Optimizing models for faster inference

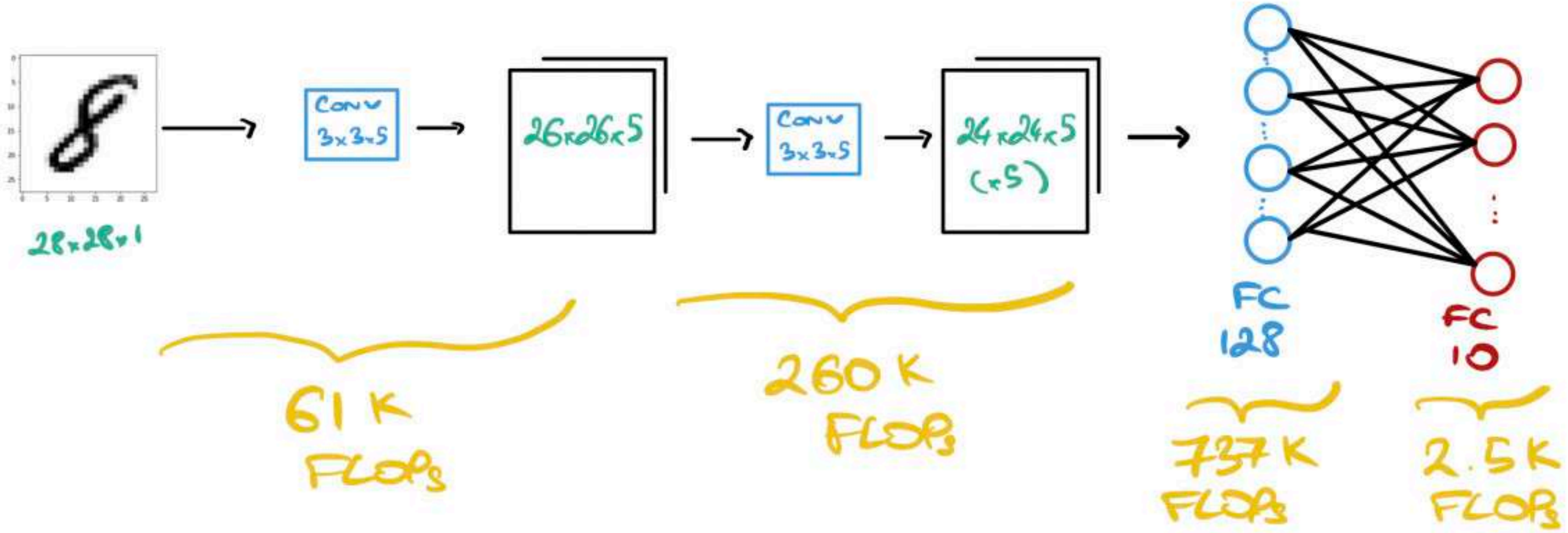
Markovich Alexander

2022

How to measure the speed of inference?

- The inference time is **how long** is takes for a forward propagation
- In order to measure this time, we must understand 3 ideas: FLOPs, FLOPS, and MACs
- **FLOPs** or **Floating Point Operations** are total number of calculations such as addition, subtraction, division, multiplication
- **FLOPS** are the Floating Point Operations per Second
- **MACs** or **Multiply-Accumulate Computations** are operations that perform addition and multiplication, that is, 2 operations.
  As a rule, we consider **1 MAC = 2 FLOPs**.

Calculating the FLOPs

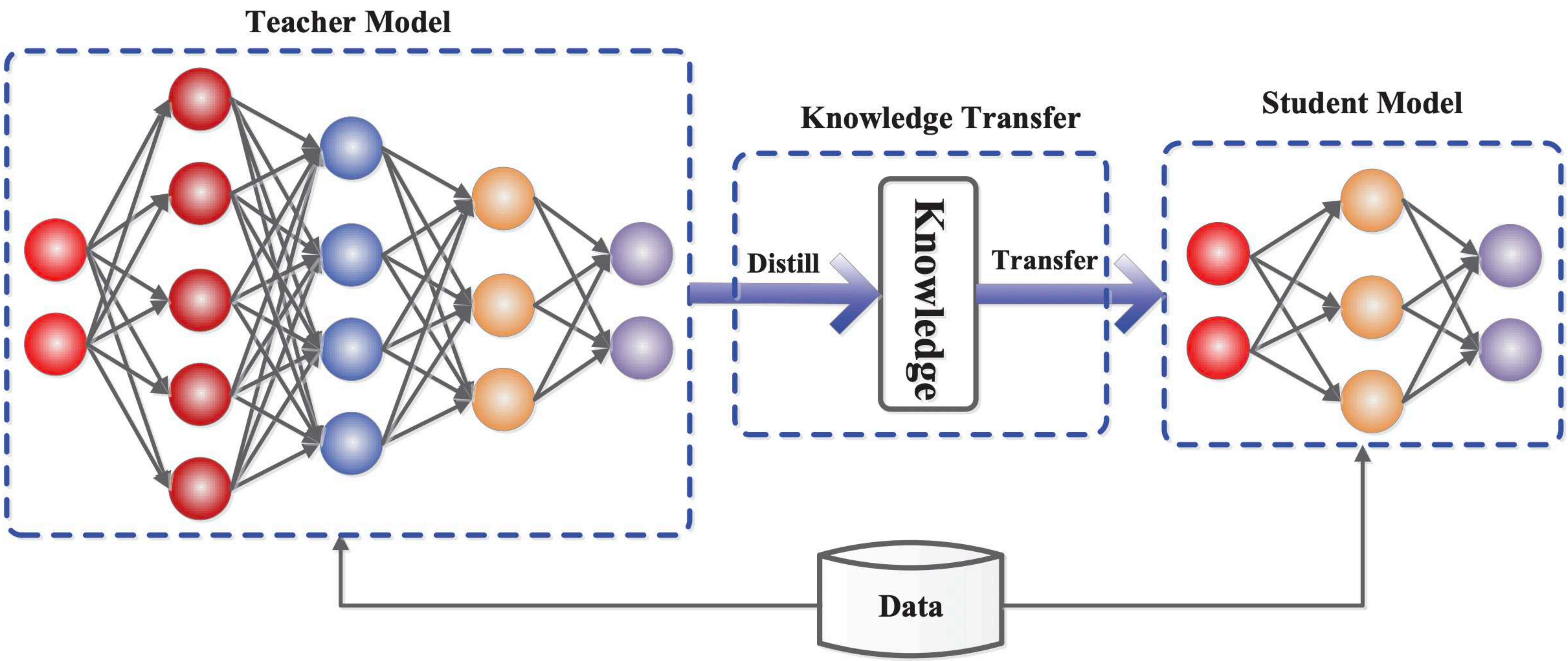


- The model will do **FLOPs = 60,840 + 259,200 + 737,280 + 2,560 = 1,060,400** operations
- Say we have a CPU that performs 1 GFLOPS FLOPs/FLOPS = (1,060,400)/(1,000,000,000) = **0,001 s or 1ms**

So that we want…

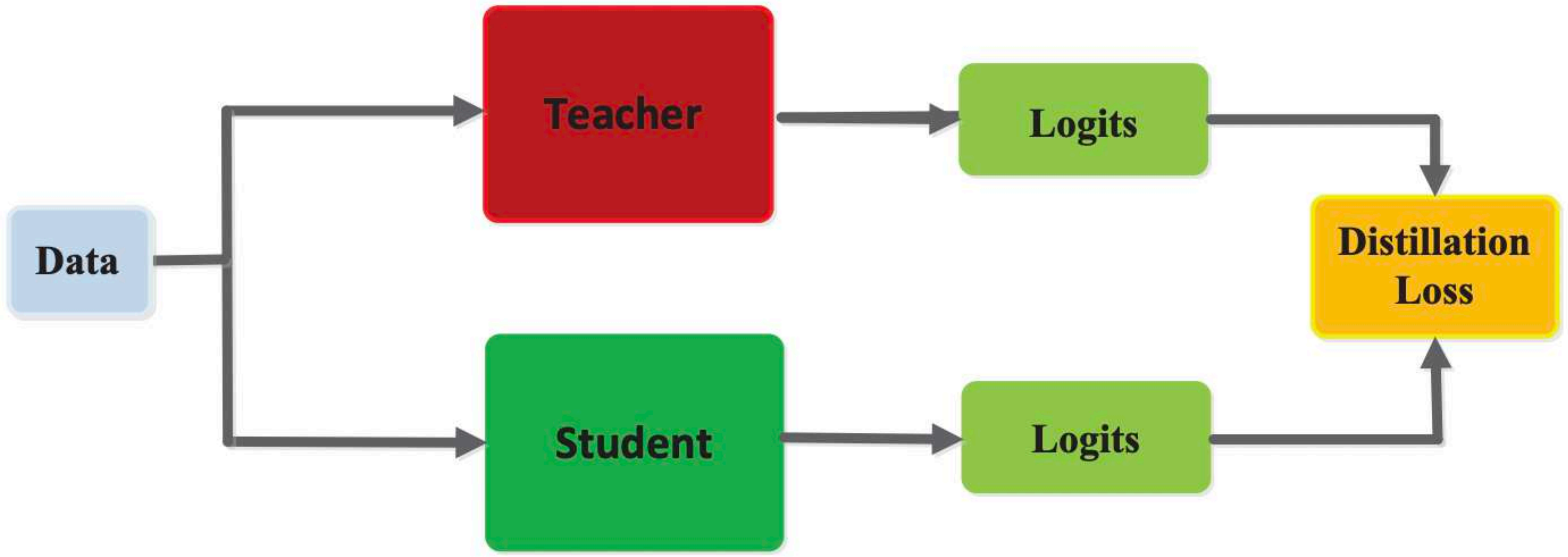
- A **low number of FLOPs in our model**, but keeping it complex enough to be good
- A high number of **FLOPS** in our hardware (previous week)
- To reduce the model size
  This gives **faster loading**, smaller **size** in storage, faster **compilation**

Knowledge Distillation



Knowledge Distillation

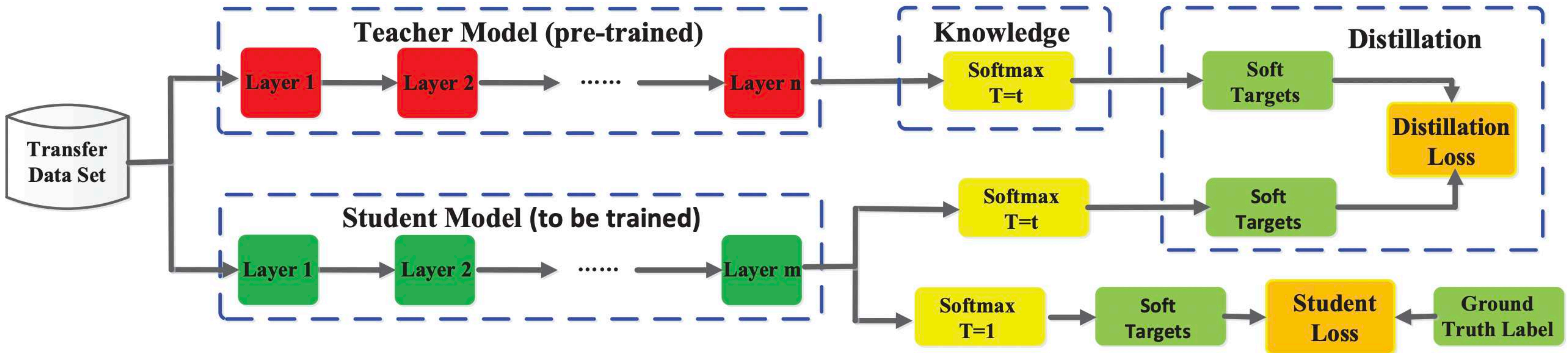
Response-Based Knowledge



- Update **student** weights and freeze **teacher** weights
- The **responses** can be logits, offsets, heatmaps and so on

Knowledge Distillation

Response-Based Knowledge



- Optimise weighted combination of **student** and **distillation** losses
- As usual, **student loss** is cross-entropy and **distillation loss** is Kullback-Leibler divergence

Knowledge Distillation

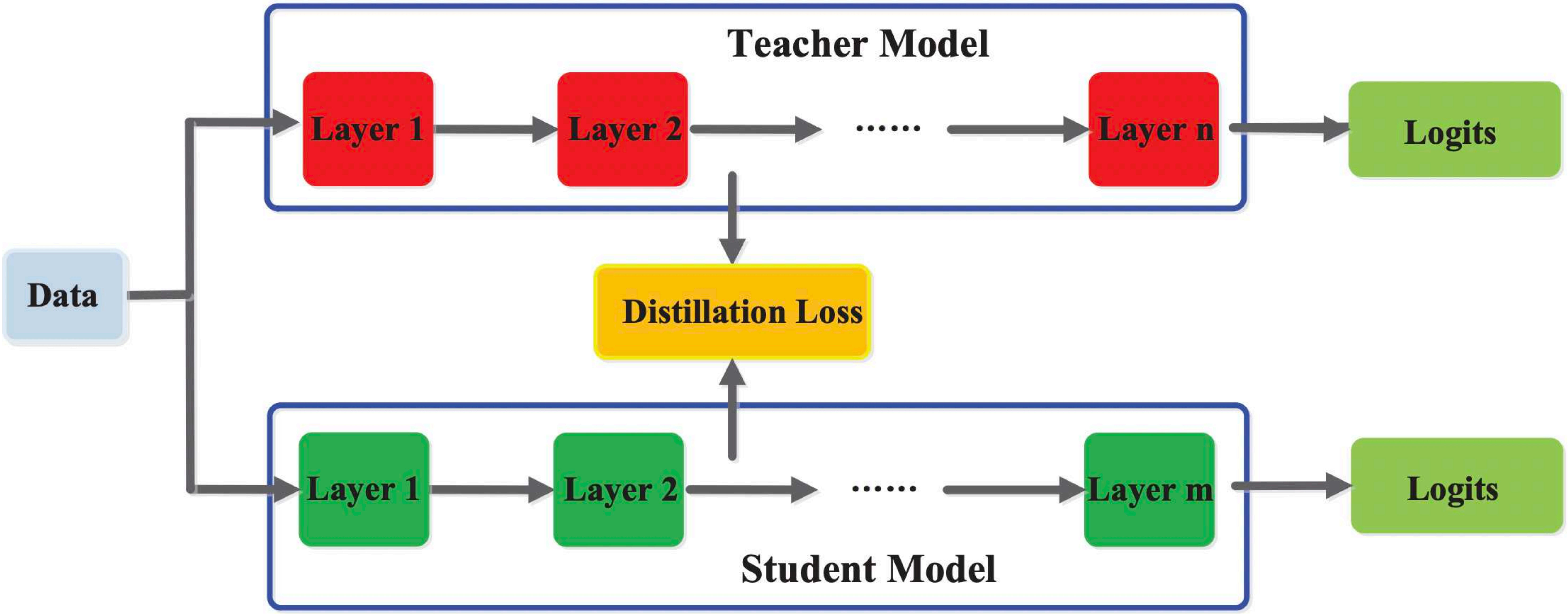
Why we need temperature?

Temperature	Logits	Softmax Probabilities
1	$[30, 5, 2, 2]$	$[1\mathrm{e}+0, 1.38\mathrm{e}-11, 6.91\mathrm{e}-13, 6.91\mathrm{e}-13]$
10	$[3, 0.5, 0.2, 0.2]$	$[0.8308, 0.0682, 0.0505, 0.0505]$

- Soft targets contain the informative **dark knowledge** from the teacher model
- Higher temperatures produce softer probabilities which provides a stronger signal to the student

Knowledge Distillation

Feature-Based Knowledge



- Directly match the feature activations of the teacher and the student

Knowledge Distillation

Feature-Based Knowledge

$$L = \mathcal{L}_F(\Phi_t(f_t(x)), \Phi_s(f_s(x)))$$

Similarity Function
(L1, L2, MMD, CE)

Feature Map

Alignment Function
(MLP, Conv)

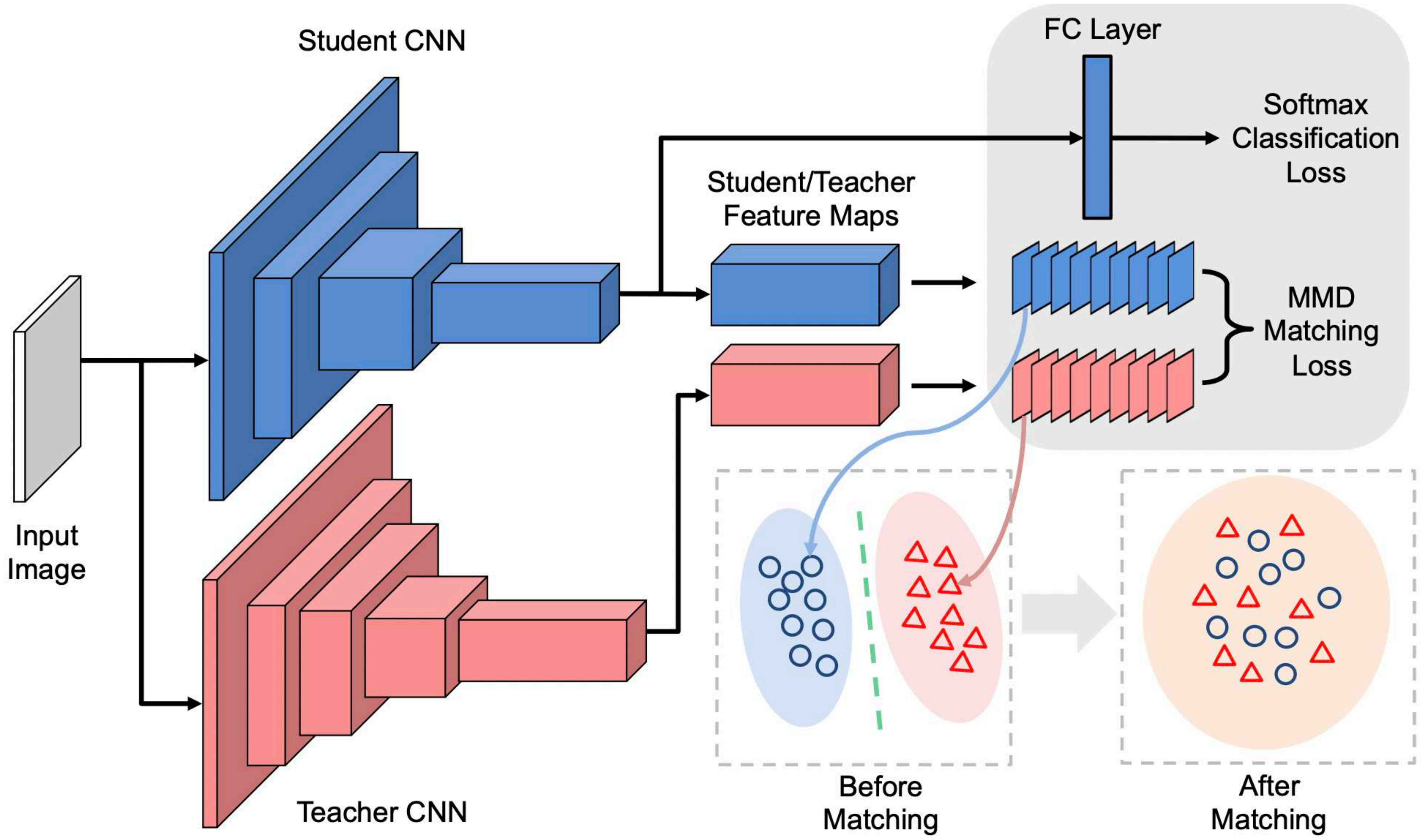
Knowledge Distillation

Feature-Based Knowledge

$$p_{i|j} = \frac{K\left(\mathbf{x}_i, \mathbf{x}_j; 2\sigma_t^2\right)}{\sum_{k=1,k\neq j}^{N} K\left(\mathbf{x}_k, \mathbf{x}_j; 2\sigma_t^2\right)} \qquad q_{i|j} = \frac{K\left(\mathbf{y}_i, \mathbf{y}_j; 2\sigma_t^2\right)}{\sum_{k=1,k\neq j}^{N} K\left(\mathbf{y}_k, \mathbf{y}_j; 2\sigma_s^2\right)}$$

$$K_{\text{cosine}}\left(\mathbf{a}, \mathbf{b}\right) = \frac{1}{2}\left(\frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\|_2\|\mathbf{b}\|_2} + 1\right) \in [0, 1]$$

$$\mathcal{KL}(\mathcal{P}\|\mathcal{Q}) = \int_{\mathbf{t}} \mathcal{P}(\mathbf{t}) \log \frac{\mathcal{P}(\mathbf{t})}{\mathcal{Q}(\mathbf{t})} d\mathbf{t}$$



Note that you can vary the **kernel function** and the **divergence metric**

Knowledge Distillation

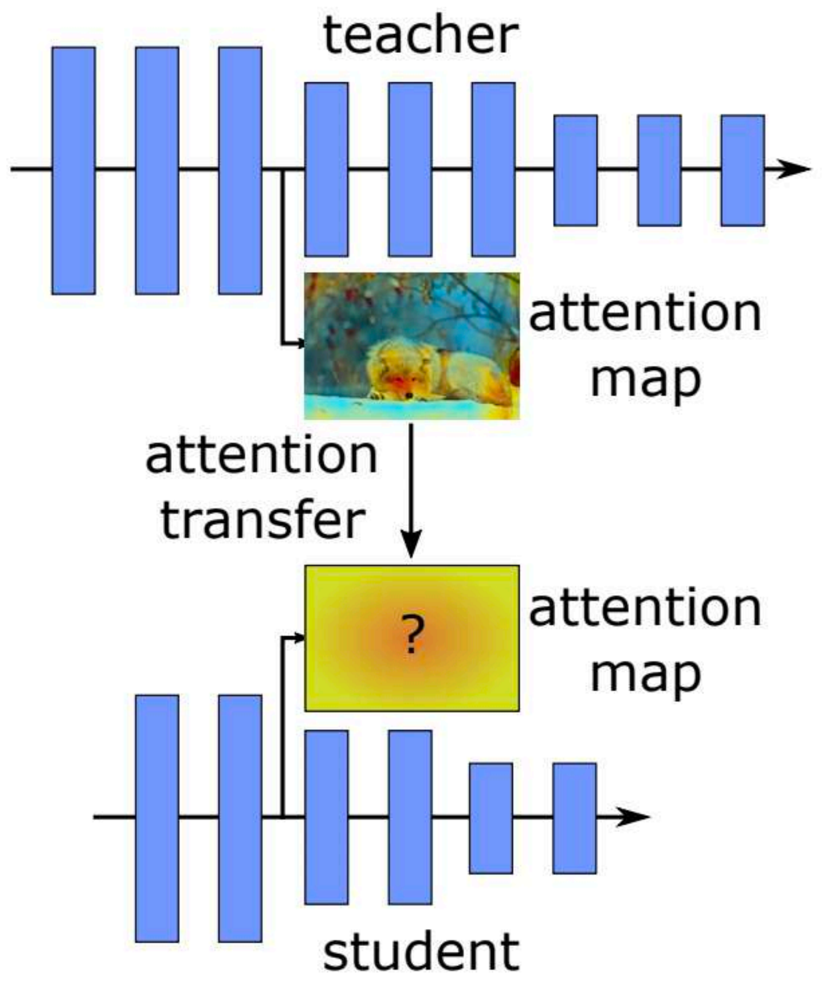
Feature-Based Knowledge

Activation-based Attention Map

$$F_{\mathrm{sum}}(A) = \sum_{i=1}^{C} |A_i|$$
$$F_{\mathrm{sum}}^{p}(A) = \sum_{i=1}^{C} |A_i|^p$$
$$F_{\max}^{p}(A) = \max_{i=1,C} |A_i|^p$$

input image

attention map



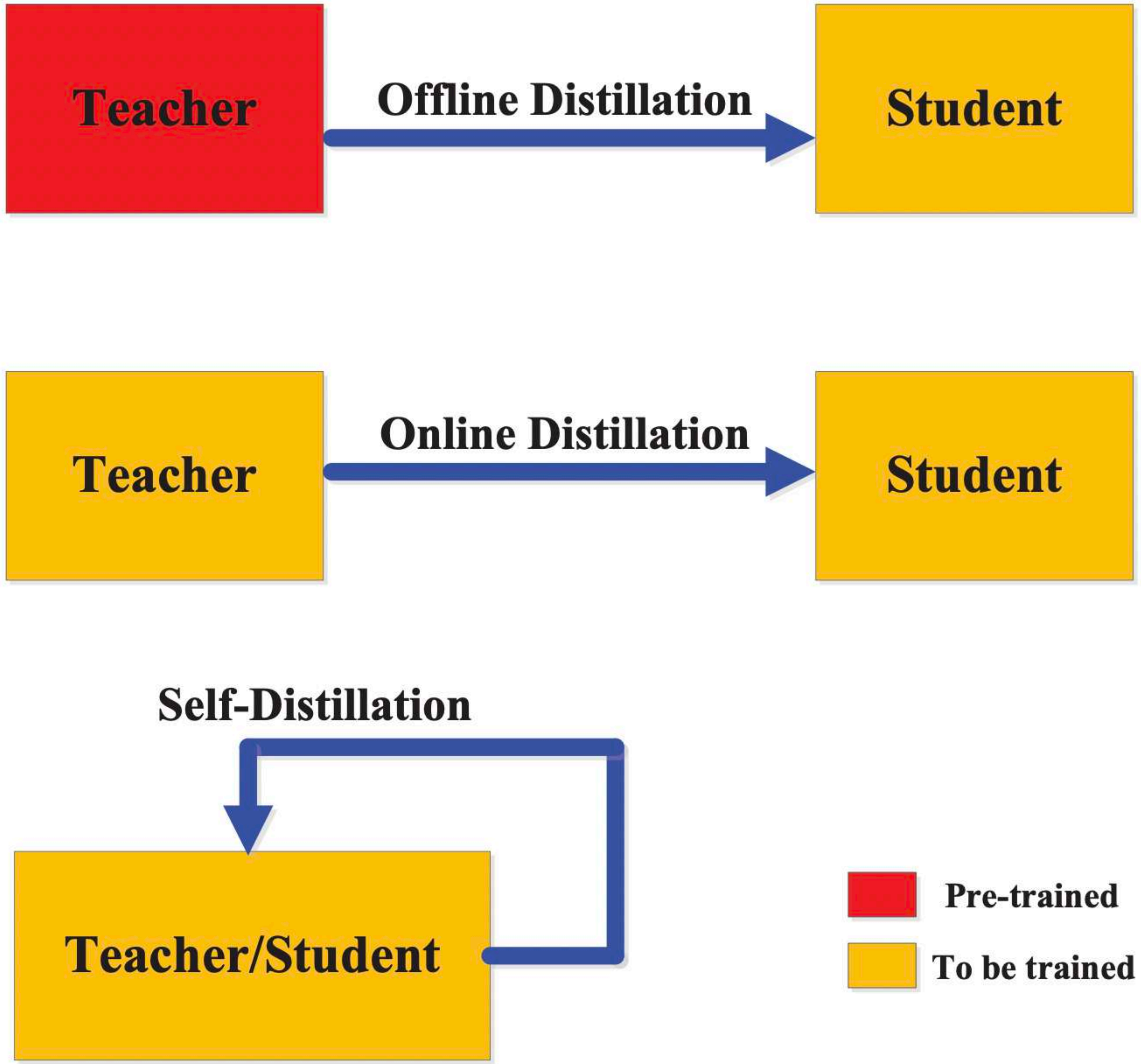
Gradient-based Attention Map

$$J_S = \frac{\partial}{\partial x}\mathcal{L}(\mathbf{W_S}, x), J_T = \frac{\partial}{\partial x}\mathcal{L}(\mathbf{W_T}, x)$$
$$\mathcal{L}(\mathbf{W_S}, \mathbf{W_T}, x) = \mathcal{L}(\mathbf{W_s}, x) + \frac{\beta}{2}\|J_S - J_T\|_2$$

Note that same technique in the **attention mechanism** can be applied

Knowledge Distillation

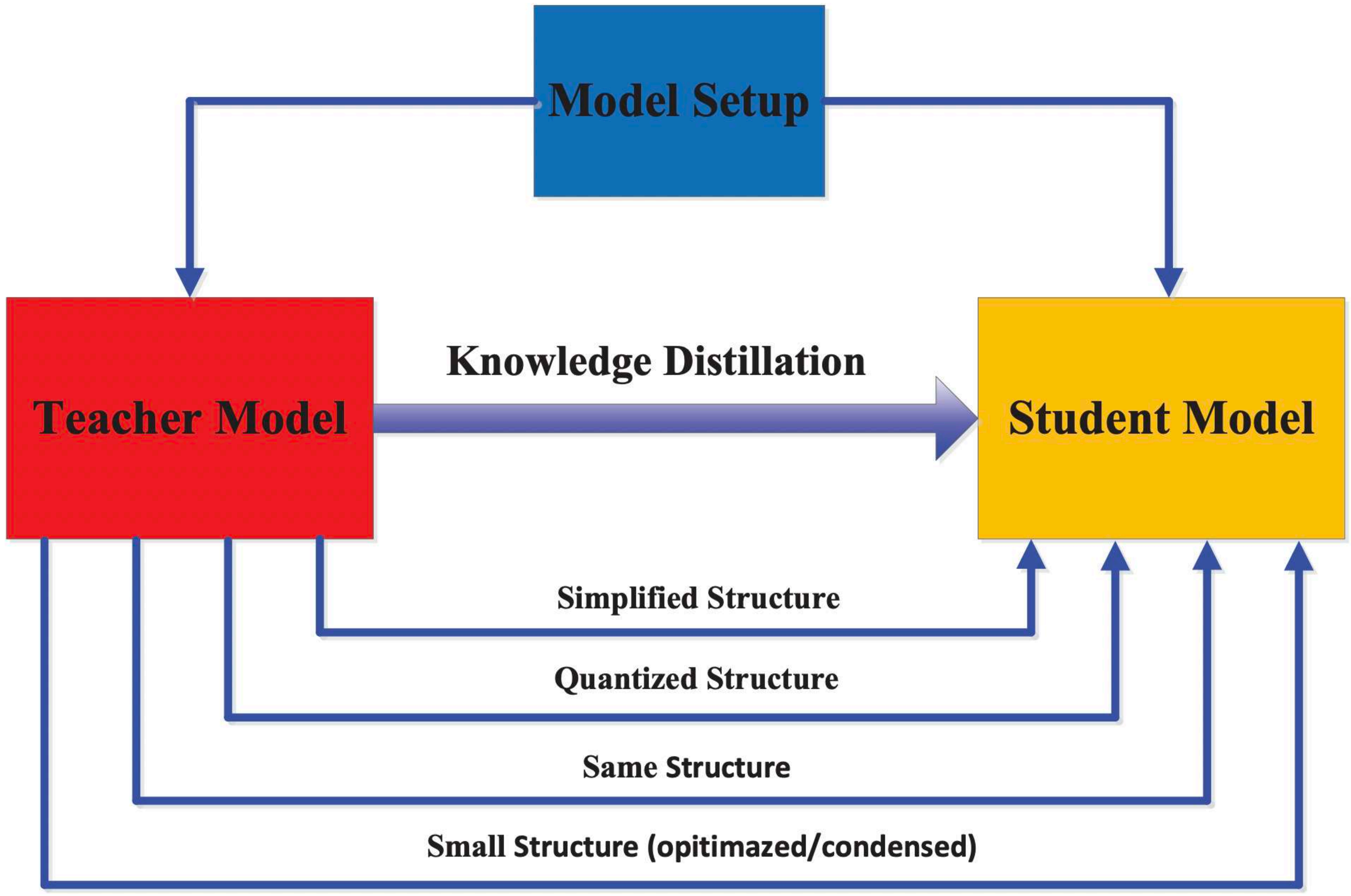
Distillation Schemes



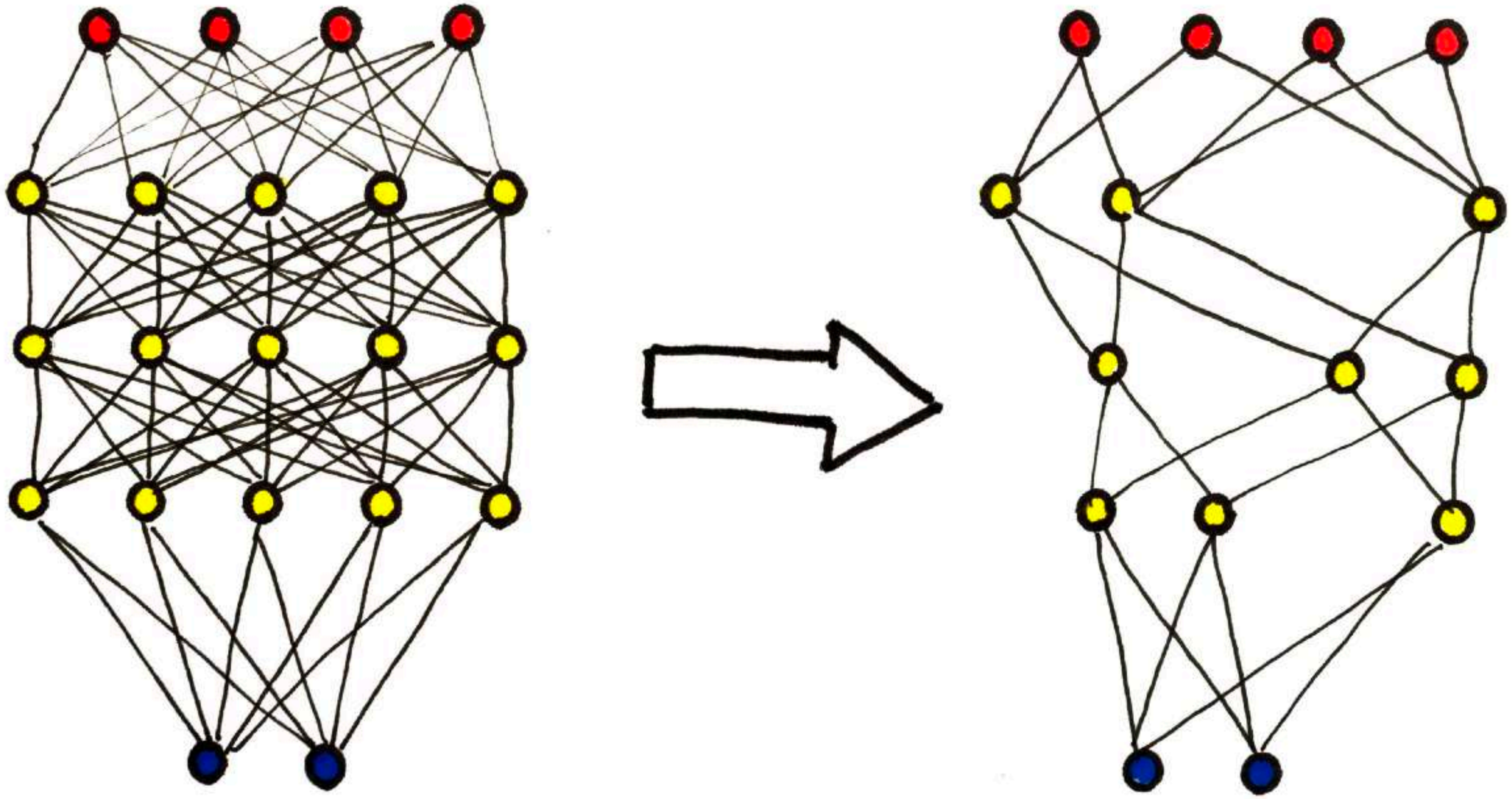
Knowledge Distillation

Teacher-Student Architecture

- Fewer layers or fewer channels in each layer
- Quantized version
- Efficient basic operations
- Optimized global network structure

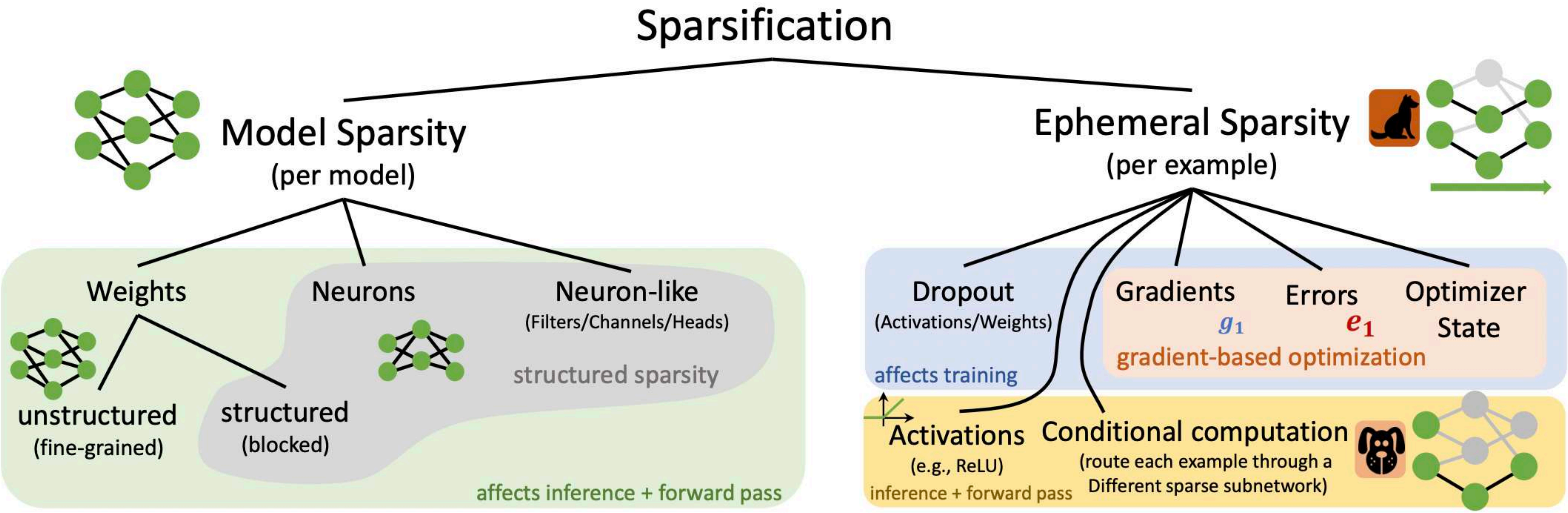


Pruning



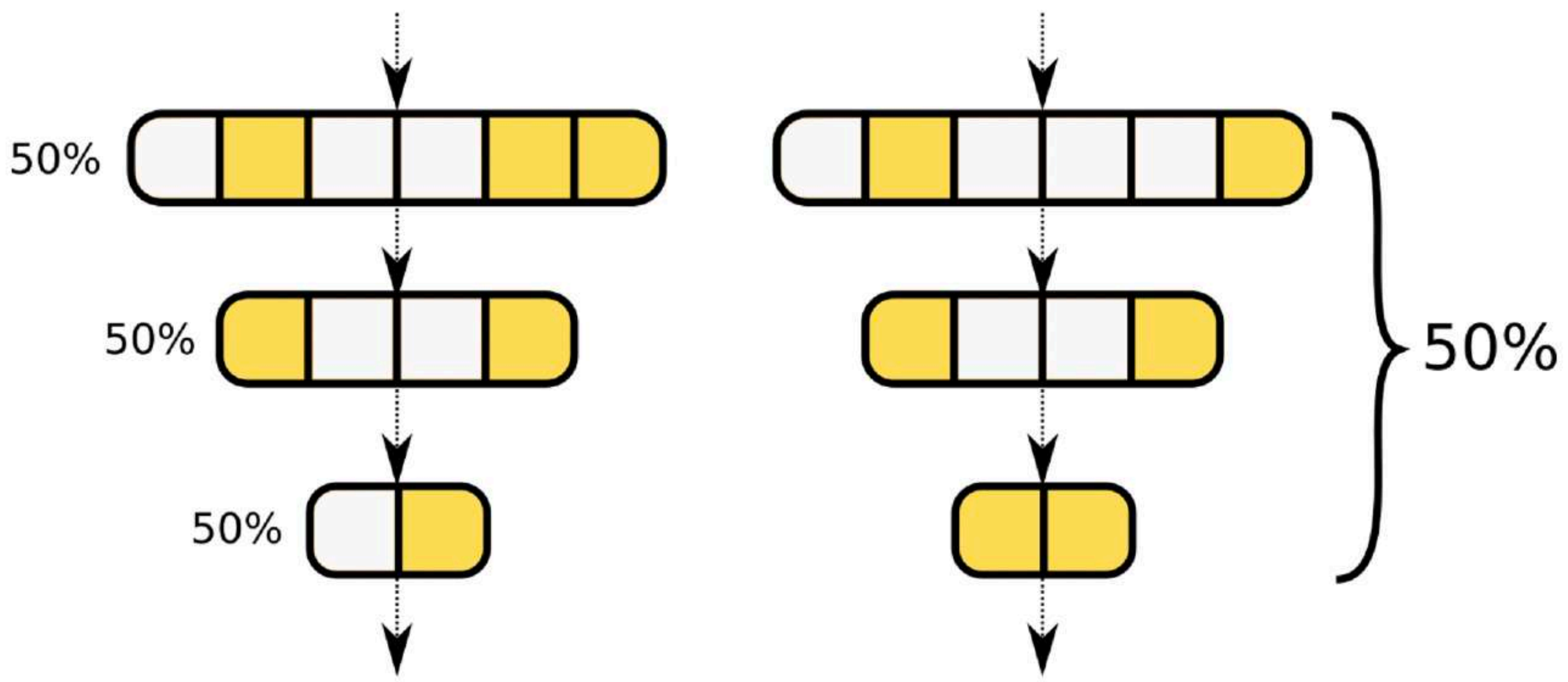
Pruning

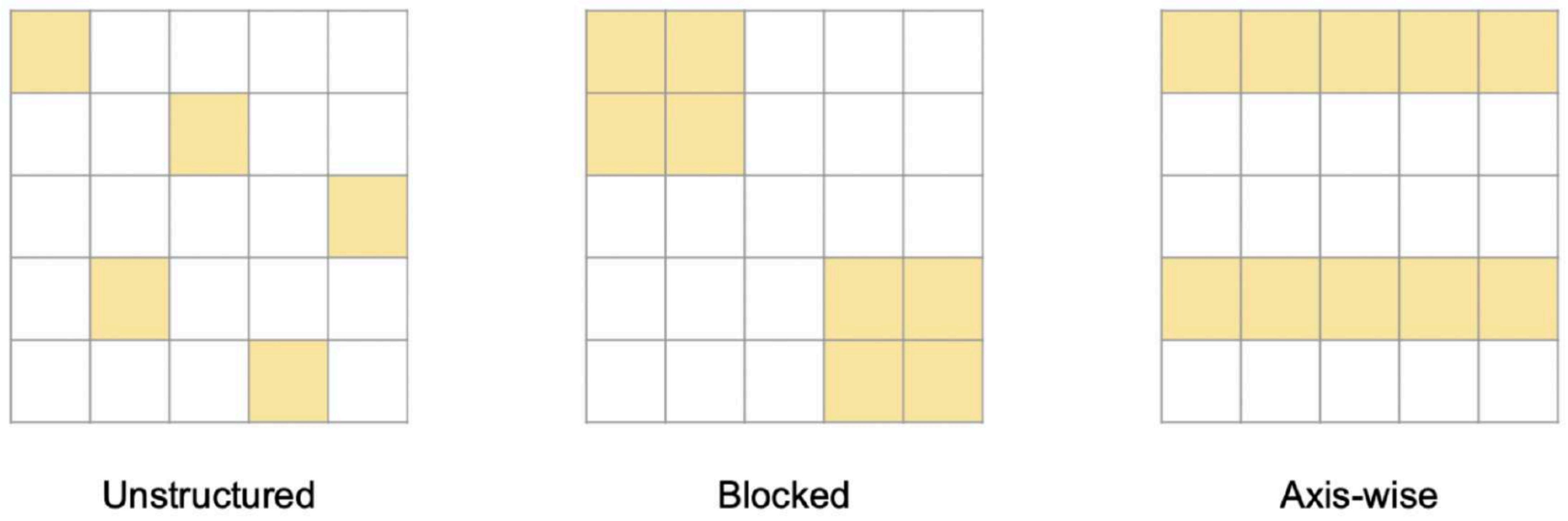
What can be sparsified?



Pruning

local/global and Individuals/groups

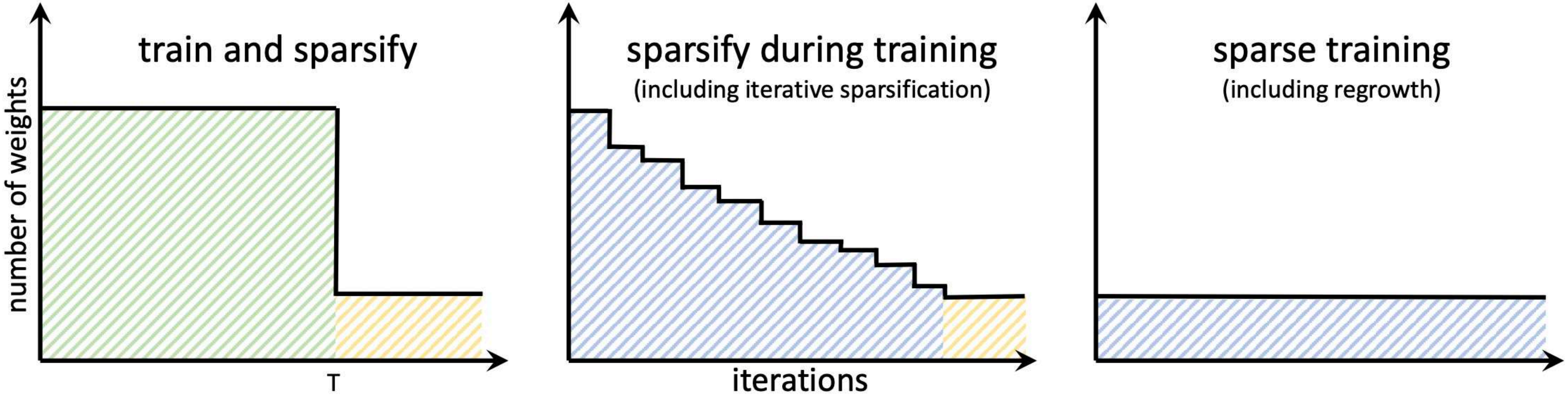




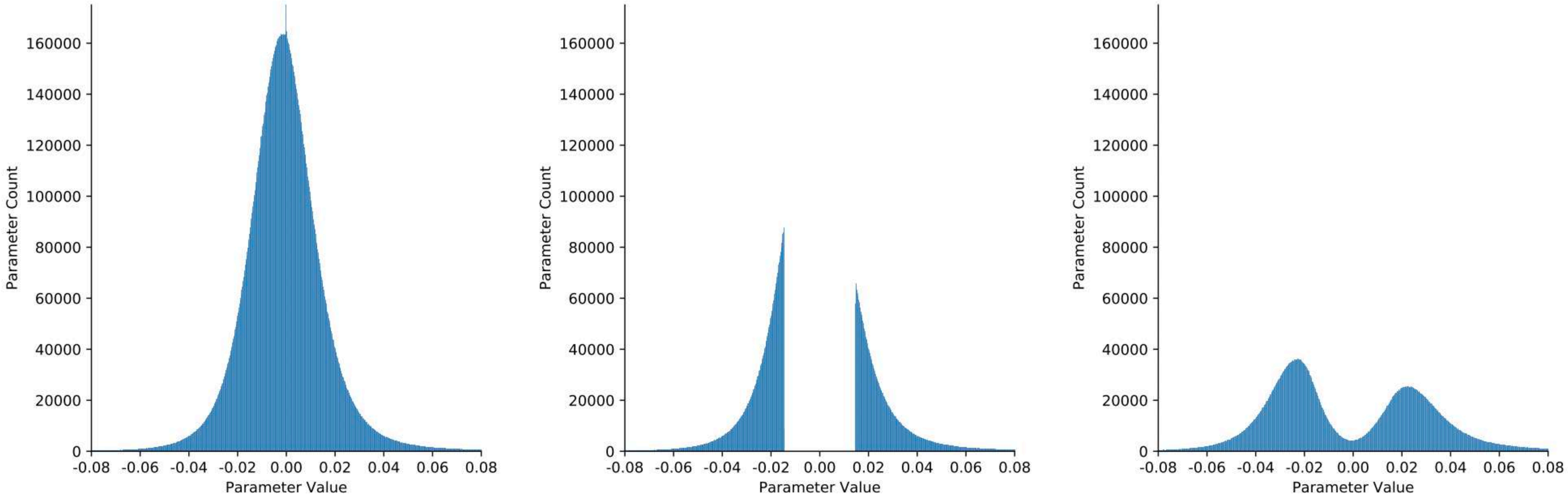
$$-\sum_{i=1}^{N} \log p(y_i \mid x_i, W) + \lambda \sum_{\eta \in \mathcal{H},\ w_\eta \in W} \|w_\eta\|_2 \to \min$$

Pruning

Sparsification Schedules



Pruning
Magnitude-based (not only weights)



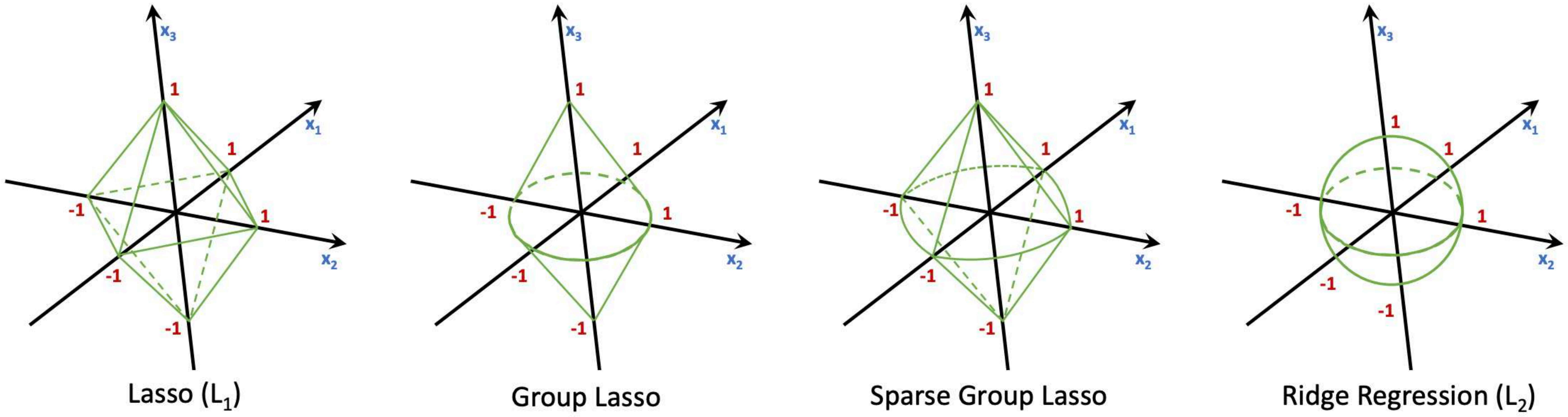
(a) Dense Network (76.0%) (b) 70% Pruned (36.1%) (c) After 3-epoch Retraining (71.4%)

$$\text{threshold}(w_i) = \begin{cases} w_i, & \text{if } |w_i| > \lambda \\ 0, & \text{if } |w_i| \leq \lambda \end{cases}$$

$$\text{threshold}_\tau(w_i) = \begin{cases} w_i, & \text{if } |w_i| \text{ in TOP-p} \\ 0, & else \end{cases}$$

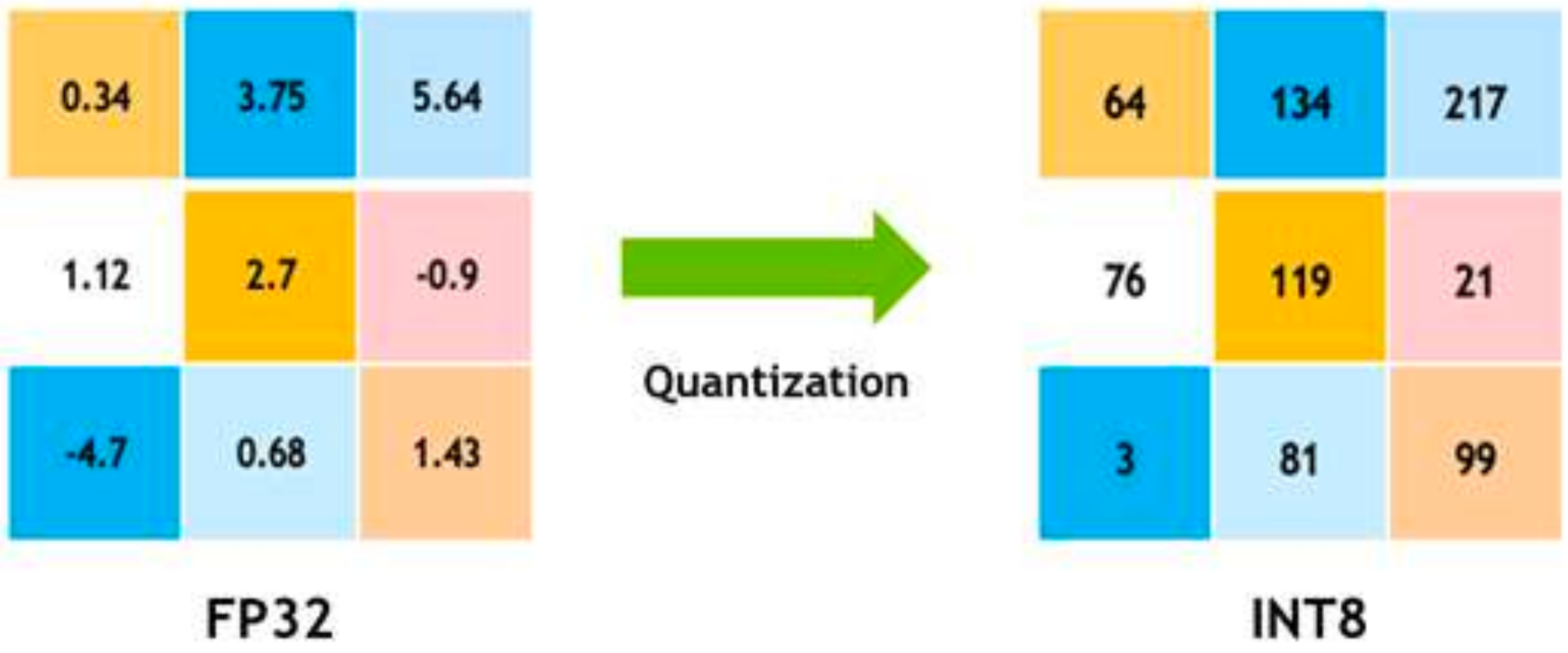
Pruning

Regularizations



$$\min_{\beta \in \mathbb{R}^p} \left(\left\| \mathbf{y} - \sum_{g=1}^{G} \mathbf{X_g} \beta_g \right\|_2^2 + \lambda_1 \sum_{g=1}^{G} \|\beta_g\|_2 + \lambda_2 \|\beta\|_1 \right)$$

Quantization

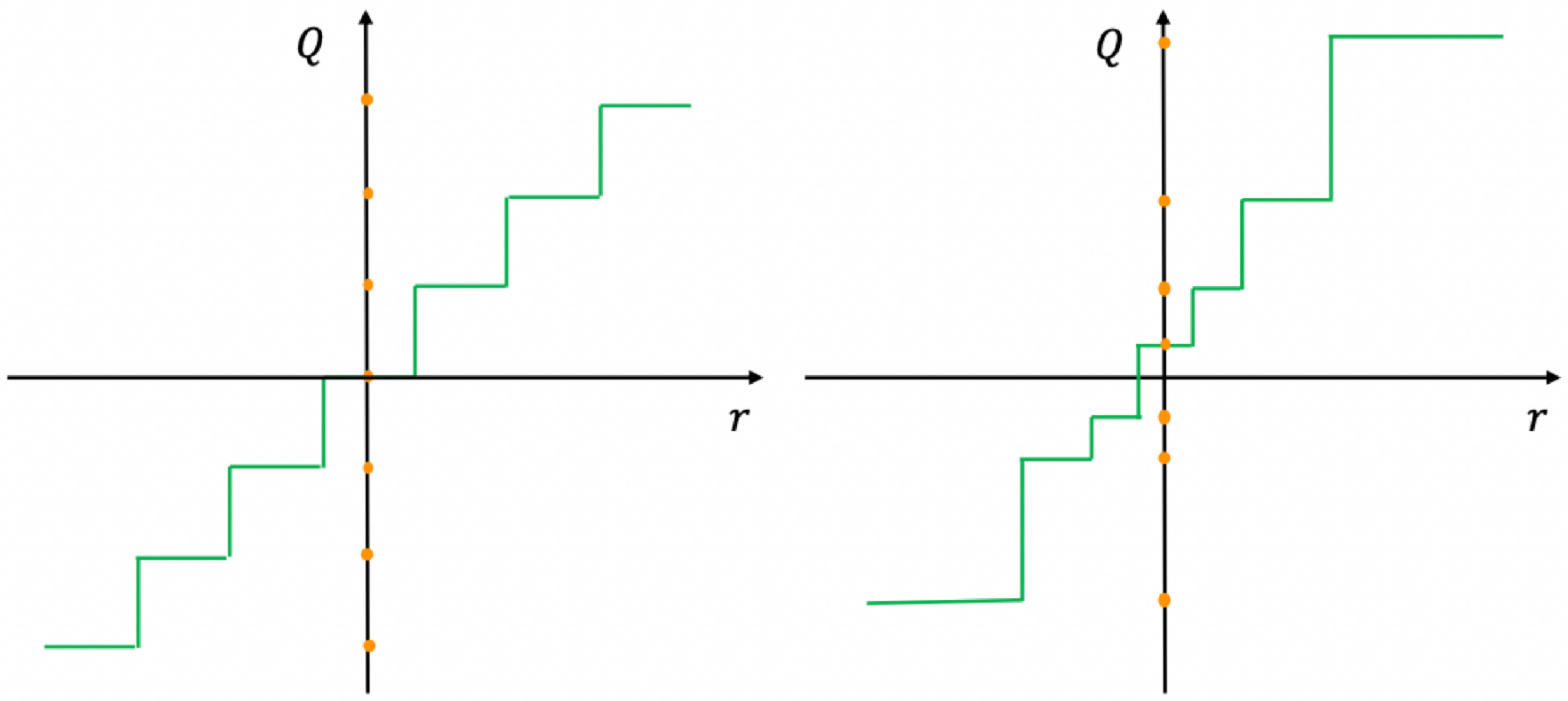


Quantization

Uniform, Symmetric and Asymmetric

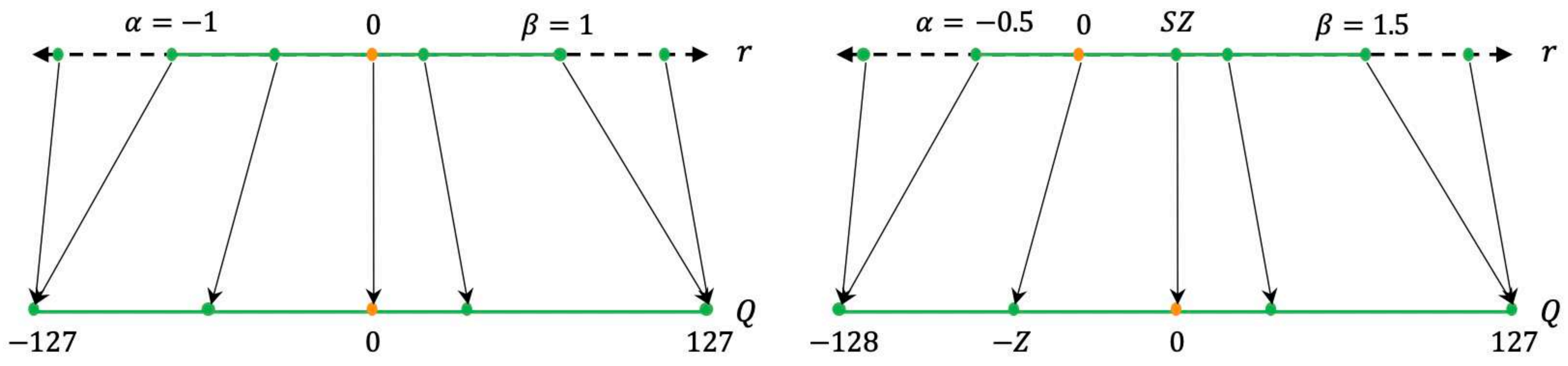
$$Q(r) = \mathrm{Int}(r/S) - Z$$

$$\tilde{r} = S(Q(r) + Z)$$



$$Q(r) = \mathrm{Int}\left(\frac{r}{S}\right)$$

$$S = \frac{\beta - \alpha}{2^b - 1}$$



Quantization

Static vs Dynamic

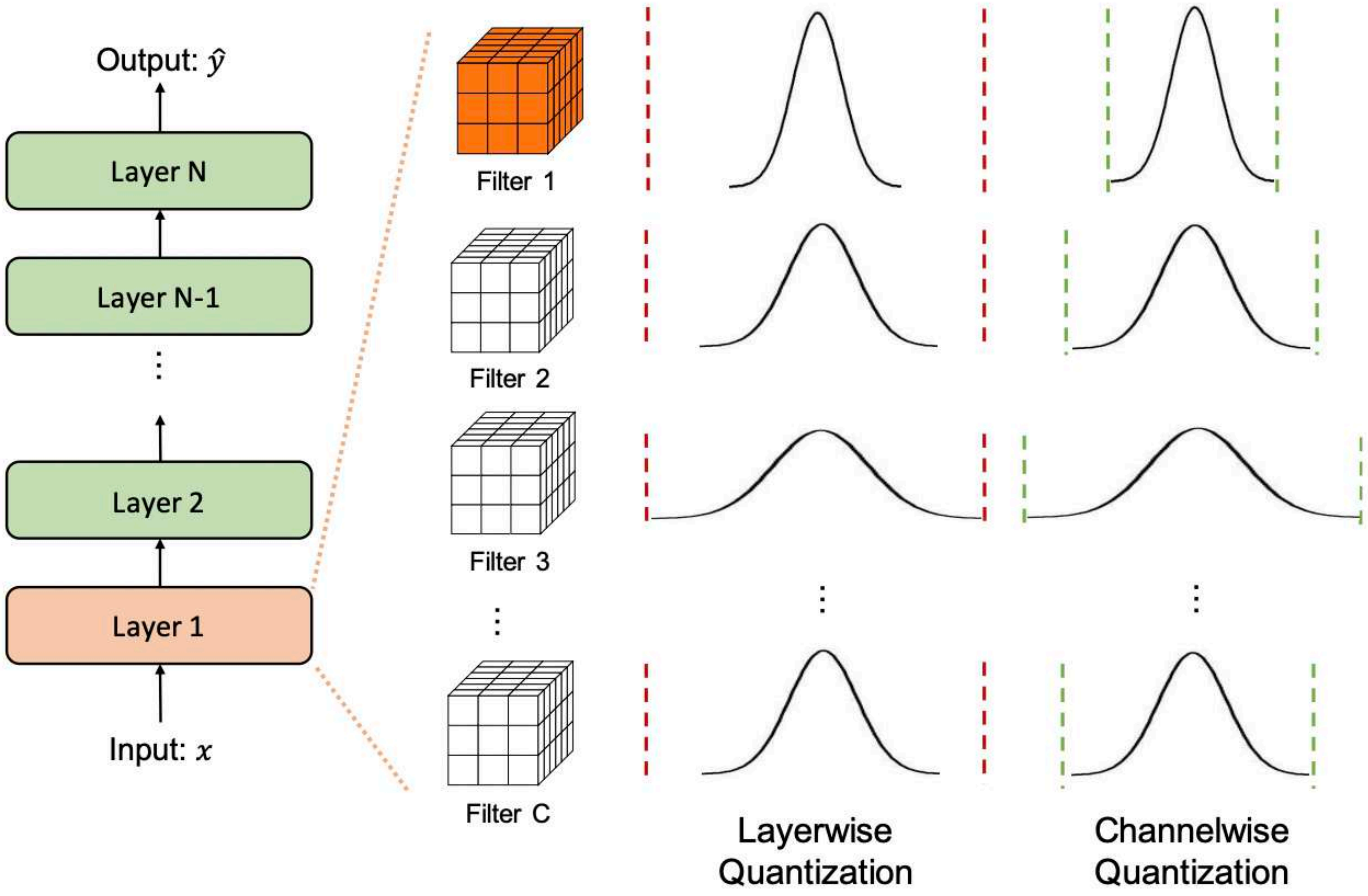
- In dynamic quantization, range is dynamically calculated for each activation map during runtime
  Suitable for RNNs

- In static quantization clipping range is pre-calculated and static during inference.
  Calibrate range with calibration dataset

Quantization

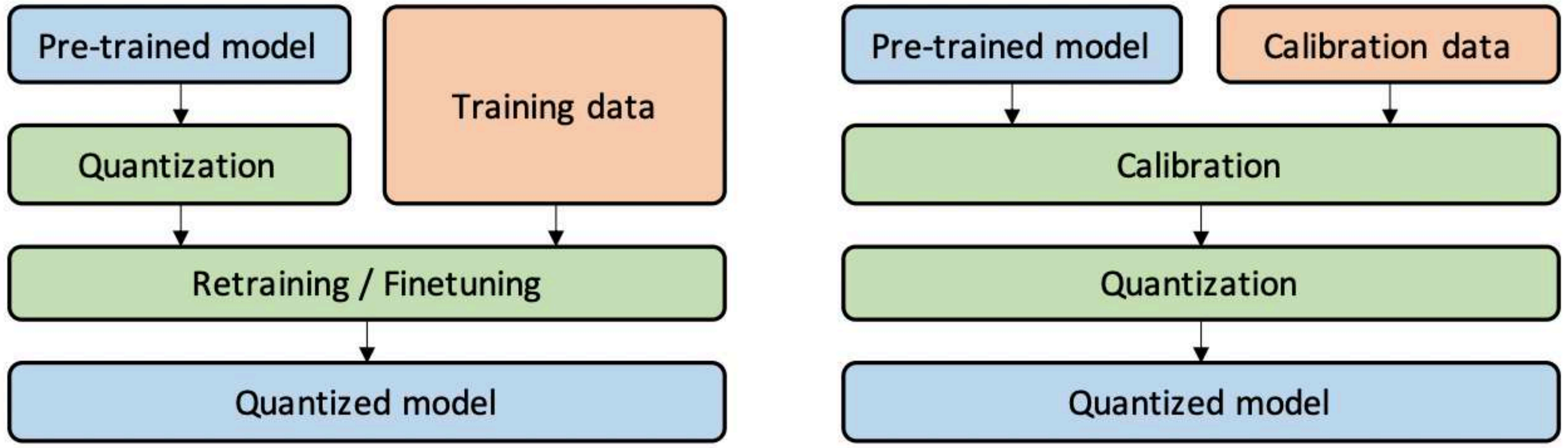
Granularity

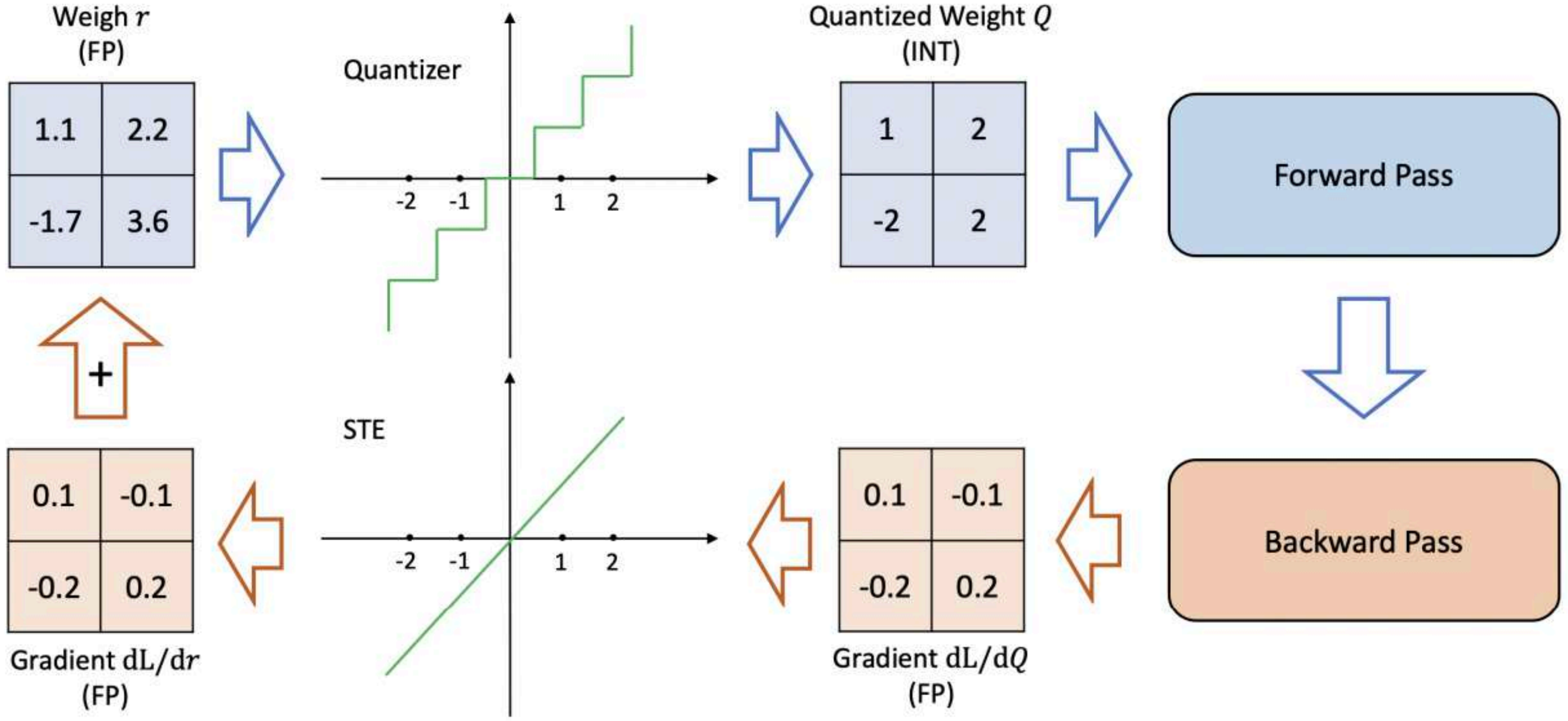
- Layerwise
- Groupwise
- Channelwise
- Sub-channelwise



Quantization

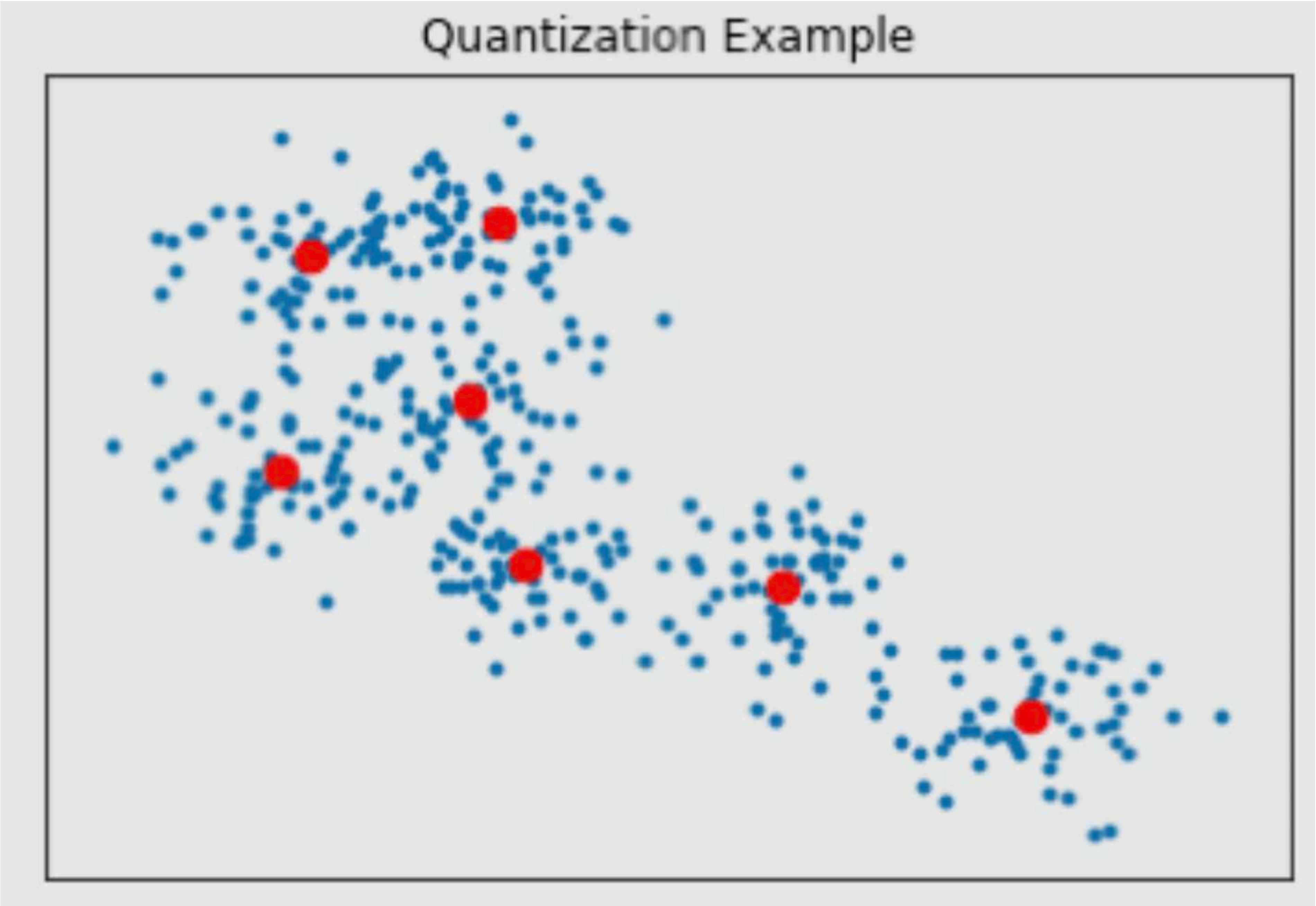
QAT vs PTQ

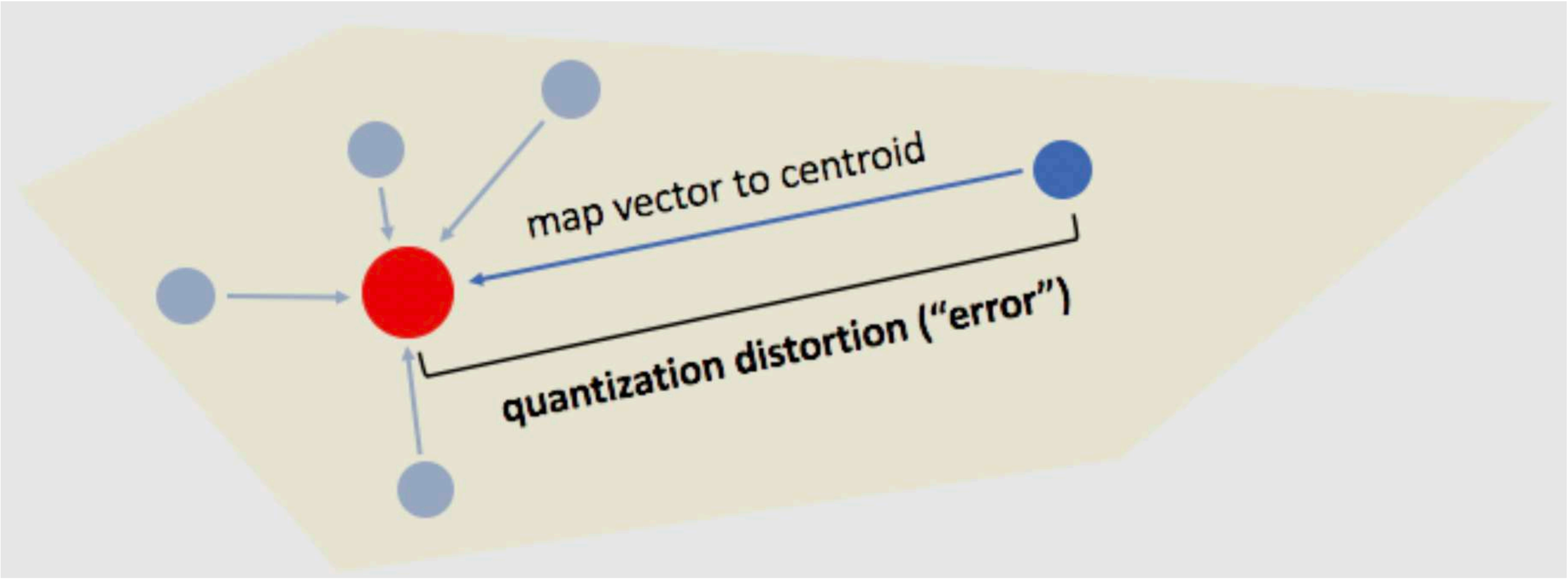




Quantization

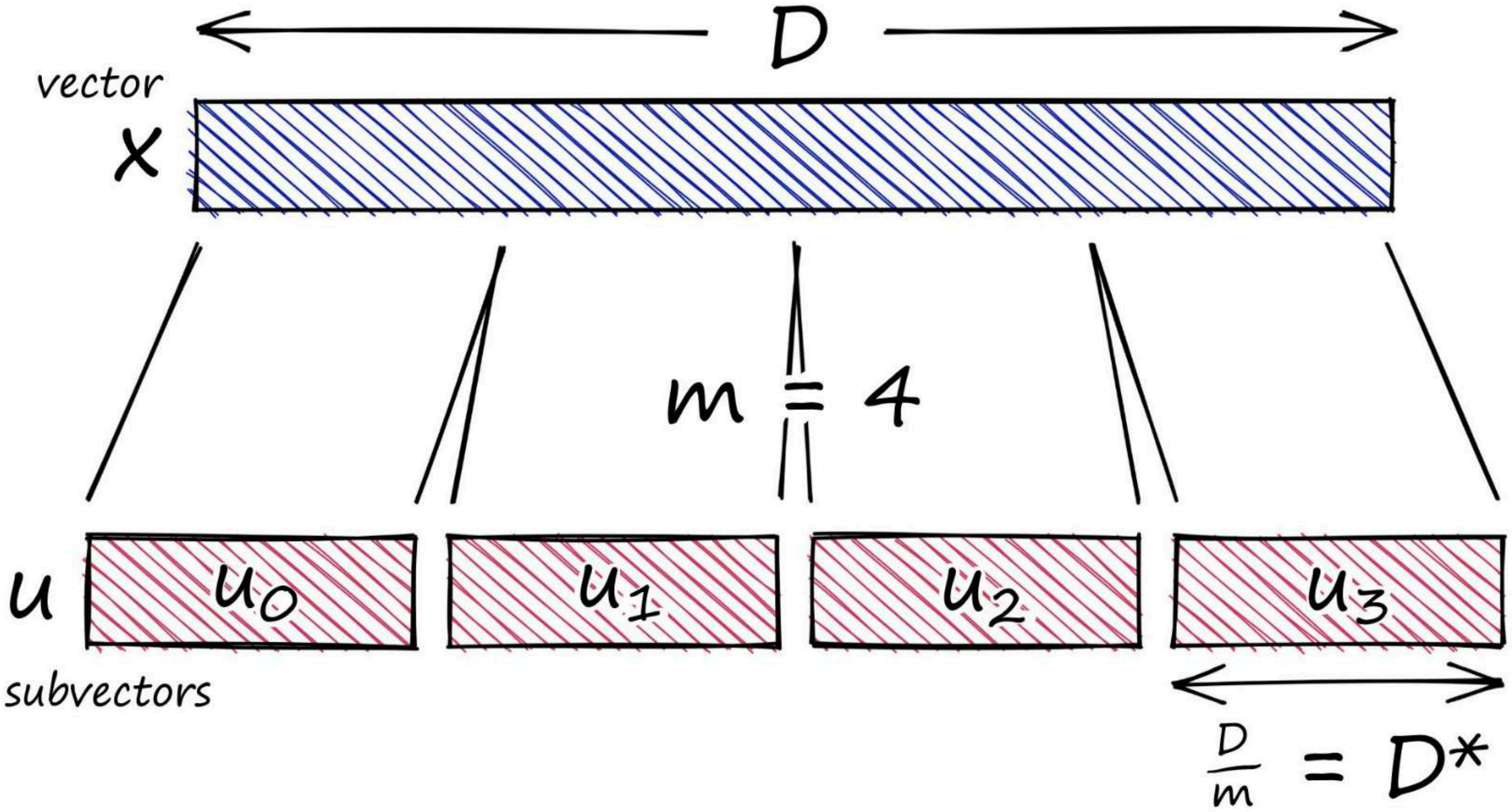
Clustering





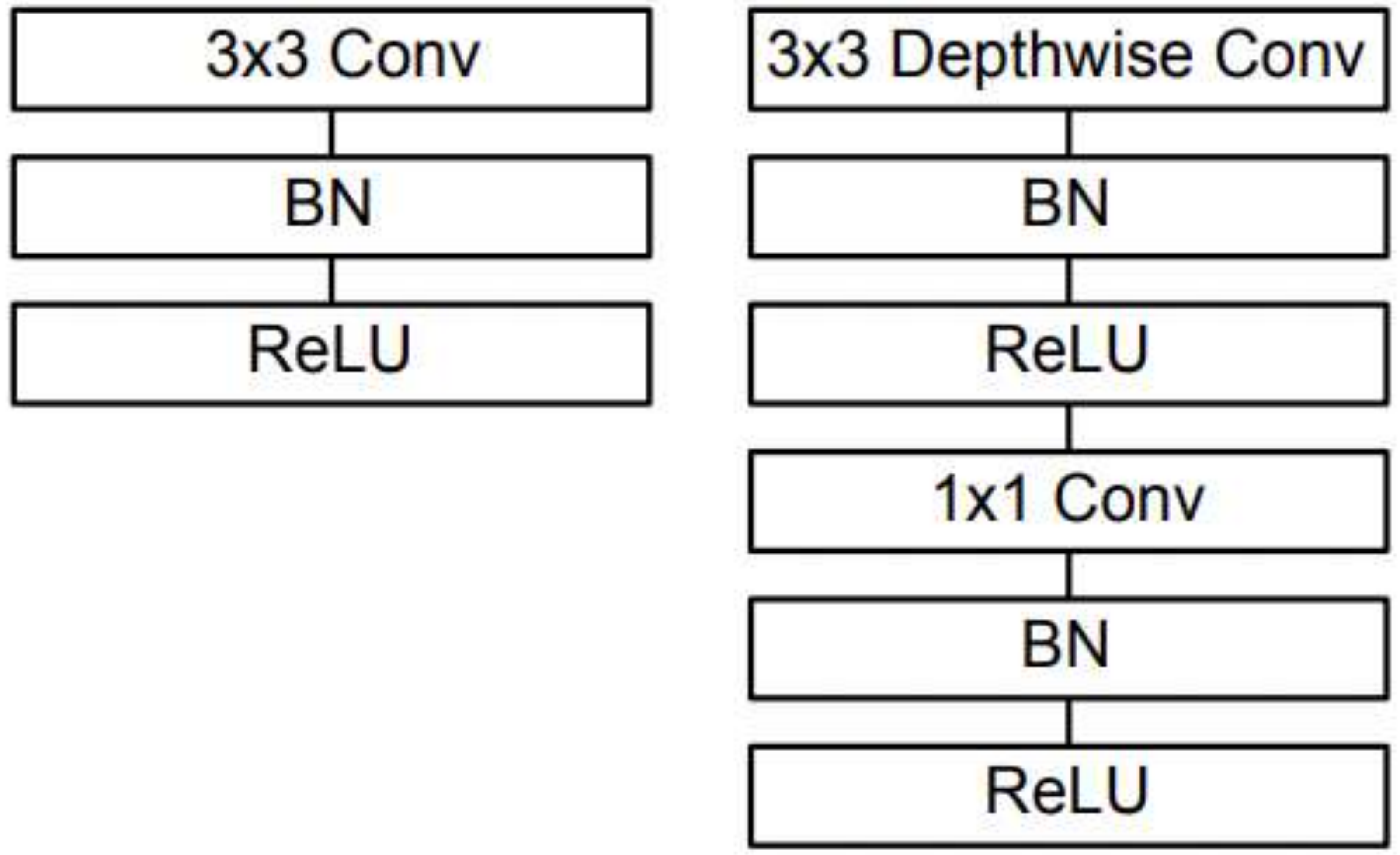
Quantization

Product

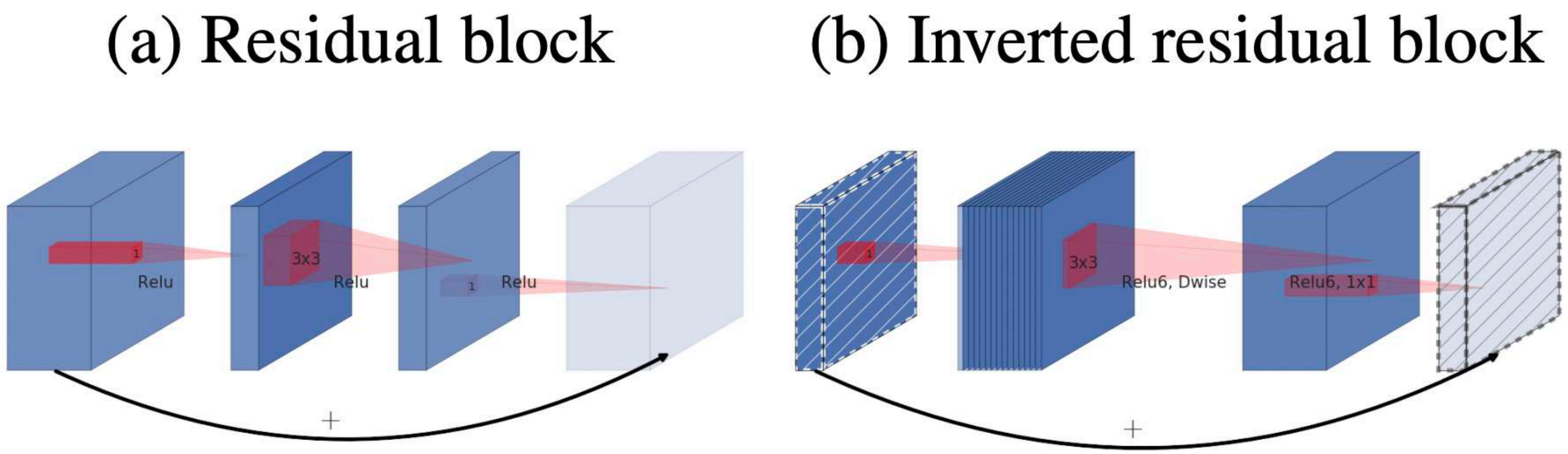


Efficient Neural Architecture

MobileNet

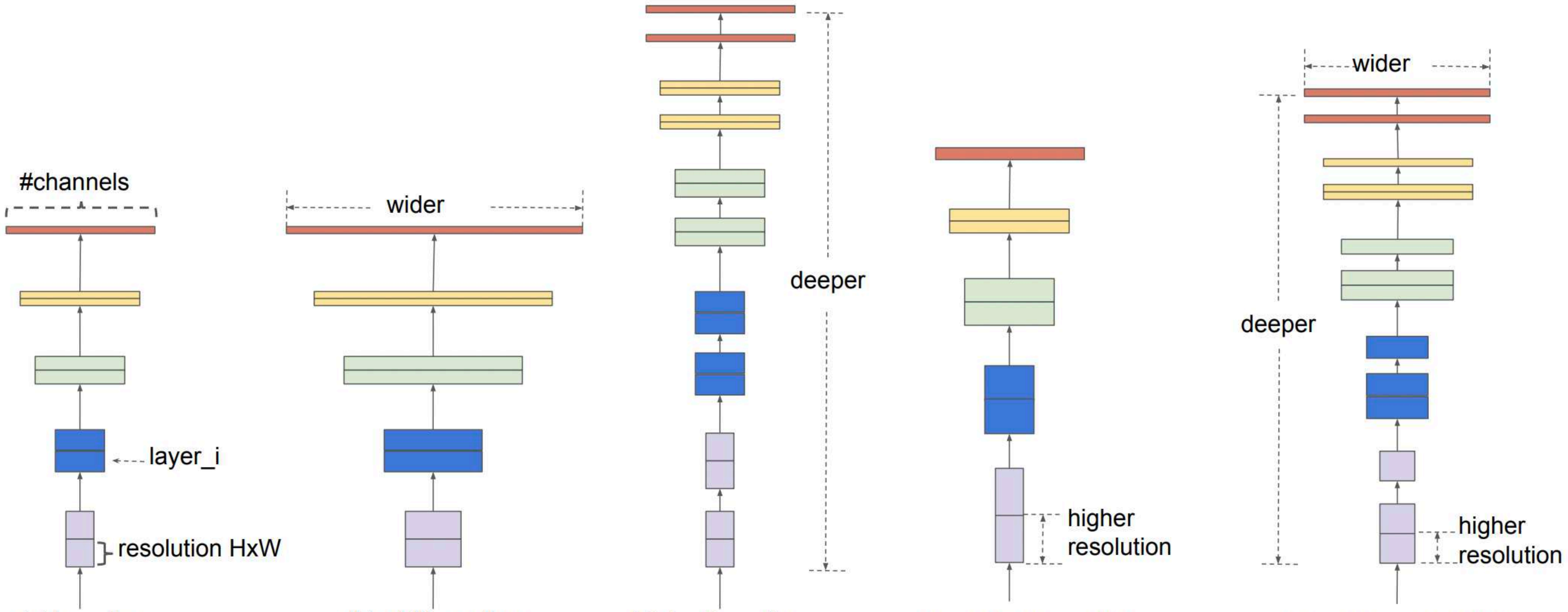


	Params	MACs	Arithmetic intensity
Spatial conv	MNK^2	$BMNK^2F^2$	$\frac{BMNK^2F^2}{BF^2(M+N)+K^2MN}$
Spatially separable conv	MNK	$BMNKF^2$	$\frac{BMNKF^2}{BF^2(M+N)+KMN}$
Pointwise conv	MN	$BMNF^2$	$\frac{BMNF^2}{BF^2(M+N)+MN}$
Group conv	MNK^2/G	$BMNK^2F^2/G$	$\frac{BMNK^2F^2/G}{BF^2(M+N)+K^2MN/G}$
Depthwise conv	MK^2	BMK^2F^2	$\frac{BMK^2F^2}{2BMF^2+K^2M}$



Efficient Neural Architecture

EfficientNet



(a) baseline (b) width scaling (c) depth scaling (d) resolution scaling (e) compound scaling

$$\max_{d,w,r} \quad Accuracy\big(\mathcal{N}(d,w,r)\big)$$

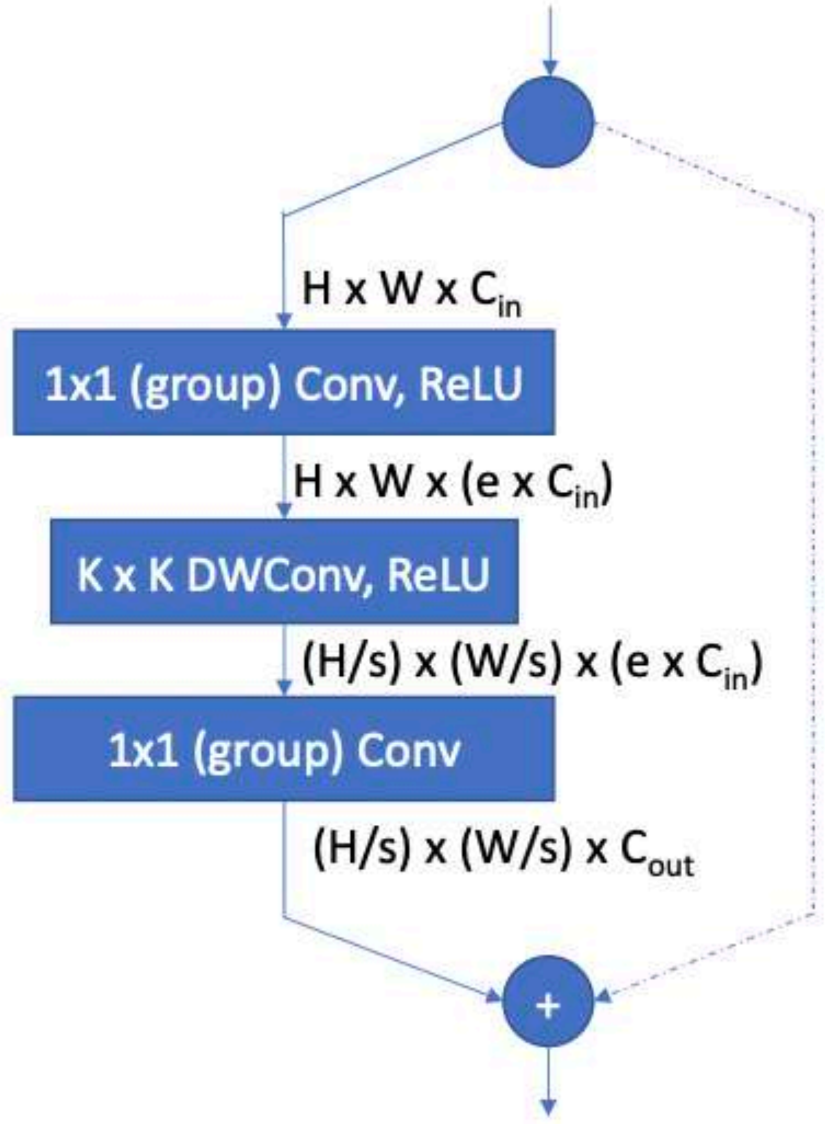
$$s.t. \quad \mathcal{N}(d,w,r) = \bigodot_{i=1\ldots s} \hat{\mathcal{F}}_i^{d\cdot\hat{L}_i}\big(X_{\langle r\cdot\hat{H}_i, r\cdot\hat{W}_i, w\cdot\hat{C}_i\rangle}\big)$$

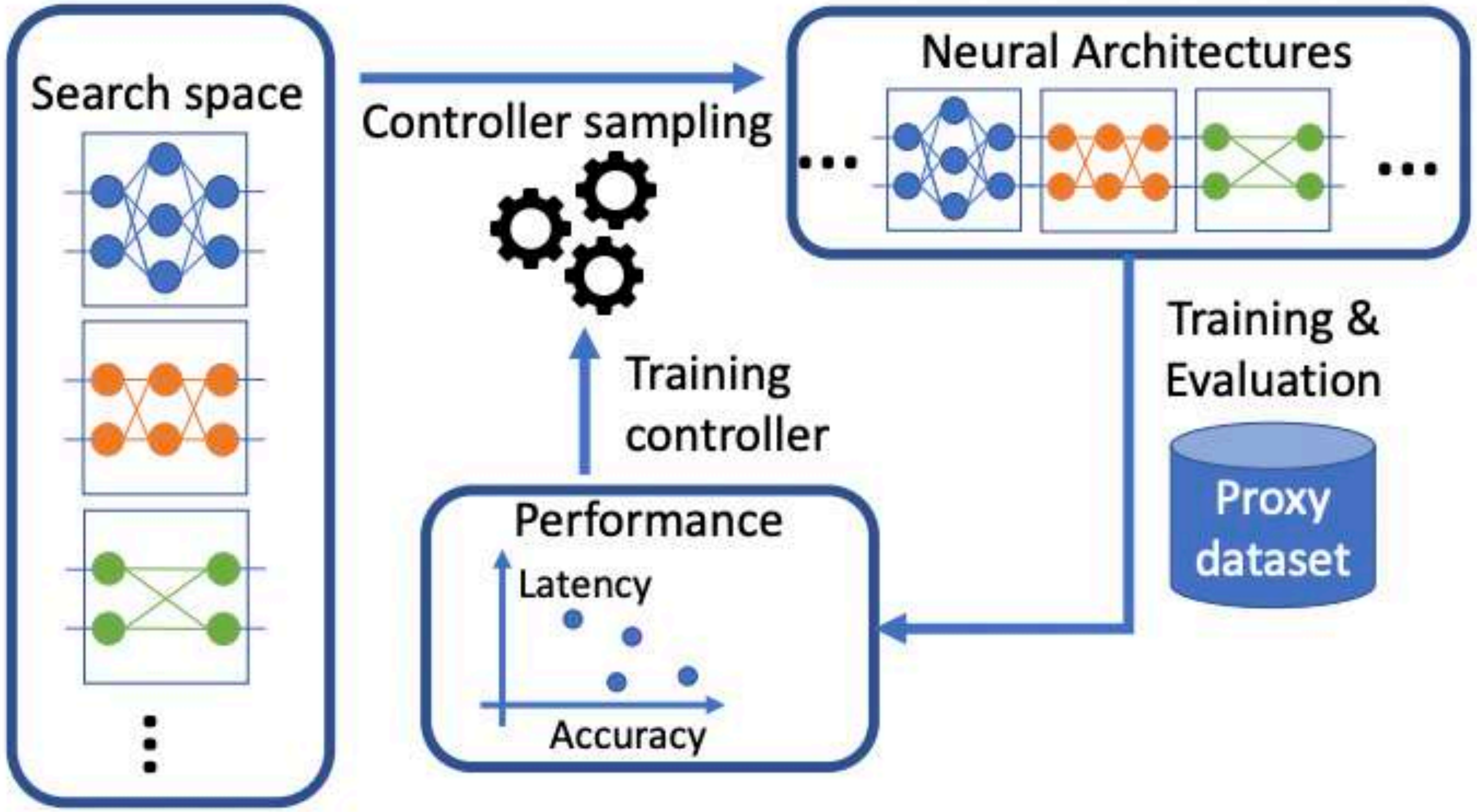
$$\text{Memory}(\mathcal{N}) \leq \text{target\_memory}$$

$$\text{FLOPS}(\mathcal{N}) \leq \text{target\_flops}$$

Neural Architecture Search

FBNet





$$\mathcal{L}(a, w_a) = \mathrm{CE}(a, w_a) \cdot \alpha \log(\mathrm{LAT}(a))^\beta$$

$$\min_{\boldsymbol{\theta}} \min_{w_a} \mathbf{E}_{a \sim P_{\boldsymbol{\theta}}}\{\mathcal{L}(a, w_a)\},\ P \text{ is GumbelSoftmax}$$

Examples

TinyBERT: Distilling BERT for NLU

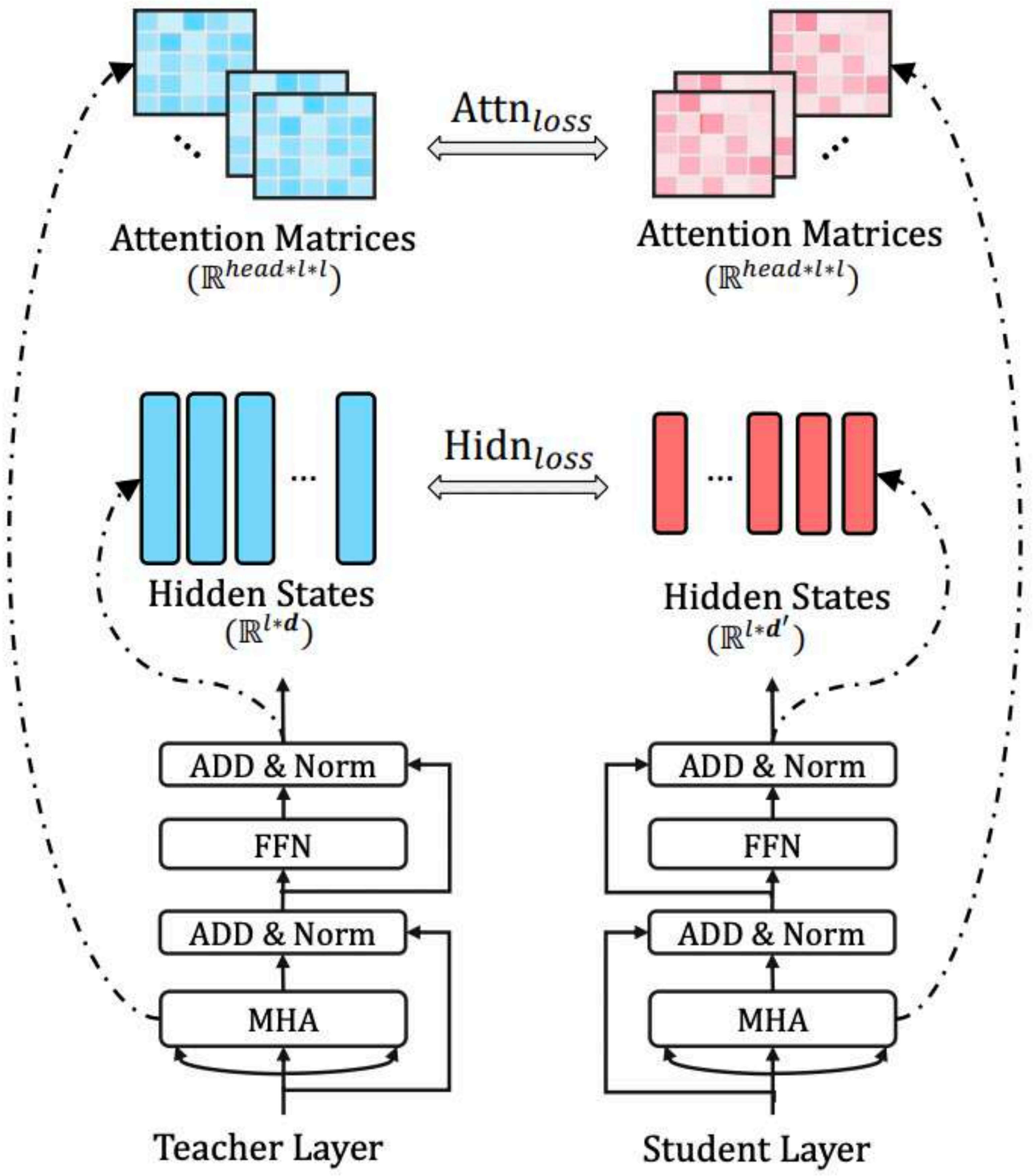
$$\mathcal{L}_{\text{attn}} = \frac{1}{h}\sum_{i=1}^{h} \text{MSE}(\boldsymbol{A}_i^S, \boldsymbol{A}_i^T)$$

$$\mathcal{L}_{\text{hidn}} = \text{MSE}(\boldsymbol{H}^S W_h, \boldsymbol{H}^T)$$

$$\mathcal{L}_{\text{embd}} = \text{MSE}(\boldsymbol{E}^S \boldsymbol{W}_e, \boldsymbol{E}^T)$$

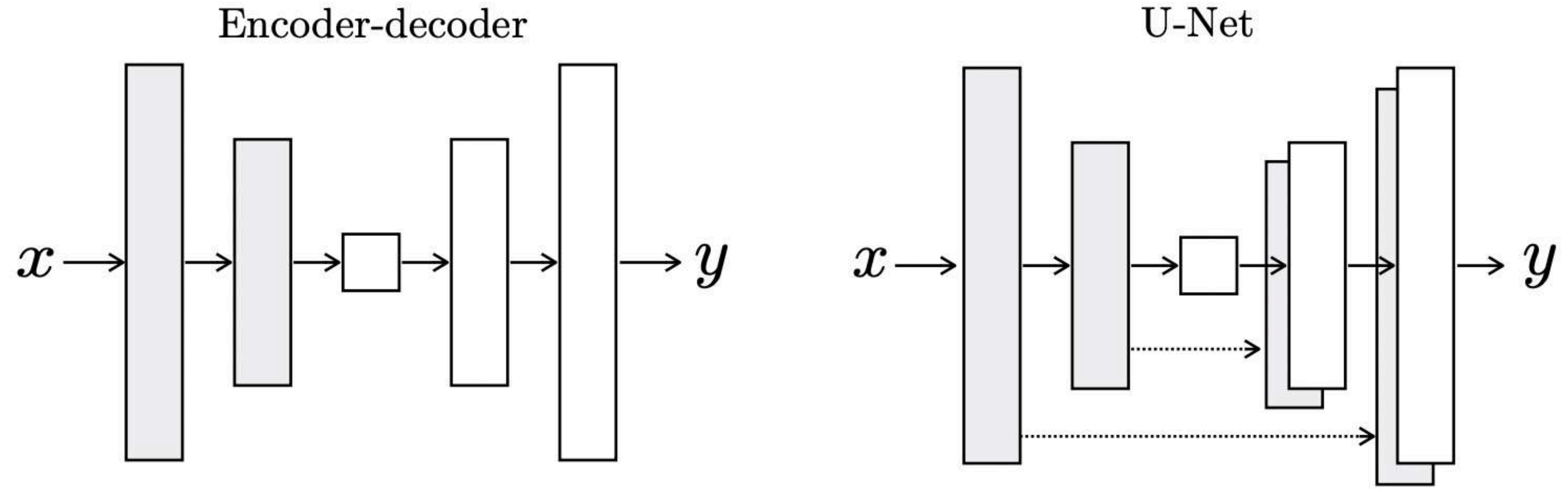
$$\mathcal{L}_{\text{pred}} = \text{CE}(\boldsymbol{z}^T/t, \boldsymbol{z}^S/t)$$

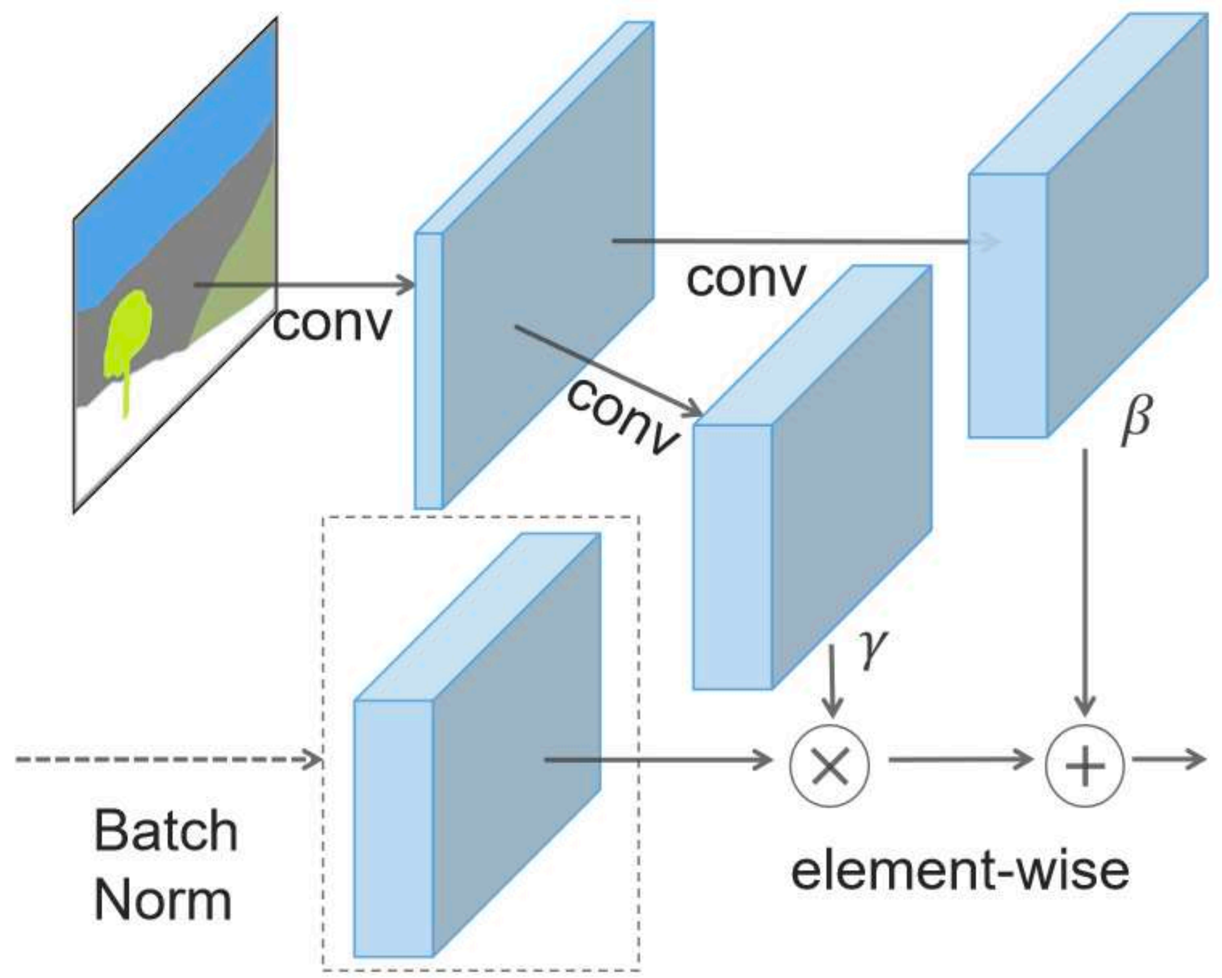
$$\mathcal{L}_{\text{model}} = \sum_{x \in \mathcal{X}} \sum_{m=0}^{M+1} \lambda_m \mathcal{L}_{\text{layer}}\left(f_m^S(x), f_{g(m)}^T(x)\right)$$

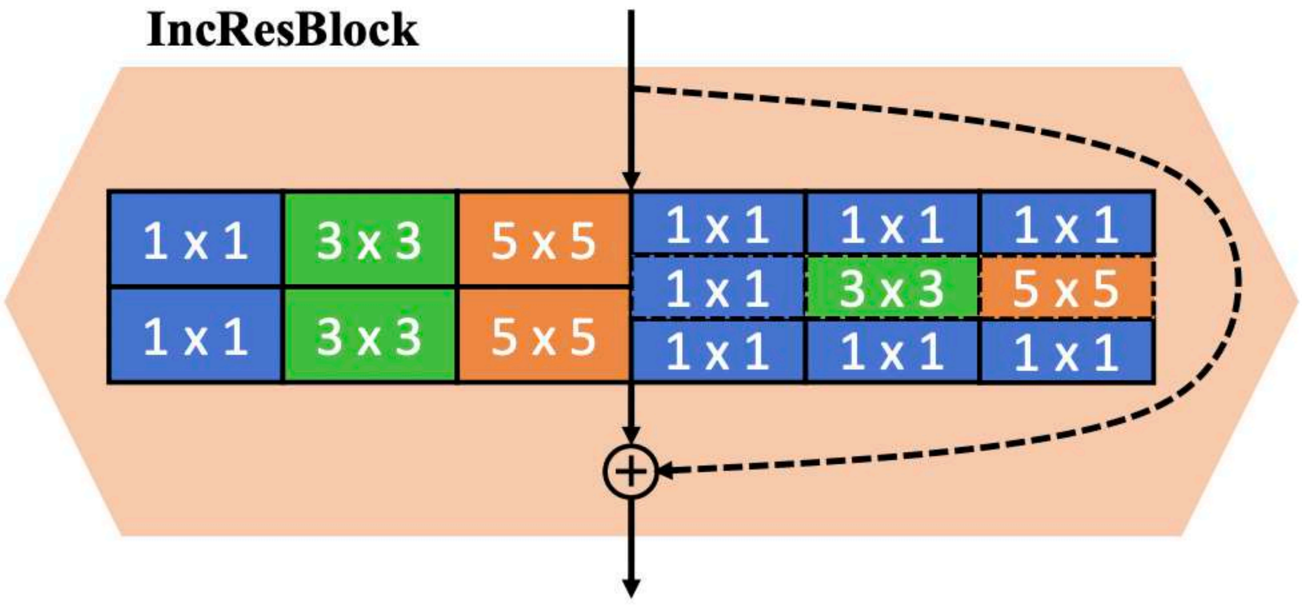


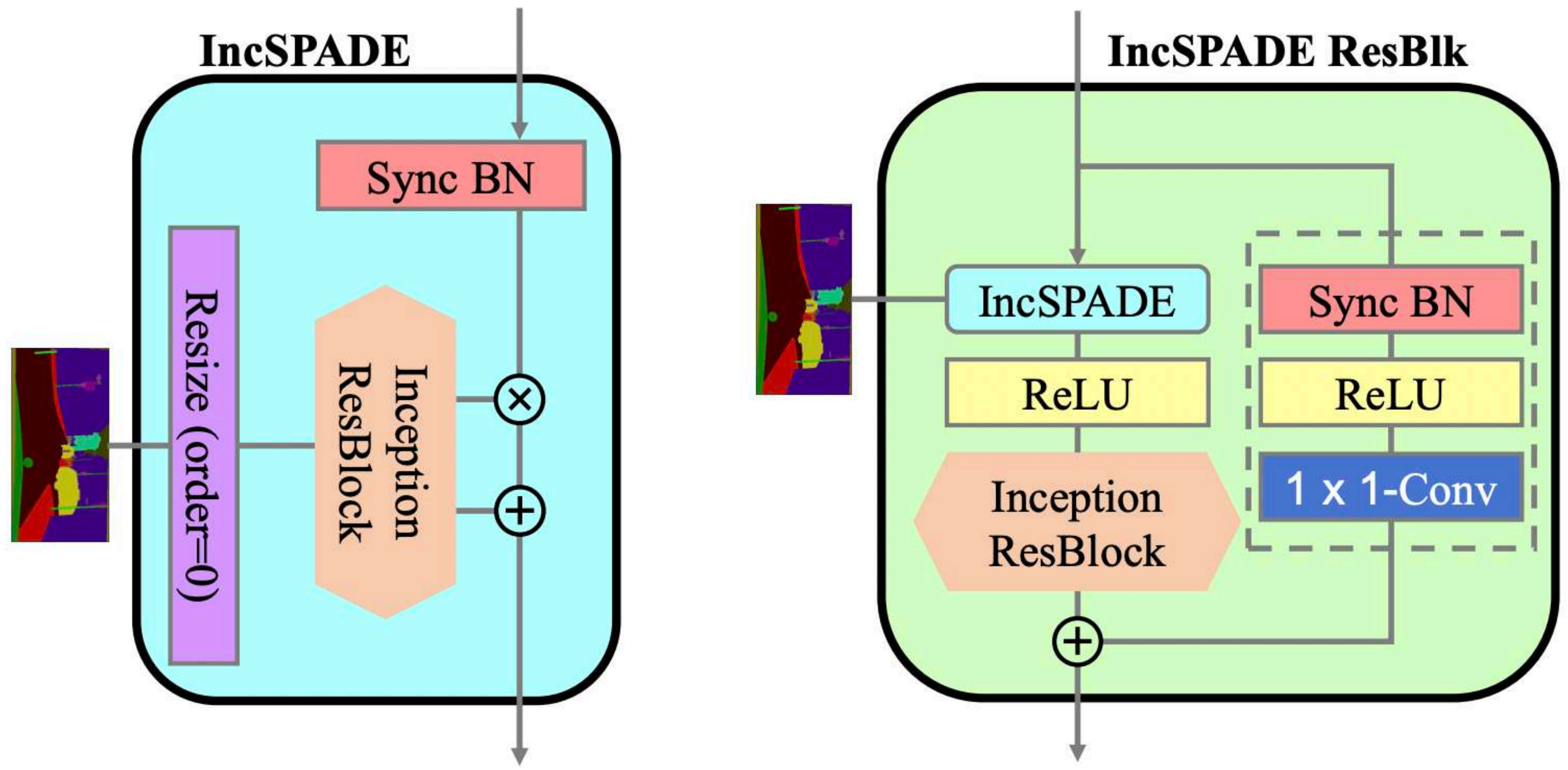
Examples

Teachers Do More Than Teach: Compressing Image-to-Image Models









Examples

Teachers Do More Than Teach: Compressing Image-to-Image Models

Algorithm 1 Searching via One-Step Pruning.

Require: Computational budget T_{b}, teacher model $\mathrm{G_T}$, scaling factors $\gamma_i^{(l)}$ (used for pruning) of the i-th channel in normalization layers $N^{(l)} \in \mathrm{G_T}$, minimum # output channels c_{lb} for convolution layers (outside the IncResBlock).

Ensure: pruned student architecture $\mathrm{G_S}$.

Initialize scale lower bound γ_{lo}: $\gamma_{lo} \leftarrow \min\limits_{i,l} |\gamma_i^{(l)}|$.

Initialize scale upper bound γ_{hi}: $\gamma_{hi} \leftarrow \max\limits_{i,l} |\gamma_i^{(l)}|$.

while $\gamma_{lo} < \gamma_{hi}$ **do**

    $\gamma_{th} \leftarrow (\gamma_{lo} + \gamma_{hi})/2$

    Prune channels satisfying $|\gamma_i^{(l)}| < \gamma_{th}$ on $\mathrm{G_T}$ while keep c_{lb} to get $\mathrm{G_S}$

    $T \leftarrow$ computational cost of $\mathrm{G_S}$

    if $T > T_{\mathrm{b}}$ **then**

        $\gamma_{lo} \leftarrow \gamma_{th}$

    else

        $\gamma_{hi} \leftarrow \gamma_{th}$

    end if

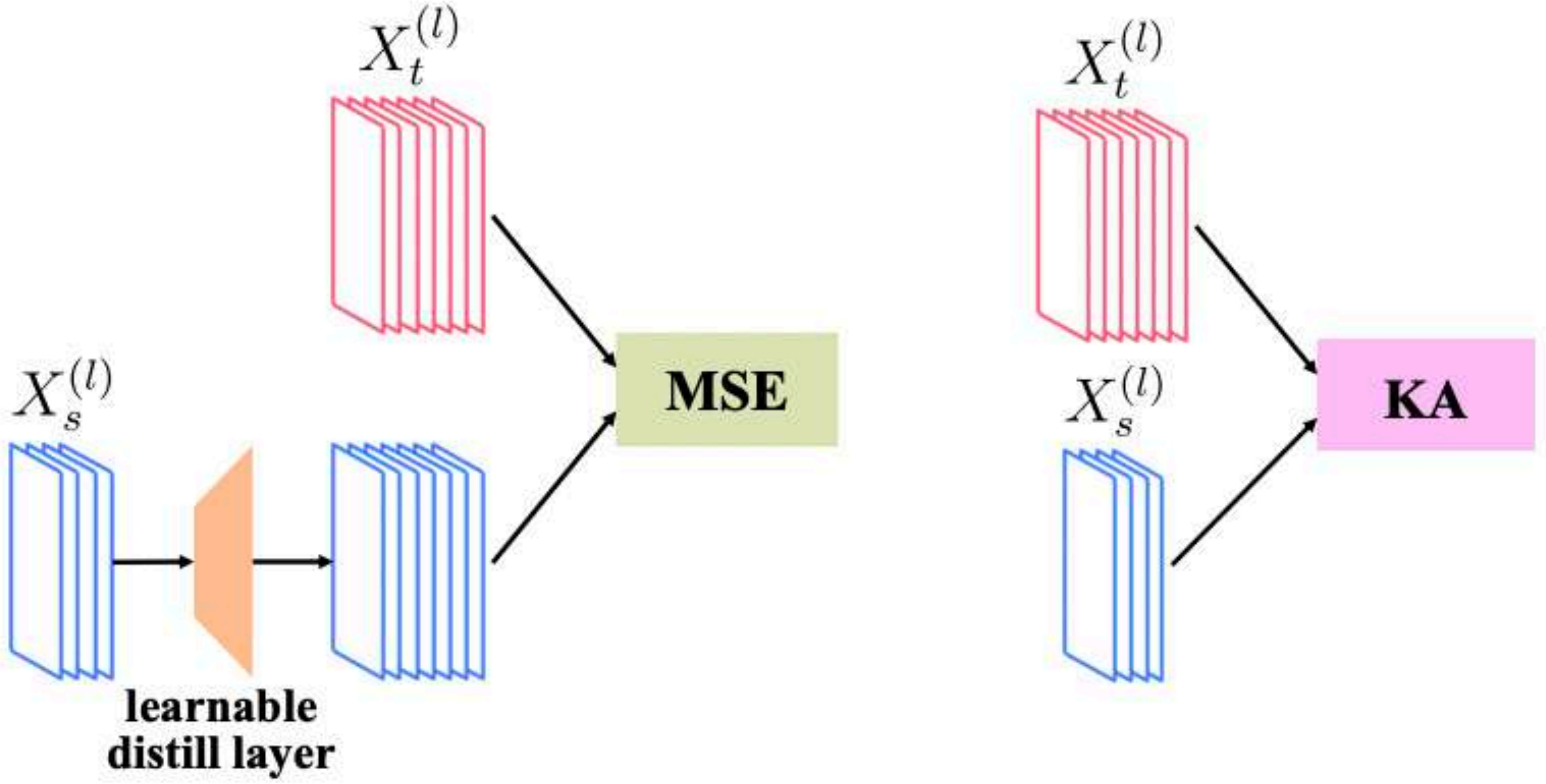
end while

Examples

Teachers Do More Than Teach: Compressing Image-to-Image Models

$$\mathrm{KA}(X, Y) = \frac{\|Y^{\mathrm{T}} X\|_{\mathrm{F}}^2}{\|X^{\mathrm{T}} X\|_{\mathrm{F}} \|Y^{\mathrm{T}} Y\|_{\mathrm{F}}}$$

$$\mathcal{L}_{\text{dist}} = - \sum_{l \in \mathcal{S}_{\text{KD}}} \mathrm{KA}\left(X_t^{(l)}, X_s^{(l)}\right)$$



Examples

Teachers Do More Than Teach: Compressing Image-to-Image Models

Table 1: Quantitative comparison between different compression techniques for Image-to-Image models. We use mIoU to evaluate the generation quality of Cityspaces and FID for other datasets. Higher mIoU or lower FID indicates better performance.

Model	Dataset	Method	MACs	FID↓	mIoU↑
CycleGAN	Horse→Zebra	Original [85, 36]	56.8B	61.53	-
		Shu *et al.* [64]	13.4B	96.15	-
		AutoGAN Distiller [20]	6.39B	83.60	-
		GAN Slimming [70]	11.25B	86.09	-
		GAN Lottery [11]	~11.35B†	~83.00†	-
		Li *et al.* [36]	2.67B	71.81	-
		CAT (Ours)	**2.56B**	**53.48**	-
Pix2pix	Cityscapes	Original [29, 36]	56.8B	-	42.06
		Li *et al.* [36]	5.66B	-	40.77
		CAT (Ours)	**5.57B**	-	**42.65**
	Map→Aerial photo	Original [29, 36]	56.8B	47.76	-
		Li *et al.* [36]	4.68B	48.02	-
		CAT (Ours)	**4.59B**	**45.63**	-
GauGAN	Cityscapes	Original [58, 36]	281B	-	62.18
		Li *et al.* [36]	31.7B	-	61.22
		CAT-A (Ours)	**29.9B**	-	**62.35**
		CAT-B (Ours)	5.52B	-	54.71