Rethinking Experience Replay: a Bag of Tricks for Continual Learning

P. Buzzega, M. Boschini, A. Porrello, S. Calderara
AImageLab - University of Modena and Reggio Emilia, Modena, Italy | pietro.buzzega@unimore.it

ICPR 2020 — 25th International Conference on Pattern Recognition — Milan, Italy 10 | 15 January 2021

Motivation

In **Continual Learning** (CL), we train models that **retain past knowledge** while **learning new tasks**

$$\underset{\theta}{\operatorname{argmin}} \sum_{t=1}^{t_c} \mathcal{L}_t, \quad \mathcal{L}_t \triangleq \mathbb{E}_{(x,y)\sim D_t}\Big[\ell(y, f_\theta(x))\Big].$$

Rehearsal CL methods use a memory buffer $\mathcal{B}$ to store and replay previously seen examples. The simplest solution in this class is **Experience Replay** (ER), which interleaves replay items with the current training batch.

$$\mathcal{L}' = \mathbb{E}_{(x,y)\sim \mathcal{D}_{t_c}}\Big[\ell(y, f_\theta(x))\Big] + \mathbb{E}_{(x,y)\sim \mathcal{B}}\Big[\ell(y, f_\theta(x))\Big].$$

However, ER is affected by some drawbacks:

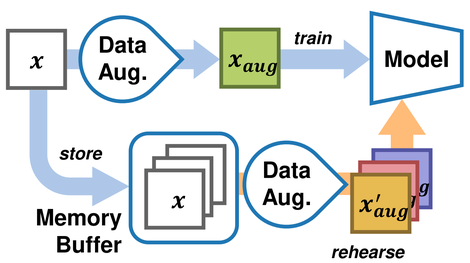
- repeated optimization of a small buffer: **overfitting**;
- ◆ implicit **bias** of the network towards **newer tasks** [3];
- ■ **reservoir sampling** [2] is not always ideal (*e.g.*: if the buffer is small, entire classes could be left out).

We address this issues by applying some **tricks**.

Tricks

● Independent Buffer Augmentation (IBA)

We store not augmented input items in $\mathcal{B}$ and augment them **independently** when drawn for replay.



◆ Bias Control (BiC)

As done in [3], we add a **bias correction layer** to the model which compensates the k^{th} output logit o_k with learned parameters α, β as follows:

$$q_k = \begin{cases} \alpha \cdot o_k + \beta & \text{if } k \text{ was trained in the last task} \\ o_k & \text{otherwise} \end{cases}$$

BiC is trained at the **end of each task** on $\mathcal{B}$.

Tricks

◆ Exponential LR Decay (ELRD)

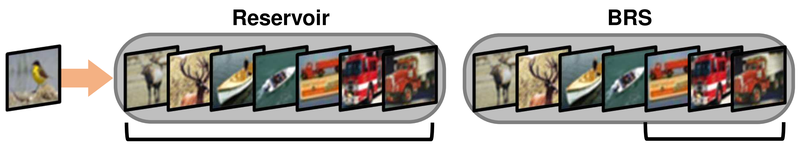
We progressively **slow learning down** in later tasks. Given an initial learning rate lr_0, we set it to

$$lr_j = lr_0 \cdot \gamma^{N_{ex}}$$

for the j^{th} example, where N_{ex} is the number of examples seen so far, and γ is a hyper-parameter s.t. $\gamma \approx 6^{-1/N_{ex}}$.

■ Balanced Reservoir Sampling (BRS)

Probability of *reservoir* leaving $\geqslant 1$ class out of $\mathcal{B} \approx 36.7\%$ (for $|\mathcal{B}| \approx C$). We propose a simple modification to it: inserted samples **must replace** a random item from the **most represented class**.

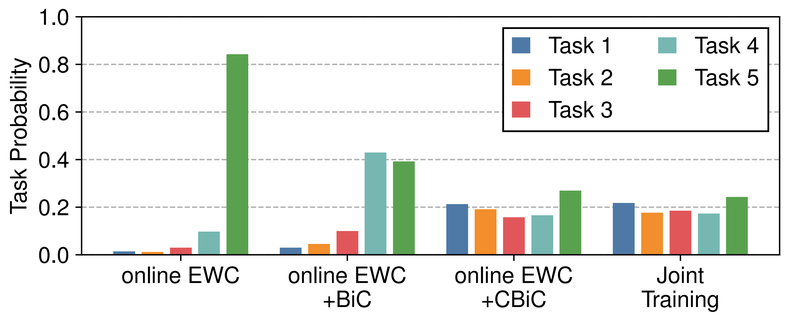


■ Loss-Aware Reservoir Sampling (LARS)

We additionally modify *reservoir* to retain the **most meaningful examples** depending on its corresponding **training loss** (similar to GSS [1], but much faster to compute).

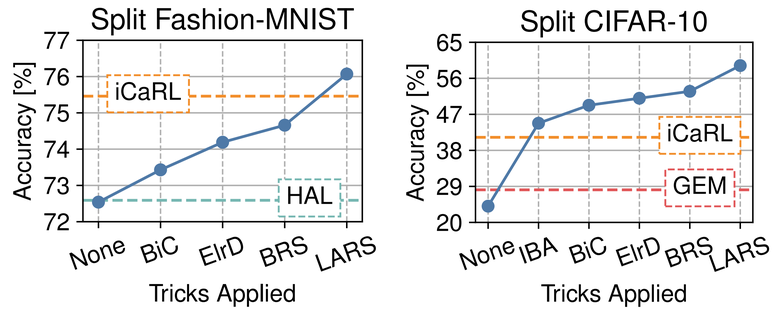
Applicability to other methods

IBA can be easily applied to other **rehearsal methods**.
BiC and **ElrD** are also effective in reducing the bias towards the last task for **Regularization CL methods**.
To better account for the discrepancy among old tasks in them, we modify BiC to apply **distinct offsets** to logits from each task (Complete Bias Correction – CBiC).

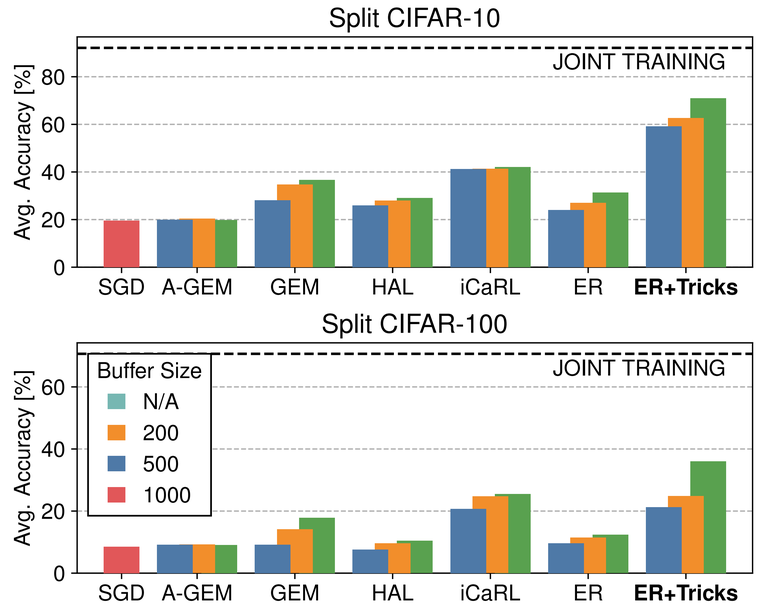


Results

The incremental application of the proposed tricks enhances the final accuracy of ER over competing methods.



ER with tricks outperforms state-of-the-art reharsal methods on multiple settings at multiple buffer sizes.



References

[1] R. Aljundi, M. Lin, B. Goujaud, and Y. Bengio. Gradient based sample selection for online continual learning. *NeurIPS 2019*.

[2] J. S. Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software*, 11, 1985.

[3] Y. Wu, Y. Chen, L. Wang, Y. Ye, Z. Liu, Y. Guo, and Y. Fu. Large scale incremental learning. *CVPR 2019*.