Explaining why Lottery Ticket Hypothesis Works or Fails

Shantanu Ghosh
University of Pittsburgh
shantan2@andrew.cmu.edu, shg121@pitt.edu

Kayhan Batmanghelich
University of Pittsburgh
kayhan@pitt.edu

Abstract

Discovering a high performing sparse network within of a massive neural network can be advantageous for deploying them on devices with limited storage space, such as mobile phones. The Lottery Ticket Hypothesis (LTH) aims to find a sub-network within a deep network with similar or better performance than the original deep network. However not much research has been performed to examine the success / failure of LTH in terms of explainability. In this work, we examine why the performance of the pruned networks gradually increases or decreases. Specifically, we evaluate the pruned networks from LTH against a complex dataset, CUB-200. Also, we study the explainability of the pruned networks in terms of both pixels and high-level concepts using GRAD-CAM and TCAV to compute the local and global explanations respectively. We perform extensive experiments and observe that the performance of the pruned network consistently decreseas for CUB-200. Also, the explanations from the pruned networks are not consistent with the original network – a possible reason for the drop in performance. Our code is available publicly at `https://github.com/Shantanu48114860/Explainability-with-LTH`

1. Introduction

Neural network pruning [21, 14, 22] is a technique to get rid of irrelevant parameters to optimize storage requirements, reduce energy consumption, and perform efficient inference. The Lottery Ticket Hypothesis (LTH) [13] aims to find a subnetwork within a deep network by pruning the superfluous weights based on their magnitudes. However, the LTH paper evaluates their algorithm on relatively easier datasets like MNIST, CIFAR *etc*. Also, a rigorous study on whether LTH improves/degrades explainability needs to be included in the literature. In this study, first we aim to evaluate LTH algorithm on a complex dataset like CUB-200. Furthermore, we illustrate whether the pruned networks using LTH, rely on the relevant pixels / interpretable concepts for prediction. We accomplish this by quantifying local (explaining an individual sample) and global explanations (explaining a class) from the pruned networks using GRAD-CAM-based saliency maps and TCAV-based concept relevance scores, respectively.

The literature of explaining a network is quite extensive. The methods such as model attribution (*e.g*. Saliency Map [29, 27]), counterfactual approach [1, 30], and distillation methods [3, 10] are examples of post hoc explainability approaches. Those methods either identify important features of input that contribute the most to the network's output [28], generate perturbation to the input that flips the network's output [26], [23], or estimate simpler functions that locally approximate the network output. However, [2] demonstrates that the saliency maps highlight the correct regions in the image even though the backbone's representation was arbitrarily perturbed. In [17], the authors argue that the saliency maps do not correspond to the high-level interpretable *concepts*, understood by humans. They propose TCAV to quantify a concept's role in the model's final prediction. However, little emphasis has been given to evaluating the explanations from the networks obtained by network pruning. In [12], the authors discover the neuron-concept relationship by applying Net-dissection [7]. They should have investigated whether the LTH-pruned networks only rely on the important concept or pixel as the original model, thereby amplifying the performance.

In this project, we investigate the relationship between pruning and explainability. Initially, we prune the deep model using LTH. Next, we validate the following hypotheses on pruning and LTH:

Hypothesis A Pruning does not modify the global / local explanations, as the pruned networks prioritize the same relevant concepts / pixels for prediction as the original network. Consequently, LTH is able to find the *winning tickets*, *i.e*. the subnetworks with similar or better performance compared to the original network.

Hypothesis B Pruning modifies the global / local explanations as the pruned networks prioritize different relevant concepts / pixels for prediction as the original network.

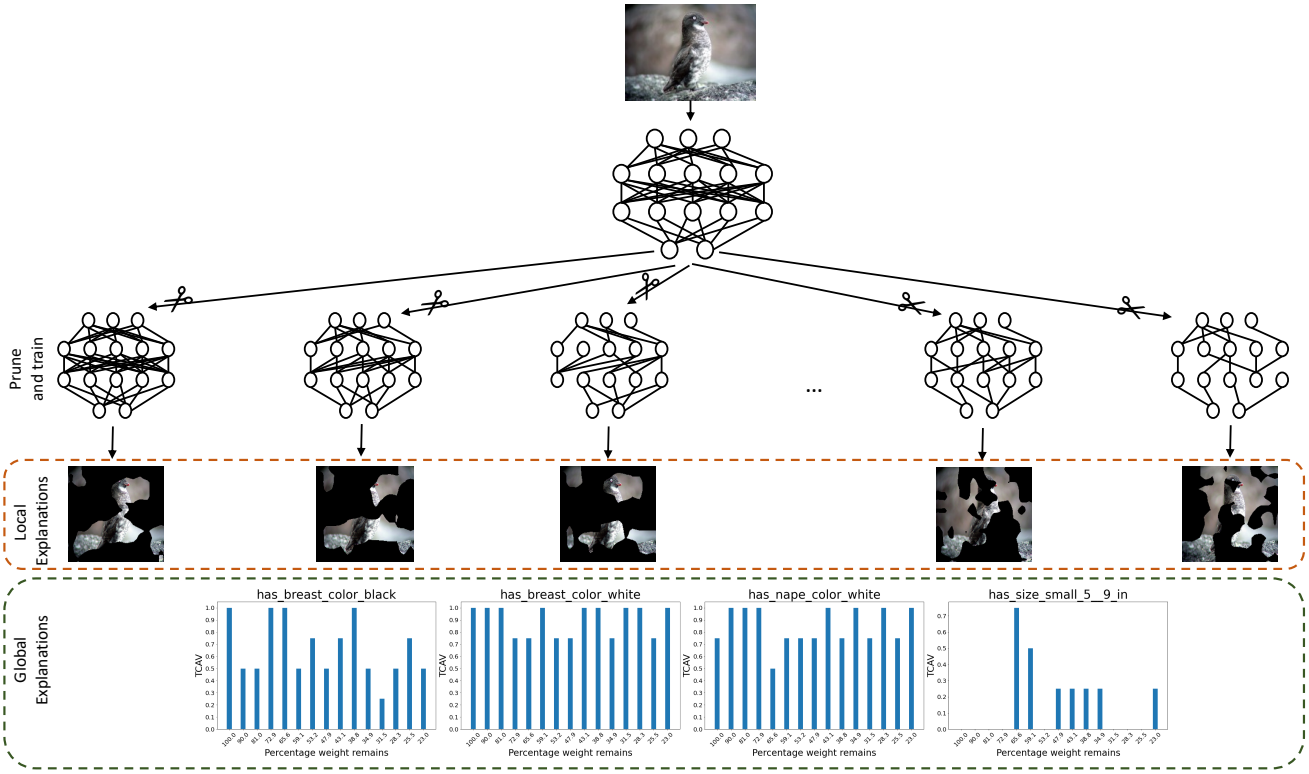

Figure 1. Overview of our method. (1) First we prune a deep neural network using *Lottery Ticket Hypothesis* [13]. (2) For each of the pruned subnetwork, we compute the local explanation using Grad-CAM [27]. (3) Furthermore, we compute the global explanation for different concepts using TCAV [17] score.

Consequently, LTH is unable to find the *winning tickets*, *i.e.* the subnetworks with similar or better performance compared to the original network.

To validate the two hypotheses, we borrow tools from explainable AI to quantify explanations for different pruned networks. Specifically, we compute the Grad-CAM [27] and TCAV score [17] to estimate the local / global explanations and identify the essential pixels / concepts, respectively. We perform extensive experiments across two datasets to study this phenomenon.

2. Related Work

Network pruning In practice, neural networks are frequently overparameterized. Distillation [5, 16] and pruning [21, 14] both rely on the ability to reduce parameters while maintaining accuracy. Even with adequate memory capacity for training data, networks naturally learn more specific functions [35, 24, 4] Later research shows that the overparameterized networks are easier to train [8, 16, 36]. Recently Lottery Ticket Hypothesis [13] aims to find a subnetwork within a deep neural net performs similarly or even better as the original deep network.

Saliency map based explanation methods First [29] developed a saliency map technique to highlight the relevant pixels in the image for the model's prediction. In CAM [36], we take the global average pooling of the feature map from the final convolution layer. Then we train a linear classifier to get the weights corresponding to each feature map, denoting the importance of a feature map to the final prediction. Later in [27, 9, 31, 32] amalgamate the gradient information with relevant weights of the necessary pixels to generate more localized saliency maps. In summary, the saliency maps aim to provide local explanations in terms of pixels. However, these saliency maps are often criticized for being inconsistent [25]. [2, 18] demonstrates that the saliency maps highlight the correct regions in the image even though the backbone's representation was arbitrarily perturbed. The explanations in terms of pixel intensities do not correspond to the high-level interpretable *concept*, understood by humans. In this project, we also aim to provide the post hoc explanation of the all the pruned models in terms of the interpretable concepts as a global explanation, rather than the pixel intensities.

Concept based explanation methods In *concept* based explanation methods, researchers aim to quantify the im-

portance of a humanly interpretable *concept* for the model's prediction. In TCAV [17], the researchers first learn the concept activation vectors (CAV) by learning a linear classifier that separates the concept images and the random images. After that, they estimate the TCAV score by taking the derivative of the prediction probabilities w.r.t. the concept activation vectors. In the concept completeness paper [34], they derived a metric known as the "concept completeness score", which shows whether the given concepts are sufficient to explain the prediction of the black box. In the Network, dissection method [7], the activations of each unit are segmented to find its association with a given concept by taking the intersection over the union metric. Recently, in [11], the researchers relax the assumption in [17] that concepts are linearly separable in the latent space; instead, they are represented by clusters, and the researchers obtain the concepts by utilizing the gaussian kernels for SVM based classifier.

In this project, we first train and prune a deep neural network using LTH [13]. Next, We utilize Grad-CAM [27] and TCAV [17] to estimate and compare the local and global explanations for all the pruned networks. We leave using the other saliency maps and concept-based methods for future work.

3. Methods

Notation Assume we have a dataset $\{\mathcal{X}, \mathcal{Y}, \mathcal{C}\}$, where $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{C}$ are the input images, class labels, and human interpretable concepts (stripness), respectively. We train a neural network $f : \mathcal{X} \rightarrow \mathcal{Y}$, trained only with supervision from the labels $\mathcal{Y}$. We denote the output logit for the k^{th} class label and image x as $f_k(x) = h_k(\Phi(x)$. We assume that f is a composition $h \circ \Phi$, where $\Phi : \mathcal{X} \rightarrow \mathbb{R}^l$ is the image embedding and $h : \mathbb{R}^l \rightarrow \mathcal{Y}$ is a transformation from the embedding, Φ, to the class labels $\mathcal{Y}$. We denote the binary classifier $t_C : \mathbb{R}^l \rightarrow \mathbb{R}$ aiming to estimate a unit linear CAV $v_C \in \mathbb{R}^l$ as for the concept C, orthogonal to the classification boundary.

Method overview Figure 1 shows an overview of our approach. We aim to train and prune the network f simultaneously for n rounds using the strategy of iterative magnitude pruning in *LTH*. Thus we obtain $f^1, f^2, \ldots, f^n$ subnetworks where the network f for the i^{th} round is denoted as $f^i(.) = h^i(\Phi^i(.))$. Next, we analyze the relationship between pruning and interpretability by estimating the saliency maps and conceptual sensitivity as the explanation metrics to measure local and global explanations, respectively. For the local explanation, we compute and compare the saliency maps $\{e^i(x)\}_{i=1}^n$ for all pruned networks $\{f^i(x)\}_{i=1}^n$, given a specific image x, yielding n-saliency maps $\{e^i(x)\}_{i=1}^n$. Furthermore, in the i^{th} round of pruning, we use the embedding Φ^i to obtain the CAV v_C^i from the classifier t_C^i for a specific concept C. We finally estimate the TCAV score using the CAV v_C^i in the particular concept C and class k, with $\text{TCAV}_{C,k}^i$ being the global explanation. For brevity, we drop *round index* i from the notations in the following sections.

3.1. Pruning Methodology

LTH [13] aims to find a subnetwork within a deep neural network that achieves similar accuracy as the original network – *when trained in isolation*. Specifically we perform the following steps to train and prune the smallest magnitude weights of the neural network f:

1. Randomly initialize $f(.; \theta_0)$ (where $\theta_0 \sim \mathcal{D}_\theta$).
2. Prune $p\%$ of the parameters θ_j with smallest magnitude weights, creating a mask m.
3. Optimize the network for j iterations using stochastic gradient descent (SGD) on a training set, arriving at parameters θ_j.
4. Reset the remaining parameters to their values in θ_0, creating a subnetwork $f(x; m \odot \theta_0)$.
5. Finetune and continue pruning for n rounds obtaining n-pruned subnetworks.

As pruning gradually reduces the network in size, we measure the performance and explainability for the all n-pruned subnetworks.

3.2. Global explanations using conceptual sensitivity

To estimate global explanations, we TCAV score [17] as a measure of conceptual sensitivity. Specifically, the TCAV score quantifies the role of a human interpretable concept (stripes) for the model prediction (zebra). For each concept C, we accumulate positive (P_C) and negative (N_C) sets of images in which the concept is present and absent, respectively. For example, corresponding to stripes, P_C and N_C contain images of stripped and random objects, respectively. Next, we train a binary linear classifier t_C to distinguish between image embeddings of two sets: $\{\Phi(x); x \in P_C\}$ and $\{\Phi(x); x \in N_C\}$. Consequently, we obtain the CAV $v_C \in \mathbb{R}^l$, unit vector normal to the decision boundary of t_C. Next, we estimate the conceptual sensitivity as a directional derivative $S_{C,k} = \nabla h_k(\Phi(x)).v_C$. Finally, we compute TCAV score as:

$$\text{TCAV}_{C,k} = \frac{|x \in X_k : S_{C,k}(x) > 0|}{|X_k|} \tag{1}$$

$\text{TCAV}_{C,k}$ is defined by the the fraction of k-class inputs whose image embedding was positively influenced by concept C, $\text{TCAV}_{C,k} \in [0, 1]$. Note that for a particular concept C and class label k, we compute $\text{TCAV}_{C,k}$ for all the n-pruned subnetworks.

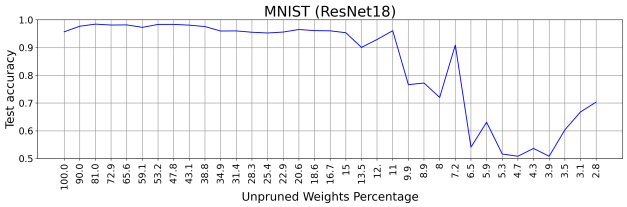

Figure 2. Accuracy of the pruned subnetworks for MNIST dataset classifying even/odd digits. The Y axis displays the accuracy of the test set. The X axis displays all the pruned networks identified by the percentage of the remaining weights in network (indicated by *Unpruned Weights Percentage*).

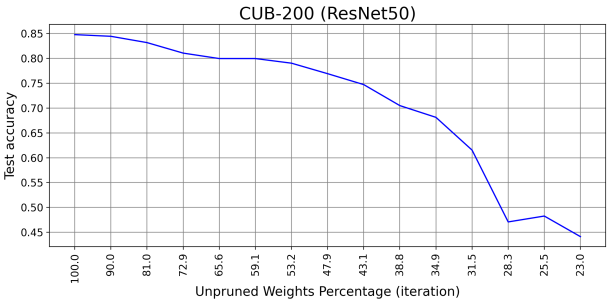

Figure 3. Accuracy of the pruned subnetworks for CUB-200 dataset classifying 200 bird species. For CUB-200, the accuracy of the model decreases consistently as we prune more weights.

3.3. Local explanations using saliency maps

Saliency maps are heatmap based technique, highlighting important features (pixels for images) in the input space responsible for the model's prediction as a class label k. In this project, we utilize GRAD-CAM method [27]. We calculate the heatmap by choosing an intermediate convolutional layer and then linearizing the rest of the network to be interpretable. Specifically, we estimate the derivative of the predicted output w.r.t. each channel of the convolutional layer averaged over all spatial locations as follows:

$$w_k^m = \sum_i \sum_j \frac{\partial Y^k}{\partial A_{i,j}^m}, \tag{2}$$

where w_c^k is the weight for k^{th} class and m^{th} feature map, Y^k is the prediction for k^{th} class and $A_{i,j}^m$ is the i, j^{th} location of m^{th} feature map. This results in a scalar for each channel that captures the importance of that channel in making the current prediction. Then, we calculate a weighted average of all activations of the convolutional layer with the above importance weights for each channel to get a 2D matrix over spatial locations Finally, we keep only positive numbers and resize them to the size of an input image to get the interpretation heatmap e. In this project corresponding to an image, we estimate this heatmap for all the n-pruned networks.

4. Experiments

Using a ResNet [15]-based model, we investigate the explainability metrics of the different pruned subnetworks using MNIST [20] and Caltech-UCSD CUB-200-2011 [33] datasets. The original LTH paper [13] discovers high-performing subnetworks for simpler data sets like MNIST. In this project, we aim to examine how LTH performs for a complicated dataset like CUB-200 and develop a relationship between the success/failure cases of LTH and explanations. First, we estimate the accuracy scores to perform a quantitative evaluation of the predictive performance of the different pruned networks. Second, we estimate the TCAV scores for different concepts for the pruned network to compare the global explanations. Third, we compute the Grad-CAM-based saliency maps for each pruned network to evaluate the qualitative comparison of the local explanations. The final two experiments evaluate if the pruned networks rely / ignore the relevant concepts / pixels for the final prediction as the original network. In all the subsequent plots, we denote the original network as the one with "100% weight remaining". We use the test-set images for all the qualitative and quantitative comparisons. For Grad-CAM, we follow an identical configuration in [27]. We use till the 3^{rd} ResNet block as Φ for each of the pruned networks to estimate the TCAV score. For h, we utilize the remaining ResNet blocks. Also, we flatten out the activation vector coming out of Φ to train t as [17]. We use a logistic regression-based classifier as t to estimate TCAV in equation 1.

4.1. Dataset and training configurations

MNIST The MNIST dataset includes 60000 training and test images of handwritten digits. We intend to solve a problem other than standard digit classification. Assuming Y $\subset \{0, 1\}^2$, we are interested in determining if a digit is either odd or even and explaining the assignment to one of these classes in terms of the digit labels (concepts in $\mathcal{C}$). We employ ResNet-18 to train and prune until 2.8% of the weights of the original model with the largest magnitude remain. Due to time constraints, we only analyze the TCAV scores for the pruned networks for MNIST, leaving the computation of Grad-CAM-based saliency maps as future work.

CUB-200 CUB-200 is a fine-grained classification dataset comprising 11788 images and 312 noisy visual concepts. The goal is to select the correct bird species from among 200 possible classes. To extract 108 denoised visual concepts, we use the strategy described in [19]. In addition, we use training and validation splits that were shared in reference [6]. We employ ResNet-50 to train

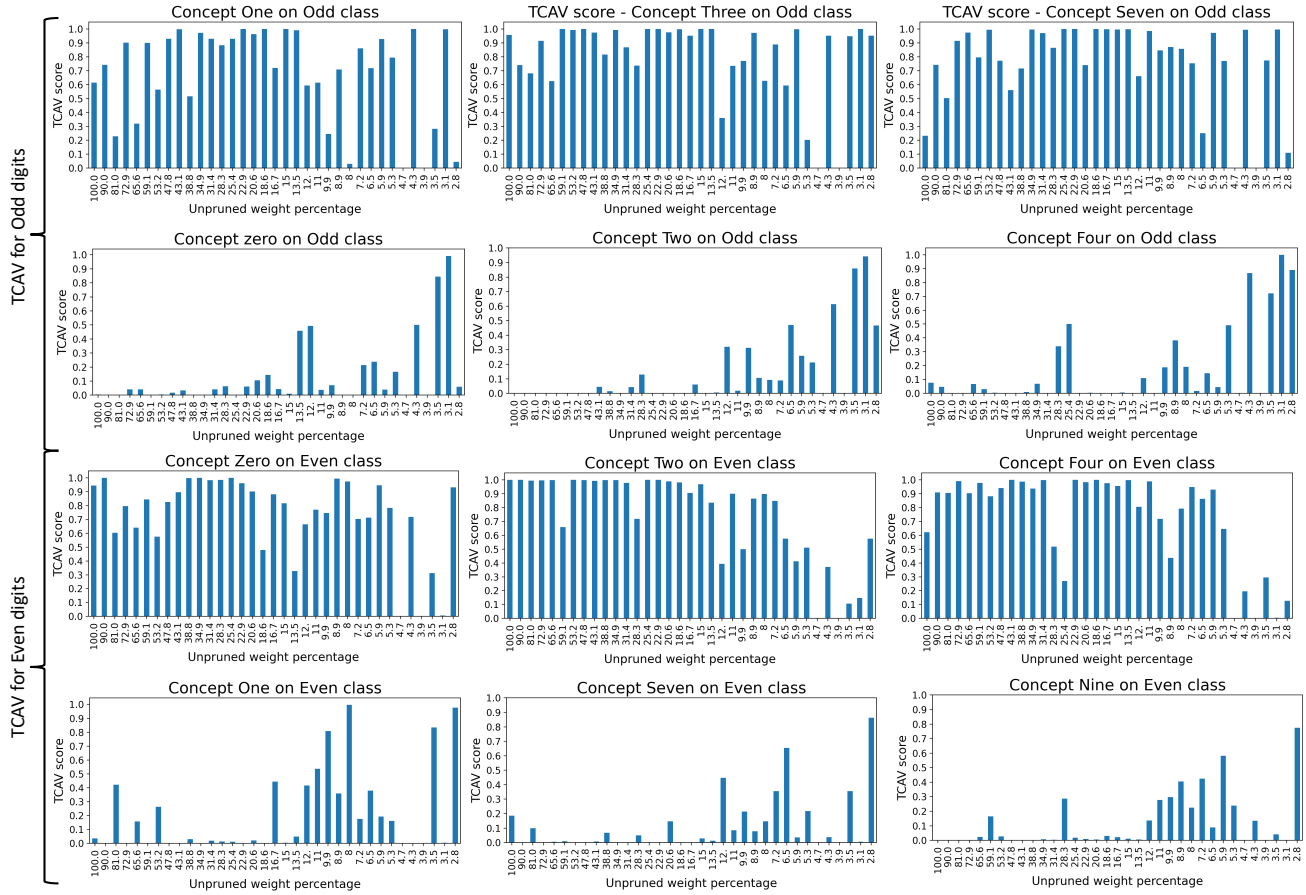

Figure 4. TCAV scores of different concepts for MNIST dataset classifying even / odd digits. The later pruned models have ins-consistent high TCAV scores for unimportant concepts for the respective class.

and prune until 23% of the weights of the original model with the largest magnitude remain. Also, we compute the TCAV score for the concepts indicated in the First Order Logic (FOL) formula for every bird class in Table 14 of the Appendix from the paper Entropy-Lens [6].

4.2. Results

4.2.1 Evaluating performance of the pruned subnetworks

Figure 2 and 3 demonstrate the accuracies for each of the pruned subnetworks. For a easy dataset like MNIST, LTH is able to find high performing subnetworks from the original ResNet18. For instance, the pruned network with 53.2% of the weights (*these weights have larger magnitude compared to the pruned weights*) achieves the highest accuracy of approximately 98.3%. Even pruned network with 7.2% of the weights achieves an accuracy of more than 90%. However, the accuracy of the pruned subnetworks drops consistently for CUB-200 dataset. The drop is steep after we continue pruning the networks after 53.2% of the remaining weights. Next we evaluate the success/failure of LTH in terms of global and local explanations.

4.2.2 Evaluating global explanations of the pruned subnetworks

Figure 4 demonstrates the TCAV scores as a global explanation for various concepts in the MNIST dataset to classify a digit as even / odd. Concept *zero*, *two*, *four* should have a significant impact on the classifier to predict a digit as even. Likewise, *one*, *seven*, *nine* should have a little influence on the classifier to predict a digit as even. The 1^{st} and 2^{nd} row of the figure 4 illustate this. However, this behavior is not consistent as we prune more weights. For example, the TCAV scores of concepts *two* and *four* are very low to predict even digits for the pruned networks with 4.7% or less remaining weights. Further, we observe high TCAV scores for the concept *one* to classify even digits for pruned-networks with 9.9%, 8%, 3.5% and 2%. As per figure 2, these smaller subnetworks perform poorly on the test set.

Similarly, figure 5 demonstrates the TCAV scores for different concepts to classify bird “Rusty Blackbird”, “Red Winged Blackbird”, “Least Auklet” bird species in the

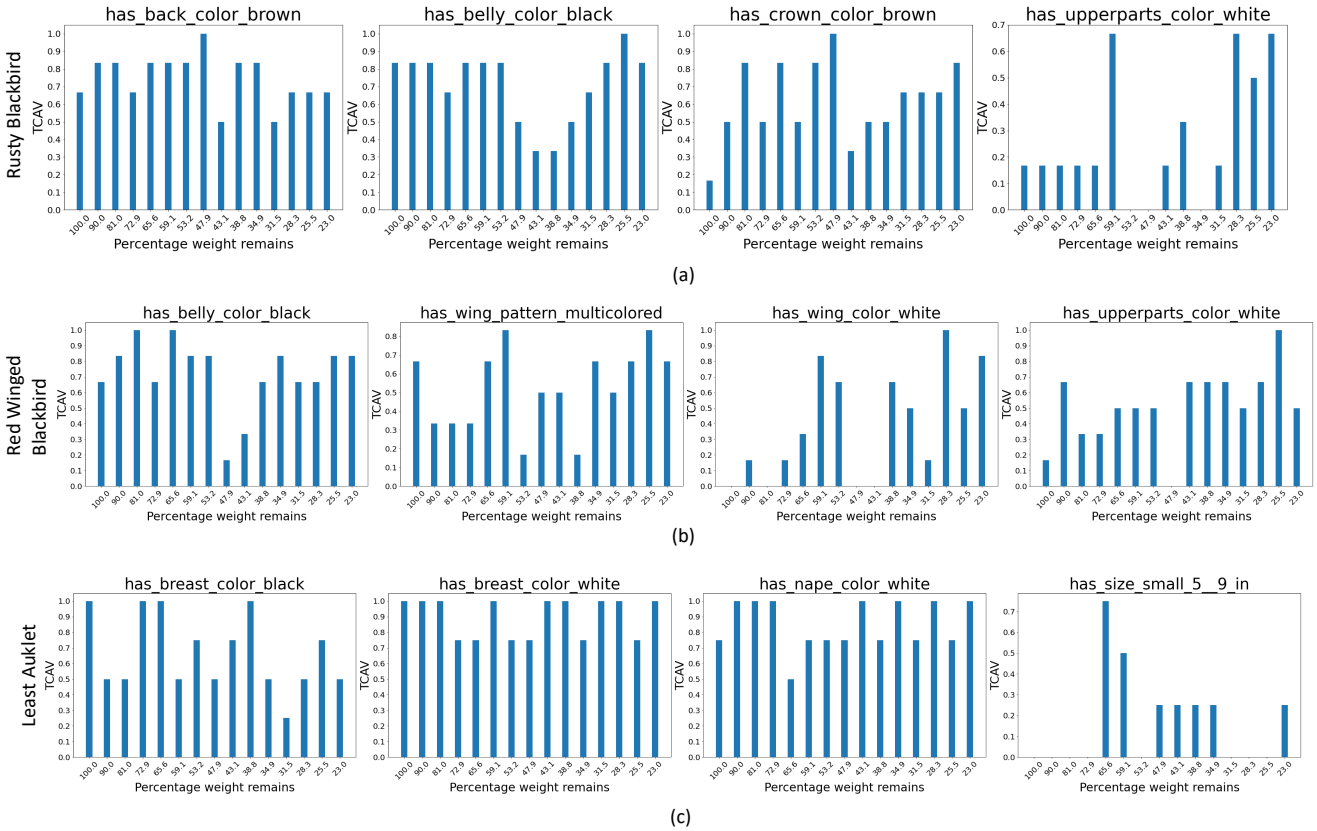

Figure 5. TCAV scores of different concepts for CUB-200 dataset classifying (a) "Rusty Blackbird"; (b)"Red Winged Blackbird"; (c)"Least Auklet" bird species. We compute the TCAV scores for those concepts, discovered in FOL formula indicated in [6]. The later pruned models have ins-consistent high TCAV scores for unimportant concepts for the respective class.

CUB-200 dataset. The 1^{st} three concepts in the figure are the essential concepts included in the class-level explanation formula in [6]. These concepts should have high importance in predicting individual bird species. The 4^{th} concept is chosen randomly from the set of concepts absent from the formula. So, the TCAV score of this concept for predicting the corresponding class should be low or random. The TCAV scores of the prune networks are consistent in the initial rounds and then behave randomly. For instance, the TCAV scores of the concepts *has_back_color_brown* and *has_belly_color_black* are mostly in the higher side for predicting "Rusty Blackbird" (1^{st} row). As per our expectation, the initial pruned networks have low TCAV scores for the random concept *has_upperparts_color_white* for predicting "Rusty Blackbird". However, the TCAV score is high for the same, corresponding to the pruned-networks with 59.1%, 28.3%, 25.5%, and 23% of the remaining weights. Figure 3 highlights the poor performance of these pruned models. For more results of TCAV scores for CUB-200, refer to the figure 7 in the Appendix. As we prune more weights, the network learns a new representation and relies on concepts deemed unimportant by the original model, degrading its performance.

4.2.3 Evaluating local explanations of the pruned sub-networks

Figure 6 illustrates the qualitative comparison of the Grad-CAM-based saliency maps for various birds corresponding to the different pruned models. Grad-CAM-based saliency maps highlight the input image's important pixels, playing a significant role in the model's prediction. This figure demonstrates that the original model and the model with more weight focus on the object of interest. However, the more pruned network uses many features, including irrelevant ones, for prediction. For example, the Grad-CAM highlights the pixels of the bird's body to detect it as "Rusty Blackbird" for the original model with 100% weights. The saliency map highlights the background and the hand where the bird sits as we prune. We discover similar trends for "Parakeet Auklet" as well. The saliency map of the original model only highlights the head's pixels and some of the body parts. However, the pruned model, with 28.3% of the remaining weights, ignores the head when predicting the

Figure 6. Qualitative plots of saliency maps using Grad-CAM for “Black Footed Albatross”, “Rusty Blackbird”, “Least Auklet”, “Parakeet Auklet”. Saliency maps of the later pruned models highlight irrelevant image pixels to classify the corresponding bird, hindering the accuracy.

class. As the pruned models learn new representations during finetuning, they rely on inferior features for the downstream prediction, hindering the performance compared to the original model shown in figure 3. For more Grad-CAM results, refer to the figure 8 in the Appendix.

5. Conclusion & Future Work

We study the success / failure modes of LTH using global and local explainability. Specifically, we prune a deep neural network using iterative LTH. Next we estimate the TCAV scores and Grad-CAM based saliency maps highlighting the concept-level and pixel-level importance respectively. We observe that LTH struggles for complicated datasets. Furthermore, the pruned-networks with more remaining weights highlights relevant concepts and pixels; the networks with less weights do not. As a result, we conclude that magnitude iterative pruning does not emphasize the relevant concepts or pixels as the original model; in fact the opposite is true. In future, alongside of pruning the teacher-student framework can be employed. The original network and the subsequent pruned networks will be considered as teacher and student models respectively. The saliency maps of the teacher model will ensure the student model to focus on the relevant pixels even with the limited network capacity. Also, we aim to apply our method to other modalities such as medical images or video in future.

6. Reproducibility

Refer to the github repository for the code - `https://github.com/Shantanu48114860/Explainability-with-LTH`

References

[1] A. Abid, M. Yuksekgonul, and J. Zou. Meaningfully explaining model mistakes using conceptual counterfactuals. *arXiv preprint arXiv:2106.12723*, 2021.

[2] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

[3] R. Alharbi, M. N. Vu, and M. T. Thai. Learning interpretation with explainable knowledge distillation. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 705–714. IEEE, 2021.

[4] D. Arpit, S. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pages 233–242. PMLR, 2017.

[5] J. Ba and R. Caruana. Do deep nets really need to be deep? *Advances in neural information processing systems*, 27, 2014.

[6] P. Barbiero, G. Ciravegna, F. Giannini, P. Lió, M. Gori, and S. Melacci. Entropy-based logic explanations of neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6046–6054, 2022.

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

[8] Y. Bengio, N. Roux, P. Vincent, O. Delalleau, and P. Marcotte. Convex neural networks. *Advances in neural information processing systems*, 18, 2005.

[9] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-cam: Improved visual explanations for deep convolutional networks. *arXiv: 1710.11063*, 2017.

[10] X. Cheng, Z. Rao, Y. Chen, and Q. Zhang. Explaining knowledge distillation by quantifying the knowledge. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12925–12935, 2020.

[11] J. Crabbé and M. van der Schaar. Concept activation regions: A generalized framework for concept-based explanations. *arXiv preprint arXiv:2209.11222*, 2022.

[12] J. Frankle and D. Bau. Dissecting pruned neural networks. *arXiv preprint arXiv:1907.00262*, 2019.

[13] J. Frankle and M. Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

[14] S. Han, J. Pool, J. Tran, and W. Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

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

[16] G. Hinton, O. Vinyals, J. Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[17] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, et al. Interpretability beyond feature attribution: quantitative testing with concept activation vectors (tcav). arxiv. *arXiv preprint arXiv:1711.11279*, 2017.

[18] P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan, and B. Kim. The (un) reliability of saliency methods. arxiv e-prints, page. *arXiv preprint arXiv:1711.00867*, 2017.

[19] P. W. Koh, T. Nguyen, Y. S. Tang, S. Mussmann, E. Pierson, B. Kim, and P. Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

[20] Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[21] Y. LeCun, J. Denker, and S. Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

[22] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[23] G. Montavon, W. Samek, and K.-R. Müller. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018.

[24] B. Neyshabur, R. Tomioka, and N. Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv preprint arXiv:1412.6614*, 2014.

[25] V. Pillai, S. A. Koohpayegani, A. Ouligian, D. Fong, and H. Pirsiavash. Consistent explanations by contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10213–10222, 2022.

[26] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K.-R. Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.

[27] R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: visual explanations from deep networks via gradient-based localization. 2016. *arXiv preprint arXiv:1610.02391*, 2016.

[28] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*, 2016.

[29] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

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

[31] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

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

[33] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[34] C.-K. Yeh, B. Kim, S. Arik, C.-L. Li, P. Ravikumar, and T. Pfister. On concept-based explanations in deep neural networks. 2019.

[35] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[36] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

Appendix

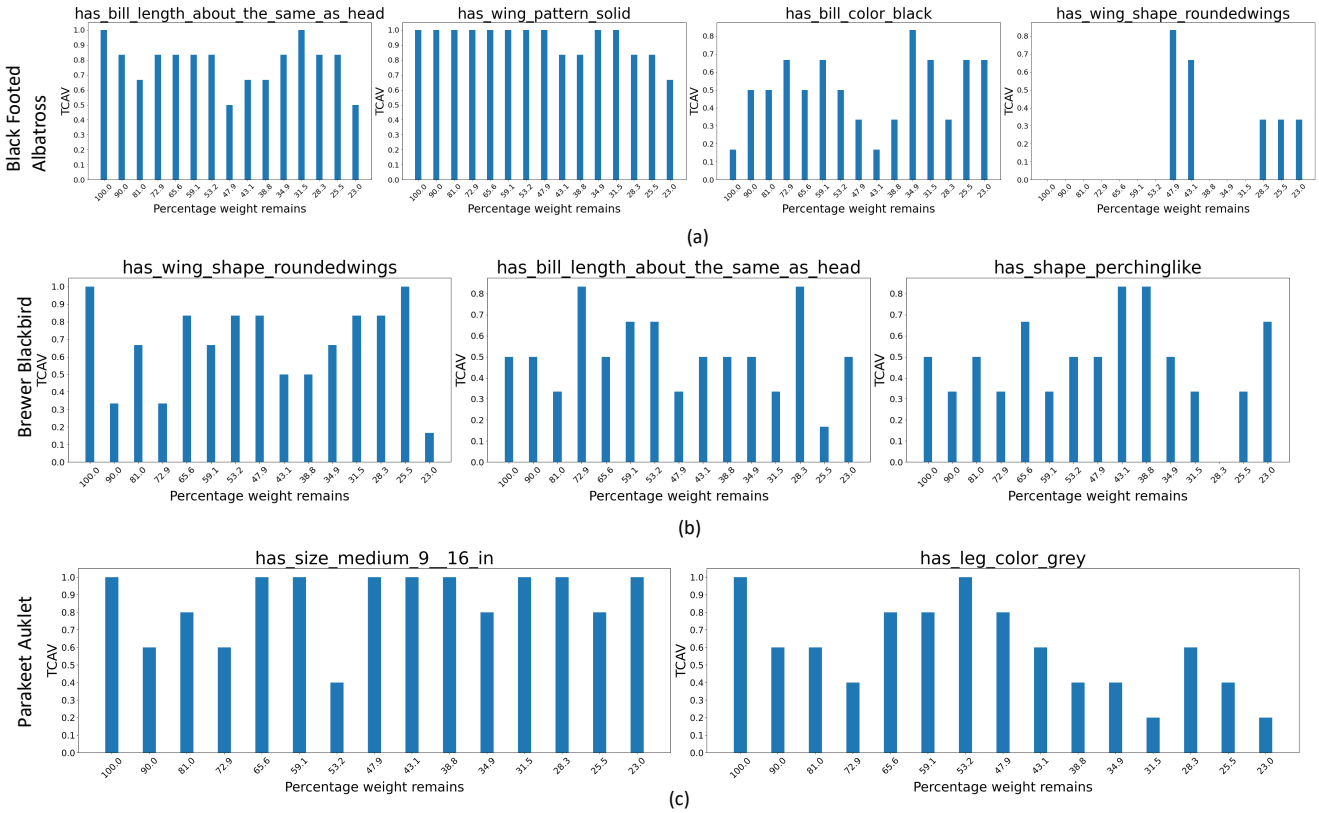

Figure 7. TCAV scores of different concepts for CUB-200 dataset classifying (a)"Black Footed", (b)"Brewer Blackbird Albatross", (c)"Parakeet Auklet" bird species. We compute the TCAV scores for those concepts, discovered in FOL formula indicated in [6].

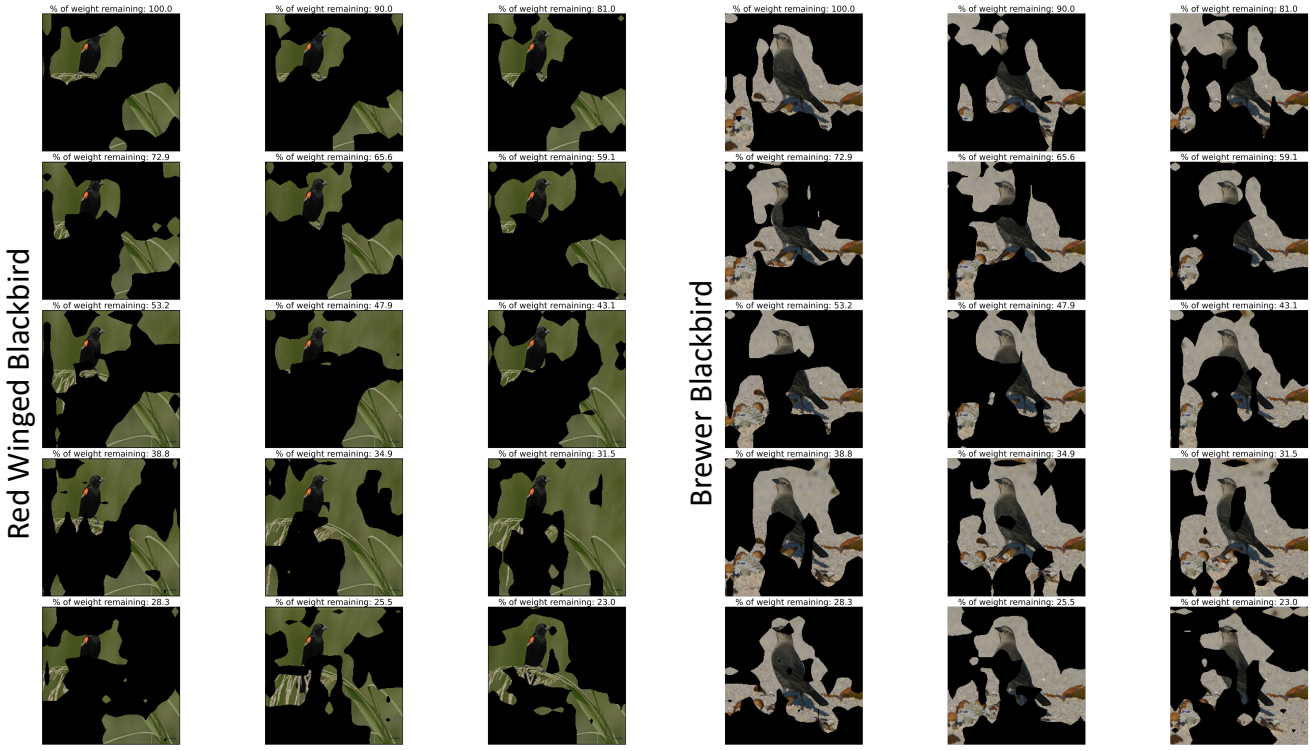

Figure 8. Qualitative plots of saliency maps using Grad-CAM for "Red Winged Blackbird", "Brewer Blackbird". Saliency maps of the later pruned models highlight irrelevant image pixels to classify the corresponding bird, hindering the accuracy.