Lecture 10: *"Training Deep Neural Networks"*

Book Chapter

Please read the *Chapter11* **"Training Deep Neural Networks"** of the 3rd edition without the sections Implementing batch Normalization, Gradient Clipping, Faster Optimizers, Learning Rate Scheduling, Monte Carlo Dropout and Max-Norm-Regularization) (that means: read p. 357 – 369, 373 – 379, 392-397, 400-401) and answer the following Questions.

Keep in mind: If you answer these questions and write a detailed summary, you won't need to read these chapters again while preparing for the exam

Questions

1. List some problems that are typical of deep neural networks.
2. Describe the vanishing gradient problem.
3. Why can the sigmoid function as activation function cause a vanishing gradient?
4. Is it useful to randomly initialize the weights of a neural network? What other methods do exist?
5. Which initialization method is used from Keras by default?
6. Describe the problem of dying ReLUs.
7. Which activation functions exist to outperform the ReLU activation function?
8. How does SELU work and what are the advantages and disadvantages?
9. Draw the function of the Swish activation function.
10. What is the final recommendation for the activation function to start with?
11. How does batch normalization work?
12. What are the advantages of batch normalization, what the disadvantages?

Leave out the subsections: Implementing batch Normalization, Gradient Clipping

13. In which situations is the use of transfer learning useful? How does transfer learning work?
14. How can you find a good model when you have little labeled data but lots of unlabeled training data?
15. Explain self-supervised learning (with an example).

Leave out the subsection Faster Optimizers and Learning Rate Scheduling

16. Which regularization techniques exist to avoid overfitting?

17. Describe the l_1 and l_2 regularization.
18. How does Dropout work?
19. Is there a disadvantage in Dropout?

Leave out the subsections Monte Carlo Dropout and Max-Norm-Regularization

Homework Assignment

Please work on the exercises given in 10-ANN3.ipynb .

Answers

1. Typical are vanishing or exploding gradients, a too small training set, a slow training progress and the risk of overfitting.
2. It can happen that the gradients get smaller and smaller the longer the learning runs. These small gradients cause in the backpropagation algorithm only very small changes in the first layers. So the weigths will not reach an optimal value and the updating process diverges.
3. The sigmoid function is very flat for high or low values, so the gradient for these values is near zero.
4. Yes, a random initialization is useful. There exist other methods, called Glorot or He initialization.
5. Glorot initialization with a uniform distribution
6. If the weighted sum of all inputs to a neuron is negative for all instances the ReLU value will be zero. Therefore it stays at the same value. If this happens to many neurons, the network can't store much information. When all neurons of a layer have negative values the whole layer is lost.
7. SELU > ELU > leaky ReLU > ReLU
8. SELU is scaled ELU, i.e. alpha is approximately 1.67. If we have an MLP and all neurons are initialized LeCun initialization, SELU will normalize all neurons during traning, that means all neurons in a layer will have mean 0 and variance 1. Under these conditions SELU outperforms the other activation functions. A disadvantage is that other techniques like drop out are not possible.
9. For z positive, it is similar to z, in the negative it has a minimum near z=-0.75.
10. Start with ReLU, p. 366
11. p. 367: During training in every layer the input data is scaled and normalized for each mini batch. That means for each mini batch the mean and standard deviation of the input values is computed. With them the values are scaled. It is learned to scale and shift the output y_i. It is rescale with the learned values.
12. p. 369: Batch normalization avoids vanishing or exploding gradients. It allows a higher learning rate and therefore a faster convergence. It is not necessary to scale the data. It is less sensitive to the weight initialization. It also regularize the

model. Disadvantages are: The computation time for each step is longer (but the wall time is usually shorter).

13. p. 376: If you have a similar problem to one that is solved already by another neural network than you should do transfer learning with this network. You take the first lower layers and freeze the weights. Then you add some new layers and train them with your training set. Then you unfreeze all layers and train further. For this step you should reduce the learning rate.
    It works good for big dense layers.
14. p. 378: You can use *unsupervised pretraining*: With the unlabeled data you train an autoencoder or a GAN. Then you use the first layers of the encoder network from the autoencoder or the first layers of the discriminator of the GAN for your new neural network. You add some final layers and train it with the labeled data.
15. You create labeled data from the unlabeled data, e.g. for a text you leave out some words and train a model to find these missing words. This pretrained model can be used as a starting point for another NLP task.
16. p. 393: early stopping, Batch Normalization, l_1 and l_2 regularization, dropout, max-norm regularization.
17. p. 393: The l_2 regularizer computes the l2 loss of the layer. It is added to the loss.
18. According to the given probability p this portion of neurons is set to zero for one training run. They are randomly chosen. The remaining weights must be multiplied with the reverse proportion to keep it fair.
19. The training takes more time. If p is too high, the model will not learn well.

Not used

1. What should be considered when using a neural network with batch normalization for prediction?
2. For what type of neural networks is gradient clipping usually used?
3. For which type of neural network does transfer learning work best?
4.
    5. How does gradient clipping work?

       p. 373: In gradient clipping the gradient is restricted to an upper and lower boundary. Every gradient over or under this boundary is set to the boundary. Gradient clipping avoids exploding gradients. It is used for RNN, where Batch Normalization is not possible.