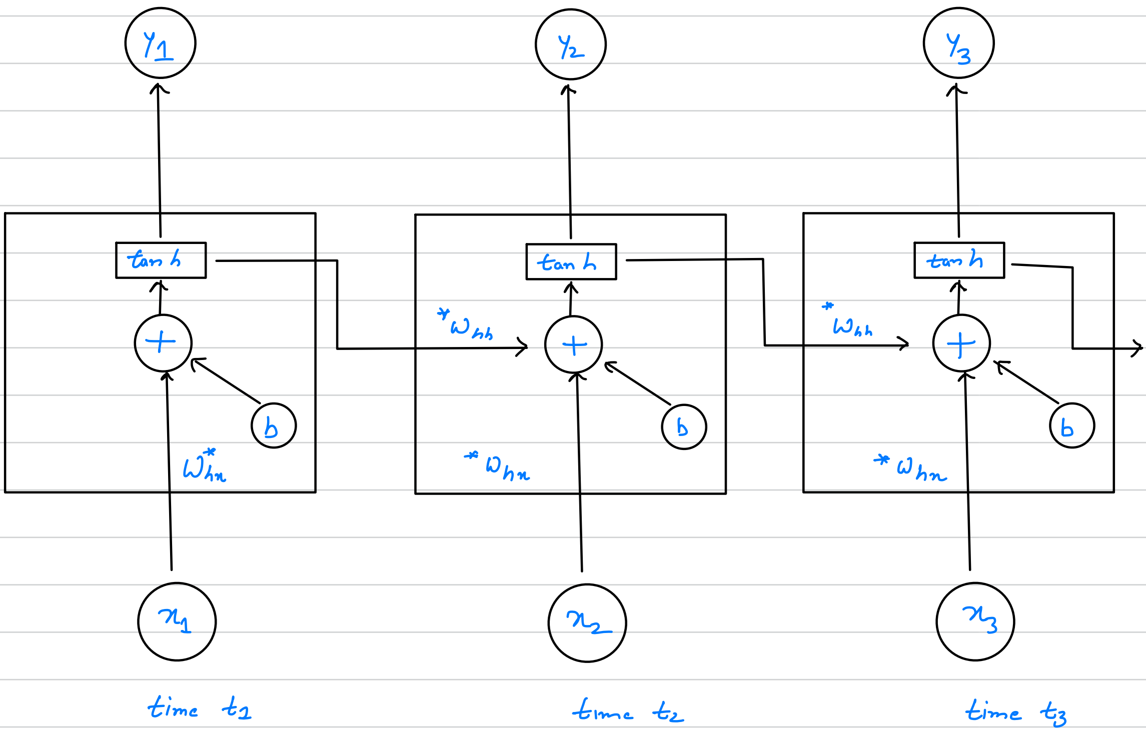
M20 AIE 318
Utkarsh Thusoo

Q 2) Vanilla RNN



In simple terms, when we consider Mean squared error, it would be computed for all training examples by

$$\frac{\text{square all the errors}}{\text{mean total error}}$$

let us suppose $\hat{x}_i$ be the vector denoting n number of prediction values. Also x_i be the vector denoting n number of true values

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (\hat{x}_i - x_i)^2$$

We know,

$$h_t = \tanh(\omega_{hh} h_{t-1} + \omega_{xh} x_t)$$

$$= \tanh\left(\begin{pmatrix} \omega_{hh} & \omega_{hx} \end{pmatrix} \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \right)$$

$$= \tanh\left[\omega \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \right)$$

$$\frac{\partial h_t}{\partial h_{t-1}} = \tanh'(\omega_{hh} h_{t-1} + \omega_{xh} x_t) \, \omega_{hh}$$

let L_{t_1} is loss at step 1

L_{t_2} is loss at step 2

L_{t_3} is loss at step 3

Total loss using MSE =

$$MSE = \frac{(Lt_1 + Lt_2 + Lt_3)^2}{3} \rightarrow \quad (1)$$

let y_1 be correctly predicted values

y_1' be true values

$\therefore \quad Lt_1 = (y_1 - y_1')$

Similarly $\quad Lt_2 = (y_2 - y_2')$

$Lt_3 = (y_3 - y_3')$

$$\therefore \quad \text{Loss, } L = \left(\frac{(y_1 - y_1) + (y_2 - y_2) + (y_3 - y_3')}{3}\right)^2$$

$$\frac{\partial L}{\partial w} = \frac{1}{3}\left[\frac{\partial}{\partial w}(y_1 - y_1')^2\right] + \frac{\partial}{\partial w}(y_2 - y_2')^2 + \frac{1}{3}\frac{\partial}{\partial w}(y_3 - y_3')^2$$

w.r.t $\quad W_{hh}$ & W_{hx}

Q3) Five things learned from assignment

1. Predict labels within an image using CNN.

2. How to extract features by removing certain layers.

3. Using Encoders to encode the label to numbers for easily splitting train & test data

4. Image net and how useful it is in providing images on variety of things

5. RNN advantages over CNN and how can we compute loss in RNN.