Supplementary Material for LINe: Out-of-Distribution Detection by Leveraging Important Neurons

Yong Hyun Ahn[1], Gyeong-Moon Park[2,*], Seong Tae Kim[2,*]
[1]Department of Artificial Intelligence, Kyung Hee University
[2]Department of Computer Science and Engineering, Kyung Hee University

A. Detailed CIFAR Benchmark Results

Table S.1 and Table S.2 are detailed results of CIFAR-10 and CIFAR-100 benchmark experiments (detailed results for Table 1 in the main text). For both tables, all the results except DICE + ReAct and LINe are taken from Sun et al. [11]. We choose hyperparameters for DICE + ReAct as sparsity p = 90 and ReAct threshold = 1.0, as in [10, 11].

B. LINe on Other Models

In this section, we show LINe also works well with other models. In the main text, we show LINe with pre-trained DenseNet [4] and ResNet-50 [2] on CIFAR and ImageNet datasets, respectively. In this section, we show LINe can be used for MobileNetV2 [9], which is pre-trained on the ImageNet-1k dataset from PyTorch. Experiment settings are the same in Section 4.2. We choose hyperparameters for LINe as pruning percentile $p_w = p_a = 10$ and clipping threshold $\delta = 0.6$. As shown in Table S.3, our method implemented on MobileNetV2 outperformed all the other methods.

C. LINe with Other Shapley-value Approximation

We use the Taylor approximation in the main text to compute the Shapley value. To see the difference of changing approximation to compute the Shapley value, we use IntGrad approximation, which is also introduced in [5]. For input $x^l \in D$, where x^l denotes the sample of class l from dataset D, a contribution(i.e., Shapley value) of i-th neuron a_i in class l, s_i^l is calculated as

$$s_i^l = a_i^l \int_{\alpha=0}^{1} \frac{\partial f_\theta(\alpha a_i^l; x)}{\partial a_i^l} d\alpha. \tag{S.1}$$

Contribution matrix C_{int} can be defined with contribution calculated by Equation S.1. With this contribution matrix C_{int}, we can apply LINe. Table S.4 show the result of LINe with Taylor and IntGrad approximation. The results of both methods are the same. Calculated contributions from both methods are different, but the order of *top-k* neurons is still the same. However, the precomputing time of IntGrad is almost 11 times larger than the Taylor approximation, so it is better to choose Taylor as an approximation method.

D. Additional Theoretical Analysis

The outstanding performance of LINe is grounded on three different groups of papers in the related work section (Sec 2.1-2.3). In Network Dissection [1] and HINT [12], neurons in the deep layer (e.g., penultimate layer) represent a specific concept (e.g., window, mammal). Also, in Khazar et al. [5], neurons with high Shapely values have critical fragments of the encoded input information. We draw an insight from the above studies that a group of neurons in the penultimate layer with high Shapley values for a specific class has essential concepts for classifying that class. We call this group of neurons class-specific neurons. Therefore, we can select important class-specific neurons and mask less important neurons by ranking the contribution of neurons. The pruning parts in LINe (i.e., AP and WP) improve the performance by masking less important neurons which trigger noisy outputs. Since class-specific neurons are activated only for essential concepts for each class, OOD samples with different visual features (i.e., concept) cannot activate most of the class-specific neurons. This simple idea motivates AC by limiting the size of activation, which makes AC treat class-specific features equally and improves OOD detection performance.

*Corresponding authors: Gyeong-Moon Park (gmpark@khu.ac.kr) and Seong Tae Kim (st.kim@khu.ac.kr).

Table S.1. **Comparison on CIFAR-10 benchmark.** Table shows comparison with competitive post-hoc OOD detection methods on CIFAR-10 benchmark. All values in this table are percentages. The average over six OOD test datasets is also reported.

Method	OOD Datasets												Average	
	SVHN		Textures		iSUN		LSUN		LSUN-Crop		Places365			
	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC
MSP [3]	47.24	93.48	64.15	88.15	42.31	94.52	42.10	94.51	33.57	95.54	63.02	88.57	48.73	92.46
ODIN [7]	25.29	94.57	57.50	82.38	3.98	98.90	3.09	99.02	4.70	98.86	52.85	88.55	24.57	93.71
Mahalanobis [6]	6.42	98.31	21.51	92.15	9.78	97.25	9.14	97.09	56.55	86.96	85.14	63.15	31.42	89.15
Energy [8]	40.61	93.99	56.12	86.43	10.07	98.07	9.28	98.12	3.81	99.15	39.40	91.64	26.55	94.57
ReAct [10]	41.64	93.87	43.58	92.47	12.72	97.72	11.46	97.87	5.96	98.84	43.31	91.03	26.45	94.67
DICE [11]	25.99	95.90	41.90	88.18	4.36	99.14	3.91	99.20	0.26	99.92	48.59	89.13	20.83	95.24
DICE + ReAct [11]	12.49	97.61	25.83	94.56	5.27	99.02	3.95	99.14	0.43	99.89	50.94	89.63	16.48	96.64
LINe (Ours)	11.38	97.75	23.44	95.12	4.90	99.01	4.19	99.09	0.61	99.83	43.78	91.12	14.72	96.99

Table S.2. **Comparison on CIFAR-100 benchmark.** Table shows comparison with competitive post-hoc OOD detection methods on CIFAR-100 benchmark. All values in this table are percentages. The average over six OOD test datasets is also reported.

Method	OOD Datasets												Average	
	SVHN		Textures		iSUN		LSUN		LSUN-Crop		Places365			
	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC	FPR95	AUROC
MSP [3]	81.70	75.40	84.79	71.48	85.99	70.17	85.24	69.18	60.49	85.60	82.55	74.31	80.13	74.36
ODIN [7]	41.35	92.65	82.34	71.48	67.05	83.84	65.22	84.22	10.54	97.93	82.32	76.84	58.14	84.49
Mahalanobis [6]	22.44	95.67	62.39	79.39	31.38	93.21	23.07	94.20	68.90	86.30	92.66	61.39	55.37	82.73
Energy [8]	87.46	81.85	84.15	71.03	74.54	78.95	70.65	80.14	14.72	97.43	79.20	77.72	68.45	81.19
ReAct [10]	83.81	81.41	77.78	78.95	65.27	86.55	60.08	87.88	25.55	94.92	82.65	74.04	62.27	84.47
DICE [11]	54.65	88.84	65.04	76.42	48.72	90.08	49.40	91.04	0.93	99.74	79.58	77.26	49.72	87.23
DICE + ReAct [11]	55.52	88.02	41.54	86.26	44.32	91.44	54.44	89.84	7.56	98.61	94.05	56.26	49.57	85.07
LINe (Ours)	31.10	91.90	39.29	87.84	24.07	94.85	25.32	94.63	5.72	98.87	88.50	63.93	35.67	88.67

Table S.3. **LINe with MobileNetV2 on ImageNet benchmark.** Results compared with competitive post-hoc OOD detection methods on ImageNet benchmark are reported. All values in this table are percentages and averaged over four OOD test datasets. ↓ indicates smaller value means higher performance and ↑ indicates vice versa.

Method	OOD Datasets								Average	
	iNaturalist		SUN		Places		Textures			
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
MSP [3]	64.29	85.32	77.02	77.10	79.23	76.27	73.51	77.30	73.51	79.00
ODIN [7]	55.39	87.62	54.07	85.88	57.36	84.71	49.96	85.03	54.20	85.81
Mahalanobis [6]	62.11	81.00	47.82	86.33	52.09	83.63	92.38	33.06	63.60	71.01
Energy score [8]	59.50	88.91	62.65	84.50	69.37	81.19	58.05	85.03	62.39	84.91
ReAct [10]	42.40	91.53	47.69	88.16	51.56	86.64	38.42	91.53	45.02	89.47
DICE [11]	43.09	90.83	38.69	90.46	53.11	85.81	32.80	91.30	41.92	89.60
DICE + ReAct [11]	32.30	93.57	31.22	92.86	46.78	88.02	16.28	96.25	31.64	92.68
LINe (Ours)	24.95	95.53	33.19	92.94	47.95	88.98	12.30	97.05	29.60	93.62

Table S.4. **Comparison on different approximation methods.** Results with different Shapley value approximation are reported on CIFAR-10, CIFAR-100, ImageNet benchmarks. All values in this table are percentages and averaged. ↓ indicates smaller value means better performance and ↑ indicates vice versa.

Method	CIFAR-10		CIFAR-100		ImageNet	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
Taylor	14.71	96.99	35.67	88.67	20.70	95.03
IntGrad	14.71	96.99	35.67	88.67	20.70	95.03

References

[1] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549, 2017. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Proceedings of the European conference on computer vision*, pages 630–645. Springer, 2016. 1

[3] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017. 2

[4] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 1

[5] Ashkan Khakzar, Soroosh Baselizadeh, Saurabh Khanduja, Christian Rupprecht, Seong Tae Kim, and Nassir Navab. Neural response interpretation through the lens of critical pathways. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13528–13538, 2021. 1

[6] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pages 7167–7177, 2018. 2

[7] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *Proceedings of International Conference on Learning Representations*, 2018. 2

[8] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33:21464–21475, 2020. 2

[9] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 1

[10] Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. *Advances in Neural Information Processing Systems*, 34:144–157, 2021. 1, 2

[11] Yiyou Sun and Yixuan Li. Dice: Leveraging sparsification for out-of-distribution detection. In *European Conference on Computer Vision*, 2022. 1, 2

[12] Andong Wang, Wei-Ning Lee, and Xiaojuan Qi. Hint: Hierarchical neuron concept explainer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10254–10264, 2022. 1