1 LSTM

In LSTM, the input x_t is a sequence with length T with $1 \leq t \leq T$. The word recurrent means for the t's layer, it not only takes the external input tensor x_t, it also takes the output from the (t-1)'s layer and feed into the current layer. We first calculate z as the following

$$z = tanh(W_o^{(z)} o_{t-1} + W_i^{(z)} x_t)$$

where W are the parameter matrix to be optimized. All the variables above are vectors, matrices or tensors. And the function tanh is broadcast across the entire tensor, in other words, it means we take the tanh value of all the elements of the tensor in (). Then we calculate the input gate i_g, and it is a vector with elements taking values between 0 and 1.

$$i_g = sigmoid(W_o^{(i)} o_{t-1} + W_i^{(i)} x_t)$$

Then the forget gate f_g

$$f_g = sigmoid(W_o^{(f)} o_{t-1} + W_i^{(f)} x_t)$$

Then the output gate o_g

$$o_g = tanh(W_o^{(z)} o_{t-1} + W_i^{(z)} x_t)$$

Based on the value of forget gate and input gate, we update the memory cell c_t.

$$c_t = f_g \circ c_{t-1} + i_g \circ z_t)$$

Where the circle denotes the element wise product, not the dot product. Finally, update the output gate.

$$o_t = o_g \circ tanh(c_t)$$

Let's walk through the dimensions to gain a better understanding. Say o_t's dimension is $n_o \times 1$ and x_t's dimension is $n_i \times 1$. Then similar to the operations we define in basic neural network, if we define the number of neurons in the first layer is $n^{[1]}$, then $W_o^{(i)}$ has dimension $n^{[1]} \times n_o$, and $W_i^{(i)}$ has dimension $n^{[1]} \times n_i$. Therefore, z has the dimension $n^{[1]}x1$.