SEOUL NATIONAL UNIV.
VISION & LEARNING

Continual Learning on Noisy Data Streams via Self-Purified Replay

Chris Dongjoo Kim*, Jinseo Jeong* , Sangwoo Moon and Gunhee Kim

Seoul National University

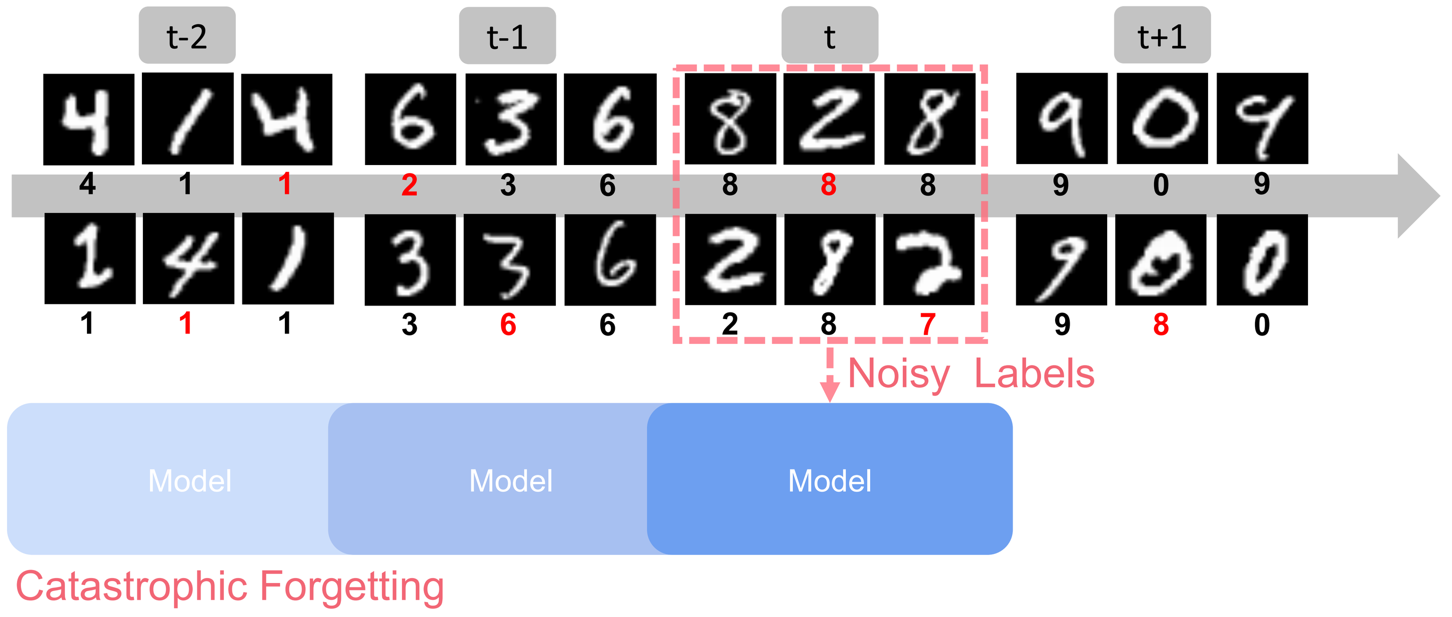
*Equal Contribution

2021 ICCV OCTOBER 11-17 VIRTUAL

Code available at http://vision.snu.ac.kr/projects/SPR

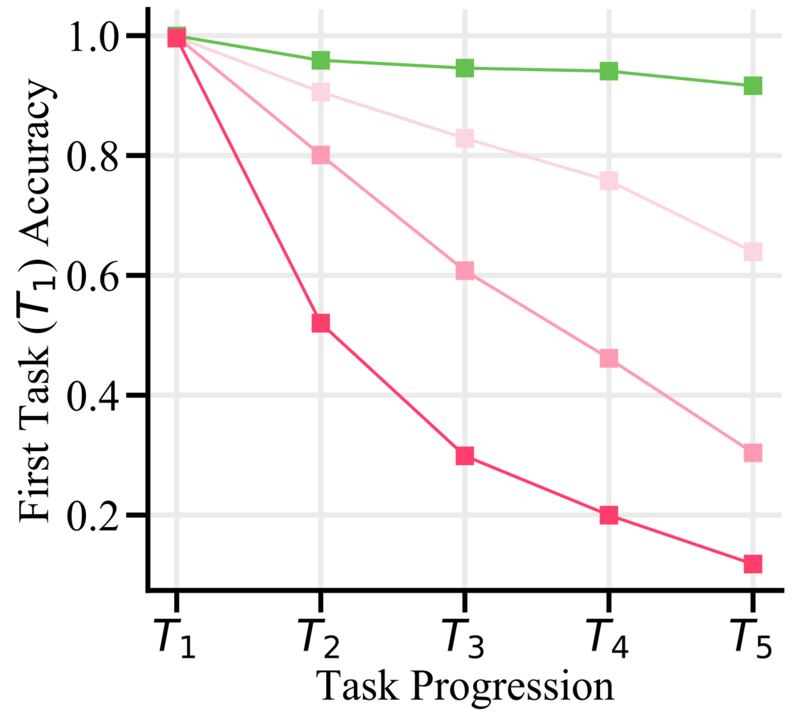
Introduction

- Noisy Labels & Continual Learning are inevitable real-world machine learning problems which are bound to converge.
- First work to explore this intersection.



Motivation

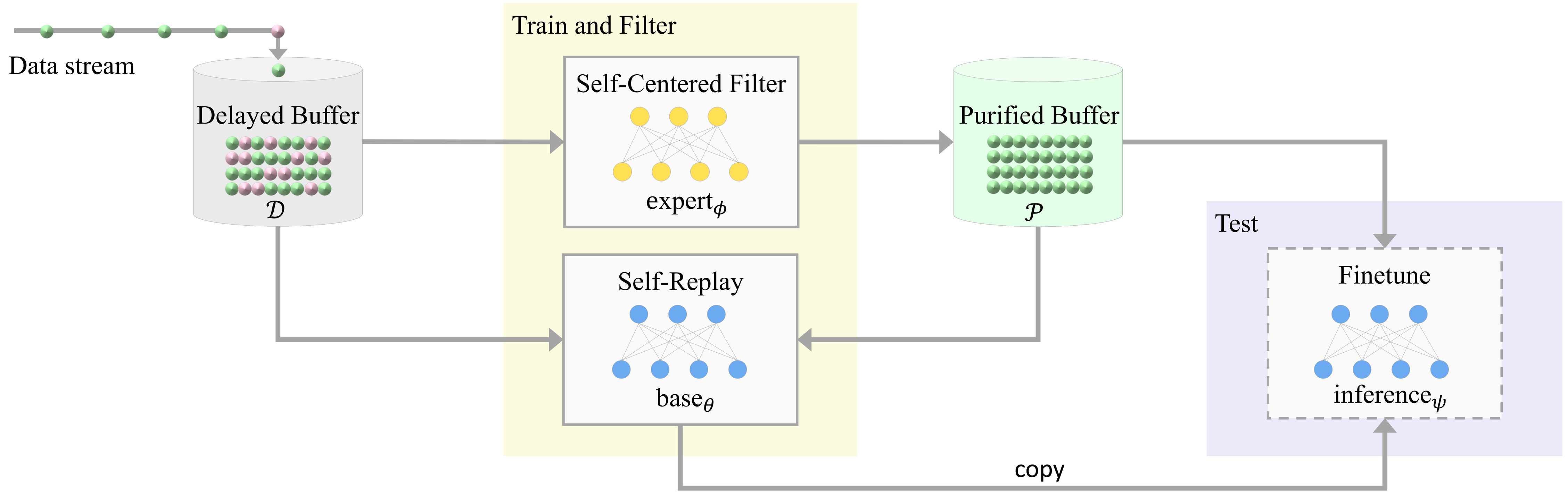
- Replay based continual learning from a noisy data stream.
- Fatal amounts of forgetting with increasing amounts of noise.



Approach

- **Goal**. Continually learn from a stream of noisy labeled data
- Break into two interrelated sub-goals:
  1. **Reduce forgetting even with noisy labels.** Catastrophic forgetting must be mitigated amidst noisy labels.
     - ✓ Self-Replay
  2. **Filter clean data.** Noise should be identified from *small portions* of data to generalized to online continual learning.
     - ✓ Self-Centered Filter
- *Self-Replay + Self-Centered Filter = Self-Purified Replay*

Self-Purified Replay Framework



- Delayed buffer $\mathcal{D}$: temporarily stocks the incoming data stream.
- Purified buffer $\mathcal{P}$: maintains the cleansed data.
- Base network θ : addresses sub goal 1 via self-supervised replay (Self-Replay) training.
- Expert network ϕ : tackles sub goal 2 by obtaining confidently clean samples via centrality

▪ Self-Replay

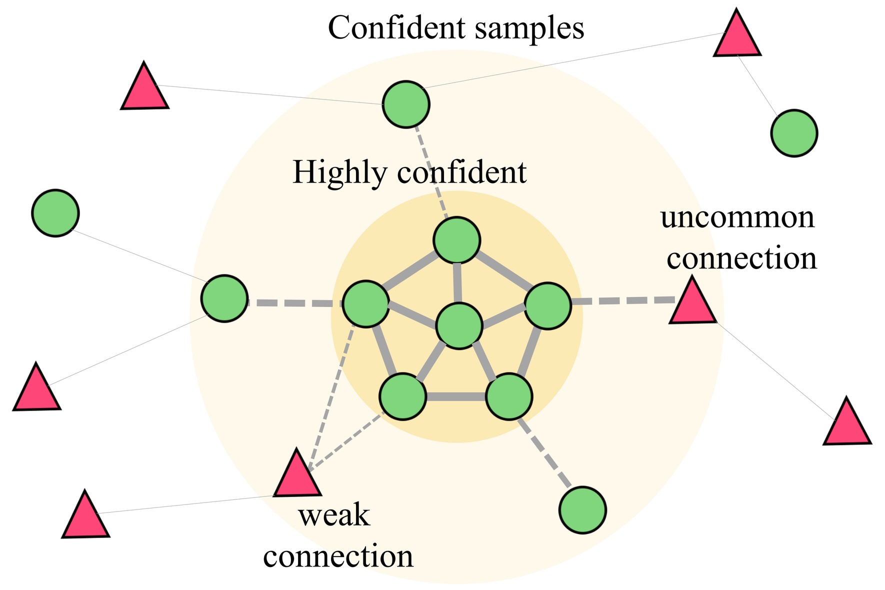
Self-supervised replay for continually relevant and rich representations.

$$L_{self} = -\sum_{i=1}^{2(B_D+B_P)} \log \frac{e^{u_i^T u_j/\tau}}{\sum_{k=1}^{2(B_D+B_P)} \mathbb{1}_{k\neq i} e^{u_i^T u_k/\tau}}$$

- **circumvent error signals** via learning only from x (without y) using contrastive self-supervised learning techniques.
- **mitigate forgetting** while learning general representations via self-supervised replay of the samples in the delayed and purified buffer ($\mathcal{D} \cup \mathcal{P}$).

▪ Self-Centered-Filter

Filter clean data (using only Delayed Buffer contents).



1. Representation learning
   - self-supervised training only using the delayed buffer.

$$L_{self} = -\sum_{i=1}^{2(B_D)} \log \frac{e^{u_i^T u_j/\tau}}{\sum_{k=1}^{2(B_D)} \mathbb{1}_{k\neq i} e^{u_i^T u_k/\tau}}$$

2. Centrality scoring
   - eigenvector centrality based on similarity between the representations.

$$c_v = \frac{1}{\lambda}\sum_{u\in N(v)} c_u = \frac{1}{\lambda}\sum_{u\in V} a_{v,u} c_u$$

3. Probabilistic discrimination
   - EM algorithm to fit a beta mixture model to the centrality scores.

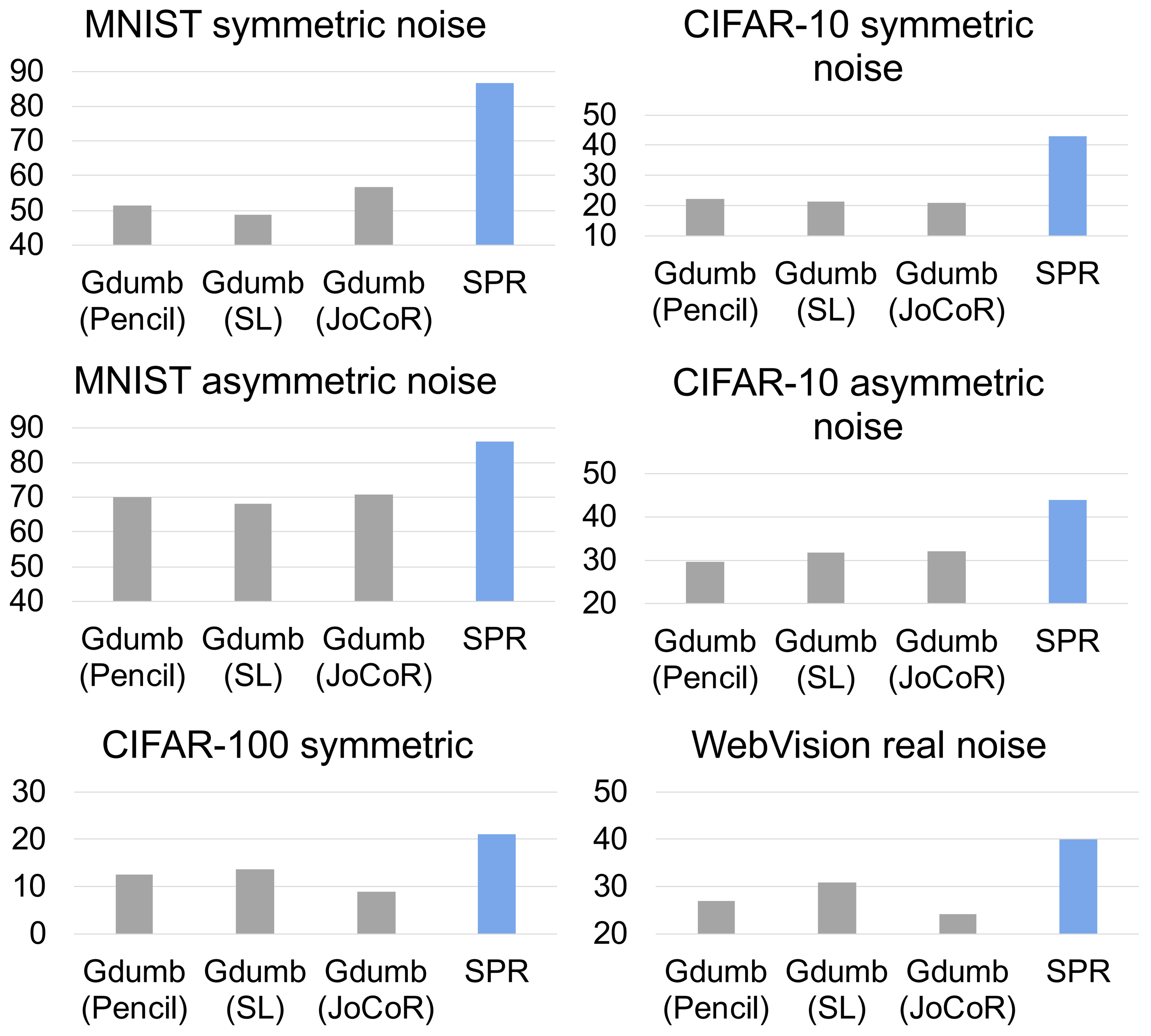
$$p(z|c) = \frac{\pi_z p(c|\alpha_z,\beta_z)}{\sum_{j=1}^{Z} \pi_j\, p(c|\alpha_j,\beta_j)}$$

4. Stochastic Ensembles
   - Monte Carlo sampling to approximate robust posterior $p(z|D_l)$.

$$p(z|D_l) \propto \int_A p(z|\text{cent}(A))dp(A|D_l)$$

$$A = (a_{v,u})_{|V|\times|V|}$$

Overall Accuracy



Filtered Percentage

Below results are based on 40% noise rates