Assignment 3

Convolutional Neural Networks

Submission deadline: 25.06.2020

General Instructions:

- Submission via moodle, must include code (and all necessary files to run it) and a written report of no more than two pages long.
- Submission in pairs or solitary (include both participants info in written report)
- The code must be written in Python 3.6 or higher, and must run.

Assignment 1 (Code):

In this part we will build an Auto-Regressive model using LSTM, to predict a noised sequence.

1. Load the file "hw3_data_loader.py" from moodle
2. Use the data loader to generate a sequence, and train an LSTM that can predict it.

Assignment 2 (Code):

Build LSTM network that inputs two binary sequences and outputs a binary sequence that is a sum of the two. Note that both inputs are of the same length, but that length is not fixed.

For example:

t	x	y	output
1	0	1	1
2	1	1	0
3	1	1	1
4	0	0	1

At time 1 the network gets the least significant bit and at time 4 it gets the most significant bit. Evaluate you LSTM on long sequences.

Assignment 1 (Report):

1. Describe your network
2. Plot your training loss and final test loss.
3. Plot your final test loss as a function of "noise": increase the value of 'noise', train your network on the new data, and test it for a single value. Plot 10 different noise values.

Assignment 2 (Report):

1. Describe your network
2. Plot your training loss and final test loss. Also, plot the final test accuracy.
3. After the model is trained, create many test samples x, y, and their target vector as defined in Assignment 2. Plot the average prediction accuracy by bit.
   For clarification: your average accuracy for bit i, is

$$\frac{number\ of\ correct\ predictions\ of\ f(x_i, y_i)}{number\ of\ samples}$$

   Can any conclusions be drawn from this result?

Good luck!