A Survey of the Current State of Self-Distillation

Ethan Glaser
Cornell Tech
New York, NY 10044
eg492@cornell.edu

1 Introduction

In recent years, neural networks have exploded in size with hardware and software developments making the creation of tremendously deep networks possible, as well as the ensembling of several models, whose outputs are averaged to determine an output. Unfortunately, despite these options being high performing, they are generally not feasible to implement in practical contexts, especially those where compute is limited or timing is crucial (1).

Knowledge distillation is the process of distilling information from a large pre-trained network onto a smaller, more deployable network. The training of the smaller network involves both the hard targets (the normal training targets of a typical training process) as well as soft targets produced by the large "teacher" network (2). This process seeks to replicate and distill the abilities of the larger model onto the smaller model, with the process shown in Figure 1.

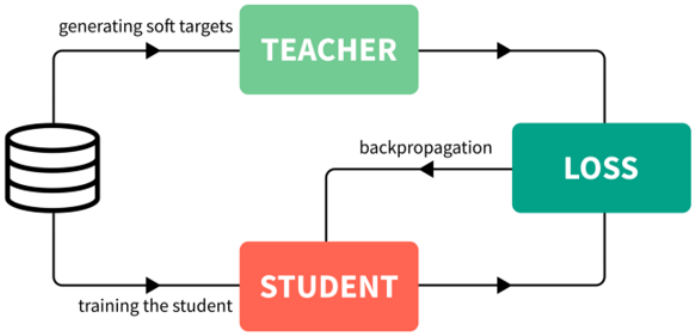


Figure 1: A visualization of the knowledge distillation of the teacher and student networks

There is no tried and true mathematical process for selecting the right teacher for a student, selecting the nuances of the smaller network, and identifying the hyperparameters of which to use during knowledge distillation. Therefore, there are some existent flaws with the process. Due to the lack of a scientific process, there is often uncertainty about the design and training of the teacher models, leading to difficulties with creating effective teacher models as well as identifying the right architecture of which to distill the knowledge onto. There is also relatively low efficiency in the knowledge transfer process - student networks are unable to replicate the performance of their teachers (2). Additionally the training of the student network is still very time consuming and compute heavy, due to the inference of the larger network being required. This is especially true if training or fine-tuning of the teacher network is required before knowledge distillation can begin. Therefore, knowledge distillation is a field of active research and exploration of opportunities to

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

improve the current state of the art, especially in optimizing the training process of student models (3).

Self-distillation is a variant of knowledge distillation in which no external teacher network is trained or used, but instead the architecture and weights of the student network is used in a true self distillation - distilling knowledge from the network itself (4). This bypasses the concern that there is no tried and true method for selecting the right teacher architecture for a student since the architecture is the student itself. There is also a significant reduction in the computation and training speed required to train a self-distilled due to the lack of the external teacher. Despite all of this, preliminary results show great experimental performance of self-distillation as well as beneficial practical applications that suggest self-distillation has all around potential moving forward in the topic of knowledge distillation.

2 Implementation & Training

Self-distillation is implemented such that the deeper layers of the network distill information onto the shallower layers. In order to do so, sections of layers need to be able to produce a consistent output, which is achieved by attaching attention-based classifiers to the output of each "sub-network" or section of layers that comprises a teacher or student within the self distilled architecture (5). Instead of a large pre-trained network providing the soft targets for training, the deeper (or often just the deepest) classifiers provide the soft targets for the shallower classifiers, updating the weights of the shallower layers in the network based on this knowledge distillation, in a process better visualized than explained with text, shown below.

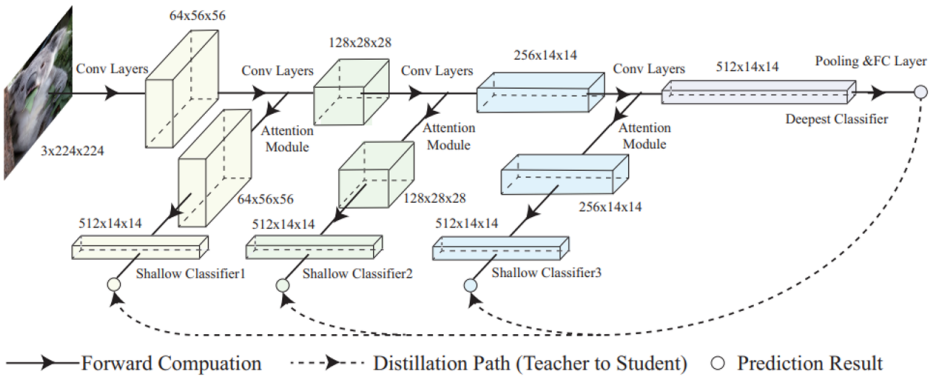


Figure 2: A visual of self-distillation, with classifiers attached to the sub networks (5)

The loss function involved with this training process is therefore slightly modified compare with traditional knowledge distillation, with 3 major components:

- Cross-entropy loss between training labels and classifier softmax output (used with essentially any model training)
- KL divergence computed between softmax output of students and teachers (also used in traditional knowledge distillation)
- L2 loss between the feature maps of shallow and deep classifiers (unique to self-distillation)

$$loss = \sum_{i}^{C} loss_i = \sum_{i}^{C} \Big((1-\alpha) \cdot CrossEntropy(q^i, y) + \alpha \cdot KL(q^i, q^C) + \lambda \cdot ||F_i - F_C||_2^2 \Big)$$

Figure 3: The 3 components of the self-distillation loss function (2)

There are several significant advantages of self-distillation. The use of a compact teacher model instead of a large over-parameterized model leads to high compression and faster training. The magnitude of improvement in training speed is displayed in the following figure, with the complexity

of the traditional knowledge distillation training process divided up between the process of training or fine-tuning the large teacher network followed by the process of distilling the knowledge onto the student network (2).

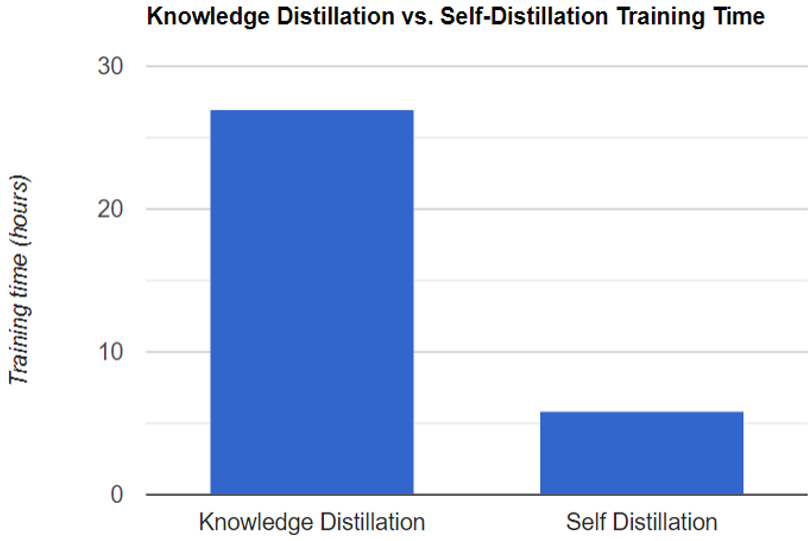


Figure 4: Comparison of train time between knowledge and self distillation

The breakdown of runtime of these two processes was 14.67 hours of time devoted to training the teacher and 12.31 hours of compute towards distilling this knowledge onto the student, compared to a total time of 5.87 hours with self distillation - just over 20% of the time of knowledge distillation.

Another advantage of using self distillation is the division of the network into sub-sections that are individually trained greatly discourages the vanishing gradient problem (2), as there is a lot more training updates made and a lot less distance between the beginning and end of layers being trained in the network.

The performance of self-distillation has been impressive so far, with a variety of experimentation showing very promising results, which will be discussed more in detail in the following section, but this is yet another advantage of self-distillation (2).

3 Experimental Results

The experimental results of self-distillation from several papers show relative success (2) (5) and promise for the future of knowledge distillation. The original self-distillation paper tested 5 architectures on two of the most common image datasets - ImageNet and CIFAR100 – averaging a 2.65% average boost in classification accuracy without slower response time in comparison to traditional training methods, with all other factors equal. Along with the overall average boost, self-distillation outperformed standard training on every single combination of dataset and model, as displayed in the following figure.

The results in these two images display the performance at each layer within a 4 section self-distilled models, with even some of the shallower classifiers outperforming the traditional training, and every single combination outperforming the baseline when considering the deepest classifier, with an even further performance boost when ensembling the 4 classifiers together.

Additional performance comparisons are made with traditional knowledge distillation (5). The following figure shows the performance results of self-distillation in comparison with various state of the art knowledge distillation methods on the same datasets described previously, with self-distillation again running the table and outperforming each knowledge distillation methods.

Self-distillation showed performance increase related to the depth of the network - the deeper the network, the more improvement self-distillation yielded.

Neural Networks	Baseline	Classifier 1/4	Classifier 2/4	Classifier3/4	Classifier 4/4	Ensemble
VGG19(BN)	64.47	63.59	67.04	68.03	67.73	68.54
ResNet18	77.09	67.85	74.57	78.23	78.64	79.67
ResNet50	77.68	68.23	74.21	75.23	80.56	81.04
ResNet101	77.98	69.45	77.29	81.17	81.23	82.03
ResNet152	79.21	68.84	78.72	81.43	81.61	82.29
ResNeXt29-8	81.29	71.15	79.00	81.48	81.51	81.90
WideResNet20-8	79.76	68.85	78.15	80.98	80.92	81.38
WideResNet44-8	79.93	72.54	81.15	81.96	82.09	82.61
WideResNet28-12	80.07	71.21	80.86	81.58	81.59	82.09
PyramidNet101-240	81.12	69.23	78.15	80.98	82.30	83.51

Table 1. Experiments results of accuracy (%) on CIFAR100 (the number marked in red is lower than its baseline).

Neural Networks	Baseline	Classifier 1/4	Classifier 2/4	Classifier 3/4	Classifier 4/4	Ensemble
VGG19(BN)	70.35	42.53	55.85	71.07	72.45	73.03
ResNet18	68.12	41.26	51.94	62.29	69.84	68.93
ResNet50	73.56	43.95	58.47	72.84	75.24	74.73

Table 2. Experiments results of top-1 accuracy (%) on ImageNet (the number marked in red is lower than its baseline).

TABLE 1
Experiment results of accuracy (%) on CIFAR100. "Baseine" in the table indicates a model trained without knowledge distillation. "Ensemble" indicates the prediction ensemble of "Ensemble" indicates the prediction ensemble of all the classifiers.

Models	Baseline	Classifier1	Classifier2	Classifier3	Classifier4	Ensemble
ResNet18	79.01	76.31	79.32	81.24	81.76	82.64
ResNet50	80.88	82.60	83.55	84.42	83.66	85.42
ResNet101	82.37	81.64	82.73	84.12	84.03	85.48
ResNet152	82.92	81.25	82.94	84.37	84.52	85.41
WRN50-2	81.26	82.85	84.02	84.91	84.33	85.78
WRN101-2	82.37	82.56	83.79	84.87	84.33	86.03
SENet18	79.53	75.60	79.81	81.77	81.84	83.10
SENet50	81.01	81.80	82.93	83.91	83.51	85.21
SENet101	82.75	82.20	82.69	83.17	82.97	84.82
ResNeXt50-4	82.65	82.03	83.50	83.78	83.42	85.12
ResNeXt101-8	82.96	82.84	83.70	84.70	84.31	85.81
ResNeSt50	83.12	83.09	84.01	84.98	85.19	86.40
MobileNetV2	65.49	62.93	66.03	67.95	67.17	68.87
ShuffleNetV2	71.61	73.20	73.87	75.66	/	76.45

TABLE 2
Experiment results of accuracy (%) on ImageNet. "Baseine" in the table indicates a model trained without knowledge distillation. "Ensemble" indicates the prediction ensemble of all the classifiers.

Models	Baseline	Classifier1	Classifier2	Classifier3	Classifier4	Ensemble
ResNet18	69.21	55.03	60.94	64.70	70.51	70.63
ResNet50	76.30	71.72	74.58	77.45	77.89	78.28
ResNet101	77.03	71.75	74.39	79.47	79.70	78.87
ResNet152	77.62	71.50	75.36	80.22	80.32	80.56
ResNeXt50-4	77.29	71.95	75.76	79.02	79.96	80.32
WideResNet50	77.46	72.37	75.99	79.22	79.87	80.17

Figure 5: Experimental results in comparison to standard training from two sources (2) (5)

4 Self-Distillation Application: BERT

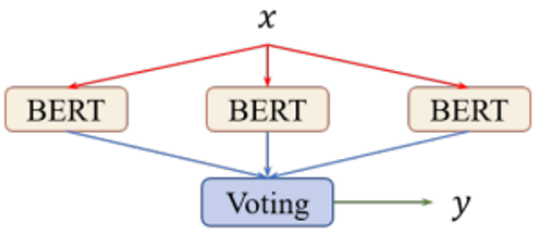
The benefits of self-distillation have been explored in various practical applications, one of which includes a comparison of self-ensembling and self-distillation for fine-tuning BERT. In the context of this paper, self-ensembling is the process of creating an ensemble of models from different time steps of the model training process (6). This paper also defines a slight variation to the self-distillation process, although the core idea is very similar - the teacher is self-ensemble model along with labeled training data. Experimentation involved evaluation of Self-Ensemble, Averaged Self-Distillation (average parameters of models), and Voted Self-Distillation (averaged output of models).

All of these methods were able to improve the performance of BERT, with the improvements shown in the following figure.

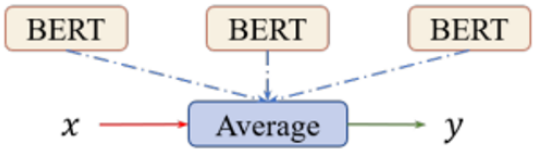
The results of this application indicate that a higher potential of BERT can be achieved when using a better fine-tuning strategy even without leveraging external knowledge or data. BERT performance can be improved without significantly decreasing the training efficiency using self-ensemble with

Teacher Model Student Model	ResNet50 ResNet18	ResNet101 ResNet18	ResNet101 ResNe50
Teacher Accuracy	80.88	82.37	82.37
Student Accuracy	79.01	79.01	80.88
KD [29]	80.49	80.31	82.09
FitNet [35]	80.67	80.54	82.14
AT [36]	80.43	80.39	81.92
DML [84]	80.52	80.57	82.37
RKD [42]	80.69	80.67	82.29
SPKD [43]	80.57	80.45	82.16
Feat [85]	80.91	80.80	82.40
Ours	81.76	81.76	85.42
KD + Ours	82.23	82.17	85.92
AT + Ours	82.34	82.21	85.65
DML + Ours	82.09	82.14	85.91

Figure 6: Experimental results in comparison to various knowledge distillation techniques (5)



(a) Voted BERT



(b) Averaged BERT

Figure 7: A Visualization of voted and averaged self-distillation (6)

parameter averaging. If this is the case with this application, it may be possible for any state of the art network to improve performance without utilizing external knowledge or data but by simply learning from its own architecture and parameters using a self-distillation approach.

This application of self-distillation reinforces the relevance of self-distillation and its ability to improve network performance in a practical manner.

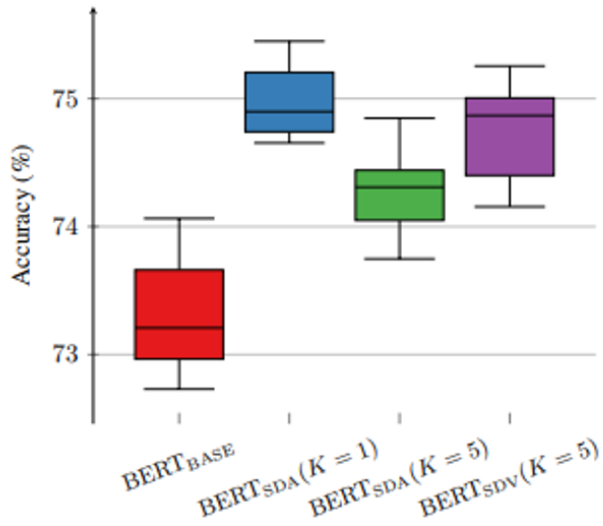


Figure 8: Experimental results in comparison to various knowledge distillation techniques (6)

5 Theoretical Justification of Self-Distillation

Previously cited papers provide a description of how self-distillation works, the advantages in comparison to traditional knowledge distillation, promising experimental performance results, and analysis of the practical applications of self-distillation. But these papers fail to identify why. Why does self-distillation work? What is the reason behind the success, and what considerations need to be made when applying self-distillation? It is critical to understand the theoretical limitations of an algorithm, and this is one critique of some of the introductory papers on this topic, granted they recognize this limitation and the focus of their research was not specifically on this topic. Recent publications (7) (8) dive more into the theoretical justification of self-distillation to develop an understanding of limitations and have a better sense of the applications of self-distillation.

One of these papers focuses on the relationship between the number of iterations of self-distillation and the impact on regularization (9). When considering self-distillation as a process of fitting a non-linear function to training data, this implementation shows that with each iteration of self-distillation, the number of basis functions used to represent the non-linear solution is progressively limited. This is followed with both a theoretical and experimental verification that overfitting of the non-linear model can be reduced by implementing a few rounds of self-distillation, but overdoing it with too many rounds can lead to too much regularization and an underfit model that does not perform well and eventually collapses to zero (9). The following figure outlines the findings of this paper by presenting how the number of iterations of self distillation impacts performance, specifically showing how initial reduction of overfitting improves performance, but continuing on for too long leads to a poor performing underfit model.

The more specific implementation and technical details detail that self-distillation results in a power iteration where the linear operation is modified at each step, with each subsequent iteration largely dependent on previous steps. This theoretical justification also yielded a closed form solution for the lower bound of number of distillations that would yield a collapse, which enables the avoidance of this situation when applying self-distillation (9).

Another paper that seeks to provide theoretical justification for self-distillation relates it to label smoothing, which provides more consistency with model predictions (7). They dive into technical details that outline an amortized MAP interpretation of knowledge distillation, providing technical insights about self-distillation and its flaws. By using this strategy to encourage either more diverse or confidence in the predicted soft labels. The implementation of self-distillation, similar to label smoothing, leads to better calibration performance, which can enable more efficient discovery of regularization priors. The paper also discusses empirical results to verify that the diversity of teacher soft outputs is correlated with the student model's performance (7).

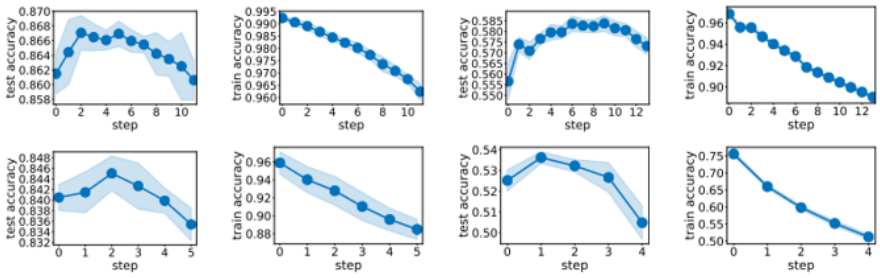


Figure 9: Experimental results of the number of iterations of self-distillation and model performance (9)

Overall, these two papers provide more theoretical insight and justification about self-distillation, particularly how it acts as a regularization technique and provides details on how to utilize it effectively to get the most out of models without the need for new data.

6 Literature Review Recap

As far as a literature review goes, there were 5 papers primarily used in creating the bulk of the content of this survey (5) (2) (7) (6) (9) with an additional 5 used to provide context around knowledge distillation or minor details related to self distillation (1) (8) (4) (3) (10). The papers generally fell into 3 main trajectories as far as content goes - introducing the topic and providing context and preliminary experimental results (5) (2), exploring a practical application of self distillation (9), and attempting to provide theoretical justification for self distillation (7) (6). Because the topic and content are all relatively new, the papers aren't so much as competing with each other but more so aiming to explore the problem and identifying how useful self distillation will be moving forward.

As explained earlier, one of the main critiques of the context and experimental papers is that they provided little to no theoretical explanation of why self-distillation works so well, more so introducing the topic and identifying it as something of interest and more exploration moving forward. These papers (5) (2) do provide a very concise and clear overview of what self distillation is and some of the advantages of it over knowledge distillation, as well as providing experimental results that strongly reinforce the relevance of self distillation - with dozens of different network and dataset combinations explored - overall very thorough experimentation. Therefore these two papers do a very nice job in accomplishing their goal of introducing a potentially exciting topic and providing results to back up this claim, paving the way for more research moving forward.

This transitions smoothly into the exploration of a more practical application of self distillation, in fine tuning a large NLP model, BERT. This paper (6) provides context and details about the process of fine-tuning BERT using self distillation methods and again reports successful experimental results. One suggestion for this paper is to dive a bit deeper into why this works with BERT and whether a similar approach is generalizable to be relevant and applicable towards other models.

Lastly, two papers (7) (9) dive into exploring the theoretical reasoning behind self distillation, revealing that self distillation as a form of regularization and that it is important to not under or over self distill, as this can lead to overfitting or underfitting. Much of the theory was a bit over my head, but the papers generally did a good job of providing understandable high level details in the abstract, intro and conclusion. Overall the papers don't necessarily provide the entire picture but represent a step in the right direction towards understanding why self distillation is effective.

Overall these papers mesh nicely together to create an overview of self distillation, and it will be interesting to see how their research is leveraged moving forwards.

7 Conclusion

Self-distillation is a novel approach towards accurate, compact networks that avoid the massive training overhead of traditional knowledge distillation. As this topic is being explored, several experi-

mentation results show surprisingly strong accuracy among networks trained using self-distillation, both in comparison to traditional training and normal knowledge distillation. Additionally, the theoretical justification of the performance of self distillation is being uncovered, which will enable more scientific approaches regarding the use of self distillation. Overall, self distillation is a major development in the knowledge distillation space, which is significant for the future of training and implementing high performing but deployable models.

References

[1] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[2] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma, "Be your own teacher: Improve the performance of convolutional neural networks via self distillation," *ICCV 2019*, 2019.

[3] D. Y. Park, M.-H. Cha, C. Jeong, D. Kim, and B. Han, "Learning student-friendly teacher networks for knowledge distillation," 2021.

[4] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," 2021.

[5] L. Zhang, C. Bao, and K. Ma, "Self-distillation: Towards efficient and compact neural networks," *IEEE*, 2015.

[6] Y. Xu, X. Qiu, L. Zhou, and X. Huang, "Improving bert fine-tuning via self-ensemble and self-distillation," 2020.

[7] M. S. Zhilu Zhang, "Self-distillation as instance-specific label smoothing," *NeurIPS 2020*, 2020.

[8] K. Borup and L. Andersen, "Even your teacher needs guidance: Ground-truth targets dampen regularization imposed by self-distillation," *NeurIPS 2021*, 2021.

[9] H. Mobahi, M. Farajtabar, and P. Bartlett, "Self-distillation amplifies regularization in hilbert space," *NeurIPS 2020*, 2020.

[10] Z. Allen-Zhu and Y. Li, "Towards understanding ensemble, knowledge distillation and self-distillation in deep learning," 2020.