Dynamic Neural Response Tuning

Tian Qiu, Wenxiang Xu, Lin Chen, Linyun Zhou, Zunlei Feng, Mingli Song

https://github.com/horrible-dong/DNRT

Highlights

- This paper explores the application of biological neural response patterns in transmitting and aggregating information within various artificial neural network (ANN) architectures.
- A novel mechanism called Dynamic Neural Response Tuning (DNRT) is proposed in this study to align the response patterns of ANNs with those of biological neurons. DNRT consists of Response-Adaptive Activation (RAA) and Aggregated Response Regularization (ARR), mimicking the biological neuron's information transmission and aggregation behaviors.
- In terms of information transmission, RAA dynamically adjusts the response condition based on the characteristics and strength of the input signal. In terms of information aggregation, ARR is devised to enhance the network's ability to learn category specificity by imposing constraints on the network's response distribution.
- Extensive experimental studies demonstrate that the proposed DNRT is highly interpretable and applicable to various mainstream ANN architectures including MLPs, ViTs and CNNs. Additionally, the proposed DNRT can achieve remarkable performance compared with existing neural response mechanisms across multiple tasks and domains.

Motivation

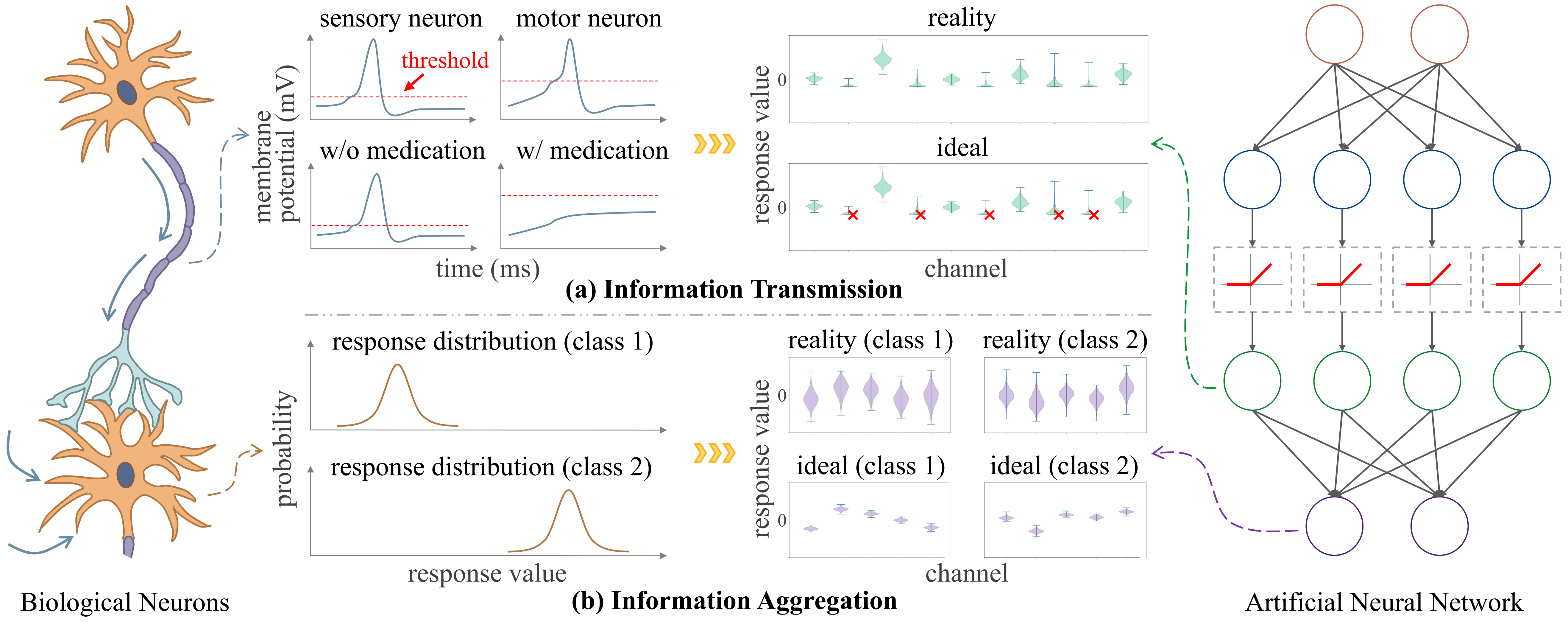


Figure 1: Observations on biological neurons and the way they have inspired the proposed Dynamic Neural Response Tuning (DNRT).

Observation (a): Information Transmission Response. The information transmission in biological neurons is often achieved by triggering action potentials that propagate through axons. ANNs utilize activation mechanisms to simulate such biological behavior. However, previous studies on ANNs' activations have only considered static/fixed response conditions, while the biological neuron's response conditions (i.e., the threshold membrane potential) are typically dynamic, depending on multiple factors such as neuronal properties and the real-time environment. Therefore, the dynamic response conditions of biological neurons could provide insights to improve the static ones of existing activations in ANNs.

Observation (b): Information Aggregation Response. The information aggregation in biological neurons is often performed by integrating impulses from other neurons. The biological aggregated response exhibits high specificity for different categories, allowing the nervous system to differentiate and identify objects. Similarly, ANNs aggregate the activated signals from upstream neurons. However, ANNs' aggregated responses tend to exhibit high variances, which indicates the response is pretty noisy and overlaps between categories, finally interfering with the network's decision. The class-specific aggregated responses of biological neurons inspire us to devise techniques to enhance the aggregated ones of ANNs.

Method

(a) Response-Adaptive Activation (RAA)

Define an activation A that operates under static response conditions, for example, the GELU:

$$A(x) = x \cdot \Phi(x). \tag{1}$$

Here, the activation probability $\Phi(x) = P(X \leq x)$ is the CDF of $X \sim \mathcal{N}(0,1)$, which rises as the input x gets larger. Since the curve of $\Phi(x)$ is static/fixed, any input to A will consistently follow the same response condition, potentially leading to the phenomenon where some irrelevant signals may incorrectly trigger an activation response and negatively impact the network's final decision. To address this issue, a learnable mapping f is introduced. It allows the original activation to assess the relevance of the input and adjust the response condition accordingly. The RAA operates in the feature vector level for efficiency:

$$\begin{aligned} RAA(\mathbf{x}) &= \mathbf{x} \cdot \Phi(\mathbf{x} + f(\mathbf{x})) \\ &= \mathbf{x} \cdot \Phi(\mathbf{x} + (\mathbf{w}^{\mathrm{T}}\mathbf{x} + b)), \end{aligned} \tag{2}$$

where $\mathbf{x}$ is an input d-dimensional feature vector. The d-dimensional mapping vector $\mathbf{w} = [w_1, w_2, ..., w_d]^{\mathrm{T}}$ and scalar bias b are learnable parameters that are initialized with zero to ensure an initial offset of zero. *Furthermore, this behavior is not limited to a specific context and can also be extended to other static activation forms such as ReLU etc.*

(b) Aggregated Response Regularization (ARR)

K is the number of categories of the dataset. For input data of each category $k \in \{1, 2, ..., K\}$, the network's historical mean/average aggregated responses $\boldsymbol{\mu}_k$ are tracked in real time:

$$\boldsymbol{\mu}_k^t = (1 - m) \cdot \boldsymbol{\mu}_k^{t-1} + m \cdot \mathbf{x}_k^t, \tag{3}$$

where the vector $\mathbf{x}_k^t = [x_1, x_2, ..., x_d]^{\mathrm{T}}$ is the network's aggregated response at time t when inputting a sample from category k, and d is the vector's dimension; *for ViTs,* $\mathbf{x}$ *is the class token, and for CNNs,* $\mathbf{x}$ *is the globally pooled feature map.* m is the momentum for updating the moving mean. Next, the network's aggregated response can be made more concentrated by utilizing the historical mean. At each time t, a loss constraint $\mathcal{L}_{arr}$ is applied:

$$\mathcal{L}_{arr} = \|\mathbf{x}_k^t - \boldsymbol{\mu}_k^{t-1}\|_1 \,/\, d. \tag{4}$$

The final loss $\mathcal{L}$ can be expressed as follows:

$$\mathcal{L} = \mathcal{L}_{task} + \lambda \cdot \frac{1}{J} \textstyle\sum_{j=1}^{J} \mathcal{L}_{arr}^{j}, \tag{5}$$

where $\mathcal{L}_{task}$ is the primary loss for the specific task; for example, in the context of a standard classification task, $\mathcal{L}_{task}$ represents the cross-entropy loss. J is the number of layers in the network that have ARR applied, and λ is the balanced parameter.

Experiments

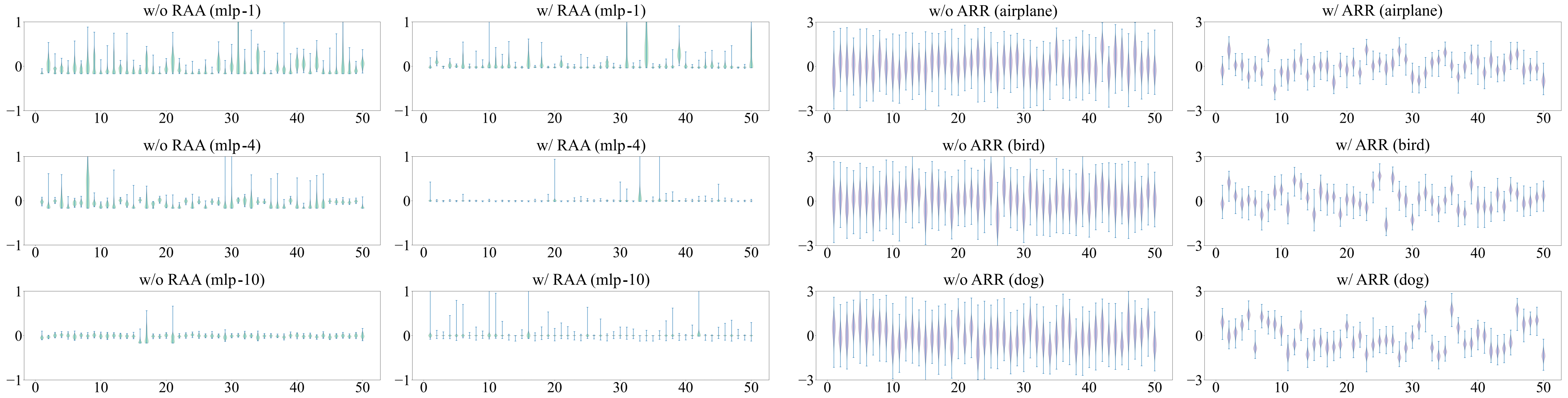
Table 1: The proposed DNRT on Vision Transformer (ViT) and its variants.

Top-1 Acc / %		Softplus	ELU	SELU	SiLU	ReLU	GELU	GDN	DNRT
CIFAR-10	ViT-Tiny	84.3	82.0	79.4	85.5	89.9	89.2	81.8	**92.1**
	DeiT-Tiny	84.7	81.4	79.9	86.6	89.6	89.2	83.0	**92.4**
	CaiT-XXS	82.5	80.7	78.4	86.6	89.4	88.7	80.0	**91.6**
	PVT-Tiny	90.6	89.3	85.4	92.5	93.0	92.8	82.8	**94.3**
	TNT-Small	88.3	85.4	83.7	90.5	90.9	91.2	85.1	**92.5**
CIFAR-100	ViT-Tiny	62.4	60.0	57.5	65.5	65.7	65.4	59.4	**71.4**
	DeiT-Tiny	63.4	60.0	58.3	67.1	67.0	67.0	59.8	**71.4**
	CaiT-XXS	60.4	59.3	55.8	63.9	65.8	65.5	56.2	**70.6**
	PVT-Tiny	69.5	69.3	65.7	70.2	70.9	70.6	64.4	**71.6**
	TNT-Small	65.2	63.8	60.9	65.1	65.4	64.4	62.5	**71.9**
ImageNet-100	ViT-Tiny	74.1	68.9	66.4	74.1	75.4	76.4	67.9	**80.9**
	DeiT-Tiny	75.3	69.4	67.0	75.1	75.6	74.6	66.3	**81.1**
	CaiT-XXS	70.9	69.1	65.9	76.1	76.0	76.7	69.5	**80.4**
	PVT-Tiny	79.5	77.1	76.1	79.5	81.9	81.4	75.8	**84.1**
	TNT-Small	78.9	79.3	76.4	77.6	79.9	77.2	76.9	**82.3**

Table 2: The proposed DNRT on various CNN architectures.

Top-1 Acc / %		Softplus	ELU	SELU	SiLU	ReLU	GELU	GDN	DNRT
CIFAR-10	AlexNet	76.1	84.3	82.6	84.3	86.0	85.2	83.8	**86.4**
	VGG-11	89.6	90.4	89.1	91.7	92.2	92.0	89.3	**92.6**
	ResNet-18	94.0	93.9	93.0	95.2	95.0	95.2	82.3	**95.5**
	MobileNet	**90.0**	89.8	87.7	89.7	87.4	89.4	86.6	**90.0**
	ShuffleNet	90.2	90.2	87.8	90.8	89.4	90.9	87.6	**91.2**
CIFAR-100	AlexNet	44.0	57.6	55.7	57.3	57.2	57.4	56.5	**59.2**
	VGG-11	64.7	68.8	66.0	70.8	70.2	70.7	70.2	**71.0**
	ResNet-18	75.6	75.5	74.7	75.7	75.7	75.8	71.7	**76.5**
	MobileNet	66.3	**67.5**	64.1	66.0	66.0	66.2	55.3	66.9
	ShuffleNet	68.0	68.2	63.8	68.0	66.3	68.2	57.3	**68.7**
ImageNet-100	AlexNet	74.8	77.5	75.7	77.8	76.3	77.7	74.4	**78.8**
	VGG-11	80.1	83.6	80.6	87.0	87.7	87.7	85.3	**88.7**
	ResNet-18	85.4	84.9	83.9	85.7	84.9	85.9	80.2	**86.8**
	MobileNet	80.5	80.7	77.5	80.9	80.6	81.0	73.6	**81.7**
	ShuffleNet	83.0	82.3	79.4	82.9	81.6	82.6	75.3	**83.4**

Note: "GELU" etc. denotes the network with the GELU etc. activation and the original response aggregation mechanism. "DNRT" denotes the network with the proposed RAA and ARR.



(a) Visual results of RAA

(b) Visual results of ARR

Figure 2: Response distributions on the class token in ViT's MLP block with and without the proposed DNRT on CIFAR-10. (a) MLP's activation responses in ViT layer-1, 4, and 10. (b) MLP's aggregated responses to airplane, bird, and dog images. The baseline is the network with the GELU activation (the default used in ViT) and the original response aggregation mechanism. The horizontal coordinate represents the element's (channel's) index within the feature vector, the vertical coordinate represents the response value, and the area depicts the response density. The figure displays the response distributions of the first 50 elements (channels) in each vector (the class token).

Table 3: The proposed DNRT on Vision Transformer (ViT) and its variants. (on ImageNet-1K)

Top-1 Acc / %		ViT-Tiny	DeiT-Tiny	CaiT-XXS	PVT-Tiny	TNT-Small
ImageNet-1K	ReLU	70.9	73.2	74.0	73.7	73.4
	GELU	70.4	73.0	73.6	73.5	73.3
	DNRT	**73.0**	**73.5**	**75.3**	**76.1**	**75.6**

Table 4: Computational complexity (FLOPs / G) & Inference speed (Latency / ms) of the networks with the proposed DNRT compared to those of the original networks.

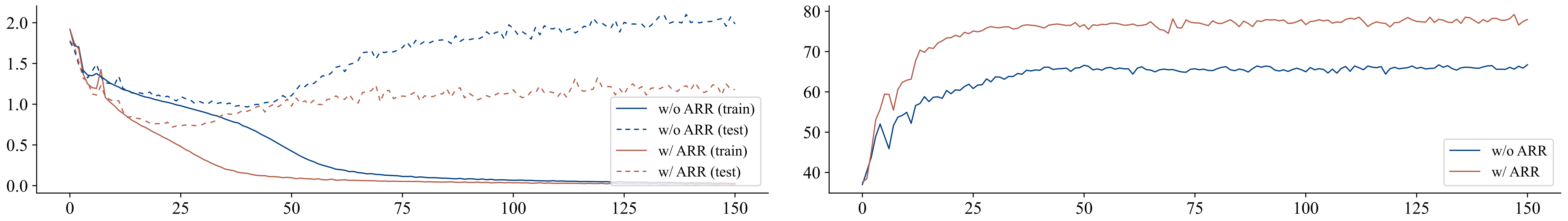
Network / Metric	ReLU (FLOPs)	ReLU (Latency)	DNRT (FLOPs)	DNRT (Latency)
ViT-Tiny	1.078 G	4.5 ms	1.080 G	4.6 ms
TNT-Small	4.849 G	10.1 ms	4.856 G	10.5 ms

Table 5: Node classification on DGraph & Tackling highly imbalanced data on Long-Tailed CIFAR-10.

Node classification			Tackling highly imbalanced data (β=10, 50, 100)		
AUC / %	GCN	GraphSAGE	Top-1 Acc / %	ViT-Tiny	ResNet-32
ReLU	72.5	75.1	ReLU	53.0, 50.0, 44.9	86.8, 83.5, 80.0
GELU	71.9	74.4	GELU	52.6, 49.3, 44.4	86.5, 83.3, 79.6
DNRT	**74.9**	**76.7**	DNRT	**56.5**, **53.6**, **47.3**	**87.7**, **84.1**, **82.0**

Table 6: Ablation study of the proposed DNRT on CIFAR-100.

Top-1 Acc / %	ReLU	GELU	RAA	ReLU + ARR	GELU + ARR	DNRT (RAA + ARR)
ViT-Tiny	65.7	65.4	66.5	69.6	68.5	**71.4**
ResNet-18	75.7	75.8	76.1	76.3	76.2	**76.5**



(a) Loss CE over epochs

(b) Top-1 Acc / % over epochs

Figure 3: The classification performance of ViT-Tiny with and without the proposed ARR on CIFAR-10 using weak data augmentations comprising only "random horizontal flipping" and "normalization". (a) Training & Testing curves of Loss CE over epochs. (b) Testing curves of Top-1 Acc / % over epochs. The model *without* ARR experiences severe overfitting, while the one *with* ARR overcomes it and yields massive performance improvements.

ICLR 2024 Poster Session 7

Fri, 10 May, 2024