EVOLVE: Enhancing Unsupervised Continual Learning with Multiple Experts

Xiaofan Yu[1], Tajana Rosing[1], Yunhui Guo[2]

[1] University of California San Diego [2] University of Texas at Dallas

Our code is available at: https://github.com/Orienfish/EVOLVE

Introduction

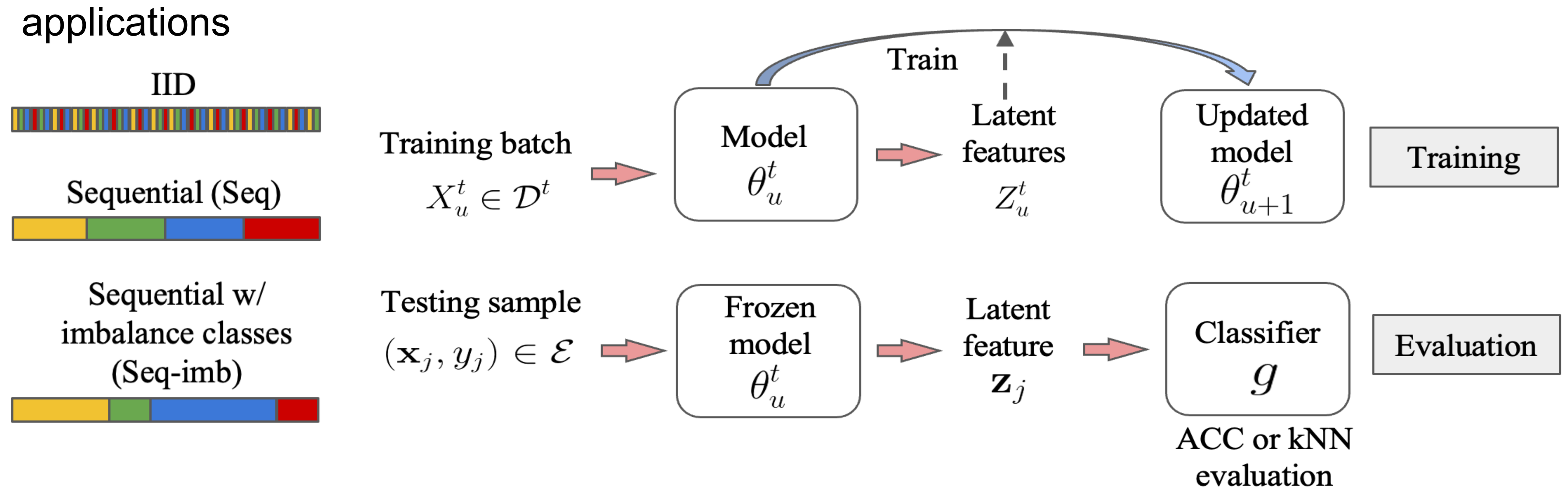
- While great progress has been made in continual learning, it is still challenging to deploy the existing algorithms in the wild to learn over time in a real-world application
- The barrier primarily stems from two factors:
  - The unpredictable streaming input
  - The lack of supervision and prior knowledge
- Most existing works in unsupervised continual learning rely on various prior knowledge to produce good results

Papers	Single-pass	Non-*iid*	No task labels	No class labels
VASE [2], CURL [61], L-VAEGAN [77]	×	✓	✓	✓
He *et al.* [31], CCSL [46], CaSSLe [24], LUMP [49]	✓	✓	×	✓
Tiezzi *et al.* [70], KIERA [57]	✓	✓	✓	×
STAM [68]	✓	✓	✓	✓

- We aim at closing the gap towards real-world continual learning

Problem Definition: Unsupervised Continual Learning (UCL)

- Online unsupervised continual learning without prior knowledge
  - Non-iid and single-pass data streams
  - No task or class labels
  - No prior knowledge, e.g., task/class shift boundaries
- We consider three different types of *class-incremental* streams inspired from real-world applications

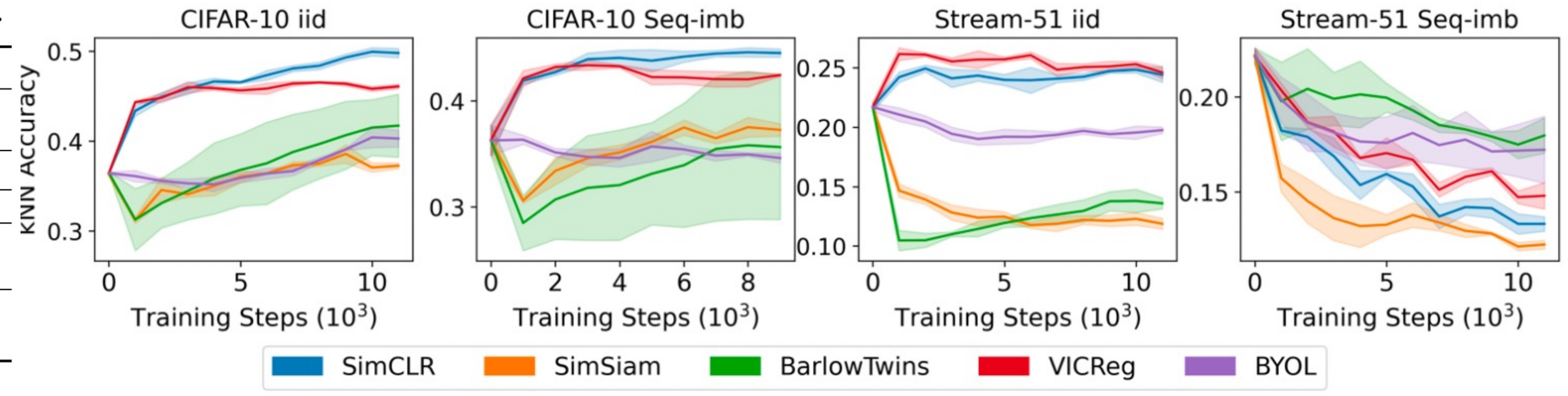


An Empirical Study of Existing Self-Supervised Learning (SSL) Baselines

- Recent studies have indicated that combining SSL with memory replay holds great promise for continual representation learning in the wild [1]
- We conduct empirical study of existing SSL methods with memory replay
  - **Datasets:** CIFAR-10 (image), Stream-51 (video) [2];
  - **Data streams:** iid vs. Seq-imb
- Our results show that SSL baselines experience a **significant accuracy drop** when applied on video datasets (with temporal correlation) with Seq-imb data streams, thus impeding their practical utility for real-world applications

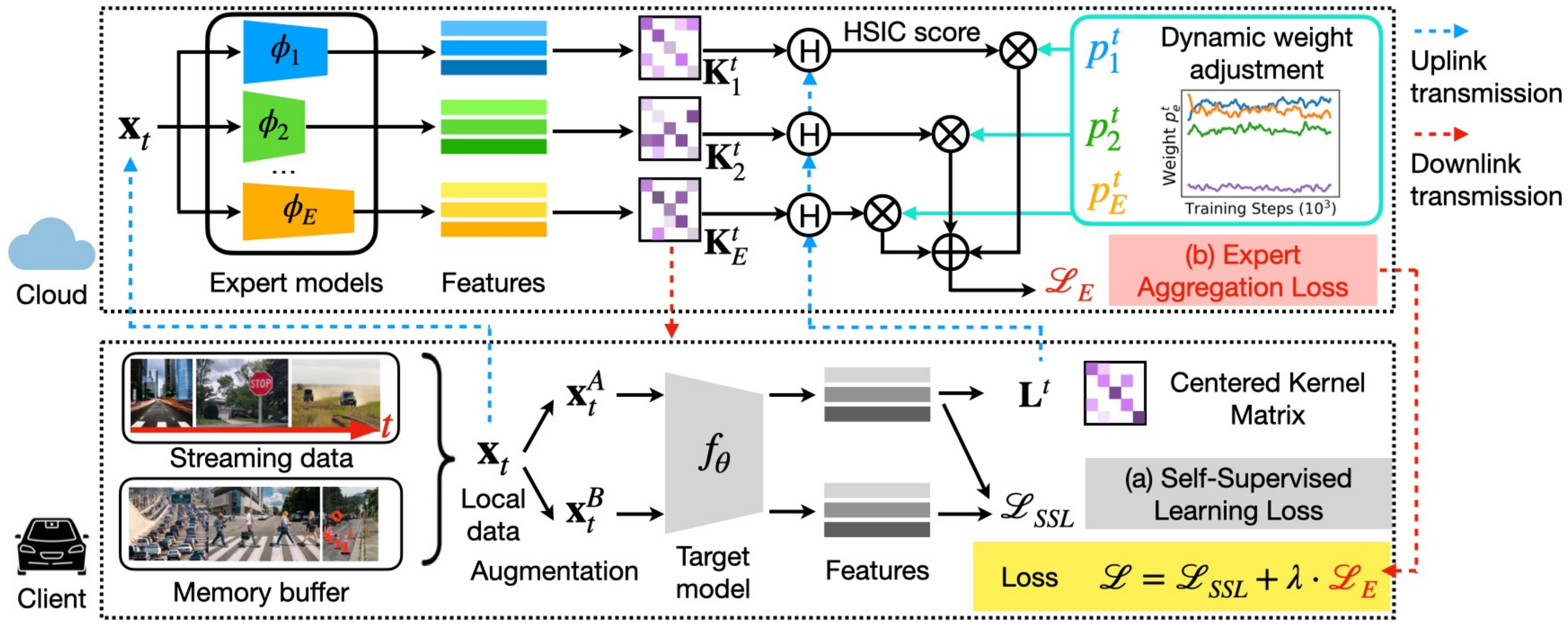
Table 1. Overview of state-of-the-art SSL methods and losses.

Methods	Loss	Loss Function $\mathcal{L}_{SSL}$
SimCLR [20]	InfoNCE	$-\log \frac{\exp(\mathbf{z}_i^A \cdot \mathbf{z}_j^B/\tau)}{\sum_{k \neq i} \exp(\mathbf{z}_i^A \cdot \mathbf{z}_k/\tau)}$
BYOL [35]	MSE	$\left\|\mathbf{q}_t^A - \mathbf{z}_t^B\right\|_2^2$
SimSiam [23]	MSE	$\mathcal{D}(\mathbf{q}_t^A, \mathbf{z}_t^B)$
Barlow Twins [92]	Cross-Correlation	$\sum_u (1-\mathcal{C}_{uu})^2 + \psi \sum_u \sum_{v \neq u} \mathcal{C}_{uv}^2$
VICReg [7]	MSE + Variance + Cross-Correlation	$\psi s(\mathbf{z}_t^A, \mathbf{z}_t^B) + \mu v(\mathbf{z}_t^A) + \nu c(\mathbf{z}_t^A)$



Our Method: EVOLVE

- Our essential idea is to enhance UCL in a wild environment with diverse pretrained models treated as *experts*
- We propose EVOLVE, a hybrid UCL framework with (1) SSL training on the local client and (2) expert-guided training on the cloud, transmitting a small set of data and intermediate features to the cloud



Two key designs of EVOLVE

1. Expert Aggregation Loss based on Hilbert-Schmidt independence criterion (HSIC)

$$\mathcal{L}_E = -\sum_{e=1}^{E} \underbrace{p_e^t}_{\text{Expert weights}} \cdot \mathrm{HSIC}(\mathbf{K}_e^t, \mathbf{L}^t),$$

$$HSIC(X,Y) = HSIC(K,L) = \frac{1}{(n-1)^2} Tr(KHLH),$$

$$H = I - \frac{1}{N} 11^T$$

2. Dynamic weight adjustment based on a confidence metric, updated in moving average

$$q_e^t = \sum_i \frac{\exp(\mathbf{h}_{e,i}^A \cdot \mathbf{h}_{e,i}^B/\tau)}{\sum_{k \neq i} \exp(\mathbf{h}_{e,i}^A \cdot \mathbf{h}_{e,k}^A/\tau) + \sum_k \exp(\mathbf{h}_{e,i}^A \cdot \mathbf{h}_{e,k}^B/\tau)},$$

$$w_e^{t+1} = \alpha w_e^t + (1-\alpha) q_e^t. \qquad p_e^t = w_e^t / \sum_l w_l^t$$

- Key advantages of EVOLVE:

Expert-guided learning helps adapt in natural and unpredictable environments

The hybrid scheme avoids the high computational costs induced by running experts on clients

Experimental Setup

- **Datasets:** CIFAR-10, TinyImageNet, CORe50 [3], Stream-51 [2]
- **UCL baselines:**
  - Synaptic Intelligence (SI) [ICML'17]
  - Progressive Neural Network (PNN) [arXiv'16]
  - Dark Experience Replay (DER) [NeurIPS'20]
  - CaSSLe [CVPR'22]
  - Lifelong Unsupervised Mixup (LUMP) [ICLR'22]
- **Experts:** pretrained ResNet-50, Swin Transformer
- **Metrics:** kNN accuracy, linear evaluation accuracy

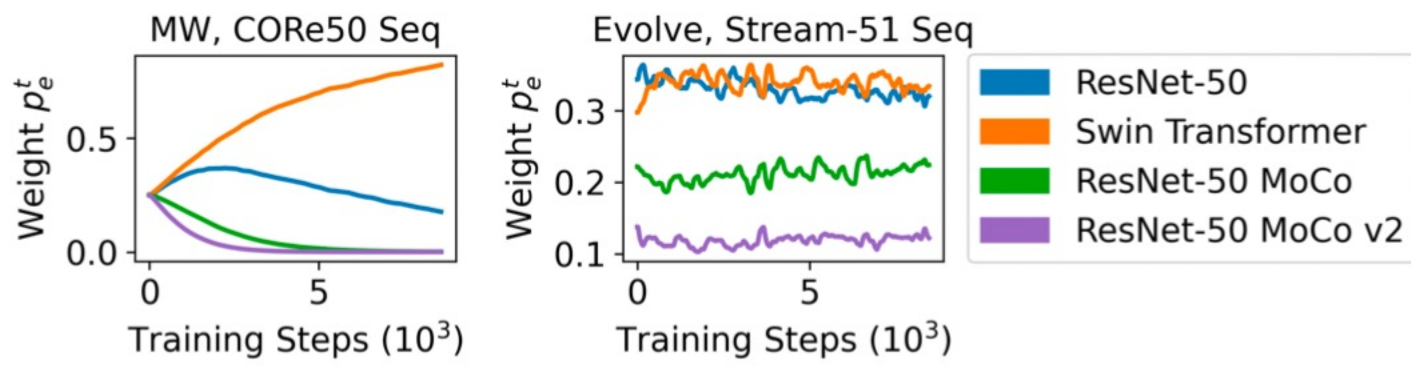
Results

- **Comparison with existing UCL baselines:** EVOLVE outperforms the top baseline using the same SSL by 3.6-20.0% in kNN accuracy and 6.1- 53.7% in top-1 linear evaluation accuracy across diverse data streams.

The final kNN accuracy and linear evaluation accuracy on Stream-51

Method	*k*NN Accuracy(↑)					Linear Evaluation Accuracy(↑)				
	SimCLR	**BYOL**	**SimSiam**	**BarlowTwins**	**VICReg**	**SimCLR**	**BYOL**	**SimSiam**	**BarlowTwins**	**VICReg**
SSL	14.0±0.4	17.1±0.1	13.1±1.0	18.1±0.4	14.8±0.3	33.1±1.5	27.7±2.6	11.9±3.5	51.4±1.1	40.8±0.2
SI	12.9±0.5	17.0±1.0	12.3±0.6	12.2±0.8	11.1±0.4	21.3±1.9	27.0±8.3	13.2±5.9	26.2±0.2	23.5±2.0
PNN	12.0±0.2	17.1±1.1	12.5±0.5	12.7±1.2	11.6±0.8	13.5±0.6	29.9±0.1	9.8±0.9	26.9±4.6	25.4±0.1
DER	13.6±0.6	16.0±0.4	14.4±1.3	13.0±1.2	10.7±0.4	31.5±1.5	37.5±2.4	28.0±5.0	28.9±0.1	24.0±1.7
CaSSLe	14.7±0.9	26.5±1.6	21.9±0.8	16.7±0.3	12.6±2.0	7.5±3.2	27.3±5.0	20.6±6.1	10.0±1.2	38.5±2.4
LUMP	20.5±0.7	14.5±0.5	12.7±0.1	13.9±0.5	20.7±1.3	48.2±0.1	27.2±0.8	8.4±0.1	16.9±3.4	55.1±1.5
EVOLVE	**30.1±1.6**	**31.6±1.3**	**31.5±1.7**	**30.1±1.7**	**24.8±0.4**	**82.2±0.9**	**84.4±1.0**	**81.7±1.0**	**75.7±2.4**	**61.2±1.7**

- **Comparison with other weight update policies for using the experts:** the commonly used Multiplicative Weight Update (MW) [4] in online optimization can converge to extremes, while EVOLVE yields a dynamic pattern



References

[1] Purushwalkam, Senthil, et al. "The challenges of continuous self-supervised learning." *ECCV'22*.

[2] Roady, Ryne, et al. "Stream-51: Streaming classification and novelty detection from videos." CVPRW'20

[3] Lomonaco, Vincenzo, and Davide Maltoni. "Core50: a new dataset and benchmark for continuous object recognition." PMLR'17

[4] Littlestone, Nick, and Manfred K. Warmuth. "The weighted majority algorithm." Information and computation 108.2 (1994): 212-261.