I Fought the Low

Decreasing Stability Gap with Neuronal Decay

Author:
Kirill Zhankov

Supervisors:
Gido van de Ven
Tom Viering

Contact me:
k.zhankov@student.tudelft.nl
linkedin.com/in/kzhankov

TU Delft

1. Introduction

Deep artificial neural networks sometimes need to learn new tasks (say, the same set of images but rotated) throughout their lifetime, without relearning from scratch [1]. This is known as continual learning (CL). In this context, Stability gap (Figure 1) refers to a dip in performance you may see when the network switches to learning a new task – it temporarily forgets how to do the previous tasks [2]. This is something we want to avoid in to keep our applications safe and train efficiently [2-4].

Neuronal decay (ND) is a regularization method that modifies the loss function in a way to encourage the model to remain sparse, i.e. keep small activations (Figure 2) [5]. Smaller activations (presumably) lead to more capacity left for future tasks. Previously, ND was not assessed on stability gap, so we tried ND to see if it helps to decrease the gap and at what cost.

2. Neuronal Decay

Modify the loss function to account for the activation magnitude:

Loss function | Cross-entropy | Neuronal decay

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{ND}$$

$$\mathcal{L}_{ND} = \sum_{x \in B} \sum_{m \in M} \sum_{i \in N^{(m)}} \left(f^{(m)}(x)_i \right)^2$$

Go over:

every image x in batch B
every hidden layer $m \in M$
every neuron $i \in N^{(m)}$

Apply activation function

3. Research Questions

Q1. Does inclusion of neuronal decay reduce the stability gap, compared to the baseline that uses replay but not decay?
Q2. Can neuronal decay on its own (with no replay) outperform the baseline that uses replay but not decay?
Q3. Is there a significant computational overhead associated with using neuronal decay?

4. Methodology

Metrics: gap depth GD in each interval (percentage points) and time-to-recover TTR relative to the length of the interval (%), see Figure 1. To answer Q3, we analytically computed and compared he number of multi-accumulate operations (MACs) and run a profiler.

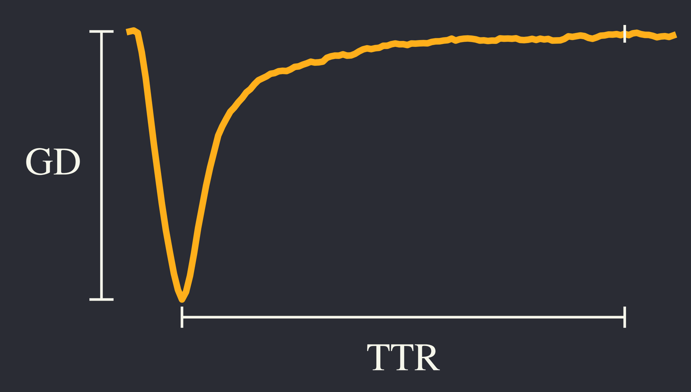

Figure 1: An example of how gap depth GD (in p.p.) and time-to-recover TTR (expressed as the percentage of the total number of batches in the task) metrics are computed for the accuracy curve with a stability gap.

Architecture: a simple vanilla multi-layer perceptron, suitable for the chosen dataset (Figure 2).

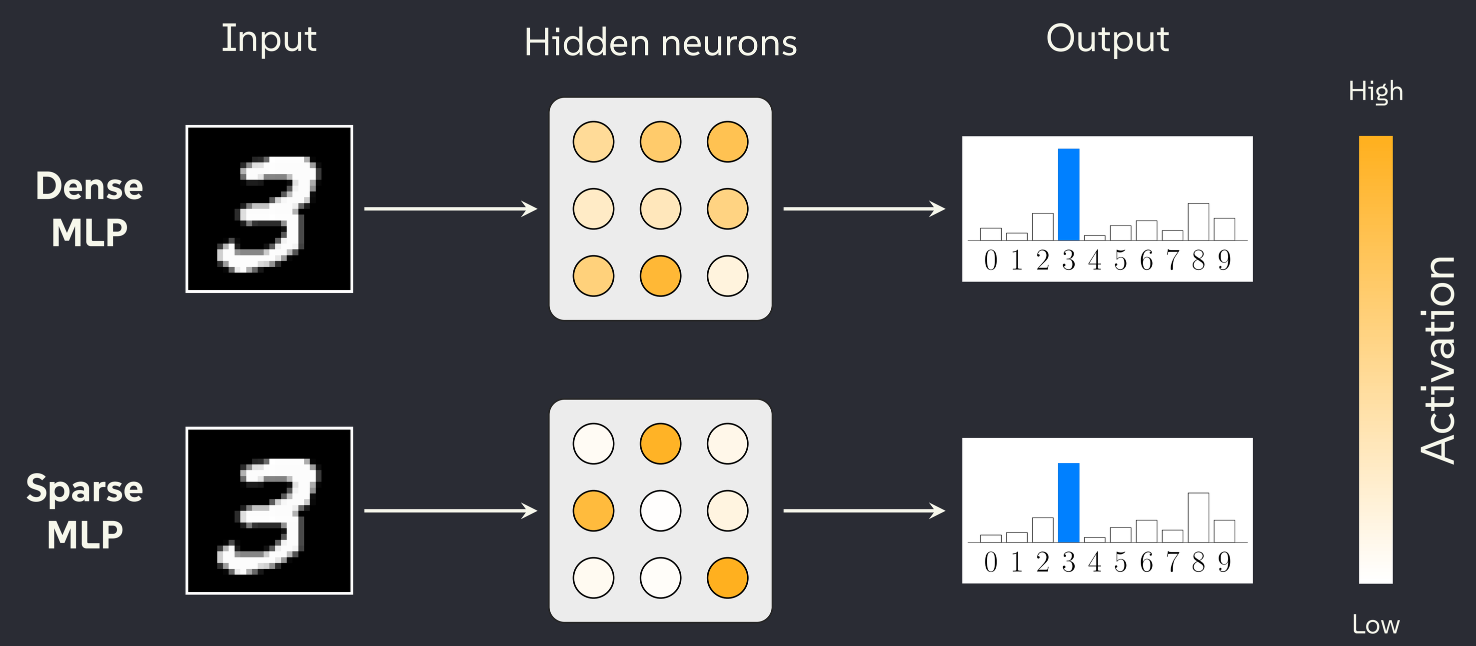

Figure 2: a dense network with many active neurons and a sparse network with few active neurons. Both networks achieve comparable performance, yet neuronal decay method prefers the sparser network.

Dataset: Rotated MNIST, grayscale images of handwritten digits with different rotations applied to them (Figure 3).

Figure 3: A single training sample showing the three tasks used in the study. Each task corresponds to a distinct rotation.

Setup: to answer the research questions, we

1. Set up a baseline with the best state-of-the-art method (full replay) on a multi-layer perceptron.
2. Introduced the neuronal decay.
3. Compared the results visually and with metrics.

5. Results

- We found a decrease in gap depth when using ND (Table 1). TTR was similar, although there was too much variance to draw a conclusion.
- Higher values of lambda were associated with smaller depth but also lower accuracy (Figure 4).
- ND used 0.007% more MACs for the chosen size. Increase in time per batch was 6.66% and 8.56% for CPU and CUDA respectively (Table 1).
- ND without replay performed better than the baseline without replay but a lot worse than the baseline with replay (result not shown).

Table 1: Task 1 gap depth GD, Task 1 time-to-recover TTR, average accuracy ACC, computed in the Task 3 interval and average CPU and CUDA training time per batch for baseline and neuronal decay (ND) models. The ND model shows a decrease in GD while sacrificing some accuracy and, on average, spends slightly more time per batch for both the CPU and CUDA events.; TTR varies greatly in both models.

Model	$\mathtt{GD}_{1,3}$ (p.p.) ↓	$\mathtt{TTR}_{1,3}$ (%) ↓	$\mathtt{ACC}_{\mathrm{AVG},3}$ (%) ↑	CPU time (ms) ↓	CUDA time (ms) ↓
Baseline	16.4 ± 1.4	$\mathbf{15.0 \pm 6.4}$	$\mathbf{97.67 \pm 0.17}$	$\mathbf{18.01 \pm 0.61}$	$\mathbf{14.60 \pm 0.16}$
ND	$\mathbf{5.5 \pm 1.0}$	16.1 ± 9.6	97.04 ± 0.16	19.21 ± 1.20	15.85 ± 0.48

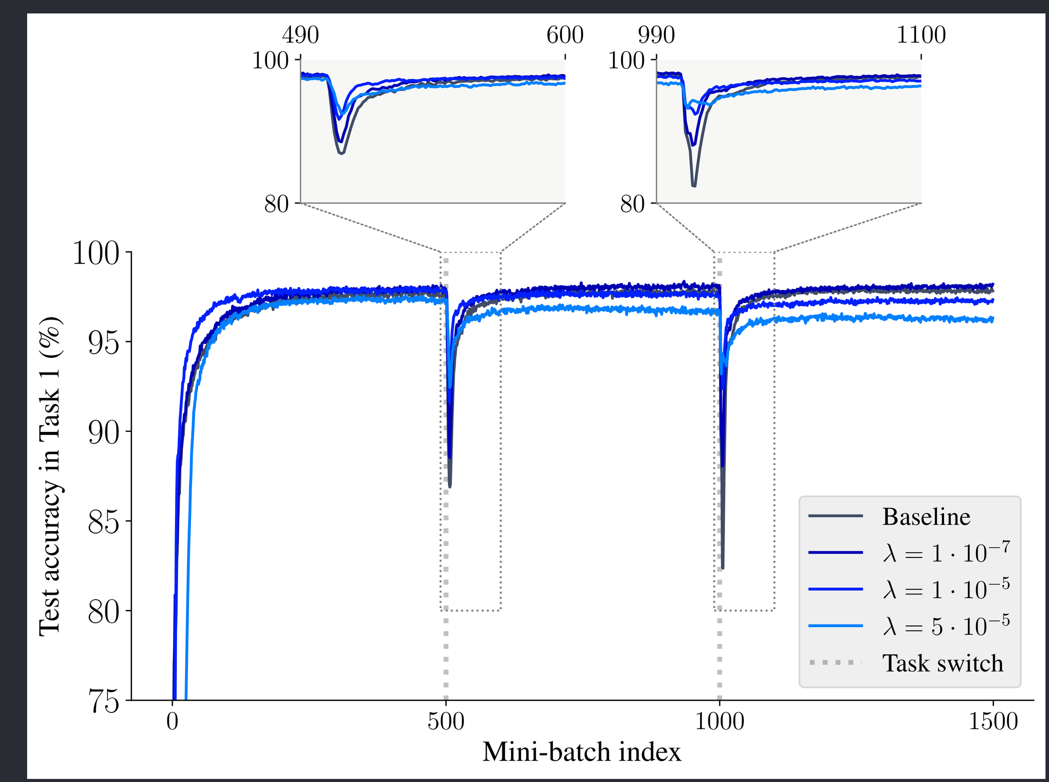

Figure 4: Test accuracy (%) for the baseline model and three neuronal decay models with different values of coefficient lambda. Lower values of lambda result in smaller gap depth but preserve a higher level of accuracy.

6. Future Work

- Getting more conclusive results for the TTR metric
- Testing the approach in CNNs and transformers (for example)
- Assessing the model's complexity quantitatively to gain insights

7. Conclusion

- ND is a solid way to reduce the stability gap and a good candidate for scenarios where adequate worst-case performance is vital.
- ND was not powerful enough to mitigate the stability gap on its own.
- ND introduced little computational overhead during training – a property that can be very desirable in CL.
- The proper choice of hyperparameters (especially, lambda) is crucial.

References
[1] G. M. van de Ven, T. Tuytelaars, and A. S. Tolias "Three types of incremental learning," Nature Machine Intelligence, vol. 4, no. 12, pp. 1185–1197, 2022.
[2] M. D. Lange, G. M. van de Ven, and T. Tuytelaars, "Continual evaluation for lifelong learning: Identifying the stability gap," in The Eleventh Int. Conf. Learn. Representations, 2023.
[3] T. Hess, T. Tuytelaars, and G. M. van de Ven, "Two complementary perspectives to continual learning: Ask not only what to optimize, but also how," in Proc. of the 1st ContinualAI Unconference ser. Proc. of Machine Learning Research, vol. 249, PMLR, 2024, pp. 37–61.
[4] S. Kamath, A. Soutif-Cormerais, J. Van De Weijer, and B. Raducanu, "The expanding scope of the stability gap: Unveiling its presence in joint incremental learning of homogeneous tasks," in Proc. of the IEEE/CVF Conf. on Comput. Vis. and Pattern Recognit. Workshops, 2024, pp. 4182–4186.
[5] R. O. Malashin and M. A. Mikhalkova, "Avoiding catastrophic forgetting via neuronal decay," in 2024 Wave Electronics and its Application in Information and Telecommunication Systems, 2024, pp. 1–6.