基于 CLIP 的类增量学习：自适应表征调整与参数融合

Linlan Huang[1], Xusheng Cao[1], Haori Lu[1], and Xialei Liu[1,2](✉)

[1] VCIP, CS, Nankai University
[2] NKIARI, Shenzhen Futian
{huanglinlan, caoxusheng, luhaori}@mail.nankai.edu.cn
{xialei}@nankai.edu.cn

摘要 类增量学习是一个具有挑战性的问题，其目标是训练一个模型，使其能够对类别不断增加的数据进行分类。随着视觉-语言预训练模型（如 CLIP）的发展，它们现在具有良好的泛化能力，这使得即使在完全冻结参数的情况下，模型仍在类增量学习中表现出色。但是如果通过微调模型以适应下游任务则会导致严重的遗忘问题。大多数使用预训练模型的现有工作都假设模型在获取新知识时对旧类别的遗忘是均匀的。在本文中，我们提出了一种名为自适应表示调整和参数融合（RAPF）的方法。在对新数据进行训练时，我们评估新类别对旧类别的影响，并使用文本特征进行表示。训练后，我们采用分解的参数融合方法来进一步减轻适配器模块微调期间的遗忘。在几个传统基准测试上的实验表明我们的方法取得了最优成果。我们的代码可在 https://github.com/linlany/RAPF 上找到。

关键词: 类增量学习 · 视觉语言模型

1 引言

现实世界在不断变化，这要求模型在保留旧知识的同时学习新知识。如果不进行更新，模型可能会变得过时，并且其性能会随着时间的推移而下降 [10]。隐私和存储限制可能会限制模型对旧数据的访问，导致数据分布严重失衡。当模型更新时，这种失衡会导致模型偏向当前数据并忘记之前学习的知识，这种现象被称为灾难性遗忘 [19]。因此，持续学习的困难在于平衡可塑性和稳定性 [26]，使得模型在不忘记旧知识的情况下学习新知识，并在不同任务中重用和扩展先前经验中的知识。

类增量学习作为持续学习中一个极具挑战性的方向，最近受到了越来越多的关注。它需要从一个不断增加新类别的数据流中学习。目前提出了三种

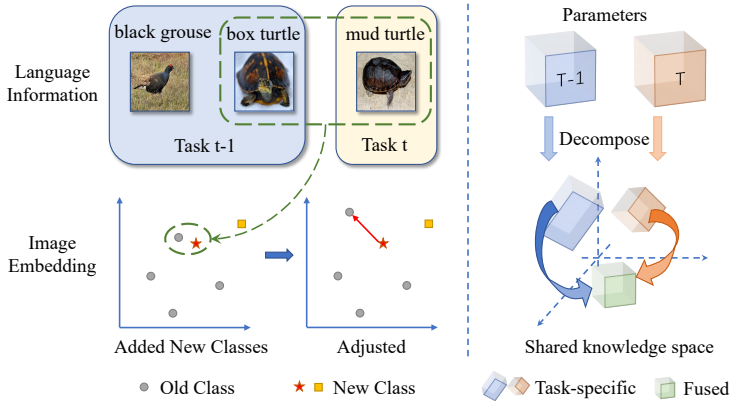


图 1: 在跨任务的类别增量学习（CIL）中，语义相似的类别会带来很大困难，这时语言信息可以帮助在遇到新数据时识别相邻的旧类别和新类别。然后可以相应地调整旧类别的图像特征表示。除此之外，为了减少遗忘程度，我们还进一步采用了分解的参数融合策略。我们将从两个连续任务中学到的参数分解为共享知识和任务特定知识。接着我们基于这种分解方式来融合参数。

方法以处理这类问题：基于正则化的方法、基于重放的方法和基于参数隔离的方法 [4]。这些方法通过向损失函数添加正则化项、重放旧数据样本或为每个类别分配专用参数来保留先前的知识。然而，这些方法大多依赖于随机初始化参量的模型，这可能不是增量学习的最佳选择。因此，对已经在大规模数据集上经过预训练的模型应用一些方法来减少遗忘是一个重要的研究方向。

在大规模数据集上预训练的模型具有很强的泛化能力并且对于下游任务中灾难性遗忘展示出较好的鲁棒性 [18,28,43,46]。此外，视觉语言预训练模型（CLIP） [30] 在持续学习中展现了强大的零样本能力 [38]，对下游任务也具有高适应性 [8,41,54]。利用预训练模型出色的特征提取能力，每个增量步骤只需要更新少量参数，从而降低了遗忘的风险。相比之下，从头开始训练的模型缺乏这种优势，其随着时间增加可能会导致性能严重下降。因此，预训练模型在持续学习中发挥了重要作用。目前使用预训练模型进行持续学习的两种主要策略是：微调模型 [48] 或者在模型上持续扩展少量参数，如基于提示的方法 [33,43–45] 或者添加适配器 [52]。

微调模型可能会损害原始模型的特征提取能力并导致灾难性遗忘，即使有正则化约束也会如此。扩展参数可能会减少对原始模型的干扰，但时间和空间成本会随时间不断增加。对于视觉语言预训练模型，语言编码器为其提

供了丰富的信息，而这对持续学习是有益的。但是现有的大多数方法只使用文本特征进行分类，并没有充分发掘它们在减少遗忘程度上的潜力。

在本文中，我们提出了一种使用文本特征来增强类增量学习中邻近类别分类能力的方法。这一方法基于这样的观察：CLIP 依赖于固定的文本特征进行分类。文本特征决定了决策边界。当新类别出现时，新的决策边界可能会将旧类别样本的一部分划分到新类别中。为了解决这个问题，我们需要增强邻近类别之间的距离。而使用文本特征可以推断旧类别和新类别之间的关系。我们通过计算新旧类别文本特征之间的距离来决定邻近类别的配对。由于新类别有足够的数据进行学习，我们不需要修改其表示。相反，我们专注于调整受新类别影响的旧类别，如图 1所示。我们训练一个单独的线性层，以防止损害预训练模型的特征提取能力。

此外，我们为线性自适应层设计了一种分解的参数融合方法，该方法不会随着任务的增加而增加参数数量。与直接计算参数平均值不同，这种方法相比之下会更加精细，并且考虑了不同任务之间共用的知识。为了平衡稳定性和可塑性，我们根据当前任务所引起的参数变化合并了增加任务前后的参数。这意味着我们不需要在训练过程中添加额外的蒸馏损失来约束参数变化，这大大降低了训练成本。

本文的主要成果如下：我们通过使用类别名称的文本特征开发了一种可以减少 CLIP 模型遗忘的方法；我们提出了一种简单有效的分解参数融合方法，可以用于预训练模型线性层的适配器；我们目前在几个数据集上取得了最优的结果。

2 相关工作

2.1 类增量学习（CIL）

增量学习方法可以根据它们使用的主要策略分为三种类型 [4]。基于正则化的方法通过在模型的参数或输出上施加一些约束以抑制对旧类别的遗忘。MAS [1] 和 EWC [15] 计算每个参数对旧任务的重要程度，并且添加一个正则化项以避免灾难性遗忘。Coil [51] 使用基于最优传输的知识蒸馏在不同任务中共享知识。一些方法 [6, 20, 49] 使用知识蒸馏的损失作为正则化项。基于重放的方法通过旧类别的一些样本或特征（示例）来保留对旧类别的记忆，然后在训练新类别时与新类别样本一起重新使用它们。还有一些方法正在探索如何减少示例的存储占用 [24]，或者寻找更好的策略来选择可以被添

加到示例中的样本 [22,23,36]。除此之外，还可以保存额外的模型和样本以协助当前模型的训练 [2,31,32]。基于动态架构的方法动态调整模型的结构以适应新类别的学习 [7,42,46,47]。

使用预训练模型的 CIL 使用预训练模型的方法主要有两种类型。一种是微调模型本身的参数来调整特征表示。ZSCL [50] 利用大量外部数据对预训练模型进行蒸馏，以维持稳定的特征空间。Zhang 等人 [48] 采用不同的学习率来更新预训练的主干网络和分类器。另一种方法则保持预训练模型不变，通过添加参数来调整特征表示。Liu 等人 [21] 在预训练的 CLIP 模型中引入了一个适配器以适应增量任务。PROOF [53] 为每个任务训练一个适配器，并使用跨模态注意力来融合 CLIP 的语言和视觉信息。RanPac [27] 使用高维投影来分离特征。最近提出的基于提示的方法 [14,33,44,45] 根据预训练模型输出的特征选择相应的提示添加到模型中，随后重新获取特征并进行分类。其中，LGCL [14] 尝试引入语言引导。Ostapenko 等人 [29] 在预训练模型上添加了一个分类网络，使用预训练的潜在特征空间进行重放。

不依赖于样本存储的 CIL 有时由于隐私和内存限制的问题，模型无法存储旧类别的样本 [34]。一些现有方法没有使用示例，而是通过高斯分布来对数据进行建模并分类 [12,37,48]。其他方法则对原型进行过采样 [56] 或增强原型以模拟重放样本 [25,55]。最近有些工作使用模型合成旧任务的数据作为示例的替代品 [3,9] 。基于提示的方法使用冻结的主干和相对隔离的提示参数以避免使用示例 [14,33,44,45]。

3 预备信息

类增量学习定义 类增量学习算法对每个任务 t 训练一个模型 M_t，该模型能够对见过的所有类别进行分类，无需任务标识，即 $C_1 \cup C_2 \cup ... \cup C_t$。并且类别集合不相交，对于所有 $\forall i \neq j, C_i \cap C_j = \emptyset$。模型 M_t 只使用当前数据集 D_t 中的数据来更新前一个模型 M_{t-1}，而不需要访问以前的数据集。

CLIP 适配器 模型中加入一个小网络作为适配器 [8] 是一个将预训练的视觉-语言模型适应到下游任务的有效方法。我们用 $f_{img}(\cdot)$ 表示 CLIP 骨干网络的视觉特征提取器，用 $f_{text}(\cdot)$ 表示文本特征提取器，用 $A(\cdot)$ 表示线性适配器。给定图像输入 $\mathbf{x}_i$，类别 y_i 与固定的提示模板，例如，“a photo of a [CLS]”，记为 $\mathbf{t}_i$，输出结果如下：

$$p(y_i|\mathbf{x}_i) = \frac{\exp(\cos(A(f_{img}(\mathbf{x}_i)), f_{text}(\mathbf{t}_i))/\boldsymbol{\tau})}{\sum_{j=1}^{|y|} \exp(\cos(A(f_{img}(\mathbf{x}_i)), f_{text}(\mathbf{t}_j))/\boldsymbol{\tau})}, \tag{1}$$

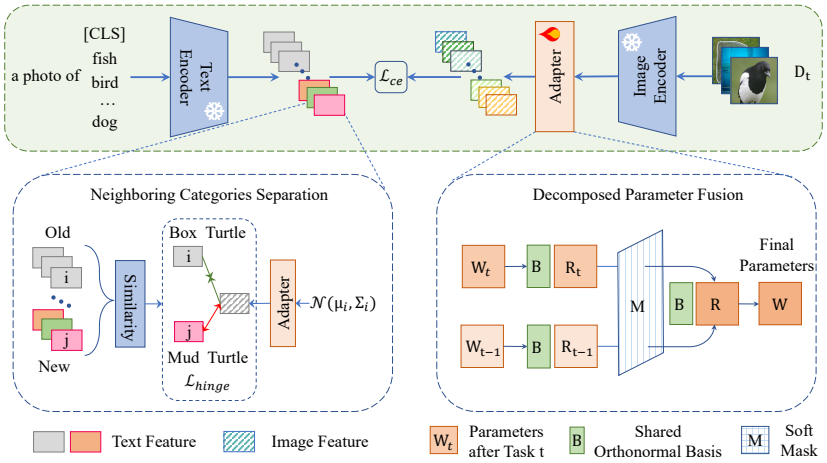


图 2: 我们方法的框架如下：邻近类别分离模块计算文本特征的相似性以识别邻近的类别。我们对旧类别的分布进行采样，并计算 hinge 损失。在参数融合模块中，我们首先将 $\mathbf{W}t$ 和 $\mathbf{W}t-1$ 分解到相同的标准正交基 $\mathbf{B}$ 中。然后，我们从分解后的参数 $\mathbf{R}t$ 和 $\mathbf{R}t-1$ 中计算出一个软掩码 $\mathbf{M}$，这个掩码作为融合权重。最后，我们使用融合后的参数 $\mathbf{R}$ 和基 $\mathbf{B}$ 来重构参数 $\mathbf{W}$。

其中 τ 是温度参数。如果文本编码器是冻结的，我们只需要在 CIL 中保留文本的嵌入而不是类别名称。我们采用交叉熵损失来微调适配器的参数，公式如下：

$$\mathcal{L}_{ce}(\mathbf{y}, \mathbf{p}) = -\sum_{i=1}^{n} y_i \log p_i. \tag{2}$$

特征生成 利用预训练骨干网络学习到的优良表示，我们可以使用高斯分布来近似每个类别的特征分布 [48]。我们从类别 c 的图像嵌入 $\mathbf{e}_c = f_{img}(\mathbf{x}_c)$ 中计算出质心 $\boldsymbol{\mu}_c$ 和协方差矩阵 $\boldsymbol{\Sigma}_c$。然后通过从高斯分布 $\mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ 中采样以为类别 c 生成图像嵌入。

4 方法

4.1 概述

如图 2所示，我们的框架包括一个预训练的 CLIP 模型和一个线性适配器。图像特征是将图像传递给图像编码器和适配器得到的，而标签特征则是将类别标签传递给文本编码器后生成。最终的分类结果是由特征向量之间的相似性来确定的。旧类别特征可以从它们各自的高斯分布中进行采样，并与

新数据一起输入适配器以计算分类损失。同时，我们根据文本特征的相似性选择邻近类别对。这些类别对对应的旧类别是受新数据影响的，对此我们从其高斯分布中采样更多数据并输入适配器，但是在此过程中仅计算分离损失。这样我们可以调整受相似新类别影响的旧类别的特征表示，从而减少因学习新类别而对旧类别造成的遗忘。在第 t 个任务训练阶段完成后，我们将前一阶段的适配器参数和当前适配器的参数融合，以获得该阶段的最终适配器。参数的自适应融合可以更好地平衡模型的可塑性和稳定性。

4.2 使用语言引导以增强邻近类别的分离

灾难性遗忘的一个表现是模型错误地将旧类别数据识别为新类别。当新类别与旧类别相似时，这种现象更容易发生。这些相似的类别属于邻近类别。它们通常在语言中具有相似的语义含义和相似的外观，例如‘金刚鹦鹉’和‘鹦鹉’。在添加新类别时，区分它们对于模型来说尤其困难。通过使用 CLIP 文本编码器，我们可以衡量类别名称之间的相似性，并将其用来指导适配器的学习过程。为了选择相似的类别对，我们使用类别名称的归一化文本特征来计算新旧类别之间的距离：

$$\mathbf{D} = \mathrm{dist}(f_{text}(\mathbf{t}_{new}), f_{text}(\mathbf{t}_{old})), \tag{3}$$

其中 $\mathbf{D}$ 是新旧类别的归一化文本特征 $f_{text}(\mathbf{t}_{new})$ 和 $f_{text}(\mathbf{t}_{old})$ 之间的欧几里得距离矩阵。学习新类别会干扰旧类别的性能。新旧类别之间的距离反映了干扰的程度，因此我们对其进行计算。然后，我们筛选出邻近类别集 $\mathcal{P}$：

$$p(y_i|\mathbf{x}_i) = \frac{\exp(\cos(A(f_{img}(\mathbf{x}_i)), f_{text}(\mathbf{t}_i))/\boldsymbol{\tau})}{\sum_{j=1}^{|y|}\exp(\cos(A(f_{img}(\mathbf{x}_i)), f_{text}(\mathbf{t}_j))/\boldsymbol{\tau})}, \tag{4}$$

其中 $\boldsymbol{\alpha}$ 代表阈值。$\mathbf{D}_{ij}$ 是旧类别 i 和新类别 j 的文本特征之间的距离。这个标准通过选择一个一对一关系的子集来降低多对多关系的复杂性。针对每对邻近类别，我们从正态分布中采样旧类别的特征来引入合页损失。由于我们没有旧类别的真实数据，而过度调整可能会损害旧类别的分类性能。因此，我们只使用合页损失来进行小而有效的调整。旧类别 c 的采样数据由 $\hat{\mathbf{e}}_c$ 表示，适配器由 A 表示：

$$\mathcal{L}_{hinge} = \sum_{k=1}^{|\mathcal{P}|}\max(\mathrm{dist}(A(\hat{\mathbf{e}}_c), f_{text}(\mathbf{t}_c)) - \mathrm{dist}(A(\hat{\mathbf{e}}_c), f_{text}(\mathbf{t}_{\not c})) + m, 0), \tag{5}$$

其中 m 是一个常数间隔，c 和 $\notin$ 分别表示集合 $\mathcal{P}$ 第 k 对邻近类别中的旧类别和新类别。这个损失函数可以使适配器更多地关注有可能混淆的邻近类别，同时最小化对旧类别表示的干扰。最终的损失函数与交叉熵损失 $\mathcal{L}_{ce}$ 结合如下：

$$\mathcal{L}_{ce}(\mathbf{y}, \mathbf{p}) = -\sum_{i=1}^{n} y_i \log p_i. \tag{6}$$

4.3 通过分解的参数融合以提高稳定性

我们引入了一个参数融合机制来维持线性适配器的稳定性，从而减少遗忘。我们评估每个新参数的重要性。由于梯度指示了基于新数据的参数更新方向，而参数在任务间的差异是梯度的加权和。因此，变化较大的参数对于学习新知识更为重要。为了获得细粒度权重的重要性矩阵以进行参数融合，我们计算训练后参数与前一任务参数之间的差异，并将其归一化到最大值：

$$\mathbf{M} = \min(1, \frac{|\mathbf{W}_{new} - \mathbf{W}_{old}|}{\max(|\mathbf{W}_{new} - \mathbf{W}_{old}|)} + b), \tag{7}$$

其中 $\mathbf{M}$ 表示每个新参数的重要性，$\mathbf{W}_{new}$ 表示当前任务训练得到的参数，$\mathbf{W}_{old}$ 表示前一任务的参数，b 是常数偏置。为了在相同的标准下比较 $\mathbf{W}_{new}$ 和 $\mathbf{W}_{old}$ 之间的差异，我们通过 SVD [11] 将 $\mathbf{W}_{old}$ 分解为正交基 $\mathbf{B}$，并计算矩阵 $\mathbf{W}_{new}$ 到 $\mathbf{B}$ 的投影：

$$\mathbf{W}_{old} \xrightarrow{decompose} \mathbf{B}\mathbf{R}_{old}, \tag{8}$$

$$\mathbf{R}_{new} = \mathbf{B}^{\mathsf{T}}\mathbf{W}_{new}. \tag{9}$$

参数矩阵 $\mathbf{W}_{old}$ 和 $\mathbf{W}_{new}$ 被表示为正交矩阵 $\mathbf{B}$ 的线性组合：$\mathbf{R}_{old}$ 和 $\mathbf{R}_{new}$，二者代表了特定任务的知识。矩阵 $\mathbf{B}$ 代表参数矩阵间的共享知识空间。因此可以将矩阵的差异转化为计算同一正交基的两个不同权重的差异。接着将分解得到的 $\mathbf{R}_{new}$ 和 $\mathbf{R}_{old}$ 代入方程式 (7)中的 $\mathbf{W}_{new}$ 和 $\mathbf{W}_{old}$ 来计算 $\mathbf{M}$。然后我们计算矩阵 $\mathbf{R}$ 和最终参数 $\mathbf{W}$：

$$\mathbf{R} = (\mathbf{J} - \mathbf{M}) \odot \mathbf{R}_{old} + \mathbf{M} \odot \mathbf{R}_{new}, \tag{10}$$

$$\mathbf{W} = \mathbf{B}\mathbf{R}, \tag{11}$$

其中 $\mathbf{J}$ 表示全 1 矩阵，$\odot$ 表示逐元素乘积。

5 实验

5.1 实验设置

数据集 我们使用三个数据集进行实验：CIFAR-100 [16]、ImageNet-1K [5]、ImageNet-100、ImageNet-R [13] 和 CUB-200 [40]。CIFAR-100 数据集包含 100 个类别，每个类别有 600 张彩色图像，图像分辨率为 32×32 像素。其中有 500 张图像分配给训练集，100 张图像分配给测试集。ImageNet-1K 数据集包含 1000 个类别，其子集 ImageNet-100 包含 100 个选定的类别。而 ImageNet-R 数据集则是从 ImageNet 中衍生而来，包括艺术、卡通、涂鸦、刺绣、电子游戏等各种风格的图像，它们是 ImageNet 数据集中 200 个类别的不同表现形式。我们遵循 [38,44] 的工作将数据集分为训练集和测试集。CUB-200 [40] 数据集广泛用于细粒度视觉分类任务，包含 11,788 张属于各色鸟类亚种的图像。

方法比较 我们与以下 CIL 方法进行比较：L2P++ [45]、Dual-Prompt [44]、CODA [33]、SLCA [48],ADAM-Adapter [52] 和 PROOF [53]。Continual-CLIP [38] 指的是 CLIP 模型的零样本性能。PROOF [53] 是一种基于 CLIP 并且使用示例的方法。为确保比较的公平性，所有方法都使用相同的 OpenAI CLIP 预训练权重 [30]。Dual-Prompt、L2P++ 和 CODA 三种方法的结果是通过运行 CODA 的公开可用代码获得的。SLCA、ADAM-Adapter、PROOF 和 Continual-CLIP 方法的结果分别来自它们各自的公共代码。

评估指标 在测试数据中，训练完第 t 个任务后对前 t 个任务的平均精度表示为 At。Avg 是所有任务准确率的平均值。Last 是经过最后任务后的平均准确率。

实验细节 我们使用 PyTorch 来开发这个模型，并在 RTX 3090 GPU 上运行它。其骨干网络是 CLIP 的 ViT-B/16 版本。我们使用 Adam 优化器以 15 个周期来训练模型，将初始学习率设置为 0.001。并且使用了 MultiStepLR 调度器，在第 4 个和第 10 个周期将学习率降低 0.1 倍。文本特征距离的默认阈值设为 0.65 以选择相邻类别对。我们的方法每周期大约使用 2000 个采样数据来模拟重放样本，而这恰与常规设置中添加的数据量相匹配。在每次迭代中，我们为每个被阈值选中的类别额外采样 20 个特征。由于一些类别的可用数据不足，有时无法获得一个满秩的协方差矩阵。对此我们按照之前的工作 [17,39]，使用协方差收缩来获得满秩矩阵。最终对类别顺序采用几种不同的洗牌方式运行实验，并获得这些顺序的平均值。

表 1: 在 CIFAR100 上进行持续学习的实验结果。B 代表基础类别的数量，而 Inc 代表增量类别的数量。所有基于基线的结果都是根据已发布的代码，并使用 VIT-B/16 的 CLIP 预训练权重进行复现的。我们对类别顺序进行了几种不同的随机打乱，并得到了这些顺序的平均值。

Method	Examplar	B0 Inc5		B0 Inc10		B0 Inc20		B50 Inc5		B50 Inc10	
		Avg	Last	Avg	Last	Avg	Last	Avg	Last	Avg	Last
PROOF [53]	✓	85.12	76.13	84.88	76.29	84.11	76.86	83.22	76.25	83.17	76.5
L2P ++ [45]	✗	79.18	68.67	81.90	73.08	84.39	77.37	58.57	18.04	76.51	48.52
DualPrompt [44]	✗	79.74	69.91	81.45	72.51	85.19	77.47	58.55	15.26	72.00	45.05
CODA [33]	✗	69.78	41.98	76.98	62.25	78.65	65.29	58.45	15.99	67.88	28.77
Continual-CLIP [38]	✗	75.93	66.68	75.15	66.68	74.01	66.68	70.79	66.68	70.77	66.68
SLCA [48]	✗	78.96	66.84	80.53	67.58	85.25	76.99	**86.99**	76.8	**86.55**	**79.92**
ADAM-Adapter [52]	✗	70.18	58.12	75.76	65.50	77.28	67.89	83.38	76.94	83.21	76.94
ours	✗	**86.87**	**79.26**	**86.19**	**79.04**	**85.73**	**79.24**	85.03	**79.64**	84.73	79.36

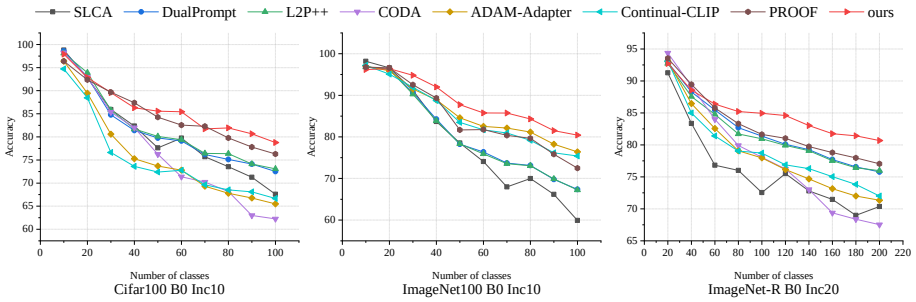


图 3: 我们的方法与其他最新技术水平基线在 CIFAR100、ImageNet100 和 ImageNet-R 上的准确率曲线。

5.2 结果比较

表 1、表 2和表 3展示了我们的方法与现有方法在三个数据集 CIFAR100、ImageNet100 和 ImageNet-R 上的比较结果。我们的方法在大多数实验设置下以显著的优势超过其他方法。在 ImageNet100 数据集上，我们的方法的最终准确率至少比其他方法高出 1.58%。在 ImageNet-R 数据集的基础 0 设置实验中，我们的方法在最终准确率上至少比其他方法高出 1.08%。这表明通过使用文本信息和参数融合的方法可以在减少遗忘和维持模型稳定性方面取得良好的效果。与零样本 Continual-CLIP 相比，我们的方法仅在模型结构中添加了一个可学习的线性层，但是也有明显的进步。这说明了我们的方法不仅仅依赖于预训练模型的泛化能力以实现良好的性能。图 3展示了随

表 2: 在 ImageNet100 上进行持续学习的实验结果。

Method	Exemplar	B0 Inc5		B0 Inc10		B0 Inc20		B50 Inc5		B50 Inc10	
		Avg	Last	Avg	Last	Avg	Last	Avg	Last	Avg	Last
PROOF [53]	✓	86.92	75.52	84.71	72.48	81.92	68.56	84.16	74.44	82.78	71.04
L2P ++ [45]	✗	75.43	62.10	80.51	67.22	84.12	73.70	62.00	22.15	74.11	49.46
DualPrompt [44]	✗	75.40	61.10	80.65	67.38	84.65	74.24	62.10	22.36	74.20	49.78
CODA [33]	✗	51.64	24.94	64.13	34.76	69.78	43.96	57.33	19.95	65.14	28.80
Continual-CLIP [38]	✗	85.74	75.40	84.98	75.40	84.03	75.40	81.35	75.40	81.09	75.40
SLCA [48]	✗	78.40	63.36	78.63	59.92	84.08	71.08	86.47	72.22	86.26	71.18
ADAM-Adapter [52]	✗	85.78	75.72	85.84	76.40	85.85	77.08	84.90	78.58	84.60	78.58
ours	✗	**87.59**	**79.87**	**87.51**	**80.23**	**86.72**	**80.10**	**86.53**	**80.16**	**86.36**	**80.22**

表 3: 在 ImageNet-R 上进行持续学习的实验结果。

Method	Exemplar	B0 Inc10		B0 Inc20		B0 Inc40		B100 Inc10		B100 Inc20	
		Avg	Last	Avg	Last	Avg	Last	Avg	Last	Avg	Last
PROOF [53]	✓	82.69	77.25	82.83	77.05	82.63	77.12	81.61	77.10	81.78	77.17
L2P ++ [45]	✗	76.87	68.78	81.67	75.98	82.81	77.87	56.17	17.90	67.73	43.28
DualPrompt [44]	✗	77.07	69.41	82.01	75.77	83.77	78.64	57.37	19.18	69.18	45.37
CODA [33]	✗	75.23	64.53	78.00	67.52	78.80	71.27	56.62	17.64	65.62	35.06
Continual-CLIP [38]	✗	79.84	72.00	79.12	72.00	77.59	72.00	76.93	72.00	76.76	72.00
SLCA [48]	✗	80.18	73.57	75.92	70.37	83.35	79.1	82.85	78.57	83.50	79.67
ADAM-Adapter [52]	✗	76.71	68.75	78.65	71.35	79.87	73.02	79.87	75.37	79.75	75.37
ours	✗	**86.28**	**79.62**	**85.58**	**80.28**	**84.69**	**80.18**	**84.12**	**80.04**	**83.99**	**80.35**

着类别数量增加，不同数据集和设置下平均准确率的下降趋势。这种方法可以显著减轻遗忘程度。

三种基于提示的方法在 base50 和 base100 的实验设置下会产生严重的提示使用失衡，这是因为它们在默认情况下对每个任务使用了相同数量的提示。由此会导致模型的预测偏向基础类别，使得注入新知识会变得困难。L2P 和 DualPrompt 在 ImageNet-R 数据集上的表现优于其在 ImageNet21k 上预训练的原始模型的性能。然而，CODA 在三个数据集上表现不佳，这说明其方法可能与特定的预训练模型高度耦合，例如在 Imagenet21k 上预训练的 ViT。SLCA 通过微调骨干网络和分类器在初始任务上取得了很好的性能。因此其在 base50 和 base100 设置下的实验结果相对较好。但是在长序列设置中，由于微调骨干网络而累积的遗忘会变得更加明显。

表 4: CUB200（基础类别 0 个，增量类别 20 个）和 ImageNet1K（基础类别 0 个，增量类别 100 个）上的结果

		PROOF	L2P ++	DualPrompt	CODA	Continual-CLIP	SLCA	ADAM-Adapter	ours
CUB200	Avg	**83.11**	71.90	71.74	66.61	60.60	73.30	78.80	83.04
	Last	75.53	62.99	62.14	50.88	51.16	60.39	70.61	**76.34**
ImageNet1K	Avg	76.23	79.30	79.39	76.99	72.96	79.10	76.60	**81.73**
	Last	65.26	69.60	69.79	66.96	64.44	68.27	68.74	**72.58**

ADAM 的主要思路是在初始任务上训练一个适配器模型，而在后续任务中不会再训练模型。因此其性能取决于初始任务中数据的分布比例。模型没有在后续任务上进行训练会保持其稳定性，但不会再学习新知识。而这种低可塑性影响了其性能。由于 ImageNet-R 中图像风格具有较大差异，所以该数据集要求模型学习更多知识以适应多种图像风格，导致这种方法在该数据集上的表现比统一风格的 CIFAR100 和 ImageNet100 更差。

PROOF 是为 CLIP 设计的，它随着每个任务扩展适配器，并使用跨模态注意力方法来融合文本和图像信息。但是这种方法没有考虑 CLIP 分类中邻近类别的影响。即使没有示例和扩展参数，我们的方法相比之下仍然具有优势。

我们还在大型数据集 ImageNet1K 和细粒度数据集 CUB200 上评估了我们的方法。结果如表 4所示。实验结果表明，我们的方法在大型数据集中仍然具有优势。尤其是在更困难的细粒度数据集 CUB200 上，我们的方法相比于 CLIP 的零样本性能有了很大的提升。

总的来说，我们的方法不会随着任务的增加而扩展模型，但是会同时保持对新数据的学习，并减轻增量任务中的遗忘程度。更多的结果在补充材料中展示。

5.3 关于消融的研究与其他分析

模块消融分析 表 5展示了我们方法中不同组件的结果。”随机”意味着随机选择类别对而不是使用文本特征的距离。在这种情况下准确率非常接近基线，而使用完整模块和文本特征可以比基线提高 2.24% 的准确率。这表明通过文本引导来选择最近类别是有效的。通过对没有矩阵分解的参数融合方法和具有矩阵分解的完整参数融合方法进行对比，可以发现使用细粒度重要性矩阵的参数融合方法很有效果，而矩阵分解进一步增强了它的性能。

表 5: 在 ImageNet100 上，基础类别 0 个，增量类别 10 个的模块消融实验结果如下：SG 表示从高斯分布中采样少量旧类别的特征。$\mathcal{L}hinge$(random) 是指为 $\mathcal{L}hinge$ 随机选择类别对。PF w/o MD 的含义是仅使用公式7进行参数融合，而 MD 表示矩阵分解。

Ablation	Last Accuracy ↑
Adapter-finetune + SG (Baseline)	73.80
Baseline + $\mathcal{L}_{hinge}$(random)	74.08
Baseline + $\mathcal{L}_{hinge}$	76.04
Baseline + $\mathcal{L}_{hinge}$ + PF w/o MD	79.28
Baseline + $\mathcal{L}_{hinge}$ + PF w/ MD (Full)	80.23

表 6: 在不同的消融实验设置下，模型对“kingsnake”（王蛇）的 50 张测试图像的预测结果如下。另外三种蛇类是通过文本特征相似性选出的邻近新类别。在补充材料中提供了更多相似性更高的结果。

	w/o $\mathcal{L}_{hinge}$	w/ $\mathcal{L}_{hinge}$
kingsnake	25	35
night snake	11	8
worm snake	2	1
eastern hog-nosed snake	10	5
others	2	1
accuracy	0.5	0.7

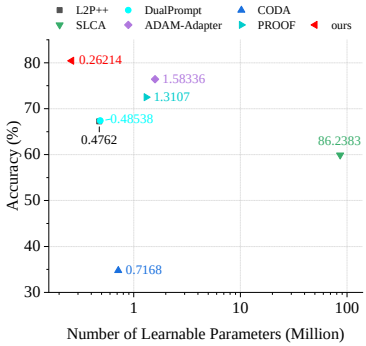
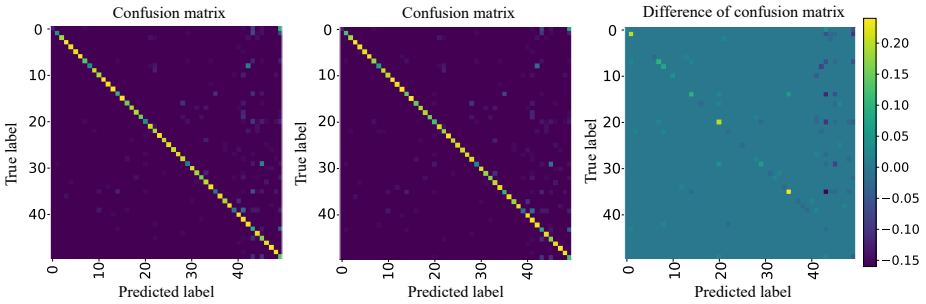


图 4: 在准确性和可学习参数方面对不同方法的比较。

训练成本分析 图 4展示了 ImageNet100 数据集上，我们的方法与其他方法在增量参数大小上的比较。如 SLCA 所做的一样，在使用相同骨干网络的情况下，冻结骨干参数的更新成本远低于微调整个模型。在冻结骨干并添加可学习参数的方法中，我们的方法添加的参数最少。尽管我们在纯视觉编码器方法中增加了一个文本编码器，但由于标签文本是固定的，因此只需要在整个训练过程中一次性计算标签的文本特征即可，而且标签的数量远远低于数据量。因此，与处理许多图像和多次迭代的视觉编码器相比，文本编码器的计算成本可以忽略不计。对于分解的参数融合，我们只在每个任务训练后分解一次矩阵。与大量的训练时间相比，这也是可以忽略不计的。

邻近类别分类 如图 5所示，我们的方法有效地纠正了将旧类别错误分类到新类别的问题。图 5c显示，错误分类到新类别的样本数量减少，正确分类的样本数量增加，而对其他类别的负面影响很小。表 6展示了一些受影响类别的例子，说明通过文本特征相似性而选出的与旧类别相近的新类别是造成错误预测的主要原因。其中选出的类别占 25 个错误预测中的 23 个。更多类似的例子请参见补充材料。用于选定类别的 $\mathcal{L}_{hinge}$ 可以有效减少对于受影响旧类别的错误预测。



(a) Baseline **(b)** Baseline + $\mathcal{L}_{hinge}$ **(c)** Difference between Fig. b and Fig. a (i. e. b - a)

图 5: 在 ImageNet100 B0 Inc10 实验中，前 5 个任务的混淆矩阵及其差异。为了提高可读性，我们仅展示了前 5 个任务的混淆矩阵。

阈值消融 在图图 6中，我们研究了用于邻近类别的阈值 的效果。由于大多数距离都大于 0.5，我们将阈值从 0.5 开始逐渐增加。随着阈值的增加，更多的邻近类别将被选中，进而提高了性能。但是过大的阈值会选出许多不相似的类别对，这将干扰分类并增加计算量。所以选择较多的不同类别并无益处。因此，我们在所有实验中使用 0.65 作为阈值。

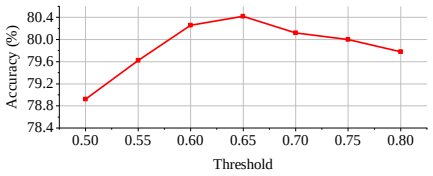


图 6: 在 ImageNet100 B0 Inc10 实验中使用不同阈值 α 进行的实验。

基于提示的方法与 CLIP 文本编码器 为了进行公平的比较，我们将 CLIP 的文本编码器作为分类器添加到了与分类器无关的基于提示的方法中。结果如表表 7所示。在 L2P++ 和 DualPrompt 中，不同任务的键和值相对独立，这导致旧类别的表示变化很小。当新类别的文本特征与旧类别相似时，使用固定的 CLIP 文本特征而不是可训练的线性层作为分类器可能会导致旧类别样本的误分类。因此，这两种方法在使用 CLIP 文本编码器时会出现性能下降的情况。CODA 使用权重来组合所有提示组件，使旧类别能够从新扩展的参数中受益。当使用信息丰富的文本分类器时，性能会得到提高。与这些方法相比，我们的方法可以更好地利用 CLIP 文本编码器。

训练时不同的采样方法 PASS [55] 和 Napa-vq [25] 并不是基于预训练模型设计的，故不能直接应用于预训练模型。因此，我们在方法中使用 PASS [55]

表 7: 在 ImageNet-R B0 Inc20 设置下，不同方法的最后准确率

Backbone	CODA	DualPrompt	L2P++
w/o text encoder	67.52	75.77	75.98
w text encoder	69.93	72.30	71.97

表 8: 在 CIFAR100 上，基础类别 0 个，增量类别 20 个的设置下，不同采样方法的结果。

Method	PASS	Napa-vq	ours
Avg	84.38	84.23	**86.19**
Last	77.18	77.22	**79.04**

表 9: 在 ImageNet-R B0 Inc20 设置下，传统方法的比较结果如下。这些传统方法的结果是根据 PILOT [35] 在 ImageNet21K 上预训练的权重复现的。

Method	Coil [51]	DER [47]	iCaRL [31]	FOSTER [42]	ours
Avg	80.48	81.16	72.76	82.49	**85.58**
Last	73.12	75.10	61.62	76.00	**82.28**

和 Napa-vq [25] 采样方法替换了特征采样和类别分离过程，如表 8所示。PASS 忽略了邻近类别，而 Napa-vq 更多地依赖于从零开始的 Na-vq 建模。我们的策略更有效地利用了 CLIP 模型。

5.4 与传统方法的比较

传统方法即使没有使用预训练模型但也使用了示例，并在类增量学习中也取得了良好的结果。在表 9中，我们比较了使用预训练的 VIT-B/16 以进行初始化的传统方法（在 ImageNet21K 上）。我们还用与我们相同的初始化方式评估了这些基线，但是它们的性能要差得多。所以即使没有使用示例，我们的方法仍然比这些较为先进的方法有明显优势。

6 结论

在本文中，我们研究了使用预训练视觉-语言模型的增量学习问题。我们发现，引入类别的文本特征来调整受新数据影响较大的旧类别表示可以有效缓解遗忘问题。此外，一个简单的线性适配器配合参数融合策略可以高效地维持模型稳定性并减少遗忘。实验证明了我们方法的有效性。手动选择阈值在一定程度上限制了我们的方法。未来的工作可以设计一个机制来动态适应阈值，并设计一个更有效的参数融合机制。文本和图像之间的相互影响也可以进一步研究。

致谢

这项研究由国家自然科学基金（编号 62206135）、中国科协青年精英科学家资助计划（编号 2023QNRC001）、天津市自然科学基金（编号 23JCQNJC01470）以及中央高校基本科研业务费（南开大学）资助。计算支持由南开大学超算中心提供。

参考文献

1. Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., Tuytelaars, T.: Memory aware synapses: Learning what (not) to forget. In: Eur. Conf. Comput. Vis. pp. 139–154 (2018)
2. Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P.K., Torr, P.H., Ranzato, M.: On tiny episodic memories in continual learning. arXiv preprint arXiv:1902.10486 (2019)
3. Choi, Y., El-Khamy, M., Lee, J.: Dual-teacher class-incremental learning with data-free generative replay. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3543–3552 (2021)
4. De Lange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., Tuytelaars, T.: A continual learning survey: Defying forgetting in classification tasks. IEEE Trans. Pattern Anal. Mach. Intell. **44**(7), 3366–3385 (2021)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 248–255. Ieee (2009)
6. Douillard, A., Cord, M., Ollion, C., Robert, T., Valle, E.: Podnet: Pooled outputs distillation for small-tasks incremental learning. In: Eur. Conf. Comput. Vis. pp. 86–102. Springer (2020)
7. Douillard, A., Ramé, A., Couairon, G., Cord, M.: Dytox: Transformers for continual learning with dynamic token expansion. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 9285–9295 (2022)
8. Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., Qiao, Y.: Clip-adapter: Better vision-language models with feature adapters. Int. J. Comput. Vis. pp. 1–15 (2023)
9. Gao, Q., Zhao, C., Ghanem, B., Zhang, J.: R-dfcil: Relation-guided representation learning for data-free class incremental learning. In: Eur. Conf. Comput. Vis. pp. 423–439. Springer (2022)

10. Geng, C., Huang, S.j., Chen, S.: Recent advances in open set recognition: A survey. IEEE Trans. Pattern Anal. Mach. Intell. **43**(10), 3614–3631 (2020)

11. Golub, G.H., Reinsch, C.: Singular value decomposition and least squares solutions. In: Handbook for Automatic Computation: Volume II: Linear Algebra, pp. 134–151. Springer (1971)

12. Hayes, T.L., Kanan, C.: Lifelong machine learning with deep streaming linear discriminant analysis. In: IEEE Conf. Comput. Vis. Pattern Recog. Worksh. pp. 220–221 (2020)

13. Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., Song, D., Steinhardt, J., Gilmer, J.: The many faces of robustness: A critical analysis of out-of-distribution generalization. Int. Conf. Comput. Vis. (2021)

14. Khan, M.G.Z.A., Naeem, M.F., Van Gool, L., Stricker, D., Tombari, F., Afzal, M.Z.: Introducing language guidance in prompt-based continual learning. In: Int. Conf. Comput. Vis. pp. 11463–11473 (2023)

15. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al.: Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences **114**(13), 3521–3526 (2017)

16. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)

17. Kumar, S., Zaidi, H.: Gdc-generalized distribution calibration for few-shot learning. arXiv preprint arXiv:2204.05230 (2022)

18. Lee, K.Y., Zhong, Y., Wang, Y.X.: Do pre-trained models benefit equally in continual learning? In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 6485–6493 (2023)

19. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Trans. Pattern Anal. Mach. Intell. **40**(12), 2935–2947 (2017)

20. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Trans. Pattern Anal. Mach. Intell. **40**(12), 2935–2947 (2017)

21. Liu, X., Cao, X., Lu, H., Xiao, J.w., Bagdanov, A.D., Cheng, M.M.: Class incremental learning with pre-trained vision-language models. arXiv preprint arXiv:2310.20348 (2023)

22. Liu, Y., Su, Y., Liu, A.A., Schiele, B., Sun, Q.: Mnemonics training: Multi-class incremental learning without forgetting. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 12245–12254 (2020)

23. Lopez-Paz, D., Ranzato, M.: Gradient episodic memory for continual learning. Adv. Neural Inform. Process. Syst. **30** (2017)

24. Luo, Z., Liu, Y., Schiele, B., Sun, Q.: Class-incremental exemplar compression for class-incremental learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 11371–11380 (2023)

25. Malepathirana, T., Senanayake, D., Halgamuge, S.: Napa-vq: Neighborhood-aware prototype augmentation with vector quantization for continual learning. In: Int. Conf. Comput. Vis. pp. 11674–11684 (2023)

26. Masana, M., Liu, X., Twardowski, B., Menta, M., Bagdanov, A.D., Van De Weijer, J.: Class-incremental learning: survey and performance evaluation on image classification. IEEE Trans. Pattern Anal. Mach. Intell. **45**(5), 5513–5533 (2022)

27. McDonnell, M.D., Gong, D., Parvaneh, A., Abbasnejad, E., van den Hengel, A.: Ranpac: Random projections and pre-trained models for continual learning. Adv. Neural Inform. Process. Syst. **36** (2024)

28. Mehta, S.V., Patil, D., Chandar, S., Strubell, E.: An empirical investigation of the role of pre-training in lifelong learning. arXiv preprint arXiv:2112.09153 (2021)

29. Ostapenko, O., Lesort, T., Rodriguez, P., Arefin, M.R., Douillard, A., Rish, I., Charlin, L.: Continual learning with foundation models: An empirical study of latent replay. In: Conference on Lifelong Learning Agents. pp. 60–91. PMLR (2022)

30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: Int. Conf. Mach. Learn. pp. 8748–8763. PMLR (2021)

31. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2001–2010 (2017)

32. Sarfraz, F., Arani, E., Zonooz, B.: Error sensitivity modulation based experience replay: Mitigating abrupt representation drift in continual learning. In: Int. Conf. Learn. Represent. (2023), `https://openreview.net/forum?id=zlbci7019Z3`

33. Smith, J.S., Karlinsky, L., Gutta, V., Cascante-Bonilla, P., Kim, D., Arbelle, A., Panda, R., Feris, R., Kira, Z.: Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 11909–11919 (2023)

34. Smith, J.S., Tian, J., Halbe, S., Hsu, Y.C., Kira, Z.: A closer look at rehearsal-free continual learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2409–2419 (2023)

35. Sun, H.L., Zhou, D.W., Ye, H.J., Zhan, D.C.: Pilot: A pre-trained model-based continual learning toolbox. arXiv preprint arXiv:2309.07117 (2023)

36. Sun, Z., Mu, Y., Hua, G.: Regularizing second-order influences for continual learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 20166–20175 (2023)

37. Tang, Y.M., Peng, Y.X., Zheng, W.S.: When prompt-based incremental learning does not meet strong pretraining. In: Int. Conf. Comput. Vis. pp. 1706–1716 (2023)

38. Thengane, V., Khan, S., Hayat, M., Khan, F.: Clip model is an efficient continual learner. arXiv preprint arXiv:2210.03114 (2022)

39. Van Ness, J.: On the dominance of non-parametric bayes rule discriminant algorithms in high dimensions. Pattern Recognition **12**(6), 355–368 (1980)

40. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)

41. Wang, E., Peng, Z., Xie, Z., Yang, F., Liu, X., Cheng, M.M.: Unlocking the multimodal potential of clip for generalized category discovery (2024), `https://arxiv.org/abs/2403.09974`

42. Wang, F.Y., Zhou, D.W., Ye, H.J., Zhan, D.C.: Foster: Feature boosting and compression for class-incremental learning. In: Eur. Conf. Comput. Vis. pp. 398–414. Springer (2022)

43. Wang, Y., Huang, Z., Hong, X.: S-prompts learning with pre-trained transformers: An occam' s razor for domain incremental learning. Adv. Neural Inform. Process. Syst. **35**, 5682–5695 (2022)

44. Wang, Z., Zhang, Z., Ebrahimi, S., Sun, R., Zhang, H., Lee, C.Y., Ren, X., Su, G., Perot, V., Dy, J., et al.: Dualprompt: Complementary prompting for rehearsal-free continual learning. In: Eur. Conf. Comput. Vis. pp. 631–648. Springer (2022)

45. Wang, Z., Zhang, Z., Lee, C.Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J., Pfister, T.: Learning to prompt for continual learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 139–149 (2022)

46. Wu, T.Y., Swaminathan, G., Li, Z., Ravichandran, A., Vasconcelos, N., Bhotika, R., Soatto, S.: Class-incremental learning with strong pre-trained models. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 9601–9610 (2022)

47. Yan, S., Xie, J., He, X.: Der: Dynamically expandable representation for class incremental learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3014–3023 (2021)

48. Zhang, G., Wang, L., Kang, G., Chen, L., Wei, Y.: Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. arXiv preprint arXiv:2303.05118 (2023)

49. Zhang, J., Zhang, J., Ghosh, S., Li, D., Tasci, S., Heck, L., Zhang, H., Kuo, C.C.J.: Class-incremental learning via deep model consolidation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1131–1140 (2020)

50. Zheng, Z., Ma, M., Wang, K., Qin, Z., Yue, X., You, Y.: Preventing zero-shot transfer degradation in continual learning of vision-language models. In: Int. Conf. Comput. Vis. pp. 19125–19136 (2023)

51. Zhou, D.W., Ye, H.J., Zhan, D.C.: Co-transport for class-incremental learning. In: ACM Int. Conf. Multimedia. pp. 1645–1654 (2021)

52. Zhou, D.W., Ye, H.J., Zhan, D.C., Liu, Z.: Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. arXiv preprint arXiv:2303.07338 (2023)

53. Zhou, D.W., Zhang, Y., Ning, J., Ye, H.J., Zhan, D.C., Liu, Z.: Learning without forgetting for vision-language models. arXiv preprint arXiv:2305.19270 (2023)

54. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. Int. J. Comput. Vis. **130**(9), 2337–2348 (2022)

55. Zhu, F., Zhang, X.Y., Wang, C., Yin, F., Liu, C.L.: Prototype augmentation and self-supervision for incremental learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 5871–5880 (2021)

56. Zhu, K., Zhai, W., Cao, Y., Luo, J., Zha, Z.J.: Self-sustaining representation expansion for non-exemplar class-incremental learning. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 9296–9305 (2022)

A ImageNet100 的类别

遵循先前研究的设置 [7, 38, 47]，我们从 ImageNet1k [5] 数据集中选取了 100 个类别作为 ImageNet100 子集。ImageNet100 的类别名称包括：

丁鱼，金鱼，大白鲨，虎鲨，双髻鲨，电鳐，黄貂鱼，公鸡，母鸡，鸵鸟，燕雀，金翅雀，家雀，朱顶雀，美洲知更鸟，画眉，松鸦，喜鹊，山雀，美洲水鸫，风筝（猛禽），白头鹰，秃鹫，大灰猫头鹰，火蜥蜴，光滑蝾螈，蝾螈，斑点蝾螈，墨西哥钝口螈，美洲牛蛙，树蛙，尾蛙，红海龟，棱皮海龟，泥龟，陆龟，箱龟，带状壁虎，绿鬣蜥，卡罗莱纳安乐蜥，沙漠草原鞭尾蜥，变色龙，褶颈蜥，鳄蜥，吉拉怪兽，欧洲绿蜥，变色龙，科莫多龙，尼罗鳄，美洲鳄，三角龙，蟒蛇，颈环蛇，东猪鼻蛇，光滑绿蛇，王蛇，束带蛇，水蛇，藤蛇，夜蛇，蟒蛇，非洲岩蟒，印度眼镜蛇，绿曼巴，海蛇，撒哈拉角蝰，东菱背响尾蛇，侧行响尾蛇，三叶虫，收割者，蝎子，黄园蛛，谷仓蜘蛛，欧洲园蛛，南黑寡妇，狼蛛，蜱，蜈蚣，黑松鸡，松鸡，草原松鸡，孔雀，鹌鹑，鹧鸪，非洲灰鹦鹉，金刚鹦鹉，硫冠凤头鹦鹉，鹦鹉，蜂虎，犀鸟，蜂鸟，翠鸟，巨嘴鸟，鸭，红胸秋沙鸭，鹅。

表 10: 在 ImageNet-R B0 Inc20 设置下传统方法的比较结果。这些传统方法的结果是根据 PILOT 研究 [35] 并使用 OpenAI 的 CLIP 预训练权重 [30] 复现的。

Method	Exemplar	Avg	Last
Coil [51]	✓	45.55	20.65
DER [47]	✓	81.85	73.27
iCaRL [31]	✓	77.25	64.52
FOSTER [42]	✓	76.81	70.23
ours	✗	85.58	80.28

B 更多实验结果

B.1 对 CLIP 权重进行初始化的传统方法

在10中，我们与传统方法进行了比较（该传统方法使用 CLIP 预训练的 VIT-B/16 模型进行初始化）。我们的方法具有明显的优势。

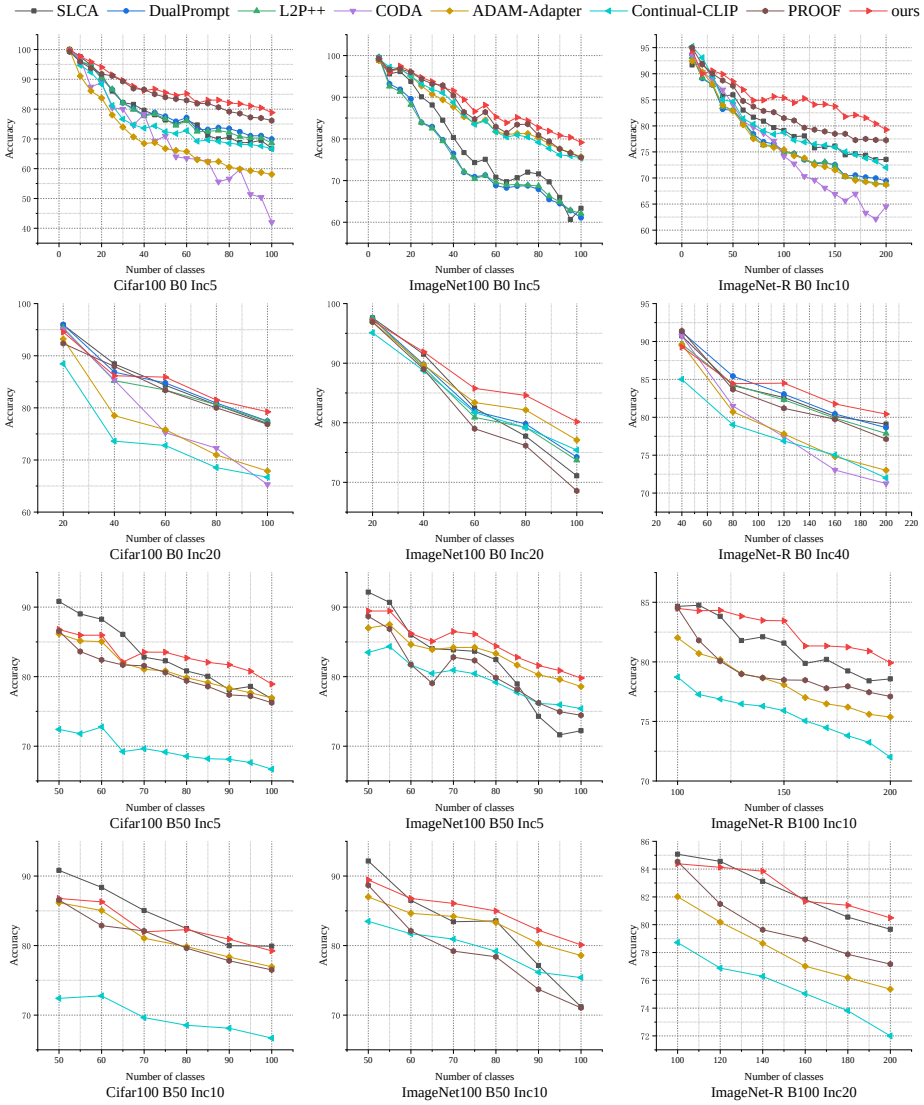


图 7: 在 CIFAR100 [16]、ImageNet100 [5] 和 ImageNet-R [13] 数据集上，我们的方法与其他最新技术水平基线在准确率曲线上的比较结果如下。

表 11: 在不同的消融实验设置下，对 50 张“海蛇测试图像的模型预测结果。“夜蛇”、“蠕蛇”和“东猪鼻蛇”是通过计算当前任务中新添加类别的文本特征相似性选择的邻近类别种类。

	w/o $\mathcal{L}_{hinge}$	$\mathcal{L}_{hinge}$
sea snake	37	42
night snake	5	2
worm snake	1	0
eastern hog-nosed snake	4	1
others	3	5
accuracy	0.74	0.84

B.2 更多设置下的准确率曲线

如图 7所示，我们的方法在大多数情况下都优于其他方法。SLCA [48]对整个模型进行了微调。正因为如此，SLCA 在某些设置的初始任务上一开始表现更好，但随着任务的增加其遗忘速度迅速增加。

B.3 更多邻近类别的例子

在不同的消融设置下,所选邻近类别的一些实验结果展示在表 11、表 12和表 13中。每个类别包含 50 张测试图像。结果表明，我们的方法在调整旧类别的表示时，对新类别表示的干扰要小得多。

表 12: 在不同的消融实验设置下，对“光滑蝾螈”和“蝾螈”的模型预测结果。“光滑蝾螈”是旧类别，而“蝾螈”是选择得到的邻近类别（通过计算当前任务中新添加类别的文本特征相似性进行选择）。

	w/o $\mathcal{L}_{hinge}$		$\mathcal{L}_{hinge}$	
	smooth newt	newt	smooth newt	newt
smooth newt	4	0	10	2
newt	22	46	16	44
others	24	4	24	4
accuracy	0.08	0.92	0.2	0.88

表 13: 在不同的消融实验设置下，对“母鸡”、“鹌鹑”和“鹅”测试图像的模型预测结果。“母鸡”是旧类别。“鹌鹑”和“鹅”选择得到的邻近类别。（通过计算当前任务中新添加类别的文本特征相似性进行选择）

	w/o $\mathcal{L}_{hinge}$			$\mathcal{L}_{hinge}$		
	hen	quail	goose	hen	quail	goose
hen	39	0	0	43	0	0
quail	9	50	0	5	50	0
goose	1	0	49	1	0	48
others	1	0	1	1	0	2
accuracy	0.78	1	0.98	0.86	1	0.96

B.4 和 RanPAC 的比较

在表 14中展示了与 RanPAC [27]（使用 OpenAI CLIP）进行性能比较的结果。我们的设置与 RanPAC 的可学习参数数量相匹配，为 0.26M。而 RanPAC 则有 1M 到 2M 的额外参数。尽管在 CIFAR100 数据集上性能相近，但我们的方法在更多样化的 ImageNet-R 数据集上表现更佳。

表 14: 与 RANPAC 的比较（10 任务下）

	CIFAR100		ImageNet-R	
	Avg	Last	Avg	Last
RanPAC	87.15	80.96	84.15	77.66
RanPAC *	86.03	79.01	80.22	71.09
ours	86.19	79.04	85.58	80.28

如表 15所示，我们还探索了在 ImageNet-R 数据集上，对标签添加不同比例的随机噪声后的结果。可以明显看出，我们的方法对噪声具有相对较强的鲁棒性

表 15: 在 ImageNet-R (10 任务) 上对不同噪声比例的研究

	0%	5%	10%	20%
RanPAC	77.66	77.54	77.08	70.7
ours	80.28	79.55	79.25	77.12