Advancing Interpretability of CLIP Representations with Concept Surrogate Model

Nhat Hoang-Xuan
University of Florida

Xiyuan Wei
Texas A&M University

Wanli Xing
University of Florida

Tianbao Yang
Texas A&M University

My T. Thai*
University of Florida

Abstract

Contrastive Language-Image Pre-training (CLIP) generates versatile multimodal embeddings for diverse applications, yet the specific information captured within these representations is not fully understood. Current explainability techniques often target specific tasks, overlooking the rich, general semantics inherent in the representations. Our objective is to reveal the concepts encoded in CLIP embeddings by learning a surrogate representation, which is expressed as a linear combination of human-understandable concepts evident in the image. Our method, which we term EXPLAIN-R, introduces a novel approach that leverages CLIP's learned instance-instance similarity to train a surrogate model that faithfully mimics CLIP's behavior. From the trained surrogate, we derive concept scores for each input image; these scores quantify the contribution of each concept and act as the explanation for the representation. Quantitative evaluations on multiple datasets demonstrate our method's superior faithfulness over the baseline. Moreover, a user study confirms that our explanations are perceived as more relevant, complete, and useful. Our work provides a novel approach for interpreting CLIP image representations, enhancing the user interpretability of representations and fostering more trustworthy AI systems.

1 Introduction

The CLIP model [1] exemplifies the success of representation learning, which aims to create general-purpose embeddings applicable to a multitude of downstream tasks. These representations have become integral to numerous applications, including text-to-image generation (e.g., Stable Diffusion [2]), Large Multimodal Models [3, 4, 5], and open-set object detection [6, 7]. The widespread adoption and diverse applications of learned representations like CLIP emphasize the need to understand their underlying semantics. Therefore, effective explanation methods are required to characterize the information encoded within these representations. As these learned embeddings are inherently task-agnostic and depend only on the input data, their explanations should therefore describe the information they hold, regardless of any specific downstream application.

Explaining general-purpose representations presents a significant challenge for traditional eXplainable AI (XAI) techniques. Methods such as GradCAM [8], LIME [9], and Integrated Gradients [10] are fundamentally designed to explain model predictions for specific tasks or classes. For instance, GradCAM utilizes class-specific gradients to generate activation maps, while LIME approximates local decision boundaries. Critically, this inherent focus on task-specific predictions prevents their generalization to explaining the underlying representations. A more recent line of work [11, 12] approaches

*Corresponding author. Email: mythai@cise.ufl.edu

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

this challenge using matrix-factorization techniques, such as Non-negative Matrix Factorization (NMF), to find low-rank decompositions of embeddings. While such methods can produce bases that effectively reconstruct the original embeddings, the resulting basis is not inherently interpretable and requires further analysis to discern its meaning.

To address these limitations, we propose EXPLAIN-R (EXPLAIN-Representations), a novel method designed to generate interpretable conceptual explanations for CLIP image representations, independent of downstream tasks. By design, EXPLAIN-R discovers human-understandable concepts from each input image. Our method utilizes these discovered concepts to construct a surrogate representation, which is then trained to mimic the behavior of the original CLIP embeddings. Post-training, our method computes concept scores that quantify the influence of each discovered concept on an input's representation. Our contributions are as follows:

- We introduce EXPLAIN-R, a novel method for learning a faithful surrogate representation of CLIP image embeddings, which is formed by linearly combining interpretable concepts, and propose a theoretically-motivated algorithm for its training.
- Extensive quantitative experiments across multiple datasets validate the faithfulness of EXPLAIN-R, demonstrating that the surrogate representation accurately preserves the predictive behavior of the original CLIP model.
- We establish via a user study that EXPLAIN-R produces explanations considered relevant to the input image, sufficiently complete to explain the model's capabilities, and useful for overall model comprehension.

2 Related Work

Interpreting CLIP vision encoder. Existing methodologies for explaining CLIP models can be broadly categorized into several distinct lines of work, each adopting a unique approach. Pixel attribution techniques [13, 14, 15, 16] identify input regions influencing model outputs via heatmaps. Being visually accessible, they can explain the "where" but fall short in explaining the "what" [11]. For instance, a visual explanation highlighting the poodle in an image does not clarify if the model recognizes a "poodle" specifically or merely a "dog". Mechanistic interpretability methods [17, 18, 19] pursue a different strategy, associating human concepts with internal model components (e.g., neurons, attention heads). Concept Bottleneck Model [20, 21, 22, 23] aims to substitute the opaque representation with an explicit concept layer. However, CBMs typically focus on task-specific predictions rather than explaining the versatile representation that is applicable to multiple tasks. In contrast, our method focuses on generating interpretable concept-based explanations for these abstract image representations directly. Furthermore, our method can be extended to provide concept explanations for zero-shot tasks without additional training.

Concept-based representation explanations. One specific line of work aims to find a concept basis spanning the representation space using matrix decomposition [12, 11, 24, 25]. These methods typically factorize dataset embeddings via SVD or NMF into a smaller basis and corresponding coefficient matrices. Although capable of achieving low reconstruction error, the resulting basis vectors are not inherently interpretable by design. Their semantic meaning needs to be inferred through subsequent analysis, like visualizing top activating inputs [12, 11]. SpLiCE [26], the most relevant prior work, decomposes CLIP image features into sums of text features using the CLIP text encoder. However, imperfections in the text encoder can lead to spurious or unrelated concepts (as observed in Section 4.2), potentially causing confusion for the user. Our method distinguishes itself from these approaches through its design for inherent interpretability: the discovered concepts are, by construction, explicitly linked to the input image, and do not rely on the text encoder. Furthermore, our proposed similarity-matching training strategy yields superior faithfulness while preserving the surrogate model's simplicity and interpretability.

3 Methodology

This section details EXPLAIN-R, a novel method for interpreting CLIP image representations through a learned concept surrogate model. EXPLAIN-R constructs a surrogate embedding from a linear combination of interpretable concepts grounded in the image. The process encompasses three key

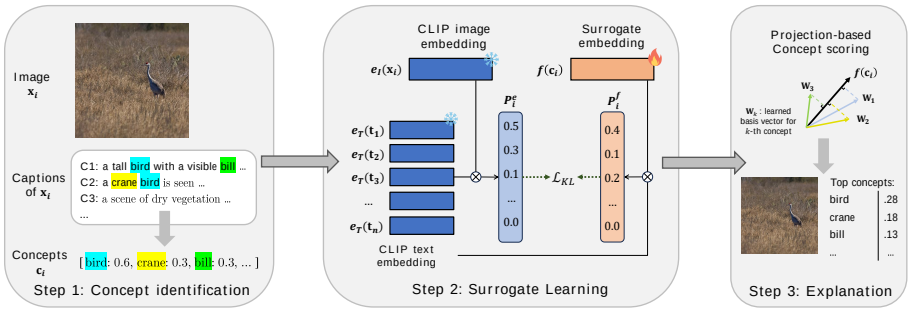

Figure 1: EXPLAIN-R methodology overview. (1) **Concept Identification**: Image captions are processed into a concept vector $\mathbf{c}_i$. (2) **Surrogate Learning**: The surrogate f, with parameters $\mathbf{W}$,$\mathbf{b}$, is trained to mimic the CLIP image embedding by matching image-to-text (P_i^f,P_i^e) and text-to-image (not visualized) similarity distributions using a KL divergence loss. Symbol $\bigotimes$ denotes cosine similarity followed by softmax. (3) **Explanation**: Explanatory concept scores for the representation of the image $\mathbf{x}_i$ are obtained by projecting the rows of $\mathbf{W}$ onto $f(\mathbf{c}_i)$ and weighing by $\mathbf{c}_i$.

steps, visualized in Fig. 1: (1) discovery of interpretable concepts from the image dataset to build a vocabulary C; (2) learning a surrogate model that transforms these image-specific concepts into an embedding that maintains similarity with text embeddings; and (3) computation of concept scores to quantify each concept's contribution, thus providing an explanation for the image's CLIP representation.

Notations. Let $\{(\mathbf{x}_i, \mathbf{t}_i)\}_{i=1}^n$ be an image-text dataset. CLIP's image encoder $e_I(\cdot)$ and text encoder $e_T(\cdot)$ yield L_2-normalized representations in $\mathbb{R}^d$. For each image $\mathbf{x}_i$, we obtain the concept vector $\mathbf{c}_i \in \mathbb{R}^{|C|}$ and form the augmented dataset $\mathcal{S} = \{(\mathbf{x}_i, \mathbf{c}_i, \mathbf{t}_i)\}$. The surrogate model $f : \mathbb{R}^{|C|} \to \mathbb{R}^d$ is defined as $f(\mathbf{c_i}; \mathbf{W}, \mathbf{b}) = \sigma(\mathbf{c}_i\mathbf{W} + \mathbf{b})$. Here $\mathbf{W} \in \mathbb{R}^{|C|\times d}$ is a matrix where rows are the (learned) concept basis vectors, $\mathbf{b} \in \mathbb{R}^d$ is a bias, and $\sigma(\cdot)$ is the L_2 normalization operator. The surrogate function linearly transforms the concepts into the embedding space, which is then normalized.

3.1 Captions-based concept identification

Explaining generic, task-agnostic image representations with open-ended concepts necessitates concept identification methods that are scalable, task-independent, and yield intuitive, human-centric concepts. Our work is directed towards developing such an approach.

However, existing concept identification paradigms are largely incompatible with achieving these combined objectives. For instance, (i) supervised concept detectors [27, 28, 29] require labeled data for predefined concepts, limiting their scalability for open-ended concept discovery. (ii) LLM-based concept generation [30, 22, 21] is typically designed to yield task-specific concepts (e.g., for class discrimination), which lacks the task-independence required for general representation understanding. (iii) Furthermore, directly employing CLIP's alignment scores [31, 23] can produce unintuitive concept associations, as CLIP's training objective aligns images with full sentences or captions rather than prioritizing the descriptive accuracy of individual words or concepts in a human-like manner.

To obtain explanations that are intuitive, broadly applicable, and thus more aligned with our objectives, we advocate sourcing concepts from data reflecting typical human image descriptions. Image captions are ideal as they tend to describe a diverse range of image features (such as objects, attributes, and actions). This descriptive characteristic means that concepts derived from captions are largely purpose-neutral, making them well-suited for explaining task-agnostic representations. This caption-based approach also offers benefits such as transparency into how concept scores are derived. While human-annotated captions are preferred, those from advanced models [3, 4] (trained to emulate human styles via metrics correlated with human judgment [32, 33]) offer a scalable alternative. Based on these ideas, we formulate the concept representation for an image $\mathbf{x}_i$ as a vector $\mathbf{c}_i \in [0, 1]^{|C|}$, where each component $\mathbf{c}_{i,k}$ quantifies the prominence of the k-th concept from a global vocabulary C within human-like descriptions of $\mathbf{x}_i$, empirically measuring its descriptive likelihood.

The construction of the vocabulary C and the concept vectors $\mathbf{c}_i$ proceeds systematically. First, we compile an initial, comprehensive vocabulary by extracting all nouns, verbs, and adjectives from the image captions across the dataset. This raw vocabulary is then pruned to create C by removing overly frequent terms, as these are often less discriminative, guided by an adjustable threshold to control the desired concept sparsity (i.e., the average number of active concepts per image). Subsequently, for each image $\mathbf{x}_i$, its concept vector component $\mathbf{c}_{i,k}$ is computed as z/m, where m is the number of captions associated with $\mathbf{x}_i$, and z is the frequency of the k-th concept (from C) within those m captions. The resulting concept vector, $\mathbf{c}_i$, reflect the qualities of an image that are emphasized in typical human descriptions.

3.2 Learning interpretable surrogate representation

Previous works [26, 11] on finding concept-based explanations focus on high-fidelity numerical reconstruction of the embeddings. For example, the baseline SpLiCE attempts to maximize the cosine similarity between the reconstructed and the original embedding. While a numerically identical reconstruction would entail perfect faithfulness, achieving this ideal is often infeasible in the presence of interpretability constraints like sparsity. Moreover, given that CLIP is trained to capture bimodal relationships, unimodal similarities can behave unintuitively, as pointed out in [34]. This makes optimizing for the similarity between an original embedding and its reconstruction a less reliable method to capture CLIP's bimodal behavior.

Our proposed approach alternatively focuses on what is learned with CLIP's contrastive objective. We posit that CLIP's contrastive training, which requires distinguishing an image's paired caption from numerous alternatives, leads to a rich similarity distribution over texts, and vice versa. This pattern of similarity to other inputs, we argue, offers insights into CLIP's behavior that previous methods overlooked. Consequently, our surrogate model, f, is trained to reproduce this distribution of similarities, without specific downstream task supervision.

We now formally state the surrogate learning problem. The distribution of image-to-text similarities of the i-th image for the original model is defined as:

$$P_i^e(j) = \frac{\exp(e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_j)/\tau)}{\sum_k \exp(e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_k)/\tau)}. \tag{1}$$

where τ is the temperature. The similarity distribution for the surrogate is analogously given as:

$$P_i^f(j) = \frac{\exp(f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)/\tau)}{\sum_k \exp(f(\mathbf{c}_i)^\top e_T(\mathbf{t}_k)/\tau)}. \tag{2}$$

Likewise, the text-to-image similarity distribution Q_j^e and Q_j^f are obtained by modifying the normalizing factor in the denominator of (1) and (2) to sum over image indices. Faithfulness in a surrogate model necessitates that its output similarity distributions, P_i^f and Q_j^f, closely mirror those of the target model, P_i^e and Q_j^e, as these distributions determine predicted probabilities and outputs. To promote such faithfulness, we propose minimizing the Kullback-Leibler (KL) divergence between these corresponding distribution pairs. This makes the distributions more similar by reducing the information available to distinguish between them [35]. In this case, the KL divergence also has an intuitive interpretation: the term $D_{KL}(P_i^e \parallel P_i^f)$, for instance, is the average difference of log probability between P_i^e and P_i^f, weighted by P_i^e. Taking the expectation of the divergence over the dataset, we obtain the final loss function:

$$\mathcal{L}(\mathbf{W}, \mathbf{b}) = \frac{1}{2n}\left(\sum_{i=1}^{n} D_{KL}(P_i^e \parallel P_i^f) + \sum_{j=1}^{n} D_{KL}(Q_j^e \parallel Q_j^f)\right). \tag{3}$$

We note that this loss function have been used in conjunction with various other mechanisms (weak/strong augmentation, momentum encoders, etc.) for self-supervised learning from scratch [36]. However, our motivation (encouraging faithfulness) and use case (XAI) is completely different, and to the best our ability, the idea of training a surrogate by distilling instance-instance similarity has not

Top concepts:	
bird	.28
crane	.18
dry (a.)	.13
tall	.08
bill (n.)	.05
prey	.05
vegetation	.05

Top concepts:	
microwave (n.)	.40
counter	.08
sign	.08
stainless	.06
microwave (v.)	.05
tray	.04
metal	.04

Top concepts:	
barometer	.15
clock	.13
change (n.)	.11
meter	.08
gauge	.06
dial	.06
fair (n.)	.05

Figure 2: Example images from ImageNet and their top concepts, ranked by CLIP representation concept score v_i. In case of ambiguity, the concepts are annotated with their part of speech. The images are labeled as *crane*, *microwave*, and *barometer* respectively. We visualize the top seven concepts for brevity; our surrogate representations contain 15-20 concepts.

been explored for XAI. Finally, note that our choice of a linear surrogate promotes interpretability by allowing computation of projection-based concept scores as in Section 3.4. It is also consistent with assumptions in existing works [12, 11], including the baseline [26].

3.3 Algorithm for Surrogate Learning

In this section, we present our algorithm for optimizing (3). As we will show, naively using mini-batch optimization for (3) leads to a biased gradient. This is a common problem for contrastive losses and their variants, which existing works circumvent by either using a larger batch size (for example, OpenAI uses a batch size of 32,768 on 256 GPUs to train the CLIP model [1]), using a queue of past samples [37, 36], or using non-contrastive losses [38, 39]. Instead, our solution to correct this bias does not require large batch size and it is theoretically motivated.

We will focus only on $D_{KL}(P_i^e \parallel P_i^f)$ since the procedure for optimizing $D_{KL}(Q_j^e \parallel Q_j^f)$ can be derived analogously. From the definition of KL divergence, the first part of (3) can be equivalently written as (c.f. Appendix B.2 for more detailed derivation):

$$\frac{1}{|\mathcal{S}|^2}\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}}\frac{1}{g(e_I,e_T;\mathbf{x}_i,\mathbf{t}_j,\mathcal{S})}\cdot\log g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S}), \tag{4}$$

where $g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S}) = \frac{1}{|\mathcal{S}|}\sum_{k\in\mathcal{S}}\exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_k)-f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right)$. The gradient w.r.t. f is then given by:

$$\frac{1}{|\mathcal{S}|^2}\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}}\frac{1}{g(e_I,e_T;\mathbf{x}_i,\mathbf{t}_j,\mathcal{S})}\cdot\frac{1}{g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S})}\cdot\nabla g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S}).$$

At each iteration, we only have access to a mini-batch of triplets $\mathcal{B}=\{(\mathbf{x}_i,\mathbf{c}_i,\mathbf{t}_i)\}$ of batch size B. The obtained mini-batch gradient estimator is simply replacing $\mathcal{S}$ with $\mathcal{B}$ in the above equation. However, due to the non-linearity of the reciprocal function $1/\cdot$, the expectation over $\mathcal{B}$ is not equal to the true gradient. Thus the mini-batch estimator is a biased estimator of the true gradient. To solve this problem, we use two moving average sequences u and v to approximate $g(e_I,e_T;\mathbf{x}_i,\mathbf{t}_j,\mathcal{S})$ and $g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S})$ respectively:

$$u_{t+1,i} = (1-\gamma_1)u_{t,i} + \gamma_1\frac{1}{|\mathcal{B}_t|}\sum_{k\in\mathcal{B}_t}\exp\left(\frac{e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_k)}{\tau}\right), \tag{5}$$

$$v_{t+1,i} = (1-\gamma_2)v_{t,i} + \gamma_2\frac{1}{|\mathcal{B}_t|}\sum_{k\in\mathcal{B}_t}\exp\left(\frac{f(\mathbf{x}_i)^\top e_T(\mathbf{t}_k)}{\tau}\right), \tag{6}$$

where $\gamma_1,\gamma_2\in(0,1]$ are two hyperparameters. Then we can approximate $g(e_I,e_T;\mathbf{x}_i,\mathbf{t}_j,\mathcal{S})$ and $g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S})$ using $u_{t+1,i}/\exp\left(e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_j)/\tau\right)$ and $v_{t+1,i}/\exp\left(f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)/\tau\right)$, respectively. The benefit of using the moving average technique is that now we can guarantee that the distance between the estimators and the true values ($g(e_I,e_T;\mathbf{x}_i,\mathbf{t}_j,\mathcal{S})$ and $g(f,e_T;\mathbf{c}_i,\mathbf{t}_j,\mathcal{S})$) diminishes to 0 in expectation, instead of remaining at a constant level $\mathcal{O}(1/B)$ if the moving average technique is not used [40, 41]. We present the pseudocode and full derivation in Appendix B.2.

Concepts for 'mixing bowl'	
clay	.09
bowl	.05
pottery	.04
wheel	.02
potter	.02
spin	.01
hand	.01

(a) Image of class "potter's wheel" misclassified as "mixing bowl" by CLIP. Note the significant contribution of the *bowl* concept, which is a main predictor for the "mixing bowl" class, as seen in Fig. 6.

Concepts for 'barometer'	
barometer	.060
clock	.047
meter	.025
gauge	.022
dial	.020
change (n.)	.016
mechanical	.013

Concepts for 'wall clock'	
clock	.054
barometer	.042
meter	.020
change (n.)	.019
gauge	.016
dial	.016
mechanical	.012

(b) Class concept scores of the same image for different classes. The contribution of concepts varies in strength depending on the target class.

Figure 3: Class concept attribution score for ImageNet images.

3.4 Interpreting representations with projection-based concept scores

This section introduces a method for attributing the CLIP image representations to their underlying concepts. Standard attribution methods, such as SHAP [42], typically rely on the notion of a prediction to operate. In contrast, our approach leverages the linearity of the explainer f to efficiently computes concept attribution scores $v_i(k)$. Building upon this, we further develop a cross-modal attribution score $v_{i,j}(k)$ to quantify how concept k modulates the similarity between image i and text j.

We begin by expanding the functional form of the surrogate $f(\mathbf{c}_i) = \sigma(\mathbf{W}\mathbf{c}_i + \mathbf{b})$, which yields:

$$f(\mathbf{c}_i) = \sigma\left(\sum_{k \in C} \mathbf{c}_{i,k}\mathbf{W}_k + \mathbf{b}\right), \tag{7}$$

where $\mathbf{W}_k$ is the column of $\mathbf{W}$ corresponding to the embedding of the k-th concept. Equation (7) shows that each concept contributes a term $\mathbf{c}_{i,k}\mathbf{W}_k$ to the pre-normalized representation. A term is considered to have a greater influence if it aligns with the direction of the final representation $f(\mathbf{c}_i)$. We quantify this alignment with the scalar projection of $\mathbf{c}_{i,k}\mathbf{W}_k$ onto $f(\mathbf{c}_i)$, which we define as the initial score: $\tilde{v}_i(k) = \mathbf{c}_{i,k}\mathbf{W}_k^\top f(\mathbf{c}_i)$. Note that summing the projection recovers the pre-normalization magnitude, i.e., $\sum_{k \in C} \tilde{v}_i(k) + \mathbf{b}^\top f(\mathbf{c}_i) = \|\mathbf{W}\mathbf{c}_i + \mathbf{b}\|$. To obtain scores reflecting the relative contribution of each concept while ignoring the bias term, we normalize them by $\lambda_i = \left(\|\mathbf{W}\mathbf{c}_i + \mathbf{b}\| - \mathbf{b}^\top f(\mathbf{c}_i)\right)^{-1}$. We define the final attribution score $v_i(k)$ as:

$$v_i(k) := \lambda_i \tilde{v}_i(k) = \lambda_i \mathbf{c}_{i,k}\mathbf{W}_k^\top f(\mathbf{c}_i). \tag{8}$$

This score $v_i(k)$ quantifies the normalized contribution of the k-th concept, based on the linear relationship between the k-th concept embedding and the final surrogate representation $f(\mathbf{c}_i)$, such that $\sum_{k \in C} v_i(k) = 1$.

Cross-model concept scores. We adapt this projection-based approach to attribute the cross-modal similarity between the i-th image and the j-th text to individual concepts k. Recall that the text embedding is $e_T(\mathbf{t}_j)$, and the similarity is computed as $s_{i,j} = f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)$. We aim to understand how each concept k influences the similarity $s_{i,j}$. Analogous to before, we project the contribution vector onto the direction of the text embedding $e_T(\mathbf{t}_j)$ to obtain the unnormalized score $\tilde{v}_{i,j}(k) = \mathbf{c}_{i,k}\mathbf{W}_k^\top e_T(\mathbf{t}_j)$. The final cross-modal attribution score is then:

$$v_{i,j}(k) = \lambda_{i,j}\mathbf{c}_{i,k}\mathbf{W}_k^\top e_T(\mathbf{t}_j), \tag{9}$$

in which the normalization factor $\lambda_{i,j} = \frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{(W\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}$ makes the scores $v_{i,j}(k)$ sum to $s_{i,j}$ over all concepts.

Ours top concepts:	
bird	.28
crane	.18
dry (a.)	.13
tall	.08
bill (n.)	.05
prey	.05
vegetation	.05

SpLiCE top concepts:	
great blue	.11
crane	.07
individual	.04
stalk	.03
wildlife	.03
banded	.03
pheasant	.03

Ours top concepts:	
microwave (n.)	.40
counter	.08
sign	.08
stainless	.06
microwave (v.)	.05
tray	.04
metal	.04

SpLiCE top concepts:	
microwave	.08
cleaning machine	.07
push button	.04
classical music	.04
bilingual	.03
sounds	.03
fax	.02

Ours top concepts:	
screw	.44
stainless	.12
bolt	.08
fastener	.07
surface	.07
screw (v.)	.06
metal	.03

SpLiCE top concepts:	
manufacturers	.13
screwed	.06
zinc alloy	.06
implant	.05
bolts	.04
cheap key	.04
repair kit	.04

Ours top concepts:	
sand	.37
castle	.18
beach	.16
kid	.09
sandy	.05
try	.05
reach	.02

SpLiCE top concepts:	
sand	.13
team building	.06
archaeological	.05
geology	.03
interacting	.02
small group	.02
bullying	.01

Figure 4: Comparison between our concepts and the baseline. Images are from the SUN397 (lower right) and ImageNet (remaining) datasets. The top-left image depicts a crane (bird), while great blue and pheasant are other species of birds. We show only the top seven concepts for each image. Concept scores from our method sum to one for each image; SpLiCE's scores do not have this property.

4 Experiments

We conduct experiments using the CLIP ViT-B/32 [1] model to demonstrate that explanations generated by our method are faithful, interpretable, and useful for users.

Datasets. We use the COCO 2017 [43] validation set, Flickr30k [44], SUN397 [45] test set, and ImageNet validation set [46] to study the CLIP model's behavior. These datasets cover a variety of image themes, including objects in context, internet images, scene understanding, and object recognition. Details and experiments on more datasets can be found in the Appendix.

Setup. For datasets that contain captions (COCO and Flickr30k), we use the captions as the text $\mathbf{t}_i$. For image-only datasets, we use the BLIP-2 [3] OPT 2.7B model fine-tuned on COCO to obtain 10 captions per image. Then, we perform concept identification (Section 3.1) to obtain 15-20 concepts per image. We train the surrogate with Algorithm 1 for 150 epochs, with batch size 1024, default temperature $\tau = 0.1$, and $\gamma_1 = \gamma_2 = 0.9$. The optimizer used is AdamW with learning rate 10^{-3}. All experiments are performed on a single A100 GPU. For the baseline SpLiCE [26], we use the official implementation linked in the paper. We obtain a similar sparsity (average number of concepts per image) to our method by setting the l_1 penalty for each dataset to facilitate a fair comparison. The vocabulary used for SpLiCE is based on LAION and has a size of 15,000.

4.1 Explanation faithfulness

Emphasizing our focus on post-hoc explanation, we assess the surrogate's faithfulness quantitatively by comparing the predictions made using the surrogate representation against those of the original image embedding. Faithfulness is measured using conventional metrics (e.g., accuracy), with the target model's prediction treated as the ground truth. We focus on zero-shot tasks as opposed to linear probing, since the former relies directly on the image representations and does not depend on a particular trained probe's behavior. For zero-shot classification, we report the accuracy. For zero-shot retrieval tasks, we report the rsum metrics [47, 48] (defined as R@1 + R@5 + R@10) as a concise summary, instead of individual Recall@K (R@K) values. Detailed retrieval metrics and performance metrics on the original tasks are provided in Appendix A.1 and A.2.

Table 1: Faithfulness of our method and the baseline on zero-shot retrieval and zero-shot classification. Higher values are better. The means and 2σ intervals are computed over five runs. We note that the official implementation of SpLiCE is deterministic.

Method	COCO val	Flickr30k	SUN397	ImageNet val
SpLiCE [26]	393.18	381.41	59.78	51.33
Ours	**482.94** $\pm$ 4.08	**476.47** $\pm$ 3.21	**62.77** $\pm$ 0.22	**53.38** $\pm$ 0.21

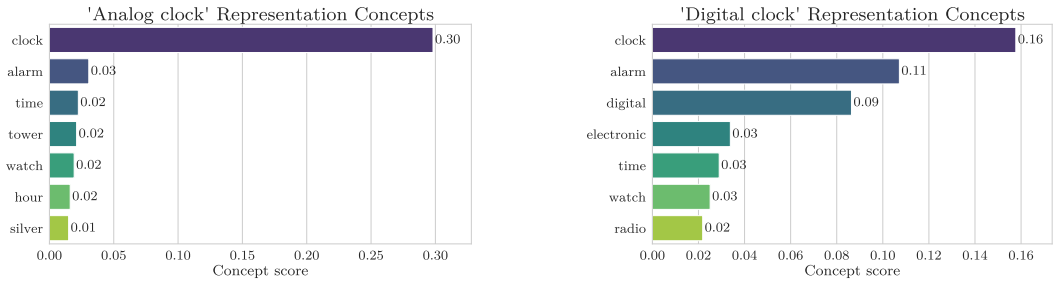

Figure 5: Average representation concept scores for images in ImageNet "Analog clock" and "Digital clock" classes.

As shown in Table 1, our surrogate model consistently outperforms the baseline in faithfulness across diverse datasets and tasks, indicating its effectiveness in replicating the target model's behavior. This success demonstrates that faithful surrogate concept representations can be achieved without solely relying on the minimization of reconstruction mean-squared error, a strategy prevalent in the baseline and other methods [11, 20, 12].

4.2 Qualitative assessment of representation concepts

We visualize the representation concepts generated by our methods and the baseline to evaluate their relevance to the image content. We show several images from ImageNet and SUN397 and their corresponding explanations in Fig. 4, with more details and examples in Appendix A.3. Our method predominantly discovers concepts highly pertinent to the image content. Our explanations suggest that the CLIP image representation attends to both the primary subject and its surrounding context; the top-ranked concept often identifies the main object with high score, while others capture secondary objects or specific attributes of the primary subject. Conversely, the baseline (SpLiCE) sometimes produces concepts that, while potentially related, are not depicted (e.g., "great blue," "cleaning machine," "geology"), or are entirely unrelated (e.g., "classical music," "cheap key").

Representation Concept score Histograms. Per-image concept scores (exemplified in Fig. 4) can be aggregated to offer insights into how the CLIP model represents classes of images in a dataset. To illustrate, Figure 5 presents the aggregated top seven concept scores for two related classes: "analog clock" and "digital clock". The analysis reveals distinct patterns: "analog clock" image embeddings are predominantly characterized by the "clock" concept, whereas "digital clock" images show a more uniform concept distribution. Notably, while several top concepts like "alarm", "time", and "watch" are common to both, the concepts "digital" and "electronic" effectively distinguish the two classes.

4.3 Class prediction analysis

This section illustrates how class concept scores ($v_{i,j}$) can be applied to understand model predictions for a specific class, enabling misprediction analysis. To reiterate for clarity in this context, these scores, $v_{i,j}$, quantify how each concept from an image i contributes to its similarity $s_{i,j}$ with class j, and are defined such that their sum equals $s_{i,j}$. This class-specific nature contrasts with task-agnostic

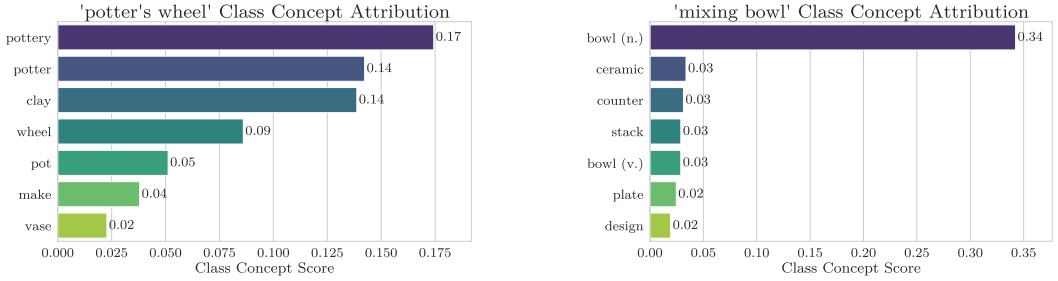

Figure 6: Average class concept scores for ImageNet classes "potter's wheel" and "mixing bowl".

representation concept scores (v_i), which describe the overall image content. Figure 3b analyzes the class concept scores ($v_{i,j}$) for a specific image; the task-agnostic representation concepts (v_i) for this same image are depicted in the rightmost panel of Figure 2. For this image, the "barometer" class prediction is primarily driven by the "barometer" concept (rank 1) followed by "clock" (rank 2), while the "wall clock" class prediction prioritizes "clock" (rank 1) over "barometer" (rank 2). This shows that the contribution of individual concepts to $v_{i,j}$ varies significantly with the class.

Misprediction analysis. Analysis of class concept scores for mispredicted images can reveal concepts responsible for incorrect classifications. For instance, Figure 3a presents an ImageNet image of class "potter's wheel" that both CLIP and the surrogate misclassified as "mixing bowl" (a bowl used for cooking). The concepts contributing to this prediction are shown alongside the image. To determine the typical concepts used for each class, we aggregated the class concept scores $v_{i,j}$ for both "potter's wheel" and "mixing bowl", visualizing the results in Figure 6. This combined information suggests that the "bowl" concept is the primary driver of the misclassification. Specifically, the aggregation indicates that "bowl" is, on average, the dominant concept for predicting "mixing bowl", while it is notably absent from the top concepts associated with "potter's wheel".

4.4 User study

We performed a small scale user study to evaluate the interpretability of our explanations (results in Fig. 7), largely following the protocol of [26] but with modified criteria. Users were shown 20 random ImageNet images and the top ten concepts from EXPLAIN-R and SpLiCE. The evaluation centered on three criteria: (1) **Relevance**: the degree to which concepts pertain to the input image; (2) **Completeness**: the extent to which explanations reflect the semantic richness and task-agnostic characteristics of the image representations; and (3) **Utility**: the perceived usefulness of each method for understanding the model's behavior. Responses were captured using a 5-point Likert scale. User study findings reveal a significant preference for our explanations across all evaluated criteria. The statistical significance of these results was established via a one-sided t-test, employing a significance level of $p < 0.05$. Additional details of the user study protocol are available in the Appendix.

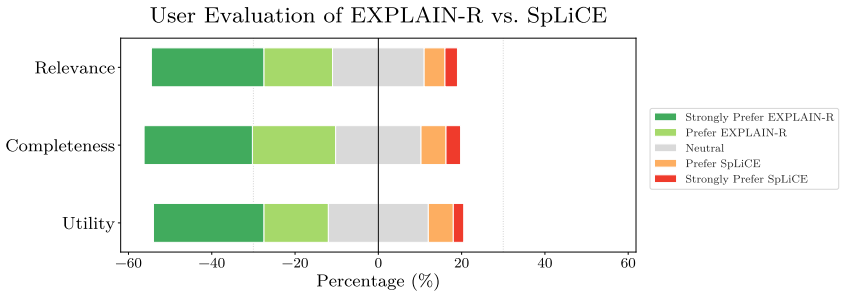

Figure 7: User study results comparing representation explanations generated by EXPLAIN-R and SpLiCE. Overall, users indicated a clear preference for EXPLAIN-R's explanations across all three evaluated criteria.

5 Discussion

In this work, we show that the instance similarities learned by CLIP can be utilized to generate concept explanations for CLIP image embeddings. Quantitative experiments demonstrated EXPLAIN-R's superior faithfulness over baselines in preserving original model predictions, while a user study confirmed its explanations are more relevant, complete, and useful for model understanding. These results suggest that our similarity-matching approach offers a promising direction for developing more faithful and human-aligned explanations for general-purpose representations.

Limitations Our approach, like prior studies [26, 11, 12], assumes that concepts interact linearly in the embedding space. While EXPLAIN-R significantly improves faithfulness over the baseline, perfect fidelity remains challenging. This gap may stem from several factors: the target model might learn concepts that are not easily captured by concise textual descriptions, or there are some

non-linear concept interactions. A further consideration is our use of a captioning model instead of human captions. While this makes the approach more feasible and scalable, it also risks the captioner hallucinating concepts not present in the image, which can negatively affect the relevance of the explanations. Selecting a properly evaluated captioning model that is suitable for the target image domain can help minimize this issue.

Acknowledgments and Disclosure of Funding

We express appreciation for the volunteers in our user study. We thank the anonymous reviewers for their constructive feedback. This work is partially supported by National Science Foundation (NSF) grants SCH-2123809 and SCH-2306572, Learning Engineering Virtual Institute grant G-23-2137070 and University of Florida Presidential Strategic Funding Award. Any opinions, findings, and conclusions or recommendations expressed in this paper, however, are those of the authors and do not necessarily reflect the views of the funding agency.

References

[1] 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. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[2] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 10684–10695, June 2022.

[3] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *ICML'23*, pages 19730–19742, Honolulu, Hawaii, USA, July 2023. JMLR.org.

[4] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved Baselines with Visual Instruction Tuning, October 2023. arXiv:2310.03744 [cs].

[5] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The twelfth international conference on learning representations*, 2024.

[6] Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Open-Vocabulary DETR with Conditional Matching. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IX*, pages 106–122, Berlin, Heidelberg, October 2022. Springer-Verlag.

[7] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, Xiao Wang, Xiaohua Zhai, Thomas Kipf, and Neil Houlsby. Simple Open-Vocabulary Object Detection. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*, pages 728–755, Berlin, Heidelberg, October 2022. Springer-Verlag.

[8] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *International Journal of Computer Vision*, 128(2):336–359, February 2020.

[9] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 1135–1144, New York, NY, USA, August 2016. Association for Computing Machinery.

[10] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 3319–3328, Sydney, NSW, Australia, August 2017. JMLR.org.

[11] Thomas Fel, Agustin Picard, Louis Béthune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadène, and Thomas Serre. CRAFT: Concept Recursive Activation FacTorization for Explainability. pages 2711–2721, 2023.

[12] Mara Graziani, Laura O'Mahony, An-phi Nguyen, Henning Müller, and Vincent Andrearczyk. Uncovering Unique Concept Vectors through Latent Space Decomposition. *Transactions on Machine Learning Research*, August 2023.

[13] Ying Wang, Tim G. J. Rudner, and Andrew Gordon Wilson. Visual Explanations of Image-Text Representations via Multi-Modal Information Bottleneck Attribution. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[14] Zhiyu Zhu, Zhibo Jin, Jiayu Zhang, NAN YANG, Jiahao Huang, Jianlong Zhou, and Fang Chen. Narrowing Information Bottleneck Theory for Multimodal Image-Text Representations Interpretability. In *The Thirteenth International Conference on Learning Representations*, 2025.

[15] Peijie Chen, Qi Li, Saad Biaz, Trung Bui, and Anh Nguyen. gScoreCAM: What objects is CLIP looking at? In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 1959–1975, December 2022.

[16] Chenyang Zhao, Kun Wang, Xingyu Zeng, Rui Zhao, and Antoni B. Chan. Gradient-based Visual Explanation for Transformer-based CLIP. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 61072–61091. PMLR, July 2024.

[17] Sriram Balasubramanian, Samyadeep Basu, and Soheil Feizi. Decomposing and Interpreting Image Representations via Text in ViTs Beyond CLIP. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[18] Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting CLIP's Image Representation via Text-Based Decomposition. October 2023.

[19] Jayneel Parekh, Pegah KHAYATAN, Mustafa Shukor, Alasdair Newson, and Matthieu Cord. A Concept-Based Explainability Framework for Large Multimodal Models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[20] Sukrut Rao, Sweta Mahajan, Moritz Böhle, and Bernt Schiele. Discover-then-Name: Task-Agnostic Concept Bottlenecks via Automated Concept Discovery. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXXVII*, pages 444–461, Berlin, Heidelberg, October 2024. Springer-Verlag.

[21] Yuan Zang, Tian Yun, Hao Tan, Trung Bui, and Chen Sun. Pre-trained Vision-Language Models Learn Discoverable Visual Concepts. *Transactions on Machine Learning Research*, 2025.

[22] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a Bottle: Language Model Guided Concept Bottlenecks for Interpretable Image Classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19187–19197, June 2023.

[23] Konstantinos P. Panousis, Dino Ienco, and Diego Marcos. Coarse-to-Fine Concept Bottleneck Models. November 2024.

[24] Johanna Vielhaben, Stefan Bluecher, and Nils Strodthoff. Multi-dimensional concept discovery (MCD): A unifying framework with completeness guarantees. *Transactions on Machine Learning Research*, 2023.

[25] Ruihan Zhang, Prashan Madumal, Tim Miller, Krista Ehinger, and Benjamin Rubinstein. Invertible Concept-based Explanations for CNN Models with Non-negative Concept Activation Vectors. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:11682–11690, May 2021.

[26] Usha Bhalla, Alex Oesterling, Suraj Srinivas, Flavio Calmon, and Himabindu Lakkaraju. Interpreting CLIP with Sparse Linear Concept Embeddings (SpLiCE). November 2024.

[27] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network Dissection: Quantifying Interpretability of Deep Visual Representations. pages 6541–6549, 2017.

[28] Xinyue Xu, Yi Qin, Lu Mi, Hao Wang, and Xiaomeng Li. Energy-Based Concept Bottleneck Models: Unifying Prediction, Concept Intervention, and Probabilistic Interpretations. In *The Twelfth International Conference on Learning Representations*, 2024.

[29] Eunji Kim, Dahuin Jung, Sangha Park, Siwon Kim, and Sungroh Yoon. Probabilistic concept bottleneck models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023. event-place: Honolulu, Hawaii, USA.

[30] Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free Concept Bottleneck Models. September 2022.

[31] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc Concept Bottleneck Models. September 2022.

[32] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, June 2015.

[33] Alon Lavie and Abhaya Agarwal. Meteor: an automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, USA, June 2007. Association for Computational Linguistics.

[34] Shashank Goel, Hritik Bansal, Sumit Bhatia, Ryan A. Rossi, Vishwa Vinay, and Aditya Grover. CyCLIP: Cyclic Contrastive Language-Image Pretraining. October 2022.

[35] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, March 1951. Publisher: Institute of Mathematical Statistics.

[36] Mingkai Zheng, Shan You, Fei Wang, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. ReSSL: Relational Self-Supervised Learning with Weak Augmentation. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[37] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, June 2020. ISSN: 2575-7075.

[38] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 21271–21284. Curran Associates, Inc., 2020.

[39] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow Twins: Self-Supervised Learning via Redundancy Reduction. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12310–12320. PMLR, 2021.

[40] Bokun Wang and Tianbao Yang. Finite-sum coupled compositional stochastic optimization: Theory and applications. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23292–23317. PMLR, 17–23 Jul 2022.

[41] Zhuoning Yuan, Yuexin Wu, Zi-Hao Qiu, Xianzhi Du, Lijun Zhang, Denny Zhou, and Tianbao Yang. Provable stochastic optimization for global contrastive learning: Small batch does not harm performance. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 25760–25782. PMLR, 17–23 Jul 2022.

[42] Scott M Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[43] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014.

[44] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

[45] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, June 2010. ISSN: 1063-6919.

[46] 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, June 2009. ISSN: 1063-6919.

[47] Hui Chen, Guiguang Ding, Xudong Liu, Zijia Lin, Ji Liu, and Jungong Han. IMRAM: Iterative Matching With Recurrent Attention Memory for Cross-Modal Image-Text Retrieval. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12652–12660, June 2020. ISSN: 2575-7075.

[48] Yan Huang, Wei Wang, and Liang Wang. Instance-Aware Image and Sentence Matching with Selective Multimodal LSTM. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7254–7262, July 2017. ISSN: 1063-6919.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide quantitative, qualitative experiments, a human evaluation, and a case study to support our claims in Section 4. Additional metrics and visualizations are provided in the Appendix (Sect. A.1, A.2, and A.3).

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: We provide a limitation section in the discussion section (Sect. 5).

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: Our paper does not introduce theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide pseudocode for our Algorithm 1. We describe the dataset used, metrics, hyper-parameters in the Experiment section (Sect. 4), and provide further experimental details in Appendix B.3. We provide source code and all of the used datasets are publicly accessible.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide code along with instructions to reproduce our results in the supplemental materials. The datasets we used are publicly accessible.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We describe the experiments in the Experiments section 4, provide additional details in Appendix B.3, and provide code in supplemental materials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We repeat experiments 5 times and report 2σ error intervals in Table 1 in the main paper. We state the confidence level and p-value for the user study in Appendix B.4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide this information in the experiments section of the paper.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our work adheres to NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts in Appendix B.5.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

Justification: Our paper explains an existing model. We do not release any data nor train a new model; therefore our work poses no such risk.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes] .

Justification: We cite and credit all used asset.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: Our code comes with documentation.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [Yes]

Justification: We describe the protocol for our small-scale user study in Section 4.4 and Appendix B.4.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [Yes]

Justification: Our study pose no more than minimal risk to participant and is ruled exempt by the instituion's IRB.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs are not involved in the core method development of our paper.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Appendix Contents

A Additional results

A.1 Retrieval faithfulness metrics

Table 2: COCO retrieval faithfulness metrics. For our method, we report the mean over 5 runs.

Method	COCO I→T			COCO T→I		
	R@1	R@5	R@10	R@1	R@5	R@10
SpLiCE [26]	38.12	72.40	86.00	38.26	72.74	85.66
Ours	**61.99**	**89.31**	**95.51**	**55.03**	**86.64**	**94.46**

Table 3: Flickr30k retrieval faithfulness metrics. For our method, we report the mean over 5 runs.

Method	Flickr30k I→T			Flickr30kT→I		
	R@1	R@5	R@10	R@1	R@5	R@10
SpLiCE [26]	40.25	70.41	80.65	40.50	69.56	80.04
Ours	**66.37**	**86.32**	**91.26**	**60.31**	**82.79**	**89.42**

Table 2 and Table 3 shows the detailed R@K metrics for COOC and Flickr30k of our method and SpLiCE, which is supplementary to Table 1 of the main paper.

A.2 Task performance metrics

Table 4: Performance metrics. For our method, we report the mean over 5 runs.

Method	COCO I→T			COCO T→I			SUN397	ImageNet
	R@1	R@5	R@10	R@1	R@5	R@10		
SpLiCE [26]	32.26	62.94	76.44	29.92	59.22	73.00	52.41	42.91
CLIP ViT-B/32	51.90	81.26	90.24	47.48	77.10	87.08	**60.71**	**61.91**
Ours	**58.03**	**85.68**	**93.19**	**56.30**	**84.98**	**93.09**	57.98	52.37

Table 4, 5 presents a comparison of performance metrics (evaluated against dataset ground truth) for our method, SpLiCE, and the original CLIP model. The baseline is setup similar to the experiment in Tab. 1. As can be seen in Table 4, our method consistently outperforms the baseline SpLiCE across tasks given the same sparsity.

Furthermore, although designed for post-hoc explanation, our surrogate representation exhibits the ability to sometimes outperform the CLIP model it explains on the zero-shot tasks we tested, despite never having access to the dataset labels and only being trained on the similarities produced by the CLIP model. We hypothesize that performance improvement (when they exists) stems from the increased robustness of concept-based inputs, which may be less susceptible to common image degradations such as occlusion, blurriness, or general noise, compared to raw image inputs.

A.3 More visualizations

In Figure 8, we provide a more comprehensive list of concepts of the images visualized in Fig. 4.

A.4 More datasets

In Table 6, we provide more zero-shot classification results on more datasets (Flowers102, Food101). Our method continues to yield consistent improvements on these fine-grained classification tasks. In Figure 9, we visualize some results from these datasets.

Table 5: Performance metrics (continued). For our method, we report the mean over 5 runs.

Method	Flickr30k I→T			Flickr30k T→I		
	R@1	R@5	R@10	R@1	R@5	R@10
SpLiCE [26]	36.43	65.03	75.36	36.83	63.50	74.20
CLIP ViT-B/32	67.17	88.90	93.80	63.60	86.78	92.26
Ours	**74.53**	**92.58**	**96.24**	**73.60**	**92.67**	**96.30**

Table 6: Faithfulness metrics for more datasets.

Method	Flowers102	Food101
SpLiCE	26.12	51.80
Ours	**37.73**	**63.62**

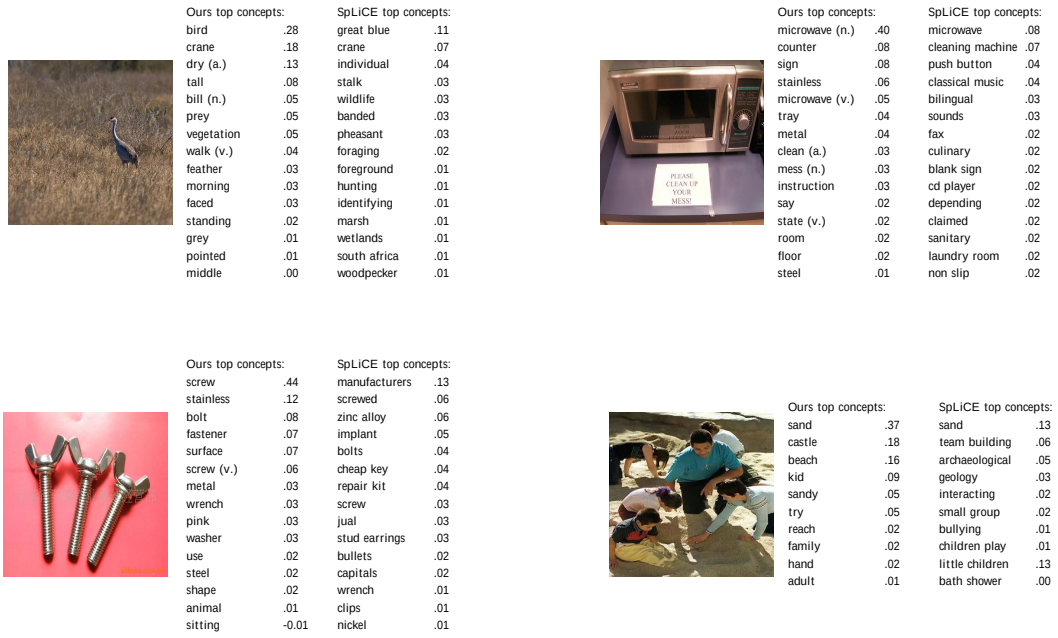

Figure 8: More comprehensive depiction of concept list from our method and the baseline.

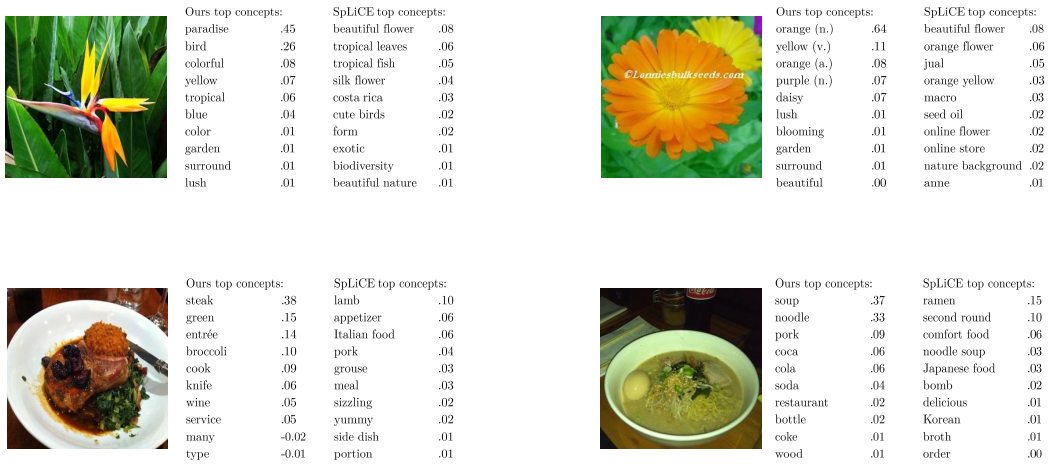

Figure 9: Visualizations for samples from the Flowers102 dataset (upper) and Food101 dataset (lower).

B Method details

B.1 Concept identification details

We redescribe our concept identification process step-by-step with more details:

1. (If not already present) obtain the captions for each image by using a captioning model
2. Extract nouns, verbs, and adjectives concepts via part-of-speech tagging with the `nltk` library
3. Filter out words not presenting in WordNet, infrequent and overly frequent concepts to obtain the vocabulary C
4. Estimate concept prevalence score for each image-concept pair to obtain the concept vectors $\mathbf{c}_i$

B.2 Algorithm for Surrogate Learning

Below is the pseudocode for the algorithm described in Section 3.3.

Algorithm 1: Algorithm for Surrogate Learning

Input: CLIP encoders e_I, e_T, surrogate model f, dataset $\mathcal{S}$, concepts $\mathcal{C}$, temperature τ, batch size B, number of iterations T

for $t = 0, \dots, T-1$ **do**
- Sample a mini-batch of triplets $\mathcal{B}_t = \{(\mathbf{x}_i, \mathbf{c}_i, \mathbf{t}_i)\}$ from the dataset
- Update $u_{t+1,i}, v_{t+1,i}$ using (5) and (6) for $i \in \mathcal{B}_t$
- Set $u_{t+1,i} = u_{t,i}, v_{t+1,i} = v_{t,i}$ for $i \notin \mathcal{B}_t$
- Compute gradient estimator w.r.t. f_t:

$$\frac{1}{|\mathcal{B}_t|^2} \sum_{i \in \mathcal{B}_t} \sum_{j \in \mathcal{B}_t} \frac{\exp\left(\frac{e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right)}{u_{t+1,i}} \cdot \frac{\exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right)}{v_{t+1,i}} \cdot \nabla g(f_t, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{B}_t).$$

- Update f_{t+1} from f_t using an optimizer

We now present the full derivation of Algorithm 1. We will focus only on $D_{KL}(P_i^e \parallel P_i^f)$ since the procedure for optimizing $D_{KL}(Q_j^e \parallel Q_j^f)$ can be derived analogously. From the definition of KL divergence, we can write the first part of (3) as

$$\frac{1}{2n} \sum_{i=1}^{n} D_{KL}(P_i^e \parallel P_i^f) = -\frac{1}{2n} \sum_{i=1}^{n} \sum_{j=1}^{n} P_i^e(j) \log P_i^f(j) + \frac{1}{2n} \sum_{i=1}^{n} \sum_{j=1}^{n} P_i^e(j) \log P_i^e(j).$$

Note that e is fixed and we want to optimize only f, we will discard the second term on the right hand side since it does not involve f. Plugging (1) and (2) into the above equation, we get

$$\begin{aligned}
& -\frac{1}{2n} \sum_{i=1}^{n} \sum_{j=1}^{n} P_i^e(j) \log P_i^f(j) \\
= & -\frac{1}{2n} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\exp(e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_j)/\tau)}{\sum_k \exp(e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_k)/\tau)} \cdot \log \frac{\exp(f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)/\tau)}{\sum_k \exp(f(\mathbf{c}_i)^\top e_T(\mathbf{t}_k)/\tau)} \\
= & \quad \frac{1}{2n} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\sum_{k=1}^{n} \exp\left(\frac{e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_k) - e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_j)}{\tau} \right) \right)^{-1} \\
& \qquad \cdot \log \sum_{k=1}^{n} \exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_k) - f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{\tau} \right).
\end{aligned} \tag{10}$$

Recall that $\mathcal{S}$ denotes the whole dataset, and

$$g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{S}) := \frac{1}{n}\sum_{k=1}^{n} \exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_k) - f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right)$$
$$= \frac{1}{n}\sum_{k=1}^{n} \exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_k)}{\tau}\right) / \exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right).$$

Then (10) can be equivalently written as

$$\frac{1}{|\mathcal{S}|^2}\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}} \frac{1}{g(e_I, e_T; \mathbf{x}_i, \mathbf{t}_j, \mathcal{S})} \cdot \log g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{S}). \tag{11}$$

The gradient w.r.t. f is given by

$$\frac{1}{|\mathcal{S}|^2}\sum_{i\in\mathcal{S}}\sum_{j\in\mathcal{S}} \frac{1}{g(e_I, e_T; \mathbf{x}_i, \mathbf{t}_j, \mathcal{S})} \cdot \frac{1}{g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{S})} \cdot \nabla g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{S}).$$

Since we only have access to one mini-batch of data $\mathcal{B}$ at each iteration, the obtained mini-batch gradient estimator is

$$\frac{1}{|\mathcal{B}|^2}\sum_{i\in\mathcal{B}}\sum_{j\in\mathcal{B}} \frac{1}{g(e_I, e_T; \mathbf{x}_i, \mathbf{t}_j, \mathcal{B})} \cdot \frac{1}{g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{B})} \cdot \nabla g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{B}).$$

However, due to the non-linearity of the reciprocal function, the expectation over $\mathcal{B}$ is not equal to the true gradient. Thus the mini-batch estimator is biased. To address this problem, we use two moving average sequences u and v to approximate $g(e_I, e_T; \mathbf{x}_i, \mathbf{t}_j, \mathcal{S})$ and $g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{S})$ respectively:

$$u_{t+1,i} = (1-\gamma_1)u_{t,i} + \gamma_1 \frac{1}{|\mathcal{B}_t|}\sum_{k\in\mathcal{B}_t} \exp\left(\frac{e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_k)}{\tau}\right),$$

$$v_{t+1,i} = (1-\gamma_2)v_{t,i} + \gamma_2 \frac{1}{|\mathcal{B}_t|}\sum_{k\in\mathcal{B}_t} \exp\left(\frac{f(\mathbf{x}_i)^\top e_T(\mathbf{t}_k)}{\tau}\right),$$

where $\gamma_1, \gamma_2 \in (0, 1]$ are two hyperparameters. Then we can approximate $g(e_I, e_T; \mathbf{x}_i, \mathbf{t}_j, \mathcal{S})$ and $g(f, e_T; \mathbf{c}_i, \mathbf{t}_j, \mathcal{S})$ using

$$u_{t+1,i} / \exp\left(\frac{e_I(\mathbf{x}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right), \text{ and } v_{t+1,i} / \exp\left(\frac{f(\mathbf{c}_i)^\top e_T(\mathbf{t}_j)}{\tau}\right).$$

B.3 Experimental details

Datasets. For the Flickr30k dataset, we explain on the full dataset. For the SUN397 dataset, we use the first official testing split. For the COCO 2017 and ImageNet dataset, we use the validation split. Following [26], for computational efficiency, the retrieval metrics are computed in batches of size 1000 and averaged over the full dataset.

Training. We use the Amsgrad variant of the AdamW optimizer with learning rate 10^{-3} and weight decay 10^{-6}. During training, we distill image-text similarities within the batch following Algorithm 1. We perform augmentations on both modalities: selecting a random caption as text augmentation, and random center crop and horizontal flip as image augmentation.

B.4 User study details

The user study involved 10 volunteers who did not receive monetary compensation. The interface is shown in Figure 10. Notably, we observed that SpLiCE's weights are often lower than that of us, since they do not sum to one. To reduce the chance of the user differentiating the two methods based on the weights, we scaled SpLiCE's weights for each individual image so that their sum equals ours.

The study was ruled exempt by our institution's IRB, as no more than minimal risk is posed to the participants. No identifying information was collected.

Criteria	CI	p-value
Relevance	1.61 ± 0.54	9×10^{-8}
Completeness	1.69 ± 0.72	1×10^{-6}
Utility	1.66 ± 0.62	4×10^{-7}

Table 7: The p-values and confidence intervals (CI) for hypothesis testing in our user study. A value of 1 denotes strong preference for EXPLAIN-R, averaged across the 20 shown samples. The hypothesis tested is whether the population mean is less than 3, which denotes neurality (no preference for SpLiCE or EXPLAIN-R).

B.5 Broader impacts

Our work addresses the problem of interpreting CLIP image representations in a task independent manner. EXPLAIN-R provides a tool for users and researchers to inspect the semantic content of the representation and provide a simple, intuitive summarization of the learned concepts for each image. For the users, this will enhance transparency and trustworthiness in multimodal representations, which traditionally relies on downstream evaluation. Researchers can use EXPLAIN-R to inspect individual embeddings and model predictions, as well as aggregate them over the dataset to obtain a more holistic view of the concepts learned by the model.

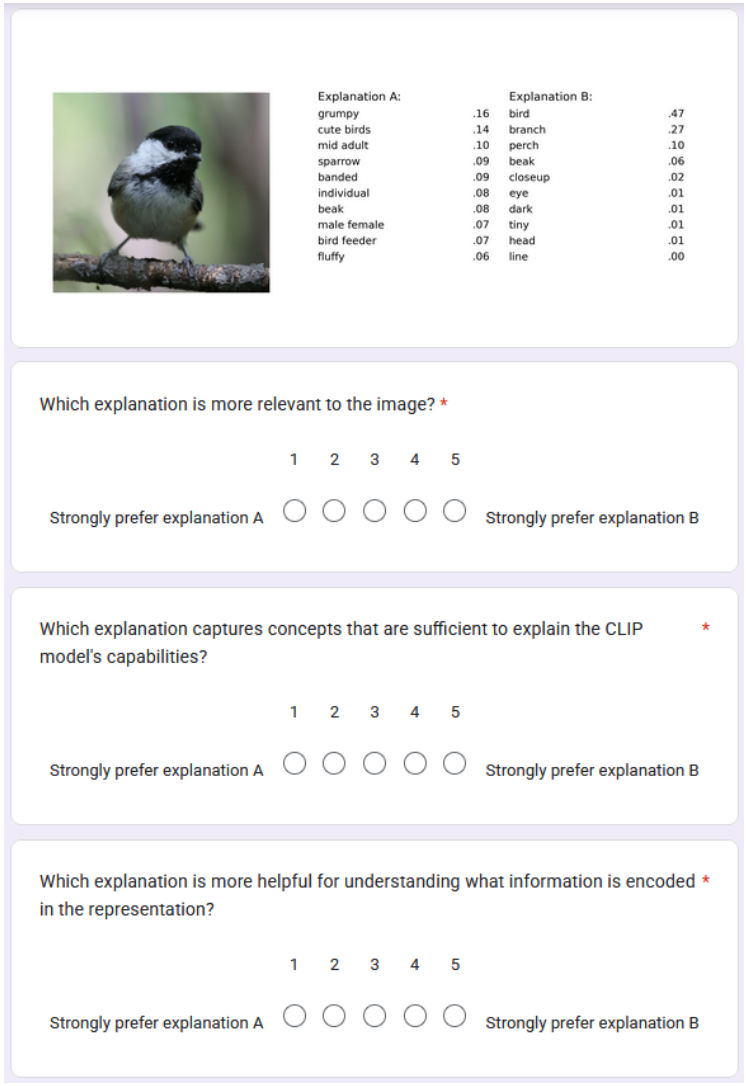

Figure 10: The interface for our user study. For each input image, we show the top 10 concepts from both methods, along with the weights. We scale SpLiCE's weights so they have the same mean as ours.