Understanding Distributed Representations of Concepts in Deep Neural Networks without Supervision

Wonjoon Chang[1] *, **Dahee Kwon**[1] *, **Jaesik Choi**[1, 2]

[1] Korea Advanced Institute of Science and Technology
[2] INEEJI
{one_jj, daheekwon, jaesik.choi}@kaist.ac.kr

Abstract

Understanding intermediate representations of the concepts learned by deep learning classifiers is indispensable for interpreting general model behaviors. Existing approaches to reveal learned concepts often rely on human supervision, such as pre-defined concept sets or segmentation processes. In this paper, we propose a novel unsupervised method for discovering distributed representations of concepts by selecting a principal subset of neurons. Our empirical findings demonstrate that instances with similar neuron activation states tend to share coherent concepts. Based on the observations, the proposed method selects principal neurons that construct an interpretable region, namely a Relaxed Decision Region (RDR), encompassing instances with coherent concepts in the feature space. It can be utilized to identify unlabeled subclasses within data and to detect the causes of misclassifications. Furthermore, the applicability of our method across various layers discloses distinct distributed representations over the layers, which provides deeper insights into the internal mechanisms of the deep learning model.

Introduction

Despite the remarkable performance of Deep Neural Networks (DNNs) in learning intricate data relationships (LeCun, Bengio, and Hinton 2015), their inherent lack of transparency remains a significant challenge. This opacity makes it difficult to understand the decision-making process, reducing model reliability and weakening the applicability in risk-sensitive domains where careful decisions are needed (Gunning et al. 2019; Samek et al. 2019). To gain insights into the general behaviors of DNNs, it is essential to reveal the semantic representations that DNNs learn. Our primary goal is to understand the distributed representations of concepts embedded within a trained model without external supervision. This approach facilitates the identification of diverse concepts within the model, including subclass distinctions, class-agnostic concepts, and even concepts that might contribute to misclassification.

Various eXplainable Artificial Intelligence (XAI) methods have been developed to enhance the transparency of a model. The gradient-based methods reveal which parts of input significantly contribute to the model's classification result based on the gradient information (Simonyan, Vedaldi, and Zisserman 2014; Bach et al. 2015; Selvaraju et al. 2017; Sundararajan, Taly, and Yan 2017; Chattopadhay et al. 2018). Yet, they focus on individual instances rather than the representative concepts that the model has learned in terms of generality. In this context, concept-based explanation methods emerge to provide more nuanced and general explanations (Kim et al. 2018; Ghorbani et al. 2019; Crabbé and van der Schaar 2022). Despite their perceptually intuitive results, most methods heavily rely on human supervision. Not only does it take substantial cost to require a refined concept dataset, but also there is no guarantee that the pre-defined concepts truly reflect the model behavior.

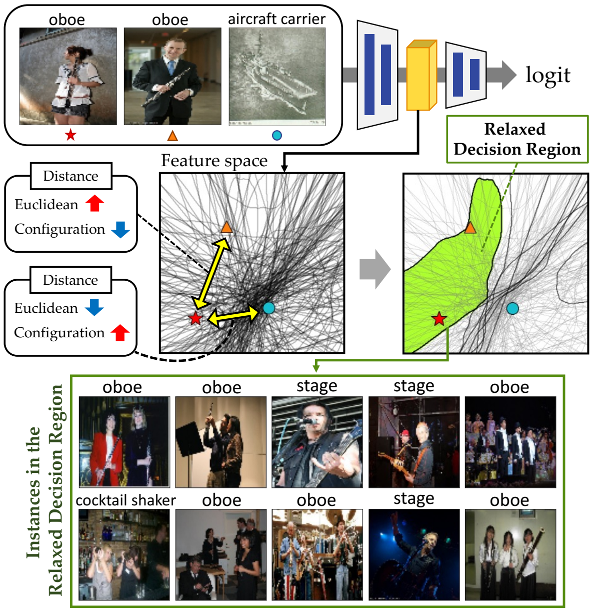

Figure 1: Relaxed Decision Region (RDR). Top: Description of the target sample and its neighbors. Middle: Visualization of the feature space and the RDR. Our RDR framework groups instances that have similar neuron activation states in the feature space. Bottom: Instances in the RDR share the coherent concept of 'a person with a stick'.

[*] These authors contributed equally.

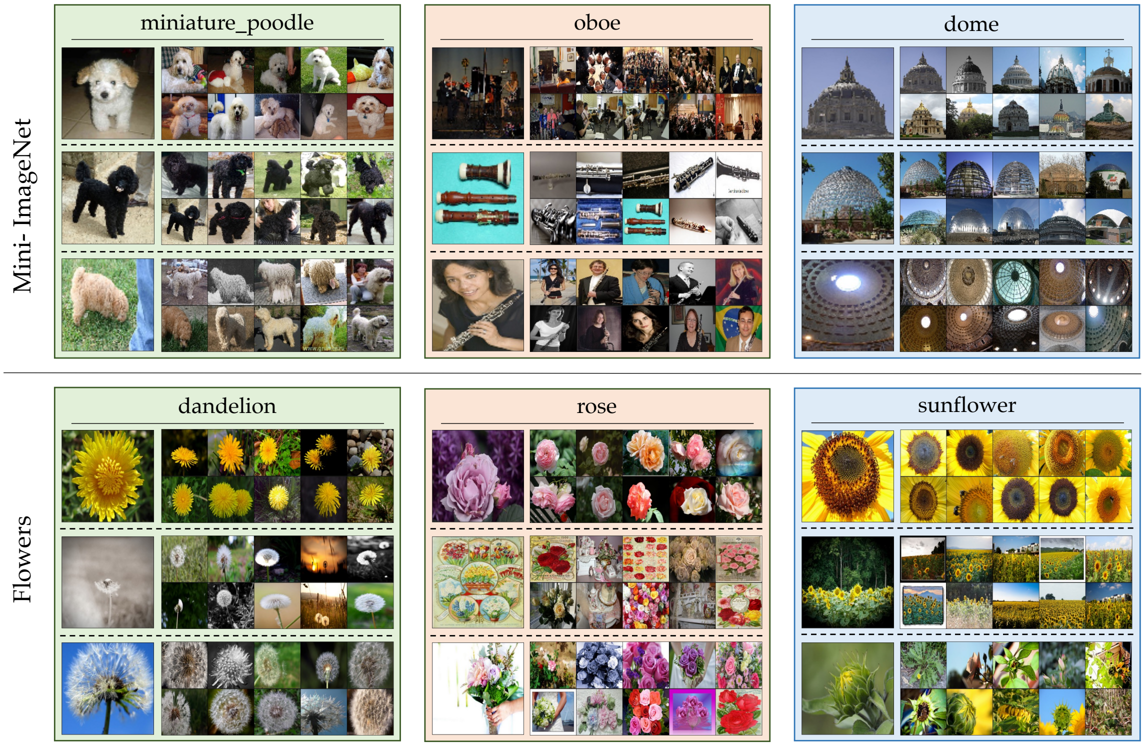

Figure 2: Without prior knowledge of label information, our RDR framework successfully captures learned concepts such as subclasses, shapes, crowds, composition, and the degree of flowering, as well as simple color schemes.

Another line of research takes a distinct approach that directly discloses the concepts embedded in the model by observing the role of internal neurons (Bau et al. 2017; Fong and Vedaldi 2018). In particular, under the assumption of a *distributed representation* (Hinton 1984), Fong and Vedaldi (2018) find combinations of neurons that represent learned concepts with segmentation information. Nevertheless, they still require human-annotated information and often entail computationally intensive manual searching.

In this paper, we present an interpretation framework aiming to elucidate learned concepts in DNNs. The proposed framework captures concept representations by leveraging the inherent information in the intermediate layers and offers example-based explanations without supervision. We first empirically demonstrate that instances with similar activation states share coherent concepts, and select a subset of relevant neurons, namely the *principal configuration*. Using the *principal configuration*, our approach constructs an interpretable region named *Relaxed Decision Region* (RDR) (Figure 1). Our RDR framework can reveal various learned concepts, including subclasses (Figure 2), concepts leading to misclassification, and diverse concepts across different layers.

Related Work

Concepts learned by DNNs can be unveiled by leveraging human supervision directly. Concept discovery methods, when given a predefined concept set, compute a relevant concept vector or a region in the internal feature space (Kim et al. 2018; Schrouff et al. 2021; Crabbé and van der Schaar 2022; Sajjad, Durrani, and Dalvi 2022; Oikarinen and Weng 2023). Although attempts have been made to bypass the need for pre-defined concept sets (Ghorbani et al. 2019; Küsters et al. 2020; Koh et al. 2020), they entail other types of costs, such as the segmentation process.

Other notable approaches aim to identify the role of internal components in DNNs, such as convolutional filters, by aligning activation patterns with pre-defined information (Bau et al. 2017; Fong and Vedaldi 2018; Angelov 2020; Achtibat et al. 2022). These approaches commonly utilize segmentation information to evaluate individual concepts captured through internal neurons. Recently, Oikarinen and Weng (2023) introduced a method providing textualized explanations for internal neurons, leveraging a CLIP model with pre-defined textual concept sets. Although these approaches still involve human supervision, they offer strong empirical evidence that internal neuron activations potentially encode information about learned semantics.

To avoid the requirement of supervision, example-based explanation methods select representative exemplars that summarize data distribution (Kim, Khanna, and Koyejo 2016; Khanna et al. 2019; Cho et al. 2021). Despite the advantages of their unsupervised algorithms, there is no guarantee that the exemplars precisely reflect the decision logic of the model. Another way is to explicitly design a specific structure capable of learning prototypical representations itself (Chen et al. 2019; Nauta et al. 2023). In such cases, however, the model structure is necessarily constrained.

Lam et al. (2021) proposed the Representative Interpretation (RI) method, which establishes a region in the feature space encompassing the maximum number of instances of a target class. While RI focuses on a specific target class, our objective is to construct a region where instances share coherent concepts without being constrained by class distinctions. We achieve it by leveraging activation patterns in an unsupervised manner. The idea is based on the fact that activation patterns are closely linked to model decisions (Chu et al. 2018; Gopinath et al. 2019) and captured representations (Bau et al. 2017; Fong and Vedaldi 2018).

Problem Definition

To set the groundwork for our discussion, this section introduces necessary terminologies and definitions. We start by clarifying the key properties of the concept sets we aim to find from a trained DNN. (1) ***Learned representation***: The concept should originate within the model and be extractable from the set of neurons. (2) ***Coherence***: The concept should convey a common semantic meaning that is consistently observed across multiple instances. (3) ***Discrimination***: The extracted concept set should be distinguishable from non-concept sets, ensuring straightforward human comprehension.

To identify a concept with the aforementioned properties, we select principal neurons that represent the corresponding concept, enabling interpretation for a target instance. The chosen neurons constitute the *Relaxed Decision Region*, described in the next section. The motivation of our neuron-selection approach is rooted in the principle of distributed representation (Hinton et al. 1986; Fong and Vedaldi 2018), suggesting that a model's learned concept representations are distributed across multiple internal components, akin to neurons in DNNs.

Terminologies

Consider a neural network $F : \mathbb{R}^{n_0} \to \mathbb{R}^{n_{\text{out}}}$ with L layers. For each layer $l \in [L]$, let N_l denote the set of neurons in layer l and n_l the number of neurons in layer l. With the assumption of the piecewise linear activation function, such as the family of ReLU (Montufar et al. 2014; Chu et al. 2018), the neural network can be represented as a composition of piecewise linear functions $F(\mathbf{x}) = f^{\text{out}} \circ f^L \circ f^{L-1} \circ \cdots \circ f^2 \circ f^1(\mathbf{x})$ where $\mathbf{x} \in \mathbb{R}^{n_0}$, $f^l : \mathbb{R}^{n_{l-1}} \to \mathbb{R}^{n_l}$ is a piecewise linear function for $l \in [L]$, and f^{out} is a linear mapping to the final logit. The output of the l-th layer is denoted by $\mathbf{x}^l = [\mathbf{x}^l_1, \ldots, \mathbf{x}^l_{n_l}]^\top$. Note that the output of the layer refers to the post-activation value of the layer. To express the internal process of the network F, we define a function $F^{(i+1):j}(\mathbf{x}^i) = f^j \circ f^{j-1} \circ \cdots \circ f^{i+2} \circ f^{i+1}(\mathbf{x}^i)$ as the successive partial representation of F, meaning the mapping from layer i to layer j.

Decision region. Given instance $\mathbf{x}$, the computation from the intermediate layer to the final logit can be represented as a linear projection

$$F(\mathbf{x}) = f^{\text{out}} \circ f^{(l+1):L}(\mathbf{x}^l) = \mathbf{W}\mathbf{x}^l + \mathbf{b}. \tag{1}$$

Note that $\mathbf{W}, \mathbf{b}$ depend on $\mathbf{x}$ and l. The preimage of $f^{\text{out}} \circ f^{(l+1):L}$ is divided into convex polytopes where the function becomes linear for each polytope (Chu et al. 2018). We call each polytope as *decision region* since the network applies the same linear projection for the belonging instances to obtain the final logit values. A decision region in the l-th layer is determined by the *activated states* of neurons in the higher layers, namely *configuration*.

Definition 1 (**Activation State).** *Given an input* $\mathbf{x} \in \mathbb{R}^{n_0}$ *and a neuron* i *in layer* l *of the neural network, the activation state is*

$$\mathbf{c}_i(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x}^l_i \le 0 \\ 1, & \text{if } \mathbf{x}^l_i > 0. \end{cases} \tag{2}$$

This is the case of the network with ReLU activation. We can easily extend the above definition to the other piecewise linear activation functions. The activation states can be represented as a vector code with discrete values. We define this code as a configuration.

Definition 2 (**Configuration).** *Given an input* $\mathbf{x} \in \mathbb{R}^{n_0}$ *and a set of neurons* N*, the configuration is*

$$\mathbf{c}^N(\mathbf{x}) = [\mathbf{c}_{N[1]}(\mathbf{x}), \ldots, \mathbf{c}_{N[|N|]}(\mathbf{x})] \tag{3}$$

where $N[i]$ *denotes the* i*-th neuron in set* N*.*

In consequence, a decision region where the given $\mathbf{x}$ located in the l-th layer is determined by $\mathbf{c}(\mathbf{x}) = \text{concat}([\mathbf{c}^{N_{l+1}}(\mathbf{x}), \ldots, \mathbf{c}^{N_L}(\mathbf{x})])$. Unless we specify the neuron set N, a configuration of $\mathbf{x}$ denotes $\mathbf{c}(\mathbf{x})$, considering every neurons in higher layers.

Internal decision boundary. We define an *internal decision boundary* as the boundary within the feature space where a transition in the activation state of each neuron occurs. In other words, each element of $\mathbf{c}(\mathbf{x})$ implies which state $\mathbf{x}$ is located with respect to the corresponding internal decision boundary. Note that the term decision boundary typically refers to the boundary where the classification results change in other literature. However, in this paper, we use the term decision boundary as the internal decision boundary.

Methods

The configuration, representing the activated states of the internal decision boundaries, determines the decision region to which an instance pertains at the target layer. This region serves as a guide for the model to extract pivotal information from the feature space. From the perspective of distributed representations, this information is captured by a specific subset of neurons, as experimentally observed by Fong and Vedaldi (2018). In this spirit, to understand the nature of the captured information for a target instance, it is imperative to identify a subset of principal neurons shared with relevant instances.

Following this logic, we present an interpretation framework designed to identify an interpretable region that aligns with the desired concept properties of learned concepts: learned concepts with coherence and discrimination. Our method automatically finds a set of instances sharing a concept of target instance through *Configuration distance*, and forms a *Relaxed Decision Region* by extracting principal neurons that represent this concept.

Configuration Distance

Before identifying a subset of principal neurons, the initial step involves finding the concept set. In contrast to other methods that heavily rely on pre-defined concept sets, we automatically discover a group of instances that share learned concepts with a given target instance. To enable this process, we introduce a metric to evaluate the difference in configurations as follows.

Definition 3 (Configuration Distance). *Given an instance $\mathbf{x}, \tilde{\mathbf{x}} \in \mathcal{X}$, the Configuration distance for a set of neurons N is defined as follows:*

$$d_C(\mathbf{x}, \tilde{\mathbf{x}}) = d_H(\mathbf{c}^N(\mathbf{x}), \mathbf{c}^N(\tilde{\mathbf{x}})) \tag{4}$$

where d_H denotes the Hamming distance.

N can be selected from either a single layer or multiple layers. If we want to focus on the specific projection from the l-th layer to the (l+1)-th layer, the configuration at the (l+1)-th layer, denoted as $d_C^{N_l}$, would be considered. We empirically verify in the following sections that concept sets are effectively found through the Configuration distance.

Relaxed Decision Region

From the configuration of a target, we select a principal subset of neurons. The activation states of these neurons construct an integrated decision region in the feature space where encompassed instances share the learned concept. We call the states of these selected neurons as the *principal configuration* $\mathbf{c}_\text{p}$ and the corresponding region as a *Relaxed Decision Region (RDR)*, $\mathcal{R}$. Finding a principal configuration can be formulated as follows:

$$\begin{aligned} \min_{\substack{\mathbf{c}_\text{p}\in\{0,1\}^t \\ N^*\subset N}} \; & \mathbb{E}_\mathbf{x}[d_H(\mathbf{c}^{N^*}(\mathbf{x}), \mathbf{c}_\text{p})] - \mathbb{E}_\mathbf{y}[d_H(\mathbf{c}^{N^*}(\mathbf{y}), \mathbf{c}_\text{p})] \\ \text{s.t.} \; & |N^*| = t \end{aligned} \tag{5}$$

where $\mathbf{x}, \mathbf{y}$ represent random variables corresponding to the positive (concept) set of inputs and the negative (concept) set of inputs, respectively. We find a principal subset N^* that consists of t number of neurons from a neuron set N. Minimizing the objective function encourages the principal configuration to exhibit strong coherence with the positive set (the first term) while also ensuring distinctiveness from the negative set (the second term).

In practice, we construct a positive set S and a negative set S_neg from training data. For a given target instance, we collect k-nearest neighbors (including itself) based on the d_C^N and assign them to the positive set S. The negative set S_neg can be easily set to the remaining data points. One of the strengths of our framework here is that it does not require a pre-defined concept set. To address the optimization problem in Equation (5), we employ a greedy algorithm for assigning neurons to N^*. Our greedy algorithm is described in Algorithm 1.

Theorem 1. *The optimal solution N^* of the problem in Equation (5) can be obtained by the greedy algorithm.*

Proof. See Appendix. □

Algorithm 1: Finding a Relaxed Decision Region

```
Input: $S$, $S_\text{neg}$, a set of neurons $N$, layer $l$
Parameter: the number of neurons to select $t$
Output: $N^*$, $\mathbf{c}_\text{p}$, $\mathcal{R}$

Initialize $N^*=\{\}$, $\mathbf{c}_\text{p} \in \{0,1\}^t$
$\bar{\mathbf{c}} = \frac{1}{|S|}\sum_{\mathbf{x}\in S}\mathbf{c}^N(\mathbf{x})$
$\bar{\mathbf{c}}_\text{neg} = \frac{1}{|S_\text{neg}|}\sum_{\mathbf{y}\in S_\text{neg}}\mathbf{c}^N(\mathbf{y})$
for $i = 1, \ldots, t$ do
    $i^* = \text{argmax}_{i\in N}|\bar{\mathbf{c}}_i - \bar{\mathbf{c}}_{\text{neg},i}|$
    $\mathbf{c}_{\text{p},i^*} = \bar{\mathbf{c}}_{i^*}$
    $N^* = N^* \cup \{i^*\}$ and $N = N \setminus \{i^*\}$
end for
$\mathcal{R} = \{\mathbf{x}^l \mid d_H(\mathbf{c}^{N^*}(\mathbf{x}), \mathbf{c}_\text{p}) = 0, \mathbf{x} \in \mathcal{X}\}$
```

In our problem, the candidate neuron set N consists of neurons that exhibit identical activation states for all instances in S so that RDR encompasses all instances in S. $\bar{\mathbf{c}}$, $\bar{\mathbf{c}}_\text{neg}$ denote the frequencies of neuron activations for instances in S and S_neg, respectively. Our greedy algorithm then iteratively selects a neuron that has the largest frequency difference between S and S_neg, indicating a high information gain to explain S. As each neuron is associated with an internal decision boundary in the feature space, we select a subset of the boundaries to form a larger region for coherent interpretation. This is why we use the term 'Relaxed' in RDR.

The parameter t controls the number of principal neurons. A smaller t makes the RDR looser so that it captures a more general concept while a larger t leads to detecting more specific properties. The number of neighbors k gives a similar but opposite effect to the results with t. To mitigate the concern about the difficulty of parameter selection, we provide guidance on selecting t and k in Appendix. We empirically check that the RDR works effectively with the parameters $k \in [5, 10]$ and $t \in [9, 15]$ in the penultimate convolutional block of the models in our experiments.

Geometric understanding of RDR. Figure 1 illustrates how RDR captures instances with learned concepts in the feature space from the geometric view. We selected three instances, two of which seem similar while the last one has distinct semantics. Then, we drew the 2-d plane that passes the feature maps of three given instances on the feature space of the 12th layer in VGG19. Each line on the plane represents an internal decision boundary corresponding to each neuron in the higher layers. Although the third instance has a smaller Euclidean distance value, the second instance (with a smaller Configuration distance value) is much more akin to the first instance. Indeed, We can easily check that there are numerous internal decision boundaries between the third instance and the others.

This intricate space partition enables DNN to apply different mappings according to inputs (Chu et al. 2018; Gopinath et al. 2019). Our RDR framework finds core internal decision boundaries and relaxes the decision regions where mappings are similar (highlighted in green in Figure 1). Further explanations are provided in Appendix.

Analysis for the Distance metrics

We demonstrate the effectiveness of the Configuration distance in disclosing learned concepts in the feature space compared to other standard metrics. Our analysis gives insights into the characteristics of decision regions.

We compare the Configuration distance with two standard distance metrics, the Euclidean distance and the Cosine distance, in terms of conceptual similarity among the nearest instances. The Configuration distance effectively captures instances with similar concepts in the feature space, whereas the Euclidean distance tends to fail in evaluating resemblance. In the case of the Cosine distance, while the closest neighbors usually include appropriate instances, some less relevant instances are also present among the neighbors.

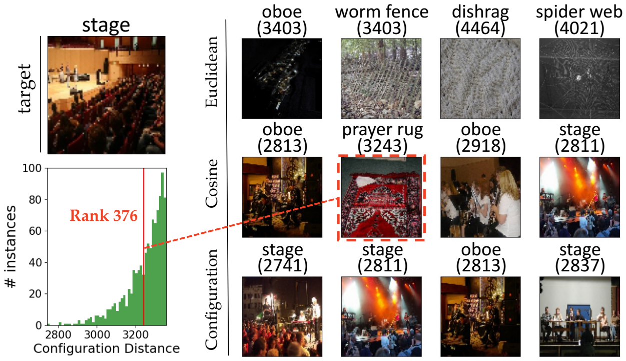

Figure 3: Top 4 nearest instances with different distance metrics. In the case of the Euclidean distance and the Cosine distance, the irrelevant instances are detected. These instances have large Configuration distances from the target.

In Figure 3, we visualize the top 4 nearest instances of the target 'stage' image, based on each distance metric. The number in parentheses denotes the Configuration distance to the target. It helps to successfully detect 'people in the stage-like place', while we cannot attain meaningful information from the Euclidean distance. For the Cosine distance, the second image, which is semantically distinct, is far away from the target with respect to the Configuration distance. The histogram shows the distribution of the 1000 smallest Configuration distance values within the training data. Among these, 375 instances are closer to the target than the 'prayer rug' image, and none of the top 4 Euclidean images are included. This serves as compelling evidence of the effectiveness of the Configuration distance in measuring differences in learned concepts.

Insight. We observe that for a given target instances with smaller Cosine distances tend to have smaller Configuration distances, which supports the efficacy of Cosine distance in evaluating similarity in the intermediate feature space compared to the Euclidean distance (more examples are in Appendix). We conjecture that this phenomenon is attributed to the geometry created by DNN structures. For example, in CNNs, decision regions are divided by polyhedral cones (Carlsson 2019) so that the angular difference between feature maps becomes highly related to the Configuration distance. This aligns with the empirical successes of prior work using the Cosine similarity in the feature space (Fong and Vedaldi 2018; Kim et al. 2018; Bachman, Hjelm, and Buchwalter 2019; Jeon, Jeong, and Choi 2020). We plan to explore this phenomenon further in our future work.

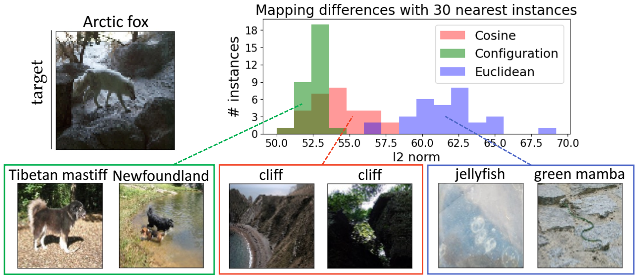

Figure 4: Mapping differences with 30 nearest neighbors. The Configuration distance indeed captures instances whose mappings are close to the target's one. With a smaller mapping difference, the image is more similar to the target.

The viewpoint of Mapping. So far, we have explained that DNNs extract different information from instances due to changes in mapping according to the configuration. In Figure 4, we compute the difference in mappings of 30 nearest instances. The difference is quantified using the L2 norm of the weight matrices in two successive layers. Compared to the other distance metrics, the Configuration distance captures instances whose mappings are close to the target's one, leading to the extraction of more similar information.

Experiments

In this section, we present the qualitative and quantitative evaluation results of our proposed method as well as various use cases. Our experiments are conducted on the Mini-ImageNet (Vinyals et al. 2016), Flowers Recognition (denoted by Flowers), Oxford pet, Broden (Bau et al. 2017), Imagenet-X (Idrissi et al. 2022) datasets, using VGG19 (Simonyan and Zisserman 2014), ResNet50 (He et al. 2016), and MobileNetV2 (Sandler et al. 2018) models. The choice of parameters t and k adheres to the criteria outlined in the previous section unless explicitly stated. Detailed settings of each experiment are provided in Appendix.

Coherence of Captured Concepts

To assess the coherence of captured concepts, we identify which parts in the image correspond to concepts with principal configurations from a convolutional layer. Following the methods of Bau et al. (2017) and Fong and Vedaldi (2018), we visualize activation maps of the channels that contain the neurons in the principal configuration.

Figure 5 shows our visualization results compared to those from other interpretability methods: Grad-CAM (Selvaraju et al. 2017), IG (Sundararajan, Taly, and Yan 2017) and ACE (Ghorbani et al. 2019). To apply IG for the intermediate layer, we compute the attribution scores for each neuron in the feature map and sum the scores across channels at each spatial location. In ACE, we adhere to the set-

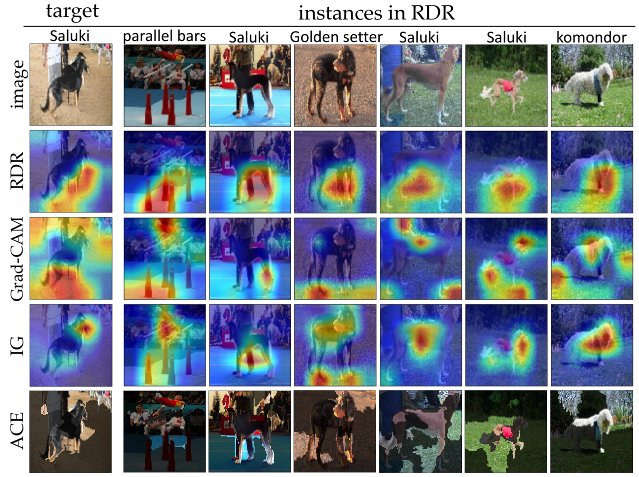

Figure 5: Retrieval results using RDR and masking for Concept Location. Without supervision, RDR successfully groups similar instances and related parts. The concept of 'legs' is learned for the images in the 12th layer.

tings outlined in the original paper and visualize images by masking all except the top 30% of significant segmentation patches. These results are obtained from the 12th layer in VGG19 using d_C^{12} with k=8, t=10. The instances in RDR share the concept of 'legs' and the selected channels focus on the 'legs' parts. Grad-CAM fails to find appropriate parts in the middle layer. While IG emphasizes specific parts of an object, it lacks consistency across the images. ACE prioritizes features for classification rather than maintaining a coherent conceptual interpretation. These results support the necessity of group-level interpretation to understand distributed representations, moving beyond the consideration of class-specific information such as gradients.

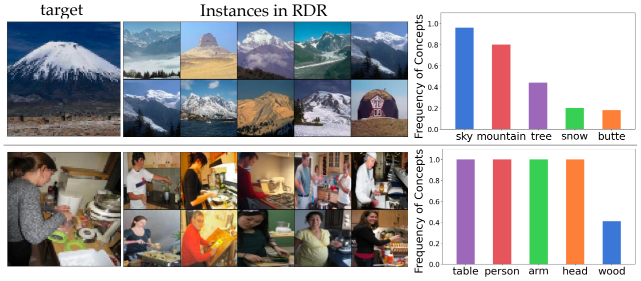

Figure 6: Identifying the coherent properties using human-annotated information in the Broden dataset.

We conduct additional investigations to assess whether concepts identified across instances in RDR align with predefined concept labels in the Broden dataset. The Broden dataset provides pixel-wise segmentation concept labels for each image. In Figure 6, we initially collect instances within an RDR (left) and identify the top 5 concept labels that they share (right). The experiment follows the same setting in Figure 5 but with k=10, t=15. The experimental results clearly demonstrate that the concepts captured with an RDR align well with human-labeled concepts.

Identifying Learned Concepts over Layers

By constructing RDR across various layers, we investigate how DNN recognizes instances at different stages of its architecture. Figure 7 illustrates the layer-wise differences of RDRs in the feature space of VGG19. We observe that the lower layers tend to capture spatial information, such as the object shape, whereas higher layers learn more detailed and class-specific features. The results align with the observations identified by Bau et al. (2017). Additional examples are provided in Appendix.

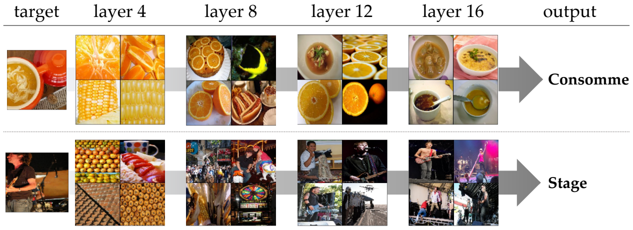

Figure 7: Differences in RDR over layers. The higher the layer is, the more class-specific concepts are captured.

Reasoning Misclassified Cases

We leverage RDR to comprehend the causes of misclassifications, under the assumption that certain internal neurons encode spurious correlations with actual labels, leading to classification errors. In Figure 8, we present which parts in the image contribute to misclassifications by designating S_{neg} as instances with the true label of the target. By visualizing the channels associated with the principal configuration, we can obtain evidence for each misclassification case. For instance, the second row demonstrates that the misclassification of the target instance stems from the presence of long legs, a characteristic commonly linked with Saluki. To further validate our findings, we double-check whether the extracted mislearned concepts align with human-labeled failure reasons from ImageNet-X. Additional illustrative examples are presented in the Appendix.

Finding Unlabeled Subclasses

Discovering unlabeled subclasses without human supervision is a challenging task. Our framework can reveal subclasses inherent in data without any prior knowledge. In Figure 2, we detect subclasses in the Mini-ImageNet dataset and the Flowers dataset captured by VGG19. We compute the Configuration distance at target layers $\{12, 14, 16, 17\}$. For each class, three target instances are chosen, followed by displaying 10 randomly selected images from the corresponding RDRs. Consequently, RDRs successfully capture learned concepts, including subclasses, in an unsupervised way. This qualitative evidence supports why our framework achieves good performance on the quantitative comparison shown in Table 1.

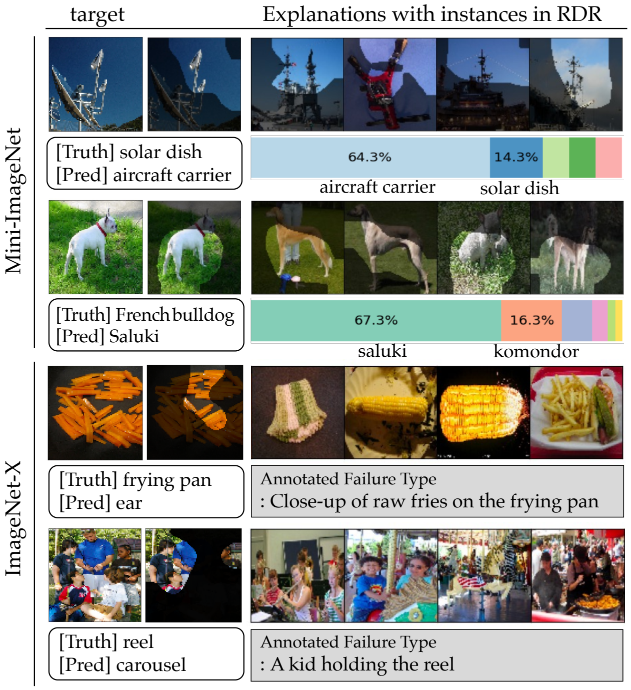

Figure 8: Misclassification reasoning can be investigated by examining RDR. In the Mini-ImageNet case, we provide the ratio of classes in RDR. In the ImageNet-X case, we describe the annotated failure type.

Quantitative Evaluation

We quantitatively evaluate the coherence of subclass groups identified by RDR, comparing with other methods on the Oxford-pet dataset. Although each model is trained to solely distinguish between cat and dog images, our RDR framework can implicitly identify specific breeds (subclasses) of target instances, even in the absence of explicit breed information. For a thorough verification, we form RDRs with 50 randomly chosen target instances, ensuring an equal number of instances within each group across all methods.

In Table 1, we employ two metrics, namely *purity* and *entropy*, for comprehensive quantitative evaluations (Zhao and Karypis 2001). Purity assesses the proportion of samples containing the target subclass within the group. Entropy measures the uncertainty of subclasses within the group.

$$\text{Purity} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}_{[y_t=\tilde{y}]}$$

$$\text{Entropy} = \sum_{y:\mathcal{P}_y \neq 0} -\mathcal{P}_y * \log \mathcal{P}_y$$

where $\mathcal{P}_y = \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}_{[y_t=y]}$ (empirical distribution). T is the number of samples in each group, and y_t is the subclass of t-th sample in a group. Subclass is denoted as y and that of a target instance is $\tilde{y}$. High purity or low entropy indicates that the group consistently extracts the subclass of the target.

We compare RDR with other approaches that can define interpretable regions: K-Nearest Neighborhood, Representative Interpretation (Lam et al. 2021), CAR (Crabbé and van der Schaar 2022) and CAV (Kim et al. 2018). To ensure a fair comparison, we consider the nearest neighbors as a concept set for CAR and CAV, as they require pre-defined concept sets. In the case of CAV, since the original CAV does not output a region, we define a region for CAV by containing instances with have high cosine similarities to the computed CAV. As shown in Table 1, RDR generally outperforms others in both purity and entropy.

	Purity			Entropy		
	VGG	RSN	MBN	VGG	RSN	MBN
RDR	**0.351**	**0.408**	**0.346**	**1.527**	**1.372**	1.531
KNN_C	0.303	0.328	0.329	1.588	1.497	**1.498**
CAR_C	0.022	0.038	0.036	2.264	2.153	2.527
CAV_C	0.314	0.387	0.323	1.549	1.416	1.575
RI	0.045	0.056	0.056	2.161	1.971	2.369
RDR_{Euc}	0.241	0.252	0.303	1.76	1.779	1.76
KNN_{Euc}	0.183	0.166	0.275	1.835	1.862	1.791
CAR_{Euc}	0.039	0.037	0.037	2.272	2.17	2.476
CAV_{Euc}	0.207	0.240	0.283	1.811	1.787	1.745
RDR_{Cos}	0.309	0.307	0.346	1.613	1.7	1.628
KNN_{Cos}	0.250	0.232	0.283	1.672	1.771	1.635
CAR_{Cos}	0.042	0.027	0.036	2.251	2.14	2.576
CAV_{Cos}	0.261	0.283	0.274	1.596	1.734	1.651

Table 1: Quantitative results for evaluating the coherence of subclass grouping. VGG, RSN, MBN represent VGG19, ResNet50, and MobileNetV2, respectively.

Following its original settings, CAR exhibits an excessively broad region, grouping various subclasses into a single region. Similarly, RI also generates a wide concept region, as it is formulated to maximize the number of samples in a target class, potentially overlooking implicit concepts. The competitive results observed in CAV can be attributed to the phenomenon where a high cosine similarity may result in a low Configuration distance, as elucidated in our analysis section. We also ablate the methods using different distance metrics, confirming the effectiveness of configuration.

Conclusion

We introduce a novel interpretation framework that reveals the learned concepts in DNNs without human supervision. Our key approach is to leverage the activation states to identify the distributed representations of concepts. We propose the Configuration Distance, a novel metric that effectively assesses the disparity in decision regions. It enables the automatic collection of concept sets, avoiding the need for pre-defined information. By extracting the principal configuration from the set, we construct a Relaxed Decision Region (RDR) that provides consistent interpretation for the related instances. In our experiments, we present various applications of RDR for interpreting DNNs, including subclass detection, reasoning misclassification, and exploring layer-wise concepts. We expect that our work guides the direction to understanding the decision-making process of DNNs, which is a crucial step for real-world applications.

Acknowledgements

This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2022-0-00984, Development of Artificial Intelligence Technology for Personalized Plug-and-Play Explanation and Verification of Explanation, 50%; No. 2022-0-00184, Development and Study of AI Technologies to Inexpensively Conform to Evolving Policy on Ethics, 30%; No. 2021-0-02068, Artificial Intelligence Innovation Hub, 10%; No. 2019-0-00075, Artificial Intelligence Graduate School Program (KAIST), 10%).

References

Achtibat, R.; Dreyer, M.; Eisenbraun, I.; Bosse, S.; Wiegand, T.; Samek, W.; and Lapuschkin, S. 2022. From" where" to" what": Towards human-understandable explanations through concept relevance propagation. *arXiv preprint arXiv:2206.03208*.

Angelov, D. 2020. Top2Vec: Distributed Representations of Topics. arXiv:2008.09470.

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

Bachman, P.; Hjelm, R. D.; and Buchwalter, W. 2019. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32.

Bau, D.; Zhou, B.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 6541–6549.

Carlsson, S. 2019. Geometry of deep convolutional networks. *arXiv preprint arXiv:1905.08922*.

Chattopadhay, A.; Sarkar, A.; Howlader, P.; and Balasubramanian, V. N. 2018. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, 839–847. IEEE.

Chen, C.; Li, O.; Tao, D.; Barnett, A.; Rudin, C.; and Su, J. K. 2019. This looks like that: deep learning for interpretable image recognition. *Advances in neural information processing systems*, 32.

Cho, S.; Chang, W.; Lee, G.; and Choi, J. 2021. Interpreting internal activation patterns in deep temporal neural networks by finding prototypes. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 158–166.

Chu, L.; Hu, X.; Hu, J.; Wang, L.; and Pei, J. 2018. Exact and consistent interpretation for piecewise linear neural networks: A closed form solution. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1244–1253.

Crabbé, J.; and van der Schaar, M. 2022. Concept Activation Regions: A Generalized Framework For Concept-Based Explanations. arXiv:2209.11222.

Fong, R.; and Vedaldi, A. 2018. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8730–8738.

Ghorbani, A.; Wexler, J.; Zou, J. Y.; and Kim, B. 2019. Towards automatic concept-based explanations. *In Proceedings of the Advances in Neural Information Processing Systems*, 32.

Gopinath, D.; Converse, H.; Pasareanu, C.; and Taly, A. 2019. Property inference for deep neural networks. In *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 797–809. IEEE.

Gunning, D.; Stefik, M.; Choi, J.; Miller, T.; Stumpf, S.; and Yang, G.-Z. 2019. XAI—Explainable artificial intelligence. *Science robotics*, 4(37): eaay7120.

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

Hinton, G. E. 1984. Distributed representations.

Hinton, G. E.; et al. 1986. Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society*, volume 1, 12. Amherst, MA.

Idrissi, B. Y.; Bouchacourt, D.; Balestriero, R.; Evtimov, I.; Hazirbas, C.; Ballas, N.; Vincent, P.; Drozdzal, M.; Lopez-Paz, D.; and Ibrahim, M. 2022. Imagenet-x: Understanding model mistakes with factor of variation annotations. *arXiv preprint arXiv:2211.01866*.

Jeon, G.; Jeong, H.; and Choi, J. 2020. An Efficient Explorative Sampling Considering the Generative Boundaries of Deep Generative Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04): 4288–4295.

Khanna, R.; Kim, B.; Ghosh, J.; and Koyejo, S. 2019. Interpreting black box predictions using fisher kernels. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 3382–3390.

Kim, B.; Khanna, R.; and Koyejo, O. O. 2016. Examples are not enough, learn to criticize! criticism for interpretability. *Proceedings of the Advances in Neural Information Processing Systems*, 29.

Kim, B.; Wattenberg, M.; Gilmer, J.; Cai, C.; Wexler, J.; Viegas, F.; et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *Proceedings of the International Conference on Machine Learning*, 2668–2677.

Koh, P. W.; Nguyen, T.; Tang, Y. S.; Mussmann, S.; Pierson, E.; Kim, B.; and Liang, P. 2020. Concept bottleneck models. In *International conference on machine learning*, 5338–5348. PMLR.

Küsters, F.; Schichtel, P.; Ahmed, S.; and Dengel, A. 2020. Conceptual explanations of neural network prediction for time series. In *2020 International joint conference on neural networks (IJCNN)*, 1–6. IEEE.

Lam, P. C.-H.; Chu, L.; Torgonskiy, M.; Pei, J.; Zhang, Y.; and Wang, L. 2021. Finding representative interpretations on convolutional neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1345–1354.

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

Montufar, G. F.; Pascanu, R.; Cho, K.; and Bengio, Y. 2014. On the number of linear regions of deep neural networks. *Proceedings of the Advances in neural information processing systems*, 27.

Nauta, M.; Schlötterer, J.; van Keulen, M.; and Seifert, C. 2023. PIP-Net: Patch-Based Intuitive Prototypes for Interpretable Image Classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2744–2753.

Oikarinen, T.; and Weng, T.-W. 2023. CLIP-Dissect: Automatic Description of Neuron Representations in Deep Vision Networks. arXiv:2204.10965.

Sajjad, H.; Durrani, N.; and Dalvi, F. 2022. Neuron-level interpretation of deep nlp models: A survey. *Transactions of the Association for Computational Linguistics*, 10: 1285–1303.

Samek, W.; Montavon, G.; Vedaldi, A.; Hansen, L. K.; and Müller, K.-R. 2019. *Explainable AI: interpreting, explaining and visualizing deep learning*, volume 11700. Springer Nature.

Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; and Chen, L.-C. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510–4520.

Schrouff, J.; Baur, S.; Hou, S.; Mincu, D.; Loreaux, E.; Blanes, R.; Wexler, J.; Karthikesalingam, A.; and Kim, B. 2021. Best of both worlds: local and global explanations with human-understandable concepts. *arXiv preprint arXiv:2106.08641*.

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

Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Visualising image classification models and saliency maps. *Deep Inside Convolutional Networks*, 2.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328. PMLR.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. *Advances in neural information processing systems*, 29.

Zhao, Y.; and Karypis, G. 2001. Criterion functions for document clustering: Experiments and analysis.