18 Oct 2024

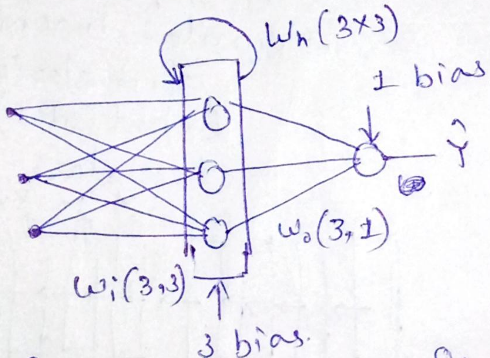
BACKPROPAGATION IN RNN (BPTT)

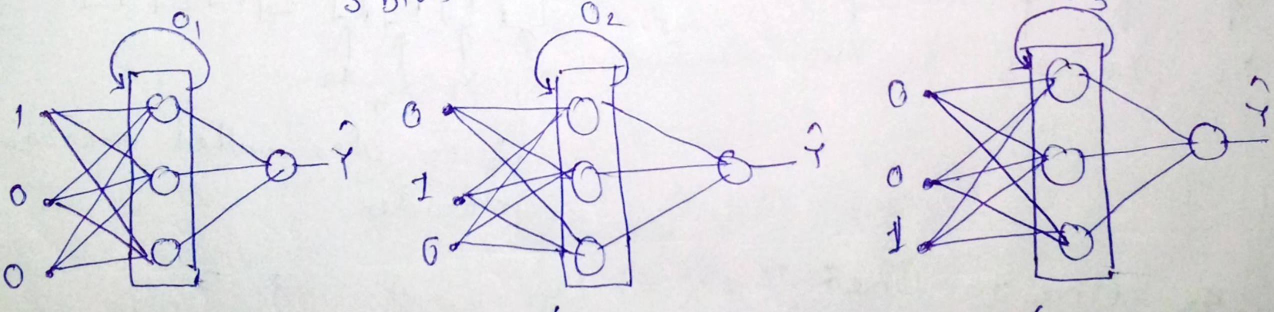
- BPTT (Backpropagation Through Time)
- Here we take a Many to one RNN Example
- Example:

text	output
cat mat rat	1
rat rat mat	1
mat mat cat	0

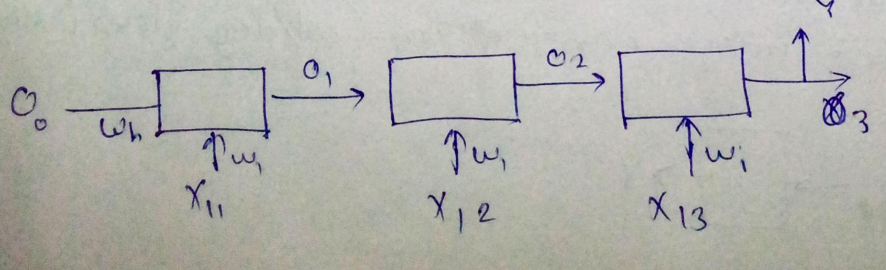
→

	X	Y
X_1	[1 0 0] [0 1 0] [0 0 1]	1
X_2	[0 0 1] [0 0 1] [0 1 0]	1
X_3	[0 1 0] [0 1 0] [1 0 0]	0





t=1 (X_{11}) t=2 (X_{12}) t=3 (X_{13})



$$O_1 = f(X_{11} W_i + O_0 W_h)$$

$$O_2 = f(X_{12} W_i + O_1 W_h)$$

$$O_3 = f(X_{13} W_i + O_2 W_h)$$

$$\hat{Y} = \text{sigmoid}(O_3 . W_o)$$

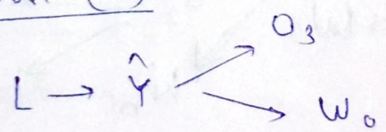
$$\text{loss}(L) = -Y_i \log \hat{Y}_i - (1 - Y_i) \log (1 - \hat{Y}_i)$$

- must minimize the loss using gradient descent where W_i, W_h, W_o values (optimal) where the loss (L) is minimum by weights update

$$W_i = W_i - \eta \frac{\partial L}{\partial W_i} \quad \text{—— (i)} \qquad W_h = W_h - \eta \frac{\partial L}{\partial W_h} \quad \text{—— (ii)}$$

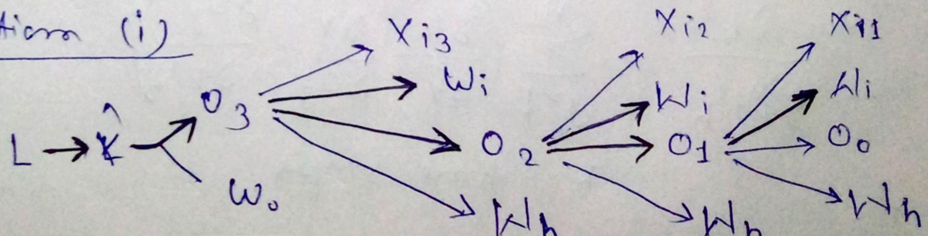
$$W_o = W_o - \eta \frac{\partial L}{\partial W_o} \quad \text{—— (iii)}$$

* in equation (iii)



$$\frac{\partial L}{\partial W_o} = \frac{\partial L}{\partial \hat{Y}} \cdot \frac{\partial \hat{Y}}{\partial W_o}$$

* in equation (i)



$$\cancel{\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial w_i}}$$

$$\cancel{\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{\partial w_i}}$$

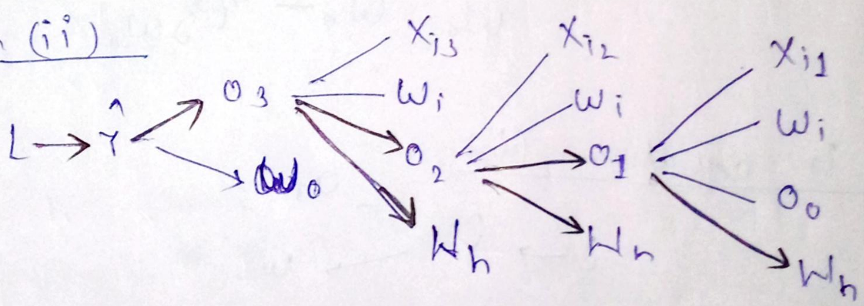
$$\frac{\partial L}{\partial W_i} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial w_i} + \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{\partial w_i} + \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{\partial O_1} \cdot \frac{\partial O_1}{\partial w_i}$$

OR

$$\frac{\partial L}{\partial w_i} = \sum_{j=1}^{n} \left(\frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_j} \cdot \frac{\partial O_j}{\partial w_i} \right)$$

n = time steps

* in equation (ii)



$$\frac{\partial L}{\partial W_h} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial W_h} + \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{\partial W_h} + \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{O_1} \cdot \frac{\partial O_1}{\partial W_h}$$

OR

$$\frac{\partial L}{\partial W_h} = \sum_{j=1}^{n} \left(\frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial O_j} \cdot \frac{\partial O_j}{\partial W_h} \right)$$

n = time steps

PROBLEM WITH RNN

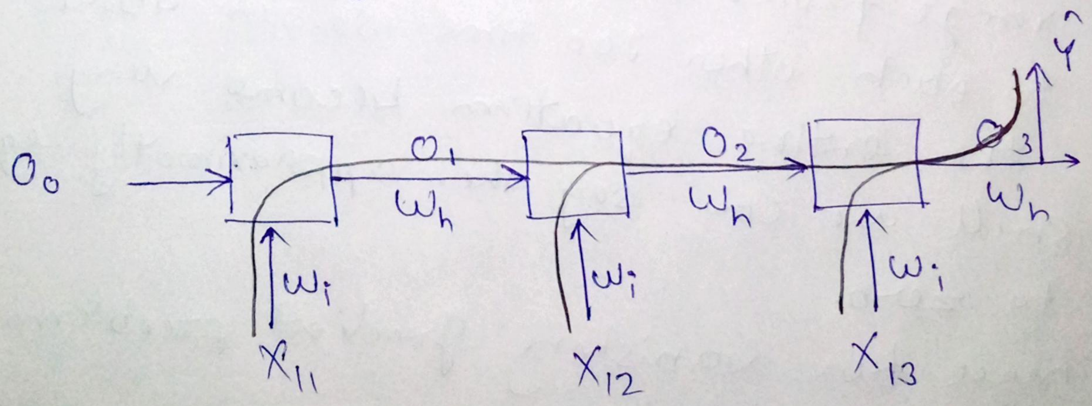
problem with RNN:

- problem of long term dependency.
- unstable Training/stagnated training.
- The above two problems are come due to unstable gradients

① Problem of long term dependency:

- this problem is occur due to vanishing gradient
- $\frac{\partial L}{\partial w_i}$ is depends on three terms.

$$\frac{\partial L}{\partial w_i} = \underbrace{\frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial w_i}}_{\text{short term dependency}} + \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{\partial w_i} + \underbrace{\frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_3} \cdot \frac{\partial O_3}{\partial O_2} \cdot \frac{\partial O_2}{\partial O_1} \cdot \frac{\partial O_1}{\partial w_i}}_{\text{long term dependency}}$$



- for more number of time steps the number of terms in $\frac{\partial L}{\partial w_i}$ is very long due to that long term dependency occurs (becomes small)

- hence the calculating the $\frac{\partial L}{\partial w_i}$ the major contribution is on short term dependency.

for very large time step.

$$\frac{\partial L}{\partial W_i} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_{100}} \prod_{t=2}^{100} \left(\frac{\partial O_t}{\partial O_{t-1}}\right) \frac{\partial O_1}{\partial W_i}$$

where

$$O_t = \tanh(X_{it} W_i + O_{t-1} W_h)$$

$$\frac{\partial O_t}{\partial O_{t-1}} = \underbrace{\tanh'(X_{it} W_i + O_{t-1} W_h)}_{\text{range } 0 \text{ to } 1} W_h$$

hence,

$$\frac{\partial L}{\partial W_i} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial O_{100}} \prod_{t=2}^{100} \left(\tanh'(X_{it} W_i + O_{t-1} W_h) W_h\right) \frac{\partial O_1}{\partial W_i}$$

- tanh'( ) value range from 0 to 1 and Wh value range from 0 to 1 and they are multiply to each other 100 time. so in above the entire equation become very small we can say the approximatly equal to zero.

- hence her vanishing gradient problem occur.

Solutions to resolve this problem:

- Use different activation functions.
- Better weight initialization. example: we can use the identity matrix
- different type of RNN (skip RNN)
- use LSTM

2. Unstable Training (Exploding gradients):

- Here the long term dependency becomes very large as compared to short term dependency
- This problem is occurs may be due to very large learning rate
- use reLU activation function with $W_h = 1$ initialisation.

Solutions to resolve this problem:

- use gradient clipping
- controlled learning rate
- use LSTM