Sesión 4:

Redes Neuronales Recurrentes y LSTM Layers para Reconocimiento de Voz

---

Elvin Mark Munoz Vega

Table of contents

Objetivos

Objetivos

1. Entender que son las redes neuronales recurrentes. Aprender sobre los Long-Short Term Memory Layers.
2. Aprender los básico sobre preprocesamiento de audio: Transformadas de Fourier y Espectrogramas.
3. Aprender sobre diferentes tipos de arquitectura usados para speech recognition.

Redes Recurrentes

Porque usar redes recurrentes?

- En ocasiones se necesita que nuestra red “retenga” información (que tenga memoria) para producir una salida precisa.
- Ejemplo: Las palabras dentro de una oración, o segementos de audio dentro del audio general.
- Queremos enseñarle a la red neuronal que “recordar” y que “olvidar” cuando le vamos alimentado con una secuencia de entradas.

Red Recurrente Simple

La forma matemática de este tipo de redes se podria resumir en la siguiente ecuación:

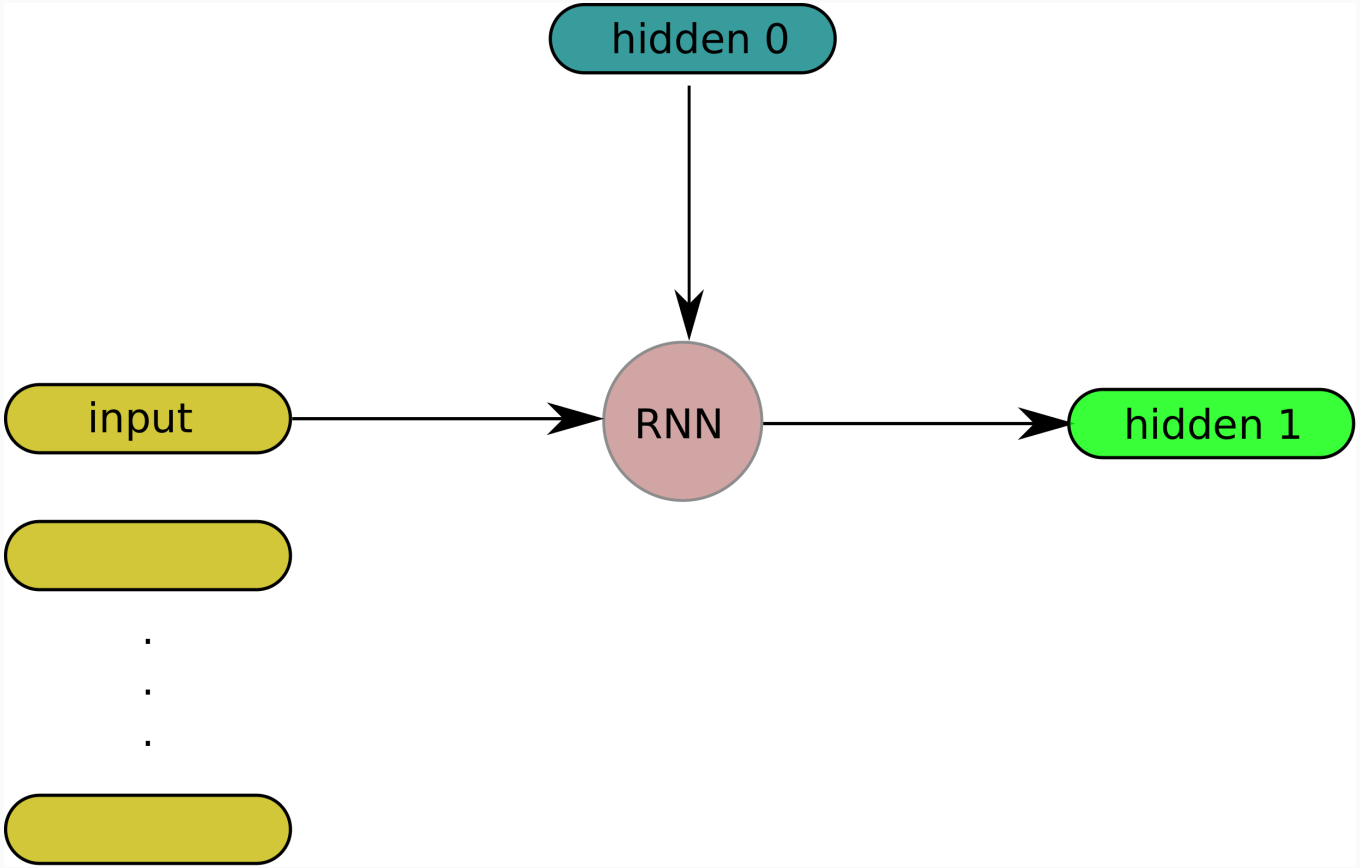
$$h_t = tanh(W_{ih}x_t + b_{ih} + W_{hh}h_{t-1} + b_{hh})$$

Red Recurrente Simple

La forma matemática de este tipo de redes se podria resumir en la siguiente ecuación:

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

y se puede interpretar de la siguiente manera gráficamente:

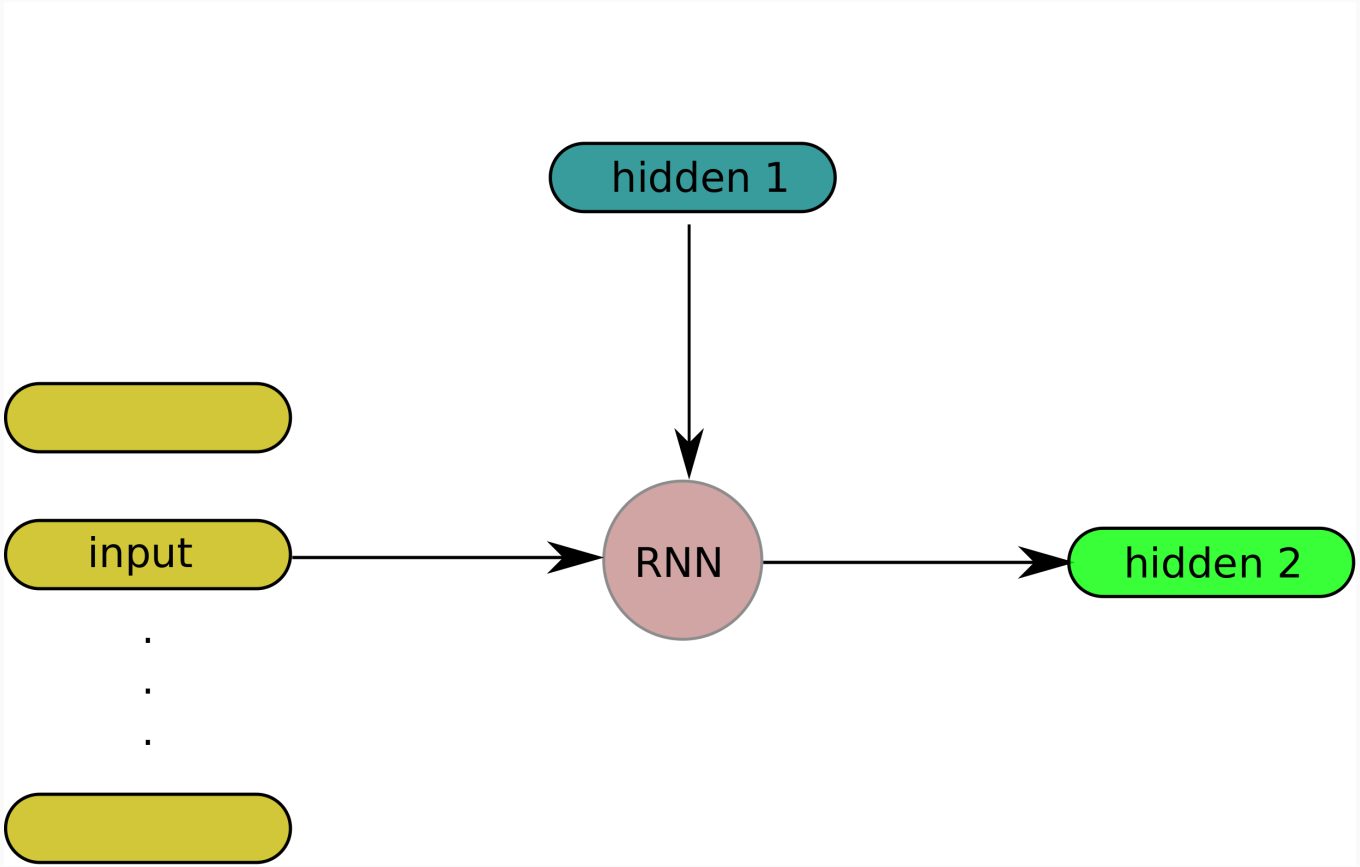


Red Recurrente Simple

La forma matemática de este tipo de redes se podria resumir en la siguiente ecuación:

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

y se puede interpretar de la siguiente manera gráficamente:



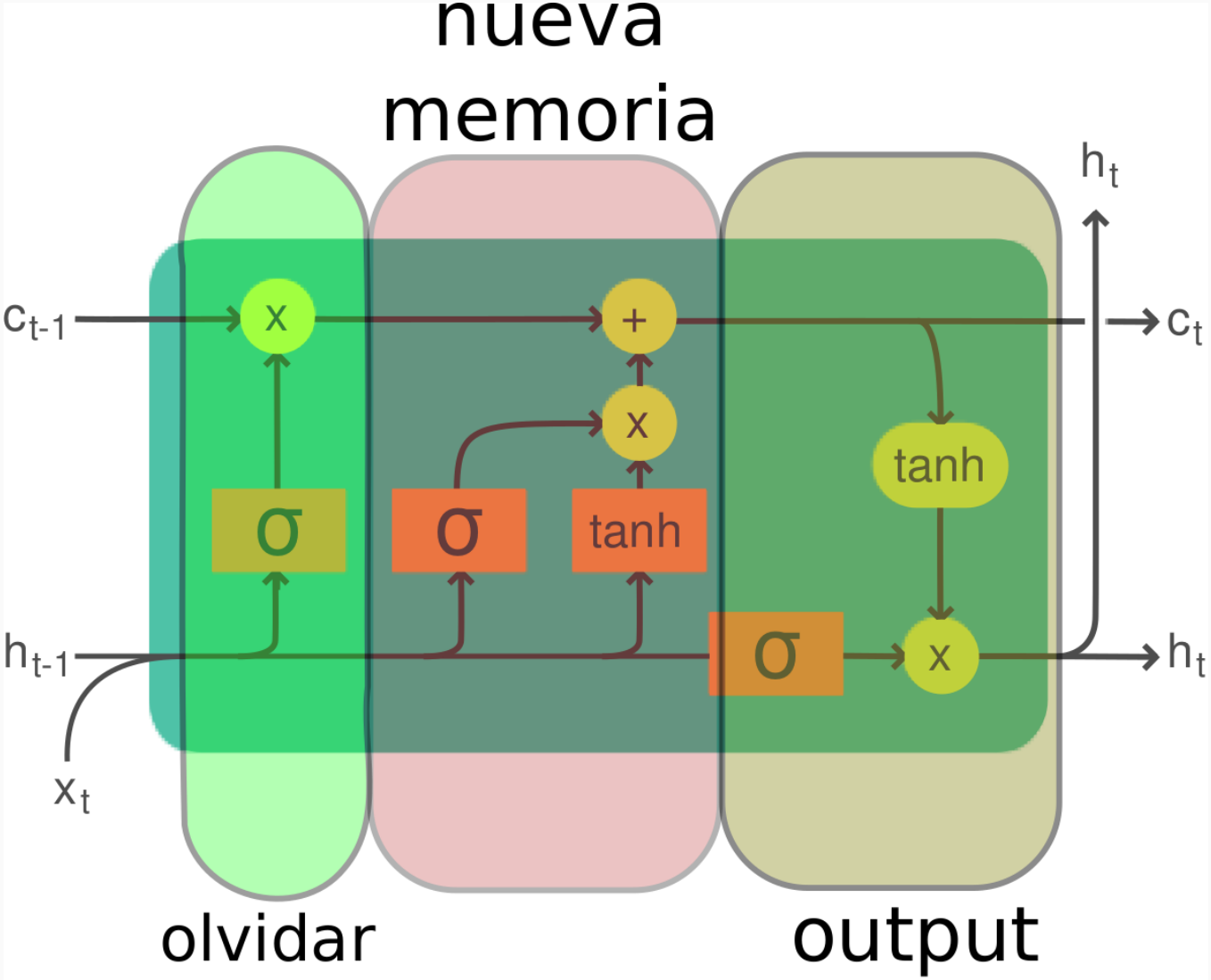
LSTM: Long-Short Term Memory

La formulación matemática de este tipo de redes es como se muestra acontinuación.

$$
\begin{aligned}
i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\
f_t &= \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \\
g_t &= tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \\
o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\
c_t &= f_t * c_{t-1} + i_t * g_t \\
h_t &= o_t * tanh(c_t)
\end{aligned}
$$

LSTM: Long-Short Term Memory

Podemos dividr a la red LSTM en 3 etapas principales.

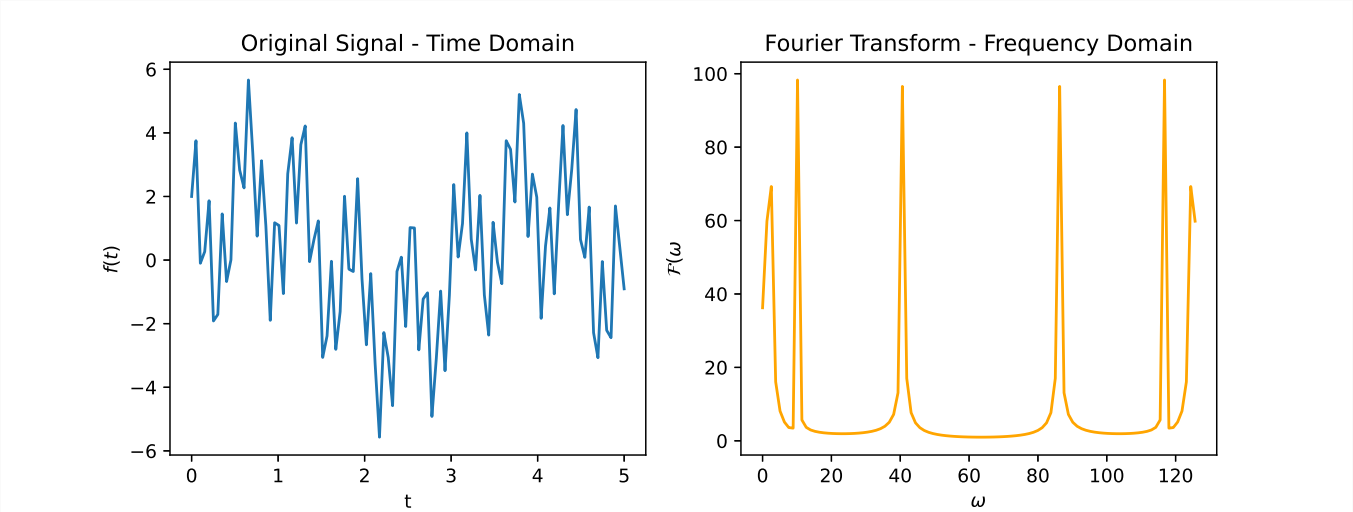


Procesamiento de audio

Transformada de Fourier

Time Domain $\Longrightarrow$ Frequency Domain.

$$\mathcal{F}(\omega) = \int f(t) e^{-j\omega t} dt$$



DFT: Discrete Fourier Transform y FFT: Fast Fourier Transform

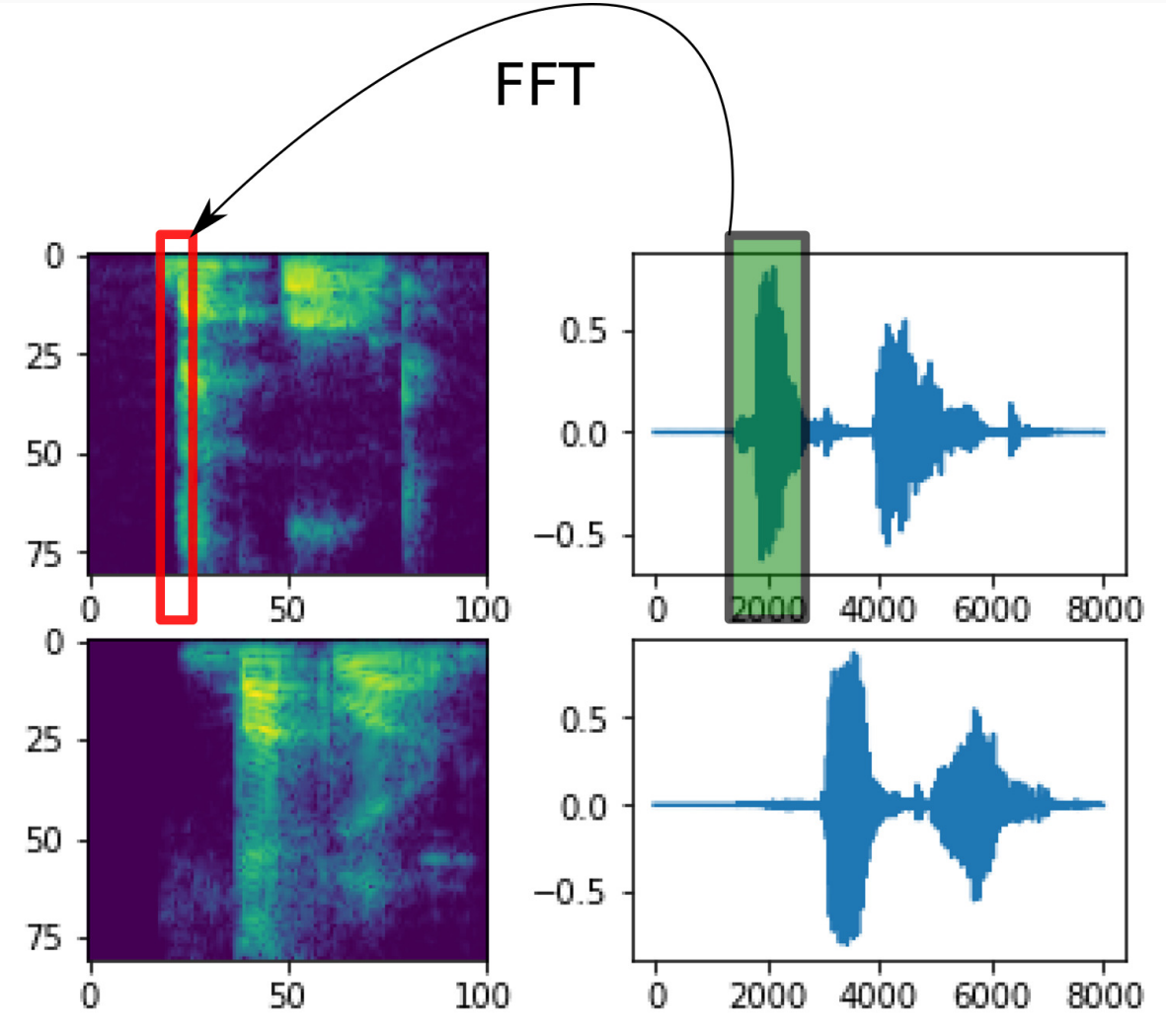
Frecuencia de sampleo: $f_0 \leftarrow$ limita la frecuencia maxima que se puede detectar.

$$\mathcal{F}_k = \sum_{i=1}^{N} x_i e^{-j\frac{2\pi ik}{N}}$$

El FFT es solo un método que agiliza el cálculo de el DFT, haciendo uso de la periodicidad del termino $e^{-j\frac{2\pi ik}{N}}$ en la sumatoria.

Espectrograma

Aplicamos la transformada rapida de Fourier (FFT) a pequeñas ventas de tiempo de la señal original.



Speech Recognition

Usando Conv1d, BatchNorm1d, MaxPool1d

Solo hay que hacer pequeños cambios a las redes convolucionales que hemos venido usando.

[N, C, H, W] $\rightarrow$ [N, C, L]

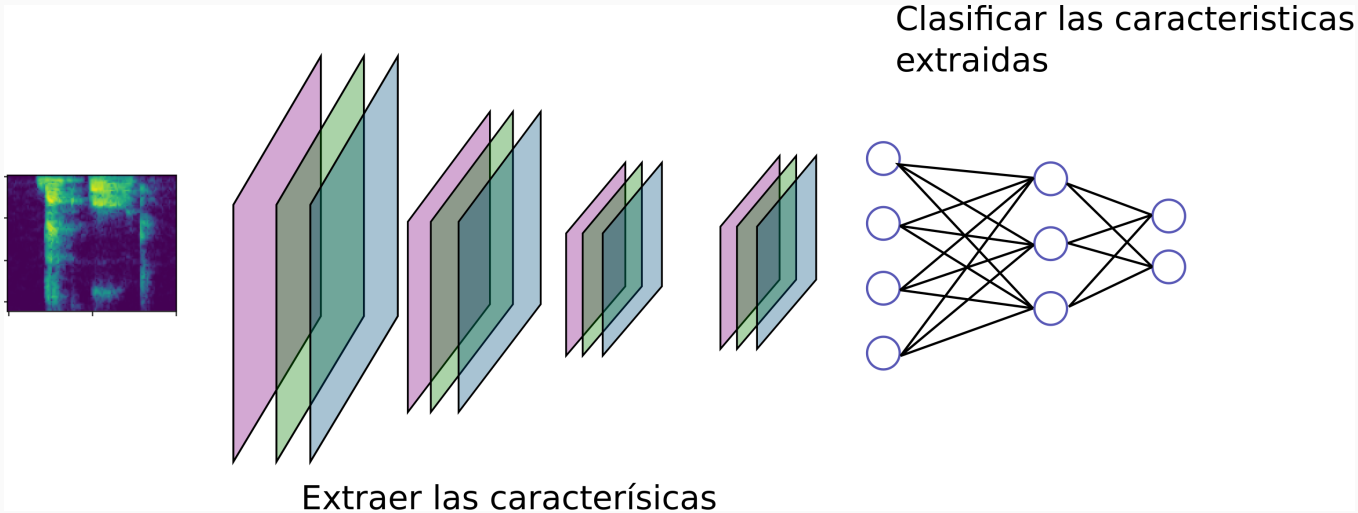
Conv2d $\rightarrow$ Conv1d

BatchNorm2d $\rightarrow$ BatchNorm1d

MaxPool2d $\rightarrow$ MaxPool1d

Usando Espectrogramas y Conv2d

Este método simplemente consiste en transformar la señal de audio en un espectrograma para luego poder usar las redes convolucionales 2D que ya sabemos usar.



Usando Espectrogramas y LSTM

En este método nuevamente utilizamos el espectrograma del audio para que luego una red LSTM pueda extraer las caracterisiticas importantes de la sequencia proporcionada en el espectrograma.