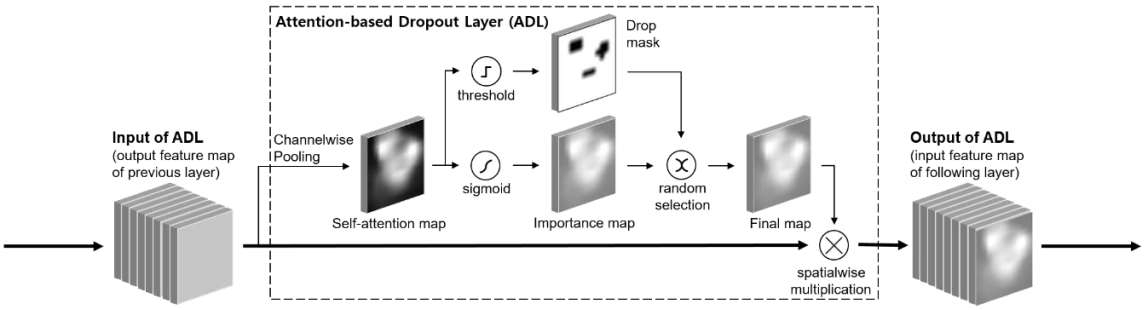
Attention-based Dropout Layer for Weakly Supervised Object Localization
(2019)

Junsuk Choe, and Hyunjung Shim
Notes

Into

In weakly supervised setting, in order to localize objects based using only the image labels, one way to do so is to use class activation mapping (CAM) to find the most dicriminative region the model focuses on for classification, given that a classifier with a reasonable accuracy should observe the most telling region of the object and that co-occur between images to decide its class, but thsese discriminative regions are very limited and do not cover the whole object in question. Some methods have been proposed to over come this limitation, such as erasing these regions to prevent the model from reling only on these regions, an other approche is to randomly remove some patches of the image, but the dicriminative regions might be missed in these cases.

Generally these methods require training the model for mutlipple times or perform multipple farward passes, which require huge computationnal resources, the goad of the authors is to find the dicriminative region accurately in an effective and efficient way, for this they propose Attention based dropout layer, as a lightweight way to erase the discriminative region during a forward pass, a self attention map is obtained using channel wise pooling, and then two masp are generated (1) a drop mask using a threshold γ and (2) and importance map sith a sigmoid contraining the most discrimnative region, and using a *drop_rate*, either the importance map is passed or the drop mask, which is then applied to the input feature maps using a spatial wise multiplication, and the results is that either the most discriminative region are highlited or removed.



ADL: Attention based Dropout Layer

As we mentionned, ADL is applied to the feature maps of the classification model in order to induce the model to learn the entire region of the object, with an input $\mathbf{F} \subseteq \mathbf{R}^{H \times W \times C}$ with H,W as the sptial dimensions, and C the number of channels, using a channel wise average pooling, a self-attention maps is generated $\mathbf{M}_{att} \subseteq \mathbf{R}^{H \times W}$, using a drop threshold γ, the drop mask is obtained $\mathbf{M}_{drop}$ with zeros for the most dicriminative regions and one for the rest. The importance map is generated using a sigmoid od the self attention map for a values between zero and one, the drop mask is then applied stochastically based on *drop_rate*, if not the importance map is passed, and the passed final map is applied to the input of ADL using spatial multiplication.

ADL can applied to each convolutionnal feature map idependently, in a VGG for example, we have 5 blocks, first two: ((conv - relu) x 2- pool), the third and fourth ((conv - relu) x 3- pool), and the last one like the first, and then finnaly a (conv - relu) x 3 followed by a GAP and FC, ADL can be added after each convolution of at the end of each blocl after the pooling, the best location is at the end of block 3 & 4, at the pool3 and pool4, and also in the last convoltion of the last block. In resnet the ADL is added in the end of each resnet block or after the first 7x7 convoltuion or pooling (6 total possible locations), for more details see the impelemntation `https://github.com/junsukchoe/ADL`

Results

Drop mask (%)	Importance map (%)	GT-known Acc (%)	Top-1 Clas (%)	Top-1 Loc (%)
100	0	72.43	57.37	44.11
75	25	**74.78**	62.25	**49.69**
50	50	71.51	64.93	49.33
25	75	67.29	*68.99*	47.98
0	100	47.51	67.78	32.24
N/A	N/A	51.09	67.55	34.41
75	N/A	73.23	61.55	47.67
N/A	25	50.62	*68.50*	33.91
75	25	**74.78**	62.25	**49.69**

Applied feature map	GT-Known Acc (%)	Top-1 Clas (%)	Top-1 Loc (%)
N/A	51.09	67.55	34.41
conv 5-3	57.99	*68.95*	41.73
+ *pool4*	68.22	67.17	48.02
+ *pool3*	**75.41**	65.27	**52.36**
+ *pool2*	71.85	63.76	48.46
+ *pool1*	74.78	62.25	49.69

Table 1. Upper: Accuracy according to *drop_rate*. Middle: Baseline accuracy. Lower: Accuracy when each component has been deactivated. Bold text refers the best localization accuracy, while *italic text* refers the best classification accuracy. N/A indicates that ADL outputs the raw input feature map instead of applying drop mask or importance map.

Method	Backbone	# of Params (Mb)	Overheads		CUB-200-2011		ImageNet-1k	
			parameter (%)	computation (%)	Top-1 Loc (%)	Top-1 Clas (%)	Top-1 Loc (%)	Top-1 Clas (%)
CAM	VGG-GAP [34, 63]	78	0	0	34.41	67.55	42.80*	66.60*
ACoL	VGG-GAP [34, 63]	181	132.05	37.63	45.92*	71.90*	**45.83***	67.50*
ADL	VGG-GAP [34, 63]	78	0	0.00	**52.36**	65.27	44.92	69.48
CAM	MobileNetV1 [11]	16	0	0	43.70	71.94	41.66	68.38
HaS-32	MobileNetV1 [11]	16	0	0	44.67	66.64	41.87	67.48
ADL	MobileNetV1 [11]	16	0	0.00	**47.74**	70.43	**43.01**	67.77
CAM	ResNet50-SE [10, 12]	107	0	0	42.72	80.65	46.19	76.56
ADL	ResNet50-SE [10, 12]	107	0	0.00	**62.29**	80.34	**48.53**	75.85
CAM	InceptionV3 [40, 60]	101	0	0	43.67*	-	46.29*	-
SPG	InceptionV3 [40, 60]	146	44.55	30.05	46.64*	-	**48.60***	-
ADL	InceptionV3 [40, 60]	101	0	0.00	**53.04**	74.55	**48.71**	72.83

Table 3. Quantitative evaluation results on CUB-200-2011 and ImageNet-1k. Bold text refers the best localization accuracy for each backbone network. We also underline the best score in each dataset. Overheads are computed based upon their backbone networks. The accuracy with asterisk* indicates that the score is from the original paper. We leave some *Top-1 Clas* scores blank, because they are not reported in the original paper [60]. For reproducing baseline methods, we use hyperparameters suggested by their original papers [63, 35]. Also, we train and test HaS and ADL under the same setting for a fair comparison.