Redes Neuronales Recurrentes

Procesamiento de Lenguaje Natural

Francisco Cervantes

Noviembre, 2019

HOY ...

- Representación de palabras
- Word embeddings
- Uso de Word embeddings

Representación de palabras

Para abordar problemas del área de Lenguaje Natural se requiere de un vocavulario

Supongamos que utilizamos un volcabulario **V** con 10,000 palabras.

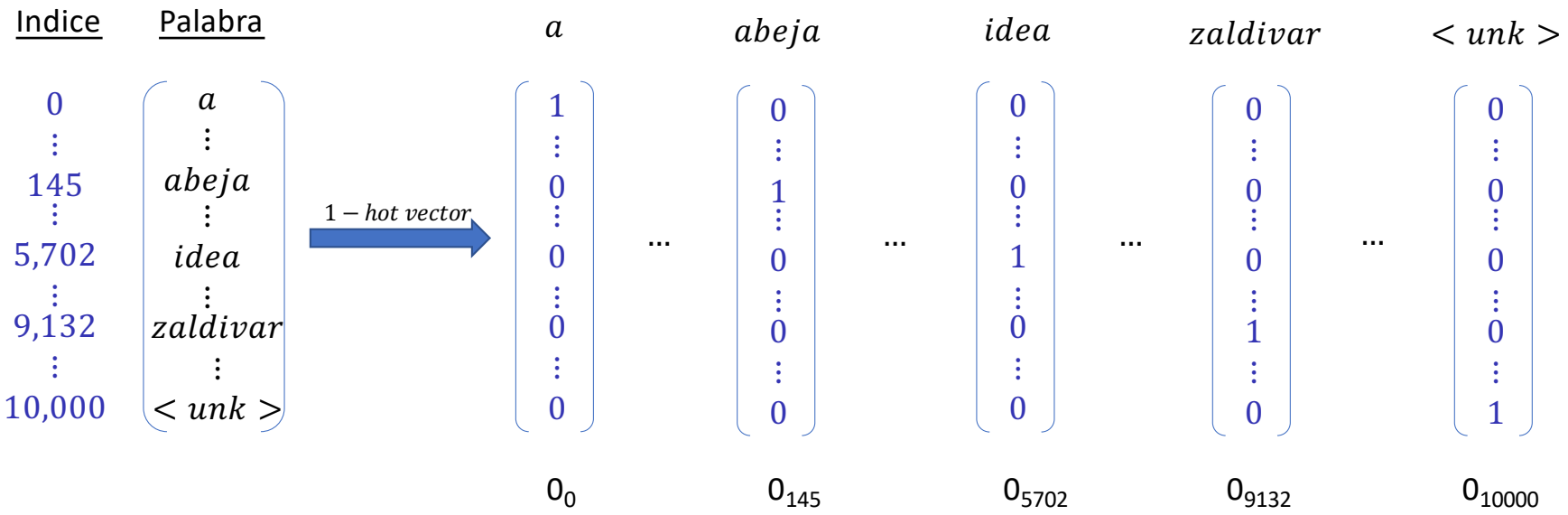
V = [a, ,…, abeja,…, idea, …, zaldivar, … , <unk>] | V | = 10,000

Sin embargo, si queremos utilizar redes LSTM, no podemos utilizar de manera directa las palabras ya que las redes neuornales requieren datos númericos en la entrada.

Una opción es representar las palabras mediante: **One hot vector**

Representación: 1-hot vector

V = [a, ,…, abeja,…, idea, …, zaldivar, … , <unk>] | V | = 10,000



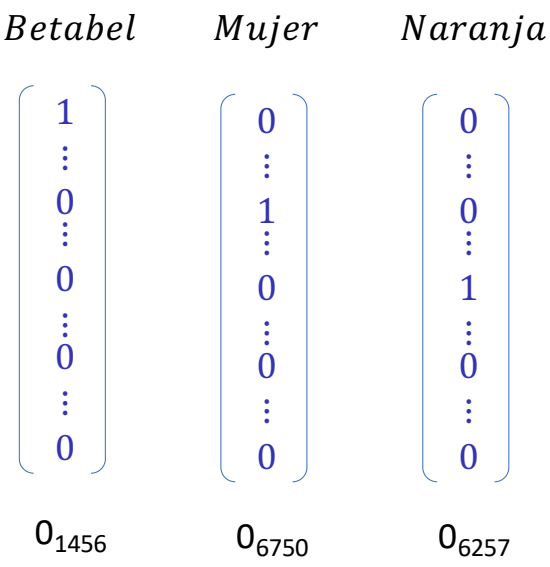
Representación: 1-hot vector

- Un problema de la representación 1-hot vector es que representa cada palabra como un elemento aislado.
- Dificulta la generalización al no considerar la relación que existe entre las palabras (por ejemplo: sinónimos, antonimos, clases, etc.)
- Veamos un ejemplo. Supongamos que tenemos un modelo de lenguaje que ha aprendido que cuando se encuentra la sentencia:

"Quiero un vaso de _____ de naranja"

"Quiero un vaso de _____ de betabel"

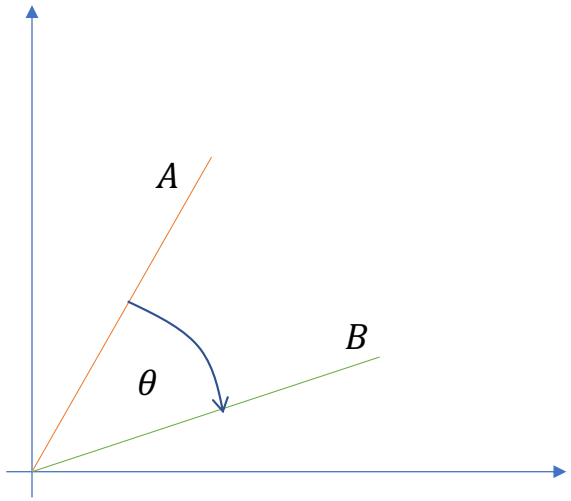
- No podemos saber que la palabra naranja es más cercana a la palabra betabel o a la palabra mujer, hombre, banco, ventana, etc.



Distancia entre dos palabras

- Existen diversas formas de medir la distancia entre dos palabras o texto, por ejemplo: distancia euclidiana, similaridad coseno, similaridad Jaccard.

- Similaridad coseno:



1. Representar las palabras como vectores (por ejemplo A y B).

2. El método consiste en determinar el ángulo entre dos vectores A y B.

3. La similaridad puede ser un valor en el rango de 0 y 1.
   - 1, significa que tienen la misma orientación (no magnitud).

$$similaridad(A,B) = cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$$

Representación con características (Word embeddings)

	Hombre (3941)	Mujer (5391)	Rey (7057)	Reyna (7059)	Betabel (1456)	Naranja (6257)
Género	-1	1	-0.95	0.95	0.0	0.01
Real	0.01	0.02	0.93	0.95	-0.01	0.00
Comida	0.01	0.01	0.02	0.01	0.95	0.97
Antiguedad						
Tamaño						
Costo						
Verbo						
Adjetivo						

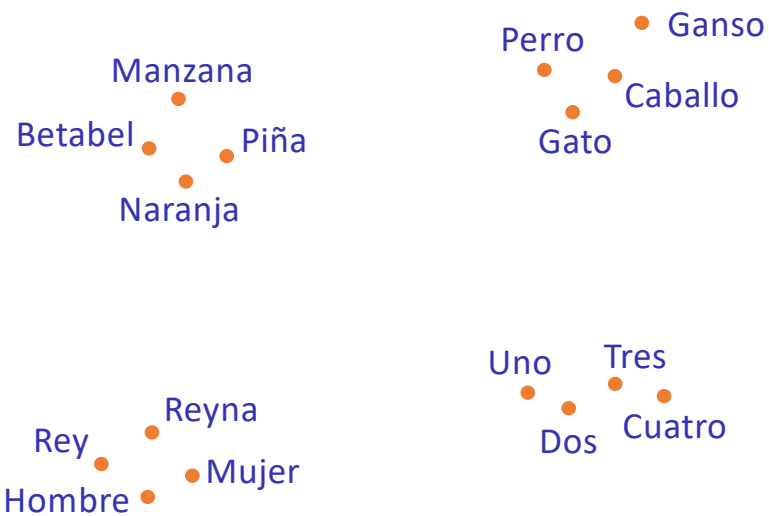
Ahora ¿qué tan similares son las palabras Betabel y Naranja?

- Podríamos tener 200 características.
- Así, cada palabra sería representa por un vector de longitud 200.

Notación:

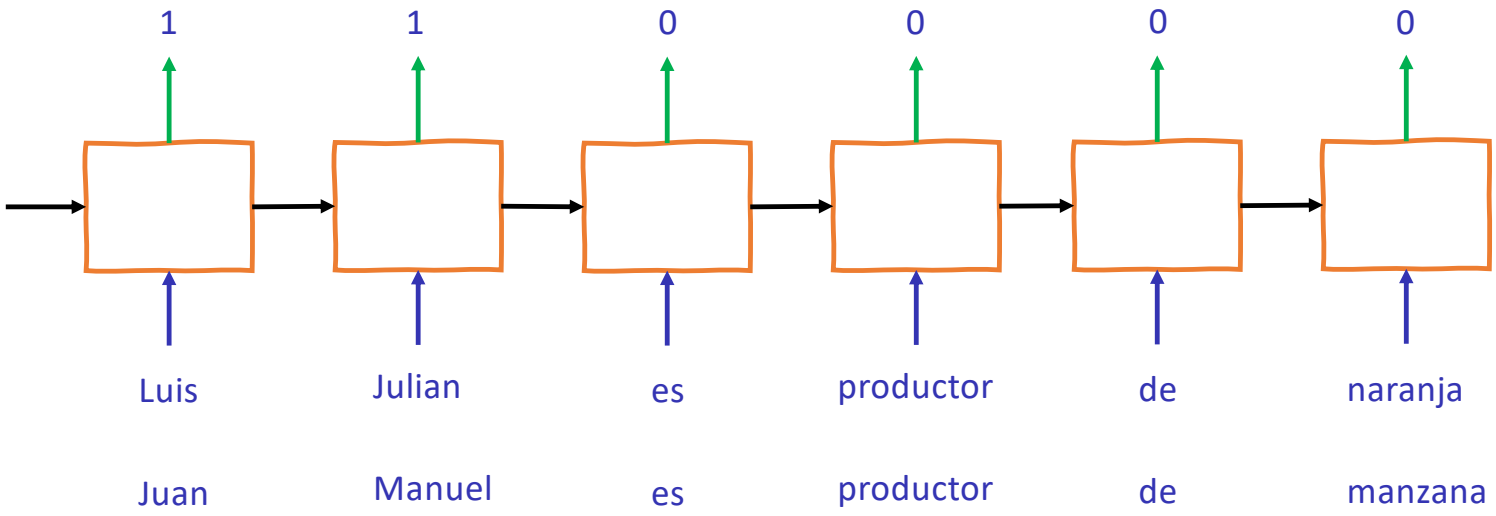
e_{3941} e_{5391} e_{7057} e_{7059} e_{1456} e_{6257}

Visualización de word embeddings



Uso de Word embeddings

- Veamos un ejemplo utilizando el caso del reconocimiento de nombres.



Transfer learning (Word embeddings)

1) Generar Word embedding utilizando grandes corpus de texto (1000 millones – 100,000 millones)

- también puede descargar un word embedding pre-entrenado

2) Transferir el word embedding a una nueva tarea.

- la nueva tarea podría tener un pequeño conjunto de datos de entrenamiento (por ejemplo 100,000 palabras)
- utilizar las características del word embbedding pre-entrenado para representar tus palabras

3) Opcional: continuar entrenando el word embedding con nuevos datos.

Word Embeddings: analogías

	Hombre (3941)	Mujer (5391)	Rey (7057)	Reyna (7059)	Betabel (1456)	Naranja (6257)
Género	-1	1	-0.95	0.95	0.0	0.01
Real	0.01	0.02	0.93	0.95	-0.01	0.00
Comida	0.01	0.01	0.02	0.01	0.95	0.97
Antiguedad	0.03	0.02	0.70	0.69	0.03	-0.02

Hombre es a **Mujer** como

Rey es a **_____ ?**

Veamos una característica interesante del word embedding:

Hombre - Mujer = ?

Rey - Reyna = ?

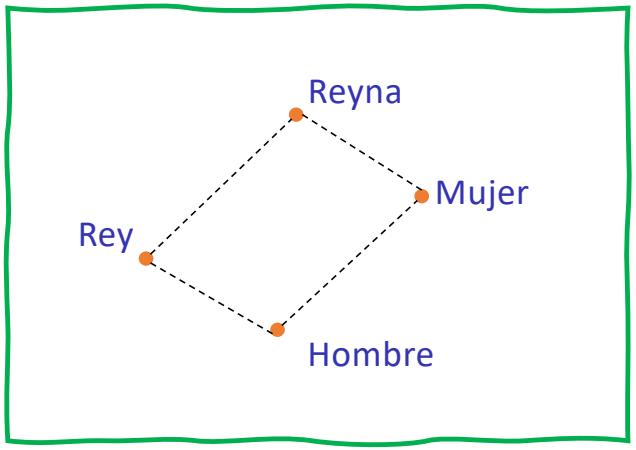
Betabel - Hombre = ?

Word Embeddings: analogías

Hombre es a **Mujer** como

Rey es a **_____ ?**

Representación visual



$$e_{Hombre} - eMu_{jer} \approx e_{Rey} - e_{?}$$

$Buscar\ una\ palabra\ ew$:

$$\max_{e_w} arg \quad sim(ew, e_{Rey} - e_{Hombre} + e_{Mujer})$$