Poisoned	Network Dissection					MILAN				
images %	std = 0.01	0.02	0.03	0.04	0.05	0.01	0.02	0.03	0.04	0.05
20	0.586	1.367	3.125	3.906	4.492	6.45	11.33	13.45	12.5	16.21
40	0.195	2.930	4.492	5.859	6.836	8.79	13.48	17.97	19.73	21.48
60	0.781	3.711	6.641	9.375	11.328	9.38	16.21	20.31	23.04	24.80
80	0.781	4.883	8.398	12.109	15.430	11.72	18.55	24.41	25.58	27.34
100	0.781	6.055	10.352	13.477	17.578	18.36	28.71	35.94	38.28	41.80

Table A.1: Percentage of units manipulated (higher means corruption technique has stronger effects) in *VGG16-Places365* conv5_3 by gradual poisoning of the probing dataset with Gaussian random noise for Network Dissection.

A. Appendix

A.1. Noise corruption of probing dataset

This section extends our experiments with random noise data corruption to manipulate neuron explanations. In Section A.1.1, we will study the effect of increasing the fraction of images poisoned with Gaussian random noise for an larger set of standard deviations compared to Fig 2.b in the main text. In Sec A.1.2, we will analyze the effect of uniform and Bernoulli bounded noise data corruption on neuron explanations.

A.1.1 Gradual data poisoning with Gaussian noise

Table A.1 shows the percentage of neurons manipulated by gradual poisoning of the probing dataset with Gaussian random noise. We observe that even adding noise to 20% of probing dataset can manipulate around 5% neurons and 17% neurons in the conv5_3 layer for Network Dissection and MILAN respectively.

A.1.2 Robustness analysis with bounded noise

In this section, we study the effect of data corruption with bounded noise on neuron explanations. We poison probing dataset with uniform and Bernoulli random noise and obtain robustness estimates for *VGG16-Places365*. Fig A.1 visualizes images in the probing dataset with added Gaussian, uniform, and Bernoulli noise. Even with noise standard deviation of 0.05, the images are visually unchanged, making it hard to discern data corruption through manual examination.

Layer	Type	std = 0.01	0.02	0.03	0.04	0.05
conv3_3	G	0.0	0.0	0.0	0.390	0.781
	U	0.0	0.0	0.0	0.0	0.0
	B	0.0	0.0	0.0	**1.172**	**1.953**
conv4_3	G	0.195	1.171	3.125	4.687	6.0546
	U	0.195	0.390	1.953	3.320	5.860
	B	0.195	**2.539**	**5.664**	**10.938**	**12.500**
conv5_3	G	0.781	5.859	10.156	13.281	17.773
	U	**0.586**	1.171	7.6171	13.867	21.679
	B	0.390	**12.500**	**21.680**	**36.328**	**41.601**

Table A.2: Percentage of neurons manipulated (higher means corruption technique has stronger effects) in *VGG16-Places365* by addition of Gaussian noise(G), Uniform noise(U), and Bernoulli noise(B) for Network Dissection. We do not observe any successful manipulation in the layers conv1_1 and conv2_2. **Highlighted** values indicate maximum percentage of units manipulated for a given layer and noise level. Bernoulli noise leads to highest number of manipulated neurons.

Network Dissection Table A.2 shows the percentage of neurons manipulated in *VGG16-Places365* with bounded noise data corruption. Bernoulli noise manipulates the highest percentage of neurons at different noise levels, with a maximum of 42% units in the conv5_3 layer.

MILAN Table A.3 shows the average F1-BERT score between clean and manipulated descriptions with bounded noise data corruption. Similar to Network Dissection, we observe that Bernoulli noise results in maximum change in the neuron descriptions, with a noise level of 0.05 reducing the average F1-BERT score to 0.73 in the conv5_3 layer.

(a) Gaussian noise

(b) Uniform noise

(c) Bernoulli noise

Figure A.1: Visualizing images poisoned with (a) Gaussian, (b) Uniform, and (c) Bernoulli noise. The first image of each row visualizes the original image, followed poisoned images with a standard deviation from 0.01 to 0.05 in step size of 0.01. The images are visually unchanged even with noise std=0.05 added to it.

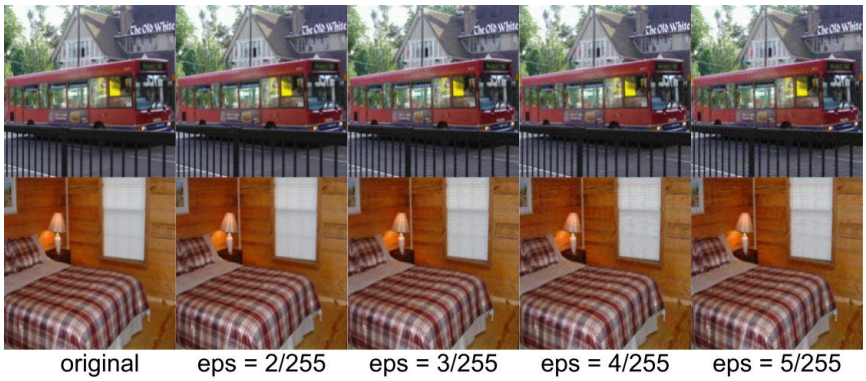

Figure A.2: Visualizing images poisoned with designed data corruption for different values of corruption magnitude ϵ.

Layer	Type	Noise				
		0.01	0.02	0.03	0.04	0.05
conv1_2	G	0.936 ± 0.172	$\mathbf{0.892 \pm 0.208}$	0.884 ± 0.222	0.870 ± 0.196	$\mathbf{0.843 \pm 0.203}$
	U	0.967 ± 0.117	0.933 ± 0.152	0.900 ± 0.202	0.865 ± 0.230	0.876 ± 0.192
	B	$\mathbf{0.918 \pm 0.189}$	0.896 ± 0.189	$\mathbf{0.867 \pm 0.206}$	$\mathbf{0.843 \pm 0.202}$	0.866 ± 0.213
conv2_2	G	0.918 ± 0.169	0.862 ± 0.202	0.856 ± 0.212	0.829 ± 0.214	0.848 ± 0.206
	U	0.933 ± 0.155	0.910 ± 0.177	0.867 ± 0.208	$\mathbf{0.853 \pm 0.216}$	0.866 ± 0.200
	B	$\mathbf{0.914 \pm 0.176}$	$\mathbf{0.870 \pm 0.201}$	$\mathbf{0.849 \pm 0.205}$	0.860 ± 0.204	$\mathbf{0.836 \pm 0.209}$
conv3_3	G	0.922 ± 0.181	0.886 ± 0.214	0.845 ± 0.226	0.813 ± 0.235	0.813 ± 0.238
	U	0.937 ± 0.167	0.909 ± 0.192	0.889 ± 0.201	0.875 ± 0.213	0.842 ± 0.224
	B	$\mathbf{0.928 \pm 0.167}$	$\mathbf{0.852 \pm 0.232}$	$\mathbf{0.836 \pm 0.222}$	$\mathbf{0.793 \pm 0.247}$	$\mathbf{0.795 \pm 0.252}$
conv4_3	G	0.896 ± 0.207	0.834 ± 0.238	0.778 ± 0.259	0.765 ± 0.259	0.735 ± 0.263
	U	0.931 ± 0.174	0.881 ± 0.218	0.840 ± 0.238	0.820 ± 0.243	0.785 ± 0.259
	B	$\mathbf{0.897 \pm 0.208}$	$\mathbf{0.812 \pm 0.249}$	$\mathbf{0.797 \pm 0.248}$	$\mathbf{0.746 \pm 0.264}$	$\mathbf{0.733 \pm 0.270}$
conv5_3	G	0.886 ± 0.223	0.832 ± 0.251	0.787 ± 0.269	0.767 ± 0.269	0.742 ± 0.279
	U	0.935 ± 0.174	0.868 ± 0.237	0.842 ± 0.253	0.807 ± 0.261	0.801 ± 0.257
	B	$\mathbf{0.888 \pm 0.224}$	$\mathbf{0.823 \pm 0.256}$	$\mathbf{0.783 \pm 0.275}$	$\mathbf{0.758 \pm 0.271}$	$\mathbf{0.730 \pm 0.283}$

Table A.3: Average F1-BERT score (lower score means our corruption technique has stronger effects) between clean and manipulated descriptions in *VGG16-Places365* with Gaussian noise(G), Uniform noise(U), and Bernoulli noise(B) data corruption for MILAN. Lower score indicates higher dissimilarity and more successful untargeted data corruption. **Highlighted** values indicate minimum F1-BERT score for a given layer and noise level. Bernoulli noise leads to lowest F1-BERT score.

Clean description	Manipulated description	Score
"vehicle windows"	"car windows"	0.942
"doors"	"wall"	0.858
"the top of round objects"	"circular shaped objects"	0.659
"containers"	"circular object"	0.638
"doors"	"buildings"	0.604
"signs and grids"	"red colored object with text"	0.564
"blue areas in pictures"	"blue skies"	0.552
"trees and flowers"	"green colored objects"	0.422
"vertical lines"	"fencing"	0.327

Table A.4: Sampled F1-BERT scores with untargeted data poisoning of MILAN

A.2. Designed corruption of probing dataset

This section extends our experiments with the designed data corruption to manipulate neuron explanations. We present our results with untargeted data corruption without using ground-truth segmentation for Network Dissection in Sec A.2.1, discuss challenges with targeted data corruption for MILAN in Sec A.2.2, provide an ablation study with a reduced form of Eq 9 in untargeted setting in Sec A.2.3, and perform robustness analysis of adversarially-trained Resnet50 model in Sec A.2.4. Fig A.2 visualizes poisoned images with our objective function, and Fig A.3 visualizes successful targeted data corruption for Network Dissection on selected neuron units.

A.2.1 Untargeted data corruption on Network Dissection with $L_c(x)$ from uncorrupted runs

In this section, we analyze untargeted data corruption for Network Dissection under the assumption that the ground-truth segmentation information is not accessible, and hence we cannot obtain $L_c(x)$ directly for data corruption with Eq 9. This is a typical setting when Network Dissection (and other NEMs) are offered as a cloud service, with the user providing the

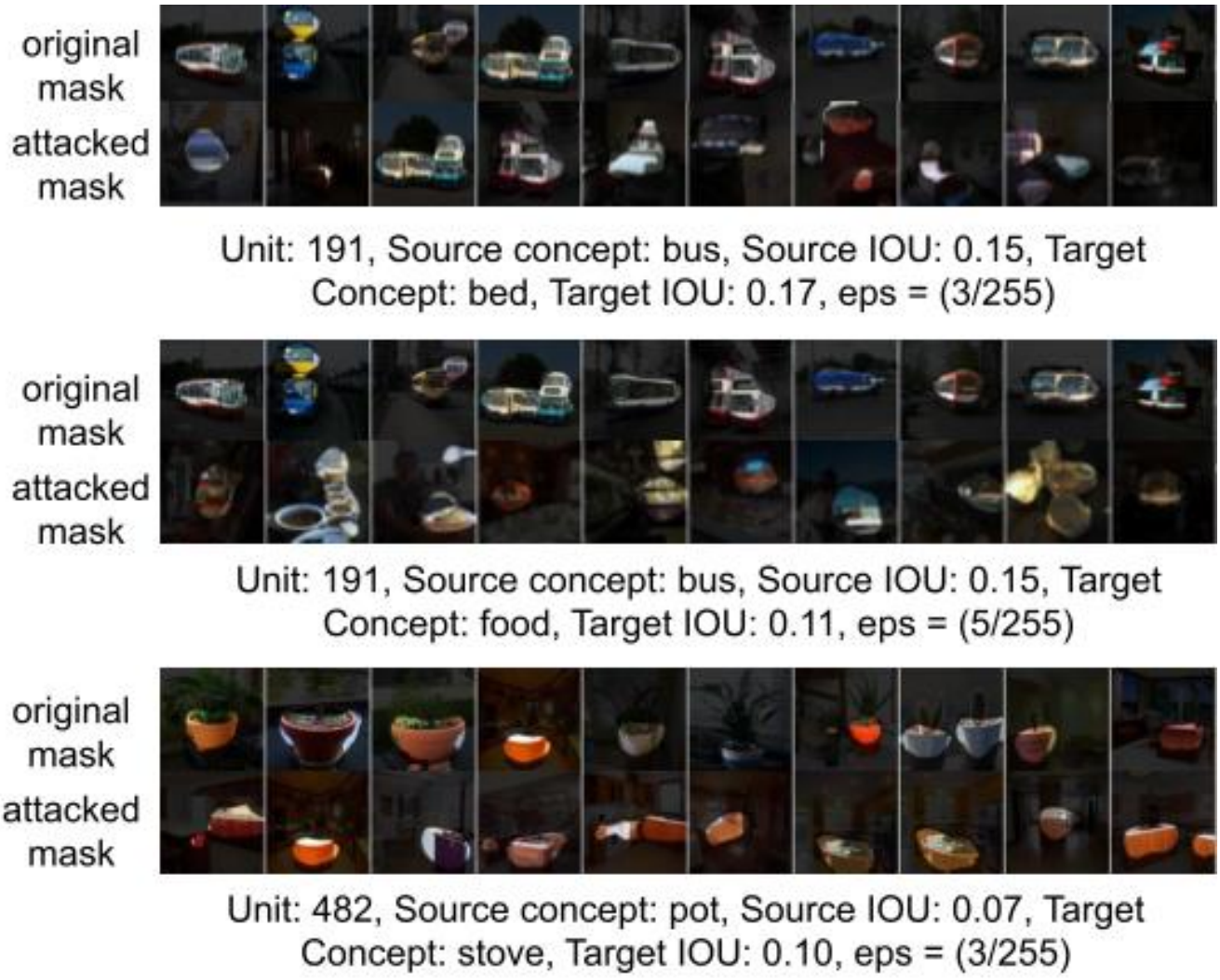

Figure A.3: Successful targeted data corruption with least corruption magnitude ϵ to manipulate unit 191 and unit 482. First row in subfigure visualizes top activating images from $\mathcal{D}_{probe}$. Second row in each subfigure highlights pixels with value greater than activation threshold T_{neuron}.

probing dataset and the per-pixel segmentation generation being handled securely in the cloud. We follow our formulation in Sec 3.2 for obtaining $L_c(x)$ from the general NEMs. The results are shown in Table A.5. We observe that the designed data corruption is less effective than using the baseline segmentation data; however, we can achieve 45% success rate on $conv5\_3$ layer by poisoning less than 10% images. This shows that hiding the `sim` function is not a viable defense against our designed data corruption method.

Layer	$\epsilon = \frac{2}{255}$	$\frac{4}{255}$	$\frac{6}{255}$
conv4_3	5.41	8.11	13.51
conv5_3	5.13	23.08	46.15

Table A.5: Percentage of units manipulated (higher means our corruption technique has stronger effects) in *VGG16-Places365* by untargeted data corruption with $L_c(x)$ obtained from uncorrupted runs for Network Dissection. We can obtain 45% manipulation success rate without using ground truth segmentation data.

A.2.2 Targeted data corruption on MILAN

The targeted data corruption aims to manipulate clean descriptions of MILAN to a target description by poisoning images with designed perturbations. We consider a targeted data corruption successful if the F1-BERT score between manipulated and target description is more significant than 0.642. We observe in our experiments that the targeted designed data corruption on MILAN can manipulate a maximum of 40% neurons in *VGG16-Places365*. This can be explained by the fact that the manipulated description is dependent on the number of poisoned images. MILAN obtains $L_c(x)$ for computing corruption using standard (non-corrupted) runs and requires hyperparameter for the number of poisoned images. We argue that the manipulated description is dependent on the number of poisoned images, and increasing the number of poisoned images after a threshold results affects the activations of unrelated concepts. Fig A.4 shows the targeted designed data corruption on Unit 191 to change its concept from "buses" to "beds" with the varying number of poisoned images. We observe that the F1-BERT score between manipulated and target description decreases to a very low value if the number of corrupted images exceeds 400, implying that the manipulated description is very different from target description.

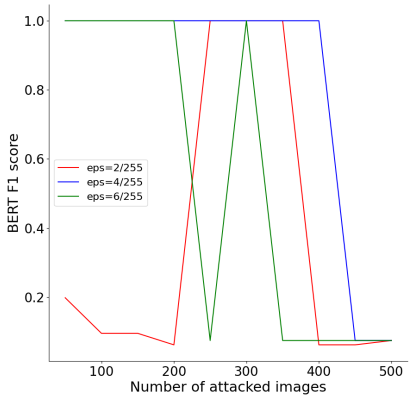

Figure A.4: F1-Bert score between clean and manipulated description with increasing number of poisoned images for Unit 191 conv5_3 of *VGG16-Places365* in MILAN. Higher score means manipulated description and target description are similar, with F1-BERT score being 1.0 if they match. The F1-BERT score increases and then decreases, implying that the manipulated description is a function of the number of poisoned images.

A.2.3 Ablation: Untargeted data corruption

An alternative untargeted data corruption objective for neuron i with concept c_i^*, and image $x_j \in \mathcal{D}_{probe}$ can be formulated as

$$\begin{aligned} \min_{\delta_j} \quad & act_{i,c_i^*}^{avg}(x_j + \delta_j) \\ \text{s.t.} \quad & ||\delta_j||_\infty \leq \epsilon \\ & x_j + \delta_j \in [0,1]^l \end{aligned} \tag{1}$$

This formulation can be intuitively understood as trying to reduce the activations of pixels associated with the source category. We refer to this formulation as U1 and our formulation in Eq 9 as U2. Table A.6 and Table A.7 compare the strength of untargeted data corruption with U1 and U2 for Network Dissection and MILAN respectively. We observe that U2 consistently outperforms U1, implying that using a random target label results in a stronger data corruption in an untargeted setting.

A.2.4 Robustness of Adversarially-trained models

In this section, we manipulate descriptions of adversarially-trained (ϵ=$\frac{2}{255}$) Resnet50. The attack success rate (ASR) on Network Dissection decreases to 41.2%, 76.0%, 88.0% for PGD ϵ=$\frac{2}{255}$, $\frac{4}{255}$, $\frac{6}{255}$ in layer 3, from 75.3%, 98.6%, 98.6% of the standard model respectively. This result suggests that adversarial training may only help to alleviate our proposed attacks a bit since the ASR is still pretty high, demonstrating the need of a stronger defense.

Layer	type	$\epsilon = \frac{2}{255}$	$\frac{4}{255}$	$\frac{6}{255}$
conv1_2	U1	0.0	0.0	0.0
	U2	**47.73**	**52.27**	**54.55**
conv2_2	U1	0.0	0.0	0.0
	U2	**4.08**	**22.45**	**38.78**
conv3_3	U1	6.67	33.33	46.67
	U2	**34.88**	**58.14**	**67.44**
conv4_3	U1	53.85	53.85	69.23
	U2	**57.45**	**95.74**	**100.0**
conv5_3	U1	38.46	53.85	53.85
	U2	**70.59**	**96.08**	**96.08**

Table A.6: Percentage of units manipulated (higher score means corruption technique has stronger effects) in *VGG16-Places365* by untargeted data corruption with objective function U1 and U2 for Network Dissection. **Highlighted** values indicates stronger corruption between U1 and U2 for a given layer and corruption magnitude ϵ. Objective function U2 consistently outperforms objective function U1 and achieves a higher manipulation success rate.

Layer	type	$\epsilon = \frac{2}{255}$	$\epsilon = \frac{4}{255}$	$\epsilon = \frac{6}{255}$
conv1_2	U1	0.91 ± 0.23	0.95 ± 0.17	0.92 ± 0.20
	U2	$\mathbf{0.89 \pm 0.26}$	$\mathbf{0.89 \pm 0.26}$	$\mathbf{0.86 \pm 0.26}$
conv2_2	U1	0.95 ± 0.13	0.89 ± 0.18	0.88 ± 0.19
	U2	$\mathbf{0.94 \pm 0.16}$	$\mathbf{0.90 \pm 0.19}$	$\mathbf{0.84 \pm 0.21}$
conv3_3	U1	0.86 ± 0.19	0.76 ± 0.24	0.71 ± 0.24
	U2	$\mathbf{0.79 \pm 0.28}$	$\mathbf{0.73 \pm 0.26}$	$\mathbf{0.66 \pm 0.24}$
conv4_3	U1	0.73 ± 0.25	0.69 ± 0.26	0.68 ± 0.26
	U2	$\mathbf{0.63 \pm 0.27}$	$\mathbf{0.58 \pm 0.24}$	$\mathbf{0.51 \pm 0.18}$
conv5_3	U1	0.75 ± 0.27	0.59 ± 0.27	0.53 ± 0.21
	U2	$\mathbf{0.54 \pm 0.24}$	$\mathbf{0.46 \pm 0.16}$	$\mathbf{0.47 \pm 0.14}$

Table A.7: Average F1-BERT score (lower score means our corruption technique has stronger effects) in *VGG16-Places365* by untargeted data manipulation with objective function U1 and U2 for MILAN. **Highlighted** values indicates strong corruption between U1 and U2 for a given layer and corruption magnitude ϵ. F1-BERT score between similar strings has its maximum value 1.0, and lower F1-BERT score indicates higher dissimilarity and more successful untargeted data corruption. Objective function U2 consistently outperforms objective function U1 and achieves a lower F1-Bert Score.