NEO-KD: Knowledge-distillation-based Adversarial Training for Multi-exit Neural Network

Seokil Ham[1], Jungwuk Park[1], Dong-Jun Han[2*], Jaekyun Moon[1]

Korea Advanced Institute of Science and Technology[1]
Purdue University[2]

NEURAL INFORMATION PROCESSING SYSTEMS

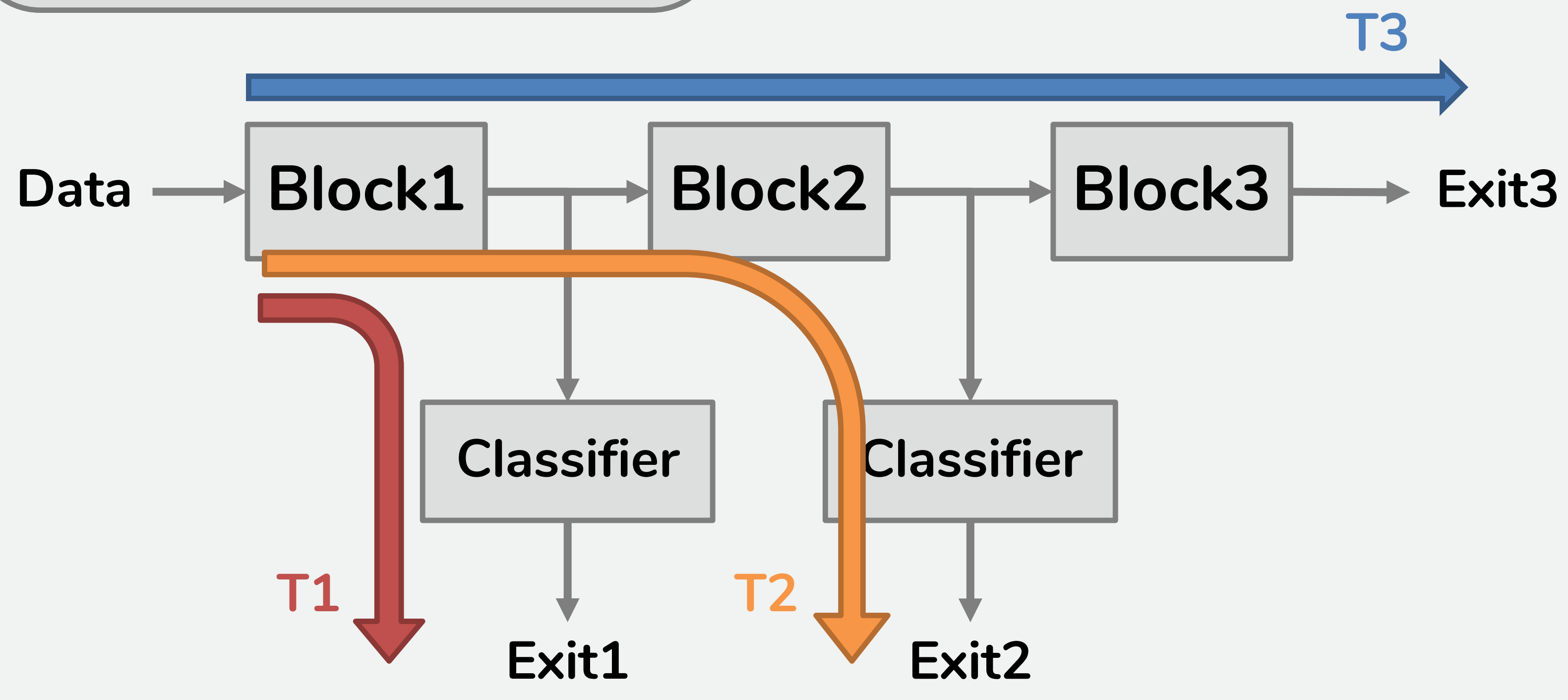
KAIST

NEURAL INFORMATION PROCESSING SYSTEMS

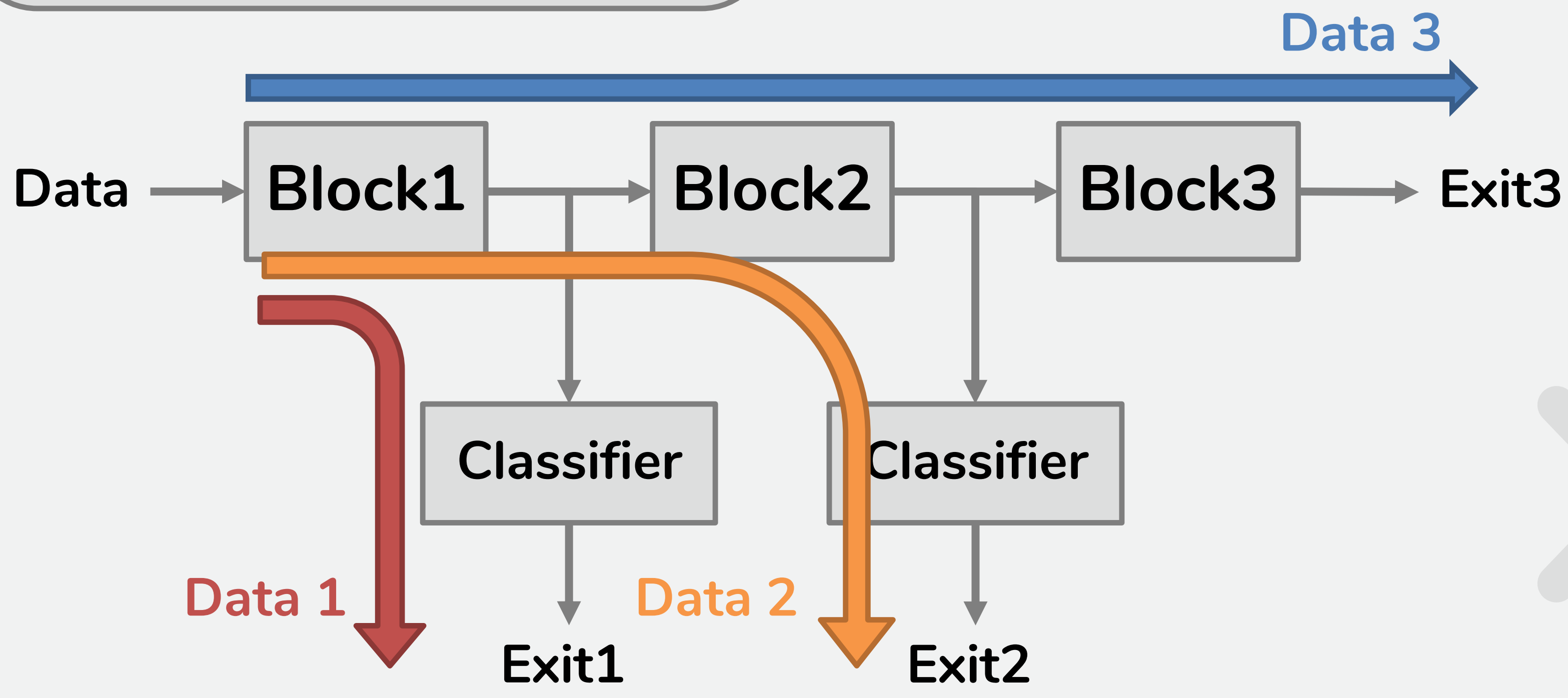
INTRODUCTION

Multi-exit Neural Network makes dynamic predictions in resource-constrained applications.

Anytime Prediction



Budgeted Prediction



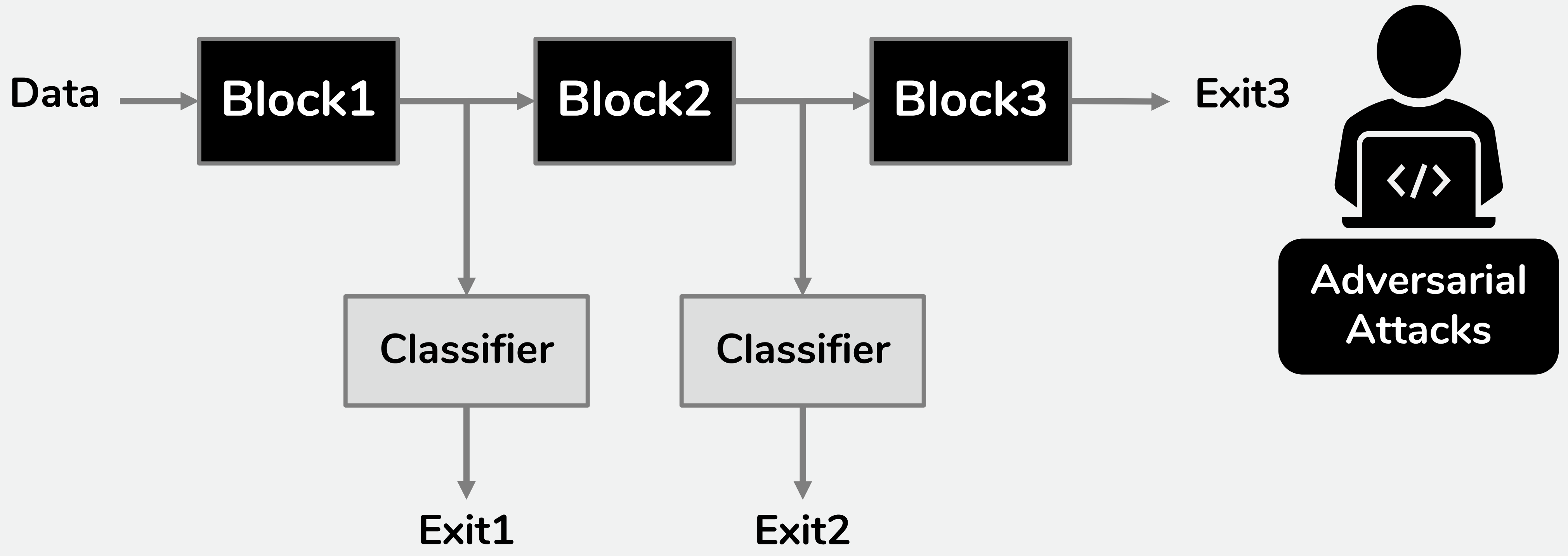
INTRODUCTION

Challenges in Robust Multi-exit Neural Network :

A multi-exit model is highly vulnerable to simple adverasrial attacks.

Cause: Different submodels (exits) in multi-exit neural network have high correlations by sharing parameters.

→ Strong adverversarial transferability across subdmodels.

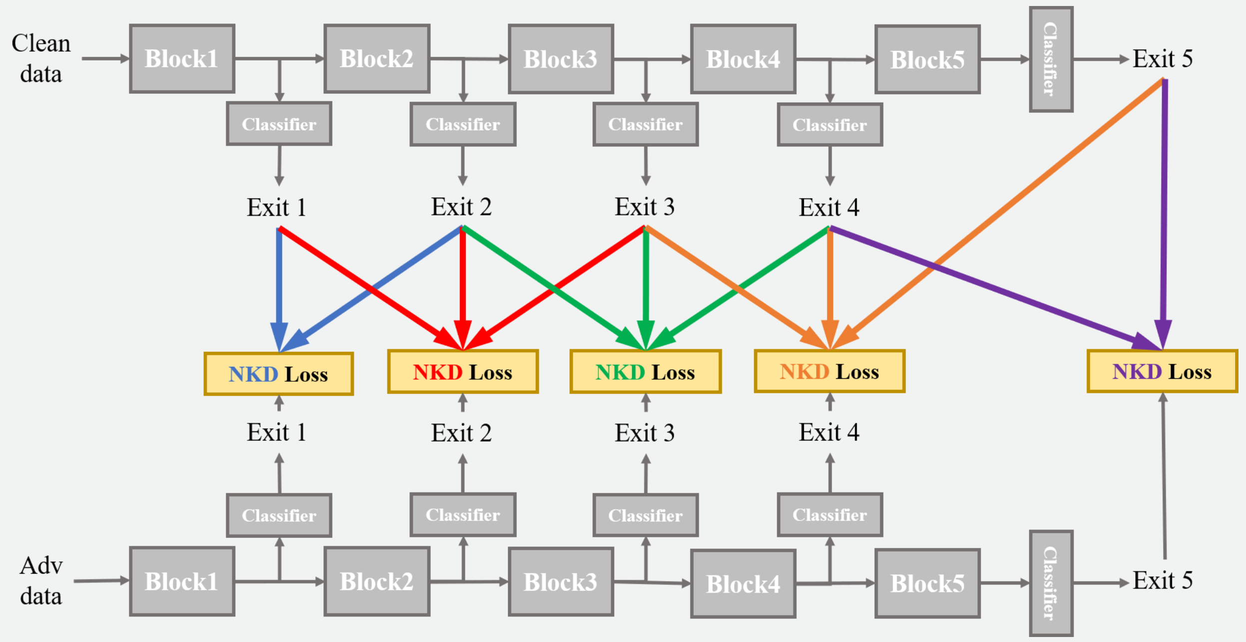


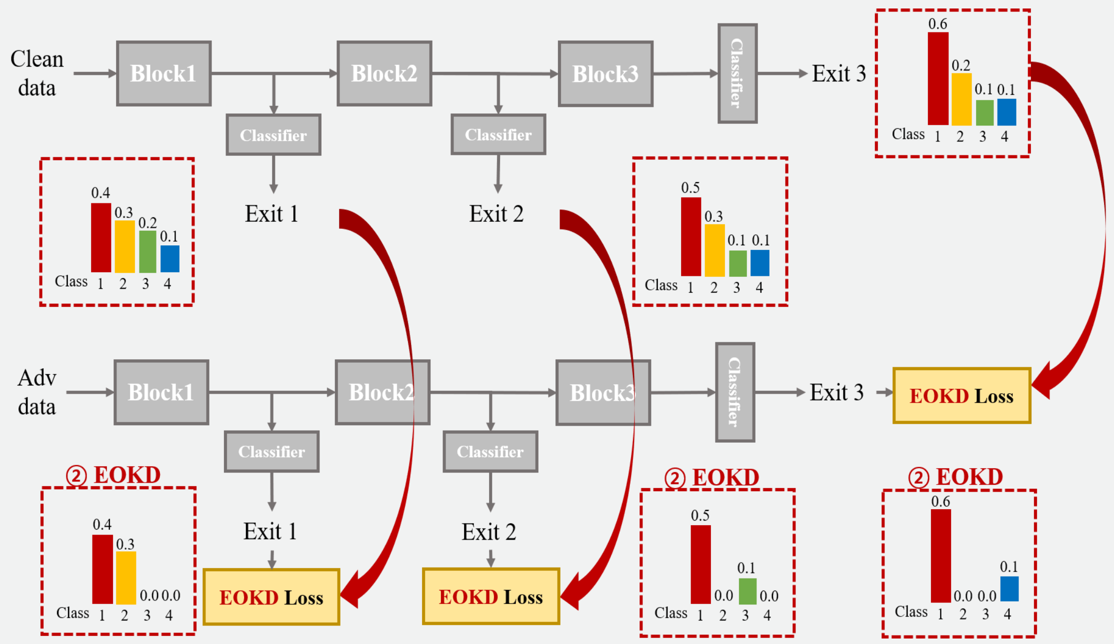
Contribution

We propose NEO-KD, a knowledge-distillation-based adversarial training strategy for robust multi-exit neural networks.

Component 1: Neighbor Knowledge Distillation (NKD)

Component 2: Exit-wise Orthogonal Knowledge Distillation (EOKD)





NEURAL INFORMATION PROCESSING SYSTEMS

KAIST

BACKGROUND

Adversarial Training for Multi-exit Neural Network

Triple Wins[1]: Adversarial training strategy by generating adversarial examples targeting a specific exit or multiple exits.

Single Attack

$$x_i^{adv} = \underset{x' \in |x'-x|_\infty \le \epsilon}{argmax} \left| \ell\left(f_{\theta_i}\left(x^{adv}\right), y\right) \right|$$

Average Attack

$$x_{avg}^{adv} = \underset{x' \in |x'-x|_\infty \le \epsilon}{argmax} \left| \frac{1}{L} \sum_{j=1}^{L} \ell\left(f_{\theta_j}\left(x^{adv}\right), y\right) \right|$$

Max-Average Attack

$$x_{max}^{adv} \leftarrow x_{i^*}^{adv},$$

$$where\ i^* = \underset{i}{argmax} \left| \frac{1}{L} \sum_{j=1}^{L} \ell\left(f_{\theta_j}\left(x_i^{adv}\right), y\right) \right|$$

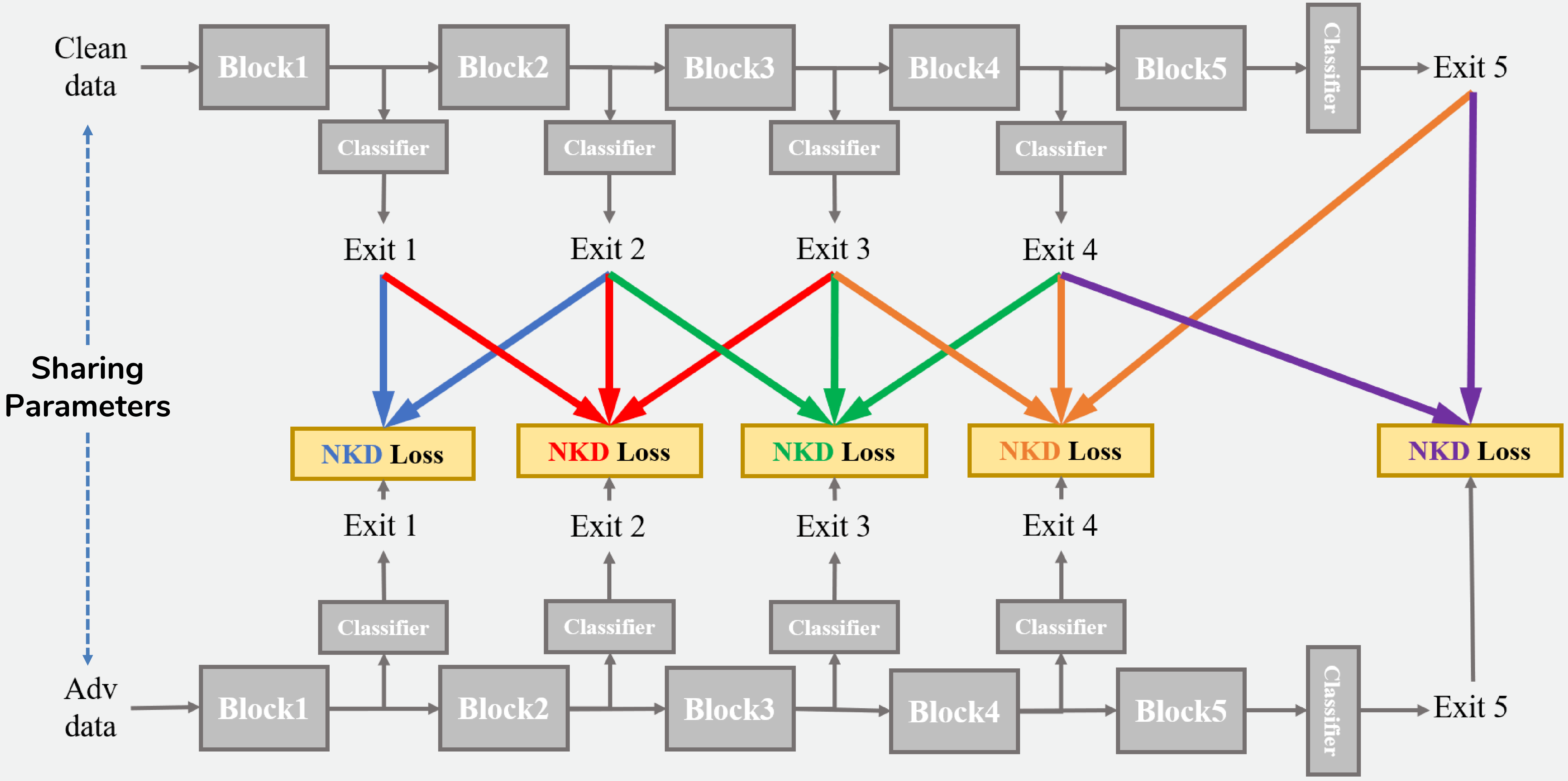
[1] Ting-Kuei Hu, Tianlong Chen, Haotao Wang, and Zhangyang Wang. Triple wins: Boosting accuracy, robustness and efficiency together by enabling input-adaptive inference. In International Conference on Learning Representations, 2020.

NEURAL INFORMATION PROCESSING SYSTEMS

KAIST

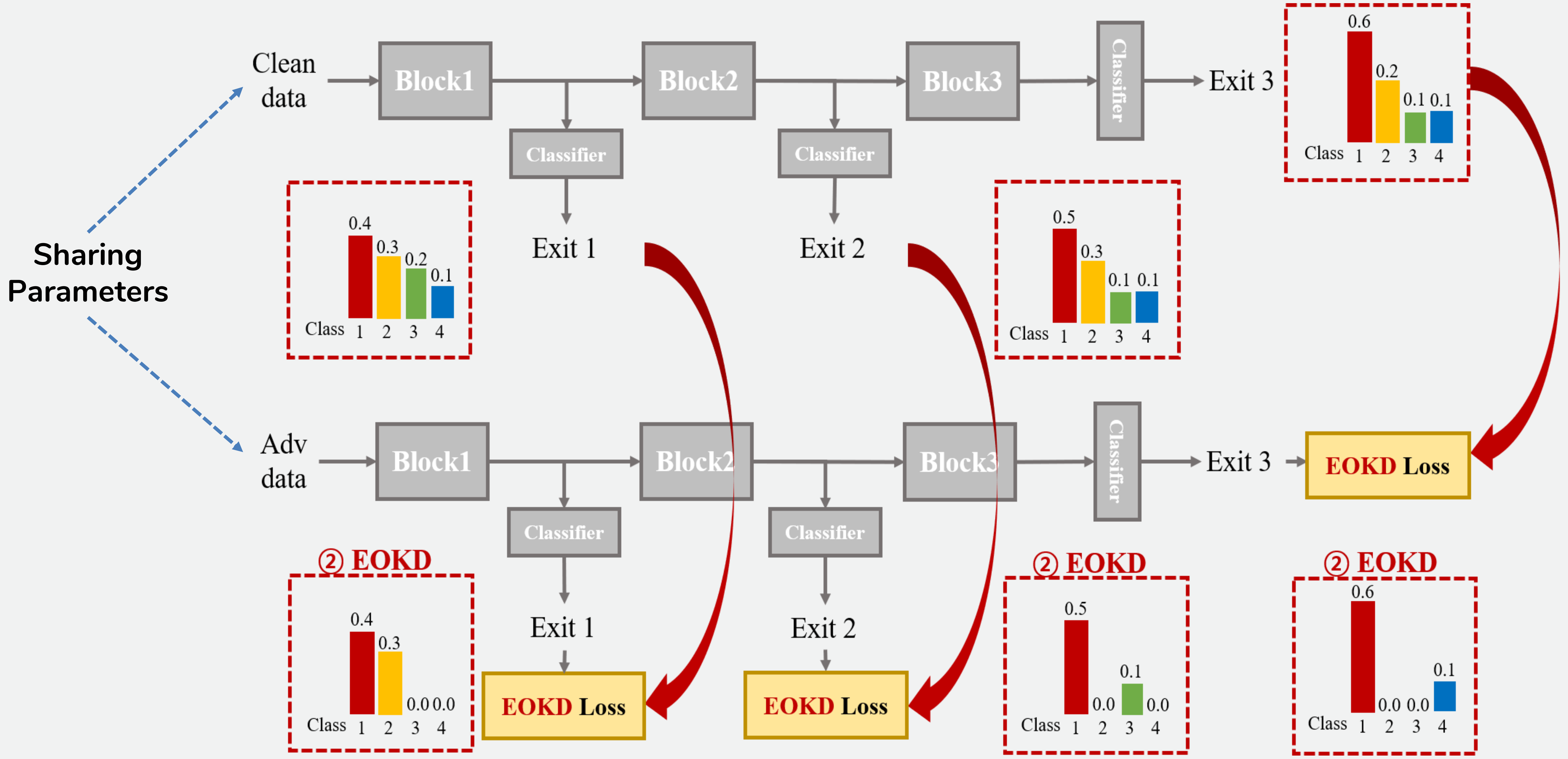
METHODOLOGY

Component 1: Neighbor Knowledge Distillation (NKD)

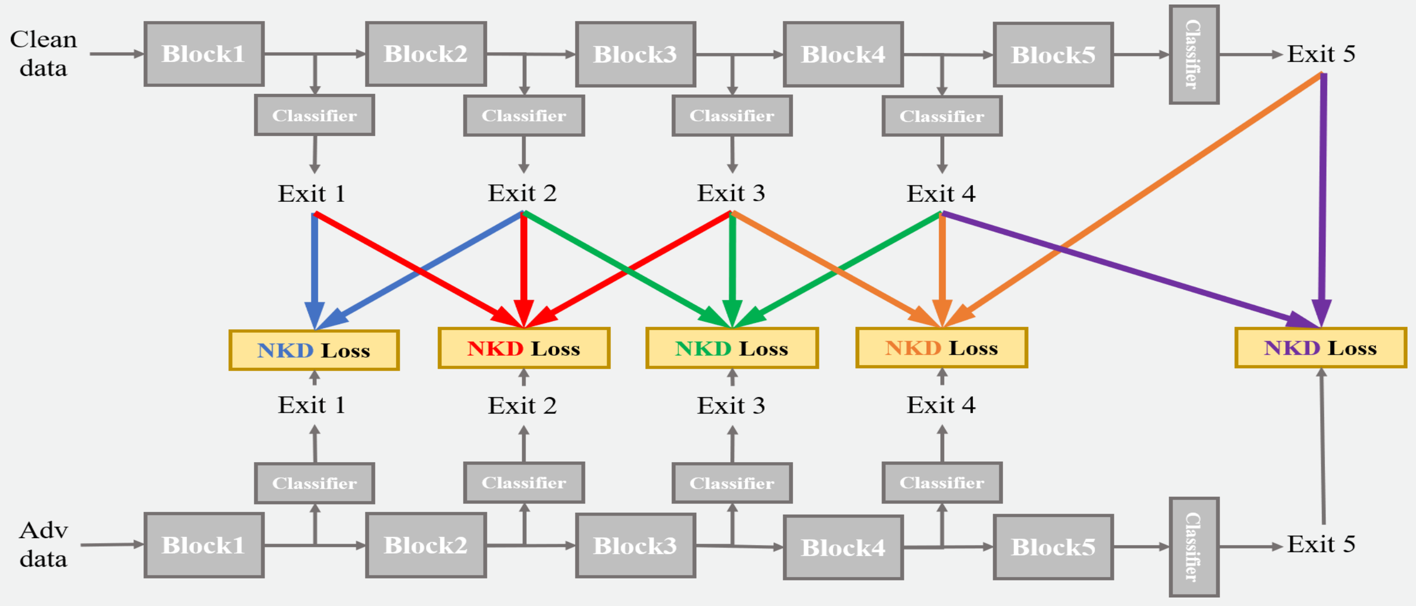


METHODOLOGY

Component 2: Exit-wise Orthogonal Knowledge Distillation (EOKD)



METHODOLOGY: NKD



$$NKD_{i,j} = \begin{cases} \ell\left(f_{\theta_i}(x_j^{adv}), \frac{1}{2}\sum_{k=1}^{2} f_{\theta_k}(x_j)\right), & i = 1 \\ \ell\left(f_{\theta_i}(x_j^{adv}), \frac{1}{2}\sum_{k=L-1}^{L} f_{\theta_k}(x_j)\right), & i = L \\ \ell\left(f_{\theta_i}(x_j^{adv}), \frac{1}{3}\sum_{k=i-1}^{i+1} f_{\theta_k}(x_j)\right), & otherwise \end{cases}$$

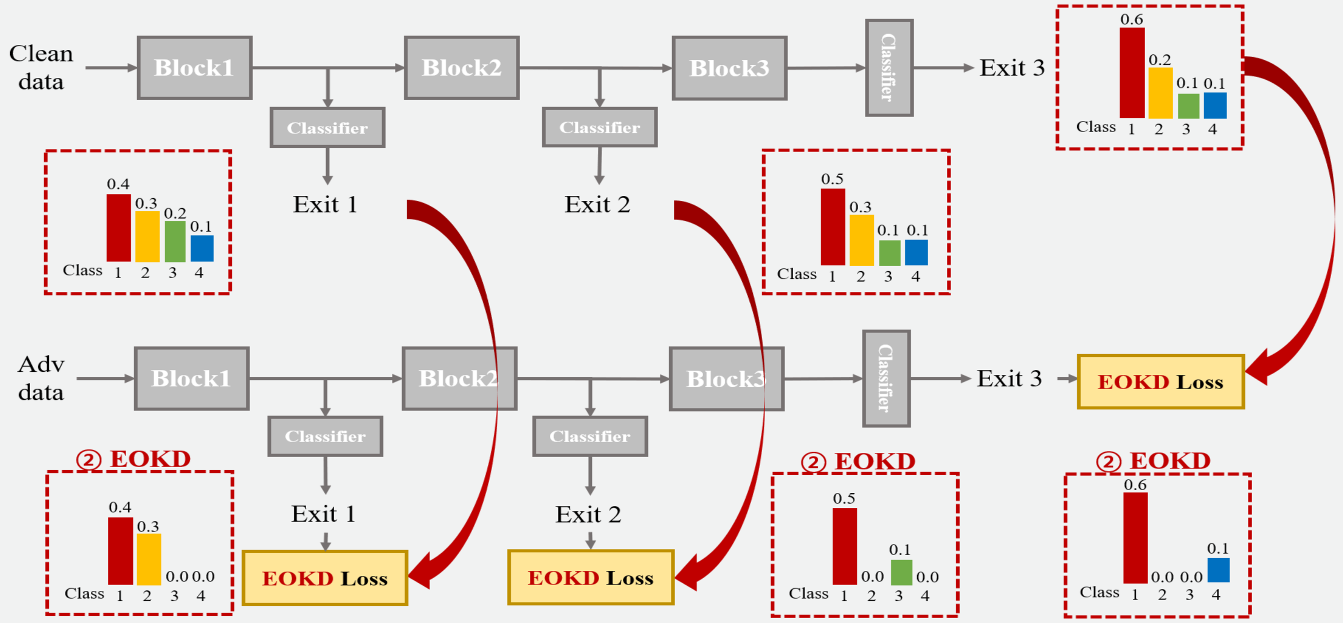
Role:

1. Generate a teacher prediction by ensembling neighbor predictions of clean data and distills it to each prediction of adversarial examples.
2. Guide the output feature of adversarial data at each exit to mimic the output feature of clean data.

Effect:

1. Provide a higher quality feature of original data to the corresponding exit.
2. Reduce adversarial transferability compared to the strategies that distill the same prediction to all exits.

METHODOLOGY: EOKD



$$EOKD_{i,j} = \ell\left(f_{\theta_i}(x_j^{adv}), O\left(f_{\theta_i}(x_j)\right)\right)$$

Role:

1. Provide orthogonal soft labels to each exit, in an exit-wise manner, reducing adversarial transferability.
2. Discard some predictions to encourage that the non-maximal predictions of individual exits become mutually orthogonal.

Effect:

1. EOKD reduces the dependency among different submodels (exits), reducing the adversarial transferability to the multi-exit network.

NEURAL INFORMATION PROCESSING SYSTEMS

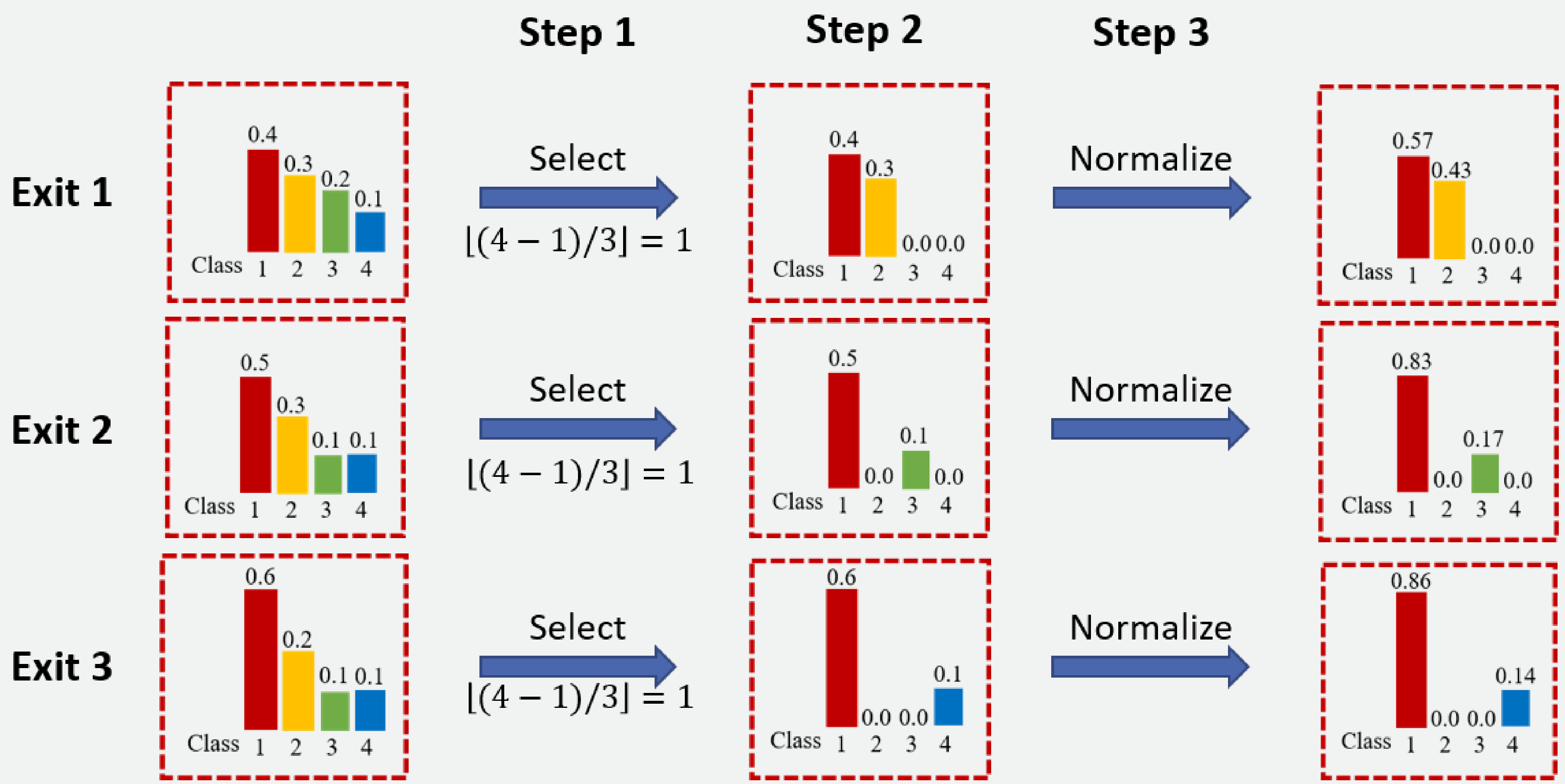
METHODOLOGY: EOKD

Orthogonal Labeling Operation O(·)

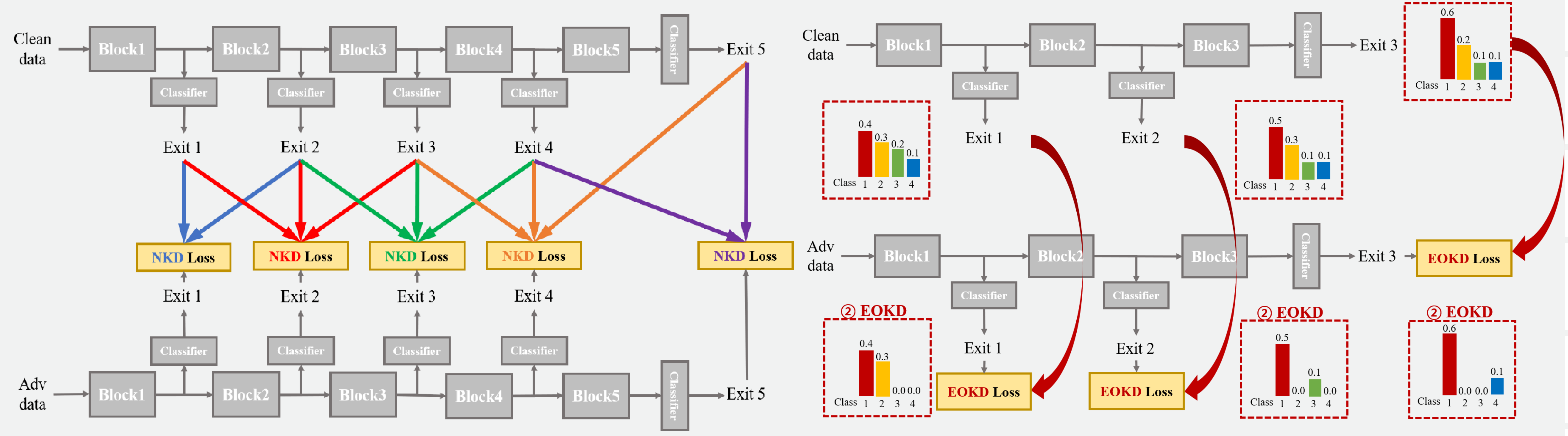
Step 1. Randomly select non-ground-truth $\lfloor (C-1)/L \rfloor$ classes among C classes for each exit.

Step 2. Unselected non-maximal labels becomes zero.

Step 3. Normalize the likelihood to be summed to 1.0.



METHODOLOGY: NEOKD



$$\mathcal{L}_{NEO-KD} = \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{L}\left[\ell\left(f_{\theta_i}(x_j), y_j\right) + \ell\left(f_{\theta_i}(x_j^{adv}), y_j\right) + \gamma_i(\alpha * NKD_{i,j} + \beta * EOKD_{i,j})\right]$$

NEURAL INFORMATION PROCESSING SYSTEMS

KAIST

Experiment Setup

Backbone	Dataset
Small CNN	MNIST
3-exit MSDNet	CIFAR10
7-exit MSDNet	CIFAR100
5-exit MSDNet	Tiny-ImageNet
5-exit MSDNet	ImageNet

Training
- PGD-based Max-average Attack
- PGD-based Average Attack

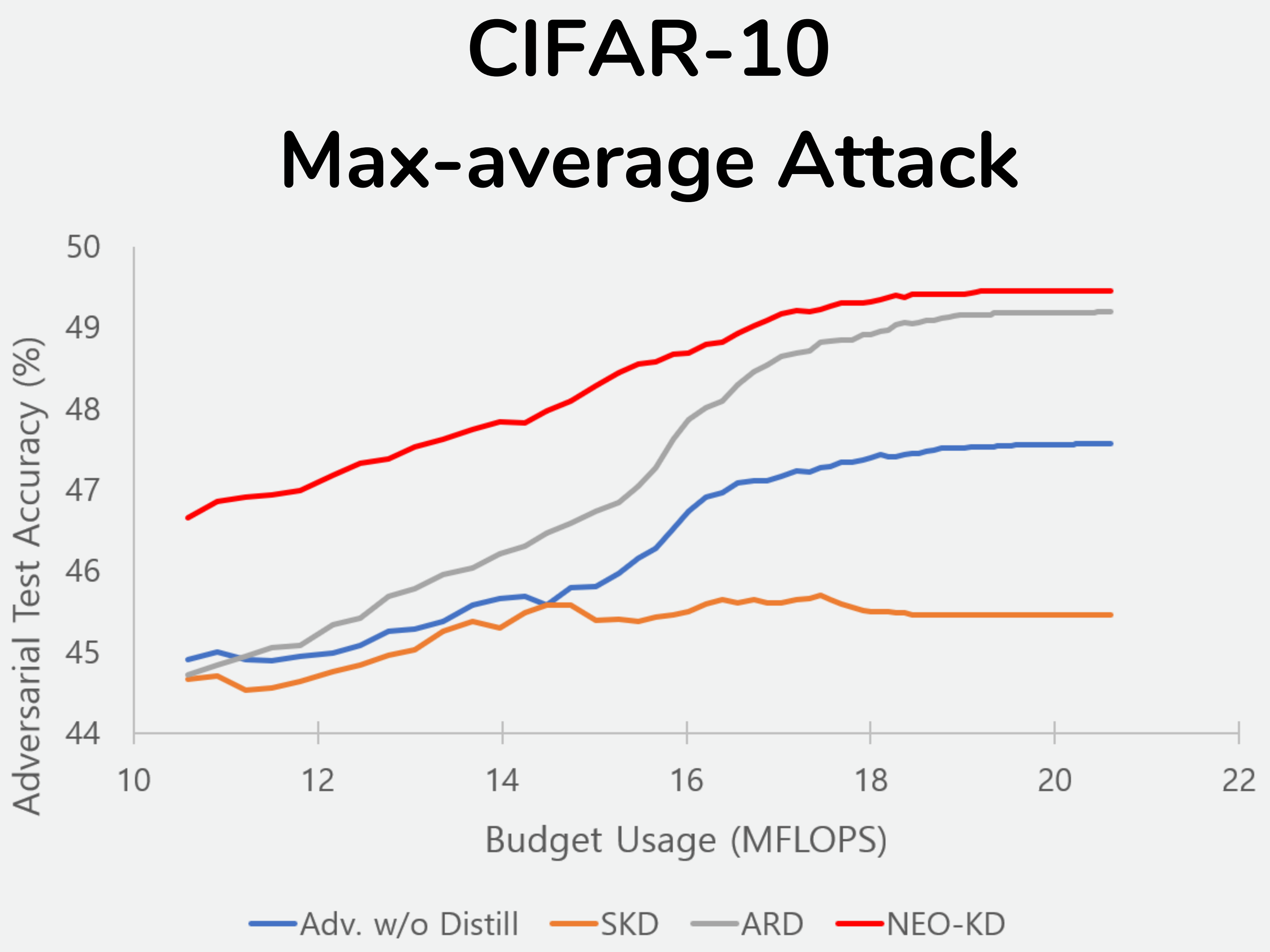
Test
- PGD-based Max-average Attack
- PGD-based Average Attack

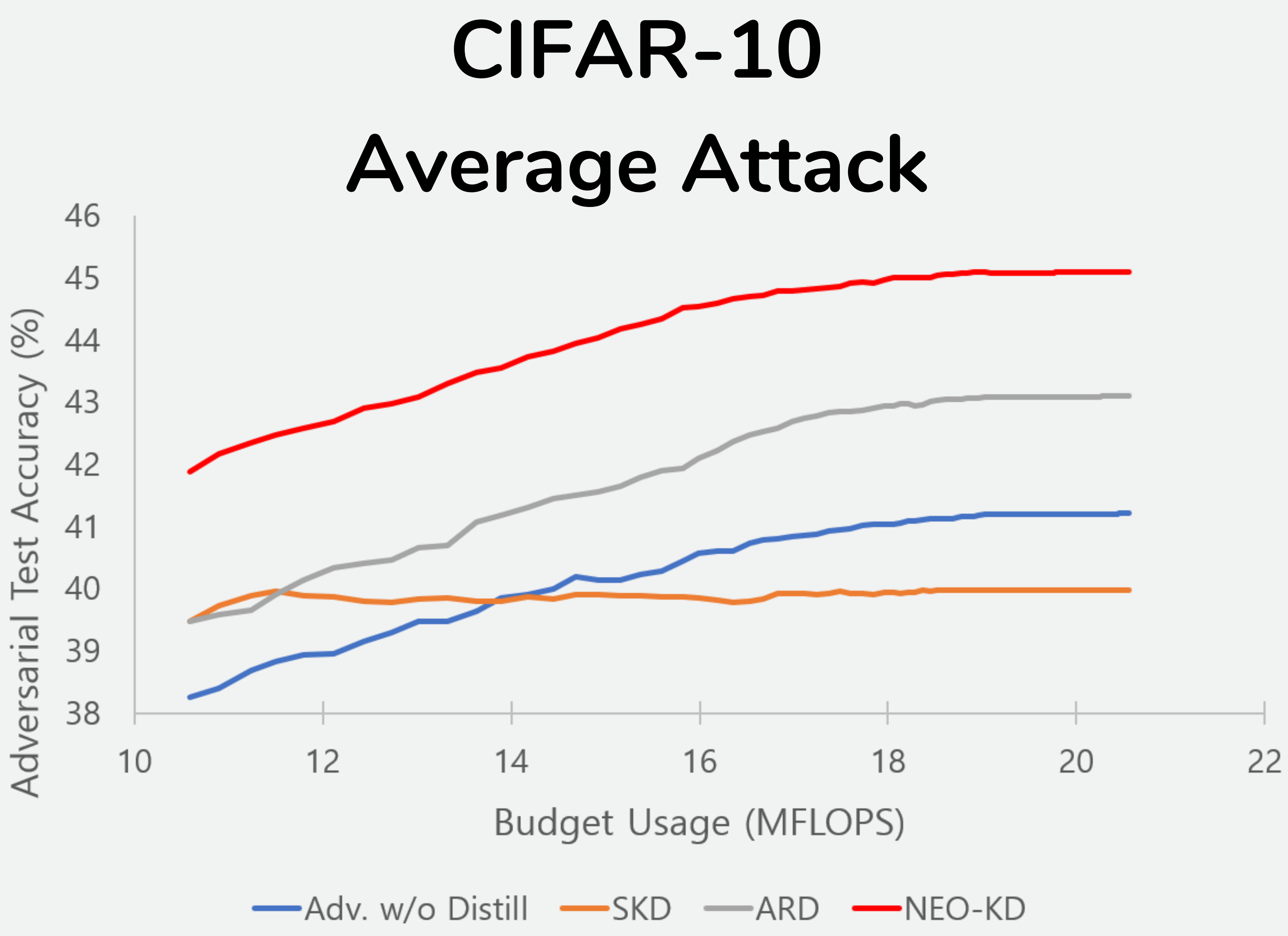
RESULT: Anytime Prediction Setup

Dataset	Average Adversarial Test Accuracy	Max-average Attack	Average Attack
MNIST	Adv. w/o Distill (ICLR 2020)	94.15%	92.75%
	SKD (ICCV 2019)	94.36% (+0.21%)	92.82% (+0.07%)
	ARD (AAAI 2020)	93.97% (-0.18%)	92.82% (+0.07%)
	LW (Neural Networks 2022)	92.12% (-2.03%)	91.95% (-0.56%)
	NEO-KD (ours)	94.49% (+0.34%)	93.50% (+0.75%)
CIFAR10	Adv. w/o Distill (ICLR 2020)	45.91%	40.04%
	SKD (ICCV 2019)	44.69% (-1.22%)	39.71% (-0.33%)
	ARD (AAAI 2020)	47.39% (+1.48%)	41.63% (+1.59%)
	LW (Neural Networks 2022)	35.87% (-10.04%)	30.62% (-9.42%)
	NEO-KD (ours)	48.30% (+2.39%)	44.20% (+4.16%)

Dataset	Average Adversarial Test Accuracy	Max-average Attack	Average Attack
CIFAR100	Adv. w/o Distill (ICLR 2020)	27.12%	18.13%
	SKD (ICCV 2019)	24.26% (-2.86%)	18.06% (-0.07%)
	ARD (AAAI 2020)	27.95% (+0.83%)	18.73% (+0.60%)
	LW (Neural Networks 2022)	19.91% (-7.21%)	14.42% (-3.71%)
	NEO-KD (ours)	28.96% (+1.84%)	22.88% (+4.75%)
ImageNet	Adv. w/o Distill (ICLR 2020)	30.45%	25.58%
	SKD (ICCV 2019)	28.43% (-2.02%)	23.57% (-2.01%)
	ARD (AAAI 2020)	30.89% (+0.44%)	24.71% (-0.87%)
	NEO-KD (ours)	32.37% (+1.92%)	29.98% (+4.40%)

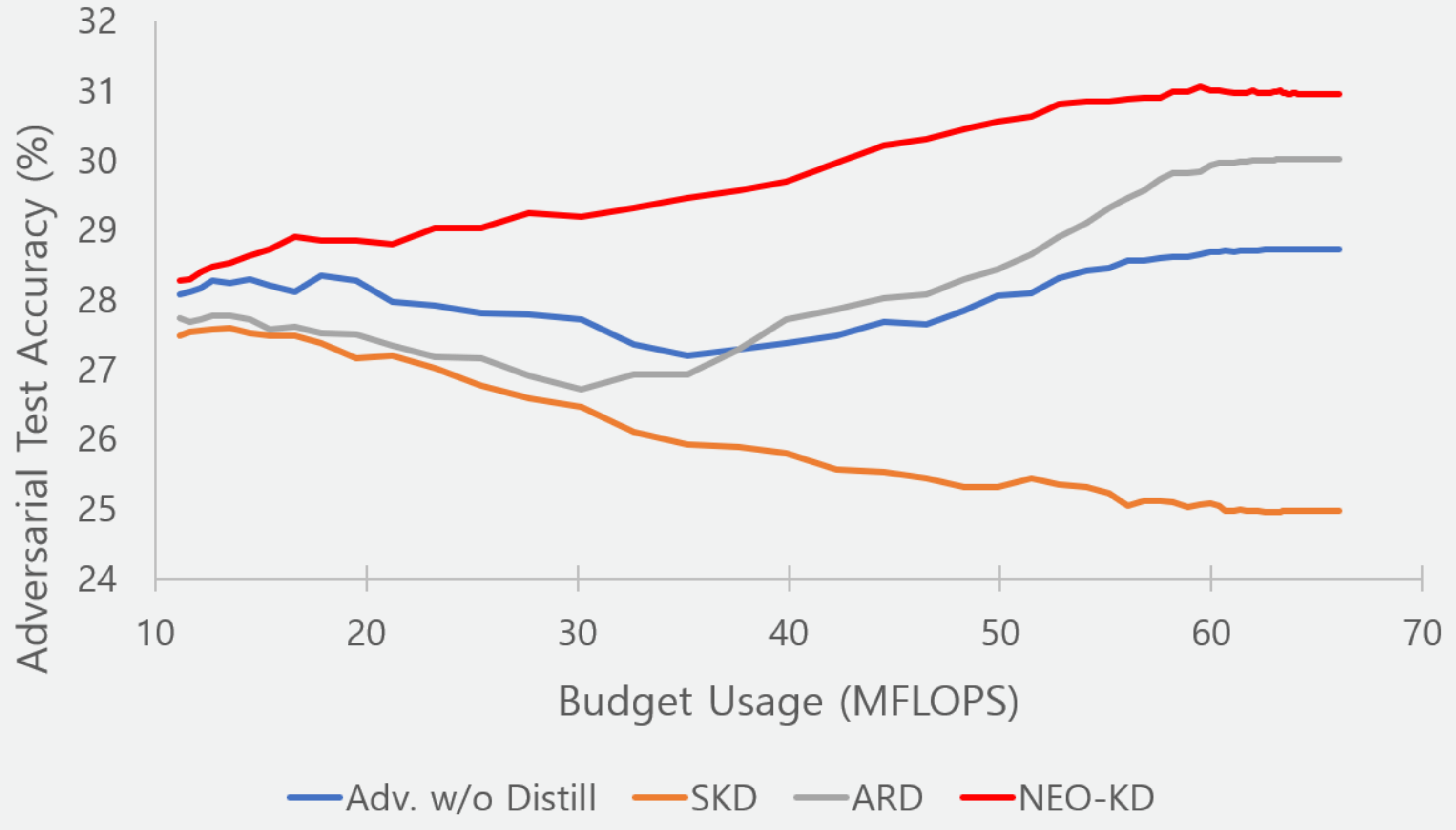
RESULT: Budgeted Prediction Setup



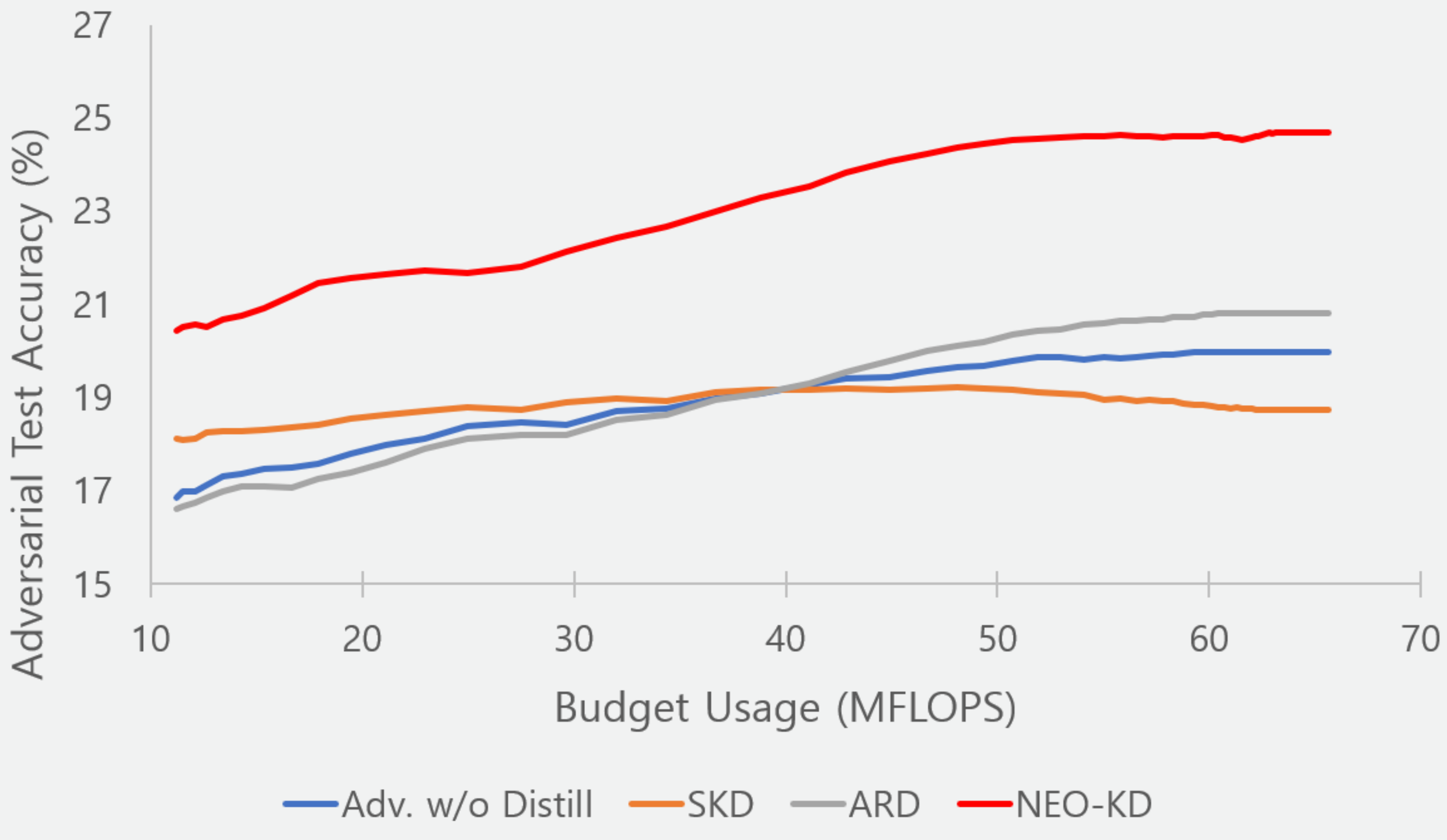


RESULT: Budgeted Prediction Setup

CIFAR-100
Max-average Attack

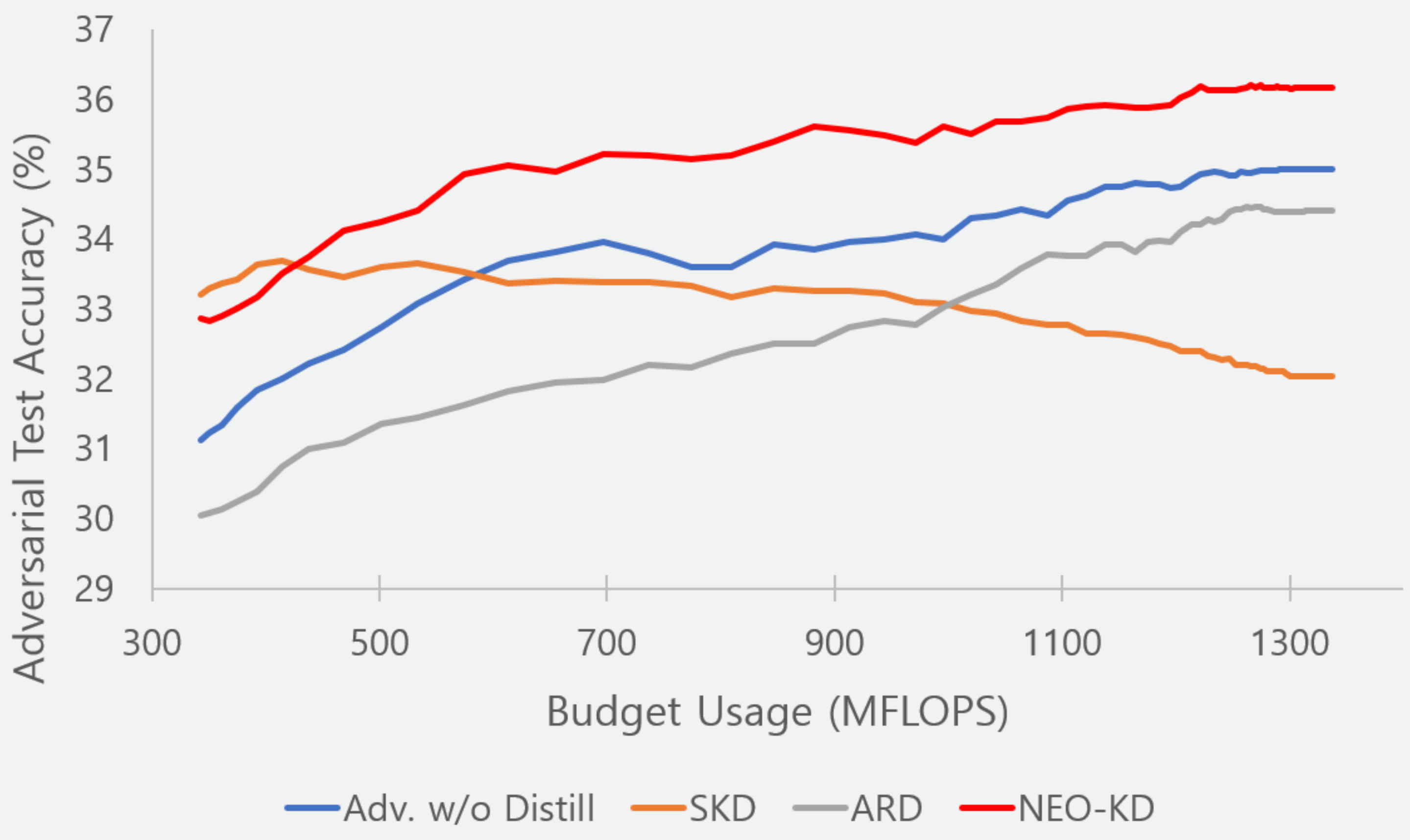


CIFAR-100
Average Attack

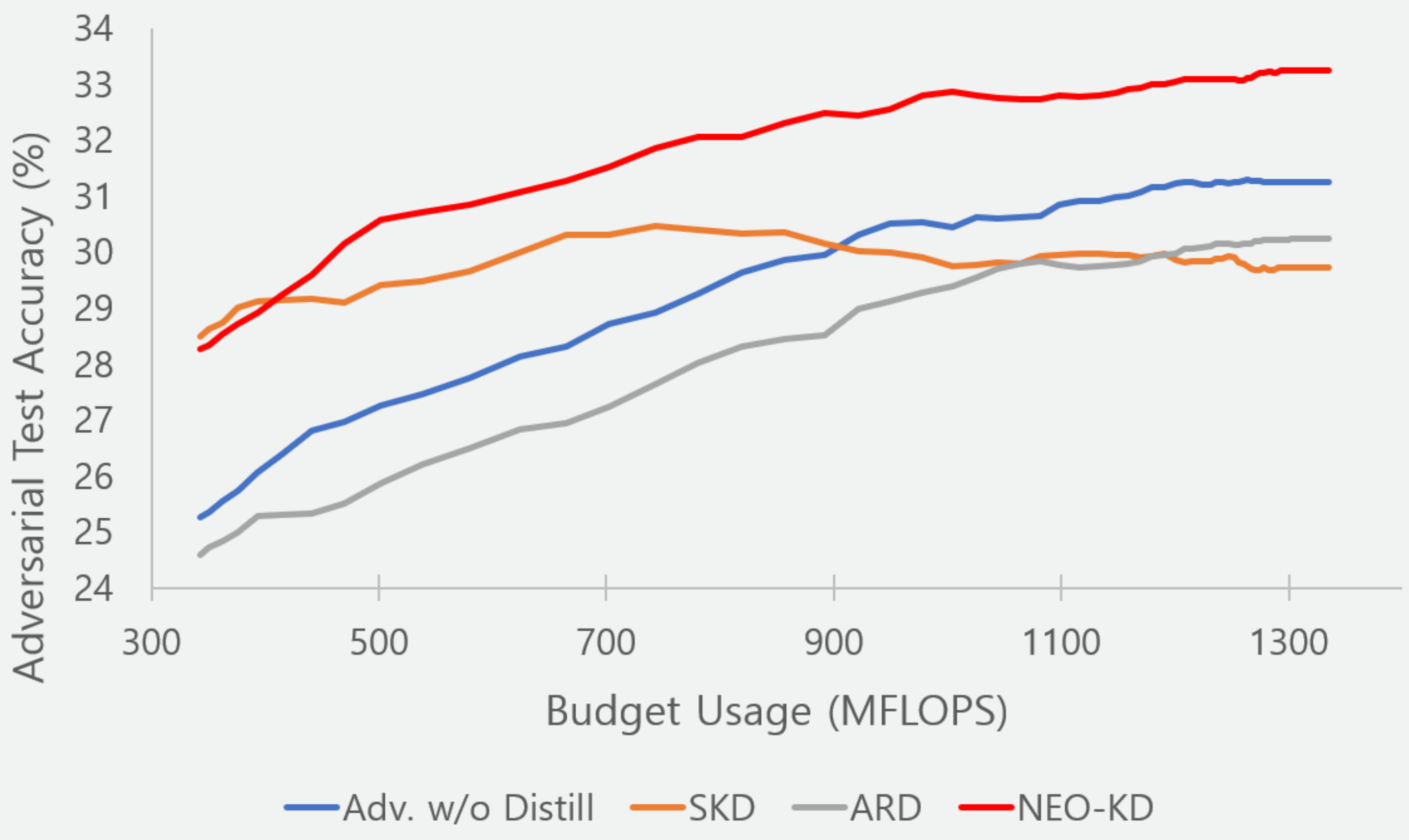


RESULT: Budgeted Prediction Setup

Tiny-ImageNet
Max-average Attack



Tiny-ImageNet
Average Attack



RESULT: Adversarial Transferability

Adversarial Transferability: the attack success rate of adversarial single attack.

	Exit 1	Exit 2	Exit 3	Exit 4	Exit 5	Exit 6	Exit 7
Exit 1	81.55	29.40	21.23	17.77	14.96	12.59	12.89
Exit 2	33.73	79.43	25.10	21.53	18.28	15.15	14.88
Exit 3	30.33	31.04	73.87	30.52	26.29	21.01	21.36
Exit 4	27.30	27.52	32.40	73.32	30.35	24.63	23.38
Exit 5	23.27	23.43	26.81	29.48	75.40	28.26	28.15
Exit 6	19.86	18.37	19.81	22.53	27.71	76.84	41.66
Exit 7	16.38	15.01	15.89	17.63	21.01	35.59	80.03

[Adv. w/o Distill]

Avg. w/o Diag: 23.68%

	Exit 1	Exit 2	Exit 3	Exit 4	Exit 5	Exit 6	Exit 7
Exit 1	79.38	38.22	29.08	24.39	20.94	19.78	19.52
Exit 2	43.55	76.42	36.37	30.56	26.36	24.11	24.15
Exit 3	38.49	41.15	71.50	41.78	36.73	32.08	31.40
Exit 4	37.24	38.46	44.49	69.32	44.56	37.77	36.85
Exit 5	33.21	34.21	38.87	43.72	68.91	43.36	41.63
Exit 6	24.51	25.64	28.16	31.84	36.40	75.41	63.31
Exit 7	18.92	18.49	20.36	22.50	25.35	52.70	81.18

[SKD]

Avg. w/o Diag: 33.36%

	Exit 1	Exit 2	Exit 3	Exit 4	Exit 5	Exit 6	Exit 7
Exit 1	66.68	31.06	20.19	15.37	11.61	9.41	10.12
Exit 2	35.34	62.46	24.30	17.85	13.82	11.61	12.41
Exit 3	26.95	27.45	58.90	25.24	18.46	15.14	15.48
Exit 4	25.66	24.44	28.11	54.76	24.58	19.72	19.48
Exit 5	21.02	19.83	21.27	24.14	57.13	23.37	22.32
Exit 6	17.66	16.83	17.27	18.98	23.01	59.20	35.67
Exit 7	13.57	12.83	12.80	13.63	16.61	30.43	66.37

[NEO-KD]

Avg. w/o Diag: 20.12%

CONCLUSION

Multi-exit neural network makes flexible predictions in resource-constraint environments.
However, multi-exit network is challenging to be robust because of high correlation across different exits from sharing parameters.

We propose a knowledge-distillation-based adversarial training strategy for robust multi-exit networks, NEO-KD.
→Correctly guiding the predictions of clean/adversarial data at each exit.
→Reduce the adversarial transferability in the multi-exit neural network.

THANK YOU

NEURAL INFORMATION PROCESSING SYSTEMS

KAIST