Dense Residual Connection Finetuning (Review)

Tejas Gaikwad
gaikwad.2@iitj.ac.in
Department of Computer Science and Engineering.
Indian Institute of Technology, Jodhpur Rajasthan, India

CSL7370 (Dependable Artificial Intelligence)

Keywords: Transfer Learning, Finetuning, DenseNet, ResNet, DRCNet, Connection Finetuning, Sparse Connection DRCNet, Cross-Data Connection Finetuning

1. Introduction

This paper discusses a deep learning architecture for tackling the limitations of Transfer learning. Transfer learning is the fastest and easiest way to build a deep learning model by taking an available model off the shelf and adapting it to some other problem. It is done by chopping off the top layer of the already initialized model and replacing it with a randomly initialized one and then training the parameters of the top layer and keep all other fixed, especially for the models which have already trained on the data set. The top layer is the fully connected neural network and the rest layers are the feature extractors. Transfer learning is very helpful when there is a shortage of training data.

The paper discusses an algorithm named DRCNet(Dense Residual Connection Network) along with some unconventional techniques of connection-finetuning(Transfer learning)i.e. Sparse Connection Dense Residual Connection Network(SC-DRCNet).

2. Motivation

Transfer learning, a technique widely used nowadays has a lot many applications but has some limitations. When an already trained model is used for some other data-set then it is difficult for the model to adapt the intermediate layer. This paper has discussed two models that are being used for transfer learning which are Residual Network(ResNet) and DenseNet(Dense Network) and have compared them with the results obtained by SC-DRCNet.

3. Summary

DenseNet and ResNet are the most widely used CNN architectures. The authors have upgraded this architecture to enhance transfer learning. The methods used for transfer learning has challenges like, if the last few layers are trained, intermediate layers do not receive feedback while performing backpropagation for the target task. Whereas if the entire network is fine-tuned then it has a large number of parameters to train and therefore required a substantial database. The proposed algorithm is most useful when there is a minimal dataset available for training the model.

3.1. Proposed Method

Each DenseNet and ResNet have their own benefits in terms of learning the dataset. In ResNet the output of an intermediate-convolution layer is connected with immediately previous layer's output, it has skip connections as a baseline and in DenseNet, intermediate layers are connected densely to all the succeeding layers.

In the provided solution SC-DRCNet, a connection fine tunning is done for DenseNet architecture to remove the redundant layer which does not have much importance to the output. The finetuning is done by providing the Connection Strenght parameter to learn the importance of each connection. Below is the suggested solution block diagram

Connection finetuning is proposed to learn a Strenght parameter unique to each connection during the training. It's basically an indicator of a connection's contribution to the complete learning process. Finetuning connections refers to retaining or dropping a connection from the network in order to boost accuracy and robustness.

The ResNet skip connections can be written as,

$$H(x) = f(x) + x$$

Modifying the above equation with strength-connection and named as DRCNet, with each block have *x* input, the output of the last block is given as,

$$H(x_n) = f(x_n) + \frac{1}{n} \times \sum_{i=1}^{n} k_i \odot x_i$$

Further, In order to retain important connections and remove some redundant ones while training, The l_1 norm has been applied to the strength parameter and optimized with cross-entropy loss, thus giving sparsity. The loss function is given as

$$\mathcal{L}(scores, l) = -scores[l] + log(\sum_{c=1}^{M} exp(scores[c])) + \lambda ||K||_{l_1}$$

This modified loss function of the model is a combination of l_1 norm of k values and cross-entropy loss.
Scores: a final array of scores for each class obtained after the forward pass for given input *x*
M: The total number of classes
L: a class label for which loss is being calculated
λ: sparsity regularisation constant which helps to control the amount of sparsity that is imposed on the vector of all parameters
K: Strength parameters
The implementation is done with varying network depths, each model is a modification of ResNet, the number of layers is 18, 50 and 152. Each layer is further sub-divided into ResNet counterparts and is divided into 2,4 and 10 dense layers. Further, each dense layer consists of 4 or 5 basic blocks and has dense-skip connections. Average pooling is also applied to decrease the size of each feature map.

3.2. Experiments and Results

The authors have tested their proposed solution over 4 different datasets namely CIFAR10, CIFAR100, SVHN, Tiny-Imagenet. They performed 3 experiments on this dataset,

1. Same Dataset Connection Finetuning
2. Cross Dataset Connection Finetuning
3. Small Samples Cross Dataset Connection Finetuning

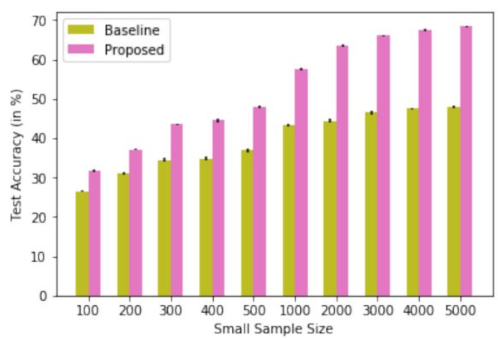
For Same Dataset Connection Finetuning, connection tuning has been performed on the above-mentioned dataset and with 3 different network depths (18,50 and 152). This connection finetuning included 3 experiments, 1) training a conventional ResNet architecture (Our Baseline model), 2) training a DRCNet architecture, and 3) training an SC-DRCNet architecture from scratch for each dataset.
There was an improvement in performance noticed because of unimportant connections. Also, it was noted that about 20-30% of all k-values are nullified which resulted. The below-mentioned table shows the performance of the proposed solution for this experiment.

Dataset		C10			C100			SVHN			Tiny-Imagenet		
Model		ResNet	DRCNet	SC-DRCNet	ResNet	DRCNet	SC-DRCNet	ResNet	DRCNet	SC-DRCNet	ResNet	DRCNet	SC-DRCNet
Network Depth	**18**	93.21	91.55	**93.88**	70.79	65.19	**72.5**	91.04	**91.4**	91.314	62.33	55.3	**62.86**
	50	93.22	92.56	**93.60**	70.82	70.7	**73.1**	91.11	87.28	**91.33**	65.13	61.76	**65.61**
	152	93.02	92.8	**93.43**	73.75	72.41	**73.91**	-	-	-	-	-	-

For Cross-Data Connection-Finetunning, a standard ResNet model is pre-trained on the Tiny-Imagenet database following which only the softmax layer is finetuned for CIFAR-10/100. Here, a boost of 5-10% was observed in test accuracies for the adaption of pre-trained networks on different targets. The below-mentioned table shows the performance of the proposed solution for this experiment.

Dataset	Total Images	Image Resolution	No. Of Classes	Train Set	Test Set
CIFAR10	60,000	32x32x3	10	50,000	10k
CIFAR100	60,000	32x32x3	100	50,000	10k
SVHN	99,298	32x32x3	10	73,257	26,032
Tiny-Imagenet	110,000	64x64x3	200	100,000	10k

For Connection Finetunning on Cross Datasets for Small Sample size, the evaluation was for the effectiveness of proposed architecture for small sample datasets. For this experiment 10k samples of images were taken to test on the baseline and proposed models. SC-DRCNet-50 yields an improvement of 5-20% over standard ResNet-50. Below is the graph for the obtained results.



4. Conclusion

The authors have performed several studies on 1) the same database, 2) cross-database and 3) random drop of skip connections. The proposed approach helped in discarding the redundant or less important residual connections and retaining the important connections with the help of connection finetuning thus boosting the accuracy and robustness.