DISCOVER THE EFFECTIVE STRATEGY FOR FACE RECOGNITION MODEL COMPRESSION BY IMPROVED KNOWLEDGE DISTILLATION

Mengjiao Wang[1] *(menjiawan@gmail.com)* ***Rujie Liu[1]***

Narishige Abe[2] Hidetsugu Uchida[2] Tomoaki Matsunami[2] Shigefumi Yamada[2]

1. Fujitsu R&D Center Co.,LTD., Beijing, China
2. Fujitsu Laboratories Ltd., Kawasaki, Japan

Introduction

For the sake of better accuracy, the face recognition model is becoming larger and larger, which makes them difficult to be deployed on embedded systems. This work proposes an effective model compression method using knowledge distillation, where a fast student model is trained under the guidance of a complex teacher model.

Our main contribution can be summarized as three folds:

1. Comprehensive study on the loss functions and the student architectures: we conduct comprehensive analysis of different loss combination and student architectures to explore the most effective approach. To the best of our knowledge, this is the first report of these experiments for knowledge distillation based face recognition model compression.

2. Improvement of hint learning method by feature normalization: we unveiled the main reason of performance degradation after adding the hint layer (feature layer) is the difference between optimization goal of feature L2 loss and face recognition cosine similarity. This problem is alleviated by introducing feature normalization strategy.

3. Introducing the teacher weighting method for improving accuracy: we propose a teacher weighting strategy for accuracy improvement to address the issue when teacher provide wrong guidance. That is, when the teacher is less confident about a sample, the weight for the teacher's guidance will be lower so that the teacher will exert less influence on the student training.

Compression Strategy Using Knowledge Distillation

1. Strategy for Loss Function Combination

In this part, we explore comprehensive study for different loss function combination to find out the most effective strategy. The loss functions include: soft loss; hard loss; hint learning with feature.

Here, let's denote the final score output as Z, the soft label for teacher model T can be defined as $X_T^\tau = softmax(\frac{Z_T}{\tau})$ where τ is the temperature parameter. Similarly, the soft label for student network S is $X_S^\tau = softmax(\frac{Z_S}{\tau})$. The soft loss is the cross entropy between X_T^τ and X_S^τ:

$$L_{soft} = H(X_T^\tau, X_S^\tau)$$

The hard loss is the cross entropy between unsoften class probability X_S and ground truth y:

$$L_{hard} = H(X_S, y)$$

Here, $H(\cdot)$ represents for cross entropy.

For the hint learning, we used the feature layer as hint to train the student model. The hint loss is actually feature L2 loss:

$$L_{Feature} = \|F_S - F_T\|$$

F_S and F_T are the features from student and teacher.

2. Strategy for Student Model Architecture Design

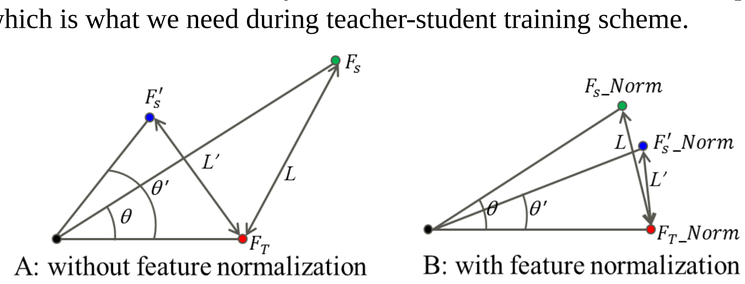
For the student model, the network architecture should be light and fast while be able to preserve accuracy. Therefore, the candidate architectures for student model should be efficient and superior comparing to the teacher model. In order to meet these requirements, we explored three types of candidate architectures, which include: efficient architecture specifically designed for fast models, i.e. SqueezeNet, MobileNet, and ShuffleNet; state-of-art architectures, i.e. Inception-ResNet and DenseNet; and tailored teacher architecture, i.e. shallower-wider and thinner-deeper, are explored for the tailored teacher architecture.

Improved Knowledge Distillation

1. Improved hint learning by feature normalization

In our evaluation, it is found that adding the feature loss to the total loss can cause degradation to the result. The main reason is that the most commonly used cosine similarity for face recognition and feature L2 loss have different optimization direction, as shown in figure 1. In figure 1.a, let F_T be the feature generated by teacher model. F_S is the student's feature. The L2 loss corresponds to the distance between F_S and F_T which is denoted as L in figure 1.a. The angle θ between vector F_S and F_T determines the cosine similarity $cos\theta$. The goal of optimization is to reduce the L2 distance while increase the cosine similarity. However, from figure 1.a, we can see that after optimization with respect to L2 loss, the feature vector moves from F_S to F_S', which decrease the L2 distance from L to L'. However, the cosine similarity also decreases from $cos\theta$ to $cos\theta'$. Therefore, we can draw the conclusion that L2 loss may decrease the cosine similarity. This will cause degradation of the face recognition performance, where cosine similarity metric is commonly used.

However, after feature normalization, as illustrated in figure 1.b, the optimization of feature L2 loss becomes consistent with cosine similarity. This means that when the L2 loss drops, the cosine similarity will increase, which is what we need during teacher-student training scheme.



A: without feature normalization B: with feature normalization

Figure 1. Illustration of relationship between cosine similarity and feature L2 loss without and with feature normalization.

2 Teacher Weighting Method

In the original idea of knowledge distillation, the student model learns the knowledge from the teacher model regardless of its correctness. This issue can be solved by weighting the teacher's guidance. When the teacher is less confident about this sample, the teacher's guidance will have a smaller weight, and vice versa. The proposed idea can be formulated as:

$$\mu = 1 - \alpha \cdot w$$

where, $w = H(X_T, y)/(H(X_S, y) + k)$

$$\alpha = \begin{cases} k/MIN, if\ w > 1 \\ 0, \quad if\ w \le 1 \end{cases}$$

Here, μ is used to weight teacher's guidance as shown:

$$L_{Total} = \lambda_{Hard} L_{Hard} + \mu(\lambda_{Soft} L_{Soft} + \lambda_{Feature} L_{Feature})$$

Result

In our implementation, a 27-layer ResNet is used as the teacher model. The CASIA-WebFace dataset is used as the training dataset. The performance is tested on LFW and our own dataset – FRDCMobile. The run time is tested on an Intel Core 2.5GHz CPU to simulate the environment of embedded device.

1. Evaluation of Different Loss Combination Strategies

Here, a 5-layer convolution network with 1.5x teacher model channel number is adopted as the student model to evaluate the different combination of loss functions. The run time and complexity comparison between the 5-layer convolution model and teacher model are shown in table 1, from which we can see the 5-layer model can achieve more than 3x acceleration of teacher model.

Model	RunTime	Model Size	Million Mult-Adds
Student (5-layer Conv)	35ms	27M	570
Teacher	132ms	126M	2049

Table 1 Run time and complexity comparison between teacher and student model

The different loss function combination strategy include:

(1) Hard: Only using hard loss L_{Hard}; ***(2) Feature:*** Only using hint learning $L_{Feature}$; ***(3) Hard+Soft:*** Combination of L_{Hard} and L_{Soft}; ***(4) Hard+Feature:*** Combination of L_{Hard} and $L_{Feature}$; ***(5) Hard+Soft+Feature:*** Combine L_{Hard}, L_{Soft} and $L_{Feature}$

From table 2, we can see that other combination strategies all outperform using hard loss alone, which proves that it's beneficial to use the teacher model's guidance. Directly adding the feature loss to the total loss function can cause some degradation to the accuracy. To solve this problem, we can normalize the feature. It was seen that '***Hard+Soft+FeatureNorm***' can improve the performance and in fact yields the best result among all these combination strategies.

Model	ACC on LFW	TPR@FAR=0.1% on FRDCMobile		
		Frontal	Pose	Lighting
Hard	95.82%±0.75%	52.64%	40.34%	45.00%
Feature	97.08%±0.64%	54.36%	42.23%	48.66%
Hard+Soft	97.10%±1.02%	67.86%	47.72%	59.23%
Hard+Feature	97.03%±0.85%	57.70%	41.23%	50.96%
Hard+Soft+Feature	96.90%±0.97%	62.80%	45.26%	57.61%
Hard+Soft+ FeatureNorm	97.15%±1.23%	69.49%	47.84%	60.40%
Teacher	97.73%±0.62%	71.91%	50.90%	60.60%

Table 2 Result of different loss combination

2. Performance of Different Architectures

The performance of different architectures are shown in table 3. The thinner-deeper architectures, which achieve the best performance by deploying the same network structure with teacher model with increased depth and reduced channels. From our analysis, we find that besides deep and convolutional, the student model also need to have exact same architecture with teacher model.

We also evaluate three thinner-deeper architectures to analyze the trade-off between depth and width. From this experiment, we empirically find out that 1.5x of teacher's depth (43 layers) yields the best result, and the width can be chosen according to the computation limit.

	Model	ACC on LFW	TPR@FAR=0.1% on FRDCMobile		
			Frontal	Pose	Light
Baseline	5-layer Conv	97.15±1.23%	69.49%	47.84%	60.40%
Efficient Architecture	SqueezeNet	94.83±0.93%	54.06%	34.87%	43.50%
	MobileNet	94.42±1.03%	50.54%	30.42%	42.53%
	ShuffleNet	95.16±1.05%	54.60%	38.30%	50.80%
State-of-art	DenseNet	97.18±0.81%	71.05%	50.99%	60.71%
	Inception-Resnet	97.20±1.26%	73.60%	52.21%	61.87%
Thinner-Deeper	27-layer Thinner-Deeper	97.18±0.65%	74.96%	53.77%	61.88%
	43-layer Thinner-Deeper	97.48±0.81%	76.00%	54.59%	65.28%
	53-layer Thinner-Deeper	97.27±0.76%	77.18%	53.88%	56.79%
Teacher	Teacher	97.73±0.62%	71.90%	50.90%	60.60%

Table 3 Results for Different Architecture

3. Performance of teacher weighing method

The architecture used is the 43-layer thinner-deeper model, and the combination of 'Hard+Soft+FeatureNorm' is chosen as the loss function. With the weighting strategy, we can finally surpass the teacher model.

Model	ACC on LFW	TRP@FAR=0.1% on FRDCMobile		
		Frontal	Pose	Light
WO Teacher Weighting	97.48%±0.81%	76.00%	54.59%	65.28%
W Teacher Weighting	97.85%±0.60%	76.60%	57.21%	70.87%
Teacher	97.73%±0.62%	71.90%	50.90%	60.60%

Conclusion

In this paper, we propose an effective method for the compression of face recognition model by knowledge distillation:

1. ***Combine hard loss, soft loss and normalized feature Euclidean loss as an effective strategy to guide the training of student model.***
2. ***The thinner-deeper architecture with 1.5X teacher's depth can achieve best performance.***
3. ***The teacher weighting method can further improve the performance while maintain 3x acceleration.***

Acknowledge

If you have further questions, please feel free to reach me at menjiawan@gmail.com