# Transfer Learning

By Anju Pandey

## What is Transfer Learning?

Transfer Learning – These two words are very descriptive by themselves. Transfer the learning. What you have learned, you are transferring that knowledge to someone else.

For example, a mom is teaching her kids how to make simple cheese pizza. Kids learn that and then they think of making different kind of pizzas with some different ingredients, different toppings, using different type of flours etc.

In the above example, mom transferred her knowledge of pizza making to her kids and kids utilized that knowledge and created something new.

### What is transfer learning in case of AI?

In case of machine or AI, a model transfers its learning to other model/s. Other model/s use/s this knowledge as base and perform the required task without going under rigorous training which they would have gone if no pre-learning was available to them.

Our problem statement is to classify correctly human expressions. There are many pretrained models available which are doing similar job with higher accuracy. We will see some of these models and will use them to build our classification model.

There are three models which we will be using: VGG16, ResNet v2, and Efficient Net.

## Brief information about VGG16, ResNet v2 and Efficient Net

**VGG16**: It is object detection and classification model/algorithm. It is 16 layers deep. This model has 92.7% accuracy on the ImageNet dataset. ImageNet dataset has 14 million images belonging to 1000 different classes.

**ResNet v2**: This model is 164 layers deep and can classify images into 1000 object categories on ImageNet dataset. It uses global average pooling rather than fully-connected layers — this reduces the model size. VGG16- due to its depth and fully connected nodes, its slower than ResNet.

**EfficientNet:** It improves **Accuracy and Efficiency through AutoML and Model Scaling**. It uses compound coefficient to uniformly scale all dimensions of depth/width/resolution. It is smaller and faster than ResNet.

Now, lets see how our **models performed** in this **capstone project based on these pretrained models**:

## VGG16:

We created two models based on VGG16, named **vggmodel and vggmodel_1**. For vggmodel, architecture guideline was provided in the project. For vggmodel_1,‘’block4_pool” layer was used from VGG16 and model was built after adding flattening and fully connected layers. This model showed the higher accuracy level than the previous one but during prediction both the models were not able to classify simple ‘happy’ face and they generated wrong label. Below table shows the accuracy percentage on test data.

| Model | Accuracy level on test data |
|---|---|
| Vggmodel | 43.75% |
| Vggmodel_1 | 50.78% |

## ResNet V2:

Two models were created (**resnetmodel and resnetmodel_2**). Resnetmodel was built as per the provided guidelines and resnetmodel_2 used “conv3_block4_out” from ResNet V2 and after flattening, classification layer was applied. For resnetmodel, accuracy was not improving hence training was stopped after 5th epoch. To compare both the models, resnetmodel_2 was also provided training only for 5 epochs. Images were provided to both the models for prediction. ‘resnetmodel_2’ predicted two expressions correctly and ‘resnetmodel’ predicted one expression correctly out of four.

Both the models were trained only for 5 epochs and result suggests that **‘resnetmodel_2’ has better accuracy**.
We need to study more about parameters/hyperparameters tuning and model architecture for getting better result. Below table shows the accuracy percentage on test data.

| Model | Accuracy level on test data |
|---|---|
| resnetmodel | 25% |
| resnetmodel_2 | 49.21% |

## Efficient Net:

Again, two models were built(**Efficientnetmodel and Efficientnetmodel_1**). Efficientnetmodel used **‘block6e_expand_activation’** layer whereas Efficientnetmodel_1 used **‘block2c_project_conv’** layer from EfficientNet model. Only this small change showed huge difference in accuracy level. Both the models were trained for 8 epochs. Performance was verified with confusion matrix and classification report. Based on these reports below conclusion was drawn:

*“Efficientnetmodel is classifying everything under ‘happy' label. So Efficientnetmodel is absolutely at its raw stage, and it needs lot of work for being good model. Efficientnetmodel_1 is clearly doing better job than the previous model. It is misclassifying some labels but with more fine tuning and architectural modification we can improve this model.”*

| Model | Accuracy level on test data |
|---|---|
| Efficientmodel | 24.21% |
| Efficientmodel_1 | 50.78% |

Table on the right shows the accuracy percentage on test data.

## Complex Neural Network Architecture:

As per the provided guidelines complex neural network was built. It was named ‘model3’. Using model3 as base model, another model was built named “model3_extension”. “model3”, certainly outperformed all the other model which were built upon the other pretrained models. It achieved 63.28% accuracy on test data in 35 epochs. “model3_extension” achieved even higher accuracy level, 67.96% on test data in 35 epochs. Using classification report and confusion matrix , performance of “model3” and “model3_extension” were compared and below are the findings:

*“If we compare model3 and model3_extension then model3_extension is performing marginally better. It is having 4% more accuracy than model3. Both the models are having lots of scope for the improvement. Among all these models,* ***model3_extension is having better accuracy*** *level.”*

## In summary:

After building many models , looking at their performances , accuracy level, prediction power, it can be said that no model is performing to the expectation level. Model3 and Model3_extension are better among all these models but still they are average performer. However all the models certainly need architectural modifications to perform better.

VGG16, ResNet and EfficientNet are pretrained image classifying models with high accuracy. Architecturally, they are different from each other but they all can be used for image classification problem.

Therefore, if we design or architect our model carefully using pretrained high performance models then we can achieve better result in less training and less time.

In any field, architecture is the key and here also we have to get the model architecture correct for satisfying result.

## References:

https://keras.io/api/applications/

https://arxiv.org/abs/1409.1556

https://arxiv.org/abs/1905.11946

**Thank you !**