Assignment 1
Transfer Learning in CNNs

Isac Paulsson & Mahmut Osmanovic

March 2025

1 Approach

1.1 Data

The data used for the transfer learning experiments consist of two image data sets. The first is the Stanford Dogs dataset [1], comprised of 20,580 labeled images, that are categorized into 120 different dog breeds. This dataset contains approximately 150 images for each class. The second dataset is the Cats vs. Dogs dataset [2], which contains 25000 images divided evenly amongst the two classes. Its images have been filtered due to a subset of them being in invalid *jpeg* format, leaving in total 23412 images. Both datasets are split into training and validation sets with a validation split size of 20%.

1.2 Model Design

A pre-designed CNN model was utilized for the experiments [3]. The total amount of model parameters are about 2.7 million. Alterations were only made to the learning rate and batch size. The learning rate was set to $1e^{-4}$ and the batch size to 64. A detailed model implementation is provided by the Keras Team. The CNN was implemented in tensorflow, version 2.14.0.

1.3 Experiment Design & Evaluation

An initial base model is trained on the Stanford Dogs dataset. Evaluation of this model is done with Categorical Cross Entropy and Categorical Accuracy. This model provides the initial weights that will be transferred to the experimental models (2-4). Additionally a baseline model is trained for the Cats vs. Dogs dataset (with random initialized weights). For the evaluation of the experimental models, as well as the baseline, Binary Cross Entropy and Binary Accuracy were utilized.

Experimental model alterations are detailed in table 1. For each configuration we train two models. In one, transferred weights are frozen. In the other, transferred weights are trainable. The accuracy and loss for both train and test sets are logged for each epoch.

Experiment	**Model alteration**
only output	Dense output layer is replaced for binary prediction
3-first-conv2d & output	Dense output layer is replaced for binary prediction, first 3 convolutional layers are re-initialized
2-last-conv2d & output	Dense output layer is replaced for binary prediction, last 2 convolutional layers are re-initialized

Table 1: Model weight alterations by experiment.

2 Insights & Analysis

Both graphs in figure 1 demonstrate the advantages of using pretrained weights. One can clearly observe that the initial model accuracy is ostensibly better in all six experiments (2,3 and 4 - frozen and and unfrozen) in comparison to the baseline. The best performing model is the one highlighted in red color (2-last-conv2d & output). Not only does it begin from the highest accuracy but also manages to obtain the highest accuracy across all epochs.

The last convolutional layer contains 1024 channels. Its size enables the model to capture a substantial amount of complexities within the new dataset. We observe that *3-first-conv2d & output* (green line) has surpassed *only output* (blue line) in performance by epoch 10. The weights being randomly reinitialized rather than finetuned on previous tasks leaves the models in a worse starting position. Hence, the initial layers have a large impact on how the activations evolve through the network.

Note that the left image is smooth without having any smoothening techniques applied to it. Since the transferred weights are frozen, there are fewer weight updates by epoch. The statistical entailment being that there are fewer combinations of weights that can result in low model performance. The principle is highlighted when comparing the variation in model performance across models with frozen and unfrozen weights.

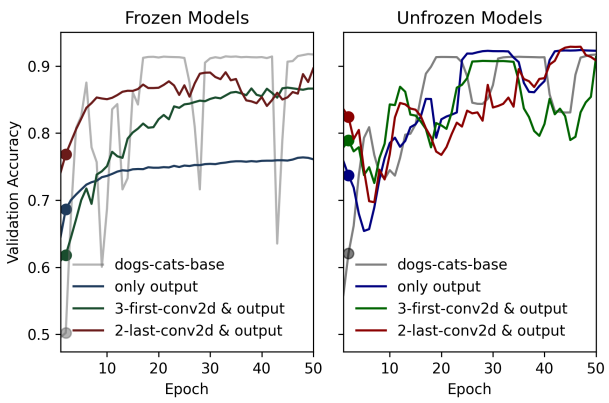

Figure 1: Validation accuracy of different transfer learning strategies. Left: models with transferred weights frozen (except dogs-cats-base). Right: models with transferred weights unfrozen (except dogs-cats-base). The right image is smoothed through the use of a sliding window of size 5. The label details what layer weights were reinitialized through *GlorotUniform*.

For instance, when comparing the performance variation of the *dogs-cats-base* model (as shown in the left image) with that of the other models.

The frozen weights effectively provide a substantial regularizing effect to our models. With fewer parameters, the risk of them settling into suboptimal values is reduced. The regularization of our models enables the use of a higher learning rate whilst maintaining model stability.

Noteworthy in the right image (figure 1) is that each of the graphs with transferred weights substantially drop in performance during the first couple of epochs. Since the output layer is randomized, the error is detected to be large, thus passing through large gradients in the backwards pass. Large gradients entail substantial updates to the transferred weights, this disrupts the internal structural dependence of our transferred weights, which in turn results in the descending accuracy.

The chaotic learning behavior of the models with non-frozen weights may be as a consequence to several factors. The primary one relating to the magnitude of the learning rate in relation to the model complexity. Large models with millions of parameters are sensitive to large gradient updates. Hence, enhanced regularization is expected to smoothed non-frozen model learning curves.

Table 2 highlights the maximum obtained validation accuracy score achieved by each model. The best performing final model for each class (frozen or not frozen weights) was the models where the last two convolutional layers and the output were re-initialized. This allowed the model to capture new insights from the Cats vs. Dogs Dataset, without obstructing the feature extraction capabilities of the transferred weights in the initial model layers.

Exp.	Freeze	Model	Test Acc.
1	False	dogs120-base	0.1407
	False	dogs-cats-base	0.9178
2	True	only output	0.7638
	False	only output	0.9244
3	True	3-first-conv2d & output	0.8678
	False	3-first-conv2d & output	0.9180
4	True	2-last-conv2d & output	0.8964
	False	2-last-conv2d & output	0.9297

Table 2: Best Test Accuracy by Model

3 Conclusions

A noteworthy insight is that the transferred weights enable the models to begin training from better initial weight configurations. Additionally, the models with sufficient capacity and more favorable initial conditions also tend to obtain marginally greater final test accuracies. Freezing weights has a regularizing effect, and enables stable learning. The base model, being trained for a sufficient amount of epochs is able to catch up in accuracy score but does not display the same stability during training. It is crucial to strike a balance between giving the model enough capacity to learn whilst preserving the structural dependencies of the transferred weights. Hence, providing additional capacity in the end of a model (rather than, say in the initial layers) is more advantageous for learning.

References

[1] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

[2] Jeremy Elson, John Douceur, Jon Gulley, and Jacob Howell. Asirra: A captcha that exploits interest-aligned manual image categorization. `https://www.microsoft.com/en-us/research/project/asirra/`, 2010. Microsoft Research.

[3] Keras Team. Image classification from scratch. `https://keras.io/examples/vision/image_classification_from_scratch/`, 2023.