

APRENDIZAJE PROFUNDO

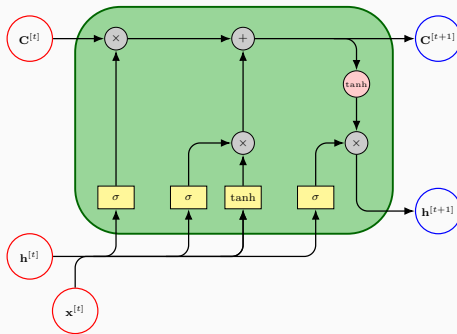
OTRAS CELDAS RECURRENTES

Gibran Fuentes Pineda

Septiembre 2025

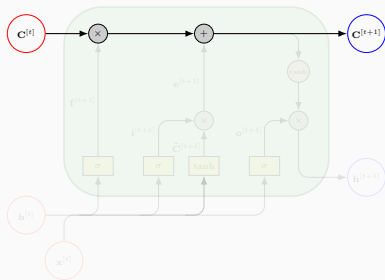
LONG-SHORT TERM MEMORY (LSTM)

- Agregan elementos internos a la celda básica que permiten capturar dependencias a corto y largo plazo



LONG-SHORT TERM MEMORY (LSTM): SALIDA DE LA CAPA ANTERIOR

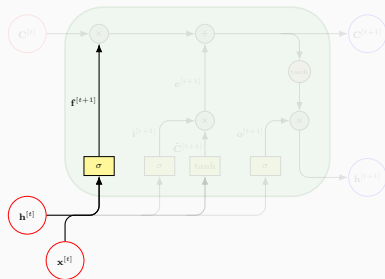
- Agrega o elimina información del estado $\mathbf{c}^{[t]}$ a partir de la entrada actual $\mathbf{x}^{[t+1]}$ y la salida anterior $\mathbf{h}^{[t]}$



LONG-SHORT TERM MEMORY (LSTM): COMPUERTA DE OLVIDO

- Determina qué olvidar del estado $C^{[t]}$ y en qué proporción a partir de la entrada actual $x^{[t+1]}$ y la salida anterior $h^{[t]}$

$$f^{[t+1]} = \sigma \left(W_f \cdot [h^{[t]}, x^{[t+1]}] + b_f \right)$$



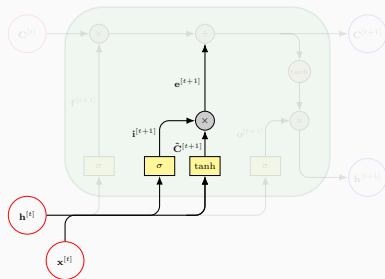
LONG-SHORT TERM MEMORY (LSTM): COMPUTERTA DE ENTRADA

- Determina qué agregar al estado $C^{[t]}$ y en qué proporción a partir de la entrada actual $x^{[t+1]}$ y el estado oculto anterior $h^{[t]}$

$$i^{[t+1]} = \sigma \left(W_i \cdot [h^{[t]}, x^{[t+1]}] + b_i \right)$$

$$\hat{C}^{[t+1]} = \tanh \left(W_C \cdot [h^{[t]}, x^{[t+1]}] + b_C \right)$$

$$e^{[t+1]} = i^{[t+1]} \odot \hat{C}^{[t+1]}$$

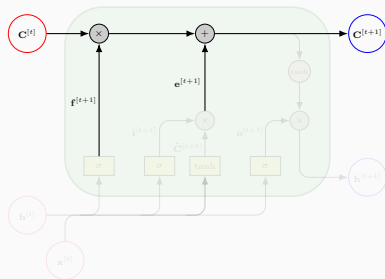


LONG-SHORT TERM MEMORY (LSTM): NUEVO ESTADO

- El nuevo estado $\mathbf{C}^{[t+1]}$ se obtiene como una combinación del estado $\mathbf{C}^{[t]}$, la salida $\mathbf{f}^{(t)}$ de la compuerta de olvido y la salida $\mathbf{e}^{[t+1]}$ de la compuerta de entrada

$$\mathbf{C}^{[t+1]} = \mathbf{f}^{[t+1]} \odot \mathbf{C}^{[t]} + \mathbf{e}^{[t+1]}$$

donde \odot denota el producto de Hadamard

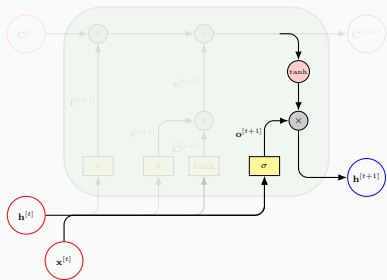


LONG-SHORT TERM MEMORY (LSTM): COMPUTERTA DE SALIDA

- El siguiente estado oculto $h^{[t+1]}$ se obtiene como una combinación de la entrada actual $x^{[t+1]}$, el estado oculto anterior $h^{[t]}$ y el nuevo estado $c^{[t+1]}$

$$o^{[t+1]} = \sigma(W_o \cdot [h^{[t]}, x^{[t+1]}] + b_o)$$

$$h^{[t+1]} = o^{[t+1]} \odot \tanh(c^{[t+1]})$$



- Combina compuertas de olvido y entrada en una sola

$$\mathbf{z}^{[t+1]} = \sigma \left(\mathbf{W}_z \cdot \left[\mathbf{h}^{[t]}, \mathbf{x}^{[t+1]} \right] + \mathbf{b}_z \right)$$

$$\mathbf{r}^{[t+1]} = \sigma \left(\mathbf{W}_r \cdot \left[\mathbf{h}^{[t]}, \mathbf{x}^{[t+1]} \right] + \mathbf{b}_r \right)$$

$$\tilde{\mathbf{h}}^{[t+1]} = \tanh \left(\mathbf{W}_h \cdot \left[\mathbf{r}^{[t+1]} \odot \mathbf{h}^{[t]}, \mathbf{x}^{[t+1]} \right] + \mathbf{b}_h \right)$$

$$\mathbf{h}^{[t+1]} = \left(1 - \mathbf{z}^{[t+1]} \right) \odot \mathbf{h}^{[t]} + \mathbf{z}^{[t+1]} \odot \tilde{\mathbf{h}}^{[t+1]}$$

¹K. Cho et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv:1406.1078*, 2014.