Towards Open Set Deep Networks: Supplemental

Abhijit Bendale*, Terrance E. Boult
University of Colorado at Colorado Springs
{abendale,tboult}@vast.uccs.edu *

In this supplement, we provide we provide additional material to further the reader as understanding of the work on Open Set Deep Networks, Mean Activation Vectors, Open Set Recognition and OpenMax algorithm. We present additional experiments on ILSVRC 2012 dataset. First we present experiments to illustrate performance of OpenMax for various parameters of EVT calibration (Alg. 1, main paper) followed by sensitivity of OpenMax to total number of "top classes" (i.e. α in Alg. 2, main paper) to consider for recalibrating SoftMax scores. We then present different distance measures namely Euclidean and cosine distance used for EVT calibration. We then illustrate working of OpenMax with qualitative examples for open set evaluation performed during the testing phase. Finally, we illustrate the distribution of Mean Activation Vectors with a class confusion map.

1. Parameters for OpenMax Calibration

1.1. Tail Sizes for EVT Calibration

In this section we present extended analysis of effect of tail sizes used for EVT fitting in Alg 1 in main paper on the performance of the proposed OpenMax algorithm. We tried multiple tail sizes for estimating parameters of Weibull distributions (line 3, Alg 1, main paper). We found that as the tail size increased, the OpenMax algorithm became very robust at rejecting images from open set and fooling set. OpenMax continued to perform much better than SoftMax in this setting. The results of this experiment are presented in Fig 1. However, beyond tail size 20, we saw performance drop on the validation set. This phenomenon can be seen in Fig 2, since F-Measure obtained on OpenMax starts to drop beyond a tail size of 20. Thus, there is an optimal balance to be maintained between rejecting images from open set and fooling set, while maintaining correct classification rate on the validation set of ILSVRC 2012.

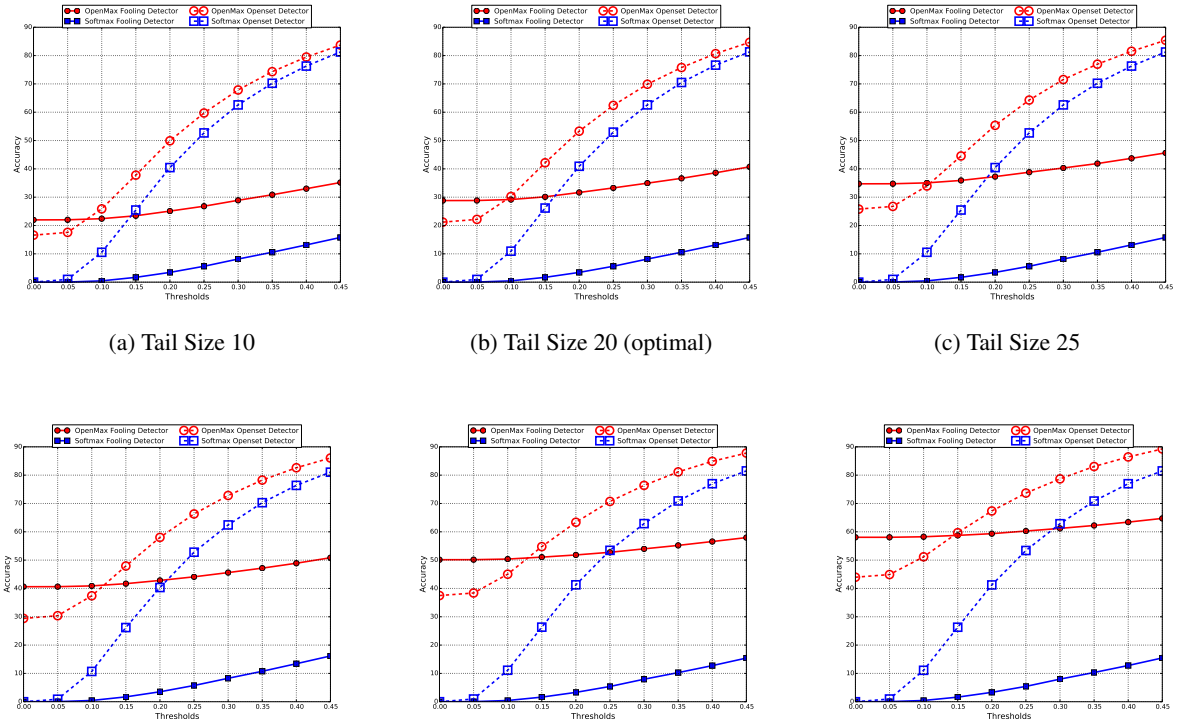
1.2. Top Classes to be considered for revision α

In Alg 2 of the main paper, we present a methodology to calibrate FC8 scores via OpenMax. In this process, we also incorporate a process to adjust class probability as well as estimate the probability for the unknown unknown class. For this purpose, in Alg 2 (main paper), we consider "top" classes to revise (line 2, Alg 2, main paper), which is controlled by parameter α. We call this parameter α rank, where the value of α suggests the total number of "top" classes to revise. In our experiments we found that optimal performance is obtained when $\alpha = 10$. At lower values of α we see a drop in F-Measure performance. If we continue to increase α values beyond 10, we see almost no gain in F-Measure performance or fooling/open set detection accuracy. The most likely reason for this lack of change in performance beyond $\alpha = 10$ is that lower ranked classes have very small FC8 activations and do not provide any significant change in OpenMax probability. The results for varying values of α are presented in Figs 3 and 4.

1.3. Distance Measures

We tried different distance measures to compute distances between Mean Activation Vectors and the Activation Vector of an incoming test image. We tried cosine distance, Euclidean distance and Euclidean-cosine distance. Cosine distance and Euclidean distances performed marginally worse compared to Euclidean-cosine distance. Cosine distance does not provide for a compact abating property and hence may not restrict open space for points that have a small degree of separation in terms of angle but are still far away in terms of Euclidean distance. Euclidean-cosine distance finds the closest points in a hyper-cone, thus restricting open space and finding closest points to Mean Activation Vector. Euclidean distance and Euclidean-cosine distance performed very similarly in terms of performance. In Fig 5 we show effects of different distances on over all performance. We see that OpenMax still performs better than SoftMax, and Euclidean-cosine distance perform the best of those tested.

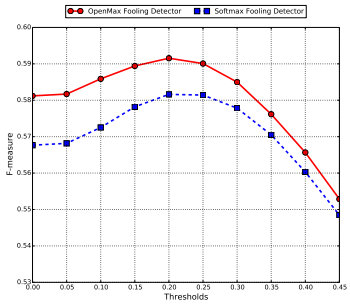
*The research was performed at University of Colorado at Colorado Springs. Abhijit Bendale is currently with Samsung Research America, Mountain View, CA



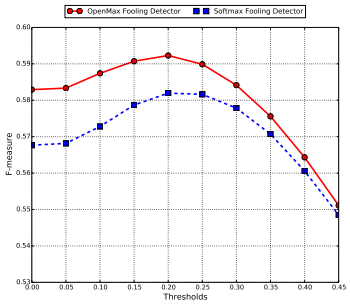
(a) Tail Size 10 (b) Tail Size 20 (optimal) (c) Tail Size 25

(d) Tail Size 30 (e) Tail Size 40 (f) Tail Size 50

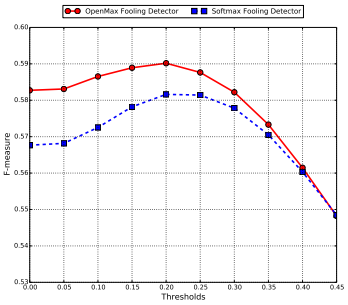
Figure 1: The graphs shows fooling detection accuracy and open set detection accuracy for varying tail sizes of EVT fitting. The graphs plot accuracy vs varying uncertainty threshold values, with different tails in each graph. We observe that OpenMax consistently performs better than SoftMax for varying tail sizes. However, while increasing tail size increases OpenMax rejections for open set and fooling, it also increases rejection for true images thereby reducing accuracy on validation set as well, see Fig 2. These type of accuracy plots are often problematic for open set testing which is why in Fig 2 we use F-measure to better balance rejection and true acceptance. In the main paper, tail size of 20 was used for all the experiments.



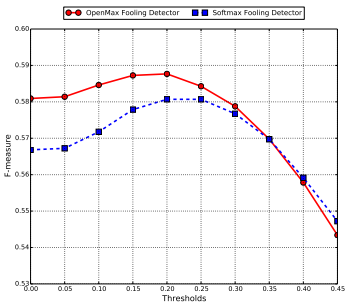
(a) Tail Size 10



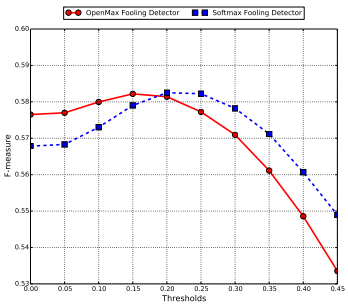
(b) Tail Size 20 (optimal)



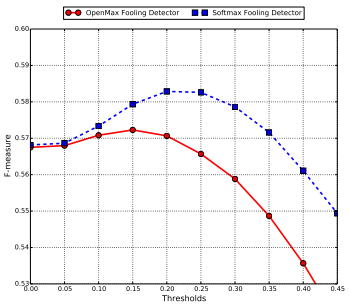
(c) Tail Size 25



(d) Tail Size 30

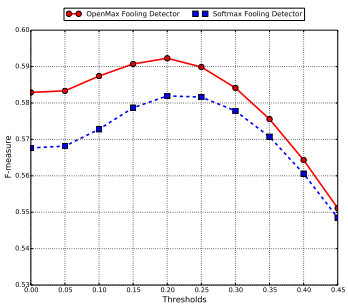


(e) Tail Size 40

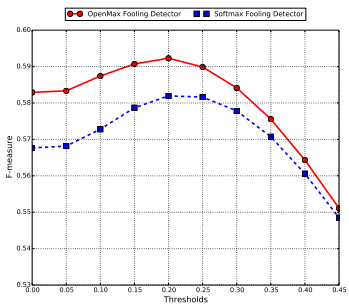


(f) Tail Size 50

Figure 2: The graphs shows F-Measure performance of OpenMax and Softmax with Open Set testing (using validation, fooling and open set images for testing). Each graph shows F-measure plotted against varying uncertainity threshold values. Tail size varies in different plots. OpenMax reaches its optimal performance at tail size 20. For tail sizes larger than 20, though OpenMax becomes good at rejecting images from fooling set and open set (Fig 1), it also rejects true images thus reducing accuracy on validation set. Hence, we choose tail size 20 for our experiments in main paper.

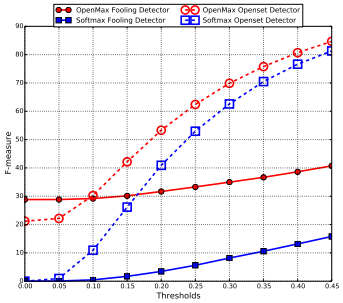


(a) Tail Size 20, Alpha Rank 5

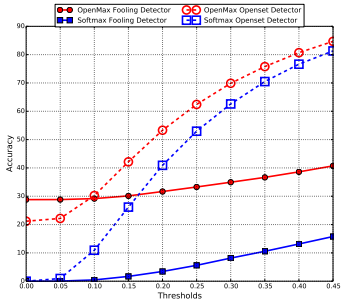


(b) Tail Size 20, Alpha Rank 10 (optimal)

Figure 3: The above figure shows performance of OpenMax and Softmax as number of top classes to be considered for recalibrating are changed. In our experiments, we found best performance when top 10 classes (i.e. $\alpha = 10$) were considered for recalibration.

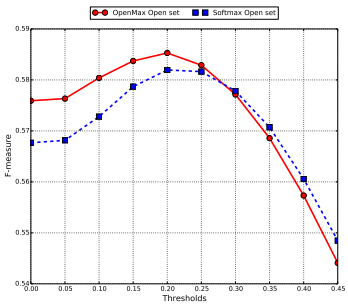


(a) Tail Size 20, Alpha Rank 5

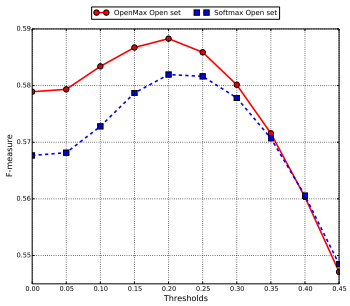


(b) Tail Size 20, Alpha Rank 10 (optimal)

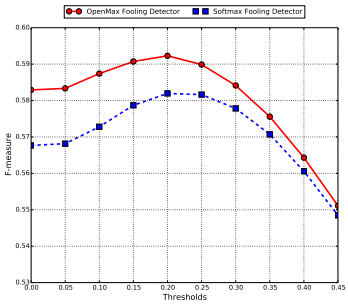
Figure 4: The figure shows fooling detection and open set detection accuracy for varying $alpha$ sizes. In our experiments, $alpha$ rank of 10 yielded best results. Increasing $alpha$ value beyond 10 did not result in any performance gains.



(a) Cosine Distance, Tail Size 20, Alpha Rank 10



(b) Euclidean Distance, Tail Size 20, Alpha Rank 10



(c) Euclidean-Cosine distance, Tail Size 20, Alpha Rank 10 (optimal) Note scale difference!

Figure 5: The above figure shows performance of OpenMax and Softmax for different types of distance measures. We found the performance trend to be similar, with euclidean-cosine distance performing best.

2. Qualitative Examples

It is often useful to look at qualitative examples of success and failure. Fig. 6 – Fig. 7 shows examples where OpenMax failed to detect open set examples. Some of these are from classes in ILSVRC 2010 that were close but not identical to classes in ILSVRC 2012. Other examples are objects from distinct ILSVRC 2010 classes that were visually very similar to a particular object class in ILSVRC 2012. Finally we show an example where OpenMax processed an ILSVRC 2012 [1] validation image but reduced its probability; thus Caffe with SoftMax provides the correct answer but OpenMax gets this example wrong.

3. Confusion Map of Mean Activation Vectors

Because detection/rejection of unknown classes depends on the distance mean activation vector (MAV) of the highest scoring FC8 classes. Note this is different from finding the distance from the input to the closest MAV. However, we still find that for unknown classes that are only fine-grained variants of known classes, the system will not likely reject them. Similarly for adversarial images, if an image is adversarially modified to a "nearyby" image, it is much less likely the OpenMax will reject/detect it. Thus it is useful to consider the confusion between existing classes.

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, pages 1–42, April 2015. 5

Figure 6: Left is an Image from ILSVRC 2010, "subway train", n04349306. OpenMax and Softmax both classfiy as n04335435. Instead of 'unknown". OpenMax predicts that the image on left belongs to category "n04335435:streetcar, tram, tramcar, trolley, trolley car" from ILSVRC 2012 with an output probability of 0.6391 (caffe probability 0.5225). Right is an example image from ILSVRC 2012, "streetcar, tram, tramcar, trolley, trolley car", n04335435 It is easy to see such mistakes are bound to happen since open set classes from ILSVRC 2010 may have have many related categories which have different names, but which are semantically or visually very similar. This is why fooling rejection is much stronger than open set rejection.

(a) A validation example of OpenMax failure. SoftMax labels it correctly as n03977966 (Police van/police wagon) with probability 0.6463, while OpenMax incorrect labels it n02701002 (Ambulance) with probability 0.4507.)

(b) Another validation failure example, where SoftMax classifies the image as n13037406 with probability 0.9991, while OpenMax rejects it as unknown. n13037406 is a gyromitra, which is genus of mushroom.

Figure 7: The above figure shows an examples of validation image misclassification by OpenMax algorithm.

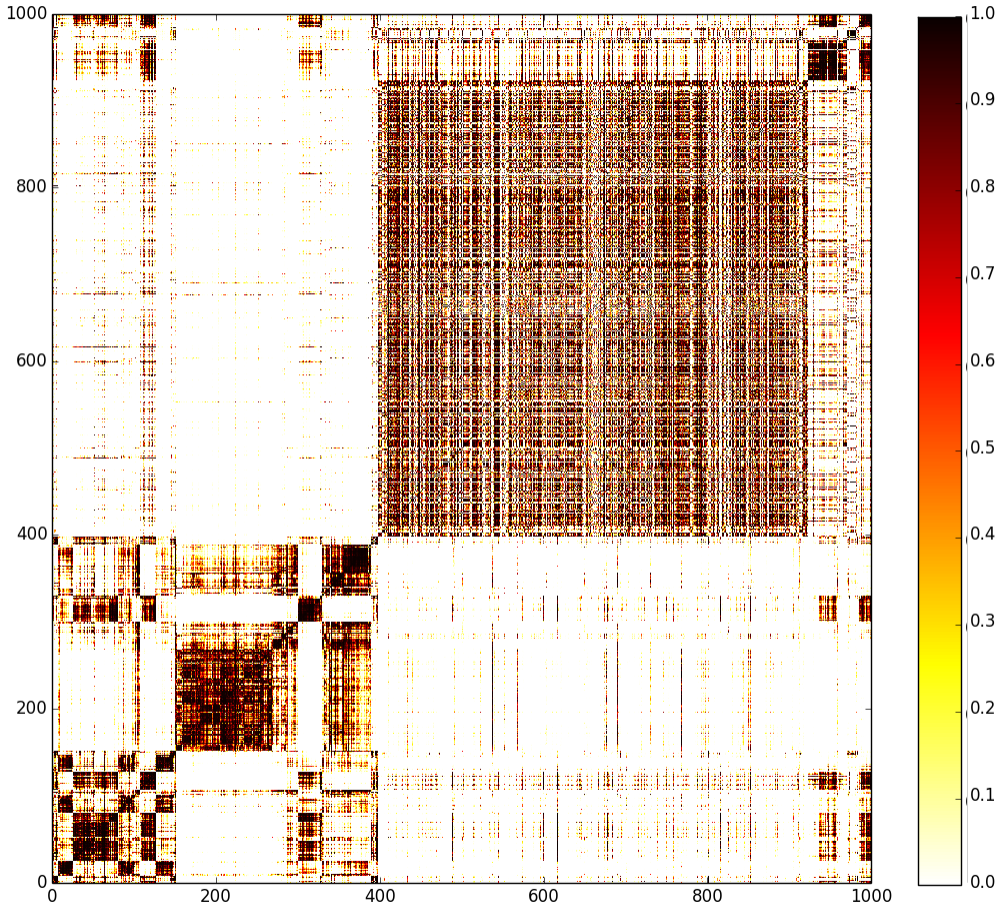


Figure 8: The above figure shows confusion matrix of distances between Mean Activation Vector (MAV) for each class in ILSVRC 2012 with MAV of every other class. Lower values of distances indicate that MAVs for respective classes are very close to each other, and higher values of distances indicate classes that are far apart. Majority of misclassifications for OpenMax happen in fine-grained categorization, which is to be expected.