CONFNET: PREDICT WITH CONFIDENCE

Sheng Wan[1]*,Tung-Yu Wu*[2]*, Wing H. Wong*[2,3] *and Chen-Yi Lee*[1]

[1]Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan
[2]Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, USA
[3]Department of Statistics, Stanford University, Stanford, CA, USA

ABSTRACT

In this paper, we propose Confidence Network (ConfNet) which not only makes predictions on input images but also generates a confidence score that estimates the probability of correctness of each prediction. Furthermore, Confidence Loss is proposed to make ConfNet automatically learn confidence scores in the training phase. The experiments on two public datasets show that the confidence scores generated by ConfNet are highly correlated with the model accuracy and outperforms two related methods. When stacking two ConfNets in a cascade structure, **3.8x** computational cost can be saved compared to the single state-of-the-art model with only **0.1%** increase of error rate.

Index Terms— Convolutional Neural Network, Deep Learning, Confidence score, Model Cascade

1. INTRODUCTION

Recently, Convolutional Neural Network (CNN) has been widely applied to real-world tasks, such as speech recognition [1], face detection [2], and audio classification [3]. Despite its high accuracy, each CNN model tends to have difficulty in predicting certain classes due to the common training issues including unbalanced training dataset and limited model capacity. Therefore, estimating how much confidence a CNN model has in its prediction is a crucial problem. In this paper, we take one step further to ask, *"Can CNN models explicitly estimate the probability of correctness of each classification prediction?"* In other words, we aim to design a network which gives high confidence scores when it has strong faith in its predictions and provides low confidence scores when facing unrecognized input samples.

This topic has been implicitly discussed by many recent works in different computer vision tasks. Redmon et al. [4] [5] propose an end-to-end object detection network which predicts a number of bounding boxes and the probability of each bounding box containing an object. This probability can be considered as the confidence score for the bounding box.

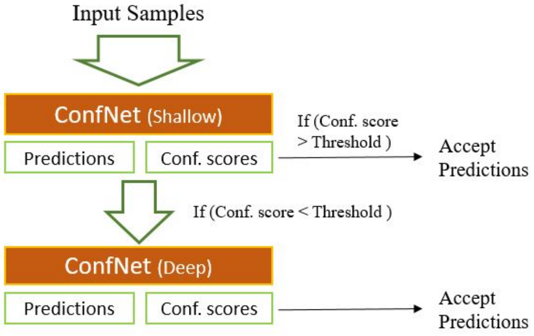

Fig. 1: The illustration of stacked ConfNets.

In face detection tasks, some works [2] [6] use the class probability generated by CNN models as the confidence score to quickly reject noises and only the predictions with high class probabilities are preserved. In [7], Wang et al. utilize the entropy of class probabilities to classify the hard samples into an additional *"I don't know"* class.

In this paper, we propose an end-to-end CNN framework for image classification, called Confidence Network (ConfNet). Given a test image, ConfNet simultaneously makes the classification prediction and provides a confidence score that estimates the correctness of the prediction. In comparison with traditional CNN frameworks, additional layers are added after the Softmax layer to generate the confidence score. Since the training dataset itself does not contain training labels for confidence scores, a novel loss function, called Confidence Loss, is proposed which allows ConfNet to learn its confidence score automatically in the training phase by leveraging the negative correlation between the model accuracy and the cross entropy loss.

We evaluate our approach on two public image classification datasets, Cifar10 [8] and ImageNet [9]. The results show that the confidence scores predicted by our CNN models are highly positively correlated with the model accuracy and outperform two related methods. In addition, we stack two ConfNets to form a cascade structure as shown in Fig. 1. All input samples are first classified by a shallow model. Then, the confidence score of each prediction is employed as a threshold to identify hard samples. These samples are sent to a deeper CNN model for more accurate prediction. Us-

Work supported by MOST of Taiwan under Grant 105-2218-E-009-001 and NSF of USA under Grant DMS-1407557 and DMS-1721550.

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

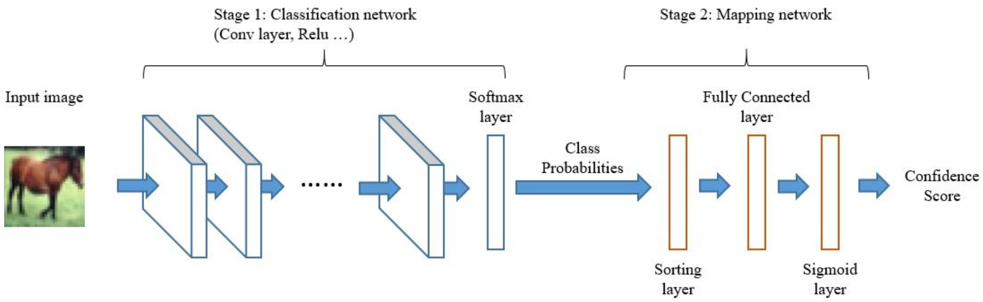

Fig. 2: The structure of Confidence Network.

ing this design, the stacked framework can significantly save computational cost and achieve high accuracy.

2. PROPOSED METHOD

In this section we first discuss the limitations of two previous methods and introduce ConfNet and its core layers. Next, Confidence Loss is proposed to allow ConfNet to learn confidence scores without additional labels in the training datasets.

2.1. Confidence of Prediction

Most modern deep learning models for classification are trained to predict $P_{model}(y|x)$, the class probabilities conditioned on input sample x, by optimizing the cross entropy loss, which is given as

$$J_{Entropy}(x,y) = -\sum_{k=1}^{K} y^k \cdot logP_{model}(y^k|x) \qquad (1)$$

where K denotes the number of classes and y^k is the label of kth class given by the training dataset. In order to minimize the cross entropy loss, the distribution of class probabilities predicted by CNN model is encouraged to present as a one hot vector when the prediction is correct [10]. By contrast, when a CNN model is not certain about prediction, the smooth distribution of class probabilities is generally observed. This suggests a conceptual connection between the distribution of class probabilities and the model accuracy.

Many recent works try to link class probabilities distribution to model accuracy. Li et al. [2] and Ranjan et al. [6] utilize the maximum of class probabilities as the confidence score to classify hard samples. Empirically, we notice that the distribution of class probabilities is able to provide more information than the maximum probabilities alone. The differences among class probabilities should also be considered to estimate the prediction correctness. Wang et al. [7] use the entropy of class probabilities as the confidence score. However, the range of entropy varies with respect to the number of classes. This cause difficulty in building a mapping from entropies to confidence scores. Inspired by these works, we propose an end-to-end network to estimate confidence scores for classification by leveraging the great representation power of neural networks.

2.2. Confidence Network

In this subsection we describe the structure of ConfNet and its core layers. As shown in Fig. 2, ConfNet consists of two stages. The first stage is a general classification network which takes images as input and generates the class probabilities. In general, the class with maximum probability is selected as the model prediction for the input sample. It is worth noting that the framework of the classification network can be replaced with different CNN frameworks that match the resource restrictions (latency, accuracy) such as Alexnet [11], VGG-16 [12], and Resnet [13].

The second stage of ConfNet is a mapping network that maps the class probabilities generated in the previous stage to a single confidence score. The mapping network is composed of a Sorting layer, a Fully Connected layer and a Sigmoid layer. The Sorting layer takes class probabilities as input and sorts the probabilities in descending order. The number of output neurons of the Sorting layer is the same as the number of classes. This layer removes the class information from the class probabilities and the following layers are hence able to focus on mapping the distribution of class probabilities to the model accuracy. After the Sorting layer, a Fully Connected layer with one output neuron is employed to derive the mapping. Weights in Fully Connected layer represent the contribution of each probability to the confidence score. Since the confidence score is expected to estimate the probability of prediction correctness, a Sigmoid layer is adopted as the last layer of the mapping network to limit the output between 0 and 1.

2.3. Confidence Loss

The probability of correctness of each prediction depends on the difficulty of each input sample and the performance of CNN model. This uncertainty results in the lack of ground truth labels for the confidence score. To solve this problem, we propose Confidence Loss which leverages the negative correlation between the model accuracy and the cross entropy loss. Confidence score of each prediction can be estimated by observing the cross entropy loss. Confidence Loss on a mini-batch $\{(x_i, y_i)\}_{i=1}^{N}$ is given as:

$$J(\alpha) = \frac{1}{N}\sum_{i=1}^{N} CS_i \cdot J_{Entropy}(x_i, y_i) - \alpha log(CS_i) \quad (2)$$

where N denotes the batch size, CS_i denotes the confidence score of ith prediction generated by ConfNet in this batch, and α is a hyper parameter to balance the regularization strength.

The first term of the objective function is the cross entropy loss scaled by the confidence score and the second term is a regularization term. Optimizing this objective function leads to two cases. (i) When the input sample can be easily classified and the expected cross entropy loss is low, a higher confidence score is preferred to reduce the regularization term. (ii) When a hard sample results in a high cross entropy loss, lower confidence score can effectively reduce the first term of the objective function. In the training phase, the mapping network of ConfNet tries to estimate the cross-entropy loss first and generates the corresponding confidence score to minimize the overall loss. This objective function allows ConfNet to learn confidence scores automatically without additional labels from the training dataset.

Our proposed approach is different to the *"I don't know'* (IDK) methods [7] [14] [15] in many aspects. First, the IDK methods classify the ambiguous samples into an additional IDK class. The classification results are within either the original classes or the IDK class. In ConfNet, we still classify samples into the original classes and further provide a confidence score for estimating the reliability of this prediction. Moreover, the IDK methods focus on identifying hard samples whereas ConfNet aims to estimate the expected accuracy for each prediction.

3. EXPERIMENTS

We evaluate our ConfNet on two public image classification datasets, Cifar10 [8] and ImageNet [9], with a number of popular CNN frameworks including NIN [16] and Resnet series [13]. All experiments are conducted in *Caffe* [17]. For model training, we first transfer the weights from the pre-trained classification model, which is downloaded from *Caffe Model Zoo*, to the classification network of ConfNet and randomly initialize the weights of the mapping network. Then, we fine-tune ConfNet to optimize the confidence loss with hyper parameter $\alpha = 0.5$ until the confidence loss converges. All results reported in following experiments are evaluated on the validation set of the two classification datasets.

3.1. Cifar10

We first show the relation between the confidence score and the model accuracy. We record the precision and the recall on the testing samples when their corresponding confidence scores are higher than threshold. As shown Fig. 3, the confidence score generated by ConfNet highly correlates to the model accuracy in both frameworks with higher confidence score corresponding to more accurate predictions. To our surprise, ConfNet achieves over 95% precision with nearly 95% recall in both frameworks. It means that most of predictions is accurate and we can identify those predictions by using confidence scores.

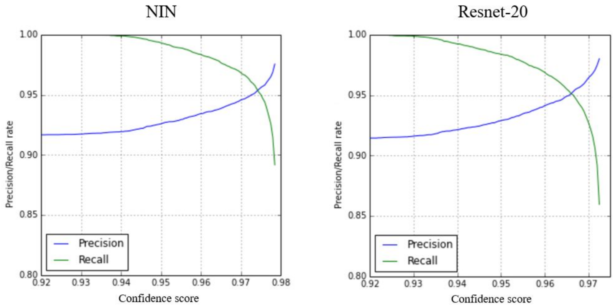

Fig. 3: Precision and Recall under different thresholds of the confidence score. The confidence score highly correlates to model accuracy in both frameworks.

We further define an evaluation metric, called **Mean Effective Confidence (MEC)**, for comparative evaluation as follow:

$$MEC = \frac{1}{n}\sum_{i=1}^{n} C_i * Normalize(CS_i), \quad (3)$$

$$C_i = \begin{cases} 1, & i\text{th prediction is correct.} \\ -1, & i\text{th prediction is wrong.} \end{cases} \quad (4)$$

where n is the number of testing samples, C_i denotes the correctness of ith prediction and $Normalize(.)$ denotes the linear normalization function which sets the confidence score of each method between 0 to 1 for fair comparison. Note that since there are 10 classes in Cifar10, the maximum of class probabilities ranges from 0.1 to 1 and the entropy of class probabilities ranges from $log_2 10$ to 0. This metric evaluates how much the confidence score relates to the correctness of prediction. Correct predictions with high confidence scores and wrong predictions with low confidence scores lead to large **MEC**.

Table 1: Performance comparison of MEC with two related methods on Cifar10 dataset

Methods	MEC	
	NIN	Resnet-20
Maximum of class probabilities	80.14%	80.74%
Entropy of class probabilities	81.75%	82.50%
ConfNet	**83.48%**	**83.33%**

Table 2: Performance comparison with state-of-the-art single models on Cifar10 dataset

Single model / Cascade structure	Threshold of Confidence score	Error rate	Average flops per image ($\times 10^7$)
NIN [16]	-	9.34%	22
Resnet-20 [13]	-	8.75%	4.1
Resnet-44 [13]	-	7.17%	9.7
Resnet-56 [13]	-	6.97%	12.5
Resnet-110 [13]	-	6.43%	24.3
DenseNet(L=100,k=12) [18]	-	5.77%	146.2
Stacked ConfNets (Resnet-20 + Resnet-110)	0.960	***7.22%***	***5.3***
Stacked ConfNets (Resnet-20 + Resnet-110)	0.965	***6.93%***	***5.7***
Stacked ConfNets (Resnet-20 + Resnet-110)	0.970	***6.53%***	***6.6***

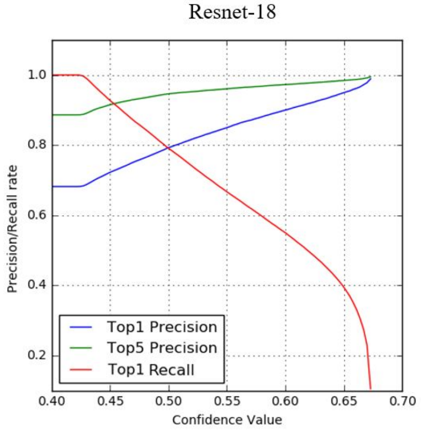

Fig. 4: Precision and Recall under different thresholds of the confidence score. The confidence score highly correlates to top-1 accuracy and top-5 accuracy.

As shown in table 1, our method outperforms two related methods. This result demonstrates that the confidence score generated by ConfNet can predict the probability of prediction correctness better.

In addition, we stack two ConfNets to form a Cascade structure, which is widely used in computer vision tasks [19] [2] [6] [7] to accelerate the classification. As shown in Fig. 1, we cascade two ConfNets with different classification frameworks. One image will be first classified by the first ConfNet with a shallow classification framework. If the confidence score is lower than a pre-defined threshold, this image would be considered as a hard sample and fed into the deeper ConfNet for more accurate prediction.

We evaluate the performance of stacked ConfNets on Cifar10. The error rate and the average flops per image are shown in table 2. By varying the threshold of confidence score, it is easy to find different trade-offs between testing accuracy and computational cost. Compared with the single Resnet-110 model, our stacked ConfNets which consists of Resnet-20 model and Resnet-110 model can save **3.8** times computational cost with only **0.1%** increase of error rate.

3.2. ImageNet

To show the generalization of Confidence Network, we also evaluate our proposed method on ImageNet [9]. We select pre-trained Resnet-18 as the classification framework of ConfNet and fine-tune until confidence loss converges. Fig 4 shows that the confidence scores generated by ConfNet are highly correlated with the top-1 accuracy and top-5 accuracy in this large scale dataset.

4. CONCLUSION

In this paper, we propose Confidence Network (ConfNet) which not only generates accurate predictions, but also provides a confidence score to estimate the probability of correctness of each prediction. With this confidence score, we can easily estimate the reliability of model prediction. For model training, we design Confidence Loss to allow ConfNet to learn confidence scores automatically without extra training labels.

We evaluate our proposed network on two popular datasets with different CNN frameworks. The results prove that confidence scores generated by ConfNet are highly correlated with model accuracy and our approach outperforms two related methods. Moreover, by stacking two ConfNets, we pass the images first through a shallow light model and further process the images which are hard to classify with a deeper model. The cascade structure for classification can significantly save the computation cost compared to the single state-of-the-art model.

5. REFERENCES

[1] Dimitri Palaz, Mathew Magimai Doss, and Ronan Collobert, “Convolutional neural networks-based contin-

uous speech recognition using raw speech signal," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4295–4299.

[2] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5325–5334.

[3] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al., "Cnn architectures for large-scale audio classification," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 131–135.

[4] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[5] Joseph Redmon and Ali Farhadi, "Yolo9000: better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.

[6] Rajeev Ranjan, Vishal M Patel, and Rama Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *arXiv preprint arXiv:1603.01249*, 2016.

[7] Xin Wang, Yujia Luo, Daniel Crankshaw, Alexey Tumanov, and Joseph E Gonzalez, "Idk cascades: Fast deep learning by learning not to overthink," *arXiv preprint arXiv:1706.00885v2*, 2017.

[8] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," 2009.

[9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[10] Ziyong Feng, Zenghui Sun, and Lianwen Jin, "Learning deep neural network using max-margin minimum classification error," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 2677–2681.

[11] 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.

[12] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[13] 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.

[14] Fereshte Khani, Martin Rinard, and Percy Liang, "Unanimous prediction for 100% precision with application to learning semantic mappings," *arXiv preprint arXiv:1606.06368*, 2016.

[15] Thomas P Trappenberg and Andrew D Back, "A classification scheme for applications with ambiguous data," in *Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on*. IEEE, 2000, vol. 6, pp. 296–301.

[16] Min Lin, Qiang Chen, and Shuicheng Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[17] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[18] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten, "Densely connected convolutional networks," *arXiv preprint arXiv:1608.06993*, 2016.

[19] Paul Viola and Michael J Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.