# CoSy: Evaluating Textual Explanations of Neurons

**Laura Kopf** [1 2] **Philine Lou Bommer** [1 3] **Anna Hedström** [1 3 4] **Sebastian Lapuschkin** [4] **Marina M.-C. Höhne** [1 2] **Kirill Bykov** [1 3 5]

## Abstract

A crucial aspect of understanding the complex nature of Deep Neural Networks (DNNs) is the ability to explain learned concepts within their latent representations. While methods exist to connect neurons to human-understandable textual descriptions, evaluating the quality of these explanations is challenging due to the lack of a unified quantitative approach. We introduce CoSy (Concept Synthesis), a novel, architecture-agnostic framework for evaluating textual explanations of latent neurons. Given textual explanations, our proposed framework uses a generative model conditioned on textual input to create data points representing the explanations, comparing the neuron's response to these and control data points to estimate explanation quality. We validate our framework through meta-evaluation experiments and benchmark various concept-based textual explanation methods for Computer Vision tasks, revealing significant differences in quality. We provide an open-source implementation on GitHub[1].

## 1. Introduction

One of the key obstacles to the wider adoption of Machine Learning is the inherent opacity of Deep Neural Networks (DNNs)—we do not understand why these machines make the predictions they do. To address this, the field of Explainable AI (XAI) (Samek & Müller, 2019; Xu et al., 2019) has emerged, aiming to reveal DNN decision-making processes in a human-understandable fashion, thereby improving model transparency, fairness, interpretability, and safety. XAI has expanded from explaining specific inputs using saliency maps (Simonyan et al., 2014; Bach et al., 2015; Selvaraju et al., 2017; Smilkov et al., 2017) to analyzing the global behavior of models by examining the functional purpose of neurons within DNNs (Bau et al., 2017; Mu & Andreas, 2020; Hernandez et al., 2022; Kalibhat et al., 2023; Oikarinen & Weng, 2023; Bykov et al., 2023b). A popular approach is to label neurons with human-understandable textual concepts based on the concepts they detect. These methods have evolved from label-specific descriptions (Bau et al., 2017) to more complex compositional (Mu & Andreas, 2020; Bykov et al., 2023b) and open-vocabulary explanations (Hernandez et al., 2022; Oikarinen & Weng, 2023). Previous work has shown that neurons can be affected by biases (Goh et al., 2021), spurious correlations (Bykov et al., 2023a), and backdoor attacks (Casper et al., 2023). Therefore, validating neuron-labeling methods is crucial given their potential impact. Despite their importance, the lack of a universally accepted quantitative evaluation measure for open-vocabulary neuron descriptions remains a significant challenge, hindering comprehensive cross-comparisons.

With our work, we aim to bridge this gap by introducing a novel quantitative evaluation framework named CoSy, for evaluating open-vocabulary explanations for neurons in Computer Vision (CV) models (illustrated in Figure 1). We summarize our contributions as below:

**(C1)** We provide the first general-purpose, quantitative evaluation framework CoSy (Section 3) that enables the evaluation of individual or a set of textual explanation methods for CV models.

**(C2)** In a series of meta-evaluation experiments (Section 4), we analyze the choice of generative models and prompts for synthetic image generation, demonstrating framework reliability.

**(C3)** We benchmark existing explanation methods (Section 5) and extract novel insights, revealing substantial variability in the quality of explanations. Generally, textual explanations for lower layers are less accurate compared to those for higher layers.

[1]UMI Lab, ATB Potsdam, Germany [2]University of Potsdam, Germany [3]TU Berlin, Germany [4]Fraunhofer Heinrich-Hertz-Institute, Berlin, Germany [5]BIFOLD, Berlin, Germany. Correspondence to: Laura Kopf <lkopf@atb-potsdam.de>, Philine Lou Bommer <pbommer@atb-potsdam.de>, Anna Hedström <ahedstroem@atb-potsdam.de>, Sebastian Lapuschkin <sebastian.lapuschkin@hhi.fraunhofer.de>, Marina M.-C. Höhne <mhoehne@atb-potsdam.de>, Kirill Bykov <kbykov@atb-potsdam.de>.

[1]https://github.com/lkopf/cosy

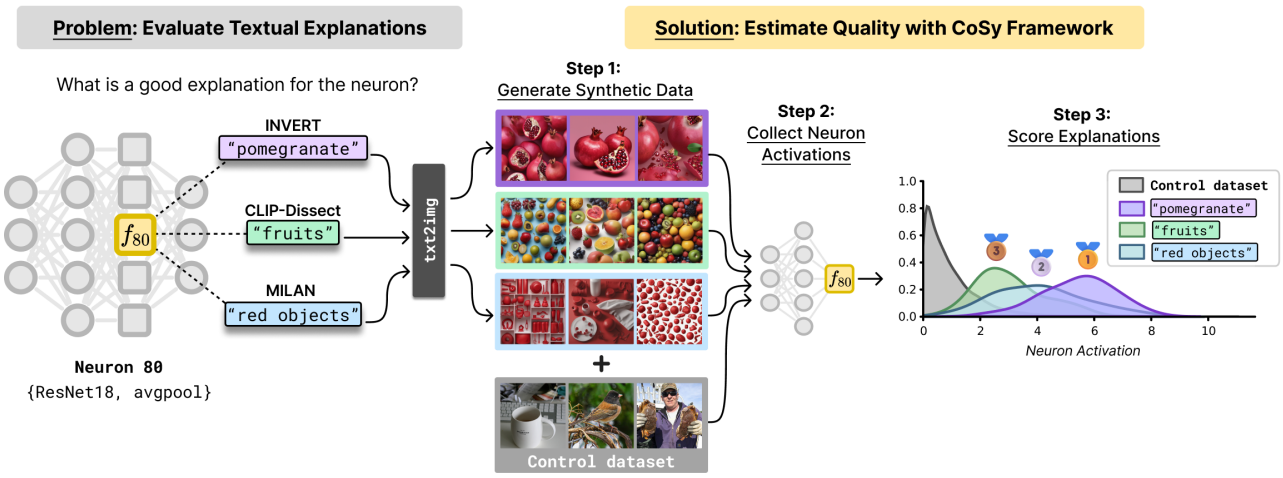


*Figure 1.* A schematic illustration of the CoSy evaluation framework for Neuron 80 in `ResNet18`'s avgpool layer. The current challenge lies in the absence of dataset- and architecture-agnostic evaluation measures to benchmark textual explanations of neurons. To address this, we propose CoSy, a framework consisting of three steps: first, a generative model translates textual concepts into the visual domain, creating synthetic images for each explanation using a text-to-image model. Then, inference is performed on synthetic images, along with a control image dataset, to collect neuron activations. Finally, by comparing the activations on synthetic images with activations on the control dataset, we can quantitatively assess the quality of the textual explanation and compare the results between different explanation methods. The implementation details of this example can be found in Appendix A.2.

## 2. Related Work

**Activation Maximization** Activation Maximization is a commonly used methodology to understand what a neuron has learned to detect (Erhan et al., 2009). Such methods work by identifying input signals that trigger the highest activation in a neuron. This can be achieved synthetically, where an optimization process is employed to create the optimal input that maximizes the neuron's activation (Olah et al., 2017; Nguyen et al., 2016; Fel et al., 2023), or naturally, by finding such inputs within a data corpus (Borowski et al., 2020). Activation Maximization has been employed for explaining latent representations of models (Goh et al., 2021; Yoshimura et al., 2021), including probabilistic models (Grinwald et al., 2023), detection of backdoor attacks (Casper et al., 2023) and spurious correlations (Bykov et al., 2023a). However, one of the key limitations of this methodology lies in its inability to scale; its scalability is limited due to its dependency on users to manually audit maximization signals.

**Automatic Neuron Interpretation** One approach to understanding a neuron's function involves linking neurons with human-understandable concepts through textual descriptions. Network Dissection (NetDissect) (Bau et al., 2017) pioneered this by associating neurons with concepts based on the Intersection over Union (IoU) of activation maps and segmentation masks. Compositional Explanations of Neurons (CompExp) (Mu & Andreas, 2020) enhanced this with compositional concepts. MILAN (Hernandez et al., 2022) expanded to open-vocabulary explanations, allowing descriptions beyond predefined labels. INVERT (Bykov et al., 2023b) used compositional labels and the Area Under the Receiver Operating Characteristic Curve (AUC) for explanations without segmentation masks. FALCON (Kalibhat et al., 2023) and CLIP-Dissect (Oikarinen & Weng, 2023) utilized CLIP models (Radford et al., 2021) for image-text similarity. Each method defines its own optimization criteria, lacking a unified consensus on what constitutes a good explanation. For detailed descriptions of the methods and their optimization objectives, please refer to Appendix A.1.

**Prior Methods for Evaluation** Significant effort has been made towards evaluating *local* explanations (Agarwal et al., 2022; Hedström et al., 2023; Hedström et al., 2023), but there is limited focus on *global* methods, with no unified benchmarking approach. INVERT and CLIP-Dissect evaluate accuracy by comparing generated neuron labels with ground truth descriptions. However, this evaluation is limited to output neurons and fixed labels only. CLIP-Dissect also measures explanation quality by computing cosine similarity in a sentence embedding space. FALCON uses human studies via Amazon Mechanical Turk for evaluation, and MILAN uses BERTScores (Zhang et al., 2019) to compare neuron labeling methods against human annotations. While human studies are generally beneficial, the conventional setup can be misleading and may not fully capture the desired evaluation criteria. Typically, annotators describe

images that activate a neuron, which is then compared to an automatic explanation. However, this approach primarily evaluates alignment with the most activating images rather than the accuracy of the explanation in describing the neuron's function.

## 3. Method

Consider a Deep Neural Network (DNN) represented by the function $g : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{X} \subset \mathbb{R}^{h \times w \times c}$ denotes the input image domain and $\mathcal{Z} \subset \mathbb{R}^l$ represents the model's output domain. We can view the model as a composition of two functions, $F : \mathcal{X} \to \mathcal{Y}$, and $L : \mathcal{Y} \to \mathcal{Z}$, such that $g = L \circ F$. Here $\mathcal{Y} \subset \mathbb{R}^{k \times w^* \times h^*}$, where $k \in \mathbb{N}$ is the number of neurons in the layer, and $w^*, h^* \in \mathbb{N}$ represent the width and height of the feature map, respectively. The function $F$, which we refer to as the *feature extractor*, can be chosen based on the layer of the model we aim to inspect. This could be an existing layer within the model or a concept bottleneck layer (Yuksekgonul et al., 2022). We refer to the $i$-th neuron within the layer as $f_i(\boldsymbol{x}) = F_i(\boldsymbol{x}) : \mathcal{X} \to \mathbb{R}^{w^* \times h^*}$. Within the scope of this paper, we refer to *explanation method* as an operator $\mathcal{E}$ that maps a neuron to the textual description $s = \mathcal{E}(f_i) \in \mathcal{S}$, where $\mathcal{S}$ is a set of potential textual explanations. The specific set of explanations depends on the implementation of the particular method.

### 3.1. COSY: Evaluating Open-Vocabulary Explanations

A good textual explanation for a neuron should be a human-understandable description of an input that activates the neuron strongly. Modern methods often use open-vocabulary explanations, complicating the quantitative collection of natural data that represents the explanation. To address this, COSY uses generative models to create data points corresponding to the textual explanation. The neuron's response to these synthetic images is compared to its activation on control images representing random concepts, enabling a quantitative evaluation of the alignment between the explanation and the neuron.

Parameters of the proposed method include a control dataset $\mathbb{X}_0 = \{\boldsymbol{x}_1^0, \ldots, \boldsymbol{x}_n^0\} \subset \mathcal{X}, n \in \mathbb{N}$ – containing natural images that represent the concepts the model was originally trained on, a generative model $p_M$, that is used for synthesizing images, and a number of generated images $m \in \mathbb{N}$. The control dataset typically includes a balanced selection of validation class images. Given a neuron $f$ and explanation $s \in \mathcal{S}$, COSY evaluates the alignment between the explanation and a neuron in 3 consecutive steps, which are illustrated in Figure 1.

1. **Generate Synthetic Data.** The first step involves generating synthetic images for a given explanation $s \in \mathcal{S}$, which we use as a prompt to a generative model $p_M$ to create a collection of synthetic images, denoted as $\mathbb{X}_1 = \{\boldsymbol{x}_1^1, \ldots, \boldsymbol{x}_m^1\} \sim p_M(\boldsymbol{x} \mid s)$. This collection consists of $m \in \mathbb{N}$ images, where $m$ is adjustable as a parameter of the evaluation procedure.

2. **Collect Neuron Activations.** Given the control dataset $\mathbb{X}_0$ and the set of generated synthetic images $\mathbb{X}_1$, we collect activations as follows:

$$\begin{aligned} \mathbb{A}_0 &= \{\sigma(f(\boldsymbol{x}_1^0)), \ldots, \sigma(f(\boldsymbol{x}_n^0))\} \in \mathbb{R}^n, \\ \mathbb{A}_1 &= \{\sigma(f(\boldsymbol{x}_1^1)), \ldots, \sigma(f(\boldsymbol{x}_m^1))\} \in \mathbb{R}^m, \end{aligned} \tag{1}$$

where $\sigma$ is an aggregation function for multi-dimensional neurons. Within the scope of our paper, we use Average Pooling as aggregation function

$$\sigma(\boldsymbol{y}) = \frac{1}{w^* h^*} \sum_{i \in [1, w^*], j \in [1, h^*]} \boldsymbol{y}_{i,j}, \quad \boldsymbol{y} \in \mathcal{Y} \subset \mathbb{R}^{w^* \times h^*}. \tag{2}$$

3. **Score Explanations.** The final step of the proposed method relies on the evaluation of the difference between neuron activations on the control dataset $\mathbb{A}_0$ and neuron activations given the synthetic dataset $\mathbb{A}_1$. To quantify this difference, we utilize a *scoring function* $\Psi : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ to measure the difference between the distributions of activations.

In the context of our paper, we employ the following scoring functions:

- **Area Under the Receiver Operating Characteristic (AUC)**

  AUC is a widely used non-parametric evaluation measure for assessing the performance of binary classification. In our method, AUC measures the neuron's ability to distinguish between synthetic and control data points

$$\Psi_{\text{AUC}}(\mathbb{A}_0, \mathbb{A}_1) = \frac{\sum_{a \in \mathbb{A}_0} \sum_{b \in \mathbb{A}_1} \mathbf{1}[a < b]}{|\mathbb{A}_0| \cdot |\mathbb{A}_1|}. \tag{3}$$

- **Mean Activation Difference (MAD)**

  MAD is a parametric measure that quantifies the difference between the mean activation of the neuron on synthetic images and the mean activation on control data points

$$\Psi_{\text{MAD}}(\mathbb{A}_0, \mathbb{A}_1) = \frac{\frac{1}{m} \sum_{b \in \mathbb{A}_1} b - \frac{1}{n} \sum_{a \in \mathbb{A}_0} a}{\sqrt{\frac{1}{n-1} \sum_{a \in \mathbb{A}_0} (a - \bar{a})^2}}, \tag{4}$$

  with mean control activation $\bar{a} = \frac{1}{n} \sum_{a \in \mathbb{A}_0} a$.

These two chosen metrics complement each other. AUC, being non-parametric and stable to outliers, evaluates the classifier's ability to rank synthetic images higher than control images (with scores ranging from 0 to 1, where 1 represents a perfect classifier and 0.5 is random). On the other hand, MAD allows us to parametrically measure the extent to which images corresponding to explanations maximize neuron activation.

## 4. Meta-Evaluation Analysis

In this section, we assess the reliability of our COSY evaluation framework through meta-evaluation.

### 4.1. Synthetic Image Reliability

We investigate the effectiveness of different generative models and prompts in producing visually similar images to natural concepts. Using open-source text-to-image models, namely Stable Diffusion XL 1.0-base (SDXL) (Podell et al., 2023) and Stable Cascade (SC) (Pernias et al., 2023), we varied five different prompts for image generation and measured the cosine similarity (*CS*) between synthetic and natural images of the same concept. Figure 2 illustrates the comparison across all generative models and prompts in terms of *CS*. Refer to Appendix A.3 for further details on the experimental setup. Our results indicate that SDXL, particularly with detailed prompts, achieves higher similarity to natural images than SC. Prompt 5 yields the highest similarity, thus we employed Prompt 5 with the SDXL model for subsequent experiments.

### 4.2. Sanity Check

To ensure our evaluation framework reliably differentiates between random and non-random explanations, we compared COSY scores for ground truth explanations with those for randomly selected explanations. Using a set of 10 output neurons, we found that true explanations consistently achieved high scores while random explanations received low scores, as shown in Table 1. This confirms the effectiveness of our evaluation metric. Further details on the experimental setup are provided in Appendix A.4, with an experiment on whether models respond differently to synthetic and natural images in Appendix A.5, and additional robustness analyses discussed in Appendix A.7.

## 5. Evaluating Explanation Methods

Within the scope of this section, we conduct a thorough cross-comparison of different methods for textual neuron explanations. We evaluate models trained on various datasets and analyze their performance on latent layers where ground truth is unavailable.

*Table 1.* Comparison of true and random explanations on output neurons with known ground truth labels. This table presents the quality estimates for true explanations, derived from target class labels, and random explanations, derived from randomly selected synthetic image classes, across four models pre-trained on ImageNet. Higher values are better. Our results consistently show high scores for true explanations and low scores for random ones.

| MODEL | AUC (STD) (↑) | | MAD (STD) (↑) | |
|---|---|---|---|---|
| | *True* | *Random* | *True* | *Random* |
| RESNET18 | 0.94±0.20 | 0.48±0.17 | 5.84±2.31 | -0.12±0.50 |
| DENSENET161 | 0.95±0.17 | 0.47±0.21 | 6.63±2.42 | -0.12±0.62 |
| GOOGLENET | 0.95±0.16 | 0.40±0.19 | 7.22±2.68 | -0.29±0.48 |
| VIT-B/16 | 0.97±0.11 | 0.54±0.23 | 12.13±4.79 | 0.05±0.57 |

*Table 2.* Benchmarking of explanation methods, explaining neurons on the second to last layers for different models. Explanations are generated with respect to a randomly selected set of 50 neurons where both AUC and MAD are reported. Higher values indicate better performance; **bold** numbers represent the highest scores. Models marked with * were pre-trained on ImageNet, while those marked with † were pre-trained on Places365.

| MODEL (LAYER) | METHOD | AUC (STD) (↑) | MAD (STD) (↑) |
|---|---|---|---|
| RESNET18* (AVGPOOL) | MILAN | 0.61±0.23 | 0.69±1.35 |
| | INVERT | **0.93±0.11** | 3.23±1.72 |
| | CLIP-DISSECT | **0.93±0.11** | **3.85±1.88** |
| VIT-B/16* (FEATURES) | MILAN | 0.53±0.19 | 0.12±0.76 |
| | INVERT | **0.89±0.17** | **1.67±0.82** |
| | CLIP-DISSECT | 0.78±0.19 | 1.29±1.01 |
| DENSENET161† (FEATURES) | MILAN | 0.56±0.28 | 0.44±1.30 |
| | INVERT | **0.85±0.16** | 2.21±1.95 |
| | CLIP-DISSECT | 0.82±0.21 | **2.52±2.33** |
| RESNET50† (AVGPOOL) | MILAN | 0.65±0.28 | 1.11±1.67 |
| | INVERT | **0.94±0.08** | 3.54±1.99 |
| | CLIP-DISSECT | 0.92±0.11 | **3.73±2.39** |

### 5.1. Benchmarking Explanation Methods

In this section, we evaluated three recent textual explanation methods, namely MILAN, INVERT, and CLIP-Dissect. Our analysis involves four distinct models: two pre-trained on the ImageNet dataset (Russakovsky et al., 2015) (ResNet18 (He et al., 2016), ViT-B/16 (Dosovitskiy et al., 2020)) and two pre-trained on the Places365 dataset (Zhou et al., 2017) (DenseNet161 (Huang et al., 2017), ResNet50 (He et al., 2016)). The ImageNet dataset focuses on objects, whereas the Places365 dataset is designed for scene recognition. Consequently, we customized our prompts accordingly: Prompt 5 performs best for object recognition, while for scene recognition, we found that Prompt 4 is more effective. Therefore, Prompt 4 was utilized in the Places365 experiment. For further implementation details, refer to Appendix A.8, and for more information on compute resources, see Appendix A.9.

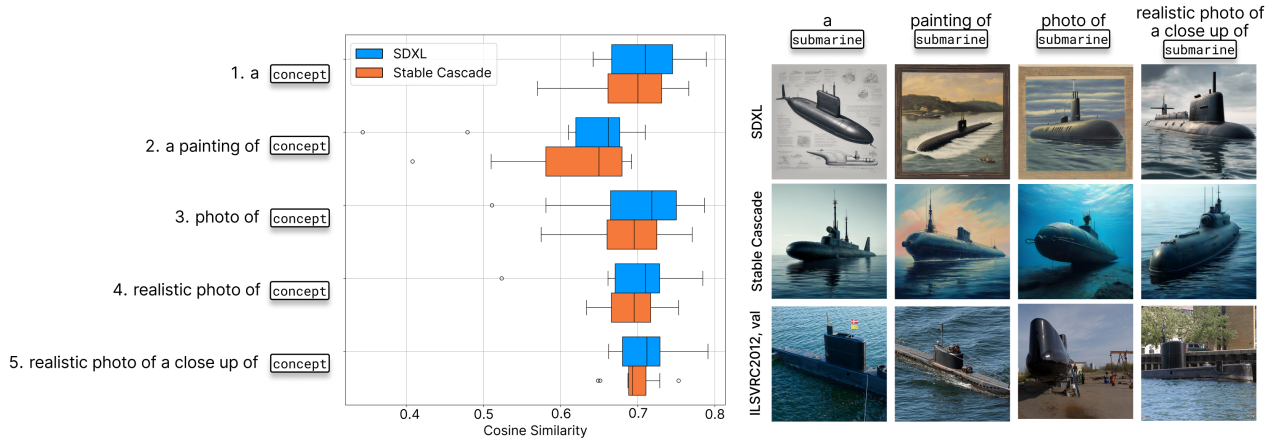


*Figure 2.* An overview of the impact of varying the prompt on the similarity between natural and synthetic images, using two text-to-image models. Left: average Cosine similarity (*CS*) across all natural and synthetic images over all classes are reported. Higher *CS* values are better, indicating greater similarity between the images. Right: an illustration of the visual differences produced by the `SDXL` and `SC` models in response to diverse prompts for the explanation concept "submarine," and natural images from the ImageNet validation dataset (Russakovsky et al., 2015). Our results show that `SDXL` and `SC` generate similar images, with `SDXL` generally being more closely aligned with natural images than `SC`.

Results of the evaluation are presented in Table 2. INVERT generally achieves the highest AUC scores across most models and datasets, although CLIP-Dissect performs similarly on the `ResNet18` applied to ImageNet. CLIP-Dissect consistently demonstrates good results across various models and datasets. However, since INVERT optimizes AUC in explanation generation, it may be biased towards AUC in our evaluation, leading to higher scores. MILAN tends to perform poorly, with an average AUC below 0.65 across all tasks, indicating performance close to random guessing. This is expected as MILAN often generates highly abstract explanations such as "white areas" or "nothing," which are challenging for text-to-image models to accurately generate, contributing to its low scores. While MAD scores suggest that CLIP-Dissect outperforms INVERT for convolutional neural networks on both datasets, INVERT concepts also achieve consistently high scores in these cases. Otherwise, we find similar outcomes for both metrics $\Psi$, with MILAN achieving poor scores in all experimental settings.

### 5.2. Explanation Methods Struggle to Explain Lower Layer Neurons

In addition to general benchmarking, we studied the quality of explanations for neurons in different layers of a model. Lower-layer neurons typically encode lower-level concepts (LeCun et al., 2015), making it interesting to see if explanation methods can capture these. We examined explanations across layers 1 to 4 and the output layer of an ImageNet pre-trained `ResNet18`, including the FALCON method. For more details on the implementation of FALCON see Appendix A.1.4. For each layer, we randomly selected 50 neurons for analysis.

Figure 3 presents the AUC results for all explanation methods across these layers, with MAD scores reported in Appendix A.10. Generally, we observed increasing scores for later layers across all methods and metrics $\Psi$, suggesting higher concept quality in later layers. MILAN and FALCON achieved lower scores across various metrics, with AUC scores of 0.5 indicating random guessing.

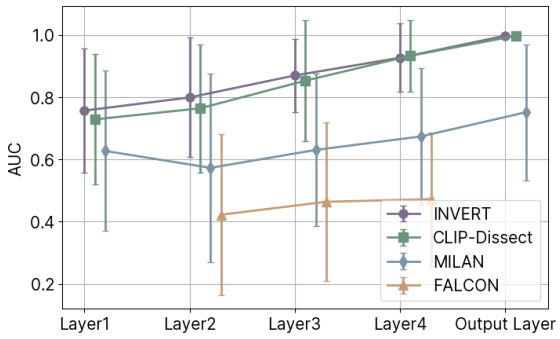


*Figure 3.* A comparison of how different explanation methods vary in their quality, as measured by AUC, across different layers in `ResNet18`. INVERT and CLIP-Dissect similarly high AUC scores across all layers, while MILAN, and FALCON scores are comparably low. Generally, all methods perform increasingly worse on lower layers.

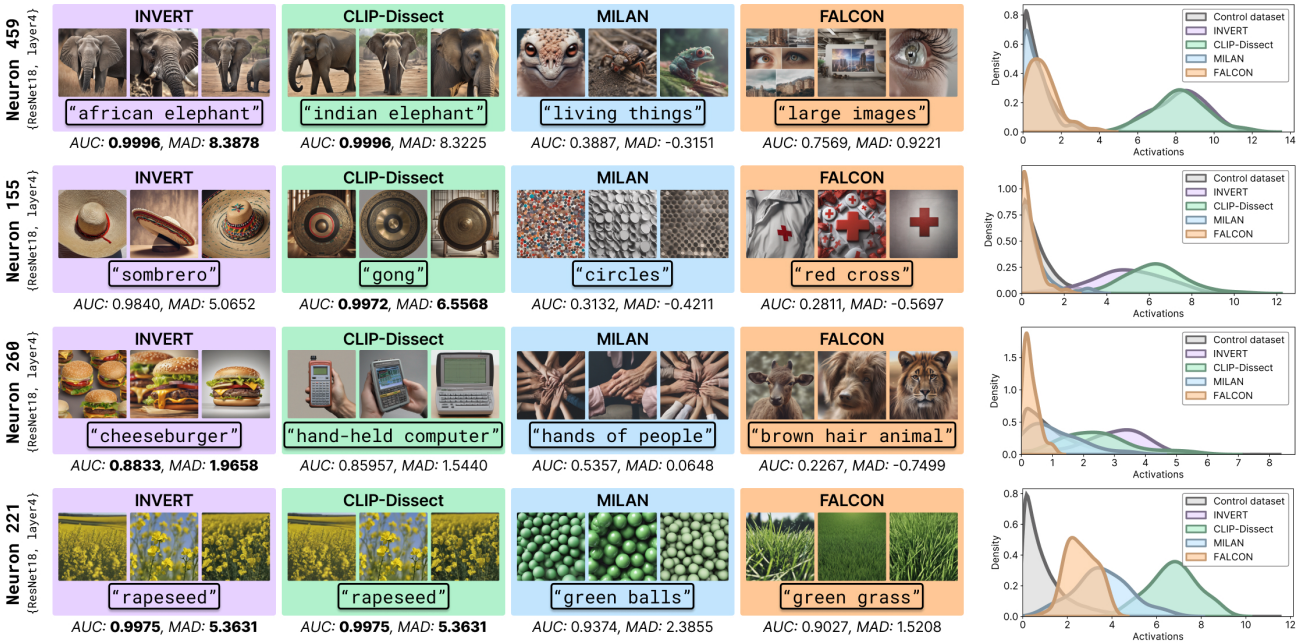


*Figure 4.* A qualitative example, of neuron explanations across 4 neurons. The first four panels include the textual explanation across INVERT, FALCON, CLIP-Dissect, and MILAN alongside three corresponding generated images. The respective AUC and MAD scores are reported below each panel. The last panel shows the activation distributions across 50 generated images for each method and the distribution of the control data.

### 5.3. What are Good Explanations?

In our approach, we propose that testing visual representations of textual explanations on neurons can provide insights into what constitutes good explanations. Building on this premise, CLIP-Dissect and INVERT consistently performed well, as evidenced by qualitative examples in Figure 4, where their explanations share visually similar concepts (neurons 155 and 459) or even identical concepts (neuron 221) while both achieving high AUC and MAD scores. However, INVERT's functionality is restricted to input data labels, unlike CLIP-Dissect, which offers a broader selection of concepts, albeit with reduced interpretability due to its reliance on a black-box model.

There are instances, such as neuron 260 in Figure 4, where all explanations vary significantly, with FALCON and MILAN often providing nearly random explanations, as indicated by their activation distributions overlapping with the control dataset. This observation aligns with our overall findings: both the AUC and MAD scores consistently indicate the low performance of FALCON and MILAN explanations in CoSy evaluation. Neurons 459 and 155 further demonstrate the gap between consistently higher and lower-performing explanation methods.

## 6. Conclusion

In this work, we propose the first automatic evaluation framework for concept-based textual explanations of neurons. Unlike existing ad-hoc evaluation methods, we can now quantitatively compare different concept-based textual explanation methods against each other with CoSy and test, whether the given explanation describes the neuron accurately. Our meta-evaluation reveals that INVERT and CLIP-Dissect excel in providing high-quality concepts for the last layers, while MILAN and FALCON often offer lower-quality explanations.

Our method faces limitations with the generative model, particularly in including certain concepts and generating highly abstract ones like "white objects." However, these challenges with abstract concepts also highlight the descriptive quality of the provided explanations. Explanations should be inherently understandable to humans. Addressing these issues may involve analyzing pre-training datasets and employing more specialized models. In future work, we plan to incorporate additional metrics for explanation quality and extend the framework to domains like NLP and healthcare, focusing on evaluating recent autointerpretable methods given by large language models (LLMs) (Kroeger et al., 2023; Bills et al., 2023) and examining high-impact areas such as healthcare datasets.

## Acknowledgements

This work was partly funded by the German Ministry for Education and Research (BMBF) through the project Explaining 4.0 (ref. 01IS200551). Additionally, this work was supported by the European Union's Horizon Europe research and innovation programme (EU Horizon Europe) as grant TEMA (101093003); the European Union's Horizon 2020 research and innovation programme (EU Horizon 2020) as grant iToBoS (965221); and the state of Berlin within the innovation support programme ProFIT (IBB) as grant BerDiBa (10174498).

## Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

## References

Agarwal, C., Krishna, S., Saxena, E., Pawelczyk, M., Johnson, N., Puri, I., Zitnik, M., and Lakkaraju, H. OpenXAI: Towards a Transparent Evaluation of Model Explanations. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=MU2495w47rz.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. Network Dissection: Quantifying Interpretability of Deep Visual Representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Bills, S., Cammarata, N., Mossing, D., Tillman, H., Gao, L., Goh, G., Sutskever, I., Leike, J., Wu, J., and Saunders, W. Language models can explain neurons in language models. https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023.

Borowski, J., Zimmermann, R. S., Schepers, J., Geirhos, R., Wallis, T. S., Bethge, M., and Brendel, W. Natural images are more informative for interpreting cnn activations than state-of-the-art synthetic feature visualizations. In *NeurIPS 2020 Workshop SVRHM*, 2020.

Bykov, K., Deb, M., Grinwald, D., Muller, K. R., and Höhne, M. M. DORA: Exploring Outlier Representations in Deep Neural Networks. *Transactions on Machine Learning Research*, 2023a.

Bykov, K., Kopf, L., Nakajima, S., Kloft, M., and Höhne, M. M. Labeling Neural Representations with Inverse Recognition. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Casper, S., Bu, T., Li, Y., Li, J., Zhang, K., Hariharan, K., and Hadfield-Menell, D. Red teaming deep neural networks with feature synthesis tools. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. *Introduction to algorithms*. MIT press, 2022.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, 2020.

Erhan, D., Bengio, Y., Courville, A., and Vincent, P. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

Fel, T., Boissin, T., Boutin, V., Picard, A. M., Novello, P., Colin, J., Linsley, D., ROUSSEAU, T., Cadene, R., Goetschalckx, L., Gardes, L., and Serre, T. Unlocking Feature Visualization for Deep Network with MAgnitude Constrained Optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=J7VoDuzuKs.

Goh, G., Cammarata, N., Voss, C., Carter, S., Petrov, M., Schubert, L., Radford, A., and Olah, C. Multimodal neurons in artificial neural networks. *Distill*, 6(3):e30, 2021.

Grinwald, D., Bykov, K., Nakajima, S., and Höhne, M. M. Visualizing the Diversity of Representations Learned by Bayesian Neural Networks. *Transactions on Machine Learning Research*, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hedström, A., Weber, L., Krakowczyk, D., Bareeva, D., Motzkus, F., Samek, W., Lapuschkin, S., and Höhne, M. M.-C. Quantus: An Explainable AI Toolkit for Responsible Evaluation of Neural Network Explanations and Beyond. *Journal of Machine Learning Research*, 24(34): 1–11, 2023.

Hedström, A., Weber, L., Lapuschkin, S., and Höhne, M. Sanity Checks Revisited: An Exploration to Repair the Model Parameter Randomisation Test. In *XAI in Action: Past, Present, and Future Applications*, 2023. URL https://openreview.net/forum?id=vVpefYmnsG.

Hernandez, E., Schwettmann, S., Bau, D., Bagashvili, T., Torralba, A., and Andreas, J. Natural Language Descriptions of Deep Features. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=NudBMY-tzDr.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Kalibhat, N., Bhardwaj, S., Bruss, C. B., Firooz, H., Sanjabi, M., and Feizi, S. Identifying Interpretable Subspaces in Image Representations. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 15623–15638. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/kalibhat23a.html.

Kroeger, N., Ley, D., Krishna, S., Agarwal, C., and Lakkaraju, H. Are Large Language Models Post Hoc Explainers? *CoRR*, abs/2310.05797, 2023.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436–444, 2015.

Miller, G. A. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

Molnar, C. *Interpretable Machine Learning*. 2 edition, 2022. URL https://christophm.github.io/interpretable-ml-book.

Mu, J. and Andreas, J. Compositional Explanations of Neurons. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 17153–17163. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c74956ffb38ba48ed6ce977af6727275-Paper.pdf.

Nguyen, A., Dosovitskiy, A., Yosinski, J., Brox, T., and Clune, J. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. *Advances in neural information processing systems*, 29, 2016.

Oikarinen, T. and Weng, T.-W. CLIP-Dissect: Automatic Description of Neuron Representations in Deep Vision Networks. *International Conference on Learning Representations*, 2023.

Olah, C., Mordvintsev, A., and Schubert, L. Feature visualization. *Distill*, 2(11):e7, 2017.

Pernias, P., Rampas, D., Richter, M. L., Pal, C., and Aubreville, M. Würstchen: An Efficient Architecture for Large-Scale Text-to-Image Diffusion Models. In *The Twelfth International Conference on Learning Representations*, 2023.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. In *The Twelfth International Conference on Learning Representations*, 2023.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Samek, W. and Müller, K.-R. Towards explainable artificial intelligence. *Explainable AI: interpreting, explaining and visualizing deep learning*, pp. 5–22, 2019.

Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., and Komatsuzaki, A. LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs. In *NeurIPS Workshop Datacentric AI, online (online), 14 Dec 2021 - 14 Dec 2021*, pp. 5 p., Dec 2021. URL https://juser.fz-juelich.de/record/905696.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: visualising image classification models and saliency maps. In *Proceedings of the International Conference on Learning Representations (ICLR)*. ICLR, 2014.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Xu, F., Uszkoreit, H., Du, Y., Fan, W., Zhao, D., and Zhu, J. Explainable AI: A brief survey on history, research areas, approaches and challenges. In *Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, Part II 8*, pp. 563–574. Springer, 2019.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057. PMLR, 2015.

Yoshimura, N., Maekawa, T., and Hara, T. Toward understanding acceleration-based activity recognition neural networks with activation maximization. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2021.

Yuksekgonul, M., Wang, M., and Zou, J. Post-hoc Concept Bottleneck Models. In *The Eleventh International Conference on Learning Representations*, 2022.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*, 2019.

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

# CoSy: Evaluating Textual Explanations of Neurons

## A. Appendix

### A.1. Concept-based Textual Explanation Methods

Concept-based textual explanation methods aim to provide insights into human-understandable concepts learned by DNNs, enabling a deeper understanding of their decision-making mechanisms. These methods provide textual descriptions for neurons in CV models. This creates a connection between the abstract representation of a concept by the neural network and a human interpretation. In general, a concept can be any abstraction, such as a color, an object, or even an idea (Molnar, 2022). Concept-based textual descriptions of a neuron $f_i$ can originate from various spaces depending on their generation process.

As defined in Section 3, we refer to *explanation method* as an operator $\mathcal{E}$ that maps a neuron to the textual description $s = \mathcal{E}(f_i) \in \mathcal{S}$, where $\mathcal{S}$ is a set of potential textual explanations. The specific set of explanations depends on the implementation of the particular method. We define the following subsets of textual descriptions $s \in \mathcal{S}$:

- $\mathcal{C}$ represents the space of individual concepts,
- $\mathcal{L}$ represents the space of logical combinations of concepts,
- $\mathcal{N}$ represents the space of open-ended natural language concept descriptions.

These textual descriptions serve as explanations for $f_i$ generated by explanation methods.

Examples for such explanation methods are MILAN (Hernandez et al., 2022), FALCON (Kalibhat et al., 2023), CLIP-Dissect (Oikarinen & Weng, 2023), and INVERT (Bykov et al., 2023b). Figure 5 shows the general principle of how $\mathcal{E}$ works. In Table 3 we outline the origin of textual descriptions and their corresponding set memberships for each $\mathcal{E}$.

#### A.1.1. NetDissect

Network Dissection (NetDissect) (Bau et al., 2017) is a method designed to explain individual neurons of DNNs, particularly convolutional neural networks (CNNs) within the domain of CV. This approach systematically analyzes the network's learned concepts by aligning individual neurons with given semantic concepts. To perform this analysis, annotated datasets with segmentation masks are required, where these masks label each pixel in an image with its corresponding object or attribute identity. The Broadly and Densely Labeled Dataset (Broden) (Bau et al., 2017) combines a set of densely labeled image datasets that represent both low-level concepts, such as colors, and higher-level concepts, such as objects. It provides

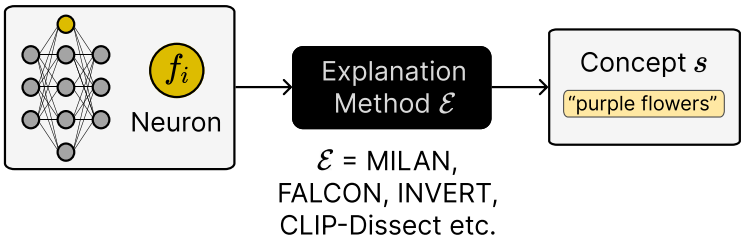


*Figure 5.* Concept-based Textuaĺ Explanation Methods. A neuron $f_i$ is selected, and a concept-based textual explanation method $\mathcal{E}$ is applied to generate a textual description $s$ explaining $f_i$.

*Table 3.* Set Membership and Origin of Descriptions. Generated textual descriptions $s$ have varying set membership and origin across all $\mathcal{E}$. These descriptions can originate from distinct spaces: individual concepts $\mathcal{C}$, logical combinations of concepts $\mathcal{L}$, and open-ended natural language concept descriptions $\mathcal{N}$. A labeled dataset refers to a collection of images paired with individual concept labels. Generated captions are produced by image-to-text models, such as Show-Attend-Tell (Xu et al., 2015). An image caption dataset consists of image-caption pairs. A concept set consists of textual concept labels.

| METHOD | SET | ORIGIN |
|---|---|---|
| NETDISSECT | $\mathcal{C}$ | LABELED DATASET |
| COMPEXP | $\mathcal{L}$ | LABELED DATASET |
| MILAN | $\mathcal{N}$ | GENERATED CAPTION |
| FALCON | $\mathcal{N}$ | IMAGE CAPTION DATASET |
| CLIP-DISSECT | $\mathcal{C}$ | CONCEPT SET |
| INVERT | $\mathcal{L}$ | LABELED DATASET |

a comprehensive set of ground truth examples for a broad range of visual concepts such as objects, scenes, object parts, textures, and materials in a variety of contexts.

A concept $s \in \mathcal{C} \subset \mathcal{S}$ is defined as a visual concept in NetDissect and is provided by the pixel-level annotated Broden dataset. Given a CNN and the Broden dataset as input, NetDissect explains a neural representation $f_i$ by searching for the highest similarity between concept image segmentation masks and neuron activation masks. Concept image segmentation masks are provided by the Broden dataset $B_s(\boldsymbol{x}) \in \{0,1\}^{H \times W}$, where a value of 1 signifies the pixel-level presence of $s$, and 0 denotes its absence. Neuron activation masks are obtained by thresholding the continuous neuron activations of $f_i$ into binary masks $A(\boldsymbol{x}) \in \{0,1\}^{H \times W}$. Then the similarity $\delta_{\text{IoU}}$ between image segmentation masks and binary neuron masks can be evaluated using the Intersection over Union score (IoU) for an individual neuron within a layer:

$$\delta_{\text{IoU}}(f_i, s) = \frac{\sum_{\boldsymbol{x} \in \boldsymbol{X}} \mathbf{1}\left(B_s(\boldsymbol{x}) \cap A(\boldsymbol{x})\right)}{\sum_{\boldsymbol{x} \in \boldsymbol{X}} \mathbf{1}\left(B_s(\boldsymbol{x}) \cup A(\boldsymbol{x})\right)}. \tag{5}$$

The NetDissect method is optimized to identify the concept that yields the highest IoU score between binary masks and image segmentation masks. This can be formalized as:

$$\mathcal{E}_{\text{NetDissect}}(f_i) = \underset{s \in \mathcal{C} \subset \mathcal{S}}{\arg\max}\, \delta_{\text{IoU}}\left(f_i, s\right). \tag{6}$$

NetDissect is constrained to segmentation datasets, relying on pixel-level annotated images with segmentation masks. Moreover, its labeling capabilities are confined to concepts provided within a labeled dataset. Furthermore, only individual concepts can be associated with each neuron.

### A.1.2. COMPEXP

To overcome the limitation of explaining neurons with only a single concept, the Compositional Explanations of Neurons (CompExp) method was later introduced (Mu & Andreas, 2020), enabling the labeling of neurons with compositional concepts. The method obtains its explanations by merging individual concepts into logical formulas using composition operators AND, OR, and NOT. The formula length $L \in \mathbb{N}$ is defined beforehand. The initial stage of explanation generation is similar to NetDissect, a set of images is taken as input, and convolutional neuron activations are converted into binary masks. The explanations are constructed through a beam search algorithm (Cormen et al., 2022), beginning with individual concepts and gradually building them into more complex logical formulas. Throughout the beam search stages, the existing formulas in the beam are combined with new concepts. These new formulas are measured by the IoU. The maximization of the IoU score is desired to get a high explanation quality.

The approach for obtaining $\delta_{\text{IoU}}$ is the same as in Equation 5. In contrast to NetDissect, the explanations can be a combination of concepts, where $s \in \mathcal{L} \subset \mathcal{S}$. The procedure of finding the best neuron description can be formalized as:

$$\mathcal{E}_{\text{CompExp}}(f_i) = \underset{s \in \mathcal{L} \subset \mathcal{S}}{\arg\max}\, \delta_{\text{IoU}}\left(f_i, s\right). \tag{7}$$

Similar to NetDissect, CompExp requires datasets containing segmentation masks and is primarily applicable to convolutional neurons.

### A.1.3. MILAN

MILAN (Hernandez et al., 2022) is a method that aims to describe neural representations within a DNN through open-ended natural language descriptions. First, a dataset of fine-grained human descriptions of image regions (Milannotations) is collected. These descriptions can be defined as concepts that are open-ended natural language descriptions, where $s \in \mathcal{N} \subset \mathcal{S}$. Given a DNN and input images $\boldsymbol{x} \in \boldsymbol{X}$, neuron masks $M(\boldsymbol{x}) \in \mathbb{R}^{H \times W \times C}$ are collected of highly activated image regions for $f_i$.

Two distributions are then derived: the probability $p(s|M(\boldsymbol{x}))$ that a human would describe an image region with $s$, and the probability $p(s)$ that a human would use the description $s$ for any neuron. The probability $p(s|M(\boldsymbol{x}))$ is approximated with the Show-Attend-Tell (Xu et al., 2015) image-to-text model trained on the Milannotations dataset. Additionally, $p(s)$ is approximated with a two-layer LSTM language model (Hochreiter & Schmidhuber, 1997) trained on the Milannotations dataset.

These distributions are then utilized to find a description that has high pointwise mutual information with $M(\boldsymbol{x})$. A hyperparameter $\lambda \in \mathbb{R}$ adjusts the significance of $p(s)$ during the computation of pointwise mutual information (PMI) between descriptions $s$ and $M(\boldsymbol{x})$ sets, where the similarity $\delta_{\text{WPMI}}$ is weighted PMI (WPMI). The objective for WPMI is given by:

$$\delta_{\text{WPMI}}(s) = \log p\left(s|M(\boldsymbol{x})\right) - \lambda \log p(s). \tag{8}$$

MILAN aims to optimize high pointwise mutual information between $s$ and $M(\boldsymbol{x})$ to find the best description for $f_i$:

$$\mathcal{E}_{\text{MILAN}}(f_i) = \underset{s \in \mathcal{N} \subset \mathcal{S}}{\arg\max}\, \delta_{\text{WPMI}}\left(f_i, s\right). \tag{9}$$

The requirement of collecting the curated labeled dataset, Milannotations, limits MILAN's capabilities when applied to tasks beyond this specific dataset. Additionally, another drawback is the requirement for model training.

### A.1.4. FALCON

The FALCON (Kalibhat et al., 2023) explainability method has a similar approach to MILAN. Initially, it gathers the most highly activating images corresponding to a neural representation. GradCam (Selvaraju et al., 2017) is subsequently applied to identify highlighted features in these images, which are then cropped to focus on these regions. These cropped images, along with large captioning dataset LAION-400m (Schuhmann et al., 2021) with concepts $s \in \mathcal{N} \subset \mathcal{S}$, are input to CLIP (Contrastive Language-Image Pre-training) (Radford et al., 2021), which computes the image-text similarity between the text embeddings of captions and the input cropped images. The top 5 captions are then extracted. Conversely, the least activating images are collected, and concepts are extracted and removed from the top-scoring concepts, ultimately yielding the explanation of the neural representation.

The similarity $\delta_{\text{CLIPScore}}$ is obtained by calculating the CLIP confidence matrix, which is essentially a cosine similarity matrix. The aim is to find the maximum image-text similarity score between image embeddings and their closest text embeddings from a large captioning dataset:

$$\mathcal{E}_{\text{FALCON}}(f_i) = \underset{s \in \mathcal{N} \subset \mathcal{S}}{\arg\max}\, \delta_{\text{CLIPScore}}\left(f_i, s\right). \tag{10}$$

This restriction significantly narrows down the range of models suitable for analysis, setting it apart considerably from other explanation methods.

**FALCON Implementation** In its original implementation, FALCON restricts the set of "explainable neurons" based on specific parameters. These include the parameter $\alpha \in \mathbb{N}$, which determines the set of highly activating images for a given feature by requiring $\alpha > 10$. Additionally, it employs a threshold $\gamma \in \mathbb{R}$ for CLIP cosine similarity, with a set value of $\gamma > 0.8$.

These parameter settings significantly restrict the number of explainable neurons, resulting to fewer than 50 explainable neurons. This constraint prevents the necessary randomization for comparison with other methods. To address this, we set $\alpha = 0$ and $\gamma = 0$. However, for FALCON-original, we retain the original settings of $\alpha$ and $\gamma$ and calculate $\Psi$ across all "explainable neurons." In our experiments on `ResNet18`, FALCON can only be applied to layers 2 to 4.

### A.1.5. CLIP-DISSECT

CLIP-Dissect (Oikarinen & Weng, 2023) is an explanation method that describes neurons in vision DNNs with open-ended concepts, eliminating the need for labeled data or human examples. This method integrates CLIP (Radford et al., 2021), which efficiently learns deep visual representations from natural language supervision. It utilizes both the image encoder and text encoder components of a CLIP model to compute the text embedding for each concept $s \in \mathcal{C} \subset \mathcal{S}$ from a concept dataset and the image embeddings for the probing images in the dataset, subsequently calculating a concept-activation matrix.

The activations of a target neuron $f_i$ are then computed across all images in the probing dataset $\boldsymbol{X}$. However, as this process is designed for scalar neural representations, these activations are summarized by a function that calculates the mean of the activation map over spatial dimensions. The concept corresponding to the target neuron is determined by identifying the most similar concept $s$ based on its activation vector. The most highly activated images are denoted as $\boldsymbol{X}_s \subset \boldsymbol{X}$.

SoftWPMI is a generalization of WPMI where the probability $p\left(\boldsymbol{x} \in \boldsymbol{X}_s\right)$ denotes the chance an image $\boldsymbol{x}$ belongs to the example set $\boldsymbol{X}_s$. Standard WPMI corresponds to cases where $p(\boldsymbol{x} \in \boldsymbol{X}_s)$ is either 0 or 1 for all $\boldsymbol{x} \in \boldsymbol{X}$, while SoftWPMI relaxes this binary setting to real values between 0 and 1. The function can be formalized as:

$$\delta_{\text{SoftPMI}}(s) = \log \mathbb{E}\left[p\left(s|\boldsymbol{X}_s\right)\right] - \lambda \log p(s). \tag{11}$$

The similarity function $\delta_{\text{SoftWPMI}}$ aims to identify the highest pointwise mutual information between the most highly activated images $\boldsymbol{X}_s$ and a concept $s$. This optimization search is expressed as:

$$\mathcal{E}_{\text{CLIP-Dissect}}(f_i) = \underset{s \in \mathcal{C} \subset \mathcal{S}}{\arg\max}\, \delta_{\text{SoftWPMI}}\left(f_i, s\right). \tag{12}$$

A drawback of CLIP-Dissect lies in its interpretability; descriptions are generated by the CLIP model, which itself is challenging to interpret.

### A.1.6. INVERT

Labeling Neural Representations with Inverse Recognition (INVERT) (Bykov et al., 2023b) shares the capability of constructing complex explanations like CompExp (Mu & Andreas, 2020) but with the added advantage of not relying on segmentation masks and only needing labeled data. The method obtains its explanations by merging individual concepts into logical formulas using composition operators AND, OR, and NOT. It also exhibits greater versatility in handling various neuron types and is computationally less demanding compared to previous methods such as NetDissect (Bau et al., 2017) and CompExp (Mu & Andreas, 2020). Additionally, INVERT introduces a transparent metric for assessing the alignment between representations and their associated explanations. The non-parametric Area Under the Receiver Operating Characteristic (AUC) measure evaluates the relationship between representations and concepts based on the representation's ability to distinguish the presence from the absence of a concept, with statistical significance. The probing dataset with the concept present is labeled as $\boldsymbol{X}_1$, while the dataset without the concept is labeled as $\boldsymbol{X}_0$.

The goal of INVERT is to identify the concept $s \in \mathcal{L} \subset S$ that maximizes $\delta_{\text{AUC}}$ with the neural representation $f_i$. Here, $s$ can be a combination of concepts. The optimization process resembles that of CompExp, employing beam search (Cormen et al., 2022) to find the optimal compositional concept. The top-performing concepts are iteratively selected until the predefined compositional length $L \in \mathbb{N}$ is reached.

The similarity measure $\delta_{\text{AUC}}$ is defined as:

$$\delta_{\text{AUC}}(f_i) = \frac{\sum_{\boldsymbol{x}_0 \in \boldsymbol{X}_0} \sum_{\boldsymbol{x}_1 \in \boldsymbol{X}_1} \mathbf{1}[f_i(\boldsymbol{x}_0) < f_i(\boldsymbol{x}_1)]}{|\boldsymbol{X}_0| \cdot |\boldsymbol{X}_1|}. \tag{13}$$

*Table 4.* Comparison of characteristics of concept-based textual explanation methods. The columns (from left to right) represent the explanation method used, its textual output type (fixed-label, compositional, or open-vocabulary), the type of neuron targeted for analysis (convolutional, scalar, or predetermined), the target metric the method optimizes (IoU, WPMI, AUC, etc.), whether the method relies on auxiliary black-box models for finding or generating explanations (img2txt, CLIP), and whether the explanation method is architecture-agnostic, meaning it can be applied to any CV model.

| METHOD | EXPLANATION | NEURON TYPE | TARGET | BLACK-BOX DEPENDENCY | ARCHITECTURE-AGNOSTIC |
|---|---|---|---|---|---|
| NETDISSECT (BAU ET AL., 2017) | FIXED-LABEL | CONV. | IOU | — | ✓ |
| COMPEXP (MU & ANDREAS, 2020) | COMPOSITIONAL | CONV. | IOU | — | ✓ |
| MILAN (HERNANDEZ ET AL., 2022) | OPEN-VOCABULARY | CONV. | WPMI | IMG2TXT MODEL | ✓ |
| INVERT (BYKOV ET AL., 2023B) | COMPOSITIONAL | SCALAR | AUC | — | ✓ |
| CLIP-DISSECT (OIKARINEN & WENG, 2023) | OPEN-VOCABULARY | SCALAR | SOFTWPMI | CLIP | ✓ |
| FALCON (KALIBHAT ET AL., 2023) | OPEN-VOCABULARY | PREDETERMINED | AVG. CLIP SCORE | CLIP | — |

The objective of INVERT is to maximize the similarity $\delta_{\text{AUC}}$ between a concept $s$ and the neural representation $f_i$, which can be described as:

$$\mathcal{E}_{\text{INVERT}}(f_i) = \arg\max_{s \in \mathcal{L} \subset \mathcal{S}} \delta_{\text{AUC}}\left(f_i, s\right). \tag{14}$$

INVERT is constrained by the requirement of a labeled dataset and is computationally more expensive compared to CLIP-Dissect.

#### A.1.7. COMPARISON OF CHARACTERISTICS OF CONCEPT-BASED TEXTUAL EXPLANATION METHODS

Table 4 illustrates an overview of the different characteristics of concept-based textual explanation methods.

### A.2. Schematic Illustration of COSY Implementation Details

In the example shown in Figure 1, we used the default settings of the explanation methods to generate explanations for neuron 80 in the avgpool layer of `ResNet18`. For CLIP-Dissect, we used the 20,000 most common English words as the concept dataset and the ImageNet validation dataset (Russakovsky et al., 2015) as the probing dataset . We employed Stable Diffusion XL 1.0-base (`SDXL`) (Podell et al., 2023) as the text-to-image model, using the prompt "realistic photo of a close up of `[concept]`" to generate concept images, with `[concept]` being replaced by the textual explanation from the methods. We generated 50 images per concept for 50 randomly chosen neurons from the avgpool layer of `ResNet18`. For evaluation, we also used the ImageNet validation dataset as the control dataset.

### A.3. Prompt and Text-to-Image Model Comparison

One of the key features of COSY is its reliance on generative models to translate textual explanations of neurons into the visual domain. Thus, it is essential that the generated images reliably resemble the textual concepts. In the following section, we present an experiment where we varied several parameters of the generation procedure and evaluated the visual similarity between generated images and synthetic ones, focusing on concepts for which we have a collection of natural images.

For our analysis, we used only open-source and freely available text-to-image models, namely Stable Diffusion XL 1.0-base (`SDXL`) (Podell et al., 2023) and Stable Cascade (`SC`) (Pernias et al., 2023). We also varied the prompts for image generation. To measure the similarity between synthetic images and natural images corresponding to the same concept, we employed cosine similarity (*CS*) in the CLIP embedding space with the CLIP-ViT-B/32 model (Radford et al., 2021). We select a set of 10 random concepts from the 1,000 classes in the ImageNet validation dataset (Russakovsky et al., 2015). For each `[concept]` we use 5 different prompts and employ them with `SDXL` and `SC` models, generating 50 images per concept. We then measure the *CS* between image pairs of the same class.

### A.4. Sanity Check Setup

Experimental setup for comparing true and random explanations on output neurons with known ground truth labels involved employing four different models pre-trained on ImageNet: `ResNet18` (He et al., 2016), `DenseNet161` (Huang et al., 2017), `GoogleNet` (Szegedy et al., 2015), and `ViT-B/16` (Dosovitskiy et al., 2020). For each model, we randomly

*Table 5.* Comparison of true and random explanations on output neurons with known ground truth labels. This table presents the quality estimates for true explanations, derived from target class labels, and random explanations, derived from randomly selected synthetic image classes (excluding the target class), across four models pre-trained on ImageNet. Higher values are better. Our results consistently show high scores for true explanations and low scores for random ones.

| Model | AUC (↑) | | MAD (↑) | |
|---|---|---|---|---|
| | *True* | *Random* | *True* | *Random* |
| ResNet18 | 0.94±0.20 | 0.54±0.23 | 5.93±2.35 | 0.17±1.90 |
| DenseNet161 | 0.95±0.17 | 0.60±0.23 | 6.81±2.50 | 0.56±1.56 |
| GoogLeNet | 0.95±0.16 | 0.41±0.24 | 7.44±2.77 | -0.31±0.90 |
| ViT-B/16 | 0.97±0.11 | 0.59±0.23 | 13.51±5.56 | 0.17±0.48 |

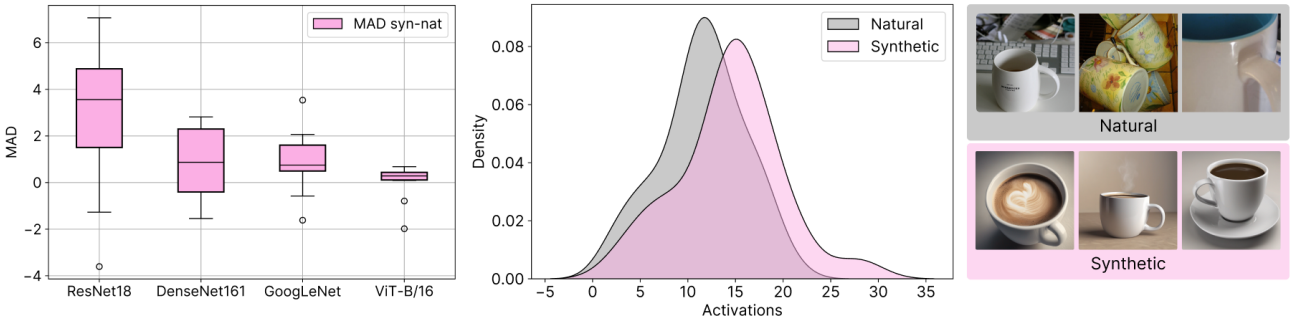


*Figure 6.* An overview of analyses performed to study the similarity between natural and synthetic images. From left to right: (1) an overview of MAD scores between synthetic and natural image activations of the output neuron's ground truth classes for each model studied in this work, (2) activations collected for neuron 504 in `ResNet18` for the class "coffee mug," showcasing the difference between the natural and synthetic distributions and (3) examples of natural versus synthetic images. In both analyses, we observe a substantial overlap in the activations of synthetic and natural images, suggesting that the models respond similarly to both types of images.

selected 10 output classes and generated 50 images per class using class descriptions. We then passed both true and randomly generated explanation images through the models, alongside the ImageNet validation dataset (Russakovsky et al., 2015) as a control dataset. We collected activations of the output neurons corresponding to each class. Next, we compared the CoSy scores of the ground truth explanations, indicated by the neuron labels, with those of randomly selected explanations. In addition to the results presented in Table 1, we also performed the same experiment excluding the ground truth images from the control dataset; these results are shown in Table 5.

### A.5. Do Models Respond Differently to Synthetic and Natural Images?

Given the visual similarity between natural and synthetic images of the same concepts, we investigate whether CV models respond differently to these groups and if the activation differences indicate adversarial behavior. Following the experimental setup in Appendix A.4, we selected 10 output neurons for each model. We passed both synthetic and natural images through the models and collected the activations of the output neurons corresponding to each class.

Figure 6 (left) illustrates the distributions of the MAD between synthetic and natural images for the same class across the 10 classes. Generally, we observe that the activation of synthetic images is slightly higher than that of natural images of the same class. However, this difference is small, given the 0 value lies within 1 standard deviation. We also illustrated (Figure 6, right) the activations of neuron 504 in the `ResNet18` output layer, corresponding to the "coffee mug" class. The results indicate a strong overlap in the neural response to both synthetic and natural images. While synthetic images activate the neuron slightly more, this does not constitute an artifactual behavior or affect our framework, which we demonstrate in the following experiment.

*Table 6.* Intraclass Image Similarity. This table illustrates the impact of varying parameters within our COSY framework. We evaluate 5 different prompts as input to 2 different text-to-image models. A random selection of 10 ImageNet classes is made, with each class name used as input to the prompt, denoted as `[concept]`, resulting in 50 images generated per prompt using a text-to-image model. We compute the average intraclass similarity across all classes. Higher *CS* and lower *ED* values indicate greater similarity between the images. In intraclass image similarity, neither excessively high nor excessively low scores are desirable.

| PROMPT | TEXT-TO-IMAGE | *CS* (↑) | *ED* (↓) |
|---|---|---|---|
| 1. "A [CONCEPT]" | SDXL | 0.83±0.07 | 5.85±1.41 |
| | SC | 0.92±0.03 | 4.03±1.00 |
| 2. "A PAINTING OF [CONCEPT]" | SDXL | 0.87±0.05 | 4.94±1.13 |
| | SC | 0.92±0.03 | 3.95±0.88 |
| 3. "PHOTO OF [CONCEPT]" | SDXL | 0.81±0.07 | 6.13±1.36 |
| | SC | 0.90±0.04 | 4.46±1.05 |
| 4. "REALISTIC PHOTO OF [CONCEPT]" | SDXL | 0.86±0.06 | 5.41±1.34 |
| | SC | 0.93±0.03 | 3.79±0.85 |
| 5. "REALISTIC PHOTO OF A CLOSE UP OF [CONCEPT]" | SDXL | 0.88±0.05 | 5.09±1.29 |
| | SC | 0.93±0.03 | 3.95±0.92 |

### A.6. Intraclass Image Similarity

In addition to comparing natural and synthetic images as in Section 4.1, we also analyze the intraclass distance to compare the similarity among synthetic images. Intraclass distance refers to the degree of diversity or dissimilarity observed within a set of images of the same class. It quantifies how much the individual images deviate from the average or central tendency of the image set. In this context, intraclass distance is desirable, reflecting how visual concepts can appear in natural images. Higher similarity scores indicate greater divergence from natural occurrences of concepts.

Cosine similarity (*CS*) and "Euclidean distance" (*ED*) are commonly used metrics for measuring image similarity because they capture different aspects of similarity and complement each other. We compute the average *CS* and *ED* for each class and determine the overall class average. Table 6 provides a detailed overview of the results quantifying the similarity within synthetic images using *CS* and *ED*. When evaluating these results, it is important to note that high scores do not necessarily indicate optimal outcomes, as they suggest nearly identical images, which may lack intraclass distance. Conversely, very low scores imply significant differences among images, which might not capture the essence of the concept adequately. Ideally, we aim for somewhat similar yet slightly varied images representing the same class. The results show that the Stable Cascade (`SC`) model consistently achieves higher scores across all prompts compared to the Stable Diffusion XL 1.0-base (`SDXL`) model. Notably, it obtains the highest score for the two most elaborate prompts (4, 5). This indicates that the `SC` model tends to offer less intraclass distance in visually representing concepts.

### A.7. Model Stability

In this experiment, our goal is to evaluate the stability of the image generation method employed, aiming to ensure consistent results within our COSY framework. We achieve this by varying the seed of the image generator and observing the impact on image generation. We anticipate consistent image representations across different model initializations, thus ensuring the stability of our framework.

For our analysis, we utilize `ResNet18` and focus on its output neurons, as the ground-truth labels associated with these neurons are known. We randomly select six classes $s$ from the ImageNet validation dataset (Russakovsky et al., 2015) and examine the corresponding $f_i$ class output neurons using COSY. Here, we exclude the $s$ class from $\mathbb{A}_0$ and let $\mathbb{A}_1$ represent the $s$ class. To ensure robustness, we initialize the text-to-image model across a random set of 10 seeds. Our analysis involves calculating the first (mean) and second moment (STD) using $\Psi_{\text{AUC}}$, as well as evaluating the intraclass image similarity (refer to Section A.6) within each synthetic ground truth class.

The results for our experiment, as shown in Table 7, demonstrate remarkably high AUC scores, indicating near-perfect detection of synthetic ground truth classes across all image model initializations. Furthermore, the standard deviation is exceptionally low, suggesting consistent image generation regardless of the chosen seed. The intraclass similarity values indicate a certain degree of distance in the generated images, indicating high similarity yet distinctiveness. This intraclass distance is desirable, ensuring that the images are not identical but share common characteristics.

*Table 7.* Model Stability. A comparison of various model initializations across 10 random seeds using `SDXL`.

| Concept | AUC (↑) | *CS* (↑) | *ED* (↓) |
|---|---|---|---|
| Bulbul | 0.9996±0.0002 | 0.91±0.03 | 3.99±0.66 |
| China cabinet | 0.9999±0.0001 | 0.89±0.04 | 5.00±0.90 |
| Leatherback turtle | 0.9994±0.0001 | 0.91±0.04 | 4.65±0.87 |
| Beer bottle | 0.9919±0.0038 | 0.80±0.08 | 6.79±1.41 |
| Half track | 0.9998±0.0000 | 0.88±0.04 | 5.12±0.91 |
| Hard disc | 1.0000±0.0001 | 0.90±0.05 | 4.64±1.17 |
| **Overall Mean** | **0.9984±0.0007** | **0.88±0.02** | **5.03±0.26** |

These findings underscore the reliability and consistency of our image generation pipeline within our CoSy framework. The high stability of text-to-image generation across different seeds and the diversity of image similarity contribute to the robustness of our approach.

### A.8. Benchmarking Implementation

For generating explanations with the explanation methods, we use a subset of 50,000 images from the training dataset on which the models were trained. For evaluation with CoSy, we use the corresponding validation datasets the models were pre-trained on as the control dataset. Additionally, for CLIP-Dissect, we define concept labels by combining the 20,000 most common English words with the corresponding dataset labels. For INVERT we set the compositional length of the explanation as $L = 1$, where $L \in \mathbb{N}$.

### A.9. Compute Resources

For running the task of image generation for CoSy we use distributed inference across multiple GPUs with PyTorch Distributed, enabling image generation with multiple prompts in parallel. We run our script on three Tesla V100S-PCIE-32GB GPUs in an internal cluster. Generating 50 images for 3 prompts in parallel takes approximately 12 minutes.

### A.10. Additional Results for Method Comparison across ResNet18 Layers

In addition to the AUC results for various methods across different `ResNet18` layers, as discussed in Section 5.2, we also report the MAD scores. This provides a more comprehensive evaluation of each method's performance. As shown in Figure 7, the MAD scores resemble the AUC results but are higher in the upper layers.

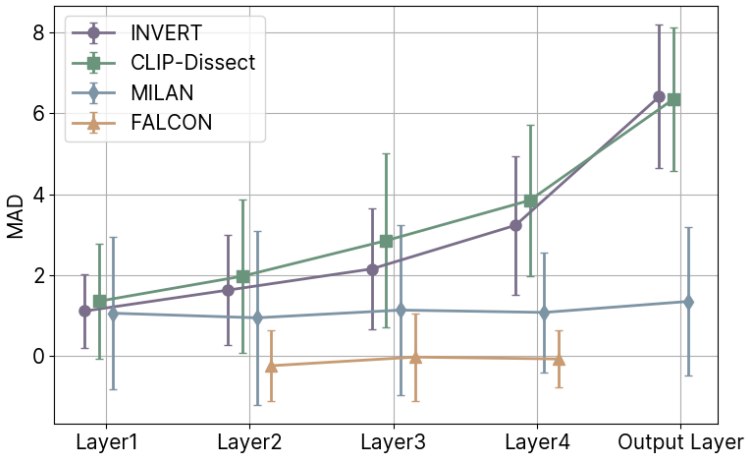


*Figure 7.* A comparison of explanation methods in `ResNet18` shows that INVERT and CLIP-Dissect maintain high MAD scores across all layers, while MILAN, and FALCON have lower scores. Overall, performance declines in the lower layers for all methods.

Given that the original implementation of FALCON only provides results for their defined "explainable neurons" (see Appendix A.1.4), we included additional results comparing all methods based on this subset of neurons. Specifically, there are 7 explainable neurons in layer 2, 5 in layer 3, and 15 in layer 4. Figure 8 presents these results.

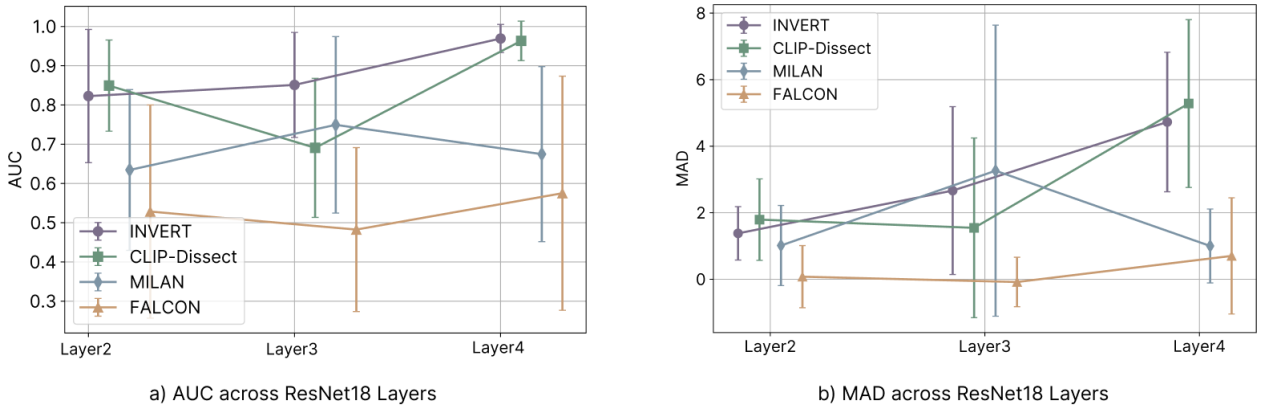


*Figure 8.* A comparison of explanation methods in `ResNet18` shows that INVERT and CLIP-Dissect maintain high MAD scores across all layers, while MILAN, and FALCON have lower scores. Overall, performance declines in the lower layers for all methods.

### A.11. Concept Broadness

While COSY focuses on measuring the explanation quality, another open question is how broad or abstract are the concepts provided as textual explanations. This question of how specific or general an individual neuron is described by the explanation, might be relevant to different XAI applications. For example, research fields where the user aims to deploy the same network for multiple tasks with varying image domains. In this case, describing a neuron's more general concept such as "a round object" might be more informative than a more (domain-)specific concept such as "a tennis ball" for the network assessment. In an effort to provide insight on the broadness of concepts, we assessed whether the similarity between images generated based on the same concept changes for more general to more specific concepts.

In our experiment, we define the broadness of a concept based on the number of hypernyms in the WordNet hierarchy (Miller, 1995). The more specific a concept the larger the number of hypernyms. We choose two ImageNet classes ("ladybug," "pug") and generate 50 images for each concept as well as each hypernym of both concepts (with the most general concept being "entity"). Then, we measure the cosine similarity of all images generated based on the same concept. The box plot of the cosine similarity across both concepts and all hypernyms, in Figure 9 indicates that we do not find a correlation. Thus, we hypothesize that the chosen temperature of the diffusion model has a stronger effect on image similarity than the broadness of the prompt used for image generation.

### A.12. Synthetic Image Examples with Different Prompts and Text-to-Image Models

Figure 10 showcases additional examples of synthetically generated images using both `SDXL` and `SC` across various prompts, highlighting the diversity and accuracy of concept representation.

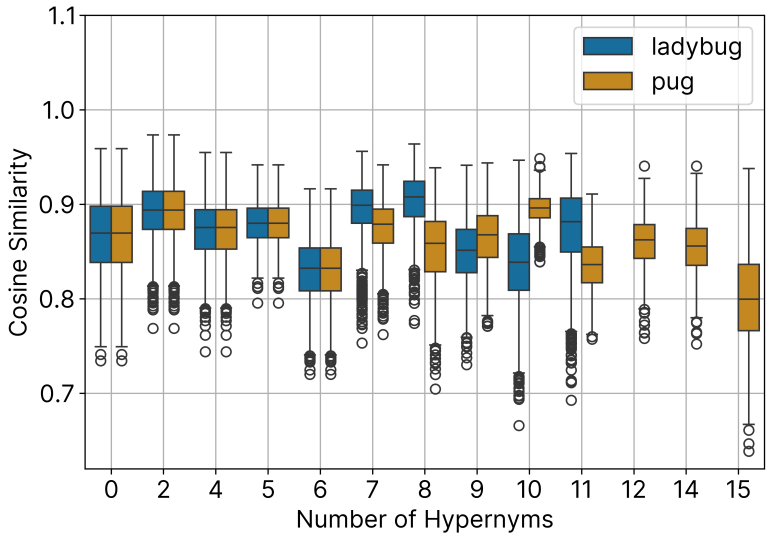


*Figure 9.* The figure demonstrates the independence of the concept broadness measured by the number of hypernyms as defined in WordNet (Miller, 1995) to the inter-image similarity of corresponding generated images.

*Figure 10.* Example images for "coffee mug" generated by the text-to-image models SDXL and SC across various prompts. (1) and (3) present examples of synthetic images with relatively low intraclass similarity and relatively high natural-to-synthetic similarity scores. (2) shows examples of synthetic images with the lowest similarity to natural images. (4) illustrates examples of synthetic images with the highest similarity to other synthetic images within the same class. (5) showcases examples of synthetic images with the highest similarity to natural images. (6) displays examples of natural images from the ImageNet validation dataset (Russakovsky et al., 2015) belonging to the class "coffee mug."