Transfer Learning and Fixmatch on CIFAR10

Victor Bros **Shuo Chen**

Abstract

Deep Convolutional Neural Networks are widely used in Computer Vision applications to perform complex tasks. However a restriction of the amount of available images may severely hinder the performance of such complex systems. This issue can be tackled with the help of several techniques. The first training method presented in this paper is Transfer Learning. This technique gives the possibility to transfer the knowledge learned on a wide data set to another problem with limited data samples. The second method is a Semi-Supervised Learning training which exploits unlabeled resource for a supervised learning, FixMatch. Hence this method can significantly increase the amount of data to be used in combining two Semi-Supervised techniques : Pseudo-labeling and Consistency Regularisation. In experimenting with a restriction on image resources, we achieved a performance of 92% classification performance with Transfer Learning on CIFAR10 and 89% classification performance with FixMatch.

Keywords: Convolutional Neural Network, Transfer Learning, Semi-Supervised Learning, Fixmatch

1 Introduction

For the last decade, Deep Neural Networks have received a lot of attention for Computer Vision tasks. The reason of their success seems to be their empirical linear scalability between the size of the data set and their performance. However their success is based on supervised learning which requires labels with the data and this resource may be expensive. Whether it be due to the amount (ImageNet[1] with more than ten millions of images) or due to their complexity (Breast Cancer Wisconsin data set[2]), it requires time or expertise, sometimes both, to constitute a clean data set. It is then interesting to reduce this dependence.

Recently, a wide range of experiments on Deep Networks have been conducted in using a limited amount of resource. Two major approaches have emerged to train Deep Convolutional Neural Network with a restricted amount of images.

Transfer Learning can be roughly viewed as transferring the knowledge learned on a task to a new similar task. The application to Neural Networks is to retrieve the architecture of a successful model and its weights, and then fine-tune them to fit for the new problem. In this report, we studied the characteristics of the Transfer Learning that have major impacts on the classification performance.

The second approach, FixMatch[3], is categorized as a Semi-Supervised Learning technique. The core idea is to use unlabeled data as a resource for the training in generating pseudo-labels. FixMatch has the particularity to generate augmented images to perform the prediction of the label. With consistency regularisation, the training may include the augmented images and their pseudo-labels in the training set. In this report, we will experiment with the capacity of such systems and their robustness on the ratio of unlabeled images among the annotated data.

1.1 Contribution

This report showcases two different experiments to improve performance of systems with limited training resources. Thus we performed a study on the capacity of learning with Transfer Learning, depending on the amount of knowledge transferred from the pretraining.

Then we performed a study on the ratio of unlabeled data to use in the training with Fixmatch method. We will then try to infer the different restrictions on the use of unknown knowledge or resources to improve the learning capacity of classification models.

1.2 Outline

The remainder of the report is structured the following way: In Section 2 we introduce the related work in this field to better conceptualise the approach and the reasoning. We will then present the methods we used in the experiments we conducted in Section 3. After obtaining a firm understanding of our approach, in Section 4, we will illustrate our obtained results and analyse them. Finally, in Section 5, we will conclude this paper by reflecting on our findings and summarising the key takeaways of this report.

2 Related Work

2.1 Transfer Learning

Transfer Learning has seen significant successes in numerous real-life applications. This training method has the purpose to reduce the serious threat of data dependency on the capacity of learning of Deep Networks. A survey on Deep Transfer Learning[4] showcases four different categories for Transfer Learning algorithms that tackle this issue. They are categorized as follows:

- Instance based : These algorithms utilize instances in the source domain by defining appropriate weights to the data depending of the relevance for the new task.
- Mapping based : The purpose is to map instances from the source domain and the target domain into a new data domain that will have a better similarity for the new task.
- Network based : It reuses parts of the network pretrained in the source domain, especially the feature extractor, and apply it to the target domain.
- Adversarial based : It uses adversarial techniques to find transferable features that are suitable for the source domain and the target domain.

Considering all the previous possibilities, the one with the most success and the better reliability is to transfer parts of the network's trained parameters to the new task. This will be our approach in this report.

2.2 Semi-supervised learning

Semi-supervised learning algorithms represent a middle ground between supervised and unsupervised algorithm. It is a mature field with a huge diversity of approaches. The recent success of MixMatch[5] and FixMatch has reopened the possibility to use these techniques to improve performance of training with unlabeled data for a classification task.

First, MixMatch was able to reduce significantly the error rate on the canonical data sets of the Virtual Adversarial Training (reducing by factor 2 to 4 on CIFAR10 depending on the ratio of labeled images). MixMatch achieved this performance in augmenting stochastically the unlabeled images with AutoAugment, finding automatically the best augmentation policy depending on the data. The algorithm is as follows:

- For each unlabeled images, we generate K different augmented images.
- Each augmented image's label is predicted with the current network and the average of the K predictions is sharpened in adjusting the distribution's temperature.
- Finally the cross-entropy is computed between the pseudo-labeled images and the labeled images to perform the training of the network.

The second algorithm, FixMatch, has this same principle to augment the unlabeled images and use their predicted labels for the training. The core idea of data augmentation in the algorithm is to weakly augment the image to perform the prediction of the label on it. If the confidence on the label is higher than a defined threshold, the image is used in the training iteration. However to perform the loss computation, the image is strongly augmented with RandAugment, which chooses among several data enhancement to transform the image. Fixmatch achieved to reduce the error rate on CIFAR10 with 250 labels from 11% for MixMatch to 5%.

3 Proposed methods

3.1 Technologies and data

For these two experiments, we have implemented the solutions using the Pytorch framework 1.7.0 with Python 3.8. The computation has been made using GPU Tesla K80-12GB on Google Colab. Hence the computation was restricted by the resources in time and in space, in the spirit of the presented experiments.

The data set used to conduct the experiments is CIFAR10. This collection is composed of 60,000 32x32 images, equally divided in 10 classes. It has been chosen for its low complexity, to check quickly what works or what does not work. However we will see the implication of this low complexity on the results.

For both experiments, the images have been normalized with the mean and the standard deviation of the data set, CIFAR10.

3.2 Transfer learning

Our experiment on Transfer Learning has the purpose to compare the different possibilities of training from an already pretrained network. We retrieve the trained weights of a Resnet convolutional architecture on the ImageNet data set[6]. We then train the model on the new CIFAR10 data set depending on the approach. In order to compare the performance, we conduct four training with the same hyper-parameters (described in section 4. The four different training are as follows:

- From an empty Resnet architecture without pretraining, all the layers are trained.
- From a fully trainable architecture with pretrained weights, all the layers are trained.
- From a partially frozen feature extractor, all but the batch normalization layers are trained.
- From a frozen feature extractor, only the classifier layers are trained.

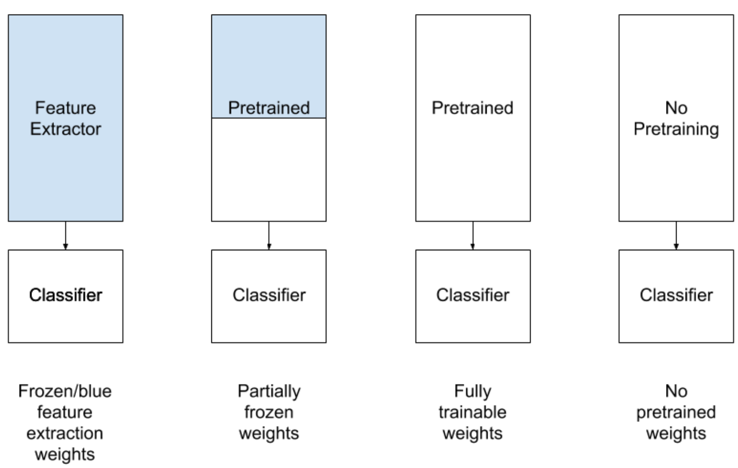


Figure 1: Representation of the different training possibilities for Transfer Learning (frozen layers in blue)

3.3 Fixmatch

The intuition behind FixMatch is to combine pseudo-labeling and consistency regularization to perform the data augmentation. The idea is to add unlabeled images to the training set when their prediction by the model is strongly confident for one label. This may mean that the image is close enough to the already seen data to be able to infer a reliable label on it.

The implementation is divided in two different parts. The first part is the augmentation of the unlabeled data. There are two different augmentation methods used in the experiment. The first one, performed before the inference on the label, is the weakly augmentation. It consists in using two random transformations on the unlabeled image, a random Horizontal Flip with a probability of $p = 0.5$ and a random translation vertical and horizontal of 12.5% of the image size. The second augmentation is a RandAugment technique. Two among the fourteen possible transformations are randomly selected for each image: identity, rotate, posterize, sharpness, translate-x, translate-y, auto-contrast, solorize, contrast, shear-x, shear-y, equalize, colorize and brightness. Also, added to the two random enhancements, there is a random Cutout that is applied to the strongly augmented image.

Figure 2: Illustration of the augmented images

We then use the principle of consistency regularization to determine if we use the pseudo-labeled image in the training. For that, we determined a threshold τ to compute the loss value :

$$\mathcal{L} = \frac{1}{\mu B} \sum_{b=1}^{\mu B} \mathbb{1}(max(q_b) \geq \tau) H(\hat{q_b}, q_b) \tag{1}$$

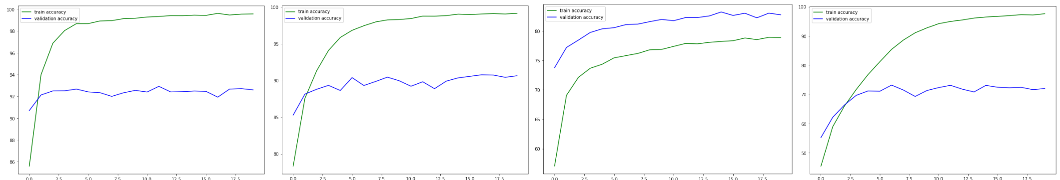
We will use this loss calculation to train the wideresnet architecture[7] with a Stochastic Gradient Descent Optimizer with momentum $\rho = 0.9$, using Nesterov Accelerated Gradient method. The learning rate is scheduled as $\eta = \eta_0 \times cos(\frac{7\pi k}{16K})$, with an initial learning rate $\eta_0 = 0.03$, k the current iteration and K the number of iteration.

4 Experimental results

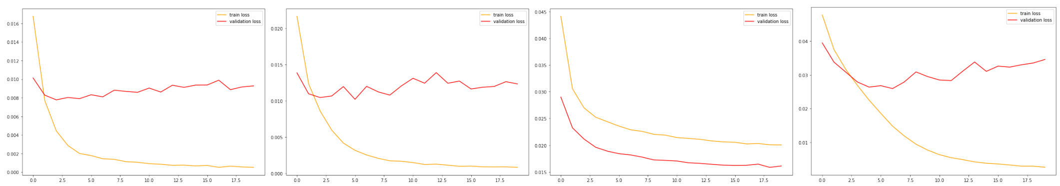
4.1 Transfer Learning

We evaluate the performance of the different Transfer Learning training with the same optimisation algorithm and hyper-parameters. We will also analyse the influence of this choice. The optimizer is Adam with a learning rate $\eta = 1e-3$, on 20 epochs of 32 images.

The results are as follows:



(a) accuracy-all layers-**92%** (b) accuracy-partial-**90%** (c) accuracy-frozen-**78%** (d) accuracy-no pretraining-**72%**



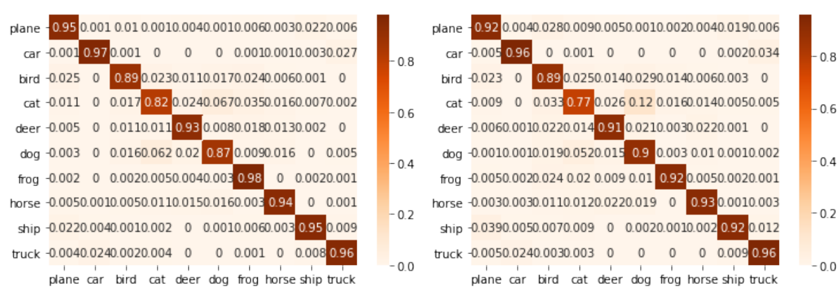
(e) loss-all layers (f) loss-partial (g) loss-frozen (h) loss-no pretraining

Figure 3: Graphs of accuracy and loss of the different training configurations with their accuracy on the test set

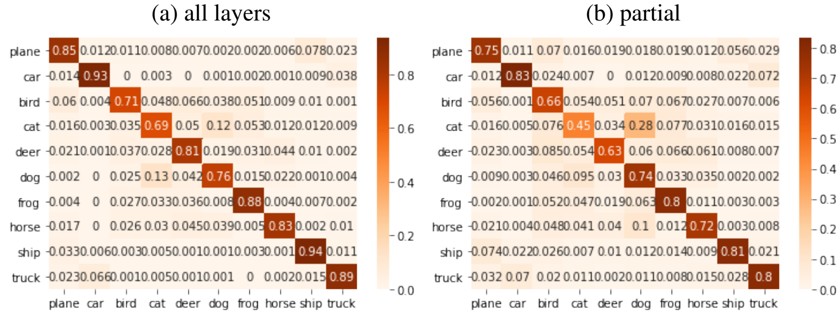
From these results, we can see that already pretrained models are faster to reach high performance. Also their capacity seems higher, even reaching 92% accuracy in fine tuning all the layers. Hence we can see that the knowledge retrieved from ImageNet is helpful and is transferred to be able to classify fairly CIFAR10.

However all but the model with frozen features extraction layers are over-fitting on the data. This is understandable since the architecture has a high-complexity to be able to learn the classification task on ImageNet. We could then use regularisation techniques to improve the performance. Also we can see that freezing the layers act as a regularisation technique in preventing to fit the model too much on the training data. The other option is to design a more appropriate architecture such as ResneXt[8].

In order to analyse in more details the performance, we have computed the confusion matrix of each training:



(a) all layers (b) partial



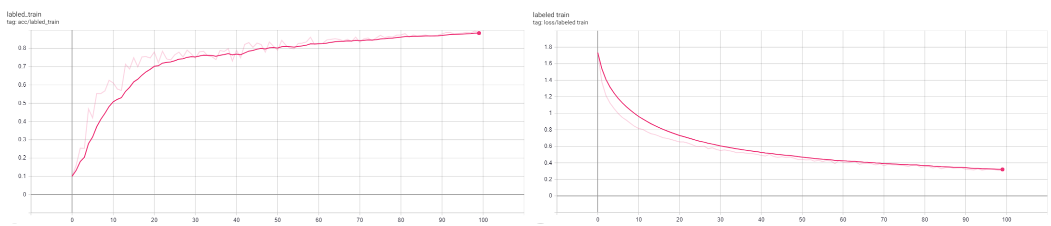
(c) frozen (d) no pretraining

Figure 4: Confusion matrices of the different training configurations

For all the training, the main category where their error is high comparing to the other is the *cat* class. However this difficulty really appears without pretraining, confusing it with *dog* often. This is understandable since there are both animals with similar body type (four legs, fur, etc). However this error is shared between all the models, so Transfer Learning can significantly improve the convergence of the training but maybe the intrinsic difficulty of the data set (separation between *cats* and *dogs*) remain.

4.2 Fixmatch

We evaluate the performance of Fixmatch with the total epochs 100, threshold $\tau = 0.95$, ratio $\mu = 0.3$, on 100 epochs.



(a) accuracy on test set-**89%** (b) loss on train set

Figure 5: Graphs of accuracy and loss in FixMatch

From the results, we can see that the semi-supervised learning performed pretty well even though it has less labelled data than supervised learning. It achieved 80% accuracy after 50 epochs, and improved a bit during the last 50 epochs.

We produced the confusion matrix also when we test the model on the test set, from which we can tell that the problem in previous transfer learning, the animals including dogs and cats and be easily confused, still exists, but for the vehicles, the network tends to have a better performance on them.

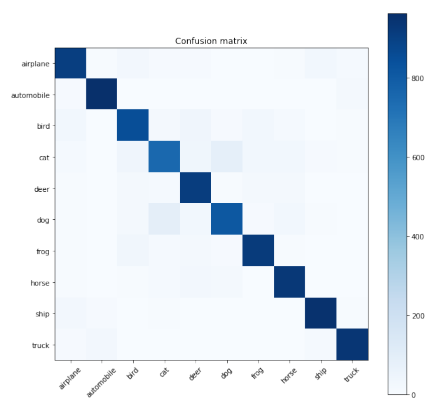
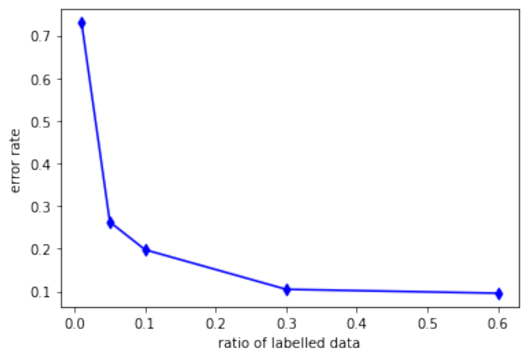


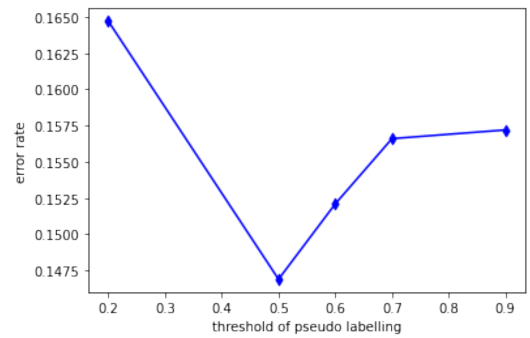
Figure 6: confusion matrix on the test set

To have a better understanding of how different factors impact on the performance of FixMatch, we applied varying ratio of labeled images and threshold of pseudo labeling on FixMatch, performing an extensive study on that.

Firstly, we set the ratio μ ranging from 0.01 to 0.6, with threshold $\tau = 0.95$, on 100 epochs, to see how different ratios impact on the network. Then, we changed the threshold ranging from 0.2 to 0.8 with ratio $\mu = 0.3$ and on 50 epochs. The result shows below:



(a) Varying the ratio of unlabeled data



(b) Varying the threshold of pseudo labeling

Figure 7: Plots of different factors' impact on FixMatch

From the plots, we can tell as the ratio goes larger, the semi-supervised learning is closer to supervised learning and more images with labels are fed to the network, thus accuracy increases. In realistic application, we can search for a optimum point which has a relative higher accuracy but with a lower cost of labelling, for example, the optimum point should be around 0.3 in our experiment.

As the threshold goes up, the error rate first drops down but later on it goes up. The reason could be at first, many false labeled images are introduced into the network, which give the network false information. Then as the threshold increases, the pseudo labelling becomes more reliable, so the performance gets better. Later, as the threshold continued to increase, there are less augmented images with high reliability, and the performance becomes a bit worse again. But threshold impact on error rate is slight when compared with ratio impact.

5 Conclusion

Our report examined two different approaches to solve the classification problem of CIFAR10.

The proposed method of Transfer Learning showcased the possibility to transfer the knowledge between similar tasks. However the results indicates that the complexity of the data sets have a major component in this transfer. Hence using a model with a too high complexity may imply an over-fitting even with high performance. It is a balance to find between the amount of transferable knowledge and the ability to generalize.

The second method, FixMatch, showcased the ability of using unlabeled images as part of the training set in augmenting them. The results are pretty promising even with a low ratio of labeled images. However the results highlight the importance of relevant labeled images to have a wide representation of the class.

Overall it is possible to train models with limited training data to solve complex tasks with complex structures such as Deep Convolutional Neural Network. Nevertheless one should take into account the balance between the knowledge of the model and the relevance of the data used to reach high performance.

References

[1] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115, 09 2014.

[2] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[3] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. 2020.

[4] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. 2018.

[5] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. 2019.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. 2015.

[7] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. 2016.

[8] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. 2016.