Deep Learning for NLP (236601) - HW3

Segev Arbiv

1 RNNs (Recursive Neural Network)

1.1 (a)

For all of the nodes, the update equations are the same - except for node 3, which doesn't have posses δ_{above} as it has no parent, and for it $\delta_{above} = 0$
We should keep in mind the the error message is the error messages from parent (δ_{above})+ error message from own score (δ_{below}). Hence, for each node we evaluate:

$$\delta_3 = \hat{y} - y$$
$$\delta_2 = (U^T \delta_3 + \delta_{above}) \circ \mathbb{1}[h^{(1)} > 0]$$

During forward prop, the parent is computed using 2 children, Hence, the errors need to be computed w.rt. each of them:

$$\delta_{below} = [\delta_{below,left} \ \delta_{below,right}] = W^{(1)T} \delta_2$$

And the derivatives, known from previous homeworks:

$$\frac{\partial J}{\partial U} = \delta_3^T h^{(1)}$$
$$\frac{\partial J}{\partial b^{(s)}} = \delta_3$$
$$\frac{\partial J}{\partial W^{(1)}} = \delta_2^T \begin{bmatrix} h_{left}^{(1)} \\ h_{right}^{(1)} \end{bmatrix}$$
$$\frac{\partial J}{\partial b^{(1)}} = \delta_2$$
$$\frac{\partial J}{\partial L_i} = \delta_{below}^T \cdot W^{(1)}$$

1.2 (b)

Python implementation.

1.3 (c)

Figure **??** present the training accuracy Vs. the dev accuracy over epochs, with *wvecdim*=30. We can see that after enough epochs, an **overfitting** to the training data is applied, and the dev accuracy begin to drop.
At figure **??** and **??** we can see the confusion matrices of the training set and the dev set, respectively.
Finally, at figure **??** we can see the dev accuracy vs *wvecdim*.