Energy-efficient Amortized Inference with Cascaded Deep Classifiers

Jiaqi Guan (Tsinghua)
Yang Liu (UIUC)
Qiang Liu (UT Austin)
Jian Peng (UIUC)

2018.7.16

Introduction

- Deep neural networks have been remarkable successful in various AI tasks

- More complex architectures

  → further improved performance,

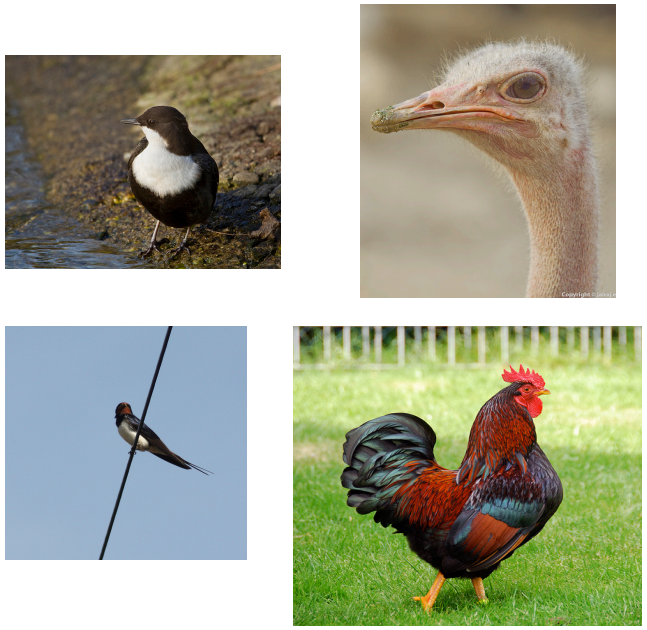
  but more expensive computation

- Constraint of computational cost for real-time inference

  → energy-efficient inference

Introduction

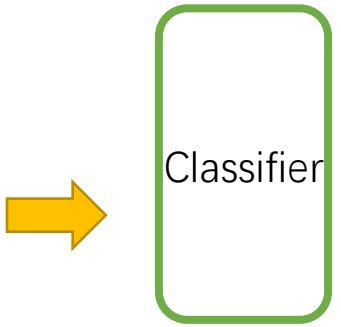
- Focus on test-time energy-efficient inference of image classification
- × sacrifice accuracy for speed
- ✓ predict both accurately and fast



Simple images

Small classifier

Difficult images



Large classifier

Related Work

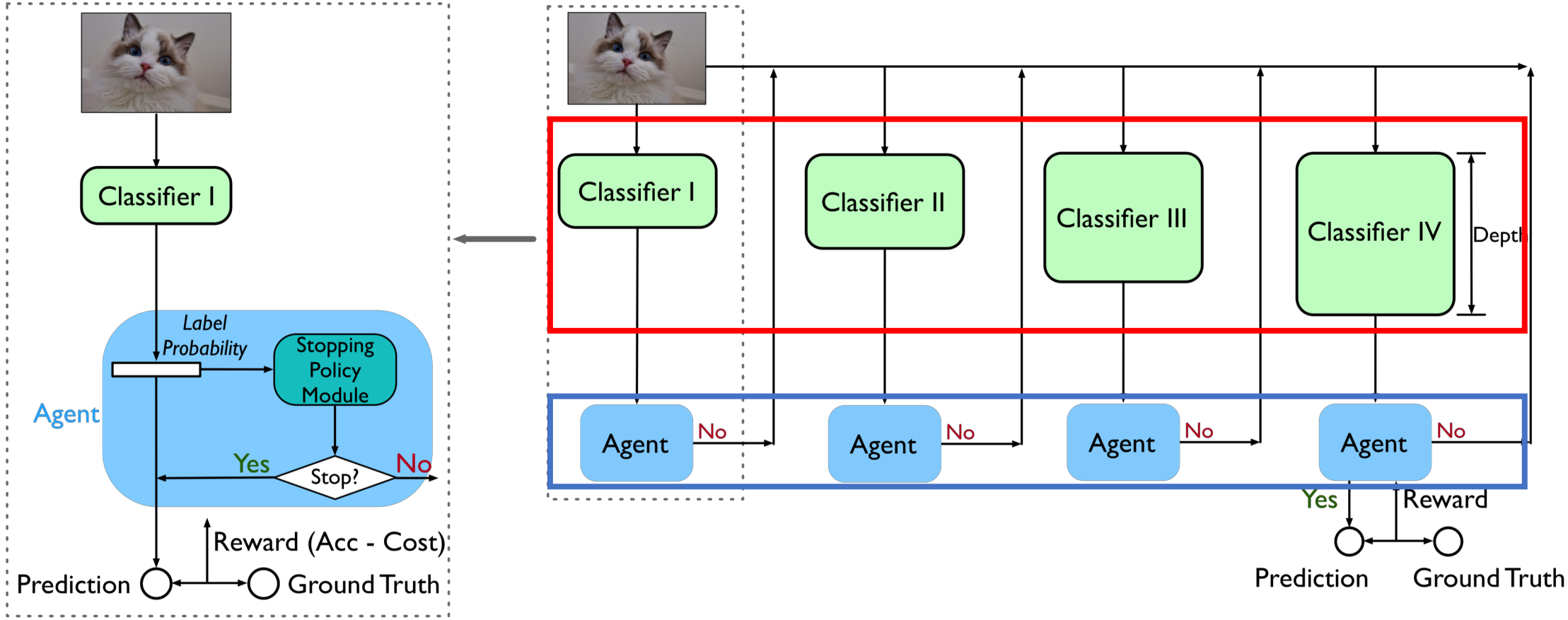
Static Techniques:

- simplifying network structure, e.g. MobileNet [Howard *et al.*, 2017]
- improving basic convolution operations numerically, e.g. ShuffleNet [Zhang *et al.*, 2017]

Dynamic Techniques:

- Anytime Neural Networks [Hu *et al.*, 2017]: generate anytime predictions by minimizing a carefully constructed weighted sum of losses.
- Adaptive Neural Network (ANN) [Bolukbasi *et al.*, 2017]: cascaded classifiers are trained and fixed. Only the termination policies are optimized to select the classifiers in a sequential manner.

Method


Classifier I
Label
Probability
Stopping
Policy
Module
Agent
Yes
No
Stop?
Reward (Acc - Cost)
Prediction
Ground Truth
Classifier I
Classifier II
Classifier III
Classifier IV
Depth
Agent
No
Agent
No
Agent
No
Agent
No
Yes
Reward
Prediction
Ground Truth

Method

- K classifiers $\{\mathrm{C_k}\}_{\mathrm{k=1}}^{K}$ with different energy cost $\{\mathrm{F_k}\}_{\mathrm{k=1}}^{K}$
- Input x with true label y, predicted label $\hat{y}$
- policy $\Pi(k|x)$

constrained optimization

$$\max_{\Pi,\{C_t\}_{t=1}^{K}} \mathbb{E}_{(x,y)\sim\mathcal{D},k_x\sim\Pi(\cdot|x),\hat{y}\sim C_{k_x}(\cdot|x)}\left[-\mathcal{L}\left(\hat{y},y\right)\right]$$
$$s.t \quad \mathbb{E}_{(x,y)\sim\mathcal{D},k_x\sim\Pi(\cdot|x)}\left[\mathcal{F}_{k_x}\right] < \mathcal{B},$$

(k_x : classifier ID assigned to x)

unconstrained optimization

$$\max_{\Pi,\{C_t\}_{t=1}^{K}} \mathbb{E}_{(x,y)\sim\mathcal{D},k_x\sim\Pi(\cdot|x),y'\sim C_{k_x}(\cdot|x)}\left[-\mathcal{L}\left(y',y\right)-\alpha\mathcal{F}_{k_x}\right]$$

(α controls the trade-off between the predictive loss function and the energy cost)

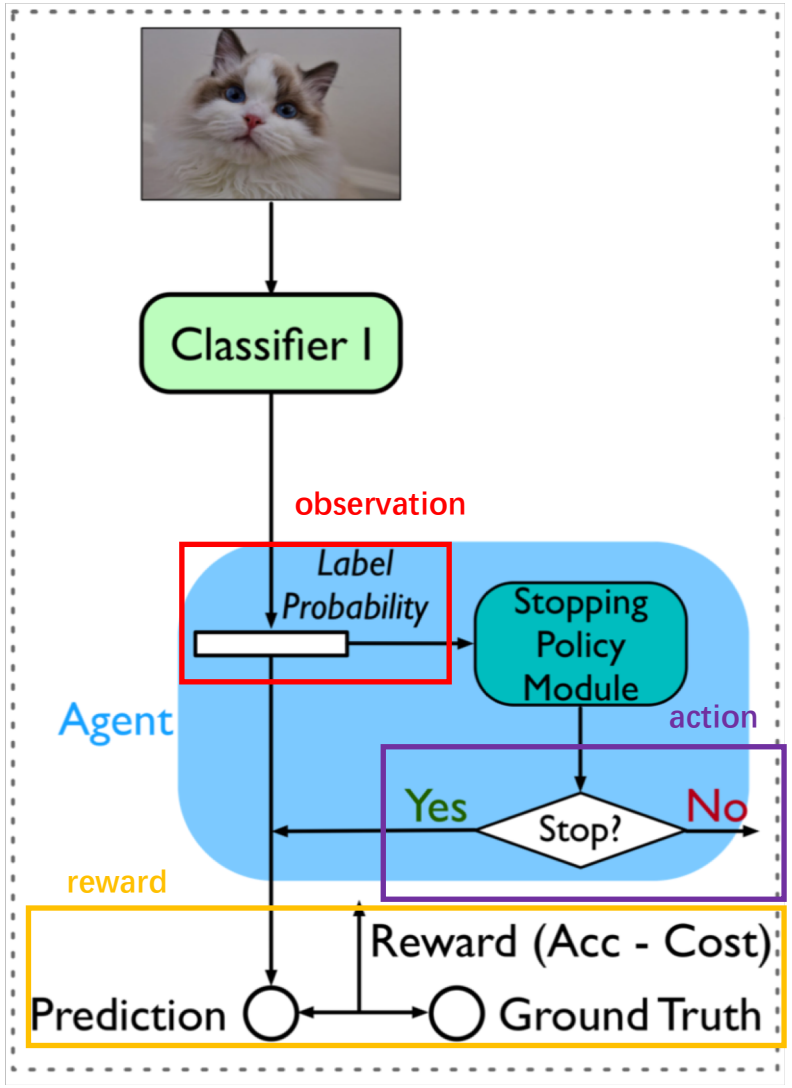
Method

- K-step optimal stopping process

$$\Pi(k|x) = \pi_k(s_k(x)) \prod_{t=1}^{k-1} (1 - \pi_t(s_t(x)))$$

($\mathbf{s_t}$: feature related to classifier $\mathbf{C_t}$, $\boldsymbol{\pi_t}$: stop probability)

- Markov decision process
  - Observation: $\mathbf{s_t}$, label probability
  - Action: stop / forward
  - Reward: $R(k, x, y, \hat{y}) = -\mathcal{L}(\hat{y}, y) - \alpha \sum_{t=1}^{k-1} \mathcal{F}_t$



Method

- Final goal:

$$\max_{\Pi,\{C_t\}_{t=1}^{K}} \mathbb{E}_{(x,y)\sim\mathcal{D},k_x\sim\Pi(\cdot|x),y'\sim C_{k_x}(\cdot|x)}\left[-\mathcal{L}\left(y',y\right)-\alpha\mathcal{F}_{k_x}\right]$$

maximize

$$\begin{aligned} J(\theta) &= \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathbb{E}_{k\sim\Pi(\cdot|x),\hat{y}\sim C_k(\cdot|x)}R(k,x,y,\hat{y})\right] \\ &= \mathbb{E}_{(x,y)\sim\mathcal{D}}\Big[\sum_{k=1}^{K}\prod_{t=1}^{k-1}\left(1-\pi_t(s_t(x);\theta)\right)\cdot\pi_k(s_k(x);\theta)\cdot\sum_{\hat{y}}C_k(\hat{y}|x;\theta)\cdot R(k,x,y,\hat{y})\Big] \end{aligned}$$

Solving by REINFORCE

$$\widehat{\nabla_\theta J} = \nabla_\theta\Big(\sum_{t=1}^{k-1}\log(1-\pi_t(s_t(x);\theta))+\log(\pi_k(s_k(x);\theta)) + \log(C_k(\hat{y}|x;\theta))\Big)\cdot\left(R(k,x,y,\hat{y})-b\right)$$

baseline (to reduce variance)

Experiment

- Compare with static ResNet classifiers:

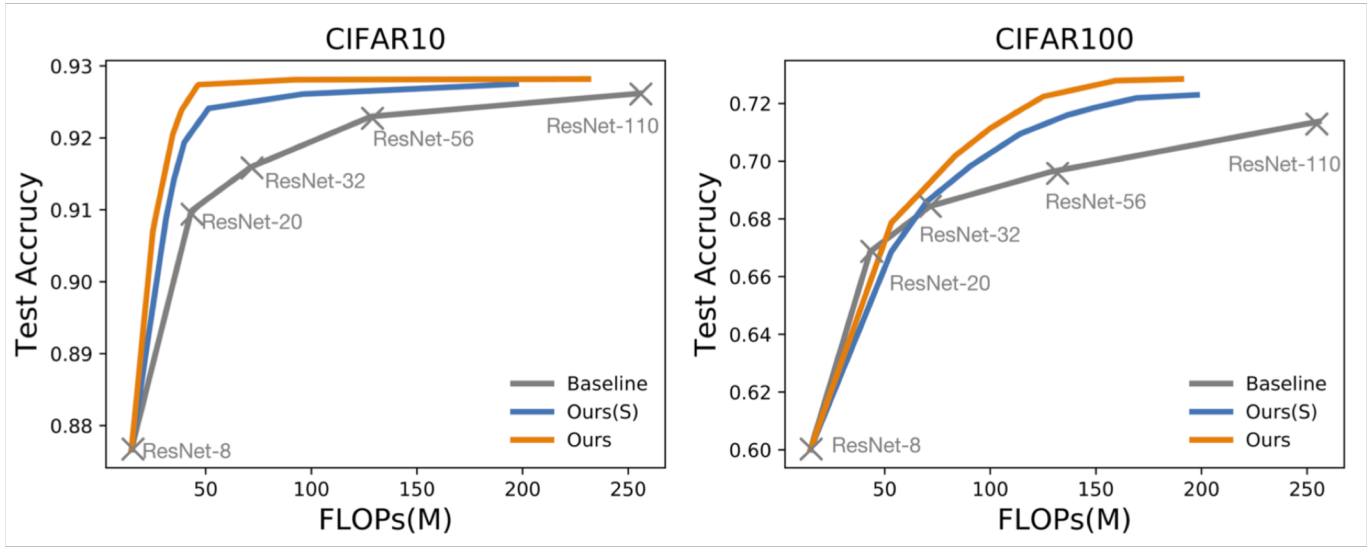
CIFAR-10		
Model	Error	FLOPs
ResNet-8	12.33%	5.82%
ResNet-20	9.00%	16.90%
ResNet-32	8.40%	27.98%
ResNet-56	7.70%	50.14%
ResNet-110	7.38%	100.00%
Ours	**7.26%**	**18.16%**

CIFAR-100		
Model	Error	FLOPs
ResNet-8	39.98%	5.82%
ResNet-20	33.13%	16.90%
ResNet-32	31.56%	27.98%
ResNet-56	30.38%	50.14%
ResNet-110	28.63%	100.00%
Ours	**27.76%**	**48.98%**

ImageNet32x32 (Top-5 Error)		
Model	Error	FLOPs
ResNet40-1	39.72%	6.27%
ResNet40-1.5	32.76%	14.09%
ResNet40-2	29.64%	25.03%
ResNet40-3	24.67%	56.27%
ResNet40-4	22.22%	100.00%
Ours	**22.21%**	**66.22%**

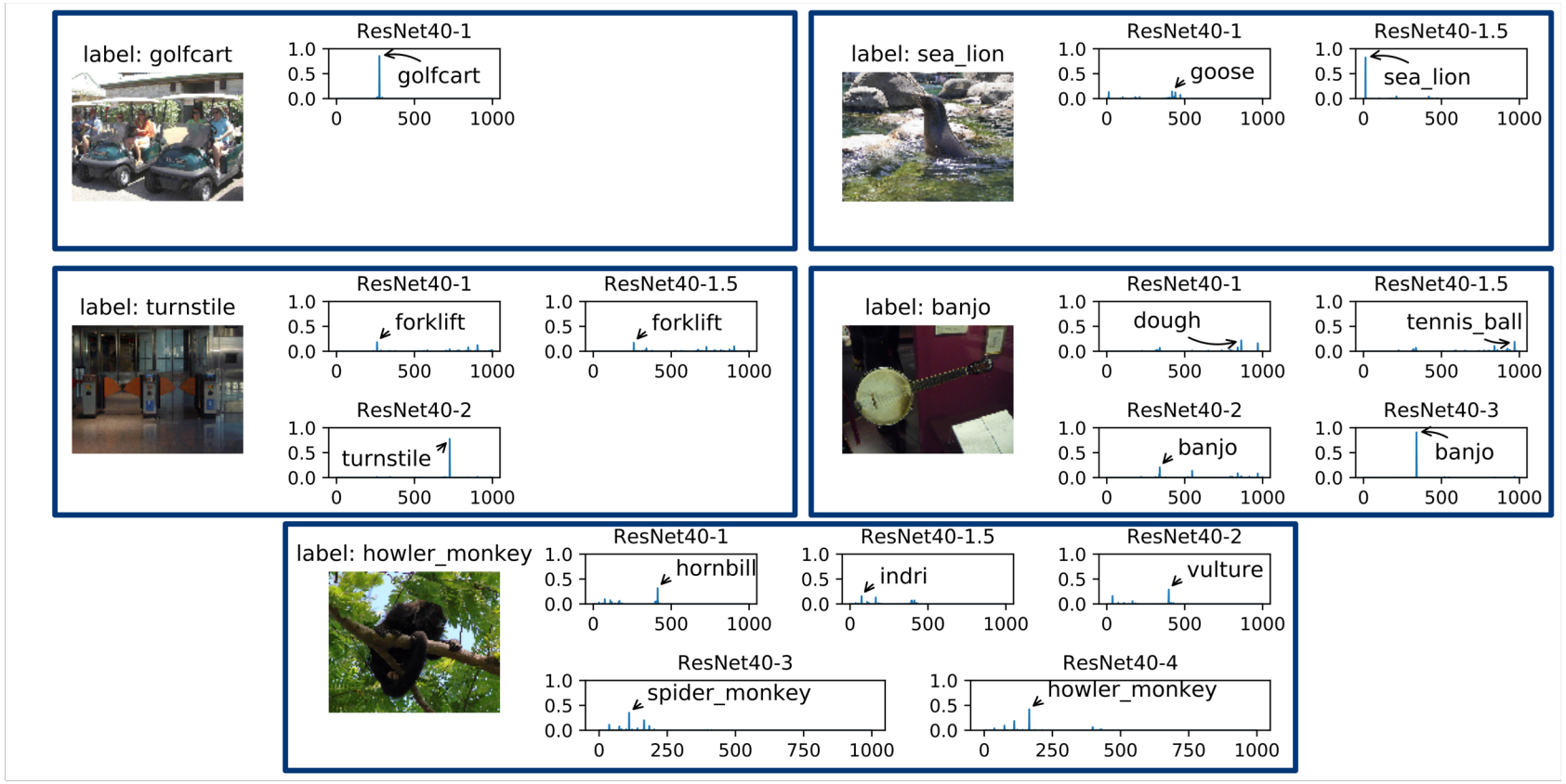
ImageNet (Top-5 Error)		
Model	Error	FLOPs
AlexNet	20.08%	18.42%
GoogleNet	10.99%	39.47%
ResNet-50	7.80%	100.00%
ANN	8.14%	56.87%
Ours	**7.82%**	**53.47%**

- Comparison of joint training:



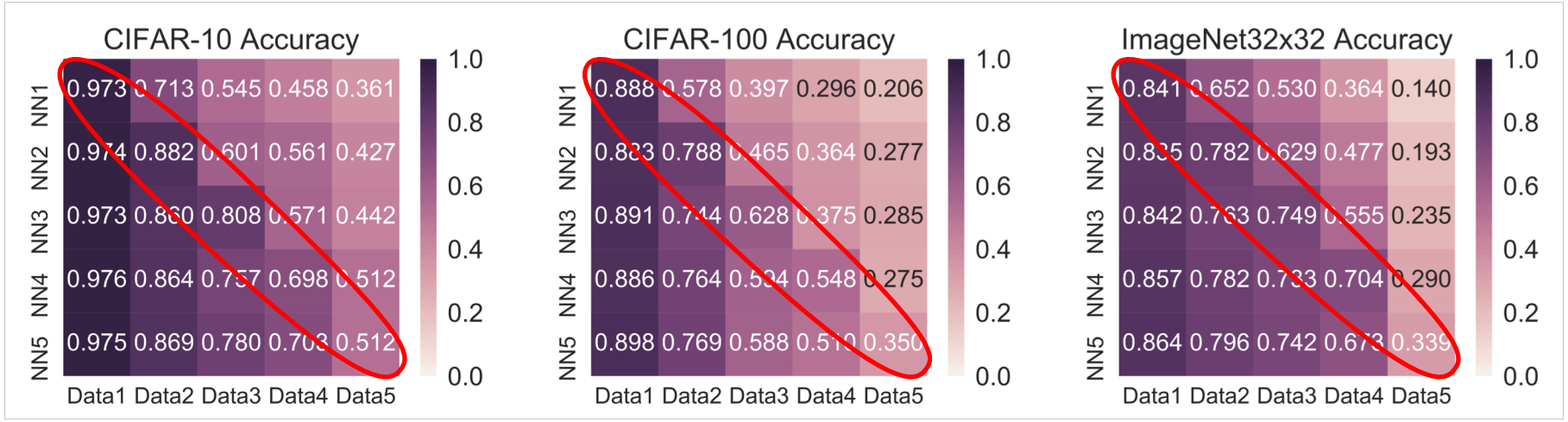
Experiment

- ImageNet Visualization

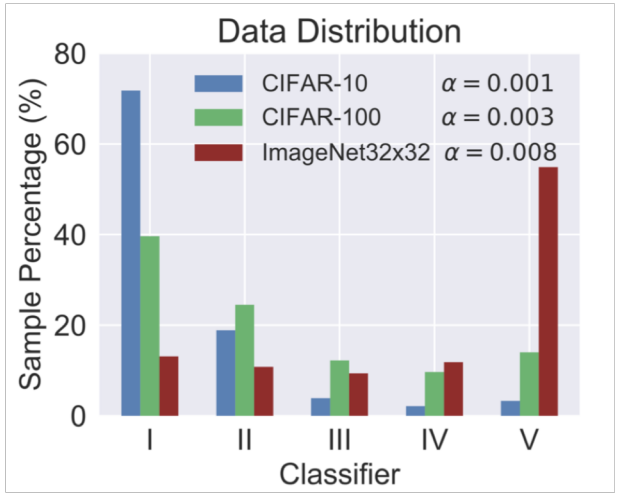


Experiment

- Accuracy Distribution



- Sample Distribution



Conclusion

- Propose an energy-efficient model by cascading deep classifiers with a policy module
- The policy module and cascading classifiers are **jointly** trained by ***REINFORCE*** to choose the smallest classifier which is sufficient to make accurate prediction for each input instance
- Achieve both high accuracy and amortized efficiency on CIFAR and ImageNet datasets

Thanks!
&
Questions?