Experiment No - 06.

Page No.
Date

Aim :-

Object detection using transfer learning of CNN architectures.

Objective :-

a. Load in pre-trained CNN model trained on a large dataset.
b. Freeze parameters (weights) in model's lower convolutional layers.
c. Add custom classifier with several layer of trainable parameters of model.
d. Train classifier layer on training data available for task
e. fine-tune hyper parameters and unfreeze more layers as needed.

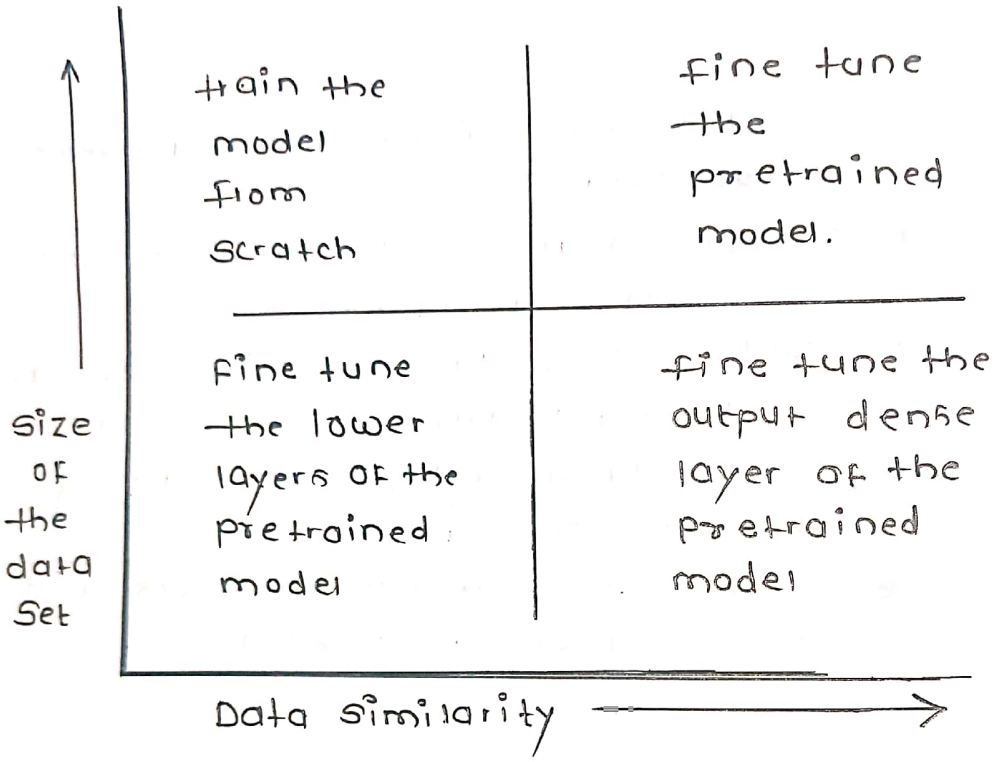
Theory :-

a. Load in pre-trained CNN model trained on a large dataset.

Artificial neural networks such as Recurrent neural network's Convolutional Neural Network's, GANs, and Autoencoders.

The learning rate used in training of the pre-trained model.

Learning rate gradually increases from 0.00002 to 0.002 at epoch 100, and linearly decreases until reaching 0.0002 at epoch 800



pre-trained model is a model created by some one else to solve a similar problem. Instead of building a model from scratch to solve a similar problem

b. Freeze parameters (weights) in model's lower convolutional layers.

freezing a layer prevents its weights from being modified. This technique is often used in transfer learning where the base model

```
# setting trainable = false for freezing the layer
model.add (conv2D (64, (3,3), trainable = false))
```

layer freezing is process of freezing the weights of specific layer in deep learning network so that these wights don't change during training.

c. Add custom classifire with several layer of ~~traine~~ trainable parameter's of model.

We have compile information about the date of development and trainable parameter counts of if you're not familiar with the MINIST dataset it's a collection of 0-9 digits as images. These images are gray-scale and thus each image can be represented with an input shape of 28 x 28 x 1. as shown in Line 5.

Page No.
Date

us. to take advantage of that the model has already learned without having to develop it from scratch.
used transfer leaning with efficientnet_B0, and to do is to gradually.

d. Train classifier layer on training data available for task.

- The weights in those layers to be re-initialized.
  base_model.trainable = False
- The next step is to add new trainable layers that will turn old features into predictions on the new dataset.
- This is important because the pre-trained model is loaded without the final output layer
- A final dense layer with units corresponding to the number of output expected by your model

e. Fine-tune hyper parameters and unfreeze more layer as needed.

- The optimal hyper parameters, let use first understand these hyper parameters:
- learning rate, batch size, momentum, and high weight decay. These hyper parameters act as

- The learning rate is high, then training may not converge or even diverge.
- hyper-parameters for your model is hard. Especially if yo do it manually
- select the
- steps hyperparameter tuning:
  1. Select the ~~list of parameter~~ right type of model.
  2. Review the list of parameter's of the model build the HP space.
  3. Finding the method for searching the hyperparameter space.
  4. Applying the cross-validation scheme approach.
  5. Assess the model score to evaluate the model.

Conclusion :-

We have disscussed in detailed study of object detection using transferflow learning of CNN architecture.