On the Impact of Knowledge Distillation for Model Interpretability

Hyeongrok Han [1] **Siwon Kim** [1] **Hyun-Soo Choi** [2 3] **Sungroh Yoon** [1 4]

Abstract

Several recent studies have elucidated why knowledge distillation (KD) improves model performance. However, few have researched the other advantages of KD in addition to its improving model performance. In this study, we have attempted to show that KD enhances the interpretability as well as the accuracy of models. We measured the number of concept detectors identified in network dissection for a quantitative comparison of model interpretability. We attributed the improvement in interpretability to the class-similarity information transferred from the teacher to student models. First, we confirmed the transfer of class-similarity information from the teacher to student model via logit distillation. Then, we analyzed how class-similarity information affects model interpretability in terms of its presence or absence and degree of similarity information. We conducted various quantitative and qualitative experiments and examined the results on different datasets, different KD methods, and according to different measures of interpretability. Our research showed that KD models by large models could be used more reliably in various fields. The code is available at https://github.com/Rok07/KD_XAI.git.

1. Introduction

In knowledge distillation (KD), information is transferred from the teacher to student model, improving the performance of the student model (Hinton et al., 2015). In general, a student model is a small neural network with a lower learning capacity compared to that of the teacher model. Many attempts have been made to reduce the size of large models using KD (Liu et al., 2021; Wang et al., 2022; West et al., 2021). This is because the huge size of large pre-trained models such as CLIP and GPT results in increased resource consumption and inference costs, limiting their usage in downstream applications (Brown et al., 2020; Radford et al., 2021).

Several recent studies have elucidated why KD improves model performance (Yuan et al., 2020; Tang et al., 2020; Zhou et al., 2021). However, few studies have researched the other advantages of KD besides its improving model performance. Through this study, we demonstrated that KD could improve not only the generalization performance of models but also the interpretability, which indicates the reliability of models.

Researchers have attempted to understand the internal decision-making processes of neural networks, which essentially seem to be black boxes (Singla et al., 2019; Sundararajan et al., 2017; Ribeiro et al., 2016). For large models such as CLIP and GPT to be applied to various studies, it is necessary to secure explainability (Gerlings et al., 2021; van der Velden et al., 2022). Many studies consider the interpretability of a model high if the activation is object-centric (Fong & Vedaldi, 2017; Dabkowski & Gal, 2017; Zintgraf et al., 2017). In this study, we found that KD promoted the object-centricity of the activation map of student models and thereby enhanced their interpretability.

Figure 1 summarizes the main arguments of this study. First, to compare the interpretability of the models, we adopted *the number of concept detectors* introduced in network dissection (Bau et al., 2017) as a measure of interpretability. The number of concept detectors represents the degree of the object-centricity of activation maps and is directly proportional to the model interpretability. According to the defined terms of interpretability, we compared the interpretability of models trained from scratch (f_{scratch}) and trained using KD (f_{KD}), as shown in Figures 1 (a) and (b). Comparing the activation maps shown in Figures 1 (a) and (b), the activation map of f_{KD} is more object-centric than that of f_{scratch}.

We attributed this improvement in interpretability to the class-similarity information transferred from the teacher to

[1]Department of Electrical and Computer Engineering, Seoul National University, Seoul, Republic of Korea [2]Department of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul, Republic of Korea [3]ZIOVISION Inc., Chuncheon, Republic of Korea [4]Interdisciplinary Program in Artificial Intelligence, Seoul National University, Seoul, Republic of Korea. Correspondence to: Hyun-Soo Choi <choi.hyunsoo@seoultech.ac.kr>, Sungroh Yoon <sryoon@snu.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

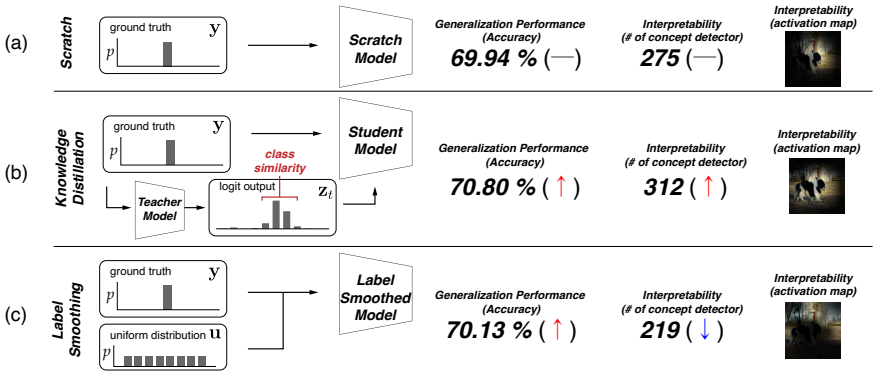

Figure 1. Illustration of main argument of the proposed study. The number of concept detectors of different models, namely models trained (a) from scratch (f_{scratch}), (b) using KD (f_{KD}), and (c) using LS (f_{LS}), have been measured for a quantitative comparison of the model interpretability. LS enhances the model performance but reduces the interpretability while KD boosts both. The transfer of class-similarity information from the teacher to student model enhances the model interpretability.

student models. The distribution of a teacher model had a high similarity between the semantically similar classes. For example, when the input image was a Border Collie, the student model was trained to minimize the distance from the distribution of the teacher model $\boldsymbol{z}_t$, which had a high probability of classes belonging to "dog." Thus, whenever "dog" samples were used as input, the student model could learn the typical characteristics of a "dog," which supported the object-centricity of the learned representation of the student model.

To demonstrate that class-similarity information enhances the interpretability of student models, we measured the entropy of semantically similar classes to confirm the transfer of class-similarity information from the teacher to student model via logit distillation. Then, we compared the interpretability between the model trained by label smoothing (f_{LS}), which did not contain (rather negatively affected) class-similarity information, and f_{KD}. As shown in Figures 1 (b) and (c), f_{LS} learns other features than objects, such as the background, which reduces the model interpretability. Referring to the previous example, for f_{LS}, the probability of an irrelevant class (*e.g.*, valley) increases because the model reduces the distance to a uniform distribution $\boldsymbol{u}$, causing the map of f_{LS} to become less object-centric. The results showed that KD enhanced the model performance and interpretability, whereas LS decreased the interpretability.

In addition to analyzing the effect of the presence or absence of class-similarity information on model interpretability, we also analyzed the effect of the *degree* of this information. Chandrasegaran et al. (2022) analyzed the effect of a teacher model trained by LS ($f_{\text{LS}}^{\text{teacher}}$) on the performance of the student model. They showed that the amount of class-similarity information transferred from the teacher to student model could be calibrated by adjusting the temperature of the KD using $f_{\text{LS}}^{\text{teacher}}$. Accordingly, we analyzed the effect of $f_{\text{LS}}^{\text{teacher}}$ on the interpretability of student models at different temperatures. Our results showed that 1) the interpretability of the student model improved with the increase in the class-similarity information learned by the students; and 2) adding logit distillation to feature distillation increased the model interpretability, demonstrating the effect of transferring class-similarity information.

We empirically showed the consistent effect of KD on model interpretability regardless of the KD method and how the interpretability measurement was defined. In addition to analyzing the effect of vanilla KD, we showed that the interpretability of models trained with various KD methods increased compared to that of f_{scratch}, as discussed in Section 3.3. Then, we generated a synthesized dataset with the ground truth of the heatmap and measured the five-band-scores proposed by Tjoa & Guan (2020) for f_{scratch} and f_{KD}, as described in Section 5.1. We measured the DiffROAR score (Shah et al., 2021) and loss gradient (Tsipras et al., 2018), as described in Sections 5.2 and 5.3, respectively, to assess the improvement in model interpretability according to various measures of interpretability. In Section 5.4, we also showed that KD improves model interpretability, not only in specific domains (vision), but also diverse domains (NLP).

The main contributions of our study are as follows.

- To the best of our knowledge, this is the first study to show that KD enhances the interpretability as well as the accuracy of models.
- A comparison of the interpretability of the LS model without similarity information and student models with different amounts of similarity information learned clearly showed that class-similarity information improved the interpretability of the student models.
- Various quantitative and qualitative experimental re-

sults support that KD improves model interpretability across KD methods, notions, datasets, and domains.

2. Related work

2.1. Knowledge distillation

Under KD, the performance of a student model with a low learning capacity is improved by receiving the output distribution from a large pre-trained teacher model with a high learning capacity (Hinton et al., 2015). Several studies have proposed various KD methods, including logit distillation (Hinton et al., 2015) and feature distillation (Romero et al., 2014; Zagoruyko & Komodakis, 2016; Yim et al., 2017; Kim et al., 2018; Xu et al., 2020; Tian et al., 2019), to improve the performance of student models.

Attention transfer (AT) averages the n-channel features in the intermediate layer of the teacher model without any regressor, allowing student models to learn attention (Zagoruyko & Komodakis, 2016). Factor transfer (FT) uses an auto-encoder to provide concise information that student models can easily understand (Kim et al., 2018). Self-supervision KD (SSKD) uses self-supervision signals to transfer intrinsic semantics and provide information to students (Xu et al., 2020). Contrastive representation distillation (CRD) trains students to maximize the lower bound of mutual information between the representations of two models, as done in contrastive learning (Tian et al., 2019).

Several studies (Yuan et al., 2020; Tang et al., 2020; Zhou et al., 2021) have analyzed how KD enhances the generalization performance of student models. They have confirmed that KD is an adaptive version of LS, which produces a regularization effect on models. Yuan et al. (2020) analyzed the relationship between KD and LS and proposed teacher-free KD. However, they did not explain the additional information provided to student models through KD. In the proposed study, we compared the models trained by LS and KD. The results revealed that the class-similarity information transferred by the teacher model promoted student models to capture conceptual representations more effectively.

2.2. Label smoothing

LS trains a model using a vector that combines a one-hot vector with a uniform distribution as a label (Szegedy et al., 2016). LS employs regularization during training, thereby improving the generalization performance of a model. Müller et al. (2019) demonstrated that LS renders each example in the training set equidistant from all other classes by visualizing the penultimate layer representations of the image classifiers. They demonstrated that $f_{\mathrm{LS}}^{\mathrm{teacher}}$ worsened the performance of the student model because it erased information about the similarities between teacher logits.

On the other hand, Shen et al. (2021) argued that KD and LS were compatible. They demonstrated that $f_{\mathrm{LS}}^{\mathrm{teacher}}$ improved the performance of the student model by increasing the distance between the embeddings of semantically similar classes. Chandrasegaran et al. (2022) introduced systematic diffusion and analyzed these contradictory findings. They demonstrated that systematic diffusion curtailed the performance of the student models trained by $f_{\mathrm{LS}}^{\mathrm{teacher}}$, thereby rendering KD at low temperatures effective. They also showed that the knowledge distilled from $f_{\mathrm{LS}}^{\mathrm{teacher}}$ resulted in a loss in class-similarity information with the decrease in temperature.

2.3. Explainable AI

Researchers have attempted to explain the reasoning processes of deep neural networks (DNNs). Post-hoc approaches interpret a trained model by localizing the attended input pixels (Simonyan et al., 2014; Sundararajan et al., 2017; Zeiler & Fergus, 2014; Ribeiro et al., 2016) or generating counterfactual explanations (Goyal et al., 2019; Singla et al., 2019). In contrast, several researchers (Chen et al., 2018; Alvarez-Melis & Jaakkola, 2018) have designed a new explainable architecture in which decision-making is inherently interpretable without any post-hoc explanation. Most approaches have demonstrated the interpretability of the proposed *methods* via qualitative visualizations or deteriorations in predictions after the elimination of the most important pixels.

The quantification of *model* interpretability is relatively underexplored. Li et al. (2020) theoretically defined interpretability as local linearity. Barceló et al. (2020) suggested that the computational complexity required to obtain explanations represents interpretability; the lower the complexity, the higher the interpretability. However, this cannot be empirically applied because real data distribution is considered, which results in high computational complexity. By contrast, network dissection (Bau et al., 2017) provides an intuitive and efficient method for quantifying the interpretability of DNNs. Therefore, we adopted network dissection as a measure of interpretability; the details are provided in the next section.

3. On the impact of KD for model interpretability

This section investigates the impact of KD on model interpretability. Section 3.1 describes the process of defining and quantifying model interpretability. Section 3.2 compares the interpretabilities of f_{scratch}, f_{KD}, and f_{LS}. Section 3.3 presents the comparison of the model trained using various KD methods to verify that enhancements are not limited to just those done by vanilla KD.

3.1. Interpretability quantification via network dissection

As described in the previous section, inspired by network dissection, we measured the interpretability of the models using the number of concept detectors. First, we shall describe the broadly and densely labeled (Broden) dataset and explain the process of counting the concept detectors. This dataset comprises the following datasets: ADE20k (Zhou et al., 2017), OpenSurfaces (Bell et al., 2014), PASCAL-Context (Mottaghi et al., 2014), PASCAL-Part (Chen et al., 2014), and Describable Textures (Cimpoi et al., 2014). The samples in Broden include objects, scenes, parts, textures, materials, and color concepts. Annotation masks for visual concepts are permitted in this dataset. All the pixels of a sample were annotated based on the corresponding concept. Therefore, by comparing the activation map for each unit in a neural network with the annotation mask, a unit aligned with human-interpretable concepts could be obtained (Bau et al., 2017).

The concept detectors were determined as follows. We measured the interpretability of model f with its frozen weights. One sample x of Broden was inputted to the model, and the activation map $A_i(x)$ was obtained for the i-th convolutional unit, following which the distribution a_i of this map was obtained. The threshold T_i corresponding to the top 0.5% of the activation value was calculated from a_i to satisfy $P(a_i \geq T_i) = 0.005$. Since the activation map $A_i(x)$ had a smaller resolution than the annotation mask M_c did for concept c, we interpolated $A_i(x)$ to ensure that the resolutions of M_c and $A_i(x)$ would be identical. Subsequently, we performed binary masking on the interpolated $A_i(x)$ such that only regions greater than or equal to T_i would appear. Finally, we calculated the intersection of union (IoU) scores between the masked activation map and annotation mask M_c. When the IoU score exceeded a predetermined threshold value (0.05), a unit i was recognized as the concept detector of the corresponding concept c. We have included a pseudocode for obtaining concept detectors in Appendix C to facilitate the understanding of network dissection.

A unique detector is a unit that is aligned with only a single concept. Network dissection (Bau et al., 2017) reports the number of unique detectors to measure the degree of disentanglement of intermediate representations. In this study, we measured the number of unique detectors and total number of concept detectors. We examined the overall concept detection capability of the unit and the degree of disentanglement.

3.2. Impact of KD on model interpretability

This section investigates the impact of KD on model interpretability, which was defined in the previous section. The experimental settings for training f_{scratch}, f_{KD}, and f_{LS} are presented in Appendix B.2. Figure 2 shows the interpretabilities of f_{scratch}, f_{KD}, and f_{LS}. We measured the number of concept detectors from the last convolutional layer of the model. A comparison of the interpretabilities for the lower layers of the models is provided in Appendix A.2. When KD was implemented, the total number of concept detectors, especially object detectors, increased significantly. Meanwhile, compared with those in f_{scratch}, both the number of concept detectors and unique detectors decreased in f_{LS}; although the number of object detectors decreased significantly, the number of scene detectors increased. The results are discussed in Section 4.2.

Table 1 lists the accuracy and interpretability of f_{scratch}, f_{LS}, and f_{KD}. We verified that KD could improve both the accuracy and interpretability of the models. To ensure the reliability of our results, we compared the interpretability of various architectures other than ResNet-18. In addition, we varied the quantile (top-k%) and IoU threshold, which are essential hyper-parameters for obtaining the concept detector. The results showed that KD enhanced model interpretability regardless of the architecture and hyper-parameters (these results are presented in Appendices A.1 and A.3).

3.3. Verification of various KD methods

In the previous section, we compared the interpretability of f_{KD} trained using vanilla KD; f_{scratch}; and f_{LS}. As discussed in Section 2, various KD methods other than vanilla KD have been proposed. We verified whether model interpretability improved, even if the teacher model transferred knowledge other than $\boldsymbol{z}_t$ to the student model. We measured the interpretability of the models trained using various methods, such as AT (Zagoruyko & Komodakis, 2016), FT (Kim et al., 2018), CRD (Tian et al., 2019), SSKD (Xu et al., 2020), and self-KD (Furlanello et al., 2018) (Table 1).

The accuracy of the model and total number of concept detectors increased regardless of the KD method used. This implied that when the teacher model transferred knowledge to the student model, its ability to capture the conceptual features of the student model could improve. Although the number of unique detectors in the AT models decreased compared to that of f_{scratch}, the number of object detectors increased. In particular, the self-KD model had a significant increase in interpretability; this can be explained as follows. In general, teacher models have architectures with a higher learning capacity than that of student models. Under self-KD, the architectures of both the teacher and student models were the same. Cho & Hariharan (2019) argued that if there was a gap between the learning capacities of the teacher and student models, the student model might not effectively understand the content. Similarly, the interpretability of the student model that received class-similarity information

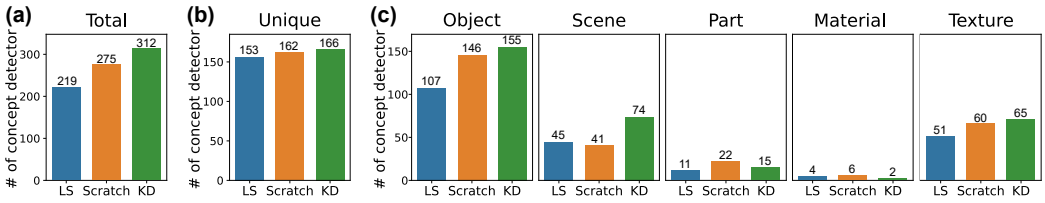

Figure 2. Comparison of interpretabilities among f_{scratch}, f_{KD}, and f_{LS}: (a) Number of concept detectors, (b) unique detectors, and (c) concept detectors per concept; the number of concepts and unique detectors increases for f_{KD} and decreases for f_{LS} (particularly, object detector).

from the ResNet-18 model, which had a similar learning capacity, improved significantly.

4. Impact of class-similarity information on model interpretability

In this section, we analyze the impact of transferred class-similarity information from the teacher to student model. First, in Section 4.1, we discuss that class-similarity information is transferred from the teacher to student model via logit distillation. Then, we contrasted the interpretability between f_{LS}, which does not contain class-similarity information, and f_{KD}, in Section 4.2. We visualized the activation maps of f_{scratch}, f_{KD}, and f_{LS}, empirically confirming that the activation map became object-centric as class similarity was transferred to the models. The student interpretability increased as the class-similarity information learned by the student model increased through class-similarity calibration; this is discussed in Section 4.3.

4.1. Examination of provision of class-similarity information by teacher model

To assess whether f_{KD} contained more class-similarity information than the other models did, we compared the entropy of the entire class and those within the same category. ImageNet dataset is a hierarchical dataset that comprises 1,000 classes. First, we divided these classes into 67 categories according to the coarse classification scheme proposed by Eshed (2020). Among the 67 categories, we excluded the "other" category because we could not state that similar classes had been grouped in this category. We measured the entropy of classes within the same category, which represents the amount of information contained in the model for that category, from the output distribution of f_{KD}. A larger entropy implied that the model contained more class-similarity information. The detailed experimental setting for the entropy measurement is presented in Appendix B.3.

Table 2 lists the results of the entropy measurements. Entropy (entire) was measured based on the output of all 1,000 classes, and entropy (category) was measured based on the output of the classes in the category to which the true class belongs. For all the classes, the entropy of f_{LS} was extremely high because the model was trained with a uniform distribution. By contrast, f_{LS} had the lowest entropy within the same category; f_{KD} had the highest entropy. The results showed that f_{KD} contained more class-similarity information than the other models did.

Table 1. Comparison of interpretabilities of various models on layer 4; Acc represents the Top-1 test accuracy on the ImageNet dataset. Each model was trained thrice based on different initial points to avoid variations. The accuracy and interpretability based on the average value of the three models are shown.

Model	Object	Scene	Part	Material	Texture	Color	Unique	**Total**	Acc
Scratch	146	41	22	6	60	0	162	275	69.94
LS	107	45	11	4	51	0	153	219	70.13
KD (Vanilla)	155	74	15	2	65	1	**166**	**312**	**70.80**
AT	163	35	25	10	54	0	158	287	70.52
FT	162	32	33	11	65	1	172	304	71.40
CRD	156	34	26	9	57	1	156	283	70.68
SSKD	162	63	15	3	66	1	164	310	70.09
Self-KD	173	46	22	6	69	0	169	316	70.63

4.2. Impact of presence of class-similarity information on model interpretability

We compared KD and LS to analyze how class-similarity information affected the degree of object-centricity of an activation map. First, we compared the loss functions of KD and LS to mathematically analyze the origin of the difference in class-similarity information. Equations (1) and (2) represent the loss functions of KD ($\mathcal{L}_{KD}$) and LS ($\mathcal{L}_{LS}$), respectively:

$$\mathcal{L}_{KD} = (1-\alpha)\cdot\mathcal{L}_{CE}(\sigma(\boldsymbol{z}_s), \boldsymbol{y}) + \alpha T^2 \cdot \mathcal{L}_{CE}(\sigma(\boldsymbol{z}_s^T), \sigma(\boldsymbol{z}_t^T)), \quad (1)$$

$$\mathcal{L}_{LS} = (1-\alpha)\cdot\mathcal{L}_{CE}(\sigma(\boldsymbol{z}), \boldsymbol{y}) + \alpha \cdot \mathcal{L}_{CE}(\sigma(\boldsymbol{z}), \boldsymbol{u}), \quad (2)$$

Table 2. Comparison of the entropy measured based on the output of all classes (Entire) and output of classes in the same category to which the correct class belongs (Category) for f_{scratch}, f_{KD}, and f_{LS}. Averaged entropy for 1,000 test samples of ImageNet was measured. Averages were only obtained for samples for which all models were correct.

Model	Entire	Category
Scratch	0.4851	2.8986
KD	0.5412	**2.9041**
LS	**3.3537**	2.6040

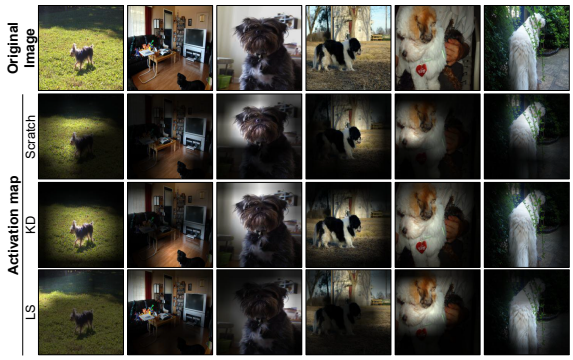

Figure 3. Comparison among activation maps of dog detector under f_{scratch}, f_{KD}, and f_{LS}; the brighter the pixel, the higher the activation value. The dog detector is the concept detector with the highest IoU score on the sample belonging to the "dog" concept.

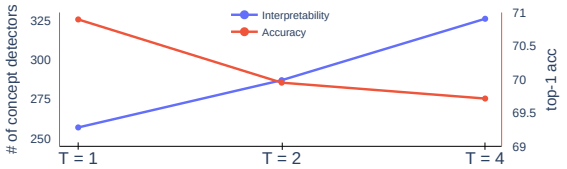

Figure 4. Comparison of interpretability and accuracy of student model trained from $f_{\text{LS}}^{\text{teacher}}$; $f_{\text{LS}}^{\text{teacher}}$ was trained using LS with $\alpha = 0.1$. The accuracy (red line) of students declines as T increases. On the other hand, the interpretability (blue line) of the student models increases as T increases. This result shows that the interpretability of the model improves as it learns class-similarity information better.

where $\mathcal{L}_{CE}$ denotes the cross-entropy loss; σ denotes the softmax function; $\boldsymbol{z}_s$ and $\boldsymbol{z}_t$ represent the output logits (distributions) of the student and teacher models, respectively; $\boldsymbol{z}$ is the logits of the target model for LS; $\boldsymbol{y}$ denotes the one-hot encoded ground truth vector; and $\boldsymbol{u}$ denotes the uniform distribution; α and T are the hyper-parameters, where α is the mixture parameter and T is the temperature for adjusting the smoothness of the distributions. We used the upper index, T, for the distributions smoothed by temperature T ($\boldsymbol{z}_s^T$ and $\boldsymbol{z}_t^T$). When $T = 1$ and the student model was considered the target model, the two loss functions differed only for the distribution $\boldsymbol{z}_t$ of the teacher model and the uniform distribution $\boldsymbol{u}$. Unlike $\boldsymbol{u}$, $\boldsymbol{z}_t$ contained information regarding the similarity between classes. Therefore, the difference in interpretability could be potentially attributed to the transfer of class-similarity information to the target model.

We visualized the activation maps generated by the concept detectors; Figure 3 shows the activation maps of f_{scratch}, f_{KD}, and f_{LS} overlaid on the sample. We observed that the activation maps of f_{KD} are more object-centric and activate in the entire object than those of f_{scratch} and f_{LS}. We shall explain the improvement in the model interpretability owing to the transferred class-similarity information. As an input, let us consider the image of a `Border Collie` belonging to the "dog" category. The logits of classes belonging to the "dog" category, such as the `German Shepherd` and `Komondor`, are higher than those of other classes that do not belong to the same category in $\boldsymbol{z}_t$. Even if the image of a `German Shepherd` was input, the teacher model transferred a similar distribution, which had a high logit of classes in the "dog" category, to the student model. These similar distributions enabled the student models to learn the typical characteristics of dogs from various images of them. The activation map of the student model could be more object-centric (*e.g.*, torso, ear, tail), as shown in Figure 3. Object-centric representation increases the IoU score corresponding to the correct concept. Meanwhile, f_{LS} was trained to learn all the other classes (*e.g.*, `seatbelt` and `valley`), even when the image of a `Border Collie` was the input. The activation map of f_{LS} was activated in regions independent of the image of the dog. This resulted in a lower IoU score for the correct concept. It can be inferred that KD improved the model interpretability by transferring class-similarity information, whereas LS reduced it.

Next, we shall interpret the experimental results in Table 1 and Figure 2. The total number of concept detectors of f_{LS} decreased, but the number of scene detectors increased compared with that of f_{scratch}. Figure 3 shows that the activation map of f_{LS} is scattered compared with that of the others. This implied that the activation map had a higher activation value in the scene than the object-centric map did. The number of object detectors decreased because f_{LS} captured more locally in the object region, reducing the IoU score compared with that of the other models. For f_{KD}, the activation map captured a wider area of the object, including even a part of the background, increasing the number of scene detectors. The improved interpretability of f_{KD}, decreased interpretability of f_{LS}, and increased number of scene detectors of f_{LS} can be explained by comparing the visualizations of the activation map.

4.3. Impact of amount of class-similarity information on model interpretability

In this section, we shall discuss how the interpretability of the student model improved as the class-similarity information that the student models learned increased from calibrating information. First, we investigated the effect of $f_{\text{LS}}^{\text{teacher}}$ on the interpretability of student models with varying T. Chandrasegaran et al. (2022) showed that through KD from $f_{\text{LS}}^{\text{teacher}}$ with a higher T, the distances between embeddings belonging to similar and dissimilar classes had become relatively reduced and increased, respectively. This implied that the higher the T, the greater the amount of class-similarity information the student model learned from $f_{\text{LS}}^{\text{teacher}}$, which allowed us to calibrate the class-similarity

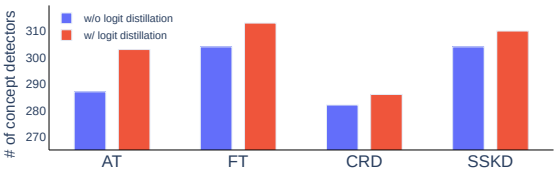

Figure 5. Comparison of interpretability between student models with (red bar) and without (blue bar) logit distillation added to feature distillation. For logit distillation, T and α are 4 and 0.5, respectively. We confirmed that adding logit distillation to feature distillation improved the interpretability for all KD methods.

information that the student model learned.

Figure 4 shows the interpretability and accuracy of the student models trained by $f_{\mathrm{LS}}^{\mathrm{teacher}}$. The accuracy of these models gradually decreased as T increased, aligning with the analysis reported by Chandrasegaran et al. (2022). On the other hand, the interpretability of these models gradually increased as T increased. The detailed settings of KD using $f_{\mathrm{LS}}^{\mathrm{teacher}}$ are presented in Appendix B.5.

Second, we examined the impact of adding logit distillation to feature distillation on model interpretability. As shown in Table 1, the interpretability of student models trained using various KD methods other than logit distillation (vanilla KD) also improved. This aligned with our insight that the information transferred to the student models improved their interpretability. Intuitively, $\boldsymbol{z}_t$ contained more class-similarity information than features. We compared the interpretability of student models that learned more class-similarity information by adding logit distillation to feature distillation with that of student models without this addition. Figure 5 shows that interpretability improved when feature distillation was combined with logit distillation. These results supported our argument that model interpretability improved when models learned more class-similarity information.

5. Improvements in different measures of model interpretability through KD

The improvement in measures of interpretability other than the number of concept detectors, namely five-band-scores, DiffROAR scores, and loss gradient, shall be discussed. Tjoa & Guan (2020) proposed five-band-scores to measure model interpretability by using a synthesized dataset with the ground truth of the heatmap. Shah et al. (2021) proposed the DiffROAR score, a metric used to probe whether an instance-specific explanation of a model highlighted its discriminative features. Tsipras et al. (2018) claimed that the degree of alignment between pixels more relevant to human perception and gradients indicated their degree of interpretability. We also show that KD improves model interpretability, not only in specific domains (vision), but also across diverse domains (NLP). In addition to the number of concept detectors, we demonstrated that consistent results could be obtained for various notions, datasets, and domains.

Table 3. Comparison of five-band-scores of f_{scratch} and f_{KD} on the synthesized dataset. Higher values of pixel accuracy, recall, and precision and lower values of FPR indicate that the model has a higher interpretability; f_{KD} has a higher interpretability than f_{scratch} does for all the metrics.

Model	Pixel_acc(↑)	Recall(↑)	Precision(↑)	FPR(↓)
Scratch	0.8803	0.5014	0.3011	0.1911
KD	**0.8974**	**0.5770**	**0.3545**	**0.1871**

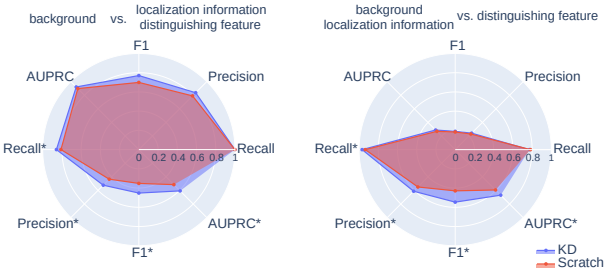

Figure 6. Results based on synthesized dataset. Left: class 0 vs. classes 1,2; Right: classes 0,1 vs. class 2; and * indicates the maximum value for the metric. In both scenarios, f_{KD} has a higher interpretability than f_{scratch} does for the averaged and maximum scores of AUROC, Precision, Recall and F1.

5.1. Five-band-scores

We measured the interpretability of the KD models using a dataset other than Broden. For an objective and quantitative evaluation, we used a synthesized dataset with the ground truth for the heatmap proposed by Tjoa & Guan (2020). The examples for each class and the ground truth generated are shown in Appendix D. There are three regions on the ground truth of the synthesized dataset: class 0) a background without any classification information (shown as a white region); class 1) localization information, which describes the location of an object (light pink region); and class 2), the distinguishing feature, which is crucial for distinguishing classes (dark pink area). We trained the model from scratch and KD using this synthesized dataset; details regarding the training are provided in Appendix B.6.
Since the distinguishing features could be regarded as the correct answer for interpretation, we could measure the interpretability by performing a pixel-by-pixel comparison of the saliency map and ground truth. However, obtaining the general recall and precision values was difficult because the ground truth was a ternary class rather than a binary class. Since the five-band-score reflects pixel accuracy, recall, precision, and the false positive rate (FPR) based on ternary

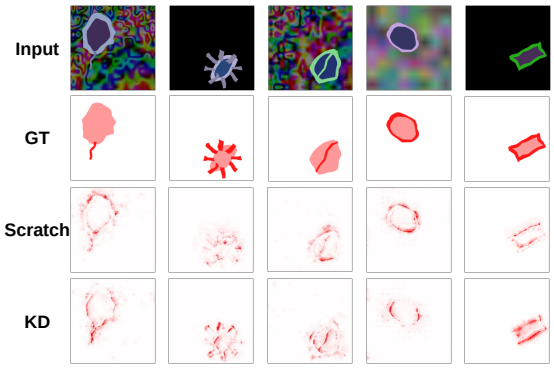

Figure 7. Qualitative results based on synthesized dataset; the first and second rows represent the input image and corresponding ground truth for each sample, respectively, while the third and last rows show the saliency maps of f_{scratch} and f_{KD}, respectively.

classification, we first measured the five-band-scores for the two models; Table 3 shows the results. In addition, we converted the ternary classification task to a binary classification task. Subsequently, we measured the performance by considering the localization information and distinguishing features as one class and the background and localization information as one class; the results are shown in Figure 6. The left radar plot shows the experimental results for class 0 vs. classes 1 and 2, while the one on the right shows the results for classes 0 and 1 vs. class 2. We measured the average precision, recall, AUPRC, and F1 scores for f_{scratch} and f_{KD}. When reporting the binary classification performance, we excluded samples belonging to a class with no objects (last column in Figure 11) from the evaluation. Both Table 3 and Figure 6 show that the interpretability of f_{KD} is higher than that of f_{scratch}.

For the qualitative evaluation, we visualized the heatmaps of f_{scratch} and f_{KD}, as shown in Figure 7. We discovered that the saliency map of f_{scratch} was activated more significantly in the region of the localization information than it had in the region of the distinguishing feature. In contrast, the saliency map of f_{KD} was activated more significantly for the distinguishing features. The hierarchical structure of the synthesized samples resulted in the synthesized dataset containing similarity information between the classes. The pre-trained teacher model provided class-similarity information to the student models, which improved their ability to capture the distinguishing features. Using a synthetic dataset, we showed that KD could improve the model interpretability within various datasets.

5.2. DiffROAR

DiffROAR is the difference in the predictive power of the datasets, with the top and bottom k% of the pixels removed by ordering the feature attribution of the model. A higher DiffROAR score implies that the attributes of the model are well aligned with the task-relevant features. We measure the DiffROAR scores of f_{scratch} and f_{KD} for the CIFAR-100, CIFAR-10, and MNIST test sets. The results are in Table 4. For all the datasets, f_{KD} had a higher DiffROAR score than f_{scratch} did.

Table 4. DiffROAR scores on various datasets (the higher, the better). Nine DiffROAR scores were measured for each dataset by varying the masking fractions (from 0.1 to 0.9 in increments of 0.1). Measurement was repeated with ten different initialization settings. Each value in the table is the average DiffROAR score of these 90 measurements.

Dataset	CIFAR-100	CIFAR-10	MNIST
Scratch	3.9747	3.3873	18.3628
KD	**4.2001**	**4.3850**	**22.3209**

5.3. Loss gradients

Figure 8 shows the loss gradients for the input pixels of f_{scratch} and f_{KD} for the ImageNet dataset; the gradients of f_{KD} are more aligned with the salient characteristic (i.e., the edge of the object) than those of f_{scratch}, showing that f_{KD} learned more human-perceptionally relevant features than f_{scratch} did. The results of the DiffROAR and loss gradient experiments revealed that KD enhanced model interpretability across various interpretability notions, in addition to the number of concept detectors.

5.4. NLP distillation

In this section, we conduct an experiment using BERT (Devlin et al., 2018) model for a text classification task, we demonstrated that KD enhances model interpretability in NLP tasks. To measure model interpretability in line with the convention in NLP, we used the Standard Sentiment Treebank (SST) dataset (Hase et al., 2021; Yin et al., 2021; Bastings et al., 2021). Below is the protocol we followed to evaluate the interpretability of an NLP model using the SST dataset.

- Data description
  - Input: sentences (movie review)
  - Class: 4-class (‘very negative’, ‘negative’, ‘positive’, and ‘very positive’)
- Model description
  - Teacher model: 12-layer BERT (acc: 0.623)
  - Scratch model (f_{scratch}): 3-layer BERT (acc: 0.484)
  - Student model (f_{KD}): 3-layer BERT (**acc: 0.516**)

It is noteworthy that the similarity between ‘very negative’ and ‘negative’ as well as between ‘very positive’ and ‘positive’ can be considered class-similarity information. The

Figure 8. Visualization of loss gradients of f_{scratch} and f_{KD} on the test set of ImageNet. Gradients of f_{KD} are more aligned with the semantically meaningful regions than those of f_{scratch} are, showing that f_{KD} learned more human-perceptually relevant features than f_{scratch}.

SST dataset provides a label for each word as either positive or negative, which serves as the ground truth for saliency (attribution), similar to the synthesized dataset in section 5.1. Therefore, we evaluated how well the saliency (attribution) obtained from f_{scratch} and f_{KD} aligns with the ground truth of attribution. We computed Integrated Gradients (IG) attribution scores from the validation and test samples.

To quantitatively measure model interpretability, we followed the process outlined below for samples where the model correctly predicts the answer: 1) For samples labeled as 'very positive/positive' in sentiment, we measured whether the words labeled as 'positive' have positive attribution scores and the words labeled as 'negative' have negative attribution scores. 2) For samples labeled as 'very negative/negative' in sentiment, we measured whether the words labeled as 'negative' have positive attribution scores and the words labeled as 'positive' have negative attribution scores. Table 5 shows the *interpretability* of f_{scratch} and f_{KD}. We show the average value of the models trained three times with different initial point. Through the above experiment, we demonstrated that KD can enhance model interpretability not only in image classification but also in the NLP domain. The detailed experimental settings of NLP distillation and the results for various model are presented in the Appendix.

Table 5. Comparison of model interpretability of f_{scratch} and f_{KD} on the SST dataset (the higher the better); f_{KD} has a higher interpretability than f_{scratch} does for NLP tasks.

Model	Accuracy	AUROC	AUPRC
f_{scratch}	0.677	0.689	0.810
f_{KD}	**0.722**	**0.720**	**0.831**

6. Conclusions

In this study, we demonstrated that KD could improve both the interpretability and accuracy of models. We measured the number of concept detectors of f_{scratch}, f_{KD}, and f_{LS} to quantitatively compare their interpretabilities. The results showed that the number of concept detectors had been significantly increased in f_{KD} and that the activation of f_{KD} was more object-centric than those of f_{scratch} and f_{LS} were. We attributed this improvement in interpretability to the class-similarity information transferred from the teacher to student model. Comparing the interpretability of models with and without class-similarity information showed that class-similarity information had improved the interpretability of the student model. Additionally, it was revealed that interpretability of the student models improved as the class-similarity information they learn increased with the calibration of information. Then, we measured the interpretability of the models trained using various KD methods and empirically showed their improved interpretability and performance. In addition to the Broden dataset, we measured the interpretability of the synthesized dataset using the ground truth label of the heatmap, the DiffROAR scores, and the loss gradients. The consistent results for varying measures of interpretability, KD methods, and datasets supported our argument that KD enhanced model interpretability.

Future scope of this paper With the emergence of foundation models such as CLIP and GPT, AI is increasingly becoming integrated into various aspects of human life. To democratize large foundation models at a reasonable cost and with fewer resources, it is essential that KD becomes prevalent in the future. Considering the concerns regarding the reliability of AI in human life, we believe that our discovery of KD improving model interpretability in crucial areas such as vision and NLP is envisioning. Furthermore, more interpretable models make debugging easier, so our findings are also important for engineers from a model development perspective. In particular, one possible application direction of our work is demonstrated by the fact that even simple techniques, such as Self-KD, can make the model more interpretable, facilitating easier debugging and improvement. We are envisioning further studies to determine whether KD can yield other useful results (*e.g.*, the robustness of the model) and whether additional metrics can be used to measure model interpretability.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korea government (Ministry of Science and ICT, MSIT) (2022R1A3B1077720 and 2022R1A5A708390811), Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (2021-0-02068, 2022-0-00959, and 2021-0-01343: AI Graduate School Program, SNU), Basic Science Research Program through NRF funded by the Korea government (Ministry of Education) (2022R1F1A1076454), and the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2023.

References

Alvarez-Melis, D. and Jaakkola, T. S. Towards robust interpretability with self-explaining neural networks. *arXiv preprint arXiv:1806.07538*, 2018.

Barceló, P., Monet, M., Pérez, J., and Subercaseaux, B. Model interpretability through the lens of computational complexity. *arXiv preprint arXiv:2010.12265*, 2020.

Bastings, J., Ebert, S., Zablotskaia, P., Sandholm, A., and Filippova, K. " will you find these shortcuts?" a protocol for evaluating the faithfulness of input salience methods for text classification. *arXiv preprint arXiv:2111.07367*, 2021.

Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

Bell, S., Bala, K., and Snavely, N. Intrinsic images in the wild. *ACM Transactions on Graphics (TOG)*, 33(4):1–12, 2014.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Chandrasegaran, K., Tran, N.-T., Zhao, Y., and Cheung, N.-M. Revisiting label smoothing and knowledge distillation compatibility: What was missing? In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2890–2916. PMLR, 17-23 Jul 2022.

Chen, C., Li, O., Tao, C., Barnett, A. J., Su, J., and Rudin, C. This looks like that: deep learning for interpretable image recognition. *arXiv preprint arXiv:1806.10574*, 2018.

Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., and Yuille, A. Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1971–1978, 2014.

Cho, J. H. and Hariharan, B. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4794–4802, 2019.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3606–3613, 2014.

Dabkowski, P. and Gal, Y. Real time image saliency for black box classifiers. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/0060ef47b12160b9198302ebdb144dcf-Paper.pdf.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Eshed, N. Novelty detection and analysis in convolutional neural networks. Master's thesis, Cornell University, 2020.

Fong, R. C. and Vedaldi, A. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., and Anandkumar, A. Born again neural networks. In *International Conference on Machine Learning*, pp. 1607–1616. PMLR, 2018.

Gerlings, J., Shollo, A., and Constantiou, I. Reviewing the need for explainable artificial intelligence (xai). In *Proceedings of the 54th Hawaii International Conference on System Sciences*, Proceedings of the Annual Hawaii International Conference on System Sciences, pp. 1284–1293, United States, 2021. Hawaii International Conference on System Sciences (HICSS). doi: 10.24251/HICSS.2021.156. 54th Annual Hawaii International Conference on System Sciences, HICSS 2021 ; Conference date: 05-01-2021 Through 08-01-2021.

Goyal, Y., Wu, Z., Ernst, J., Batra, D., Parikh, D., and Lee, S. Counterfactual visual explanations. In *International Conference on Machine Learning*, pp. 2376–2384. PMLR, 2019.

Hase, P., Xie, H., and Bansal, M. The out-of-distribution problem in explainability and search methods for feature importance explanations. *Advances in neural information processing systems*, 34:3650–3666, 2021.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Kim, J., Park, S., and Kwak, N. Paraphrasing complex network: Network compression via factor transfer. *arXiv preprint arXiv:1802.04977*, 2018.

Li, J., Nagarajan, V., Plumb, G., and Talwalkar, A. A learning theoretic perspective on local explainability. *arXiv preprint arXiv:2011.01205*, 2020.

Liu, P., Gao, Z.-F., Zhao, W. X., Xie, Z.-Y., Lu, Z.-Y., and Wen, J.-R. Enabling lightweight fine-tuning for pre-trained language model compression based on matrix product operators. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 5388–5398, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.418. URL https://aclanthology.org/2021.acl-long.418.

Mottaghi, R., Chen, X., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., Urtasun, R., and Yuille, A. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 891–898, 2014.

Müller, R., Kornblith, S., and Hinton, G. E. When does label smoothing help? In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/f1748d6b0fd9d439f71450117eba2725-Paper.pdf.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Ribeiro, M. T., Singh, S., and Guestrin, C. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

Shah, H., Jain, P., and Netrapalli, P. Do input gradients highlight discriminative features? *Advances in Neural Information Processing Systems*, 34, 2021.

Shen, Z., Liu, Z., Xu, D., Chen, Z., Cheng, K.-T., and Savvides, M. Is label smoothing truly incompatible with knowledge distillation: An empirical study. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PObuuGVrGaZ.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *In Workshop at International Conference on Learning Representations*. Citeseer, 2014.

Singla, S., Pollack, B., Chen, J., and Batmanghelich, K. Explanation by progressive exaggeration. *arXiv preprint arXiv:1911.00483*, 2019.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Tang, J., Shivanna, R., Zhao, Z., Lin, D., Singh, A., Chi, E. H., and Jain, S. Understanding and improving knowledge distillation. *arXiv preprint arXiv:2002.03532*, 2020.

Tian, Y., Krishnan, D., and Isola, P. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

Tjoa, E. and Guan, C. Quantifying explainability of saliency methods in deep neural networks. *arXiv preprint arXiv:2009.02899*, 2020.

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018.

van der Velden, B. H., Kuijf, H. J., Gilhuijs, K. G., and Viergever, M. A. Explainable artificial intelligence (xai) in deep learning-based medical image analysis. *Medical*

Image Analysis, 79:102470, 2022. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2022.102470. URL https://www.sciencedirect.com/science/article/pii/S1361841522001177.

Wang, Z., Codella, N., Chen, Y.-C., Zhou, L., Dai, X., Xiao, B., Yang, J., You, H., Chang, K.-W., Chang, S.-f., et al. Multimodal adaptive distillation for leveraging unimodal encoders for vision-language tasks. *arXiv preprint arXiv:2204.10496*, 2022.

West, P., Bhagavatula, C., Hessel, J., Hwang, J. D., Jiang, L., Bras, R. L., Lu, X., Welleck, S., and Choi, Y. Symbolic knowledge distillation: from general language models to commonsense models. *arXiv preprint arXiv:2110.07178*, 2021.

Xu, G., Liu, Z., Li, X., and Loy, C. C. Knowledge distillation meets self-supervision. In *European Conference on Computer Vision*, pp. 588–604. Springer, 2020.

Yim, J., Joo, D., Bae, J., and Kim, J. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4133–4141, 2017.

Yin, F., Shi, Z., Hsieh, C.-J., and Chang, K.-W. On the sensitivity and stability of model interpretations in nlp. *arXiv preprint arXiv:2104.08782*, 2021.

Yuan, L., Tay, F. E., Li, G., Wang, T., and Feng, J. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3903–3911, 2020.

Zagoruyko, S. and Komodakis, N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

Zhou, H., Song, L., Chen, J., Zhou, Y., Wang, G., Yuan, J., and Zhang, Q. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. *arXiv preprint arXiv:2102.00650*, 2021.

Zintgraf, L. M., Cohen, T. S., Adel, T., and Welling, M. Visualizing deep neural network decisions: Prediction difference analysis. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=BJ5UeU9xx.

This document is a supplement to our study titled **'On the Impact of Knowledge Distillation for Model Interpretability'**. We demonstrate that f_{KD} improves model interpretability compared to f_{scratch} across various settings, including different hyper-parameters, layers and architectures. To ensure reproducibility, we provide a detailed description of the experimental settings used in the main paper. Additionally, we offer pseudocode for obtaining concept detectors, which aids in further understanding network dissection. Through entropy measurement within two similar classes, we demonstrate that f_{KD} contains more class-similarity information than f_{LS}. In addition to ImageNet, we present the visualization of loss gradients for f_{scratch} and f_{KD} on the MNIST dataset.

A. On the impact of KD for model interpretability in various settings

We conducted a comparative analysis of model interpretability using a vanilla KD method across various settings. In Section A.1, we present the model interpretability results considering different hyper-parameters, namely α, T, top k% activation value and IoU threshold. Section A.2 provides a comparison of the interpretability between f_{scratch}, f_{KD} and f_{LS} specifically focusing on the lower layers. In Section A.3, we evaluate the model interpretability for various architectures other than ResNet-18. Our experimental results consistently align with the results presented in the main paper.

A.1. Model interpretability for various hyper-parameters

In accordance with Equation (1) presented in the main paper, the KD loss function ($\mathcal{L}_{KD}$) incorporates hyper-parameters α and temperature T. We provide the number of concept detectors for each concept, the number of unique detectors, the total number of concept detectors and the top-1 accuracy on the ImageNet for f_{KD}, with variations of α, in Table 6 ($\alpha = 0.1$), Table 7 ($\alpha = 0.5$) and Table 8 ($\alpha = 0.9$). Within each table, we demonstrate the model interpretability by varying the temperature by values of 1, 4, 8, and 16. Consistent with the main paper, the presented results are averages from three separate model training runs initiated from different starting point.

When α was set to 0.9, the test accuracy of the other models, except for the case where T was set to one, was lower than that of f_{scratch}. Teacher output probability of incorrect class compared to the correct class increases as the α value increases. Training the student model to minimize the distance of the teacher distribution, which was greatly smoothed by a high temperature, resulted in more instances where the student learned information from a class that was not the correct answer. Consequently, we can explain that when both α and T values were high, the test accuracy of f_{KD} could be lower than that of f_{scratch}.

We verified that the total number of concept detectors in f_{KD} increased compared to f_{scratch}, regardless of the combination of hyper-parameters. Additionally, we demonstrated that the number of object detectors in f_{KD} increased due to the class-similarity information provided by the teacher, enabling the student to learn more object-centric representations. These results support our claim that KD enhances model interpretability. Moreover, in most cases, f_{KD} exhibited a higher number of unique detectors compared to f_{scratch}, indicating better capture of disentangled representations. If the α value remained the same, we observed an improvement in model interpretability as the temperature increased in the majority of cases. This improvement can be attributed to the increased transfer of class-similarity information from the teacher to the student model as the temperature increased.

Table 9 lists the number of concept detectors per concept, the number of unique detectors, the total number of concept detectors, and test accuracy of the f_{LS} with different α values. We trained the models with two α values (0.1 and 0.5). Compared to f_{scratch}, both models exhibited a decrease in interpretability. Specifically, the number of object detectors significantly decreased, while the number of scene detectors increased compared to f_{scratch}. This can be attributed to f_{LS} being trained to minimize the distance with the uniform distribution, enabling it to learn all other classes in addition to the target class. Consequently, the activation map of f_{LS} showed more active in the scene surrounding the object rather than at the center of the object.

In the main paper, we conducted a comparison of model interpretability using quantile and IoU thresholds of 0.005 and 0.05. The quantile represents the top k% of the activation value used to obtain T_i (as described in Section 3.1 of the main paper). A unit is considered a concept detector if the IoU score between the masked activation map and annotation mask exceeds the IoU threshold. Both the quantile and IoU threshold are important hyper-parameters for obtaining the concept detectors. Figure 9 and 10 present the results of the interpretability measurements for various quantiles and IoU thresholds. We show that KD enhances model interpretability regardless of these two hyper-parameters.

Table 6. Interpretability and accuracy of f_{KD} with $\alpha = 0.1$

Temperature(T)	Object	Scene	Part	Material	Texture	Color	Unique	Total	Acc
1	154	40	23	7	59	0	166	285	70.61
4	146	47	21	7	70	0	163	292	70.62
8	147	46	18	4	70	0	158	285	70.64
16	156	48	19	7	65	0	164	295	70.64

Table 7. Interpretability of f_{KD} with $\alpha = 0.5$

Temperature(T)	Object	Scene	Part	Material	Texture	Color	Unique	Total	Acc
1	150	43	19	6	58	0	156	277	71.17
4	155	74	15	2	65	1	166	311	70.80
8	161	70	21	4	72	1	168	330	70.77
16	167	71	20	5	79	0	171	343	70.85

Table 8. Interpretability and accuracy of f_{KD} with $\alpha = 0.9$

Temperature(T)	Object	Scene	Part	Material	Texture	Color	Unique	Total	Acc
1	162	48	24	7	69	0	174	310	70.87
4	137	78	13	4	66	1	166	298	69.59
8	149	61	20	6	81	0	170	318	68.91
16	152	72	18	3	79	1	160	325	69.54

Table 9. Interpretability and accuracy of f_{LS} with various α

α	Object	Scene	Part	Material	Texture	Color	Unique	Total	Acc
0.1	107	45	11	4	51	0	150	219	70.01
0.5	109	81	4	4	56	0	153	255	68.01

Table 10. The comparison of model interpretability of f_{scratch}, f_{KD} and f_{LS} in the lower layers; we obtained consistent results with the comparison of model interpretability of the main paper.

	Unique			Total		
Model	Layer1	Layer2	Layer3	Layer1	Layer2	Layer3
Scratch	0	7	30	0	7	37
LS	1	8	29	1	9	35
KD	**1**	**10**	**39**	**1**	**10**	**52**

A.2. Model interpretability for various layers

Table 10 lists the number of unique detectors and the total number of concept detectors in lower layers of f_{scratch}, f_{KD} and f_{LS}. Each model was the same model presented in Table 1 of our main paper. The quantile and IoU threshold were also equal to 0.005 and 0.05. We demonstrate that f_{KD} had more interpretable units, even in the lower layers. In particular, the interpretability gap with f_{scratch} widened from layer 3.

A.3. Model interpretability for various architectures

In the main paper, we focused a single setup for comparing the model interpretability with ResNet-18. It is essential to show that KD enhances model interpretability for architectures other than ResNet-18. We additionally measured interpretability of four different architectures. Table 11 present the interpretability for various architectures. Regardless of the model architecture, KD enhances the interpretability of models. In the case of MobileNet_v2, model interpretability was further improved when the teacher was MobileNet_v2 than the teacher was ResNet-34. We obtain the consistent results with the measurement of the main paper that self-KD enhances the model interpretability better than vanilla KD.

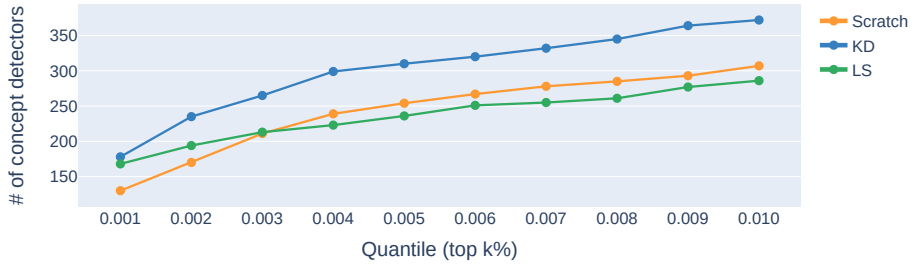

Figure 9. The total number of concept detectors of f_{scratch}, f_{KD} and f_{LS} for varying the quantiles (from 0.001 to 0.010 in increments of 0.001). The quantile indicates the top k% of activation value for obtaining a T_i.

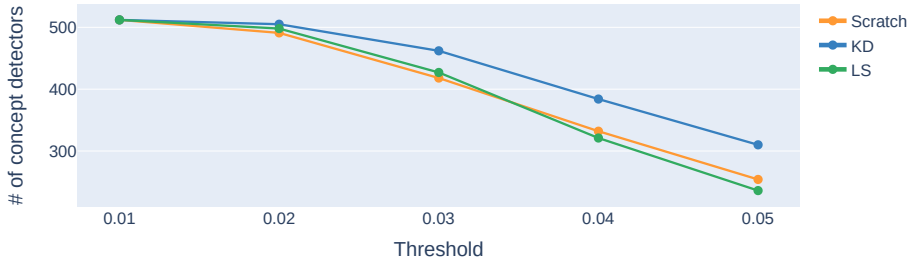

Figure 10. The total number of concept detectors of f_{scratch}, f_{KD} and f_{LS} for varying the IoU thresholds (from 0.01 to 0.05 in increments of 0.01). If IoU score between the masked activation map and annotation mask is higher than IoU threshold, we call that unit a concept detector.

Table 11. The comparison of model interpretability with various architectures; R, M, and E in the first and second rows represent ResNet (He et al., 2016), MobileNet (Howard et al., 2017), and EfficientNet (Tan & Le, 2019), respectively. The symbol "-" indicates that the model was trained from scratch, not KD.

Teacher	-	R-50	-	R-50	-	R-34	M_v2	-	E_b2
Student	R-34	R-34	R-50	R-50	M_v2	M_v2	M_v2	E_b2	E_b2
Unique	161	**185**	602	**685**	244	351	**393**	96	**469**
Total	267	**329**	1008	**1196**	348	655	**694**	129	**882**

B. Experimental details

B.1. Experimental environment

We conduct all experiments introduced in the main paper with the following environments.

1. **CPU**: Intel(R) Xeon(R) Gold 6258R
2. **GPU**: NVIDIA A40 48GB GDDR6
3. **CUDA version**: 11.4
4. **Library**: PyTorch (Paszke et al., 2019)

B.2. Experimental setup for training f_{scratch}, f_{KD}, and f_{LS}

The detailed experimental settings for training f_{scratch}, f_{KD}, and f_{LS} are as follows: All models were trained on the ImageNet dataset (Russakovsky et al., 2015), and the total number of epochs was 100. We performed various KD experiments using the TorchDistill library (Matsubara, 2021). For the teacher model, we used the pre-trained ResNet-34 architecture provided by Torchvision (Paszke et al., 2017). We used the ResNet-18 architecture for f_{KD} and the same architecture for f_{scratch} and f_{LS} for an unbiased comparison. We used SGD optimization as the optimizer. We set the initial

learning rate to 0.1 and divided it by ten every 30, 60, and 90 epoch for scheduling. We set the temperature to four. α, a hyper-parameter to determine the ratio of the correct answer to $\boldsymbol{z}_t$, was set to 0.5 for training. For the LS, we set the α value to 0.1. We report the results of the hyper-parameter setting with the highest generalization performance (top-1 test accuracy).

B.3. Experimental setup for the entropy measurement experiments

To show that class-similarity information is actually transferred from teacher to student model via logit distillation, we compared the entropy measured from entire and within same category of f_{scratch}, f_{KD}, and f_{LS} in Section 4.1. For the entropy measurement experiment, we used the pre-trained ResNet-34 architecture provided by Torchvision as a teacher model. We used the ResNet-18 architecture for f_{scratch}, f_{KD}, and f_{LS}. We used f_{scratch} as a pre-trained model provided by PyTorch. The hyper-parameters of f_{KD} were $\alpha = 0.5$, $T = 1$, and α of the f_{LS} was 0.5. The other setting (e.g., the total number of epochs, optimization, and learning rate) were the same as in the previous section.

B.4. Experimental setup for various KD methods

In the main paper, we compared the interpretability of the models trained with various KD methods (in Table 1). We describe the experimental setup of models trained with various KD. We conducted most of KD experiments using the TorchDistill library. For all experiments except self-KD, we used ResNet-34 architecture as teachers and ResNet-18 architecture as students.

B.4.1. Attention transfer (AT) (Zagoruyko & Komodakis, 2016)

- **training epochs**: 100
- **batch size**: 256
- **attention pair**: layer3, and layer4
- **attention loss factor**: 1,000

B.4.2. Factor transfer (FT) (Kim et al., 2018)

- **training epochs for paraphraser**: 1
- **number of input channels for paraphraser and translator**: 512
- **number of output channels for paraphraser and translator**: 256
- **training epochs for student**: 90
- **batch size**: 256
- **norm type**: 1
- **transferred layer**: layer4
- **factor transfer loss factor**: 1,000

B.4.3. Contrastive representation distillation (CRD) (Tian et al., 2019)

- **training epochs**: 100
- **batch size**: 85
- **number of negative samples**: 16384
- **feature dimension**: 128
- **temperature for contrastive learning**: 0.07
- **momentum**: 0.5
- **contrastive loss factor**: 0.8

B.4.4. SELF-SUPERVISION KNOWLEDGE DISTILLATION (SSKD) (XU ET AL., 2020)

- **training teacher SS module**
  - **training epochs**: 30
  - **batch size**: 85
  - **feature dimension**: 512
  - **optimizer**: SGD
  - **learning rate**: 0.1 divided by 10 at 10, 20 epoch
- **training student**
  - **training epochs**: 100
  - **feature dimension**: 512
  - **KD temperature**: 4.0
  - **SS temperature**: 0.5
  - **TF temperature**: 4.0
  - **SS ratio**: 0.75
  - **TF ratio**: 1.0
  - **loss weights [CE, KD, SS, TF]**: [1.0, 0.9, 10.0, 2.7]

B.4.5. SELF KNOWLEDGE DISTILLATION (SELF-KD) (FURLANELLO ET AL., 2018)

- **teacher architecture**: ResNet-18
- **training epochs**: 100
- **batch size**: 256
- **learning rate**: 0.1 divided by 0.1 at 30, 60, 90 epoch
- **momentum**: 0.9
- $\boldsymbol{\alpha}$: 0.1
- $\boldsymbol{T}$: 4.0

B.5. Experimental setup for KD using $f_{\text{LS}}^{\text{teacher}}$

In addition to the effect of the presence or absence of class-similarity information on model interpretability, we also analyzed how the degree of transferred class-similarity information affects the model interpretability in Section 4.3 of the main paper. This section presents the detailed experimental settings for KD using $f_{\text{LS}}^{\text{teacher}}$ shown in Figure 4. We used the ResNet-34 architecture for $f_{\text{LS}}^{\text{teacher}}$, and $f_{\text{LS}}^{\text{teacher}}$ is trained using LS with $\alpha = 0.1$. For the student model, we used the ResNet-18 architecture and we set the $\alpha = 0.5$ for KD. We varied the temperature to 1, 2, and 4 to analyze the impact of transferred class-similarity information on the model interpretability. We trained each model thrice based on different initial points to avoid variations. The other settings (e.g., the total number of epochs, optimization, and learning rate) were the same as in Section B.2.

B.6. Experimental setup for synthesized dataset experiments

In the main paper, we demonstrate KD enhances the model interpretability with various notions and dataset (in Section 5). We describe the setup of experiments on the synthesized dataset. For experiments on synthesized dataset, we used the ResNet-34 architecture without pre-training provided by Torchvision for a teacher model. We used the ResNet-18 architecture for a student model. We used the same architecture for the models trained from scratch. Since the number of classes for the synthesized dataset is 10, we added one fully connected (FC) layer to the ResNet backbone, and the output of FC layer is 10. Because the dataset is not complicated and overfitting easily occurs, as suggested by (Tjoa & Guan, 2020), we trained the models with small epochs (training epoch = 4), and the number of batch size was 4. We set the first learning rate to 0.001 with 0.00005 weight decay. Adam optimization was used as the optimizer. For KD training, α and T were set to 0.5 and 4, respectively. We used the Saliency function of the Captum library to get the saliency map of the models for evaluations (Kokhlikyan et al., 2020a).

B.7. Experimental setup for calculating DiffROAR

In the main paper, we measure the DiffROAR scores (in Section 5.2). DiffROAR is the difference in the predictive power of datasets, with the top-k% and bottom-k% of pixels removed by ordering the feature attribution of the model. We present the experimental setup for calculating DiffROAR scores. We used the Saliency function of the Captum library to obtain the feature attribution of the model. We used ResNet-18 as the teacher and student (Self-KD). For the CIFAR dataset, α and T were set to 0.1 and one. We re-trained the ResNet-18 model for top-k% and bottom-k% removed datasets with 60 epochs. For MNIST, we set α and T to 0.1 and four. We re-trained the ResNet-18 model for top-k% and bottom-k% removed datasets with 10 epochs. The differences in accuracy for top-k% and bottom-k% are presented in Table 4 of the main paper.

B.8. Experimental setup for NLP distillation

In the main paper, we demonstrate that KD enhances model interpretability in NLP tasks. We conducted an experiment using BERT for a classification task and utilized the Standard Sentiment Treebank (SST) dataset to measure model interpretability. The original SST dataset comprises five classes ('very negative', 'negative', 'neutral', 'positive', and 'very positive'). However, we train the model using only four classes ('very negative', 'negative', 'positive', and 'very positive') because unlike the negative and positive classes, the 'neutral' class does not contain similarity information with other classes. To perform distillation, we set the values of α and T to 0.5 and four, respectively. We trained the BERT-student model with three and six layers for 20 epochs.

C. Pseudocode of obtaining concept detectors

We present the pseudocode of obtaining the concept detector to facilitate the understanding of network dissection, and the code is shown in Algorithm 1.

Algorithm 1 Obtaining the concept detectors

Require: Broden dataset X, target model f, and target concept c

$N \leftarrow$ the number of convolutional units in fourth layer of f
for $x \in R^{n \times n}$ in X **do**
  for $i = 1, 2, ..., N$ **do**
    Collect the activation map $A_i(x) \in R^{d \times d}$, where $d < n$
  end for
end for
$a_i \leftarrow$ the distribution of individual unit activation
for $x \in R^{n \times n}$ in X **do**
  for $i = 1, 2, ..., N$ **do**
    Calculate T_i to satisfy $P(a_i \geq T_i) = 0.005$
    Interpolate $A_i(x)$ to be $\in R^{n \times n}$
    $A_i(x) \leftarrow A_i(x) \geq T_i$
    $M_c(x) \leftarrow$ annotation mask of x for concept c
    Compute $IoU_{i,c}$ value between $A_i(x)$ and $M_c(x)$
    if $IoU_{i,c} \geq 0.05$: **then**
      Unit i is the concept detector of the concept c
    end if
  end for
end for

D. The example samples of synthesized dataset

To verify that KD improves model interpretability except for the Broden dataset and the number of concept detectors, we present the result of five-band-scores and radar plots using the synthesized dataset in Section 5.1 of the main paper. This section presents the example samples of synthesized that we generated. The synthesized dataset has the ground truth for the heatmap and was proposed by Tjoa & Guan (2020). The synthesized dataset comprised 10 classes, and examples for each

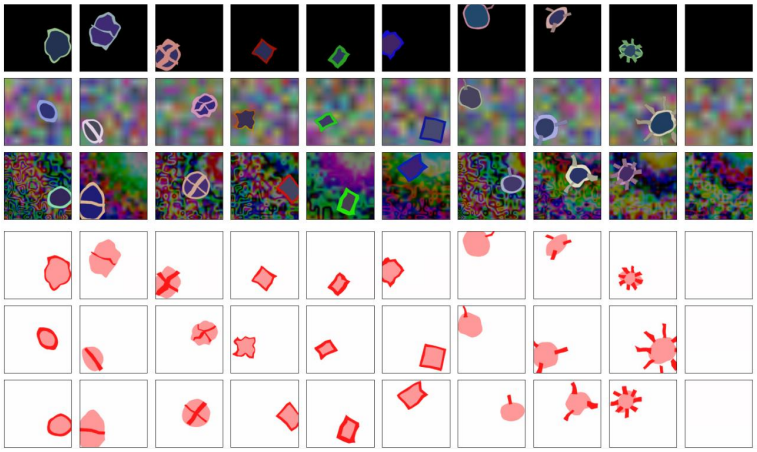

Figure 11. Samples and ground truths of the synthetic dataset that we generated. Row 1-3: Sample images of the 10 classes. Row 4-6: Ground truths of the sample images. The synthesized dataset formed a hierarchy with a circular (columns 1 to 3), rectangular (columns 4 to 6), and tail category (columns 7 to 9). Samples of the last column is a class with no objects. Each sample exhibited one among the three types (dark, blurred, and noisy) of random background.

class and the ground truth that we generated are shown in Figure 11. We generated 6,400 training and 1,600 test samples. The first three rows represent the ground truths of the sample images. The synthesized dataset formed a hierarchy with a circular (columns 1 to 3), rectangular (columns 4 to 6), and tail category (columns 7 to 9). Each sample exhibited one among the three types (dark, blurred, and noise) of random background. The ground truth of the synthesized dataset has three regions: class 0), a background that does not contain any classification information, shown as a white region; class 1), localization information, the location of an object, shown as a light pink region; and class 2), the distinguishing feature, which is crucial for distinguishing between classes, is shown as a dark pink area.

E. Additional experimental results

E.1. Qualitative results of entropy measurement experiments

In this section, we demonstrate the entropy measured in two classes with high similarity to show that KD contains class-similarity information well. Two classes with high similarity are `komondor` and `old English sheepdog` belonging to "sheepdog" category, and we present the example image in Figure 12. Both the `komondor` and the `old English sheepdog` are dogs with their faces covered in hair, with the difference that the former has a white fur, whereas the latter has a grayish fur on the back of the body. We obtained the output distribution of f_{scratch}, f_{KD} and f_{LS} when the correct answer class was `komondor` or `old English sheepdog` as the input sample. Table 12 lists the results of measuring entropy values using the output logit values of the two classes when all models had the correct answer. Even for two classes with high similarity, the entropy of f_{KD} was the largest, and the entropy of f_{LS} decreased significantly compared to f_{scratch}. Through qualitative entropy measurement experiments, we confirmed that the f_{KD} contained class-similarity information well, but not the f_{LS}.

Table 14 presents the number of classes for each category used in the entropy measurement experiment in the main paper.

Table 12. Comparison of entropy within two similar classes (`komondor` and `old English sheepdog`)

Model	Entropy
Scratch	0.944
Knowledge distillation	**0.953**
Label smoothing	0.872

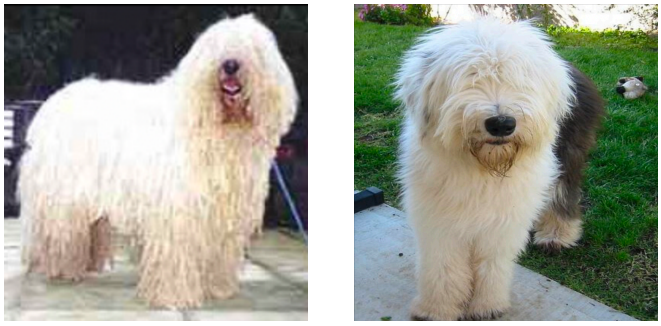

Figure 12. Example images of `komondor` (left) and `old English sheepdog` (right).

We divided 1,000 classes into 67 categories based on the coarse ImageNet category classification proposed by (Eshed, 2020). Among the 67 categories, we excluded the classes of "other" category because we could not state that similar classes were grouped together in that category.

E.2. Visualizations of loss gradient for MNIST dataset

We presented visualization of the loss gradients on the ImageNet in the Section 5.3 of the main paper. In addition to ImageNet, we present the visualization of the loss gradients of f_{scratch} and f_{KD} on the MNIST dataset, and the results are in Figure 13. The gradients of f_{KD} were more aligned with the semantic important regions (region of the numbers) than f_{scratch}. We demonstrate that f_{KD} learned more human-perceptually relevant features than f_{scratch} for various datasets.

E.3. BERT model interpretability for various layers

We demonstrated the model interpretability using the BERT model in Section 5.4. The SST dataset provides a label for each word as either positive or negative, serving as the ground truth for saliency (attribution), similar to the synthesized dataset in the main paper. We computed Integrated Gradients (IG) attribution scores from the validation and test samples using the LayerIntegratedAttribution function of the Captum library (Kokhlikyan et al., 2020b). Accuracy, AUROC, and AUPRC were listed as measures of model interpretability in Table 5. In Table 5, we presented only the experimental results of applying KD with a 12-layer BERT as the teacher model and a 3-layer BERT as the student model. In Table 13, we show the model interpretability when varying the layers of the student model. Our results show that KD enhances model interpretability, even when the layers of the student model are varied.

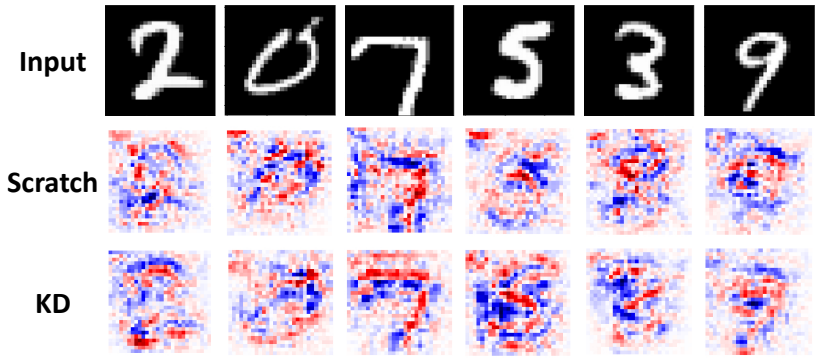

Figure 13. Visualization of loss gradients of f_{scratch} and f_{KD} on the testset of MNIST.

Table 13. Comparison of model interpretability of f_{scratch} and f_{KD} on the SST dataset with various layers of student model (the higher the better); the subscript refers to the layers of the student model

Model	Accuracy	AUROC	AUPRC
$f_{\text{scratch}}^{\text{3-layer}}$	0.677	0.689	0.810
$f_{\text{KD}}^{\text{3-layer}}$	**0.722**	**0.720**	**0.831**
$f_{\text{scratch}}^{\text{6-layer}}$	0.668	0.670	0.723
$f_{\text{KD}}^{\text{6-layer}}$	**0.722**	**0.793**	**0.829**

Table 14. The number of classes belonging to each category

Category	# of classes	Category	# of classes	Category	# of classes
arachnid	8	mollusk	6	building	37
armadillo	1	mongoose	3	clothing	47
bear	5	monotreme	2	container	18
bird	59	person	4	cooking	27
bug	25	plant	3	decor	22
butterfly	6	primate	19	electronics	49
cat	4	rabbit	3	fence	3
coral	5	rodent	7	food	28
crocodile	2	salamander	5	furniture	34
crustacean	9	shark	4	hat	8
dinosaur	1	sloth	2	instrument	28
dog	119	snake	16	lab equipment	2
echinoderms	3	trilobite	1	other	19
ferret	7	turtle	5	outdoor scene	32
fish	13	ungulate	16	paper	9
flower	5	vegetable	7	sports equipment	13
frog	3	wild cat	9	technology	27
fruit	14	wild dog	11	tool	43
fungus	7	accessory	18	toy	4
hog	3	aircraft	5	train	4
lizard	11	ball	9	vehicle	49
marine mammals	4	boat	15	weapon	10
marsupial	3				

Reference

Eshed, N. Novelty detection and analysis in convolutional neural networks. Master's thesis, Cornell University, 2020.

Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., and Anandkumar, A. Born again neural networks. In *International Conference on Machine Learning*, pp. 1607–1616. PMLR, 2018.

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

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Kim, J., Park, S., and Kwak, N. Paraphrasing complex network: Network compression via factor transfer. *arXiv preprint arXiv:1802.04977*, 2018.

Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020a.

Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020b.

Matsubara, Y. torchdistill: A modular, configuration-driven framework for knowledge distillation. In *International Workshop on Reproducible Research in Pattern Recognition*, pp. 24–44. Springer, 2021.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Tan, M. and Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pp. 6105–6114. PMLR, 2019.

Tian, Y., Krishnan, D., and Isola, P. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

Tjoa, E. and Guan, C. Quantifying explainability of saliency methods in deep neural networks. *arXiv preprint arXiv:2009.02899*, 2020.

Xu, G., Liu, Z., Li, X., and Loy, C. C. Knowledge distillation meets self-supervision. In *European Conference on Computer Vision*, pp. 588–604. Springer, 2020.

Zagoruyko, S. and Komodakis, N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.