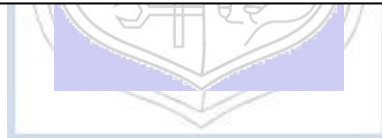




Experiment No.: 6

Title: Transfer learning with CNN



Aim: To implement transfer learning with Convolutional Neural Network.

Theory:

Transfer learning is the reuse of a pre-trained model on a new problem. It's currently very popular in deep learning because it can train deep neural networks with comparatively little data. This is very useful in the data science field since most real-world problems typically do not have millions of labelled data points to train such complex models. In transfer learning, the knowledge of an already trained machine learning model is applied to a different but related problem. For example, if you trained a simple classifier to predict whether an image contains a backpack, you could use the knowledge that the model gained during its training to recognize other objects like sunglasses. With transfer learning, we basically try to exploit what has been learned in one task to improve generalization in another. We transfer the weights that a network has learned at "task A" to a new "task B."

When to Use Transfer Learning-

When we don't have enough annotated data to train our model with. When there is a pre-trained model that has been trained on similar data and tasks. If you used TensorFlow to train the original model, you might simply restore it and retrain some layers for your job. Transfer learning, on the other hand, only works if the features learnt in the first task are general, meaning they can be applied to another activity. Furthermore, the model's input must be the same size as it was when it was first trained.

1. TRAINING A MODEL TO REUSE IT

Consider the situation in which you wish to tackle Task A but lack the necessary data to train a deep neural network. Finding a related task B with a lot of data is one method to get around this. Utilize the deep neural network to train on task B and then use the model to solve task A. The problem you're seeking to solve will decide whether you need to employ the entire model or just a few layers.

If the input in both jobs is the same, you might reapply the model and make predictions for your new input. Changing and retraining distinct task-specific layers and the output layer, on the other hand, is an approach to investigate.

2. USING A PRE-TRAINED MODEL

The second option is to employ a model that has already been trained. There are a number of these models out there, so do some research beforehand. The number of layers to reuse and retrain is determined by the task. Keras consists of nine pre-trained models used in transfer learning, prediction, fine-tuning. These models, as well as some quick lessons on how to utilise them, may be found here. Many research institutions also make trained models accessible. The most popular application of this form of transfer learning is deep learning.

Activity:

- 1) Import requisite libraries using Tensorflow and Keras.
 - 2) Load the selected dataset.
 - 3) Visualize and display random images belonging to each class.
 - 4) Select the model to be used for transfer learning.
 - 5) Using pre-trained model. Develop CNN for your dataset.
 - 6) Print Model Summary and display architecture diagram.
 - 7) Compile and fit the model on train dataset.
 - 8) Calculate training and the cross-validation accuracy.
 - 9) Redefine the model by using appropriate regularization technique to prevent overfitting.
 - 10) Fit the data on the regularized model.
 - 11) Calculate and plot loss function and accuracy using suitable loss function.
 - 12) Display classification Report for regularized CNN model.
 - 13) Comment on output.
-

CO:

Conclusion:

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

References:

Books/ Journals/ Websites:

