"Discover-then-Name: Task-Agnostic Concept Bottlenecks via Automated Concept Discovery" - A Reproducibility Study

Anonymous authors
Paper under double-blind review

Abstract

Concept Bottleneck Models (CBMs) are a class of interpretable deep learning frameworks that improve transparency by mapping input data into human-understandable concepts. Recent advances, including the Discover-then-Name CBM proposed by Rao et al. (2024), eliminate reliance on external language models by automating concept discovery and naming using a CLIP feature extractor and sparse autoencoder. This study is focused on replicating the key findings reported by Rao et al. (2024). We conclude that the core conceptual ideas are reproducible, but not to the extent presented in the original work. Many representations of the active neurons appear to be disaligned with their assigned concepts. To address this discrepancy, we suggest a model extension; we propose an enhanced alignment method evaluated through a user study. Our extended model provides more interpretable concepts (with statistical significance), at the cost of a slight decrease in accuracy.

1 Introduction

Interpretable frameworks like Concept Bottleneck Models (CBMs) have gained attention for their ability to enhance explainability in deep learning. CBMs accomplish this by mapping input data into a human-understandable concept space, which can then be used for downstream tasks such as classification (Koh et al., 2020). This is achieved through a linear combination of concepts, which allows to explain the predictions made by a classifier. Traditional CBMs require labeled attribute datasets, but recent innovations use large language models (LLMs) (Brown et al., 2020) and vision-language models (Radford et al., 2021) for attribute-label-free concept learning. Despite this, relying on LLMs can lead to unfaithfulness to the model's reasoning process (Margeloiu et al., 2021).

To address these limitations, Rao et al. (2024) propose the Discover-then-Name CBM (DN-CBM), which automates the discovery and naming of concepts without relying on external LLMs. It uses a CLIP-based feature extractor (Radford et al., 2021) and sparse autoencoder (SAE) to disentangle input embeddings into human-understandable concepts for classification.

In this study, we reproduce and evaluate the findings presented by Rao et al. (2024), focusing on the performance and explainability of the DN-CBM framework. Building on their findings, we investigate the impact of cosine similarity on concept explainability and introduce a loss function which promotes more interpretable concepts. This loss function encourages alignment between the SAE neurons and their assigned concepts, which we then evaluate through a user study.

2 Scope of reproducibility

We investigate the following (main) claims from Rao et al. (2024) and label them (**C1-C3**) for reference.

- **C1**: **Automated concept discovery.** The DN-CBM framework can successfully discover latent concepts in the data without pre-selecting them. The SAE effectively identifies meaningful and human-understandable concepts directly from the CLIP feature space.

- **C2**: **Interpretability.** The method demonstrates that the discovered dictionary vectors align well with text embeddings of the concepts they represent in CLIP space. This alignment enables intuitive naming of the concepts, facilitating model interpretability across different tasks. As a result, the approach supports task-agnostic explanations of the model's decision process.
- **C3**: **Performance.** The DN-CBM achieves competitive performance on classification tasks across a variety of downstream datasets, ensuring task-agnosticity.

3 Methodology

This section outlines the methods used in this study. Sections 3.1, 3.2, and 3.3 discuss the models, datasets, and hyperparameter considerations respectively. Section 3.4.1 details the experimental setup used to validate the claims, and Section 3.4.2 presents the motivation and theoretical foundation of our extensions.

3.1 Model descriptions

Concept discovery. We follow the SAE approach proposed by Bricken et al. (2023) to transform CLIP features into a more interpretable latent space. This is achieved using a linear encoder $f(\cdot)$ with weights $\boldsymbol{W}_E \in \mathbb{R}^{d \times h}$, followed by a ReLU activation function $\phi(\cdot)$. The SAE is trained in a self-supervised manner by reconstructing the original features using a linear decoder $g(\cdot)$ with weights $\boldsymbol{W}_D \in \mathbb{R}^{h \times d}$. The latent space dimension, denoted by h, is much larger than the CLIP embedding dimension, denoted by d. For a given embedding $\boldsymbol{a} \in \mathbb{R}^d$, that is produced by a CLIP image encoder $\mathcal{I}$, the loss function is defined as:

$$\mathcal{L}_{\text{SAE}}(\boldsymbol{a}) = \|\text{SAE}(\boldsymbol{a}) - \boldsymbol{a}\|_2^2 + \lambda_1 \|\phi(f(\boldsymbol{a}))\|_1, \tag{1}$$

where λ_1 is a hyperparameter that enforces sparsity in activations. The SAE is typically trained on a large dataset, denoted as $\mathcal{D}_{extract}$, to extract a wide range of concepts.

Concept naming. After training, we automatically assign names to individual feature dimensions in the SAE's hidden representation. To achieve this, a vocabulary set $\mathcal{V} = \{v_1, \ldots, v_{|\mathcal{V}|}\}$ is embedded using a CLIP text encoder $\mathcal{T}$. Neurons in the SAE's latent space are labeled according to the highest cosine similarity between their dictionary (decoder) weights and the CLIP vocabulary representations. This is a natural choice as CLIP was trained to optimize cosine similarities between text and image embeddings. The dictionary weight vector $\boldsymbol{p}_c$ for neuron c is defined as the cth row of the decoder weight matrix:

$$\boldsymbol{p}_c = [\boldsymbol{W}_D]_{c,:} \in \mathbb{R}^d. \tag{2}$$

The corresponding label s_c is then determined as:

$$s_c = \underset{v \in \mathcal{V}}{\text{argmax}} \cos\left(\angle\left(\boldsymbol{p}_c, \mathcal{T}(v)\right)\right). \tag{3}$$

We define a vector representing alignment as a distribution over all cosine similarities so that we can refer to this later. Specifically, we introduce a vector $\boldsymbol{v} \in \mathbb{R}^h$, where the cth element is defined as:

$$v_c = \cos\left(\angle\left(\boldsymbol{p}_c, \mathcal{T}(s_c)\right)\right). \tag{4}$$

We refer to v_c as the *cosine similarity score* of concept c throughout this work. It quantifies the alignment between the assigned vector in CLIP space and the dictionary vector of neuron c.

Constructing CBMs. A CBM is constructed by connecting a linear probe $h(\cdot)$ to the encoder's output for downstream classification tasks. The probe is trained on a separate dataset, denoted $\mathcal{D}_{probe} = \{(\boldsymbol{x_1}, y_1)\}, (\boldsymbol{x}_2, y_2), \ldots\}$, where y_i represents the ground truth label of $\boldsymbol{x}_i$. The CBM $t(\cdot)$ is defined as:

$$t(\boldsymbol{x}_i) = (h \circ \phi \circ f \circ \mathcal{I})(\boldsymbol{x}_i). \tag{5}$$

During probe training, the SAE encoder layer is frozen, and the probe weights ($\boldsymbol{\omega}$) are adjusted based on the following loss function, where λ_2 is a sparsity hyperparameter and CE denotes the cross-entropy loss:

$$\mathcal{L}_{\text{probe}}(\boldsymbol{x}_i) = \text{CE}\left(t(\boldsymbol{x}_i), y_i\right) + \lambda_2 |\boldsymbol{\omega}|_1. \tag{6}$$

3.2 Datasets

We train and evaluate the DN-CBM on the following datasets:

CC3M. CC3M consists of image-caption pairs generated by extracting text from alt-texts of images on the web (Sharma et al., 2018). It contains 3,318,333 training images and a validation split of 15,840 images, with captions comprising a total of 51,201 unique token types. The validation split consists of 15,840 images. Due to link rot, approximately 68% of the originally collected images were retained in the final dataset. The dataset can be downloaded here.

ImageNet. Imagenet contains 1,000 categories, with over 1.2 million training images (Deng et al., 2009). Each category has approximately 1,000 training images. Additionally, there are 50,000 validation images (50 per class) and 100,000 test images (100 per class).

Places365. Places365 contains 1.8 million training images spanning 365 unique scene categories, representing a wide range of real-world environments (Zhou et al., 2017). The dataset also includes 36,000 validation images. These scene categories are distributed to reflect diverse locations encountered in everyday life, leading to a non-uniform distribution of images across categories. Due to computational constraints, we use 10% of the training dataset in our experiments, which we refer to as Places365*.

CIFAR10. CIFAR10 consists of 60,000 32x32 colour images in 10 classes (Krizhevsky, 2009). There are 6,000 images per class. The classes are airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. The training split contains 50,000 images, and the test split consists of 10,000 images.

CIFAR100. CIFAR100 consists of 60,000 images in total (Krizhevsky, 2009), with 100 classes and 600 images per class. It consists of 50,000 training images and 10,000 test images.

CC3M is used for training and evaluating the SAE ($\mathcal{D}_{extract}$), whereas ImageNet, Places365*, CIFAR10 and CIFAR100 are used for training and evaluating the linear probe ($\mathcal{D}_{probe}$) for downstream tasks.

3.3 Hyperparameters

In reproducing the experiments, we adhered to the hyperparameters specified by the authors, which are presented in Table 1. For this study, we will utilize the probe hyperparameter settings v_2 unless stated otherwise.

Table 1: Hyperparameters. We use the same hyperparameters as Rao et al. (2024) but note that the original paper details a hyperparameter sweep without specifying the probe hyperparameters. Thus, we consider two configurations for the linear probe: v_1, from the GitHub README example for Places365, and v_2, the default settings in the code for each probe dataset. The Adam optimizer is configured using the default hyperparameter settings (Kingma and Ba, 2015).

General		**SAE**		**Probe**		
Hyperparameter	Value	Hyperparameter	Value	Hyperparameter	v_1	v_2
text encoder ($\mathcal{T}$)	CLIP ResNet-50	latent dim (h)	8192	learning rate[1]	10^{-2}	10^{-3}
image encoder ($\mathcal{I}$)	ResNet-50	L_1 sparsity (λ_1)	3×10^{-5}	batch size	512	512
vocabulary ($\mathcal{V}$)	Google 20k	learning rate	5×10^{-4}	epochs	200	200
vocabulary size ($\lvert\mathcal{V}\rvert$)	20000	epochs	200	L_1 sparsity (λ_2)	0.1	1
embedding dim (d)	1024	batch size	4096	optimizer	Adam	Adam
		batch resample freq	10			
		optimizer	Adam			

3.4 Experimental setup and code

Below is a summary of the resources used in our study, followed by an explanation of the methods to validate the claims from Section 2, along with our proposed extension to the original work. The full code is available at our GitHub repository.

[1]For CIFAR100, a probe learning rate of 10^{-2} was used in v_2.

3.4.1 Reproducibility setup

We utilized the publicly available codebase by Rao et al. (2024) to reproduce the key findings of the original study. This comprehensive GitHub repository includes all necessary scripts to generate the plots and the final results. The computational tasks were carried out using GPU resources provided by the Dutch national supercomputer, Snellius, with funding support from the University of Amsterdam. A Nvidia A100 Tensor Core GPU was used to run the experiments. The total computational expense for the reproduction equals 91.91 GPU hours. This has an estimated emissions of 10.87 kgCO2eq.[2]

To investigate **C1**, we train an SAE with the same architecture and hyperparameters as the original paper. We then align the dictionary vectors (Equation 2) with the CLIP feature representations. Finally, we visualize the ranked cosine similarity scores for the DN-CBM and our reproduced DN-CBM. This will help us determine if our reproduced latent space is as closely associated with CLIP features as in the original work. We use this same figure as an initial indication for **C2**, which asserts that the vectors should align well. To further investigate **C2**, we reproduce both the qualitative and quantitative analyses from Rao et al. (2024).

The qualitative analysis involves reproducing a figure that illustrates examples of named concepts along with the top images that activate these concepts across four datasets. In the original work, this qualitative analysis includes only concepts that have high cosine similarity scores. We extend this by incorporating both highly aligned and lower-aligned concepts to assess their representational quality. Furthermore, we reproduce the explanation of the decision of the DN-CBM by classifying random images from the Places365* dataset and reporting the top concepts that contribute to the decision. We assess generalizability by reporting similar results for other datasets in Appendix B.1.

The quantitative support for **C2** includes a human feedback survey to validate the alignment of concepts with neurons across varying cosine similarity scores. Participants were asked to rate concept consistency and naming accuracy for 12 images under one concept. Similar to the qualitative analysis, these are the top images that activate the concept. High consistency is defined as a set of images with a consistent overarching theme. This survey uses a 1-5 rating scale for accuracy, where 1 indicates poor alignment, and 5 indicates strong alignment between concept and images. This survey was conducted with 22 participants; we replicated the method and compared the results.

To reproduce **C3**, we follow the original DN-CBM model specification by attaching a linear probe to the SAE to classify images. We compute the classification accuracy on ImageNet, Places365*, CIFAR10, and CIFAR100, specified in Section 3.2. We compare our accuracies to the original accuracies.

In Rao et al. (2024), the authors conduct two additional analyses beyond the core claims. First, they conduct a quantitative evaluation using the SUNAttributes dataset (Patterson et al., 2014) to compare discovered concepts from their SAE to ground truth labels, filtering and merging nodes based on cosine similarity with text embeddings. We did not reproduce this analysis. We emphasize that the core claims emphasize alignment with CLIP embeddings and interpretability across diverse datasets, which offer broader generalizability than the specific evaluation with SUNAttributes. Second, the authors assess the effectiveness of concept interventions in their DN-CBM model using the Waterbirds-100 dataset (Petryk et al., 2022; Sagawa et al., 2020) to test robustness against spurious correlations between bird type and background. We chose not to reproduce this analysis as it focuses on human-driven interventions, whereas our emphasis is on automation and task-agnostic adaptability.

Additionally, we did not include an analysis of semantic consistency using k-means clustering on concept activation vectors or the CLIP-Dissect component of the user study. We believe these analyses are unnecessary in determining the paper's core claims, as our focus remains on reproducibility and the alignment of concepts with CLIP embeddings.

[2]The emissions were calculated using Machine Learning calculator (Lacoste et al., 2019). The estimated carbon efficiency of The Netherlands was 0.473 kgCO2eq in January 2025 (Nowtricity, 2025).

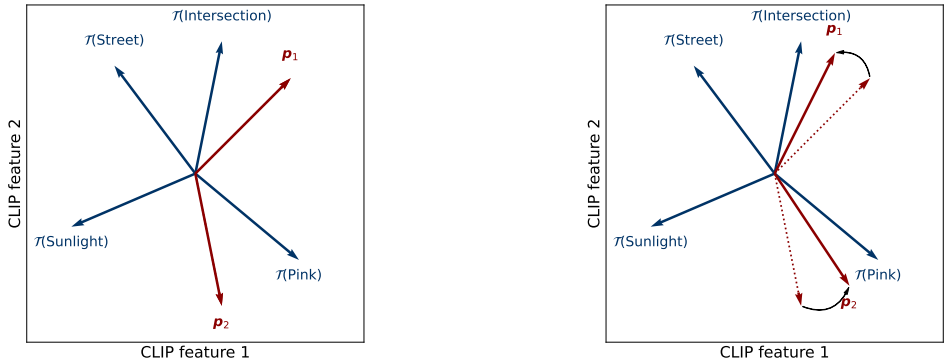
3.4.2 Extension to Original Work

Motivation. In their novel method, Rao et al. (2024) propose a post-hoc approach to explainable AI. This approach is compelling for maintaining the model's accuracy, as it does not constrain the model to be inherently explainable during training. However, the degree of explainability is theoretically limited as certain aspects of a freely trained model cannot be fully captured by a single word or concept due to the constraints of human language. Rao et al. (2024) report good alignment between dictionary vectors and text embeddings in CLIP space (high cosine similarity score). However, defining "good" alignment in high-dimensional space is challenging, especially when the alignment varies significantly across the neurons. We investigate this alignment in **C2** and propose a model extension to improve the alignment of dictionary vectors and text embeddings.

Model extension. We introduce a loss function that encourages SAE neurons to align with more interpretable concepts. This loss function is designed for fine-tuning, as it depends on a set of pre-defined concepts learned during training. This loss function is defined as:

$$\mathcal{L}_{\text{FSAE}}(\boldsymbol{a}) = \|\text{SAE}(\boldsymbol{a}) - \boldsymbol{a}\|_2^2 + \lambda_1 \|\phi(f(\boldsymbol{a}))\|_1 - C\left(\frac{\phi(f(\boldsymbol{a}))}{\|\phi(f(\boldsymbol{a}))\|_2} \cdot \boldsymbol{v}\right), \tag{7}$$

where C is the cosine penalty parameter that dictates the strength of our introduced penalty term. This term encourages alignment between the latent space and explainable features using the cosine similarity scores of $\boldsymbol{v}$. The conceptual idea behind our fine-tuning loss is shown in Figure 1.



(a) Dictionary vectors before fine-tuning.

(b) Dictionary vectors after fine-tuning.

Figure 1: Conceptual overview of the fine-tuning process, in CLIP embedding space. In this example, the vocabulary set is $\mathcal{V} = \{\text{Street}, \text{Pink}, \text{Intersection}, \text{Sunlight}\}$, with both CLIP and latent dimensions $h = d = 2$. Neuron 1 is assigned $s_1 = \{\text{Intersection}\}$, and neuron 2 is assigned $s_2 = \{\text{Pink}\}$. After fine-tuning, the dictionary vectors are better aligned with the CLIP embeddings of their respective assigned names. In practice, the vocabulary set, CLIP embedding dimension, and number of neurons are much larger.

A key advantage of scaling by $\phi(f(\boldsymbol{a}))$ is that the incentive for alignment is proportional to the activation magnitude. This prevents rarely activated neurons from arbitrarily aligning with CLIP embeddings without capturing meaningful representations. Consequently, the neurons involved in inference are more likely to exhibit a high cosine score. The normalization prevents the optimization routine from generating excessively large activations to exploit the cosine loss.

Excessively large C values, however, introduce an issue. When C becomes too large, the cosine similarity term dominates the loss function, causing the distribution of $\boldsymbol{v}$ to shift towards higher values. This reduces variation in similarity values. As more explainable neurons are incentivized to activate, their activations may lose meaning. Furthermore, because we normalize $C(\phi(f(\boldsymbol{a})) \cdot \boldsymbol{v})$ by $\|\phi(f(\boldsymbol{a}))\|$, there is no incentive to increase the activation magnitude. This leads to the diffusion of neuron activations, meaning that more

neurons become active but with lower activation values. This reduces the model's interpretability because when it relies on a vast number of weakly activated neurons, it becomes difficult to pinpoint specific neurons as the primary contributors to the decision-making process. This creates a trade-off: increasing neuron explainability can sometimes come at the cost of making their activations less meaningful.

To determine an appropriate value for C, we experiment with different values and analyze their impact on neuron interpretability, activations, and accuracy. We fine-tune our model seven times on CC3M, each time using a different value of the hyperparameter $C \in [10^{-6}, 10^{-5}, \ldots, 10^{0}]$. We use the mean cosine similarity score as a proxy for neuron interpretability, while activation levels are quantified by the average magnitude of nonzero elements in $\phi(f(\boldsymbol{a}))$. Additionally, we measure the validation accuracy on the Places365* dataset. A balanced value of C is then selected based on a qualitative assessment of these factors.

To evaluate the impact of our method, we compared the reproduced DN-CBM model with our fine-tuned extension using the optimal C value. We do this by conducting a user study. For our user study, we selected all concept explanations for image classifications that appeared more than five times in a subset of the Places365* test set. Next, we ranked these concepts by cosine similarity and selected the bottom 40, middle 40, and top 40 aligning concepts for both models (resulting in a total of 6 groups). For each concept, we sampled five images without replacement corresponding to the classification explanation, resulting in a total of 240 concepts, each paired with five images. We asked each participant to rate two randomly sampled concepts from each group on a scale from 0 to 5, reflecting how many images aligned with the concept. This sampling approach ensured that each participant evaluated only 12 images while the entire dataset was thoroughly assessed across all participants through the randomized selection process. A total of 203 participants completed the questionnaire. Further details of the outline of our user study are given in Appendix A.1. To assess whether the differences in user ratings between the two models are statistically significant, we employ the Wilcoxon signed-rank test (Wilcoxon, 1945). This non-parametric test is appropriate given the paired nature of our data and the lack of a normal distribution in the ratings. The details of the test and its implementation are further discussed in Appendix A.2.

Note that our user study differs from the user study conducted by Rao et al. (2024). In their work, they analyzed neuron activations in response to concepts, we argue that classification explanations are more meaningful for our objective, because they align with the model's primary task. For example, a neuron representing the concept of "turquoise" may have a list of top-activating images, such as a car in CIFAR10. However, this does not necessarily indicate that the neuron plays a key role in classifying the object (car).

4 Results

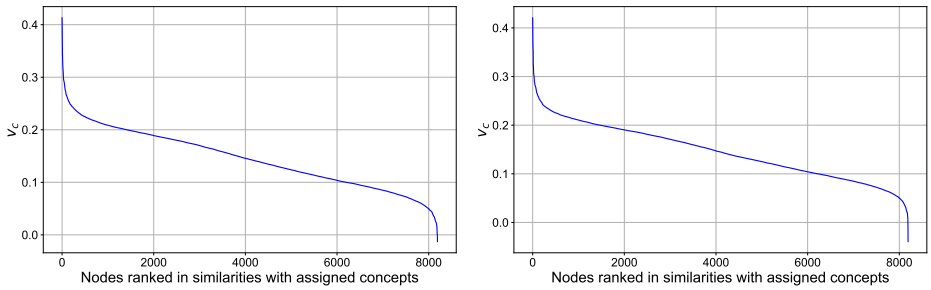
This section begins by presenting findings on the reproducibility of **C1** through **C3**. Subsequently, the results of extensions to DN-CBM are discussed. The findings confirm **C1** and **C3**, while contradictory results are observed for **C2**.

4.1 Result reproducing original paper

To assess the reproducibility of **C1** we visualize the ranked cosine scores for the original and reproduced DN-CBM in Figure 2. We observe that we successfully reproduce a latent space that maps concepts within a certain cosine similarity range. The distribution of values closely aligns with the original findings, demonstrating that our approach achieves a comparable mapping. This validates the reproducibility of **C1**.

To validate **C2**, we extracted the top-activating images with the highest cosine similarity score, across the four datasets, as shown in Figure 3. Notably, the top-activating images strongly correspond to their respective concepts for ImageNet and Places365, indicating a high degree of semantic consistency. This is in line with the results of Rao et al. (2024) and supports **C2**. For CIFAR10 and CIFAR100, the images correspond less with the concepts "plaid" and "sweater", which may be attributed to the limited expressiveness of the dataset.

To further assess the reproducibility of **C2**, we examine Figure 2. The cosine scores range broadly from approximately -0.01 to 0.42, indicating the presence of lower-aligned concepts. The original study focuses



(a) DN-CBM. **(b)** Reproduced DN-CBM.

Figure 2: Ranked cosine similarity scores of the assigned concepts. Comparison of the cosine similarity scores from the original DN-CBM (Figure 2a) and the reproduced DN-CBM (Figure 2b).

Figure 3: Task-agnosticity of concept extraction. We present the concepts with the highest cosine similarity score, alongside their top-activating images from four datasets.

only on highly aligned concepts (Figure 3), which may not generalize to the explainability of all neurons. To assess **C2** across the entire network, we examine Figure 4, which presents concepts from the lower end of the cosine similarity distribution and their top-activating images across four datasets. This analysis reveals that not all concepts are as explainable as those in the highly aligned Figure 3, leading us to conclude that **C2** cannot be fully reproduced.

Figure 4: Lower aligned task-agnosticity of concept extraction. We present low-aligned concepts alongside their top-activating images from four datasets. The images associated with each concept demonstrate low consistency with the assigned concept name across datasets.

To continue the qualitative analysis of **C2**, we display the top concepts contributing to the decision-making process for two randomly selected samples from the Places365* dataset in Figure 5. Rao et al. (2024) show similar examples with concepts that, indeed, all describe aspects of the image's theme. This qualitative analysis supports their claim that concepts are associated with the predicted class, thus aiding interpretability. Upon examining Figure 5, we find that not all displayed concepts contribute meaningfully to the decision-making process, which partially challenges **C2**. The left figure illustrates that the concepts are thematically consistent with the corresponding class, supporting the validity of the approach. The right figure raises concerns. Specifically, the presence of concepts such as "kayaking", "dams", and "trivium", do not clearly correspond to the class. Consequently, we find that the original results cannot be reproduced to the same

extent. This qualitative analysis is extended to different datasets in Appendix B.1, which demonstrates similar results.

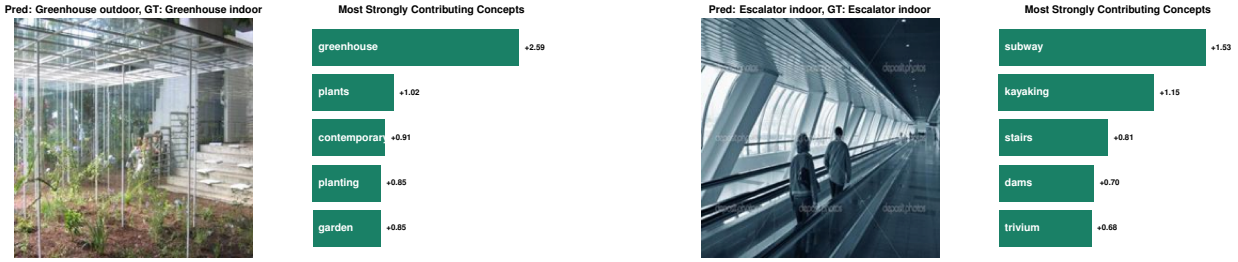
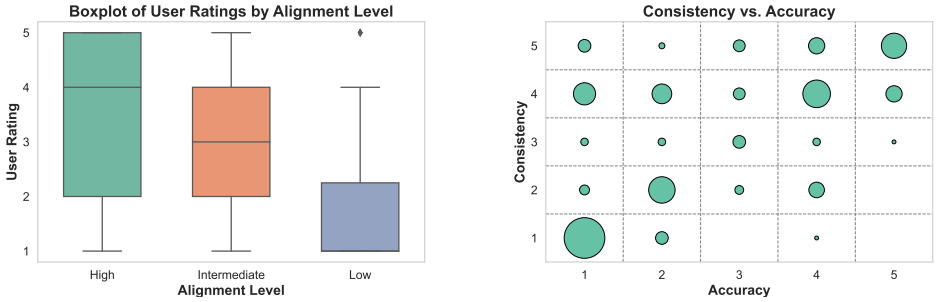


Figure 5: Explaining decisions using the reproduced DN-CBM. We present randomly drawn examples of images from the Places365* dataset alongside the top concepts contributing to their classification.

To reproduce the quantitative analysis for **C2**, the survey results are presented in Figure 6. Figure 6a reveals an overall decline in semantic consistency as concept alignment decreases, aligning with prior findings. However, the results are not an exact match, as the original work reported the highest scores for intermediate-aligned concepts, likely due to the small sample size of both user studies. Figure 6b shows a positive relation between accuracy and consistency, with consistency generally improving as accuracy improves. Nonetheless, some cases exhibit high consistency despite low accuracy. This trend aligns with the original findings, suggesting reasonable reproducibility of the user study.



(a) Semantic consistency of concepts. **(b)** Consistency vs. Accuracy.

Figure 6: User study on concept accuracy. Semantic consistency is plotted for nodes with high, intermediate, and low alignment to their assigned text embeddings of the reproduced SAE (Figure 6a). In Figure 6b, the semantic consistency scores are plotted against name accuracy. The survey has 22 participants.

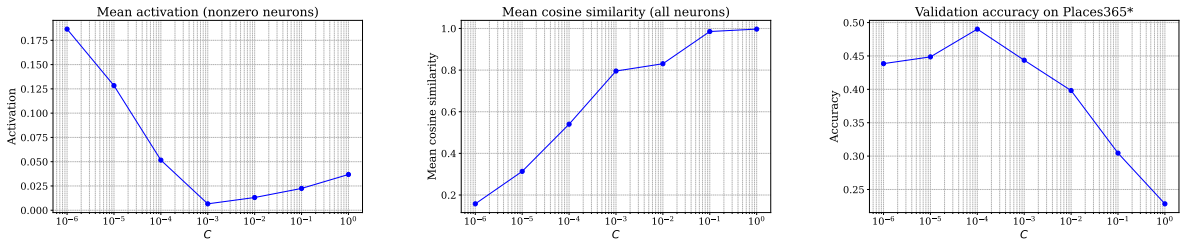
To assess **C3**, we report the classification accuracy for the reproduced model and the original work in Table 2. The classification accuracies of ImageNet, CIFAR10 and CIFAR100 are similar for the original and reproduced DN-CBM. The performance on Places365* is worse for the reproduced model. This could be attributed to the use of a smaller version of this dataset. Overall our results are in agreement with **C3**.

Table 2: Comparison of performance of the original and reproduced results. We report the classification accuracy (%) of the original paper and our reproduction using CLIP ResNet-50 on ImageNet, Places365*, CIFAR10 and CIFAR100. "Finetuned" is our model suggestion, which incorporates a cosine loss function with $C = 10^{-4}$ (Equation 7), leading to improved explainability. '*' indicates the use of a smaller dataset compared to the original paper; 10% of Places365.

Dataset	ImageNet	Places365	CIFAR10	CIFAR100
Original	72.9	53.5	87.6	67.5
Reproduced	72.7	50.0*	86.7	68.6
Finetuned (Ours)	70.5	49.3*	83.9	64.5

4.2 Result beyond original paper

Figure 7 shows the results of our experiments with different cosine penalty parameters, C. After completing standard training, we fine-tuned for an additional 30 epochs using the loss function in Equation 7, which was generally sufficient for convergence. We observe the theoretical trends discussed in Section 3.4.2. As C increases, activations become more diffused, the average cosine similarity score rises, and at some point it comes at the cost of accuracy. A value of $C = 10^{-4}$ strikes a balance, significantly improving cosine similarity (from 0.146 to 0.540), still reasonable activations and the highest validation accuracy. This value of C is selected to represent our extension in the experiments we conducted. To further illustrate the effect of the cosine penalty, Appendix B.2 provides a visualization of the cosine score distribution across nodes for different values of C.



(a) Mean activation. **(b)** Mean cosine similarity. **(c)** Validation accuracy on Places365.

Figure 7: Impact evaluation of C. Comparison of mean activation across nonzero neurons, mean cosine score, and accuracy for different C values. These plots are obtained with probe hyperparameters v_1.

Referring to Table 2, we compare the accuracy of our model to the original DN-CBM using the same hyperparameters (v_2). We observe that, under these hyperparameters[3], our finetuned model underperforms, with accuracy drops ranging from -0.7% to -4.1%, depending on the dataset. This decline is attributed to the additional constraint imposed by neuron interpretability, highlighting a trade-off between accuracy and interpretability. We further assess the interpretability component in our user study.

The results of our user study are presented in Figure 8. Our findings indicate that, across all alignment levels, our model, on average, achieves higher user ratings. This effect is particularly pronounced for low and intermediate-aligned concepts, where the Wilcoxon signed-rank test produces p-values of 0.000. This indicates that, even under a conservative significance threshold of 0.1%, the null hypothesis—that there is no average difference in ratings between the reproduced and fine-tuned models—would still be rejected. For the highly aligned concepts, the ratings are similar, and the statistical test yields a p-value of 0.176. Notably, the intermediate-aligned concepts receive higher user ratings than the high-aligned concepts in our model. This occurs because enhancing neuron explainability can diminish the meaningfulness of activations, as discussed in Section 3.4.2.

Figure 9 provides a qualitative intuition regarding the model's improvement. We present a randomly selected image from Places365* and from CIFAR10, classified and explained by both the original DN-CBM (left) and our extended model with $C = 10^{-4}$ (right). Looking at the Places365* figure, using the original model (9a), the concept "ivy" is logically related to the predicted class "Vegetable garden." However, other concepts such as "arnold," "cosmos," "labrador," and "eleven" lack clear interpretability. In contrast, our model (9b) predicts "Field wild", which is more coherently supported by concepts such as "meadow", "fields", "flower", and "crops". Similarly for CIFAR10, the original model (9c) generates concepts "pelican", "Michigan","aaliyah", "busty" and "elephants", which do not contribute meaningfully to the prediction "Horse". The same image classified using our model (9d) generates the contributing concepts "horses", "equine" (horse-like) and "horseback". This suggests that the extended model provides more interpretable and semantically relevant explanations. For generalization, we plot local explanations for CIFAR100 and ImageNet in Appendix B.2.

[3]Note that these hyperparameters were explicitly optimized for the original DN-CBM, placing our model at a slight disadvantage.

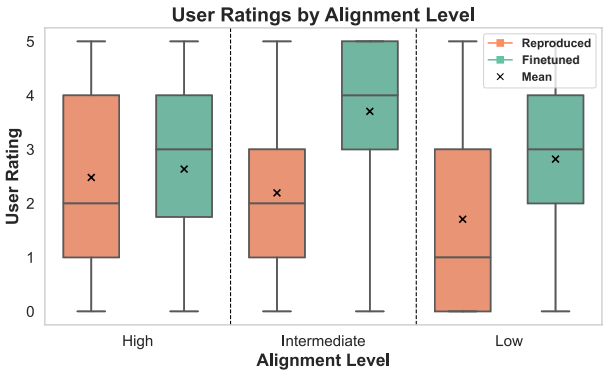
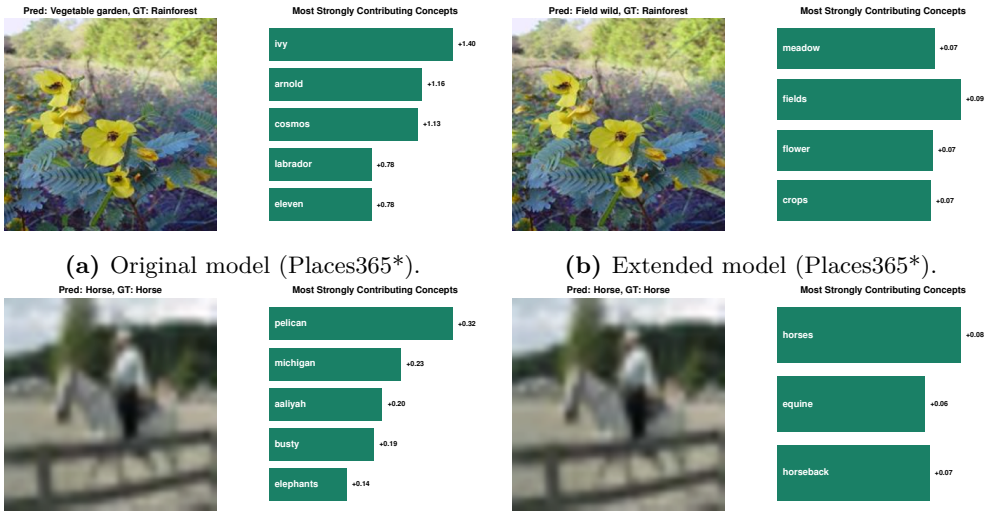


Figure 8: User study results. User ratings for the reconstructed DN-CBM (orange) and our model with $C = 10^{-4}$ and probe hyperparameters v_1 (green) on Places365*. The ratings are evaluated across three groups based on descending relative cosine similarity scores: high alignment (top 40 highest-aligning concepts of the model), intermediate alignment (40 concepts from the middle), and low alignment (bottom 40 concepts).



(a) Original model (Places365*). **(b)** Extended model (Places365*).

(c) Original model (CIFAR10). **(d)** Extended model (CIFAR10).

Figure 9: Explaining decisions using DN-CBM and our extension. The top row shows an example from Places365* with the predicted class, ground truth, and top contributing concepts for both the original DN-CBM (left) and our extended model with $C = 10^{-4}$ (right). The bottom row presents a similar comparison for CIFAR10.

5 Discussion

Our study successfully reproduces **C1** and **C3**, demonstrating that the DN-CBM framework can effectively uncover latent concepts in the data without pre-selecting them while maintaining competitive classification accuracy. Our results for **C2** indicate partial reproducibility, as especially lower-aligned concepts often fail to contribute meaningfully to explaining the decision-making process. Fine-tuning the DN-CBM with an extended loss function that drives the dictionary vector of neurons towards explainable CLIP vectors enhances the interpretability of these neurons, as supported by both qualitative and user study analyses. However, this comes at the cost of classification accuracy, introducing a trade-off that can be controlled by adjusting the value of C, provided that C does not increase to the point where activations in the hidden state of the SAE diffuse excessively. This raises an intriguing direction for future research, which could involve

conducting user studies across a range of values for the cosine penalty parameter C. Such studies could provide a more nuanced understanding of the trade-off between accuracy and interpretability.

The concept of an alignment loss through the C penalty term could also be generalized to incorporate different proximity measures. While cosine similarity is one approach to assessing neuron-concept alignment, it may not always be the most effective metric. Determining similarity between vectors in high-dimensional spaces is inherently challenging, however (Aggarwal et al., 2001).

Lastly, we propose several general directions for future research that are not necessarily specific to our fine-tuned model: removing vague concepts from the vocabulary set, using a larger dataset than CC3M to develop a more general off-the-shelf SAE, and discouraging low-aligning concepts from contributing to classifications through the probe. All of these directions hold promise in pushing the boundaries of explainable AI.

What was easy. Reproducing the study was relatively straightforward due to the author's well-documented, publicly available code. The GitHub repository included clear instructions on setting up the environment, running experiments, and reproducing figures. The modularity of the code and the comprehensive documentation made it easy to verify the majority of the original claims. We considered it unnecessary to contact the original authors

What was difficult. Despite the code being well-documented, we encountered some minor issues, such as errors related to storing results. These were manageable but required some extra debugging. Due to Snellius' limitations on uploading large datasets, we had to use a smaller version of Places365. While it was a necessary adaptation, it might have affected the performance in ways that were not fully comparable to the original paper's results. Additionally, the paper did not clearly specify all hyperparameter settings, and while the GitHub code provided some guidance, inconsistencies in argument values added to the difficulty of replicating the results exactly.

References

Aggarwal, C. C., A. Hinneburg, and D. A. Keim (2001). On the Surprising Behavior of Distance Metrics in High Dimensional Space. In J. Van den Bussche and V. Vianu (Eds.), *Database Theory — ICDT 2001*, Berlin, Heidelberg, pp. 420–434. Springer Berlin Heidelberg.

Bricken, T., A. Templeton, J. Batson, B. Chen, A. Jermyn, T. Conerly, N. Turner, C. Anil, C. Denison, A. Askell, et al. (2023). Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread 2*.

Brown, T., B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, and A. Askell, et al. (2020). Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin (Eds.), *NeurIPS*, Volume 33, pp. 1877–1901. Curran Associates, Inc.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei (2009). ImageNet: A large-scale hierarchical image database. In *CVPR*, pp. 248–255.

Kingma, D. P. and J. Ba (2015). Adam: A Method for Stochastic Optimization. In *ICLR*.

Koh, P. W., T. Nguyen, Y. S. Tang, S. Mussmann, E. Pierson, B. Kim, and P. Liang (2020, 13–18 Jul). Concept Bottleneck Models. In *ICML*, Volume 119, pp. 5338–5348. PMLR.

Krizhevsky, A. (2009). Learning Multiple Layers of Features from Tiny Images. Technical report, Computer Science Department, University of Toronto.

Lacoste, A., A. Luccioni, V. Schmidt, and T. Dandres (2019). Quantifying the Carbon Emissions of Machine Learning. In *CoRR*, Volume abs/1910.09700.

Margeloiu, A., M. Ashman, U. Bhatt, Y. Chen, M. Jamnik, and A. Weller (2021). Do Concept Bottleneck Models Learn as Intended? In *ICLRW*.

Nowtricity (2025, 1). CO2 emissions per kWh in Netherlands.

Patterson, G., C. Xu, H. Su, and J. Hays (2014, 05). The SUN Attribute Database: Beyond Categories for Deeper Scene Understanding. *IJCV 108*(1–2), 59–81.

Petryk, S., L. Dunlap, K. Nasseri, J. Gonzalez, T. Darrell, and A. Rohrbach (2022). On Guiding Visual Attention with Language Specification. In *CVPR*, pp. 18092–18102.

Radford, A., J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever (2021). Learning Transferable Visual Models From Natural Language Supervision. In *ICML*, pp. 8748–8763.

Rao, S., S. Mahajan, M. Böhle, and B. Schiele (2024). Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery. In A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol (Eds.), *Computer Vision – ECCV 2024*, Cham, pp. 444–461. Springer Nature Switzerland.

Sagawa, S., P. W. Koh, T. B. Hashimoto, and P. Liang (2020). Distributionally Robust Neural Networks. In *ICLR*.

Sharma, P., N. Ding, S. Goodman, and R. Soricut (2018). Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *ACL*, pp. 2556–2565.

Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin 1*(6), 80–83.

Zhou, B., A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba (2017). Places: A 10 Million Image Database for Scene Recognition. In *IEEE TPAMI*.

A Additional explanation

A.1 Survey method

In our user study, we aimed to evaluate concept alignment across different models. Below, we provide additional details on participant selection and survey structure.

Participant recruitment and survey distribution. The survey was distributed to a diverse group of participants, including university students, colleagues, friends, and family. To ensure unbiased evaluation, participants were not informed which concepts originated from our model versus the reproduced DN-CBM model.

Survey structure and model parameters. The parameter C in our model was set to 10^{-4}. Before answering the main survey questions, participants were provided with example questions to familiarize them with the task (Figure 10). An example question from the study is included in Figure 11).

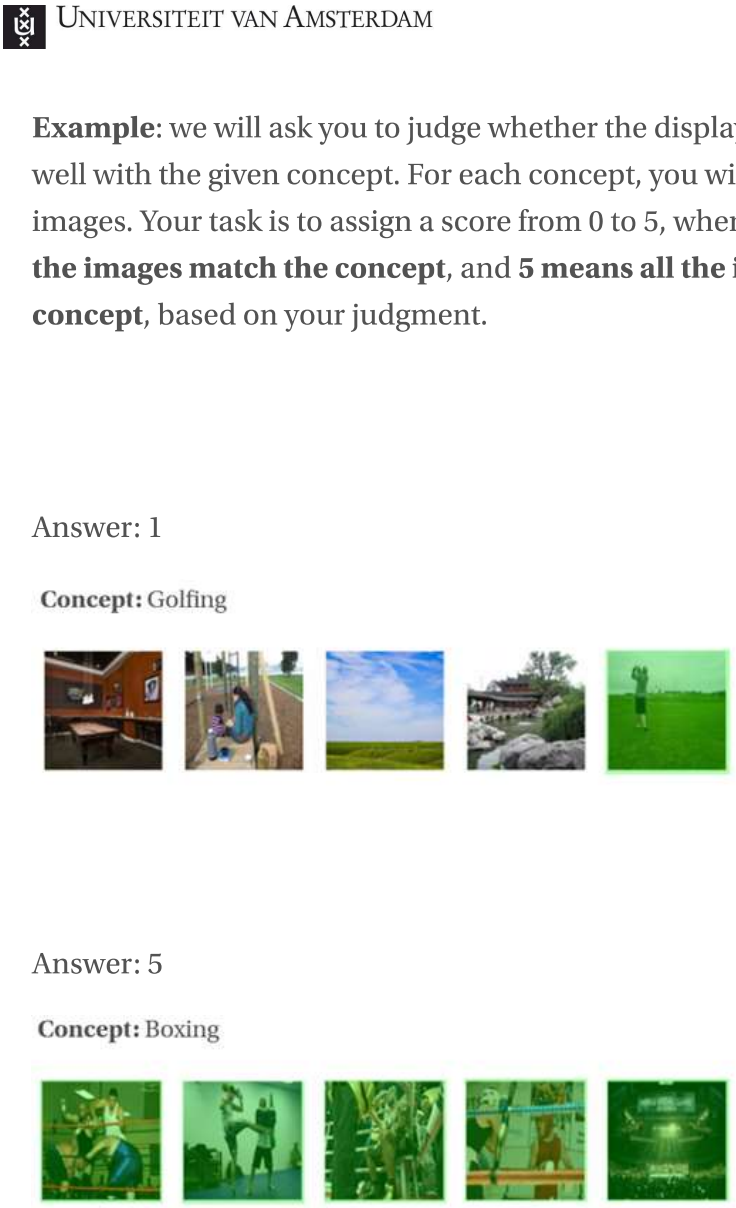


Figure 10: Examples presented at the start of the survey. At the start of the survey, we included three sample questions (two of which are shown in this figure) to help participants become familiar with the question format and response process.

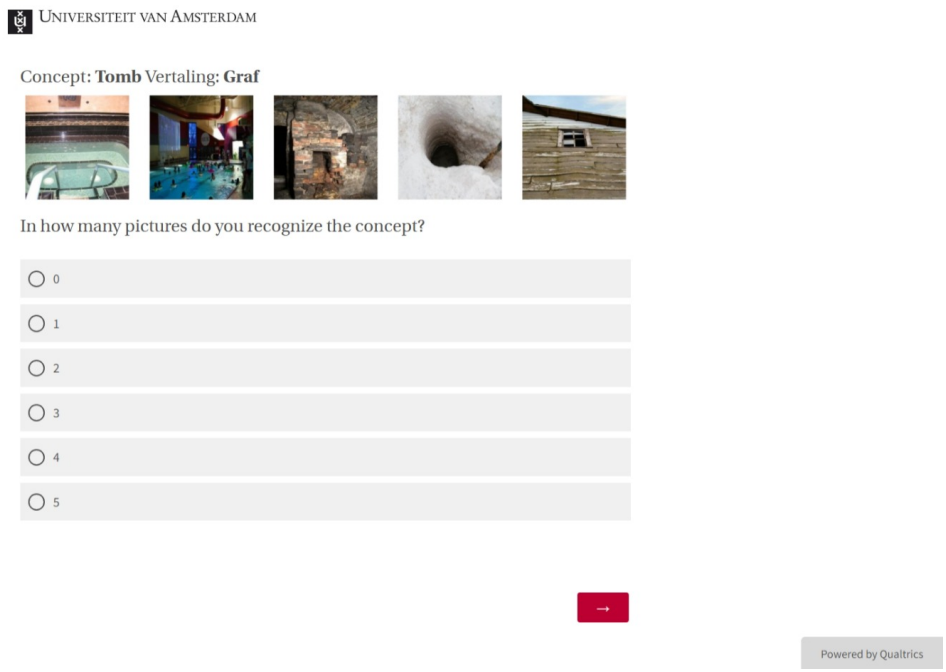


Figure 11: Example of a question in the user study. We present five randomly selected images in which "Tomb" appears among the top five explaining nodes in the local explanation.

A.2 Wilcoxon signed-rank test

To determine whether the differences in user ratings between the two models are statistically significant, we employed the Wilcoxon signed-rank test (Wilcoxon, 1945). This test is a non-parametric alternative to the paired t-test and is particularly suitable when the assumption of normality is violated.

Let X_i and Y_i represent the average user rating given by participant i for the reproduced DN-CBM and our model respectively, at a specific alignment level (high, intermediate, or low). The difference in ratings for participant i is:

$$D_i = Y_i - X_i, \tag{8}$$

where D_i represents whether the participant preferred one model over the other for that alignment level.

For each alignment level, we test:

Null hypothesis (H_0). The average difference in ratings across participants is zero ($\overline{D} = 0$). This implies that there is no significant preference for either model.

Alternative hypothesis (H_A). The average difference is not zero ($\overline{D} \neq 0$), meaning participants systematically rate one model higher.

Assumptions. The Wilcoxon Signed-Rank Test assumes that the two samples are dependent, meaning the data consists of paired samples. Moreover, the distribution of D_i should be approximately symmetric around the median. Lastly, the test requires that the data is ordinal. These assumptions hold for our data, as the ratings range from 0 to 5, and each participant's ratings are dependent.

B Additional qualitative results

B.1 Generalization local explanation

This section provides an extra local explanation for the reproduced DN-CBM and examines the local explanation for ImageNet and CIFAR10.

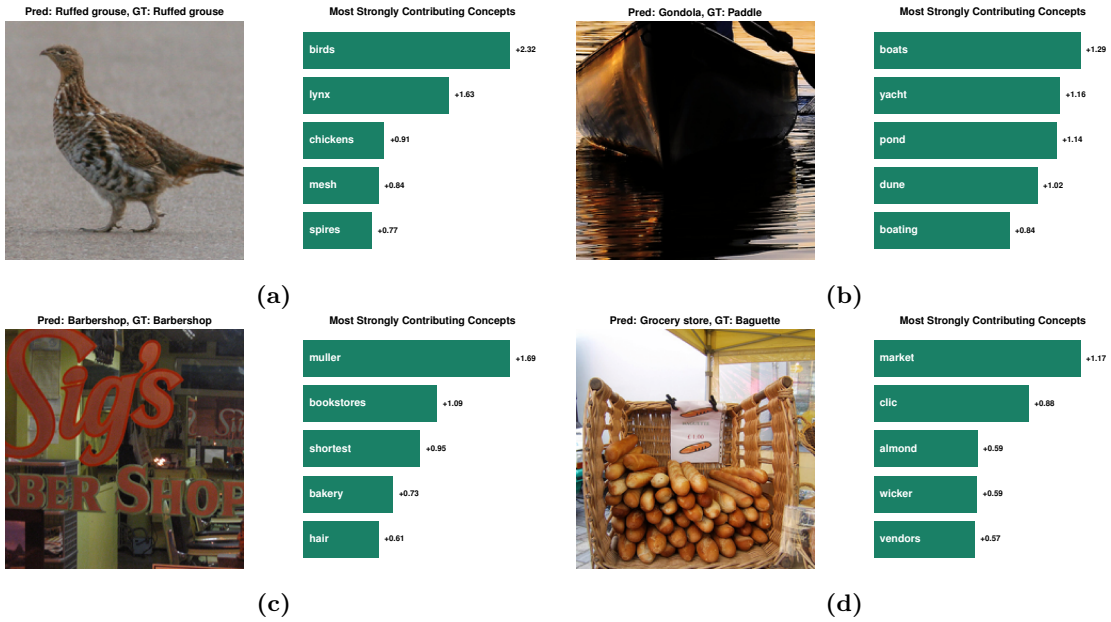


Figure 12: Explaining decisions using the reproduced DN-CBM. We present randomly drawn examples of images from the ImageNet dataset alongside the top concepts contributing to their classification. Figures 12a and 12c are correctly classified, whereas Figures 12b and 12d deviate from the ground truth labels. While the predicted labels for Figure 12b appear reasonable, and most labels for Figure 12a are also interpretable—except for the label "lynx"—the classifications for Figures 12c and 12d are less coherent. Specifically, concepts such as "muller", "bookstores", "shortest", and "bakery" fail to provide a meaningful rationale for the barbershop classification. Furthermore, in Figure 12d, one of the highest contributing concepts is "clic", which lacks a clear semantic interpretation, making it difficult to understand its role in the model's decision-making process.

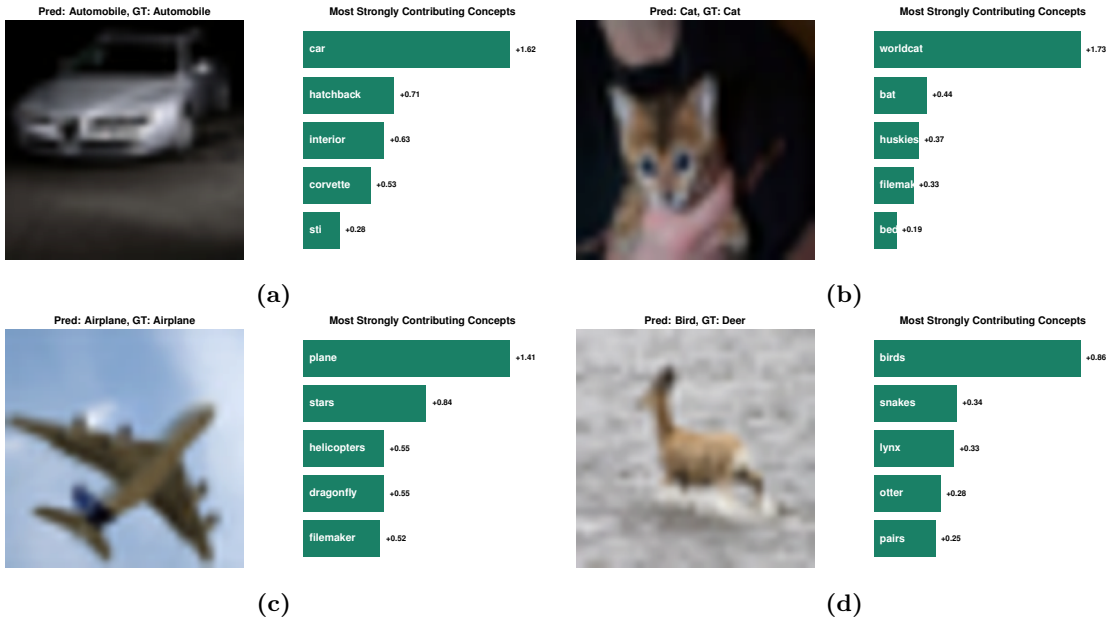


Figure 13: Explaining decisions using the reproduced DN-CBM. We present randomly drawn examples of images from the CIFAR10 dataset alongside the top concepts contributing to their classification. Our observations indicate that Figures 13a, 13b, and 13c are correctly classified, while Figure 13d is misclassified. The explanations for the classifications of "Automobile", "Airplane", and "Cat" appear reasonable, as they include relevant concepts such as "car", "plane", and "worldcat." In the case of Figure 13d, which is misclassified as a "Bird", the contribution of the "birds" node can be interpreted as a plausible factor. However, the inclusion of concepts such as "snakes", "lynx", and "otter" is less intuitive and does not provide a clear rationale for the model's prediction.

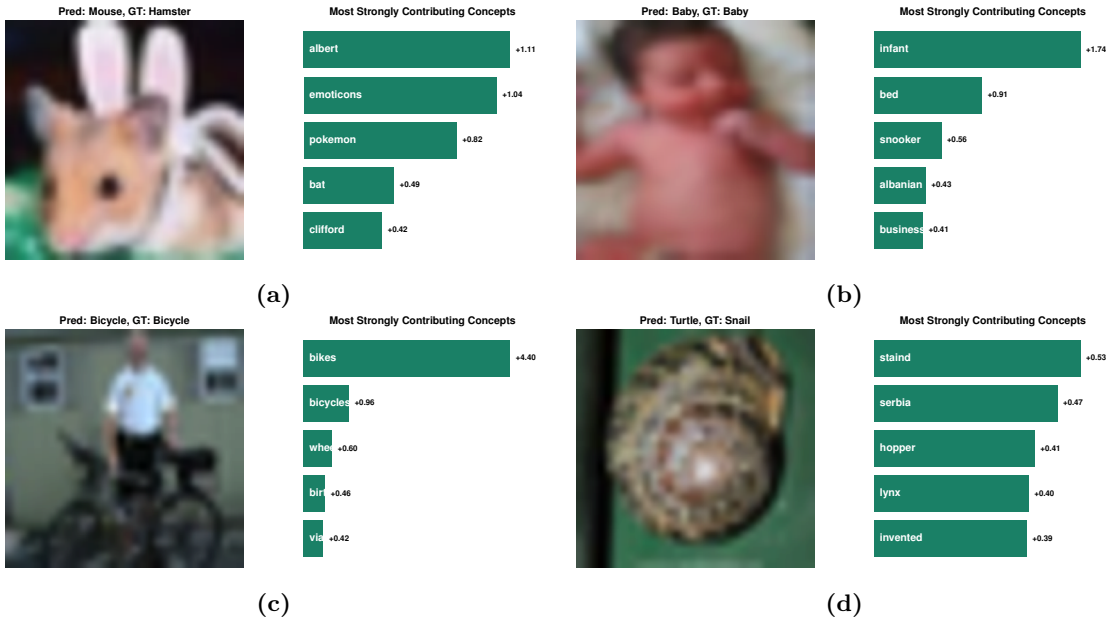
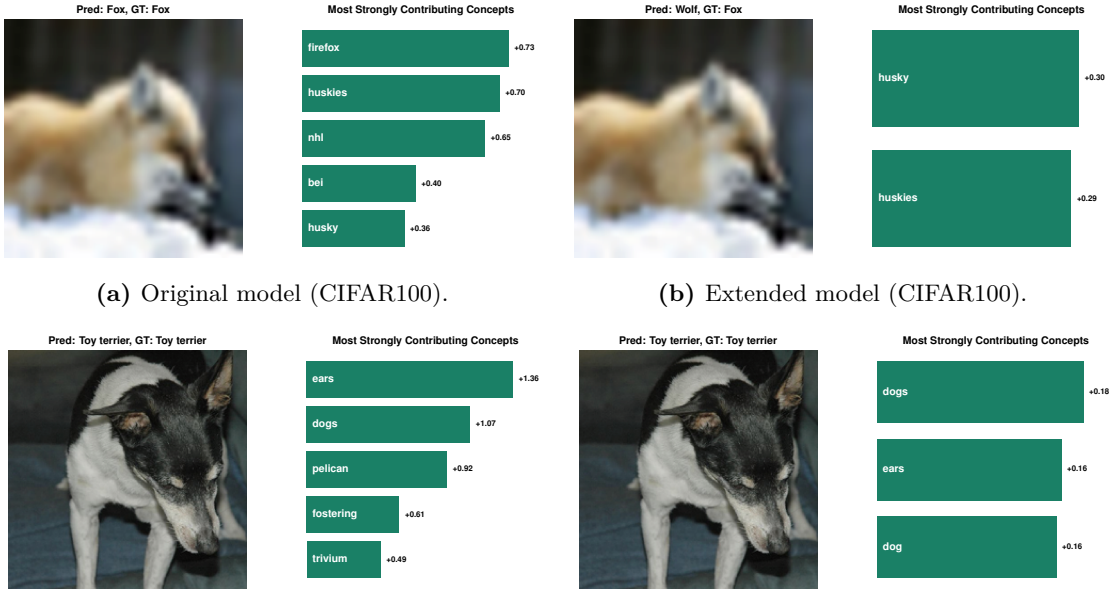


Figure 14: Explaining decisions using the reproduced DN-CBM. We present randomly drawn examples of images from the CIFAR100 dataset alongside the top concepts contributing to their classification. Our analysis reveals that Figures 14b and 14c are correctly classified, whereas Figures 14a and 14d are misclassified. The primary contributing concepts for the classifications of "Bicycle" and "Baby", such as "bikes" and "infant", provide meaningful and interpretable justifications. However, in the case of the misclassification as "Mouse," the concepts "albert", "emoticons", and "pokemon" do not offer a coherent explanation for the model's decision. Similarly, for the misclassification as "Turtle," the contributing concepts "staind" (a band), "Serbia", and "hopper" lack clear semantic relevance, making the prediction difficult to interpret.

B.2 Extended model

In this section, we will provide more explanation of how the extended model behaves. This is done by plotting cosine similarity for different C in Figure 16. Moreover, for generalization, we show local explanations of the original model and our model for CIFAR100 and ImageNet in Figure 15.



(a) Original model (CIFAR100).

(b) Extended model (CIFAR100).

(c) Original model (ImageNet).

(d) Extended model (ImageNet).

Figure 15: Explaining decisions using DN-CBM and our extension. The top row shows an example from CIFAR100 with the predicted class, ground truth, and top contributing concepts for both the original DN-CBM (left) and our extended model with $C = 10^{-4}$ (right). It is observed that our model predicts the incorrect label "wolf", yet the rationale behind this classification is interpretable, with contributing concepts such as "husky" and "huskies". Given that CIFAR100 does not contain a "husky" class, "wolf" is identified as the next closest match. In contrast, the original model correctly predicts the label, with explainable concepts such as "firefox", "husky", and "huskies", although concepts like "nhl" and "bei" are less interpretable. The bottom row compares similar examples for ImageNet. In this case, both models correctly classify the image as "Toy terrier". Our model provides a set of fully interpretable concepts, including "dogs", "ears", and "dog". The original model also identifies several explainable concepts such as "ears", "dogs", and "fostering", but includes less interpretable concepts like "pelican" and "trivium".

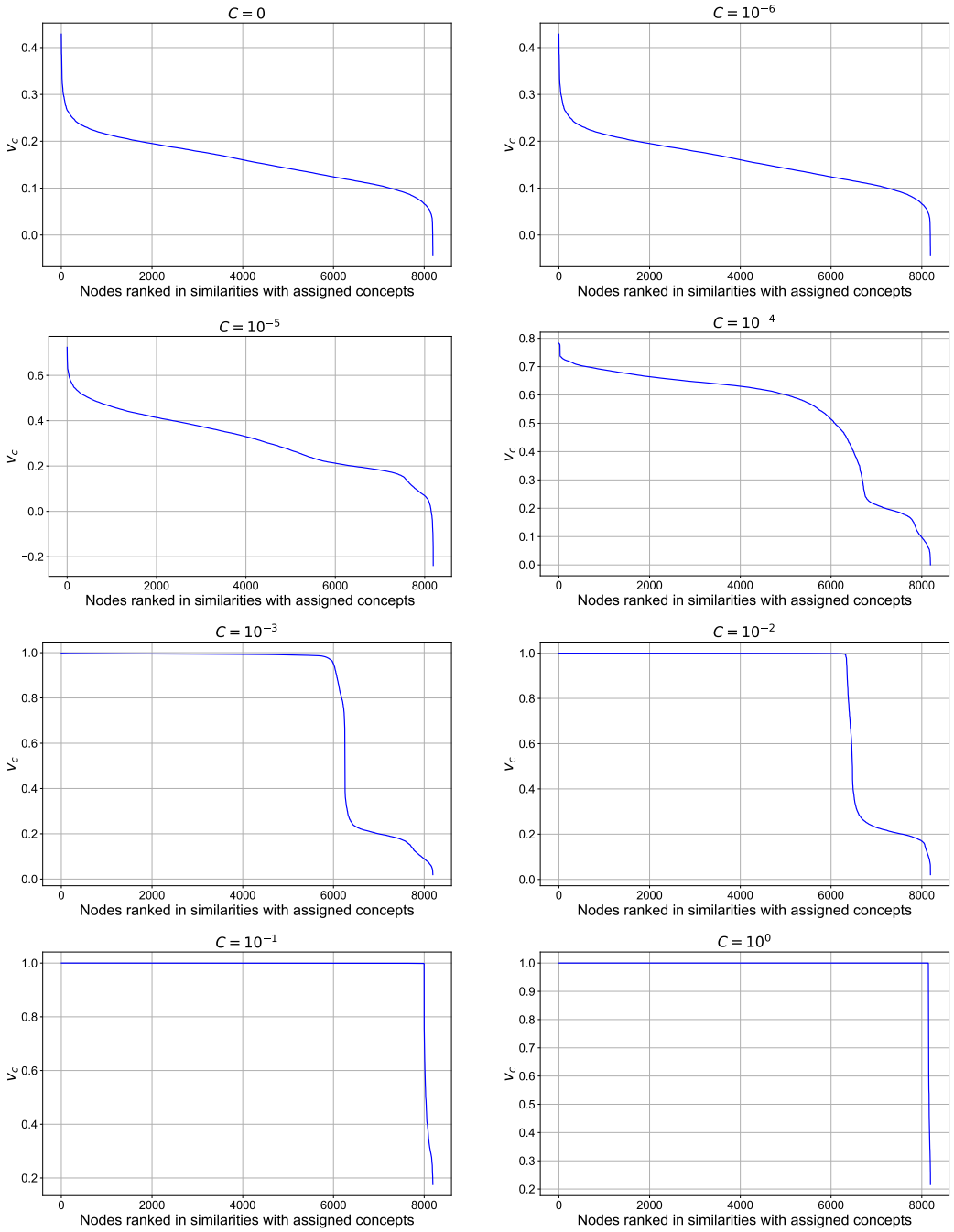


Figure 16: Cosine similarity distribution for different parameters C. The ranked cosine similarity values of the assigned concepts after fine-tuning with varying penalty parameters are presented. $C = 10^{-6}$ yields a similar distribution to the original cosine distribution of $C = 0$. For $C = 10^{-5}$, the shape of the distribution of cosine scores remains similar to that for $C = 10^{0}$, but with a substantially larger range. When $C = 10^{-4}$, a noticeable shift in the distribution emerges, with most similarity values becoming positive and a larger proportion of concepts exhibiting higher cosine similarity. For $C = 10^{-3}$ and $C = 10^{-2}$, the cosine similarity scores reach 1 for the first 6,000 nodes before rapidly declining to 0.2 or lower. For $C = 10^{-1}$ and $C = 10^{0}$, the cosine similarity score goes to 0.2 after 8000 nodes.