N8104: Artificial Neural Networks

Transfer Learning

Sachchida Nand Chaurasia
Assistant Professor

Department of Computer Science
Banaras Hindu University
Varanasi
Email id: snchaurasia@bhu.ac.in, sachchidanand.mca07@gmail.com

June 25, 2023

Transfer Learning I

"The choice of CNN architecture depends on the complexity of the image classification task, the size of the dataset, the available computational resources, and the desired level of accuracy."

- *If the input data is small and simple, such as images with low resolution, then a smaller CNN architecture such as LeNet or AlexNet might be sufficient.*
- *If the input data is large and complex, such as high-resolution images or videos, then a larger and more complex CNN architecture such as VGG, Inception, or ResNet might be needed to extract relevant features.*
- *If the task involves object detection or segmentation, then architectures like YOLO, RCNN, or Mask R-CNN might be suitable.*

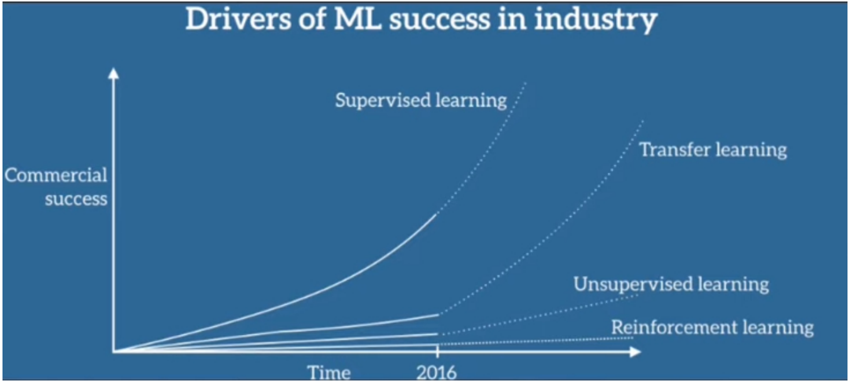
Transfer Learning II

- *If the task involves processing sequential data such as speech or text, then architectures such as Convolutional LSTM or Time Distributed CNN might be used.*
- *If the available computational resources are limited, then smaller architectures with fewer layers and parameters may be preferred to reduce training time and memory usage.*

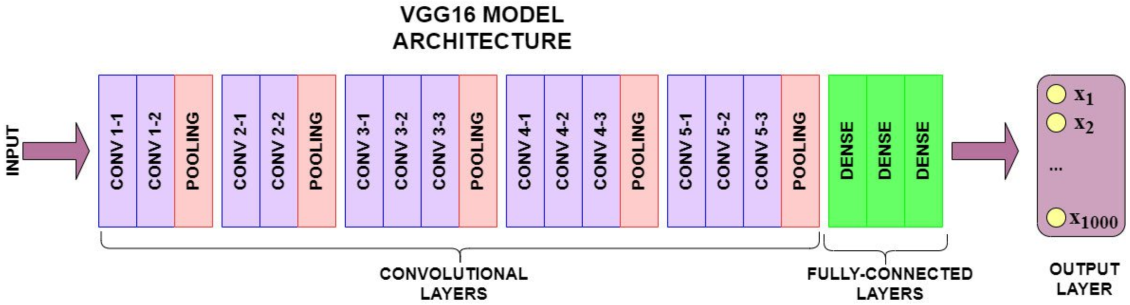
Transfer Learning III

Year	CNN	Developed by	Place	Top-5 error rate	No. of parameters
1998	LeNet(8)	Yann LeCun et al			60 thousand
2012	AlexNet(7)	Alex Krizhevsky, Geoffrey Hinton, Ilya Sutskever	1st	15.3%	60 million
2013	ZFNet()	Matthew Zeiler and Rob Fergus	1st	14.8%	
2014	GoogLeNet(19)	Google	1st	6.67%	4 million
2014	VGG Net(16)	Simonyan, Zisserman	2nd	7.3%	138 million
2015	ResNet(152)	Kaiming He	1st	3.6%	

Transfer Learning IV



Transfer Learning V

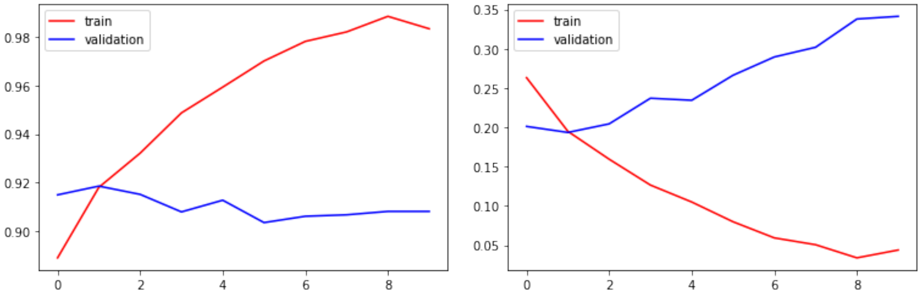


Ways of doing Transfer Learning

1. Feature Extraction
2. Fine Tuning

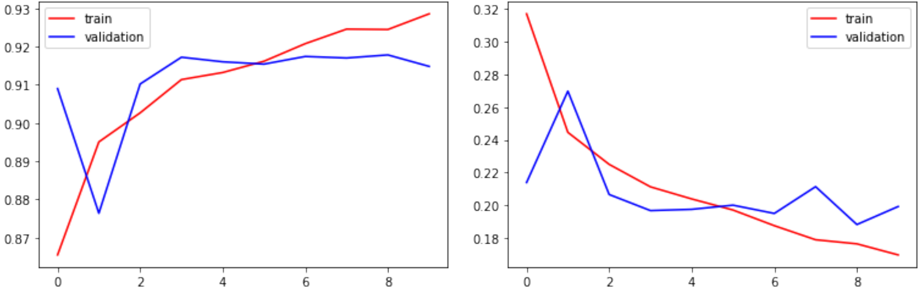
Transfer Learning VII

Without data augumentation:



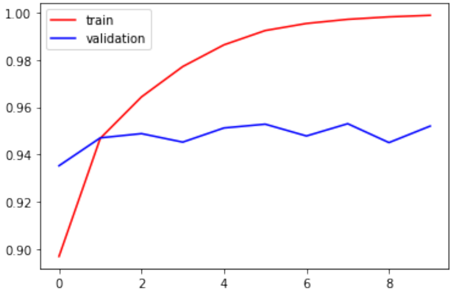
Transfer Learning VIII

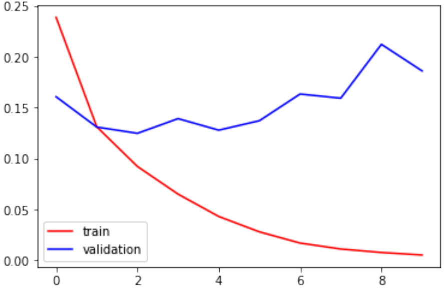
With data augumentation:



Transfer Learning IX

With Fine Tuning:





Transfer Learning X

Without Data augumentation:

Transfer Learning XI

```
model = Sequential()

model.add(conv_base)
model.add(Flatten())
model.add(Dense(50,activation='relu'))
model.add(Dense(25,activation='relu'))
model.add(Dense(1,activation='sigmoid'))
```

```
model.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg16 (Functional)          (None, 4, 4, 512)         14714688

 flatten_4 (Flatten)         (None, 8192)              0

 dense_9 (Dense)             (None, 50)                409650

 dense_10 (Dense)            (None, 25)                1275

 dense_11 (Dense)            (None, 1)                 26

=================================================================
Total params: 15,125,639
Trainable params: 410,951
Non-trainable params: 14,714,688
```

Transfer Learning XII

```
625/625 [==============================] - 2184s 3s/step - loss: 0.1740 - accuracy: 0.9285 - val_loss: 0.2103 - va
l_accuracy: 0.9096
Epoch 2/10
625/625 [==============================] - 2053s 3s/step - loss: 0.1475 - accuracy: 0.9384 - val_loss: 0.2252 - va
l_accuracy: 0.9126
Epoch 3/10
625/625 [==============================] - 2046s 3s/step - loss: 0.1284 - accuracy: 0.9457 - val_loss: 0.2423 - va
l_accuracy: 0.9086
Epoch 4/10
625/625 [==============================] - 2009s 3s/step - loss: 0.1165 - accuracy: 0.9510 - val_loss: 0.2366 - va
l_accuracy: 0.9164
Epoch 5/10
625/625 [==============================] - 2044s 3s/step - loss: 0.1052 - accuracy: 0.9561 - val_loss: 0.2661 - va
l_accuracy: 0.9140
Epoch 6/10
625/625 [==============================] - 2049s 3s/step - loss: 0.0913 - accuracy: 0.9611 - val_loss: 0.2960 - va
l_accuracy: 0.9084
Epoch 7/10
625/625 [==============================] - 2052s 3s/step - loss: 0.0783 - accuracy: 0.9681 - val_loss: 0.3780 - va
l_accuracy: 0.8966
Epoch 8/10
625/625 [==============================] - 1982s 3s/step - loss: 0.0636 - accuracy: 0.9742 - val_loss: 0.3723 - va
l_accuracy: 0.8958
Epoch 9/10
625/625 [==============================] - 2034s 3s/step - loss: 0.0606 - accuracy: 0.9761 - val_loss: 0.3662 - va
l_accuracy: 0.9080
Epoch 10/10
625/625 [==============================] - 1977s 3s/step - loss: 0.0517 - accuracy: 0.9796 - val_loss: 0.3961 - va
l_accuracy: 0.9022
```

Transfer Learning XIII