

# MSI603 – Proyecto Integrador: Ciencia de Datos

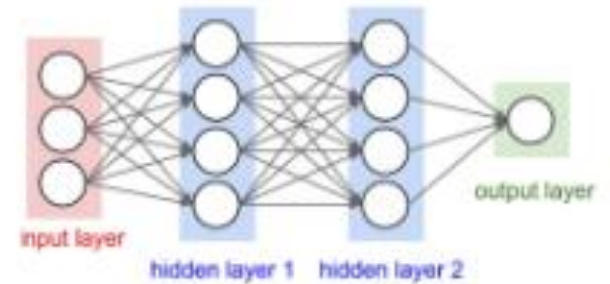
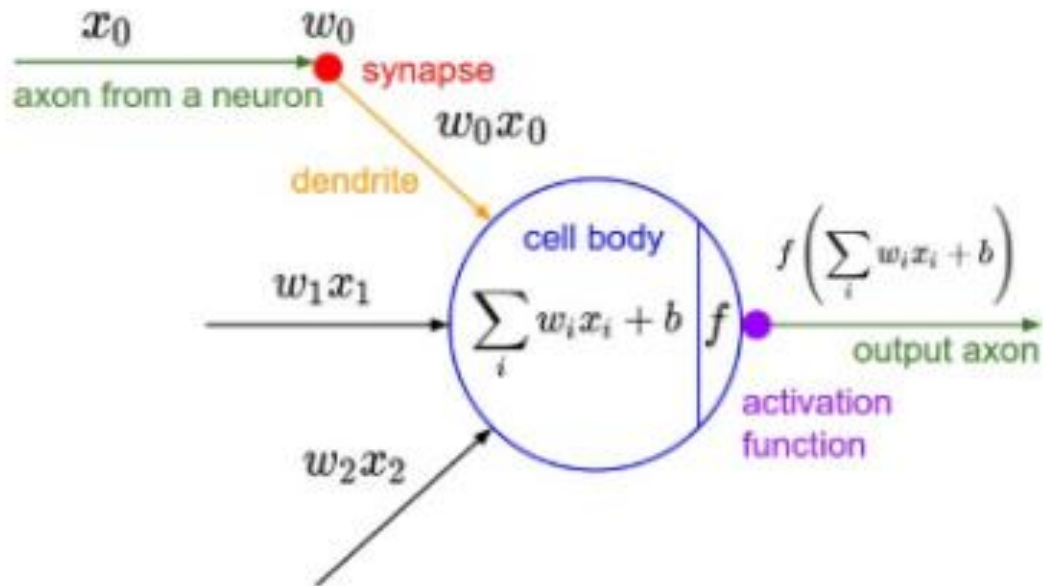


**Universidad  
Andrés Bello®**  
Conectar · Innovar · Liderar

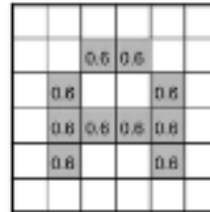


## Mag. en Ingeniería Informática

# Perceptrón



- Convolución: operador matemático que transforma dos funciones  $f$  y  $g$ , en que se superpone  $f$  y una versión invertida y trasladada de  $g$ .
- Las redes neuronales convolucionales (CNN - Convolutional Neural Networks) son útiles para determinar las características importantes a considerar dentro del conjunto de datos.
- En aplicaciones con visión, las primeras capas pueden detectar líneas, curvas hasta llegar a formas más complejas hacia capas más profundas.



		0.5	0.5			
	0.5				0.5	
	0.5	0.5	0.5	0.5		
	0.5				0.5	

- Una imagen es una matriz de píxeles, que cuando hay color presente está formado por una tupla de valores (R,G,B) con cada componente representando un valor de 0 a 255 según la intensidad del color respectivo.
- En una imagen con 28px de alto y 28px de ancho, se usa 784 neuronas en la capa de entrada (a 1 color, en escala de grises).
- Cada valor de intensidad de color en cada píxel se debe normalizar a valores entre 0 y 1.

		0.6	0.6		
	0.6			0.6	
	0.6	0.6	0.6	0.6	
	0.6			0.6	

Imagen de  
entrada

1	0	-1
2	0	-2
1	0	-1

kernel

- La primera operación consiste en aplicar un kernel a grupos de pixeles cercanos generando una nueva matriz de salida.
- El kernel inicialmente toma valores aleatorios y se va ajustando mediante backpropagation.
- El conjunto de kernels se llama *filtro*.



- Aplicación de función de activación:

IMAGEN

		0,6	0,6		
	0,6			0,6	
	0,6	0,6	0,6	0,6	
	0,6			0,6	

KERNEL

1	0	-1
2	0	-2
1	0	-1

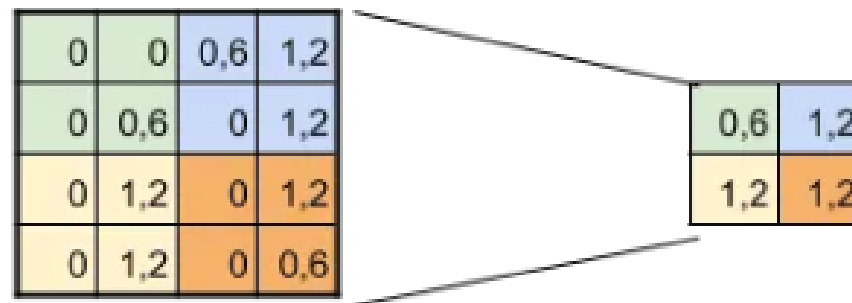
CONVOLUCION  
DEL KERNEL

-1,2	-0,6	0,6	1,2
-1,2	0,6	-0,6	1,2
-1,2	1,2	-1,2	1,2
-0,6	1,2	-1,2	0,6

APLICO RELU

0	0	0,6	1,2
0	0,6	0	1,2
0	1,2	0	1,2
0	1,2	0	0,6

- Técnica para reducir dimensionalidad antes de una nueva convolución.
- Prevalecen las características más importantes que detectó cada filtro.
- Método más popular: Max-Pooling



# Proceso de convolución

1)Entrada: Imagen	2)Aplico Kernel	3)Obtengo Feature Mapping	4)Aplico Max- Pooling	5)Obtengo "Salida" de la Convolución
28x28x1	32 filtros de 3x3	28x28x32	de 2x2	14x14x32

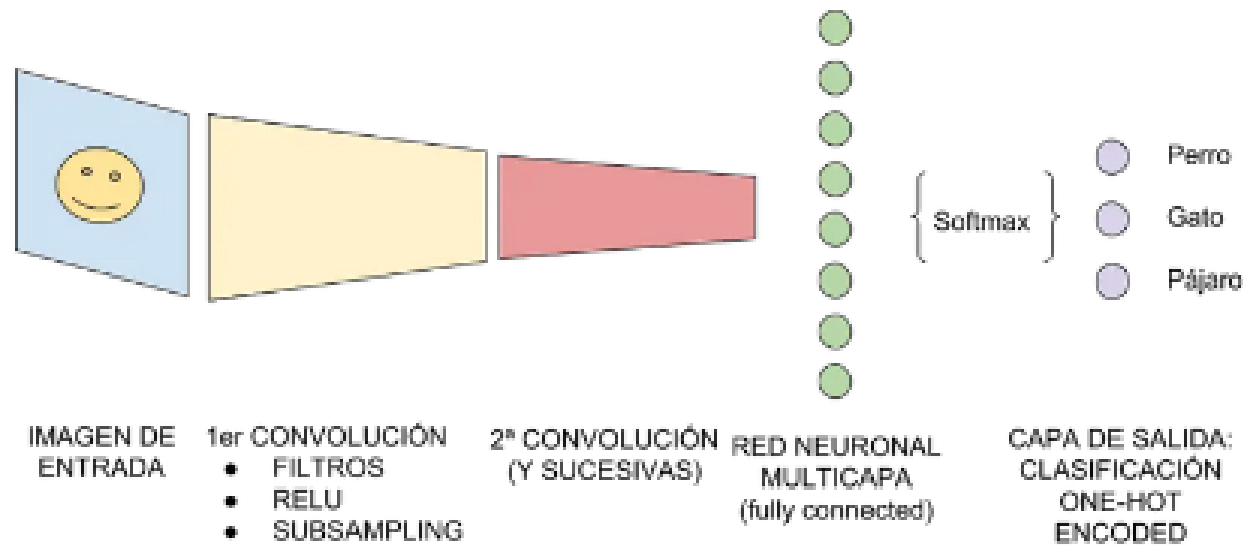


# Convoluciones sucesivas

1)Entrada: Imagen	2)Aplico Kernel	3)Obtengo Feature Mapping	4)Aplico Max- Pooling	5)Obtengo "Salida" de la Convolución
14x14x32	64 filtros de 3x3	14x14x64	de 2x2	7x7x64



1)Entrada: Imagen	2)Aplico Kernel	3)Obtengo Feature Mapping	4)Aplico Max- Pooling	5)Obtengo "Salida" de la Convolución
7x7x64	128 filtros de 3x3	7x7x128	de 2x2	3x3x128

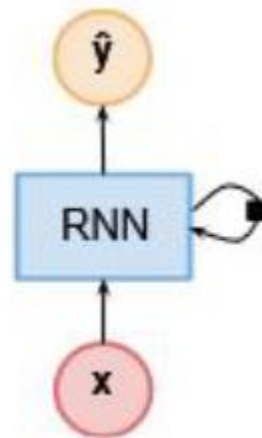


- Se conecta con la última capa mediante la función *softmax* (generalización de función logística), con salida en one-hot encoding.

## Sesión\_06\_CNN.ipynb



- **RNN** adaptadas para procesar secuencias de valores:  $x_1, x_1, \dots, x_t$ .
- Las **redes recurrentes** pueden procesar secuencias de largo arbitrario
- Utilizan la información de la salida de la red en el tiempo anterior o el valor de un estado oculto interno a la red



**Idea central:** Compartir parámetros en distintas partes del modelo (i.e., mismos parámetros de red para distintos instantes de tiempo)

## Ejemplo:

"Me fui a vivir a París en el 2008."

"En el 2008, me fui a vivir a París."

- Entrenamos un sistema de reconocimiento automático, pedimos que prediga el año en el que el narrador se mudó a París
- Deberá indicar "2008" más allá de que si el año aparece en 9na o 3era posición en la frase.
- Tiene que aprender las reglas del lenguaje para cada posible posición de palabra en la sentencia.
- Las **redes neuronales recurrentes** comparten el modelo a lo largo de los instantes de tiempo (similar a red convolucional)



Formula de Recurrencia:

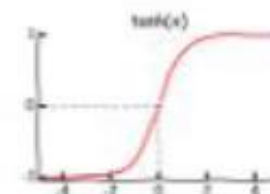
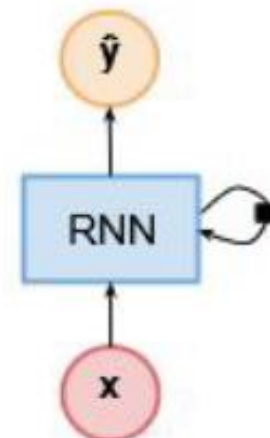
$$\mathbf{h}_t = f_{\mathbf{W}}(\mathbf{h}_{t-1}, \mathbf{x}_t)$$

Ejemplo básico

$$\mathbf{h}_t = \tanh(\mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{W}_{xh}\mathbf{x}_t)$$

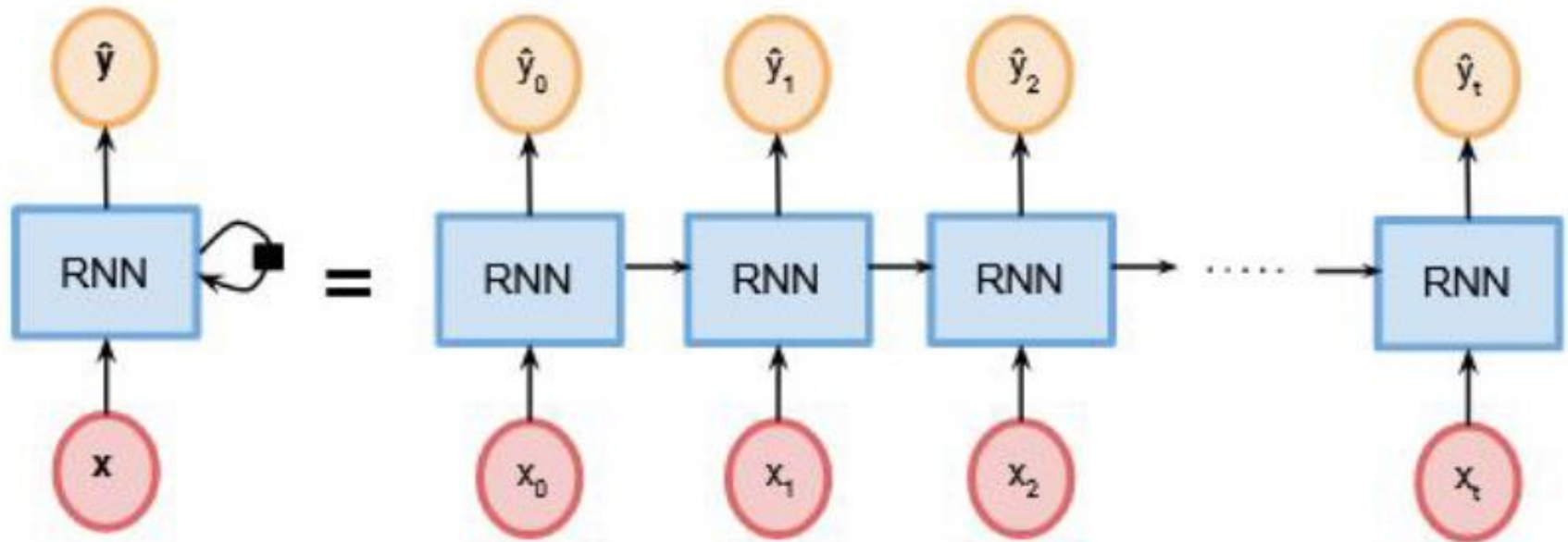
$$\hat{\mathbf{y}}_t = \mathbf{W}_{hy}\mathbf{h}_t$$

- Estado consiste en un único vector (oculto)  $\mathbf{h}_t$
- Salida en el instante  $\hat{\mathbf{y}}_t$  es una operación lineal del estado actual



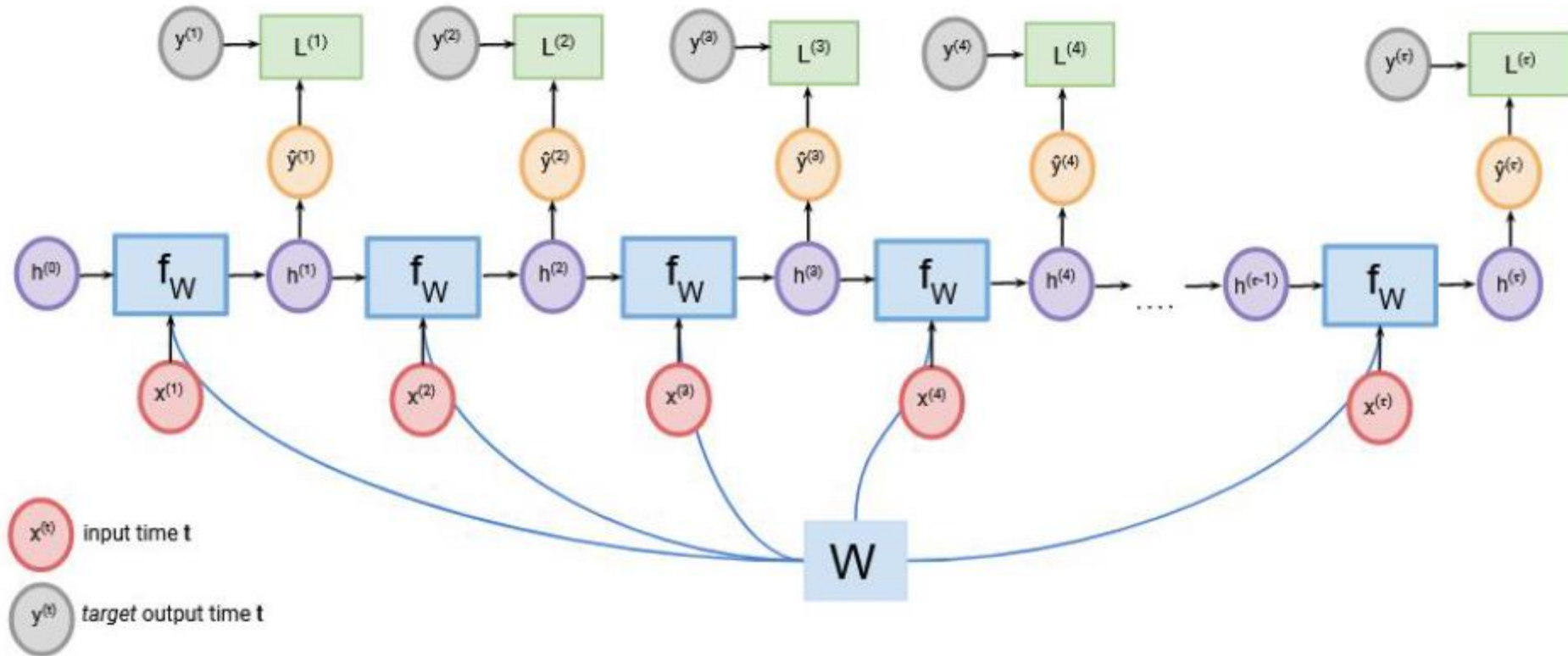


# Redes neuronales recurrentes



Una red neuronal recurrente puede pensarse como múltiples copias de la misma red donde cada una le pasa un mensaje (estado) al sucesor.

# Redes neuronales recurrentes



- Los pesos  $W$  se comparten en todos los instantes de tiempo.

$$\begin{aligned}\mathbf{h}_t &= f(\mathbf{h}_{t-1}, \mathbf{x}_t) \\ &= f(f(\mathbf{h}_{t-2}, \mathbf{x}_{t-1}), \mathbf{x}_t) \\ &= f(f(f(\mathbf{h}_{t-3}, \mathbf{x}_{t-2}), \mathbf{x}_{t-1}), \mathbf{x}_t) \\ &\vdots \\ &= f(\dots f(f(\mathbf{h}_1, \mathbf{x}_2), \mathbf{x}_3), \dots, \mathbf{x}_t) \\ &= \tilde{f}(\mathbf{h}_0, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t)\end{aligned}$$

- Estado  $\mathbf{h}_t$  en el instante  $t$  depende del estado inicial  $\mathbf{h}_0$  y todas las entradas anteriores hasta  $t$

La discrepancia total (total loss) para una secuencia es:

$$\begin{aligned} L &= \sum_{t=1}^{\tau} L_t(\hat{\mathbf{y}}_t, \mathbf{y}_t) \\ &= \sum_{t=1}^{\tau} L_t(g(\mathbf{h}_0, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{t-1}, \mathbf{x}_t), \mathbf{y}_t) \end{aligned}$$

## Backpropagation through time (BPTT)

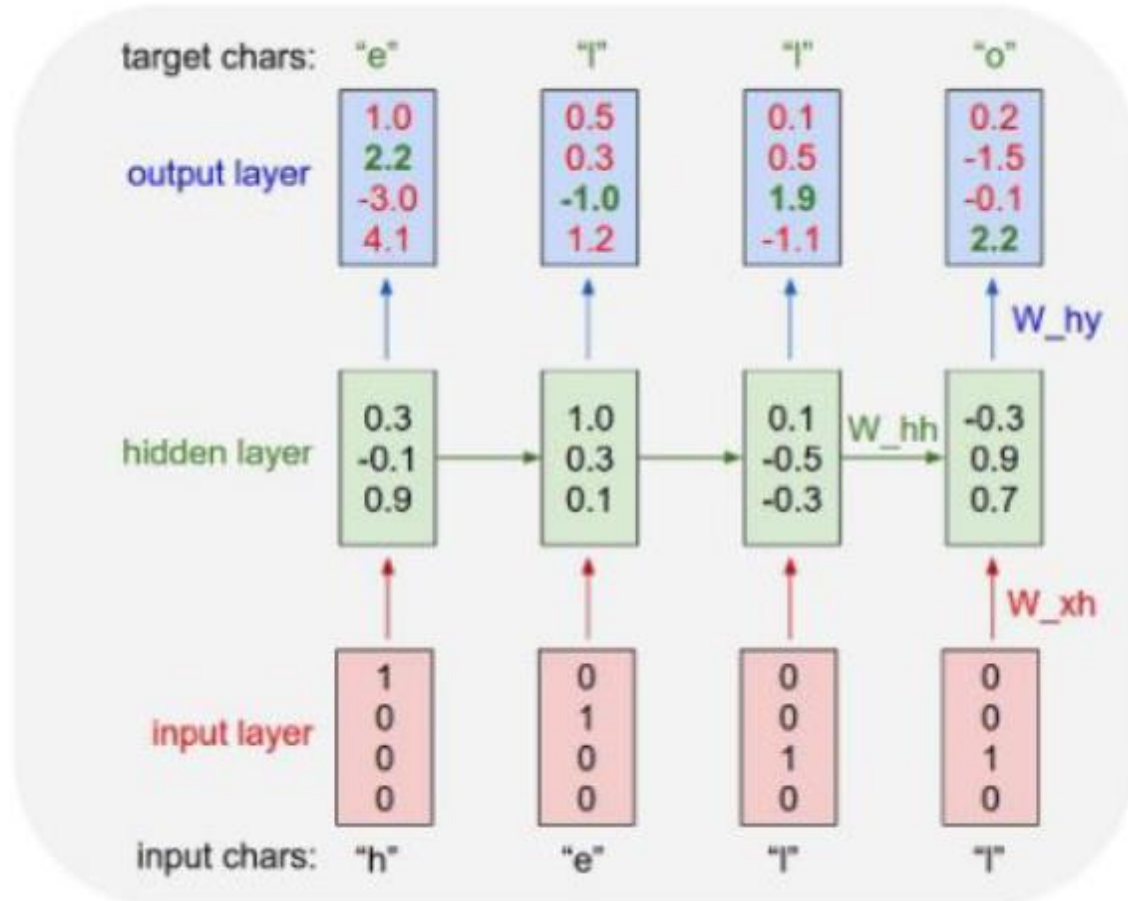
- El cálculo del gradiente de  $L$  respecto a los parámetros  $\mathbf{W}$  es costoso.
- *Forward*: Requiere  $\tau$  pasos (que no se pueden paralelizar)
- *Backward*: Requiere  $\tau$  pasos (desde  $t = \tau$  hasta  $t = 1$ )



# Ejemplo conceptual

Vocabulary:  
[h,e,l,o]

Example training  
sequence:  
“hello”



# Ejemplo conceptual

tyntd-iafhatawiaoihrdemot lytdws e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e  
plia tklrqd t o idoe ns,smtt h ne etie h,hregtrs nigtike,aoaenns lng

↓ train more

"Tmont thithey" fomesscerliund  
Keushey. Thom here  
sheulke, anmerenith ol sivh I lalterthend Bleipile shuw y fil on aseterlome  
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."

↓ train more

Aftair fall unsuch that the hall for Prince Velzonski's that me of  
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort  
how, and Gogition is so overelical and offer.

↓ train more

"Why do what that day," replied Natasha, and wishing to himself the fact the  
princess, Princess Mary was easier, fed in had oftened him.  
Pierre aking his soul came to the packs and drove up his father-in-law women.



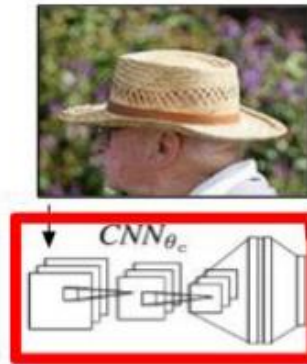
# Ejemplo conceptual (CNN+RNN)



*A cat is sitting on a tree branch*

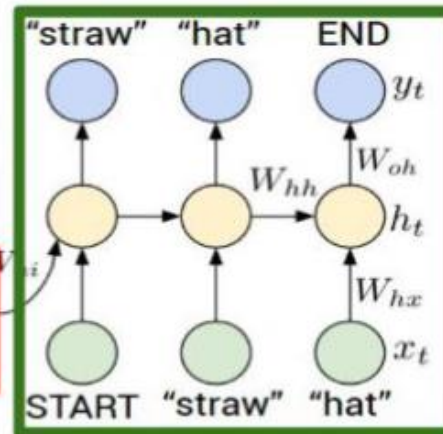


*A tennis player in action on the court*



## Convolutional Neural Network

## Recurrent Neural Network



*A woman standing on a beach holding a surfboard*



*A man in a baseball uniform throwing a ball*

## Sesión\_06\_RNN.ipynb

