Transferring Inductive Bias through Leveled Knowledge Distillation

Clement Wong
clemwong@umich.edu

Derek Fermin
ferminde@umich.edu

Gretchen Zheng
zhengji@umich.edu

Jeongsoo Park
jespark@umich.edu

Zilin Wang
zilinwan@umich.edu

Abstract

Knowledge distillation is a method with a large potential as it allows a simple model trained under constrained resources to have a great performance by learning from a larger pretrained teacher model. Our project verified that knowledge distillation could improve a simple student model's performance by learning from a more complicated pretrained teacher model through the transfer of the inductive bias. In our project, we show that we can train a simple translation-invariant MLP model using a CNN teacher model, and thus verify the previous conclusion. Moreover, we propose two new methods to further improve the transfer of inductive bias: training with a weaker teacher and applying feature matching.

1 Introduction

Convolutional neural networks (CNN) have been the dominant approach for computer vision tasks in the past decade. While this may be partially attributed to the rapidly growing computational power and improved neural network design, CNNs' success is largely based on its ability to learn filters that extract meaningful spatial features. These filters can be 2 dimensional for images, 3 dimensional for videos and 3D image, or even higher dimensional in other applications. CNN's ability to learn spatial features can be interpreted as an *inductive bias* injected to the neural network by the experts. Experts understand that these models will be used on data with spatial correlations, thus they used the model that best exploits these features. However, some neural networks such as multi-layer perceptrons (MLP) may not have the capacity to learn such features without human intervention.

Recently, there has been some attempts [4] of using Knowledge Distillation (KD) to transfer "knowledge" learned by the sophisticated neural network to a simpler one. Knowledge distillation utilizes teacher model's output as a label to train the student model. This allows the student models to learn from a more meaningful label rather than traditional one-hot encoded labels. Based on the research by Abnar et al., 2021 [1], there is a possibility that knowledge distillation may transfer the inductive bias of the teacher model to a student model. The transfer of inductive bias has the potential to improve accuracy and training acceleration of a model with a simple architecture.

In our project, we demonstrate that knowledge distillation can transfer inductive bias from sophisticated teacher model to a simple student model. We show this by transferring the inductive bias of a teacher CNN to a student MLP so that the student model exhibits translational equivariance of a CNN. This transfer of inductive bias through knowledge distillation ultimately boosted

Github Repository: https://github.com/Wayne2Wang/Knowledge-Distillation the robustness of the student model without changing the complexity of its architecture. Having identified that knowledge distillation can transfer inductive bias and improve the performance of a student model, we refine the process of knowledge distillation so that we can further transfer the inductive bias to the student. The experimental observations suggest that training with a weaker/leveled teacher and applying feature matching allows the student model to further exhibit the teacher model's inductive bias.

2 Proposed Method

In this project, we aim to prove that knowledge distillation can indeed transfer the inductive bias of the teacher model onto the student model but under certain conditions: with a leveled teacher and with feature matching. To this end, we aim to train a simple student multi-layer perceptron (MLP) model from a teacher convolutional neural network (CNN) model which has been proven to be translational invariant. Our goal will be to show that the invariance to translation from the teacher can be transferred to the student, thus showing that the inductive bias has been transferred through knowledge distillation.

In order to verify whether knowledge distillation can transfer the inductive bias, we separated our process into two main parts: defining a loss function that allows the student model to learn from the teacher and ground truth in different degrees and setting up a training and evaluation process to showcase the inductive bias has indeed been transferred from the teacher to the student.

2.1 Loss for Knowledge distillation:

We implement a special loss for student model during training. First, we use a cross-entropy loss (L_{CE}) to measure the accuracy of the hard label prediction generated by the student model. Then, we use KL divergence (L_{KL}) to measure the difference between the soft predictions generated by the student model (softmax($\psi_{student}(\mathbf{X})$)) and the soft labels generated by the teacher (softmax($\psi_{teacher}(\mathbf{X})$)). Finally we use an L1 loss (L_{FM}) between the student's and teacher's features to measure the difference between these. This feature matching loss with the KL divergence loss will allow the student model to try to approximate the soft labels and features of the teacher model, and thus perform knowledge distillation.

The CE loss between the student's hard label predictions and ground truth labels is specified as:

$$L_{CE} = -\log\left(\frac{\exp(s_{y_i})}{\sum_j \exp(s_j)}\right) \tag{1}$$

The KL Loss between student's soft predictions and teacher's soft label is specified as:

$$L_{KL} = KL((\text{softmax}(\frac{\psi_{student}(\mathbf{X})}{\tau}), \text{softmax}(\frac{\psi_{teacher}(\mathbf{X})}{\tau})) \tag{2}$$

Here, τ is a hyperparameter representing the temperature. For high temperatures, output probability becomes 'smoothed' out. That is, the values tend to get more distributed than the original prediction. For low temperatures, the probability resembles more like the original predictions.

The L1 Loss between the student's and teacher's models features is specified as:

$$L_{FM} = \sum_{i=1}^{n} |y - \hat{y}_i| \tag{3}$$

Combining the cross-entropy loss(L_{CE}) for hard label prediction, the KL divergence loss(L_{KL}) measuring the difference between student's and teacher's soft predictions, and the feature matching loss(L_{FM}) measuring the difference between the student's and teacher's features, we get our overall loss function:

$$L = \alpha L_{CE} + (1 - \alpha)(s_{KL}\tau^2 L_{KL} + s_{FM}L_{FM}) \tag{4}$$

Here the hyperparameters are α, τ, s_{KL}, and s_{FM}, where α determines how much weight we want to give to the CE loss compared to the other losses. τ is as described above. s_{KL}, and s_{FM} determine how much weight we want to give to the KL loss and the L1 loss. Take note of the τ^2 term which is there to even out the gradients when backpropagating. This ensures that no loss is much greater than the others, leading to a more distributed contribution from all the losses.

A graphical pipeline of how the losses are used during training is shown below:

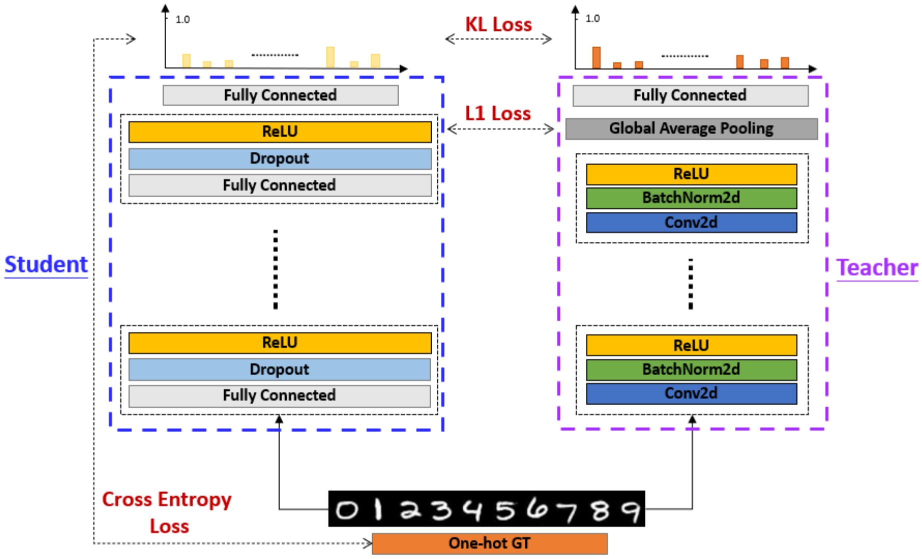


Figure 1: Diagram of loss components and how they are being used

As it can be seen in Figure 1, each of the loss components is included with an arrow designating how it is being used. The Cross Entropy loss is used between the Ground truth labels and the student predictions. The KL loss is used between the teacher's and student's predictions and the L1 Loss is used between the student's and teacher's features. These features are the output of the penultimate layer of each model.

This setup for our loss (having three separate components) allows us to test which combination of components is more effective at transferring the inductive bias from the teacher model as stated by our our objectives and shown by our experimental results.

2.2 Evaluation Methods:

Overall our approach is divided into two: we first train our models on the standard MNIST and CIFAR-10 datasets. Then, we measure the degree of such transfer of inductive bias by comparing the student MLP's accuracy on the translated CIFAR-10 and MNIST-C dataset with that of the standard MLP model. Here MNIST-C stands for corrupted MNIST and we choose a subset from which is composed of translated images of the original MNIST dataset.

For the standard MLP models, we trained a simple three/four hidden layer model with ReLU activation and a constant dropout rate of 0.3. For the teacher models, we used a CNN model with 4 blocks composed of (convolutional layer - Batch Norm - ReLU).

For quantitative evaluation, we tracked the top 1 accuracy metrics for the standard and knowledge-distilled MLP model. Measuring the accuracy difference between two MLP models when the input data is augmented (translated) vs not augmented allow us to see if the knowledge distillation actually transferred the inductive bias of a teacher. We try this on different-shaped student network as specified in the experimental result section.

2.3 Experimental Approach:

We first aim to prove that a leveled teacher is necessary for good transfer of inductive bias. To prove this, we used different snapshots of our CNN teacher model to teach the student model using normal Knowledge distillation approach with only CE and KL loss. We then measured the performance of student model per trained batch iterations of the teacher and show that a teacher closer to the student in terms of complexity allows the student to learn better.

Our second experiment builds on the result of experiment 1. In this case we aim to improve the KD scheme by implementing feature matching through an L1 loss. In this case we compare a weakly-trained and strongly-trained version of our teacher CNN model, and then compare a weak teacher versus a strong teacher with label smoothing. After we try and compare different feature matching settings to see which components are significant in transferring the inductive bias. In a same sense, we also compare the results from using different α values in order to investigate the contribution between ground truth cross entropy loss and knowledge distillation loss. These different approaches and combinations allows us to show that alongside the leveled teacher, feature matching boosts the transfer of inductive bias from the teacher model. This is further explained in our results section.

3 Related Work

Knowledge Distillation. Knowledge distillation is a technique to transfer the knowledge learned by a cumbersome teacher model to a much simpler student model. It was first introduced by Cristian Buciluă, Rich Caruana, and Alexandru NiculescuMizil [2]. With the pre-trained teacher model, student models are optimized on unlabeled, unseen data by minimizing the squared distance between the logits of the teacher model and the student model. Later, the term "knowledge distillation" was formally proposed by [4]. The difference is that they trained on unseen but labeled data, which enables them to train the student network with a composite loss. The student network now has to learn from the high temperature predictions of the teacher model and the ground truth labels. More recent works concentrate on interpreting the foundations of knowledge distillation [7] to gain deeper theoretical understanding [3, 8, 10], and to improve the methodology in various ways [5, 6, 11, 12]. However, they are mainly focused on generalization performance and data efficiency, while not exploring the significance of inductive bias in depth.

Inductive Bias. Inductive Bias is the set of assumptions that the learner/model uses to predict outputs of unseen inputs [9]. So in the case of CNNs for example, the inductive bias are the assumptions that there are some types of spatial structures present in the data. Which in turn allows the model to exhibit invariance to affine transforms such as translation or scaling to some degree.

Inductive Bias Transfer of Knowledge Distillation Recent study by Abnar et al. [1] demonstrated that the inductive bias of the teacher model could be transferred to the student model through knowledge distillation. They claim that the MLP model trained with CNN as a teacher showed robustness to translation and scaling, which MLPs are traditionally known for not having. They verified their statement by testing the accuracies of distilled-MLP models on translated and scaled MNIST-C dataset.

Our experiment is similar in a sense that we are also using knowledge distillation and exploring if the inductive bias is transferred. However, we believe that the approach by Abnar et al. is not very telling of the different subtleties in Knowledge Distillation and just testing on MNIST is too elementary. That is, MNIST is too simple of a dataset to claim that the inductive bias has been truly transferred without showing any more details. Therefore, in our experiment, we plan to expand their research by trying out a different, more complex dataset (CIFAR-10) and defining a more complex loss function to give more ways for the student to learn from the teacher. We explore if the knowledge distillation transfers the inductive bias better in this case and how using different trained snapshots of the teacher can improve the transfer of knowledge.

4 Experimental Results

In this section, we try to verify two main claims. The first claim we want to verify is that using a leveled teacher instead of a stronger one leads to student model being more successful at correctly classifying translated images. The second claim we want to check is that matching the feature space of teacher CNN and student MLP models further improves the transfer of inductive bias.

4.1 Leveled teachers are better teachers

We first experimented with how the transfer of inductive bias was affected using different teacher CNNs. In other words, we want to see what teacher CNN gives the most robust student MLP through knowledge distillation. This seems like an obvious question at the first glance, and the most common practice is to use the most powerful teachers that are well-trained and have the best robustness themselves. However, this is not the case according to our experiments. In our experiment, we first train the 4-layer teacher CNN described in the methods section on MNIST. During the training time, we take different snapshots of the model by storing the specific model weights. Once we have all the teacher snapshots, we train another set of student MLPs on MNIST by knowledge distillation, with one of the snapshots being the teacher. Finally, we compare the robustness of the student MLPs with the MLP trained without knowledge distillation by measuring their accuracy on the MNIST-C dataset(translated).

As shown in Figure 2, almost all teacher CNNs are able to at least improve their students' performance, but the weaker teachers at around 2500 iterations produced the best student performance on the translated MNIST-C images. The MLP trained without knowledge distillation can only achieve 34% accuracy on MNIST-C whereas the student MLP trained with weaker teachers

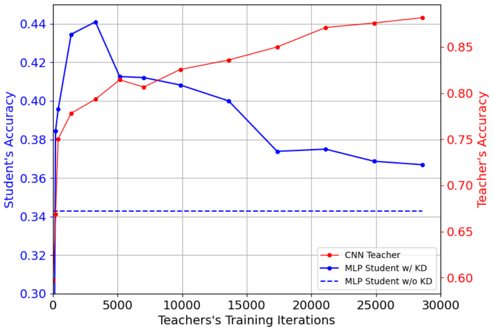


Figure 2: Student and teacher's accuracies on MNIST-C vs. teacher's training iterations on MNIST. Here, the iteration is measured as the number of batches the teacher has seen. With a batch size of 128, we take snapshots of the teacher model every 50 iterations in the first 500 iterations. Then, we increase to every 500 iterations.

can achieve 44% accuracy, which is approximately 10% boost. One might think that the teacher CNN is overfitting the training data. However, as can be observed in fig. 2, the teacher CNN actually gets more robust because it's accuracy on MNIST-C is almost monotonically increasing as the training proceeds.

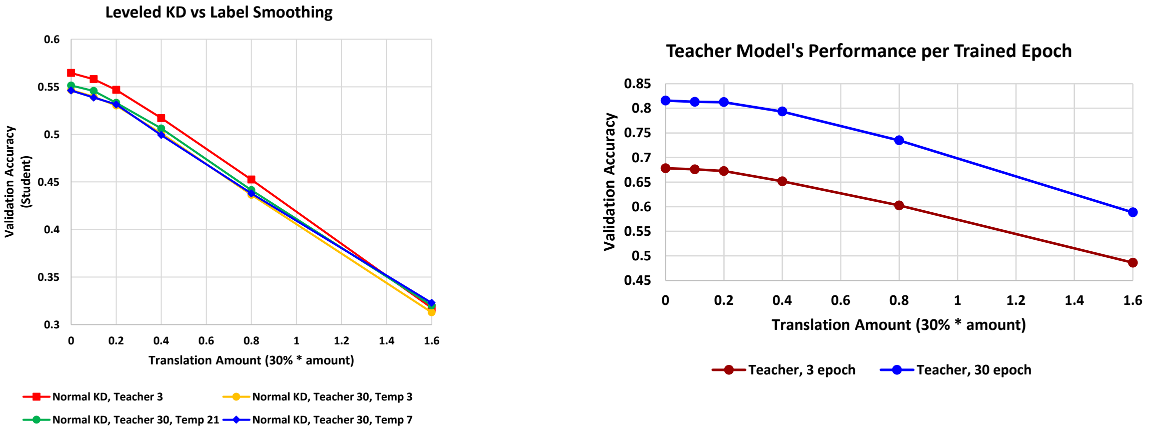
The result signals that matching the *level* of a student by using weaker teachers allows the student to learn better. This can be interpreted as, say the student model is an elementary student. Then, an elementary teacher would better at teaching that student than a college professor.

4.2 Improving the knowledge distillation scheme

Building up from our previous result, we want to understand why the weaker teacher is performing better and see if we can improve upon it. We noticed a similarity between using a weaker teacher versus label smoothing technique, so we first compare the performance of these two methods. Then, we try to improve the distillation scheme by introducing feature matching loss. For each of these experiments, the student MLP model is trained for 20 epochs.

4.2.1 Weaker Teacher vs Label Smoothing

Strong teachers generally produce more confident predictions. Sometimes, these predictions are too confident that it isn't too different from the one-hot label. Because of this, in basic knowledge distillation scheme, we use label smoothing to convey more *knowledge* to the student. We artificially modify the τ as in Equation (2) to force the teacher's prediction scores to be more evened out. Weaker teachers, due to their tendency to produce less confident predictions than strong teachers, do this more naturally. We believe this 'natural' label smoothing effect by weaker teachers replaces the need for label smoothing in knowledge distillation. To verify our claim, we compared the performance of a MLP model trained with weaker teacher versus the model trained with stronger teacher plus label smoothing. The models are trained on CIFAR-10 dataset. The result is shown in Figure 3. It shows that the model trained with weaker teacher(red line) outperforms the same model trained with stronger teacher with varying degrees of label smoothing. This result confirms that the 'natural' label smoothing effect by weaker teacher is superior than traditional label smoothing in knowledge distillation.



(a) Comparison of model trained with weaker teacher with a model with stronger teacher with varying degree of label smoothing

(b) Performance of teacher models used in this experiment

Figure 3: Comparison between weaker teacher and label smoothing of a stronger teacher

4.2.2 Feature Matching

Now that we understand that knowledge distillation can potentially transfer inductive bias, we want to see if we can improve the distillation scheme by introducing feature matching. As shown in Figure 1, we introduce another loss component which allows the student model to match the intermediate feature space of a teacher model. Then, we experimented on different combination of these loss components. Table 1 shows the combination of the loss components used in our experiment.

Type	Loss components
No KD	Ground Truth
Normal KD	Ground Truth, Teacher's Prediction
Type-1 KD	Ground Truth, Teacher's Feature Space
Type-2 KD	Ground Truth, Teacher's Prediction, Teacher's Feature Space

Table 1: Different combination of loss components

The experiment result in Figure 4 shows two main results:

(1) Type-1 Knowledge Distillation(yellow line) outperforms all other types of loss

(2) The model trained without knowledge distillation(green line) suffered more from the traslational augmentation.

Result (1) indicates that introducing feature matching not only allows the student model to exhibit more of the teacher model's inductive bias, but also potentially improves the student model's overall performance. In addition, result (2) confirms that knowledge distillation can indeed transfer inductive bias from the teacher model to a student model as the student trained with knowledge distillation performed better at correctly classifying translated images compared to the student model without knowledge distillation.

In order to understand how much effect the feature matching has on transferring the inductive bias, we trained a model by varying the parameter α. Result from this experiment is shown in Figure 5. We can observe that changing the α makes the model more susceptible to translational augmentation. This shows that the models trained with more weight on ground truth loss(bigger α) than knowledge distillation loss are displaying less translational equivariance–inductive bias of the teacher model. Therefore we can see that the feature matching is significant in improving the transfer of inductive bias to the student model.

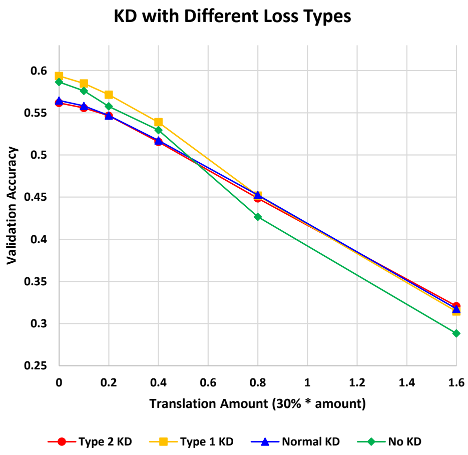


Figure 4: Performance comparison for different knowledge distillation loss types

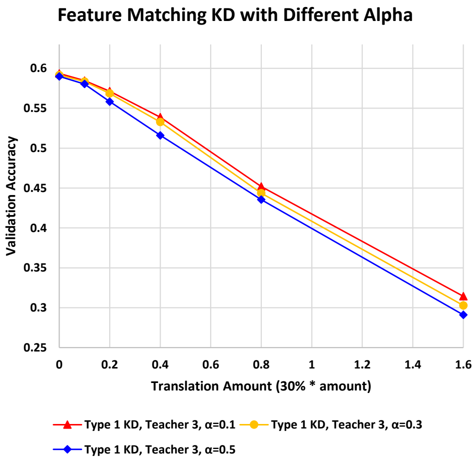


Figure 5: Achieved accuracy for Type-1 knowledge distillation with different α

5 Conclusion

Our project showed that knowledge distillation can help a simple student model to learn from a more sophisticated teacher model's inductive bias and therefore improve the student model's performance. We verified this by training a CNN model and making the simple MLP model to learn the translation invariance from the teacher model's results through the transfer of the inductive bias. Furthermore, we conducted two main experiments and found that training with a weaker/leveled teacher model and applying feature matching could improve the transfer of inductive bias by a noticeable margin while also boosting the performance of the student model.

In general, knowledge distillation is a method that has a large potential to be further deployed in the industry as it allows a simple model to learn from a large pretrained model under constrained resources. Our research verify the effectiveness of Knowledge distillation and proposed two ways: training a weaker teacher and applying feature matching to further improve the simple model performance.

Author Contributions

C. W., D. F., G. Z., J. P., Z. W. conducted experiments and analyzed the results. C. W. designed the data management pipeline. D. F. designed the knowledge distillation training pipeline, ran experiments and wrote the proposed method part. G. Z. reviewed and edited the report and presentation outline. J. P. compared weaker teacher with label smoothing, designed feature matching pipeline and wrote the second experiment part. Z. W. designed the overall pipeline, ran and wrote the first experiment.

References

[1] Samira Abnar, Mostafa Dehghani, and Willem H. Zuidema. Transferring inductive biases through knowledge distillation. *CoRR*, abs/2006.00555, 2020.

[2] 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*, KDD '06, page 535–541, New York, NY, USA, 2006. Association for Computing Machinery.

[3] Nicholas Frosst and Geoffrey E. Hinton. Distilling a neural network into a soft decision tree. *CoRR*, abs/1711.09784, 2017.

[4] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[5] Jangho Kim, Yash Bhalgat, Jinwon Lee, Chirag Patel, and Nojun Kwak. Qkd: Quantization-aware knowledge distillation, 2019.

[6] Jakob Lindqvist, Amanda Olmin, Fredrik Lindsten, and Lennart Svensson. A general framework for ensemble distribution distillation. *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep 2020.

[7] Xuan Liu, Xiaoguang Wang, and Stan Matwin. Improving the interpretability of deep neural networks with knowledge distillation. *CoRR*, abs/1812.10924, 2018.

[8] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information, 2015. cite arxiv:1511.03643.

[9] Tom M. Mitchell. The need for biases in learning generalizations. Technical report, 1980.

[10] Rafael Müller, Simon Kornblith, and Geoffrey Hinton. *When Does Label Smoothing Help?* Curran Associates Inc., Red Hook, NY, USA, 2019.

[11] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3967–3976, 2019.

[12] Zhiqiang Shen and Eric P. Xing. A fast knowledge distillation framework for visual recognition. *CoRR*, abs/2112.01528, 2021.

Project resources

- Github: `https://github.com/Wayne2Wang/Knowledge-Distillation`