

LSTM (Long Short-Term Memory)

- Goal: To remember important information for long sequences while avoiding gradients.

- Math:

An LSTM keeps a cell state c_t that acts like long-term memory and is controlled by three gates.

- The forget gate decides what to remove:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$$

- The input gate decides what new information to add:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$$

- A candidate memory is created:

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c)$$

- The output gate controls the visible hidden state:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \quad h_t = o_t \odot \tanh(c_t)$$

- Conclusion:

By using gates and a separate cell state, LSTM can keep or forget information selectively, making it effective for long-term dependencies in sequences.