Knowledge-Aware Neuron Interpretation for Scene Classification

Yong Guan[1,2], Freddy Lécué[3*], Jiaoyan Chen[4], Ru Li[2*], Jeff Z. Pan[5*]
[1] Department of Computer Science and Technology, Tsinghua University, China
[2]School of Computer & Information Technology, Shanxi University, China
[3]Inria, France
[4]Department of Computer Science, The University of Manchester, UK
[5]School of Informatics, The University of Edinburgh, UK
gy2022@mail.tsinghua.edu.cn, freddy.lecue@inria.fr, liru@sxu.edu.cn, j.z.pan@ed.ac.uk,

Abstract

Although neural models have achieved remarkable performance, they still encounter doubts due to the intransparency. To this end, model prediction explanation is attracting more and more attentions. However, current methods rarely incorporate external knowledge and still suffer from three limitations: (1) **Neglecting concept completeness**. Merely selecting concepts may not sufficient for prediction. (2) **Lacking concept fusion**. Failure to merge semantically-equivalent concepts. (3) **Difficult in manipulating model behavior**. Lack of verification for explanation on original model. To address these issues, we propose a novel knowledge-aware neuron interpretation framework to explain model predictions for image scene classification. Specifically, for concept completeness, we present core concepts of a scene based on knowledge graph, ConceptNet, to gauge the completeness of concepts. Our method, incorporating complete concepts, effectively provides better prediction explanations compared to baselines. Furthermore, for concept fusion, we introduce a knowledge graph-based method known as Concept Filtering, which produces over 23% point gain on neuron behaviors for neuron interpretation. At last, we propose Model Manipulation, which aims to study whether the core concepts based on ConceptNet could be employed to manipulate model behavior. The results show that core concepts can effectively improve the performance of original model by over 26%.

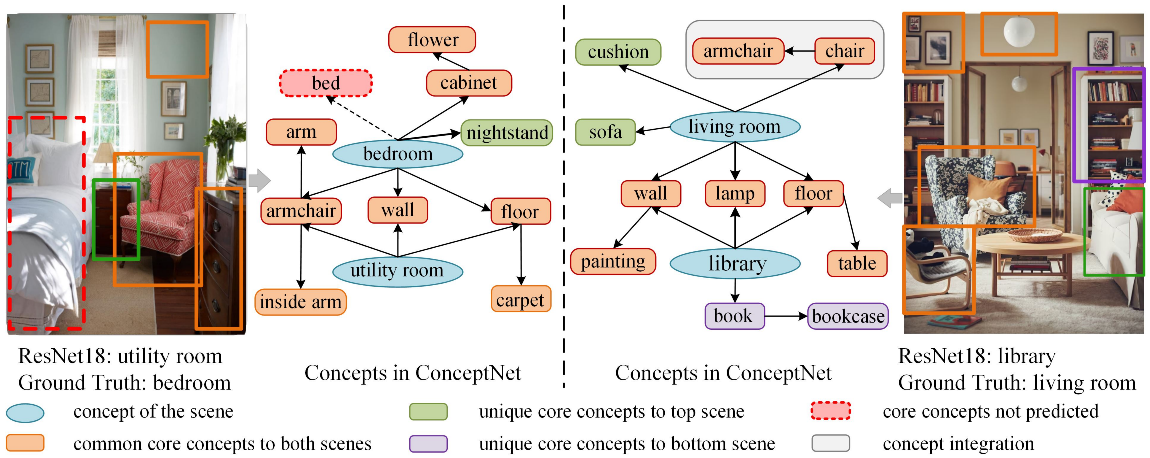
Introduction

Deep neural network (DNN) architectures are designed to be increasingly sophisticated, with scaled up the model size, and have been achieving unprecedented advancements in various areas of artificial intelligence (Thoppilan et al. 2022; OpenAI 2023). Despite their strengths, DNNs are not fully transparent and often perceived as "black-box" algorithms, which can impair users' trust and hence diminish usability of such systems (Brik et al. 2023). As shown in Figure 1(a), the model predicts the image as `utility room`, which is different from the ground truth (target label) `bedroom`. However, it is unclear why the model predicts this label, making it hard to understand, debug and improve.

There has been a growing interest in exploring explanations of model predictions (Chen et al. 2018; Deng et al. 2019), which, generally speaking, could be categorized into two methods: functional analysis and decision analysis (Shahroudnejad 2021). Functional analysis methods try to capture overall behavior by investigating the relations between the decision and the image, using saliency map (Akhtar and Jalwana 2023), occlusion techniques (Kortylewski et al. 2021), and rationale (Jiang et al. 2021). Such methods typically lack in-depth understanding of internal modules of the model, often failing to provide comprehensive insights into the decision-making process. The decision analysis methods explore explanation by analyzing the internal components' behavior, such as decomposing the network classification decision into contributions of input elements (Montavon et al. 2017; Tian and Liu 2020). Furthermore, studying neuron-level explanation enables more accurate orientation and editing of the decision-making process (Teotia, Lapedriza, and Ostadabbas 2022). However, they do not offer the most intuitive explanations that are easily understandable to humans, and the link between the decision and internal components is not obvious.

Some studies attempt to utilize concepts to enhance the interpretation of model decisions, establishing relations between the decision and the input image through a selected number of concepts, such as ACE (Ghorbani et al. 2019), ConceptSHAP (Yeh et al. 2020), and VRX (Ge et al. 2021). Although the decision is explained by presenting a set of concepts found within image, these methods still exhibit certain three key limitations. (1) **Neglecting concept completeness**. These methods select a set of concepts salient to the corresponding scene, but they do not guarantee that these concepts are sufficient to explain the prediction. As shown in Figure 1(a), the model selects a set of salient concepts, including *armchair*, *floor*, *wall*, and more. However, their prediction mislabels scene as `utility room` instead of `bedroom`, due to an incomplete concept set (Zhu et al. 2015) that overlooks the *bed* concept. (2) **Lacking concept fusion**. These methods merely group segments based on resemblance, but they do not merge semantically-equivalent concepts. As depicted in Figure 1(b), concepts *armchair* and *chair* can be fused (Wang et al. 2015), as they convey identical meanings. (3) **Hard to manipulate model behavior**. These methods mainly focus on explanation, but they do not provide guidance on how to rectify mistakes made by the original model.

*Corresponding authors



(a) Core concepts absent. (b) Non-core concepts dominate.

Figure 1: Example of false prediction explanations[1]. Concepts that are essential to the meaning of a scene are called core concepts, such as *bed* in scene `bedroom`. Concepts that are not necessary for understanding the scene and can be omitted or ignored are called non-core concepts, such as *book* and *bookcase* in `living room`.

To address the above problems, we propose a novel knowledge-aware neuron interpretation framework for scene classification. As well-defined knowledge will help to further enhance the model explanation and facilitate human understanding. Specifically, for **concept completeness**, we present core concepts (CC) of a scene to gauge the completeness of concepts. CC refers to the fundamental elements that collectively constitute the scene (Kozorog and Stanojević 2013); e.g., *bed*, *cabinet*, *armchair*, *floor*, *wall*, *nightstand* and *lamp* are the CC of scene `bedroom`. To formulate the CC, we leverage knowledge graphs (KG) (Pan et al. 2017b,a), such as ConceptNet. Additionally, we introduce the MinMax-based NetDissect method to establish links between neuron behavior and concepts at the neuron level. As for **concept fusion**, we introduce a Concept Filtering method, which effectively merge semantically-equivalent concepts based on ConceptNet in order to enhance the existing neuron interpretation. Furthermore, for **manipulate model behavior**, we propose Model Manipulation, which aims to study whether the CC obtained from ConceptNet could be employed to manipulate model behavior, such as identifying the positive/negative neurons in model, integrating CC into original model design phase. At last, our method, integrating CC, will help to answer a variety of interpretability questions across different datasets, such as ADE20k and Opensurfaces, and multiple models, such as ResNet, DenseNet, AlexNet and MobileNet.

- Do complete concepts base on core concepts provide benefit to model prediction explanation? We propose core concepts of scene derived from ConceptNet, along with three evaluation metrics, to establish the link between decisions and concepts in image. The experimental results show that our method, integrating complete concepts, achieves better results than the existing methods.
- Furthermore, do external knowledge through concept fusion improve existing neuron interpretation? We propose *Concept Filtering* method, which produces over 23% point gain on neuron behaviors for neuron interpretation.
- In addition, do explanations based on core concepts contribute to model performance? We propose both unsupervised and supervised methods based on core concepts extracted from ConceptNet to manipulate model behavior. The overall results prove that core concepts and related explanation metrics can help optimise the original model, leading to 26.7% of performance improvement[2].

Preliminaries

Neuron Interpretation

Neuron interpretation aims to improve the interpretability of models by understanding the neuron behavior. Observations of neurons (a.k.a hidden units) in neural networks have revealed that human-interpretable concepts sometimes emerge as individual latent variables. Thus, a pioneer work on interpreting neurons (Bau et al. 2017) designed a network dissection (*NetDissect*) tool to quantify the interpretability of a model and particularly its neurons.

Given a neural network f trained and used for prediction, f maps an image x_i to a latent representation that is also known as neuron features or units (e.g., unit 483 of ResNet-18 layer 4 in Figure 2), denoted as $\{f_1, f_2, ..., f_n\}$, where n denotes the dimension and f_t $(1 \leq t \leq n)$ is known as t-th neuron features. Given C the set of concepts of a given dataset, $L : (x_i, c) \mapsto \{0, 1\}$ is a concept function which

[1](a) The model does not predict the CC *bed* of `bedroom`, and thus mistakenly predicts the scene to be `utility room`. (b) The model mainly focuses on non-core concepts, including *book* and *bookcase*, which are CC of scene `library`.

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

(a) Input Image x. (b) Activation $F_{483}(x)$. (c) Activation over T_{483} (d) Concept `water`.

Figure 2: Unit 483 of layer 4 in ResNet-18, as a `water` detector (d) with a IoU score of .14 for pixelwise annotated input images x in (a) wrt. the upscaled unit activation map (b), determined by areas of significative activation (c).

indicates whether an image (region) x_i is an instance of a concept $c \in C$; e.g., $L(x_i, water) = 1$ means x_i is an image (region) containing water. *NetDissect* computes the most relevant concepts from C to the neuron feature f_t over the set of images x:

$$NWD(f_t, x, C) = argmax\{\sigma(A_t(x_i), L(x_i, c))\} \quad (1)$$

where σ is a measure function, such as intersection over union (IoU) and Jaccard index, while $A_t(x_i)$ is the activation of f_t for x_i, and can be scaled up to the mask resolution using bilinear interpolation. The unit f_t is regarded as a detector which selects the highest scoring concept. For example, unit 483 in Figure 2 is regarded as a `water` descriptor (d) for images (a) wrt. activation (b) using threshold (c).

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 scenes. Each image $x_i \in \mathcal{D}$ belongs to a scene $y_j \in Y$ and contains multiple concepts representing the scene in x_i; e.g., the image of Figure 1(a) is labelled as scene `bedroom` and contains concepts such as *wall*, *lamp*, *armchair*. For each scene y_j, it has multiple images in $\mathcal{D}$, denoted as $\mathcal{D}_{y_j} \subseteq \mathcal{D}$. $C_{y_j} \subseteq C$ refers to the set of associated concepts of y_j. $LC(x_i)$ is the set of learning concepts of neuron features f_t by computing the correlation between f_t and the corresponding concept set. In the rest of paper, assumed that the KG contains all concepts in C.[3]

For each image x_i, it has a label y_p predicted by f and a target (ground truth) label y_j. In this paper, we consider three tasks: *(T1)* model 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 interpretation: studying the effectiveness of external knowledge used in *T1* on existing neuron interpretation; *(T3)* model manipulation: exploiting CC used in *T1 and T2* to optimise model performance.

Approach

In this section, we present the knowledge-aware framework to address the three tasks mentioned above. Task *T1*, corresponding to the limitation of concept completeness, contains three sections: MinMax-based NetDissect, Core Concepts, and Model Prediction Explanations. Task *T2*, corresponding to the limitation of concept fusion, is detailed in Concept Filtering. Task *T3*, corresponding to the limitation of manipulate model behavior, is detailed in Model Manipulation.

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

MinMax-based NetDissect

MinMax-based NetDissect aims to learn which concepts are closest to the neuron. Following existing work (Bau et al. 2017; Guan et al. 2023), we use NetDissect to evaluate the alignment between each hidden unit and a set of concepts. Note that the original NetDissect method computes the general neuron behavior on the dataset level while ignoring features that are unique and useful for an individual image prediction. In contrast, we aim to learn the neuron behavior for individual scene, e.g., `bedroom`, to see whether concepts in scene can help explain model prediction. To achieve this, we propose a new variant MinMax-based NetDissect method to learn the neuron behavior for individual image. Formally, given f a neural network, f_t the *t*-th neuron in intermediate layer, C_{y_j} the associated concepts of y_j, x_i the target image, σ measure function, and A_t the activated neuron features of the *t*-th neuron, $L(x_i, c)$ the concept function where c is a concept in C_{y_j}, we have:

$$MM\text{-}NWD\ (f_t, x_i, C_{y_j}) = Ths\{\sigma(A_t(x_i), L(x_i, c))\} \quad (2)$$

We use IoU as the measure function σ. Thus, the concepts $LC(x_i)$ learned by a target neuron can be obtained from concept selection strategy[4] $Ths\{\cdot\}$. We consider three ways of selecting concepts that neuron learns.

(1) *Whole layer*: all concepts with IoU scores larger than 0 are regarded as valid concepts;

(2) *Highest IoU*: only select the concept with the highest IoU score, treating the neuron as a concept detector.

(3) *Threshold*: only utilize 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.

Core Concepts

Core concepts (CC) are the fundamental elements that are strong candidates for scene concepts introduced in (Kozorog and Stanojević 2013); e.g., *bed*, *cabinet*, *armchair*, *floor*, *wall*, *nightstand* and *lamp* are the CC of scene `bedroom`. A well-defined KG, such as ConceptNet, contains essential concepts in the scene and facilitates human understanding. Therefore, we leverage ConceptNet to help to define two types of CC to address the challenge of concept completeness, including scoping core concepts (SCC) and identifier core concepts (ICC). Informally speaking, the SCC for a

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

scene involves concepts that are shared between the scene-related concepts within the dataset and the scene-related concepts within the KG. On the other hand, the ICC for a scene involves concepts that are uniquely suited to a particular scene, such as *bed*, *nightstand* for `bedroom`. Between SCC and ICC, SCC takes into account concept coverage for the scene, while ICC considers the concepts specific to the scene. Next, two types of CC are defined as follows.

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

Definition. **(Identifier Core Concepts)** *Given a scene $y_j \in Y$ ($j \in \{1, ..., |Y|\}$), its associated images and concepts in whole dataset $\mathcal{D}$ are $\mathcal{D}_{y_j}$ and C_{y_j}, respectively.*

- *Concepts of a scene that obtained form dataset. $Count(y_j, p) \subseteq C_{y_j}$ is the set of overlapping ground truth concepts, from C_{y_j}, over at least p% of the images in $\mathcal{D}_{y_j}$,*
- *Specificity concepts for a scene obtained form dataset. P_c is the highest percentage such that, for any i, j, $i \neq j$, $Count(y_i, P_c) \neq Count(y_j, P_c)$.*
- *Concepts of a scene obtained form dataset and KG. $SCount(y_j, \mathcal{G}, p)$ 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 $\mathcal{D}_{y_j}$, where $TopkOfCount(y_j)$ is the set of the top k concepts of $Count(y_j, P_c)$ and $\mathcal{G}$ is Knowledge Graph,*
- *Specificity concepts of a scene obtained form dataset and KG. P_{sc} is the highest percentage such that, for any i, j, $i \neq j$, $SCount(y_i, \mathcal{G}, P_{sc}) \neq SCount(y_j, \mathcal{G}, P_{sc})$).*

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 concepts, no matter whether they are in the KG or not. As ICC is more specific, it often has a smaller size than SCC.

Model Prediction Explanations

Model Prediction Explanations aims to utilize CC to establish the link between decisions and concepts in image. For this purpose, we propose the following metrics accordingly.

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

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

$$SM(x_i, y_j) = \frac{|LC(x_i) \cap CC_l(y_j)|}{|LC(x_i) \cup CC_l(y_j)|} \tag{4}$$

$$DM(x_i, y_j) = \frac{|LC(x_i) \setminus CC_l(y_j)|}{|CC_l(y_j)|} \tag{5}$$

Note y_j is the predicted scene. The larger (smaller) the CM and SM (DM) scores become, the smaller the gap between the learned concepts and scene.

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 model, the first task here is to explain why the prediction is wrong, i.e., why $y_t \neq y_p$. One would expect that the LC 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., $DM(x_i, y_p) > DM(x_i, y_t)$). The consistency metric for set D_f of false predictions (as scene y_p) CM^{FP} 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 second 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 correctly predicted images CM^{TP} of D_t should be larger than that over the falsely predicted images $CM^{T\text{-}FP}$: $CM^{TP} > CM^{T\text{-}FP}$.[5] Thus, the consistency metric for the set D_t of correctly predicted images and that for the set D_f of falsely predicted images in scene y_t can be defined as follows :

$$CM^{TP} = \frac{\Sigma_{x_i \in D_t} CM(x_i, y_t)}{|D_t|} \tag{7}$$

$$CM^{T\text{-}FP} = \frac{\Sigma_{x_i \in D_f} CM(x_i, y_t)}{|D_f|} \tag{8}$$

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 $SM^{TP} > SM^{T\text{-}FP}$.

Neuron Interpretation via Concept Filtering

This section aims to optimize neuron interpretation by merging semantically-equivalent concepts based on ConceptNet. In context of image classification and object detection, there could be a large number of concepts and many of which might have similar semantics, e.g. *armchair* and *chair*. This could lead to misleading or even wrong explanations for predictions. 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, Dismult and TransD, 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 existing neuron interpretation.

[5] The symbol T refers to calculating true prediction explanation.

LC	CC	False Prediction Explanation (%)			True Prediction Explanation (%)			
		CM^{FP}	DM^{FP}	SM^{FP}	CM^{TP}	$CM^{T\text{-}FP}$	SM^{TP}	$SM^{T\text{-}FP}$
ConceptSHAP	–	51.93	43.24	50.87	21.04	18.51	19.68	17.32
CLIP-Dissect	SCC	53.31	86.57	33.45	12.64	11.78	13.72	13.16
	ICC	46.94	67.06	38.94	33.31	32.44	21.76	21.17
Whole Layer	Top_10	43.04	19.13	42.77	11.29	6.12	4.38	2.49
	SCC	**78.51**	87.30	69.85	12.95	9.94	8.17	6.85
	ICC	51.43	69.32	29.58	**53.73**	**47.33**	22.76	22.13
Highest IoU	Top_10	34.61	19.12	34.56	9.20	5.18	7.51	4.03
	SCC	65.77	85.76	64.07	6.52	5.04	5.91	4.64
	ICC	49.42	67.76	42.24	26.32	21.86	21.27	18.11
Threshold	Top_10	42.52	18,69	42.42	11.17	6.07	7.48	4.12
	SCC	78.03	**87.38**	**72.13**	11.83	9.22	10.09	8.07
	ICC	50.32	69.81	34.11	49.60	44.13	**34.97**	**32.34**

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

Model Manipulation

Model Manipulation aims to study whether the CC could help to manipulate model behavior, including Neuron Identifying via CC and Re-training via CC. In addition, we propose using PE metrics for re-training.

Neuron Identification via CC aims to 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 y_j with true prediction, can be calculated as follows:

$$Con\_Score(f_t) = \sum_{x_i \in D^T_{y_j}} (P(x_i, CC_l) - N(x_i, CC_l)) \tag{9}$$

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

For true prediction, we disable top-k positive neurons (by setting the neuron features to 0 (Mu and Andreas 2020)) for the scene and see whether the model still correctly predicts the scene. For false prediction, we disable top-k (in our evaluation, $k = 20$) negative neurons for the scene and see whether the model can make better prediction.

Re-training via CC aims to integrate CC into original model design phase to further improve its performance. In the original models, the training objective is scene loss $\mathcal{L}_s$. We add another core concept loss:

$$\mathcal{L}_c = -\sum \log \mathcal{P}(c^* | \theta) \tag{10}$$

where $c^* \in C$ is the golden concept. For example, given scene `bedroom` with concepts *bed, armchair and fridge*, the new overall objective will let model pay more attention to the CC, such as *bed* and *armchair*.

Re-training via PE aims to utilize the explanation metrics as features to optimise the original model. We use a classical classifier SVM (Cesa-Bianchi, Gentile, and Zaniboni 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 model.

Experiments

Datasets

For testing, we use two scene datasets ADE20k (Zhou et al. 2017) and Opensurfaces (Bell, Bala, and Snavely 2014). 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 existing work CEN (Mu and Andreas 2020). Opensurfaces is a large database created from real-world consumer photographs with pixel-level annotations. It contains 25,329 images which are annotated with surface properties, including material, color and scene.

Do Completing Concepts Based on Core Concepts Provide Benefit to Model Prediction Explanation?

Yes. The overall results show that our method, integrating complete concepts, achieves better results than existing methods across false and true prediction. In details, we first report the results of false/true prediction explanation, and then enhance the model prediction explanation by integrating concept filtering. Furthermore, we conduct experiments on different models.

Results of False Prediction Explanation For false prediction explanation, we expect to have higher scores on the three metrics (CM, DM, SM). The higher the score, which means the better the explanations. The results are reported in Table 1, and we have the following three observations.

(1) When compared to the results of the baseline method Top_10, both SCC and ICC achieve significant better results, indicating our proposed method is effective and reasonable.

(2) All the best scores for false prediction explanation (across CM^{FP}, DM^{FP} and SM^{FP}) come from SCC. The reason is that SCC has broader coverage than ICC: if some concepts are not in SCC, then they are most likely to be incorrect. On the other hand, ICC is more specific, thus it might exclude some (partially) correct concepts.

(3) Among the three methods to represent neurons' learned concepts in section MinMax-based NetDissect, the

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 2: 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	15.79	9.88	6.23	3.59
SCC	25.31	19.69	18.23	15.01
ICC	**66.60**	**59.78**	**41.19**	**38.16**

Table 3: Integrating concept filtering for true prediction.

threshold-based method achieves better results, demonstrating that our method can explain false predictions well.

(4) The results from ConceptSHAP (Yeh et al. 2020) and CLIP-Dissect (Oikarinen and Weng 2023) are in line with the trends in NetDissect, and all satisfy our assumptions.

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 scores as well as 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, the results in Table 1, ICC achieves the better results. Although the gap between SM^{TP} and $SM^{T\text{-}FP}$ for top 10 (Highest IoU) is bigger, these SM^{TP} and $SM^{T\text{-}FP}$ scores are very low.

Integrating Concept Filtering for Model Prediction Explanation Tables 2 and 3 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 that without concept filtering in Table 1.

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 of false prediction explanation are shown in Table 4. From the CC perspective, SCC has better results than ICC over every model, which is similar to the observation over ResNet-18 from Table 1. The results of SCC on ResNet-50 achieve the best performance across all models.

Do External Knowledge and Concept Fusion Improve Existing Neuron Interpretation?

Yes. Our method, Concept Filtering, merges semantically-equivalent concepts based on ConceptNet, which effectively produces over 23% point gain on neuron behaviors for neuron interpretation. 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. 2015), ProjE (Shi and 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 ConceptNet. In particular, we extracted all concepts in ADE20k, as well as direct 1-hop and 2-hop neighbors of ADE20k concepts in ConceptNet. We applied fuzzy matching for 0.1% of ADE20k concepts due to some misalignment between concepts in ADE20k and ConceptNet.

Models	CC	CM^{FP}	DM^{FP}	SM^{FP}
ResNet-50	SCC	**80.85**	**89.36**	**74.46**
	ICC	53.19	69.15	36.17
DenseNet-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 4: Results of false prediction explanation on different models and utilize the MinMax-based threshold to learn the neurons' concepts.

Cluster Nb.	TransD	Dismult	ProjE	TransE
160	+17.01	+22.20	+19.94	+20.11
165	+18.11	+22.84	+17.48	+25.77
167	+18.44	+23.42	+18.03	+26.05
168	+21.78	+23.15	+17.80	**+26.31**
169	+20.72	+22.84	**+23.86**	+26.01
170	+20.90	+23.22	+22.35	+25.36
175	+21.27	**+23.88**	+21.74	+22.47
180	**+25.29**	+23.07	+23.04	+22.63
185	+24.10	+23.34	+22.49	+22.32

Table 5: IoU gain (%) of different clusters.

The IoU gain is measured by capturing the interpretability improvement from (A) concept with no clustering strategy to (B) concept with a k-clustering strategy with (k: Cluster Nb.) using Embeddings. The IoU gain is defined as $(B-A)/A$. Table 5 captures the main results. We can see that: (1) Fusing of semantically-equivalent concepts leads to significant performance enhancement across all methods. In essence, reducing cluster number exposes more interpretable units in the neural model. (2) Among various embedding techniques, TransE outperforms others, achieving a remarkable 26.3% improvement with 168 clusters compared to the non-clustering strategy, i.e., 512 neurons in the context of ResNet-18. (3) From a clustering perspective, optimal performance arises within the 168-180 class range. Fewer than

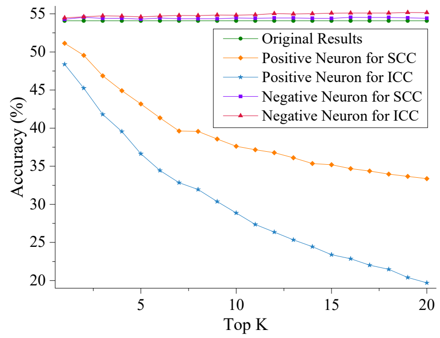


Figure 3: Results on ADE20k.

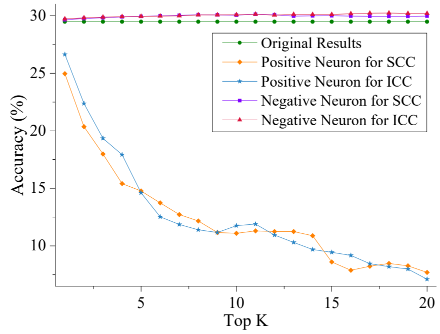


Figure 4: Results on Opensurface.

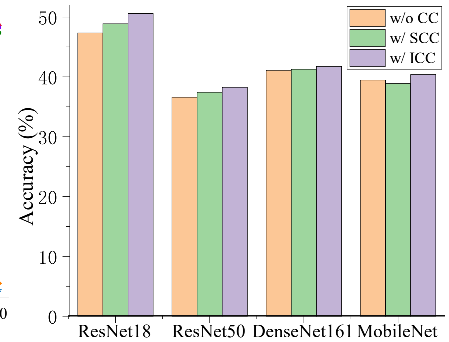


Figure 5: Model re-training via CC.

168 clusters imply fusion of less semantically related concepts, while over 180 clusters suggest the disregard of partially semantically-equivalent concepts.

Do Explanations Based on Core Concepts Contribute to Model Performance?

Yes. We show that core concepts and related explanation metrics can help optimise the model, leading to 26.7% of performance improvement. After verifying the effectiveness of CC on explanations, a natural question is whether CC can be further used to help manipulate model behavior.

Results of Neuron Identification via CC Figure 3 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 4, 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.

Results of Re-training via CC From Figures 3 and 4, 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.

Method	Accuracy (%) ADE20k	Accuracy (%) Opensurfaces
ResNet18	52.96	29.26
SVM (SCC)	66.54	31.62
SVM (ICC)	**67.11**	**32.27**

Table 6: Results of PE.

However, the 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 5. In Figure 3 and 4, the experiments are based on the model ResNet18, and the results have improved about 1.3% by removing the negative neurons. However, in Figure 5, the corresponding performance of ResNet18 has improved 3.27%. On the other models, the results by adding ICC are all improved. Compared to SCC, utilizing the ICC for re-training model is more effective.

The above two parts mainly focus on manipulating the model behavior, such as identifying the positive/negative neurons and re-training from scratch. In the following part, we further verify the effectiveness of explanation metrics CM, SM and DM; cf., Eq. (3), (4) and (5).

Results of Re-training via PE The results are shown in Table 6, 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.

Conclusion

In this study, we investigated knowledge-aware neuron interpretation for image scenes classification. To address the concept completeness, we proposed two types of core concepts (i.e., SCC and ICC) based on KGs. We show that SCC is effective on explaining false predictions, while ICC excels in neuron identification and model optimisation with concept loss. Our results also show that concept fusion and CC based metrics are effective for neuron interpretation and model optimisation, respectively, significantly outperforming state of the art approaches by over 20%.

Acknowledgments

We would like to thank the anonymous reviewers for their constructive comments and suggestions. This work is supported by a grant from the Institute for Tsinghua University Initiative Scientific Research Program, by the Science and Technology Cooperation and Exchange Special Program of Shanxi Province (No.202204041101016), by the Institute for Guo Qiang, Tsinghua University (2019GQB0003), by the Chang Jiang Scholars Program (J2019032), and the EPSRC project ConCur (EP/V050869/1).

References

Akhtar, N.; and Jalwana, M. A. A. K. 2023. Rethinking Interpretation: Input-Agnostic Saliency Mapping of Deep Visual Classifiers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 178–186. AAAI Press.

Bau, D.; Zhou, B.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Network Dissection: Quantifying Interpretability of Deep Visual Representations. In *IEEE Conference on Computer Vision and Pattern Recognition,CVPR*, 3319–3327. IEEE Computer Society.

Bell, S.; Bala, K.; and Snavely, N. 2014. Intrinsic images in the wild. *ACM Trans. Graph.*, 33(4): 159:1–159:12.

Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. *Curran Associates Inc.*, 2787–2795.

Brik, B.; Chergui, H.; Zanzi, L.; Devoti, F.; Ksentini, A.; Siddiqui, M. S.; Costa-Pérez, X.; and Verikoukis, C. V. 2023. A Survey on Explainable AI for 6G O-RAN: Architecture, Use Cases, Challenges and Research Directions. arXiv:2307.00319.

Cesa-Bianchi, N.; Gentile, C.; and Zaniboni, L. 2006. Hierarchical classification: combining Bayes with SVM. In Cohen, W. W.; and Moore, A. W., eds., *Machine Learning, Proceedings of the Twenty-Third International Conference, ICML*, volume 148, 177–184. ACM.

Chen, J.; Lecue, F.; Pan, J. Z.; Horrocks, I.; and Chen, H. 2018. Knowledge-based Transfer Learning Explanation. In *Proc. of the International Conference on Principles of Knowledge Representation and Reasoning (KR2018)*, 349–358.

Deng, S.; Zhang, N.; Zhang, W.; Chen, J.; Pan, J. Z.; and Chen, H. 2019. Knowledge-Driven Stock Trend Prediction and Explanation via Temporal Convolutional Network. In *Proc. of the World Wide Web Conference (WWW 2019)*, 678–685.

Ge, Y.; Xiao, Y.; Xu, Z.; Zheng, M.; Karanam, S.; Chen, T.; Itti, L.; and Wu, Z. 2021. A Peek Into the Reasoning of Neural Networks: Interpreting With Structural Visual Concepts. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2195–2204. Computer Vision Foundation / IEEE.

Ghorbani, A.; Wexler, J.; Zou, J. Y.; and Kim, B. 2019. Towards Automatic Concept-based Explanations. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS*, 9273–9282.

Guan, Y.; Chen, J.; Lecue, F.; Pan, J.; Li, J.; and Li, R. 2023. Trigger-Argument based Explanation for Event Detection. In *Findings of the Association for Computational Linguistics: ACL 2023*, 5046–5058. Toronto, Canada: Association for Computational Linguistics.

Ji, G.; He, S.; Xu, L.; Liu, K.; and Zhao, J. 2015. 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 of the Asian Federation of Natural Language Processing, ACL*, 687–696. The Association for Computer Linguistics.

Jiang, Z.; Zhang, Y.; Yang, Z.; Zhao, J.; and Liu, K. 2021. Alignment Rationale for Natural Language Inference. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP*, 5372–5387. Association for Computational Linguistics.

Kortylewski, A.; Liu, Q.; Wang, A.; Sun, Y.; and Yuille, A. L. 2021. Compositional Convolutional Neural Networks: A Robust and Interpretable Model for Object Recognition Under Occlusion. *Int. J. Comput. Vis.*, 129(3): 736–760.

Kozorog, M.; and Stanojević, D. 2013. Towards a definition of the concept of scene: Communicating on the basis of things that matter. *Sociologija*, 55(3): 353–374.

Montavon, G.; Lapuschkin, S.; Binder, A.; Samek, W.; and Müller, K. 2017. Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognit.*, 65: 211–222.

Mu, J.; and Andreas, J. 2020. Compositional Explanations of Neurons. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS*, 11.

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

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Pan, J.; Calvanese, D.; Eiter, T.; Horrocks, I.; Kifer, M.; Lin, F.; and Zhao, Y. 2017a. *Reasoning Web: Logical Foundation of Knowledge Graph Construction and Querying Answering*. Springer.

Pan, J. Z.; Vetere, G.; Gomez-Perez, J.; and Wu, H., eds. 2017b. *Exploiting Linked Data and Knowledge Graphs for Large Organisations*. Springer.

Shahroudnejad, A. 2021. A Survey on Understanding, Visualizations, and Explanation of Deep Neural Networks. arXiv:2102.01792.

Shi, B.; and Weninger, T. 2017. ProjE: Embedding Projection for Knowledge Graph Completion. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI*, 1236–1242. AAAI Press.

Teotia, D.; Lapedriza, À.; and Ostadabbas, S. 2022. Interpreting Face Inference Models Using Hierarchical Network Dissection. *Int. J. Comput. Vis.*, 130(5): 1277–1292.

Thoppilan, R.; Freitas, D. D.; Hall, J.; Shazeer, N.; Kulshreshtha, A.; Cheng, H.; Jin, A.; Bos, T.; Baker, L.; Du, Y.; Li, Y.; Lee, H.; Zheng, H. S.; Ghafouri, A.; Menegali, M.; Huang, Y.; Krikun, M.; Lepikhin, D.; Qin, J.; Chen, D.; Xu, Y.; Chen, Z.; Roberts, A.; Bosma, M.; Zhou, Y.; Chang, C.; Krivokon, I.; Rusch, W.; Pickett, M.; Meier-Hellstern, K. S.; Morris, M. R.; Doshi, T.; Santos, R. D.; Duke, T.; Soraker, J.; Zevenbergen, B.; Prabhakaran, V.; Diaz, M.; Hutchinson, B.; Olson, K.; Molina, A.; Hoffman-John, E.; Lee, J.; Aroyo, L.; Rajakumar, R.; Butryna, A.; Lamm, M.; Kuzmina, V.; Fenton, J.; Cohen, A.; Bernstein, R.; Kurzweil, R.; y Arcas, B. A.; Cui, C.; Croak, M.; Chi, E. H.; and Le, Q. 2022. LaMDA: Language Models for Dialog Applications. arXiv:2201.08239.

Tian, Y.; and Liu, G. 2020. MANE: Model-Agnostic Nonlinear Explanations for Deep Learning Model. In *2020 IEEE World Congress on Services, SERVICES 2020, Beijing, China, October 18-23, 2020*, 33–36. IEEE.

Wang, H.; Fang, Z.; Zhang, L.; Pan, J. Z.; and Ruan, T. 2015. Towards Effective Online Knowledge Graph Fusion. In *Proc. of 14th International Semantic Web Conference (ISWC 2015)*.

Yang, B.; tau Yih, W.; He, X.; Gao, J.; and Deng, L. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. arXiv:1412.6575.

Yeh, C.; Kim, B.; Arik, S. Ö.; Li, C.; Pfister, T.; and Ravikumar, P. 2020. On Completeness-aware Concept-Based Explanations in Deep Neural Networks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS*, 12. Red Hook, NY, USA: Curran Associates Inc.

Zhou, B.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; and Torralba, A. 2017. Scene Parsing through ADE20K Dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 5122–5130. IEEE Computer Society.

Zhu, M.; Gao, Z.; Pan, J. Z.; Zhao, Y.; Xu, Y.; and Quan, Z. 2015. TBox Learning from Incomplete Data by Inference in BelNet+. *Knowledge Based Systems*, 30–40.