Born Again Neural Networks

Tommaso Furlanello
University of Southern California
furlanel@usc.edu

Zachary C. Lipton
Carnegie Mellon University,
Amazon AI
zlipton@cmu.edu

Laurent Itti
University of Southern California
itti@usc.edu

Anima Anandkumar
California Institute of Technology,
Amazon AI
anima@caltech.edu

Abstract

Knowledge distillation techniques seek to transfer knowledge acquired by a learned *teacher* model to a new *student* model. In prior work, the teacher typically is a high-capacity model with formidable performance, while the student is more compact. By transferring knowledge, one hopes to benefit from the student's compactness while suffering only minimal degradation in performance. In this paper, we revisit knowledge distillation but with a different objective. Rather than compressing models, we train students that are parameterized identically to their parents. Surprisingly, these *born again networks* (BANs), tend to outperform their teacher models. Our experiments with born again dense networks demonstrate state-of-the-art performance on the CIFAR-100 dataset reaching a validation error of 15.5 % with a single model and 14.9 % with our best ensemble. Additionally, we investigate knowledge transfer to architectures that are different, but with capacity comparable to their teachers. In these experiments, we show that similar advantages can be achieved by transferring knowledge between dense networks and residual networks of similar capacity.

1 Introduction

In a well-known paper on algorithmic modeling [1], Leo Breiman noted that different stochastic algorithmic procedures [2, 3, 4] can lead to diverse models with similar validation performances. Moreover, he noted that we can often compose these models into an ensemble that achieves predictive power superior to each of the constituent models. Interestingly, given such a powerful ensemble, one can often find a simpler model (no more complex than one of the ensemble's constituents) that mimics the ensemble and achieves its performance. In *Born Again Trees* [5], Breiman pioneers this idea, learning single trees that are able to recover the performance of multiple-tree predictors. These born-again trees approximate the ensemble decision but offer the acknowledged interpretability of decision trees. A number of subsequent papers have rediscovered the idea of *born-again* models. In the neural network community, similar ideas emerged under the names *model compression* [6] and *knowledge distillation* [7]. In both cases, the idea is typically to transfer the knowledge of a high-capacity teacher with formidable performance to a more compact student [8, 9, 10]. Although the student cannot match the teacher when trained directly in a supervised manner, the *distillation* process brings the student closer to the predictive power of the teacher.

We propose to revisit knowledge distillation but with a different objective. Rather than compressing models, we aim to transfer knowledge from a teacher to a student of identical capacity. In doing so,

Metalearning Workshop@31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

we make the surprising discovery that the students become the masters, outperforming their teachers by significant margins. In a manner reminiscent to Minsky's *Sequence of Teaching Selves* [11], we develop a simple re-training procedure: after the teacher model converges, we initialize a new student and train it with the dual goals of predicting the correct labels and matching the output distribution of the teacher. This way the pre-trained teacher can bias the gradients from the environment and potentially lead the students toward better local minima. We call these students *Born Again Networks* (BANs) and show that applied to DenseNets, BANs have consistently lower validation errors than their teachers. We show that this procedure can be applied, albeit with diminishing returns, for multiple steps.

Furthermore, we explore whether the objective function induced by the DenseNet teacher can be used to improve a simpler architecture like ResNet bringing it close to state of the art accuracy without relying on more complicated recent innovations like grouped convolutions [12], stochastic depth [13], learning rate restarts [14], or shake-shake [15]. We construct *wide-ResNets* [16] and *bottleneck ResNets* [17] of comparable complexity to their teacher and show that these BAN-as-ResNets surpass their DenseNet teachers and drastically outperform standard ResNets.

1.1 Residual and Densely Connected Neural Networks

As first described in [18], deep residual networks employ some design principles that are rapidly becoming ubiquitous among modern computer vision models. Multiple extensions [17, 16, 12, 19] have been proposed, progressively increasing their accuracy on CIFAR100 [20] and Imagenet [21]. Densely connected networks (DenseNets) [22] are a recently proposed variation where the summation operation at the end of each unit is substituted by a concatenation between the input and output of the unit.

2 Born Again Networks

Consider the classical image classification setting where we have a training dataset composed by tuples of images and labels $\{x, y\} \in X, Y$ and we are interested in finding a function $f(x) : X \mapsto Y$ able to generalize to unseen data. Commonly, the mapping $f(x)$ is parametrized by a neural network $f(x, \theta_1)$, $\theta_1 \in \Theta_1$. Parameters are learned by empirical risk minimization, where the resulting model θ_1^* is the minimizer of a loss function:

$$\theta_1^* = \arg\min_{\theta_1} \mathcal{L}(y, f(x, \theta_1)), \tag{1}$$

typically optimized by stochastic gradient descent (SGD).

Born Again Networks (BAN) are based on the empirical finding that the solution θ_1^* found by SGD can be sub-optimal in terms of generalization error, thus can be potentially improved modifying the loss function. The most common such modification is to apply a regularization penalty in order to limit the complexity of the learned model. BANs instead exploits the idea demonstrated in knowledge distillation, that the information contained in the original model's output distribution $f(x, \theta_1^*)$ can provide a rich source of training signal, leading to a second solution $f(x, \theta_2^*)$, $\theta_2 \in \Theta_2$, with better generalization ability. We modify the original loss function by adding a knowledge distillation term based on the Kullback–Leibler divergence between the new model's outputs and the outputs of the original model

$$\min_{\theta_2} \mathcal{L}(y, f(x, \theta_2)) + \mathcal{L}(f(x, \arg\min_{\theta_1} \mathcal{L}(y, f(x, \theta_1))), f(x, \theta_2)) \tag{2}$$

Unlike the original works on knowledge distillation, we address the case when the teacher and student networks have identical architectures. Additionally, we present experiments addressing the case when the teacher and student networks have similar capacity but different architectures. For example we perform knowledge transfer from a DenseNet teacher to a ResNet student with similar number of parameters.

2.1 Sequence of Teaching Selves Born Again Networks Ensemble

In Marvin Minsky's Society of Mind [11], in the explanation of human development introduced the idea of a *sequence of teaching selves*. Minsky suggested that sudden spurts in intelligence during

childhood may be due to longer and hidden training of new "student" model under the guidance of the older self. In the same work, Minsky concluded that our perception of a long-term self is constructed by an ensemble of multiple generations of internal models, which we can use for guidance when the most current model falls short.

Inspired by these philosophical ideas and also by impressive recent results of DenseNet Ensembles [23] on CIFAR100, we apply BANs sequentially with multiple generations of knowledge transfer. In each case, the k-th model is trained, with knowledge transferred from the $k-1$-th student:

$$\min_{\theta_k} \mathcal{L}(y, f(x, \theta_k)) + \mathcal{L}(f(x, \arg\min_{\theta_{k-1}} \mathcal{L}(y, f(x, \theta_{k\_1}))), f(x, \theta_k)) \tag{3}$$

Finally, similarly to ensembling multiple snapshot [24] of SGD with restart [14], we produce Born Again Network Ensembles (BANE) by averaging the prediction of multiple generations of BANs.

$$\hat{f}^k(x) = \sum_{i=1}^{k} f(x, \theta_i)/k \tag{4}$$

We find the improvements of the sequence to saturate, but we are able to produce significant gains through ensembling.

2.2 DenseNets Born Again as ResNets

Since BAN-DenseNets perform at the same level as plain DenseNets with multiples of their parameters, we test whether the BAN procedure can be used to improve ResNets as well. Instead of using a weaker ResNet teacher we employ a DenseNet 90-60 as teacher and construct comparable ResNet students switching *Dense Blocks* with *Wide Residual Blocks* and *Bottleneck Residual Blocks*.

3 Experiments

All experiments are performed on CIFAR 100 using the same setting of wide-ResNet [16] except for Mean-Std normalization. The only form of regularization used other than the knowledge distillation loss are weight decay and, in the case of wide-ResNet drop-out.

Baselines In order to get a strong teacher baseline without the prohibitive memory usage of the original architectures, we explore multiple height and growth factors for DenseNets. We find a sweet spot in relatively shallower architectures with increased growth factor and comparable number of parameters to the largest configuration of the original paper. Finally, we construct wide-ResNet and bottleneck-ResNet networks that match the output shape of DenseNet-90-60 at each block as baselines for our BAN-ResNet experiment.

BAN-Densenet We perform BAN re-training after convergence using the same training schedule originally used to train the teacher networks. We employ DenseNet-(116-33, 90-60, 80-80,8 0-120) and train a sequence of BANs for each configuration. We test the ensemble performance for sequences of 2 and 3 BANs. We also train variations of DenseNet-90-60, with increased or decreased number of units in each block and different number of channels determined through a ratio of the original activation sizes.

Table 1: Born Again DenseNet: test error on CIFAR100 for DenseNet of different depth and growth factor, the respective sequence of BAN-DenseNet, and the BAN-ensembles resulting from the sequence. Each BAN is trained from the label loss and cross-entropy with respect to the model at its left. We include the original teacher as a member of the ensemble for Ens*3 for 80-120 since we did not train a BAN-3 for this configuration.

Network	Parameters	Baseline	BAN-1	BAN-2	BAN-3	Ens*2	Ens*3
DenseNetBC-112-33	6.3 M	18.25	17.61	17.22	**16.59**	15.77	15.68
DenseNetBC-90-60	16.1 M	17.69	16.62	**16.44**	16.72	15.39	15.74
DenseNetBC-80-80	22.4 M	17.3	16.26	16.30	**15.5**	15.46	15.14
DenseNetBC-80-120	50.4 M	16.87	**16.13**	16.13	/	**15.13**	**14.9**

Table 2: BAN-ResNets and modified BAN-DenseNets: CIFAR100 test error for BAN-ResNets and BAN-DenseNet trained from a DenseNet 90-60 teacher with different numbers of blocks and channels. In match architectures first is indicated the number of units per blocks, and then the ratio of input and output channels with respect to a DenseNet 90-60 block.

Network	Parameters	Baseline	BAN
Pre-activation ResNet-1001 [17]	10.2 M	22.71	
Match-Pre-ResNet-14-0.5	7.3 M	20.28	18.8
Match-Pre-ResNet-14-1	17.7 M	18.84	**17.39**
Wide-ResNet-28-10	36 M	18.63	/
Wide-ResNet-28-20	146 M	17.64	/
Match-Wide-ResNet-1-1	20.9 M	20.4	19.12
Match-Wide-ResNet-2-1	43.1 M	18.83	17.42
Match-Wide-ResNet-4-0.5	24.3 M	19.63	**17.13**
Match-Wide-ResNet-4-1	87.3 M	18.77	17.18
Match-DenseNet-7-1-2	21.2 M	/	16.95
Match-DenseNet-14-0.5-1	5.1 M	/	19.83
Match-DenseNet-14-0.75-1.5	10.1 M	/	17.3
Match-DenseNet-14-1-3	80.5 M	/	18.89
Match-DenseNet-28-1-2	13.7 M	/	**16.43**
Match-DenseNet-42-1-2	12.9 M	/	16.64
Match-DenseNet-56-1-2	12.6 M	/	16.64

BAN-Resnet In all the BAN-ResNet experiments, the student shares the first and last layer with its teacher. We modulate the complexity of the ResNet by changing the number of units, starting from the depth of the successful wide-ResNet28 [16] and reducing until there is only a single residual unit per block. Since the number of channels in each block is constant in every residual unit, we match it with a proportion of the corresponding dense block output after the 1x1 convolution, before the spatial down-sampling. We explore mostly architectures with a ratio of 1, but also show the effect of halving the width of the network.

4 Results

We report the surprising finding that by performing knowledge distillation across models of similar architecture, BAN-DenseNets and BAN-ResNets improve over their teachers across all configurations. The third generation of BAN-3-DenseNet-80-80 produces a single model with 22M parameters that achieves 15.5 % error on CIFAR100 as can be noted in Table 1.

To our knowledge, this is currently the SOTA non-ensemble model trained with SGD without any sort of shake-shake regularization. It is only beaten by [25] who use a *pyramidal ResNet* trained for 1800 epochs with a combination of shake-shake [15], pyramid-drop [26] and cut-out regularization [27].

Similarly our largest ensemble BAN-3-DenseNet-BC-80-120 with 150M parameters and an error of 14.9 is the lowest reported ensemble result in the same setting. BAN-3-DenseNet-112-33 is based on the building block of the best coupled-ensemble of [23] and reaches a single-error model of 16.59 with only 6.3 M parameters, furthermore the ensembles of two or three consecutive generations reach a comparable error of 15.77 and 15.68 with the baseline error of 15.68 reported in [23] where four models were used.

Finally (table 2), BAN-ResNet outperforms both their traditional counterparts, equivalent ResNets trained without DenseNet teacher, and their DenseNet teacher. Similarly BAN-DenseNet are robust to changes in the number of layers, offering a nice trade-off between memory consumption and number of sequential operations. We manage to find students worse than their master only by setting the number of blocks to 1 or by drastically reducing the number of features.

References

[1] Leo Breiman et al. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231, 2001.

[2] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10):993–1001, 1990.

[3] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[4] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

[5] Leo Breiman and Nong Shang. Born again trees. 1996.

[6] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 535–541. ACM, 2006.

[7] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[8] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems*, pages 2654–2662, 2014.

[9] Gregor Urban, Krzysztof J Geras, Samira Ebrahimi Kahou, Ozlem Aslan, Shengjie Wang, Rich Caruana, Abdelrahman Mohamed, Matthai Philipose, and Matt Richardson. Do deep convolutional nets really need to be deep and convolutional? *arXiv preprint arXiv:1603.05691*, 2016.

[10] Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. *arXiv preprint arXiv:1511.06295*, 2015.

[11] Marvin Minsky. Society of mind: a response to four reviews. *Artificial Intelligence*, 48(3):371–396, 1991.

[12] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *arXiv preprint arXiv:1611.05431*, 2016.

[13] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European Conference on Computer Vision*, pages 646–661. Springer, 2016.

[14] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[15] Xavier Gastaldi. Shake-shake regularization. *arXiv preprint arXiv:1705.07485*, 2017.

[16] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[18] 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*, pages 770–778, 2016.

[19] Dongyoon Han, Jiwhan Kim, and Junmo Kim. Deep pyramidal residual networks. *arXiv preprint arXiv:1610.02915*, 2016.

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

[21] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

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

[23] A. Dutt, D. Pellerin, and G. Quenot. Coupled Ensembles of Neural Networks. *ArXiv e-prints*, September 2017.

[24] Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E Hopcroft, and Kilian Q Weinberger. Snapshot ensembles: Train 1, get m for free. *arXiv preprint arXiv:1704.00109*, 2017.

[25] Anonymous. Shakedrop regularization. *International Conference on Learning Representations*, 2018.

[26] Yoshihiro Yamada, Masakazu Iwamura, and Koichi Kise. Deep pyramidal residual networks with separated stochastic depth. *arXiv preprint arXiv:1612.01230*, 2016.

[27] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.