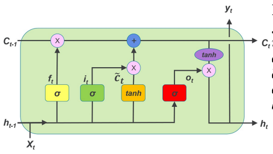
RNN: $h_t = \phi_h(W_h h_{t-1} + W_x x_t + b_h)$ and $y_t = \phi_t(W_y h_t + b_y)$ where ϕ is the activation.
Backward Pass: calculate $\mathcal{L}(\vec{\theta}) = L_{total}(\vec{\theta}) = \sum_{t=1}^{T} L(y_t)$ and $\frac{d}{d\theta}\mathcal{L}(\vec{\theta}) = \sum_{t=1}^{T} \frac{d}{d\theta} L(y_t)$ is computed for each $\theta \in \{W_h, W_x, W_y, b_h, b_y\}$.

- $\frac{d\mathcal{L}(y_t)}{dW_y} = \frac{d\mathcal{L}(y_t)}{dy_t}\frac{dy_t}{dW_y}$ and $\frac{d\mathcal{L}(y_t)}{db_y} = \frac{d\mathcal{L}(y_t)}{dy_t}\frac{dy_t}{db_y}$
- $\frac{d\mathcal{L}(y_t)}{dW_x} = \frac{d\mathcal{L}(y_t)}{dy_t}\frac{dy_t}{dh_t}\frac{h_t}{dW_x}$ and $\frac{d\mathcal{L}(y_t)}{db_h} = \frac{d\mathcal{L}(y_t)}{dy_t}\frac{dy_t}{dh_t}\frac{dh_t}{db_h}$. However, W_x, b_h contribute to h_t directly and indirectly. Next bullet-point describes why we also need $\frac{dh_t}{dW_x} = \frac{\partial h_t}{\partial W_x} + \frac{dh_t}{dh_{t-1}}\frac{dh_{t-1}}{dW_x}$ and $\frac{dh_t}{db_h} = \frac{\partial h_t}{\partial b_h} + \frac{dh_t}{dh_{t-1}}\frac{dh_{t-1}}{db_h}$
- $\frac{d\mathcal{L}(y_t)}{dW_h} = \frac{d\mathcal{L}(y_t)}{dy_t}\frac{dy_t}{dh_t}\frac{h_t}{dW_h}$ The entries in the Jacobian $\frac{dh_t}{dW_h}$ contains the total gradient of $h_t[i]$ w.r.t $W_h[m,n]$. h_t depends on h_{t-1} which depends W_h. Therefore: $\frac{dh_t}{dW_h} = \frac{\partial h_t}{\partial W_h} + \frac{dh_t}{dh_{t-1}}\frac{dh_{t-1}}{dW_h}$
  If we continue expanding: $\frac{dh_t}{dW_h} = \frac{\partial h_t}{\partial W_h} + \frac{dh_t}{dh_{t-1}}(\frac{\partial h_{t-1}}{\partial W_h} + \frac{dh_{t-1}}{dh_{t-2}}\frac{dh_{t-2}}{dW_h}) = \sum_{\tau=1}^{t}(\prod_{l=\tau}^{t-1}\frac{dh_{l+1}}{dh_l})\frac{\partial h_\tau}{\partial W_h}$ which results in the Back-propogation through time. We may truncate this to $\sum_{\tau=\max(1,t-L)}^{t}$.
  $\prod$ contains products of activation and weight matrix; gradient vanish/explode as $t \to \infty$! Therefore the long-term dependencies become harder to learn.



LSTM:
f_t : forget gate $= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$
i_t : input gate $= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
o_t : output gate $= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
$\tilde{c}_t$: candidate cell state $= \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$
c_t : memory cell state $= f_t \odot c_{t-1} + i_c \odot \tilde{c}_t$
h_t : hidden state $= o_t \odot \tanh(c_t)$