Rethinking Crowd-Sourced Evaluation of Neuron Explanations

Tuomas Oikarinen
UC San Diego, CSE
toikarinen@ucsd.edu

Ge Yan
UC San Diego, CSE
geyan@ucsd.edu

Akshay Kulkarni
UC San Diego, CSE
a2kulkarni@ucsd.edu

Tsui-Wei Weng
UC San Diego, HDSI
lweng@ucsd.edu

Abstract

Interpreting individual neurons or directions in activation space is important for mechanistic interpretability. Numerous automated interpretability methods have been proposed to generate such explanations, but it remains unclear how reliable these explanations are, and which methods produce the most accurate descriptions. While crowd-sourced evaluations are commonly used, existing pipelines are noisy, costly, and typically assess only the highest-activating inputs, leading to unreliable results. In this paper, we introduce two techniques to enable cost-effective and accurate crowdsourced evaluation of automated interpretability methods beyond top activating inputs. First, we propose Model-Guided Importance Sampling (MG-IS) to select the most informative inputs to show human raters. In our experiments, we show this reduces the number of inputs needed to reach the same evaluation accuracy by $\sim 13\times$. Second, we address label noise in crowd-sourced ratings through Bayesian Rating Aggregation (BRAgg), which allows us to reduce the number of ratings per input required to overcome noise by $\sim 3\times$. Together, these techniques reduce the evaluation cost by $\sim 40\times$, making large-scale evaluation feasible. Finally, we use our methods to conduct a large scale crowd-sourced study comparing recent automated interpretability methods for vision networks.

1 Introduction

Despite their transformative capabilities, deep learning models remain fundamentally opaque, which limits their reliability and trustworthiness, especially in high-stakes domains. To address this, the field of mechanistic interpretability aiming to understand the mechanisms inside neural networks has grown rapidly. A key part of mechanistic interpretability is understanding small components of neural networks such as individual neurons or latents in a sparse autoencoder (SAE) through text-based descriptions, which can be achieved via automated neuron descriptions [3, 19, 14, 20, 21, 7, 17, 26, 1].

To assess the quality of these neuron explanations, researchers often rely on human studies, such as crowd-sourced evaluations via platforms like Amazon Mechanical Turk (AMT). However, despite their importance, the design and analysis of these studies has received relatively little attention. Most existing evaluations examine only whether an explanation matches the top activating inputs of a neuron. As identified by [22], this corresponds to measuring only *Recall*, and ignores many important factors such as whether the explanation also describes lower activations of that neuron, or whether all inputs matching the explanation actually activate the neuron (*Precision*). This can lead to misleading conclusions and fails to reliably assess explanation quality.

Motivated by the above limitations, we aim to conduct a crowdsourced evaluation using a more principled evaluation metric: the *correlation coefficient* between neuron activations and concept (explanation) presence labels. We choose correlation as it was identified as the most reliable metric by [22] – it captures both sensitivity and specificity and it doesn't require arbitrary binarization of real-valued neuron activations. However, using correlation as the evaluation metric introduces two challenges that directly drive up total human study cost:
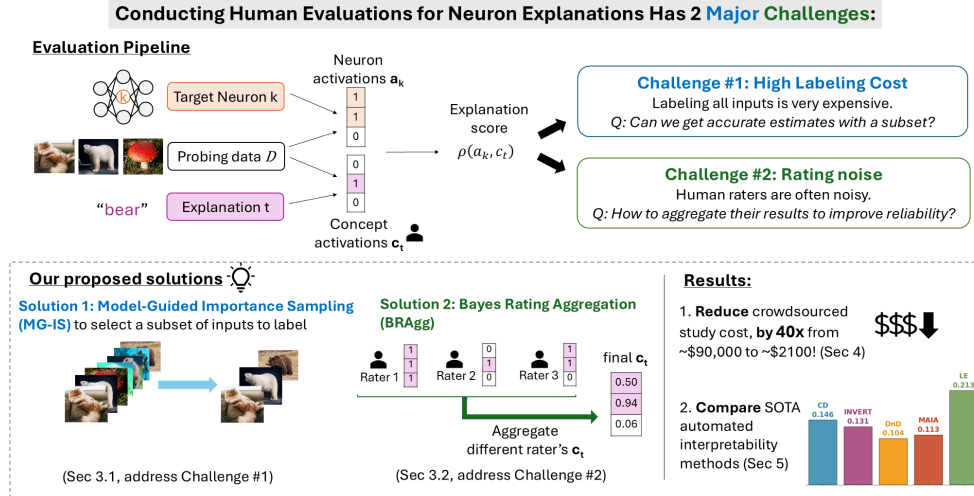
Figure 1: Overview of the explanation evaluation pipeline. We focus on two main challenges: 1: How to reduce high labeling cost? and 2: How to effectively handle noisy ratings?. Our proposed solutions are discussed in Section 3, validated in Section 4, and applied to compare automated interpretability methods in Section 5.

- **Challenge 1: High Labeling Cost**. Computing the correlation requires concept presence labels across the *entire* probing dataset. Labeling all inputs of a dataset with 50,000 images for a single (neuron, explanation) pair with three raters each costs roughly $600, and evaluating a few thousands of neurons in a single model can reach $1M.

- **Challenge 2: Rater Noise**. Human concept judgments are noisy, and even small error rates can significantly affect correlation scores, particularly for rare concepts. Increasing the number of raters per input improves reliability, but further multiplies overall cost.

In other words, *reliable, full-distribution human evaluation of neuron explanations is currently too expensive to perform at scale*. To address this, we introduce two novel techniques to enable cost-efficient and reliable crowd-sourced evaluation of automated interpretability. In summary, our contributions are:

- **1. Model-Guided Importance Sampling (MG-IS)** *for Efficient Input Selection*: We develop an importance sampling strategy that selects the most informative inputs to show raters, reducing labeling cost by around 13× while preserving correlation estimation accuracy;

- **2. Bayes Rating Aggregation (BRAgg)** *for Robust Rater Modeling*: We propose a Bayesian rater uncertainty model that reduces the number of ratings required for stable correlation estimates by an additional 3×, which equivalently reduce the labeling cost by 3×;

- **3. Systematic Comparison of Automated Interpretability Methods:** Together, our proposed MG-IS and BRAgg reduce total evaluation cost by around **40×**, taking the total costs of a systematic comparison down from around $90,000$ to 2160. Our work enables a large-scale, systematic comparison of leading neuron explanation methods across multiple vision models. Our findings demonstrate that Linear Explanations [21] overall produces the best vision neuron explanations, outperforming for example recent LLM-based explanations [1, 26].

2 Related Work

2.1 Automated Interpretability Methods for Deep Vision Models

Methods based on labeled concept datasets: Early work on automated interpretability sought to match neuron activations to human-interpretable concept annotations. Network Dissection (ND) [2] is the first automated method to generate text descriptions for individual neurons in vision models.

They introduced the Broden dataset with dense pixel-wise annotations and proposed identifying concepts with high Intersection over Union (IoU) with binarized neuron activations. However, their approach requires dense concept annotated data which is expensive to collect. Compositional Explanations [19] extends Network Dissection to deal with polysemantic neurons, which may activate on multiple unrelated concepts, by searching for logical compositions of concepts such as "cat OR dog". Clustered Compositional Explanations(CCE)[17] extends this even further, addressing the problem that Compositional Explanations only explains the highest activations of a neuron. CCE divides the activation range of a neuron into 5 buckets, and generates a separate compositional explanation for each. [3] propose a version of Network Dissection utilizing a segmentation model instead of labeled data. INVERT[7] proposes explaining neurons with logical compositions of dataset class(or superclass) labels that maximize AUC, which removes the reliance on Broden but still requires labeled data.

Methods based on Language Models: Another popular line of work leverages generative language models to generate natural-language neuron descriptions. The first such method was MILAN [14], which trains a small generative neural network to describe the most highly activating inputs of a neuron. Describe-and-Dissect (DnD) [1] is a more recent method that utilizes pre-trained language models instead of training their own to generate detailed neuron descriptions based on highest activating inputs. Multimodal Automated Interpretability Agent (MAIA)[26] is another LLM based explanation pipeline where the explaining LLM agent can interact with many tools such as looking at highly activating inputs, generating new images or editing existing ones to generate its description.

CLIP-based methods: Finally, papers such as CLIP-Dissect(CD) [20] and FALCON [16] have proposed methods that don't require labeled concept information by relying on supervision from multimodal models such as CLIP [24]. Linear Explanations (LE) [21] proposes to explain polysemantic neurons as a linear combination of concepts, such as "$3 \times \text{dog} + 2 \times \text{cat}$". To obtain these explanations, the authors utilize either the class labels from the dataset which gives the explanations denoted as LE(label), or pseudo-labels from SigLIP [27] gives gives the explanation denoted as LE(SigLIP).

2.2 Human Evaluations of Automated Interpretability Methods

Crowd-sourced human evaluations are commonly used to assess the quality of automatic neuron explanations in previous works [2, 20, 1]. These evaluations typically present raters the top-activating inputs for a neuron along with one or multiple candidate descriptions, and ask them to rate how well the description matches the top-activating inputs. However, as pointed out by NeuronEval [22], this evaluation protocol is flawed and fails to measure whether the description matches lower activations of the neuron, or whether all inputs corresponding to description cause the neuron to fire, and in fact corresponds to only measuring *Recall*. As a result, this evaluation approach can favor explanations that are overly broad or non-specific.

On the other hand, in [5] and [29], the authors conducted more objective crowd-sourced evaluations of neuron interpretability by having raters predict whether an input is highly activating or not based on a feature visualization or a set of highly activating images for that neuron. However, these studies did not study text descriptions of neurons, and their methodology, which is roughly equivalent to measuring AUC may also be biased towards overly generic neuron descriptions and does not pass the sanity checks proposed by [22].

NeuronEval [22] formalizes the task of Neuron evaluation under a unified mathematical framework which we utilize in this paper, and identifies flawed and good evaluation metrics through sanity tests. However, they do not conduct actual user studies using an evaluation metric that pass their sanity checks, which is the goal of this paper.

3 Methods

In this section, we introduce two techniques to enable cost-efficient and reliable crowdsourced evaluation of neuron explanations. Section 3.1 introduces Model-Guided Importance Sampling (MG-IS), which addresses the labeling cost challenge by selecting the most informative inputs to label. Section 3.2 introduces Bayes Rating Aggregation (BRAgg), which addresses the rater noise challenge by modeling rater errors to robustly aggregate binary judgments. Together, MG-IS and

BRAgg substantially reduce the number of labels and ratings required for stable correlation estimates, making crowdsourced evaluation feasible in practice.

Problem Formulation. Given a neural network f and a neuron of interest k (or a direction in activation space), we can obtain a neuron explanation t (described by text) either manually or via automated interpretability methods, that aims to describe the role of that neuron. To measure how good this explanation is, we follow the framework introduced by [22], and use the Pearson correlation between the neuron activation vector a_k and the explanation concept presence vector c_t (see Fig. 1 for an overview of the evaluation pipeline). A high correlation coefficient means the concept is present on inputs where the neuron activates highly, and not present when neuron activates lowly.

Here, $a_k \in \mathbb{R}^{|\mathcal{D}|}$ is the activation vector of a neuron k over all the probing images $x_i \in \mathcal{D}$ and $c_t \in \mathbb{R}^{|\mathcal{D}|}$ is the concept vector to indicate if concept t appears on the probing images. The i-th component $[a_k]_i$ and $[c_t]_i$ can be formally expressed as: $[a_k]_i = f_k^{0:l}(x_i), \quad [c_t]_i = \mathbb{P}(t|x_i), \quad \forall\ i \in \{|\mathcal{D}|\}$ where $f^{0:l}$ denotes the neural network up to l-th layer. Our goal is to evaluate Pearson's correlation coefficient ρ, which can be expressed as:

$$\rho(a_k, c_t) = \frac{1}{|\mathcal{D}|} \frac{\sum_{i \in \mathcal{D}}([a_k]_i - \mu(a_k)) \cdot ([c_t]_i - \mu(c_t))}{\sigma(a_k)\sigma(c_t)} \tag{1}$$

where $\mu(a_k), \sigma(a_k), \mu(c_t), \sigma(c_t)$ are the mean and standard deviation of the vectors a_k and c_t.

3.1 Model-Guided Importance Sampling (MG-IS)

The activation vector a_k can be easily evaluated with a forward pass of the neural network. However, obtaining a high quality concept vector c_t can be expensive and difficult, requiring human raters or computationally expensive models. Our problem then becomes: *How can we best select a subset of inputs $S \subseteq \{|D|\}$ to label, such that we can accurately estimate $\rho(a_k, c_t)$ using just the subset?*

The simplest approach is to select input for annotation using uniform (Monte Carlo) sampling, where each input has an equal probability of being selected. However, concepts of interest could be rare in the probing dataset, meaning that a uniform sample is unlikely to contain sufficient amount (or any) of positive examples of the concept, making accurate estimation of correlation impossible.

3.1.1 Method: MG-IS

To address this issue, we propose estimating ρ using importance sampling, where we draw inputs x_i from a proposal distribution $\mathcal{Q}$ that places higher probability on informative inputs such as the ones that are likely to contain the concept, rather than from the original uniform distribution $\mathcal{P}$. Importantly, importance sampling preserves the expectation we seek to estimate, i.e. $\mathbb{E}_{x \sim \mathcal{P}}[h(x)] = \mathbb{E}_{x \sim \mathcal{Q}}[\frac{p(x)}{q(x)}h(x)]$ for any distribution Q, given $q(x) > 0$ for all x where $p(x)h(x) \neq 0$. Here $p(x)$ and $q(x)$ are the probability mass functions (pmf) of $\mathcal{P}$ and $\mathcal{Q}$ respectively.

According to [25] (Sec 3.3.2, Theorem 3.12, reproduced in Appendix C.5), when estimating the expected value of a function h, i.e. $\mathbb{E}_{x \sim \mathcal{P}}[h(x)]$, the optimal proposal distribution $\mathcal{Q}^*$ that minimizes the variance of the estimator should have the pmf $q^*(x) \propto |h(x)|p(x)$. Intuitively, this distribution samples more frequently from inputs that make a larger contribution to the value of the expectation, reducing the number of samples needed for a stable estimate.

To apply importance sampling to our correlation estimation, we first rewrite (1) in the form of an expectation over inputs:

$$\rho = \mathbb{E}_{x_i \sim \mathcal{P}} \frac{([a_k]_i - \mu(a_k)) \cdot ([c_t]_i - \mu(c_t))}{\sigma(a_k)\sigma(c_t)}, \tag{2}$$

Based on the Theorem 1 and Eq (2), we can estimate the ρ by sampling x_i from $\mathcal{Q}^*$ with pmf $q(x)$ as:

$$q^*(x_i) \propto \left| \frac{([a_k]_i - \mu(a_k)) \cdot ([c_t]_i - \mu(c_t))}{\sigma(a_k)\sigma(c_t)} \right| \cdot \frac{1}{|\mathcal{D}|} \tag{3}$$

$$\implies q^*(x_i) \propto |\bar{a}_{ki} \cdot \bar{c}_{ti}|, \tag{4}$$

where $\bar{a}_{ki}$ and $\bar{c}_{ti}$ are the normalized quantities $\frac{[a_k]_i - \mu(a_k)}{\sigma(a_k)}$ and $\frac{[c_t]_i - \mu(c_t)}{\sigma(c_t)}$ respectively. So to minimize the variance of estimator ρ, we should sample x_i more from the probing images that have high absolute product values $|\bar{a}_{ki} \cdot \bar{c}_{ti}|$.

4

Since we do not know c_t before running the test (i.e. through human study or computationally expensive model), we instead use a cheaper (and less accurate) method to approximate c_t for the purposes of sampling. In particular, we use the SigLIP [27] model to predict c_t, which we denote as c_t^{siglip}. This gives us the q^{siglip} sampling distribution:

$$q^{siglip}(x_i) = \frac{|[\bar{a}_k]_i \cdot [\bar{c}_t^{siglip}]_i|}{\sum_{x_i \in \mathcal{D}} |[\bar{a}_k]_i \cdot [\bar{c}_t^{siglip}]_i|} \tag{5}$$

To account for the error in siglip estimates, and to ensure that importance sampling is unbiased ($q(x) > 0$ when $p(x)h(x) \neq 0$), our final sampling distribution $q^{\text{MG-IS}}(x)$ is a mixture of $q^{siglip}(x)$ and the original distribution $p(x) = \frac{1}{|\mathcal{D}|} \; \forall \; x \in \mathcal{D}$:

$$q^{\text{MG-IS}}(x) = (1 - \gamma)q^{siglip}(x) + \gamma p(x) \tag{6}$$

The hyperparameter γ was chosen experimentally, with best performance using $\gamma = 0.2$. See Appendix B.1 for details. This is equivalent to selecting 80% of the samples from q^{siglip} and 20% of them from the uniform distribution. We name this approach model-guided importance sampling (MG-IS) because the sampling is guided by the SigLIP models.

Sampling Correction. To obtain an unbiased estimate of the correlation using importance sampling, we must estimate not only the final coefficient, but also the mean and variance of the concept vector c_t. To evaluate them we apply the importance sampling correction $\frac{p(x)}{q(x)}$ sequentially. We first estimate the mean $\mu_S(c_t)$, and then the standard deviation $\sigma_S(c_t)$ from the samples $x_i \sim \mathcal{Q}$:

$$\mu_S(c_t) = \frac{1}{|S|} \sum_{i \in S} \frac{p(x_i)}{q(x_i)} [c_t]_i \tag{7}$$

$$\sigma_S(c_t) = \sqrt{\frac{1}{|S| - 1} \sum_{i \in S} \frac{p(x_i)}{q(x_i)} ([c_t]_i - \mu_S(c_t))^2}. \tag{8}$$

Next, we normalize the inputs: $[\bar{c}_t]_S = \frac{[c_t]_S - \mu_S(c_t)}{\sigma_S(c_t)}$. For a_k, since we have access to the entire neuron activation vector, we can directly compute the true mean and standard deviation (i.e. $\mu(a_k)$, $\sigma(a_k)$) instead of needing to estimate them from the sample. Thus, we have $[\bar{a}_k]_S = \frac{[a_k]_S - \mu(a_k)}{\sigma(a_k)}$. Finally, we obtain the estimated correlation score of ρ_S as:

$$\rho_S = \frac{1}{|S|} \sum_{i \in S} \frac{p(x_i)}{q(x_i)} [\bar{a}_k]_i \cdot [\bar{c}_t]_i, \tag{9}$$

where we use the subscript S to denote that the correlation score is estimated on the subset S.

3.1.2 Simulation study: Sampling Effectiveness

To study the accuracy of our estimated correlation ρ_S, we simulate a human study using existing labeled data. We then simulate limited data by only using a subset of these labels, and compare different sampling methods based how close their estimated correlation coefficient ρ_S is to the true correlation coefficient ρ_{gt} in Eq. (11).
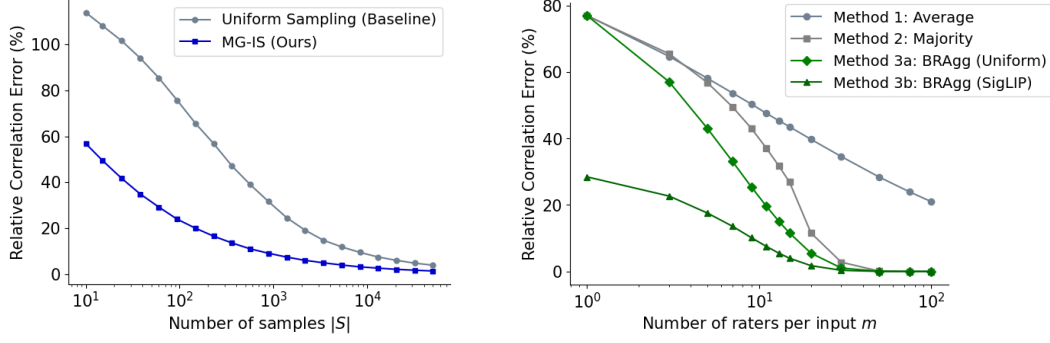
In particular, we use the ImageNet[8] dataset and its class and superclass labels, which we denote as the ground truth concept labels c_t^*. We then first generate descriptions t_k for all 2048 neurons in layer4 of ResNet-50[13] trained on ImageNet, by selecting the concept t_k from ImageNet class and superclass names $\mathcal{C}$ that maximizes correlation with neuron k's activations.

$$t_k = \text{argmax}_{t \in \mathcal{C}} \; \rho(a_k, c_t^*) \tag{10}$$

$$\rho_{gt} = \rho(a_k, c_{t_k}^*) \tag{11}$$

To compare different sampling strategies, we sample subsets of observations using their proposed distributions Q, and measure the Relative Correlation Error (RCE, Eq. 12) compared to correlation calculated from full distribution:

$$RCE(Q, |S|) = \frac{\sum_{k \in K} \mathbb{E}_{S \sim Q}[|\rho_S(a_k, c_{t_k}) - \rho_{gt}(a_k, c_{t_k}^*)|]}{\sum_{k \in K} |\rho_{gt}(a_k, c_{t_k}^*)|} \tag{12}$$

5

(a) Comparing different sampling strategies to estimate $\rho_{\mathcal{S}}$. Our Model-Guided Importance Sampling (MG-IS) using SigLIP estimates significantly outperform baseline regardless of the size of sampled subset.

(b) Comparison of different rating aggregation strategies on simulated human study with error rate $\eta = 23\%$, tested on the full dataset.

For Uniform Sampling, $Q = P$. In practice we evaluate the expectation by averaging over 10 samples from Q.

Results. Figure 2a shows the Relative Correlation Error of different strategies as a function of number of samples $|S|$. We can see that overall our MG-IS, as defined in Eq. (6), performs much better – which has low error and lower cost. Compared to the Uniform Sampling baseline, we can match the correlation estimation accuracy with around $15\times$ less samples. Alternatively, we can reach $\sim 65\%$ lower correlation estimation error with the same budget.

In Appendix B.3 we also experiment with a version of importance sampling without model guidance, based on a_k only, and show it can still be useful (better than uniform sampling) for the cases where no cheap model is available, but overall it performs worse than the MG-IS.

3.2 Bayes Rating Aggregation (BRAgg)

Another challenge in human evaluation is that the concept labels c_t obtained are noisy due to variability in rater attention, interpretation, and experience – this is particularly a problem when using crowd-sourced platforms such as Amazon Mechanical Turk. Such noise can significantly affect correlation estimation, particularly when the concept is rare as we might end up with more false positives than true positives.

To mitigate this, for each input x_i and concept t, we collect m independent binary ratings $r_{ti}^j \in \{0, 1\}, j \in \{m\}$ from different raters. Let $R_{ti} = \{r_{ti}^1, ..., r_{ti}^m\}$ denote the set of the ratings for a particular (input, concept) pair (x_i, t). The goal of rating aggregation is to combine these m judgments for each input x_i into a single estimated concept presence value $[c_t]_i$. Below, we describe three different methods for Rating Aggregation to get $[c_t]_i$.

Method 1 - Average: $[c_t]_i = \frac{\sum_{j=1}^m r_{ti}^j}{m}$

$\Rightarrow$ Ratings are simply averaged to form c_t.

Method 2 - Majority Vote:

$$[c_t]_i = \begin{cases} 1, & \text{if} \quad \frac{\sum_{j=1}^m r_{ti}^j}{m} > 0.5; \\ 0, & \text{if} \quad \frac{\sum_{j=1}^m r_{ti}^j}{m} \leq 0.5. \end{cases}$$

$\Rightarrow$ The most common rating becomes the label $[c_t]_i$.

While these methods are simple and commonly used, they may not be the optimal ways to model our confidence that a concept is present in a particular input. To better capture this uncertainty, we propose Bayes Rating Aggregation (BRAgg) to estimate $[c_t]_i$ as the posterior probability that the

6

concept is present: $[c_t]_i = \mathbb{P}([c_t^*]_i = 1 | R_{ti})$. Here c_{ti}^* denotes the "ideal" or true concept value without any labeling noise.

Method 3 - Bayes Rating Aggregation (BRAgg):

Let $C_{ti} := ([c_t^*]_i = 1)$ and $\neg C_{ti} = ([c_t^*]_i = 0)$, then:

$$[c_t]_i = \frac{\mathbb{P}(R_{ti}|C_{ti}) \cdot \mathbb{P}(C_{ti})}{\mathbb{P}(R_{ti}|C_{ti}) \cdot \mathbb{P}(C_{ti}) + \mathbb{P}(R_{ti}|\neg C_{ti})\mathbb{P}(\neg C_{ti})} \tag{13}$$

where we first use Bayes rule to expand the posterior $[c_t]_i = \mathbb{P}(c_{ti}^* = 1 | R_{ti})$ to get Eq (13), and then calculate the Likelihood and Prior terms in Eq (13) as follows:

(I) Likelihood: $\mathbb{P}(R_{ti}|C_{ti})$. We assume each rater makes errors uniformly at random with error rate η, i.e. for any input $\mathbb{P}(r_{ti}^j = c_{ti}^*) = 1 - \eta$, where η is a parameter we can estimate experimentally. Let $\alpha_{ti} = \sum_{j=1}^m r_{ti}^j$, we obtain the likelihood in below equations:

$$\mathbb{P}(R_{ti}|C_{ti}) = (1 - \eta)^{\alpha_{ti}} (\eta)^{(m - \alpha_{ti})} \tag{14}$$

$$\mathbb{P}(R_{ti}|\neg C_{ti}) = (\eta)^{\alpha_{ti}} (1 - \eta)^{(m - \alpha_{ti})} \tag{15}$$

(II) Prior: $\mathbb{P}(C_{ti})$. There are multiple ways of choosing priors, reflecting the confidence of whether a concept exists in an input x_i. In our analysis, we consider two different priors:

- **(a) Uniform Prior:** for all concepts t and all inputs x_i, set $\mathbb{P}(C_{ti}) = \beta$
- **(b) SigLIP Prior:** we leverage knowledge from a cheap evaluator, namely SigLIP to initialize the prior and set $\mathbb{P}(C_{ti}) = [c_t^{siglip}]_i$

Here β is a hyperparameter, which is set as 0.05. We evaluate the effects of different choices for β in Appendix B.2. For the SigLIP prior, we clip the prior to be between 0.001 and 0.999 to avoid extreme values which may dominate the final result. Our method with SigLIP prior can be seen as a hybrid evaluation that combines both human and model knowledge.

3.2.1 Simulation: Effect of Rating Aggregation Method

Setup: We follow the ImageNet based setup described in Section 3.1.2 for defining the ground truth correlation ρ_{gt} (Eq. 11). However, instead of using a subset of inputs, in this test we evaluate how accurately we can estimate the correlation with access to m noisy ratings for each input. For our testing, we used an error rate $\eta = 23\%$, i.e. randomly flipped 23% of the labels, which aligns with our experimental observations. We report the Relative Correlation Error (RCE) between estimated correlations and ρ_{gt} in Figure 2b.

As we can see from Figure 2b, Method 3b BRAgg(SigLIP) resulted in clearly the lowest RCE, allowing us to reach relatively accurate estimates with just a few raters, while other methods require > 10 raters per input before reaching low error rates. Overall BRAgg (Uniform) performs the second best, while Average rating aggregation fails to reach low errors even with 100 raters per input. Depending on the desired error rate, our BRAgg(SigLIP) can achieve it with 2-10$\times$ less raters than the best baseline (majority).

4 Experimental Validation and Human Study Design: Combining MG-IS and BRAgg

To further understand the performance of our proposed techniques, and to determine optimal parameters for our large scale human study in Section 5, we tested our methods in two realistic settings with both label noise and limited samples.

Setting 1 - Simulation: Our first setting is based on the simulation setups described in Section 3, but this time with *both* reduced samples and 23% label noise. The results of this setting are shown in Figure 3a.

Setting 2 - Crowd-sourced Validation Study: In this setting, we conducted a *real* crowd-sourced evaluation on AMT to annotate the concept labels c_t for samples x_i in the probing dataset, on

(a) Setting 1: Simulation with 23% label noise.

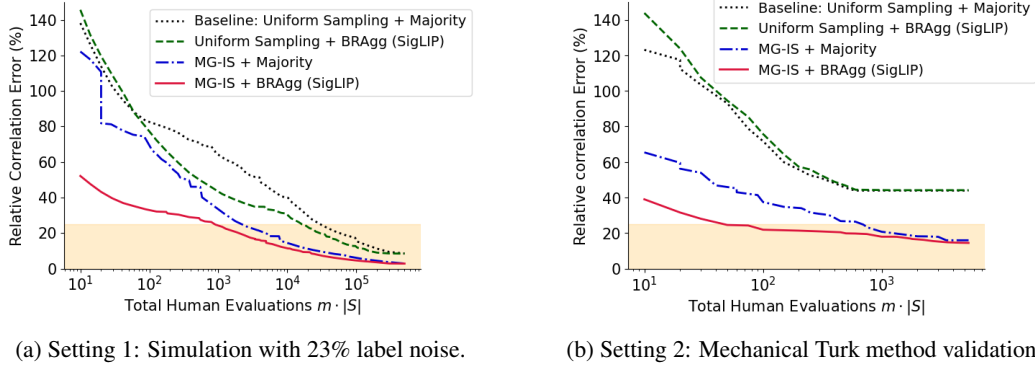(b) Setting 2: Mechanical Turk method validation.

Figure 3: Comparing the effect of our sampling technique MG-IS, rating aggregation method BRAgg independently and both together. The shaded region represents RCE$\leq 25\%$.

inputs where we also have access to the ground-truth labels c_t^* from the dataset for comparison. Specifically, we performed a crowdsourced evaluation of 10 neurons in Resnet-50 layer4 with explanations selected from ImageNet classes and superclasses following Eq. (10). We then estimated the correlation coefficient ρ_S using human ratings R on the rated subset S and measured how close our estimate is to the ground truth ρ_{gt}. This setup allows us to evaluate the error rate on Mechanical Turk and the effectiveness of our techniques on real crowdsourced data.

We randomly selected 10 neurons (excluding neurons with very vague concepts like *artifact*) and labeled 600 inputs per neuron with 9 AMT raters labeling each input. To estimate the evaluation accuracy with smaller number of raters/inputs we randomly sampled subsets of the ratings for our plots in Figure 3b (Note: each point is the average of 1000 random samples of subsets of the same size). We also used this setting estimate error rate of MTurk raters as $\eta = \mathbb{P}(r_{ti}^j \neq [c_t^*]_i)$, giving us an estimate of 23%, which we used for our simulations. We note this might slightly overestimate the noisiness of raters as some ImageNet labels are in reality ambiguous or even incorrect but it serves as a useful approximation.

Results: Figure 3 shows the results of different sampling and rating aggregation techniques across our two settings. The x-axis of Figure 3 shows the total number of ratings i.e. number of samples $|S|$ times number of ratings per input m. Since the optimal number of ratings per input depends on the total budget (for smaller budgets it is better to use fewer raters per sample), we plot the Pareto-front i.e. the optimal number of raters for each budget in Figure 3.

From the results we can see that by far the best results are achieved when using both MG-IS and BRAgg together (red line), while they also help individually with importance sampling being more important overall. In our Simulation setting (Fig. 3a, with a budget of 550 Evaluations per neuron, we can reach RCE of 27.5% with MG-IS + BRAgg. To reach the same error rate requires 1760 ($3\times$) evaluations with MG-IS only, 14607 ($25\times$) samples with BRAgg only and 22560 ($40\times$) samples using the baseline uniform sampling with majority vote, showing our techniques lead to around $40\times$ cost reduction overall.

On the MTurk evaluation (Fig. 3b), overall error rates are lower, but the trends between methods are similar, with MG-IS + BRAgg reaching 19.8% RCE within the budget of 550 evaluations per neuron, while MG-IS only requires 2000 evals to reach the same error ($4\times$). Uniform sampling methods never reach sufficiently low errors within the number of samples we evaluated, but extrapolating from current results we estimate it would require 10k-30k evaluations to reach 20% RCE.

Based on our findings and our budget, we choose to use 3 raters per input and to evaluate 180 inputs per neuron for 540 evaluations per neuron in the large scale study in Section 5. As we can evaluate 15 inputs in one task for the cost of $0.06, this evaluation will cost us $\frac{\$0.06}{15} \cdot 180 \cdot 3 = \$2.16$ per (neuron, explanation) pair. See Appendix A.3 for detailed results on the tradeoff between more inputs vs. more ratings per input, as well as how to use these experiments to decide the optimal parameters for our large-scale study.

Simple Exp. (l=1)	ND [3]	MILAN [14]	CD [20]	INVERT l=1 [7]	DnD [1]	MAIA [26]	LE(label) l=1 [21]	LE(SigLIP) l=1 [21]
RN-50 (Layer4)	0.1242 ± 0.002	0.0920 ± 0.002	<u>0.1904</u> ± 0.002	0.1867 ± 0.002	0.1534 ± 0.002	0.1396 ± 0.009	0.1793 ± 0.002	**0.2413** ± **0.002**
ViT-B-16 (Layer11 MLP)	0.0335 ± 0.002	0.0194 ± 0.002	0.1849 ± 0.004	0.1343 ± 0.004	0.1049 ± 0.003	0.1497 ± 0.021	<u>0.2704</u> ± 0.004	**0.2968** ± **0.004**

Table 1: SigLIP based simulation with correlation scoring comparing different simple explanation methods ($l = 1$). We can see Linear Explanation (LE) performs the best even with at length 1.

Accuracy of SigLIP only evaluation. Since our techniques are assisted by the SigLIP model, it is natural to ask how does it compare against fully automated evaluation? Our techniques can be seen as a hybrid method combining the strengths of human ratings and efficiency of model-based evaluation. Overall, our results indicate SigLIP-based evaluation is quite reliable by itself. When measuring the RCE of SigLIP only (using the full dataset), it reaches RCE of 32.10% on all neurons (vs. our estimated human study RCE of 27.90%), and on the subset of 10 neurons used in our MTurk validation SigLIP only RCE is 22.47% compared to 19.8% of our human study. So overall SigLIP only evaluation has around 15% higher error than our human study. This is still more accurate than a low budget human-study, and as such can be a good practical choice for future studies. However, it is important to remember that human evaluation is the gold standard for interpretability, and we should not blindly rely on automated evaluation without validating them with experiments such as ours.

5 Large-Scale Crowdsourced Study

Setup. Our study focuses on neurons in two layers of two different models: Layer4 of ResNet-50 trained on ImageNet, and MLP activations in Layer11 of ViT-B-16 [9] trained on ImageNet. We use the full ImageNet Validation set as probing dataset, see Appendix C.3 for additional details.

5.1 Method Selection via Automated Evaluation

Many different methods have been introduced to explain neurons in vision models. Due to limited budget, we are unable to compare all these methods on our crowdsourced study, so we first perform an automated evaluation to find the best methods to include in the crowdsourced study.

Explanation Complexity: Some methods explain a single neuron as a combination of concepts. For example [19, 7] use a logical composition of explanations, such as "dog OR cat".For fairness and simplicity, in our main experiments we focus on explanations with only one concept $l = 1$, but we also conduct an automated evaluation of complex experiments in Appendix B.6.

For automated evaluation we use the SigLIP based simulation with correlation scoring introduced by [21], see Appendix C.4 for details. The results of this automated evaluation are shown in Table 1.

Overall we can see Linear Explanations [21](even when restricted to a single concept) produces the highest correlation scores. Other well performing methods include INVERT [7] and CLIP-Dissect [20], followed by recent language model based explanations MAIA [26] and DnD[1]. We observed that Broden based methods [2, 19, 17] overall performed relatively poorly, likely because their concept sets do not include the relevant higher level concepts such as animal species needed to describe later neurons of ImageNet models.

5.2 Crowdsourced Evaluation

Details: Based on automated evaluation results, we chose to conduct our crowdsourced study on the 5 best performing simple explanations: LE (SigLIP)[21], CLIP-Dissect[20], INVERT[7], MAIA[26] and DnD[1]. We evaluated 100 randomly selected neurons for each of the two models, and we evaluated 180 inputs per neuron with 3 raters each. This gives us a total of 1000 (neuron, explanation) pairs to evaluate for a total cost of $2160. Our study was deemed Exempt from IRB approval by the IRB review board at our institution.
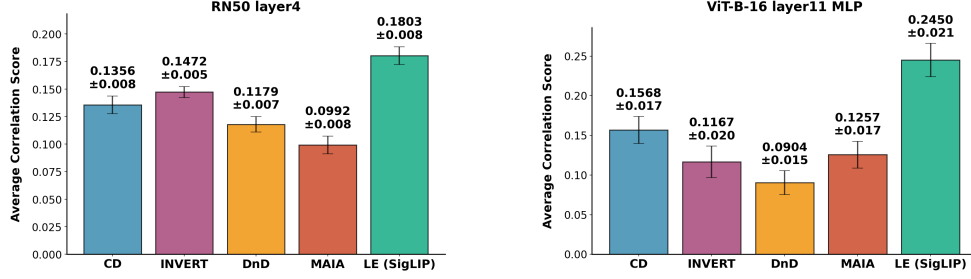
Figure 4: Results of our MTurk study. Using BRAgg(SigLIP) for rating aggregation. All methods are restricted to length 1 explanations. Overall explanations of LE(SigLIP) have the highest correlation coefficient.

User Interface. The main objective of our crowdsourced study is to have users annotate which inputs have concept t present. We do this by presenting raters with explanation t and 15 inputs per task, and asking them to select inputs where concept is present. The full study interface and additional details are available in Appendix C.2.

5.2.1 Results

The overall results from the crowdsourced evaluation are shown in Figure 4. See Appendix C.2 for more detailed results, including example explanations and their scores. We can see Linear Explanations performs the best overall despite being limited to using a predefined concept set. We think this is likely because it is the only explanation method tested that is optimized to explain the entire range of activations instead of focusing only on highest activations. The second best performing method is CLIP-Dissect [20] or INVERT [7] depending on the model evaluated. It is somewhat surprising that recent methods based on large language models [1, 26] did not outperform simpler baselines despite being able to provide very accurate and complex description in some cases. We think this is caused by a few reasons:

1. Focus on highly activating inputs only, leading to overly specific explanations that do not describe lower activations.
2. Inconsistency. While all 5 methods mostly produce relevant descriptions, more complex methods such as those relying on LLMs often have more variance in their description quality, leading to poor explanations for some neurons.

Overall the correlation scores were quite low, with best methods reaching ~ 0.2 correlation, highlighting the need for further improvement in explanation methods and/or creating more interpretable models.

6 Conclusion

In this paper, we introduced two new techniques for conduction principled crowdsourced evaluation of neuron explanations. First, we proposed Model-Guided Importance Sampling (MG-IS) to determine which inputs to show raters, reducing our labeling cost by $\sim 15\times$. Second, we introduced a new Bayes rule based method (BRAgg) to aggregate multiple ratings, which further reduces the amount of ratings needed to reach a certain level of accuracy by $\sim 3\times$. Finally, using these methods, we were able to conduct a large-scale human study with reasonable cost (lowering cost from estimated $90,000 USD to $2160 USD) without sacrificing accuracy – We evaluated 5 of the best performing methods for explaining vision neurons and discovered that Linear Explanations [21] performed the best overall.

References

[1] Nicholas Bai, Rahul A Iyer, Tuomas Oikarinen, Akshay Kulkarni, and Tsui-Wei Weng. Interpreting neurons in deep vision networks with language models. *TMLR*, 2025.

[2] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*, 2017.

[3] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *PNAS*, 2020.

[4] Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. `https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html`, 2023.

[5] Judy Borowski, Roland Simon Zimmermann, Judith Schepers, Robert Geirhos, Thomas S. A. Wallis, Matthias Bethge, and Wieland Brendel. Exemplary natural images explain {cnn} activations better than state-of-the-art feature visualization. In *International Conference on Learning Representations*, 2021.

[6] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

[7] Kirill Bykov, Laura Kopf, Shinichi Nakajima, Marius Kloft, and Marina MC Höhne. Labeling neural representations with inverse recognition. In *NeurIPS*, 2023.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[10] Yossi Gandelsman, Alexei A Efros, and Jacob Steinhardt. Interpreting the second-order effects of neurons in clip. *arXiv preprint arXiv:2406.04341*, 2024.

[11] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[12] Yoav Gur-Arieh, Roy Mayan, Chen Agassy, Atticus Geiger, and Mor Geva. Enhancing automated interpretability with output-centric feature descriptions. *arXiv preprint arXiv:2501.08319*, 2025.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[14] Evan Hernandez, Sarah Schwettmann, David Bau, Teona Bagashvili, Antonio Torralba, and Jacob Andreas. Natural language descriptions of deep visual features. In *ICLR*, 2022.

[15] Wenyi Hong, Weihan Wang, Ming Ding, Wenmeng Yu, Qingsong Lv, Yan Wang, Yean Cheng, Shiyu Huang, Junhui Ji, Zhao Xue, et al. Cogvlm2: Visual language models for image and video understanding. *arXiv preprint arXiv:2408.16500*, 2024.

[16] Neha Kalibhat, Shweta Bhardwaj, C Bayan Bruss, Hamed Firooz, Maziar Sanjabi, and Soheil Feizi. Identifying interpretable subspaces in image representations. In *ICML*, 2023.

[17] Biagio La Rosa, Leilani Gilpin, and Roberto Capobianco. Towards a fuller understanding of neurons with clustered compositional explanations. *Advances in Neural Information Processing Systems*, 36, 2023.

[18] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025.

[19] Jesse Mu and Jacob Andreas. Compositional explanations of neurons. In *NeurIPS*, 2020.

[20] Tuomas Oikarinen and Tsui-Wei Weng. Clip-dissect: Automatic description of neuron representations in deep vision networks. In *ICLR*, 2023.

[21] Tuomas Oikarinen and Tsui-Wei Weng. Linear explanations for individual neurons. In *International Conference on Machine Learning*, 2024.

[22] Tuomas Oikarinen, Ge Yan, and Tsui-Wei Weng. Evaluating neuron explanations: A unified framework with sanity checks. In *International Conference on Machine Learning*, 2025.

[23] OpenAI. GPT-4o system card. `https://openai.com/index/gpt-4o-system-card/`, 2024.

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[25] Christian Robert and George Casella. *Monte Carlo Statistical Methods, second edition*. Springer New York, 2004.

[26] Tamar Rott Shaham, Sarah Schwettmann, Franklin Wang, Achyuta Rajaram, Evan Hernandez, Jacob Andreas, and Antonio Torralba. A multimodal automated interpretability agent. In *Forty-first International Conference on Machine Learning*, 2024.

[27] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023.

[28] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[29] Roland S. Zimmermann, Thomas Klein, and Wieland Brendel. Scale alone does not improve mechanistic interpretability in vision models. In *NeurIPS*, 2023.

A Discussion

A.1 Limitations:

Due to time and budget constraints, our crowdsourced evaluation focuses on comparing descriptions on a few models/layers e.g. later layer neurons of ImageNet trained models. While we mostly find similar trends between the ResNet and ViT models, different description methods may have advantage on different types of neurons. For example, it is possible that Network Dissection based methods would perform better than they did on our current evaluation if we focus on lower layer neurons or on models trained on Places365 [28], as the labels in the Broden dataset are more suitable for these tasks. Similarly, we think LLM based methods [1, 26] might perform better when describing neurons of a sparse autoencoder [6] as these are more monosemantic and can be better described by highly activating inputs only.
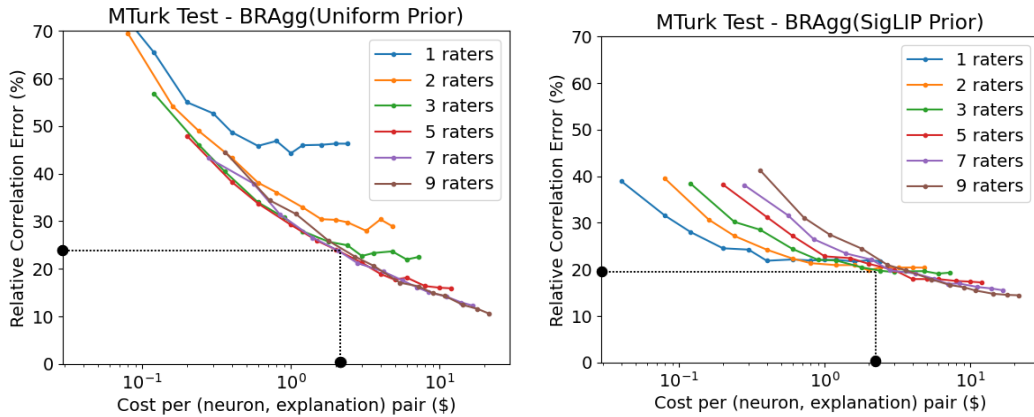
Second, our crowdsourced evaluation relies on Amazon Mechanical Turk workers, who are not experts and often make errors in labeling. While we introduced principled measures to estimate the errors as well as mitigations for error, we cannot improve their domain knowledge, which means a crowdsourced evaluation might favor simpler descriptions over more complex concepts requiring domain knowledge. To compare more complex descriptions or neuron descriptions in a specific domain it may be necessary to recruit domain expert raters.

Finally, our evaluation is focused on evaluating *input-based* neuron explanations that aim to explain the "input → neuron activation" function. Some recent work such as [10, 12] instead focus on *output-based* neuron explanations that aim to explain the "neuron activation → model output" relationship. Rigorously evaluating these *output-based* explanations will require different methodology and is an interesting problem for future work.

A.2 Broader Impact

This paper is focused on better understanding neural networks via interpretability, and as such we expect it's impact to be largely positive as better understanding of neural networks can help us for example identify failure modes before deploying and enable better control of models. As our focus is in particular on rigorous evaluation of neuron explanations, this can help avoid interpretability illusions or users over-relying on unreliable explanations.

A.3 Crowdsourced Study Design

(a) Aggregation with Method 3a: BRAgg - Uniform Prior.

(b) Aggregation with Method 3b: BRAgg - SigLIP prior.

Figure 5: Comparing rating aggregation strategies in Setting 2 (Testing on MTurk) described in Sec 4. The x-axis represents the cost of evaluations as number of inputs per neuron multiplied by number of raters per input, times cost per input. The lowest curve represents the optimal number of raters for a particular budget.

13

In this section we discuss how our methodology testing/validation settings described in Section 4 can be used to help design the finer parameters of our study such as how many raters to use per input. In our current MTurk setup, the cost of obtaining a rating for a single image by a single rater is $\frac{\$0.06}{15} = \$0.004$ as we show 15 inputs per task. We can then plot the expected error as a function of evaluation cost by plotting $n_{inputs} \cdot n_{raters} \cdot \$0.004$ on the x-axis.

In Figure 5, we plot the cost vs expected error rate for different numbers of raters we have per input. As we can see, the optimal number of raters per input depends on your budget, with smaller number of raters working better with smaller budgets, while for a larger budget it is optimal to use more raters per input.

Given our budget, we are aiming for a cost of around $2.2 per (neuron, explanation pair), and we can see the optimal number of raters and expected error rate for this budget following the black lines on our plot. Since we wish to get good results using both aggregation methods, we can see 3 raters (green line) is among the best for both methods, we choose to use 3 raters and 180 inputs per neuron for the crowd-sourced study. This gives us an expected correlation error of around 25% with Bayes - Uniform Prior and around 20% with Bayes - SigLIP prior based on the MTurk test.

B Ablation Studies
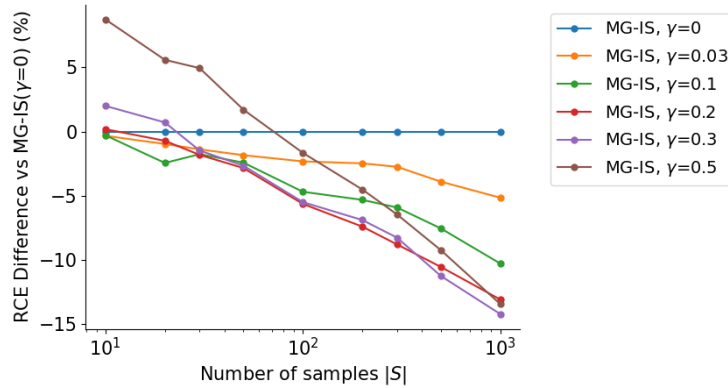
B.1 Choice of γ

Figure 6: The Effect of the γ hyperparameter on Importance sampling performance.

Figure 6 shows the effects of the hyperparameter γ (Eq. 6) on importance sampling performance on our simulated setup with no label noise. For visual clarity, we report the difference compared to baseline with $\gamma = 0$ instead of absolute numbers, so for example reducing RCE from 25% to 22.5% would show as -10% on this plot. Overall we can see our method is not very sensitive to this choice, but using a reasonable γ does reduce error by around 10%, with more relative difference on larger sample sizes. We use $\gamma = 0.2$ as it performs the best on sample sizes around 200 that we use in our large study.

B.2 Sensitivity to β in BRAgg (Uniform Prior)

Our Rating Aggregation Method 3a Bayes - Uniform Sampling described in Section 3.2 depends on the β hyperparameter to select the uniform prior. In this section, we conducted a test on Setting 2 (MTurk test, using MG-IS sampling) comparing different values for β. As shown in Fig 7, the exact choice of β has little effect on the correlation error, with values between 0.02 and 0.2 performing well. In our experiments we used $\beta = 0.05$ which performs well overall. We think the small sensitivity is mostly because changing prior has a relatively linear effect on predicted c_t, and since correlation coefficient normalizes the c_t the scale of c_t does not change the correlation. Each datapoint in Figure 7 is the average over 30 samples with different random seeds.
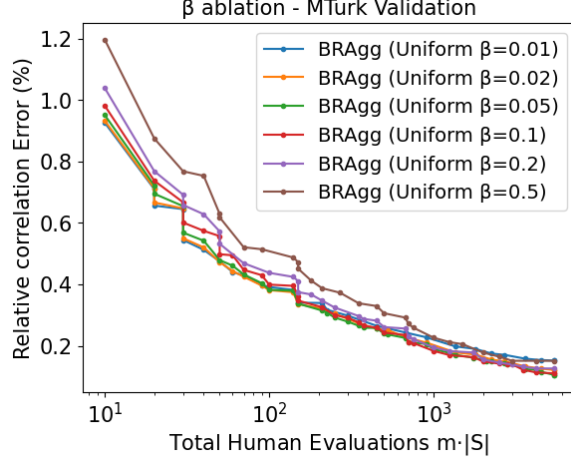
Figure 7: Comparing the error rate of using different β values for BRAgg(Uniform prior).

B.3 Activation-Guided Importance Sampling

Figure 8: Comparing different sampling strategies on simulated data with no label noise. We can see importance sampling without model guidance can still perform quite well and deliver significant gains over uniform sampling.

In addition to Model-Guided Importance Sampling (MG-IS), we also tested importance sampling guided purely by neuron activations, which we call activation guided importance sampling (AG-IS). This could be useful for cases where no cheap model is available to estimate concept presence. In particular we tested with importance sampling functions of the following form:

$$q^{act}(x_i) = \frac{|[\bar{a}_k]_i|^\alpha}{\sum_{x_i \in \mathcal{D}} |[\bar{a}_k]_i|^\alpha} \tag{16}$$

$$q^{\text{AG-IS}}(x) = (1-\gamma)q^{act}(x) + \gamma p(x) \tag{17}$$

As shown in Figure 8, we can see very significant gains over uniform sampling, with the best performing variant being $\alpha = 2$, but overall performance is not as good as our model-guided importance sampling. We used $\gamma = 0.2$ for these experiments.

B.4 Error Model for Rating Aggregation

For our main results, our Bayes modeling uses a uniform error model, i.e. we assume all have concepts and inputs share the same error rate η as described in Section 3.2. However, this is not

always a realistic assumption, as some inputs and concept are harder/more ambiguous, leading to higher error rates. To account for this, we also experimented using an alternative error model where the error rate η_{ti} is different for each pair of input i and concept t.

In this model $\mathbb{P}(r_{ti}^j = c_{ti}^*) = 1 - \eta_{ti}$, where $\eta_{ti} \sim \text{Beta}(0.75, 2.5)$. The hyperparameters $\alpha = 0.75$ and $\beta = 2.5$ were selected to maximize the probability of the MTurk user ratings we observed in Setting 2.

To do Bayesian inference with this model, we do inference on both c_{ti}^* and η_{ti} and report $[c_t]_i = \int_{\eta_{ti}} \mathbb{P}(C_{ti}, \eta_{ti}|R_{ti})d\eta_{ti}$.

Applying the Bayes rule $\mathbb{P}(C_{ti}, \eta_{ti}|R_{ti}) = \mathbb{P}(R_{ti}|C_{ti}, \eta_{ti})\mathbb{P}(C_{ti}, \eta_{ti})/\mathbb{P}(R_{ti})$ we get:

$$[c_t]_i = \frac{\int_{\eta_{ti}} \mathbb{P}(R_{ti}|C_{ti}, \eta_{ti}) \cdot \mathbb{P}(C_{ti})\mathbb{P}(\eta_{ti})d\eta_{ti}}{\int_{\eta_{ti}} \mathbb{P}(R_{ti}|C_{ti}, \eta_{ti}) \cdot \mathbb{P}(C_{ti})\mathbb{P}(\eta_{ti}) + \mathbb{P}(R_{ti}|\neg C_{ti}, \eta_{ti})\mathbb{P}(\neg C_{ti})\mathbb{P}(\eta_{ti})d\eta_{ti}} \tag{18}$$

Since $\mathbb{P}(\eta_{ti})$ and $\mathbb{P}(\_ti)$ are independent. In practice calculating exact solution for the integral is difficult, so we instead approximate it with 100 discrete bins.

Likelihood: $\mathbb{P}(R_{ti}|C_{ti}, \eta_{ti})$.

With $\alpha_{ti} = \sum_{j=1}^m r_{ti}^j$, we obtain the likelihood in below equations:

$$\mathbb{P}(R_{ti}|C_{ti}, \eta_{ti}) = (1 - \eta_{ti})^{\alpha_{ti}} (\eta_{ti})^{(m - \alpha_{ti})} \tag{19}$$

$$\mathbb{P}(R_{ti}|\neg C_{ti}, \eta_{ti}) = (\eta_{ti})^{\alpha_{ti}} (1 - \eta_{ti})^{(m - \alpha_{ti})} \tag{20}$$

Priors: For $\mathbb{P}(C_{ti})$ we use the same priors as in Section 3.2 (uniform or SigLIP prior). For $\mathbb{P}(\eta_{ti})$, we use the Beta distribution:

$$\mathbb{P}(\eta_{ti}) \propto \eta_{ti}^{\alpha} \cdot (1 - \eta_{ti})^{\beta} \tag{21}$$

The hyperparameters $\alpha = 0.75$ and $\beta = 2.5$ were selected to maximize the probability of the MTurk user ratings we observed.

Figure 9: Comparing different error models for rating aggregation. Note log-scale on both axes.

Results   As shown in Figure 9, the performance between different error models is similar, but overall the simplified uniform error model (solid lines) leads to slightly better results, so we use the simple error model for our main results. We are not fully certain why the simple model performs better, but one possibility is that even 9 ratings per user is not enough for the Bayes model to accurately infer per-input error rates.

B.5   SigLIP Only Evaluation Accuracy

Figure 10 shows comparison of different rating aggregation strategies vs a pure automated evaluation only relying on SigLIP scores. We can see that the optimal rating strategy depends on your budget.

16

Figure 10: Comparison the error of different rating aggregation strategies and only relying on SigLIP instead of conducting a human study. Using our Setting 2 results from MTurk with MG-IS sampling.

A fully automated evaluation provides similar accuracy as a hybrid evaluation (BRAgg(SigLIP)) with 100 samples per (neuron, explanation) pair, or a pure human evaluation with 800 samples per neuron. This implies that the optimal study strategy depends on your budget: If you cannot afford more than 100 human evaluations per neuron, it is best to rely on automated evaluation. For budgets between 100-1000 samples per neuron, a hybrid evaluation (BRAgg(SigLIP)) performs the best. For sufficiently large budget, i.e. >1000 evaluations per neuron, pure human evaluation (BRAgg(Uniform)) performs the best.
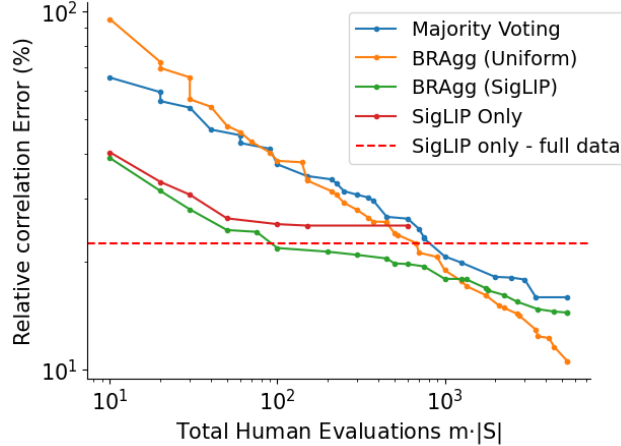
B.6 Compositional/Complex Explanations

A more complex explanation is typically more accurate, but it is harder to understand, and more expensive to evaluate as it requires labels for each of the concepts involved in the explanation. We use the length l to indicate explanation complexity, where l is the number of unique concepts in the explanation, e.g. "cat" OR "dog" would have lenght 2.

Complex Exp. (l>1)	Comp Exp [19] l=3	INVERT [7] l=3	CCE [17], l= 5	CCE [17], l=15	LE(label) [21] l=4.37/1.97	LE(SigLIP) [21] l=4.66/1.82
RN-50 (Layer4)	0.1399 ± 0.002	0.2341 ± 0.002	0.0993 ± 0.002	0.1510 ± 0.003	<u>0.2924</u> ± 0.002	**0.3772** ± **0.002**
ViT-B-16 (Layer11 MLP)	0.0468 ± 0.002	0.1101 ± 0.003	0.0534 ± 0.003	0.0570 ± 0.006	<u>0.3243</u> ± 0.005	**0.3489** ± **0.005**

Table 2: SigLIP based simulation with correlation scoring comparing complex explanations ($l > 1$). We can see Linear Explanation (LE) overall performs the best.

For fairness we split the explanations into two groups, simple explanations where $l = 1$ and complex explanations with $l > 1$. Table 1 in main text shows comparison of different simple explanations, while Table 2 compares complex explanation methods. For complex explanations, we can see Linear Explanations reaches the highest correlations with most methods having higher scores than their simple counterpart, highlighting the value of added complexity.

B.7 Model Ablation

In our main results we used the SigLIP [27] ViT-L-16-384 model both as the guide for importance sampling (MG-IS), and to provide the prior for our Bayes rating aggregation. For the Automated Evaluation described in Section 5.1 we used the SigLIP ViT-SO400M-14-384 model.

Model	Aggregation	CLIP-Dissect	INVERT l=1	DnD	MAIA	LE(SigLIP) l=1
ResNet-50	BRAgg	0.1392	<u>0.1490</u>	0.1242	0.1072	**0.1786**
(Layer4)	(SigLIP 2 Prior)	± 0.008	± 0.005	± 0.008	± 0.008	**± 0.009**
ViT-B-16	BRAgg	<u>0.1714</u>	0.1250	0.1039	0.1355	**0.2591**
(Layer11 mlp)	(SigLIP 2 Prior)	± 0.019	± 0.021	± 0.016	± 0.019	**± 0.023**
Average	BRAgg (SigLIP 2 Prior)	<u>0.1553</u>	0.1370	0.1141	0.1201	**0.2189**

Table 3: Large scale crowdsourced study results using SigLIP 2 as the prior for BRAgg.

The purpose of this section is to analyze the sensitivity of our results to the model choice. In particular, we test replacing the SigLIP ViT-L-16-384 model for sampling and rating aggregation with the following models:

1. CLIP [24] ViT-L14-336, which is an older model with slightly weaker benchmark performance.

2. SigLIP2 [?] ViT-gopt-16-384, which is a bigger and more recent model with slightly bettet benchmark scores.

For all models we used the weights and implementation from the open_clip package [?].

Results - Sampling Model: Figure 11 shows a comparison of different models for guiding sampling. We can see better models improve results but the differences are quite small. On average using CLIP instead of SigLIP leads to around 5 % (relative) increase in error with the same number of samples, while using SigLIP 2 leads to around 2 % (relative) reduction in error.

Figure 11: Comparing different models for guiding sampling in our simulated test (Setting 1). We can see stronger models improve sampling but the differences are small.

Results - Model for Bayes Prior: When changing the model used for creating the prior in BRAgg, we observe similar trends, with larger models generally performing better than smaller ones. Figure 12 shows the results of using different models as the Bayes prior We notice a somewhat bigger performance difference in this setting, with for SigLIP 2 reducing RCE at (3 raters per input and 200 inputs) from 19.70%(of SigLIP) to 17.95% for around 10% relative reduction in error. Interestingly, using CLIP as the prior performs worse with small sample sizes but beats SigLIP with large evaluation budget. Inspired by these results, we also analyze our main experiment results using SigLIP 2 instead of SigLIP as the Bayes prior and report the results in Table 3. Overall we can see the results are very similar to our results using SigLIP as the prior reported in Table 4. The only difference we notice is the correlation scores are slightly higher for all methods on ViT-B-16 neurons. This highlights our conclusions regarding the relative performance of different explanation methods are not sensitive to model choice.

Figure 12: Comparing different choices for the Bayes prior. We can see SigLIP2 performs the best overall.

Study information

▶ **Click to View Study Information**

☐ By checking this box I indicate that I am at least 18 years old, have read the study information above, and agree to participate in this research study.

Task

Select all the images that contain: **ground beetle**.

If you do not know what **ground beetle** means, use a tool like Google Image search to find out.
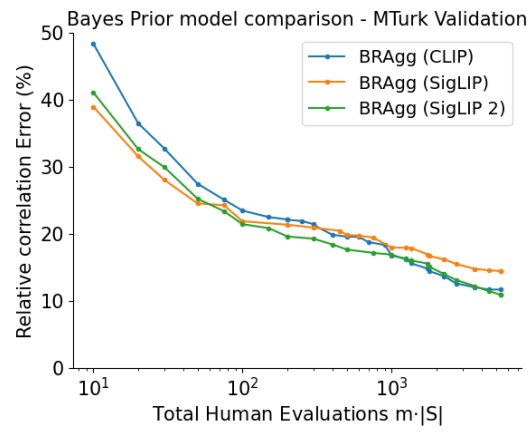
Submit

Figure 13: Our user study interface.

C Additional Details

C.1 MTurk Experimental Details:

Figure 13 showcases the full user interface displayed to the raters. We selected raters based in the US with over 10,000 tasks approved and $> 98\%$ task approval rate. Each rater was paid \$0.05 per task(and we paid another \$0.01 per task in fees to MTurk), which takes around 15 seconds to complete based on our testing, for an estimated \$12/hr earnings.

C.2 Detailed MTurk results

Table 4 shows more detailed results of our crowdsourced evaluation, including multiple rating aggregation methods. Comparing different rating aggregation methods, we can see that the correlation scores themselves vary a lot depending on the aggregation method used, but the order between explanations methods is quite consistent, for example Linear Explanations [21] score the highest regardless of aggregation method used.

Statistical Significance   Table 5 reports the statistical significance of our results using the two-sample Student's t-test. We can see we have statistically significant evidence that Linear Explanations performs better than all the other methods, with the exception of CLIP-Dissect when using BRAgg with uniform prior, and all methods with the SigLIP prior.

Example Ratings   Figures 14, 15 and 16 showcase example neurons and the descriptions assigned by different explanation methods, as well as the correlation scores estimated by our crowd-sourced study for those explanations. We colored explanations based on the estimated correlation coefficient, with Green: $\rho > 0.25$, Yellow: $0.25 \geq \rho > 0.10$ and Red: $0.10 \geq \rho$.

20

Model	Aggregation	CD[20]	INVERT[7]	DnD[1]	MAIA[26]	LE(SigLIP)[21]
ResNet-50 (Layer4)	Majority	0.0978 ± 0.010	0.0981 ± 0.010	0.0978 ± 0.010	0.0834 ± 0.010	**0.1216 ± 0.011**
	BRAgg (Uniform Prior)	0.1017 ± 0.009	0.0977 ± 0.010	0.1011 ± 0.008	0.0876 ± 0.010	**0.1261 ± 0.011**
	BRAgg (SigLIP Prior)	0.1356 ± 0.008	0.1472 ± 0.005	0.1179 ± 0.007	0.0992 ± 0.008	**0.1803 ± 0.008**
ViT-B-16 (Layer11 mlp)	Majority vote	0.1035 ± 0.016	0.0754 ± 0.016	0.0825 ± 0.017	0.0723 ± 0.013	**0.1208 ± 0.015**
	BRAgg (Uniform Prior)	0.0934 ± 0.014	0.0637 ± 0.014	0.0689 ± 0.013	0.0555 ± 0.012	**0.1018 ± 0.012**
	BRAgg (SigLIP Prior)	0.1568 ± 0.017	0.1167 ± 0.020	0.0904 ± 0.015	0.1257 ± 0.017	**0.2450 ± 0.021**
Average	BRAgg (SigLIP Prior)	0.1462	0.132	0.1042	0.1125	**0.2127**
	All	0.1148	0.0998	0.0931	0.0873	**0.1493**

Table 4: Detailed results of our large scale crowdsourced study. For each model we tested 100 random neurons, and we report the average correlation score of the explanations produced by each method, as well as the standard error of the mean. Comparing different rating aggregation methods, we can see that the correlation scores themselves vary a lot depending on the aggregation method used, but the order between explanations methods is quite consistent. Using length 1 explanations from both INVERT and LE.

		Hypothesis			
Model	Aggregation	LE >CLIP-Dissect	LE >INVERT	LE >DnD	LE >MAIA
RN-50	BRAgg(Uniform)	0.0601	**0.0079**	**0.0198**	**0.0010**
	BRAgg(SigLIP)	**0.0020**	**0.0098**	**0.0000**	**0.0000**
ViT-B-16	BRAgg(Uniform)	0.4295	**0.0037**	**0.0431**	**0.0062**
	BRAgg(SigLIP)	**0.0039**	**0.0001**	**0.0000**	**0.0001**

Table 5: Statistical significance of our results according to two-sample students t-test. We report the p-values of different hypothesis, with statistically significant p-values in bold.

We can see sometimes generative methods like MAIA [26] produce the best explanations, for example "Vibrant Green Elements in Nature" in Figure 14(Neuron 1126), but other times are too specific or fail to find the right concept 15.

Overall we did not observe very high correlation scores for any neurons in ResNet50, likely due to them being more densely activated and polysemantic, see for example neuron 108 (Fig. 14 that activates for both trains and invertebrates. On the other hand, for ViT-B-16 we observed several neurons that were extremely interpretable (correlation scores >0.5), such as Neuron 2187 (Fig. 16) that seems to only activate on Alpine ibex/Mountain goats.

ResNet50 layer4 - Neuron 108

Top:0.00%-0.01% Images, Activations:(4.67-5.32)

Top:0.01%-0.10% Images, Activations:(3.44-4.67)

Top:0.10%-1.00% Images, Activations:(2.37-3.44)

Top:1.00%-10.00% Images, Activations:(1.04-2.37)

CLIP-Dissect: trains(0.24)
INVERT: electric locomotive(0.21)
DnD: train on tracks(0.26)
MAIA: Trains in vibrant environments(0.15)
LE(SigLIP): invertebrate(0.33)

ResNet50 layer4 - Neuron 1126

Top:0.00%-0.01% Images, Activations:(5.22-6.62)

Top:0.01%-0.10% Images, Activations:(4.17-5.22)

Top:0.10%-1.00% Images, Activations:(2.73-4.17)

Top:1.00%-10.00% Images, Activations:(1.20-2.73)

CLIP-Dissect: patrol(0.03)
INVERT: granny smith(0.16)
DnD: wet green objects(0.14)
MAIA: Vibrant Green Elements in Nature(0.30)
LE(SigLIP): lime(0.07)

Figure 14: Visualization of example neurons, their descriptions and correlations scores from our crowdsourced evaluation (BRAgg with SigLIP prior). We have colored the descriptions based on the correlation score.

ResNet50 layer4 - Neuron 1556

Top:0.00%-0.01% Images, Activations:(6.81-8.20)

Top:0.01%-0.10% Images, Activations:(5.76-6.81)

Top:0.10%-1.00% Images, Activations:(4.35-5.76)

Top:1.00%-10.00% Images, Activations:(1.94-4.35)

CLIP-Dissect: pelican(0.14)
INVERT: drilling platform(0.11)
DnD: birds over water(0.17)
MAIA: pelicans over clear blue seas(0.08)
LE(SigLIP): seabird(0.26)

ResNet50 layer4 - Neuron 1839

Top:0.00%-0.01% Images, Activations:(6.52-7.41)

Top:0.01%-0.10% Images, Activations:(5.26-6.52)

Top:0.10%-1.00% Images, Activations:(3.50-5.26)

Top:1.00%-10.00% Images, Activations:(1.41-3.50)

CLIP-Dissect: taxis(0.16)
INVERT: car mirror(0.16)
DnD: side view of vehicles(0.16)
MAIA: Scenic cycling on winding roads(0.07)
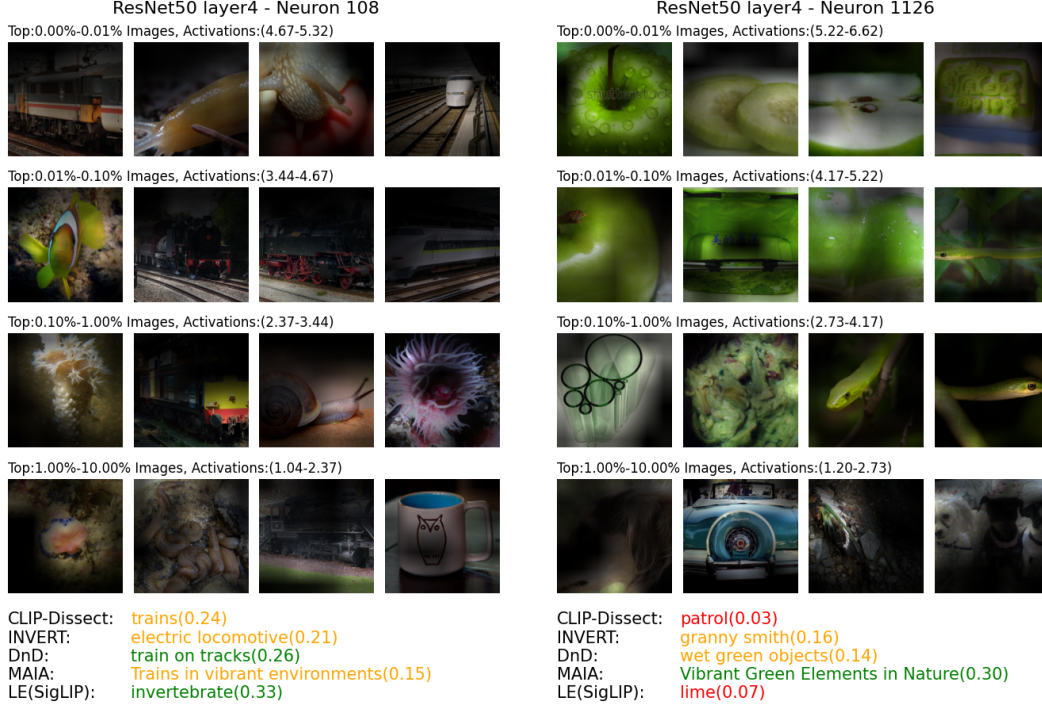LE(SigLIP): motor vehicle(0.46)

Figure 15: Visualization of example neurons, their descriptions and correlations scores from our crowdsourced evaluation (BRAgg with SigLIP prior). We have colored the descriptions based on the correlation score.

ViT-B-16 layer11 mlp - Neuron 1093
Top:0.00%-0.01% Images, Activations:(1.63-2.51)
Top:0.01%-0.10% Images, Activations:(0.77-1.63)
Top:0.10%-1.00% Images, Activations:(-0.00-0.77)
Top:1.00%-10.00% Images, Activations:(-0.01--0.00)

CLIP-Dissect: fridge(0.35)
INVERT: trifle(0.01)
DnD: food storage scenes(0.30)
MAIA: Brightly colored retail displays(0.07)
LE(SigLIP):   grocery(0.23)

ViT-B-16 layer11 mlp - Neuron 2187
Top:0.00%-0.01% Images, Activations:(1.86-2.16)
Top:0.01%-0.10% Images, Activations:(0.77-1.86)
Top:0.10%-1.00% Images, Activations:(-0.00-0.77)
Top:1.00%-10.00% Images, Activations:(-0.00--0.00)

CLIP-Dissect: wildlife(0.09)
INVERT: hen(0.01)
DnD: goats in mountainous terrain(0.45)
MAIA: Animals with Curved Horns in Mountains(0.51)
LE(SigLIP):   Alpine ibex(0.66)

Figure 16: Visualization of example neurons, their descriptions and correlations scores from our crowdsourced evaluation (BRAgg with SigLIP prior). We have colored the descriptions based on the correlation score.
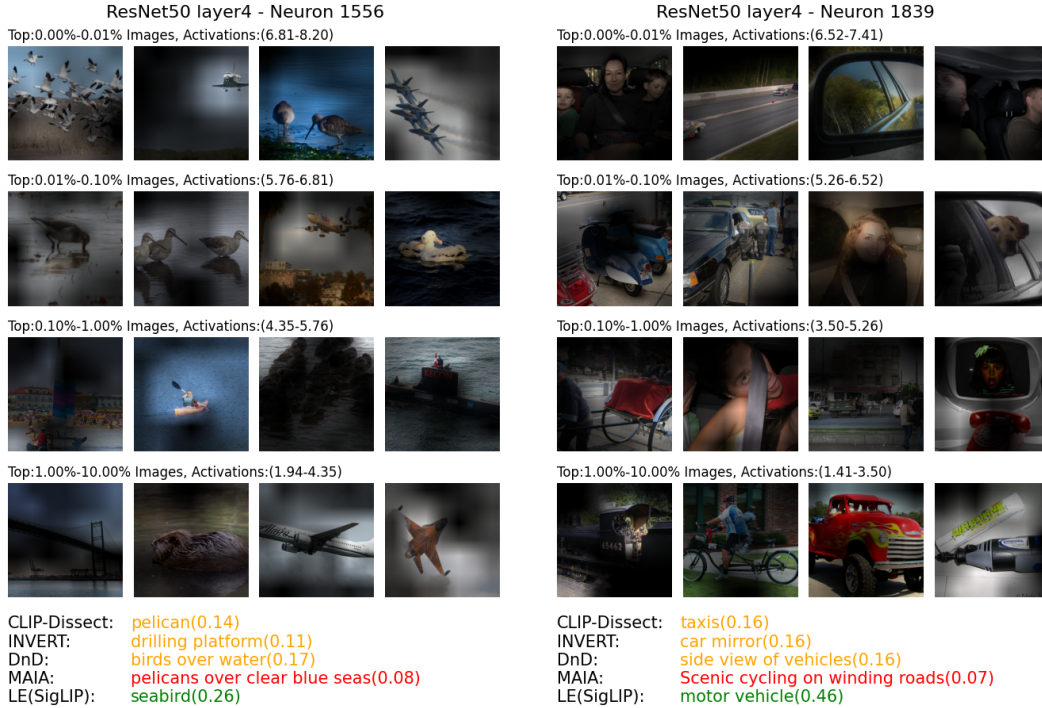
23

C.3 Experimental Details

Our evaluation focused on two models/layers:

1. Layer4 (end of residual block 4) of ResNet-50 trained on ImageNet. For simulation we use the neuron activation after average pooling, giving scalar activations, but for methods designed for 2d-activations, activations before average pooling are given as input.

2. MLP-activations in Layer 11 of ViT-B-16 [9] Encoder trained on ImageNet. We focus only on the activations of the CLS-token as this is the last layer and other tokens do not affect the prediction.

We use the full ImageNet validation dataset as $\mathcal{D}$ for the human study for all methods, and for generating explanations unless the method requires a specific dataset for explanation generation such as Broden [2].

For all methods using SigLIP guidance we use the SigLIP-SO400M-14-386 model.

C.3.1 Baseline Implementation Details

For practical purposes we made slight modifications to some baseline methods. The changes are discussed in detail below:

DnD [1]: The original implementation uses GPT-3.5 Turbo through the OpenAI API. Given the high cost of using the API and that recent open-source LLMs have strong performance compared to even closed-source LLMs like GPT-3.5, we replace GPT-3.5 Turbo with Llama 3.1-8B-Instruct [11]. DnD [1] showed that the older Llama 2 model was already better than GPT-3.5 Turbo and on-par with GPT-4 Turbo for neuron description, so our choice of Llama 3.1 does not degrade DnD's performance w.r.t. GPT-3.5 Turbo.

MAIA [26]: Compared to the original implementation, we replace GPT4-vision-preview with the newer GPT4o-2024-08-06 [23] since it has lower API costs and better performance. The method is still quite expensive, costing us approximately \$65 and \$116 to generate descriptions for 100 randomly selected neurons of ResNet-50 Layer4 and ViT-B-16 Layer11 MLP respectively. Note that this cost also includes repeating the experiment for ∼10 and ∼20 neurons from ResNet and ViT respectively which did not yield any neuron description in the first run. Initially, we also tried open-source LLMs with support for vision inputs (*i.e.* VLMs) like Llama-3.2-11B-Vision-Instruct [11], Llava-OneVision-Qwen2-7B-ov-hf [18], and CogVLM2-Llama3-Chat-19B [15]. However, these VLMs do not work well with MAIA since they fail to generate executable code, forget the image tokens and focus only on the last few text tokens if given a long prompt, and allow only one image input at a time. This is likely because these VLMs are geared towards visual question answering and do not possess the more generalized capabilities of GPT4/4o.

Methods relying on 2d activations: Many methods are designed to explain entire channels of CNNs with 2d activations [2, 19, 14, 17]. For ResNet-50 layer4 we fed the pre-avg pool activations to these methods for proper 2d input. However, for ViT-B-16 last layer only the CLS-token activations affect the output, and as such we are explaining neurons with scalar activations. In this case, we broadcasted the scalar activations into a 2d-tensor with the same value in all spatial locations. However this is not the intended way to use these methods and may partially explain poorer performance of some methods on ViT-neurons as observed in Tables 1 and 2.

CCE [17]: For Clustered Compositional Explanations, we tested two different versions: the $l = 15$ version corresponds to the default version with length 3 explanations for each of the 5 activation clusters. For the $l = 5$ version we used explanation length=1 with 5 clusters of activations. We also used the implementation of [17] to reproduce the results of Compositional Explanations [19] by setting explanation length=3 and number of clusters=1.

C.3.2 Subset of Neurons

For most methods in Table 1 and 2, we report the average correlation scores across all 2048 neurons for ResNet-50 layer4 and 3072 neurons for ViT-B-16 layer11 mlp. However for certain methods due to high computational and/or API cost we were only able to explain a subset of these neurons and report the average score of these subsets in Tables 1 and 2. We report the results for a subset of neurons for the following methods:

- MAIA [26]: Randomly selected subset of 100 neurons each for both RN50 and ViT-B-16.
- CCE [17] $l = 5$: For ViT-B-16 we used a subset of 1420 neurons. RN50 evaluated on all neurons.
- CCE [17] $l = 15$: RN50: subset of 984 neurons. ViT-B-16: subset of 422 neurons.

All other methods were evaluated on all neurons of each layer.

C.4 Automated Evaluation Details

For our automated evaluation(Sec. 5.1), we use Simulation with Correlation Scoring as described by [21]. This evaluation was originally proposed for language model neurons by [4].

The basic idea of simulation evaluation to use the *explanation* to predict neuron activations on unseen inputs. With correlation scoring we then evaluate the correlation coefficient ρ between the predicted activations s and actual neuron activations a_k on the entire test split of 10,000 inputs as done by [21].

For **simple explanations**, the predicted activation s is simply the presence of concept on that input.

$$s(x_i, t) = [c_t]_i \tag{22}$$

For a **linear explanation**, $E = \{(w_1, t_1), ..., (w_l, t_l)\}$ the predicted activation s is calculated following [21] as:

$$s(x_i, E) = \sum_{w_j, t_j \in E} w_j [c_{t_j}]_i \tag{23}$$

For **compositional explanations** [19], we calculate the predicted activation as follows using probabilistic logic. Different basic logical operators are calculated as:

$$s(x_i, t_1 \text{ AND } t_2) = [c_{t_1}]_i \cdot [c_{t_2}]_i \tag{24}$$

$$s(x_i, t_1 \text{ OR } t_2) = 1 - (1 - [c_{t_1}]_i) \cdot (1 - [c_{t_2}]_i) \tag{25}$$

$$s(x_i, \text{NOT } t) = 1 - [c_t]_i \tag{26}$$

Predictions for more complex compositions are then calculated by iteratively applying these rules.

Clustered Compositional Explanations: CCE [17] explanations are of the form $E = \{(l_1, u_1, F_1), ..., (l_r, u_r, F_r)\}$ where r is the number of activation clusters, and l_j, u_j are the lower and upper bound of activations for that cluster and F_j is a compositional explanation for activations of that cluster. To predict neuron activation based on this explanation, we use the following formula:

$$s(x_i, E) = \sum_{l_j, u_j, F_j \in E} \frac{l_j + u_j}{2} s(x_i, F_j) \tag{27}$$

This means if the concepts according to the formula of a cluster are present, we predict the neuron's activation will be in the middle of the clusters activation range.

For all automated evaluations we use SigLIP-SO400M-14-386 model to predict c_t following [21].

C.5 Theorem 1

Suppose we are estimating the expected value of function $h(x)$ when $x \sim \mathcal{P}$. Let $\mathcal{X}$ be the support of $\mathcal{P}$.

Theorem 1 ([25], Sec 3.3.2, Theorem 3.12). *For importance sampling with sampling distribution q:*

$$\mathbb{E}_{x \sim \mathcal{P}}[h(x)] = \int_{\mathcal{X}} h(x) \frac{p(x)}{q(x)} q(x) dx \approx \frac{1}{|S|} \sum_{i=1}^{|S|} \frac{h(x_i)p(x_i)}{q(x_i)}.$$

The choice of q that minimizes the variance satisfies $q(x) \propto |h(x)|p(x)$.

Proof: Reproduced from [25].

$$Var\left[\frac{h(x)p(x)}{q(x)}\right] = \mathbb{E}_q\left[\left(\frac{h(x)p(x)}{q(x)}\right)^2\right] - \left(\mathbb{E}_q\left[\frac{h(x)p(x)}{q(x)}\right]\right)^2 \tag{28}$$

Since the the second term $\left(\mathbb{E}_q\left[\frac{h(x)p(x)}{q(x)}\right]\right)^2 = \left(\int_{\mathcal{X}} h(x)p(x)dx\right)^2$ does not depend on q, in order to minimize variance we only need to minimize the first term.

From Jensen's inequality it follows that:

$$\mathbb{E}_q\left[\left(\frac{h(x)p(x)}{q(x)}\right)^2\right] \geq \left(\mathbb{E}_q\left[\frac{|h(x)|p(x)}{q(x)}\right]\right)^2 = \left(\int_{\mathcal{X}} |h(x)|p(x)dx\right)^2 \tag{29}$$

Giving us a lower bound for the first term. If we set

$$q(x) = \frac{|h(x)|p(x)}{\int_{\mathcal{X}} |h(z)|p(z)dz} \tag{30}$$

Which is a valid probability distribution, we get:

$$\mathbb{E}_q\left[\left(\frac{h(x)p(x)}{q(x)}\right)^2\right] = \left(\int_{\mathcal{X}} |h(z)|p(z)dz\right)^2 \tag{31}$$

This exactly matches the lower bound, proving that minimum variance is attained by setting

$$q(x) = \frac{|h(x)|p(x)}{\int_{\mathcal{X}} |h(z)|p(z)dz} \propto |h(x)|p(x) \tag{32}$$

C.6 Compute details

Our main contribution is focused on efficient crowd-sourced evaluation and as such our method is not computationally costly. The main computational cost associated with our method is calculating the SigLIP image encoder outputs for the entire $\mathcal{D}$, as these are needed for both importance sampling and the Bayes - SigLIP prior. However this is a relatively cheap onetime cost of around 20 minutes on a single NVIDIA RTX 6000 Ada Generation GPU.

The main computational expense associated with this paper involved running the baseline methods. In Tables 6 and 7 below, we report the approximate runtime to explain all neurons of a layer using different description methods using a single NVIDIA RTX 6000 Ada Generation GPU.

Simple Exp. (l=1)	ND [3]	MILAN [14]	CD [20]	INVERT l=1 [7]	DnD [1]	MAIA [26]	LE(label) l=1 [21]	LE(SigLIP) l=1 [21]
RN-50 (Layer4)	~ 1 hr	~ 1 hr	~ 5 mins	~ 1 hr	~ 55 hrs	~ 255 hrs	~ 1 hr	~ 1 hr

Table 6: Approximate runtime of different $l = 1$ baseline methods to explain neurons.

Complex Exp. ($l>1$)	Comp Exp [19] l=3	INVERT [7] l=3	CCE [17], l= 5	CCE [17], l=15	LE(label) [21] l=4.37/1.97	LE(SigLIP) [21] l=4.66/1.82
RN-50 (Layer4)	∼ 92 hrs	∼ 24 hrs	∼ 87 hrs	∼ 275 hrs	∼ 1 hr	∼ 1 hr

Table 7: Approximate runtime of different complex explanation baseline methods.

C.7 Inter-Annotator Consistency

To measure the quality of our ratings, we also measure inter-annotator consistency through Fleiss's Kappa. This measures the amount of agreement above chance, with 1 meaning perfect agreement and 0 random chance.

For MTurk raters in our test Setting 2, we observe a Kappa of 0.395. On the real study we observe Fleiss's Kappa of 0.191 on ViT neuron explanations and 0.274 on RN50 neurons. Overall we can see these agreement rates are relatively low, highlighting the need for error correction trough rating aggregation. While we think some of the noise in responses is underlying poor quality responses, some disagreement is also caused by underlying ambiguity in the task or poor explanations like MAIA sometimes outputting "Undetermined Selectivity".

To compare against higher quality annotations, we had 3 authors rate 150 samples each for the 10 neurons in Setting 2. For authors we observed a Fleiss Kappa of 0.774. This shows that obtaining higher quality raters has potential to improve study signal, but as we show we can still efficiently utilize noisy ratings with our methods.