EXPLAINING IMAGE CLASSIFICATION THROUGH KNOWLEDGE-AWARE NEURON INTERPRETATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Although neural networks have achieved remarkable results, they still encounter doubts due to the intransparency. To this end, neural network prediction explanation is attracting more and more attentions. State of the art methods, however, rarely introduce human-understandable external knowledge, making the explanation hard to interpret by human beings. In this paper, we propose a knowledge-aware framework to explain neural network predictions for image scene classification. We introduce two notions of core concepts, with the help of knowledge graphs, to measure the association of concepts with respect to image scenes, and analyse solutions for prediction explanation and model manipulation. In our experiments on two popular scene classification datasets ADE20k and Opensurfaces, the proposed solutions produce better results than baseline and state of the art methods, e.g., our method produces over 25% IoU gain on compositional explanation for neuron behaviors. In addition, our core concepts and related explanation metrics can help effectively manipulate the model prediction, further leading to a new training method with 26.7% performance improvement.

1 INTRODUCTION

While neural networks have been achieving unprecedented advancements in various areas of artificial intelligence, deep architectures are not fully transparent and often perceived as "black-box" algorithms Adadi & Berrada (2018). This limitation has been identified in many fields Che et al. (2017), undermining users' trust and hence decreasing usability of such systems Ribeiro et al. (2016); Stepin et al. (2021). For instance, as shown in Figure 1(a), a neural model predicts the image as *utility room*, which is different from the ground truth (target label) *bedroom*. Without proper explanations, it is unclear why the model predicts this label and what is its relationship to the target label, making it hard to debug and optimise.

There has been a growing interest in exploring explanations of model predictions, which, generally speaking, could be categorized into two methods: decision analysis and functional analysis Shahroudnejad (2021). The former methods explore explanation by analyzing the internal components' behavior. Practically, they consider the effect of each neuron by decomposing the network classification decision into contributions of its input elements Montavon et al. (2017); Tian & Liu (2020). CGL Varshneya et al. (2021) and CompositionalNet Kortylewski et al. (2021) try to learn and adjust the neuron behavior by modifying the training objectives, but they do not aim at capturing which and why an element (e.g., layer, neuron) in the neural architectures plays a more important role in predictions. Functional analysis methods try to capture overall behavior by investigating the relation between inputs and outputs, using saliency map Zeiler & Fergus (2014); Zintgraf et al. (2017), occlusion and related techniques Smilkov et al. (2017); Sundararajan et al. (2017).

We focus on concept based explanations, a sub-category of decision analysis methods. The most relevant efforts are automatic concept based explanation (ACE) Ghorbani et al. (2019), ConceptSHAP Yeh et al. (2020) and VRX Ge et al. (2021), which focus on automatically identifying higher-level concepts for the neural models. Such image segments are extracted from many input samples (of a target class) together with their importance using TCAV Kim et al. (2018) for predicting that target class. However, these approaches are mainly capturing visual concepts through images segments, which are disconnected from one another and not necessarily human interpretable (cf. Figure 5(b)).

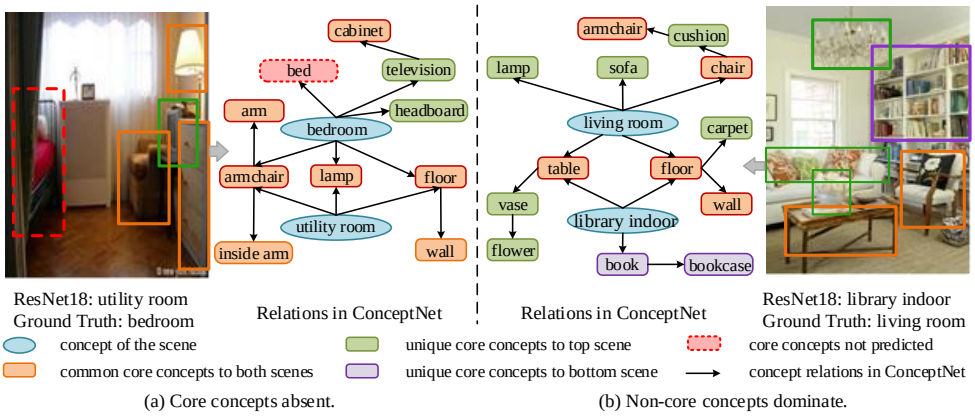


Figure 1: Two types of false predictions. We build the concept relations by ConceptNet, and the model can predict all the concepts (except concept *bed*).

In this work, we argue that concepts should be defined based on the human-understandable knowledge bases. Therefore, we propose a knowledge aware model prediction explanation approach, by introducing the notion of *core concepts*, which are knowledge graph (KG) based inter-connected concepts for scene identification. There are different ways of using core concepts. (1) *Absence of core concepts:* a model fails to capture some core concepts and thus makes wrong predictions. For example, in Figure 1(a), the model does not predict the core concept *bed* of the scene *bedroom*, and thus mistakenly predicts the scene to be *utility room*. (2) *Domination by non-core concepts:* a model predicts many non-core concepts and thus makes wrong predictions. For example, in Figure 1(b), the model predicts many non-core concepts, including *book* and *bookcase*, which are the core concepts of a wrong scene *library indoor*. Furthermore, core concept based explanations can be used to help debug the bad cases and optimise the model.

Contributions: To the best of our knowledge, this is the first effort to use general purpose knowledge graphs, such as ConceptNet, to define core concepts for image scene classification, by modelling semantic relationships between the originally disconnected concepts predicted by neurons. Accordingly, we propose solutions for the prediction explanation problem, for both prediction explanation for the model and compositional explanations for neuron behaviors, and model manipulation problem. It is worth mentioning that our approach relies on no additional image annotations and is adaptable to different CNNs. In this study, we evaluated several classic CNNs including ResNet, DenseNet, AlexNet and MobileNet, and have conducted extensive empirical experiments on ADE20k and Opensurfaces. The explanation achieves high quality according to the analysis on different metrics and human assessment. Our experiments show that core concepts and related explanation metrics can help optimise the model, leading to 26.7% of performance improvement. [1]

2 Preliminary

2.1 Knowledge Graphs

Knowledge Graphs (KGs) have become well known in knowledge representation and knowledge management applications widely across search Dietz et al. (2018); Gu et al. (2019), recommendation Guo et al. (2020), image classification Geng et al. (2021), visual question answering Chen et al. (2021) and industrial domains Bader et al. (2020).

In our study, we use ConceptNet Speer et al. (2017), a popular commonsense knowledge graph, containing large-scale triples, such as (*pillow, AtLocation, bed*) and (*wheels, Parts of, car*), with nodes representing general concepts and edges representing relations between concepts. We remove the relation pairs that connect to itself and obtain 37 relations, 1,785,572 concepts, and 3,377,895 triples, respectively.

[1] Code and data are available at: `https://github.com/neuroninterpretation/EIIC`

2.2 NEURON COMPOSITION

Neuron composition aims to improve the interpretability of neural network models by understanding the neuron behavior. Bau et al. (2017) proposed the NetDissect framework, evaluating the alignment between individual hidden units and a set of concepts. Assumed that f is a neural network and f_t is *t*-th neuron in intermediate layer. C is the pre-defined atomic concept set, and concepts are image segmentation masks for image pixels. For *bookcase* concept in Figure 1(b), it means an image region containing *bookcase*. $NetDissect\ (f_t)$ measures Intersection over Union score (IoU) between neuron features and concepts in input to find the most similar concept to neuron f_t:

$$NetDissect\ (f_t) = \underset{c_j \in C}{argmax}\ \sigma(A_t(\mathbf{x}), c_j), \tag{1}$$

where σ denotes measure function, $A_t(\mathbf{x})$ is activated neuron features of f_t with binary masks, and $\mathbf{x}$ denotes the whole input.

2.3 PROBLEM STATEMENT

Let $\mathcal{D} = \{x_1, x_2, ..., x_{|\mathcal{D}|}\}$ be a set of images, C be the overall concept set, and $Y = \{y_1, y_2, ..., y_{|Y|}\}$ be a set of scene labels; e.g., the image of Figure 1(a) is labelled as scene `bedroom`. Each image $x_i \in \mathcal{D}$ belongs to a scene $y_j \in Y$, and contains multiple concepts representing the objects in x_i such as *wall*, *floor*, *lamp*, *armchair* in the image of Figure 1(a). For each scene y_j, it has multiple images in $\mathcal{D}$, denoted as $\mathcal{D}_{y_j} \subseteq \mathcal{D}$. We assume the neural network f is trained and used for prediction. It maps an image x_i to a latent representation which is also known as neuron features or hidden states, denoted as $\{f_1, f_2, ..., f_n\}$, where f_t is known as *t*-th neuron features and n denotes the dimension. $IC(x_i)$ is the learning concepts by neuron features f_t. $CC(y_j)$ is the core concept set for scene y_j. In the rest of the paper, we assume that the KG contains all concepts in C.[2]

For each image x_i, it has a label y_p predicted by f and a target (ground truth) label y_j. Given a KG $\mathcal{G}$, we consider three tasks: *(T1)* prediction explanation: explaining why f predicts x_i as y_p, and why the prediction is correct (i.e., $y_p = y_j$) or wrong (i.e., $y_p \neq y_j$); *(T2)* neuron behavior explanation: identifying compositional logical concepts that closely approximate neuron behavior; *(T3)* model manipulation: studying methods to optimise model prediction performance.

3 APPROACH

In this section, we present a knowledge-aware framework, starting by introducing a MinMax-based NetDissect method (Sec. 3.1) learning which concepts are closely aligned with neuron behavior, as well as the notion of core concepts (Sec. 3.2). In Sec. 3.3, we propose two types of prediction explanation, based on core concepts and concepts learned by neurons. In Sec. 3.4, we propose to optimise explanations via concept filtering. Last but not least, we propose how to improve model prediction based on features from our explanations in Sec. 3.5.

3.1 MINMAX-BASED NETDISSECT

The procedure described in Eq.(1) can compute the general neuron behavior based on the whole dataset level while ignoring features that are unique and useful for an individual image prediction. Take image in Figure 1(a) as an example, based on the whole dataset, neuron 1 and 2 are related to concepts *sea* and *highway* respectively. However, this observation is useless for explaining image prediction. We would like to learn the neuron behavior on scene *bedroom* to see whether it can help explain model prediction. To achieve this, we propose a new variant of NetDissect Bau et al. (2017), named MinMax-based NetDissect, to learn the neuron behavior of individual image level. Formally, for an image x_i, measure function σ, and activated neuron features A_t of the *t*-th neuron, we have:

$$MinMax\text{-}NetDissect\ (f_t) = Ths\{\ \sigma(A_t(x_i), C_{y_j})\} \tag{2}$$

Thus the concepts $IC(x_i)$ learned by a target neuron in image classification can be obtained from the concept selection strategy[3] Ths. We consider three ways of selecting concepts that a neuron learns:

[2] In practice, if some concepts in C are not in the KG, we could often align them to some close KG concepts.

[3] In the original NetDissect, they directly use the concept with highest score for neuron. However, neurons do not express a single concept, but make predictions from multiple concepts Mu & Andreas (2020).

1) *whole layer*: use all concepts with IoU scores larger than 0; 2) *highest IoU*: only the concept with the highest IoU; 3) *threshold*: use only the concepts with IoU scores higher than a MinMax-based threshold that we compute as follows: (a) select the concept with the highest IoU for each neuron; (b) use the lowest IoU value among the IoU values of the selected concepts as the threshold.

3.2 CORE CONCEPTS

In this section, we address the problem of how to define the core concepts for each scene y_i. In the definitions of such core concepts, we exploit human understandable knowledge from a KG, which is different from implicit knowledge in visual concepts that are used in some other concept-based explanations Ghorbani et al. (2019). Towards this end, we define two kinds of core concepts: Scoping Core Concepts (SCC) and Identifier Core Concepts (ICC). Informally speaking, the SCCs of a scene y_j are the intersection of the concepts associated with y_j and the concepts in a background KG that are related to y_j. Thus the KG is used to help scope the set of core concepts.

Definition. (Scoping Core Concepts) *Given a set of scenes $y_j \in Y$ ($j \in \{1, ..., |Y|\}$), its associated concepts are denoted as C_{y_j}, a KG $\mathcal{G}$, and the set of concepts $RC(y_j, \mathcal{G})$ from $\mathcal{G}$ that are related to y_j. We define the* **scoping core concepts** *for scene y_j as follows: $SCC(y_j, \mathcal{G}) = RC(y_j, \mathcal{G}) \cap C_{y_j}$.*

Note that we do not give a specific definition of $RC(y_j, \mathcal{G})$, as one can use different similarity measures to define such relatedness, even using different ones for SCC and ICC. There are some limitations for SCC though. Firstly, SCC does not formally guarantee that different scenes come with different sets of core concepts, failing to make SCC some kind of identifier of a given scene. Secondly, SCC might include some knowledge graph related concepts that might not be crucial for a given scene. To address these two limitations, we introduce the notion of Identifier Core Concepts.

Definition. (Identifier Core Concepts) *Given a set of scenes $y_j \in Y$ ($j \in \{1, ..., |Y|\}$), its associated images and concepts $\mathcal{D}_{y_j}$ and C_{y_j}, respectively, and a KG $\mathcal{G}$, we assume that:*

- *$Count(y_j, p) \subseteq C_{y_j}$ is the set of overlapping ground truth concepts, from C_{y_j}, over at least $p\%$ of the images in $\mathcal{D}_{y_j}$,*
- *P_c is the highest percentage such that, for any $i, j \in \{1, ..., |Y|\}$, $i \neq j$, $Count(y_i, P_c) \neq Count(y_j, P_c)$,*
- *$TopkOfCount(y_j)$ is the set of the top k concepts of $Count(y_j, P_c)$,*
- *$SCount(y_j, \mathcal{G}, p)$ is the set of overlapping ground truth concepts, from $(RC(y_j, \mathcal{G}) \cap C_{y_j}) \cup TopkOfCount(y_j)$, over at least $p\%$ of the images in $\mathcal{D}_{y_j}$,*
- *P_{sc} is the highest percentage such that, for any $i, j \in \{1, ..., |Y|\}$, $SCount(y_i, \mathcal{G}, P_{sc}) \neq SCount(y_j, \mathcal{G}, P_{sc})$).*

We define the **identifier core concepts** *for scene y_j as follows: $ICC(y_j, \mathcal{G}) = SCount(y_j, \mathcal{G}, P_{sc})$.*

We consider the balance between concepts in the KG and annotated concepts of y_j, by including the top k (in our experiments, $k = 2$) most popular annotated concepts, no matter whether they are in the KG or not. As ICC is more selective, it often has a smaller size than SCC (cf. Appendix A).

3.3 MODEL PREDICTION EXPLANATIONS

In this section, we will make use of the concepts learned by neurons (Sec. 3.1) and SCC and ICC for scenes (Sec. 3.2) to provide two kinds of explanations for model predictions: prediction explanation and post-prediction explanation. For this purpose, we propose some metrics accordingly.

Prediction explanations (PE) are explanations provided together with predictions, with ground truth (target) scene unknown. Given an image x_i, the concepts $IC(x_i)$ learned by a neuron, a scene y_j, and its core concepts $CC_l(y_j)$, where $CC_l \in \{SCC, ICC\}$. We propose the consistency metric (difference metric, similarity metric) for measuring the consistency (difference, similarity, resp.) between the learned concepts from the neuron and core concepts from scene y_j:

$$CM(x_i, y_j) = \frac{|IC(x_i) \cap CC_l(y_j)|}{|CC_l(y_j)|} \tag{3}$$

$$DM(x_i, y_j) = \frac{|IC(x_i) \setminus CC_l(y_j)|}{|CC_l(y_j)|} \tag{4}$$

$$SM(x_i, y_j) = \frac{|IC(x_i) \cap CC_l(y_j)|}{|IC(x_i) \cup CC_l(y_j)|} \tag{5}$$

Note that y_j is the predicted scene for PE. The larger (smaller) the CM and SM (DM) scores become, the smaller the gap between the learned concepts and the scene. PE metrics can be used to optimise prediction performance (cf. Sec. 3.5).

Post-prediction explanations (PPE) are explanations when both predicted and target scene are known. Given an image x_i of scene y_t, and the scene y_p predicted by the model, the task here is to explain why the prediction is wrong, i.e. why $y_t \neq y_p$. One would expect that the learned concepts should be closer to the predicted scene (i.e., $CM(x_i, y_p) > CM(x_i, y_t)$ and $SM(x_i, y_p) > SM(x_i, y_t)$) and be more different from the target scene (i.e., $DM(x_i, y_p) > DM(x_i, y_t)$). Thus, the consistency metric for the set D_f of false predictions (as scene y_p) can be defined as follows:

$$CM^{FP} = \frac{|\{x_i \in D_f | CM(x_i, y_p) > CM(x_i, y_t)\}|}{|D_f|} \tag{6}$$

The difference and similarity metrics, denoted as DM^{FP} and SM^{FP} can be defined respectively.

Given an image x_i of scene y_t, the task here is to explain why the prediction is correct. We propose to compare the set of images with true prediction (D_t) against those with false prediction (D_f), with the expectation that the consistency metric over the truly predicted images CM^{TP} of D_t should be larger than that over the falsely predicted images ($CM^{TP} > CM^{T\text{-}FP}$). Thus, the consistency metric for the set D_t of truly predicted images and that for the set D_f of falsely predicted images in scene y_t can be defined as follows:

$$CM^{TP} = \frac{\Sigma_{x_i \in D_t} CM(x_i, y_t)}{|D_t|}, \qquad CM^{T\text{-}FP} = \frac{\Sigma_{x_i \in D_f} CM(x_i, y_t)}{|D_f|} \tag{7}$$

Similarly, we can define similarity metric for D_t and D_f, denoted as SM^{TP} and $SM^{T\text{-}FP}$, respectively, with the expectation that $SM^{TP} > SM^{T\text{-}FP}$.

3.4 Explanation Optimization via Concept Filtering

In this section, we propose a KG-based approach to optimize explanation via concept filtering. In the context of image classification and object detection, there could be a large number of concepts and many of which might have similar semantics, i.e. *armchair* and *chair*. This could lead to misleading or even wrong explanations for incorrect predictions, i.e. "Figure 1(a) is predicted to be a *utility room* because there is no chair" might not be correct, since an *armchair* is also a *chair* which could be seen in a *bedroom* (cf. ConceptNet description of *bedroom* in Figure 1(a)). To address this challenge, given each set of scene associated concepts C_{y_j}, we compute the embeddings of the concepts in C_{y_j} and align them to concepts in a KG like ConceptNet, using classic KG embeddings techniques, such as TransE Bordes et al. (2013), Dismult Yang et al. (2014) and TransD Ji et al. (2015), then group them w.r.t. their distances, into clusters $Cl_1(C_{y_j}), ..., Cl_r(C_{y_j})$. One can transform C_{y_j} into $CF(C_{y_j})$ by selecting one representative concept in each cluster $Cl_i(C_{y_j})(1 \leq i \leq r)$ to represent all concepts in $Cl_i(C_{y_j})$. Our *hypothesis* is that replacing C_{y_j} with $CF(C_{y_j})$ could help optimise model prediction explanation and compositional explanation of neuron behaviors.

Model Prediction Explanation Replacing C_{y_j} with $CF(C_{y_j})$ will affect the construction of SCC and ICC, and thus the metrics proposed in Sec. 3.3.

Compositional Explanation of Neuron Behavior The procedure described in Eq.(2) can only produce explanations from the fixed, pre-defined concepts in C_{y_j}. We follow CEN Mu & Andreas (2020), to combinatorially expand the set of possible explanations to include logical forms defined inductively over C_{y_j}, using three operations including disjunction (OR), conjunction (AND), and negation (NOT) for individual neurons. For each formula length, it exhaustively searches the overall concepts. Replacing C_{y_j} with $CF(C_{y_j})$ will affect the way that IoU is calculated in that the IoU values in a cluster are aggregated into one value for the representative concept for each cluster.

3.5 Model Manipulation

In this section, we study whether the core concepts (from Sec. 3.2) could help to manipulate model behavior, such as correcting a false prediction or corrupting a true prediction.We also propose using PE metrics (from Sec. 3.3) for re-training.

Neurons' learned concepts: whole layer (%)			
	CM^{FP}	DM^{FP}	SM^{FP}
Top_10	43.04	19.13	42.77
SCC	**78.51**	87.30	69.85
ICC	51.43	69.32	29.58
Neurons' learned concepts: highest IoU (%)			
Top_10	34.61	19.12	34.56
SCC	65.77	85.76	64.07
ICC	49.42	67.76	42.24
Neurons' learned concepts: threshold (%)			
Top_10	42.52	18,69	42.42
SCC	78.03	**87.38**	**72.13**
ICC	50.32	69.81	34.11

Table 1: Results of false prediction explanation. Top_10 means the top 10 concepts of scene as CC.

Neurons' learned concepts: whole layer				
	Consistency Metrics		Similarity Metrics	
	CM^{TP}	CM^{T_FP}	SM^{TP}	SM^{T_FP}
Top_10	0.11	0.06	0.04	0.02
SCC	0.13	0.10	0.08	0.06
ICC	**0.54**	**0.47**	0.23	0.23
Neurons' learned concepts: highest IoU				
Top_10	0.09	0.05	0.07	0.04
SCC	0.06	0.05	0.06	0.05
ICC	0.26	0.22	0.21	0.18
Neurons' learned concepts: threshold				
Top_10	0.11	0.06	0.07	0.04
SCC	0.12	0.09	0.10	0.08
ICC	0.50	0.44	**0.35**	**0.32**

Table 2: Results of true prediction explanation.

Neuron Identification using CC Given a trained classification model, we identify the positive and negative neurons by calculating contribution score to see the model behavior. The contribution score for the neurons f_t, over images x_i in the scene y_j with true prediction, can be calculated as follows:

$$Con\_Score(f_t) = \sum_{x_i \in D^T_{y_j}} (P(x_i, CC_l) - N(x_i, CC_l)) \tag{8}$$

where $P(x_i, CC_l)$ and $N(x_i, CC_l)$ are the number of IC in and not in CC_l, respectively.

For a true prediction, we disable top-k positive neurons (by setting the neuron features to 0 Bau et al. (2019); Mu & Andreas (2020)) for the corresponding scene and see whether the model still correctly predicts the scene. For a false prediction, we disable top-k negative neurons for the corresponding scene and see whether the model can make better prediction. k is set to 20 in our evaluation.

Re-training using CC The core concepts measure important components of the scene, which can play an important role for scene classification, as shown in Figure 1. Thus we propose to integrate core concepts into the model to further improve its performance. In the original models, the training objective is scene loss $\mathcal{L}_s$. We add another core concept loss $\mathcal{L}_c = -\sum \log \mathcal{P}(c^*|\theta)$, where $c^* \in C$ is the golden concept. For example, given a *bedroom* image with concepts of *bed, TV and fridge*, the new objective will let model pay more attention to the core concepts (*bed* and *TV*).

Re-training using PE We use a classical classifier SVM Cesa-Bianchi et al. (2006), but not an arbitrary neural network, as it will not introduce unexplained factors. For training the classifier, we utilize three types of features: (1) the features of metrics CM, SM and DM; (2) the MRR (mean reciprocal rank) feature which integrates the three metrics over all scenes; (3) the hidden states which learned by the original CNN model.

4 Experiments

4.1 Dataset and Pre-trained Model

For testing we use two scene datasets ADE20k Zhou et al. (2017) and Opensurfaces Bell et al. (2014) with atomic concepts annotated.

ADE20K is a challenging scene parsing benchmark with pixel-level annotations, which contains 22,210 images. There are 1,105 unique concepts in ADE20k, categorized by scene, object, part, and color, and each image belongs to a scene. We utilize the version from CEN Mu & Andreas (2020)[4].

Opensurfaces is a large database created from real-world consumer photographs. It contains 25,329 images which are annotated with surface properties, including material, color and scene[5]. We remove the scene with less than 10 images, and delete the images that are not annotated with material.

[4]http://github.com/jayelm/compexp

[5]http://opensurfaces.cs.cornell.edu/intrinsic/

Neurons' concepts: whole layer (%)			
Methods	CM^{FP}	DM^{FP}	SM^{FP}
Top_10	58.76	26.14	58.82
SCC	**81.17**	**85.97**	**74.74**
ICC	55.73	69.85	37.39

Table 3: Integrating concept filtering for false prediction.

Neurons' concepts: whole layer				
Methods	Consistency Metrics		Similarity Metrics	
	CM^{TP}	$CM^{T\text{-}FP}$	SM^{TP}	$SM^{T\text{-}FP}$
Top_10	0.15	0.09	0.06	0.03
SCC	0.25	0.19	0.18	0.15
ICC	**0.66**	**0.60**	**0.41**	**0.38**

Table 4: Integrating concept filtering for true prediction.

We evaluate four popular CNN models with different network architectures, namely ResNet-18 He et al. (2016), ResNet-50, DensenNet-161 Huang et al. (2017) and AlexNet Krizhevsky et al. (2017), which have 512, 2,048, 2,208 and 256 units of the final layer to probe for concepts respectively.

4.2 Evaluation of Post-Prediction Explanations

Results of False Prediction Explanation: For false prediction explanation, we expect to have higher scores on the three metrics (CM, DM, SM). Here are three observations from Table 1: (1) when compared to the results of the baseline method Top_10, both SCC and ICC achieve significant better results, indicating our core concepts is effective and reasonable; (2) all the best scores (across CM^{FP}, DM^{FP} and SM^{FP}) are from SCC. This can be explained from the fact that SCC is normally larger than ICC: if some concepts are not in SCC, then they are most likely to be incorrect. On the other hand, ICC is more selective, thus it might exclude some (partially) correct concepts; (3) among the three methods to represent neurons' learned concepts in Sec. 3.1, the threshold-based method achieves better results.

Results of True Prediction Explanation: For true prediction explanation, we expect to observe that CM^{TP} and SM^{TP} are larger than $CM^{T\text{-}FP}$ and $SM^{T\text{-}FP}$ respectively. The bigger the gap between CM^{TP} and $CM^{T\text{-}FP}$ and between SM^{TP} and $SM^{T\text{-}FP}$, the better the results are. As a whole, all the items in Table 2 are satisfied with our assumption. The best results are from ICC. This is understandable, since ICC is more selective, thus compared to SCC, they are less likely to involve noise. We also conduct the same experiments for false and true predictions on Opensurfaces, and achieve similar results as on ADE20k. The detailed results are shown in Appendix C.

Integrating Concept Filtering: Tables 3 and 4 show the results for false prediction explanation and true prediction explanation when using concept filtering to simplify the concept sets. For false prediction, SCC achieves the best performance compared to ICC and Top_10. For true prediction, once again the results of CM^{TP} and SM^{TP} are larger than $CM^{T\text{-}FP}$ and $SM^{T\text{-}FP}$ respectively. In addition, results are better than corresponding results without concept filtering in Table 1 and 2.

Model Prediction Explanation on Different Models: We further implement our method on different architectures to verify the generalization. We randomly select 1000 samples from the ADE20K data for the experiment by considering the effect of time efficiency. The results are shown in Table 5. SCC has better results than ICC over every model, which is similar to the observation over ResNet-18 from Table 1. For true prediction, all three models achieve significantly better results which can be found in Appendix D.

Models	CC	CM^{FP}	DM^{FP}	SM^{FP}
ResNet-50	SCC	**80.85**	**89.36**	**74.46**
	ICC	53.19	69.15	36.17
DensenNet-161	SCC	**78.63**	**81.32**	**45.65**
	ICC	55.47	57.54	21.29
AlexNet	SCC	**76.58**	**83.26**	**71.34**
	ICC	60.23	68.33	31.57

Table 5: Results of false prediction on different models.

4.3 Evaluation of Compositional Explanations

As KG embedding techniques could have an impact on the number of optimal clusters, as well as on the interpretability of neurons, we ran some experiments with ResNet-18 over the ADE20k dataset to evaluate their impact. In particular we evaluated the impact of TransE Bordes et al. (2013), Dismult Yang et al. (2014), ProjE Shi & Weninger (2017) and TransD Ji et al. (2015) on the (1) optimal number of clusters, and (2) quality of interpretability, measured using IoU similarly described as in CEN. The final number of cluster also captures the final number of core concepts to be considered for explanation, as a cluster is described by a unique concept in ConceptNet. The knowledge graph used for computing the embeddings is a subset of Concept-

Net. In particular, we extracted all ADE20k labels as main concepts, as well as direct 1-hop and 2-hop neighbors of ADE20k labels in ConceptNet. We applied fuzzy matching for 0.1% of ADE20k labels due to some misalignment between ADE20k labels and ConceptNet concepts. The IoU gain is measured by capturing the interpretability improvement from (A) labeling with no clustering strategy to (B) labeling with a k-clustering strategy with (k: Nb. of Cluster) using Embeddings. The IoU gain is defined as ${}^{(B-A)}/_{A}$. Table 6 captures the main results[6]. As described in this table, 168 is the best number of cluster for ADE20k using ResNet-18 with TransE embeddings. In other words, reducing the number of classes to 168 has the advantage of exposing more interpretable units in ResNet-18 for ADE20k, and therefore we reduce the number of core concepts from 512 to 168. TransE is the best performing embeddings strategy, with a best at 26.3% improvement using 168 clusters compared with no clustering strategy i.e., 512 neurons in the context of ResNet-18.

Nb. of Clusters	IoU Gain (%)	Embeddings
168	**+26.30**	TransE
166	+26.09	TransE
167	+26.05	TransE
169	+26.00	TransE
165	+25.77	TransE
170	+25.36	TransE
171	+25.33	TransE
181	+25.36	TransD
180	+25.28	TransD
182	+25.15	TransD
179	+25.03	TransD
146	+25.01	Distmult

Table 6: Embeddings Techniques vs. Interpretability Gain.

We also conduct a set of ablation studies on compositional explanations to verify the effectiveness of our MinMax-based NetDissect which is provided in Appendix F.

4.4 Evaluation of Model Manipulation

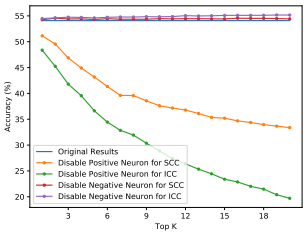


Figure 2: Neuron contribution on ADE20k.

Results of Neuron Identifying on ADE20K and Opensurfaces: Figure 2 shows the model performance when we disable the positive neurons or negative neurons. We have the following three observations: (1) when negative (resp. positive) neurons are disabled, the model performance is improved (resp. decreases), proving that the CC facilitates identifying important neurons during decision-making of the model; (2) model accuracy tends to decrease as the number of inhibitory positive neurons increases; (3) compared to SCC, ICC can better identify both the positive and negative neurons. As shown in Figure 3, we can also see that the model performance decreases when we disable the positive neurons. Note that the accuracy on ADE20k decreases more, indicating that more positive neurons could be detected on ADE20k.

When inhibiting negative neurons, the model performance is improved on both ADE20k and Opensurfaces. The results on Opensurfaces do not show as much growth as the results on ADE20k. This is probably because Opensurfaces mainly focuses on annotating the surface property, such as material, which makes the core concept of different scenes with limited differentiation. For example, concepts *painted*, *wood* will be core concepts for most scenes, such as *living_room*, *family_room*, *office* and *staircase*. From the overall experimental results, with the help of core concepts, our method can effectively identify the positive and negative neurons, and then augment the model performance.

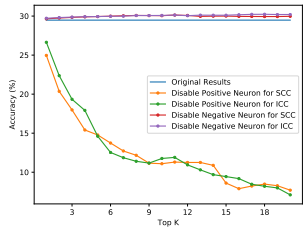
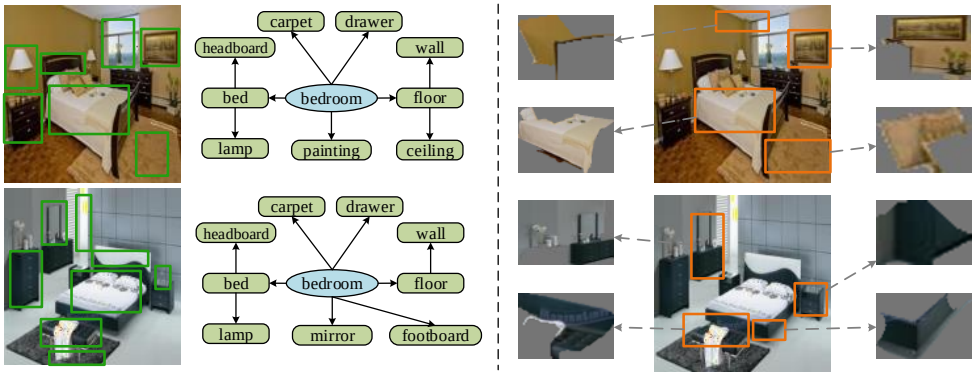


Figure 3: Neuron contribution on Opensurface.

Results of Re-training using CC: From Figure 2 and 3, we can see that the performance change of disabling negative neurons is not as large as disabling positive neurons on both datasets. This is reasonable, since the model we explain is trained with its parameters fixed and it is difficult to correct false predictions by only removing some negative neurons. However, the result improvements on different datasets still indicate that our method is effective to retrieve negative neurons.

To address this challenge, we re-train the initial models with the help of CC, and the results are shown in Figure 4. In Figure 2 and 3, the experiments are based on the model ResNet18, and the results have improved about 1.3% by removing the negative neurons. However, in Figure 4, the corresponding performance of ResNet18 has improved 3.27%. On the other models, the results by

[6]Complete results: in the material zip file provided.



(a) ICC and the relations for bedroom. (b) Concept generated by ACE.

Figure 5: Concepts of the scene bedroom by our proposed ICC (left) and ACE (right).

adding ICC are all improved. Compared to SCC, utilizing the ICC for re-training model is more effective. We also conduct case studies in Appendix G.

The above two parts mainly focus on manipulating the CNN model behavior with the help of CC, such as identifying the useful neurons and re-training from scratch. In the following part, we further verify the effectiveness of metrics CM, SM and DM on with-prediction explanation. The implementation details are shown in Appendix H.

Results of Re-training using PE: The results are shown in Table 7, and ResNet18 is the fundamental model. The results are improved for SCC and ICC on both datasets. ICC-based SVM on ADE20K achieves the best performance with 67.11, and outperforms the basic ResNet18 by a large margin of 14.15, which amouts to 26.7% improvement.

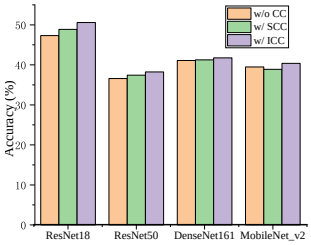


Figure 4: The results of model re-training using CC.

4.5 Evaluation of Core Concepts

Method	Acc. (%)	
	ADE20K	Opensurfaces
ResNet18	52.96	29.26
SVM (SCC)	66.54	31.62
SVM (ICC)	**67.11**	**32.27**

Table 7: Results of PE.

Analysis of SCC and ICC: SCC describes the scenario more comprehensively while keeping a higher level of coverage. ICC mainly focuses on higher precision. These characteristics can be reflected in the following three aspects: (1) the concept count of SCC is often larger than ICC; (2) SCC has better results on the false prediction explanation as shown in Table 1; (3) when identifying positive (negative) neurons to change the model performance, we need more scene-specific concepts, and thus ICC achieves better results as shown in Figures 2 and 3.

Compare to Visual Concept-based Explanation: We compare the identifier core concepts to ACE Ghorbani et al. (2019) on ADE20k. The results are shown in Figure 5. We have the following three observations: (1) many concepts generated by ACE are still difficult for human to understand; (2) the concepts in Figure 5(b) are independent of each other, and the image of the scene cannot be identified from only a limited number of concepts; (3) in contrast, ICC can be used to facilitate identifying scene, as shown in Figure 5(a). More detailed cases on SCC are given in Appendix B.

5 Conclusion and Outlook

In this study, we investigated knowledge-aware explanation of neural network predictions on image scenes. We proposed two types of core concepts (i.e., SCC and ICC) based on KGs to help identify scenes, assisting prediction explanation and further augmenting model performance. Extensive experiments demonstrate that our methods can make a better explanation with visual concepts, which enables human beings to better understand the predictions. Our experimental results also verify that inhibiting the positive and negative neurons identified with SCC and ICC can manipulate the model behavior, and that PE based retraining leads to significant performance improvements.

REFERENCES

Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 2018. doi: 10.1109/ACCESS.2018.2870052.

Sebastian R. Bader, Irlán Grangel-González, Priyanka Nanjappa, Maria-Esther Vidal, and Maria Maleshkova. A Knowledge Graph for Industry 4.0. In *Proceedings of the 17th Extended Semantic Web Conference (ESWC 2020)*, pp. 465–480, 2020.

David Bau, Bolei Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3319–3327, 2017.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B. Tenenbaum, William T. Freeman, and Antonio Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Trans. on Graphics (SIGGRAPH)*, 33(4), 2014.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. Hierarchical classification: Combining bayes with svm. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 177–184, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143867. URL https://doi.org/10.1145/1143844.1143867.

Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Interpretable deep models for icu outcome prediction. *AMIA Annual Symposium Proceedings*, 2016:371–380, 02 2017.

Zhuo Chen, Jiaoyan Chen, Yuxia Geng, Jeff Z. Pan, Zonggang Yuan, and Huajun Chen. Zero-shot visual question answering using knowledge graph. In *Proc. of The 20th International Semantic Web Conference, ISWC*, pp. 146–162, 2021.

Laura Dietz, Alexander Kotov, and Edgar Meij. Utilizing knowledge graphs for text-centric information retrieval. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2018)*, pp. 1387–1390, 2018.

Yunhao Ge, Yao Xiao, Zhi Xu, Meng Zheng, Srikrishna Karanam, Terrence Chen, Laurent Itti, and Ziyan Wu. A peek into the reasoning of neural networks: Interpreting with structural visual concepts. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2195–2204, 2021. doi: 10.1109/CVPR46437.2021.00223.

Yuxia Geng, Jiaoyan Chen, Zhuo Chen, Jeff Z. Pan, Zhiquan Ye, Zonggang Yuan, Yantao Jia, and Huajun Chen. Ontozsl: Ontology-enhanced zero-shot learning. In *Proc. of The Web Conference 2021,*, pp. 3325–3336. ACM / IW3C2, 2021.

Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. *Advances in Neural Information Processing Systems*, 32, 2019.

Yu Gu, Tianshuo Zhou, Gong Cheng, Ziyang Li, Jeff Z. Pan, and Yuzhong Qu. Relevance Search over Schema-Rich Knowledge Graphs. In *Proc. of the 12th ACM International WSDM Conference (WSDM2019)*, pp. 114–122, 2019.

Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. *CoRR*, abs/2003.00911, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, 2017. doi: 10.1109/CVPR.2017.243.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 687–696, 2015.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Adam Kortylewski, Qing Liu, Angtian Wang, Yihong Sun, and Alan Yuille. Compositional convolutional neural networks: A robust and interpretable model for object recognition under occlusion. *International Journal of Computer Vision*, 129(3):736–760, 2021.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, may 2017. ISSN 0001-0782. doi: 10.1145/3065386.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2016.11.008.

Jesse Mu and Jacob Andreas. Compositional explanations of neurons. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Marco Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pp. 97–101, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-3020.

Atefeh Shahroudnejad. A survey on understanding, visualizations, and explanation of deep neural networks. *CoRR*, abs/2102.01792, 2021.

Baoxu Shi and Tim Weninger. Proje: Embedding projection for knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017.

Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pp. 4444–4451. AAAI Press, 2017.

Ilia Stepin, Jose M. Alonso, Alejandro Catala, and Martín Pereira-Fariña. A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access*, 9:11974–12001, 2021. doi: 10.1109/ACCESS.2021.3051315.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. ICML'17, pp. 3319–3328. JMLR.org, 2017.

Yue Tian and Guanjun Liu. Mane: Model-agnostic non-linear explanations for deep learning model. In *2020 IEEE World Congress on Services (SERVICES)*, pp. 33–36. IEEE, 2020.

Saurabh Varshneya, Antoine Ledent, Robert A. Vandermeulen, Yunwen Lei, Matthias Enders, Damian Borth, and Marius Kloft. Learning interpretable concept groups in cnns. In Zhi-Hua Zhou (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 1061–1067. International Joint Conferences on Artificial Intelligence Organization, 8 2021. Main Track.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. *Advances in Neural Information Processing Systems*, 33:20554–20565, 2020.

Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 818–833, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10590-1.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5122–5130, 2017. doi: 10.1109/CVPR.2017.544.

Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *CoRR*, abs/1702.04595, 2017.

A EXAMPLES OF CORE CONCEPTS

The core concepts are shown in Table 8. SCC is based on the whole concepts in the scene, whenever a concept has relation with the scene in the ConceptNet, it is considered as a core concept, which will inevitably introduce redundant concepts. For example, *coat* and *streetcar* are not important concepts to scene *library*. In contrast to SCC, ICC considers the relations between the concept and the scene while computing the importance of the concept to the scene, so the core concepts in ICC are more concentrated. For example, in scene *library*, the concept count of ICC is 17 less than the SCC.

Scene	CC	Count	Core Concepts
library	SCC	27	wall, book, windowpane, light, desk, armchair, ceiling, lamp, fluorescent, counter, poster, cabinet, apron, top, exhibitor, chair, table, screen, grill, coat, paper, floor, stairs, streetcar, spotlight, stairway, bookcase
	ICC	10	wall, floor, bookcase, book, ceiling, windowpane, light, chair, table, desk
forest_path	SCC	10	tree, grass, path, plant, signboard, water, gate, wall, earth, person
	ICC	7	path, grass, tree, plant, earth, sky, bush

Table 8: Core concepts of the scene *library* and *forest_path*.

B LIST OF CASE ON SCC

In this section, we compare our scoping concept definition (SCC) and the concept-based explanation ACE as shown in Figure 6. We can obtain the same conclusion as that of ICC in Section 4.2 that the SCC based method improves the comprehensive description of the scene by integrating the KG, which in turn helps to identify the scene.

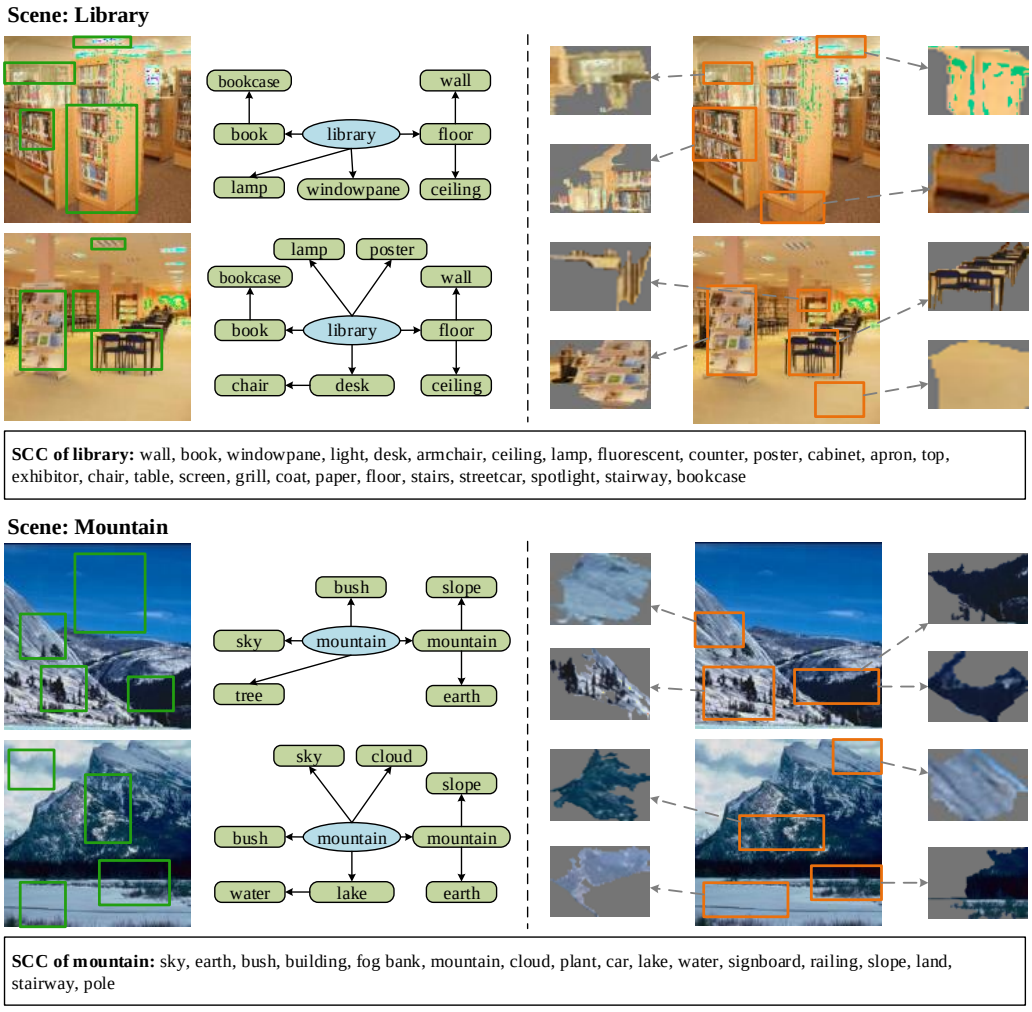
C MODEL PREDICTION EXPLANATION ON OPENSURFACES

Table 9 shows the results of false prediction, we can see that it achieves the similar performance as on ADE20k. On the other hand, SCC once again achieves the best performance.

For the true prediction explanation, as shown in Table 10, we can also obtain the same observations as on ADE20k. The results on different datasets prove that our method is effective and can be applied to different datasets.

Neurons' learned concepts: whole layer (%)			
	CM^{FP}	DM^{FP}	SM^{FP}
SCC	73.69	**84.09**	64.03
ICC	35.66	53.57	33.91
Neurons' learned concepts: highest IoU (%)			
SCC	67.63	40.03	66.61
ICC	35.26	34.41	34.75
Neurons' learned concepts: threshold (%)			
SCC	**74.51**	41.51	**73.23**
ICC	36.06	37.18	34.29

Table 9: Results of false prediction explanation on Opensurfaces. The table contains three blocks: top block, middle block, and bottom block. And each block is the results of two CC definitions on three different metrics.



(a) SCC and the relations. (b) Concept generated by ACE.

Figure 6: Concepts of scene bedroom by our proposed SCC (left) and ACE (right).

Neurons' learned concepts: whole layer				
	Consistency Metrics		Similarity Metrics	
	CM^{TP}	$CM^{F\text{-}TP}$	SM^{TP}	$SM^{F\text{-}TP}$
SCC	0.10	0.10	0.07	0.07
ICC	0.31	0.30	0.11	0.10
Neurons' learned concepts: highest IoU				
SCC	0.08	0.08	0.08	0.07
ICC	0.23	0.21	0.21	0.19
Neurons' learned concepts: threshold				
SCC	0.10	0.10	0.10	0.10
ICC	**0.32**	**0.30**	**0.28**	**0.27**

Table 10: Results of true prediction explanation on Opensurfaces.

D TRUE PREDICTION EXPLANATION ON DIFFERENT MODELS

Table 11 shows the results of true prediction explanation on different models. The results of ICC achieve better performance compared to the results of SCC, which further prove that ICC can reduce redundant concepts.

Models	CC	Consistency Metrics		Similarity Metrics	
		CM^{TP}	$CM^{F\text{-}TP}$	SM^{TP}	$SM^{F\text{-}TP}$
ResNet-50	SCC	0.14	0.11	0.11	0.09
	ICC	**0.54**	**0.50**	**0.37**	**0.35**
DesenNet-161	SCC	0.11	0.10	0.05	0.05
	ICC	**0.21**	**0.21**	**0.16**	**0.15**
AlexNet	SCC	0.09	0.08	0.07	0.06
	ICC	**0.36**	**0.34**	**0.28**	**0.26**

Table 11: Results of true prediction explanation on different models.

In the end, combining the results of false prediction explanation can prove that it is reasonable to define different core concepts.

E GROUPING DISTRIBUTION

Table 12 shows the grouping results, we test top k from 1 to 5. As the top k gets larger, the number of group get smaller. In top 5, there are only 10 groups, and each group has an average of 454.7 neurons and 410.8 concepts. Since we have 512 neurons at most, we only test up to the top 5.

Top K	Group Count	Avg Neuron	Avg Concepts
1	255	1	2.3
2	87	23.1	31.4
3	37	209.9	199.9
4	20	384.5	340.9
5	10	454.7	410.8

Table 12: Neuron grouping on the whole neurons. Avg Neuron and Avg Concepts are the average neurons and concepts for each group respectively.

Methods		IoU Score (%)			
		Len 1	Len 2	Len 3	Len 4
CEN (w/o color)		5.81	7.51	8.48	8.87
CEN (w/o color, w/o scene)		4.27	5.56	6.16	6.50
Grouping Top K (w/o color)	1	5.53	7.07	7.83	8.24
	2	5.63	7.21	8.06	8.43
	3	5.69	7.37	8.41	8.76
	4	5.77	7.48	8.46	**8.82**
	5	5.74	7.42	8.43	8.80
Grouping Top K (w/o color, w/o scene)	1	3.24	4.74	5.40	5.78
	2	3.78	4.98	5.69	6.55
	3	4.27	5.65	6.54	6.72
	4	4.38	5.66	6.59	**6.74**
	5	4.39	5.63	6.53	6.65

Table 13: Results of Compositional Explanations.

F USING MINMAX-BASED NETDISSECT

We also conduct a set of ablation studies on compositional explanations to verify the effectiveness of our MinMax-based NetDissect.

Table 13 reports our results. For each formula length, different neurons only calculate the corresponding concept set, with the following three steps: (S1) For each concept (with IoU score larger than the threshold T), we rank and get the neuron scores based on the number of images that each

neuron can predict the concept. (S2) For each concept, we select the top k neurons (regard as a concept-neuron pair). The neurons are divided into different groups by combining the concept-neuron pairs if they have the same neurons. (S3) For each neuron, we select the concepts that belong to the corresponding groups to calculate the logic expression.

The original ADE20k dataset has pixel-level annotations on scene, object, part, and color for each image. Since our goal is to explain model prediction of each image and the concepts are mainly about the part and object in the scene for per image, when explaining the model prediction we focus on explaining the concept corresponding to the neuron rather than the scene. On the other hand, there are 11 colors in total and each image has 9.92 colors on average. So color as a common concept, and our analysis does not consider the effect of color. As shown in Table 13, our grouping results (*Grouping Top K (w/o scene, w/o color)*) are significantly better than the baseline on IoU score, especially in the grouping top 4 with 0.24 improvements on formula length 4.

For a fair comparison, we also display our grouping results by adding the scene (*Grouping Top K (w/o color)*) in Table 13. Our grouping top 4 achieves comparable performance to the baseline.

Method		>20%	10-20%	5-10%	<5%
CEN	Len 1	0.2	5.27	23.44	71.09
	Len 2	0.2	10.16	33.98	55.66
	Len 3	0.59	12.11	40.23	47.07
	Len 4	0.78	13.68	43.95	41.6
Top 4	Len 1	0.39	7.81	40.43	51.37
	Len 2	0.59	12.5	47.85	39.06
	Len 3	0.78	15.04	48.24	35.94
	Len 4	0.98	14.84	47.46	36.72

Table 14: The distribution of IoU score for different formula lengths. Four columns represent different IoU score ranges.

In addition, we share the distribution of IoU score on different formula lengths in Table 14. From the results, we can see that in interval <5%, the percentage of our method is smaller than the CEN. However, for the other four intervals, the percentage of our method is larger than the CEN in terms of all four formula lengths, confirming once again that our method can effectively improve the compositional explanations of neurons.

	Target Scene: street		Predicted Scene: crosswalk	
	count	Intersection	count	Intersection
SCC	9	traffic light, bag, person, taillight, leg, windowpane, car, road, license plate	10	sidewalk, hand, taillight, leg, car, windowpane, bag, road, building, traffic light
ICC	8	sidewalk, windowpane, car, road, building, sky, person, traffic light	6	sidewalk, person, road, building, crosswalk, traffic light

Table 15: Case Study. The Intersection operation in the table means the intersection between the neurons' learned concepts (method of threshold) and CC of target scene (predicted scene).

G CASE STUDY

As shown in Table 15, the intersection count of SCC for predicted scene *crosswalk* is larger than the count of target scene *street*, indicating the neurons are closer to the predicted scene. On the other hand, For ICC, although the intersection count of *crosswalk* is smaller than the *street*, it has the key concepts *crosswalk* and *traffic light* which are the CC of scene *crosswalk*. In addition, compare to SCC, ICC can identify more key concepts to the scene while reducing redundant concepts.

As conclusion, for the current scene *street*, the model predicts to *crosswalk* is explainable and reasonable.

H IMPLEMENTATION DETAILS

Implementation details of our proposed models are as follows. In all experiments, we evaluate our method on four popular CNN models with different network architectures, i.e. ResNet-18, ResNet-50, DensenNet-161 and AlexNet. For a fair comparison with other baselines, all the models are trained on the dataset Places365[7], and test on the whole datasets ADE20K and Opensurfaces. In last two parts of Sec. 3.5, we re-train the CNN models on ADE20K and Opensurfaces. During the re-training phase, we split the data to train, dev and test by 80%, 10% and 10% for each scene. In Figure 4 and Table 7, we report model performance on the new test set.

[7] http://places2.csail.mit.edu/models_places365/