Online Continual Learning In Image Classification

Zheda Mai

Supervisor: Scott Sanner

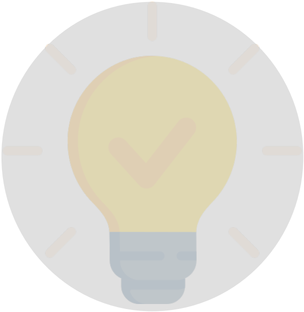
UNIVERSITY OF TORONTO

Zheda’s Continual Learning Journey

Continual Learning?

Competition

Survey



New Idea

Future Work

Zheda's Continual Learning Journey

Continual Learning? Competition Survey New Idea Future Work

1. Why do we need Continual Learning?
2. What's Continual Learning and what's the main challenge?
3. What are popular approaches in this area?

Why do we need Continual Learning

- Numerous data are generated daily on edge devices
- Model performance could be greatly improved by integrating these data
- User data can't always be uploaded to servers for training due to privacy concerns

This necessitates methods that can **continually** learn from streaming data while minimizing **memory** storage and **computation** footprint.

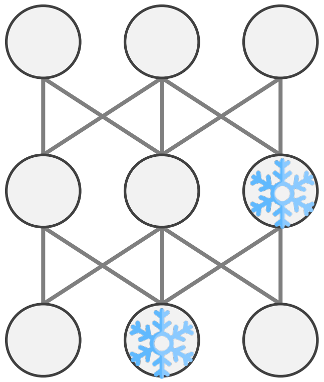
What's Continual Learning

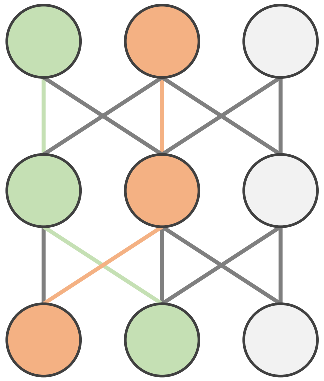
- ***Continual Learning*** (CL) studies the problem of learning from a non-i.i.d stream of data, with the goal of preserving and extending the acquired knowledge over time

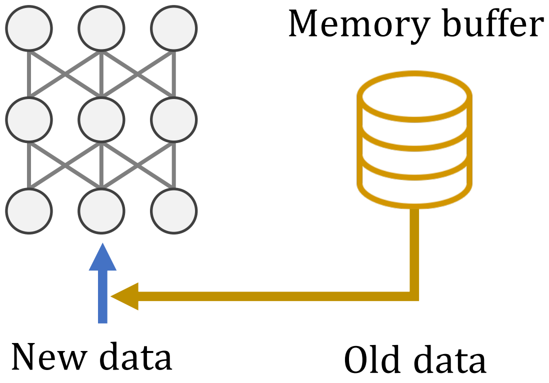
- The main challenge of CL is *catastrophic forgetting,* the inability of a network to perform well on previously seen data after updating with recent data

Continual Learning Approaches

Regularization	Parameter Isolation	Replay







- Constrain the update of key network parameters
- Knowledge Distillation to constrain the output of the network

- Assign per-task parameters
- Often require task-ID

- Memory buffer stores a subset of previous data for replay

Which method works the best?

Zheda's Continual Learning Journey

Continual Learning? | Competition | Survey | New Idea | Future Work

Which method works the best?

CVPR20 Continual Learning Competition

Three challenge tracks

- New instances(NI)

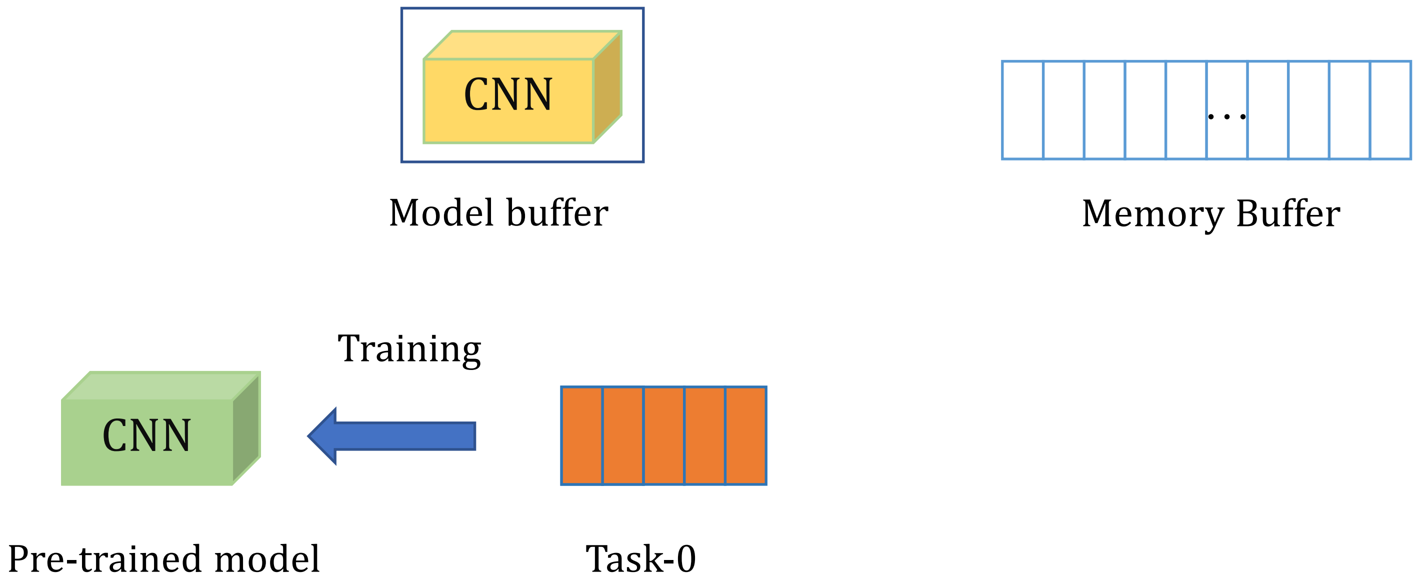
- Multi-Task New classes(NC)

- New instances & classes (NIC)

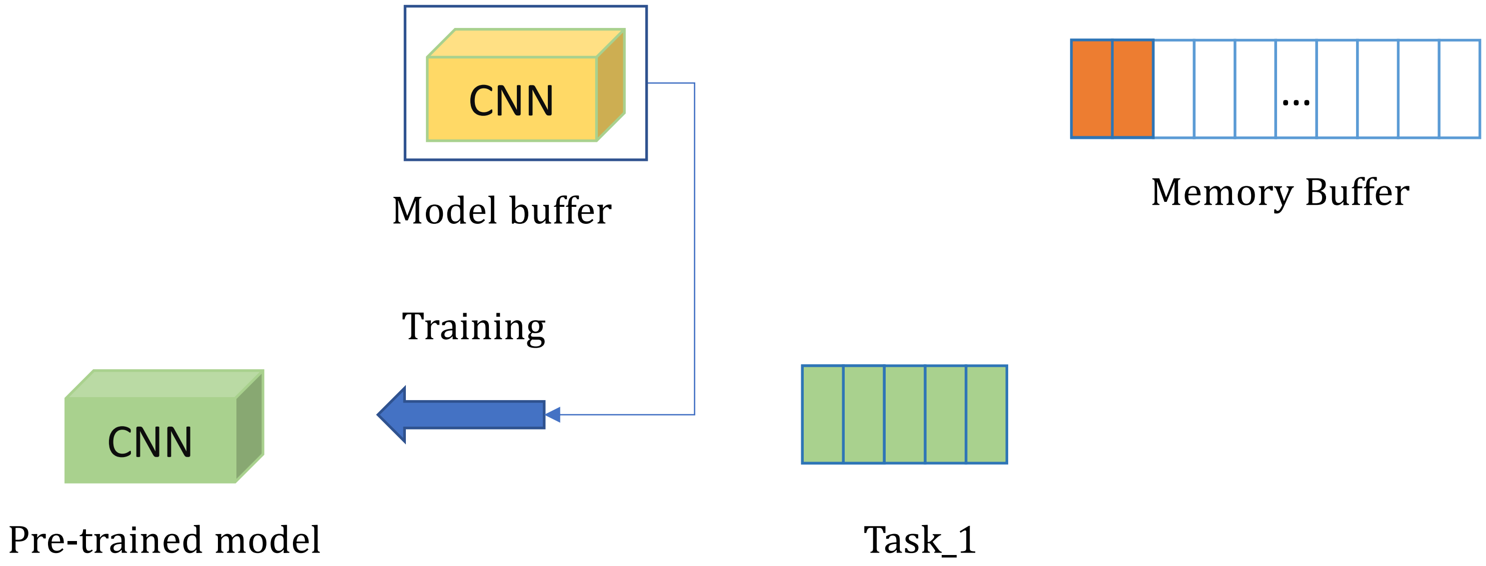
Three challenge tracks

- New instances(NI)
- Multi-Task New classes(NC)
- New instances & classes (NIC)
  - 391 tasks, each one has 300 images of the same class
  - The class can be seen or completely new
  - The model processes tasks sequentially

Batch-level Experience **Replay** with Review



Batch-level Experience **Replay** with Review

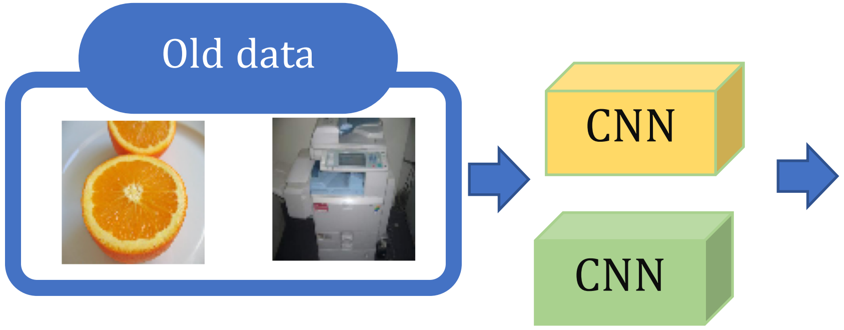


Batch-level Experience **Replay** with Review

Current model

$$\mathcal{L}_{CE}(\mathbf{x}, y) = \sum_{c=1}^{D_{new}+D_{mem}} -\delta_{c=y} \log\left(p_c(\mathbf{x})\right)$$

cross-entropy loss for new data and old data



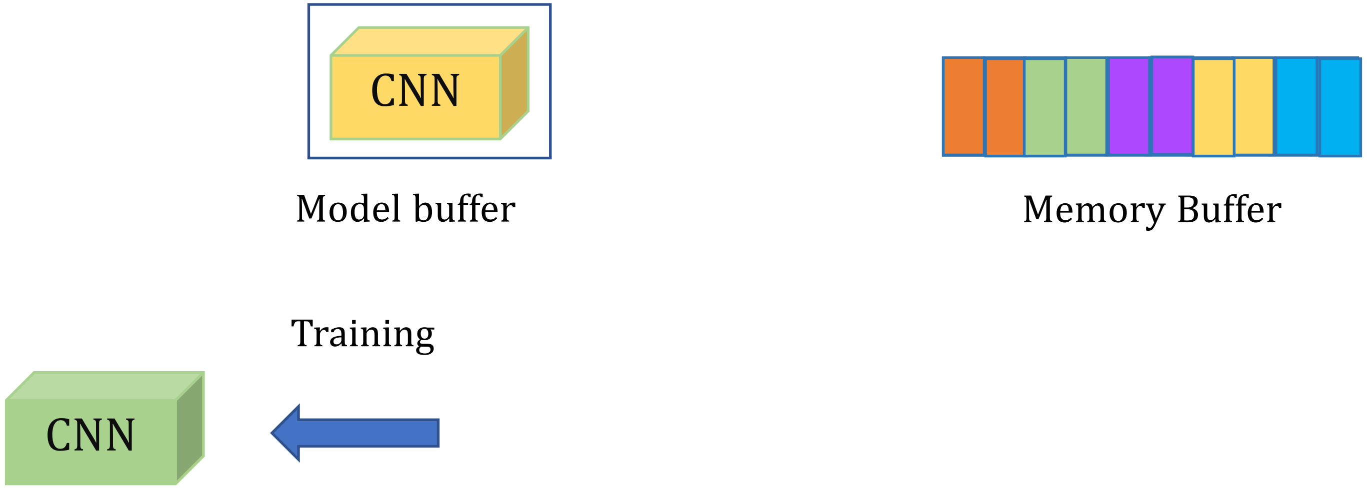
$$\mathcal{L}_{KD}(\mathbf{x}) = \sum_{c=1}^{C_{mem}} -\hat{q}_c(\mathbf{x}) \log\left(q_c(\mathbf{x})\right)$$

knowledge distillation loss for old data

Current & old models

Total Loss $$\mathcal{L}(\mathbf{x}, y) = \mathcal{L}_{CE}(\mathbf{x}, y) + \lambda \mathcal{L}_{KD}(\mathbf{x}) + L_2$$

Batch-level Experience Replay with **Review**



Model buffer

Memory Buffer

Training

Pre-trained model

By the end of training all tasks

Smaller learning rate for review

Final Ranking

TEAM NAME	TEST ACC (%)	VAL ACC_{avg} (%)	RUN_{time} (M)	RAM_{avg} (MB)	RAM_{max} (MB)	$DISK_{avg}$ (MB)	$DISK_{max}$ (MB)	CL_{score}
UT_LG	0.92	0.68	68.67	10643.25	11624.87	0	0	0.694359483
JODELET	0.88	0.64	6.59	15758.62	18169.32	0	0	0.680821395
AR1	0.80	0.58	20.46	8040.47	10092.72	0	0	0.663760006
YC14600	0.91	0.65	64.88	16425.64	19800.48	0	0	0.653114358
ICT_VIPL	0.95	0.68	76.73	2459.31	2459.68	392.1875	562.5	0.61726439
SOONY	0.88	0.63	120.33	14533.97	15763.60	0	0	0.612231922
REHEARSAL	0.75	0.52	22.87	19056.77	23174.11	0	0	0.570829566
JIMIB	0.91	0.74	242.12	17995.61	23765.51	0	0	0.542653619
NOOBMASTER	0.76	0.53	147.59	24714.06	30266.62	0	0	0.464365891
NAÏVE	0.23	0.24	5.16	15763.46	18158.02	0	0	0.32735254
AVG	0.80	0.59	77.54	14539.12	17327.49	39.22	56.25	0.58

Discussion

When I tried to find a method that works well in the competition, it took me a long time ! 😔

Most papers show that their methods surpass others in one specific setting

- What is the setting where each method works the best?
- What are the relative advantages of different tricks?

Zheda’s Continual Learning Journey

Continual Learning? | Competition | Survey | New Idea | Future Work

- What is the setting where each method works the best?
- What are the relative advantages of these tricks?

An Empirical Survey

- Summarized 40 recently proposed approaches
- Empirically scrutinized
  - 9 SOTA methods + 2 baselines
  - 7 simple but effective tricks

Experiment Setup

Small scaled, artificially created

Datasets	Task #	# of classes/task	# of images/class	Image Size
Split CIFAR-100	20	5	500	32x32x3
Split MiniImageNet	20	5	500	84x84x3
CORe50-NC	9	10	2398	128x128x3

Large scaled, designed for CL

Metrics: (1) Average Accuracy, (2) Forgetting, (3)Run time (4) Forward Transfer (5) Backward Transfer

Key Insight 1 – Which one works the best?

Method	Split CIFAR-100			Split Mini-ImageNet			CORe50-NC		
Finetune		3.7 ± 0.3			3.4 ± 0.2			7.7 ± 1.0	
OffLine		49.7 ± 2.6			51.9 ± 0.5			51.7 ± 1.8	
EWC		3.7 ± 0.4			3.5 ± 0.4			8.3 ± 0.3	
LWF		7.2 ± 0.4			7.6 ± 0.7			7.1 ± 1.9	
Buffer Size	M=1k	M=5k	M=10k	M=1k	M=5k	M=10k	M=1k	M=5k	M=10k
ER	7.6 ± 0.5	17.0 ± 1.9	18.4 ± 1.4	6.4 ± 0.9	14.5 ± 2.1	15.9 ± 2.0	23.5 ± 2.4	27.5 ± 3.5	28.2 ± 3.3
MIR	7.6 ± 0.5	18.2 ± 0.8	19.3 ± 0.7	6.4 ± 0.9	16.5 ± 2.1	21.0 ± 1.1	**27.0 ± 1.6**	**32.9 ± 1.7**	**34.5 ± 1.5**
GSS	7.7 ± 0.5	11.3 ± 0.9	13.4 ± 0.6	5.9 ± 0.7	11.2 ± 0.9	13.5 ± 0.8	19.6 ± 3.0	22.2 ± 4.4	21.1 ± 3.5
iCaRL	**16.7 ± 0.8**	19.2 ± 1.1	18.8 ± 0.9	**14.7 ± 0.4**	17.5 ± 0.6	17.4 ± 1.5	22.1 ± 1.4	25.1 ± 1.6	22.9 ± 3.1
AGEM	3.7 ± 0.4	3.6 ± 0.2	3.8 ± 0.2	3.4 ± 0.2	3.7 ± 0.3	3.3 ± 0.3	8.7 ± 0.6	9.0 ± 0.5	8.9 ± 0.6
CN-DPM	14.0 ± 1.7	-	-	9.4 ± 1.2	-	-	7.6 ± 0.4	-	-
GDumb	10.4 ± 1.1	**22.1 ± 0.9**	**28.8 ± 0.9**	8.8 ± 0.4	**21.1 ± 1.7**	**31.0 ± 1.4**	15.1 ± 1.2	28.1 ± 1.4	32.6 ± 1.7

Memory Buffer

In CIFAR-100 & Mini-ImageNet

- iCaRL shows strong performance when M is small
- GDumb dominates when M becomes larger

- iCaRL: Knowledge Distillation + Replay + Nearest Mean Classifier
- Gdumb: trains a classifier from scratch with the memory data only

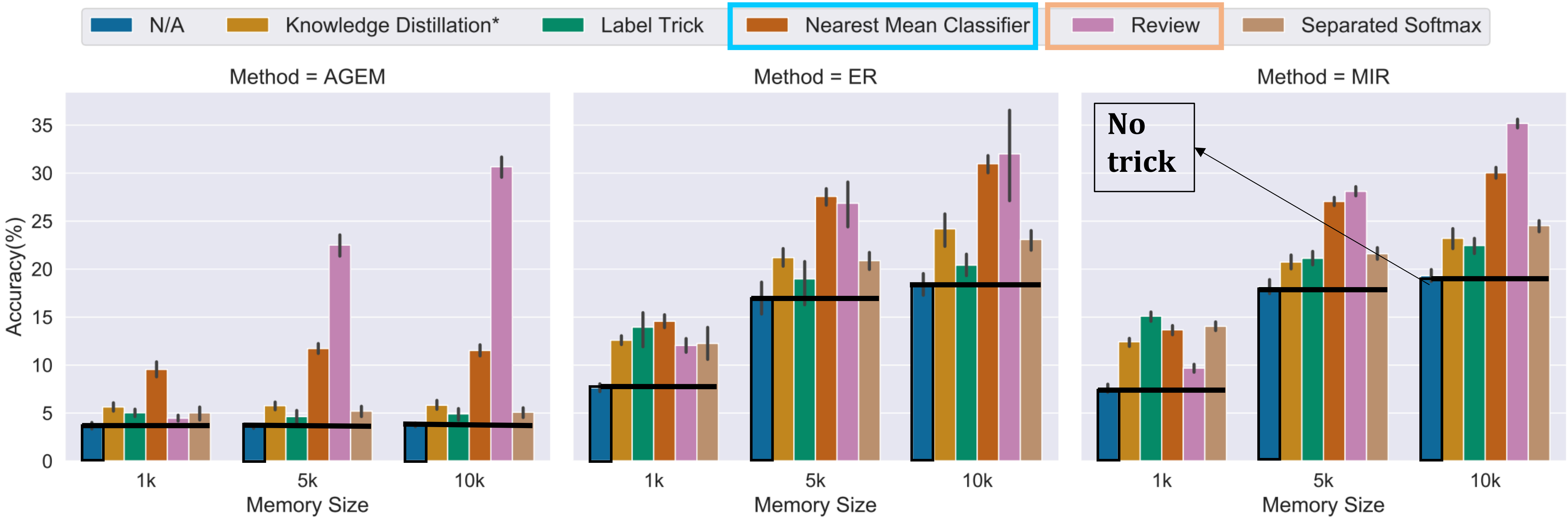
Key Insight 2 – Larger and CL-specific dataset

Method	Split CIFAR-100			Split Mini-ImageNet			CORe50-NC		
Finetune	3.7 ± 0.3			3.4 ± 0.2			7.7 ± 1.0		
OffLine	49.7 ± 2.6			51.9 ± 0.5			51.7 ± 1.8		
EWC	3.7 ± 0.4			3.5 ± 0.4			8.3 ± 0.3		
LWF	7.2 ± 0.4			7.6 ± 0.7			7.1 ± 1.9		
Buffer Size	M=1k	M=5k	M=10k	M=1k	M=5k	M=10k	M=1k	M=5k	M=10k
ER	7.6 ± 0.5	17.0 ± 1.9	18.4 ± 1.4	6.4 ± 0.9	14.5 ± 2.1	15.9 ± 2.0	23.5 ± 2.4	27.5 ± 3.5	28.2 ± 3.3
MIR	7.6 ± 0.5	18.2 ± 0.8	19.3 ± 0.7	6.4 ± 0.9	16.5 ± 2.1	21.0 ± 1.1	**27.0 ± 1.6**	**32.9 ± 1.7**	**34.5 ± 1.5**
GSS	7.7 ± 0.5	11.3 ± 0.9	13.4 ± 0.6	5.9 ± 0.7	11.2 ± 0.9	13.5 ± 0.8	19.6 ± 3.0	22.2 ± 4.4	21.1 ± 3.5
iCaRL	**16.7 ± 0.8**	19.2 ± 1.1	18.8 ± 0.9	**14.7 ± 0.4**	17.5 ± 0.6	17.4 ± 1.5	22.1 ± 1.4	25.1 ± 1.6	22.9 ± 3.1
AGEM	3.7 ± 0.4	3.6 ± 0.2	3.8 ± 0.2	3.4 ± 0.2	3.7 ± 0.3	3.3 ± 0.3	8.7 ± 0.6	9.0 ± 0.5	8.9 ± 0.6
CN-DPM	14.0 ± 1.7	-	-	9.4 ± 1.2	-	-	7.6 ± 0.4	-	-
GDumb	10.4 ± 1.1	**22.1 ± 0.9**	**28.8 ± 0.9**	8.8 ± 0.4	**21.1 ± 1.7**	**31.0 ± 1.4**	15.1 ± 1.2	28.1 ± 1.4	32.6 ± 1.7

For larger and CL-specific dataset, CORe50-NC

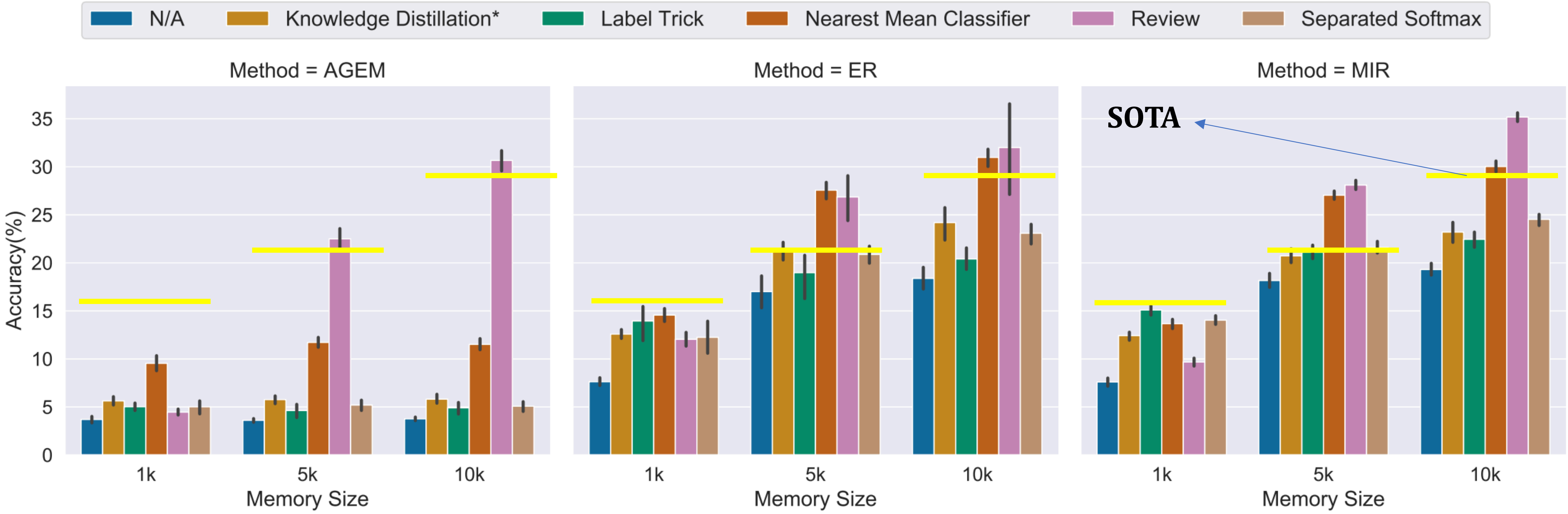
- MIR is the strongest across different M sizes
- MIR: replay based method that carefully selects which samples to replay with the new data

Key Insight 3 - Tricks



- All the tricks improve the base methods
- Two tricks are most effective (1)Nearest Mean Classifier and (2) Review

Key Insight 3 - Tricks



- All the tricks improve the base methods
- Two tricks are most effective (1)Nearest Mean Classifier and (2) Review
- Base methods with tricks outperform SOTA when M is large

Discussion

Replay based methods with memory buffers have show exceptional promise in the competition and the survey

Open question:

Which buffered images to replay, especially when the buffer is small ?

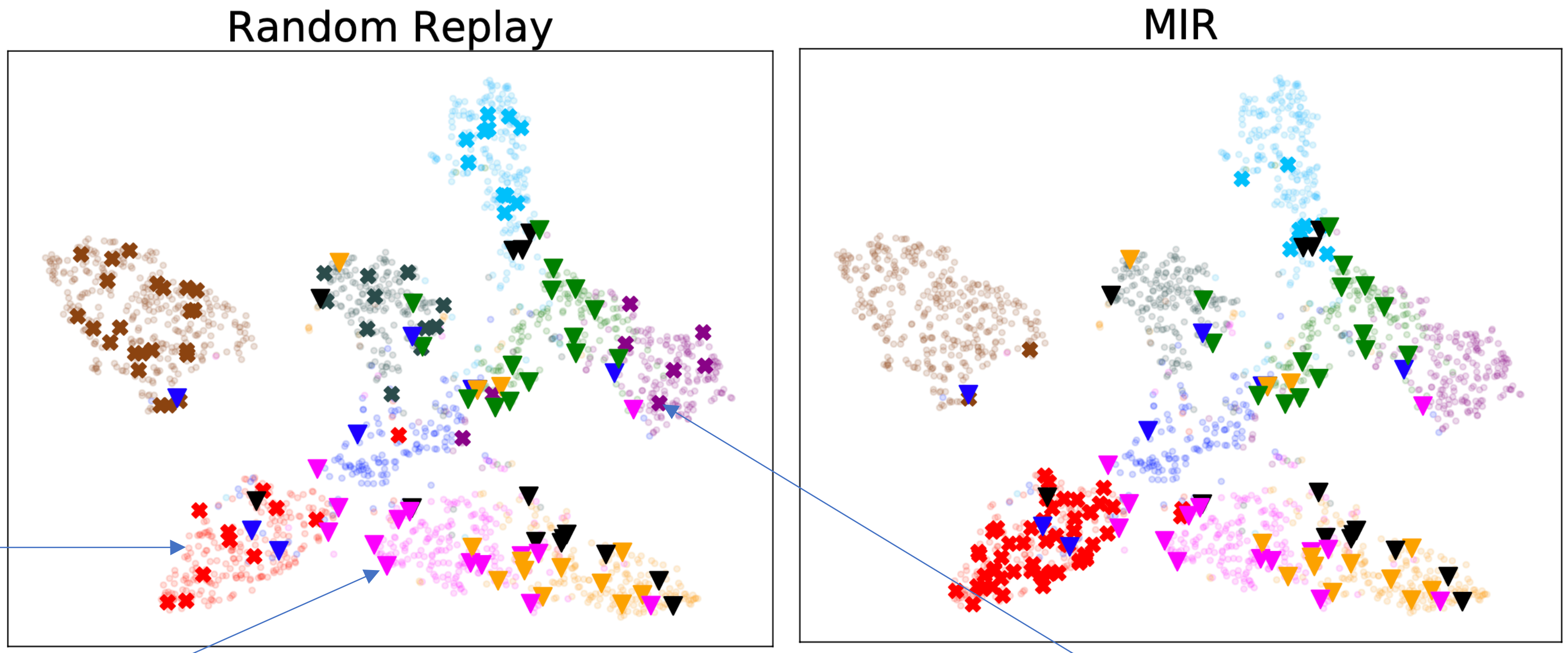
Zheda’s Continual Learning Journey

Continual Learning? | Competition | Survey | New Idea | Future Work

Which buffered images to replay, especially when the buffer is small ?

ASER:
Adversarial Shapley Value Experience Replay

How do existing methods select replay samples (t-SNE)

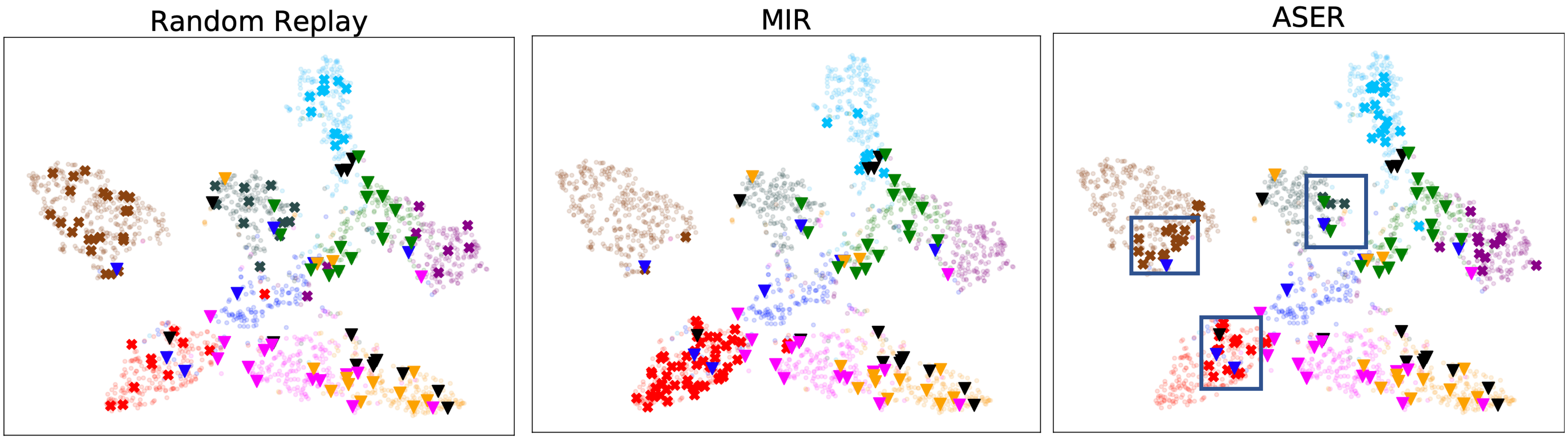


: New task samples
: Buffered samples

: retrieved buffered samples for replay
Color : represents a class

- Random Replay: randomly retrieves samples for replay
- MIR[2] selects samples whose loss most increases after a update with new data

How do existing methods select replay samples (t-SNE)



▼ : New task samples
● : Buffered samples

✖ : retrieved buffered samples for replay
Color : represents a class

- Random Replay: randomly retrieves samples for replay
- MIR[2] selects samples whose loss most increases after a update with new data

ASER strategically retrieves buffered samples that are representative of different classes but also adversarially located near class boundaries and current task samples

Shapley Value

- Shapley value (SV)
- SV for data valuation

Shapley Value

- Shapley Value (SV)
  - Proposed originally in cooperative game theory to fairly distribute total gains to each player
- SV for data valuation

Shapley Value

- Shapley Value(SV)
  - Originally proposed in cooperative game theory to fairly distribute total gains to each player

- SV for data valuation
  - Measure how much of the test accuracy is attributed to a training sample

  - $S_t(i)$ is high -> training sample i is useful for the test accuracy of test set t

ASER: Adversarial Shapley Value Experience Replay

Adversarial Shapley value (**ASV**) for CL memory retrieval to score buffered samples according to their abilities to:

- preserve latent decision boundaries for old classes (to avoid forgetting)
- interfere with latent decision boundaries for new classes (to encourage learning of new class boundaries)

How to quantify these abilities?

ASER: Adversarial Shapley Value Experience Replay

$\mathbf{ASV}_\mu(i)$ gives the buffered sample i a score. We replay buffered samples with high scores.

$$\mathbf{ASV}_\mu(i) = \boxed{\frac{1}{|S_{\text{sub}}|} \sum_{j \in S_{\text{sub}}} s_j(i)} - \frac{1}{b} \sum_{k \in B_n} s_k(i), \;\; \forall i \in \mathcal{M} \setminus S_{\text{sub}} \, ,$$

Sample j is another buffered sample

preservation

To have high **ASV**

- Average of $s_j(i)$ should be high
- Buffered sample i is useful for classification of samples in the memory buffer
- Should be replayed to preserve the old knowledge

ASER: Adversarial Shapley Value Experience Replay

$\mathbf{ASV}_\mu(i)$ gives the buffered sample i a score. We replay buffered samples with high scores.

$$\mathbf{ASV}_\mu(i) = \frac{1}{|S_{\text{sub}}|} \sum_{j \in S_{\text{sub}}} s_j(i) - \boxed{\frac{1}{b} \sum_{k \in B_n} s_k(i),} \; \forall i \in \mathcal{M} \setminus S_{\text{sub}} \,,$$

interference

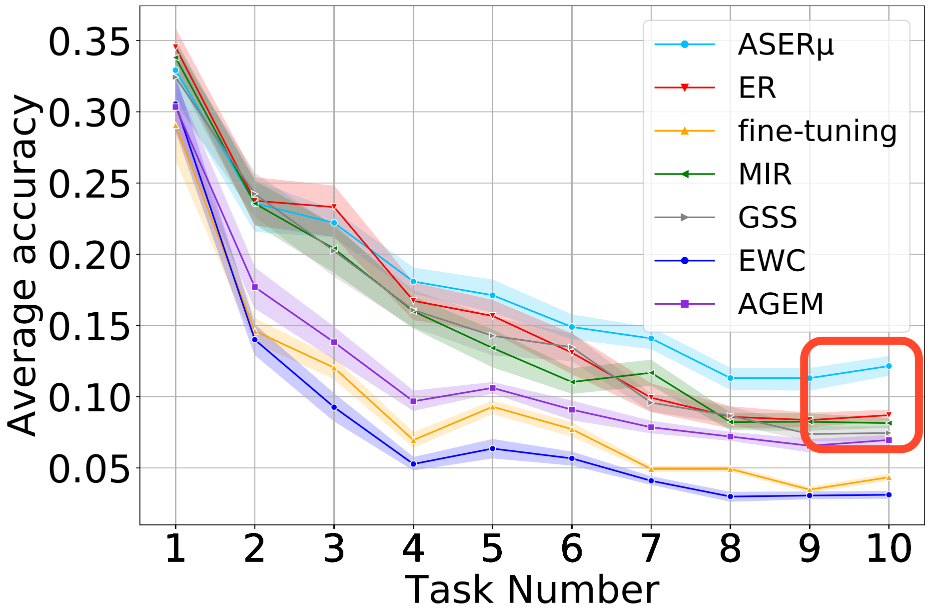
Sample k is a new task sample

To have high **ASV**

- Average of $s_k(i)$ should be negative with large magnitude
- Buffered sample i interferes with new task samples (the model has hard time classifying them)
- Should be replayed to assist the learning of new knowledge

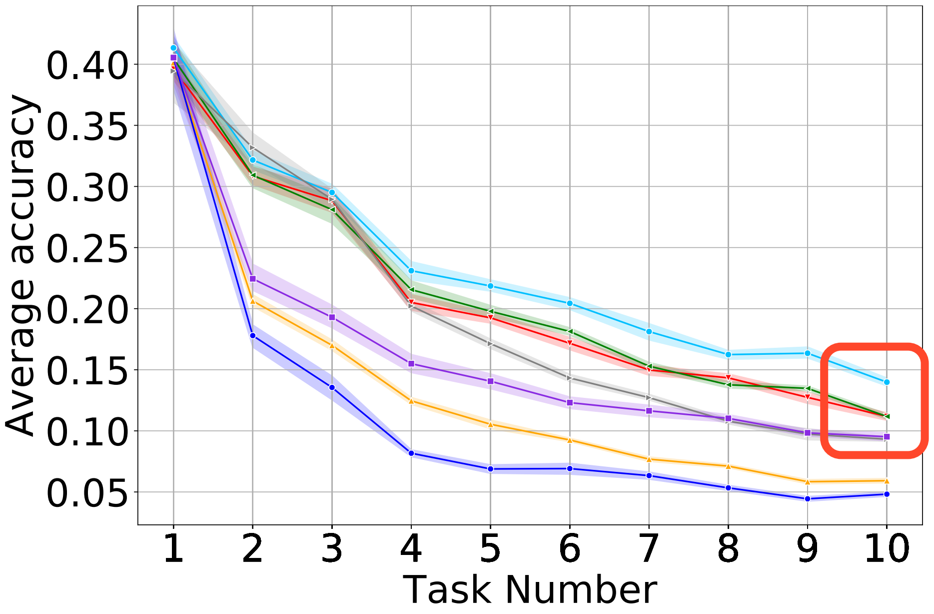
Experiment: results

Mini-ImageNet



8.7%->**12.2%**

CIFAR-100



11.2%->**14.1%**

- Average accuracy on observed tasks with buffer size 1k.
- ASER outperforms other methods when the model sees more tasks

Contributions

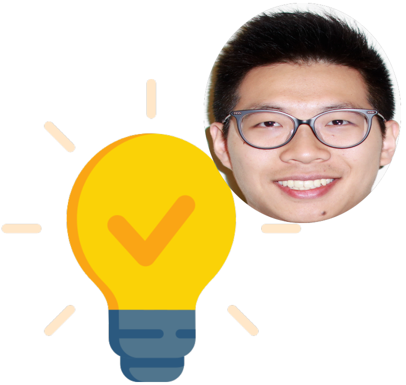
- A simple and efficient continual learning approach and won the competition at CVPR2020
- A comprehensive empirical survey for online continual learning
- A novel and effective way to use Shapaey value adversarially in continual learning to choose replay samples from the memory buffer

Zheda's Continual Learning Journey

Continual Learning?

Competition

Survey



New Idea

Future Work

What's the next step?

Future Work

- More effective way to utilize retrieved samples
  - More sophisticated methods to utilize the retrieved samples
  - Meta-learning is a potential direction
- Supervised contrastive continual learning
  - Nearest Class Mean (NCM) classifier is a competitive substitute for Softmax classifier
  - NCM classifier requires well-separated class embeddings
  - Supervised contrastive loss [8] is a promising direction

Reference

[1] Lesort, T., etc(2019). Regularization shortcomings for continual learning

[2] Aljundi, etc. (2019). Online continual learning with maximal interfered retrieval.

[3] Jia, R., etc. (2019). Efficient task-specific data valuation for nearest neighbor algorithms.

[4] Chaudhry, A., etc (2018). Efficient lifelong learning with a-gem

[5] Chaudhry, etc(2019). On tiny episodic memories in continual learning

[6] Kirkpatrick, J., etc(2017). Overcoming catastrophic forgetting in neural networks

[7] Aljundi, R., etc(2019). Gradient based sample selection for online continual learning.

[8] Khosla, P., etc(2020). Supervised contrastive learning.