ATTENDROP FOR CONVOLUTIONAL NEURAL NETWORKS

Anonymous ICME submission

ABSTRACT

Dropout lacks success in Convolutional Neural Networks (CNN) since the spatially correlated features still grant a chance for rich information to flow through the network. Recently, some structured dropout methods such as Cutout and Dropblock have been proposed to resolve such an issue. However, the randomness behind such structured methods in deciding mask positions risks the model's capability of focusing on non-trivial spatial contextual information. In addition, fixed square masks breed the model's inflexibility in dealing with heterogeneous scales and shapes of objects. Therefore, to enhance the effectiveness of structured dropout on CNNs, we propose AttentionDrop that drops units intelligently. To be more specific, AttentionDrop 1) firstly generates an adaptive mask on units of irregular regions in a feature map according to their activation values; 2) the generated adaptive mask is then optimized by a soft mask sampled from a Bernoulli distribution. Experiments on CIFAR-10, CIFAR-100 and SVHN datasets demonstrate AttentionDrop's competitive performance on image classification tasks compared with state-of-the-art methods, yielding test accuracy of 96.37%, 78.60%, and 98.31% respectively.

***Index Terms*—** Dropout, Attention, Adaptive, Regularization, CNN, Classification

1. INTRODUCTION

Deep neural networks are widely used in the field of computer vision. As a matter of fact, many state-of-the-art methods leverage CNNs for multifarious visual tasks, such as image classification [1, 2], object detection [3] and semantic segmentation [4, 5]. However, most prevalent models like ResNet [6], Inception [7], and DenseNet [8] are inclined to suffer from over-parameterization, thus giving rise to the problem of overfitting. In this regard, regularization methods, such as weight decay and dropout [9], are harnessed to alleviate the problem of overfitting.

To date, dropout [9] has been a commonly used regularization method and proved to be effective for fully connected layers of deep neural networks. However, it becomes less effective for CNNs due to the fact that features in CNNs are correlated spatially [10]. As a result, to enhance the effectiveness of dropout on CNNs, some variants of dropout like Cutout [11], SpatialDropout [12], and DropBlock [10] have been recently proposed. Cutout operates on the image level by randomly masking square regions of input, while SpatialDropout operates on the feature level where an entire channel is dropped from a feature map. Recently proposed DropBlock [10], a form of structured dropout, generalizes Cutout to every feature map and obtains better performance.

Despite the performance gain of such regularization methods, they might perform unsatisfactorily for the following reasons. First, they drop activation units randomly, thus ignoring the spatial contextual information that might be indispensable; besides, considering various scales and shapes of objects, such methods are susceptible to limited flexibility with fixed square masks. As shown in Figure 1, previous regularization methods, including dropout [9] and DropBlock [10], mask feature maps randomly. Therefore, when the mask covers the entire object, such methods might suffer from significant performance deterioration and thus make the network difficult to train.

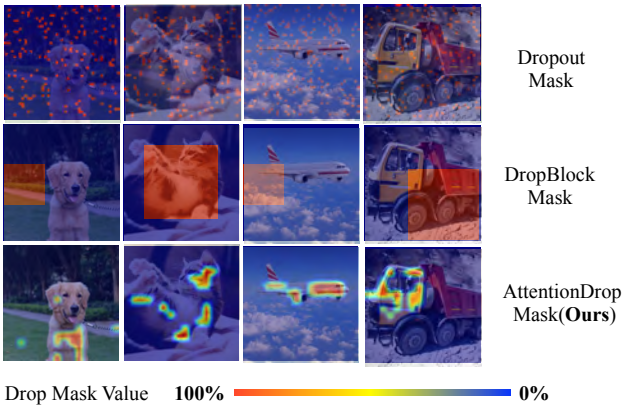

Fig. 1. Masks of naive dropout [9], Dropblock [10] and AttentionDrop. In DropBlock, it sometimes backfires because it drops overmuch non-trivial information in the feature map, which makes the network fail to learn what a cat is when the cat is completely "dropped". Naive dropout is rendered less effective because the dropped information can still be restored through surrounding features given the spatial correlation.

In order to address the above issues, we propose a novel regularization method – AttentionDrop – by considering attention information. Specifically, AttentionDrop 1) firstly produces irregularly shaped adaptive masks according to the activation values of units in a feature map; 2) a soft mask sampled from a Bernoulli distribution is utilized to optimize

the original adaptive mask. The above two approaches are combined together as AttentionDrop which is shown in Figure 2. Experiments show that feature maps are activated in a better way by AttentionDrop, which indicates that the CNN model with AttentionDrop takes into account a wider variety of features rather than only the ones with high activation values when making predictions. In addition, the heat map of AttentionDrop masks the feature map with an object-like shape, while that of Dropout and DropBlock just focus on random regions.

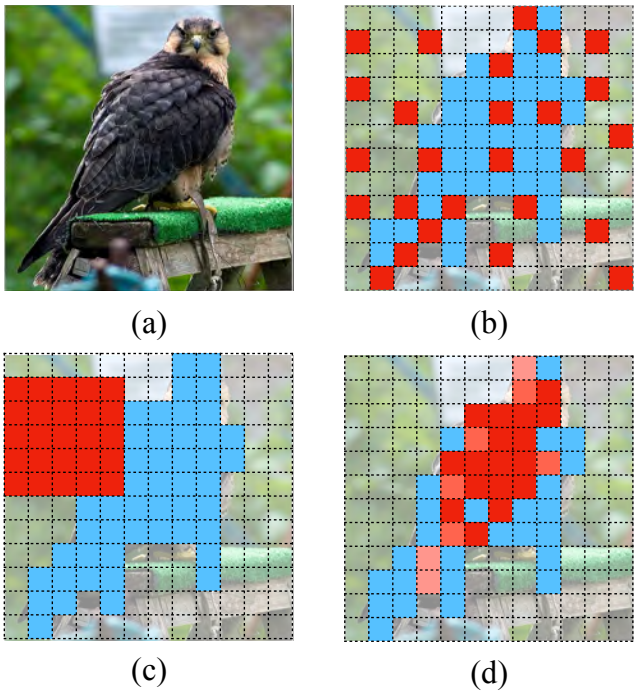

Fig. 2. The comparison among Dropout, DropBlock and AttentionDrop. The main object is marked by blue squares, and Dropout operations are marked by red squares. (a) is the input image; (b) is the standard Dropout; (c) is DropBlock; (d) is AttentionDrop.

To the best of our knowledge, AttentionDrop is the first regularization method that is capable of deciding the position and shape of the mask adaptively and dropping with an attention mask. The two characteristics enable AttentionDrop to adjust according to the activation values of a feature map. On the one hand, AttentionDrop highlights the overall context information while retaining the main object in the feature map. On the other hand, parameters for AttentionDrop does not require too much manual adjustment thanks to its intelligence in choosing features.

Adding AttentionDrop to ResNet-18 [13], which is a state-of-the-art classification model, brings an increase in performance. To be more specific, the accuracy increases on CIFAR-10, CIFAR-100 [1] and SVHN [14], are from 95.28% to 96.37%, from 77.54% to 78.60% and from 97.40% to 98.31% respectively.

2. RELATED WORK

Regularization is critical for neural networks especially for models with a large number of parameters. After Dropout [9] opened the door to regularize model through simple "drop" mechanism, a number of regularization approaches are proposed such as DropPath [15], DropConnect [16], shake-shake regularization [17], and ShakeDrop regularization [18], maxout [19], StochasticDepth [20]. DropPath set some layers in the neural network to zeros after training, rather than for a specific neuron. DropConnect drops the connections of the network instead of activations. Most of these methods insert noise information to the existing neural networks, therefore, they prevent the model from overfitting. Nonetheless, our method AttentionDrop adds attention mechanism to extend the existing dropout-based methods. Further experiments demonstrate this attention feature is beneficial for model to understand the image, thus, achieve a better performance.

3. METHODS

AttentionDrop is a simple but effective technique that originates intuitively from Dropout and DropBlock. As DropBlock, it drops contiguous regions from the feature map. However, instead of dropping only regions with fixed shapes, AttentionDrop drops regions based on the activations of the input. Moreover, the original binary mask is substituted by a soft mask.

Denoting the feature map of the current layer as $F^{(n)}$ and that of the next layer as $F^{(n+1)}$, traditional dropout can be described as

$$\gamma_{ij} = Bernoulli(\alpha) \tag{1}$$

$$M_{ij} = \begin{cases} 0 & \gamma_{ij} = 1 \\ 1 & \gamma_{ij} = 0 \end{cases} \tag{2}$$

$$F^{(n+1)} = M \odot F^{(n)} \tag{3}$$

where γ_{ij} is sampled from a Bernoulli distribution with the dropout probability $\alpha \in (0, 1)$ and M is the generated mask.

3.1. Dropout with Adaptive Mask

We first propose a simple implementation of the adaptive mask. Letting $F^{(n)}_{top\_\beta}$ denote the top βth percentile value in $F^{(n)}$, we define the shape mask as

$$S_{ij} = F^{(n)}_{ij} \leq F^{(n)}_{top\_\beta} \tag{4}$$

For example, when setting β to 20, the masked activation units tend to cover a contiguous region, such as a dog's head, the frontier part of a plane, or a tree's trunk. Therefore, we do not need extra operations to connect the high activation units

in order to make masks contiguous. Then we compute the AttentionDrop mask as

$$M_{ij} = \begin{cases} S_{ij} & \gamma = 1 \\ 1 & \gamma = 0 \end{cases} \tag{5}$$

where γ is again sampled from a Bernoulli distribution as above and M is the dropout mask with an adaptive shape. This operation differs from the original dropout in that it constrains the region to a small portion of the feature map and drops / keeps the units in the region altogether. It could be interpreted as focusing all the attention on covering important regions so that the irrelevant background is ignored by the mask. With definite shapes and positions, dropout performs more effectively through an adaptive mask.

3.2. Dropout with Soft Mask

Most Dropout techniques employ binary masks where activation units are either dropped or kept. In this section, we explore a new way of dropout with masks consisting of soft values between 1 and 0. In addition, as discussed before, the mask should have preference for highly activated units. Firstly, we normalize the current feature map through min-max normalization as

$$\widehat{F_{ij}^{(n)}} = \frac{F_{ij}^{(n)} - F_{min}^{(n)}}{F_{max}^{(n)} - F_{min}^{(n)}} \tag{6}$$

Note that $\widehat{F_n^{(n)}}$ naturally suffices for the above two properties: soft value and attention mechanism. Normalized features in a uniform range are critical for assigning appropriate weights later. Then the soft mask M_{ij} can be defined as

$$S_{ij} = 1 - \widehat{F_{ij}^{(n)}} \tag{7}$$

$$M_{ij} = \begin{cases} S_{ij} & \gamma = 1 \\ 1 & \gamma = 0 \end{cases} \tag{8}$$

Lastly, the factor $1/sum(M)$ applies appropriate scaling to the adjustment for the varying range of valid (unmasked) inputs $F_n^{(n)}$

$$F_{n+1}^{(n)} = \frac{F_n^{(n)} \odot M}{sum(M)} \tag{9}$$

As a result, soft mask M_{ij} allows features to be partially kept, and thus is more flexible than the binary mask. Experiments also substantiate the better effectiveness of the soft mask.

3.3. AttentionDrop

Combining the adaptive mask and soft mask together, the AttentionDrop algorithm can be described in Algorithm 1.

As we can see from Algorithm 1, AttentionDrop selects well-learned features and drops them intelligently, in stark contrast to the aimlessness of methods such as naive dropout [9] and DropBlock [10]. After applying AttentionDrop to $F^{(n)}$, the highly activated units in $F^{(n)}$ are dropped effectively. Without the dropped information, the network is encouraged to focus more on complimentary and less prominent features. Besides, with the attention mechanism, the model is still capable of obtaining the information from the masked features, because of the soft mask with attention mechanism instead of a hard mask that comprises only 1 and 0.

Following [10], we gradually increase α. In our experiments, we use a linear scheduler to increase the value of α from 0 to 0.1.

We visualize the masks generated by Dropout, DropBlock and AttentionDrop in Figure 3. We can see that the masks generated by AttentionDrop shed light mostly on the regions with rich semantic information instead of masking the entire object, therefore maintaining sufficient features for the classification task. In addition, AttentionDrop also generates masks effectively to remove maximally activated features and encourage the network to consider less prominent features.

Algorithm 1 The work flow of AttentionDrop.

Input: Current layer's feature $F^{(n)}$
Output: Next layer's feature: $F^{(n+1)}$

if phase == interface **then**
    return $F^{(n)}$
Normalize the current layer's feature

$$\widehat{F_{ij}^{(n)}} \Leftarrow \frac{F_{ij}^{(n)} - F_{min}^{(n)}}{F_{max}^{(n)} - F_{min}^{(n)}}$$

Get the top β activation from normalized feature map, to localize the adaptive mask with irregular shape

$$\widehat{F_{ij}^{(n)}} \Leftarrow (F_{ij}^{(n)} \geq F_{top\_\beta}^{(n)}) \odot \widehat{F_{ij}^{(n)}}$$

Get the attention weights $S_{ij} \Leftarrow 1 - \widehat{F_{ij}^{(n)}}$
For each element in feature map $F^{(n)}$, get an initial dropout possibility $\gamma = Bernoulli(\alpha)$
Get the AttentionDrop mask
if γ == 1 **then**
    $M_{ij} = S_{ij}$
else
    $M_{ij} = 1$
Apply mask to the output feature and scale the output

$$F^{(n+1)} \Leftarrow \frac{F^{(n)} \odot M}{sum(M)}$$

return $F^{(n+1)}$

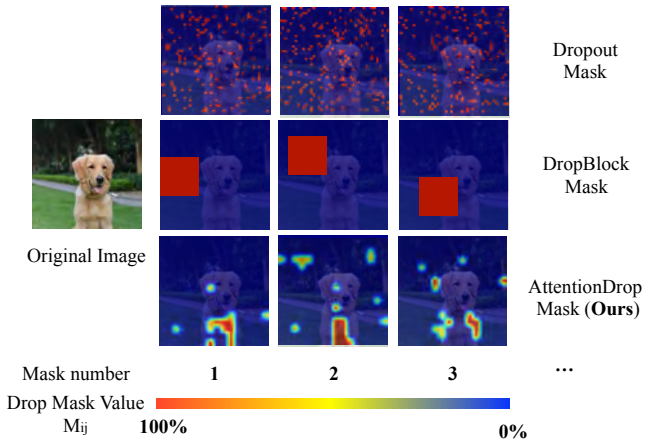

Fig. 3. Masks generated by Dropout, DropBlock and AttentionDrop. We randomly choose 9 masks generated by 3 dropout methods and visualize them through the illustration of heat maps. It can be seen that the masks of Dropout and DropBlock cover the images randomly, whereas the masks of AttentionDrop cover semantic information more purposefully. In fact, dropping out multiple randomly chosen separate regions is ineffective because the removed information can be retrieved through nearby units, thus rendering Dropout ineffective. In addition, DropBlock also risks network's capability of learning from features since it allows a chance for the entire object to be dropped. Nonetheless, AttentionDrop drops semantic information effectively since the highly activated units usually cover the object partially, which compels the model to reference global context features and learn representations in a better way while still keeping the indispensable features.

4. EXPERIMENTS

We apply three types of AttentionDrop: Dropout with **A**daptive **M**ask (AM), Dropout with **S**oft **M**ask (SM) and Dropout with **A**ttention**D**rop (AD), to three general image classification datasets – CIFAR-10, CIFAR-100 and SVHN. All experiments are based on ResNet-18 model [13].

We normalize the datasets with per-channel mean and deviation. Standard data augmentation scheme is also incorporated. Images are first zero-padded with 4 units on each side to obtain a 40×40 image, then a 32×32 cropped region is randomly extracted. Additionally, images are also randomly mirrored horizontally with 50% probability. The learning rate is decayed by the factor of 1e-1 at 0.4, 0.6, 0.8 ratio of total epochs. The model will automatically stop training when it converges. All experiments are performed on a 1080Ti GPU.

4.1. CIFAR-10 and CIFAR-100

Both CIFAR datasets consist of 60,000 images with a shape of 3×32×32 [1]. CIFAR-10 consists of 10 different classes, such as dogs, cats, horses, and buses. CIFAR-100 includes 100 classes with much fewer pictures in each class. Thus, the classification task on CIFAR-100 is far more exacting compared to that on CIFAR-10 given its wide range of image types and resemblance among classes. For example, on the average, each class in CIFAR-10 contains around 5,000 images; however, in CIFAR-100, the quantity plunges to only 500. In our experiments, each dataset is split into a training set with 50,000 images and a test set with 10,000 images.

We display the test accuracy of both datasets in Table 1 and the accuracy of CIFAR-100 during training in Figure 4. We can see that all of the AttentionDrop models, which is a combination of the adaptive mask and soft mask, outperform the state-of-art ones on the CIFAR-10 and CIFAR-100 datasets. In fact, an adaptive mask pinpoints the most non-trivial units in a feature map, offering more inspirations upon the mask position; the soft mask overshadows the hard mask, enlightening the model about a more tactful way to drop the units. Therefore, the combination of the two mechanisms boosts the model to obtain the best results.

Table 1. Test accuracy of classification on CIFAR-10 and CIFAR-100 test sets. The best results are highlighted in **bold**. DropBlock is based on [10] and Cutout is based on [11]

Models	CIFAR-10 (top-1)	CIFAR-100 (top-1)
ResNet-18[11]	95.28±0.21	77.54±0.31
ResNet-18+Cutout[11]	96.01±0.13	78.04±0.24
ResNet-18+DropBlock	95.41±0.14	78.05±0.65
ResNet-18+AM	95.31±0.05	77.58±0.12
ResNet-18+SM	95.35±0.17	78.22±0.45
ResNet-18+AD	95.49±0.16	78.47±0.07
ResNet-18+AD+Cutout	**96.37±0.09**	**78.60±0.16**

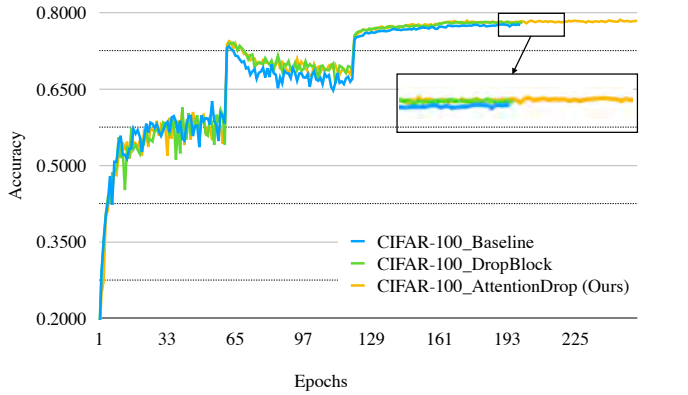

Fig. 4. Training accuracy of classification on CIFAR-100 training set. The models with AttentionDrop (AD) converge to a higher accuracy during training than the ones with either standard Dropout or DropBlock. In addition, they also achieve a better performance than the ones with AM or SM used alone.

4.2. SVHN

The Street View House Numbers (SVHN) [14] is a real-world dataset consisting of 630,420 labeled digits captured from

street view images. The official dataset contains 73,257 training images and 26,032 test images, and there are also 531,131 additional training images available. We use both training sets to train our models. The classification results are shown in Table 2 and we also display the accuracy during training in Figure 5. We can see that models with AttentionDrop again achieves the best performance.

Table 2. Test accuracy of classification on SVHN test set. The best results are highlighted in **bold**. DropBlock is based on [10] and Cutout is based on [11]

Models	SVHN (top-1)
ResNet-18[11]	97.40±0.12
ResNet-18+Cutout[11]	98.10±0.07
ResNet-18+DropBlock	97.90±0.15
ResNet-18+AM	98.16±0.05
ResNet-18+AD+Cutout	**98.31±0.09**

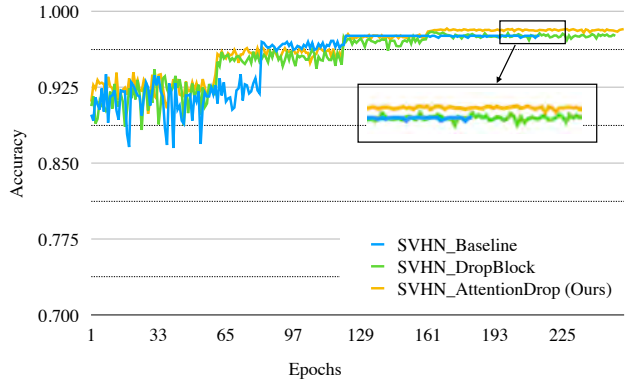

Fig. 5. Training accuracy of classification on SVHN training set. The models with AD converge to a higher accuracy than that with either standard Dropout or DropBlock.

4.3. Analysis of Activations

In this part, we design several experiments to prove the effectiveness of AttentionDrop in dropping semantic information of feature maps.

First, we use class activation mapping (CAM) [21] to visualize the activation units of ResNet-18 [13] on images as shown in Figure 6. We can see that the AttentionDrop model is able to capture discriminative spatial information, and that the feature map shape indicates a stronger correlation to the object in the image. Hence, the AttentionDrop model demonstrates strong competence in learning spatially distributed features thanks to its adaptive dropout shapes and soft masks.

Additionally, in order to gain a better understanding of the effect of AttentionDrop, we compare the average magnitude of feature activations in ResNet-18 on CIFAR-10 when trained with and without AttentionDrop. The models were trained on the same settings as illustrated in Section 4.1. We quantify feature activations in different magnitude scales averaged over 512 randomly sampled images. Specifically, to scale the output of convolutional layers for comparison, we choose the feature maps of Batch Normalization layers after each convolutional layer as the activation units for visualization. Besides, considering the predominant zero values of activations resulting from the sparsity in deep learning model, we eliminate the activations less than 0.02 for better illustration. The results are displayed in Figure 7. We observe that the activations in the AttentionDrop model significantly outstrip that in the DropBlock model, which thus demonstrates that AttentionDrop manages to prompt the network to consider a wider variety of features when making predictions.

5. CONCLUSION

In this paper, we propose an original dropout scheme – AttentionDrop, in order to make up for the deficiency of naive dropout that usually employs hard masks and drops without any priori knowledge. Specifically, AtentionDrop makes use of the activation values in each feature map and generates soft masks with adaptive shapes. Besides, it adjusts mask shapes and mask weights during the process of training. We visualize the masks and validate their better utilization of the contextual information from the input image. Extensive classification experiments are performed on different datasets using AttentionDrop and indicate a pronounced gain in performance. We believe that AttentionDrop possesses significant applicability to other tasks like object detection, semantic segmentation, and etc.

6. REFERENCES

[1] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," Tech. Rep., Citeseer, 2009.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[3] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[4] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask r-cnn," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2980–2988.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

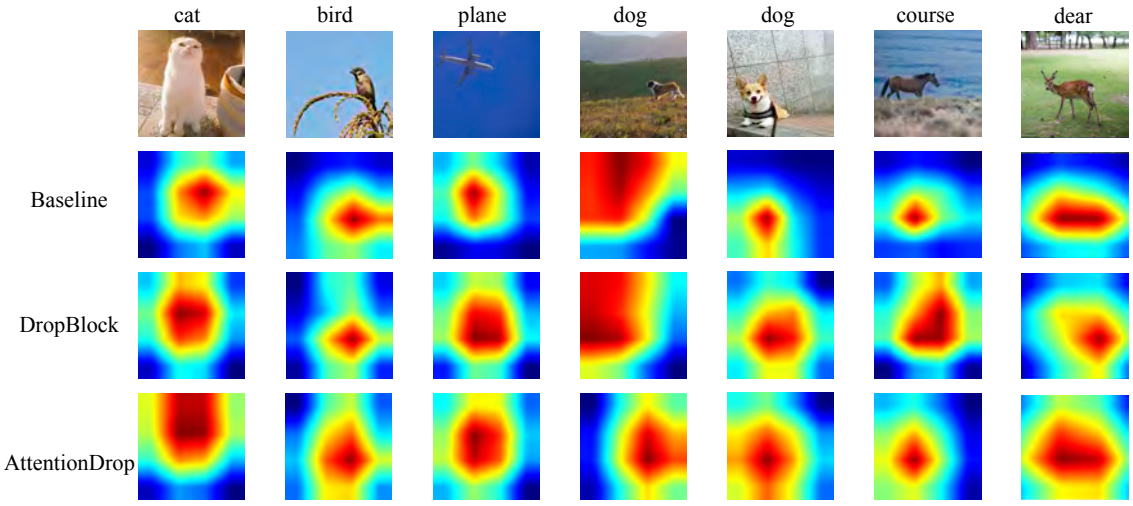

Fig. 6. Class activation mapping (CAM) [21] for ResNet-18 [13] model trained with Dropout, DropBlock, and AttentionDrop. The model trained with AttentionDrop tends to focus on semantic regions with a more precise shape.

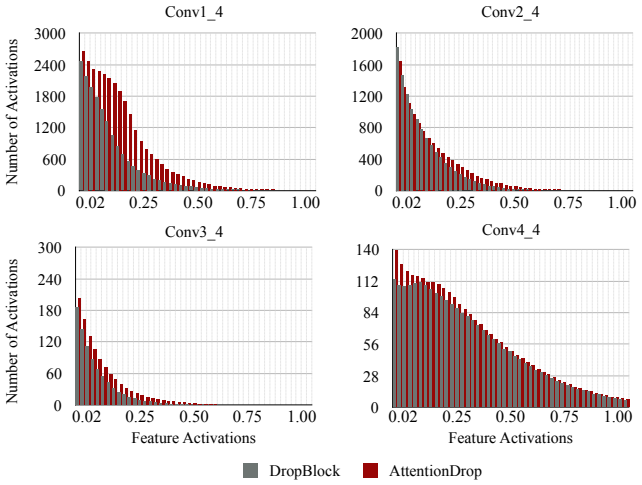

Fig. 7. Average activation range measure. Number of feature activations averaged over all 512 random samples. A standard ResNet18 is compared with a ResNet18 trained with AttentionDrop at four different depths.

[7] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[8] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks.," in *CVPR*, 2017, vol. 1, p. 3.

[9] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[10] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V. Le, "Dropblock: A regularization method for convolutional networks," *CoRR*, vol. abs/1810.12890, 2018.

[11] Terrance DeVries and Graham W Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[12] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler, "Efficient object localization using convolutional networks," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 648–656, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," in *ECCV*, 2016.

[14] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng, "Reading digits in natural images with unsupervised feature learning," 2011.

[15] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le, "Learning transferable architectures for scalable image recognition," *CoRR*, vol. abs/1707.07012, 2017.

[16] Li Wan, Matthew D. Zeiler, Sixin Zhang, Yann LeCun, and Rob Fergus, "Regularization of neural networks using dropconnect," in *ICML*, 2013.

[17] Xavier Gastaldi, "Shake-shake regularization," *CoRR*, vol. abs/1705.07485, 2017.

[18] Yoshihiro Yamada, Masakazu Iwamura, and Koichi Kise, "Shakedrop regularization," *CoRR*, vol. abs/1802.02375, 2018.

[19] Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron C. Courville, and Yoshua Bengio, "Maxout networks," in *ICML*, 2013.

[20] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q. Weinberger, "Deep networks with stochastic depth," in *ECCV*, 2016.

[21] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba, "Learning deep features for discriminative localization," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, 2016.