Assignment 7

Popescu Andrei-George

December 5, 2023

1 Model description

In order to verify the efficiency, both from the point of view of optimizing the time given to training, as well as minimizing the loss, we created two models with different structures to test them and find an optimal structure to make the previously mentioned aspects more efficient.

The first model uses two convolution layers with a kernel size of 3 and padding to preserve the dimensions of the feature maps that later pass through two fully connected layers, the last of which forms the output consisting of the 784 pixels (28 x 28 size image). The simplicity of this model led to the need for a much longer training time to reach satisfactory results from the point of view of loss minimization.

The second model was created with the aim of increasing the number of filters used to improve results and training time. It uses three convolution layers, the first using a kernel size of 7 (with the aim of creating more generic feature maps) and the others a kernel of 3. After each of these layers, normalization is applied to the data batch and pooling, halving the amount of information that reaches the next layer, in order to ease the calculations and design of the feature maps that will reach the fully connected layers. I kept the number of two connected layers from the previous model, but I modified the number of neurons present on each hidden layer, in order not to force a drastic reduction of the features learned up to that moment.

In the two graphs that follow, you can see the difference in terms of the efficiency of the two models in training.

Figure 1: Loss results using the first model

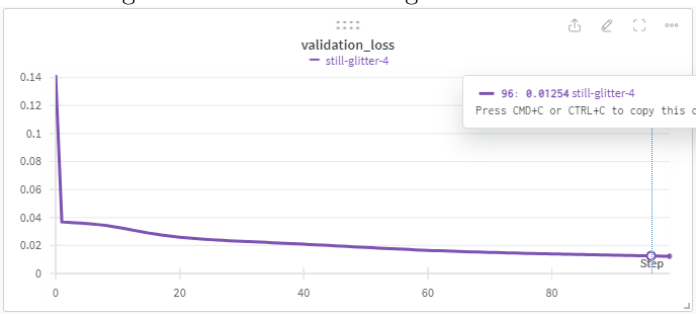
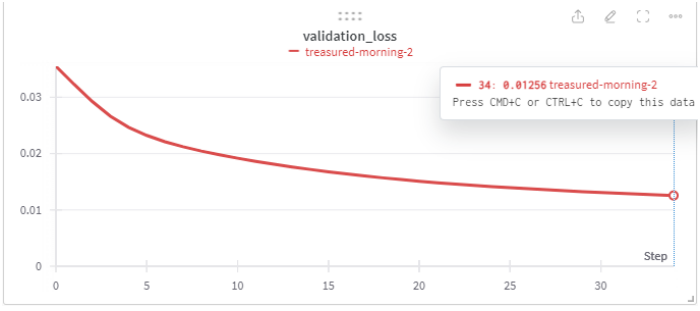


Figure 2: Loss results using the second model



It can be observed that in order to reach approximately the same results in the training, the second one shows an increased efficiency, because in the use of identical parameters (only the structure of the models being different), the first model reaches the results of the second model with 62 later epochs.

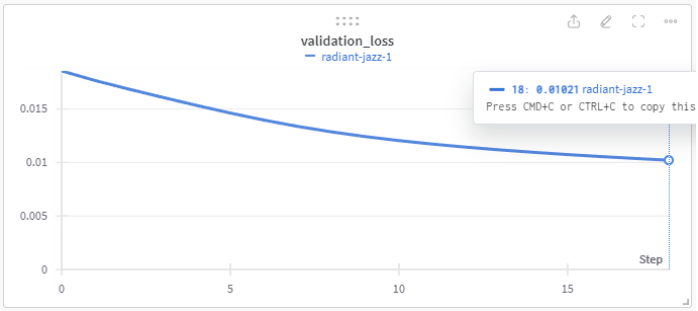
2 Loss function

The optimal loss function was chosen after testing some of those offered by pytorch and comparing the results obtained after training with a different number of epochs and with varied parameters.

In the two experiments exemplified in the previous chapter, Mean Squared Error was used, with the aim of having a model sensitive to outliers that would ensure the learning of large differences, in order to be able to ensure a relative fidelity to the ground truth even with a small number of epoch.

In the example we see in the following graph, the SmoothL1Loss function was used, which is more balanced when it comes to sensitivity to outliers. Through this method, although the loss does not decrease as suddenly, a process of its gradual decrease occurs, in which the loss does not reach an insignificant value as quickly.

Figure 3: Loss results using the second model with the smooth loss function



As can be seen, using a moderate loss function we reached better results in a smaller number of epochs. The difference between the losses that occur during the training must be taken into account in order to optimize the number of epochs in which the training takes place. In the present case, a class created with the aim of early opting for training was used if it is considered that the difference between the last losses becomes too small.

3 Early stopping

I used the early stopping criterion in order to limit the time given to training when the value of the loss does not decrease enough. The choice of the stop was made based on the value of the loss in the respective epoch, compared to the value in the previous one. If a negligible difference is reached, the early stop of the training starts to be taken into account. We considered the possibility that after a certain number of epochs a steeper decrease can be reached, for this reason a check is made on an extended number of epochs (value given as a parameter to the class constructor), and if for the specified number of consecutive epochs, there was no change greater than a given limit within the value of the loss, training is stopped.