Machine Learning and Artificial Intelligence
A.J. Tallón-Ballesteros and C. Chen (Eds.)

© 2020 The authors and IOS Press.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).
doi:10.3233/FAIA200778

A Simple Ensemble Learning Knowledge Distillation

Himel DAS GUPTA [a1], Kun ZHANG [b] Victor S. SHENG [c]

[a] *Department of Computer Science, Texas Tech University, Lubbock, Texas, 79409, USA*
[b] *Department of Computer Science, Bioinformatics facility of Xavier RCMI Center of Cancer Research, Xavier University of Louisiana, New Orleans, Louisiana, 70125, USA*
[c] *Department of Computer Science, Texas Tech University, Lubbock, Texas, 79409, USA*

Abstract. Deep neural network (DNN) has shown significant improvement in learning and generalizing different machine learning tasks over the years. But it comes with an expense of heavy computational power and memory requirements. We can see that machine learning applications are even running in portable devices like mobiles and embedded systems nowadays, which generally have limited resources regarding computational power and memory and thus can only run small machine learning models. However, smaller networks usually do not perform very well. In this paper, we have implemented a simple ensemble learning based knowledge distillation network to improve the accuracy of such small models. Our experimental results prove that the performance enhancement of smaller models can be achieved through distilling knowledge from a combination of small models rather than using a cumbersome model for the knowledge transfer. Besides, the ensemble knowledge distillation network is simpler, time-efficient, and easy to implement.

Keywords. ensemble, bagging, knowledge distillation

1. Introduction

The vast spectrum of artificial intelligence has imposed significant impacts over the years in different applications. We have witnessed the state of art results achieved in the field of computer vision [1], natural language processing [2], speech recognition [3], and autonomous vehicle [4]. The performance of machine learning models in such applications much depends on the architecture of the model even if adequate training data is available for training the networks. Integrating more layers and more parameters can certainly improve the accuracy of a machine learning model. However, it comes with the cost of requiring powerful computing systems. Small devices like mobile and embedded systems can barely provide such computational power. Besides, it takes a lot of time to complete these tasks. In order to deploy well in practical fields, such cumbersome architectures become hectic to run. The idea of knowledge distillation (KD) has brought

[1] Corresponding Author: Himel Das Gupta, Department of Computer Science, Texas Tech University, Texas, USA; E-mail: Himel.Das@ttu.edu

prominent results in such situations where a small model is trained to mimic like a large model. The fundamental idea of knowledge distillation is to transfer the generalization ability of a complex teacher model to a simpler student model. A recent popular approach is to transfer the output class probabilities (from 0 to 1) of a big neural network to a student model. The student model tries to learn from these probabilities, instead of hard class labels. Thus, the student model can imitate the generalization of the teacher model on unseen data.

The classification accuracy of Neural networks can be improved with the help of ensemble learning [5] like bagging. With bagging, we can learn multiple independent models together and average their predictions to obtain a final prediction with a lower variance. There exist different ways to improve the performance of ensemble learning like in [6] where authors have demonstrated the relation in multistrategy ensemble learning between test error reduction and the generation of diversity in ensemble membership. However, in this paper, we propose a simple ensemble knowledge distillation method (called EKD) to improve the performance of a single model, instead of the ensemble model. Specifically, it first performs knowledge distillation from ensemble learned models, and then use the knowledge distilled to train a single model. Note that this approach is also different from traditional knowledge distillation (KD), since traditional knowledge distillation usually uses a comparatively bigger model(deep learning model) to distill the knowledge from the bigger model for training a simple single model. Our experimental comparisons show that knowledge transfer from a cluster of small models can be as effective as transferring knowledge from a cumbersome model.

Previous work like [7] has achieved a significant improvement in terms of accuracy using ensemble knowledge distillation. Now, there are some differences between their work and ours. First of all, their ensemble teacher model consists of multiple convolutional neural networks (CNN) with different architectures whereas we have used same CNN architecture for our ensemble teacher model. Secondly, their student model is a composition of multiple branches where each branch is represented as separate neural network architecture. On the other hand, our student model is a simple single neural network architecture. And most importantly, their framework they have connected these two networks (ensemble teacher network + compact student network) together whereas our teacher, student model are separate. To elaborate these, we have to focus on their distillation process where the main difference is lying. To start with, though their teacher model is trained at first separately, it gets fine-tuned simultaneously and collaboratively with the student model. And the distillation process is done by the objective function which is summation of ”teacher model loss + student model loss + distillation loss” as the whole network is connected. In our model, training phase of teacher and student model is completely separate where we trained the ensemble teacher model first, calculate the softmax probabilities and then use these probabilities as constraint while training the student model. And the loss function for our student model is consists of only student model loss and distillation loss.

In a nutshell, We will discuss related work in Section 2. We are going to demonstrate the technical aspects of knowledge distillation and our EKD in Section 3. In Section 4, we will define our experiment setup and all the preprocessing steps. Analysis and empirical comparisons of our experiments will be discussed in Section 5. The paper ends with conclusions and future work opportunities in Section 6.

2. Related Works

The term ”knowledge distillation” or ”teacher-student model” has been first proposed by [8] which has been further improved by [9]. In [8] a novel method called ”Model Compression” was first introduced where a complex, large network can be compressed into a smaller model. This paper has built the foundation of knowledge distillation. Additional improvement has been done by [10] where the authors have considered not only the output layers but also the intermediate layers for transferring class probabilities to the student model using L2 loss. Knowledge distillation has been used in reinforcement learning also [11]. In [12] authors have represented the distilled transfer knowledge as FSP Matrix which is generated by two layers. Instead of compressing model, distillation can also be achieved by training parameters of a student model identically to their teacher which is defined as Born-again networks in [13]. Adversarial based learning strategy has been used in [14] to distill the diverse knowledge from a compressed large trained ensemble networks. In [15] cluster of different architectural recurrent networks has been used as ensemble distillation learning to improving accuracy in speech recognition. Ensemble based knowledge distillation can have superior performance improvement over the traditional knowledge distillation shown in [16] where it uses data augmentation. What it does is, creates multiple copies of data with respect to the soft output targets from various teachers model. As mentioned in the introduction, in [7] ensemble learning has been improvised by using multiple branches of student model where the branches are trained by a teacher model using it’s ground truth labels and information. Improvements in task like binary classification has been done using ensemble learning [17] where bagging technique is applied to an ensemble of pairs of neural networks. These networks have been used to predict degree of truth membership, indeterminacy membership, and false membership values in the interval neutrosophic sets. Ensemble learning can be very handy in improving classification performance for Deep Belief Network(DBN) too. In [18] a new mechanism called Pseudo Boost Deep Belief Network(PB-DBN) has been proposed in this regard where top layers are boosted while lower layers of the base classifiers share weights for feature extraction. A novel method called Generalized Regression Neural Network (GEFTS–GRNN) has been proposed in [19] where the authors combined a single GRNN from multiple base level GRNN to produce the final output. In our paper, we have shown a much easier way to implement a ensemble knowledge distillation network.

3. Background and Implemented Model

3.1. knowledge distillation (KD)

The architecture of a neural network is organized such way that we can get the probabilities of the classified classes by imposing ”softmax” activation function on it. The general equation of such output layer is like this $y_i = exp(x_i/T)/\sum_j exp(x_i/T)$ in [8]. Where x_i is the logit, j is the number of classes, and y_i is the class probability. Here T is denoted as temperature value which is usually 1 [8]. The higher value in temperature signifies the softer probability distribution of the classes. The main idea of distillation is to transfer these probabilities as knowledge from the cumbersome model to the smaller model. It can be achieved by making the soften probabilities of teacher model as target for the small model. For understanding, often the cumbersome model is defined as ”teacher” model and the smaller model is defined as ”student” to express the idea of student learning knowledge from the teacher.

In [8] the authors used weighted average of two functions to train the student model to produce correct labels in addition to the teacher's soft labels. The first objective function is the cross entropy with the correct labels whereas cross entropy with the soft labels is considered as second objective function. The distillation process is propagated by the custom loss function like in Eq. (1),

$$\begin{aligned} \mathscr{L}_{student} &= \alpha \mathscr{L}_{CL} + (1-\alpha)\mathscr{L}_{KD} \\ \mathscr{L}_{KD} &= T^2 KL(y_s, y_t) \end{aligned} \tag{1}$$

Here $\mathscr{L}_{KL}$ is the built in KL DIVERGENCE loss, $\mathscr{L}_{CL}$ is the normal cross-entropy loss, T is the temperature value and y_s, y_t are the targets softened for the student model. The hyperparameter α emphasizes between weighted average of the two loss functions.

3.2. Ensemble knowledge distillation (EKD)

To demonstrate our EKD framework, we have combined multiple small models as a single teacher model. That means, our teacher ensemble model is a composition of small models rather than one single cumbersome model like in a traditional knowledge distillation framework. **Figure 1** shows our combined network of ensemble knowledge distillation (EKD). Where $y_{t_1}...y_{t_n}$ symbolizes the ”softmax” predictions of the n number of ensemble models. In a traditional knowledge framework, the transferring of generalization ability of a cumbersome model to a small models is done by imposing the class probabilities of the cumbersome model as ”soft targets” for the small model during the training phase. In our implemented method, instead of taking class probabilities of a single cumbersome model, we took the arithmetic mean of the class probabilities produced by each model of the ensemble model. So, $(y_{t_1}...y_{t_n})/n$ represents the probabilities that has been used as ”soft targets” for the small student model in our implemented model.

4. Experiment Setup

4.1. Architecture of Used Networks

To conduct our experiments, we used four different neural networks which we are going to denote as **LargeCNNnet** (6 layer CNN), **SmallCIFAR10net** (2 layer CNN), **SmallCIFAR100net** (3 layer CNN), and **SmallMnistNet** (2 layer MLP). The application of these network in our experiments is summarized in **Table 1**. The motivation behind the architecture of these models was to show the implementation of the actual ensemble knowledge distillation method rather than getting the highest accuracy. We constructed the architecture of these models in such way that it doesn't take too much computation power and memory to run but significant enough to show the differences between a traditional knowledge distillation and ensemble knowledge distillation framework.

Table 1. Used neural network architectures.

Dataset	Teacher Model	Student Model
CIFAR10 (KD)	LargeCNNnet	SmallCIFAR10Net
CIFAR10 (EKD)	n*SmallCIFAR10Net	SmallCIFAR10Net
CIFAR100 (KD)	LargeCNNnet	SmallCIFAR100Net
CIFAR100 (EKD)	n*SmallCIFAR100Net	SmallCIFAR100Net
MNIST(KD)	LargeCNNnet	SmallMnistNet
MNIST(KD)	n*SmallMnistNet	SmallMnistNet

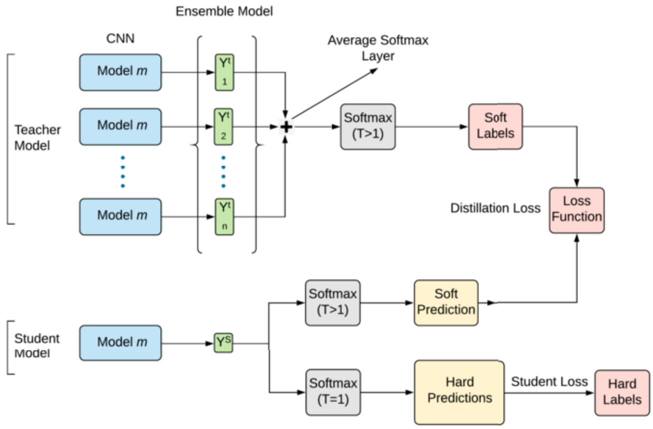

Figure 1. Combined Model: Ensemble Distillation Network

4.2. Datasets

To conduct our experiment, we have used three different datasets. The CIFAR10, CIFAR100 and MNIST. Both CIFAR10 and CIFAR100 datasets are consists of 60,000 images with pixel density of 32×32×3. There are 10 classes in CIFAR10 and 100 classes in CIFAR100 with 6000 and 600 images of each class respectively. Out of 60,000 samples we used 50,000 for our training purposes and 10,000 for testing phase. The MNIST dataset consists of 60,000 handwritten digit images including 10,000 training samples.

4.3. Parameters, Hyperparameters and Loss Functions

Both CIFAR10 and CIFAR 100 experiment has been done using 25 epochs and 64 batch. MNIST experiment has been done using 4 epochs with 64 batch because 4 epochs was enough to provide accuracy around 80% in our EKD model. For bagging ensemble we used 5 combine models of SmallCIFAR10net as teacher model for CIFAR10 experiment and SmallCIFAR100net for CIFAR100 experiment. For distillation purpose, we used temperature value of 3. We used ”adam” optimizer and ”sparse categorical crossentropy” loss function for our teacher model.

5. Result Analysis and Discussion

We have created separate python script for all the experiments. Evaluating our experiments we see some significant results shown in **Table 2**. We are going to discuss on the result one by one.

Table 2. Empirical Comparison between normal knowledge distillation (KD) and ensemble knowledge distillation (EKD) in terms of accuracy

Model	CIFAR10		CIFAR100		MNIST	
	KD (%)	EKD (%)	KD (%)	EKD (%)	KD (%)	EKD (%)
Distilled Student Model	72.2	71.76	34.59	42.05	77.57	79.95
Student Model Alone	70.28	70.44	35.85	34.39	76.32	77.83
Improvement	1.92	1.32	−1.26	7.66	1.25	2.12

Table 3. Training time Comparison between normal knowledge distillation (KD) and ensemble knowledge distillation (EKD).

Datasets	Teacher Model			Student Model		
	KD (sec)	EKD (sec)	Improvement(sec)	KD (sec)	EKD (sec)	Improvement (sec)
CIFAR100	6625	3125	3500	850	750	100
CIFAR10	3825	2750	1075	600	650	-50
MNIST	575	87.5	487.5	25	25	0

5.1. CIFAR10 and CIFAR100

In the case of CIFAR10 dataset classification, although KD has shown better improvement than EKD, we can observe that EKD has obtained 1.32% performance gain on student model which emphasize the fact that teacher model can be a collection of small model and still can improve the standalone model accuracy. In the case of the CIFAR100 experiment, significant performance gain has been observed in the ERD mechanism whereas traditional distillation was not successful using the big model as a teacher model. But using EKD, performance has been enhanced by 7.66%. Additionally, we can observe that it took less time in case for EKD to perform the whole process compared to KD in **Table** 3. Also, to check the influence of hyperparameter like ”number of ensemble network”. We conducted three separate experiments with 5, 10 and 20 ensemble networks and for CIPAR10 we got improvement in our student model in this sequence 1.32%,0.49%, and 2.83%. In our future research we will try to explore more on this. These results from the **Table 2** amplifies the fact that performance improvement in a classification task can be achieved through distillation by using a combination of small models like SmallCIFAR10net like the same way in using a cumbersome model like LargeCNNnet for distillation even better sometimes.

5.2. MNIST

In the case of MNIST dataset, our EKD network outperformed KD by providing 2.12% performance improvement. So, for both CIFAR100 and MNIST dataset, our model gained more accuracy than a traditional KD does.

These results from the **Table 2** approves the fact that performance improvement in a classification task which has been achieved by transferring distilled knowledge from a cluster of small models can be as powerful as transferring from a cumbersome model, and sometimes even better. In future, we also want to provide more statistical evaluation with additional experiments of our proposed mechanism to signifies the impact of our simple ensemble knowledge distillation framework.

6. Conclusions and Future Work

In this paper, we proposed a simple ensemble knowledge distillation approach called EKD. Our experimental results showed that EKD can improve the accuracy of a single learning model through transferring the distilled knowledge from an ensemble models. It performs better than the traditional knowledge distillation using a cumbersome model as the teach model, especially on the CIFAR100 and the MNIST dataset. Specifically, on the CIFAR100 dataset, the experimental result shows that EKD achieved the highest accuracy (around 42.05%), which is much higher than 34.59% (achieved by KD).

Although EKD performs significantly better than KD on CIFAR 100, it loses to KD on CIFAR 10. This motivates us to conduct further study to investigate potential reasons. We will also investigate the performance of hierarchical knowledge distillation, including traditional knowledge distillation and ensemble distillation.

References

[1] Forsyth DA, Ponce J. Computer vision: a modern approach. Prentice Hall Professional Technical Reference; 2002 Mar 1.

[2] Manning C, Schutze H. Foundations of statistical natural language processing. MIT press; 1999 May 28.

[3] Graves A, Mohamed AR, Hinton G. Speech recognition with deep recurrent neural networks. In2013 IEEE international conference on acoustics, speech and signal processing 2013 May 26 (pp. 6645-6649). IEEE.

[4] Schwarting W, Alonso-Mora J, Rus D. Planning and decision-making for autonomous vehicles. Annual Review of Control, Robotics, and Autonomous Systems. 2018 May 29.

[5] Dietterich TG. Ensemble learning. The handbook of brain theory and neural networks. 2002 Mar;2:110-25.

[6] Webb GI, Zheng Z. Multistrategy ensemble learning: Reducing error by combining ensemble learning techniques. IEEE Transactions on Knowledge and Data Engineering. 2004 Aug 2;16(8):980-91.

[7] Asif U, Tang J, Harrer S. Ensemble knowledge distillation for learning improved and efficient networks. arXiv preprint arXiv:1909.08097. 2019 Sep 17.

[8] Buciluǎ C, Caruana R, Niculescu-Mizil A. Model compression. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining 2006 Aug 20 (pp. 535-541).

[9] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531. 2015 Mar 9.

[10] Romero A, Ballas N, Kahou SE, Chassang A, Gatta C, Bengio Y. Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550. 2014 Dec 19.

[11] Schmitt S, Hudson JJ, Zidek A, Osindero S, Doersch C, Czarnecki WM, Leibo JZ, Kuttler H, Zisserman A, Simonyan K, Eslami SM. Kickstarting deep reinforcement learning. arXiv preprint arXiv:1803.03835. 2018 Mar 10.

[12] Yim J, Joo D, Bae J, Kim J. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. InProceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017 (pp. 4133-4141).

[13] Furlanello T, Lipton ZC, Tschannen M, Itti L, Anandkumar A. Born again neural networks. arXiv preprint arXiv:1805.04770. 2018 May 12.

[14] Shen Z, He Z, Cui W, Yu J, Zheng Y, Zhu C, Savvides M. Adversarial-based knowledge distillation for multi-model ensemble and noisy data refinement. arXiv preprint arXiv:1908.08520. 2019 Aug 22.

[15] Chebotar Y, Waters A. Distilling Knowledge from Ensembles of Neural Networks for Speech Recognition. InInterspeech 2016 Sep (pp. 3439-3443).

[16] Fukuda T, Suzuki M, Kurata G, Thomas S, Cui J, Ramabhadran B. Efficient Knowledge Distillation from an Ensemble of Teachers. InInterspeech 2017 Aug 20 (pp. 3697-3701).

[17] Kraipeerapun P, Fung CC. Binary classification using ensemble neural networks and interval neutrosophic sets. Neurocomputing. 2009 Aug 1;72(13-15):2845-56.

[18] Duan T, Srihari SN. Pseudo boosted deep belief network. InInternational Conference on Artificial Neural Networks 2016 Sep 6 (pp. 105-112). Springer, Cham.

[19] Gheyas IA, Smith LS. A novel neural network ensemble architecture for time series forecasting. Neurocomputing. 2011 Nov 1;74(18):3855-64.