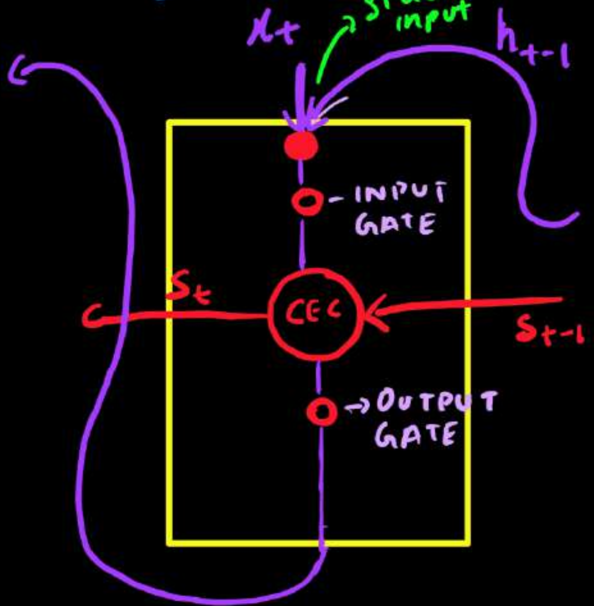
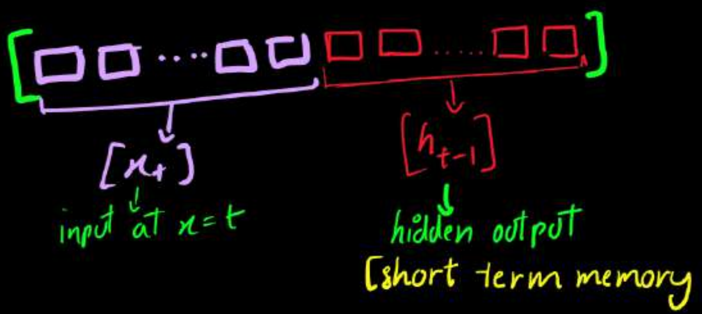
BPTT IN LSTM



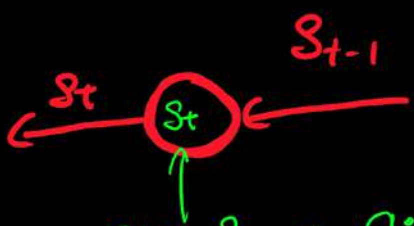
Stacked INPUT

⇒



CEC Cell [long term memory]

At every Time step

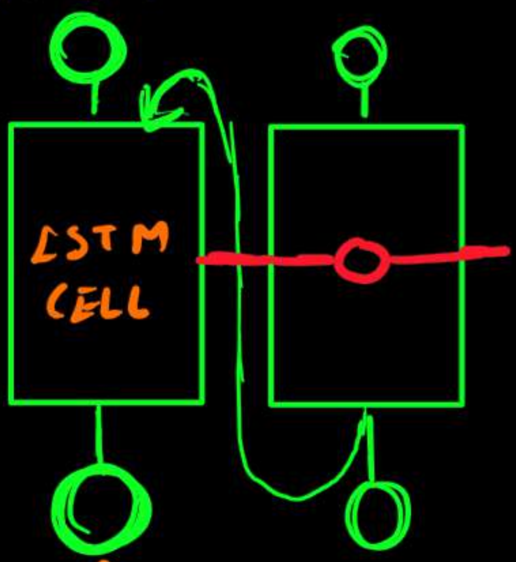


$S_t = S_{t-1} \times a_{ig}$

input gate activation

BACKPROPAGATION

1



Loss is calculated at output cell

* FOR MSE

$$\text{Loss} = (y - \hat{y})^2$$

True y

predicted y

$$\frac{dL}{d\hat{y}} = 2(y - \hat{y})$$

Now we find out how the loss affects the activation and the weights of the output cell

② $\hat{y} = a(w_y h_x + b_x)$

a → activation function
w_y → output weights
h_x → hidden output at layer x
b_x → bias at x

NOTE:- All the derivative are partial derivative here

We need to find $\frac{dL}{dw_y}$ Slope of the loss function wrt w_y

$$\frac{dL}{dw_y} = \frac{dL}{d\hat{y}} \cdot \frac{d\hat{y}}{dw_y}$$

$$\frac{d\hat{y}}{dw_y} = \text{activation derivative} * h_x$$ → this is the gradient change at time step t

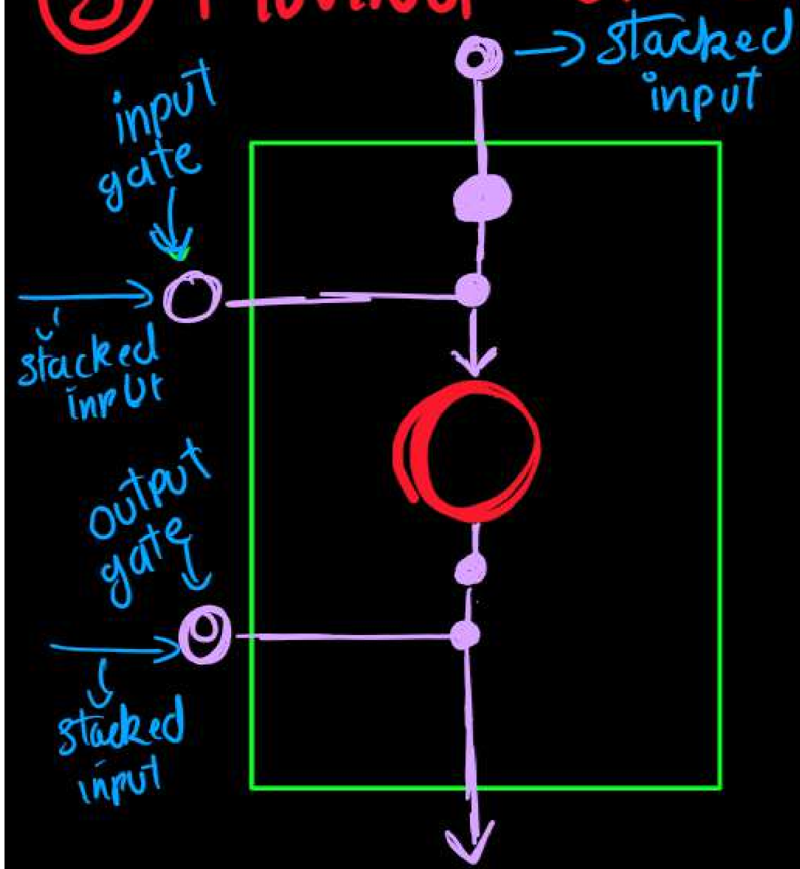
$$\frac{d\hat{y}}{d\,b_y} = \text{activation derivative} * 1$$ → Bias gradient

☆ A Gradient accumulator sums up these gradients (changes) over all the time steps

we also find $\frac{d\hat{y}}{d\,h_x}$ [Slope of the loss function wrt activation of the hidden layer]

$$\frac{d\hat{y}}{dx} = \text{activation derivative} * w_y$$

③ MOVING ONTO THE LSTM MEMORY CELL



Right now we have

→ Updated the output weights & bias to the gradient accumulator

→ we have the gradient of how loss function affects the output of the hidden layer

→ In an LSTM cell we have to update the weights and biases of 3- thing

(i) Input weights & biases

(ii) Input gate weights & biases

(iii) Output gate weights & biases

★ Taking the stacked input as is_t at time t

Backpropagation starts from the output gate

a_{og} = activation($is_t \cdot w_{og} + b_{og}$)

a_{og} → activation of output gate

is_t → stacked input

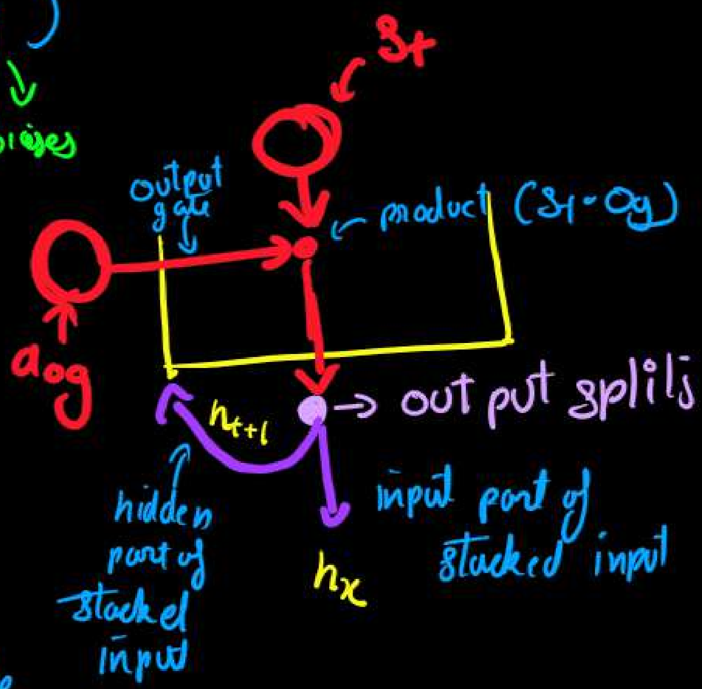
w_{og} → weights of output gate

b_{og} → biases

$[h_{next}, h_x]$ = activation($a_{og} \cdot s_t$)

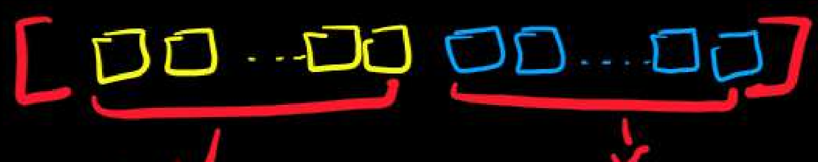
h_{next} → hidden part which goes to next time step

h_x → output of hidden layers



For now lets assume we get the gradient of the hidden part of the stacked output from the next time step

We also have d_{h_x} from the output



→ gradient stacked output ⇒

dh_{next} dh_x

Now that we have the gradient of stacked output we need

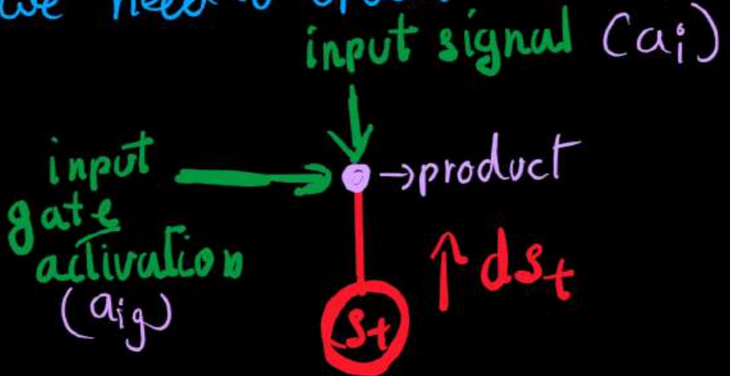
$$\frac{dL}{w_{og}} = \underbrace{\frac{dL}{do_s}}_{\text{gradient stacked output}} \cdot \underbrace{\frac{do_s}{da_{og}}}_{\text{choin rule Till we get } dw_{og}} \cdot \underbrace{\frac{da_{og}}{dw_{og}}}$$

→ This will get us the gradient to update output gate

So calculate $\frac{dL}{ds_t} = \frac{dL}{do_s} \cdot \frac{do_s}{ds_t}$

★ I dont want to calculate everything again so I will just write them in derivative form done

Now we need to update the input gate, and cell input weights



$$a_{ig} = act(i_s \cdot w_{ig} + b)$$

$$a_i = act(i_s \cdot w_i + b)$$

$$s_t = s_{t-1} * \text{product}$$

$$s_t = s_{t-1} * a_{ig} * a_i$$

same $\frac{dL}{ds_t}$ is used to calculate.

$$\frac{dL}{dw_{ig}} = \frac{dL}{ds_t} \cdot \frac{ds_t}{da_{ig}} \cdot \frac{da_{ig}}{dw_{ig}}$$

$$\frac{dL}{dw_i} = \frac{dL}{ds_t} \cdot \frac{ds_t}{da_i} \cdot \frac{da_i}{dw_i}$$

calculate this chain rule

All these calculated gradient are added to their respective gradient accumulators

∴ Gradients from all the three will consitute the gradient of the input vector

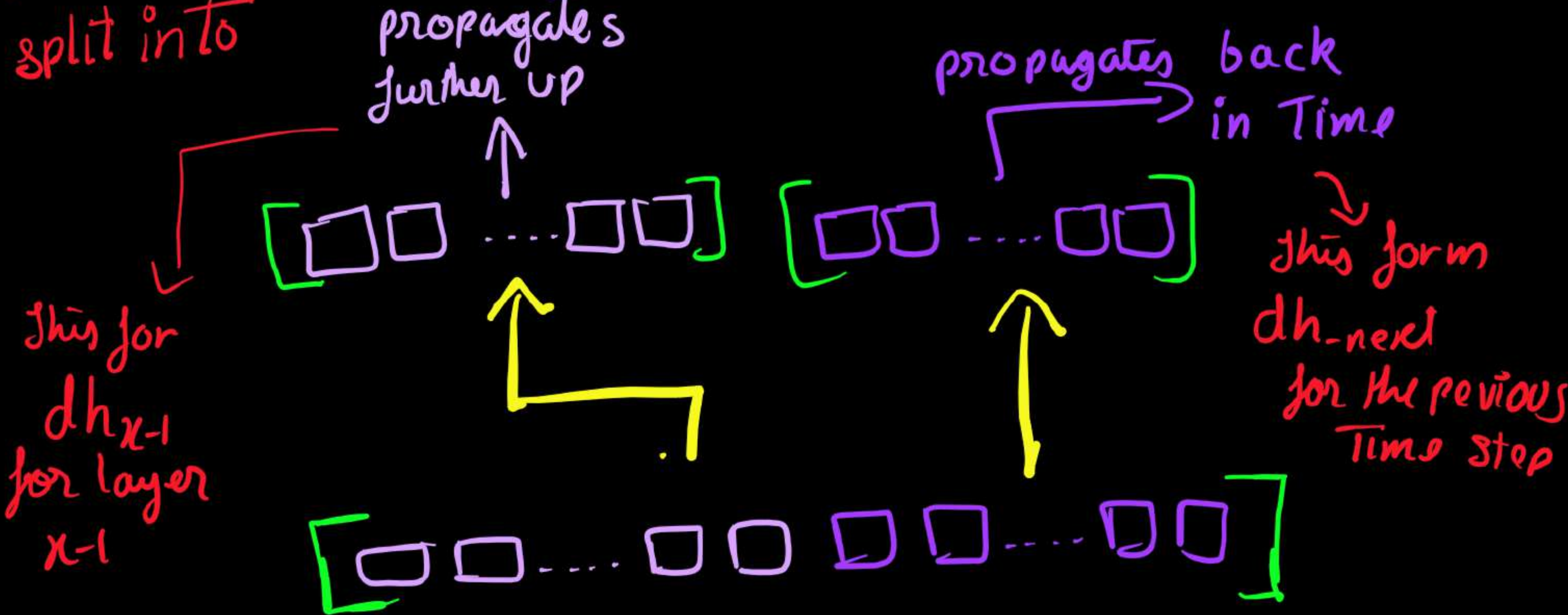
$$\frac{dL}{di_s}\left(\text{from output gate}\right) = \frac{dL}{da_{og}} \cdot \frac{da_{og}}{di_s} = w_{og} * \text{activation derivative (output gate)} \; — \; ①$$

$$\frac{dL}{di_s}\left(\text{from input gate}\right) = \frac{dL}{da_{ig}} \cdot \frac{da_{ig}}{di_s} = w_{ig} + \text{activation derivative (input gate)} \; — \; ②$$

$$\frac{dL}{di_s}\left(\text{from cell input}\right) = \frac{dL}{da_i} \cdot \frac{da_i}{di_s} = w_i + \text{activation derivative (cell input)} \; — \; ③$$

$$\frac{dL}{di_s} = ① + ② + ③$$

FINNALY this stacked input gradient is then split into



WITH THIS BPTT through an LSTM is complete