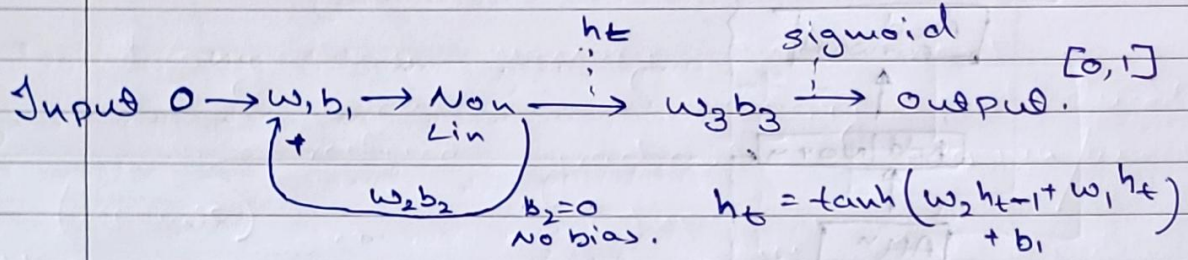
__/__/____

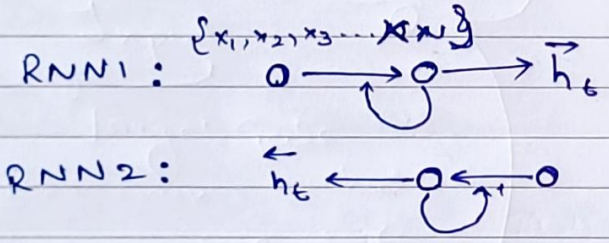
Recurrent Neural Networks



* $\{w_1 b_1, w_2 b_2, w_3 b_3\}$ is shared for all inputs to the RNN

* Very prone to the vanishing/exploding gradient problem

Bidirectional RNN



$$\vec{h}_t = \tanh(\vec{w}\,\vec{h}_{t-1} + \vec{u}\,x_t + \vec{b})$$

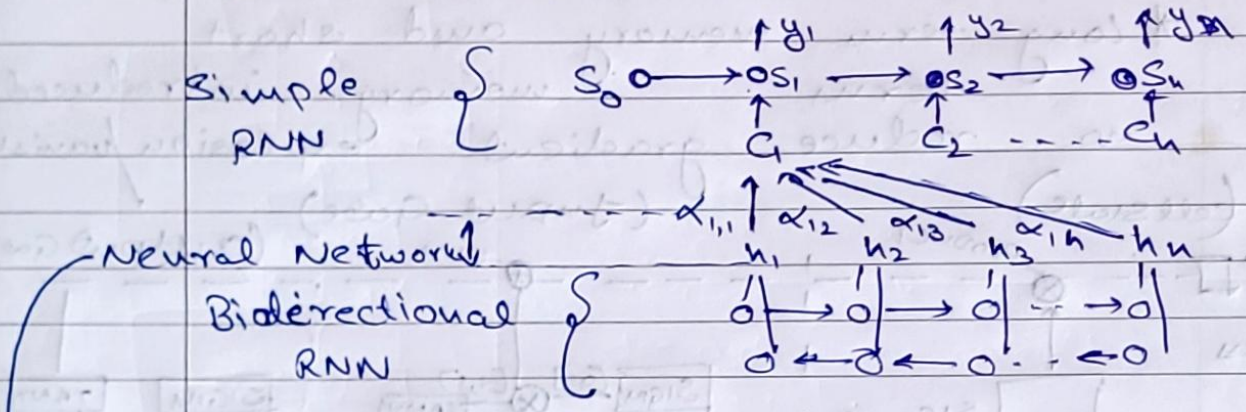
$$\overleftarrow{h}_t = \tanh(\overleftarrow{w}\,\overleftarrow{h}_{t+1} + \overleftarrow{u}\,x_t + \overleftarrow{b})$$

$$y_t = \sigma\left(V[\vec{h}_t, \overleftarrow{h}_t]_{concat} + b\right)$$

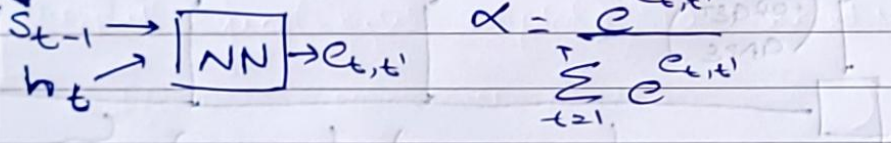
The output are concatenated at every time step before passing thru a final sigmoid.

__/__/____

RNN attention architecture



$$C_h = \sum_{t=1}^{N} \alpha_{t,h} * h_t$$



$$\alpha = \frac{e^{e_{t,t'}}}{\sum_{t=1}^{T} e^{e_{t,t'}}}$$

* There is a nural network to give attention weights only based of previous step and current embedding.