

Simple Exercise on Recurrent Neural Networks

This exercise is about understanding how a character-generating recurrent network works. We will assume that the training text is as follows: *We are the students of the Master of Science in Applied Computational Science and Engineering, and we are really interested in Machine Learning.*

We will not differentiate between upper and lower-case letters.

We will also assume that the hidden vector h_t is of dimension 100x1 and that the weights/biases are initialized randomly.

1. What is the dimension of the vocabulary vector? What does it contain?
2. Using the hot encoding notation, how would you represent the vector associated with the letter d or to the blank space character?
3. How many parameters, - or degrees of freedoms - do we have to train, in the case where we have no bias terms in the calculations of h_t and y_t , and in the case where we have bias terms?
4. Typically the value of h_0 is taken to be the null vector, meaning that:

$$h_1 = \tanh(W_{xh}x_0)$$

Since x_0 is the one hot encoding of the first letter w of the training sentence, the (transposed) vector x_0 is:

$$(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0)$$

and $W_{xh}x_0$ is the 18th column of W_{xh} . The coordinates of h_1 are between -1 and +1 because of \tanh . Suppose that the calculation of

$$y_1 = W_{hy}h_1$$

has produced the following values for the transposed of y_1 :

$$y_1^T = (0.1,-0.1,0.2,0.1,-0.3,0.2,0.4,-0.1,0.2,-0.3,0.4,-0.3,-0.5,0.3,0.2,0.5,-0.1,0.3, 0.1,0.1,0.2,-0.3)$$

What is the value of the loss function associated with this first calculation?