Assignment 6

Popescu Andrei-George

November 24, 2023

1 Layers of the model

The structure of a DenseNet121 is based on the construction of 4 dense blocks of layers, and between these blocks a transition layer is used to move the data sent to the next block. Each block consists of 2 convolution layers that are repeated successively a set number of times, depending on the number of the dense block(6, 12, 24 and 16 repetitions).

Before entering the stated blocks, a 7x7 convolution is applied on the input data and a max pooling on the result, then entering the first dense block.

Analyzing in particular the 2 convolution layers that are repeated, they are built starting with a normalization of the received data batch, following an activation function and then performing the convolution (of 1x1, respectively 3x3) after which the dropout is used. These steps are repeated as many times as each specifies the configuration of each individual block.

At the exit from a dense block, it goes through a transition layer, it normalizes the received input, passes it through an activation function, applies a 1x1 convolution and a 2x2 average pooling, with the aim of halving the size of the data that arrives in the next block.

After exiting the last dense block, the output is passed through an activation function, after which a 7x7 max pooling is done and a linear transformation is applied.

2 Forward function

The way the dense network works can best be exemplified within the forward function. We have discussed so far, from the explanation of the layers, how the input data passes through an initial convolution layer and a max pooling on the resulting feature map until entering the first dense block, how the transition layer between blocks is built to reduce the data that include the layers that are applied after exiting the last block.

Looking in detail, I stated that within each dense block there is a number of repetitions of some convolution layers, which we will refer to as a layer in the block. The connectivity of such a block is based on the fact that each layer receives all the feature maps calculated up to that moment, not just the one resulting from the previous layer. In order to achieve this, in each layer when a convolution is applied, an appropriate padding is used so that the output is of the same size. And in each block, before passing through a layer, the features from the previous layers are concatenated to be passed together with the current feature map input through the layer.

3 Gradient flow

Dense network connectivity helps to solve the vanishing gradient problem. This is due to the fact that in the backpropagation stage each layer within a dense block receives gradients on several paths.

Considering the fact that each layer receives not only the gradient that passed through all the layers before it, but all the gradients that were used within the block up to the respective layer in the backpropagation step, it helps to regulate the network, because the neurons will not it only relies on certain features, but they will also be able to use some features already learned in the succeeding layers.