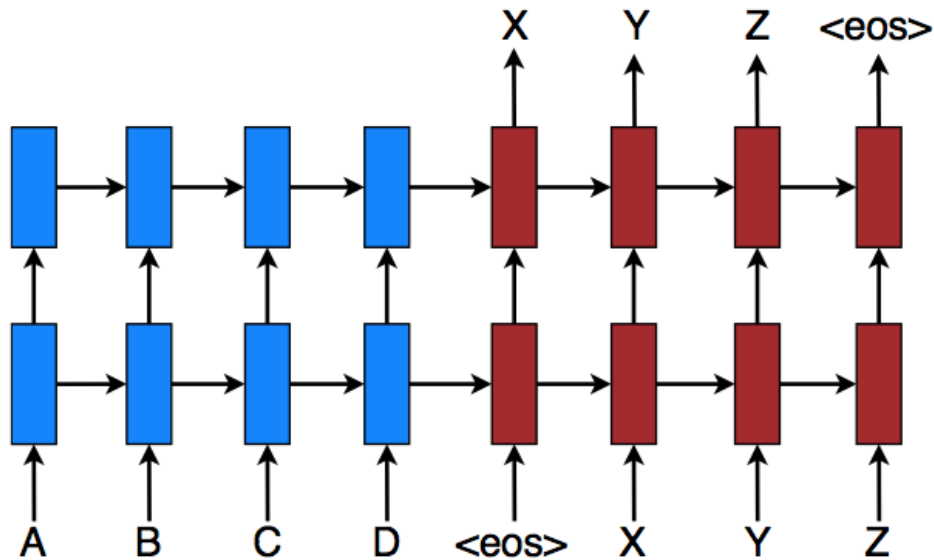


Homework on RNN

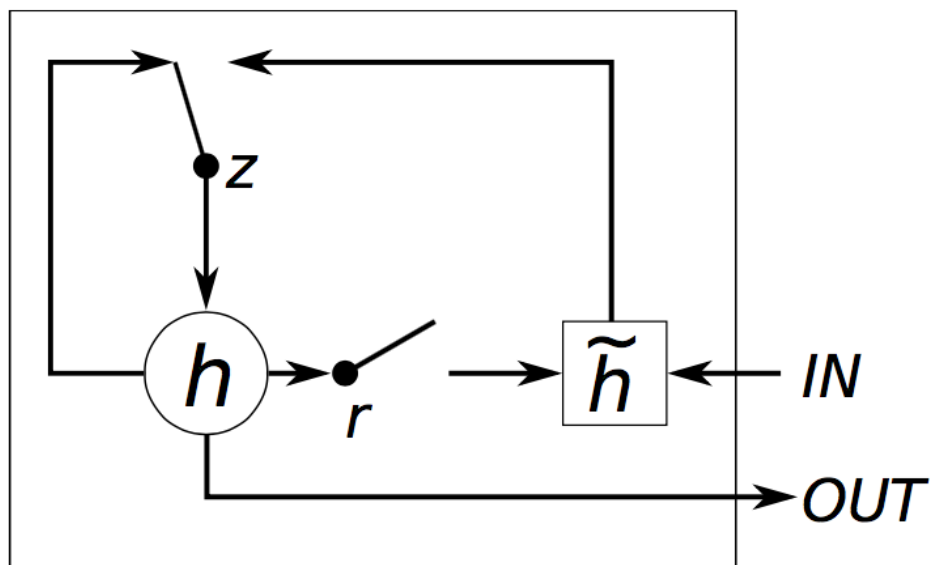
Here I decided to study the tutorial:

<https://github.com/alrojo/tensorflow-tutorial/tree/master>

It implements an encoder/decoder for reading numbers from text strings using gated recurrent units both in encoding and decoding, i.e., “one two five” → “125#”.



Encoder/decoder setup

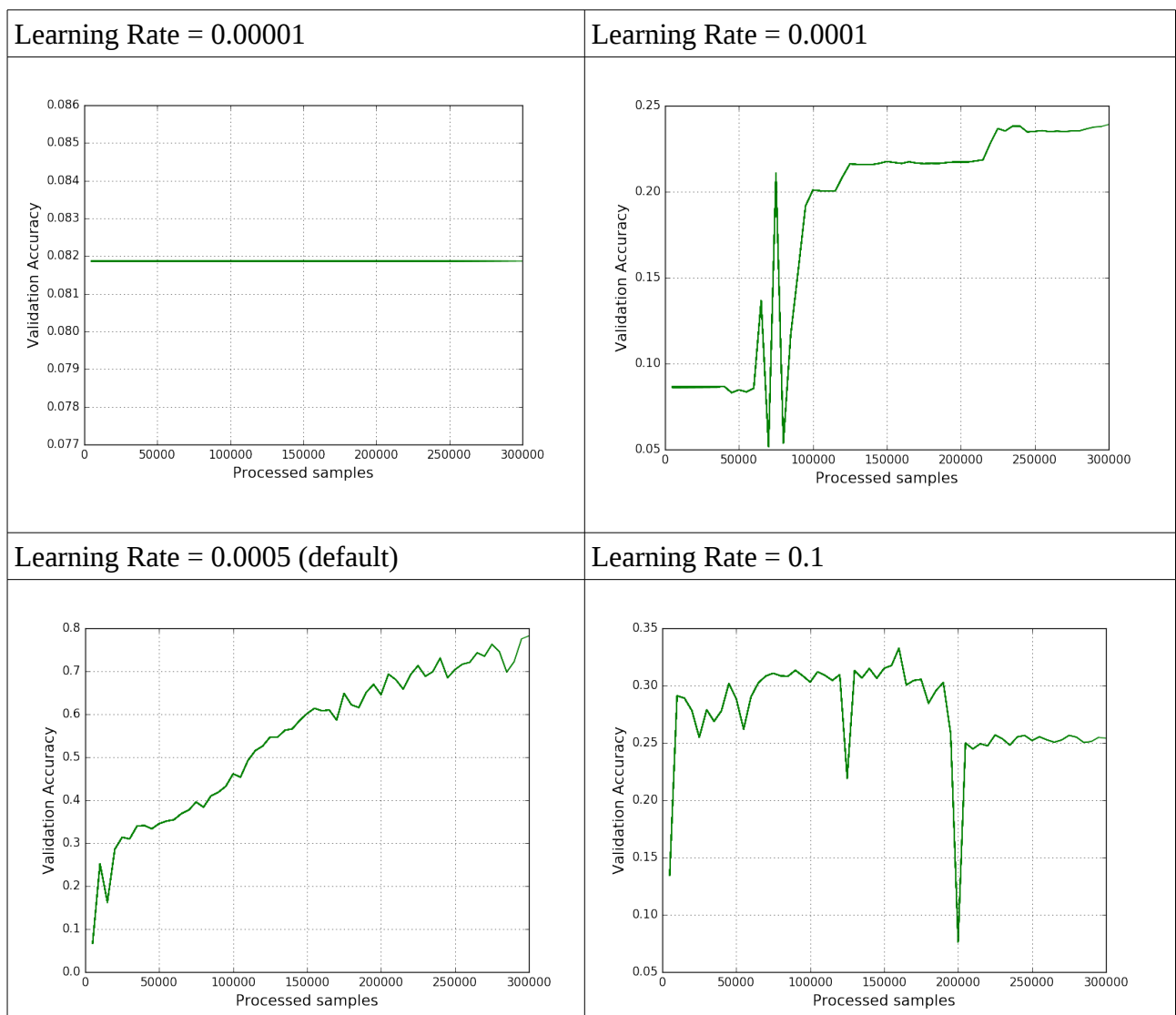


Overview of the GRU: The reset gate r determines how to combine the new input with the previous memory, and the update, z gate defines how much of the previous memory to keep around.

The strings are first coded to integers according to:

'0'=0, '1'=1, '2'=2, '3'=3, '4'=4, '5'=5, '6'=6, '7'=7, '8'=8,
'9'=9, '#'=10, ' '=11, 'e'=12, 'g'=13, 'f'=14, 'i'=15, 'h'=16, 'o'=17,
'n'=18, 's'=19, 'r'=20, 'u'=21, 't'=22, 'w'=23, 'v'=24, 'x'=25, 'z'=26,

The validation accuracy as a function of iterations and learning rate is presented in the figures below, clearly a too small or a too high rate would lead to malfunctioning learning process.



In the tutorial it was also possible to plot the accuracy as a function of target position, for some reason it seemed to be easier for the network to correctly transfer integers in the end beginning position of the sequence.

