COSE474-2022: Final Project Final Report
Analyzing Transfer Learning For Image Classification

2018320147 Seongjoon Mun

1. Introduction

Deep learning architectures such as CNN have accomplished successes in image classification tasks, and it has been shown that these models perform best when there is enough labelled data for the task. However, it is very expensive to collect abundant labelled data and some tasks such as rare disease diagnosis have limited training data. As a result, we may want to learn from a small number of training data and transfer learning can greatly improve performance in this situation. In this report, I will introduce recent progress in deep transfer learning for image classification and find which areas could be improved.

1.1. Motivation

With the increasing demand for the application of modern CNN models to real world areas, studies in transfer learning has increased at a rapid pace. Also, it is important to check the current state of the field and make suggestions about how to progress the field. Even though there are many surveys about transfer learning in specific areas, there are not much studies that focus on transfer learning for general image classification. Therefore, I believe that it is important to combine all the knowledge in the field of transfer learning and analyze the results. By summarizing current knowledge in the area and categorizing specific cases of transfer learning, we can gain a good understanding of when transfer learning performs best and when it does not perform as expected.

1.2. Problem definition

Transfer learning can be categorized by the task. To be specific, the performance of transfer learning depends on the relevance between data that is used to pretrain the model (source dataset) and the target dataset. If the source and target dataset are more closely related, transfer learning will show better performance. On the other hand, if the source dataset is not well related to the target dataset the model can be negatively impacted by pretraining (negative transfer). Therefore, it is important to find methods to improve transfer learning in diverse situations because it is difficult to always find a target dataset that is large and similar to source dataset.

1.3. Concise Description of Contribution

In the first final project proposal, my topic was semi-supervised semantic segmentation. However, in October, I became interested in the idea of transfer learning and started reading papers related to transfer learning on arxiv.org. Also, I wanted to find out what kind of factor changes can improve the performance of transfer learning. So in November, I thought about how to solve this question. Afterwards, I tried to find different target datasets, source codes I can use and designed the experiment. In December, I got the results and started writing the report.

1.3.1. Related Works

Transfer Learning. During transfer learning, a model is trained on a task for which more data is available and the trained weights are used to initialize a model for the target task. It is common to pretrain a model on a very large dataset such as ImageNet, and then use the model either as an initialization or a fixed feature extractor.

Deep transfer learning for image classification: a survey (Plested et al., 2022)) is a general transfer learning survey related to image classification. The figures show that deep learning performance scales with the size of the dataset and the model. Also, initializing weights of the model with weights that have been trained on a large source dataset results in more stable updates.

Characterizing and Avoiding Negative Transfer (Wang et al., 2019) reveals that if the source dataset is not well related to the target dataset, the model can be negatively impacted by pretraining. The higher the divergence between these values, the less information there is in the source domain that can be exploited to improve performance in the target domain.

2. Methods

2.1. Significance and novelty

In my report, I will analyze four circumstances where the target dataset is large and similar to the source dataset (pre-trained model's dataset), target dataset is large but different from the source dataset, target dataset is small and similar to the source dataset, and target dataset is small and different from the source dataset. Then I will try transfer learning for each circumstances with different learning rate, different number of model's layers, and layer freezing and find out which setting is the best. I am conducting this experience because there may be situations where we are interested in training a model on unexpected environments, sometimes on a small amount of target data, and it is important to alleviate the side effects of transfer learning such as negative transfer.

2.2. Main figure

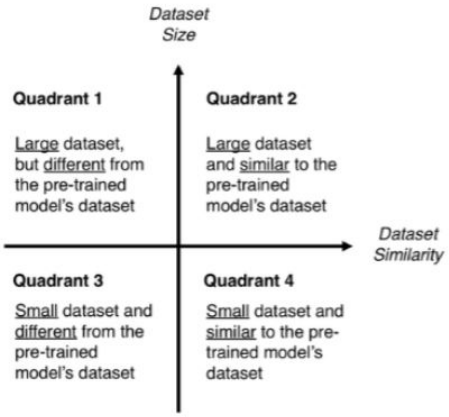

Figure 1. This figure shows the method of testing transfer learning with four datasets with different sizes and similarity. The details are explained in the significance and novelty part.

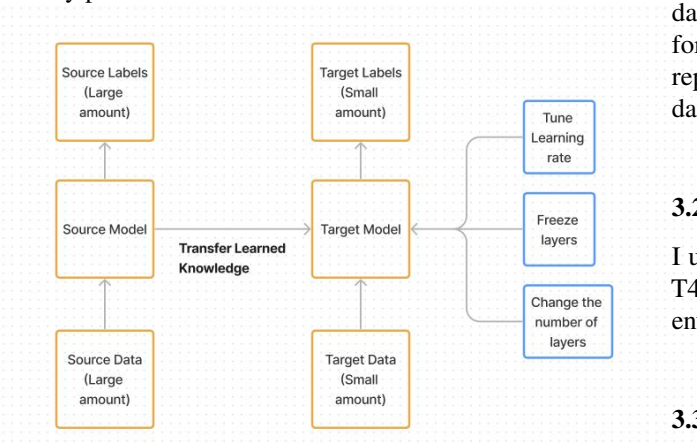

Figure 2.

2.3. Reproducibility(Algorithm)

Figure 2 shows the method of basic transfer learning adjusted with different learning rate, different number of layers, and whether the layers except for the final one are frozen or not.
Algorithm : First, the target dataset is transformed using RandomResizedCrop, RandomHorizontalFlip, and Normalize. Then, the pretrained CNN model is brought and optimizer, loss function, learning rate decay, and epoch is established. Afterwards, the model is trained until the end of epoch and prints out the best test accuracy among the epochs. Each epoch has training and test phase. Lastly, the model visualizes some images and the predicted label. The algorithm is same when we freeze all the layers except for the final layer but parameter requires-grad needs to be set to false so that gradient computation is disabled.

3. Experiments

3.1. Dataset

The model is pretrained on Imagenet 1000, which is a dataset that has 1,000 classes with 1,281,167 training images, 50,000 validation images and 100,000 test images. Also, I conducted experiments on four different datasets, which I downloaded from Kaggle. The first dataset is ant and bee, which consists of 400 images for training and 200 images for testing in total. This dataset represents small target dataset that is simlar to source dataset(pre-trained model's dataset). The second dataset is cat and dog, which consists of 4,000 images for training and 2,000 images for testing. This dataset represents large target dataset that is simlar to source dataset. The third dataset is glioma tumor and no tumor, which consists of 400 images for training and 200 images for testing. This dataset represents small target dataset that is different from source dataset. The fourth dataset is car and bike, which consists of 4,000 images for training and 2,000 images for testing. This dataset represents large target dataset that is different from source dataset.

3.2. Computing Resource

I used the GPU on google colab pro version(GPU : Tesla T4), and google colab uses virtual machine with Linux environment. Also, the pytorch version is 1.12.1.

3.3. Experimental Design and Setup

The experiment is conducted on 4 datasets under 8 different conditions(Different CNN architecture, different

learning rate, freezing the layer or not). For the model, resnet18 and resnet50 pretrained on Imagenet 1000 are used respectively. Also, the learning rate of the model is set to 0.001 and 0.0001 respectively. The learning rate is decayed by a factor of 0.1 every 7 epochs. In addition, experiment is both conducted with freezing all the layers except for the final layer during training stage and not freezing the layer. Adam optimizer is used for updating parameters and cross entropy loss is used for computing the loss. Also, some data transformations(RandomResizedCrop,RandomHorizontalFlip,Normalize) are applied to the input images. The epoch of the experiment is 25 and I picked the best accuracy among the epochs when comparing the results.

3.4. Quantitative results

The SOTA of ant and bee classification(table1) is 0.99. Compared to our best method(Freezing, Resnet-50,learning rate 0.001), SOTA outperforms us by 0.035%. Also, the SOTA of cats and dogs classification(table2) is 0.991, and this outperforms our best method(Freezing, Resnet-50,learning rate 0.001) by 0.007%. There is no public SOTA for table3 and 4 since these are datasets that I arbitrarily collected. For the baseline, CNN model without transfer learning is used. Our method with best accuracy on table1 outperforms the baseline on same conditions by 0.307189%. The best method on table 2 outperforms the baseline by 0.355%. The best method on table 3 outperforms the baseline by 0.155%. The best method on table 4 outperforms the baseline by 0.027%, but it shows more difference on other conditions.

3.5. Qualitative results

An example of qualitative results on ant and bee classification test set. Backbone is Resnet-18, and learning rate is 0.0001. The left image is the result of training without transfer learning, while the right image is training with transfer learning. The non transfer learning method wrongly classified the bee as ant, while the transfer learning method correctly classified the bee. This shows that transfer learning method is better when classifying images.

Freezing	Backbone	LR	Accuracy	Baseline
Y	R18	0.001	0.947712	0.6732
N	R18	0.001	0.810458	0.712418
Y	R18	0.0001	0.8954	0.620915
N	R18	0.0001	0.947712	0.784314
Y	R50	0.001	0.954248	0.647059
N	R50	0.001	0.751634	0.699346
Y	R50	0.0001	0.921569	0.666667
N	R50	0.0001	0.941176	0.673203

Table 1. Classification of ant and bee

Freezing	Backbone	LR	Accuracy	Baseline
Y	R18	0.001	0.982	0.636
N	R18	0.001	0.77	0.685
Y	R18	0.0001	0.982	0.601
N	R18	0.0001	0.98	0.784
Y	R50	0.001	0.984	0.629
N	R50	0.001	0.657	0.612
Y	R50	0.0001	0.982	0.612
N	R50	0.0001	0.976	0.705

Table 2. Classification of cat and dog

Freezing	Backbone	LR	Accuracy	Baseline
Y	R18	0.001	0.715	0.525
N	R18	0.001	0.665	0.63
Y	R18	0.0001	0.72	0.585
N	R18	0.0001	0.71	0.645
Y	R50	0.001	0.73	0.59
N	R50	0.001	0.625	0.61
Y	R50	0.0001	0.73	0.575
N	R50	0.0001	0.73	0.63

Table 3. Classification of glioma tumor and no tumor

Freezing	Backbone	LR	Accuracy	Baseline
Y	R18	0.001	0.993	0.755
N	R18	0.001	0.981	0.956
Y	R18	0.0001	0.984	0.693
N	R18	0.0001	0.996	0.969
Y	R50	0.001	0.995	0.676
N	R50	0.001	0.973	0.937
Y	R50	0.0001	0.993	0.625
N	R50	0.0001	0.99	0.939

Table 4. Classification of car and bike

3.6. Figures(Tables)

Freezing is about whether I froze all the layers except for the final layer of the model during training. Y is freezing and N is non freezing. Backbone is the pretrained model that I used in transfer learning, R18 is Resnet-18 and R50 is Resnet-50. LR is learning rate.

Analysis of the table : Transfer learning with every datasets and every conditions showed better performance than model with no transfer learning. Also, it is clear that layer freezing is effective in table 2 and 3, which means that layer freezing except for the final layer during training is effective when the source and the target data is similar

and the target data is large, and also when the source and the target data is not similar and the target data is small. Moreover, by looking at table 1 and 2, we can see that optimal learning rate is lower when the source and target data is similar (with non freezing method). However, it was asserted in previous research that less layer shows better result when the source and target data is not relevant, but my result was different. Also, previous research stated that transfer learning hyperparameters have less impact on performance as the size of the target dataset increases, but this was only true for table 4 which had big dataset size and least accuracy difference among different parameters.

3.7. Discussion why the proposed method is successful or unsuccessful

Unsuccessful factor : First, I tried to find a target dataset that is large enough and not similar to the source dataset(Imagenet 1000), but it was difficult to find such images so I chose the car and bike dataset. However, it was difficult to judge whether the car and bike dataset was totally different from Imagenet. Also, car and bike images are easy to distinguish compared to other datasets so the classification accuracy(table4) was too high, which made it difficult to analyze table4.

4. Future Direction

As I mentioned in the introduction, it is expensive to collect large amount of labelled data before training. So, it is important to use transfer learning properly, especially when the target data is limited. I have tried different learning rates, different number of layers for the model, and freezing layers during training to achieve better transfer learning accuracy in various datasets, but more factors such as activation functions, loss functions, pooling layers, and data augmentations have to be considered. Therefore, many combinations of diverse factors during transfer learning have to be tested and researched to reach better accuracy of image classification. Also, studies to minimize the effect of negative transfer is crucial.

References

Jo Plested, Tom Gedeon. Deep transfer learning for image classification: a survey. In Comput. Vis. Pattern Recog., 2022.

Zirui Wang, Zihang Dai, Barnab´as P´oczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11293–11302, 2019.

Jo Plested and Tom Gedeon. An analysis of the interaction between transfer learning protocols in deep neural networks. In International Conference on Neural Information Processing, pages 312–323. Springer, 2019.

Pulkit Agrawal, Ross Girshick, and Jitendra Malik. Analyzing the performance of multilayer neural networks for object recognition. In European conference on computer vision, pages 329–344. Springer, 2014.

Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge Belongie. Large scale fine-grained categorization and domain specific transfer learning. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4109–4118, 2018.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In Proceedings of the European Conference on Computer Vision (ECCV), pages 181–196, 2018.

Michal Heker and Hayit Greenspan. Joint liver lesion segmentation and classification via transfer learning. arXiv preprint arXiv:2004.12352, 2020.

A. Code on github

https://github.com/munjoon98/deeplearning-finalproject