

# Aprendizaje profundo

## OTRAS CELDAS RECURRENTES

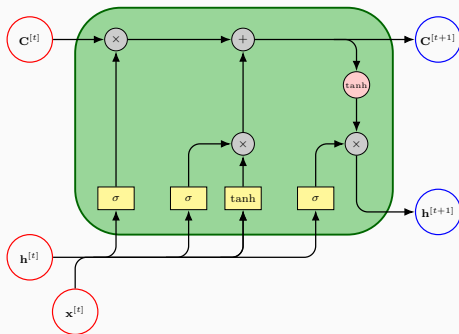
---

Gibran Fuentes Pineda

5 de octubre de 2023

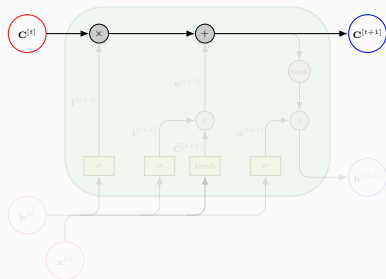
# Long-short term memory (LSTM)

- 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

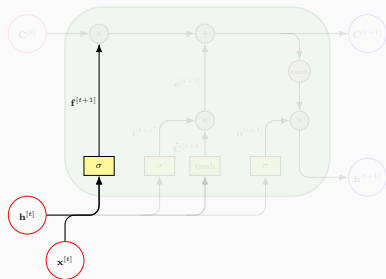
- Agrega o elimina información del estado  $\mathbf{C}^{[t]}$  a partir de la entrada actual  $\mathbf{x}^{[t+1]}$  y la salida anterior  $\mathbf{h}^{[t]}$



# 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  $h^{[t]}$

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



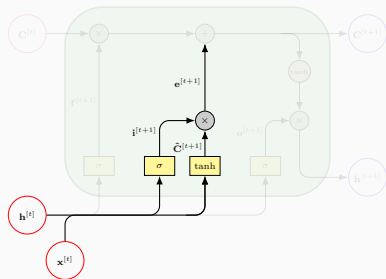
# Long-short term memory (LSTM): computerta de entrada

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

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

$$\hat{\mathbf{C}}^{[t+1]} = \tanh \left( \mathbf{W}_c \cdot \left[ \mathbf{h}^{[t]}, \mathbf{x}^{[t+1]} \right] + \mathbf{b}_c \right)$$

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

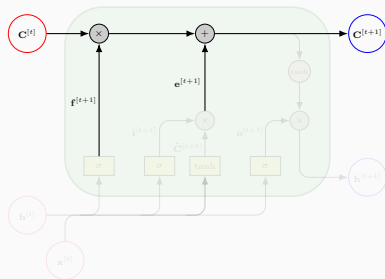


# Long-short term memory (LSTM): nuevo estado

- 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

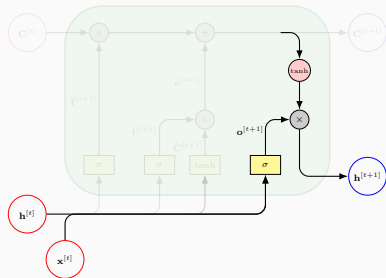


# Long-short term memory (LSTM): computerta de salida

- 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  $c^{[t+1]}$

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

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



# Gated recurrent unit (GRU)<sup>1</sup>

- Combina compuertas de olvido y entrada en una sola

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

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

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

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

---

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