中国图象图形学学会
CHINA SOCIETY OF IMAGE AND GRAPHICS

2026 中国图象图形大会
CHINESE CONGRESS ON IMAGE AND GRAPHICS

持续学习：过去，现在与未来

周大蔚

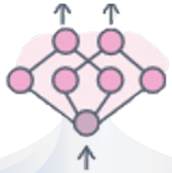
南京大学

zhoudw@nju.edu.cn

智能模型的泛化性

人工智能技术近年来快速发展，其核心问题是模型能力的泛化性

图像识别模型　文档理解模型　音频分析模型　音转文字模型　图学习模型　文本问答模型　分类回归模型　强化学习模型



人工智能技术

交通　音频　工业　图像　代码　视频　农业

自动驾驶任务面临的场景变化导致难以泛化

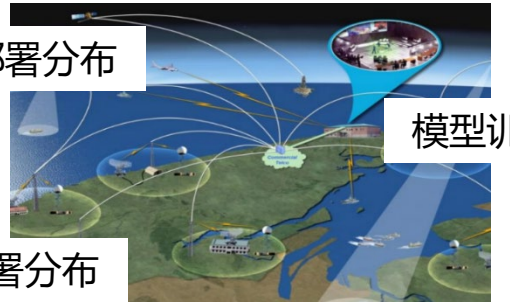
汽车厂商设想的场景

上路后遇到的场景

……

通信任务面临的数据分布变化导致难以泛化

模型部署分布



模型训练分布

模型部署分布

大模型的广泛应用

自2017年Transformer诞生以来，以大模型为代表的AI技术高速发展，泛化性显著提升

大模型诞生阶段
2017-2018年

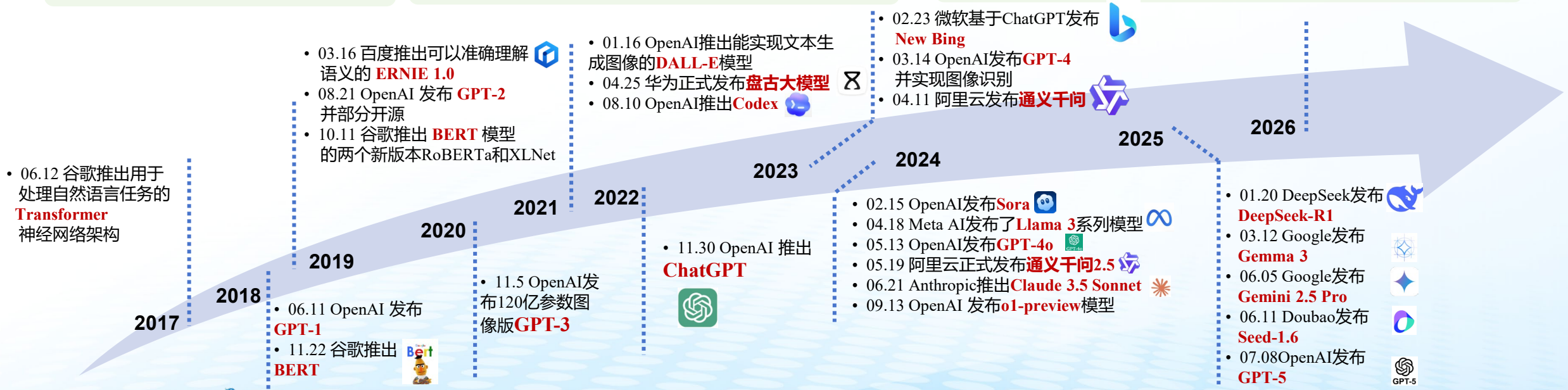
在诞生阶段，以**Transformer为代表的全新神经网络架构**，奠定了大模型的算法架构基础，使大模型技术的性能得到了显著提升。

大模型探索阶段
2019-2021年

在探索期，规模更大、性能更强的大模型涌现，**更高效的预训练、指令微调等开始出现**，被用于进一步提高推理能力和任务泛化。

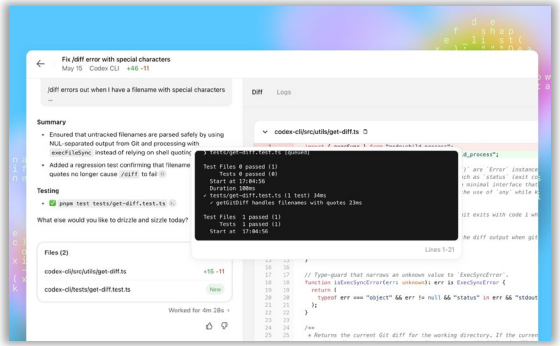
大模型爆发阶段
2022-2026年

在爆发期，**大数据、大算力和大算法的结合**，具备了**多模态理解**与**多类型内容生成能力。**

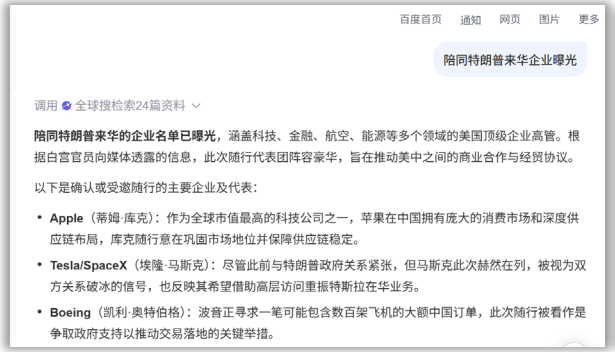


大模型与人工智能

基于海量数据上的预训练，大模型在众多场景中展现出强大的泛化能力



代码理解

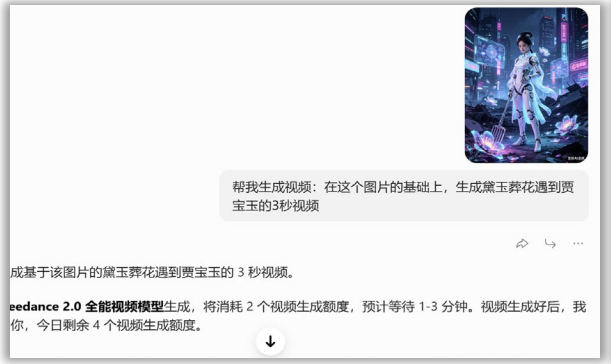


智能搜索

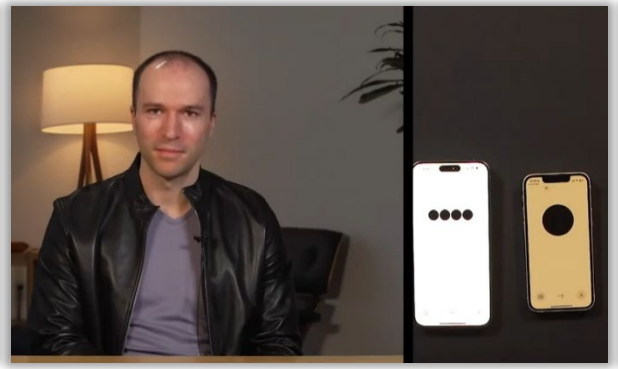
千问APP实时对话交互

“奥巴马数字人”

音乐创作



视频生成



GPT-4o与另一个GPT进行互动

具身智能登上春晚舞台

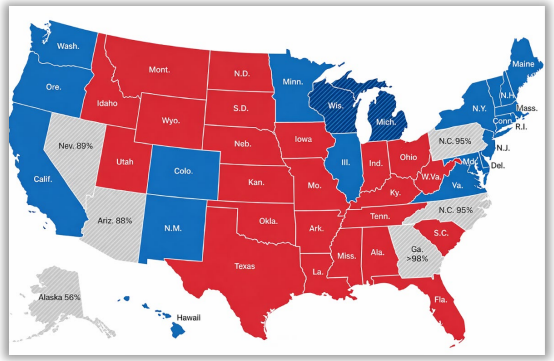
然而...

现实世界呈现动态开放特性，数据分布、知识内容、任务需求、规则与偏好等要素均会随时间推移不断变化

新的数据分布出现

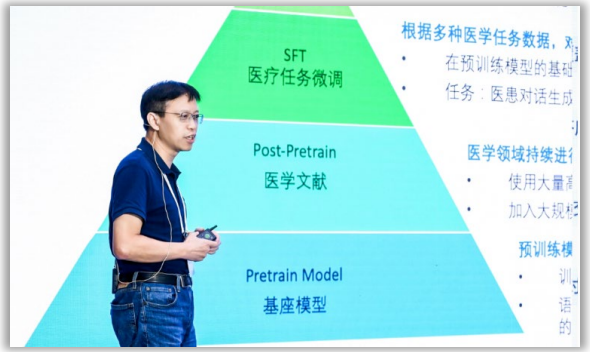
装备识别：新的装备类型随研制过程不断演进涌现，需及时识别

知识内容持续演进



知识内容：现实世界持续产生变化，随时间不短产生最新知识

任务规则偏好变动



规则偏好：医疗大模型的行为边界和安全偏好需要随政策变化

大模型应对动态环境

当下大模型虽然有很强的泛化能力，但在面临垂域场景、动态需求时依然存在“难泛化”的问题

在已知场景中，具身智能模型通过训练，能够在无遮挡的环境下抓起杯子

然而，在面临环境变化的情境下，具身智能模型的性能会受到干扰，无法具体理解“把杯子放到图片上”的具体含义

持续学习（Continual Learning）

对大模型泛化能力的进一步需求，实际上是要求大模型能够“**自主演进，持续进化**”

在**持续学习**过程中，模型接收按时间到达的数据流 $\mathcal{D}_1, \ldots, \mathcal{D}_T$，在每一阶段 t 仅基于当前可用信息更新参数

$$\theta_t = \mathcal{A}(\theta_{t-1}, \mathcal{D}_t),$$

使得模型在拟合新阶段数据后，仍然能够同时保持对历史分布泛化性能

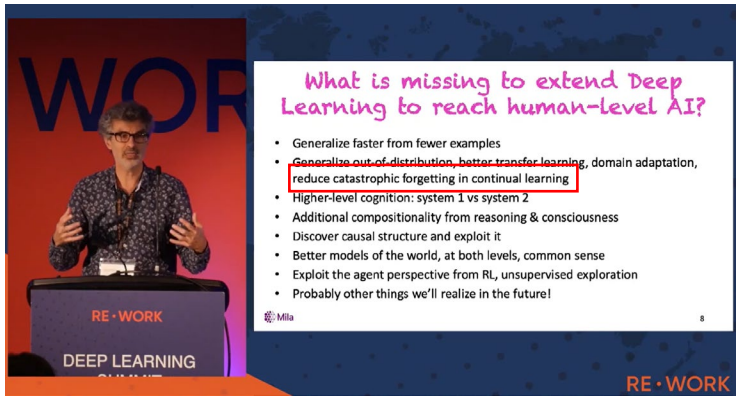
$$\min_{\mathcal{A}} \sum_{t=1}^{T} \sum_{s=1}^{t} \lambda_{t,s} \mathbb{E}_{(x,y)\sim P_s} \left[\ell\left(f_{\theta_t}(x), y \right) \right],$$

持续学习的目标是构造自主演进的智能模型

持续学习备受关注

持续学习研究已成为当下人工智能领域的核心要点之一，
是提升大模型自主演进能力的重要一环

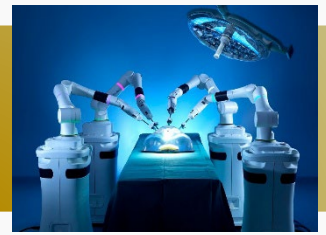
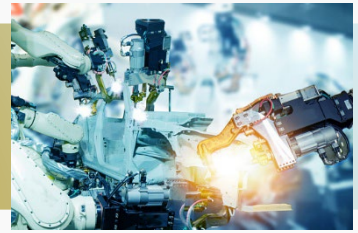
我国《“人工智能+制造”专项行动实施意见》指出：培育重点行业大模型，发展“云-边-端”模型体系，持续提升泛化能力



图灵奖得主Yoshua Bengio在报告中将“持续学习”能力作为“制约深度学习达到人类认知水平”的重要因素

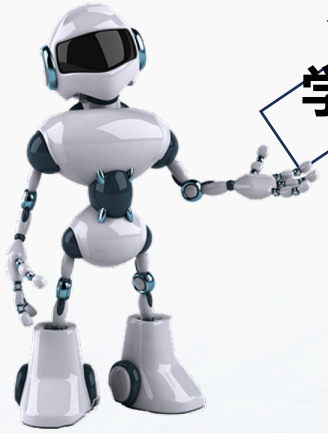
图灵奖得主Richard Sutton在报告中称“持续学习”能力是进入“经验时代”与“未来AI”的重要因素

持续学习的挑战

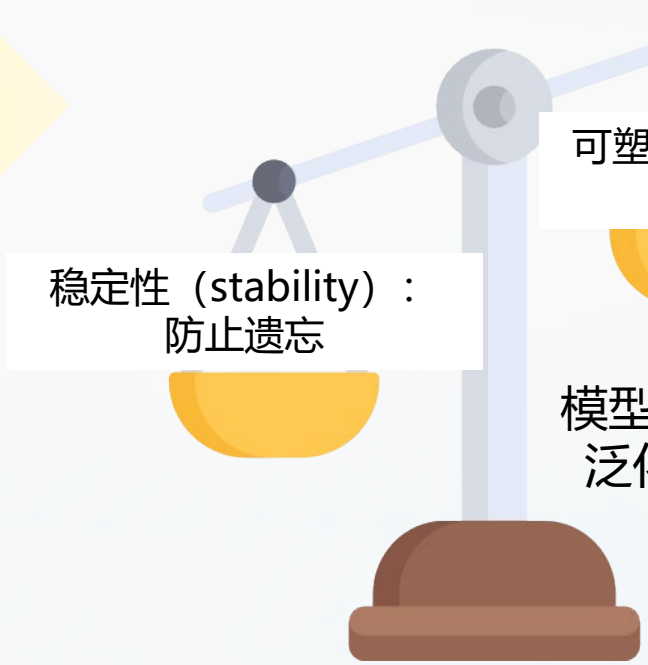

遗忘

学习



对历史分布的泛化能力

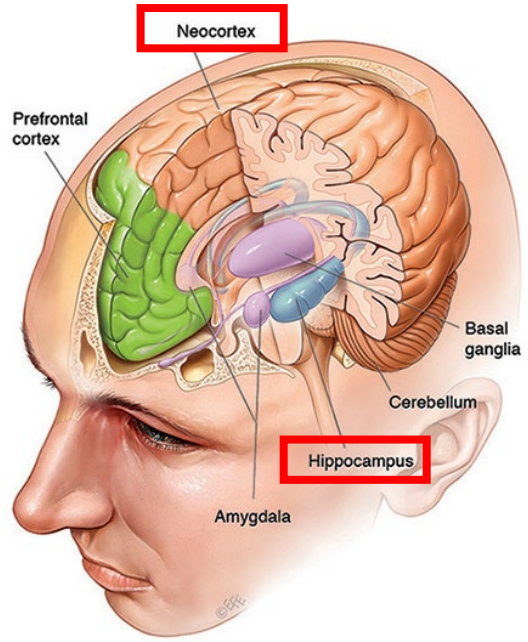
对新分布的泛化能力



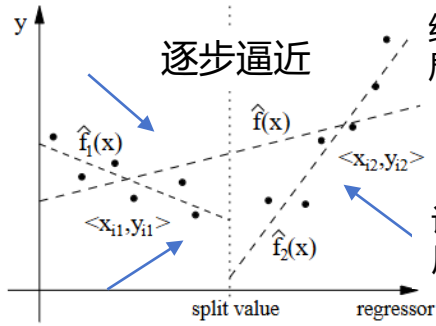
随着数据的持续积聚，在持续学习过程中，模型学习新知识的同时会导致对历史知识的“**灾难性遗忘**”，即“**学好了新的，但是忘记了旧的**”

持续学习的初期发展

持续学习最早可以追溯到1990年代神经科学的研究，
早期研究主要关注模型如何应对新增数据导致的分布变化或概念漂移

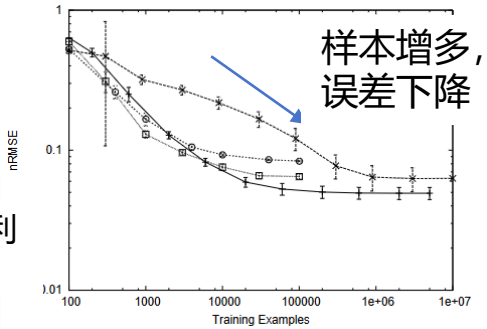


在神经科学中研究海马体和新皮质之间的互补记忆特性 [McClelland et al. 1995]



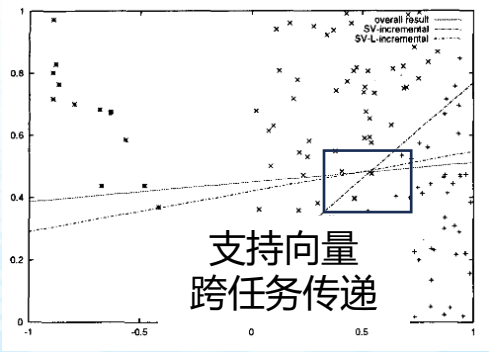
线性模型树通过动态划分输入空间、在局部拟合线性模型来逐步逼近复杂函数

训练样本增加后，整体误差下降，能够利用流式数据进行持续训练



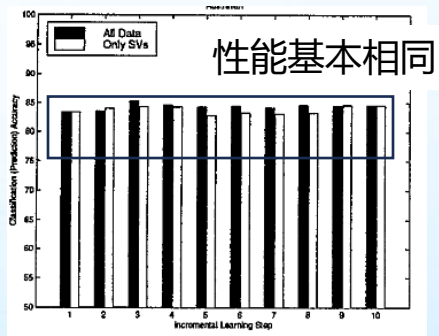
样本增多，误差下降

深度学习兴起前，持续学习研究关注线性模型，如树 [Duncan 2004]或SVM [Ahmed et al 1999]



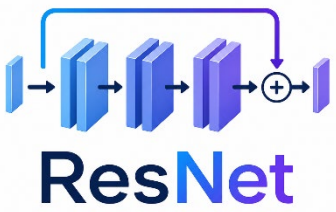
支持向量能够作为历史数据的紧凑表示，被传递到下一步继续训练

只保留支持向量也能较稳定地维持分类性能，说明其对历史决策边界具有较强的信息保真能力

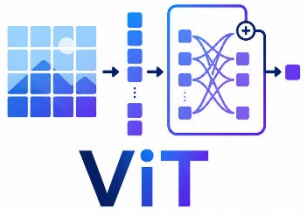


性能基本相同

模型结构在持续演进



克服新类学习时的旧类遗忘



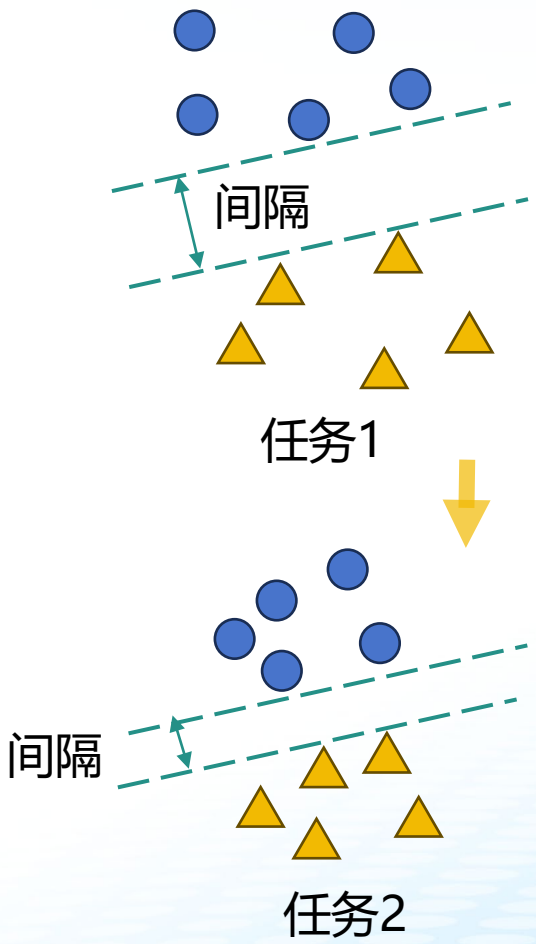
利用预训练表征轻量更新

引入跨模态信息辅助识别

扩展大模型能力的边界

进入深度学习时代以来，尽管“持续学习”的含义在不断进化，
解决这一问题的核心思路在不同阶段仍可以互相借鉴

模型遗忘与特征表示的分析



- 特征表示**间隔（margin）**的变化对模型能力的影响

单层间隔 → 全层间隔：全层间隔是让模型预测发生改变所需的最小扰动

$$m_F(\boldsymbol{x}_i, y_i) := \min_{\boldsymbol{\delta}_1,\ldots,\boldsymbol{\delta}_L} \sqrt{\sum_{l=1}^{L} \|\boldsymbol{\delta}_l\|_2^2}$$

$$\text{s.t. } \arg\max F(\boldsymbol{x}_i, \boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_L) \neq y_i$$

泛化界：所有在训练集达到零损失的模型能以$1-\delta$概率满足下式：（即期望损失有关于间隔的上界）

$$\mathbb{E}_P[\ell_{0-1}(F(\boldsymbol{x}), y)] \lesssim \frac{\sum_i C_i}{\sqrt{n}} \sqrt{\mathbb{E}_{(\boldsymbol{x},y)\sim P_n}\left[\frac{1}{m_F(\boldsymbol{x}, y)^2}\right] \log^2 n + \zeta}$$

提升模型的抗扰动能力，就可以抵抗遗忘

间隔与表示空间的变化

全间隔 → 每一层的间隔：

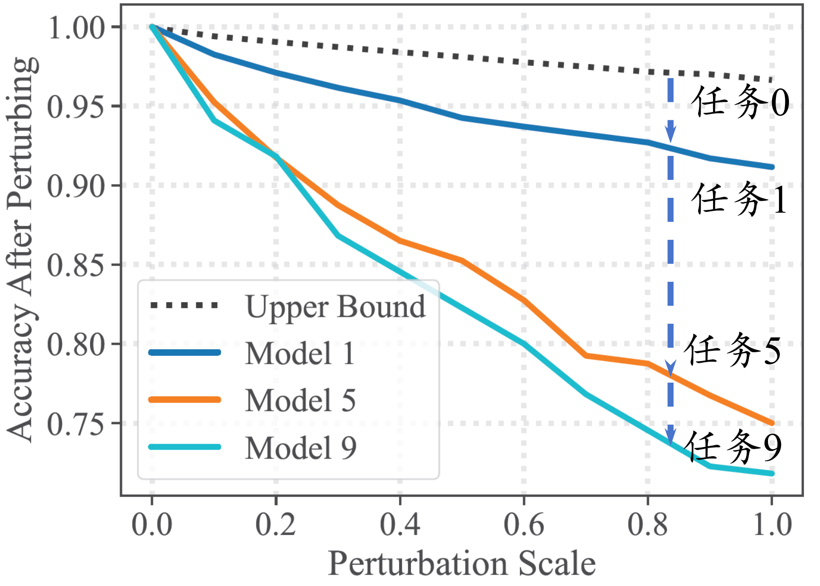
$$m_F(\boldsymbol{x}_i, y_i) \le \widetilde{m}_{F,l}(\boldsymbol{x}_i, y_i) := \min_{\boldsymbol{\delta}_l} \|\delta_l\|_2,$$
$$\text{s.t. } \operatorname{argmax} F(\boldsymbol{x}_i, \boldsymbol{0}, \dots, \delta_l, \dots, \boldsymbol{0}) \neq y_i.$$

最小扰动的方向 → 模型关于特征的梯度：

$$\widetilde{m}_{F,l}(\boldsymbol{x}_i, y_i) = \min_{\delta_l} \|\delta_l\|_2$$
$$\approx \min_{\alpha_{s,l}} \left\| \alpha_{i,l} \nabla_{\boldsymbol{z}} \ell \left(F_l(\boldsymbol{z}_{i,l}) \right) \right\|_2$$
$$\text{s.t. } \arg\max F(x_i, 0, \dots, \delta_l, \dots, 0) \neq y_i \, ,$$

$$\widetilde{m}_{F,l}(\boldsymbol{x}_i, y_i) \approx \alpha^*_{i,l} \left\| \nabla_{\boldsymbol{z}} \ell \left(F_l(\boldsymbol{z}_{i,l}) \right) \right\|_2$$

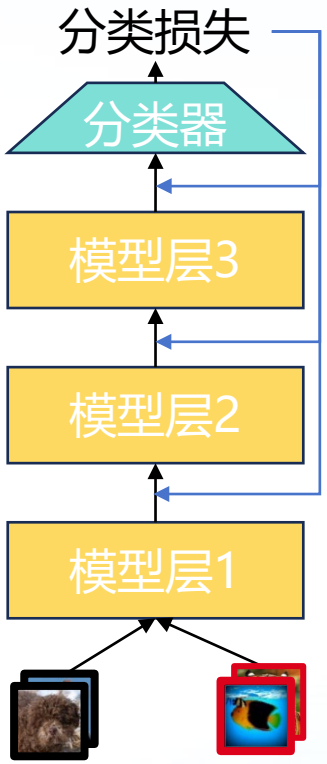
在回放样本上考察扰动强度与预测改变的样本数的关系，观测其在持续学习过程中间隔的改变趋势



- 随着模型持续学习，回放样本的抗扰动能力越来越差
- 回放样本的平均间隔在不同深度都变得越来越小

Zheng, Zhou, Ye, Zhan. *Multi-layer Rehearsal Feature Augmentation for Class-Incremental Learning*. ICML 2024.

多层回放特征增强



$$\hat{z}_{i,l} = z_{i,l} + \lambda \nabla_{z} \ell(F_l(z_{i,l})),$$

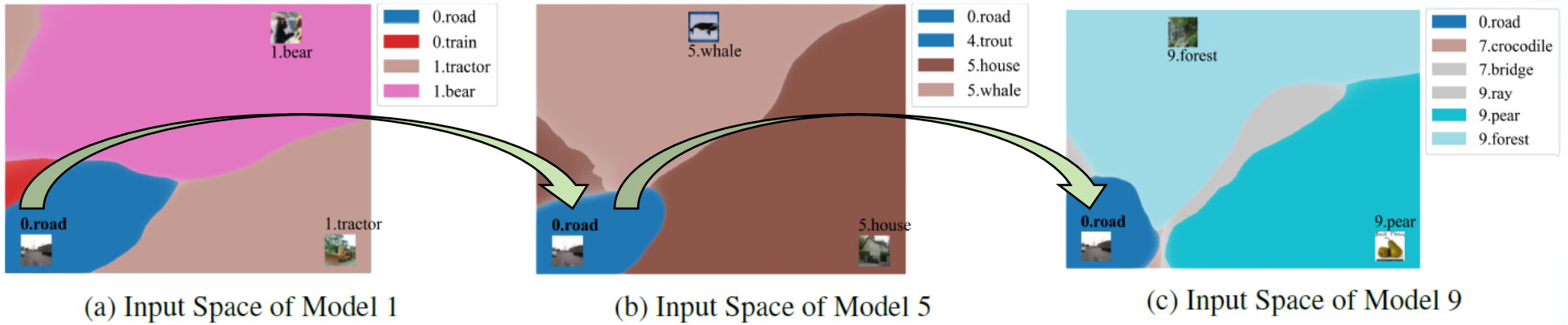
Algorithm 1 Multi-layer Rehearsal Feature Augmentation

Input: batch $\mathcal{B}$, current model $F_t(x)$
for $x_i, y_i \in \mathcal{B}$ **do**
  if $x_i, y_i \in \mathcal{M}$ **then** 仅对回放样本进行特征增强
    Sample $l \sim \mathcal{U}\{1, L\}$, $\hat{\beta} \sim \mathcal{U}(0, \beta)$
    $\mathcal{L}_{\text{cls},i} = \text{AugmentedForward}(F_t(\boldsymbol{x}), \boldsymbol{x}_i, y_i, l, \hat{\beta})$
  else
    $\mathcal{L}_{\text{cls},i} = \ell(F_t(\boldsymbol{x}_i), y_i)$
  end if
end for
$\mathcal{L}_{\text{cls}} = \frac{1}{|\mathcal{B}|} \sum_i \mathcal{L}_{\text{cls},i}$
Output: $\mathcal{L}_{\text{cls}}$
function AugmentedForward($F(\boldsymbol{x})$, $\boldsymbol{x}$, y, l, $\hat{\beta}$)
  $\boldsymbol{z}_l = f_{l-1} \circ \cdots \circ f_1(\boldsymbol{x})$ 沿损失上升方向制造更靠近边界的样本
  $\hat{\boldsymbol{z}}_l = \boldsymbol{z}_l + \hat{\beta} \|\boldsymbol{z}_l\|_2 \nabla_{\boldsymbol{z}} \ell(F_l(\boldsymbol{z}_l), y)$ {Eq. 6}
  return $\ell(F_l(\hat{\boldsymbol{z}}_l), y)$ {Eq. 7} 返回增强后的分类损失
end function

通过对回放样本的特征进行定向增强，训练模型抵抗扰动，从而扩大回放样本的间隔，让模型拥有更强的泛化能力和鲁棒性

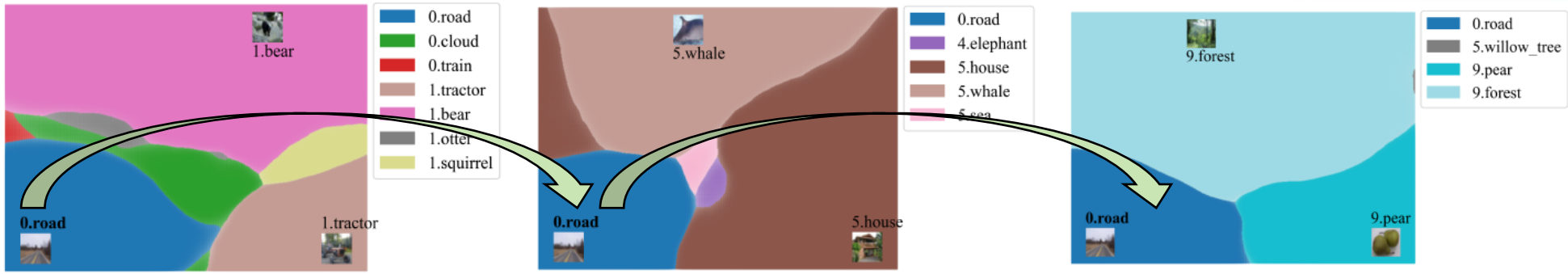
实验结果

使用前



(a) Input Space of Model 1　(b) Input Space of Model 5　(c) Input Space of Model 9

使用后



(a) Input Space of Model 1　(b) Input Space of Model 5　(c) Input Space of Model 9

使用所提出的方法，回放样本的判别边界得到了显著的扩张，模型间隔增大

持续学习的模型表示

在持续学习过程中，模型的表示空间结构难以维持，造成模型能力受损

稳定性（stability）
防止遗忘

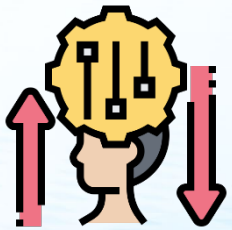
可塑性（plasticity）
快速适配

兼容性（compatibility）
新旧能力协同工作

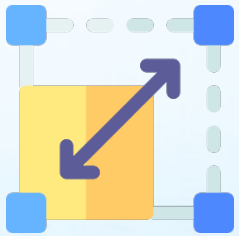
增强模型兼容性

特征表示预留

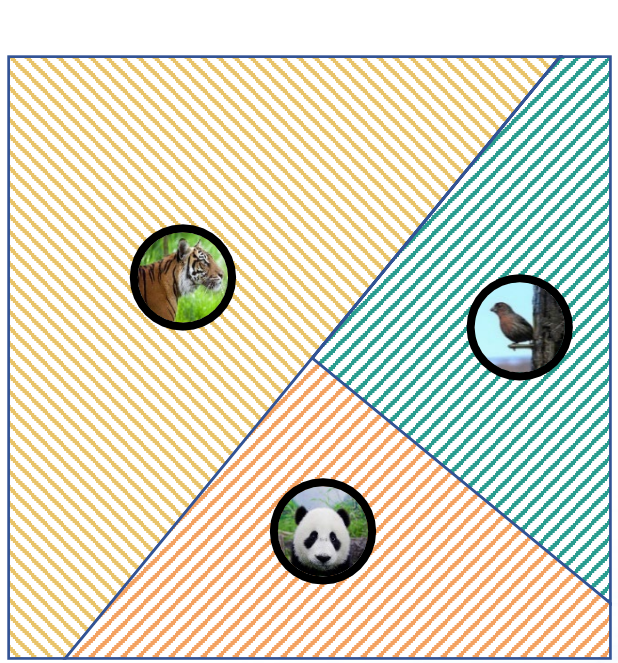
特征表示矫正



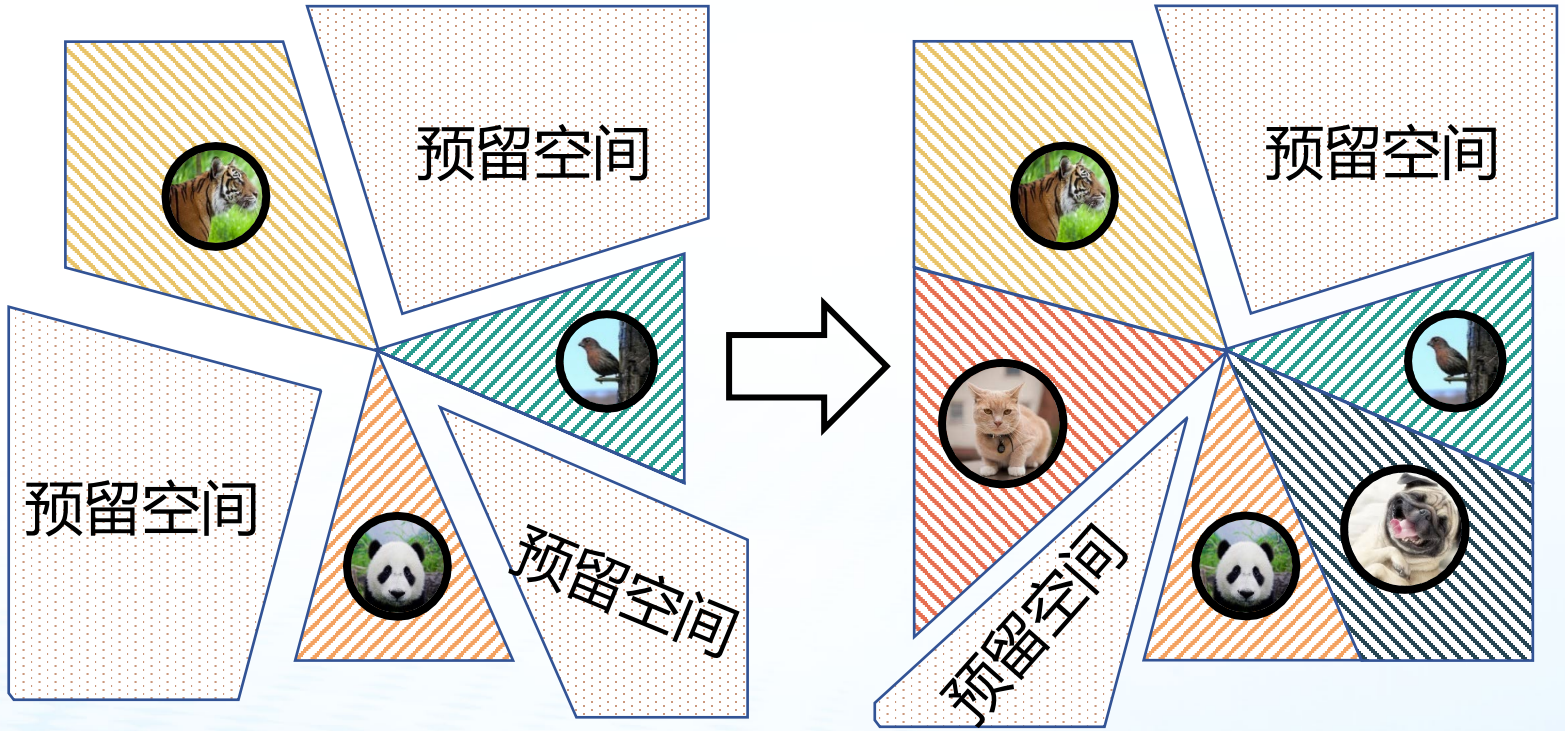
特征表示扩张



为新数据预留特征空间



传统训练模式



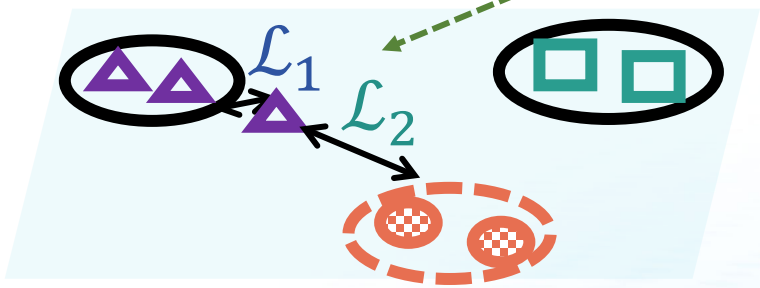
向前兼容训练模式

Zhou et al., *Forward compatible few-shot class-incremental learning*. CVPR 2022

向前兼容用于增量学习

核心思想： 为新类预留表示空间，缓解模型更新时的新旧干扰

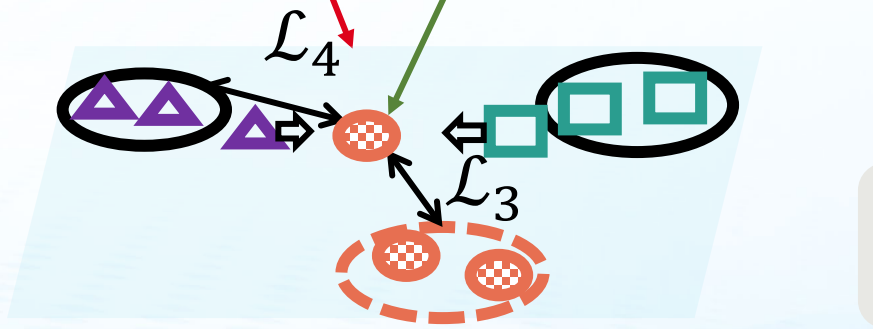
$$\mathcal{L}_v(\mathbf{x}, y) = \underbrace{\ell(f_v(\mathbf{x}), y)}_{\mathcal{L}_1} + \gamma \underbrace{\ell\left(\mathrm{Mask}(f_v(\mathbf{x}), y), \hat{y}\right)}_{\mathcal{L}_2}$$



已知类别样本距对应类中心最近、虚拟类中心次近

$$\mathcal{L}_f(z) = \underbrace{\ell(f_v(z), \hat{y})}_{\mathcal{L}_3} + \gamma \underbrace{\ell\left(\mathrm{Mask}(f_v(z), \hat{y}), \hat{\hat{y}}\right)}_{\mathcal{L}_4},$$

$$z = g\left[\lambda h(\mathbf{x}_i) + (1-\lambda) h(\mathbf{x}_j)\right],$$



表示空间

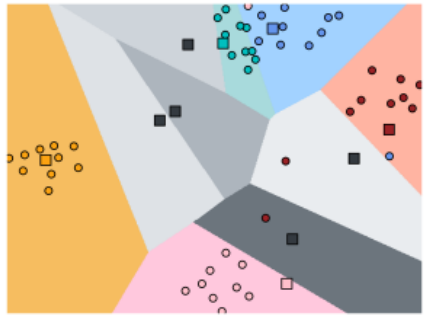
已知类

虚拟类

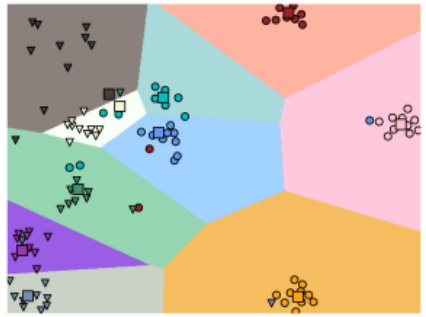
虚拟类样本距对应的虚拟类中心最近、最近的已知类中心次近

特征表示预留的实验验证

Method	Accuracy in each session (%) ↑											PD ↓	Δ PD
	0	1	2	3	4	5	6	7	8	9	10		
Finetune	68.68	43.70	25.05	17.72	18.08	16.95	15.10	10.06	8.93	8.93	8.47	60.21	**+41.25**
Pre-Allocated RPC† [32]	68.47	51.00	45.42	40.76	35.90	33.18	27.23	24.24	21.18	17.34	16.20	52.27	**+33.31**
iCaRL [33]	68.68	52.65	48.61	44.16	36.62	29.52	27.83	26.26	24.01	23.89	21.16	47.52	**+28.56**
EEIL [8]	68.68	53.63	47.91	44.20	36.30	27.46	25.93	24.70	23.95	24.13	22.11	46.57	**+27.61**
Rebalancing [21]	68.68	57.12	44.21	28.78	26.71	25.66	24.62	21.52	20.12	20.06	19.87	48.81	**+29.85**
TOPIC [41]	68.68	62.49	54.81	49.99	45.25	41.40	38.35	35.36	32.22	28.31	26.26	42.40	**+23.44**
SPPR [67]	68.68	61.85	57.43	52.68	50.19	46.88	44.65	43.07	40.17	39.63	37.33	31.35	**+12.39**
Decoupled-NegCosine† [26]	74.96	70.57	66.62	61.32	60.09	56.06	55.03	52.78	51.50	50.08	48.47	26.49	**+7.53**
Decoupled-Cosine [45]	75.52	70.95	66.46	61.20	60.86	56.88	55.40	53.49	51.94	50.93	49.31	26.21	**+7.25**
Decoupled-DeepEMD [57]	75.35	70.69	66.68	62.34	59.76	56.54	54.61	52.52	50.73	49.20	47.60	27.75	**+8.79**
CEC [58]	75.85	71.94	68.50	63.50	62.43	58.27	57.73	55.81	54.83	53.52	52.28	23.57	**+4.61**
FACT	**75.90**	**73.23**	**70.84**	**66.13**	**65.56**	**62.15**	**61.74**	**59.83**	**58.41**	**57.89**	**56.94**	**18.96**	



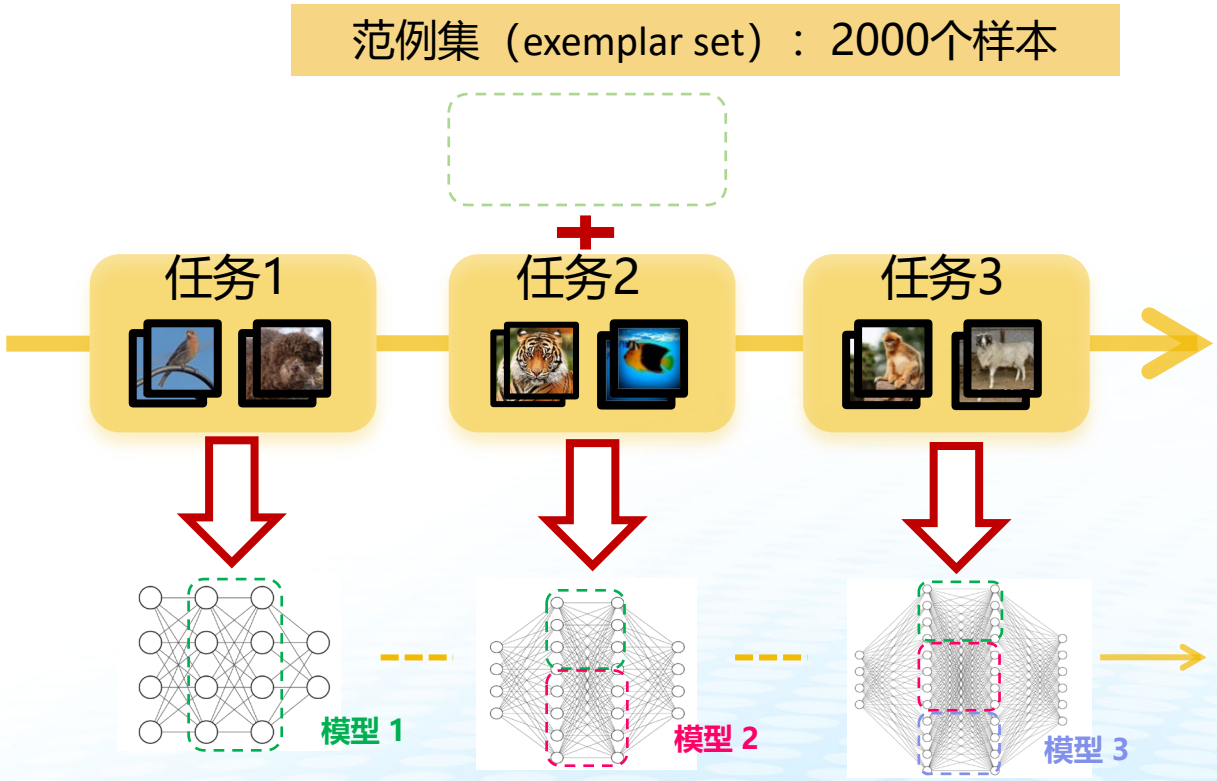
(a) Base session, 5 old classes & 5 virtual prototypes.



(b) Incremental session, 5 old classes & 5 new classes.

- CUB200数据集以100个类别作为base task, 其余类别分10阶段到来的10-way-5-shot设定中, 性能超越SOTA算法约4.5%
- **预留**的新类空间（深色）较好地为模型后续更新提供帮助.

表示空间扩张实现模型兼容



- 模型保留、特征拼接 [Yan et al. CVPR'21] [Wang et al. ECCV'22] [Wang et al. NeurIPS'22]

$$f(x) = W^{\top}\text{Concat}[\phi_1(x), \phi_2(x), \ldots \phi_b(x)]$$

保存、冻结旧模型，仅训练新模型

基于范例集进行多个模型之间的预测矫正

Wang, Zhou, Ye, Zhan. *FOSTER: Feature Boosting and Compression for Class-Incremental Learning*. ECCV 2022.

资源受限：表示扩张的难题

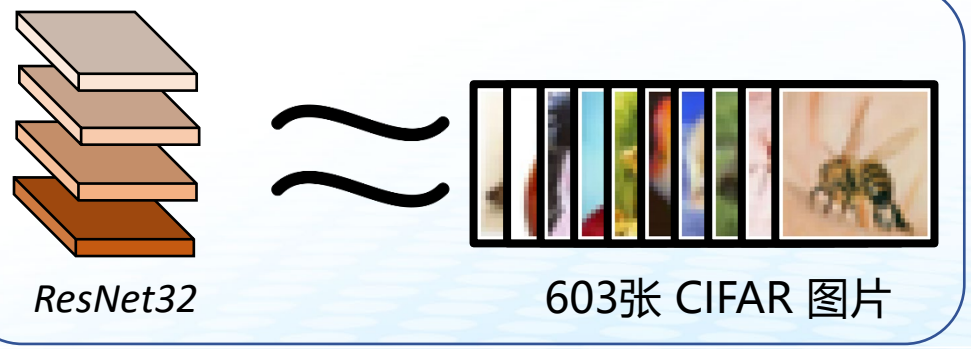
不公平对比

基于回放的方法	2000 范例	模型	
基于模型的方法	2000 范例	模型 * N	

当前的持续学习对比方式

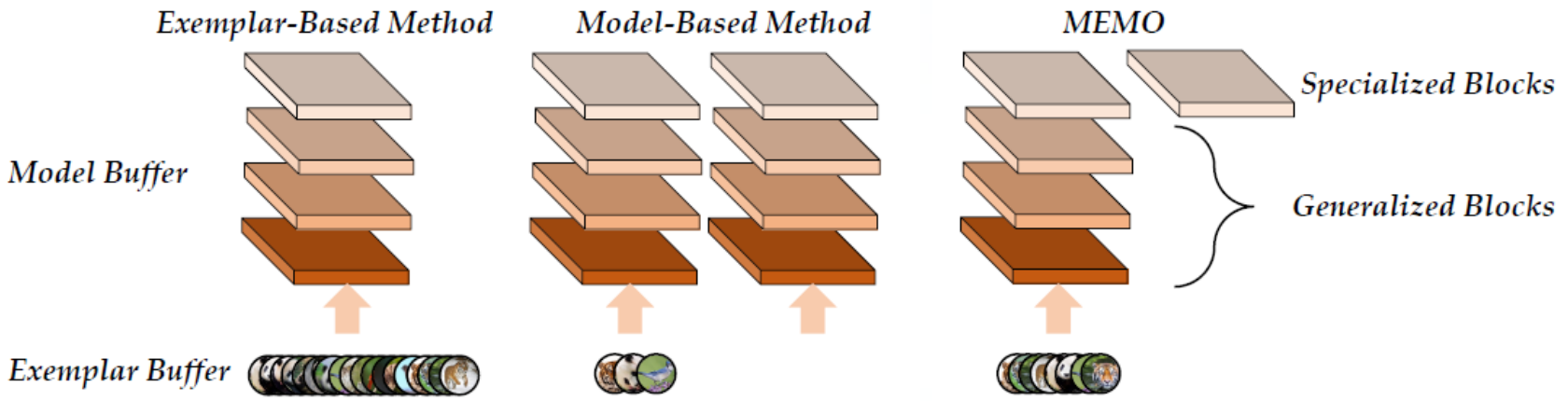
基于回放的方法	2000 范例	Δ exemplars	模型
基于模型的方法	2000 范例	模型 * N	

公平的对比方式

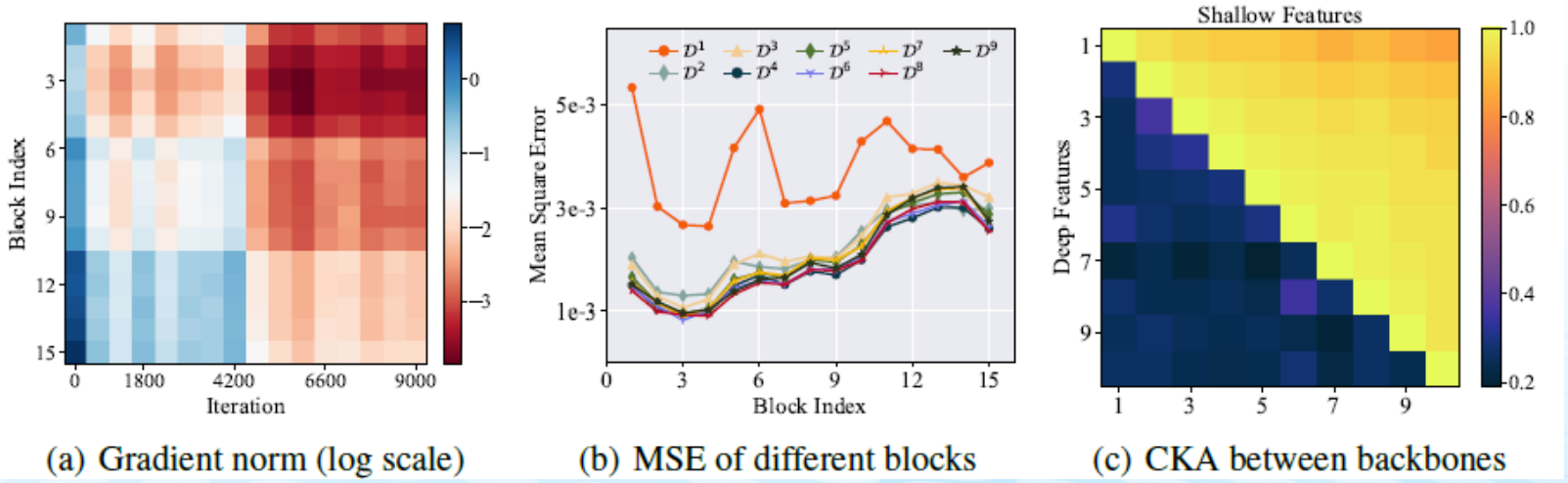


给定**相同的总存储空间**，如何合理分配用于存储数据与模型的空间使其更好兼容？

局部模型的兼容



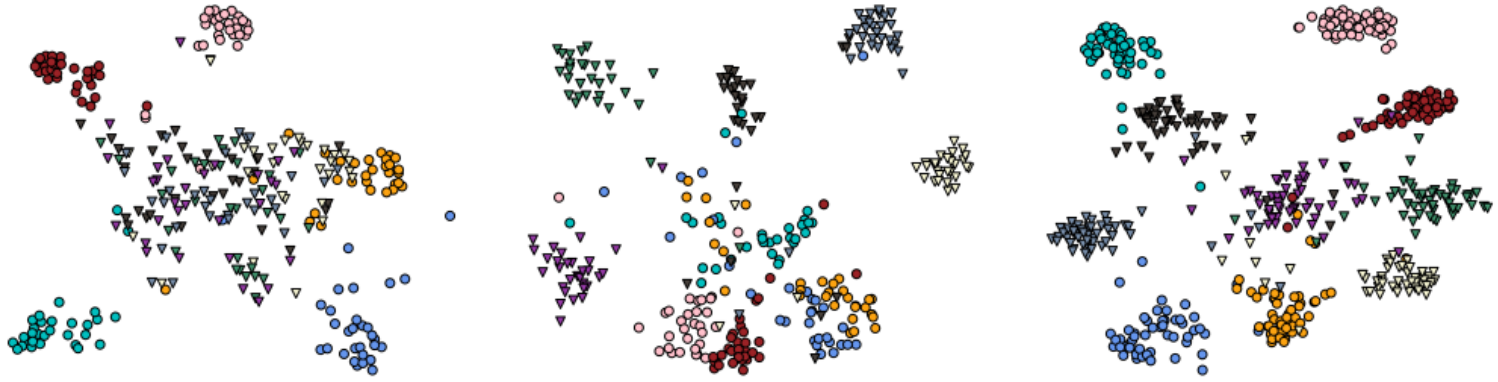
- 给定模型存储空间,如何合理分配用于存储数据与模型的空间使其更好复用表示



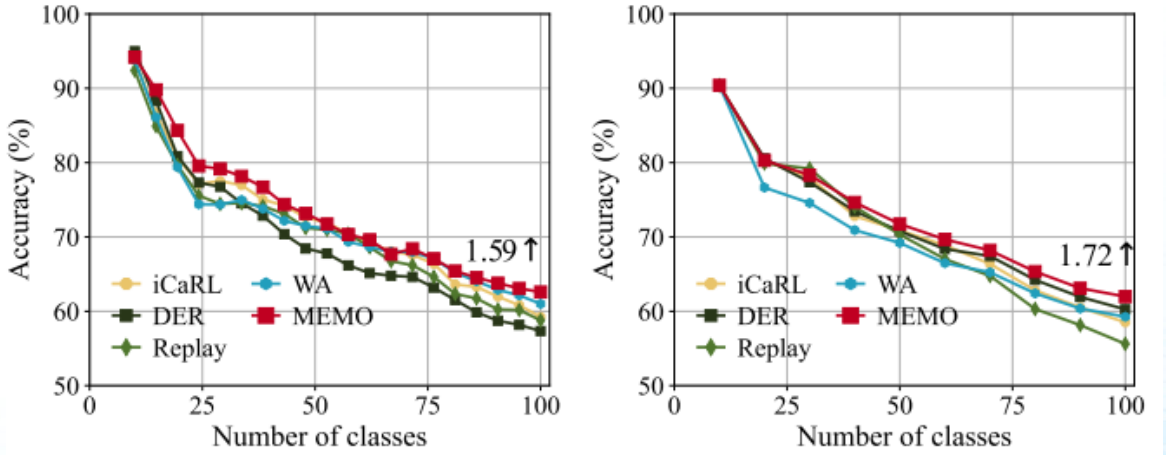
(a) Gradient norm (log scale) (b) MSE of different blocks (c) CKA between backbones

Zhou, et al., *A Model or 603 Exemplars: Towards Memory-Efficient Class-Incremental Learning*. ICLR 2023.

实验验证



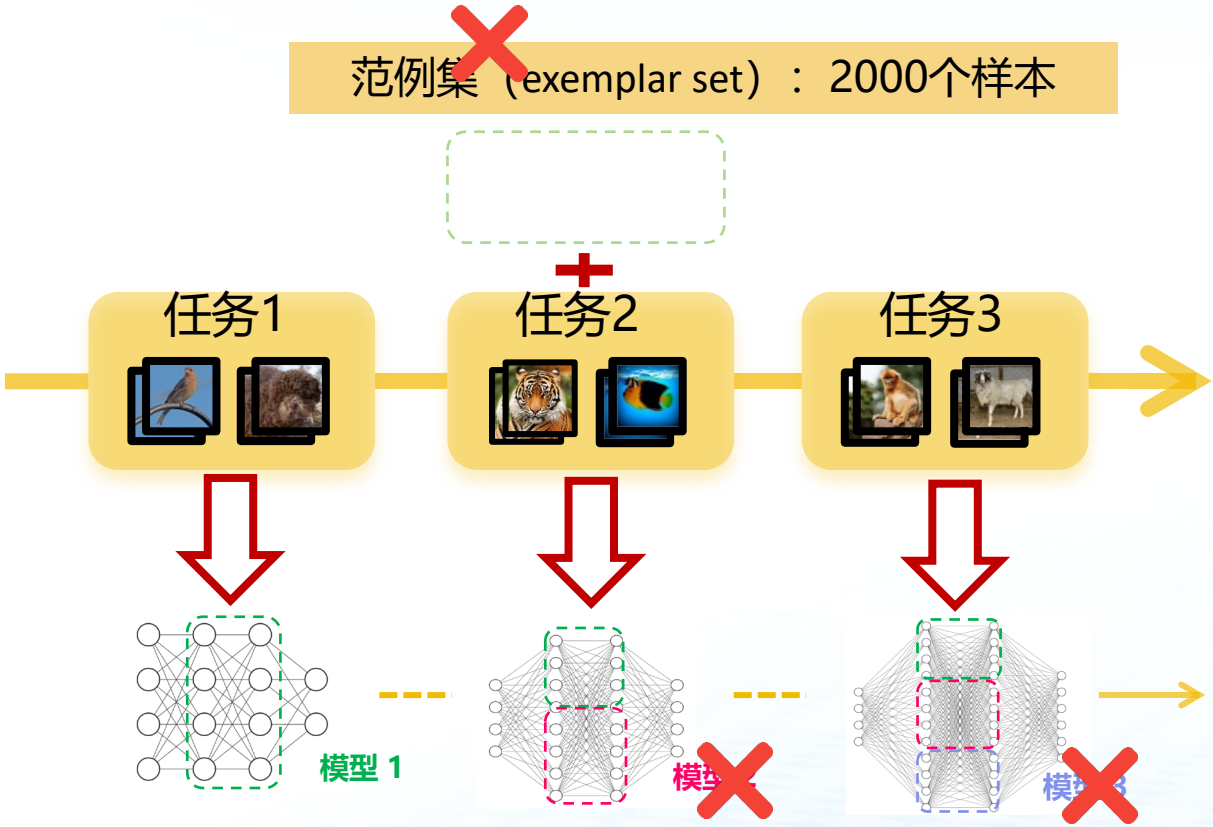
(a) $\phi_{s1}(\phi_g(\mathbf{x}))$ (b) $\phi_{s2}(\phi_g(\mathbf{x}))$ (c) $[\phi_{s1}(\phi_g(\mathbf{x})), \phi_{s2}(\phi_g(\mathbf{x}))]$



(a) CIFAR100 Base0 Inc5 (b) CIFAR100 Base0 Inc10

- 在共用浅层特征的同时，深层特征学得了**任务相关**的特征表示；
- 将不同任务的深层特征合并时，能够获得**适应所有类别**的特征表示；
- 当所有算法存储开销相同时，所提出算法实现了免费的性能提升.

不基于回放样本的表示空间扩张



范例集的使用也增加了算法对资源的消耗，若不使用范例集，则算法无法进行预测校准

若以预训练模型作为初始化，由于预训练模型使用复杂的网络结构，保存**多个主干网络**将显著消耗大量存储空间

怎样在仅使用**固定主干网络**的同时，**不使用回放样本**扩增表示?

视觉预训练模型带来的新观察

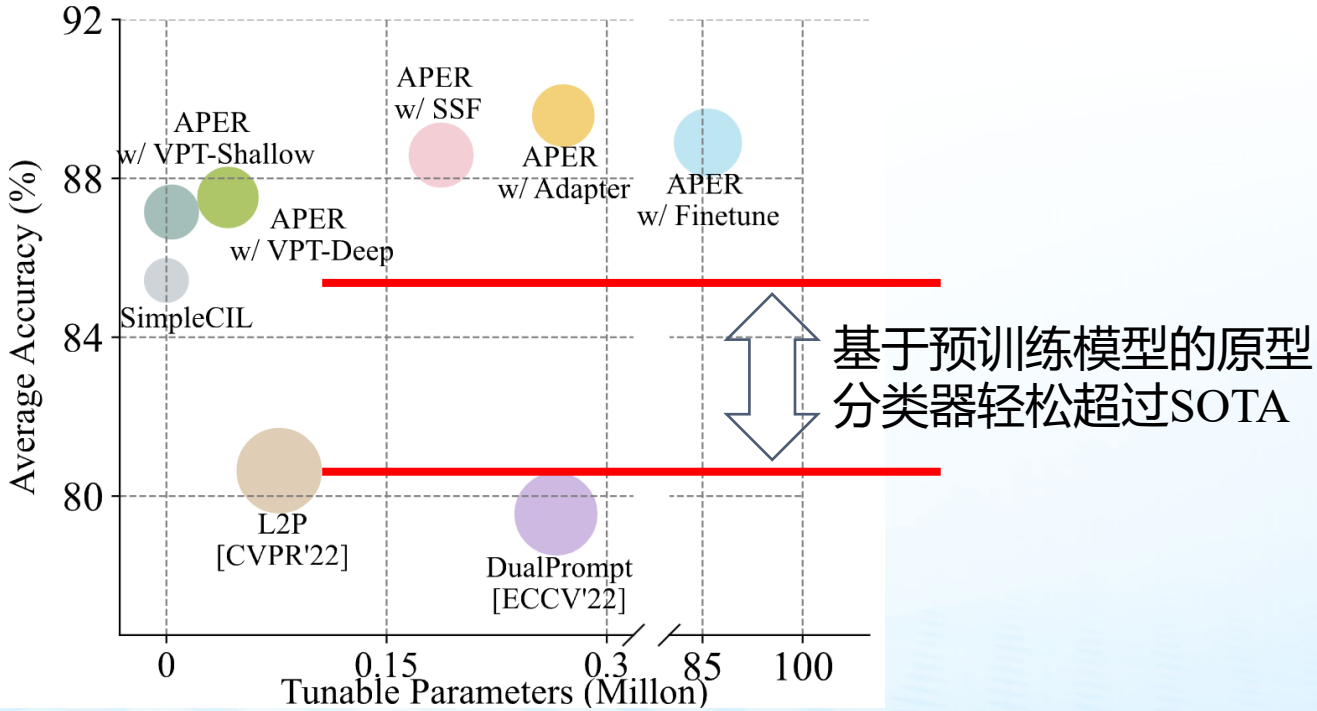
- 增量学习的目的是获得适配所有任务的特征表示，并抵抗学习过程中的特征遗忘
- 相比于传统的从零开始训练设定，预训练模型天然具有**可泛化**的特征表示

预训练模型能为特征表示复用带来何种便捷？

“从零训练”

“基于预训练模型”

$$\boldsymbol{p}_i = \frac{1}{K}\sum_{j=1}^{|\mathcal{D}^b|}\mathbb{I}(y_j = i)\phi(\mathbf{x}_j)$$



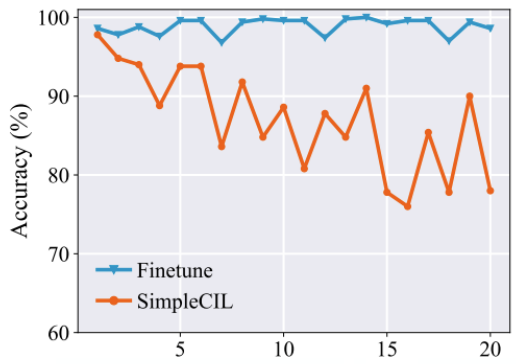
基于预训练模型是否需要增量学习？

Zhou, et al., *Revisiting Class-Incremental Learning with Pre-Trained Models: Generalizability and Adaptivity are All You Need*. IJCV 2025.

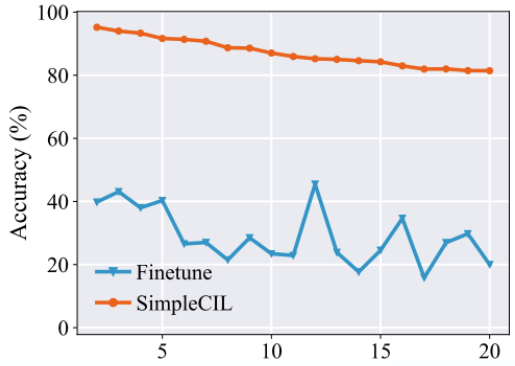
基于视觉预训练模型的解决思路

（预训练模型+原型分类器）是否足以处理任何下游任务的增量学习?

不能！利用下游任务调整可进一步提升模型的能力

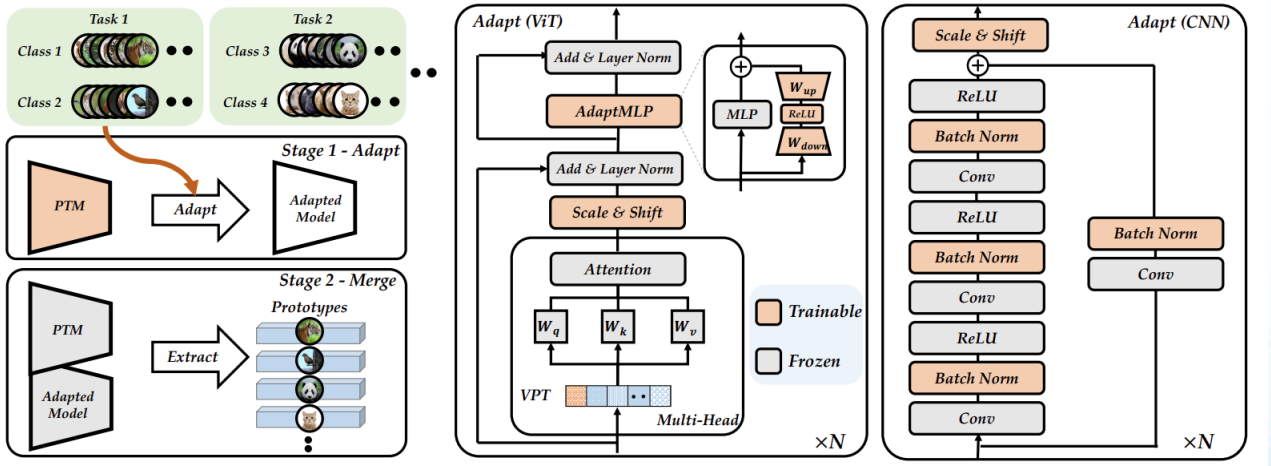


新类性能



旧类性能

如何结合预训练模型与适配后模型的优势?

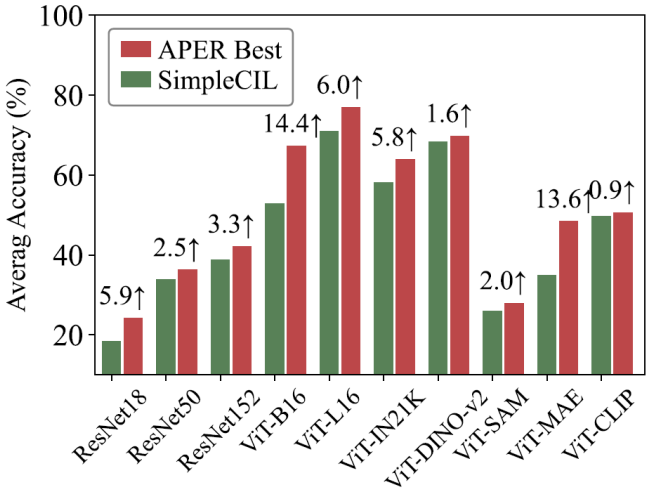


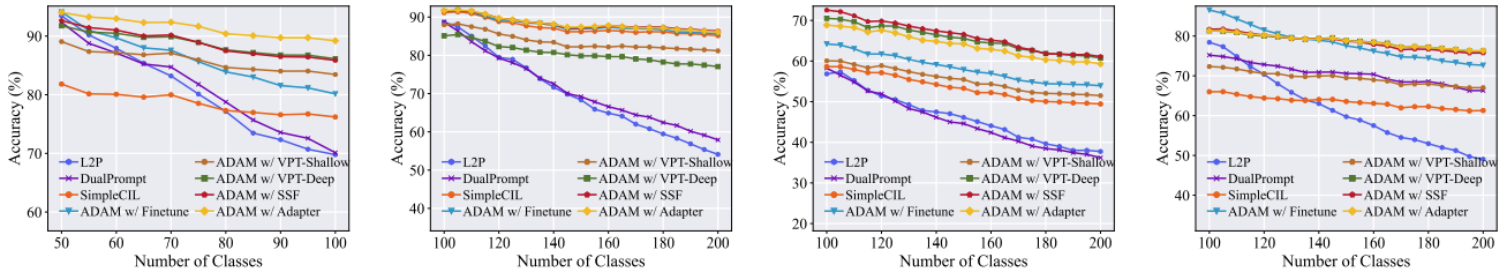
第一阶段：模型适配（Adapt）与拼接（Merge）
后续阶段：原型分类器

Adapt and Merge!

实验验证

Method	CIFAR B0 Inc5 $\bar{A}$	CIFAR B0 Inc5 A_B	CUB B0 Inc10 $\bar{A}$	CUB B0 Inc10 A_B	IN-R B0 Inc5 $\bar{A}$	IN-R B0 Inc5 A_B	IN-A B0 Inc10 $\bar{A}$	IN-A B0 Inc10 A_B	ObjNet B0 Inc10 $\bar{A}$	ObjNet B0 Inc10 A_B	OmniBench B0 Inc30 $\bar{A}$	OmniBench B0 Inc30 A_B	VTAB B0 Inc10 $\bar{A}$	VTAB B0 Inc10 A_B
Finetune	38.90	20.17	26.08	13.96	21.61	10.79	21.60	10.96	19.14	8.73	23.61	10.57	34.95	21.25
Finetune Adapter [10]	60.51	49.32	66.84	52.99	47.59	40.28	43.05	37.66	50.22	35.95	62.32	50.53	48.91	45.12
LwF [54]	46.29	41.07	48.97	32.03	39.93	26.47	35.39	23.83	33.01	20.65	47.14	33.95	40.48	27.54
SDC [111]	68.21	63.05	70.62	66.37	52.17	49.20	26.65	23.57	39.04	29.06	60.94	50.28	45.06	22.50
L2P [101]	85.94	79.93	67.05	56.25	66.53	59.22	47.16	38.48	63.78	52.19	73.36	64.69	77.11	77.10
DualPrompt [100]	87.87	81.15	77.47	66.54	63.31	55.22	52.56	42.68	59.27	49.33	73.92	65.52	83.36	81.23
CODA-Prompt [82]	89.11	81.96	84.00	73.37	64.42	55.08	48.51	36.47	66.07	53.29	77.03	68.09	83.90	83.02
CPP [55]	85.21	78.64	86.60	85.27	64.33	60.74	53.70	40.70	60.44	49.92	71.52	73.26	85.92	84.30
LAE [24]	**92.47**	**87.62**	83.13	77.78	69.05	63.17	57.19	46.41	62.28	50.57	73.80	70.63	86.14	84.39
SimpleCIL	87.57	81.26	92.20	**86.73**	62.58	54.55	60.50	49.44	65.45	53.59	79.34	73.15	85.99	84.38
APER w/ Finetune	87.67	81.27	91.82	86.39	70.51	62.42	61.57	50.76	61.41	48.34	73.02	65.03	**87.47**	80.44
APER w/ VPT-Shallow	90.43	84.57	92.02	86.51	66.63	58.32	57.72	46.15	64.54	52.53	79.63	73.68	87.15	**85.36**
APER w/ VPT-Deep	88.46	82.17	91.02	84.99	68.79	60.48	60.59	48.72	67.83	54.65	**81.05**	**74.47**	86.59	83.06
APER w/ SSF	87.78	81.98	91.72	86.13	68.94	60.60	**62.81**	**51.48**	**69.15**	**56.64**	80.53	74.00	85.66	81.92
APER w/ Adapter	90.65	85.15	**92.21**	**86.73**	**72.35**	**64.33**	60.53	49.57	67.18	55.24	80.75	74.37	85.95	84.35

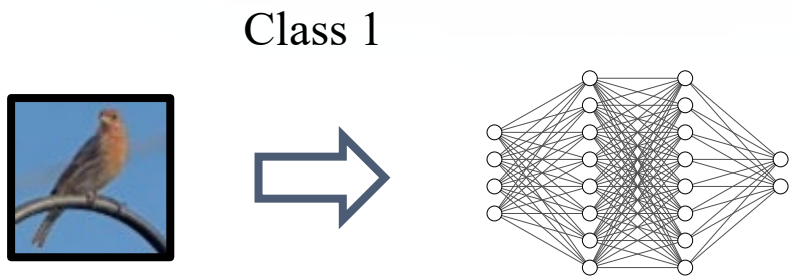


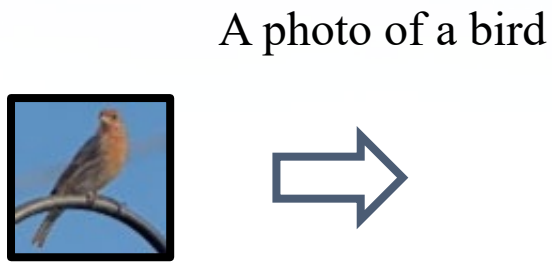


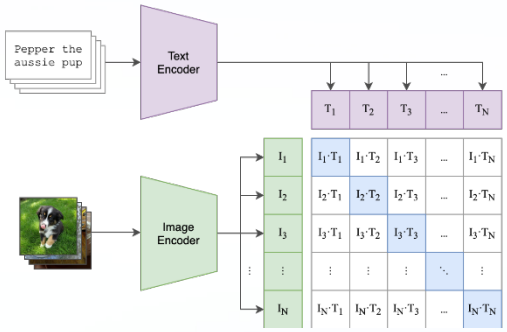
(a) CIFAR B50 Inc5 (b) CUB B100 Inc5 (c) ImageNet-A B100 Inc5 (d) ImageNet-R B100 Inc5

- 在7个数据集不同设定下的增量学习场景中，均对比当前SOTA有稳定性能提升
- 在不同骨干网络上，可以实现稳定性能提升

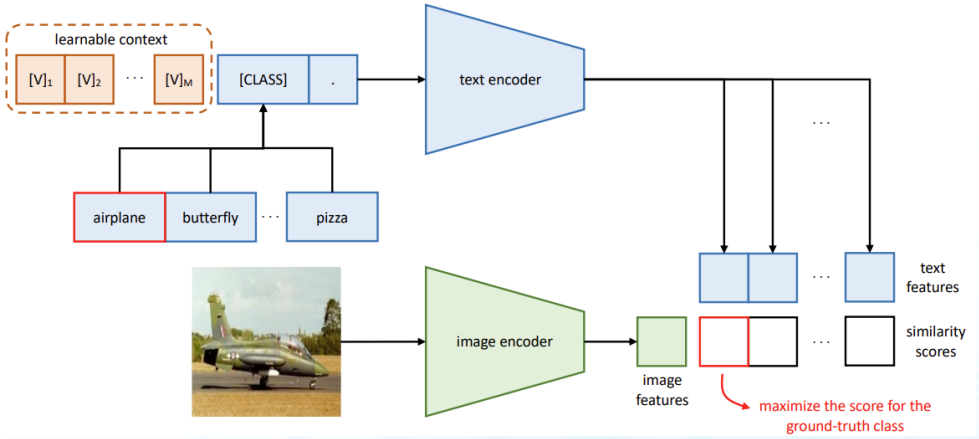
基于视觉-语言模型的表示扩张

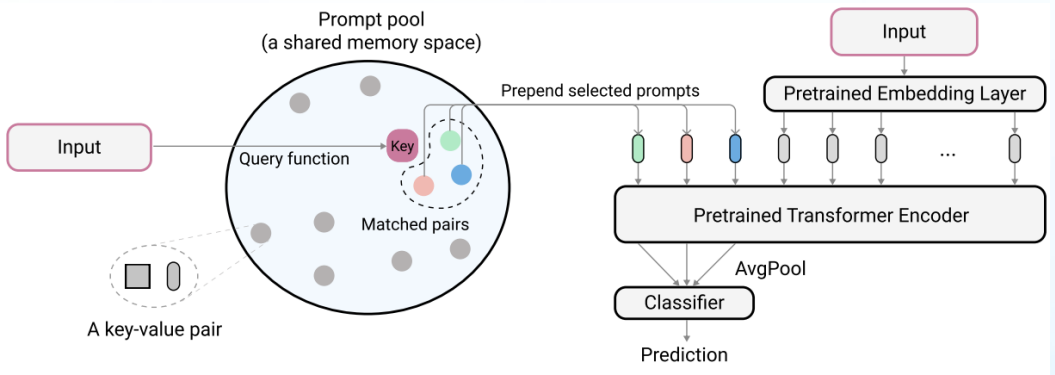






语义信息可以帮助模型更好地进行增量学习



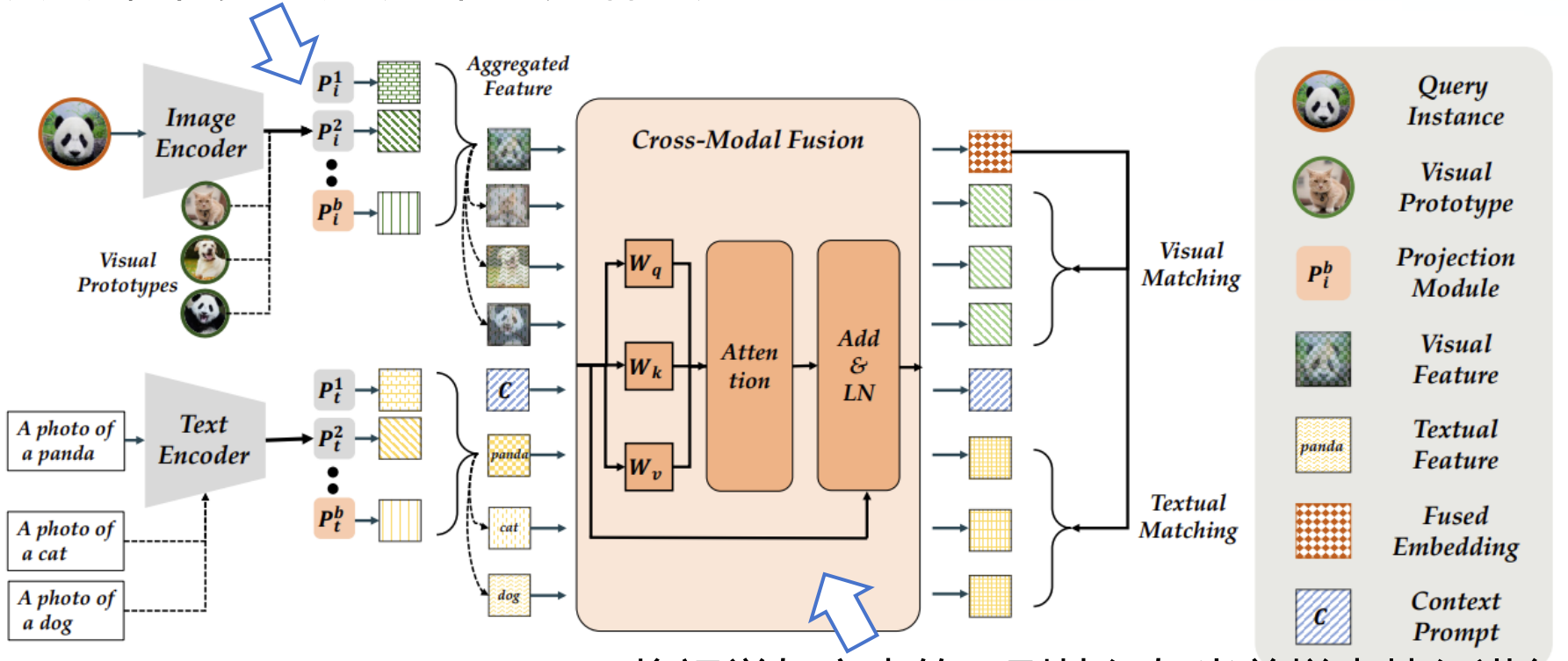


提示学习 [Zhou et al. IJCV'22] 仅关注模型适应过程，无法解决灾难性遗忘

基于预训练模型的增量学习方法[Wang et al. CVPR'22] 无法有效利用多模态信息

基于视觉-语言模型的表示扩张

在冻结预训练模型的基础上，为视觉与文本编码器学习
可扩展的投影模块，将多组投影之和作为增量表示



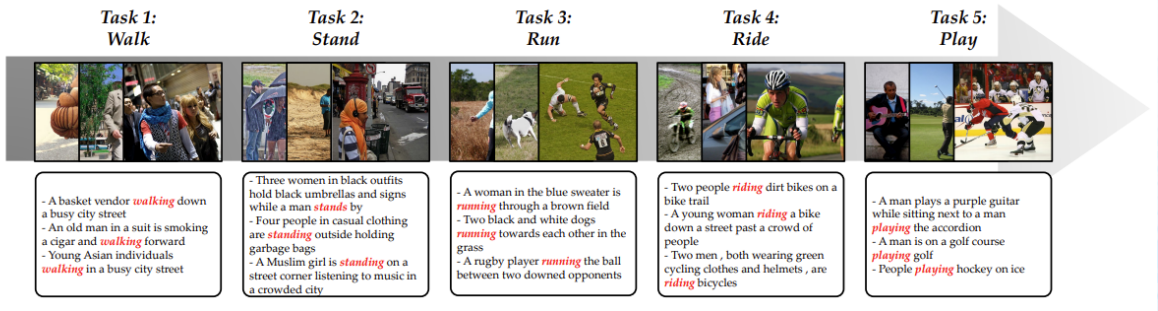
将视觉与文本的原型特征与当前样本特征进行跨模态
融合，对协同调整后的视觉-文本信息进行匹配训练

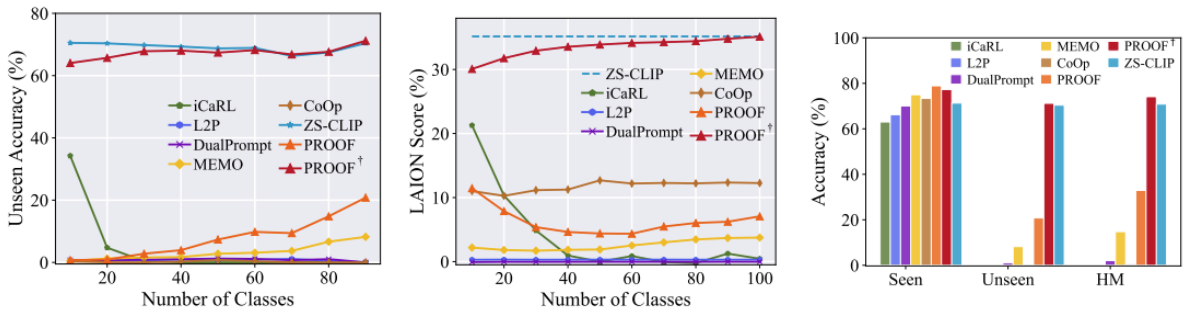
Zhou, et al., *Learning without Forgetting for Vision-Language Models*. TPAMI 2025

基于视觉-语言模型的表示扩张

Method	Exemplar	ImageNet-R B0 Inc20 $\bar{A}$	ImageNet-R B0 Inc20 A_B	ImageNet-R B100 Inc20 $\bar{A}$	ImageNet-R B100 Inc20 A_B	CUB B0 Inc20 $\bar{A}$	CUB B0 Inc20 A_B	CUB B100 Inc20 $\bar{A}$	CUB B100 Inc20 A_B	UCF B0 Inc10 $\bar{A}$	UCF B0 Inc10 A_B	UCF B50 Inc10 $\bar{A}$	UCF B50 Inc10 A_B
Finetune	✗	1.37	0.43	1.01	0.88	2.06	0.64	0.56	0.47	4.51	1.59	1.21	0.80
Finetune LiT [75]	✗	64.88	30.42	57.75	29.77	58.15	35.28	51.95	35.96	79.25	64.84	81.79	65.4
Finetune CoOp [85]	✗	60.73	37.52	54.20	39.77	27.61	8.57	24.03	10.14	47.85	33.46	42.02	24.74
SimpleCIL [83]	✗	81.06	74.48	76.84	74.48	83.81	77.52	79.75	77.52	90.44	85.68	88.12	85.68
ZS-CLIP [46]	✗	83.37	77.17	79.57	77.17	74.38	63.06	67.96	63.06	75.50	67.64	71.44	67.64
CoOp [85]	✓	82.40	76.20	79.76	77.13	77.34	68.70	74.09	67.47	90.13	86.24	88.36	85.71
iCaRL [47]	✓	72.22	54.38	68.67	60.15	82.04	74.74	78.57	75.07	89.47	84.34	88.51	84.11
MEMO [82]	✓	80.00	74.07	76.72	73.95	77.32	65.69	72.88	66.41	84.02	74.08	82.58	75.48
L2P [64]	✓	75.73	67.22	74.15	71.20	79.23	68.54	75.85	71.12	88.71	83.93	86.51	83.22
DualPrompt [63]	✓	78.47	70.82	72.98	69.18	83.21	74.94	78.06	74.27	89.48	85.41	86.96	84.65
PROOF	✓	**85.34**	**80.10**	**82.32**	**80.30**	**84.93**	**79.43**	**81.67**	**79.18**	**92.34**	**89.92**	**91.70**	**89.16**

Method	Text → Image R_B@1	$\bar{R}$@1	R_B@5	$\bar{R}$@5	R_B@10	$\bar{R}$@10
Finetune	37.35	51.33	67.38	77.77	77.95	85.55
DER [69]	66.71	74.18	89.63	93.00	94.84	96.69
MEMO [82]	69.53	76.35	91.89	94.44	96.09	97.32
PROOF	**72.10**	**78.01**	**93.10**	**95.27**	**96.92**	**97.90**



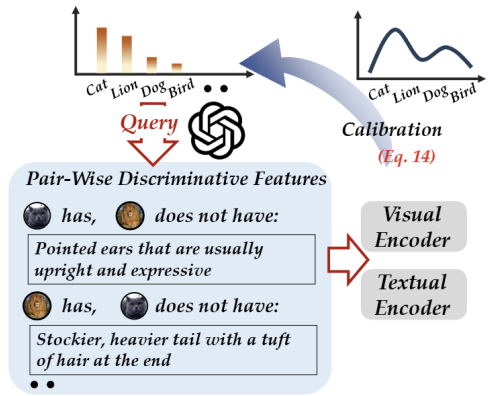


(a) Unseen class accuracy (b) LAION score (c) $\mathcal{A}_S$, $\mathcal{A}_U$, $\mathcal{A}_{HM}$

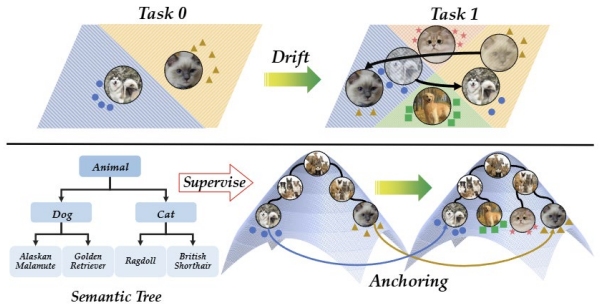
- 在9个基准数据集上取得最优性能
- 可维持模型的零样本学习能力不下降
- 除了使用CLIP进行分类任务以外，在其他视觉语言预训练模型（BEiT-3）的其他任务场景（持续跨模态检索）中也展示出优秀性能

基于视觉-语言模型的表示扩张

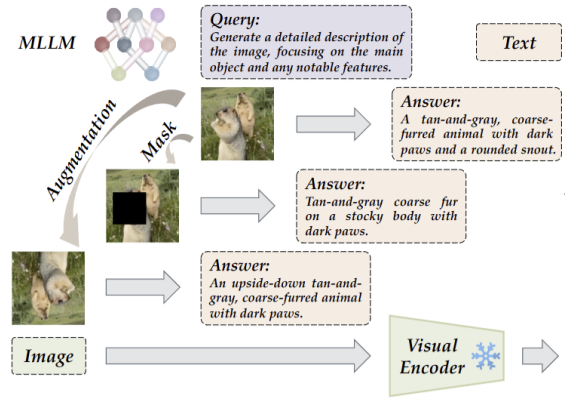
相比于 [A Photo of a Class]，大语言模型可为持续学习带来何种便利?



属性级描述[Zhou et al., ICCV 2025]：“猫是具有向上竖着的尖耳朵的动物”



层级语义 [Hu et al., CoRR 2025]：动物-猫科-英国短毛猫

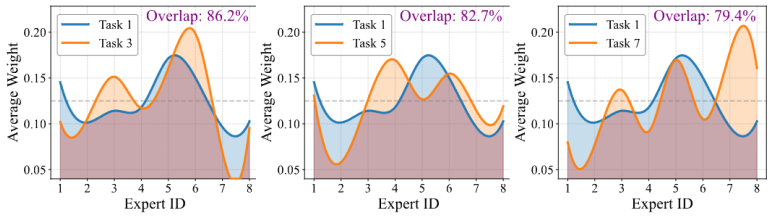


图片级描述 [Xie, Shi, Zhou, ICML 2026]：一种毛色棕褐色和灰色相间、毛发粗糙的动物，爪子呈深色，鼻子呈圆形

Zhou, et al., *External Knowledge Injection for CLIP-Based Class-Incremental Learning*. ICCV 2025
Hu, Li, Xie, Zhou. *Hierarchical Semantic Tree Anchoring for CLIP-Based Class-Incremental Learning*. CoRR 2025
Xie, Shi, Zhou. *AREA: Attribute Extraction and Aggregation for CLIP-Based Class-Incremental Learning*. ICML 2026

多模态大模型MOE内部的兼容

由于模型持续训练，MOE机制存在路由-专家的不兼容，因而“选不准、做不对”



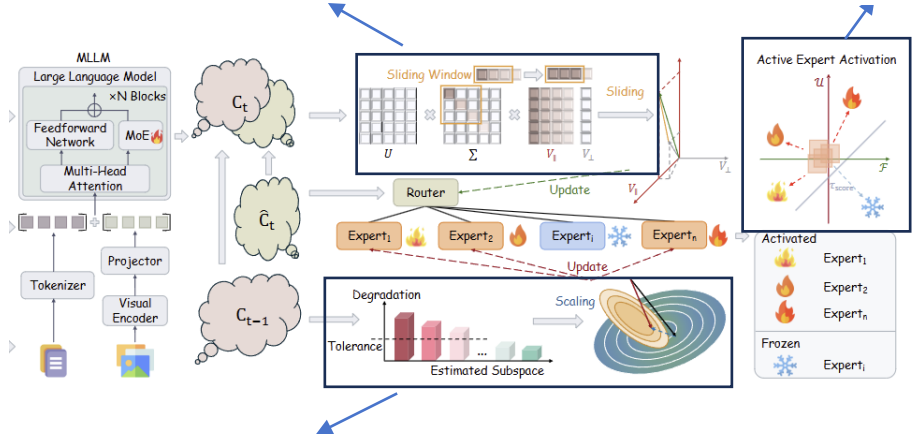
(a) Task 1 vs Task 3 *(b)* Task 1 vs Task 5 *(c)* Task 1 vs Task 7

路由漂移：随着持续学习，原本应该激活处理旧任务专家的输入，被错误地路由到了不相关的专家

专家漂移：即使路由路径是正确，专家自身的参数被更新，导致处理旧任务的特定功能退化

维护动态更新的路由输入协方差矩阵，捕捉输入数据分布的变化轨迹

量化专家参数更新对旧任务造成的影响，自适缩放梯度



路由的更新梯度仅投影到任务相关子空间进行优化，同时严格限制在历史保持子空间上的参数变化，从而保护旧任务路由

持续学习系列算法工具包

ResNet

ViT

CLIP

MLLM

PyCIL

PILOT
A Pre-trained Model-based Continual Learning Toolbox

C3Box
A CLIP-based Class-Incremental Learning Toolbox

PRISM

大模型有了进一步发展...

大模型能力的进一步提升，为大模型持续学习带来了机遇和挑战

2025.9 OpenAI :

把 Operator 整合进 ChatGPT 的 agent mode

支持大模型调用工具

2026.1 Claude:

memory tool 可以跨会话创建、读取、更新、删除持久化记忆文件

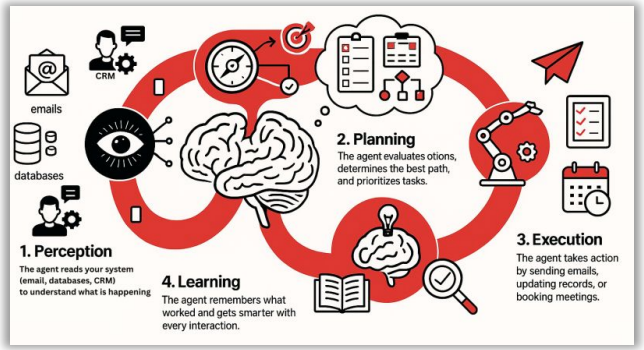
大模型拥有记忆

2026.3 OpenClaw:

GitHub 仓库里已经有技能注册与自动搜索技能的 ClawHub

智能体技术成熟

新知识不再需要写进模型参数中

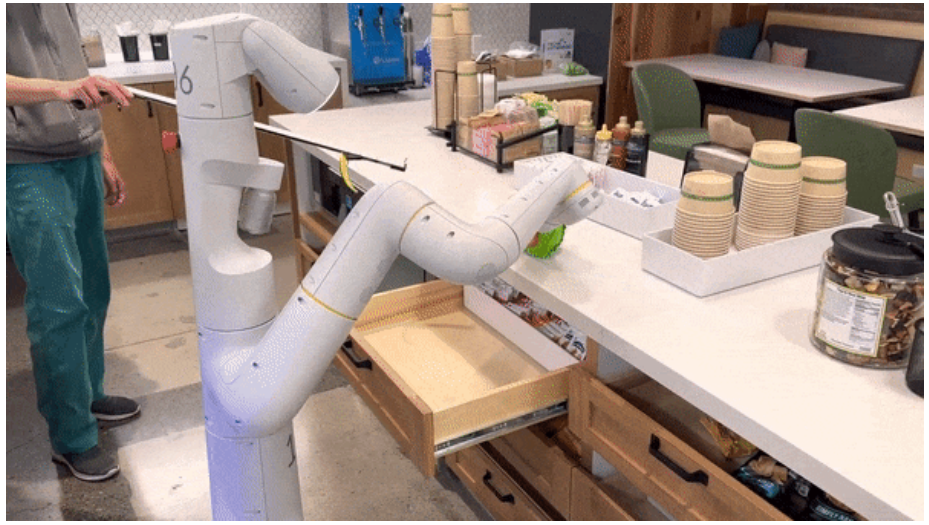


新能力可以通过接入新工具来获得，但模型可能忘记某些工具的使用方式，或由于工具更新而策略失效

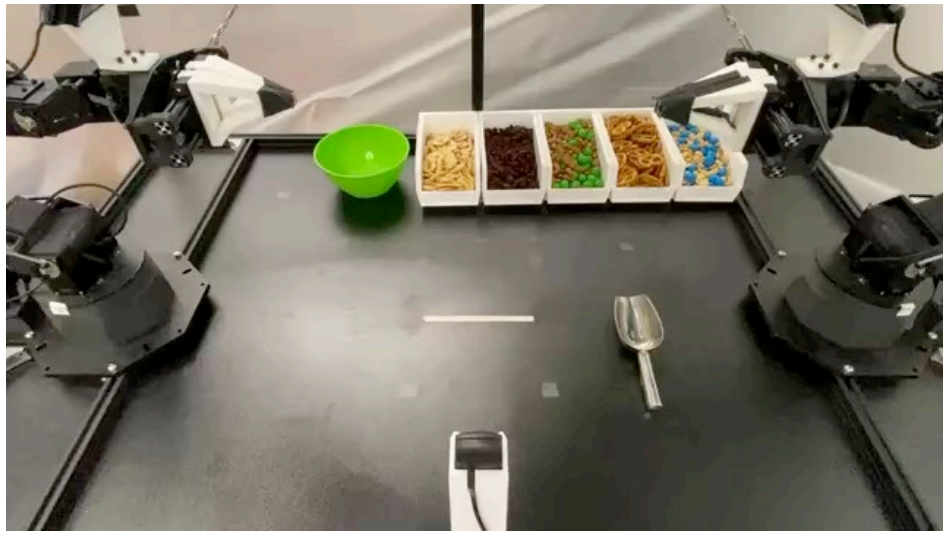
新知识也可以是记忆空间的扩展，这使得学习速度更快，但过时偏好、过期事实一旦被写入，就可能长期影响后续行为

大模型的持续演进

未来持续学习不只是“输入新数据再训练”，而是智能体在环境中
感知—行动—反馈—修正—再探索 的闭环

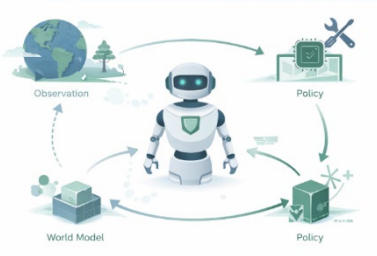


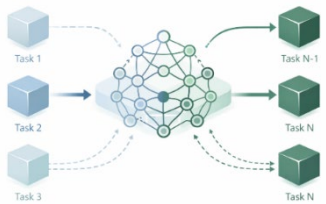
基于大模型持续学习，构成整体系统化的自我演进能力

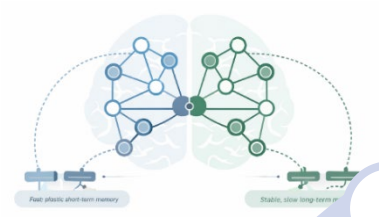


大模型可以自主地探索，并利用探索后的知识进行自主学习演进

总结与展望







未来的持续学习：
关注智能体的长期自主演化与能力兼容

现在的持续学习：
关注预训练表征与下游任务兼容

过去的持续学习：
关注旧类与新类兼容

神经科学的持续学习：
关注神经元间的互补记忆

谢谢！