SHANDONG UNIVERSITY

PRCV 2025

智汇沪上，海纳视界

持续学习：从小模型到大模型

CONTINUAL LEARNING: FROM SMALL MODEL TO LARGE MODEL

报告人：丛润民

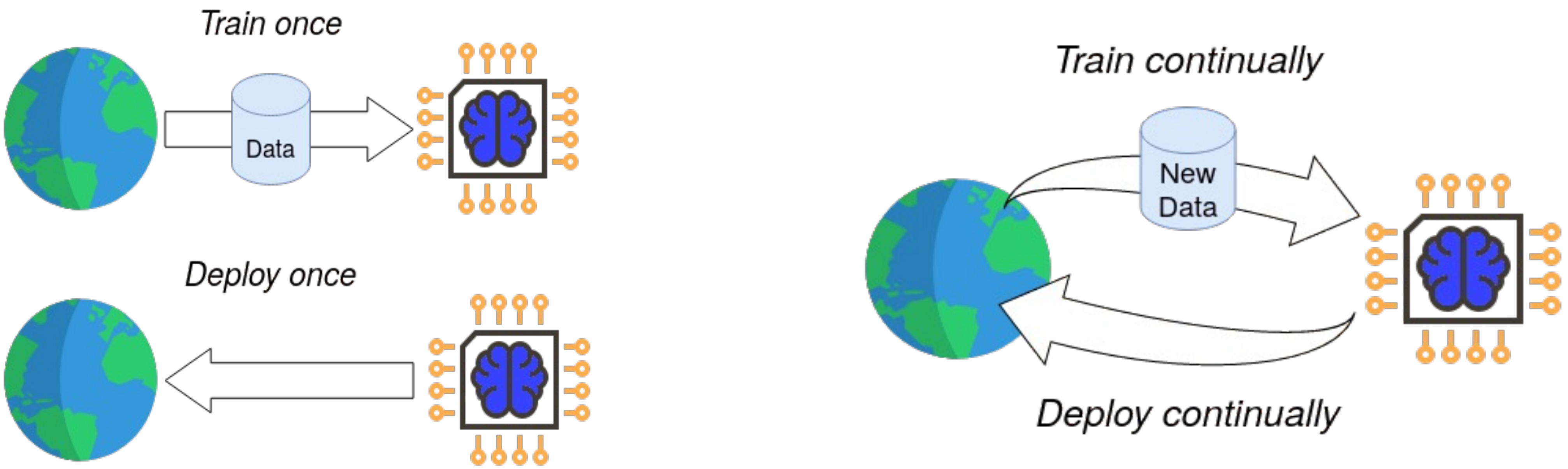
山东大学控制科学与工程学院

机器智能与系统控制教育部重点实验室

—— 学无止境 气有浩然 ——

- Introduction
- Technical Methods
  - Replay Without Saving: Prototype Derivation and Distribution Rebalance for Class-Incremental Semantic Segmentation (TPAMI’25 & NeurIPS’23)
  - SEFE: Superficial And Essential Forgetting Eliminator For Multimodal Continual Instruction Tuning (ICML’25)

Introduction —— Transient → Continual Learning



Transient Perception → Continual Perception

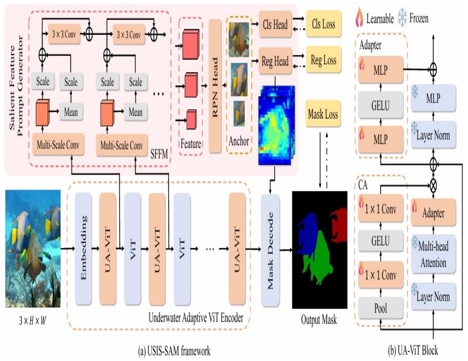
As shown in the above image, a conventional model can only be trained once and has fixed capabilities. In contrast, a model with continual learning abilities can continuously expand its capabilities to meet new requirements.

Introduction —— Transient → Continual Learning

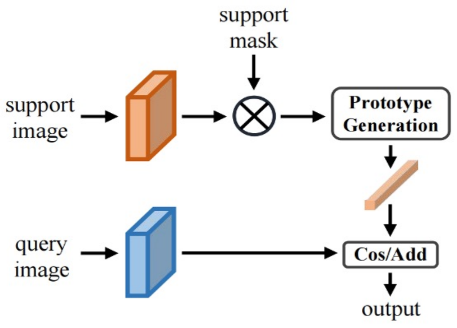
瞬态学习

持续学习

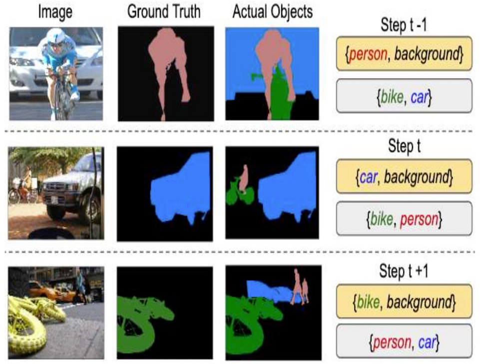
全监督 **小样本** **小模型** **大模型**



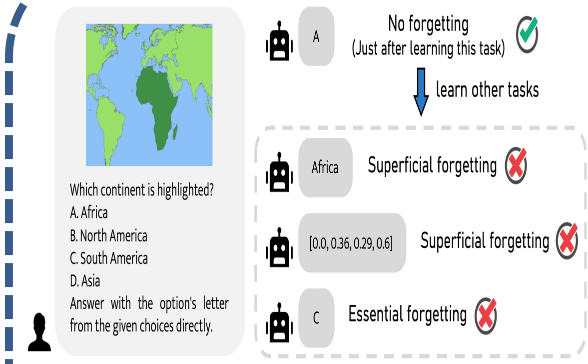
当前类别学习
全监督学习



新类别学习
小样本学习

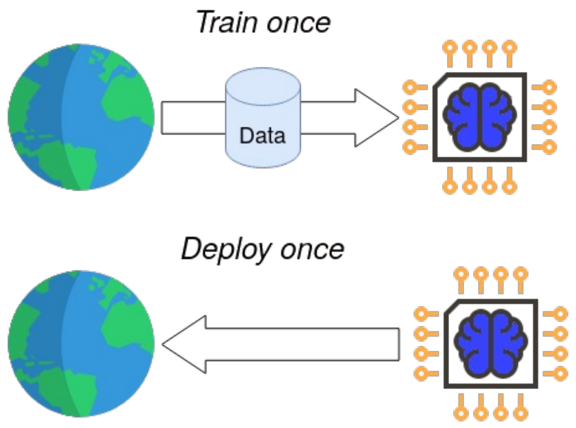


小模型语义理解
持续学习

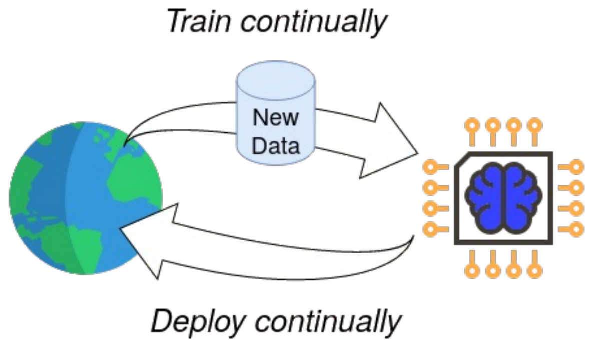


多模态大模型推理
持续学习

Introduction —— Continual Learning



Conventional Model



Continual Learning Model

- **Continual Learning** refers to the ability of a model to retain previously learned knowledge while continuously receiving new data and acquiring new knowledge. The main challenge is to **avoid "Catastrophic Forgetting"**.
- As shown in the above image, a conventional model can only be trained once and has fixed capabilities. In contrast, **a model with continual learning abilities can continuously expand its capabilities to meet new requirements**.
- The methods for continual learning can be broadly categorized into **regularization**, **replay**, and **parameter isolation**.

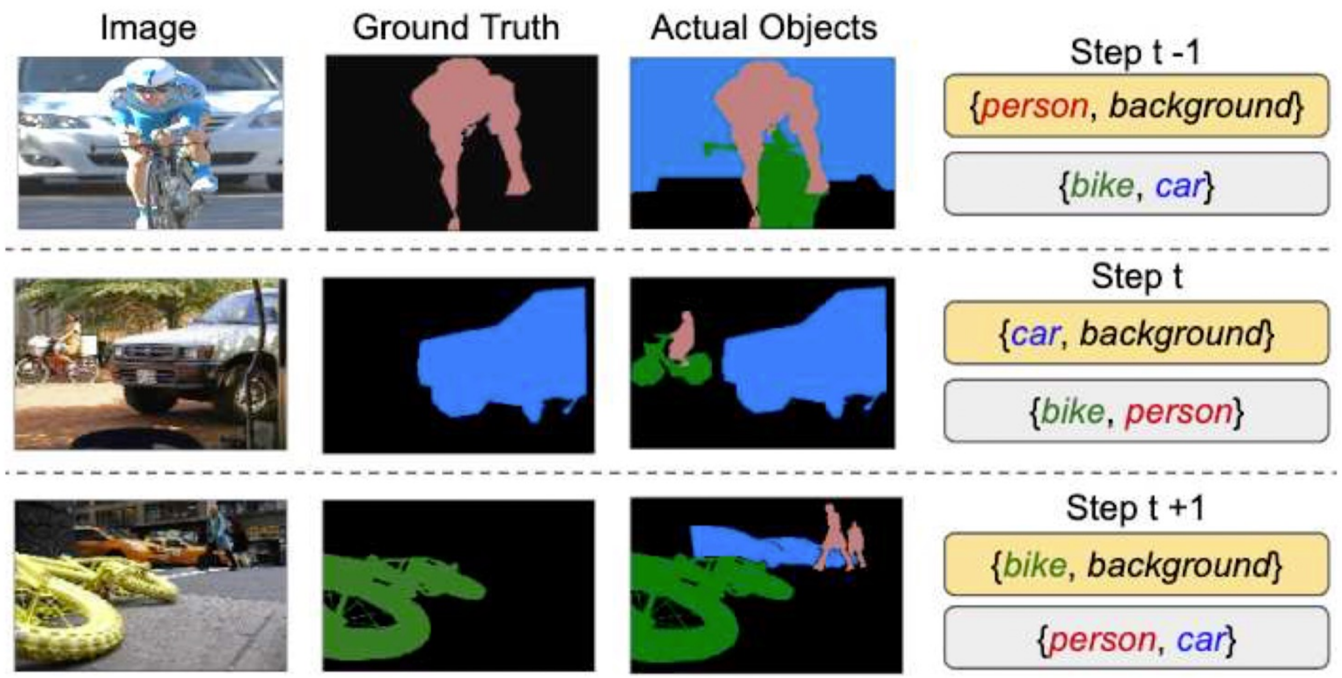
NeurIPS 2023 → TPAMI 2025

Saving 100x Storage: Prototype Replay for Reconstructing Training Sample Distribution in Class-Incremental Semantic Segmentation

Replay Without Saving: Prototype Derivation and Distribution Rebalance for Class-Incremental Semantic Segmentation

Jinpeng Chen, Runmin Cong, Yuxuan Luo, Horace Ip, and Sam Kwong

Task Definition



- In **class-incremental semantic segmentation (CISS)**, each step focuses on different classes, with its training set only annotates current classes, while previously learned classes and future classes are labeled as *background*.
- The images in each single-step training set contain at least one pixel from current classes, and images devoid of any current class are excluded.

Motivation

- There are **a lot of false positives for classes in incremental steps** (i.e., steps beyond the first).
- This is because **the proportion of current classes in the single-step training set is significantly higher than in the complete dataset, leading to classification bias**, which is especially pronounced in incremental steps with fewer classes.

To address this issue, the key is to augment past classes and background pixels in the training samples of the incremental steps, thereby reducing the proportion of the current class. At the same time, it is important to avoid triggering excessive storage requirements.

Solution

Prototype Replay	Background Repetition
At each task, the pixel occurrence count for each class is recorded. In subsequent tasks, pixel-level class prototypes are replayed based on these occurrence counts.	At each task, the cumulative pixel count of the background class is updated. In subsequent tasks, background features are duplicated according to this count.

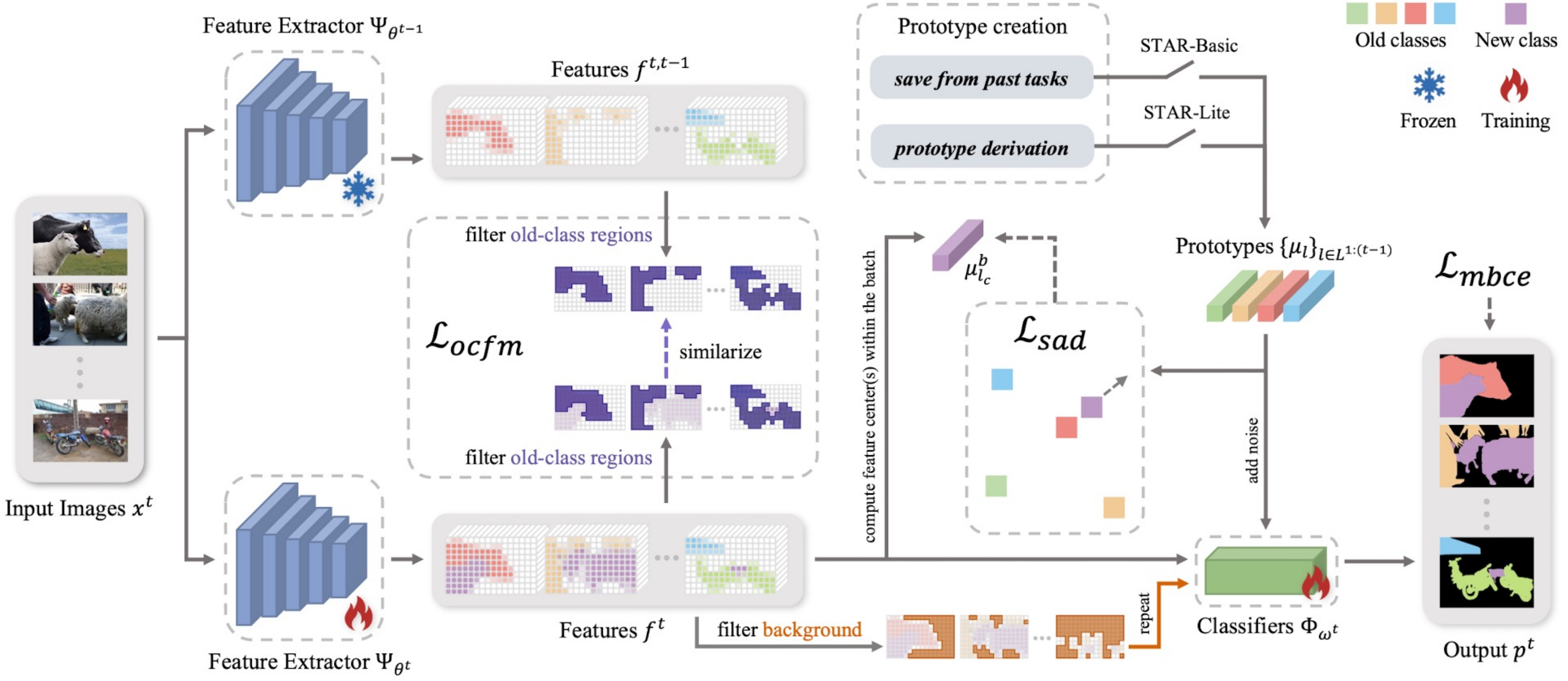
These two strategies respectively adjust the proportion of foreground and background classes within the single-step training samples to match the proportion in the "cumulative training set up to the current step", thus avoiding bias.

Contributions

- We propose a new CISS method named **STAR**. Its **basic version** stores compact prototypes and necessary statistics for each learned class. This enables a **comprehensive reconstruction of single-task training sample distributions**, aligning them with the complete dataset to mitigate classification bias.
- We develop a **prototype derivation method** that considers both the recognition and extraction patterns of the network. This empowers **prototype creation without the need for storage**, leading to a **lite version**.
- The **OCFM loss** is introduced to **retain learned knowledge in a spatially targeted manner**, maintaining old-class features while ensuring flexibility for learning new classes. Additionally, the **SAD loss** is designed to enhance the feature discriminability between similar old-new class pairs, facilitating the classification.

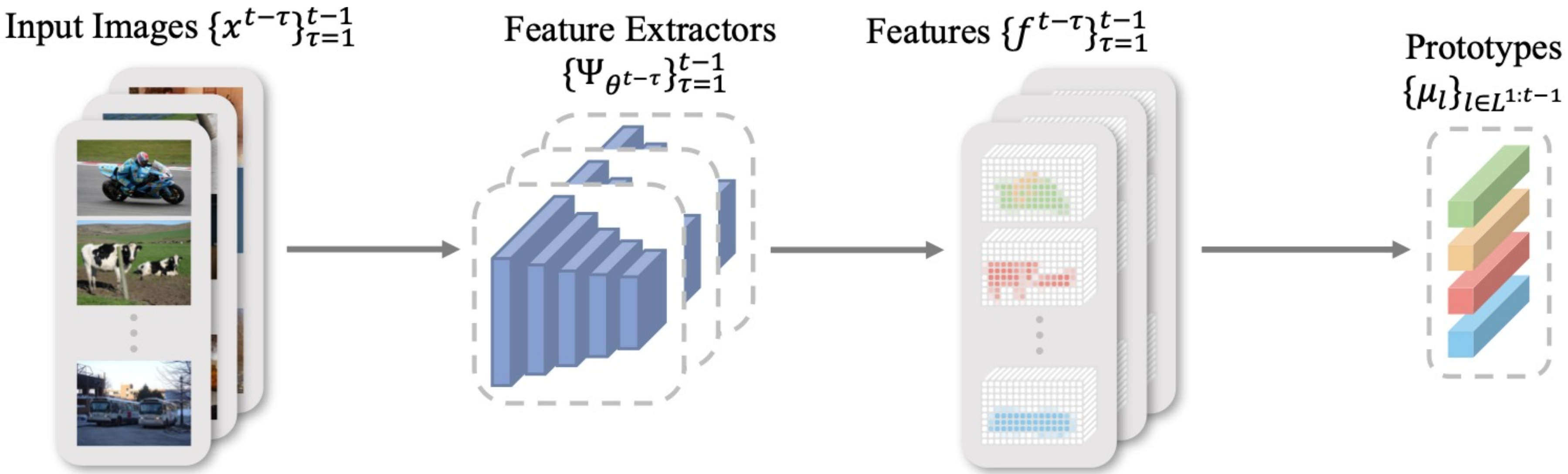
Our STAR Method

https://github.com/jinpeng0528/STAR



Prototype Replay – Basic Version

- After the learning of each task, all training samples in this task are passed through the frozen model to compute the features.
- The feature centers are stored as **<u>prototypes</u>** and replayed in subsequent tasks.
- Since prototypes are highly compact, they require only 1/100 of the storage compared to existing replay-based methods that storing raw images.



Prototype Replay – Lite Version

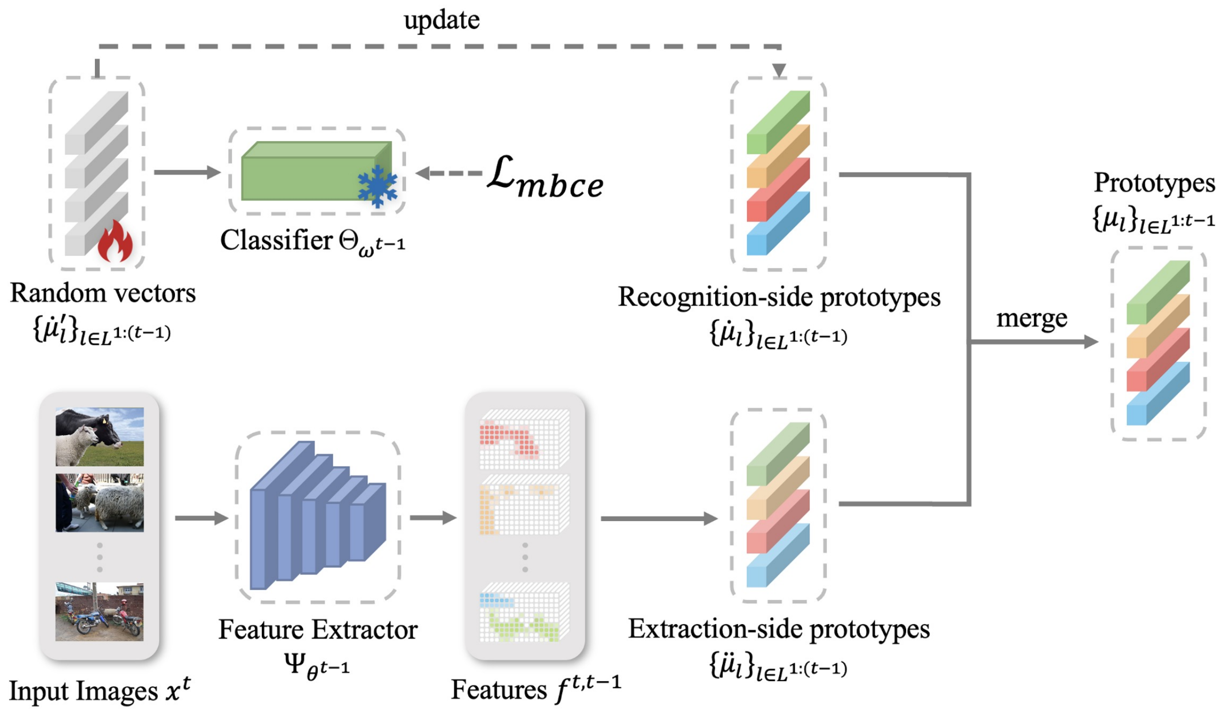
By leveraging the network's classification recognition and feature extraction patterns, prototypes are derived without the need for any storage.

Recognition Patterns

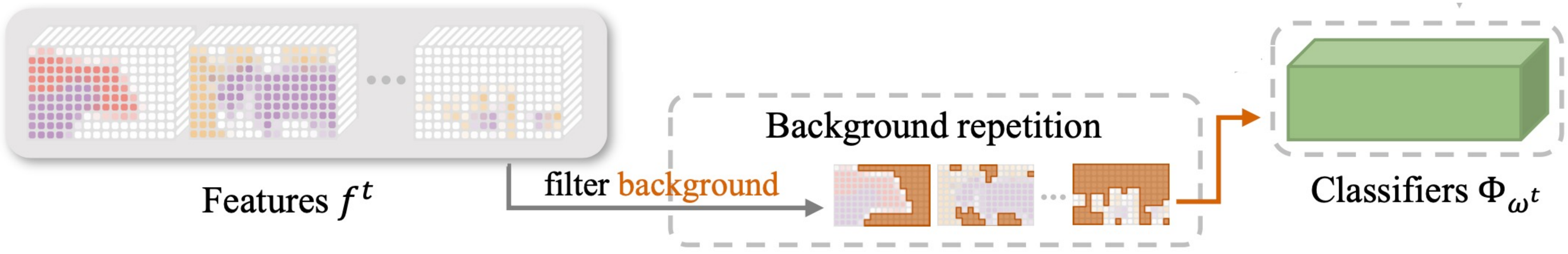
The classifier is isolated from the network. Then, it is used to infer representative features of previous classes, forming the **recognition-side prototype**.

Extraction Patterns

Images from the current task are fed into the network, and features from regions predicted as belonging to previous classes are aggregated to construct the **extraction-side prototypes**.



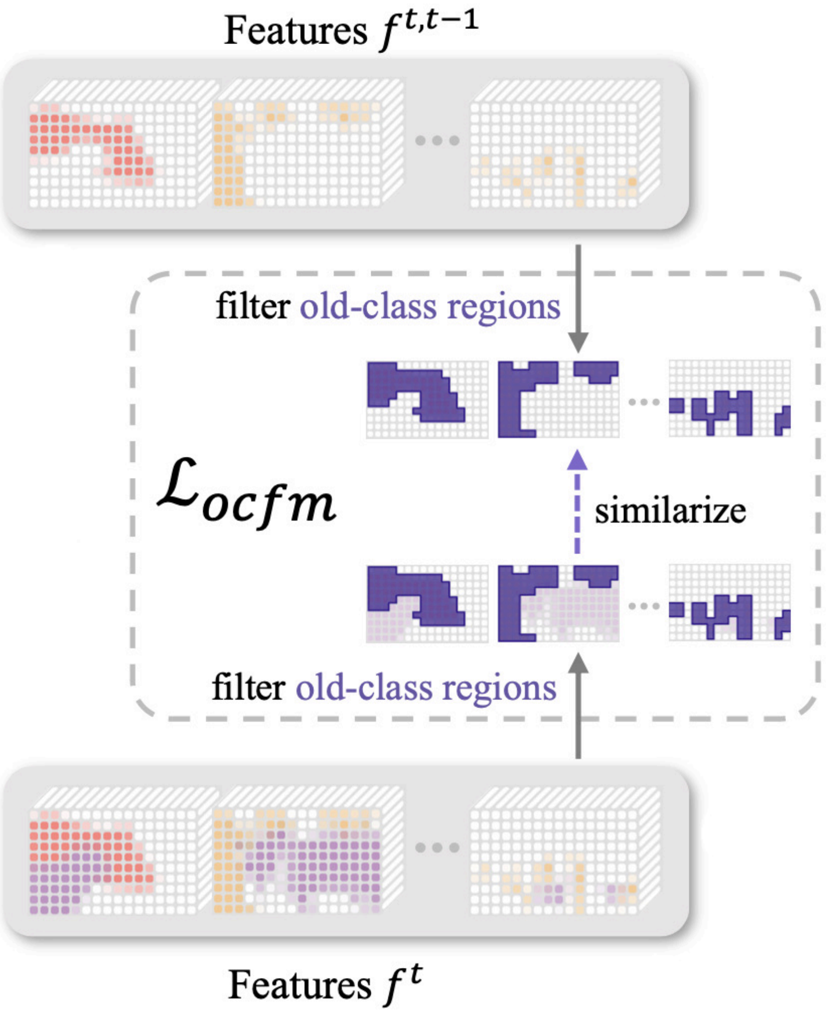
Background Repetition



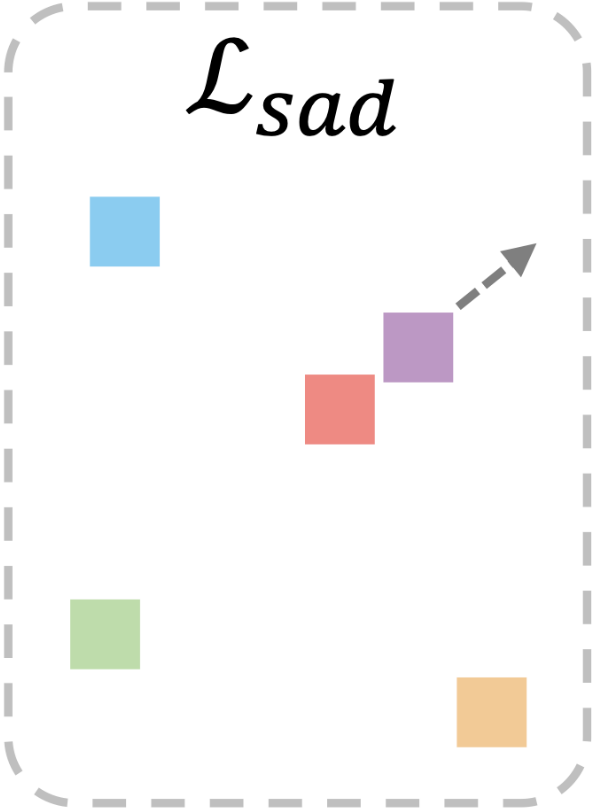
- Starting from the first step, we record and update the cumulative occurrence count of background pixels, η_{bg}.
- In subsequent steps, the background regions of input images are filtered out using current annotations and predictions from the previous model.
- **The features of these background regions are repeated multiple times and fed into the classifiers to add η_{bg} extra background pixels**, thus aligning the proportion of background in the single-step training samples with that of the ”cumulative training set up to the current step”.

Old-Class Feature Maintaining Loss

- A crucial prerequisite for effective prototype replay is the relative stability of old-class feature space.
- The old-class feature-maintaining loss utilizes current labels and predictions from the previous model to **locate old-class regions**. Within these regions, it **constrains the features extracted by the current model to be close to those extracted by the previous model**.



Similarity-Aware Discriminative Loss



- Some similar "new-old class pairs" are prone to confusion because they appear in different steps, making it challenging for the feature extractor to generate discriminative features.
- The most direct approach is to penalize the similarity of all "new-old class pairs" feature centers, increasing their distance.
- However, this method may lead to resource waste as some "new-old class pairs" are inherently dissimilar. Therefore, we **penalize the similarity between each new class feature center and its closest old class feature center**, focusing on the most challenging points.

Experiments

Method	19-1						15-5						15-1					
	Disjoint			Overlapped			Disjoint			Overlapped			Disjoint			Overlapped		
	base	*inc.*	*all*	*base*	*inc.*	*all*	*base*	*inc.*	*all*	*base*	*inc.*	*all*	*base*	*inc.*	*all*	*base*	*inc.*	*all*
MiB [10]	69.6	25.6	67.4	70.2	22.1	67.8	71.8	43.3	64.7	75.5	49.4	69.0	46.2	12.9	37.9	35.1	13.5	29.7
SDR [14]	69.9	37.3	68.4	69.1	32.6	67.4	73.5	47.3	67.2	75.4	52.6	69.9	59.2	12.9	48.1	44.7	21.8	39.2
PLOP [12]	75.1	38.2	73.2	75.4	37.4	73.5	66.5	39.6	59.8	75.7	51.7	70.1	49.0	13.8	40.2	65.7	17.3	54.2
SSUL [11]	77.4	22.4	74.8	77.7	29.7	75.4	76.4	45.6	69.1	77.8	50.1	71.2	74.0	32.2	64.0	77.3	36.6	67.6
STCISS [55]	76.6	36.0	75.4	76.1	43.4	74.5	76.9	54.3	71.3	76.7	54.3	71.1	70.1	34.3	61.2	71.4	40.0	63.6
RBC [58]	76.4	45.8	75.0	77.3	55.6	76.2	75.1	49.7	69.9	76.6	52.8	70.9	61.7	19.5	51.6	69.5	38.4	62.1
DKD [9]	77.4	43.6	75.8	77.8	41.5	76.0	77.6	54.1	72.0	78.8	58.2	73.9	76.3	39.4	67.5	78.2	44.3	70.1
UCD [56]	75.7	31.8	73.5	75.9	39.5	74.0	67.0	39.3	60.1	75.0	51.8	69.2	50.8	13.3	41.4	66.3	21.6	55.1
EWF [57]	78.2	3.2	74.6	77.9	6.7	74.5	79.3	38.2	69.5	79.4	38.2	69.5	75.3	22.5	62.7	78.5	31.6	67.3
STAR-Lite	77.9	46.4	**76.4**	78.1	49.1	**76.8**	78.5	58.3	**73.7**	79.7	59.4	**74.8**	78.5	45.9	**70.8**	80.0	51.2	**73.1**
RECALL [59]	65.0	47.1	65.4	68.1	55.3	68.6	69.2	52.9	66.3	67.7	54.3	65.6	67.6	49.2	64.3	67.8	50.9	64.8
PLOPLong [60]	-	-	-	74.8	39.7	73.1	-	-	-	76.0	48.3	69.4	-	-	-	72.0	26.7	61.2
SSUL-M [11]	77.6	43.9	76.0	77.8	49.8	76.5	76.5	48.6	69.8	78.4	55.8	73.0	76.5	43.4	68.6	78.4	49.0	71.4
DKD-M [9]	77.6	56.9	76.6	78.0	57.7	**77.0**	77.7	55.4	72.4	79.1	60.6	74.7	77.3	48.2	70.3	78.8	52.4	72.5
STAR-Basic	78.0	47.5	76.5	78.2	48.5	76.8	78.5	57.9	73.6	79.7	59.6	74.9	78.1	48.2	71.0	79.8	51.6	73.1
STAR-Basic†	77.9	53.6	**76.7**	78.1	56.3	**77.0**	78.6	58.4	**73.8**	80.1	62.2	**75.8**	77.8	50.4	**71.3**	79.8	55.5	**74.0**

Pascal VOC 2012 Dataset - 1

Method	10-1			5-3		
	base	*inc.*	*all*	*base*	*inc.*	*all*
MiB [10]	12.3	13.1	12.7	57.1	42.6	46.7
PLOP [12]	44.0	15.5	30.5	17.5	19.2	18.7
SSUL [11]	71.3	46.0	59.3	72.4	50.7	56.9
DKD [9]	73.1	46.5	60.4	69.6	53.5	58.1
EWF [57]	71.5	30.3	51.9	61.7	42.2	47.7
STAR-Lite	74.0	53.5	**64.3**	72.1	59.6	**63.2**
SSUL-M [11]	74.0	53.2	64.1	71.3	53.2	58.4
DKD-M [9]	74.0	56.7	65.8	69.8	60.2	62.9
STAR-Basic	72.6	55.4	64.4	70.7	61.8	64.3
STAR-Basic†	74.4	56.9	**66.1**	72.4	63.3	**65.9**

Pascal VOC 2012 Dataset - 2

Method	13-6			13-1		
	base	*inc.*	*all*	*base*	*inc.*	*all*
MiB [10]	52.8	17.9	41.8	51.6	22.9	42.5
PLOP [12]	53.2	10.1	39.6	52.4	15.1	40.6
DKD [9]	55.5	36.4	49.8	55.7	20.9	46.5
UCD [56]	53.0	18.6	42.1	52.2	23.4	43.1
STAR-Lite	56.6	50.5	54.7	55.7	31.2	**48.3**
STAR-Basic	56.4	50.9	**54.8**	55.7	31.1	**48.3**

CityScapes Dataset

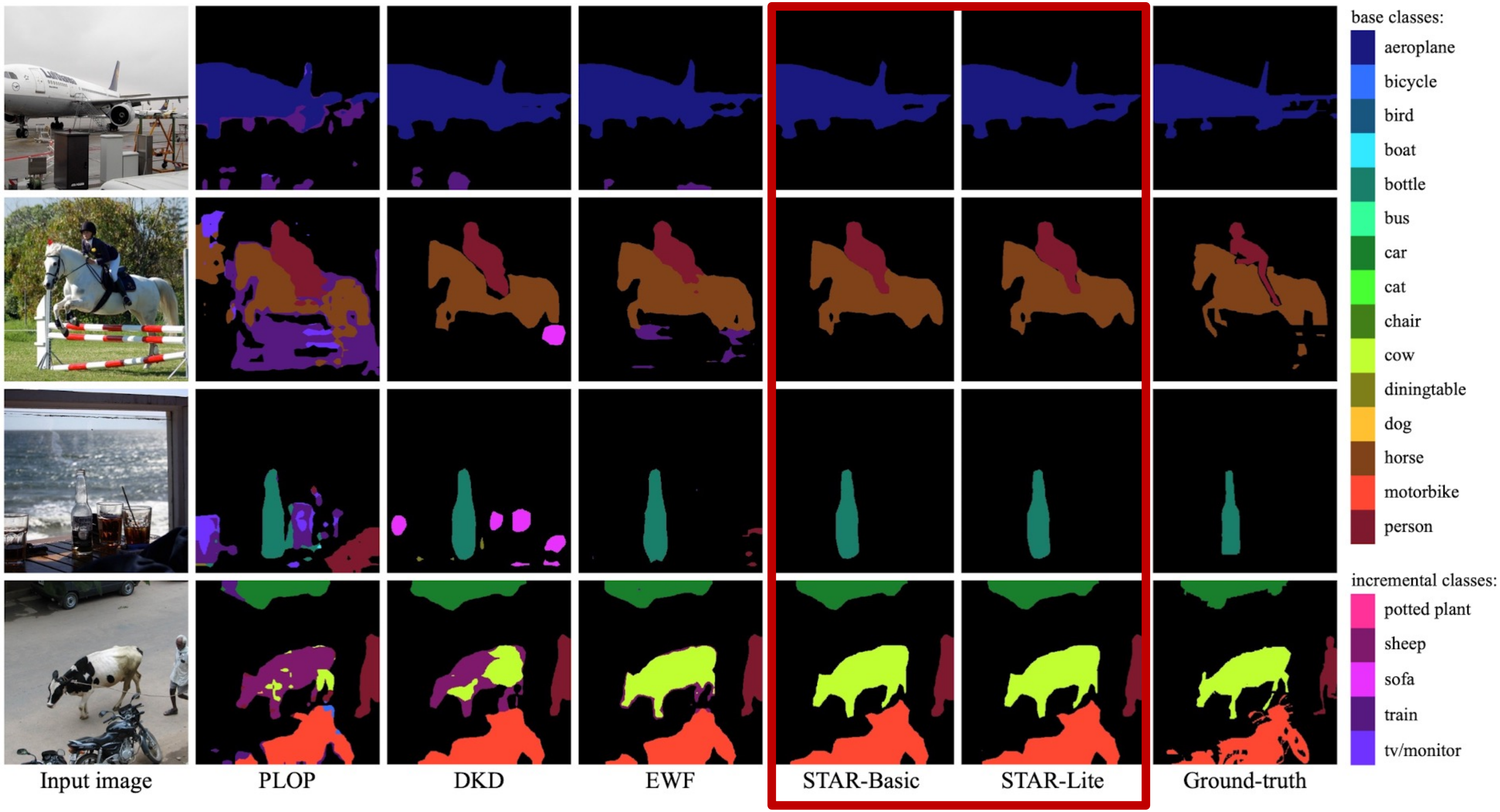
Method	100-50			100-10			50-50		
	base	*inc.*	*all*	*base*	*inc.*	*all*	*base*	*inc.*	*all*
MiB [10]	40.5	17.2	32.8	38.2	11.1	29.2	45.6	21.0	29.3
PLOP [12]	41.9	14.9	32.9	40.5	13.6	31.6	48.8	21.0	30.4
SSUL [11]	41.3	18.0	33.6	40.2	18.8	33.1	48.4	20.2	29.6
RCIL [54]	42.3	18.8	34.5	39.3	17.6	32.1	48.3	25.0	32.5
STCISS [55]	40.7	24.0	35.1	33.6	16.9	28.1	40.0	23.6	29.0
RBC [58]	42.9	21.5	35.8	39.0	21.7	33.3	49.6	26.3	34.2
DKD [9]	42.4	22.9	36.0	41.5	19.4	34.2	48.8	26.3	33.9
EWF [57]	41.2	21.3	34.6	41.5	16.3	33.2	46.1	19.8	28.5
STAR-Lite	42.4	24.3	**36.4**	42.0	20.4	**34.9**	48.7	26.9	**34.3**
PLOPLong [60]	41.9	14.9	32.9	40.5	13.6	31.6	48.8	21.0	30.4
SSUL-M [11]	42.8	17.5	34.4	42.9	17.7	34.5	49.1	20.1	29.8
DKD-M [9]	42.4	23.0	36.0	41.7	20.1	34.6	48.8	26.3	33.9
STAR-Basic	42.4	24.3	**36.4**	41.8	20.7	**34.8**	48.3	27.0	**34.2**

ADE20K Dataset

STAR-Basic: Save 100x Storage Cost

STAR-Lite: Replay Without Any Storage

Experiments



Ablation Study

	PR	*BPR*	*OCFM*	*SAD*	**disjoint 15-1** *base*	**disjoint 15-1** *inc.*	**disjoint 15-1** *all*	**overlapped 15-1** *base*	**overlapped 15-1** *inc.*	**overlapped 15-1** *all*
STAR-Lite	×	✓	✓	✓	77.9	41.8	69.3	79.1	40.4	69.9
STAR-Lite	✓	×	✓	✓	76.4	30.8	65.5	78.1	32.3	67.2
STAR-Lite	✓	✓	×	✓	76.9	37.7	67.6	79.5	45.5	71.4
STAR-Lite	✓	✓	✓	×	75.7	33.9	65.7	79.5	47.4	71.9
STAR-Lite	✓	✓	✓	✓	78.5	45.9	**70.8**	80.0	51.2	**73.1**
STAR-Basic	×	✓	✓	✓	77.9	41.8	69.3	79.1	40.4	69.9
STAR-Basic	✓	×	✓	✓	77.5	33.6	67.0	78.6	36.0	68.5
STAR-Basic	✓	✓	×	✓	76.8	36.3	67.1	79.3	46.1	71.4
STAR-Basic	✓	✓	✓	×	75.8	34.8	66.0	79.6	48.9	72.3
STAR-Basic	✓	✓	✓	✓	78.1	48.2	**71.0**	79.8	51.6	**73.1**

Ablation Study Results

PR: Prototype Replay

BPR: Background Pixel Repetition

OCFM: Old-Class Features Maintaining Loss

SAD: Similarity-Aware Discriminative Loss

Conclusion

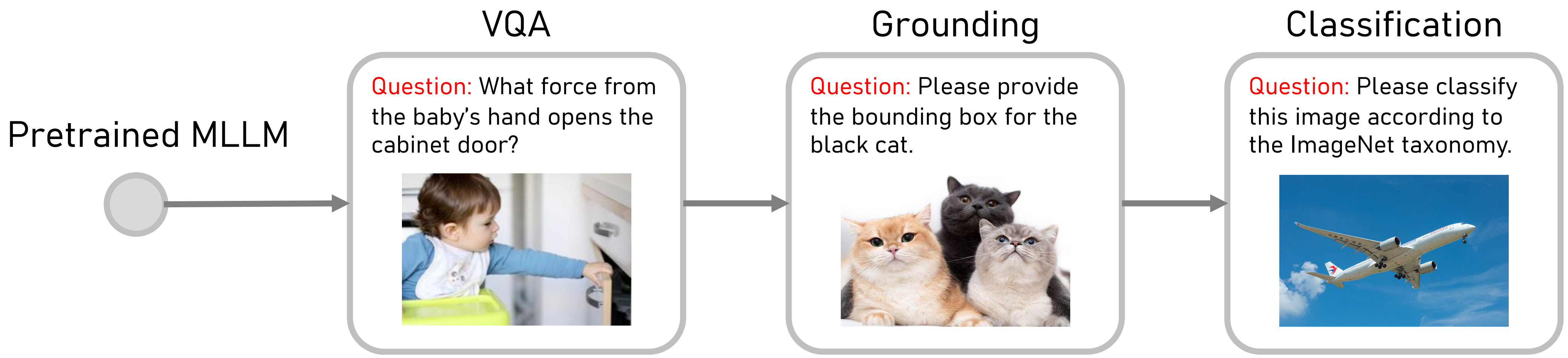
- This paper introduces **STAR**, a CISS method designed to **mitigate classification bias** arising from distribution variances between single-task training sets and the complete dataset.
- STAR employs two principal tactics: **prototype replay** and **background pixel repetition**. The former rectifies the distribution of foreground classes by replaying old-class prototypes, while the latter reintegrates missing background pixels by duplicating background pixels.
- Regarding the creation of prototypes, STAR diverges into two variants. **STAR-Basic** stores prototypes after learning each task for future replay, whereas **STAR-Lite** employs a novel prototype derivation method that considers the network's recognition and extraction patterns to deduce prototypes.
- The **OCFM loss** is introduced to maintain the features of old classes, ensuring the model's ability to learn new classes without losing prior knowledge. Additionally, the **SAD loss** is proposed to enhance feature differentiation between similar old and new class pairs, improving their distinguishability for the classifiers.

ICML 2025

SEFE: Superficial and Essential Forgetting Eliminator for Multimodal Continual Instruction Tuning

Jinpeng Chen, Runmin Cong, Yuzhi Zhao, Hongzheng Yang, Guangneng Hu,
Horace Ho Shing Ip, and Sam Kwong

Introduction



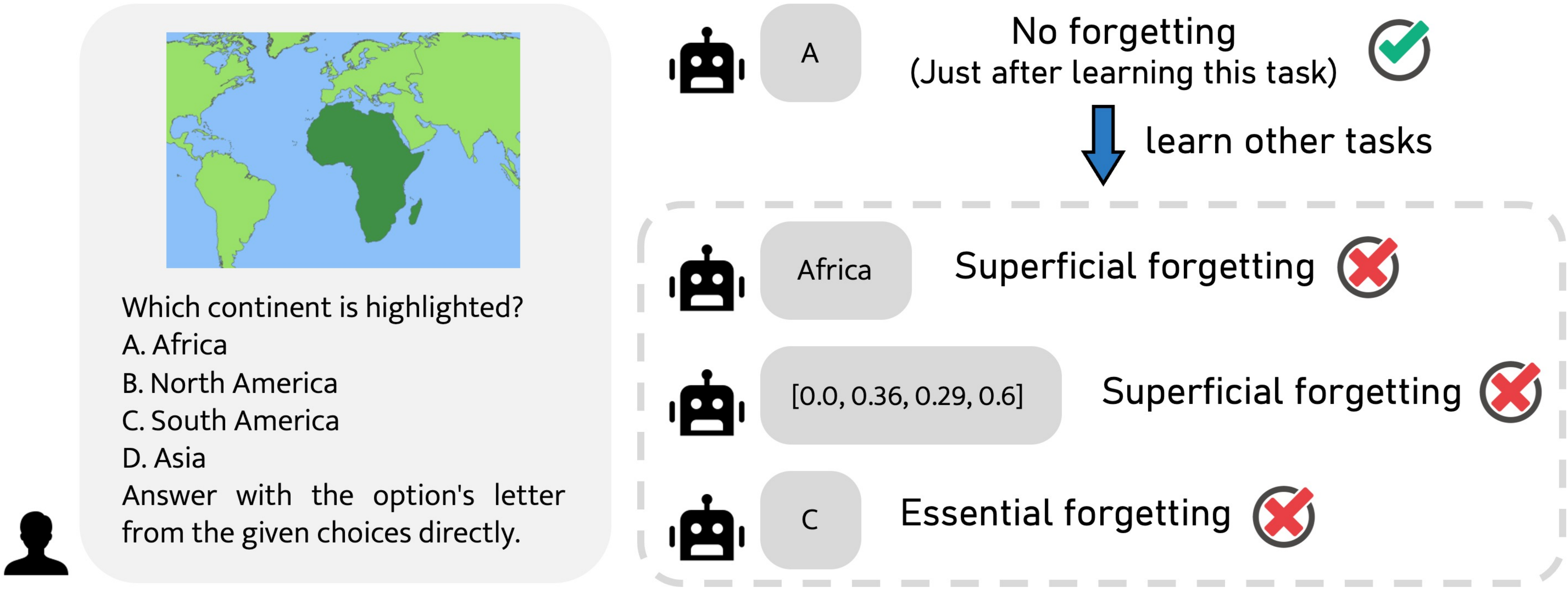
- In **Multimodal Continual Instruction Tuning (MCIT)**, a pretrained Multimodal Large Language Model (MLLM) is sequentially tuned on a series of multimodal tasks, aiming to learn new tasks while minimizing forgetting of previously learned ones.

Introduction

Does the forgetting problem become more severe or alleviated for large and small models under continual learning architectures?

Introduction

Does the forgetting problem become more severe or alleviated for large and small models under continual learning architectures?



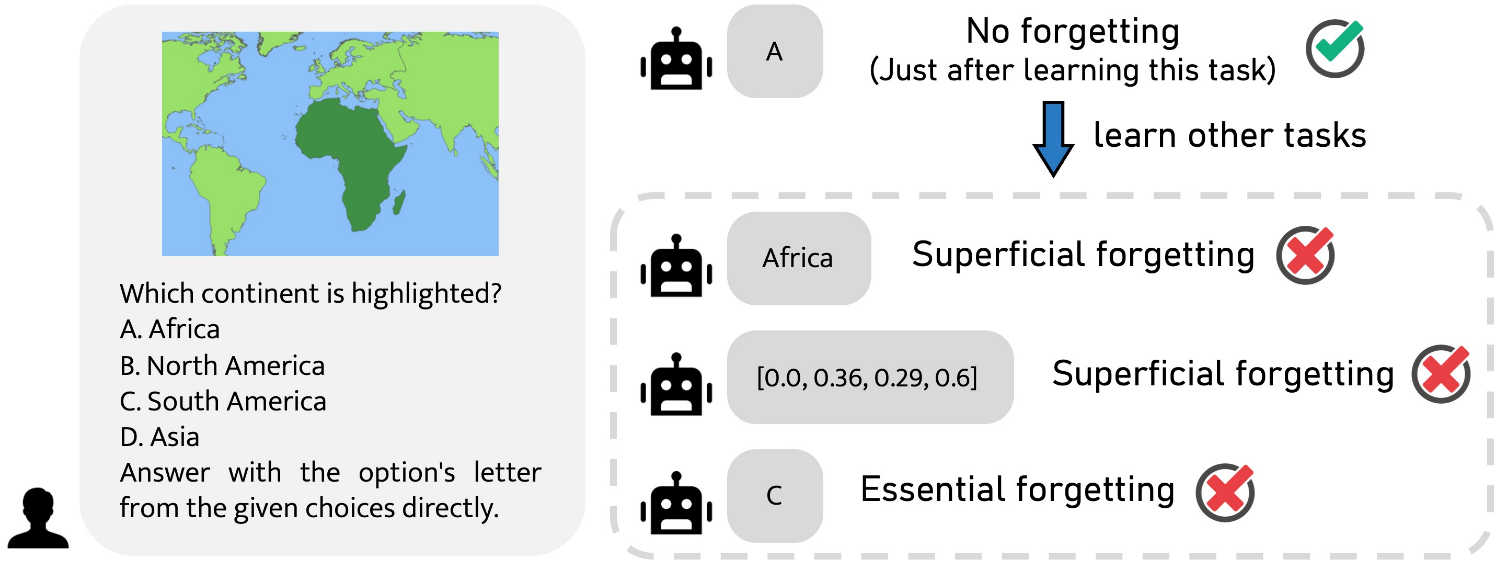
Contributions

a) We formally define *superficial forgetting* and *essential forgetting* in MCIT. Furthermore, our proposed method, SEFE, addresses these challenges and achieves state-of-the-art performance.

b) To mitigate *superficial forgetting*, we introduce the **Answer Style Diversification (ASD)** paradigm that unifies the answer domain across tasks by rephrasing questions, thereby reducing the model's bias toward specific response styles. Additionally, we create **CoIN-ASD**, an ASD-adjusted version of the CoIN benchmark, which can serve as a new benchmark for evaluating *essential forgetting* in future MCIT studies.

c) To address *essential forgetting*, we present **RegLoRA**. By identifying critical elements in the weight update matrices and applying regularization constraints, RegLoRA ensures that LoRA fine-tuning does not disrupt the model's existing knowledge.

Forgetting Types

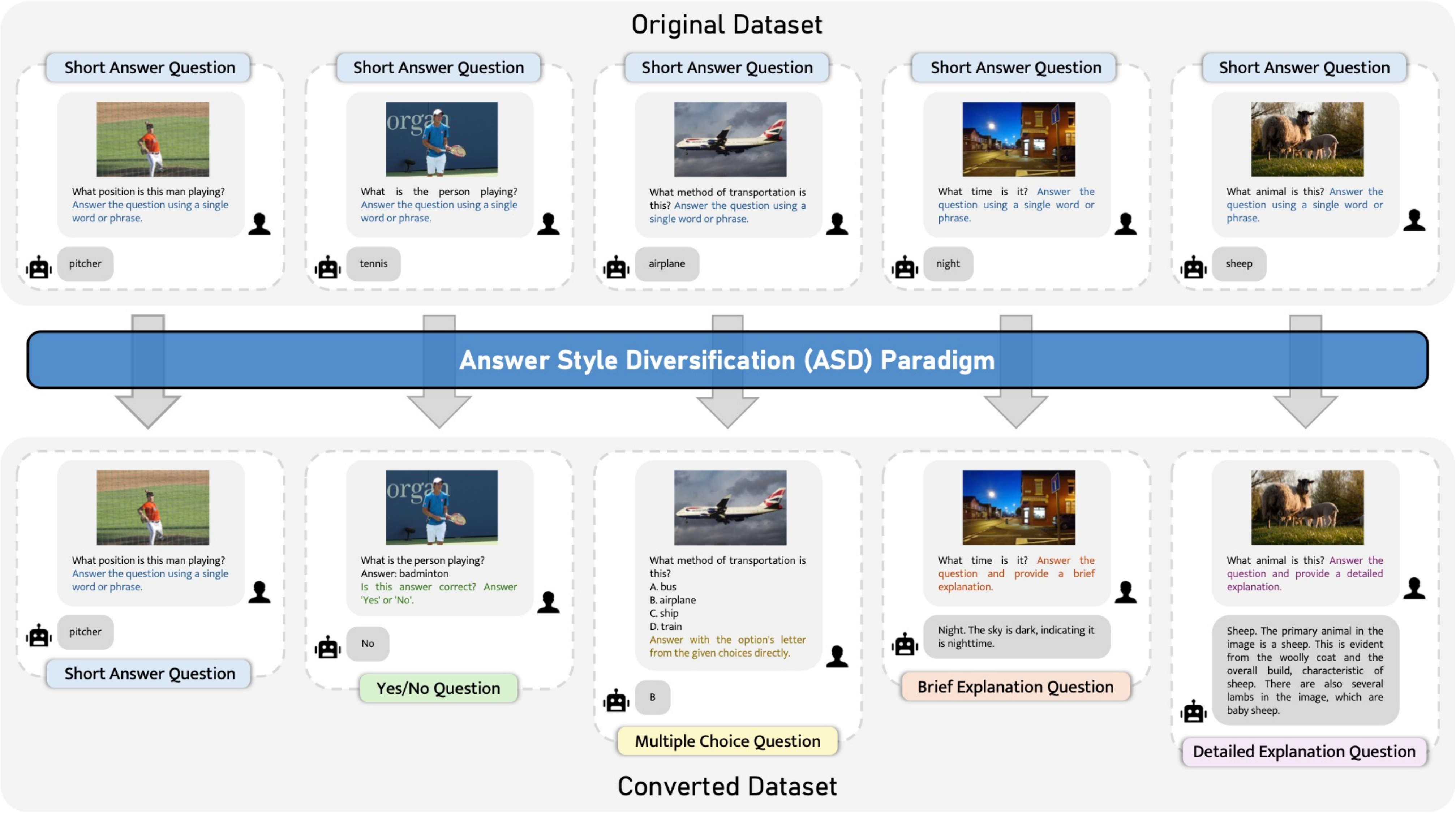


- **Superficial Forgetting:** task knowledge may be retained while the response style is forgotten.
- **Essential Forgetting:** task knowledge is forgotten.

Answer Style Diversification

- *Superficial forgetting* arises from the gap in answer space between tasks, as the model tends to respond in the answer style of the most recently learned task.
- To address this issue, the **Answer Style Diversification (ASD) paradigm** reformulate questions in each task into five unified formats, aligning the answer space across tasks.
- These five formats include Short Answer Question, Yes/No Question, Multiple Choice Question, Brief Explanation Question, and Detailed Explanation Question. After analyzing 15 mainstream benchmarks, we find that these formats sufficiently cover the requirements of all tasks.

Answer Style Diversification



Answer Style Diversification

Method	Accuracy on Each Task (%)								Aggregate Results (%)			
	SQA	VQA^{Text}	*ImgNet*	*GQA*	*VizWiz*	*Grd*	VQA^{v2}	VQA^{OCR}	MFT↑	MFN↑	MAA↑	BWT↓
FFT	2.95	36.38	52.35	46.40	33.90	0.00	61.65	50.00	65.87	35.45	36.73	-30.42
LoRA [20]	54.05	44.63	41.25	47.55	20.80	0.85	59.30	64.30	**70.21**	41.59	39.53	-28.62
O-LoRA [45]	75.40	52.89	71.85	47.30	37.35	7.10	61.85	61.20	<u>69.30</u>	51.87	49.56	-17.43
LoTA [38]	67.30	41.51	8.25	37.15	42.25	0.10	47.95	56.15	54.72	37.58	50.46	-17.14
FFT+ASD	74.50	50.12	65.40	54.35	45.50	0.00	64.40	68.50	68.28	<u>52.85</u>	57.18	-15.44
LoRA+ASD [20]	74.45	49.70	39.30	52.00	50.45	7.05	62.25	47.80	68.13	47.88	<u>59.71</u>	-20.26
O-LoRA+ASD [45]	75.20	55.36	67.50	54.70	52.90	15.40	64.45	35.05	65.59	52.57	61.63	<u>-13.02</u>
LoTA+ASD [38]	76.90	42.65	15.85	40.25	45.10	0.30	54.35	54.00	56.99	41.18	56.28	-15.82

MFT: Mean Fine-tune Accuracy

MFN: Mean Final Accuracy

MAA: Mean Average Accuracy

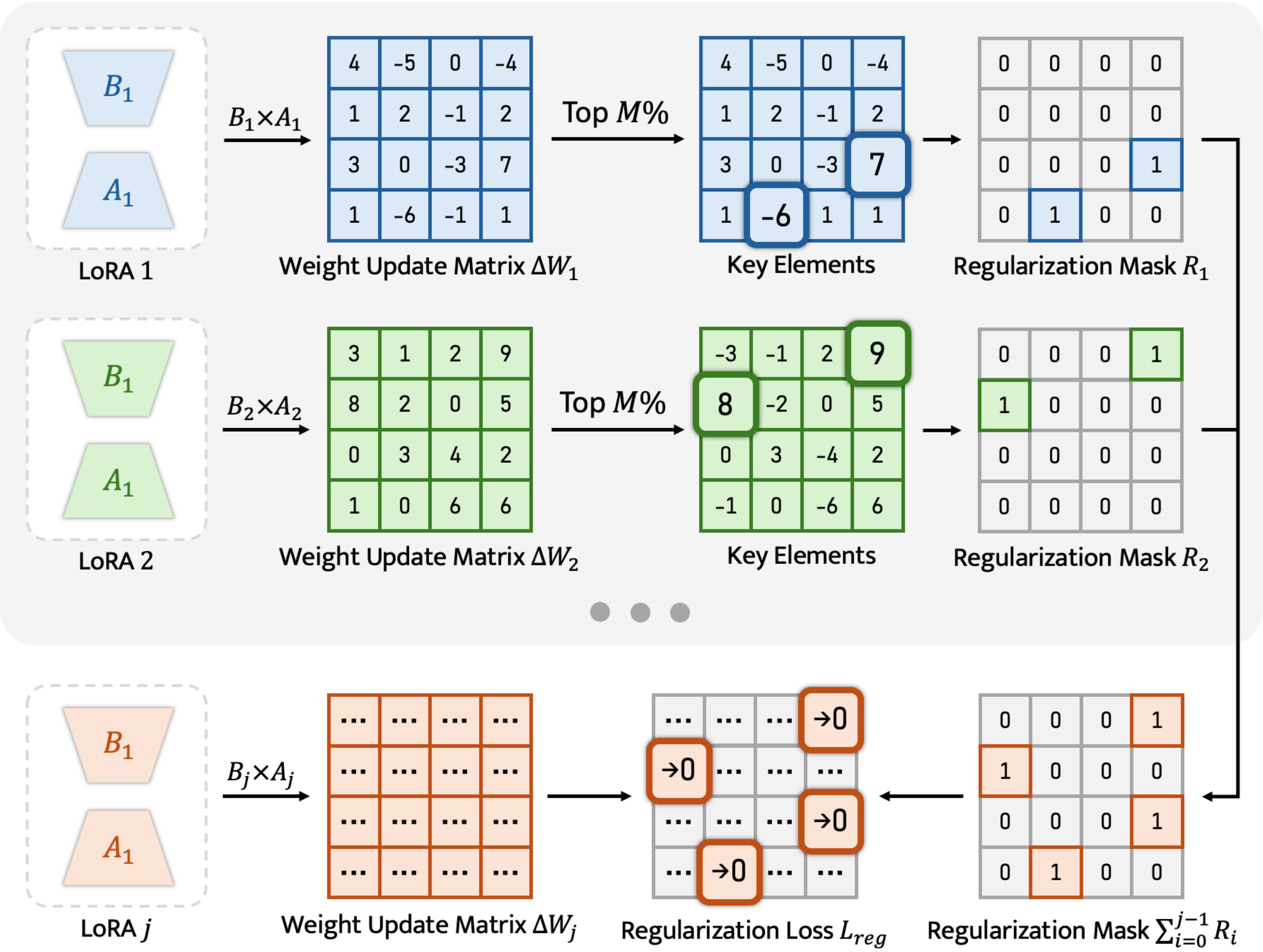
BWT: Backward Transfer

By adding ASD to existing methods, MFN, MAA, and BWT achieve average improvements of 7.00%, 14.63%, and 7.27%, respectively.

RegLoRA

- Although *superficial forgetting* is alleviated by ASD, *essential forgetting*—the true loss of past knowledge—still remains.
- Experiments reveal that only a small subset of parameters change significantly during task learning. These key parameters likely carry most of the task-specific knowledge.
- Therefore, we propose **RegLoRA**, which constrains updates to parameters significantly changed during previous tasks, thereby preserving knowledge of earlier tasks.

RegLoRA



- After each task, a **regularization mask** is saved to identify important elements for that task.
- During future training, updates to all previously identified elements are constrained.

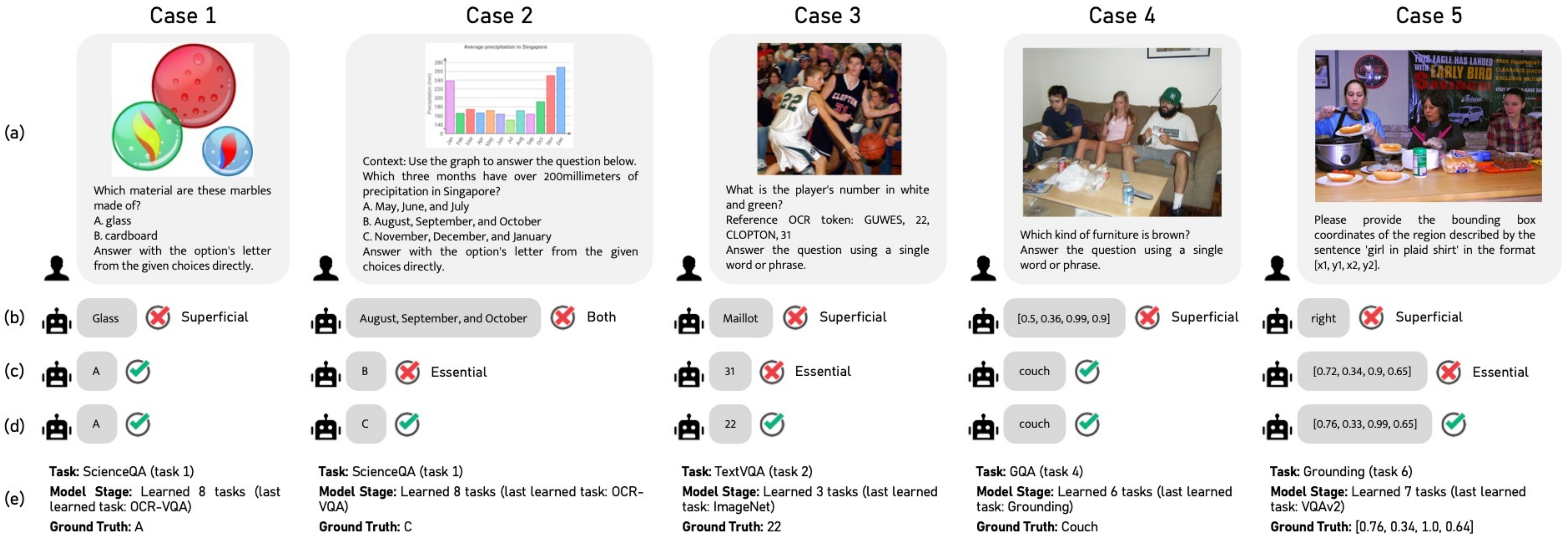
RegLoRA

Configuration	Aggregate Results (%)			
	MFT↑	MFN↑	MAA↑	BWT↑
Baseline (LoRA)	**70.21**	41.59	39.53	-28.62
+ ASD	68.13	47.88	59.71	-20.26
+ ASD + RegLoRA	69.02	**58.57**	**63.04**	**-10.45**

Quantitative Comparison

Method	Accuracy on Each Task (%)								Aggregate Results (%)			
	SQA	*VQA*^{Text}	*ImgNet*	*GQA*	*VizWiz*	*Grd*	*VQA*^{v2}	*VQA*^{OCR}	MFT↑	MFN↑	MAA↑	BWT↓
FFT	2.95	36.38	52.35	46.40	33.90	0.00	61.65	50.00	65.87	35.45	36.73	-30.42
LoRA [20]	54.05	44.63	41.25	47.55	20.80	0.85	59.30	64.30	**70.21**	41.59	39.53	-28.62
O-LoRA [45]	75.40	52.89	71.85	47.30	37.35	7.10	61.85	61.20	<u>69.30</u>	51.87	49.56	-17.43
LoTA [38]	67.30	41.51	8.25	37.15	42.25	0.10	47.95	56.15	54.72	37.58	50.46	-17.14
FFT+ASD	74.50	50.12	65.40	54.35	45.50	0.00	64.40	68.50	68.28	<u>52.85</u>	57.18	-15.44
LoRA+ASD [20]	74.45	49.70	39.30	52.00	50.45	7.05	62.25	47.80	68.13	47.88	<u>59.71</u>	-20.26
O-LoRA+ASD [45]	75.20	55.36	67.50	54.70	52.90	15.40	64.45	35.05	65.59	52.57	61.63	<u>-13.02</u>
LoTA+ASD [38]	76.90	42.65	15.85	40.25	45.10	0.30	54.35	54.00	56.99	41.18	56.28	-15.82
SEFE (Ours)	75.35	58.66	83.10	54.25	48.85	16.75	65.35	66.25	69.02	**58.57**	**63.04**	**-10.45**

Qualitative Comparison



(a) Instruction; (b) Response from the baseline model; (c) Response from the baseline model with ASD added; (d) Response from the baseline model with both ASD and RegLoRA added; (e) Basic information of the case.

Conclusion

- This paper identifies two forgetting types in MCIT—superficial forgetting, where the model's response style becomes biased, and essential forgetting, where factual knowledge is lost.
- To address these issues, we propose the SEFE method, which includes two components: the ASD paradigm and RegLoRA. ASD mitigates superficial forgetting by diversifying question types within tasks, improving response style robustness and knowledge assessment. RegLoRA combats essential forgetting by identifying and regularizing critical weight components across LoRAs to preserve knowledge.
- Experiments demonstrate that both ASD and RegLoRA are effective in tackling their respective forgetting types, and together in SEFE, they achieve state-of-the-art performance in mitigating catastrophic forgetting in MCIT.

山东大学
SHANDONG UNIVERSITY

THANKS FOR WATCHING

陈锦芃 博士

罗宇轩 博士

Sam Kwong 教授（加拿大工程院院士）

—— 学无止境 气有浩然 ——