Discovering Influential Neuron Path in Vision Transformers

上海科技大学 ShanghaiTech University

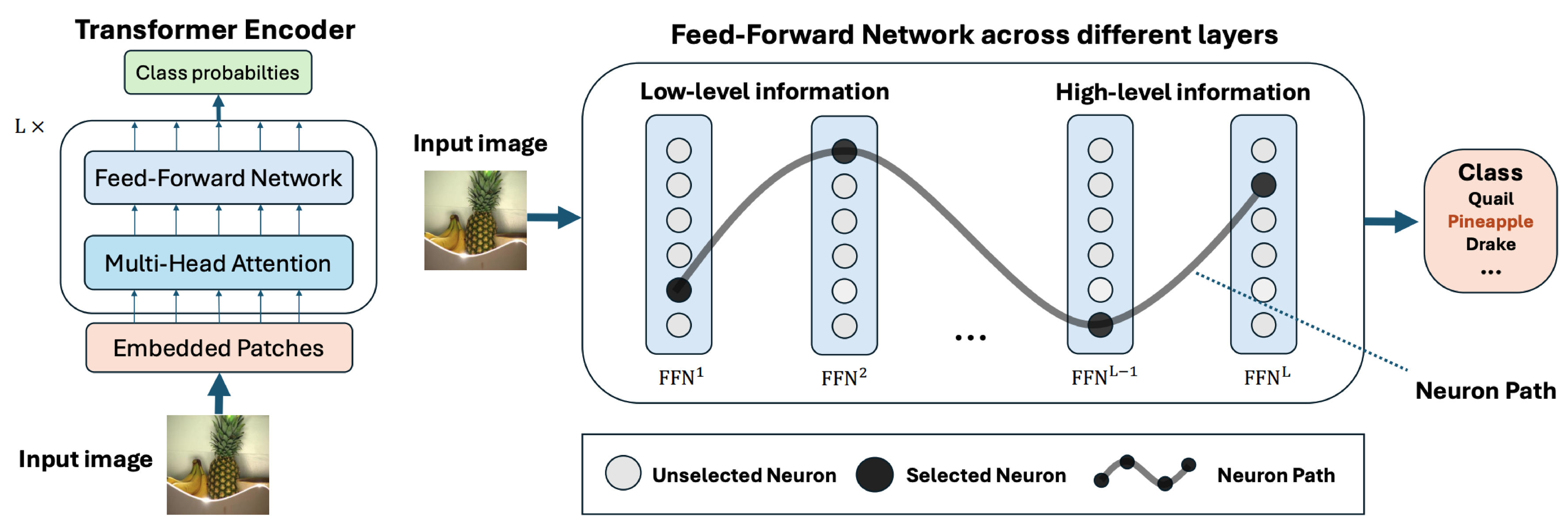
ICLR

Kan Ren

Yifan Wang, Yifei Liu, Yingdiong Shi, Changming Li, Anqi Pang, Sibei Yang, Jingyi Yu, Kan Ren
{wangyf7, renkan}@shanghaitech.edu.cn

Question: ***How is the vision Transformer model processing the input information by layer, and which part of the model is significant to derive the final outcome?***

A New Lens: The Neuron Path



Definition 1 (Joint Attribution Score) *Given a model $F : \mathbb{R}^d \to \mathbb{R}$ containing L layers, whose output with input x is defined as F_x, with a set of neuron $\{w^1_{i_1}, w^2_{i_2}, ..., w^N_{i_N}\}, N \le L$, a Joint Attribution Score is defined as*

$$\mathrm{JAS}(w^1_{i_1}, w^2_{i_2}, ..., w^N_{i_N}) = \sum_{n=1}^{N} \overline{w}^n_{i_n} \int_{\alpha=0}^{1} \sum_{l=1}^{N} \frac{\partial F_x(\alpha \overline{w}^1_{i_1}, \alpha \overline{w}^2_{i_2}, ..., \alpha \overline{w}^N_{i_N})}{\partial w^l_{i_l}} d\alpha \,. \tag{2}$$

For the convenience of computation, we use the Riemann approximation to estimate the continuous integral as follows,

$$\widetilde{\mathrm{JAS}}(w^1_{i_1}, w^2_{i_2}, ..., w^N_{i_N}) = \frac{1}{m} \sum_{j=1}^{N} \overline{w}^j_{i_j} \sum_{k=1}^{m} \sum_{l=1}^{N} \frac{\partial F_x(\frac{k}{m}\overline{w}^1_{i_1}, \frac{k}{m}\overline{w}^2_{i_2}, ..., \frac{k}{m}\overline{w}^N_{i_N})}{\partial w^l_{i_l}} \,, \tag{3}$$

where m is the sampling step.

Definition 2 (Neuron Path) *Given a model $F : \mathbb{R}^d \to \mathbb{R}$ containing L layers, with an input x, and a user-defined criterion $\mathbf{S}(\cdot)$, a neuron path $\mathcal{P}_x$ is defined as follow.*

$$\mathcal{P}_x = \{w^1, w^2, ..., w^L\} \tag{4}$$

that maximizes the $\mathbf{S}(\mathcal{P}_x)$, where $w^l, l \in \{1, 2, ..., L\}$ stands for the selected neuron within layer l.

Algorithm 1 Layer-progressive Neuron Locating Algorithm

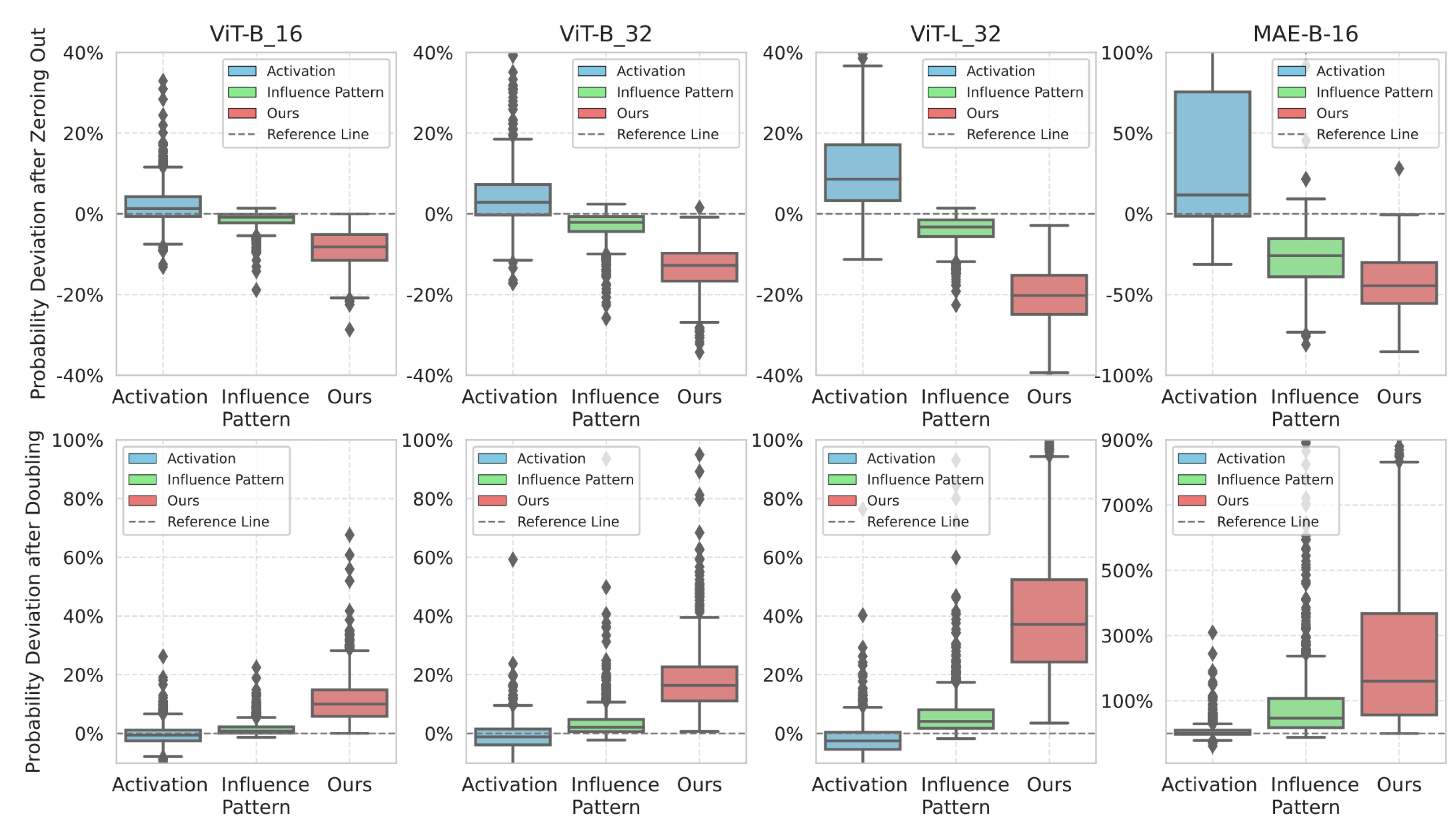
Input: Model F with L layers, input sample x
Output: neuron path $\mathcal{P}$
Initialization: $\mathcal{P} = \varnothing, l = 1$
while $l \le L$ **do**
  $\mathcal{W}$ is the set of neurons in layer l of F; Score $= 0, p =$ None
  for $w \in \mathcal{W}$ **do**
    if Score $< \widetilde{\mathrm{JAS}}(\mathcal{P}, w)$ **then**
      Score $= \widetilde{\mathrm{JAS}}(\mathcal{P}, w); p = w$
  $\mathcal{P} = \mathcal{P} \cup \{p\}; l = l + 1$

Quantitative Comparison:

Metrics	Methods	Target Models			
		ViT-B-16	ViT-B-32	ViT-L-32	MAE-B-16
Joint Attribution Score ↑	Activation	-0.0034	0.0288	-0.0065	0.0013
	Influence Pattern	0.0412	0.0841	0.1227	0.0030
	Neuron Path (ours)	**0.4078**	**0.6610**	**1.0086**	**0.0095**
Removal Accuracy Deviation ↓	Activation	0.07%	-0.15%	0.16%	-2.80%
	Influence Pattern	-0.50%	-1.24%	-1.41%	-15.67%
	Neuron Path (ours)	**-2.40%**	**-3.81%**	**-5.28%**	**-26.50%**
Enhancement Accuracy Deviation ↑	Activation	-0.33%	-0.45%	-0.86%	-1.00%
	Influence Pattern	0.46%	0.83%	1.12%	4.15%
	Neuron Path (ours)	**2.04%**	**3.06%**	**5.02%**	**7.28%**

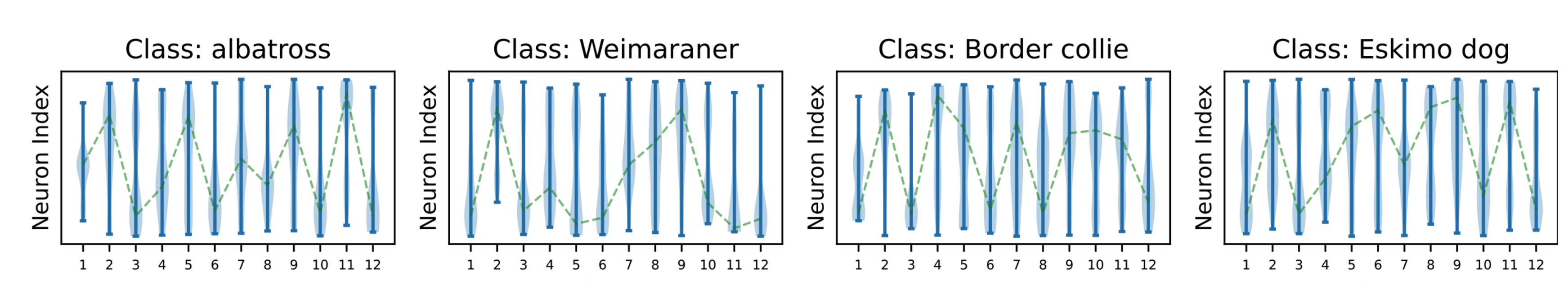
Finding 1: The Neuron Path method more effectively identifies the influential neurons within the model.

Quantitative Comparison:



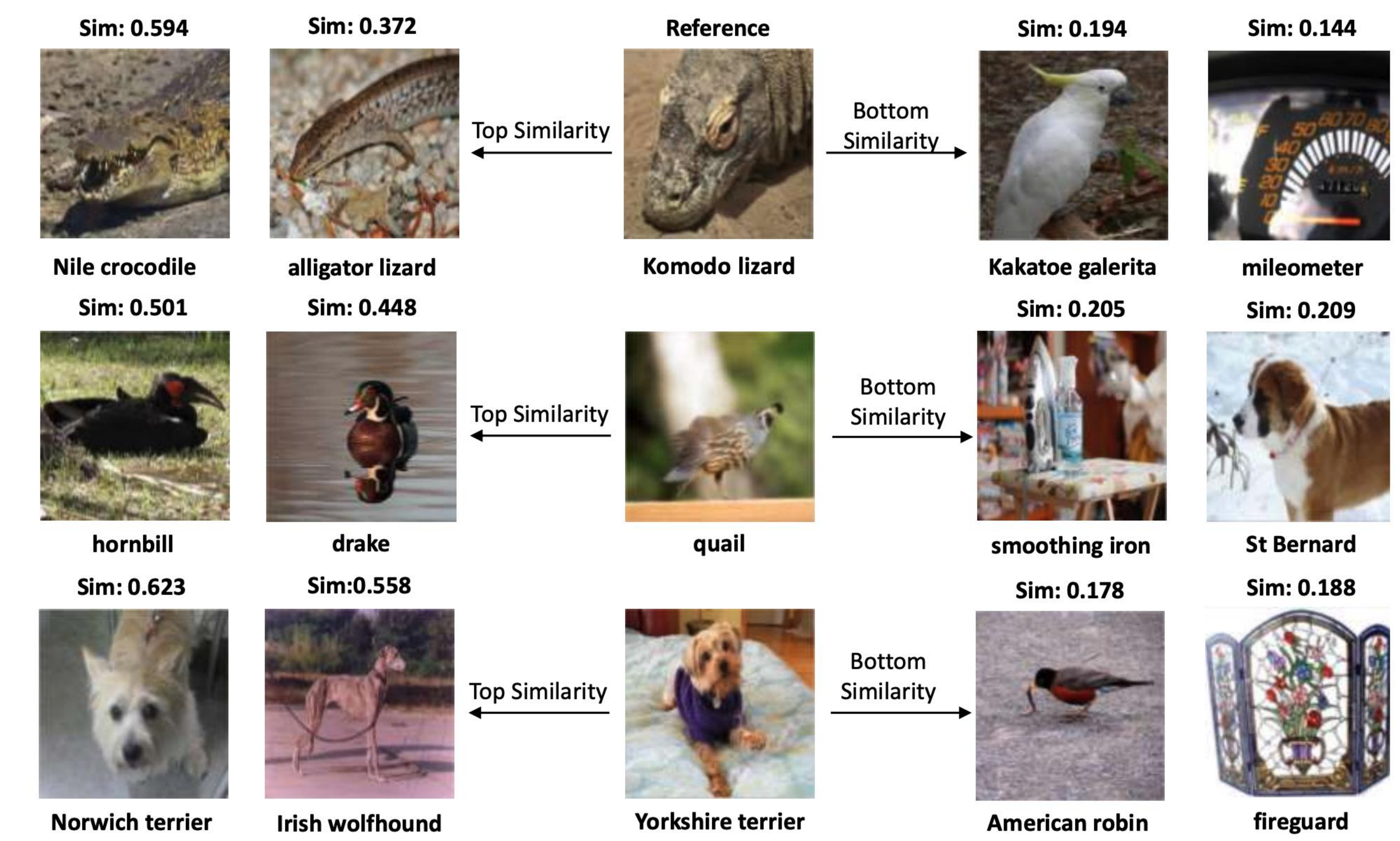
Finding 2: The discovered neuron paths play a vital role in model inference.

Intra-class analysis on neuron path:



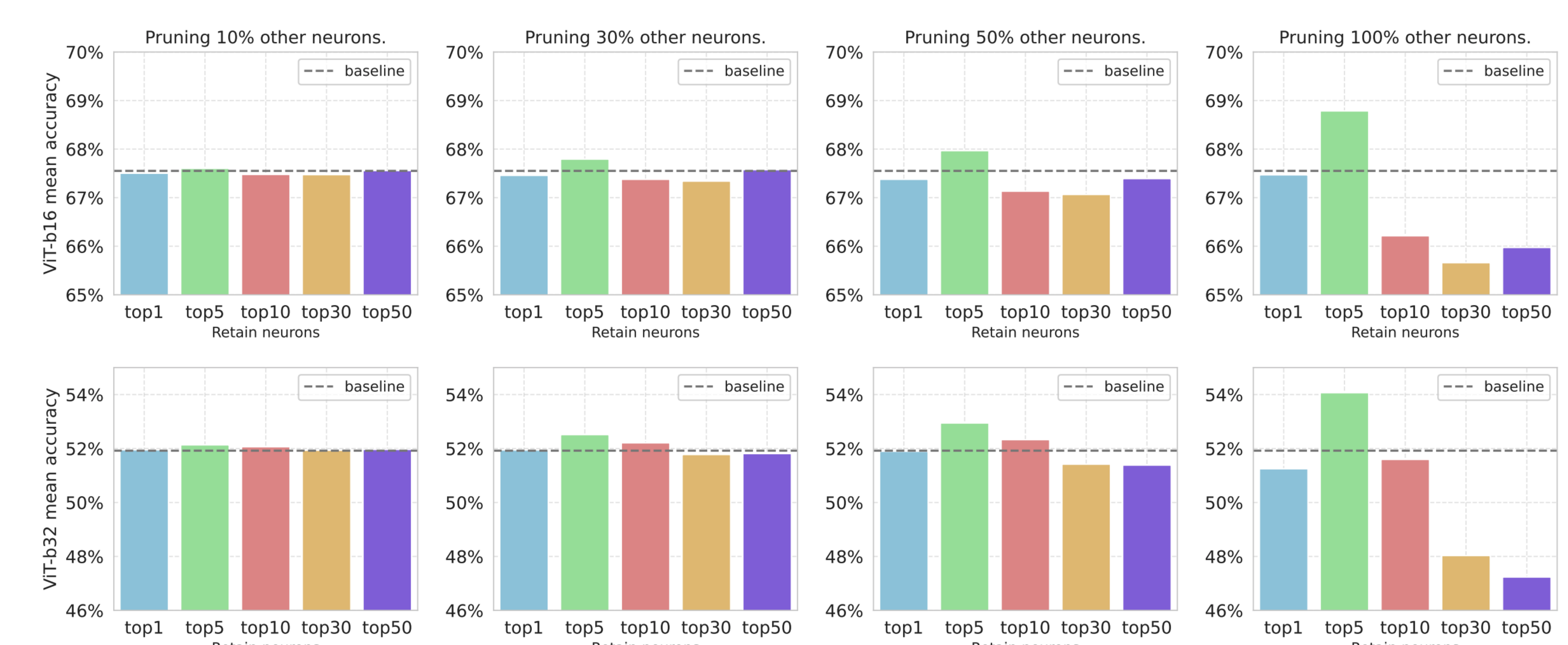
Finding 3: Some certain neurons contribute more at each layer to specific classes.

Inter-class analysis on neuron path:



Finding 4: Neuron paths reveals semantic similarity.

Multi-neuron model pruning:



Finding 5: Neurons within Vision Transformer models are largely redundant, with only a sparse subset significantly impacting model performance.

Future works:

1. Neuron Path in more modules.
2. Neuron Path in more tasks.
3. More applications with Neuron Path.