

Redes Recurrentes

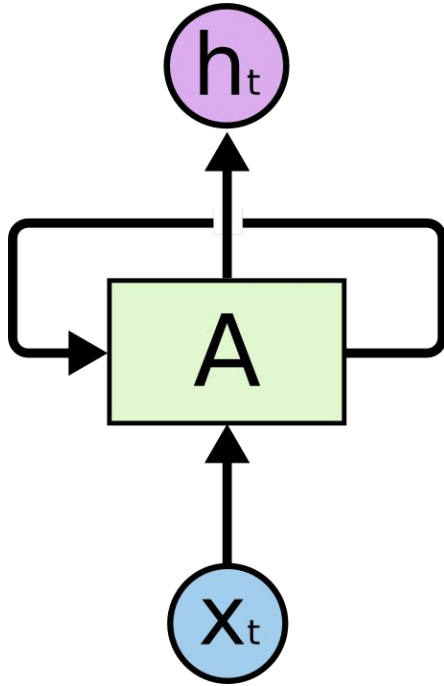
Orlando Ramos Flores

Contenido

- Introducción
- Long Short-Term Memory (LSTM)
 - Célula de memoria (primer definición)
 - Puerta de entrada, puerta de olvido y puerta de salida
 - Célula de memoria candidata
 - Célula de memoria
 - Estado oculto
 - Resumen

Introducción

Introducción: Red Recurrente



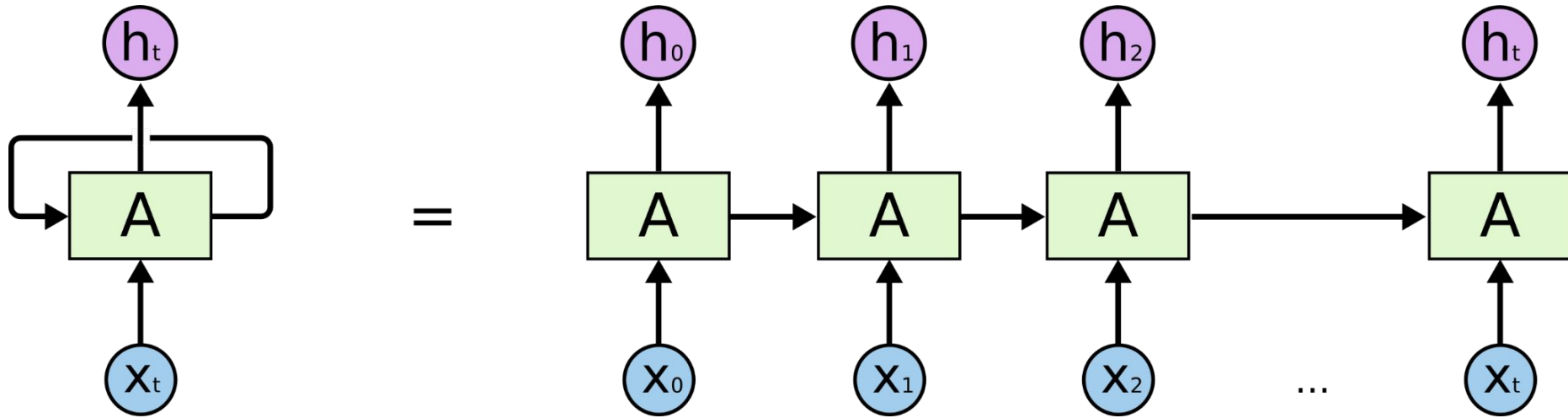
A: Red Recurrente

x_t : Entradas

h_t : Salidas

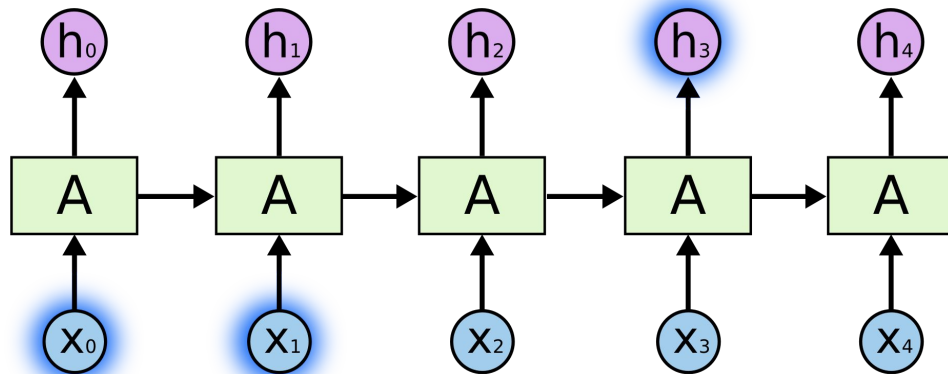
Un bucle permite que la información pase de un paso de la red al siguiente.

Introducción: Despliegue Red Recurrente



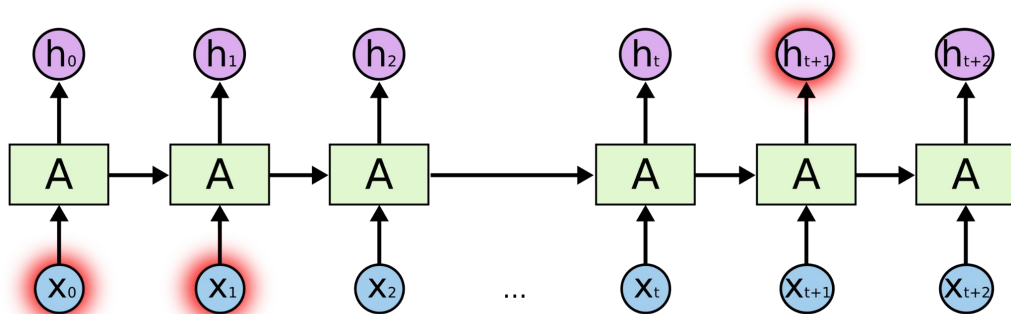
Introducción: Redes Recurrentes

- A veces, solo necesitamos mirar información reciente para realizar la tarea actual.
- Por ejemplo, considerar un modelo de lenguaje que intenta predecir la siguiente palabra en función de las anteriores.
- Si estamos tratando de predecir la última palabra en “*las nubes están en el cielo*”, no necesitamos más contexto: es bastante obvio que la siguiente palabra será “*cielo*”.
- En tales casos, donde la brecha entre la información relevante y el lugar donde se necesita sea pequeña, las RNN pueden aprender a usar la información pasada.



Introducción: Redes Recurrentes

- Pero también hay casos en los que necesitamos más contexto.
- Por ejemplo, tratar de predecir la última palabra en el texto "*Crecí en Francia... hablo con fluidez el francés*".
- La información reciente sugiere que la siguiente palabra probablemente sea el nombre de un idioma, pero si queremos acotar qué idioma, necesitamos el contexto de Francia, desde más atrás.
- Es completamente posible que la brecha entre la información relevante y el punto donde se necesita se vuelva muy grande.



Introducción: Redes Recurrentes

- En teoría, las RNN son absolutamente capaces de manejar tales "dependencias a largo plazo (Long-Term)".
- Aunque se podría elegir cuidadosamente los parámetros para resolver problemas pequeños de esta forma.
- Lamentablemente, en la práctica, las RNN no parecen poder aprender correctamente.
- El problema es conocido como *gradientes que se desvanecen y explotan* (Vanishing and exploding gradients)

Long Short-Term Memory LSTM

Long Short-Term Memory: LSTM

- El diseño de la LSTM de Hochreiter, S., & Schmidhuber, comparte muchas de las propiedades de la GRU. Las LSTM tienen un diseño un poco más complejo que las GRU, pero son anteriores a las GRU en casi dos décadas.
- LSTM introduce una **celda de memoria** que tiene la misma forma que el estado oculto (algunas literaturas consideran la celda de memoria como un tipo especial del estado oculto), diseñada para registrar información adicional.
- Para controlar la celda de memoria se necesitan varias puertas.
- Se necesita una puerta para leer las entradas de la celda (**output gate**).
- Se necesita una segunda puerta para decidir cuándo leer datos en la celda (**input gate**).
- Por último, necesitamos un mecanismo para restablecer el contenido de la celda, gobernado por una puerta de olvido (**forget gate**).
- La motivación para tal diseño es la misma que la de las GRU, es decir, poder decidir cuándo recordar y cuándo ignorar las entradas en el estado oculto a través de un mecanismo dedicado.

LSTM: Célula de memoria

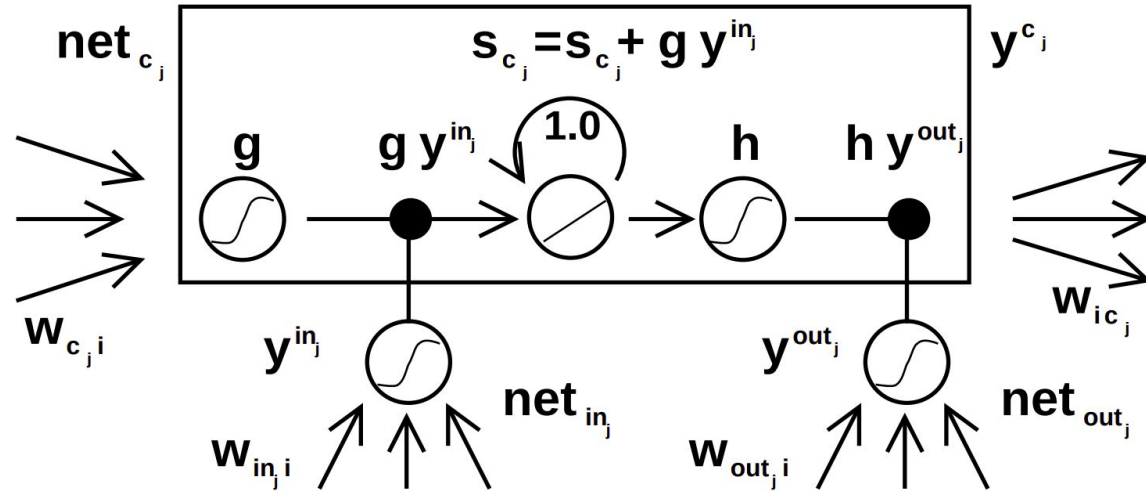
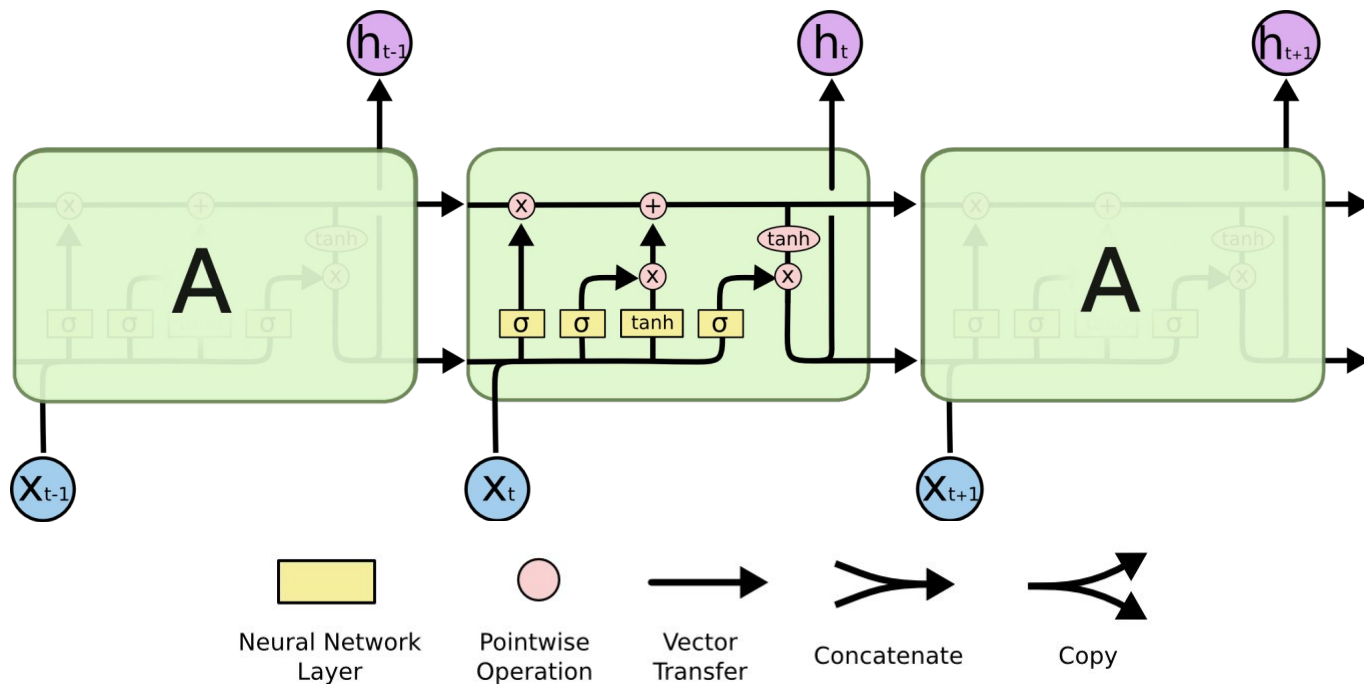
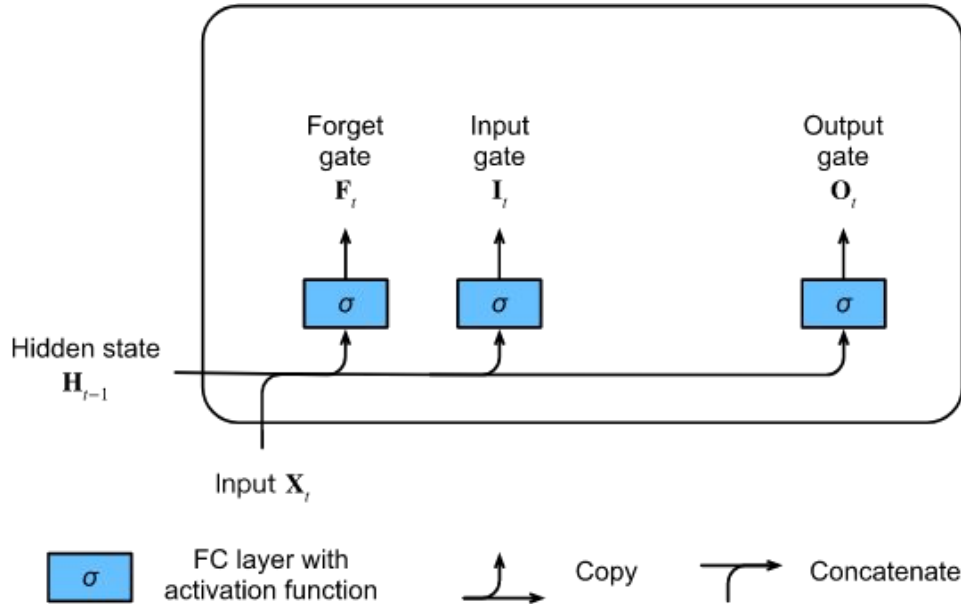


Figure 1: Architecture of memory cell c_j (the box) and its gate units in_j, out_j . The self-recurrent connection (with weight 1.0) indicates feedback with a delay of 1 time step. It builds the basis of the “constant error carousel” CEC. The gate units open and close access to CEC. See text and appendix A.1 for details.

LSTM: Célula de memoria

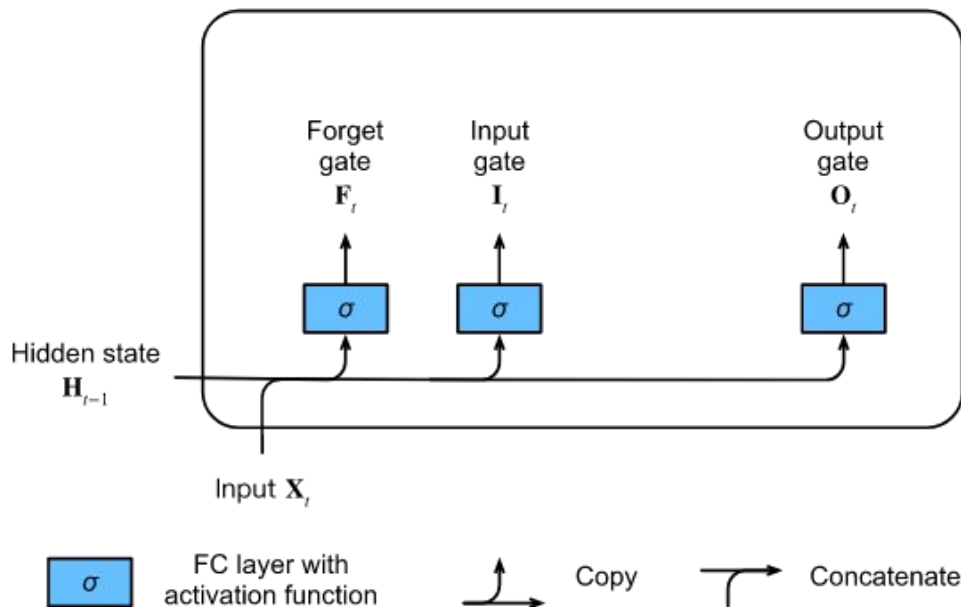


LSTM: Puertas de entrada, olvido y salida



- Al igual que en las GRU, los datos que ingresan a las puertas LSTM son la entrada en el paso de tiempo actual y el estado oculto del paso de tiempo anterior.
- Son procesados por tres capas totalmente conectadas con una función de activación sigmoideal para calcular los valores de las puertas de entrada, olvido y de salida.
- Como resultado, los valores de las tres puertas están en el rango de $(0,1)$.

LSTM: Puertas de entrada, olvido y salida



Supongamos que hay h unidades ocultas, el tamaño del batch es n , y el número de entradas es d . Entonces:

- Entradas $X_t \in \mathbb{R}^{n \times d}$
- El estado oculto del paso de tiempo anterior es $H_{t-1} \in \mathbb{R}^{n \times h}$

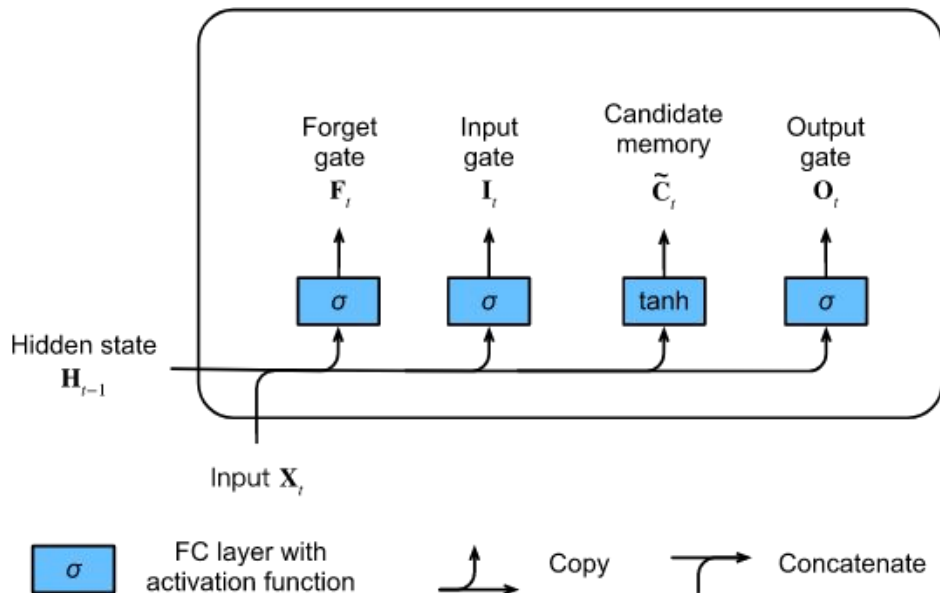
Las puertas en el tiempo t se definen:

$$I_t = \sigma(X_t W_{xi} + H_{t-1} W_{hi} + b_i), \quad O_t = \sigma(X_t W_{xo} + H_{t-1} W_{ho} + b_o)$$

$$F_t = \sigma(X_t W_{xf} + H_{t-1} W_{hf} + b_f)$$

$W_{xi}, W_{xf}, W_{xo} \in \mathbb{R}^{d \times h}$ y $W_{hi}, W_{hf}, W_{ho} \in \mathbb{R}^{h \times h}$ son parámetros de peso y $b_i, b_f, b_o \in \mathbb{R}^{1 \times h}$ son parámetros de sesgo.

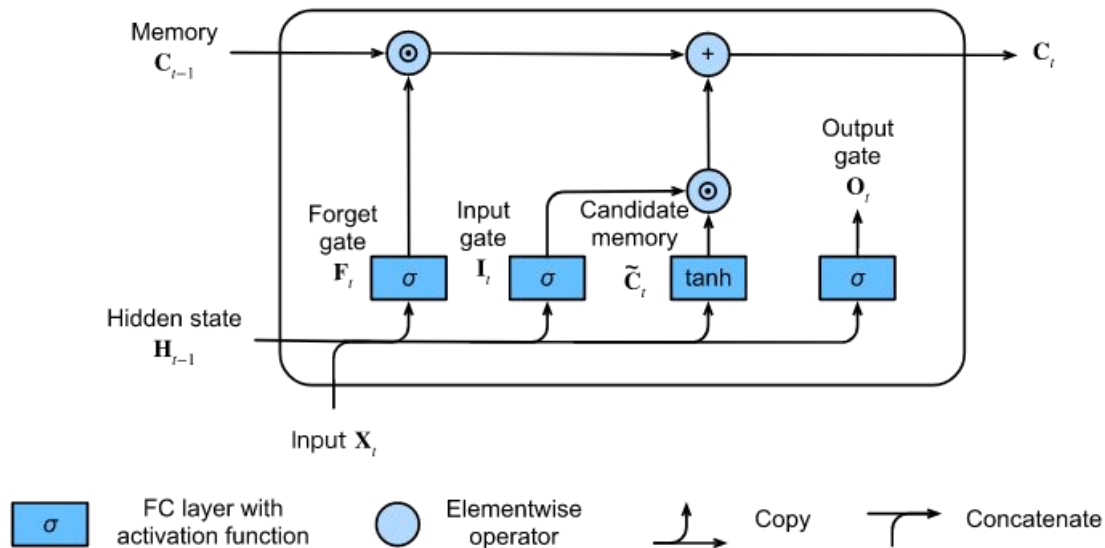
LSTM: Célula de memoria candidata



- Sea la célula de memoria $\tilde{C}_t \in \mathbb{R}^{n \times h}$
- Su cálculo es similar al de las tres puertas descritas anteriormente, pero utilizando una función \tanh con un rango de valores para $(-1,1)$ como función de activación.

$$\tilde{C} = \tanh(X_t W_{xc} + H_{t-1} W_{hc} + b_c)$$

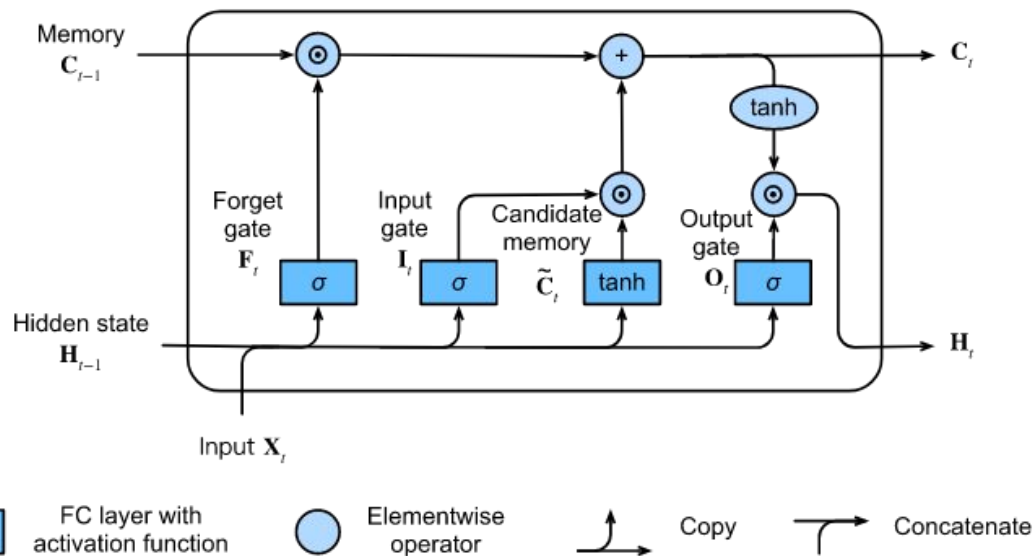
LSTM: Célula de memoria



$$C_t = F_t \odot C_{t-1} + I_t \odot \tilde{C}_t$$

- La puerta I_t rige cuánto se toma en cuenta de los nuevos datos a través de \tilde{C}_t , y la puerta F_t rige la cantidad del contenido de la celda de memoria antigua $\tilde{C}_{t-1} \in \mathbb{R}^{n \times h}$ que se retiene.
- Si la **puerta de olvido** siempre se aproxima a **1** y la **puerta de entrada** siempre se aproxima a **0**, las celdas de memoria pasadas C_{t-1} se guardarán con el tiempo y pasarán al paso de tiempo actual.
- *Este diseño se presenta para mitigar el problema del desvanecimiento del gradiente y capturar mejor las dependencias de largo alcance dentro de las secuencias.*

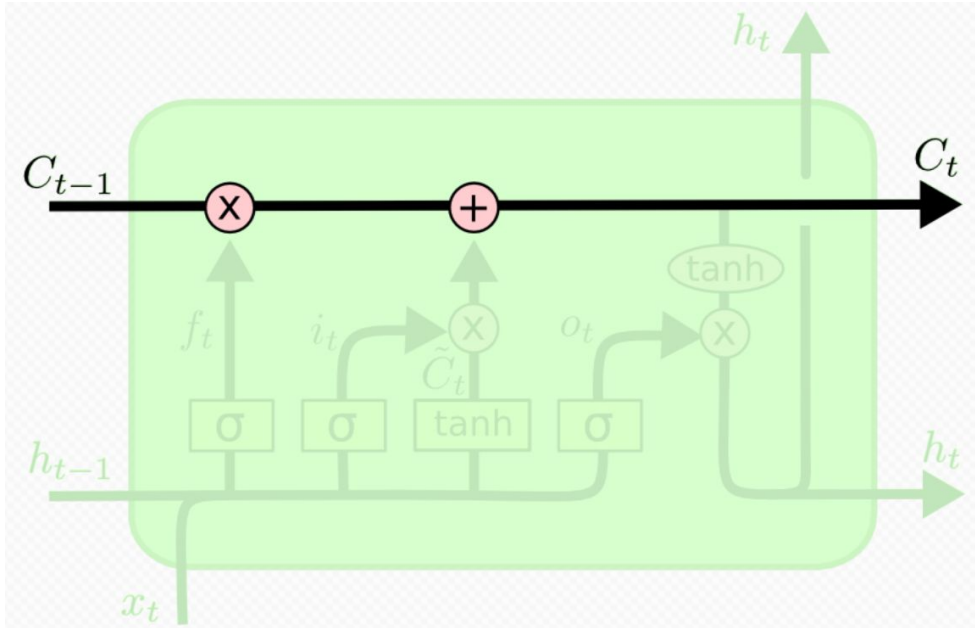
LSTM: Estado oculto



$$H_t = O_t \odot \tanh(C_t)$$

- El estado oculto sea $H_t \in \mathbb{R}^{n \times h}$
- Aquí se realizan los cálculos con la puerta de salida.
- En la LSTM es simplemente una versión cerrada del \tanh de la célula de memoria.
- Esto asegura que los valores de H_t estén siempre en el intervalo $(-1, 1)$.
- Cada vez que la **puerta de salida** se aproxima a **1**, se pasa toda la información de la memoria, y
- Cada vez que la **puerta de salida** se aproxima a **0**, se retiene toda la información solo dentro de la celda de memoria y no se realiza más procesamiento.

LSTM: Célula de memoria



- La clave de las LSTM es el estado de memoria de la célula (C_t) la línea horizontal que atraviesa la parte superior del diagrama.
- El estado C_t es como una cinta transportadora.
- Se ejecuta directamente a lo largo de toda la cadena, con solo algunas interacciones lineales menores.
- Es muy fácil que la información fluya sin cambios.

Resumen: Puerta de Olvido

- Esta puerta decide qué información debe desecharse o conservarse.
- La información del estado oculto anterior y la información de la entrada actual se pasan a través de la función *sigmoide*.
- Los valores salen entre 0 y 1.
- Cuanto más cerca de 0 sea el valor significa olvidar, y cuanto más cerca de 1 significa mantener.

Resumen: Puerta de Entrada

- Para actualizar el estado de la célula, tenemos la puerta de entrada.
- Primero, pasamos el estado oculto anterior y la entrada actual a una función *sigmoide*.
- Eso decide qué valores se actualizarán transformando los valores para que estén entre 0 y 1.
- 0 significa que no es importante y 1 significa que es importante.
- También pasa el estado oculto y la entrada actual a la función *tanh* para reducir los valores entre -1 y 1 para ayudar a regular la red.
- Luego, multiplica la salida de tanh con la salida *sigmoide*.
- La salida *sigmoide* decidirá qué información es importante mantener de la salida *tanh*.

Resumen: Célula de Memoria

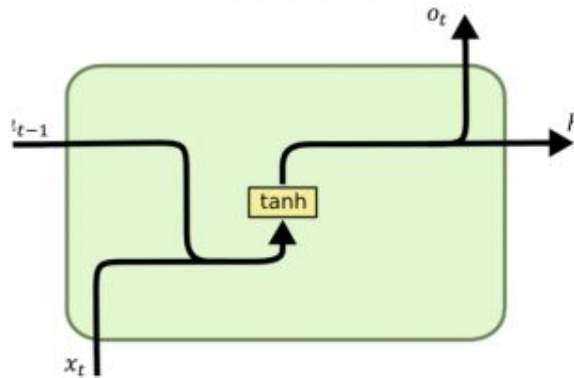
- En este punto se tiene suficiente información para calcular el estado de la célula.
- Primero, el estado de la célula se multiplica por el vector de olvido.
- Esto tiene la posibilidad de eliminar valores en el estado de la célula si se multiplica por valores cercanos a 0.
- Luego, se toma la salida de la puerta de entrada y se hace una suma que actualiza el estado de la celda a nuevos valores que la red neuronal considera relevantes.
- Eso proporciona un nuevo estado de la célula.

Resumen: Puerta de Salida

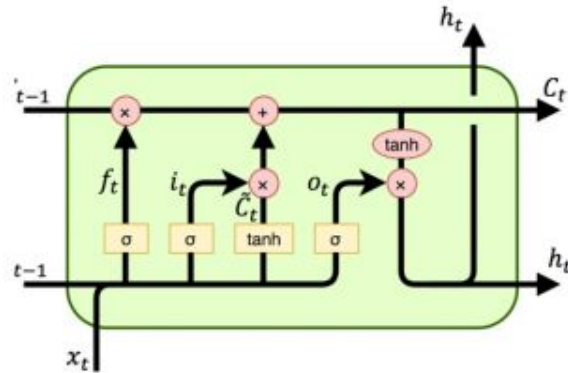
- La puerta de salida decide cuál debería ser el siguiente estado oculto.
- Recordar que el estado oculto contiene información sobre entradas anteriores.
- El estado oculto también se usa para predicciones.
- Primero, se pasa el estado oculto anterior y la entrada actual a una función *sigmoide*.
- Luego se pasa el estado de célula recién modificado a la función *tanh*.
- Se multiplica la salida de *tanh* con la salida *sigmoide* para decidir qué información debe llevar el estado oculto.
- La salida es el estado oculto.
- El nuevo estado de la célula y el nuevo estado oculto se transfieren al siguiente paso de tiempo.

Resumen: RNN, LSTM, GRU

RNN



LSTM



GRU

