Computer Vision
Assignment №14

Pooya Kabiri

Department of Computer Science

Iran University of Science and Technology

January 2021

1 Underfitting & Overfitting

1. **Underfitting in Neural Networks**

   Underfitting is simply oversimplifying of the problem by the neural Network.

   Underfitting occurs when the model doesn't have the strength and complexity of non-linearity to fit the data in a promising way.

   An underfitted model has poor performance on training and also on test. This model neither can fit the training data nor do a well generalization.

2. **Overfitting in Neural Networks**

   Overfitting is using more than necessary complexity and strength to solve a problem.

   Overfitting occurs when a model with a lot of non-linearity strength-which can fit very complex data is fitted on a simple data. The model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. In fact even every noise in training data is learned by the network.

   An overfitted neural network model has a very high performance on the training data, but in terms of test data and generalization on new data, the model performs poorly, so there is a huge gap between training and generalization performances.

   Overfitting can be reduced by:

   - **Using more data and using data augmentation:** this results in better performance for generalization of the model.

- **Reducing network capacity:** like reducing number of layers, number of neurons in layers etc. This results in the model to be less strong can less likely to overfit the data.
- **Dropout Layers:** Causes some portion of the neurons be deactivated in an epoch, and it is done randomly with a probability. In this way not all the neurons are learning simultaneously, so the overall network capacity is reduced and the model is less likely to overfit the data.
- **Regularization:** When using regularization, the loss penalty is increased as the model complexity increases, so the network is discouraged to learn a complex model
- **Cross-Validation:** Is used to measure the generalization of the model in each training epoch and can be used to detect and prevent overfitting.

2 Transfer Learning

Transfer Learning is used when a model developed for a task is reused as the starting point for a model on a second task.

In Computer Vision, a pre-trained network which is trained on large image datasets like ImageNet is used as a backbone network and works as a feature extractor. Then the produced feature map is fed into a second network which is designed for the special purpose of the task.

The first layers of a CNN usually learns basic and general features which are problem independent like lines etc. When using a pre-trained network these early learned weights can be used for basic feature extraction without the need for training and other time-consuming task.

The Backbone Network's weights can be frozen or can be trained alongside the higher problem-dependent layers.

Because these backbone networks are usually trained on very large datasets, fine-tuning can reduce the problem of overfitting and improves generalization.

This process is called **Fine-Tuning**.

In Conclusion, Fine-Tuning is effective because:

- **Greatly reduces training time**
- **Improves performance**
- **Reduces overfitting on small datasets**

3 Depth-wise Separable Convolutions

1. **Typical 5*5 Convolution**

$$Number\ of\ Parameters = (5 * 5 * 3 + 1) * 128 = 9,728$$

2. **Depth-wise Separable Convolution**

$$Number\ of\ Parameters = M * (5 * 5 * 1 + 1) + 128 * (3 * M + 1)$$

$$\Rightarrow M = 2$$

$$\Rightarrow Number\ of\ Parameters = 2 * (5 * 5 * 1 + 1) + 128 * (3 * 2 + 1) = 948$$

4 Time Tracked

The total time tracked for this assignment: **21 Hours and 50 Minutes**.

Time is tracked using **Clockify** application.