methods

图 1 持续学习方法类别示意图

2 研究现状

2.1 基于回放的方法

互补学习系统(CLS)理论上认为智能主体必须拥有 2 个学习系统, 一个是获得结构化的知识表示, 另一个是快速学习部分实例的细节. 人脑中, 海马系统具有短期适应性, 负责快速学习新的信息和知识, 而大脑新皮质系统具有长期保持性, 负责维护之前学过的知识. 该理论与持续学习有着很强的相关性. 受该理论的启发, 持续学习中基于回放的这一类方法通过引入一个额外的内存系统来充当新皮质维护旧知识的作用. 根据额外的内存系统中样本来源的不同将基于回放的方法进一步地细分.

这一类方法的核心主要是针对持续学习设定中无法接触旧任务数据这一限制, 通过让部分旧任务数据出现在后续任务的训练过程中, 缓解持续学习模型灾难性遗忘的程度. 根据获得旧任务数据的方式可以进一步分为采样回放方法、生成回放方法、

伪样本回放方法. 采样回放方法通常申请一个固定大小的内存空间用来保存从旧任务数据中采样出的一小部分样本. 生成回放方法和伪样本回放方法都属于在后续任务训练时, 通过模型合成旧任务数据. 二者区别在于, 生成回放方法通常会基于生成对抗网络(generative adversarial network, GAN)来训练并保存一个能够合成当前任务数据的生成模型, 并在后续任务的训练过程中通过合成大量旧任务数据用于训练, 来缓解灾难性遗忘. 而伪样本回放方法只借助分类模型本身, 通过对随机输入进行不断地优化, 最终合成旧任务数据.

2.1.1 采样回放方法

该方法采用的策略是学习完每个任务后保存一部分当前任务的原始样本在固定大小的内存空间里, 并在后续任务的学习过程中进行样本回放. 这一过程类似人脑将新信息快速存储在海马体中, 然后缓慢地整合到新皮质系统中. 通过新旧信息的不断回放, 以尽量减少新信息整合过程对现有知识的干扰. 如图 2 所示, 把所有存储样本 M 作为任务 T_t 的训

练样本的一部分,在第 t 个任务 T_t 结束时,从当前任务数据 U_t 中保存一小部分样本 M_t 至内存中,图2中 M_t 代表第 t 个任务所保存的样本集合.值得注意的是,由于存储空间的大小是固定的,所以随着保存的类别不断增多,分配给每个类别的样本数量将不断减少,因此需要不断地为每个类别删减样本以保持存储空间大小固定.

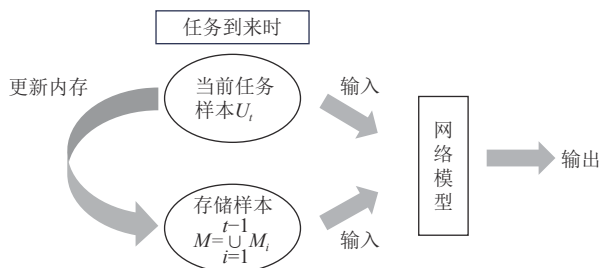


Fig. 2 Illustration of sampling replay methods

图2 采样回放方法示意图

在这类方法中,一部分工作重点关注保存怎样的样本才能更好地帮助持续学习模型减轻遗忘.为了解决这个问题,Rebuffi等人^[29]通过特征均值为每个类计算原型,并且优先保存更靠近原型的样本,认为这部分样本更能代表该类的特征.在删除样本时,优先删除距离原型较远的,即当初保存时后加入的样本.文献^[30]选择保存更“难”的样本,认为被保存的样本不仅应该能够代表这个类,更应该能够与其他类辨别开.而且认为处于类分布中心的样本更具有代表性,处于分类边界上的样本更具有辨别性.如图3所示,通过计算一个样本在不同数据增强方法下对应的模型输出的方差,来判断一个样本的不确定性,并认为方差越大的样本,不确定性越强,也更靠近分类边界.具体计算公式为:

$$S_c = \sum_{i=1}^M 1_c \arg \max_{\hat{c}} p(y = \hat{c} | \tilde{x}_t), \quad (4)$$

$$u(x) = 1 - \frac{1}{M} \max_c S_c, \quad (5)$$

其中 M 表示对一个样本通过数据增强得到的扰动样本数量; x 表示扰动样本; 1_c 为指示函数,当模型对扰动样本的预测结果为 c 时, $1_c=1$,否则 $1_c=0$; $u(x)$ 表示样本 x 的不确定性.

因为通常用于保存样本的内存大小是固定的,所以另外一部分工作重点关注内存空间的利用率,即如何利用固定大小的内存来保存更多的样本.文献^[31]指出,用uint8格式保存一张 $256 \times 256 \times 3$ 的图片需要占用192 KB,而用float32格式保存该图片对

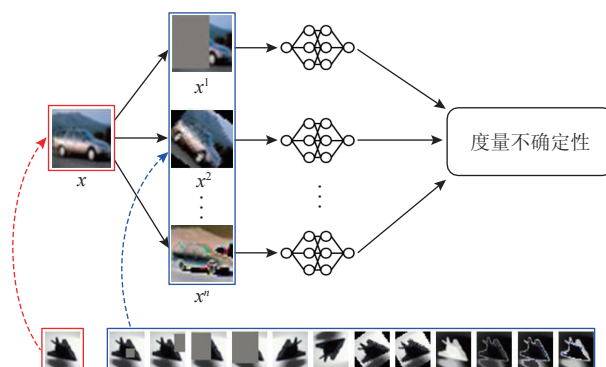


Fig. 3 Illustration of RM^[30] method

图3 RM^[30]方法示意图

应的512维的特征只需要占用2 KB,约为图片格式的1%.存储特征的方式可以极大地提高保存样本的数量,但随着模型不断地更新,会存在特征过时的问题.为了解决这个问题,文献^[31]通过保存模型副本 f_{θ_t} ,利用当前任务的数据 U_{t+1} 在 f_{θ_t} 和当前模型 $f_{\theta_{t+1}}$ 提取出的特征 V^t, V^{t+1} 来训练一个小型网络 g 学习 $f_{\theta_t}(x)$ 和 $f_{\theta_{t+1}}(x)$ 之间的映射关系,利用训练好的小型网络 g 对之前保存的特征进行更新.然后将更新后的旧类样本特征与新类样本特征混合,重新训练一个特征分类器.该分类器可以是一个线性全连接层,也可以是支持向量机(support vector machine, SVM).此外,文献^[32]提出通过传统的图像压缩算法,以降低图像分辨率为代价,降低单个样本的存储大小,提高存储空间的利用率.文献^[33]与文献^[31]类似,通过当前任务数据在整个任务中发生的语义漂移(semantic drift)来估计出各个类别原型的语义漂移,可以与保存样本特征类的工作相结合.如图4所示,这里的语

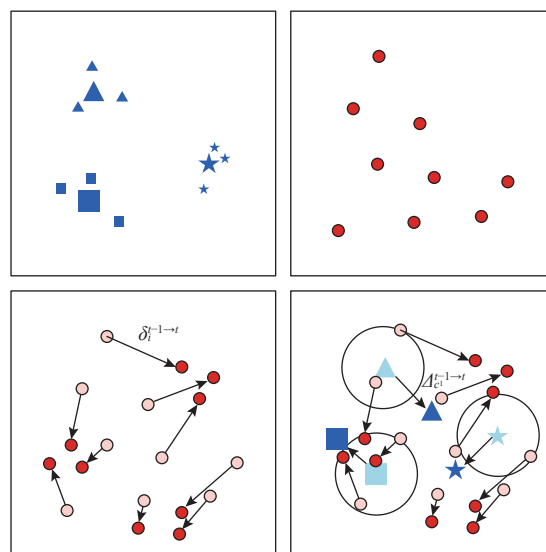


Fig. 4 Illustration of semantic drift^[33]

图4 语义漂移^[33]示意图

义漂移指的是一个样本在同一模型的不同时间点所对应的特征表示是不同的. 因为随着模型在学习过程中不断更新参数, 导致特征提取器发生变化, 从而出现特征过时. 文献 [33] 的特征更新计算方法具体为:

$$\hat{\Delta}_{c^s}^{t-1 \rightarrow t} = \sum_i [y_i \in C^t] w_i \delta_i^{t-1 \rightarrow t} / \sum_i [y_i \in C^t] w_i, \quad (6)$$

$$w_i = e^{-\frac{\|z_i^{t-1} - \mu_{c^s}^{t-1}\|^2}{2\sigma^2}}, \quad (7)$$

其中 $\hat{\Delta}_{c^s}^{t-1 \rightarrow t}$ 表示类 c^s 的原型在任务 t 上发生的语义漂移; δ_i 表示样本 i 在任务 t 上发生的语义漂移; z_i 和 μ_{c^s} 分别表示样本 i 的特征以及类 c^s 的原型; 通过高斯分布来衡量样本 i 与类 c^s 原型的相似度权重 w_i .

此外, 由于存储样本数量远小于当前任务样本数量, 文献 [34] 为解决训练数据中新旧类样本数量极度不平衡这一问题, 在正常的训练过程后增加了一个微调 (fine-tuning) 步骤, 通过从训练数据中划分出一个类别平衡的验证集, 并利用验证集和较小的学习率来微调模型以减轻模型对于新旧类别的偏差. 类似的还有文献 [35] 提出在模型分类层后增加 2 个可学习参数 α 和 β , 通过对在新类上的预测值进行缩放: $q_k = \alpha \times o_k + \beta$, 来矫正模型对于新旧类别的偏差. 如图 5 所示, 该方法通过从训练集中划分出一个新旧类平衡的验证集用于在每个任务训练结束后训练

2 个矫正参数.

文献 [36] 注意到线性分类层中新类的权重范数明显大于旧类, 从而造成模型更偏向于新类. 为此, 提出使用余弦相似性来避免这一问题, 并通过间隔损失使得各类别的特征在特征空间中分离. 文献 [37] 同样注意到新旧类别的范数存在差异, 并且通过对比实验发现知识蒸馏损失有助于模型保持旧类别间的区分度, 但不能缓解模型将旧类别误分类为新类的现象. 如图 6 所示, 通过对齐新旧类别权重的范数, 对模型的偏差进行矫正, 形式为

$$W'_{\text{new}} = \gamma \times W_{\text{new}}, \quad (8)$$

$$\gamma = \frac{\text{Mean}(\text{Norm}_{\text{old}})}{\text{Mean}(\text{Norm}_{\text{new}})}, \quad (9)$$

其中 Norm_{old} 和 Norm_{new} 分别表示旧类和新类线性分类头中对应权重参数的范数. $\text{Mean}()$ 表示取集合元素的平均值. 通过为新类在线性分类头中对应权重参数 W_{new} 乘上一个缩放系数 γ , 使新旧类权重的范数进行对齐.

文献 [38] 试图将每个任务的特征分为任务无关特征 (共享特征) 以及任务特定特征, 并认为任务特定特征更容易被遗忘, 从而导致模型在旧任务上的性能出现下降. 为此借助对抗学习, 采用正交性约束强制共享特征不会出现在任务特定特征空间中. 如图 7 所示, 将特征提取器分为 2 部分: 共享模块和多

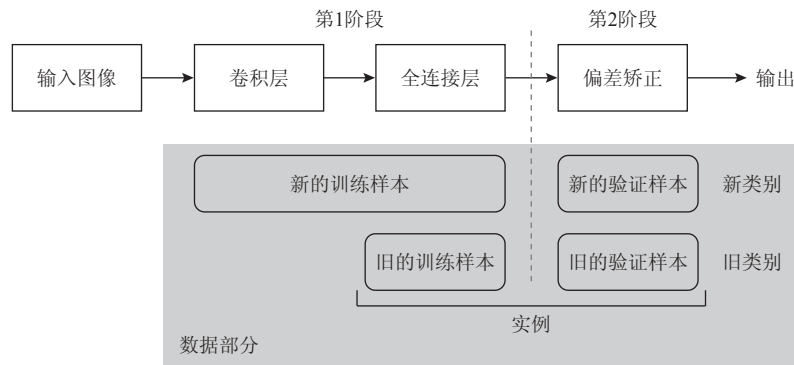


Fig. 5 Illustration of BiC^[35] method

图 5 BiC^[35] 方法示意图

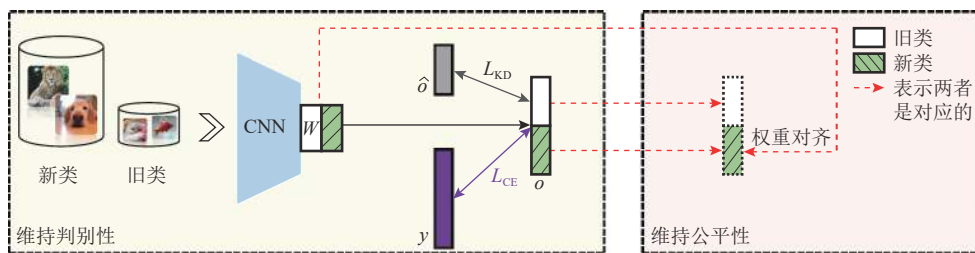
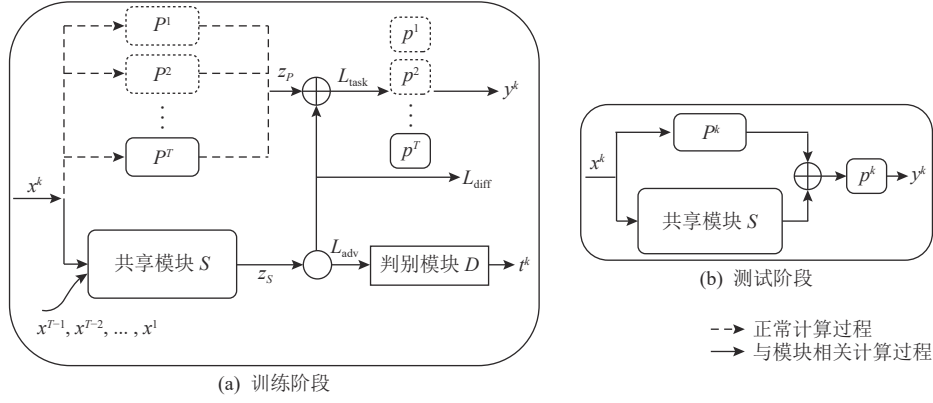


Fig. 6 Illustration of weigh align^[37] method

图 6 权重对齐^[37] 方法示意图

Fig. 7 Illustration of ACL^[38] method图7 ACL^[38]方法示意图

个任务私有模块. 共享模块需要学会从不同任务的样本中提取出任务无关的特征表示. 为此, 通过引入一个负责预测样本特征属于哪个任务的判别模块, 并将其与共享模块进行对抗训练, 形式为:

$$L_{\text{adv}}(D, S, X_k, T_k, M_k) = \min_S \max_D \sum_{k=0}^T \mathbb{1}_{[k=t_k]} \ln D(S(x_k)), \quad (10)$$

其中 D, S 分别表示判别模块与共享模块, X_k, M_k 分别表示当前任务样本与保存样本. 为了使私有模块与共享模块学到的特征尽可能正交, 提出差异性损失:

$$L_{\text{diff}}(S, P, X_k, M_k) = \sum_{k=1}^T \|(S(x_k))^T P_k(x_k)\|_F^2, \quad (11)$$

其中 P_k 表示任务 k 的私有模块.

综上所述, 采样回放方法放宽了无法接触旧样本这一限制, 存储部分旧样本, 并将其直接用于后续的模型训练中. 对于存储这一步骤, 现有方法大多关注 2 点: 存储怎样的样本和怎样提高存储空间利用率. 此外, 采样回放方法也往往会与其他类方法相结合, 在如何高效利用存储样本这一问题上进行探索. 采样回放方法相比较其他方法, 因为在后续任务中能够直接接触旧样本, 性能往往较好. 但这类方法也存在 2 个缺点: 1) 在现实场景的部署中保存样本

可能会涉及隐私问题; 2) 存储样本数量较少, 容易在训练过程中对这一小部分样本过拟合. 此外存储样本数量远小于当前任务样本数量, 使得训练集中存在类别不平衡现象, 这也会对训练效果造成一定影响.

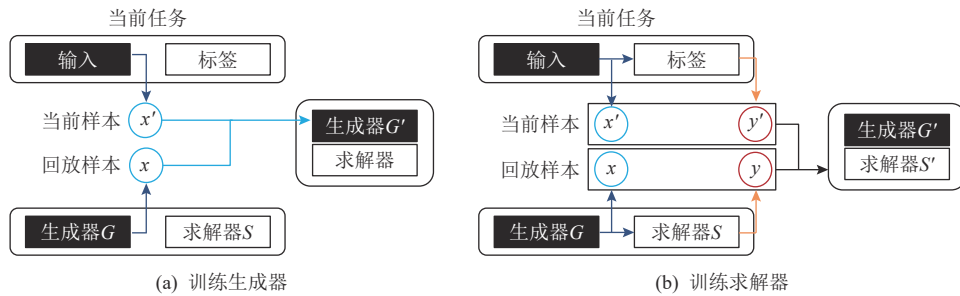
2.1.2 生成回放方法

不同于采样回放方法直接采样保存原始样本, 生成回放方法往往基于生成对抗网络, 通过训练一个能够生成旧类样本的生成网络, 并将其保存下来用于后续任务学习过程中随时生成旧类样本, 变向地解决持续学习中不能接触旧样本的限制.

文献 [39] 提出 DGR 方法, 如图 8 所示, 其中主要包含 2 个模型: 生成器 G 和求解器 S , 生成器 G 负责生成各类旧样本, 求解器 S 负责对各类样本进行分类. 当新任务到来时, 首先利用生成器 G 生成足够多的旧类样本 x , 并将其和当前新类样本 x' 按一定的比例混合. 然后, 按照传统生成对抗网络的训练方式训练一个新的生成器 G' , 使其能够生成目前见过的所有类别的样本. 生成对抗网络的训练目标为:

$$\min_G \max_D V(D, G) = E_{x \sim p_{\text{data}}(x)} [\ln D(x)] + E_{z \sim p(z)} [\ln(1 - D(G(z)))]. \quad (12)$$

当生成器 G' 训练好后, 再借助旧生成器 G 和求解

Fig. 8 Illustration of DGR^[39] model图8 DGR^[39]模型示意图

器 S ,生成旧类样本集 $U_{(x,y)}$,与当前任务数据 $U_{(x',y')}$ 混合在一起训练新的求解器 S' .

文献[40]中的模型包括生成器 G 和判别器 D ,其模型使用AC-GAN框架^[41],其中判别器 D 包含2个作用:辨别真实和生成的图片以及对目前见过的所有类进行分类.此外,为了防止生成器出现旧任务知识的重要参数被重写的情况,还使用了参数隔离和模型拓展技术,为每一层的权重学习一个稀疏的二进制掩码来避免各任务之间的知识产生冲突.另外,为了使模型有足够的容量来学习新任务,在每一个任务结束后通过增加神经元的方式来保证每一层的自由神经元数量不变以容纳新任务的知识.

综上,生成回放方法试图通过训练一个能够生成所有旧类图像的生成模型来避免灾难性遗忘.这类方法的关键在于如何训练出一个符合要求的生成模型.然而,生成模型相对于分类模型更难训练,训练时间也会更长,在持续学习设定下,由于旧类原始样本的缺失导致模型训练难度很大.

2.1.3 伪样本回放方法

不同于生成回放方法需要用额外的生成对抗网络模型来生成旧样本,伪样本生成回放方法只需要借助分类网络就可实现伪样本生成.如图9所示,这

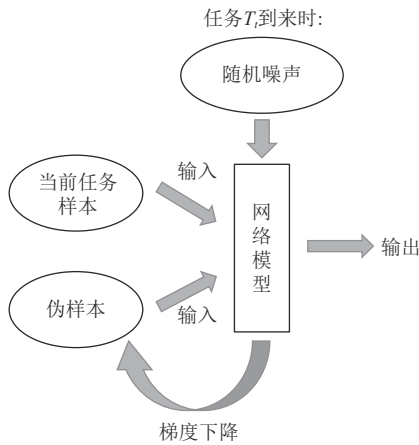


Fig. 9 Illustration of pseudo-sample generation replay methods

图9 伪样本生成回放方法示意图

类方法通常会把输入的随机噪声作为可学习参数,固定模型参数,通过梯度下降的方式优化输入,最终将随机噪声优化为伪样本,其核心在于如何合成出包含丰富旧类信息的伪样本用来传递旧类知识,减轻灾难性遗忘.

文献[42]中指出,文献[43]通过使线性分类层在指定类别上产生高响应的方式来优化输入噪声,但由于缺乏对中间特征表示的约束,导致生成的图像缺少自然图像统计信息而不自然.这样低质量的伪样本难以传递旧类知识.为此,观察到模型在训练过程中其实隐式地编码了训练样本中非常丰富的信息,即模型中的批归一化层(batch normalization layer)中存储的均值和方差,其在本质上表示了过去的训练样本在各层的表示.通过假设这些中间激活值服从高斯分布,就可以对合成伪样本的中间特征表示施加约束,使合成的图像更加接近训练集样本分布,从而能够更好地传递旧类知识.设计的特征分布正则项为:

$$R_{\text{feature}}(\hat{x}) = \sum_l \|\mu_l(\hat{x}) - BN_l(rm)\|_2 + \sum_l \|\sigma_l^2(\hat{x}) - BN_l(rv)\|_2, \quad (13)$$

其中 $\mu_l(\hat{x}), \sigma_l^2(\hat{x})$ 分别表示合成样本 \hat{x} 在第 l 层的特征表示的均值和方差, $BN_l(rm), BN_l(rv)$ 分别表示该 BN 层的均值和方差.除了合成高质量的样本以外,还希望合成样本多样化.为此,还提出一个迭代式合成图像过程,如图10所示.其核心思想为鼓励合成那些会使得教师模型和学生模型产生分歧的图像.使得合成的样本落在学生模型熟悉的分布之外,从而令合成的样本具有多样性.通过不断地迭代图10中的过程,使得学生模型不断拓展熟悉的样本分布,也使得合成的样本能够逐渐覆盖原始数据分布.

文献[44]在合成样本的时候,不同于以往合成样本方法往往指定标签为一个独热向量(one-hot vector).参考文献[45],使用从狄利克雷分布(Dirichlet distribution)中采样出的向量作为合成样本时所使用

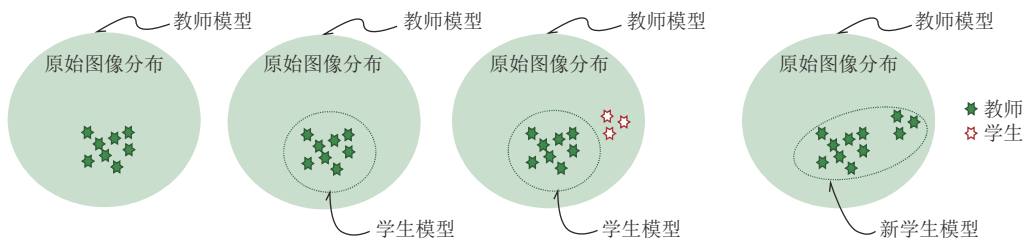


Fig. 10 Illustration of DeepInversion^[42] method

图10 DeepInversion^[42]方法示意图

的标签, 并认为软标签比独热向量包含更多的语义信息, 更有助于合成服从训练样本分布的伪样本. 但随意采样出的向量很可能导致合成的样本并不属于训练样本的分布, 必须对采样出的向量做出一定的限制. 文献[45]通过在训练时保存一个混淆矩阵 CM , 其中 $CM_{i,j}$ 表示有多少第 i 类样本被误分类为第 j 类. 在为第 i 类合成样本时, 将 CM_i 归一化后的向量作为一个参考值 γ_i , 只有当采样出的随机向量 v_i 与 γ_i 的距离小于一定阈值 η 时, 才会利用 v_i 作为标签来合成样本. 使用混淆矩阵的每一行作为参考值的前提假设是: 类别之间的相似度与各类预测值的大小应该保持一致. 即: 如果 $CM_{i,j}$ 较大, 则说明第 i 类与第 j 类相似度较高, 因此模型对于第 i 类样本在第 j 类上的预测值也应该相应较高, 反之亦然.

文献[46]指出合成样本与真实样本之间存在2种差异: 1) 语义差异. 指样本属于旧类样本或新类样本. 2) 分布差异. 指样本属于合成样本或真实样本. 而当使用合成样本和真实样本同时训练模型时, 模型往往容易学会辨别一个样本属于合成样本还是真实样本. 这导致当使用测试集来测试模型时, 模型往往会由于测试样本属于真实样本, 而将新旧类测试样本都预测为真实样本(新类), 从而导致模型性能急剧下降. 为了模型能够更好地关注于当前任务各类的特征, 在训练分类模型时使用局部交叉熵分类损失 $L_{CE}^T = L_{CE}(p(x|y \in T_n), y)$, 让模型不会通过分布差异来区分新旧类样本. 为了最小化旧数据的特征差异, 文献[46]通过分类器中各类别对应权重向量范数大小来衡量特征的重要性, 并基于此提出重要性加权特征蒸馏:

$$L_{KD}^{wfeat} = \|W(\theta_n^{L-1}(x)) - W(\theta_{n-1}^{L-1}(x))\|_2^2, \quad (14)$$

其中 θ_n, θ_{n-1} 分别表示当前模型和上个任务的模型副本. 最后, 为了分开新旧类且不让模型关注样本是真实或合成数据, 通过标准交叉熵损失来微调最后的线性分类层.

综上所述, 伪样本回放方法与生成回放方法唯一的不同点在于合成图像的方式, 伪样本回放方法将随机噪声输入看作可学习参数, 通过梯度下降方式优化输入得到伪样本. 这种做法的好处在于只需要训练分类模型, 一定程度上降低了训练难度. 这类方法的关键在于如何有效设计合成样本时的损失函数, 如文献[42]中指出的一样, 如果对生成样本的中间特征表示不施加约束, 容易导致合成的图像质量较差, 难以传递旧类信息. 因此文献[42, 44, 46]都是

通过利用样本的中间特征表示来设计约束项, 进而保证合成的图像富含旧类信息.

2.2 基于约束的方法

受到理论神经科学模型的启发, 通过对不同模型参数产生不同可塑性水平的约束, 可以保护巩固的知识不被遗忘^[15]. 从计算的角度来看, 这可以通过额外的正则化惩罚项来对神经网络进行建模, 这些正则化项惩罚神经网络映射函数的变化.

这一类方法的核心是设法通过设计模型的参数更新过程来保证模型在学习新任务时不会降低在旧任务上的表现. 根据约束的方式, 该方法可以进一步分为参数约束方法、梯度约束方法和数据约束方法. 参数约束方法通过衡量各个模型参数的重要性, 并限制重要参数的更新来保证模型性能的稳定. 梯度约束方法将模型在任务上的表现转换为关于模型参数的带约束优化问题. 通过求解得到模型参数梯度的可行空间, 并将模型参数梯度投影至可行空间中, 得到模型参数实际更新的梯度. 数据约束方法相较于前2种方法, 其约束形式更加高层, 通过让当前模型模仿旧模型副本来防止模型在学习过程中过度遗忘旧知识.

2.2.1 参数约束方法

参数约束方法的主要思想是限制对旧任务性能影响较大的参数的更新, 防止重要参数的更新导致模型对旧类知识的灾难性遗忘. 如图11所示, 通常假设参数之间是独立的, 并且由于过参数化现象存在, 使得有可能在任务1的最佳参数 θ_1^* 的附近找一个对于任务2而言的最佳参数 θ_2^* , 这样使得模型参数处于2个任务的低损失区域的相交区域. 换言之, 模型在学习任务2的同时, 保持了在任务1上的性能. 关键在于如何准确估计出各个参数对于模型性能的重要性程度.

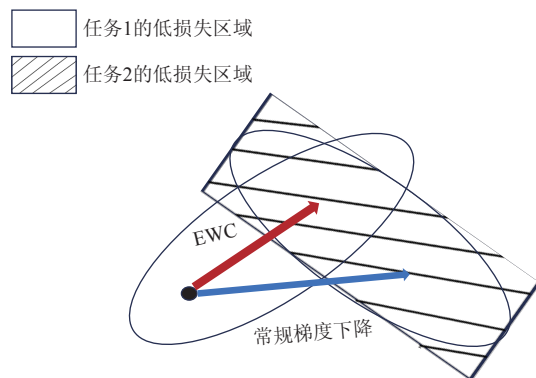


Fig. 11 Illustration of EWC^[47] method

图 11 EWC^[47] 方法示意图

Kirkpatrick 等人^[47]通过贝叶斯法则将后验概率表示为

$$\ln p(\theta|U) = \ln p(U_2|\theta) + \ln p(\theta|U_1) - \ln p(U_2), \quad (15)$$

其中 θ 表示模型参数, U 表示完整数据集, U_1, U_2 为 U 的划分, $p(\theta|U)$ 表示在给定数据集 U 上训练得到的参数 θ 的分布. 由于在持续学习场景下无法获得真实的后验概率, 参考文献[48]的做法, 将后验概率近似为高斯分布. 文献[47]将损失函数对于参数的二阶导数作为参数的重要性, 并使用 Fisher 信息矩阵的对角元素 F_i 进行近似, 其最终的优化目标为

$$L(\theta) = L_2(\theta) + \sum_i \frac{\lambda}{2} F_i (\theta_i - \theta_{i,i}^*)^2. \quad (16)$$

这一优化目标代表一边最大化任务2的似然概率 $p(U_2|\theta)$, 一边保持任务1的后验概率 $p(\theta|U_1)$ 的最大化等价于最大化参数在 U 上的后验概率. Lee 等人^[49]同样假设参数的后验分布服从高斯分布, 并且为了使得假设更加合理, 在训练过程中还使用了迁移学习技术使得2个任务的损失函数的搜索空间尽可能地光滑, 如图12所示. 另外, 为了更好地利用2个单独任务的后验分布估计出混合后的高斯分布, 还提出了合并策略 mean-IMM, 其中 mean-IMM 的目标是最小化真实后验分布 $q_{1,2}$ 与2个任务的混合后验分布 $(1-\alpha)q_1 + \alpha q_2$ 的KL散度:

$$\mu_{1,2}^*, \Sigma_{1,2}^* = \arg \min_{\mu_{1,2}, \Sigma_{1,2}} L_{KL}(q_{1,2} || (1-\alpha)q_1 + \alpha q_2). \quad (17)$$

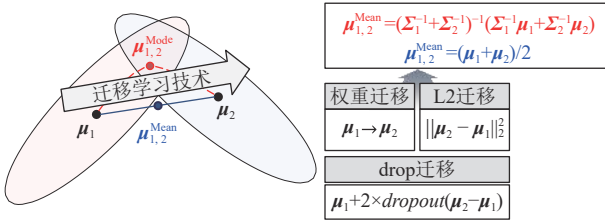


Fig. 12 Illustration of IMM^[49] method

图12 IMM^[49]方法示意图

Aljundi 等人^[50]并不是通过贝叶斯估计参数的后验分布来估计参数的重要性, 而是直接利用模型输出对参数的敏感程度作为重要性评估标准. 对于一个已经学过的任务, 将学到的模型看作一个从训练数据 X_1 到标签 Y_1 的映射 F , 其目的是尽可能地保持映射 F 不变, 即通过公式进行约束:

$$F(x_k; \theta + \delta) - F(x_k; \theta) \approx \sum_{ij} g_{ij}(x_k) \delta_{ij}, \quad (18)$$

其中 g 对应模型输出 F 对于参数 θ 的梯度, δ 是加在参数 θ 上的一个扰动. 基于此, 可以定义参数 θ_{ij} 对于模

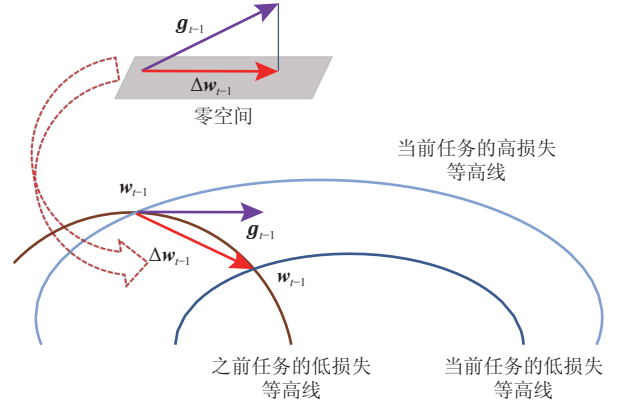


Fig. 13 Illustration of Adam-NSCL^[51] method

图13 Adam-NSCL^[51]方法示意图

型的重要程度为 $\Omega_{ij} = \frac{1}{N} \sum_{k=1}^N \|g_{ij}(x_k)\|$. 当模型输出为多维时, 为了简化计算, 将 F 定义为输出向量的 L_2 范数. 这种做法不仅可以在线计算, 而且借助无标签数据即可完成, 对于实际场景部署更加友好.

综上所述, 参数约束方法通过限制重要参数的更新来保留旧知识. 其关键在于如何估计出参数的重要性. 现有的方法^[47, 49]往往从贝叶斯概率的角度将训练建模为一个寻找参数后验分布的过程, 通过把参数在全局数据上的后验分布转化为在局部数据上的概率分布进行求解. 转化求解目标时往往需要将损失函数泰勒展开, 然后将一阶导数近似为0, 但在任务差异较大时这种近似就会带来较大的误差. 并且该方法通常会忽略参数之间的相关性从而可以使用对角元素来减轻计算量^[47].

2.2.2 梯度约束方法

不同于参数约束方法直接通过计算得到对每个参数的限制, 梯度约束方法通常需要保存部分旧任务的信息. 如图13所示, 在学习新任务时, 通过直接对参数梯度进行矫正, 来保证模型的参数更新不会导致在旧任务上的性能下降.

David 等人^[52]通过为每个旧任务 T_k 保存部分样本 M_k , 并要求模型在学习新任务时的每一次梯度更新不会导致模型在任何一个 M_k 上的损失大于上一个任务结束时在 M_k 上的损失, 即对所有旧任务 T_k 满足 $Loss(f_\theta, M_k) \leq Loss(f_{\theta}^{t-1}, M_k)$. 若当前梯度 g 无法满足约束条件, 则需将当前梯度 g 投影为 \tilde{g} , 而梯度 \tilde{g} 需满足与 g 尽可能地接近从而能够学习新知识. 此外, 还需要保证梯度 \tilde{g} 与在每个任务的数据集上的梯度的内积大于0来保证模型在旧任务上的性能不下降. 最后将梯度 \tilde{g} 的求解变为一个有不等式约束的2次规划问题, 形式为:

$$\min_{\mathbf{g}} \frac{1}{2} \|\mathbf{g} - \tilde{\mathbf{g}}\|_2^2, \quad (19)$$

$$\text{s.t. } \tilde{\mathbf{g}}^T \mathbf{g}_k \geq 0 \text{ for all } k < t. \quad (20)$$

随后, 2019 年, Chaudhry 等人^[53]对文献[52]的方法进一步优化提出 A-GEM 方法, 该方法将文献[52]方法中的约束条件由在所有旧任务上的损失均不上升, 变为在所有旧任务上的平均损失不上升. 这样的优化对算法的效率带来极大的提升: 1) 将约束条件从 $t-1$ 个降低为 1 个; 2) 由于约束条件只有 1 个, 不需要解 2 次规划问题; 3) 利用随机性从 M 中随机采样出的一个批次的样本来估计模型在所有旧任务上的性能, 形式为:

$$\min_{\mathbf{g}} \frac{1}{2} \|\mathbf{g} - \tilde{\mathbf{g}}\|_2^2, \quad (21)$$

$$\text{s.t. } \tilde{\mathbf{g}}^T \mathbf{g}_{\text{ref}} \geq 0,$$

其中 \mathbf{g}_{ref} 表示模型在旧任务样本上的梯度. 与文献[53]类似, 文献[54]提出把遗忘形式化表达成旧任务损失的增量, 并且采用递归最小损失(recursive least loss, RLL)来评估过往所有旧任务的损失和, 通过引入一个对称正定的投影矩阵 \mathbf{P} 来最小化遗忘的期望.

Saha 等人^[55]提出的方法 GPM, 将整个梯度空间分为 2 部分: 核心梯度空间(core gradient space, CGS)和剩余梯度空间(residual gradient space, RGS). 沿着 CGS 梯度下降会对任务相关性和权重表示之间的相关性造成较大的破坏, 而沿着与 CGS 正交的方向, 即 RGS, 则不会造成太大的影响. 为此, 保存了 CGS 的一组基底, 并且通过令参数梯度减去梯度在 CGS 上的投影, 得到与 CGS 正交的梯度用于实际参数更新. 并且在每个任务结束后, 需要添加新任务对应的 CGS 的基底到内存中, 用来约束后续任务的训练. 与 GPM 方法利用梯度的正交来防止对过去任务知识的干扰这一做法相似的还有文献[51], GPM 方法为每一层保存一个特征协方差矩阵, 并将每一层更新的参数梯度投影到该层特征协方差矩阵的零空间中, 来保证参数的更新不会对模型在旧任务上的性能造成太大影响.

综上所述, 梯度约束方法通常需要经过推导后解带约束的优化问题^[52-53, 55]得到模型参数梯度的可行空间, 通过将梯度投影到可行空间中, 得到最终更新用的梯度. 求解带约束的优化问题由于计算量大, 需要消耗大量时间, 因此文献[52-53]提出的方法往往应用于在线学习场景. 此外, 随着任务数量的增多, 模型参数梯度的可行空间会逐渐变小^[55], 学习新任务会逐渐变得困难.

2.2.3 数据约束方法

如图 14 所示, 数据约束方法往往会保存上一个任务结束时的模型副本, 并在学习新任务的过程中使用知识蒸馏技术, 通常会使当前模型和副本在某个数据分布上的预测尽可能保持一致, 通过模仿副本的方式, 让模型在学习新知识的同时尽可能地保留旧知识.

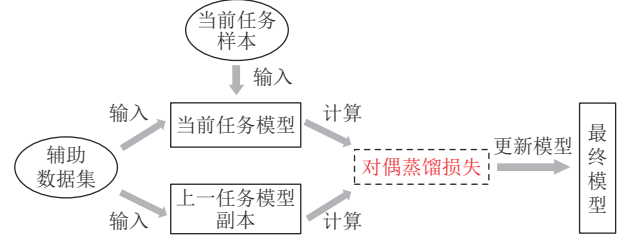


Fig. 14 Illustration of DMC^[56] method

图 14 DMC^[56]方法示意图

Li 等人^[57]提出的 LwF 方法如图 15 所示, 通过约束当前模型与旧模型对于新任务数据输出的一致性来保证模型尽可能地保持旧知识, 具体做法为在训练的目标函数中增加一个损失项 L_{old} .

$$L_{\text{old}}(y_o, \hat{y}_o) = - \sum_{i=1}^I y_o^{(i)} \ln \hat{y}_o^{(i)}, \quad (22)$$

其中 y_o, \hat{y}_o 分别表示旧模型和当前模型对样本的输出. y_o', \hat{y}_o' 分别表示对模型输出进行处理后的分布:

$$y_o'^{(i)} = (y_o^{(i)})^{\frac{1}{T}} / \sum_j (y_o^{(j)})^{\frac{1}{T}}. \quad (23)$$

通过在训练过程中使 L_{old} 最小化, 让当前模型对于样本的输出与旧模型对于样本的输出尽可能一致, 从而保留旧任务知识.

部分工作^[31]使用特征蒸馏, 即基于新旧模型的特征提取器得到数据的新旧特征表示, 其损失项为

$$L_{\text{FD}}(x) = 1 - \cos(h'_\theta(x), h_{\theta'}^{-1}(x)), \quad (24)$$

其中 $h'_\theta(x)$ 表示经过第 t 个任务训练后的模型的特征提取器对样本 x 的特征表示, $\cos(\cdot)$ 表示计算 2 个向量的余弦相似性.

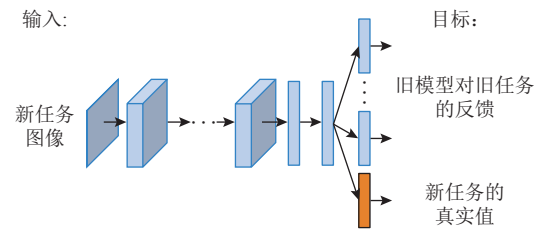


Fig. 15 Illustration of LwF^[57] method

图 15 LwF^[57]方法示意图

Zhang 等人^[56]提出的方法 DMC, 通过对偶蒸馏损失 (dual distillation loss) 作为训练目标, 借助一个无标注的辅助数据集将具有旧知识的模型副本和只在新任务数据上训练好的模型整合起来, 得到最终模型. DMC 的训练过程消除了原本由于信息不对称 (模型只能接触新任务数据, 无法接触旧数据) 而产生的固有偏差, 提出的对偶蒸馏损失使得最终的学生模型能够同时向 2 个教师模型学习, 将新旧任务知识整合到一个模型中. 在整合过程中使用无标注的辅助数据的好处在于可以从辅助数据中挖掘出丰富的可转移表示, 同时也避免使用新数据让模型产生偏差.

综上所述, 数据约束方法不像梯度约束方法和参数约束方法那样会直接通过参数或者参数梯度来设计约束, 而是通过让模型在学习新任务的过程中模仿模型副本的表现来对训练进行约束, 可以理解为一种更高层次的约束. 但这种做法往往潜在地要求任务与任务之间的数据分布不能差距太大, 文献^[56]要求无标注的辅助数据集具有通用和多样化的特点, 并与训练集有一定的相关性. 这一要求在数据比较隐私或者稀少的情况下难以得到满足.

2.3 基于结构的方法

面对持续到来的任务序列, 基于结构的方法考虑为不同任务分配不同的网络结构、模型参数等, 防止模型在学习新任务时覆盖在旧任务上学到的知识, 保证模型的稳定性, 彻底避免灾难性遗忘. 根据具体

实现的角度不同, 这类持续学习方法可以进一步细分为参数隔离方法和模型拓展方法. 参数隔离方法是一类静态的方法, 通常会固定模型容量, 为每个任务分配不同的模型参数. 而模型拓展方法更加动态与灵活, 没有限定模型的大小, 考虑为新任务开辟新的网络子结构, 并且在学习新任务的时候固定已经学好的网络部分. 一般来说, 基于结构的方法训练的模型都拥有多个分类头, 在推理阶段, 模型需要针对不同任务来激活对应的参数路径或子网络. 这类方法的优点在于能完全解决灾难性遗忘问题, 缺点也很明显, 不管是参数隔离方法还是模型拓展方法, 在无穷任务序列的理想情况下, 我们不可能无限制地给每个任务分配新的存储空间.

2.3.1 参数隔离方法

为了解决灾难性遗忘而不增加额外的网络容量, Mallya 等人^[58]提出了一种迭代剪枝的模型 PackNet. 如图 16 所示, PackNet 的核心操作是剪枝, 在对上一个任务进行学习后, 模型通过剪枝来为后续任务尽量腾出空间. 具体地, 在当前任务上训练之后, 模型会对权重参数进行排序, 根据权重的范数大小来评估其重要性, 并剪掉重要性较小的那些权重, 最后对剪枝后的网络做少轮次的再训练以保证其在当前任务上的性能. 后续任务的训练允许利用之前学到的参数, 但不能做任何修改, 只能修改被剪掉的子网络的权重. 在测试阶段, 模型针对不同任务使用对应的掩码来激活各自的权重参数来完成推理任务.

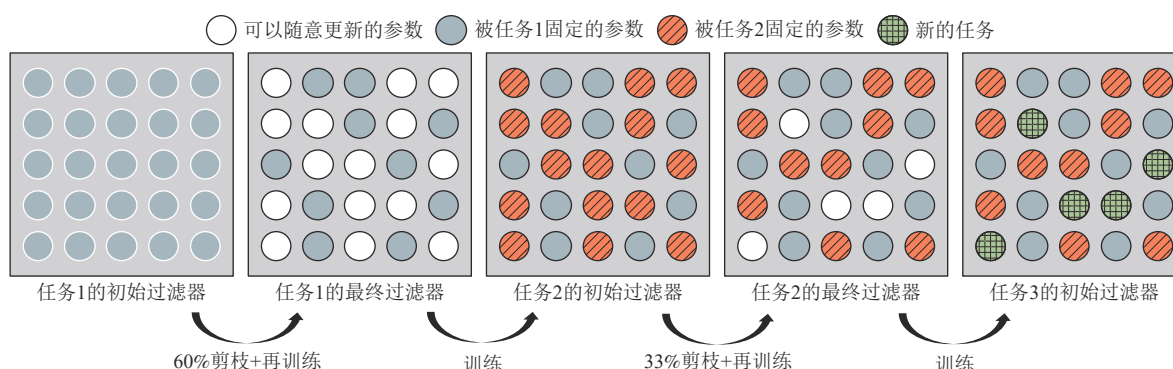


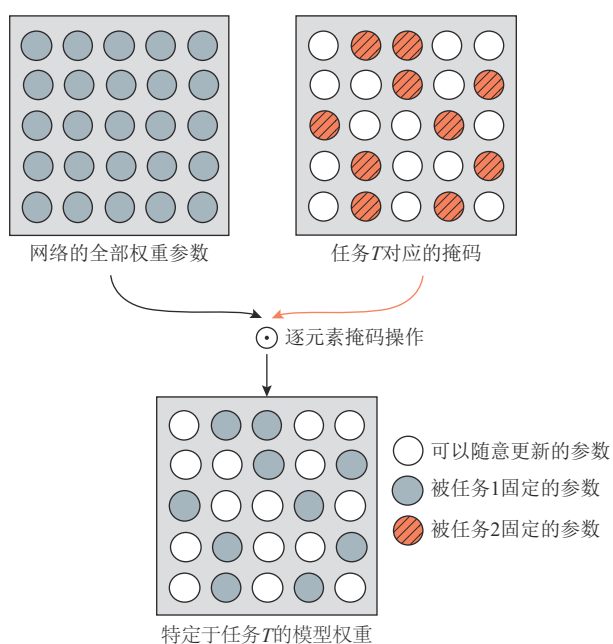
Fig. 16 Illustration of PackNet^[58] method

图 16 PackNet^[58] 方法示意图

从 PNN 模型^[59]出发, DeepMind 的 Fernando 等人^[60]提出了 PathNet. 不同于 PNN 逐步拓展模型的做法, PathNet 方法预先构建了 L 层模块化的神经网络, 每层有 M 个模块, 每个模块本身就是一个网络, 如卷积神经网络. 在学完当前任务后, 模型会冻结对应的模块, 利用剩余模块来学习后续任务. 最终, PathNet 会

为每个任务组织一条模块激活的路径, 当某个任务到来时, 模型将通过对应的路径进行推理决策.

为了避免 PackNet 中频繁的剪枝操作, Mallya 等人^[58]考虑是否必须修改网络参数才能学到适用于新任务的知识, 从这个角度上提出了 Piggyback^[61]方法, 如图 17 所示. 在训练时, Piggyback 固定骨干网络,

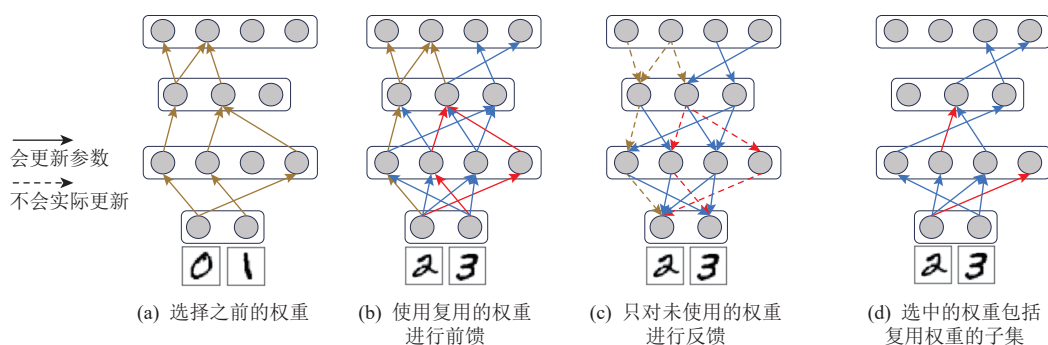
Fig. 17 Illustration of Piggyback^[61] method图 17 Piggyback^[61]方法示意图

并为每个任务端到端地学习一个实数掩码以激活部分权重参数, 解决对应的任务. 在测试阶段, 通过阈值函数, 实数掩码将会被转变为 2 值掩码. 显然, 对于 Piggyback 方法来说, 骨干网络的初始化是非常重要的,

文献 [10] 使用的是在 ImageNet 数据集上预训练的神经网络, 在当时取得了很好的模型性能.

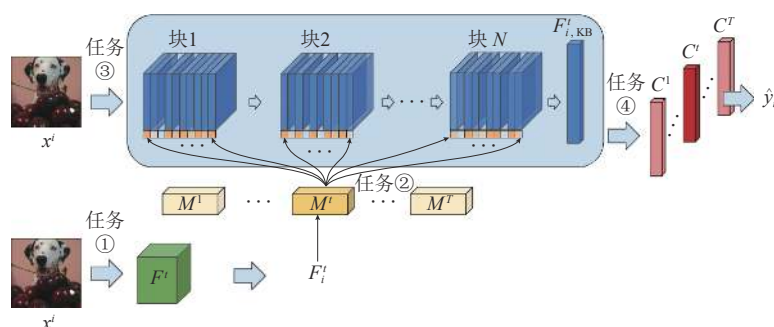
Kang 等人^[62] 将近几年引起广泛关注的彩票假设 (lottery ticket hypothesis, LTH)^[63] 与持续学习场景结合起来, 提出了基于参数隔离的 WSN 方法. 由于彩票假设在经验上证明了在一个稠密网络中存在的稀疏子网络能够保留原本模型的性能, 因此 WSN 方法考虑依次学习每个任务并选择对应的最优子网络. 不同于 Piggyback^[61] 为固定大小的网络学习最优的二进制掩码和 PackNet^[58] 强制要求模型复用之前任务学到的全部特征和权重参数, 在序列任务学习过程中引入了知识迁移的偏差, WSN 方法在稠密网络内部选择性地复用和动态拓展子网络.

如图 18 所示, WSN 单独为每个任务学习子网络的模型权重和特定于任务的二进制掩码, 同时尝试复用之前任务子网络的权重. 然后, 为了避免 WSN 方法所需内存随任务数目增加而显著增大的问题, 使用了霍夫曼编码对学到的二进制掩码进行压缩, 使网络容量随任务数量呈亚线性增长. 可以发现, 因为 WSN 为每个任务学习各自的子网络, 所以该方法不会受到灾难性遗忘的影响, 具有良好的抗遗忘特性.

Fig. 18 Illustration of WSN^[62] method图 18 WSN^[62]方法示意图

结合元学习技术, Hurtado 等人^[64] 提出了 MARK 方法. MARK 依赖于任务间存在相似信息的假设, 在学习一个新任务时促进已有权重的复用, 而非直接重写权重. 如图 19 所示, MARK 在不同任务之间维护了一组共享权重 KB (knowledge base), KB 由 N 个的卷积块组成, 负责在模型面临新任务时积累相关知识. 当新任务 t 的数据到来后, 模型首先利用其特征向量学习一个任务相关的掩码函数 M^t 来查询 KB, 从中选择相关权重以解决当前任务, 再使用元学习算法增量地更新 KB, 累积不同任务的相关知识, 进一

步提高权重复用性. 在测试时, 根据不同的任务 t , MARK 使用对应的掩码函数 M^t 来为测试样本 x_i 生成掩码 M_i^t , 借此选择 KB 中合适的网络权重参数, 之后提取用于模型预测的特征向量 $F_{i,KB}^t$, 交给任务相关的分类器 C^t 完成分类. 需要注意的是, 如果任务之间的相似性假设不显著甚至不成立, 那么 MARK 的模型性能就会很差. 此外, 因为 MARK 需要在第 1 个任务上端到端训练以初始化共享权重, 所以其对第 1 个任务的质量和结构很敏感, 即任务序列不能是完全随机的.

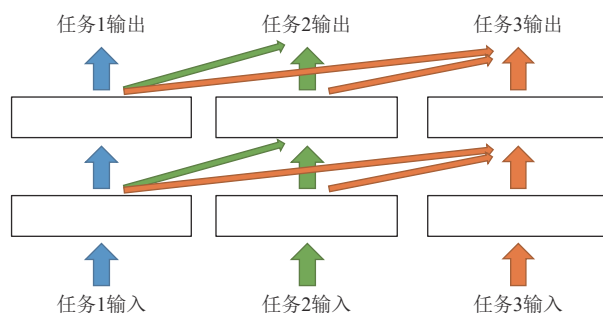
Fig. 19 Illustration of MARK^[64] method图 19 MARK^[64] 方法示意图

文献 [65] 提出了基于视觉 transformer (vision transformer, ViT) 的持续学习方法 MEAT. 如图 20 所示, 在训练过程中, MEAT 为每个任务学习一个注意力掩码 (attention mask), 用来分配给自注意力机制 (self-attention mechanism) 和部分模型参数, 完成持续学习任务. MEAT 充分利用了 ViT 的架构特性, 凭借更小的开销取得了更高的性能.

综上所述可以发现, 参数隔离方法几乎都是基于掩码的 (mask-based), 为了标识每个任务对应的权重参数等, 这些方法自然而然地使用掩码函数来实现这一目标. 参数隔离方法的主要开销是为每个任务存储对应的稀疏掩码, 相比于模型拓展方法为每个任务新建网络结构的做法, 其代价是比较小的.

2.3.2 模型拓展方法

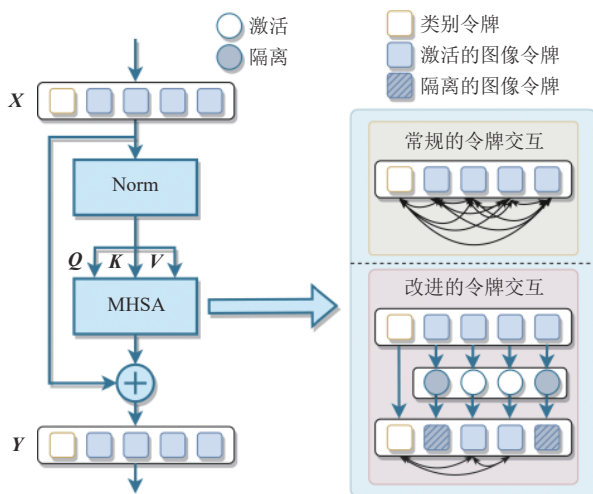
作为最早的模型拓展方法之一, DeepMind 的 Rusu 等人^[59]提出的 PNN 方法则非常简单直接. 为了保证后续任务的训练不会影响之前已经学到的知识, PNN 干脆为每个任务创建整个新的网络, 保证了模型的学习能力. 如图 21 所示, 最初 PNN 只是一个单

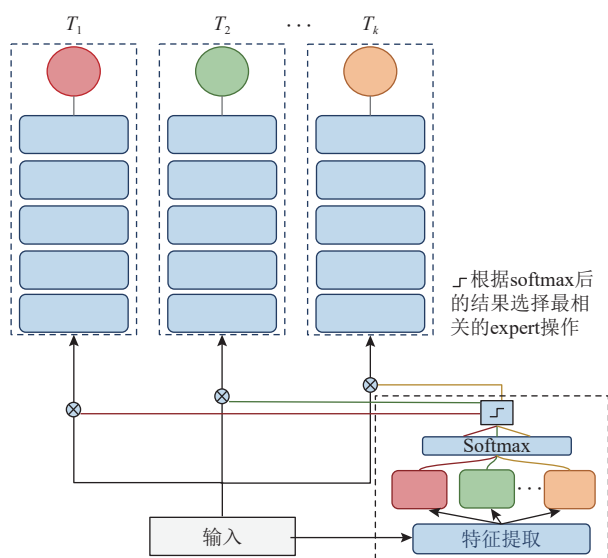
Fig. 21 Illustration of PNN^[59] method图 21 PNN^[59] 方法示意图

列神经网络, 当第 2 个任务到来时, 模型会冻结之前的参数, 重新增加一列网络并随机初始化. 在学习新任务时, 由于侧向连接的存在, 当前任务对应的网络会同时接收自己和之前各列网络的底层输入. PNN 野蛮生长般的模型拓展方式带来了鲜明的优缺点, 虽然它能最大限度地避免灾难性遗忘问题, 但巨大的存储开销也是让人难以接受的.

与 PNN 方法^[59]类似, Aljundi 等人^[66]提出的 Expert Gate 模型也会为每个任务单独创建各自的 Expert 网络. 在训练阶段, 模型中的 Gate 模块计算当前任务和之前任务的相似性, 利用最相近任务对应的参数来初始化当前任务的网络, 能够一定程度上实现旧知识的复用. Expert Gate 虽然可以取得更好的性能, 但存储开销大的缺点依然存在. Expert Gate 的模型如图 22 所示.

不同于 PNN^[59]和 Expert Gate^[66]简单地每个任务都分配一个模型, Yoon 等人^[67]提出的 DEN 方法认为何时拓展网络、要添加多少神经元是需要谨慎考虑的, 如图 23 所示. 如果旧的网络能够充分解释新任务, 那么旧网络就不需要拓展其大小. 另一方面如果新任务与现有的神经元非常不匹配则需要增加大量的神经元, 即需要对模型动态地添加所需数量的神经元.

Fig. 20 Illustration of MEAT^[65] method图 20 MEAT^[65] 方法示意图

Fig. 22 Illustration of Expert Gate^[66] method图 22 Expert Gate^[66] 方法示意图

DEN 的具体做法为: 当新任务到来时, 首先固定模型并在训练过程中借助 L_1 正则化使新增加的分类头尽可能地稀疏. 当得到一个稀疏的分类头后就可以通过广度优先搜索确定网络中哪些神经元会影响到新任务的输出, 将这一部分神经元重新微调并保持其他与新任务无关的神经元不变. 因此, 可将这一步骤的训练形式化为

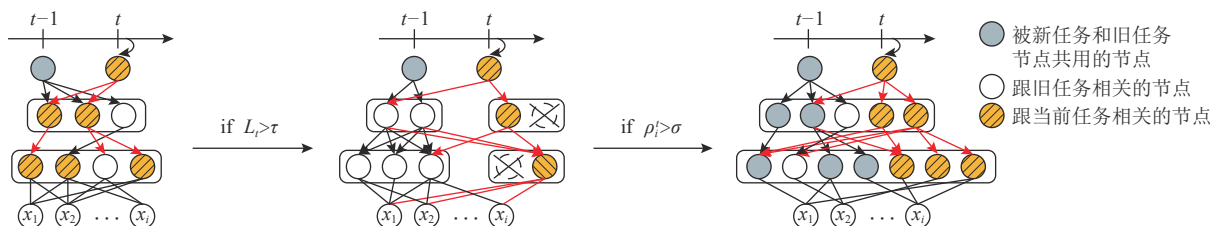
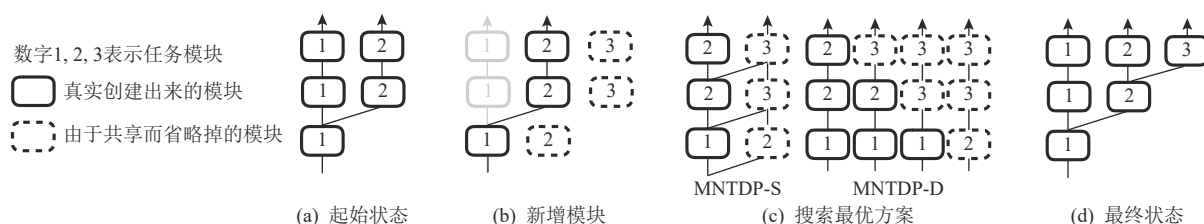
$$\min_{W'_S} L(W'_S; W_S^{-1}, D_t) + \mu \|W'_S\|_2, \quad (25)$$

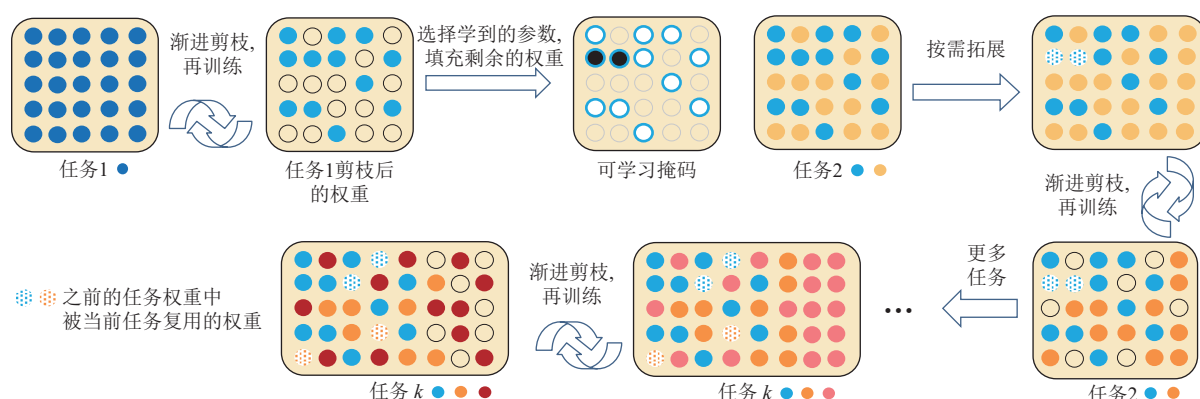
其中 W'_S , W_S^{-1} 代表是否影响新任务输出的子网络. 如果旧知识能够充分地解决新任务, 则此时模型不需要再进行拓展, 反之当模型在微调之后不足以解决

新任务, 则接下来要引入新的神经元来学习新任务所需的特征. 首先为每一层引入固定数量的神经元, 为了避免整个网络重新训练所带来的计算开销, 此时只训练新增的神经元, 并通过施加惩罚项来促进参数的稀疏化, 且在训练后移除无用的神经元来保证模型的增长不会太快.

以往的工作主要衡量的是模型在学习过程中的遗忘情况, Veniat 等人^[68] 认为持续学习方法不仅要避免遗忘, 而且还应兼顾内存效率、计算效率、可拓展性、迁移性等特点, 由此他们提出了 CTrL 基准测试来更全面地评估持续学习方法. 如图 24 所示, 他们提出了一个模块化的神经网络 MNTDP, 为每个任务复用和新建对应的网络结构. 类似于其他的模型拓展方法, MNTDP 模型会在新任务到来时尝试复用已有的模块, 并按需实例化新模块以学习当前任务的知识. 为了保证模型能有效地选择复用模块, MNTDP 设置了指空间上的任务驱动先验(task-driven prior)来组合各个网络模块, 实现长任务序列学习的目标. 通过特殊的结构设计, 如果任务间具有相似性, MNTDP 模型就可以随着任务数量的增加呈次线性扩展, 一定程度上缓解了模型拓展方法普遍存在的存储需求大的问题.

文献 [69] 结合 PackNet^[58] 和 PNN^[59] 的特点, 提出了一种新的基于结构的持续学习方法 CPG(compacting, picking, and growing). 如图 25 所示, CPG 方法主要由 3 个步骤组成: 模型压缩、关键权重选择、渐进网络拓展. 在新任务到来后, CPG 首先通过模型剪枝选出对于旧任务的关键权重并固定, 使用释放出的权重

Fig. 23 Illustration of DEN^[67] method图 23 DEN^[67] 方法示意图Fig. 24 Illustration of CTrL^[68] method图 24 CTrL^[68] 方法示意图

Fig. 25 Illustration of CPG^[69] method图 25 CPG^[69] 方法示意图

参数来学习新任务, 然后评估当前模型, 如果不满足准确率等性能指标就进行模型拓展, 增加新的网络节点继续学习. CPG 继承了参数隔离方法和模型拓展方法的优势, 既能复用网络参数, 又能适时扩展模型容量, 有效地避免了灾难性遗忘.

总的来说, 模型拓展这种渐进式的持续学习方法优缺点都很明显. 优点是可以尽可能地避免灾难性遗忘, 缺点也非常突出, 比如 PNN 要求每个任务的底层特征有一定相似性、后续的网络不能利用之前获得的高层特征等, 而且最大的问题还在于存储开销巨大, 随着训练的进行, 网络模型的参数量会成倍增加.

3 国内外研究现状分析

3.1 基于回放的研究现状

针对持续学习中造成灾难性遗忘的直接原因, 即无法接触旧类样本, 基于采样回放的方法因为可以存储部分样本, 为模型保存知识带来极大便利, 发展出许多的方法. 对于旧样本的采样策略, 国外有牛津大学提出的 iCaRL^[29] 以及 NAVER 提出的 RM^[30] 都针对保存哪些样本进行研究. 随着对灾难性遗忘的深入研究, 人们逐渐注意到基于采样回放的方法容易使得模型存在偏差. 对此, 国外有马拉加大学提出的 EEIL^[34], 国内有东北大学提出的 BiC^[35]、中国科学技术大学提出的 UCIR^[36] 和清华大学提出的 WA^[37], 针对模型偏差的不同表现形式对模型进行矫正. 近年来, 人们对隐私问题愈加关注, 也促使持续学习衍生出基于生成回放的方法和基于伪样本回放的方法. 在不直接保存真实旧数据情况下, 通过生成包含旧类信息的数据来缓解灾难性遗忘. 例如, 麻省理工大学提出的 DGR^[39]、柏林洪堡大学提出的 DGM^[40] 都是

利用 GAN 模型来生成旧样本, 而英伟达提出的 DeepInversion^[42]、佐治亚理工学院提出的 ABD^[46] 等方法都是通过在一定约束条件下将随机噪声变换为包含旧类信息的样本来减轻遗忘.

基于回放的方法是持续学习中非常重要的一类方法, 国内外都有相当多的工作在进行研究. 并且近年来持续学习朝着多领域融合的方向发展, 而基于回放的方法可以保存部分旧类样本这一特点, 也为其他领域方法的迁移带来极大的便利.

3.2 基于约束的研究现状

由于新任务的学习会干扰模型中保存的旧知识, 基于约束的方法针对这一问题, 通过在学习新任务时施加约束来避免对旧知识的过度干扰. DeepMind 提出的 EWC^[47]、首尔大学提出的 IMM^[49] 等都是通过不同的方式来衡量模型参数对于旧任务的重要性, 并通过对参数的改变来施加约束以保证模型对于旧知识的保留.

除了通过施加正则项来约束模型的学习过程, 国内外还有许多工作是直接限制模型参数在学习新任务时的梯度. Facebook 提出的 GEM^[52] 和 A-GEM^[53]、普渡大学提出的 GPM^[55] 和西安交通大学提出的 Adam-NSCL^[51] 都属于这一类工作. 伊利诺伊大学提出的 LwF^[56] 和南加州大学提出的 DMC^[56] 则通过知识蒸馏来限制模型在学习新任务时改变对旧数据的特征表示.

基于约束的方法的核心思想也往往会与其他类型的方法相结合来作为损失项的形式出现. 如牛津大学提出的 iCaRL^[29] 和清华大学提出的 WA^[37] 等方法中均使用了知识蒸馏损失项来帮助模型缓解遗忘.

3.3 基于结构的研究现状

为了防止新旧任务知识之间的干扰, 基于结构的方法通过为不同任务分配不同部分的模型来容纳

各任务的知识,以解决灾难性遗忘问题.参数隔离常见的做法是为每个任务学习一个参数掩码来标识每个任务对应的权重参数.伊利诺伊大学提出的 PackNet^[58]与 Piggyback^[61], DeepMind 提出的 PathNet^[60]以及浙江大学提出的 MEAT^[65]均属于这类工作.此外,DeepMind 提出的 PNN^[59]、巴黎大学提出的 CTrL^[68]、台湾中央研究院资讯科学研究所提出的 CPG^[69]通过为新任务适当地拓展模型来容纳新知识来避免不同任务知识之间的互相干扰,从而能够实现极小的遗忘.

基于结构的方法由于直接避免知识的互相干扰,在反向迁移精度 BWT 这一性能指标上往往表现非常出色,但如何限制模型拓展的速率以及准确地为每个任务分配掩码是目前国内外研究的重点.

4 发展趋势与展望

综上所述,围绕持续学习的理论和方法研究在国内外快速开展.近年来的工作有朝着多领域交叉的发展趋势,得益于持续学习并没有对任务序列中的单个任务做出过多的限制,研究者也越来越注意到其他领域技术在持续学习中的适用性,如强化学习、迁移学习、元学习、在线学习等.本文从3个方面对持续学习的未来发展进行展望:

1) 持续学习理论研究.目前已有的持续学习工作主要是朝着如何防止模型遗忘这一目标而努力.遗忘可以被看作是模型在学习新知识的过程中参数的更新破坏了原本被编码在参数中的旧知识.事实上,根据联合训练的结果得知,模型本身的容量足够将新旧任务知识都同时编码进一组权重参数中.因此,如何学到一组能够同时兼容新旧知识的模型参数在理论体系和理论支撑上仍然是一个十分重要且具有挑战的问题.在基于回放的方法中,能够保存的只是一小部分旧样本,远小于新任务的样本数,因此如何减轻训练数据中的类别不平衡也是一个亟需解决的问题.在基于约束的方法中,参数更新往往是一个带约束的优化问题,这也导致像文献[52]这类的工作由于解优化问题的速度限制,往往难以应用到真实在线学习场景中.如何快速求解带约束的优化问题也是一个未来亟需解决的问题.

2) 持续学习新的问题设定.现有持续学习工作的问题设定主要集中在任务数据量充足、监督信息完备、任务编号信息可知等完美假设的场景下.但是在真实任务场景中,往往存在数据量不充足、标记信

息缺失、任务编号未知等问题.因此,持续学习未来的发展趋势将会往更贴近实际场景设置的方向发展,例如,小样本类别增量学习(few-shot class-incremental learning)、无监督持续学习(unsupervised continual learning)、在线持续学习(online continual learning)和通用持续学习(general/task-free continual learning).

人类在真实环境中学习新的任务或者类别时往往只需要几个样本,因此小样本类别增量学习^[70-71]主要关注当新任务的数据量特别少时,如何兼顾旧任务的性能的同时快速自适应到新任务上.为解决真实场景中数据标注难的问题,无监督持续学习旨在从一系列无标注的序列任务中学习并抵抗灾难性遗忘的问题,并证明比有监督持续学习有显著弱的灾难性遗忘现象^[72].目前主流的持续学习方法主要采用离线的训练范式,即每个任务可以训练多个轮次,可以反复访问当前任务的数据.但在一些真实应用场景中,即便是当前任务也没办法反复访问,因此在线持续学习^[52, 73-75]假定任务数据都是在线流式输入更具有实际意义.不管是任务增量学习、类别增量学习,还是域增量学习场景,持续学习模型在训练阶段都需要知道任务的编号以及任务之间的边界.但是,在真实应用场景中,我们并不清楚任务边界,因此通用持续学习^[76-78]假设训练阶段任务边界未知的设定更符合实际应用场景.

3) 持续学习新的应用场景.持续学习除了在图像分类任务上得到广泛关注,最近2年也逐渐引入并应用到其他任务中.例如,文献[79]将持续学习应用到目标检测任务中,进而定义新的增量目标检测(incremental object detection)任务.后续一系列工作,如文献[80-83],分别从目标检测设定、小样本场景、资源受限场景、开放世界场景对增量目标检测问题进行研究.文献[84]将持续学习引入生成对抗网络,随后也掀起了终生生成对抗网络(lifelong GAN)的研究热潮^[85].文献[86-89]在图像语义分割问题上引入持续学习场景,构造了新的持续语义分割(continual semantic segmentation)任务.其他引入持续学习的应用场景还包括持续动作识别(continual action recognition)^[90]、持续学习文本分类(continual learning on text classification)^[91]、终生行人重识别(lifelong person re-identification)^[92]、自然语言生成(natural language generation)^[93]等.值得注意的是,在 Nature Communications 2022 上发表的一篇文章首次采用持续学习去解决单颗粒冷冻电子显微术中的粒子筛选(particle picking)操作,用来辅助解决生物大分子的

原子结构的测定工作^[94]. 由此可见, 持续学习未来会在其他应用场景得到更多的应用, 也会发挥更多的效能.

5 结束语

本文主要从持续学习方法在监督学习场景下的应用出发, 重点关注图像分类任务, 主要介绍了基于回放、基于约束和基于结构的三大类持续学习方法在国内外研究进展, 给出了可能的发展趋势与展望. 由于近些年持续学习研究取得了迅猛的发展, 本文并不能涉及目前持续学习的各个方面. 总的来说, 持续学习无论是基础理论方面还是实际应用方面都尚且存在许多问题亟待解决. 但可以说, 持续学习也许是下一个人工智能研究的热点.

作者贡献声明: 李文斌提出整体写作思路, 负责论文撰写和论文修改; 熊亚锐、范祉辰负责文献调研、内容设计、论文撰写和校对; 邓波、曹付元参与论文审阅; 高阳提出指导意见和审核论文.

参 考 文 献

- [1] He Kaiming, Zhang Xiangyu, Ren Shaoqing, et al. Deep residual learning for image recognition [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2016: 770–778
- [2] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition [J]. arXiv preprint, arXiv: 1409.1556, 2015
- [3] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks [C] //Proc of the 26th Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2012: 1106–1114
- [4] Brown T B, Mann B, Ryder N, et al. Language models are few-shot learners [C] //Proc of the 34th Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2020: 1877–1901
- [5] Devlin J, Chang Mingwei, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding [C] //Proc of the Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2019: 4171–4186
- [6] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C] //Proc of the 31st Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2017: 5998–6008
- [7] Wu Haiping, Chen Yuntao, Wang Naiyan, et al. Sequence level semantics aggregation for video object detection [C] //Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 9217–9225
- [8] Deng Jiajun, Pan Yingwei, Yao Ting, et al. Single shot video object detector[J]. *IEEE Transactions on Multimedia*, 2021, 23: 846–858
- [9] Shvets M, Liu Wei, Berg A C. Leveraging long-range temporal relationships between proposals for video object detection [C] //Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 9756–9764
- [10] Russakovsky O, Deng Jia, Su H, et al. Imagenet large scale visual recognition challenge[J]. *International Journal of Computer Vision*, 2015, 115(3): 211–252
- [11] French R M. Catastrophic forgetting in connectionist networks[J]. *Trends in Cognitive Sciences*, 1999, 3(4): 128–135
- [12] Barnett S M, Ceci S J. When and where do we apply what we learn?: A taxonomy for far transfer[J]. *Psychological Bulletin*, 2002, 128(4): 612–637
- [13] Lewkowicz D J. Early experience and multisensory perceptual narrowing[J]. *Developmental Psychobiology*, 2014, 56(2): 292–315
- [14] Murray M M, Lewkowicz D J, Amedi A, et al. Multisensory processes: A balancing act across the lifespan[J]. *Trends in Neurosciences*, 2016, 39(8): 567–579
- [15] Parisi G I, Kemker R, Part J L, et al. Continual lifelong learning with neural networks: A review[J]. *Neural networks*, 2019, 113: 54–71
- [16] McClelland J L, McNaughton B L, O'Reilly R C. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory[J]. *Psychological Review*, 1995, 102(3): 419–457
- [17] O'Reilly R C. The division of labor between the neocortex and hippocampus [M] //Connectionist Models in Cognitive Psychology. Brandon, VT: Psychology Press, 2004: 153–162
- [18] Qin Yulin, McNaughton B L, Skaggs W E, et al. Memory reprocessing in corticocortical and hippocampocortical neuronal ensembles [J]. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 1997, 352(1360): 1525–1533
- [19] Ji Daoyun, Wilson M A. Coordinated memory replay in the visual cortex and hippocampus during sleep[J]. *Nature Neuroscience*, 2007, 10(1): 100–107
- [20] Lange M D, Aljundi R, Masana M, et al. A continual learning survey: Defying forgetting in classification tasks[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022, 44(7): 3366–3385
- [21] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. *Nature*, 2015, 521(7553): 436–444
- [22] Zhuang Fuzhen, Qi Zhiyuan, Duan Keyu, et al. A comprehensive survey on transfer learning[J]. *Proceedings of the IEEE*, 2020, 109(1): 43–76
- [23] Crawshaw M. Multi-task learning with deep neural networks: A survey [J]. arXiv preprint, arXiv: 2009.09796, 2020
- [24] Huisman M, Van R J N, Plaat A. A survey of deep meta-learning[J]. *Artificial Intelligence Review*, 2021, 54(6): 4483–4541
- [25] Hoi S C H, Sahoo D, Lu Jing, et al. Online learning: A comprehensive survey[J]. *Neurocomputing*, 2021, 459: 249–289

- [26] Krizhevsky A. Learning multiple layers of features from tiny images [D]. Toronto: Computer Science Department, University of Toronto, 2009
- [27] Le Ya, Yang Xuan. Tiny imagenet visual recognition challenge [R]. Palo Alto, CA: Stanford University, 2015
- [28] Vinyals O, Blundell C, Lillicrap T, et al. Matching networks for one shot learning [C] //Proc of the 30th Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2016: 3630–3638
- [29] Rebuffi S A, Kolesnikov A, Sperl G, et al. ICaRL: Incremental classifier and representation learning [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 2001–2010
- [30] Bang J, Kim H, Yoo Y J, et al. Rainbow memory: Continual learning with a memory of diverse samples [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 8218–8227
- [31] Iscen A, Zhang J, Lazebnik S, et al. Memory-efficient incremental learning through feature adaptation [C] //Proc of the 16th European Conf on Computer Vision. Berlin: Springer, 2020: 699–715
- [32] Wang Liyuan, Zhang Xingxing, Yang Kuo, et al. Memory replay with data compression for continual learning [J]. arXiv preprint, arXiv: 2202.06592, 2022
- [33] Yu Lu, Twardowski B, Liu Xialei, et al. Semantic drift compensation for class-incremental learning [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 6982–6991
- [34] Castro F M, Marín-Jiménez M J, Guil N, et al. End-to-end incremental learning [C] //Proc of the 15th European Conf on Computer Vision. Berlin: Springer, 2018: 233–248
- [35] Wu Yue, Chen Yinpeng, Wang Lijuan, et al. Large scale incremental learning [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 374–382
- [36] Hou Saihui, Pan Xinyu, Loy C C, et al. Learning a unified classifier incrementally via rebalancing [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 831–839
- [37] Zhao Bowen, Xiao Xi, Gan Guojun, et al. Maintaining discrimination and fairness in class incremental learning [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 13208–13217
- [38] Ebrahimi S, Meier F, Calandra R, et al. Adversarial continual learning [C] //Proc of the 16th European Conf on Computer Vision. Berlin: Springer, 2020: 386–402
- [39] Shin H, Lee J K, Kim J, et al. Continual learning with deep generative replay [C] //Proc of the 31st Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2017: 2990–2999
- [40] Ostapenko O, Puskas M, Klein T, et al. Learning to remember: A synaptic plasticity driven framework for continual learning [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 11321–11329
- [41] Odena A, Olah C, Shlens J. Conditional image synthesis with auxiliary classifier GANs [C] //Proc of the 34th Int Conf on Machine Learning. New York: ACM, 2017: 2642–2651
- [42] Yin Hongxu, Molchanov P, Alvarez J M, et al. Dreaming to distill: Data-free knowledge transfer via deepinversion [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 8715–8724
- [43] Mordvintsev A, Olah C, Tyka M. Inceptionism: Going deeper into neural networks [EB/OL]. (2003-06-18)[2023-03-10]. <https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>
- [44] PourKeshavarzi M, Zhao Guoying, Sabokrou M. Looking back on learned experiences for class/task incremental learning [C/OL] //Proc of the 9th Int Conf on Learning Representations. 2021 [2023-03-10]. <https://openreview.net/pdf?id=RxplU3vmBx>
- [45] Nayak G K, Mopuri K R, Shaj V, et al. Zero-shot knowledge distillation in deep networks [C] //Proc of the 36th Int Conf on Machine Learning. New York: ACM, 2019: 4743–4751
- [46] Smith J, Hsu Y C, Balloch J, et al. Always be dreaming: A new approach for data-free class-incremental learning [C] //Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2021: 9374–9384
- [47] Kirkpatrick J, Pascanu R, Rabinowitz N, et al. Overcoming catastrophic forgetting in neural networks[J]. *Proceedings of the National Academy of Sciences*, 2017, 114(13): 3521–3526
- [48] MacKay D J. A practical Bayesian framework for backpropagation networks[J]. *Neural Computation*, 1992, 4(3): 448–472
- [49] Lee S W, Kim J H, Jun J, et al. Overcoming catastrophic forgetting by incremental moment matching [C] //Proc of the 31st Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2017: 4652–4662
- [50] Aljundi R, Babiloni F, Elhoseiny F, et al. Memory aware synapses: Learning what (not) to forget [C] //Proc of the 15th European Conf on Computer Vision. Berlin: Springer, 2018: 139–154
- [51] Wang Shipeng, Li Xiaorong, Sun Jian, et al. Training networks in null space of feature covariance for continual learning [C] //Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 184–193
- [52] David L P, Ranzato M A. Gradient episodic memory for continual learning[C] //Proc of the 31st Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2017: 6467–6476
- [53] Chaudhry A, Ranzato M A, Rohrbach M, et al. Efficient lifelong learning with A-GEM [J]. arXiv preprint, arXiv: 1812.00420, 2019
- [54] Liu Hao, Liu Huaping. Continual learning with recursive gradient optimization [J]. arXiv preprint, arXiv: 2201.12522, 2022
- [55] Saha G, Garg I, Roy K. Gradient projection memory for continual learning [J]. arXiv preprint, arXiv: 2103.09762, 2021
- [56] Zhang Junting, Zhang Jie, Ghosh S, et al. Class-incremental learning via deep model consolidation [C] //Proc of the 29th IEEE Winter Conf on Applications of Computer Vision. Piscataway, NJ: IEEE, 2020: 1131–1140
- [57] Li Zhizhong, Hoiem D. Learning without forgetting[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 40(12): 2935–2947
- [58] Mallya A, Lazebnik S. PackNet: Adding multiple tasks to a single network by iterative pruning[C] //Proc of the IEEE Conf on

- Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2018: 7765–7773
- [59] Rusu A A, Rabinowitz N C, Desjardins G, et al. Progressive neural networks[J]. arXiv preprint, arXiv: 1606.04671, 2016
- [60] Fernando C, Banarse D, Blundell C. et al. PathNet: Evolution channels gradient descent in super neural networks[J]. arXiv preprint, arXiv: 1701.08734, 2017
- [61] mallya A, Davis D, Lazebnik S. Piggyback: Adapting a single network to multiple tasks by learning to mask weights [C] // Proc of the 15th European Conf on Computer Vision. Berlin: Springer, 2018: 67–82
- [62] Kang H, Mina R, Madjid S. et al. Forget-free continual learning with winning subnet works[C]//Proc of the Int Conf on Machine Learning. New York: ACM, 2022: 10734–10750
- [63] Frankle J, Carbin M. The lottery ticket hypothesis: Finding sparse, trainable neural networks [J]. arXiv preprint, arXiv: 1803.03635, 2019
- [64] Hurtado J, Raymond-Saez A, Soto A. Optimizing reusable knowledge for continual learning via metalearning [C] //Proc of the 35th Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2021: 14150–14162
- [65] Xue Mengqi, Zhang Haoifei, Song Jie. et al. Meta-attention for vit-backed continual learning[C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 150–159
- [66] Aljundi R, Chakravarty P, Tuytelaars T. Expert gate: Lifelong learning with a network of experts [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2017: 3366–3375
- [67] Yoon J, Yang E, Lee J. et al. Lifelong learning with dynamically expandable networks [J]. arXiv preprint, arXiv: 1708.01547, 2018
- [68] Veniat T, Denoyer L, Ranzato M. Efficient continual learning with modular networks and task-driven priors [J]. arXiv preprint, arXiv: 2012.12631, 2021
- [69] Hung S C Y, Tu Chenghao, Wu Chengen. et al. Compacting, picking and growing for unforgetting continual learning [C] // Proc of the 33rd Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2019: 13647–13657
- [70] Tao Xiaoyu, Hong Xiaopeng, Chang Xinyuan, et al. Few-shot class-incremental learning [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 12183–12192
- [71] Zhao Hanbin, Fu Yongjian, Kang Mintong. et al. MgSvF: Multi-grained slow vs. fast framework for few-shot class-incremental learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 44: 1–13
- [72] Madaan D, Yoon J, Li Yuanchun. et al. Representational continuity for unsupervised continual learning [J]. arXiv preprint, arXiv: 2110.06976, 2021
- [73] Pham Q, Liu C, Hoi S. Continual normalization: Rethinking batch normalization for online continual learning [J]. arXiv preprint, arXiv: 2203.16102, 2022
- [74] Caccia L, Aljundi R, Asadi N. et al. New insights on reducing abrupt representation change in online continual learning [J]. arXiv preprint, arXiv: 2104.05025, 2022
- [75] Koh H, Kim D, Ha J W, et al. Online continual learning on class incremental blurry task configuration with anytime inference [J]. arXiv preprint, arXiv: 2110.10031, 2022
- [76] Aljundi R, Kelchtermans K, Tuytelaars T. Task-free continual learning [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2019: 11254–11263
- [77] Buzzega P, Boschini M, Porrello A, et al. Dark experience for general continual learning: A strong, simple baseline [C] // Proc of the 34th Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2020: 15920–15930
- [78] Sun Shengyang, Calandriello D, Hu Huiyi. et al. Information-theoretic online memory selection for continual learning [J]. arXiv preprint, arXiv: 2204.04763, 2022
- [79] Shmelkov K, Schmid C, Alahari K. Incremental learning of object detectors without catastrophic forgetting [C] // Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2017: 3400–3409
- [80] Peng Can, Zhao Kun, Maksoud S. et al. Sid: Incremental learning for anchor-free object detection via selective and inter-related distillation[J]. *Computer Vision and Image Understanding*, 2021, 210: 103229–103236
- [81] Li Pengyang, Li Yanan, Wang Donghui. Class-incremental few-shot object detection[J]. arXiv preprint, arXiv: 2105.07637, 2021
- [82] Li Dawei, Tasci S, Ghosh S, et al. Rilod: Near real-time incremental learning for object detection at the edge [C] // Proc of the ACM/IEEE Symp on Edge Computing. New York: ACM, 2019: 113–126
- [83] Joseph K J, Khan S H, Khan F S, et al. Towards open world object detection [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 5830–5840
- [84] Seff A, Beatson A, Suo D, et al. Continual learning in generative adversarial nets[J]. arXiv preprint, arXiv: 1705.08395, 2017
- [85] Zhai Mengyao, Chen Lei, Tung F, et al. Lifelong GAN: Continual learning for conditional image generation [C] // Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 2759–2768
- [86] Cermelli F, Mancini M, Bul S R, et al. Modeling the background for incremental learning in semantic segmentation [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 9233–9242
- [87] Douillard A, Chen Yifu, Dapogny A, et al. Plop: Learning without forgetting for continual semantic segmentation [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 4040–4050
- [88] Ganea D A, Boom B, Poppe R. Incremental few-shot instance segmentation [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 1185–1194
- [89] Zhang Changbin, Xiao Jiawen, Liu Xialei, et al. Representation compensation networks for continual semantic segmentation [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2022: 7053–7064
- [90] Li Tianjiao, Ke Qihong, Rahmani H, et al. Else-net: Elastic semantic network for continual action recognition from skeleton data [C] // Proc of the IEEE Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2021: 13434–13443
- [91] Huang Yufan, Zhang Yanzhe, Chen Jiaao, et al. Continual learning

for text classification with information disentanglement based regularization [C] // Proc of the 22nd Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2021: 2736–2746

- [92] Pu Nan, Chen Wei, Liu Yu, et al. Lifelong person re-identification via adaptive knowledge accumulation [C] // Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 7901–7910
- [93] Mi Fei, Chen Liangwei, Zhao Mengjie, et al. Continual learning for natural language generation in task-oriented dialog systems [C] // Proc of the Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2020, 3461–3474
- [94] Zhang Xinyu, Zhao Tianfang, Chen Jiansheng, et al. Epicker is an exemplar-based continual learning approach for knowledge accumulation in cryoem particle picking[J]. *Nature Communications*, 2022, 13(1): 1–10



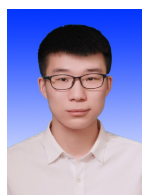
Li Wenbin, born in 1991. PhD, associate researcher. Member of CCF. His main research interests include machine learning, metric learning, few-shot learning, and computer vision.

李文斌, 1991 年生. 博士, 副研究员. CCF 会员. 主要研究方向为机器学习、度量学习、小样本学习、计算机视觉.



Xiong Yakun, born in 2000. Master candidate. Member of CCF. His main research interests include machine learning, continual learning, and computer vision.

熊亚锟, 2000 年生. 硕士研究生. CCF 会员. 主要研究方向为机器学习、持续学习、计算机视觉.



Fan Zhichen, born in 1998. Master candidate. Member of CCF. His main research interests include machine learning, continual learning, and computer vision.

范祉辰, 1998 年生. 硕士研究生. CCF 会员. 主要研究方向为机器学习、持续学习、计算机视觉.



Deng Bo, born in 1973. PhD, researcher. Senior member of CCF. His main research interests include distributed computing and intelligent service system.

邓波, 1973 年生. 博士, 研究员. CCF 高级会员. 主要研究方向为分布式计算、智能服务系统.



Cao Fuyuan, born in 1974. PhD, professor, PhD supervisor. Senior member of CCF. His main research interests include machine learning, data mining, and big data analysis.

曹付元, 1974 年生. 博士, 教授, 博士生导师. CCF 高级会员. 主要研究方向为机器学习、数据挖掘、大数据分析.



Gao Yang, born in 1972. PhD, professor, PhD supervisor. Senior member of CCF. His main research interests include reinforcement learning, multi-agent systems, and big data analysis.

高阳, 1972 年生. 博士, 教授, 博士生导师. CCF 高级会员. 主要研究方向为强化学习、多智能体系统、大数据分析.