An Effective Margin-based Product Recognition Solution to the CVPR 2020 RetailVision Challenge

Qi Xin, Zihan Ni, Qingze Wang, Rongqiao An, Yipeng Sun, Kun Yao
Beijing university of Posts and Telecommunications
xinsir@bupt.edu.cn, 598842055@qq.com

1. The proposed approach

In this competition, we use arc-margin as the loss function to increase the recognition performance, compared to softmax, arcface converges fast and has a remarkable gain. In our experiment, we found scale 35 margin 0.3 is the best setting, though there may be some subtle differences in different networks. We use the models pretrained in ImageNet, and then finetune on the Aliproduct train dataset without additional data.

We use multiple networks for this task, which include ResNet50, ResNet101, ResNet200-vd, Res2Net200-vd, InceptionV4, Xception71, Hrnet64, EfficientB4, Se-hrnet64, SENet154 and so on. The most efficient one is EfficientB4 and it achieves the best classification performance in both validation and test dataset. Besides, ResNet200-vd, Res2net200-vd, Hrnet64, Se-hrnet64 also achieve competitive performances. However, the rest perform not very well on test dataset, though they have a good classification accuracy on validation set.

For all the above networks, before the fc layer, we append a embedding layer which has a feature size of 512, the larger does not bring any gain and the smaller may drop the performance. We don?t adopt any drop out strategy in our experiment, we can?t find a way in which drop out can boost the performance. When it comes to the image process method, we only use the random crop as the data augmentation strategy and it turns to be effective. We set the scale range to [0.08, 1.0], we also try to add some other strategy such as rotate, color, contrast, brightness and so on, but we don?t find any obvious gain. Random crop is just good enough to realize the data augmentation. The image size of the input is resized to 224 * 224, and we train 120 epochs for ordinary networks and 200 epochs for those with -vd suffix.

There are two important tricks to increase the classification performance. The first one is data balance, which means we copy image multiple times for the label with few images, and we select only part of images for the label with too many images. Data balance can increase the classification performance for the labels with few images by a large margin. Exactly, it is not necessary to drop the images for the label with adequate images, in our experiment, it is not obvious that drop can bring gains. In general, more data makes the model more robust to noisy. The second is to enlarge the image size to 288*288 during test, the reason behind this is neural network tends to achieve good performance when the object scale is the same between training and testing. As we use random crop during training, the actual object size is a little larger compared to 224. In the testing stage, we resize the short side of the image to 288 and then center crop, so 288 is better than 224 when we test.

We can get multiple classification result for different models, so we need to merge the result to achieve the best performances. We adopt three types of merge strategies: 1 Classification merge. 2 Retrieval Merge 3 Correct Retrieval based on Classification. The most core notion in our merge experiment is the weight: We merge the result of different models based on their performance on the test dataset, the better model has higher weight. For the classification result of a single image, we assign 5 weight to the top5 label(top5 is enough because the others has very low probability) and sum the classification result of different models to get the final prediction. The retrieval merge is similar, we use the classification model as the backbone to extract the features of the train and test dataset; then we merge the result of top5 retrieval result with 5 weights for a single model; finally we merge the retrieval result of different models with different weight, achieving 7.06% in terms of error rates.