Batch Pruning by Activation Stability

Md Mustakin Alam[1], Shaker Islam[2], Aminul Islam[1]

[1]University of Louisiana at Lafayette | [2]BRAC University

The Fourteenth International Conference on Learning Representations

Rio de Janeiro, Brazil - 2026

Introduction

- Deep learning models are powerful but **resource-intensive** - significant GPU hours, memory, and energy
- Much computation is wasted on **redundant or less informative data**
- Reducing training cost without sacrificing performance is a longstanding challenge
- Key question: *Can internal activation stability determine which batches to skip?*

Motivation

Existing approaches fall short of:

- **Static pruning:** high preprocessing cost, not adaptive during training
- **Dynamic pruning:** relies on complex heuristics, per-sample loss tracking
- **Dataset distillation:** scalability challenges, high computational overhead

Key insight from Neural Collapse:

- Activation patterns **stabilize as training converges** - a natural signal for pruning

Contributions

1. **Batch Pruning by Activation Stability (*B-PAS*): Activation Stability-Guided Dynamic Batch Pruning:**

   Lightweight, plug-and-play method that prunes batches on-the-fly based on activation variance stability across epochs

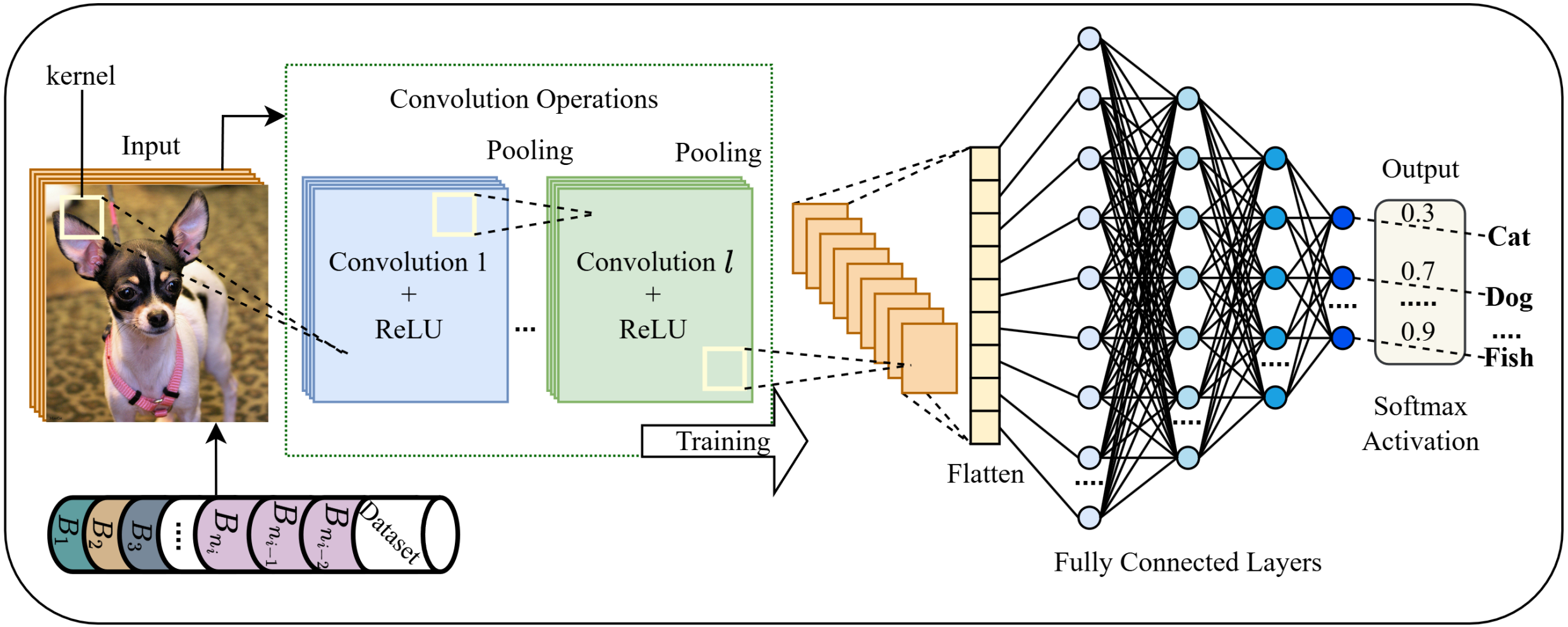
2. **Generalization across Architectures and Datasets:**

   ResNet-18, ResNet-50, CvT, GPT-2 on CIFAR-10/100, SVHN, ImageNet-1K, Alpaca

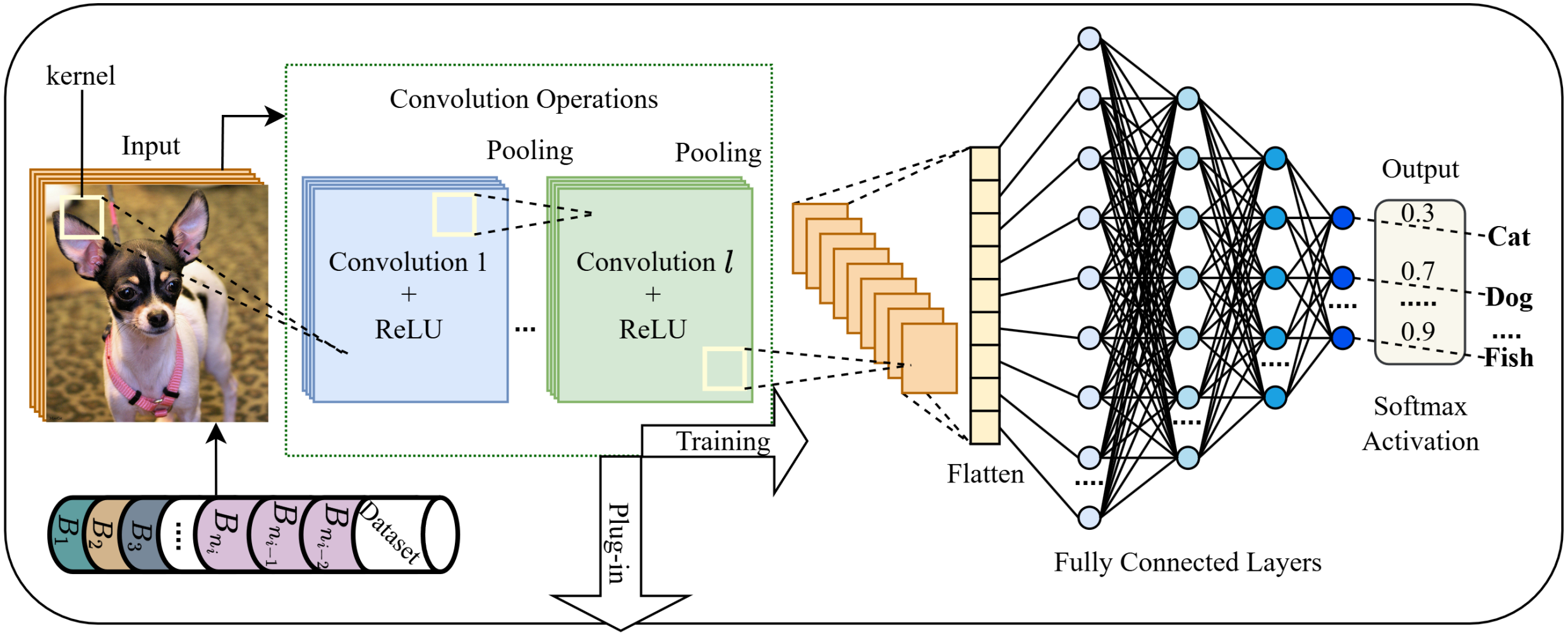
3. **Data Savings Index (DSI):**

   New metric quantifying cumulative fraction of training data saved during learning

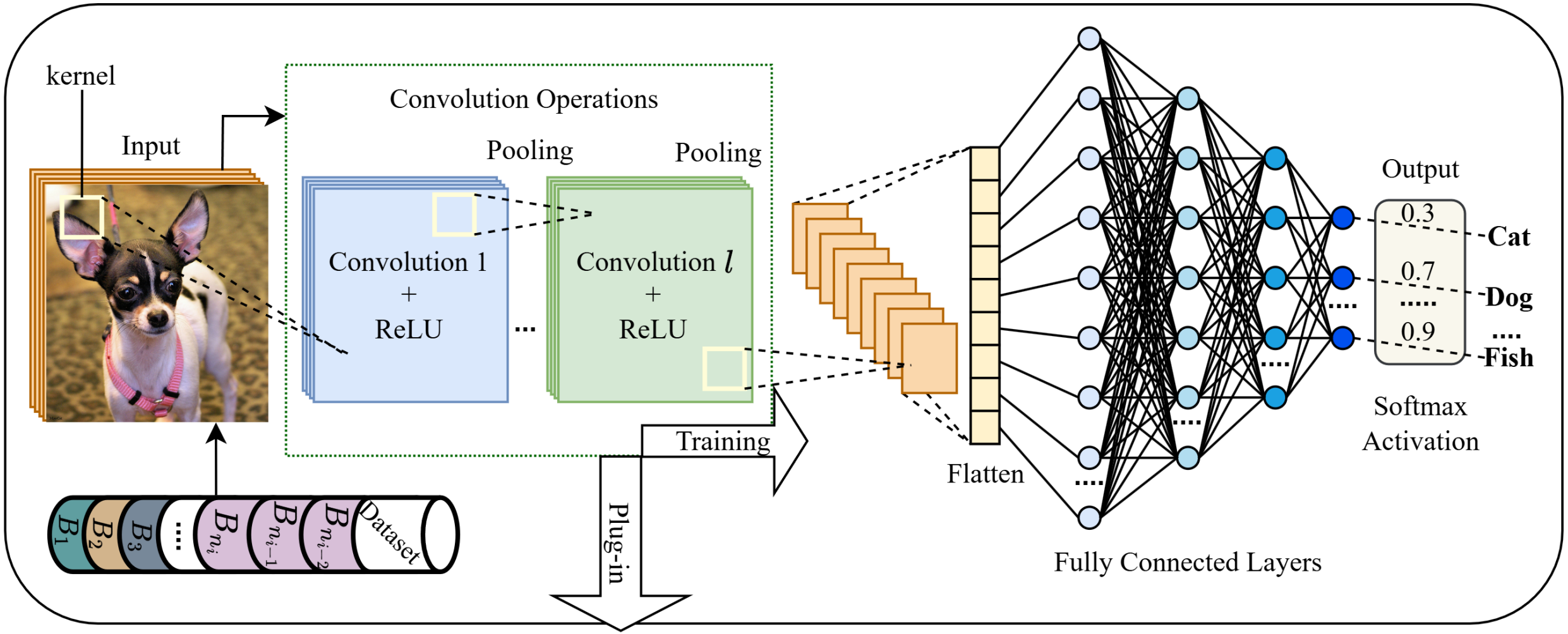
A Standard Neural Network Architecture for Image Classification



A Standard Neural Network Architecture for Image Classification

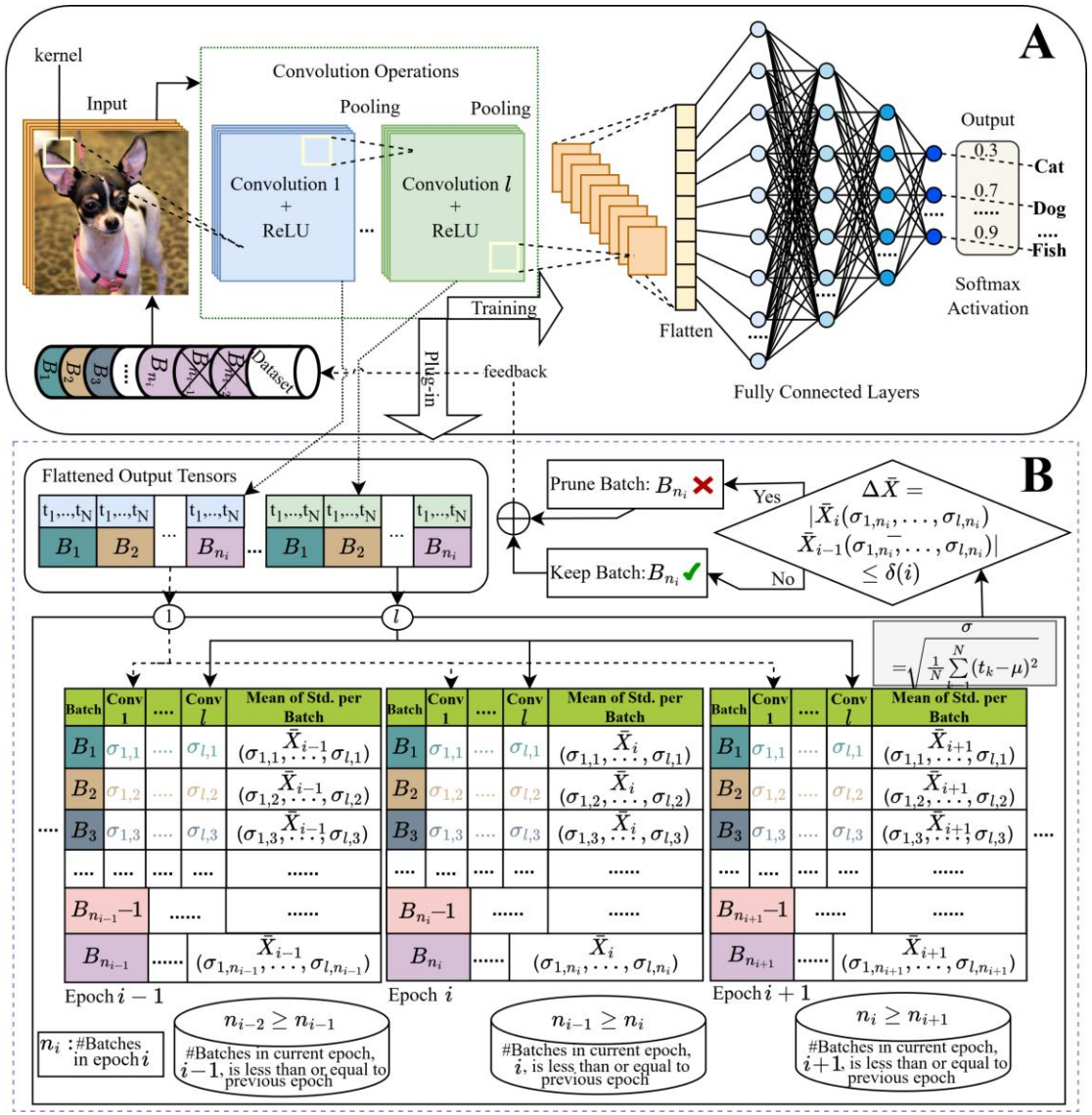


A Standard Neural Network Architecture for Image Classification



B-PAS

Batch Pruning by Activation Stability (*B-PAS*)



B-PAS: Activation Tracking & Stability

1

Activation Tracking

For each batch at every conv layer:

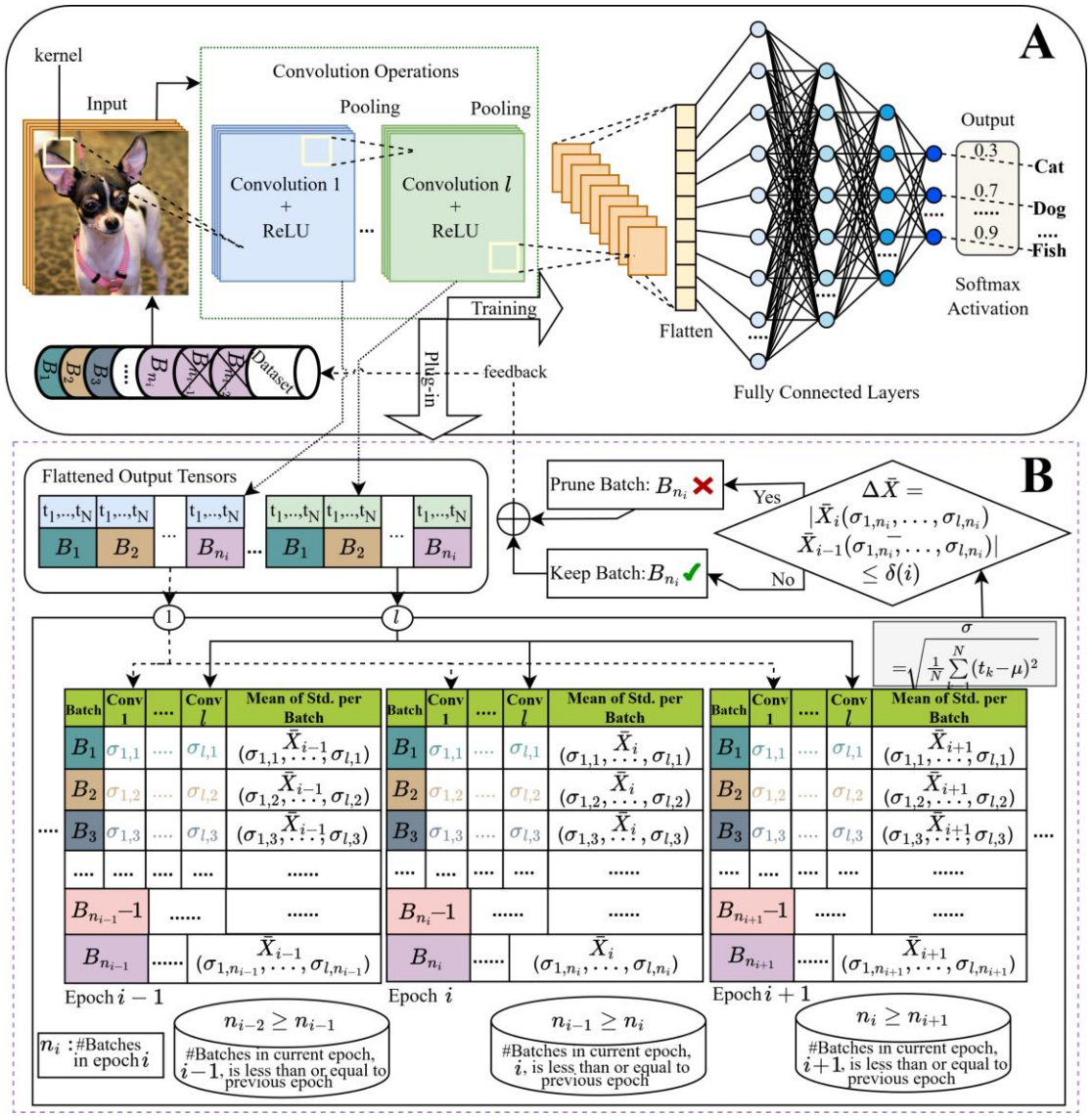
- Flatten post-ReLU output tensors
- Compute σ (std. deviation) per batch per layer
- Average across all L layers → $\overline{X}$

2

Stability Check

Compare across consecutive epochs:

- $\Delta\overline{X} = |\overline{X}_i - \overline{X}_{i-1}|$
- If $\Delta\overline{X}$ approaches zero → batch has converged
- Activations stabilized = low utility



B-PAS: Pruning & Exponential Threshold Schedule

3

Dynamic Batch Pruning

- If $\Delta\overline{X} < \delta(i)$, batch is permanently pruned
- Dataset shrinks monotonically: $n_i \leq n_{i-1}$
- Focuses training on informative batches only

4

Exponential Threshold Schedule

- $\delta(i) = \delta_s \cdot e^{\alpha i}$
- $\alpha = \frac{1}{I} \ln\left(\frac{\delta e}{\delta_s}\right)$
- Conservative early (features being learned)
- Aggressive later (learning stabilizes)

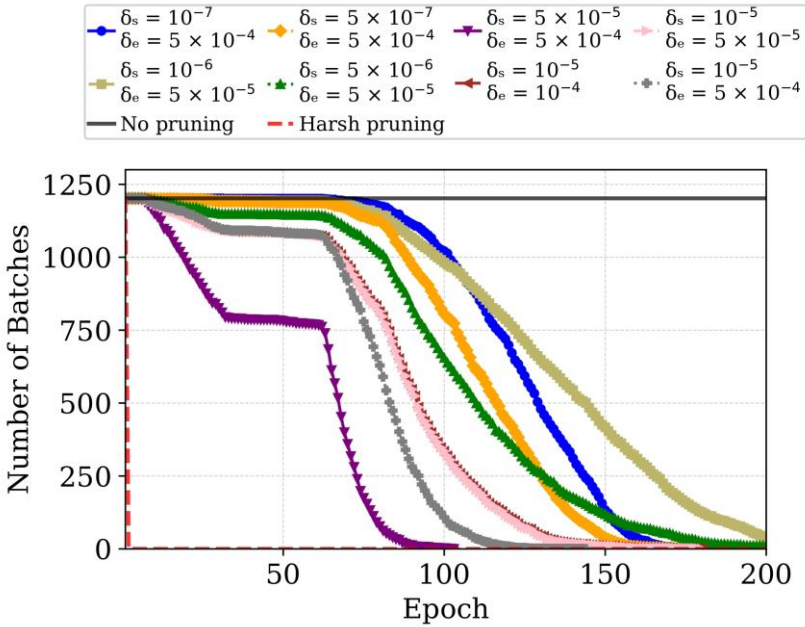


Figure: Pruning dynamics under different delta settings for ResNet-50 on ImageNet-1K (200 epochs). Lower thresholds (e.g., $\delta \in [1 \times 10^{-6}, 5 \times 10^{-5}]$) lead to conservative pruning, retaining most batches until late epochs, while higher thresholds (e.g., $\delta \in [5 \times 10^{-5}, 5 \times 10^{-4}]$) cause aggressive pruning and premature training termination. The dynamic schedule ($\delta_s = 5 \times 10^{-6}$, $\delta e = 5 \times 10^{-5}$) provides a balanced trajectory, steadily reducing data.

Results: Comparison with SOTA

ImageNet-1K with ResNet-50 — Comparison with SOTA (InfoBatch)

Approach	Saved Hrs (%)	DSI (%)	Accuracy (%)
ResNet-50 Full Dataset	0	0	78.07
+ InfoBatch (40%)	40	28	78.07
+ B-PAS ($\delta \in [10^{-5}, 10^{-4}]$)	**61**	**57**	**78.07**
+ B-PAS ($\delta \in [5 \times 10^{-6}, 5 \times 10^{-5}]$)	**48**	**47**	**78.43**

Highlights:

- Up to **57% data savings** and **61% GPU node-hour reduction** with no accuracy loss
- Outperforms InfoBatch by **29% higher data savings** and **21% greater GPU savings**
- Comparatively conservative thresholds can even **improve accuracy to 78.43%** while saving 47% data

Results: Cross-Architecture & Dataset Robustness

	CIFAR-10	CIFAR-10	CIFAR-100	CIFAR-100	SVHN	ImageNet	ImageNet
	R-18	R-50	R-18	R-50	R-50	R-50	CvT
Full Dataset	95.60	95.66	78.20	80.60	96.27	78.07	79.65
B-PAS	**95.60**	**95.66**	**78.20**	**80.60**	**96.27**	**78.43**	**79.60**
DSI (%)	**25**	**33**	**24**	**30**	**30**	**47**	**14**
Saved Hrs (%)	23	29	22	29	33	48	13

Key Takeaways:

- *B-PAS* **preserves accuracy** across all model-dataset combinations
- Largest gains on **ImageNet-1K**: 47% DSI, 48% GPU savings - where efficiency matters most
- Extended to **GPT-2 fine-tuning** - pruned batches with no performance loss
- For CvT**,** conservative thresholds yield about 14% data and 13% GPU-hour savings with no accuracy loss, while more aggressive thresholds reach about 35% savings with a small accuracy drop, reflecting slower activation stabilization in transformers compared to CNNs.

Conclusion

- *B-PAS* is a **practical, plug-and-play** approach for data-efficient deep learning
- Activation stability is a **powerful internal signal** for efficient training
- Up to **57% data savings** and **61% GPU reduction**, surpassing SOTA methods
- Generalizes across **CNNs, ViTs, and LLMs** (GPT-2 fine-tuning)

Future Work: Data-driven, adaptive thresholding mechanisms

References

[1] Qin et al., “InfoBatch: Lossless Training Speed Up by Unbiased Dynamic Data Pruning,” ICLR 2024.

[2] Papyan et al., “Prevalence of Neural Collapse During the Terminal Phase of Deep Learning Training,” PNAS 2020.

[3] Ahmad et al., “When Do Convolutional Neural Networks Stop Learning?” arXiv 2024.

[4] He et al., “Deep Residual Learning for Image Recognition,” CVPR 2016.

[5] Wu et al., “CvT: Introducing Convolutions to Vision Transformers,” ICCV 2021.

Paper

Thank You!

Questions?

Contact: md-mustakin.alam1@louisiana.edu

Code