Exercises
Deep Learning
Fall 2018

Machine Learning Institute
Dept. of Computer Science, ETH Zürich
Fernando Pérez-Cruz
Web http://www.da.inf.ethz.ch/teaching/2018/DeepLearning/

Series Monday, Nov 26, 2018
(Deep Learning, Exercise series 9 - solutions)

Solution 1 (LSTM):

a. Recall output f_t acts as a forget key. If $f_t = 1$, then the accumulated info is kept; otherwise, the information is reset. The blue characters, i.e. punctuation marks '!', '.', and '?', signal the end of sentence which is the right time of forgetting the past; therefore, f_t has to be zero for these inputs. Recall f_t expression as

$$f_t = \sigma(W^{(f)} x_t + U^{(f)} h_{t-1}) \tag{1}$$

To make f_t zero for embeddings of punctuation marks, weights $W^{(f)}$ has to keep a negative inner product with the blue points(/embeddings) [1]. Furthermore, the output f_t has to be one for nouns 'cat', 'dog', and 'horse'; hence $W^{(f)}$ should keep a positive inner product with embeddings of these words. Putting all together, $W^{(f)}$ should approximately lies in the highlighted region in the following image.

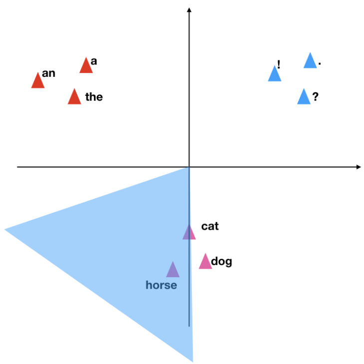

Figure 1: Solution 1.a

b. The formulation $f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$ implies that i_t has to be zero for articles that do not provide important new information. To make i_t zero for inputs 'a', 'an', and 'the', $\boldsymbol{W}^{(i)}$ has to keep a negative inner product with their embeddings (red points). Furthermore, $W^{(i)}$ should keep a positive inner product with embeddings of nouns (pink points). A set of valid candidate directions for $W^{(i)}$ is highlighted in blue in Figure 2.

Solution 3 (Differentiable memory):

In differentiable memories, reading from the memory is through linear combination of all the memory units. All the images can be obtained by a linear combination of 6 images plotted in Figure 3.

Solution 4 (Neural Turing machine):

[1] Recall $\lim_{t \to -\infty} \sigma(t) \to 0$

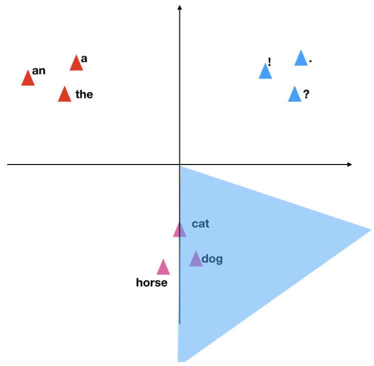

Figure 2: Solution 1.b

Figure 3: Solution 3

a. A neural turning machine learns programming through a set of input-output samples of the target program. To learn copying a sequence with length C, we need to create many different sequences with length C and feed them as input and output of the network. For example, one input-ouput sample can be: $x_t = "a"$ as input and $y_t = "a"$ as output.

b. The multiplication task (task ii) is rather difficult to be learnt from a set of input-output samples. Yet, learning the multiplication program is an open problem in the literature of neural turning machines.

c. Storing and extracting data is necessary for Turing's model of programming. Indeed, we need to learn accessing a memory to learn a program. However, accessing the memory is not differentiable and can not be back-propagated through the training process. Differentiable memories make Turing's machine trainable by back-propagation and simple gradient descent. Without differentiable memories, training of NTM reduces to an NP-hard combinatorial optimization problem.