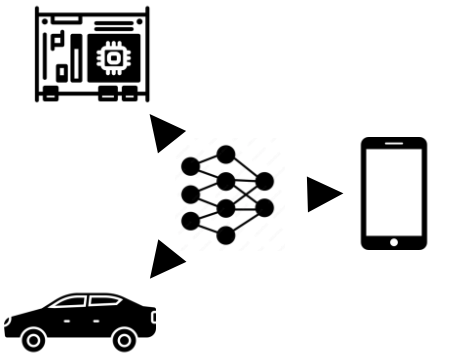
Towards On-device Learning on the Edge: Ways to Select Neurons to Update under a Budget Constraint

Aël Quélennec Enzo Tartaglione Pavlo Mozharovskyi Van-Tam Nguyen

{name.surname}@telecom-paris.fr

Motivations – Porting AI to edge devices

- Improved computational resources have expanded AI use cases, enabling **larger neural networks**.
- Lower microcontroller costs have increased embedded device adoption.
- Combining microcontrollers and deep neural networks is challenging due to **resource constraints**.
- Current research focuses on **compressed and accelerated models** for efficient MCU inference.
- Recent research worldwide is addressing **on-device learning** complexities.



Method – Dynamically selecting the fastest neurons

Our work draws inspiration from MIT's pioneering paper on On-Device Learning [1] and the Neurons at Equilibrium algorithm [2]. For each epoch, we compute the output values of every neuron over a fixed validation subset and use these outputs to calculate the "velocity" of each neuron. In our approach, we dynamically update the k fastest neurons, ensuring that the corresponding number of updated parameters stays within a specific budget. Achieving small values of k is possible as we fine-tune pre-trained networks. This enables our models to maintain high test accuracies while ensuring that the update cost never exceeds a predetermined memory budget.

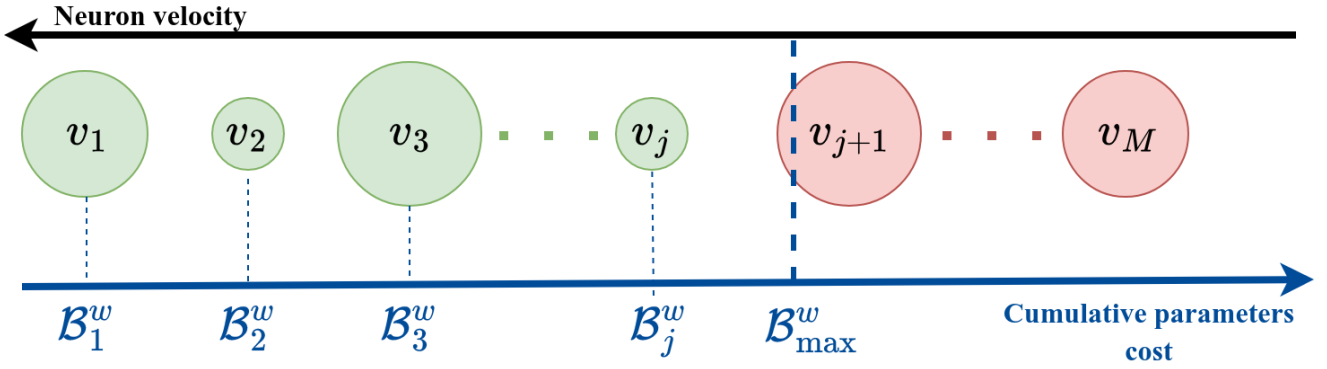


Figure 1. Selection of neurons to update given a network of M neurons and a budget $\mathcal{B}^w_{\max}$. The cost $\mathcal{C}^w_i$ is proportional to the size of the circle which represents the i-th neuron. The neurons selected for the update are in green, while those frozen in red.

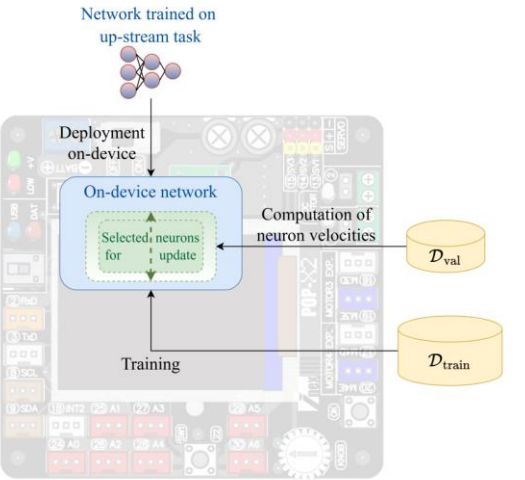


Figure 2. Overview for the on-device learning when dynamic neuron selection is applied.

Experiments and results

- Our proposal is the first to dynamically select a sub-network for updates, contrasting with a baseline of randomly selecting neurons until the maximum budget is reached.
- We compare a static method, Sparse Update (SU), with the two dynamic methods introduced. In most cases, Velocity achieves better accuracies than the other methods, while Random competes with SU, albeit falling slightly below.
- The results indicate that a well-designed dynamic neuron selection algorithm can enhance fine-tuning under memory constraints.
- Further research will focus on refining algorithm design to improve performance consistency, addressing cases where Velocity may still underperform.

Table 1. Comparison of final top1 test accuracies between Baseline, Sparse Update (SU), Random and Velocity neuron selection over various pretrained models, datasets, and budgets. For the first epoch the neurons to update are randomly selected. For each budget, highlighted results correspond to the best accuracy between neuron selection methods (green if SU is better, blue for Velocity, and red for Random, in bold the overall best performance regardless of the budget).

Model	$\mathcal{B}^w_{\max}$	Method	Cifar 10	Cifar 100	VWW	Flowers	Food	Pets	CUB
MbV2	192 311	SU	94.88±0.12	78.15±0.13	90.75±0.17	92.70±0.06	75.02±0.23	**86.93±0.22**	66.48±0.41
		Velocity	95.35±0.35	79.41±0.21	90.95±0.16	92.98±0.29	79.18±0.07	85.56±0.47	67.92±0.27
		Random	94.46±0.16	78.03±0.21	90.20±0.13	92.11±0.15	77.57±0.16	85.16±0.07	65.96±0.06
	464 639	SU	95.00±0.08	78.69±0.19	90.80±0.24	92.86±0.33	76.50±0.23	86.64±0.27	67.81±0.23
		Velocity	95.49±0.02	79.52±0.11	91.41±0.19	93.32±0.15	79.67±0.19	84.36±0.76	68.19±0.24
		Random	94.51±0.05	78.49±0.19	90.55±0.24	92.64±0.32	78.48±0.17	84.92±0.21	66.18±0.71
	675 540	SU	95.18±0.17	79.03±0.30	91.03±0.16	93.08±0.09	77.19±0.07	86.42±0.45	67.72±0.32
		Velocity	95.57±0.11	79.21±0.37	91.72±0.15	93.33±0.31	79.68±0.16	84.37±0.28	68.19±0.24
		Random	94.57±0.09	78.41±0.29	90.35±0.01	92.88±0.21	78.90±0.06	84.39±0.41	66.36±0.20
	2 189 760	Baseline	**95.93±0.14**	**79.83±0.29**	**91.80±0.03**	**94.02±0.03**	**80.63±0.10**	82.82±0.18	**69.24±0.34**
Resnet18	980 715	Velocity	95.51±0.10	78.77±0.41	88.78±0.51	90.78±0.24	75.09±0.13	**82.82±0.30**	63.64±0.35
		Random	95.20±0.20	77.98±0.38	88.33±0.45	89.39±0.47	74.57±0.15	79.49±0.51	60.93±0.54
	2 369 480	Velocity	95.36±0.15	**79.12±0.12**	89.16±0.31	**91.02±0.17**	75.72±0.23	82.01±0.60	**63.84±0.39**
		Random	95.68±0.10	78.28±0.22	89.21±0.23	89.53±0.17	75.17±0.12	79.40±0.34	61.42±0.32
	3 444 987	Velocity	95.58±0.21	78.95±0.13	89.17±0.33	90.76±0.15	75.83±0.12	81.45±0.63	63.59±0.03
		Random	95.80±0.09	78.52±0.18	88.92±0.19	89.66±0.30	75.28±0.12	79.28±0.34	61.45±0.32
	11 166 912	Baseline	**96.2±0.13**	78.86±0.08	**89.78±0.24**	90.14±0.26	**76.32±0.08**	79.76±0.63	60.97±0.52
Resnet50	2 059 888	Velocity	97.10±0.07	**82.94±0.35**	93.04±0.15	93.65±0.08	81.10±0.05	**90.11±0.25**	**73.73±0.52**
		Random	96.80±0.06	81.46±0.11	92.13±0.33	94.04±0.18	80.68±0.18	88.92±0.18	72.79±0.15
	4 976 842	Velocity	97.12±0.09	82.79±0.40	**93.37±0.14**	94.84±0.08	81.62±0.17	89.42±0.25	73.55±0.18
		Random	96.97±0.08	82.04±0.11	92.59±0.20	94.23±0.22	81.52±0.11	88.46±0.07	72.40±0.60
	7 235 830	Velocity	97.07±0.08	82.45±0.18	93.21±0.14	**95.03±0.18**	81.76±0.06	88.97±0.60	73.54±0.42
		Random	97.01±0.01	82.27±0.29	92.59±0.21	94.54±0.15	81.88±0.29	88.18±0.63	72.40±0.35
	23 454 912	Baseline	**97.30±0.04**	82.63±0.25	92.91±0.22	94.87±0.20	**82.49±0.15**	87.29±0.34	72.93±0.41

Acknowledgements This research was partially funded by Hi!PARIS Center on Data Analytics and Artificial Intelligence and by the project titled "PC2-NF-NAI" in theframework of the program "PEPR 5G et Réseaux du futur". This paper is supported by the European Union's HORIZON Research and Innovation Programme under grant agreement No 101120657, project ENFIELD (European Lighthouse to Manifest Trustworthy and Green AI).

The source code used to run experiments is available at https://github.com/aelQuelennec/-WACV-2024-Ways-to-Select-Neurons-under-a-Budget-Constraint

[1] Lin, J., Zhu, L., Chen, W. M., Wang, W. C., & Gan, C. On-Device Training Under 256KB Memory. In *Advances in Neural Information Processing Systems*.

[2] Bragagnolo, A., Tartaglione, E., & Grangetto, M. To update or not to update? Neurons at equilibrium in deep models. In *Advances in Neural Information Processing Systems.*