

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

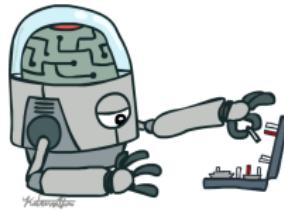
Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers



# Aprendizaje Secuencial

A. Atutxa

LSI Bilbao

November 25, 2024

---

<sup>1</sup>Basado en las lecciones de Andrew Ng (Stanford), Ava Amini (MIT), Jay Alammal

# Overview

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

## 1 Motivación

## 2 Ejemplos de Contextos Secuenciales

## 3 Entendiendo las RNNs

## 4 Cálculo del Forward: ejemplo

## 5 Backprop: problemas

## 6 RNN: Limitaciones

## 7 Razonando hacia los Transformers

# Motivación

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

- Estados dependientes del tiempo

- $E_t = f(x_1, x_2, \dots, x_n, E_{t-1})$

- Predicciones dependientes del estado anterior

# Motivación

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

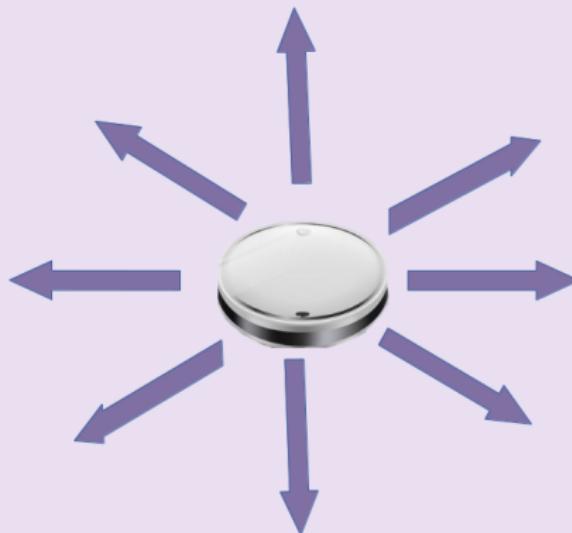
Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

- Predice dónde estará la aspiradora



# Motivación

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

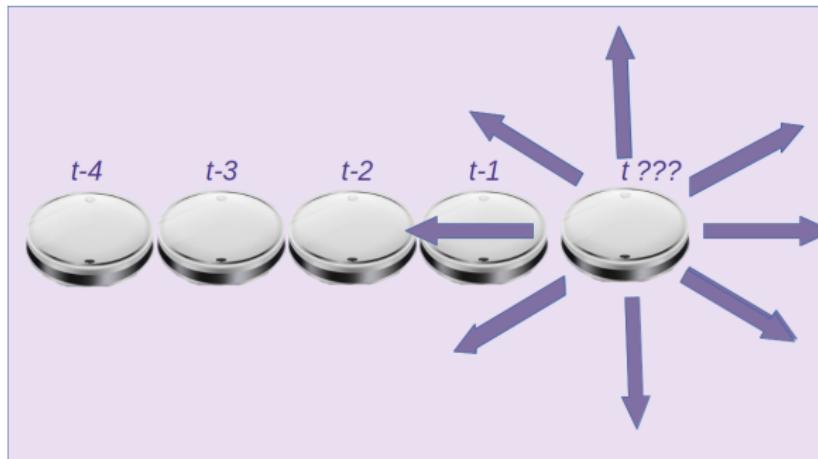
Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

- ¿Y ahora?

Es más fácil porque el problema está más acotado.  
Conocemos el pasado



# Ejemplos de Contextos Secuenciales

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

## ■ Lenguaje:

- Secuencia de caracteres: o-t-i-t-i-s-[ ]-c-r-o-n-i-c-a
- Secuencia de tokens : ot-it-is-[ ]-croni-ca (~ Ngram)
- Secuencia de palabras: otitis-[ ]-cronica

## ■ Predicción valores bolsa



## ■ Biología: Secuencias de ADN



## ■ Medicina: ECGs



# Ejemplos de Tareas

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

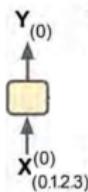
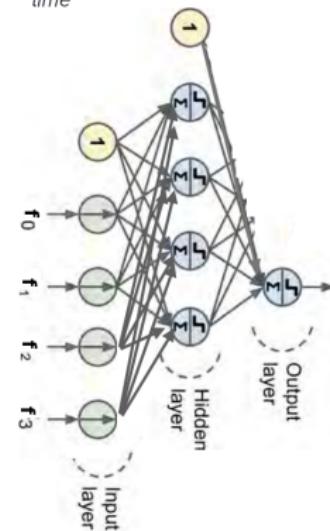
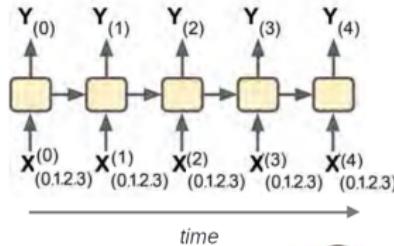
Entendiendo las RNNs

Cálculo del Forward: ejemplo

Backprop: problemas

RNN: Limitaciones

Razonando hacia los Transformers



# Ejemplos de Tareas

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

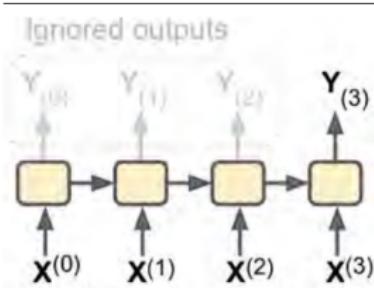
Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

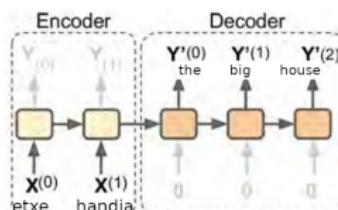
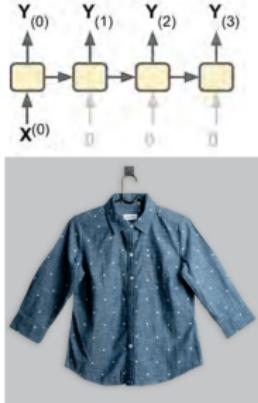
Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers



- Q:** Has the UK been hit by a hurricane?
- P:** The Great Storm of 1987 was a violent extratropical cyclone which caused casualties in England, France and the Channel Islands ...
- A:** Yes. [An example event is given.]



Traducción Automática: etxe handia (eusko)  
The big house (ing)

# Volvamos a nuestra red

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

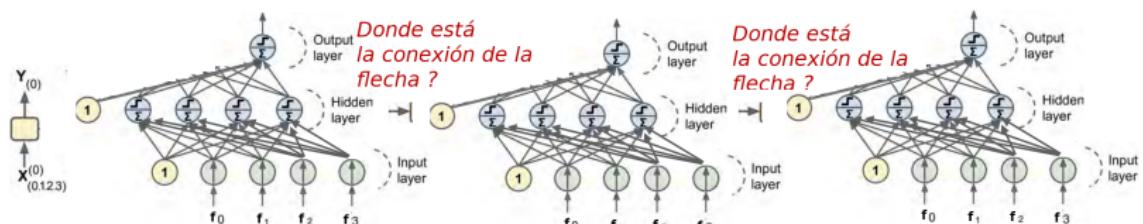
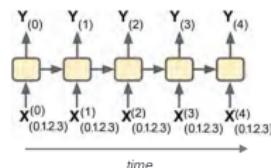
Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers



$y_t$  depende solo de  $x_t$  porque aún no hemos aprendido a conectarlas

# Entendiendo las RNNs

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

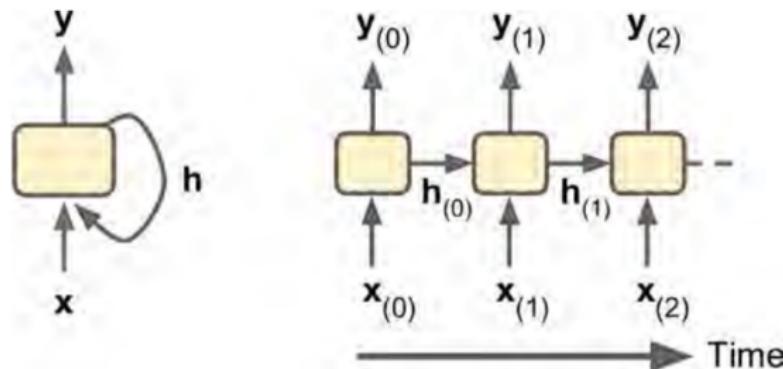
Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers



$$h_t = f_W(x_t, h_{t-1})$$

cell state      function      input      old state  
                  with weights  
                      W

Note: the same function and set of parameters are used at every time step

# Entendiendo las RNNs

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

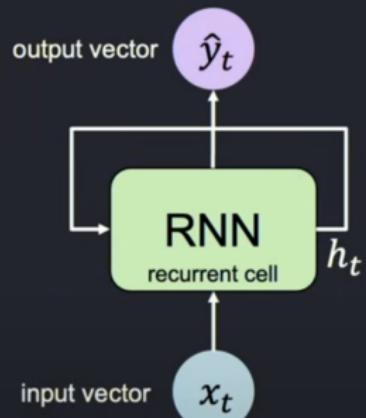
Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

```
my_rnn = RNN()  
hidden_state = [0, 0, 0, 0]  
  
sentence = ["I", "love", "recurrent", "neural"]  
  
for word in sentence:  
    prediction, hidden_state = my_rnn(word, hidden_state)  
  
    next_word_prediction = prediction  
    # >>> "networks!"
```



# Entendiendo las RNNs

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

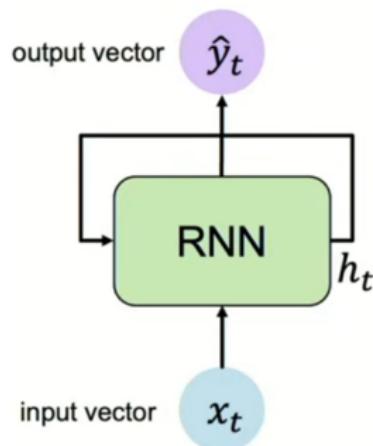
Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers



Output Vector

$$\hat{y}_t = \mathbf{W}_{hy}^T h_t$$

Update Hidden State

$$h_t = \tanh(\mathbf{W}_{hh}^T h_{t-1} + \mathbf{W}_{xh}^T x_t)$$

Input Vector

$$x_t$$

# Entendiendo las RNNS

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

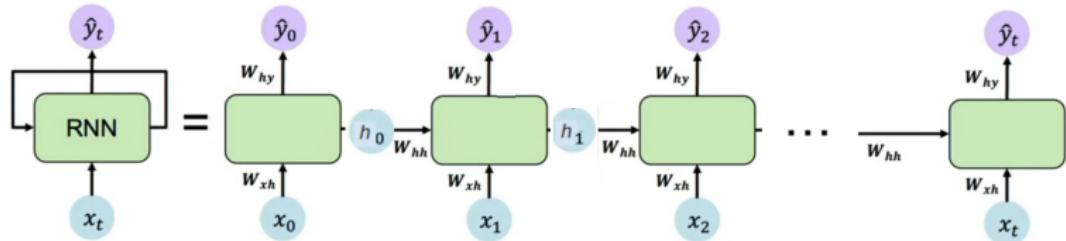
Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

Recordad que hay una sola instancia de las variables que contienen los pesos  $W_{hh}$  y  $W_{xh}$  solo que aparecen en este dibujo muchas veces porque estamos representando cada vuelta del bucle

```
for word in sentence:  
    prediction, hidden_state = my_rnn(word, hidden_state)
```



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

# Definir y Calcular el FeedForward

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

## Definir las capas y el cálculo del FeedForward - los productos escalares (Keras+Pytorch)

### Definición de una RNN y su constructora

```
class MyRNNCell(tf.keras.layers.Layer):
    def __init__(self, rnn_units, input_dim, output_dim):
        super(MyRNNCell, self).__init__()

        # Initialize weight matrices
        self.W_xh = self.add_weight([rnn_units, input_dim])
        self.W_hh = self.add_weight([rnn_units, rnn_units])
        self.W_hy = self.add_weight([output_dim, rnn_units])

        # Initialize hidden state to zeros
        self.h = tf.zeros([rnn_units, 1])
```

### Definir el cálculo de la última capa para la predicción

```
def call(self, x):
    # Update the hidden state
    self.h = tf.math.tanh( self.W_hh * self.h + self.W_xh * x )

    # Compute the output
    output = self.W_hy * self.h

    # Return the current output and hidden state
    return output, self.h
```

# Cálculos Forward

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

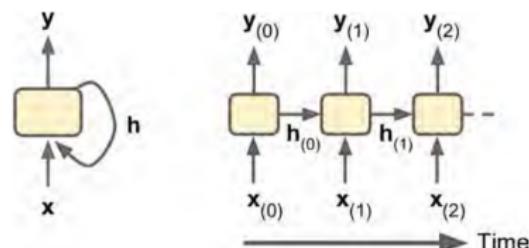
Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers

- Supongamos que la tarea es predecir la siguiente palabra dentro de nuestro diccionario de 5 palabras. ¿Qué arquitectura tendría nuestra red? (¿transparencia 8 o la siguiente?)



$$h_t = f_W(x_t, h_{t-1})$$

cell state      function      input  
                with weights  
                               $W$       old state

Note: the same function and set of parameters are used at every time step

# Cálculos Forward

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers

- ¿Cuál es el tamaño del output bajo la tarea que hemos definido?

$$x \in \mathbb{R}^5$$

[0,	1,	0,	0,	0]
-----	----	----	----	----

Supongamos dimensión de la célula RNN=3

Al comienzo se inicializa a 0

$$h \in \mathbb{R}^3$$

[0,	0,	0]
-----	----	----

¿Cuáles serán las dimensiones de  $W_{xh}$ ,  $W_{hh}$  y  $W_{hy}$ ?

Definición de una RNN y su constructora

```
class MyRNNCell(tf.keras.layers.Layer):
    def __init__(self, rnn_units, input_dim, output_dim):
        super(MyRNNCell, self).__init__()

        # Initialize weight matrices
        self.W_xh = self.add_weight([rnn_units, input_dim])
        self.W_hh = self.add_weight([rnn_units, rnn_units])
        self.W_hy = self.add_weight([output_dim, rnn_units])

        # Initialize hidden state to zeros
        self.h = tf.zeros([rnn_units, 1])
```

# Cálculos Forward

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

$$x \in \mathbb{R}^5 \\ [0, 1, 0, 0, 0]$$
$$h \in \mathbb{R}^3 \\ [0, 0, 0]$$

**IMP!!** hay que sumar los productos escalares de:

$$W_{xh} x \text{ y } W_{hh} h$$

Definir el cálculo de la última capa para la predicción

```
def call(self, x):
    # Update the hidden state
    self.h = tf.math.tanh(self.W_hh * self.h + self.W_xh * x)

    # Compute the output
    output = self.W_hy * self.h

    # Return the current output and hidden state
    return output, self.h
```

Para sumar 2 vectores (o 2 matrices), est@s tienen que tener la **misma dimensión** y demás  **$h_{t+1}$  tiene que tener el mismo tamaño que  $h_t$**

$$x \in \mathbb{R}^5$$

$$W_{xh} \in \mathbb{R}^{5 \times 3}$$

$$h \in \mathbb{R}^3$$

$$W_{hh} \in \mathbb{R}^{3 \times 3}$$

$$[0, 1, 0, 0, 0]$$

$$\begin{bmatrix} 0.2 & 0.1 & 0.2 \\ 0.3 & 0.2 & 0.1 \\ 0.6 & 0.4 & 0.1 \\ 0.8 & 0.1 & 0.2 \\ 0.1 & 0.1 & 0.4 \end{bmatrix}$$

$$W_{xh} * x \in \mathbb{R}^3$$

$$[0.3, 0.2, 0.1]$$

$$[0, 0, 0]$$

$$\begin{bmatrix} 0.2 & 0.1 & 0.2 \\ 0.3 & 0.2 & 0.1 \\ 0.6 & 0.4 & 0.1 \end{bmatrix}$$

$$W_{hh} * h \in \mathbb{R}^3$$

$$[0, 0, 0]$$

# Cálculos Forward

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

outputReal  $\in \mathbb{R}^5$   

0.	0.	0.	1.	0]
----	----	----	----	----

predicciónOutput  $\in \mathbb{R}^?$   

[?]	[?]	[...]	[?]	[?]
-----	-----	-------	-----	-----

```
def call(self, x):  
    # Update the hidden state  
    self.h = tf.math.tanh( self.W_hh * self.h + self.W_xh * x )  
  
    # Compute the output  
    output = self.W_hy * self.h  
  
    # Return the current output and hidden state  
    return output, self.h
```

IMP!!

dim. prod esc. de  $W_{hy} * h$  debe coincidir con dim. del output:  $\text{dim}(W_{hy} * h) == \text{dim}(\text{out})$

Definir el cálculo de la última capa para la predicción

$h \in \mathbb{R}^3$        $W_{hy} \in \mathbb{R}^{3 \times 5}$

$$\begin{matrix} [0. & 0. & 0.] & * & \begin{matrix} 0.2 & 0.1 & 0.2 & 0.1 & 0.1 \\ 0.3 & 0.2 & 0.1 & 0.2 & 0.1 \\ 0.6 & 0.4 & 0.1 & 0.1 & 0.3 \end{matrix} & = & [0 & 0 & 0 & 0 & 0] \end{matrix}$$

```
def call(self, x):  
    # Update the hidden state  
    self.h = tf.math.tanh( self.W_hh * self.h + self.W_xh * x )  
  
    # Compute the output  
    output = self.W_hy * self.h  
  
    # Return the current output and hidden state  
    return output, self.h
```

# Cálculos Forward

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

- Solo quedaría añadir:
  - la función de activación (sigmoide o softmax) y la función de error (o llamada de otra forma coste o loss)
  - una DNN completa (una o varias capas hidden + la función de activación (sigmoide o softmax) y la función de error. Esto añadiría más computación y por lo tanto complejidad al backpropagation.

# Cálculos Backpropagation

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers

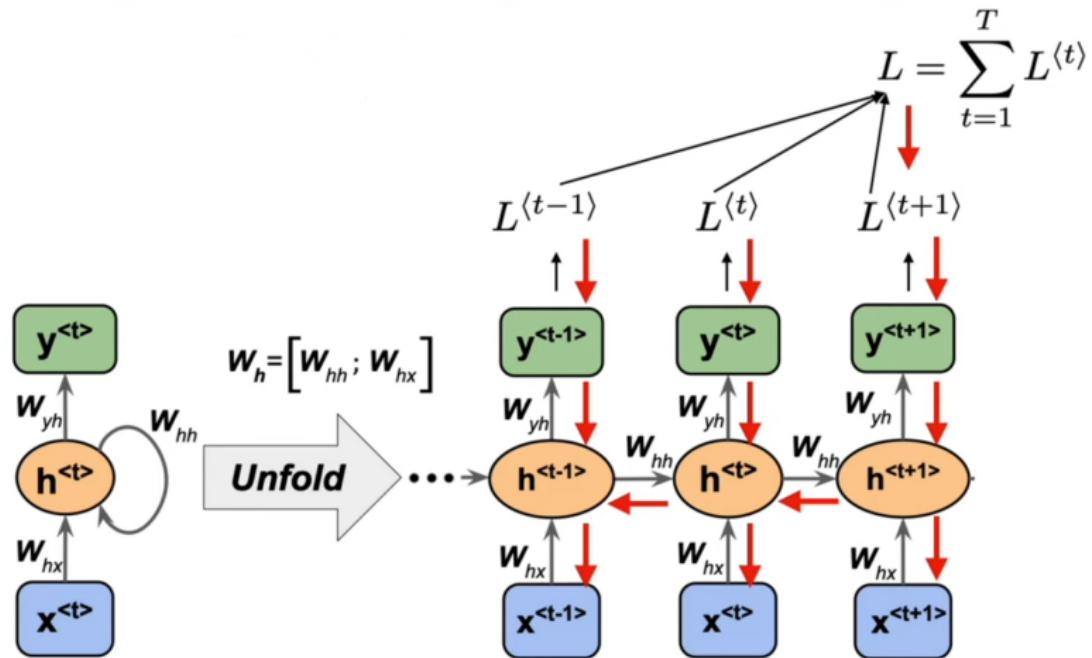


Image source: Sebastian Raschka, Vahid Mirjalili. *Python Machine Learning*. 3rd Edition. Packt, 2019

# Problemas de Vanishing/Exploding

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers

$$L = \sum_{t=1}^T L^{(t)} \quad \frac{\partial L^{(t)}}{\partial \mathbf{W}_{hh}} = \frac{\partial L^{(t)}}{\partial y^{(t)}} \cdot \frac{\partial y^{(t)}}{\partial \mathbf{h}^{(t)}} \cdot \left( \sum_{k=1}^t \boxed{\frac{\partial \mathbf{h}^{(t)}}{\partial \mathbf{h}^{(k)}}} \cdot \frac{\partial \mathbf{h}^{(k)}}{\partial \mathbf{W}_{hh}} \right)$$

computado como multiplicacion de  $\mathbf{h}$  adiacentes

$$\frac{\partial \mathbf{h}^{(t)}}{\partial \mathbf{h}^{(k)}} = \prod_{i=k+1}^t \frac{\partial \mathbf{h}^{(i)}}{\partial \mathbf{h}^{(i-1)}}$$

Vanishing/Exploding gradient!

Ilustración Sebastian Raschka

supongamos  $t=3$

$$\prod_{i=2}^{t=3} \frac{\partial \mathbf{h}^{(i)}}{\partial \mathbf{h}^{(i-1)}} = \frac{\partial \mathbf{h}^{(2)}}{\partial \mathbf{h}^{(1)}} \frac{\partial \mathbf{h}^{(3)}}{\partial \mathbf{h}^{(2)}}$$

Recordemos que  $\mathbf{h} = \text{funAct}((\mathbf{W}_{hh} * \mathbf{h}_{t-1}) + (\mathbf{W}_{xh} * \mathbf{x}_t))$  y por lo tanto la derivada será  $\mathbf{W}_{hh}$  y la de la  $\mathbf{h}$  anterior lo mismo.... es decir termino multiplicando  $\mathbf{W}_{hh}$  tantas veces como de  $i$  a  $t$ . Por lo tanto si los pesos contenidos en  $\mathbf{W}_{hh}$  son muy altos se produce lo que se denomina **Exploding gradients** y si son muy bajos **Vanishing gradients**.

# Problemas de Vanishing/Exploding

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

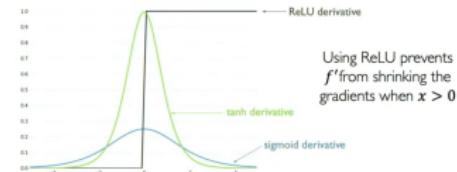
Razonando hacia los  
Transformers

## ■ Aspectos relevantes:

- Cuanto más largas son las secuencias mayor es el problema
- Cuanto peor es la inicialización mayor es el problema
- La selección de la función de activación es relevante

## ■ Soluciones:

- Seleccionar bien activación



- Estrategias de inicialización W

- **LSTM** emplear variantes matemáticas que:

- Permitan "seleccionar" h
- Permitan "sumar" derivadas en vez de multiplicarlas

# Limitaciones

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

El diseño inherentemente secuencial (la posición no es una variable sino que es inherente al sistema) supone:

- Captura limitada de dependencias entre elementos lejanos
- Computación lenta por no poder paralelizar

# Otra vuelta de tuerca

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

¿Qué pasa si la **posición es explícita** (una variable más del sistema) y podemos evitar el procesamiento secuencial pudiendo emplear lo que se llama **self-atención**?

- Esta es la propuesta en la que se basan los **Transformers**
  - GPT (Generative PreTrained **Transformers**)
  - BERT (Bidirectional Encoder Representations from **Transformers**)

# Haciendo explícita la posición

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

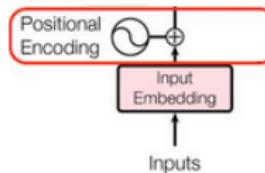
Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers

Por cada elemento (p.e. palabra):



Literalmente se suman. Pero para que esto funcione ¿qué tiene que cumplir los embeddings posicionales o que es deseable que cumplan?

- Deben representar cada posición de manera única
- Deben ser capaces de capturar la distancia relativa entre palabras
- Valores normalizados (no tienen que ser excesivamente grandes ni excesivamente pequeños mejor entre  $[0,1]$  o  $[-1,1]$ ) sino distorsionarían la suma tomando demasiada relevancia con respecto al embedding "semántico"

# Haciendo explícita la posición

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

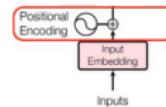
RNN:  
Limitaciones

Razonando hacia los  
Transformers

Por cada elemento (p.e. palabra):

¿índices 0,1,2,3...n cumplen?

- Deben representar cada posición de manera única
- Valores normalizados (ni excesivamente grandes ni excesivamente p.e. entre [0,1] o [-1,1]) sino distorsionarían la suma: demasiada relev. con respecto al embedding "semántico". ¿Y en binario?.
- Deben ser capaces de capturar la distancia relativa entre palabras



0 :	0	0	0	0	8 :	1	0	0	0
1 :	0	0	0	1	9 :	1	0	0	1
2 :	0	0	1	0	10 :	1	0	1	0
3 :	0	0	1	1	11 :	1	0	1	1
4 :	0	1	0	0	12 :	1	1	0	0
5 :	0	1	0	1	13 :	1	1	0	1
6 :	0	1	1	0	14 :	1	1	1	0
7 :	0	1	1	1	15 :	1	1	1	1

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

## ¿índices 0,1,2,3...n cumplen?

### ■ Problema:

- A más long de secuencia más bits, no es un vector de dim. estática
- ¿Por que no explorar los números reales y solo quedarnos con 0,1?

0:	0 0 0 0	8:	1 0 0 0
1:	0 0 0 1	9:	1 0 0 1
2:	0 0 1 0	10:	1 0 1 0
3:	0 0 1 1	11:	1 0 1 1
4:	0 1 0 0	12:	1 1 0 0
5:	0 1 0 1	13:	1 1 0 1
6:	0 1 1 0	14:	1 1 1 0
7:	0 1 1 1	15:	1 1 1 1

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

## Para más detalles <sup>1</sup>

$$\vec{p}_t^{(i)} = f(t)^{(i)} := \begin{cases} \sin(\omega_k \cdot t), & \text{if } i = 2k \\ \cos(\omega_k \cdot t), & \text{if } i = 2k + 1 \end{cases}$$

where

$$\omega_k = \frac{1}{10000^{2k/d}}$$

$$\vec{p}_t = \begin{bmatrix} \sin(\omega_1 \cdot t) \\ \cos(\omega_1 \cdot t) \\ \sin(\omega_2 \cdot t) \\ \cos(\omega_2 \cdot t) \\ \vdots \\ \sin(\omega_{d/2} \cdot t) \\ \cos(\omega_{d/2} \cdot t) \end{bmatrix}_{d \times 1}$$

---

<sup>1</sup>[https://kazemnejad.com/blog/transformer\\_architecture\\_positional\\_encoding/](https://kazemnejad.com/blog/transformer_architecture_positional_encoding/)

# ¿Se puede mejorar?

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

$$x_{\text{embSem}} \in \mathbb{R}^{512} + x_{\text{embPos}} \in \mathbb{R}^{512} = x_{\text{eSP}} \in \mathbb{R}^{512}$$
$$\begin{matrix} 1 & \dots & 512 \end{matrix} \quad W_v \quad V = x^* W_v$$

el 0.4 0.2 ... 0.1

doctor 0.2 0.2 ... 0.2

salvó 0.1 0.1 ... 0.3

al 0.2 0.2 ... 0.1

paciente 0.1 0.2 ... 0.2

$w_{1\_1}$   
 $w_{1\_2}$   
 $w_{1\_3}$   
 $w_{1\_4}$   
...  
 $w_{1\_512}$

$$= \begin{matrix} X^{\text{pos1}*w_{1\_1}} \\ X^{\text{pos2}*w_{1\_2}} \\ X^{\text{pos3}*w_{1\_3}} \\ \dots \\ X^{\text{pos5}*w_{1\_512}} \end{matrix}$$

# ¿Se puede mejorar?

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

Entendiendo las RNNs

Cálculo del Forward: ejemplo

Backprop: problemas

RNN: Limitaciones

Razonando hacia los Transformers

$$x_{\text{embSem}} \in \mathbb{R}^{512} + x_{\text{embPos}} \in \mathbb{R}^{512} = x_{\text{eSP}} \in \mathbb{R}^{512}$$

1            ...            512

$W_v$

el            

0.4	0.2	...	0.1
-----	-----	-----	-----

$w_{1,1}$     ...     $w_{n,1}$

doctor        

0.2	0.2	...	0.2
-----	-----	-----	-----

$w_{1,2}$     ...     $w_{n,2}$

salvó        

0.1	0.1	...	0.3
-----	-----	-----	-----

$w_{1,3}$     ...     $w_{n,3}$

al            

0.2	0.2	...	0.1
-----	-----	-----	-----

$w_{1,4}$     ...     $w_{n,4}$

paciente    

0.1	0.2	...	0.2
-----	-----	-----	-----

$w_{1,512}$     ...     $w_{n,512}$

$$V = x^* W_v$$

$$= \begin{array}{cccccc} x^{pos1*} w_1 & x^{pos1*} w_2 & \dots & \dots & \dots & x^{pos1*} w_n \\ x^{pos2*} w_1 & x^{pos2*} w_2 & \dots & \dots & \dots & x^{pos2*} w_n \\ x^{pos3*} w_1 & x^{pos3*} w_2 & \dots & \dots & \dots & x^{pos3*} w_n \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x^{pos5*} w_1 & x^{pos5*} w_2 & \dots & \dots & \dots & x^{pos5*} w_n \end{array}$$

Esta es una representación más potente

# ¿Qué pasa si modelamos ( añadimos) la rel. (attention) entre pal.?

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

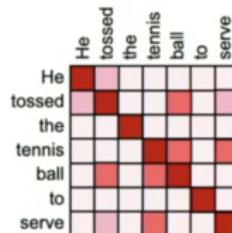
Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers



$$\text{softmax} \left( \frac{Q \cdot K^T}{\text{scaling}} \right)$$

---

Attention weighting

# Todo junto

Intro AS

A. Atutxa

Motivación

Ejemplos de Contextos Secuenciales

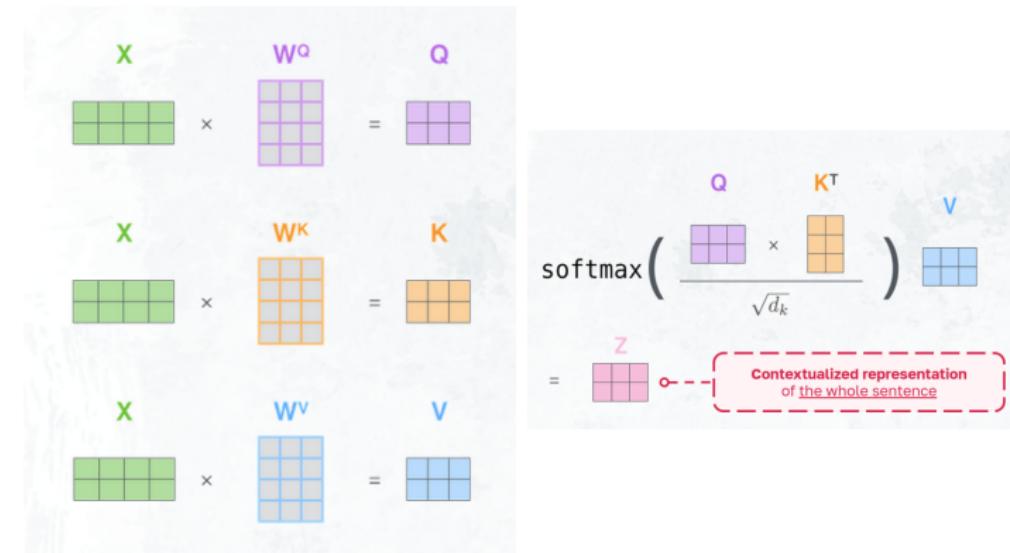
Entendiendo las RNNs

Cálculo del Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando hacia los  
Transformers



# Todo junto

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

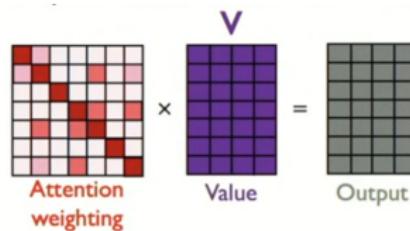
Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers



$$\underbrace{\text{softmax} \left( \frac{Q \cdot K^T}{\text{scaling}} \right)}_{\text{---}} \cdot \underbrace{V}_{\text{---}} = A(Q, K, V)$$

¿Qué tengo que aprender?  $W_Q$ ,  $W_K$  y  $W_V$ .

# Todo Junto

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

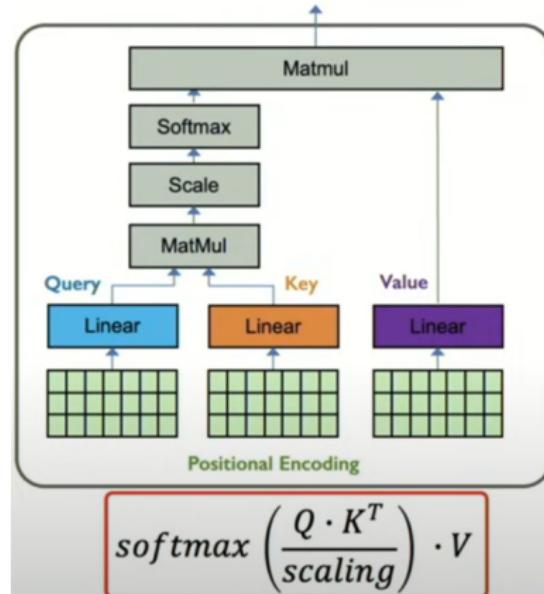
Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers



Representación de una sola Head de Encoder<sup>2</sup>

<sup>2</sup>Vaswani et al. NeurIPS 2017

# Y aún hay más amigos...

Intro AS

A. Atutxa

Motivación

Ejemplos de  
Contextos  
Secuenciales

Entendiendo  
las RNNs

Cálculo del  
Forward:  
ejemplo

Backprop:  
problemas

RNN:  
Limitaciones

Razonando  
hacia los  
Transformers

