COMP 472: Artificial Intelligence

Recurrent Neural Networks

Solutions

Question 1 Assume that we have a RNN with 1 hidden layer with 3 hidden nodes. The weights and the biases are initialized to the following:

U			
-0.1755	-0.2336	-0.1809	0.0814
-0.2765	0.1485	0.2627	-0.4786
0.4060	-0.5527	0.2739	-0.0623

W		
-0.3193	0.5178	0.1962
-0.2081	0.2824	0.4841
-0.2742	0.5378	0.5116

V		
-0.0403	0.4014	-0.1031
-0.5294	-0.3292	0.0249
0.0401	-0.0935	-0.4084
0.2457	-0.2007	-0.5622

$bias_1$		
-0.4951	0.1297	-0.2006

$bias_2$			
-0.1786	-0.1260	0.2593	-0.0886

Trace the feedforward process of RNNs for 2 time steps for a character-based language modelling task given the input string “happy”. Use the formulas as in the slides :

$$s_i = \tanh(U \times x_i + W \times s_{i-1} + bias_1)$$

$$o_i = softmax(V \times s_i + bias_2)$$

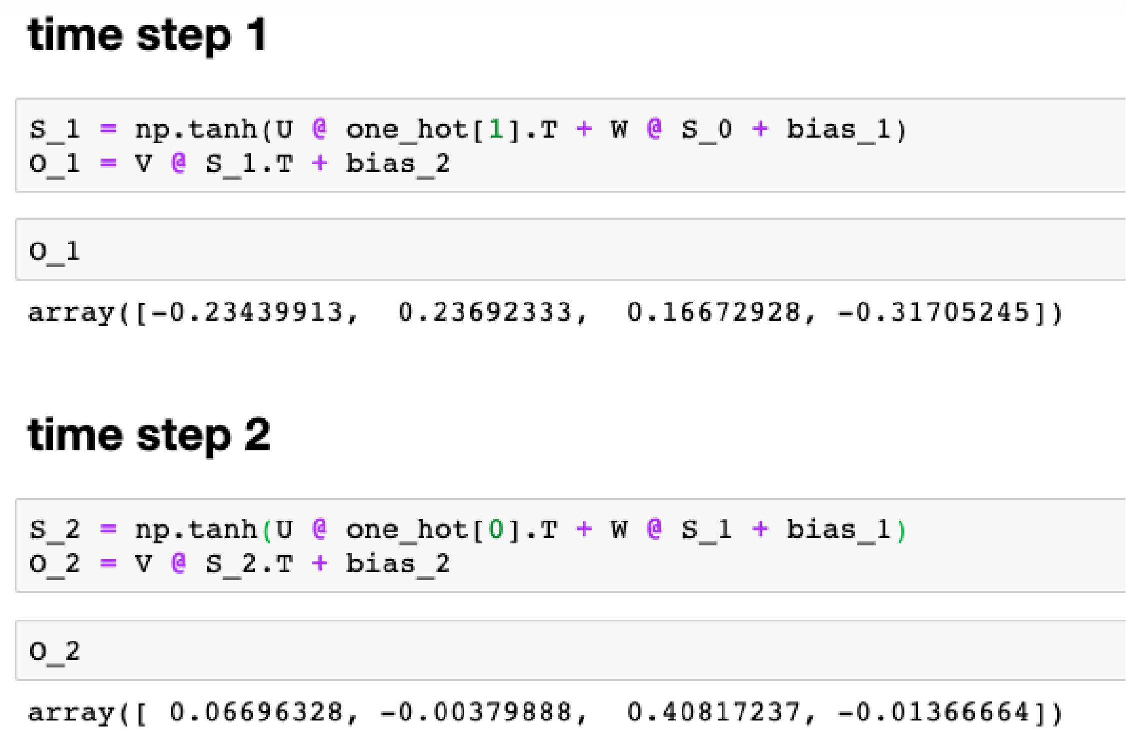
Time step 1:

$S_2 part - a$	$S_2 part - b$	bias	sum	tanh-output	O_1
-0.1755	0	-0.4951	-0.6706	-0.5853	0.1993
-0.2765	0	0.1297	-0.1468	-0.1457	0.3193
0.4060	0	-0.2006	0.2054	0.2006	0.2977
					0.1835

Time step 2:

$S_2 part - a$	$S_2 part - b$	bias	sum	tanh-output	O_2
-0.2336	0.1511	-0.4951	-0.5775	-0.5208	0.2346
0.1485	0.1787	0.1297	0.4569	0.4275	0.2186
-0.5527	0.1857	-0.2006	-0.5675	0.5135	0.3301
					0.2165

Also, here is a short script for doing the calculation. You can find the complete version on Moodle.



time step 1

```
S_1 = np.tanh(U @ one_hot[1].T + W @ S_0 + bias_1)
O_1 = V @ S_1.T + bias_2
```

```
O_1
```

```
array([-0.23439913,  0.23692333,  0.16672928, -0.31705245])
```

time step 2

```
S_2 = np.tanh(U @ one_hot[0].T + W @ S_1 + bias_1)
O_2 = V @ S_2.T + bias_2
```

```
O_2
```

```
array([ 0.06696328, -0.00379888,  0.40817237, -0.01366664])
```