REDES NEURAIS RECORRENTES

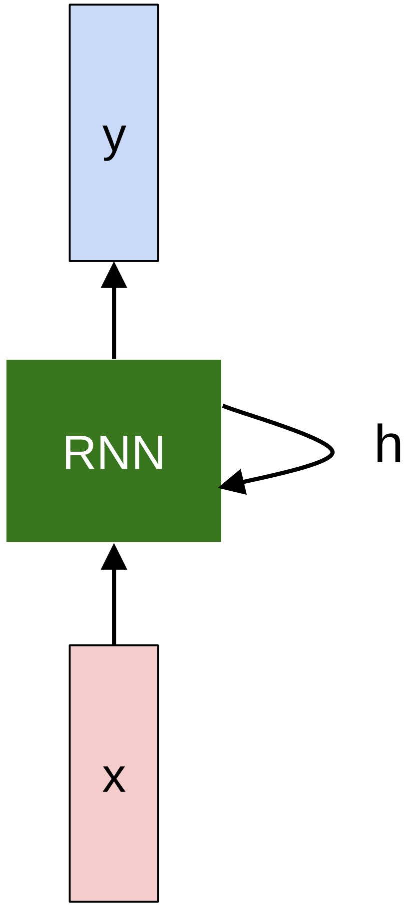
LSTM (*LONG SHORT TERM MEMORY*)

Zenilton K. G. Patrocínio Jr

zenilton@pucminas.br

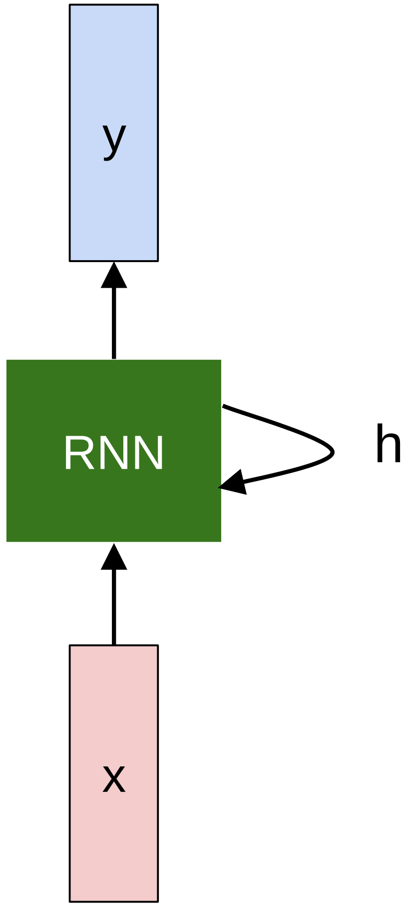
Rede Neural Recorrente (RNN)

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$



Rede Neural Recorrente (RNN)

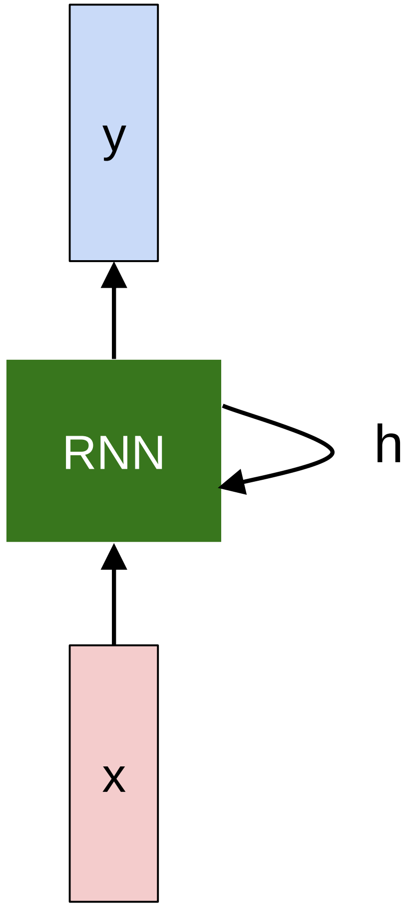
$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$



Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

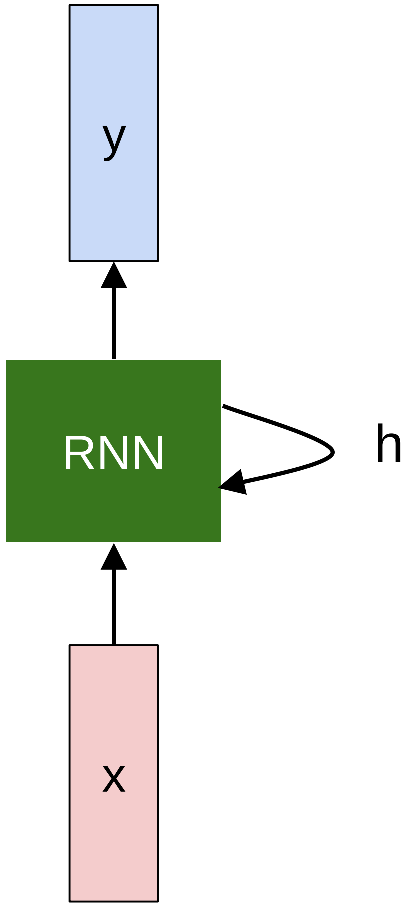


Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

$$\frac{\partial h_t}{\partial h_{t-1}} = \operatorname{sech}^2 \hat{h} \times W_{hh}$$

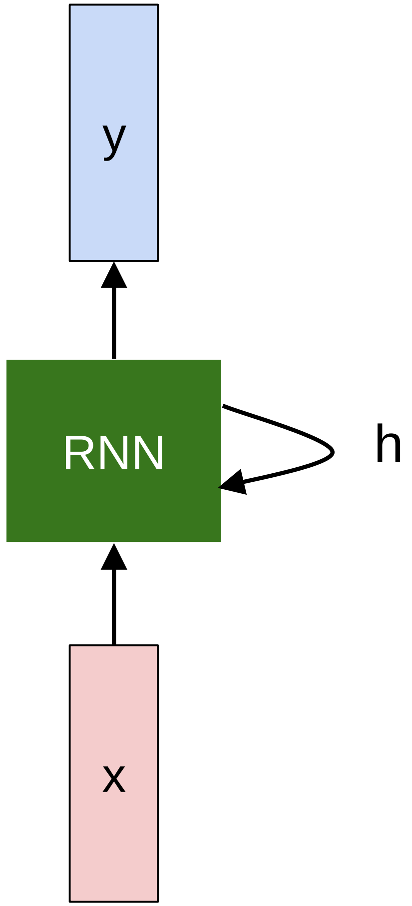


Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

$$\frac{\partial h_t}{\partial h_{t-1}} = \operatorname{sech}^2 \hat{h} \times W_{hh}$$

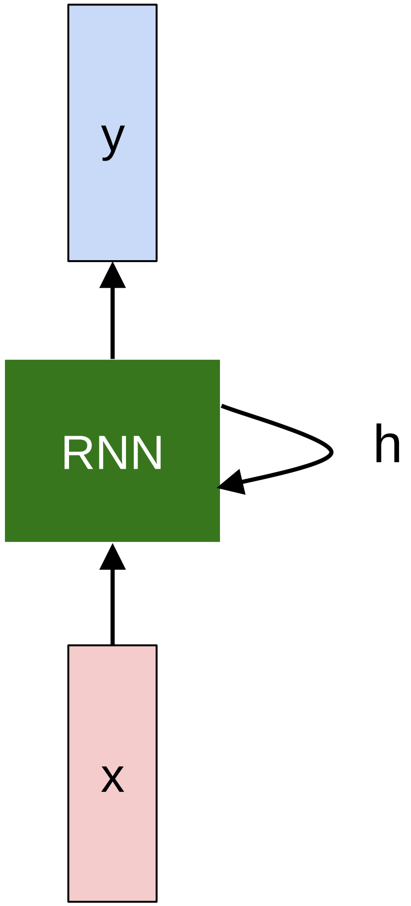


Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

$$\frac{\partial h_t}{\partial h_{t-1}} = \underbrace{\operatorname{sech}^2 \hat{h}}_{\leq 1} \times W_{hh}$$



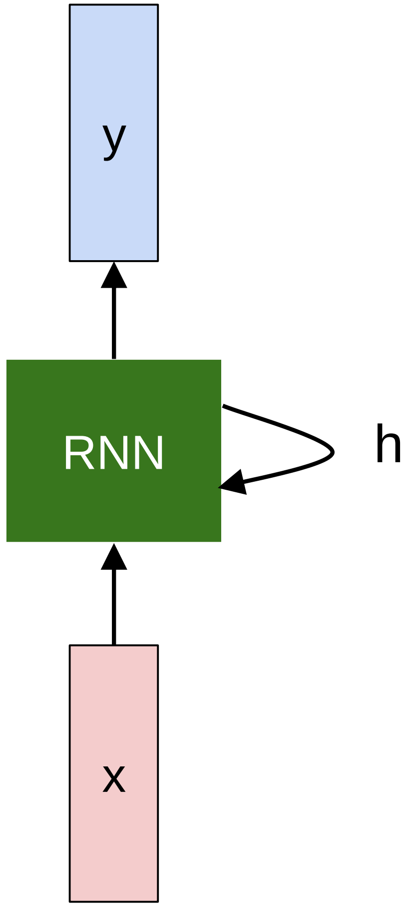
Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

$$\frac{\partial h_t}{\partial h_{t-1}} = \underbrace{\mathrm{sech}^2\, \hat{h}}_{\leq 1} \times W_{hh}$$

W_{hh} é multiplicado pelo gradiente no próximo passo, logo, o gradiente ao longo do tempo é aproximadamente uma potência de W_{hh}



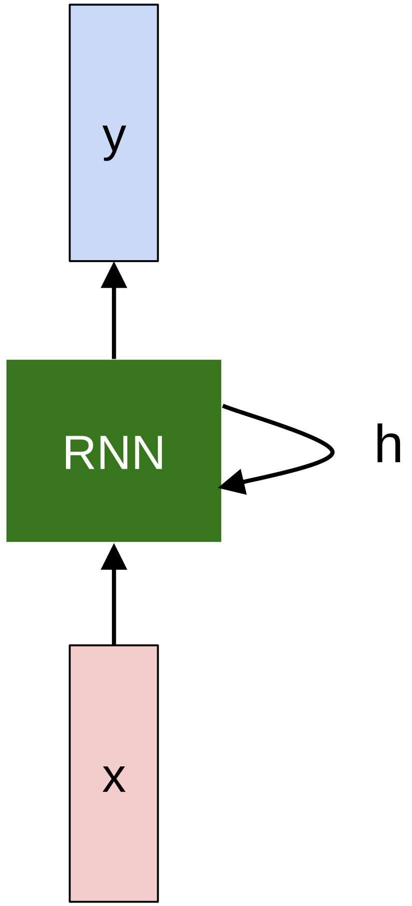
Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh} h_{t-1} + W_{xh} x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

$$\frac{\partial h_t}{\partial h_{t-1}} = \underbrace{\operatorname{sech}^2 \hat{h}}_{\leq 1} \times W_{hh}$$

W_{hh} é multiplicado pelo gradiente no próximo passo, logo, o gradiente ao longo do tempo é aproximadamente uma potência de W_{hh}



Se o maior autovalor de W_{hh} > 1, os gradientes crescerão com o tempo (explodindo)

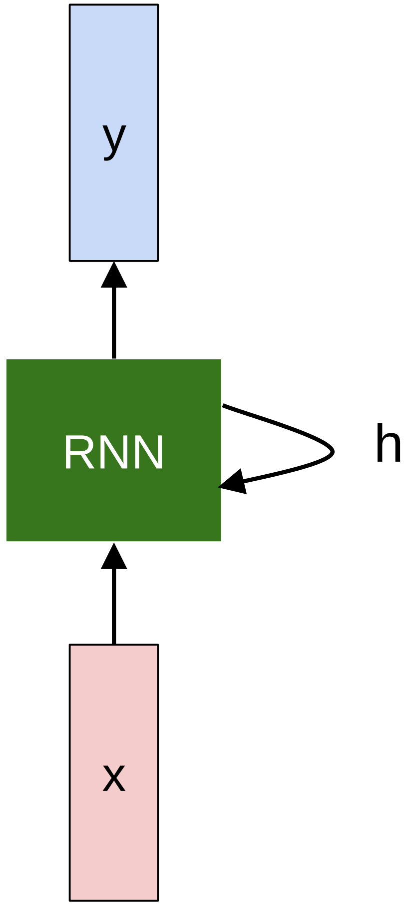
Rede Neural Recorrente (RNN)

$$h_t = \tanh(\overbrace{W_{hh}h_{t-1} + W_{xh}x_t}^{\hat{h}})$$

No ***backprop*** será necessário $\dfrac{\partial h_t}{\partial h_{t-1}} = \dfrac{\partial h_t}{\partial \hat{h}} \times \dfrac{\partial \hat{h}}{\partial h_{t-1}}$

$$\frac{\partial h_t}{\partial h_{t-1}} = \underbrace{\operatorname{sech}^2 \hat{h}}_{\leq 1} \times W_{hh}$$

W_{hh} é multiplicado pelo gradiente no próximo passo, logo, o gradiente ao longo do tempo é aproximadamente uma potência de W_{hh}



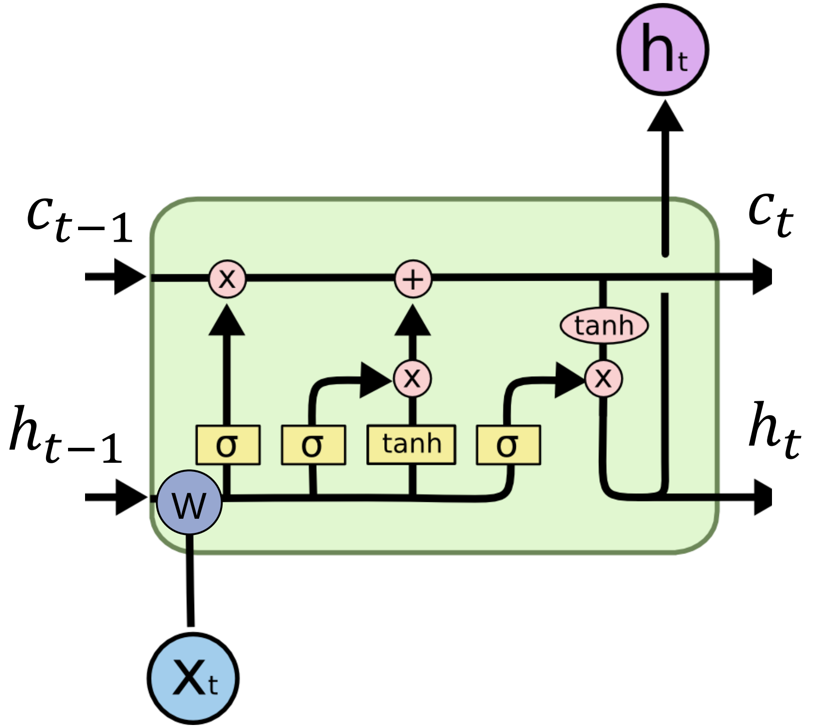
Se o maior autovalor de W_{hh} > 1, os gradientes crescerão com o tempo (explodindo)

Se o maior autovalor de W_{hh} < 1, os gradientes reduzirão com o tempo (desaparecendo)

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

Unidades LSTM (*Long Short-Term Memory*) possuem uma memória de célula c_i que é repassada de um instante de tempo para o próximo (sem transformação não linear)

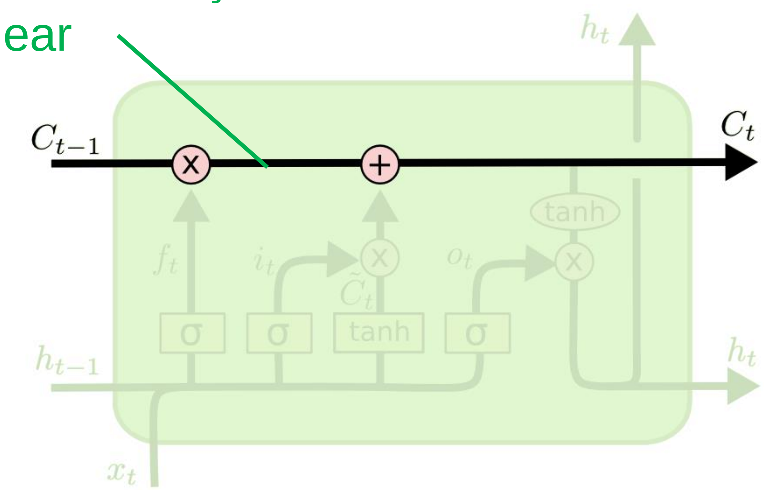


LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

Unidades LSTM (*Long Short-Term Memory*) possuem uma memória de célula c_i que é repassada de um instante de tempo para o próximo (sem transformação não linear)

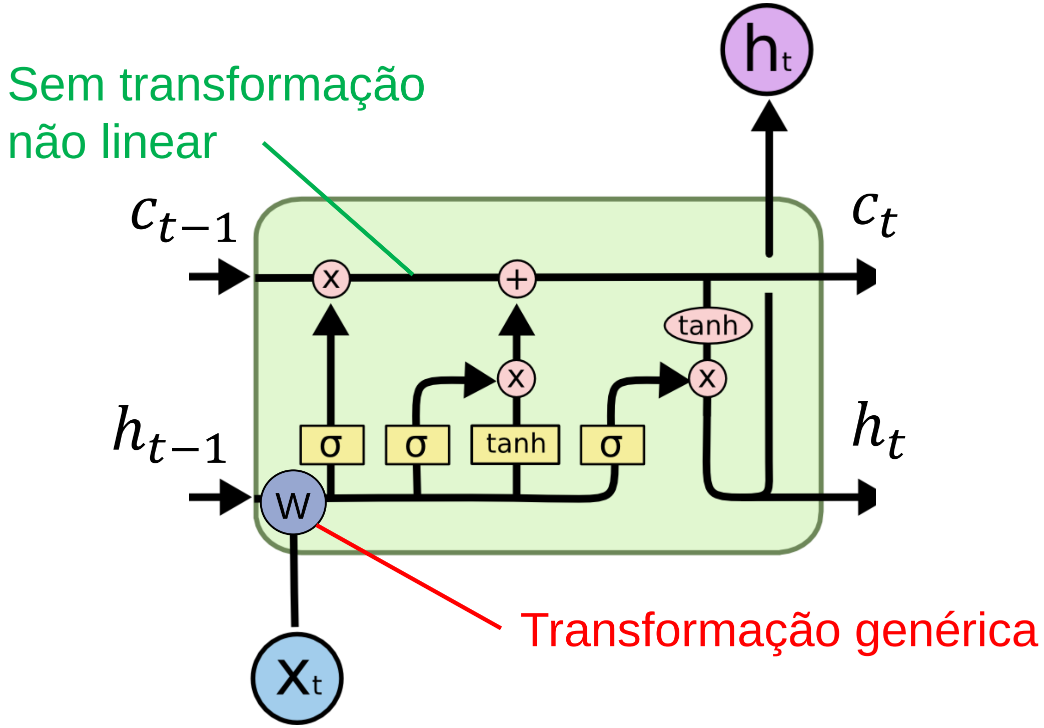
Sem transformação
não linear



LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

Unidades LSTM (*Long Short-Term Memory*) possuem uma memória de célula c_i que é repassada de um instante de tempo para o próximo (sem transformação não linear)



Elas também possuem informações de estado “escondidas” que passam por transformações não lineares como uma RNN tradicional

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

- Existem dois elementos recorrentes c_i e h_i

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

- Existem dois elementos recorrentes c_i e h_i
- h_i assume o papel da saída de uma RNN tradicional

LSTM – *Long Short Term Memory*

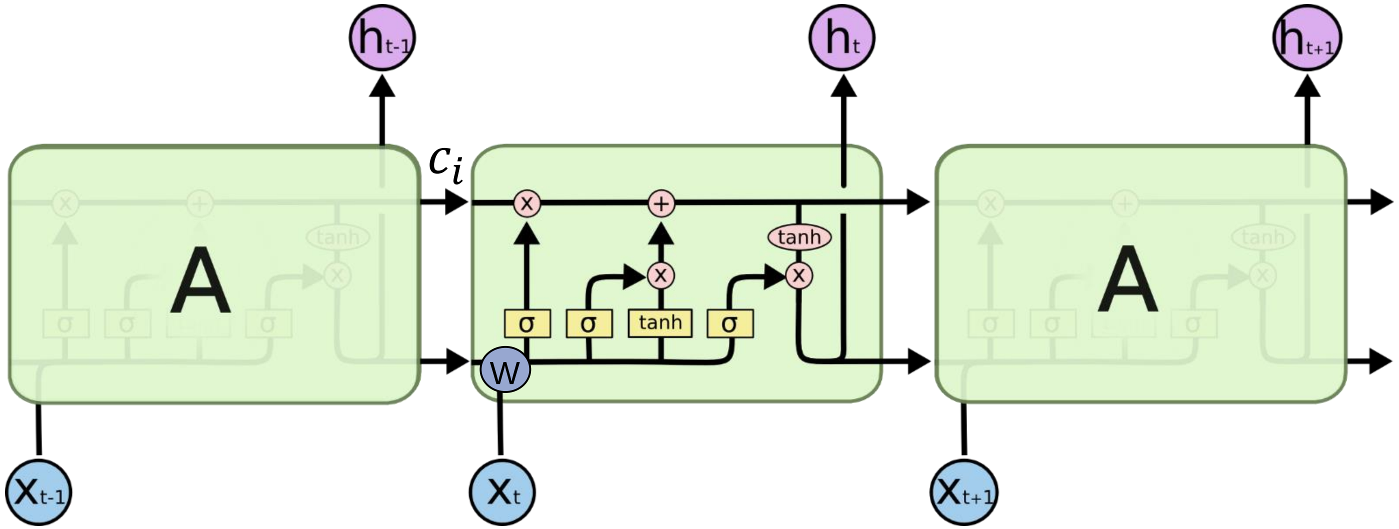
[Hochreiter et al., 1997]

- Existem dois elementos recorrentes c_i e h_i
- h_i assume o papel da saída de uma RNN tradicional
- O estado da célula c_i representa a memória da mesma e não passa por transformação

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

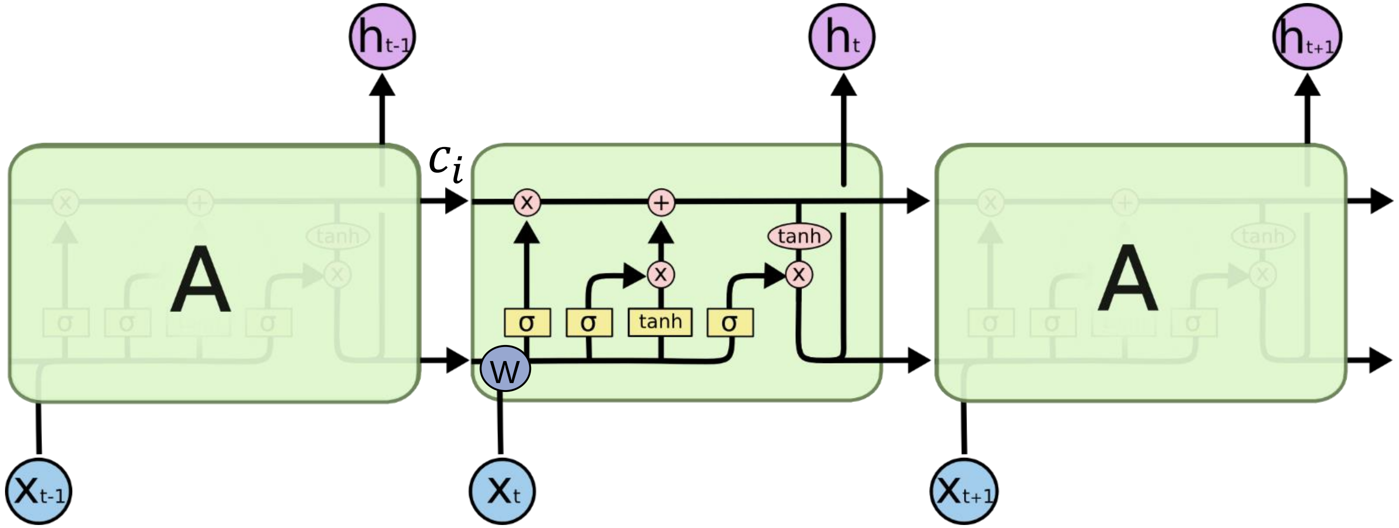
- Existem dois elementos recorrentes c_i e h_i
- h_i assume o papel da saída de uma RNN tradicional
- O estado da célula c_i representa a memória da mesma e não passa por transformação
- LSTM podem ser estendida no tempo (como uma RNN)



LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

- Existem dois elementos recorrentes c_i e h_i
- h_i assume o papel da saída de uma RNN tradicional
- O estado da célula c_i representa a memória da mesma e não passa por transformação
- LSTM podem ser estendida no tempo (como uma RNN)



- Para várias camadas, as saídas h_is de uma camada tornam-se as entradas x_is da próxima

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$
$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \mathrm{sigm} \\ \mathrm{sigm} \\ \mathrm{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$
$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$W^l \quad [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$W^l \quad [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$W^l \quad [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$
$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM pode simular uma simples RNN com valores adequados para $\boldsymbol{i}$, $\boldsymbol{f}$ e $\boldsymbol{o}$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$W^l \quad [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$$c_t^l = \underset{0}{f} \odot c_{t-1}^l + \underset{1}{i} \odot g$$

$$h_t^l = \underset{1}{o} \odot \tanh(c_t^l)$$

LSTM pode simular uma simples RNN com valores adequados para $\boldsymbol{i}$, $\boldsymbol{f}$ e $\boldsymbol{o}$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

RNN:

$$h_t^l = \tanh W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$h \in \mathbb{R}^n$

$W^l \; [n \times 2n]$

LSTM:

$W^l \; [4n \times 2n]$

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

Entrada da camada inferior $\equiv x$

Entrada do passo anterior

$$c_t^l = \underset{0}{f} \odot c_{t-1}^l + \underset{1}{i} \odot g$$

$$h_t^l = \underset{1}{o} \odot \tanh(c_t^l)$$

LSTM pode simular uma simples RNN com valores adequados para i, f e o

QUASE!

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \mathrm{sigm} \\ \mathrm{sigm} \\ \mathrm{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

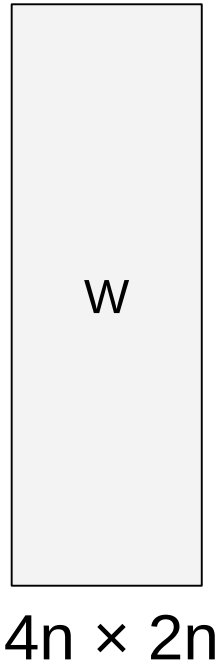
$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



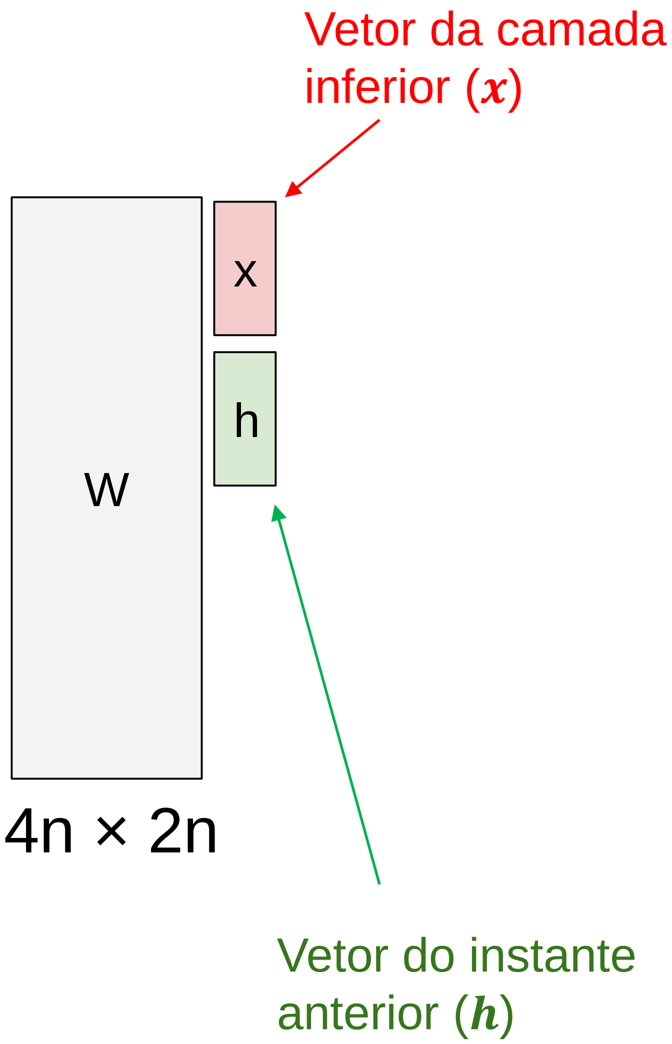
$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



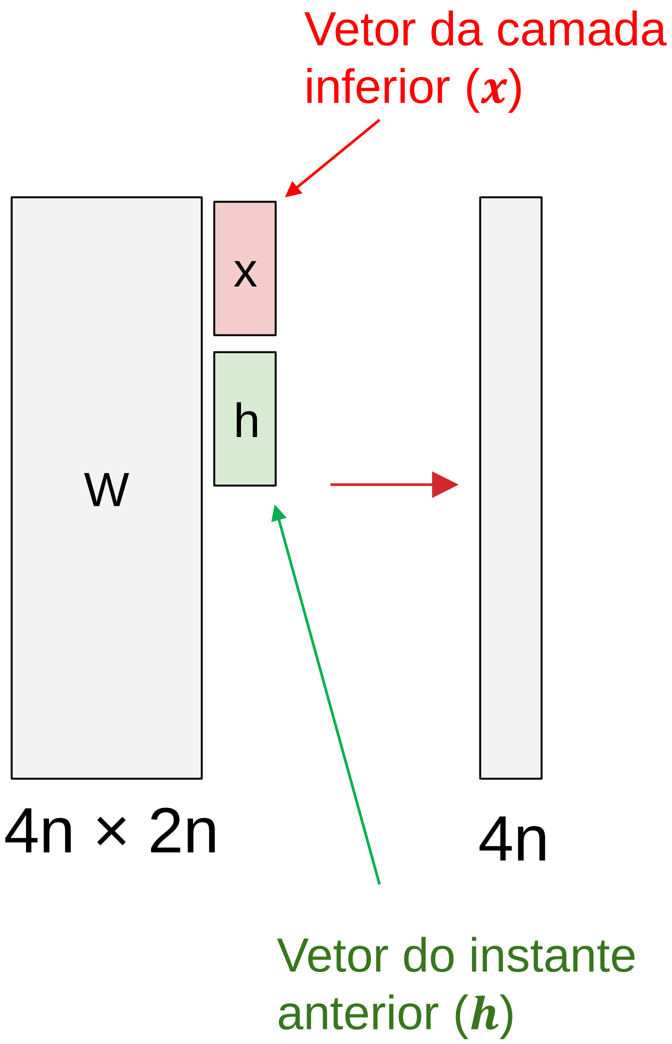
$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \mathrm{sigm} \\ \mathrm{sigm} \\ \mathrm{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



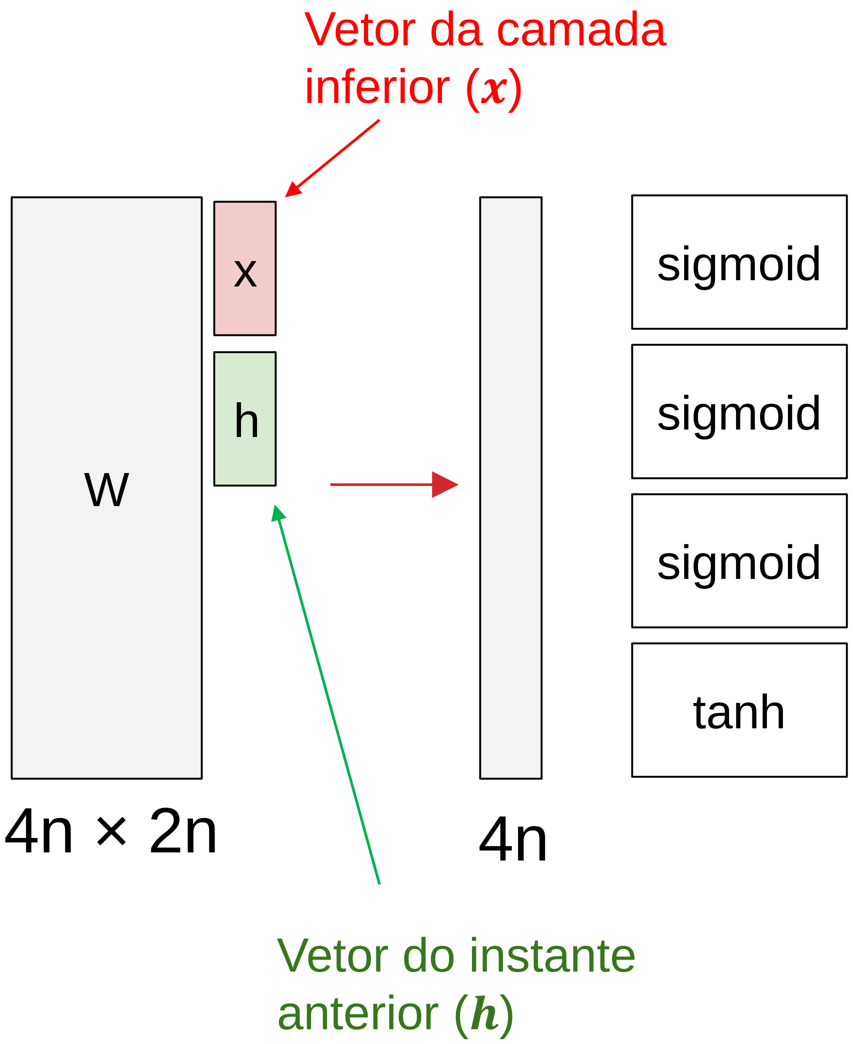
$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



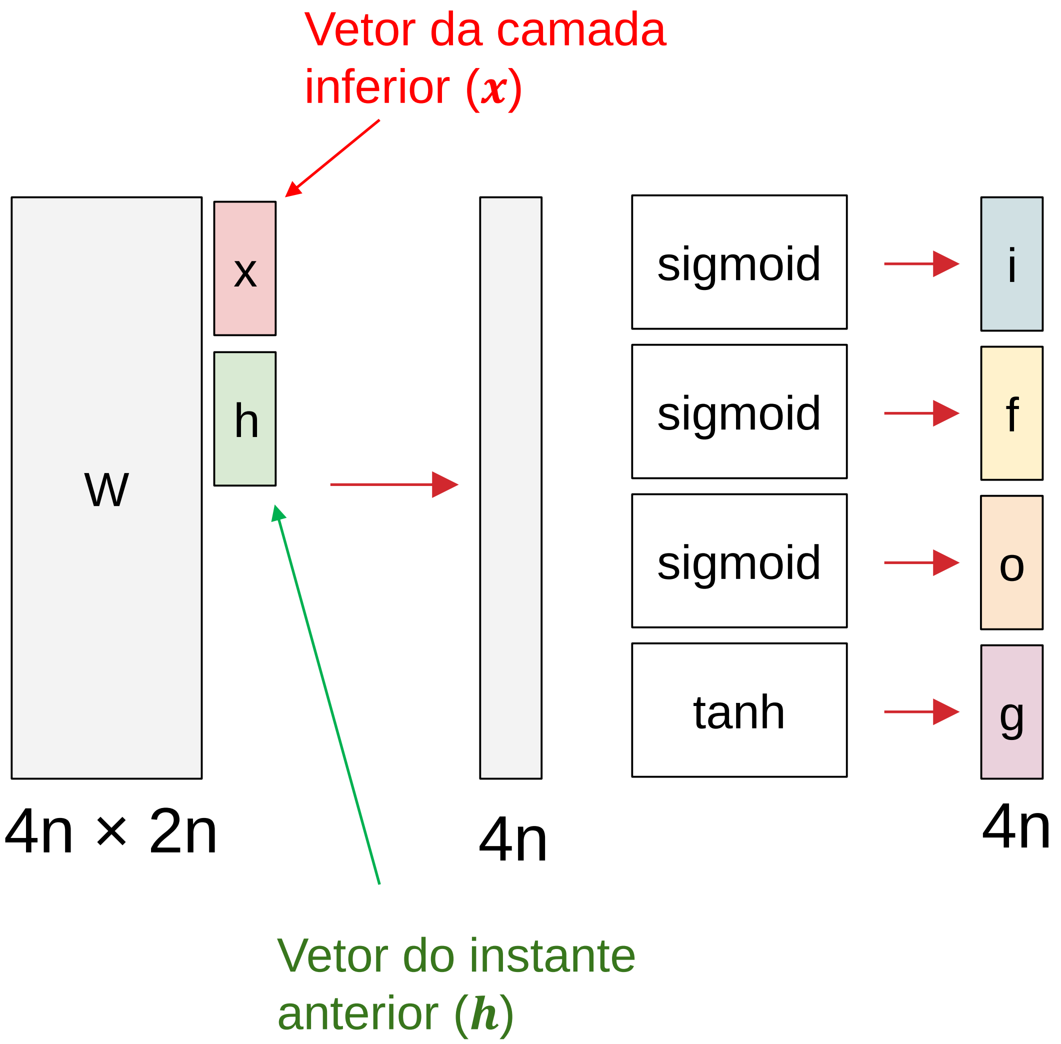
$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \mathrm{sigm} \\ \mathrm{sigm} \\ \mathrm{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = \boxed{f \odot c_{t-1}^l} + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = \boxed{f \odot c_{t-1}^l} + i \odot g$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

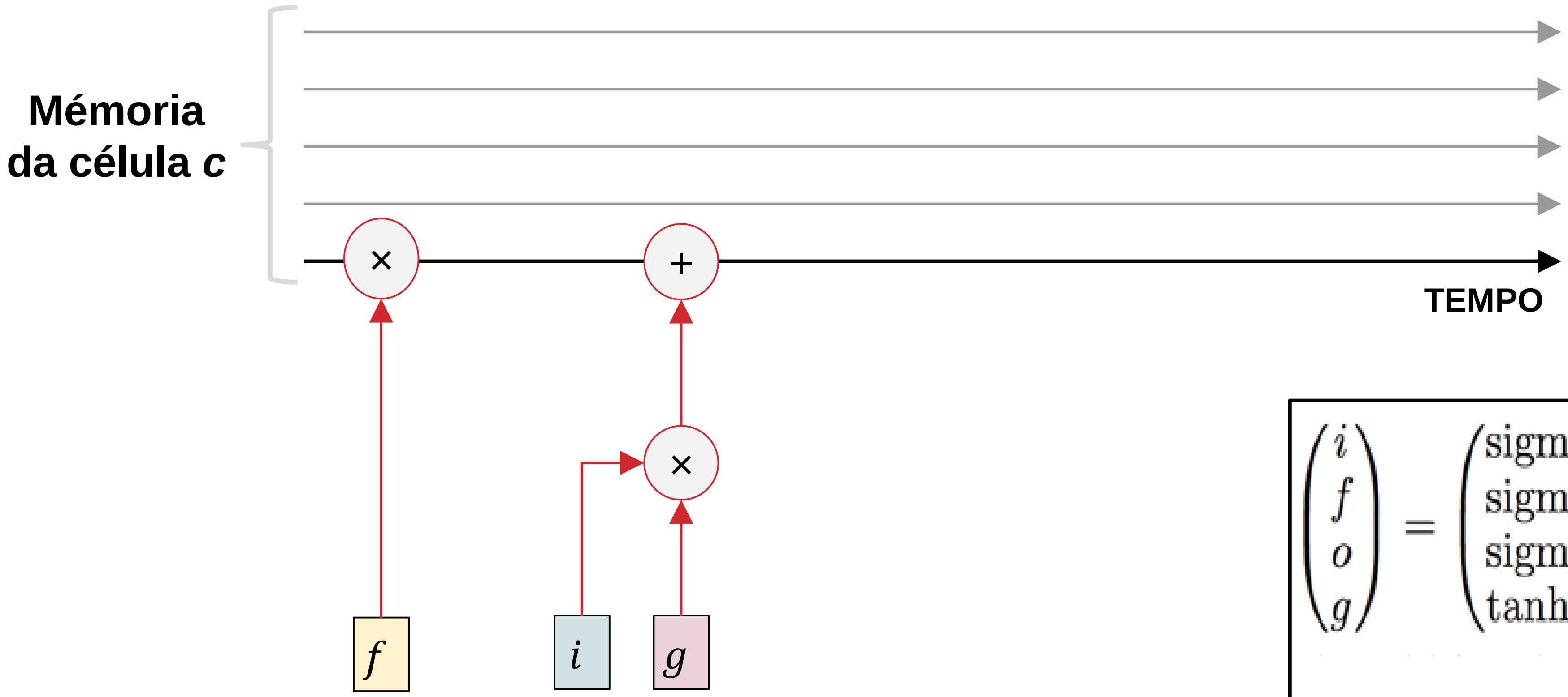
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l \boxed{+ i \odot g}$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



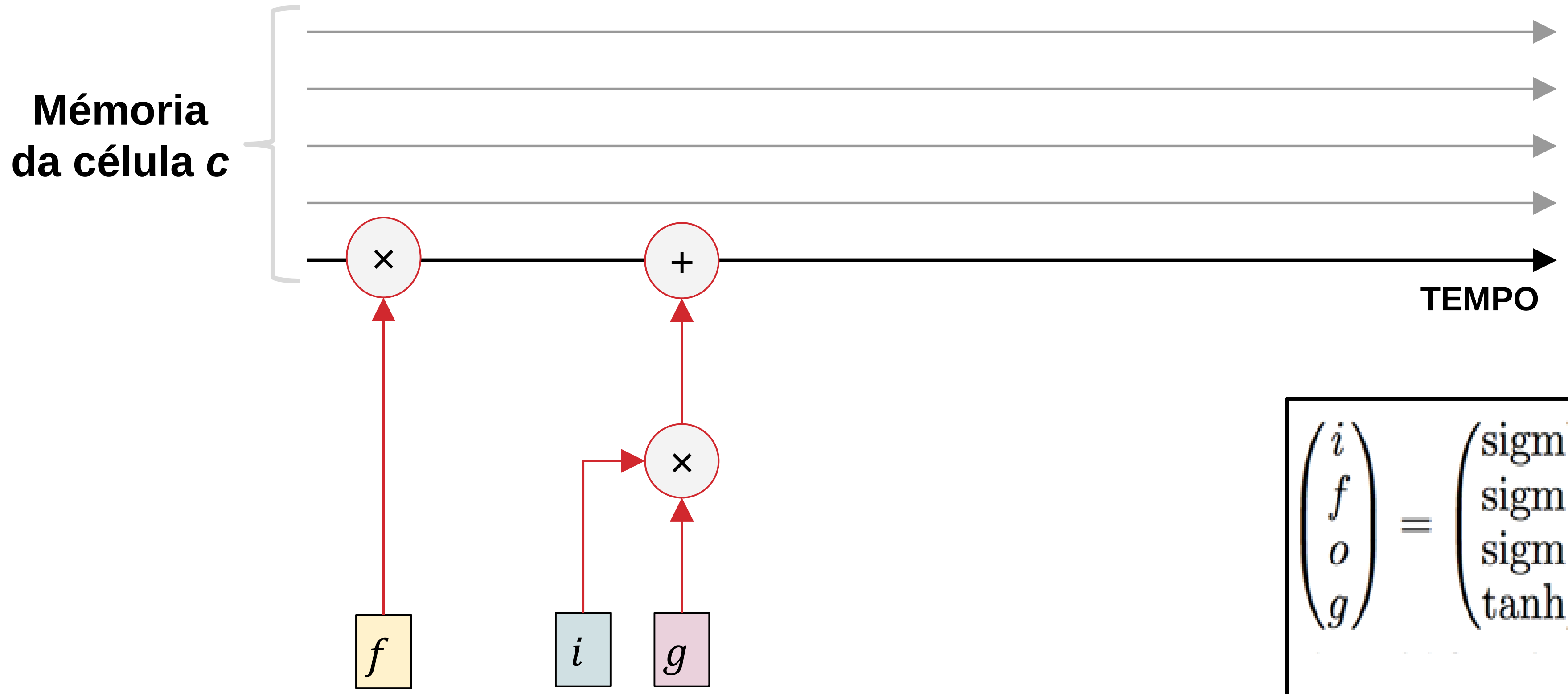
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l \boxed{+ i \odot g}$$

$$h_t^l = o \odot \tanh(c_t^l)$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



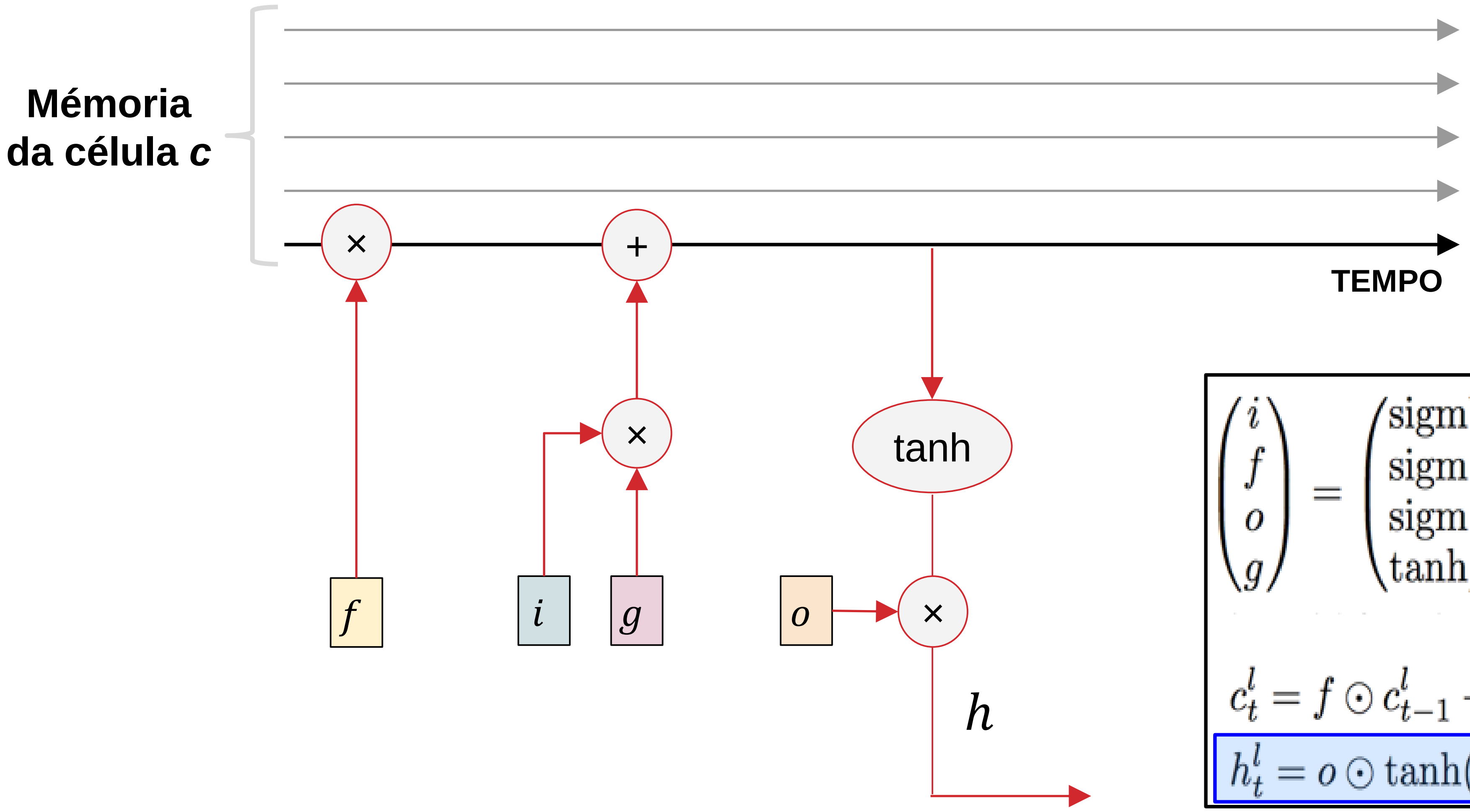
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$\boxed{h_t^l = o \odot \tanh(c_t^l)}$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



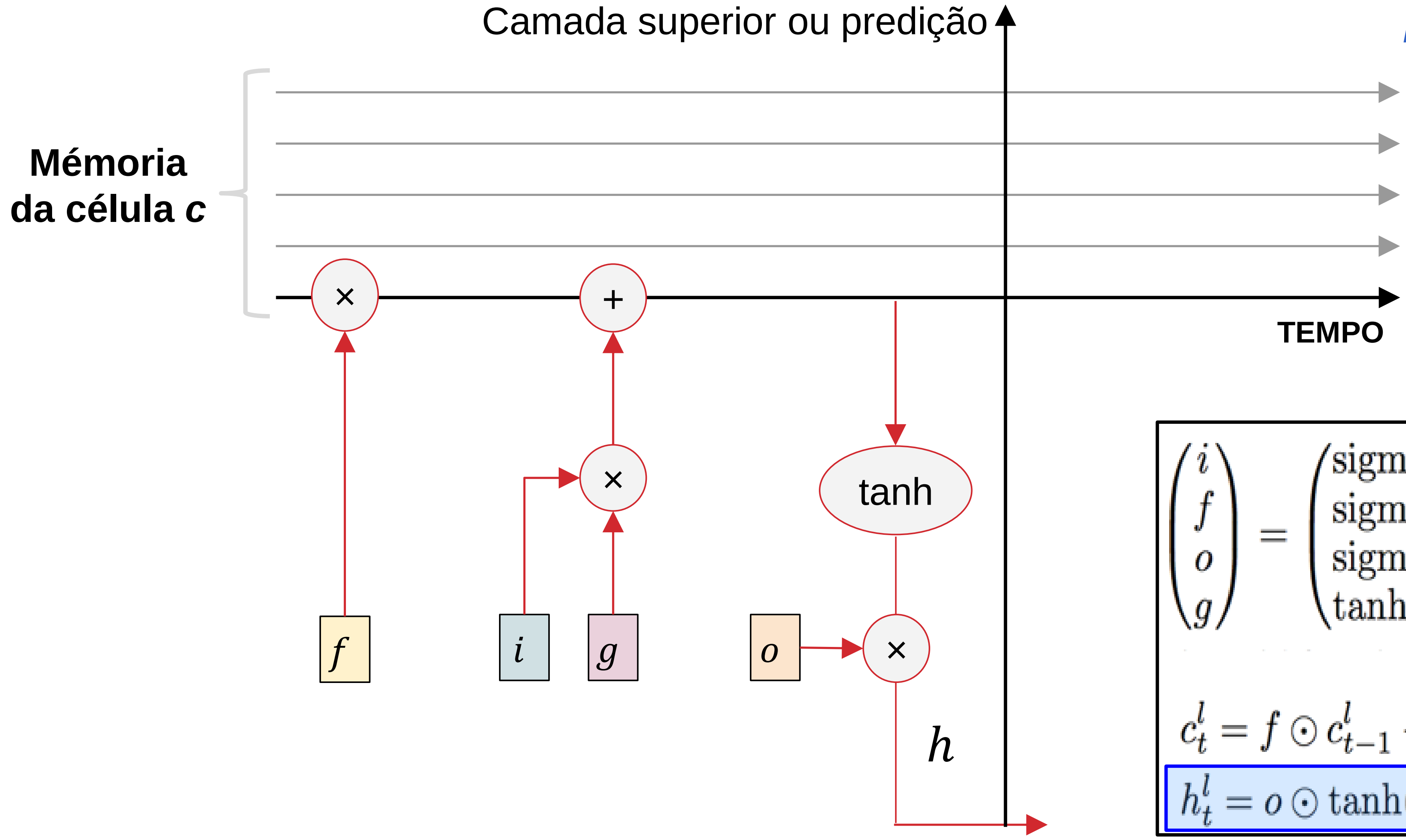
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$\boxed{h_t^l = o \odot \tanh(c_t^l)}$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



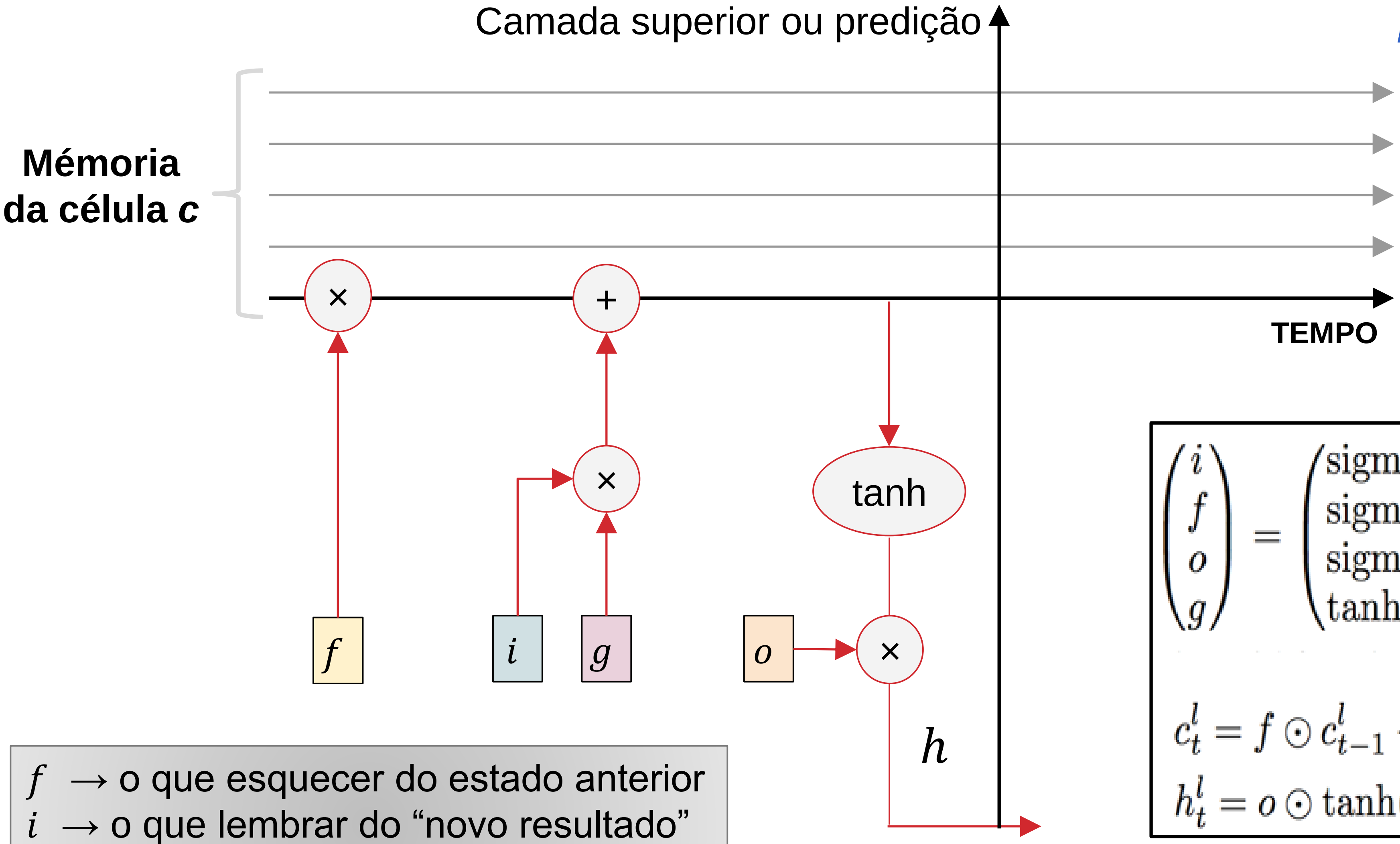
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

$$\boxed{h_t^l = o \odot \tanh(c_t^l)}$$

LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]



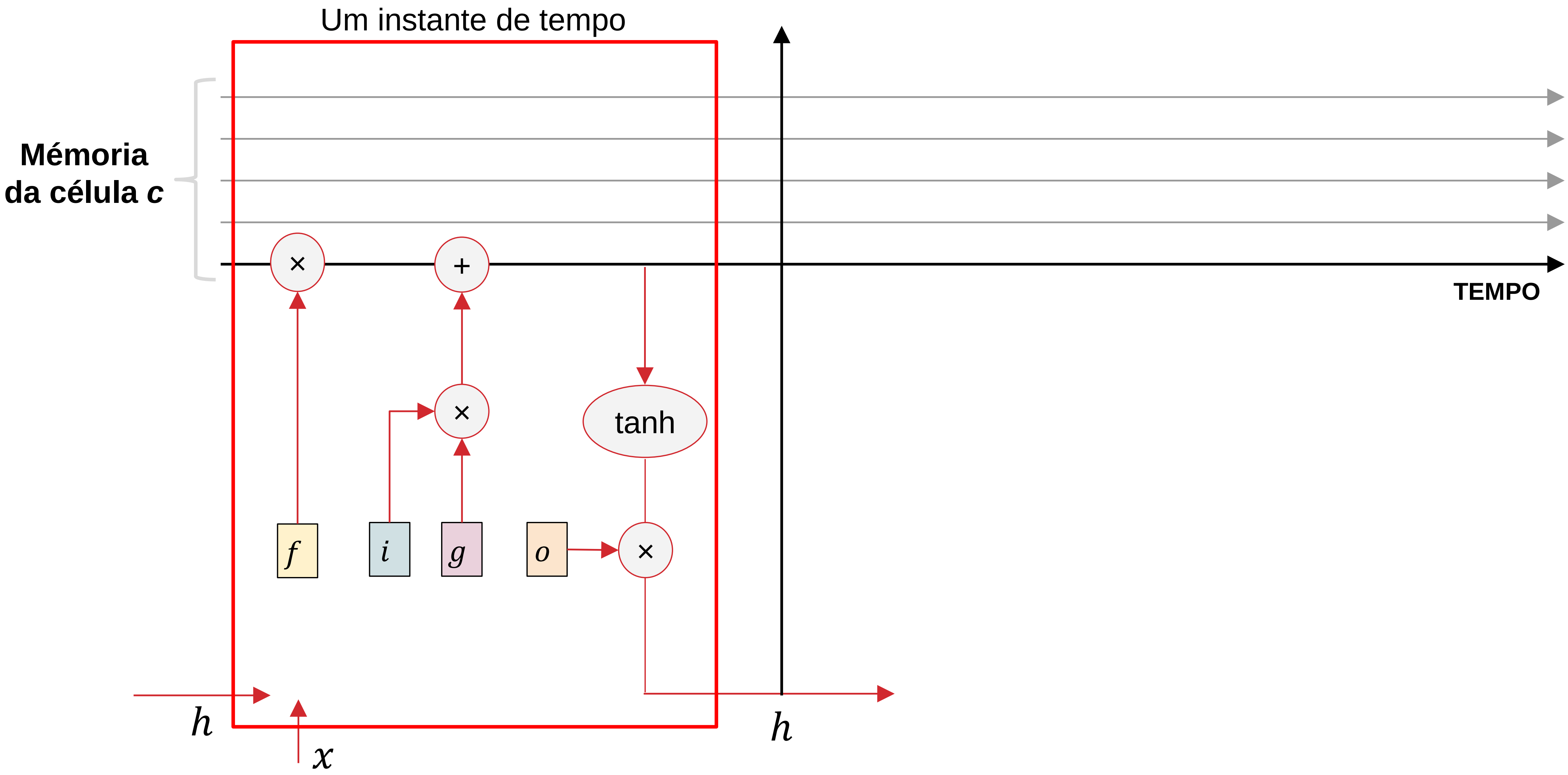
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^l \end{pmatrix}$$

$$c_t^l = f \odot c_{t-1}^l + i \odot g$$

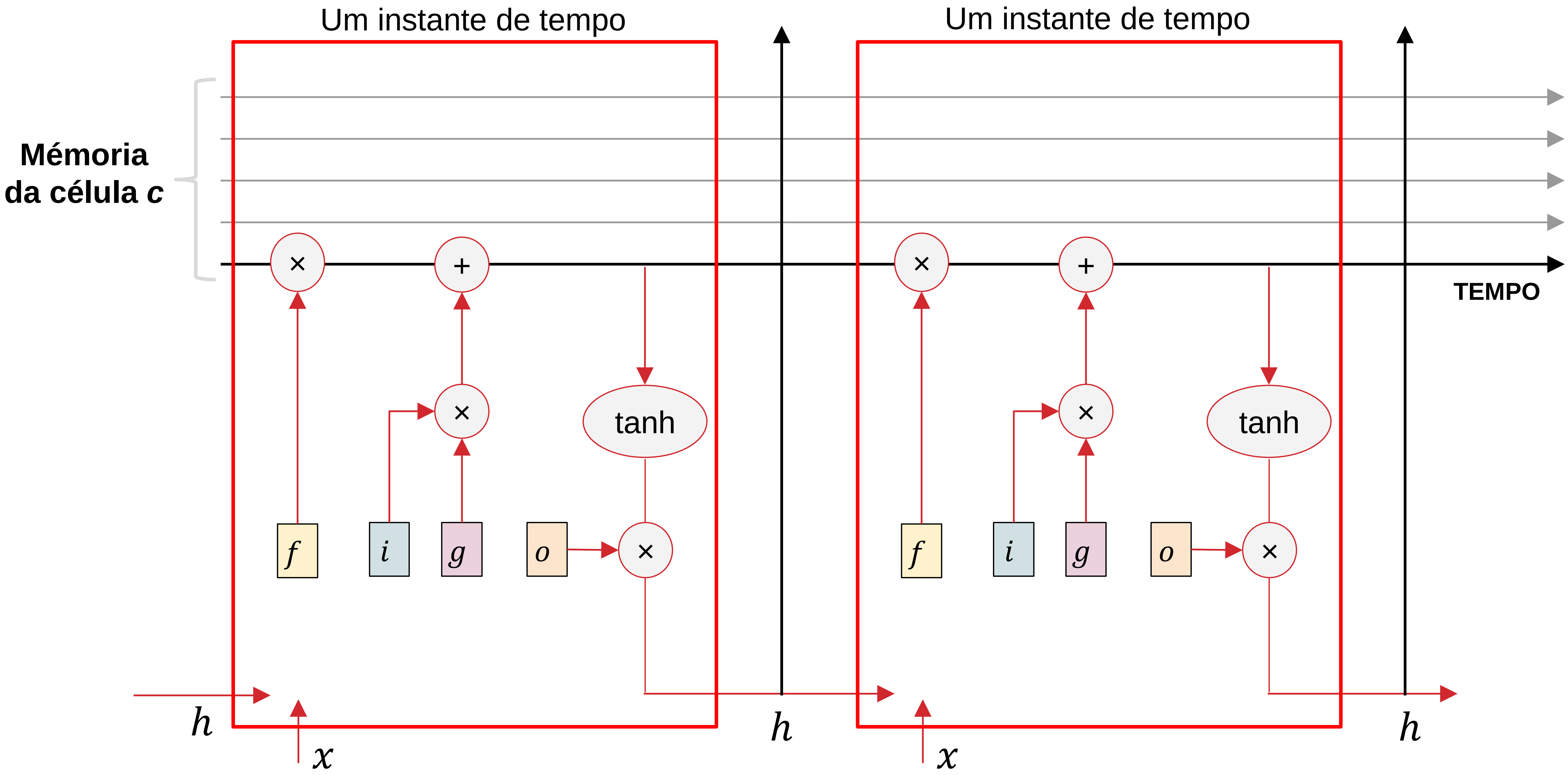
$$h_t^l = o \odot \tanh(c_t^l)$$

f → o que esquecer do estado anterior
i → o que lembrar do "novo resultado"
o → o que produzir como saída

LSTM – *Long Short Term Memory*



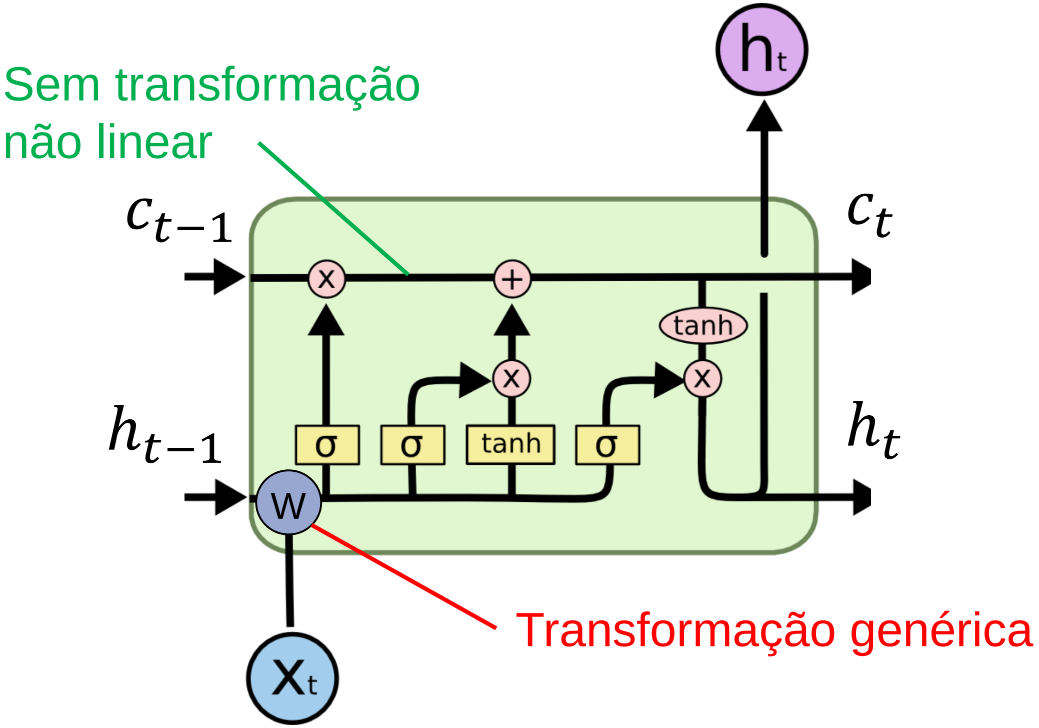
LSTM – *Long Short Term Memory*



LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

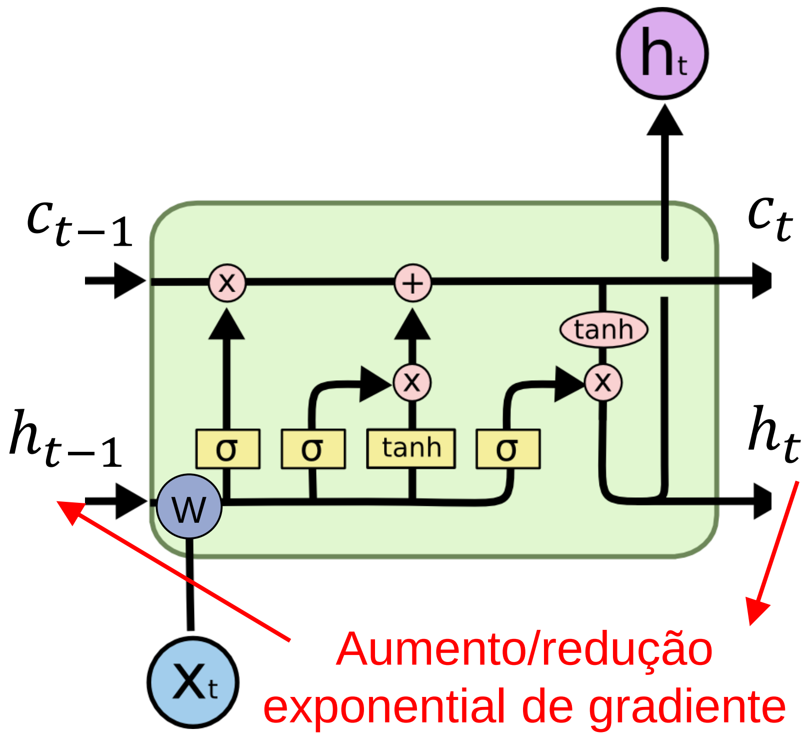
Lembre-se de que h passa por uma transformação que depende de W_{hh} em cada unidade



LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

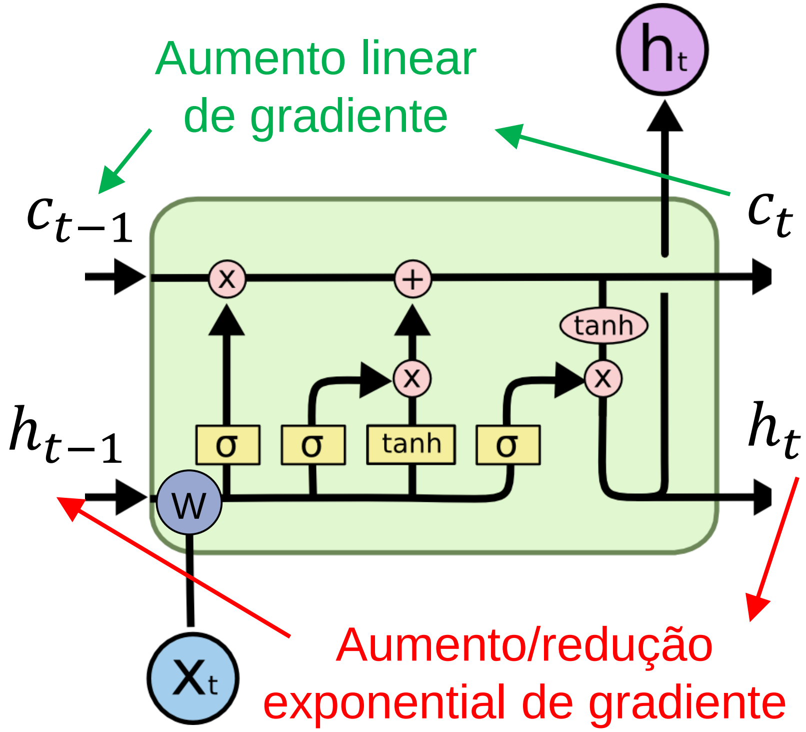
Lembre-se de que h passa por uma transformação que depende de W_{hh} em cada unidade



LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

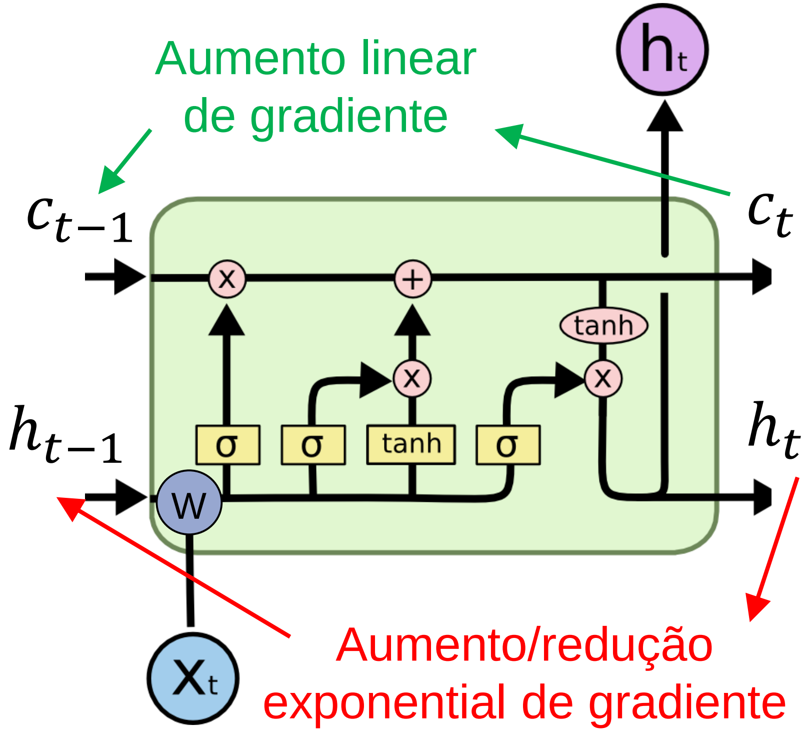
Lembre-se de que h passa por uma transformação que depende de W_{hh} em cada unidade



LSTM – *Long Short Term Memory*

[Hochreiter et al., 1997]

Lembre-se de que h passa por uma transformação que depende de W_{hh} em cada unidade



Gradientes são bem comportados no caminho de c mas não do de h, porém LSTMs aprendem a “confiar” principalmente em c para a memória de longo prazo

LSTM – Variações

GRU – *Gated Recurrent Unit* [Cho et al. 2014]

$$
\begin{aligned}
r_t &= \mathrm{sigm}\left(W_{\mathrm{xr}} x_t + W_{\mathrm{hr}} h_{t-1} + b_{\mathrm{r}}\right) \\
z_t &= \mathrm{sigm}(W_{\mathrm{xz}} x_t + W_{\mathrm{hz}} h_{t-1} + b_{\mathrm{z}}) \\
\tilde{h}_t &= \tanh(W_{\mathrm{xh}} x_t + W_{\mathrm{hh}} (r_t \odot h_{t-1}) + b_{\mathrm{h}}) \\
h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t
\end{aligned}
$$

Veja refs sobre equivalência:

- [*LSTM: A Search Space Odyssey*, Greff et al., 2015]
- [*An Empirical Exploration of Recurrent Network Architectures,* Jozefowicz et al., 2015]