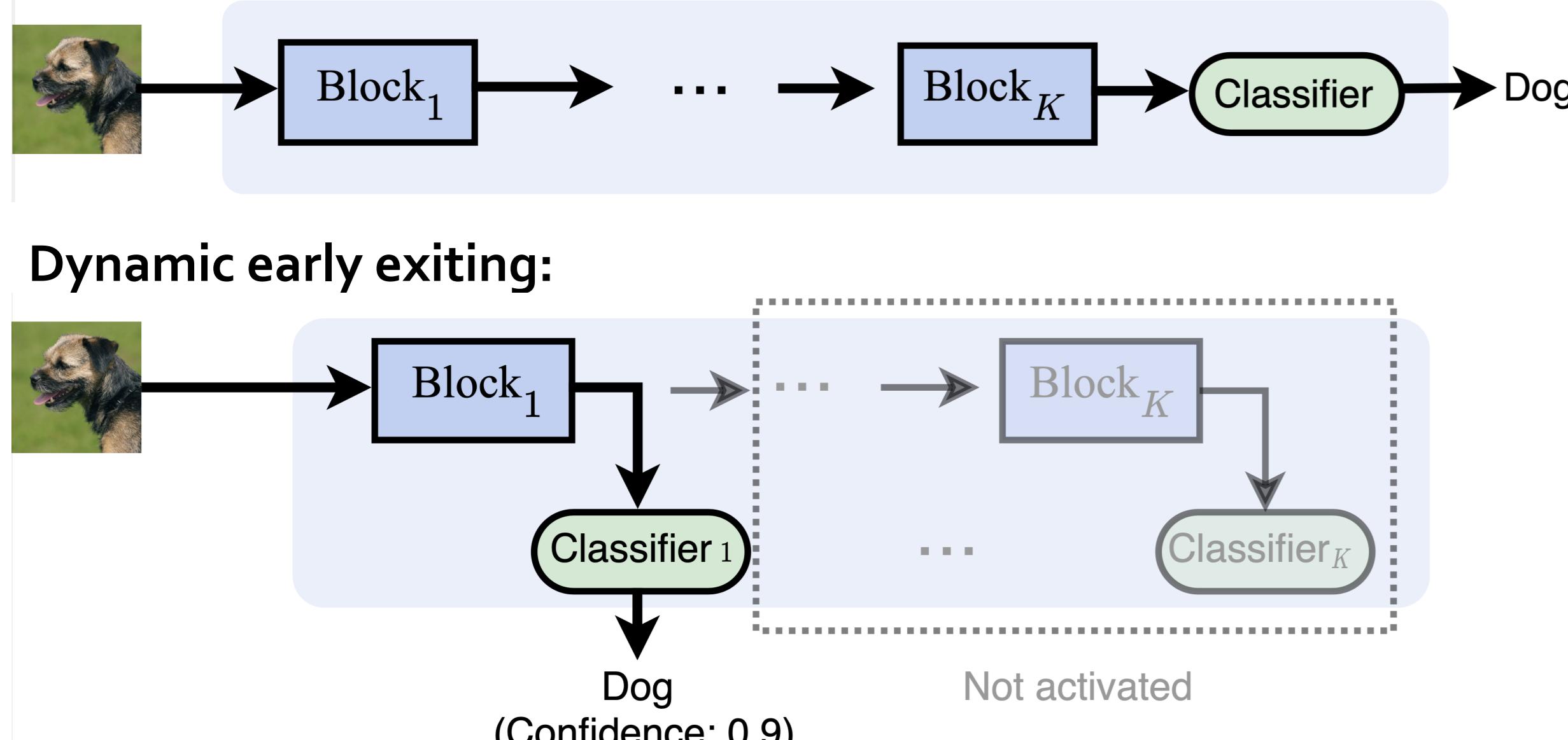


Background

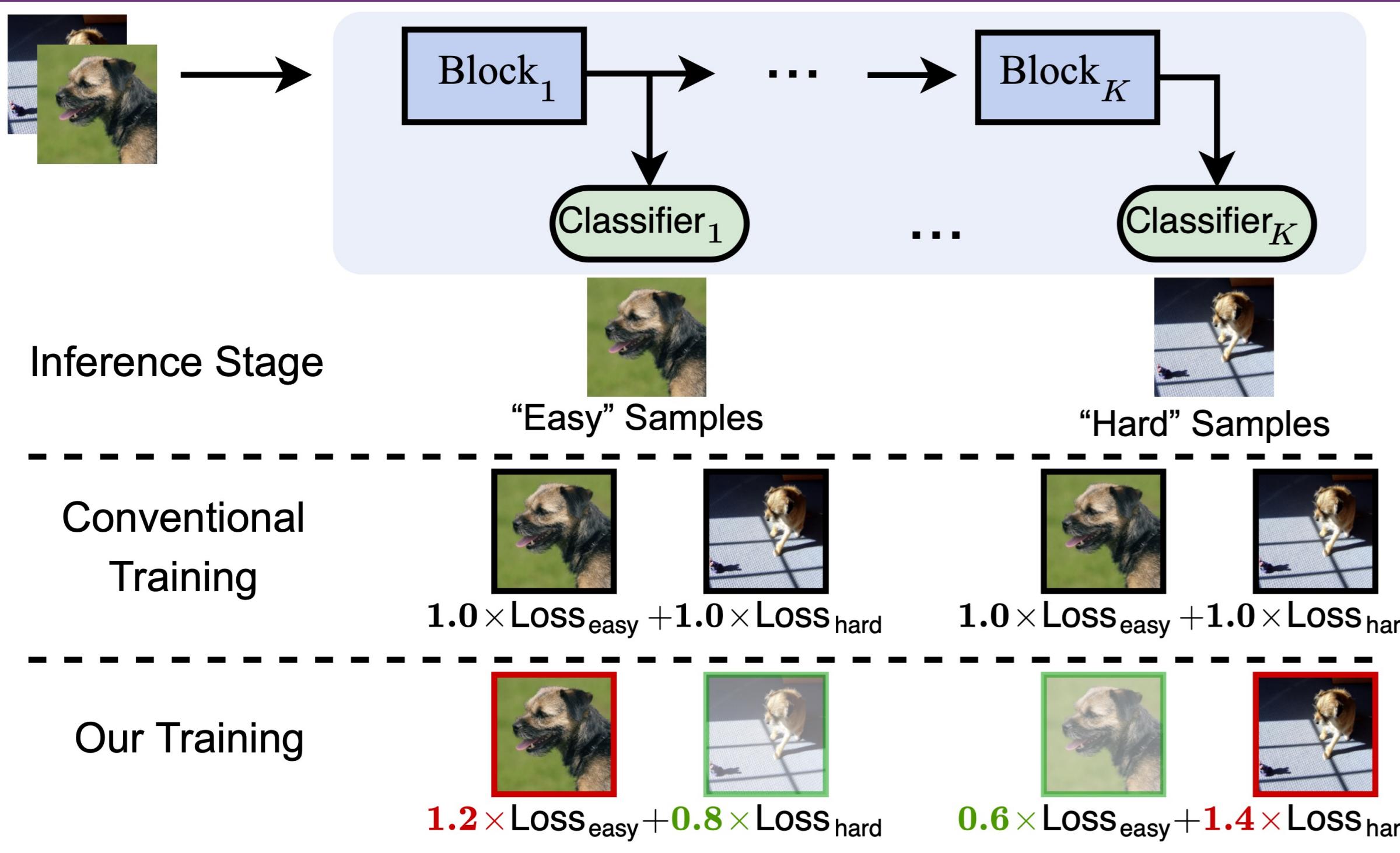
Conventional deep network:



[Observation]

- “Easy” samples can be output at early exits (classifiers) at test time
- Different exits are responsible with samples with different complexity

Motivation



- [Intuition] Samples with varying complexity should contribute unequally to exits (classifiers) with varying capacity in training.
- [Solution] Sample weighting.

\mathbf{x}_i, y_i : the i -th sample and its ground-truth label

$f(\cdot; \Theta_f)$: multi-exit classification network.

$f^{(k)}(\cdot; \Theta_f^{(k)})$: the k -th sub-network contained in f . $k = 1, 2, \dots, K$.

Conventional training objective:

$$\mathcal{L} = \sum_{k=1}^K \frac{1}{N} \sum_{i=1}^N l_i^{(k)}, \quad \text{where } l_i^{(k)} = \text{CE}\left(f^{(k)}\left(\mathbf{x}_i; \Theta_f^{(k)}\right), y_i\right)$$

Our training objective with sample weighting:

$$\mathcal{L} = \sum_{k=1}^K \frac{1}{N} \sum_{i=1}^N w_i^{(k)} l_i^{(k)}$$

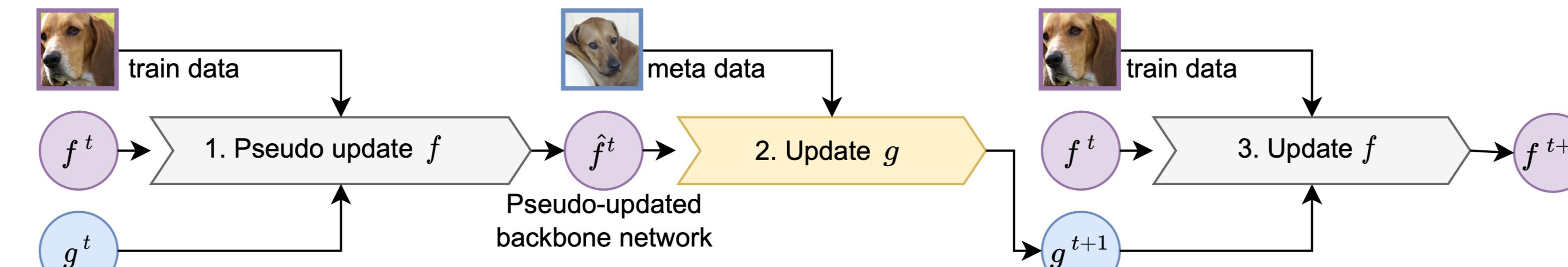
Weight prediction network g : $w_i = g(\mathbf{l}_i; \Theta_g)$,

where $w_i = [w_i^{(1)}, w_i^{(2)}, \dots, w_i^{(K)}]$,

$$\mathbf{l}_i = [l_i^{(1)}, l_i^{(2)}, \dots, l_i^{(K)}]$$

Method

Training pipeline



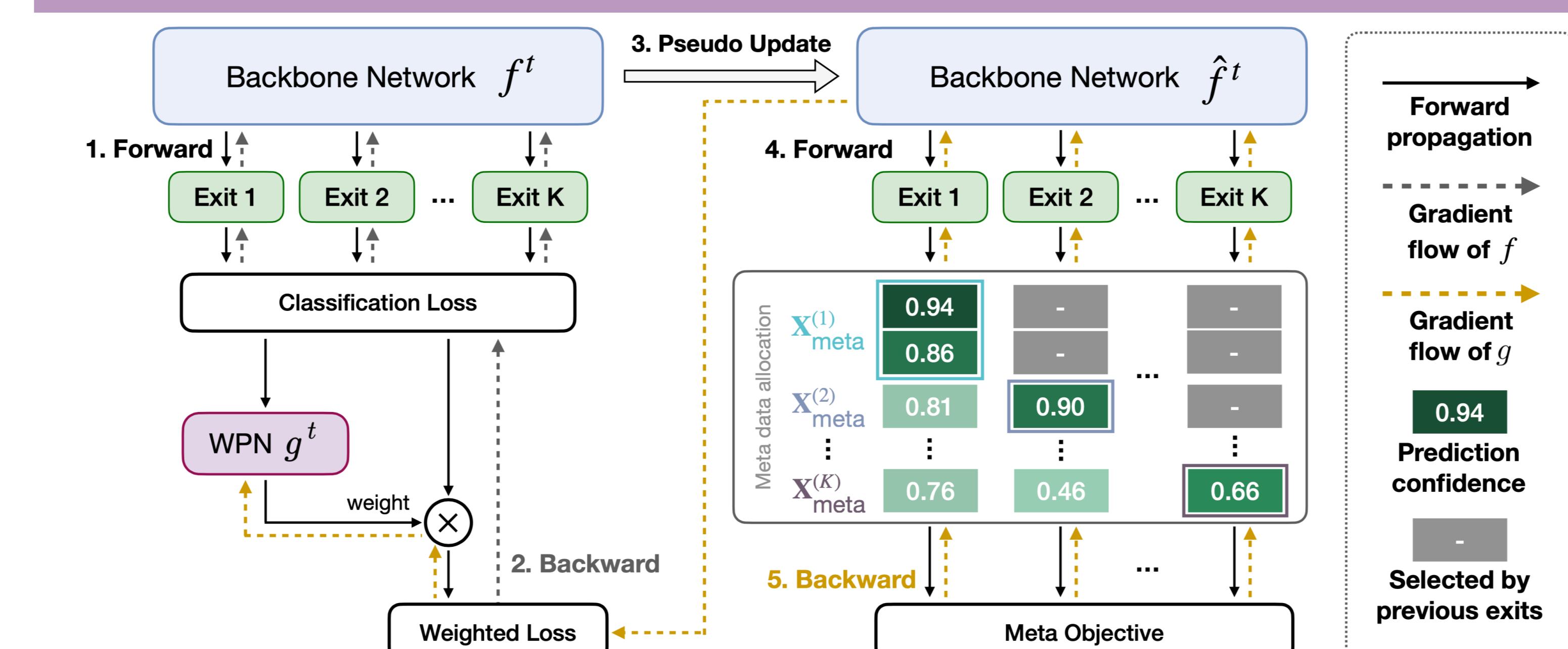
Meta-learning Objective

$$\begin{aligned} \Theta_f^*(\Theta_g) &= \underset{\Theta_f}{\operatorname{argmin}} \mathcal{L}_{\text{tr}}(\Theta_f, \Theta_g) \\ &\triangleq \underset{\Theta_f}{\operatorname{argmin}} \sum_{k=1}^K \frac{1}{N} \sum_{i: \mathbf{x}_i \in \mathbf{X}_{\text{tr}}} g^{(k)}(\mathbf{l}_i; \Theta_g) \cdot l_i^{(k)}(\Theta_f). \\ \Theta_g^* &= \underset{\Theta_g}{\operatorname{argmin}} \mathcal{L}_{\text{meta}}(\Theta_f^*(\Theta_g)) \\ &\triangleq \underset{\Theta_g}{\operatorname{argmin}} \sum_{k=1}^K \frac{1}{N_k} \sum_{j: \mathbf{x}_j \in \mathbf{X}_{\text{meta}}^{(k)}} l^{(k)}(\mathbf{x}_j, y_j; \Theta_f^*(\Theta_g)). \end{aligned}$$

Meta-learning Optimization

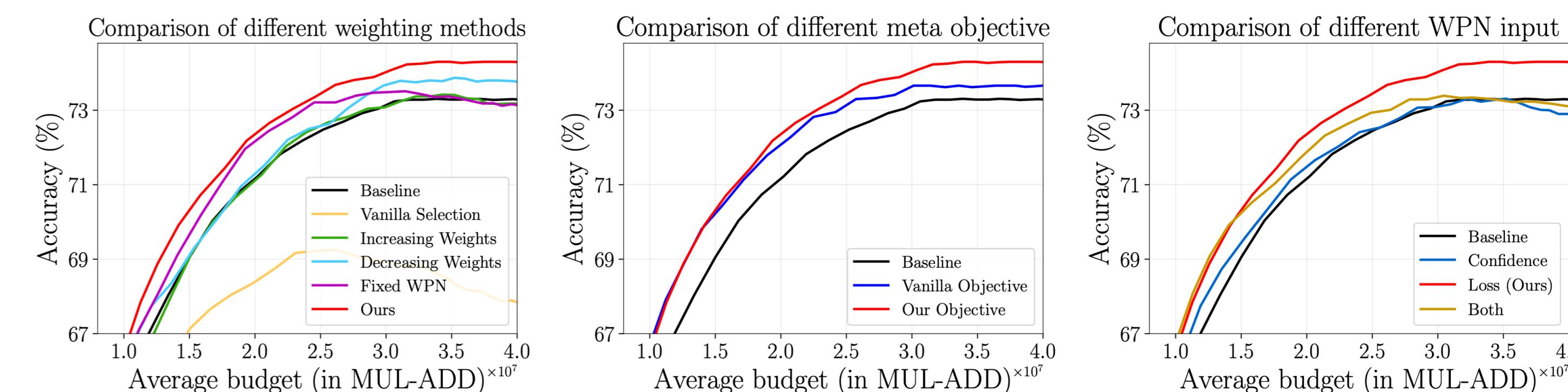
$$\begin{aligned} \widehat{\Theta}_f^t(\Theta_g) &= \Theta_f^t - \alpha \frac{\partial \mathcal{L}_{\text{tr}}(\Theta_f, \Theta_g)}{\partial \Theta_f} \Big|_{\Theta_f^t} \\ \Theta_g^{t+1} &= \Theta_g^t - \beta \frac{\partial \mathcal{L}_{\text{meta}}(\widehat{\Theta}_f^t(\Theta_g))}{\partial \Theta_g} \Big|_{\Theta_g^t} \\ \Theta_f^{t+1} &= \Theta_f^t - \alpha \frac{\partial \mathcal{L}_{\text{tr}}(\Theta_f, \Theta_g^{t+1})}{\partial \Theta_f} \Big|_{\Theta_f^t} \end{aligned}$$

Gradient Flow



Experiments

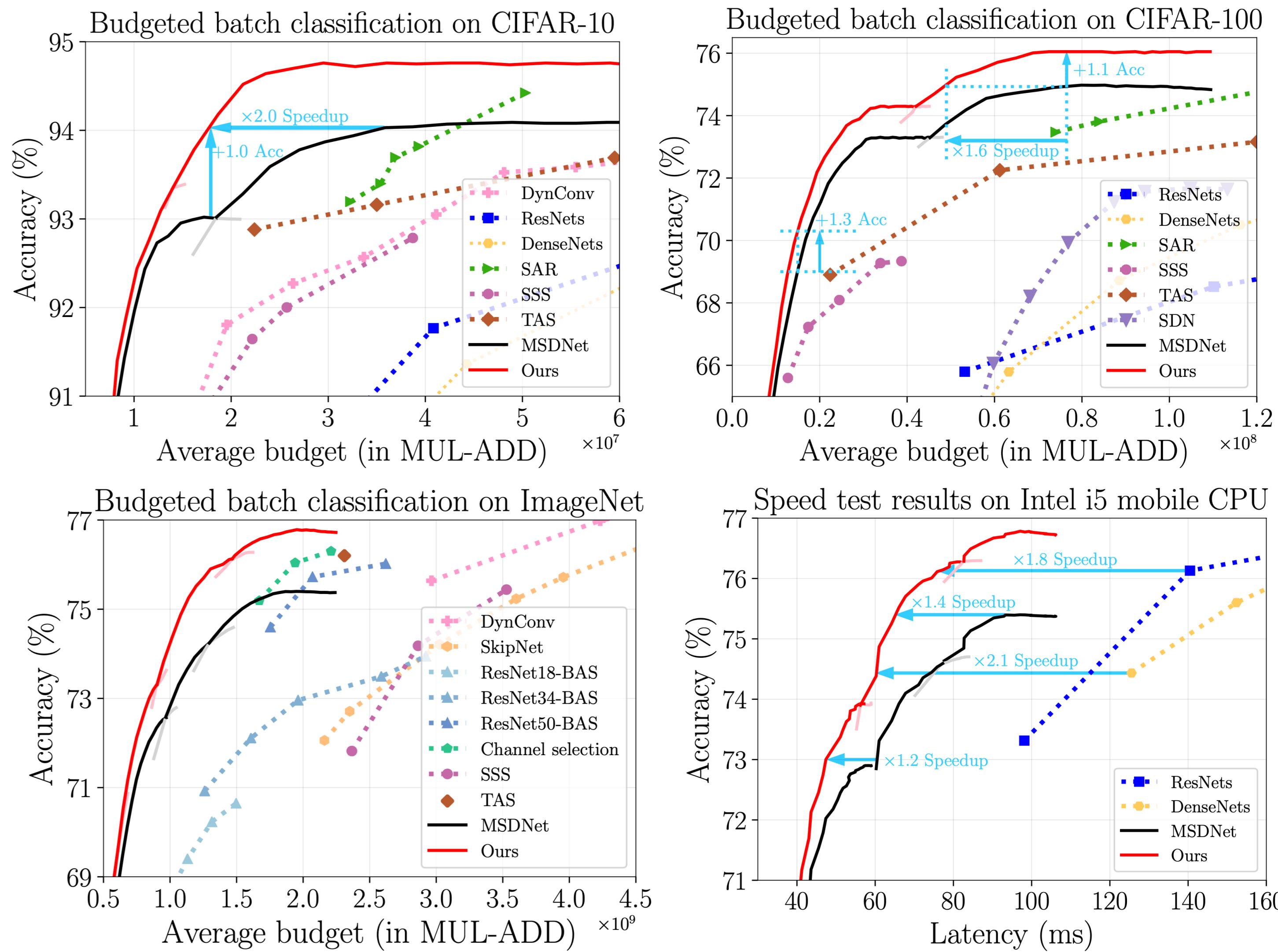
Ablation Studies



- Sample weighting > Sample selection ;
- Predicted weights > Hand-designed weights ;
- Meta-learning procedure > Frozen (learned) weight prediction;
- Our specialized meta objective > naive meta objective;
- Using loss to produce weights is preferable to using confidence.

Main Results

- Significantly improve the accuracy-efficiency tradeoff on different datasets;



- Compatible with other training techniques.

