Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

Sistemas Urbanos Inteligentes

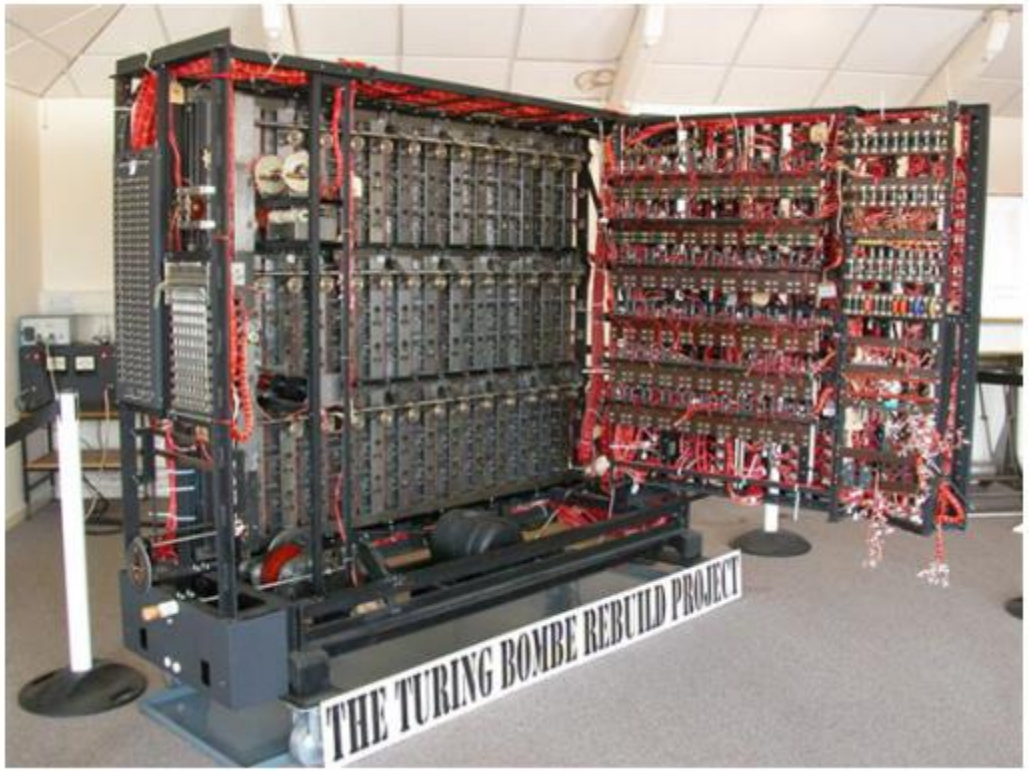
Modelos de lenguaje

Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación

El lenguaje es uno de los grandes logros de la inteligencia humana

El lenguaje puede verse como un mecanismo para codificar información…

El lenguaje es uno de los grandes logros de la inteligencia humana



…pero para extraerla, es necesario decodificarla

En esta clase cubriremos ambas tareas
desde la óptica de las redes neuronales

- Cómo codificar lenguaje de manera simple y con muchos datos.
- Cómo mejorar esta codificación integrándola con la decodificación.
- ¿Y qué tiene que ver esto con los sistemas urbanos?
- Una parte importante de las preferencias, opiniones y comportamientos de los habitantes pueden inferirse del texto que generan.
- Si bien esto lo veremos en el contexto de lenguaje, es aplicable a cualquier tipo de secuencia.

¿Cómo podemos codificar lenguaje?

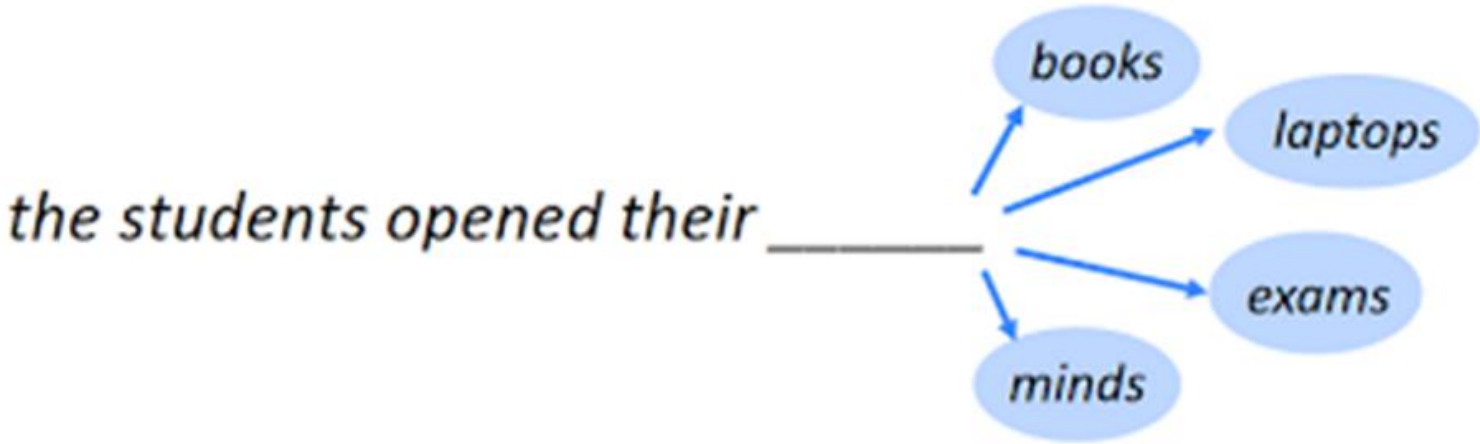
Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up… very very very good movie.

('the', 8),
(',', 5),
('very', 4),
('.', 4),
('who', 4),
('and', 3),
('good', 2),
('it', 2),
('to', 2),
('a', 2),
('for', 2),
('can', 2),
('this', 2),
('of', 2),
('drama', 1),
('although', 1),
('appeared', 1),
('have', 1),
('few', 1),
('blank', 1)
…..

¿Qué problemas tiene esta codificación?

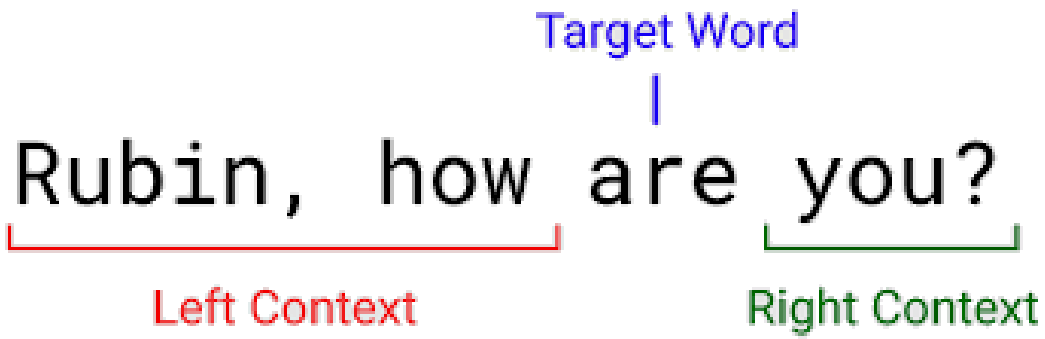
- No considera el orden de los elementos (letras, palabras, etc.)
- No genera un espacio semántico: vectores cercanos no son semánticamente similares
- Nuestro primer enfoque intentará solucionar esto utilizando representaciones continuas y muchos datos.

Modelos de lenguaje neuronales

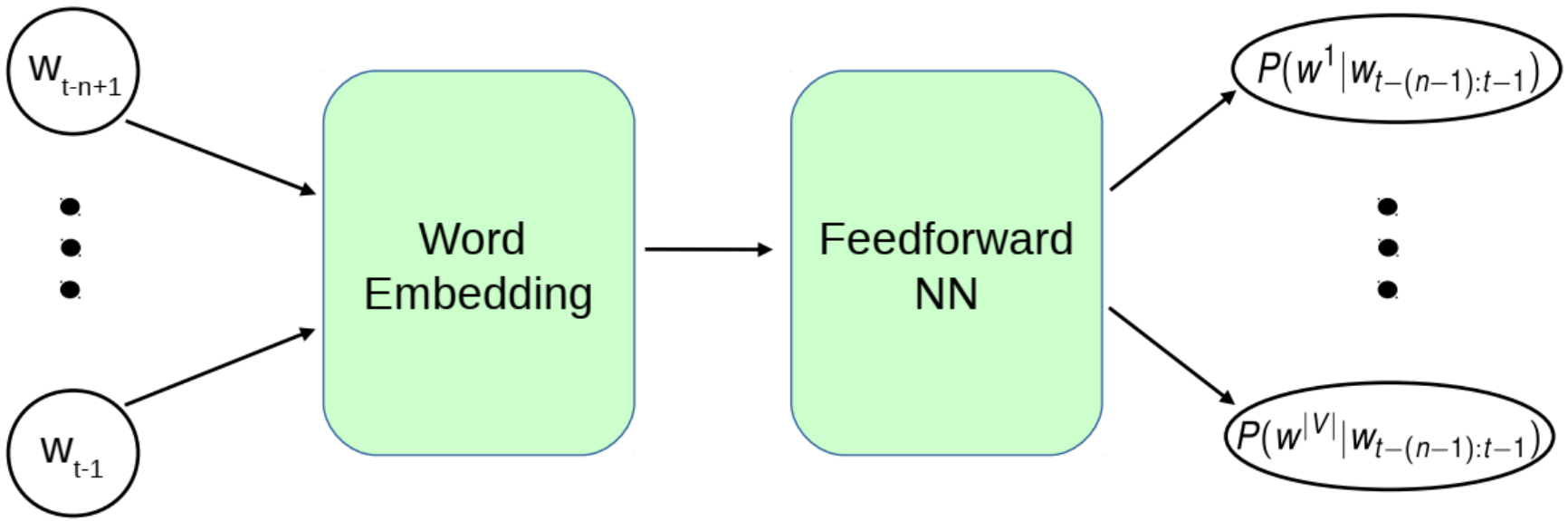


Modelos de lenguaje neuronales

Modelos de lenguaje neuronales

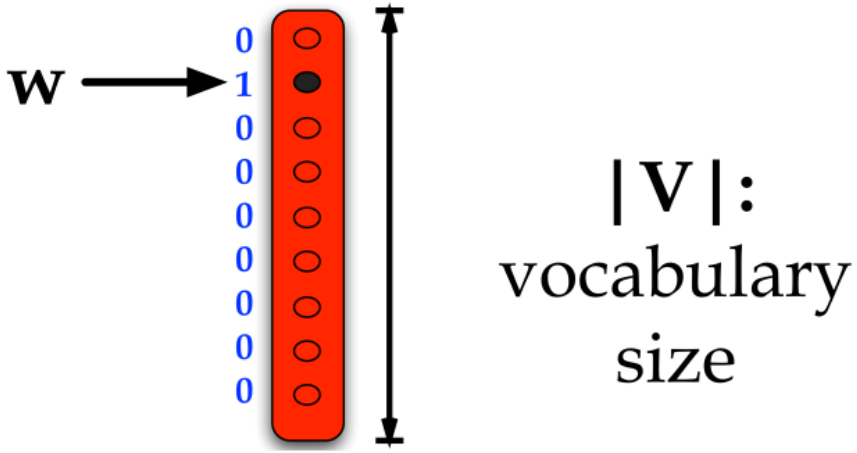


Modelos de lenguaje neuronales

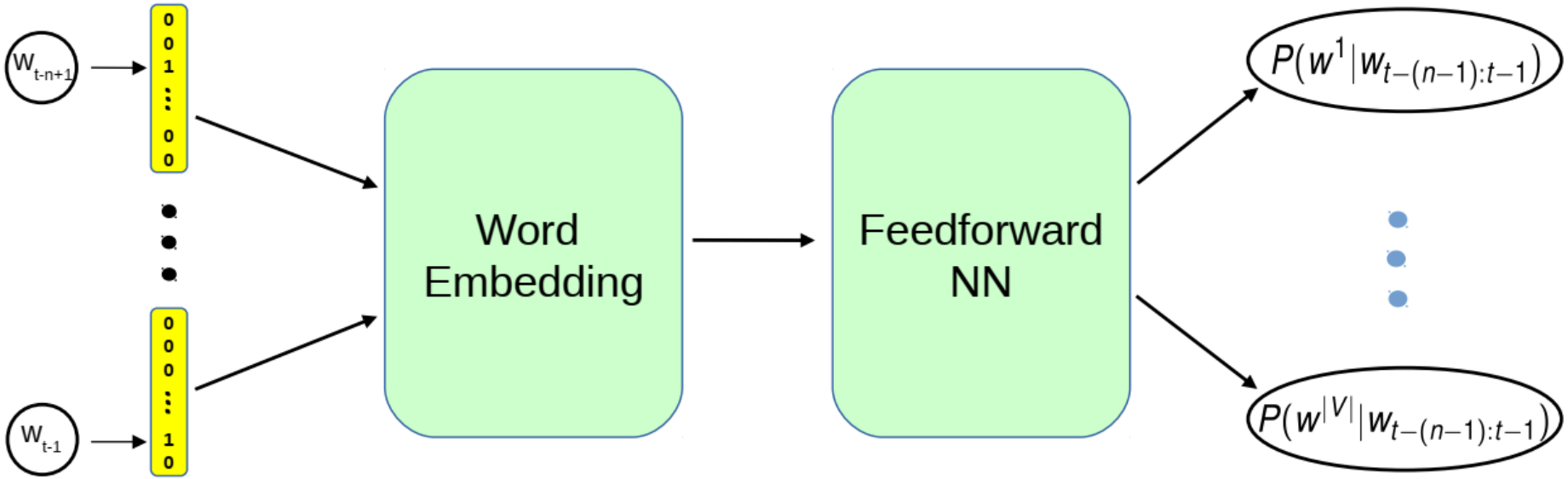


Modelos de lenguaje neuronales

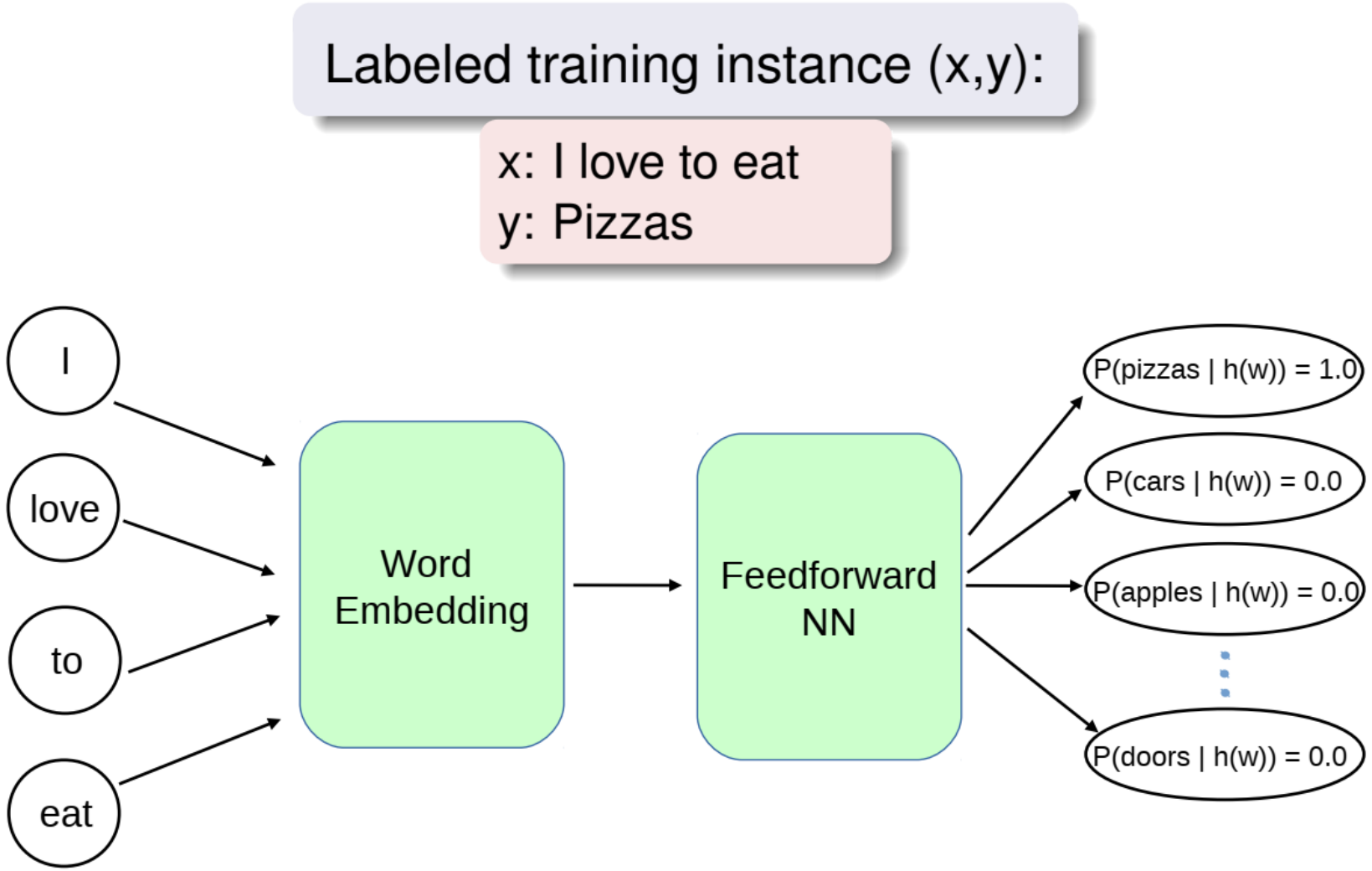
- Típicamente las palabras son representadas mediante una codificación *one-hot*



Modelos de lenguaje neuronales



Modelos de lenguaje neuronales



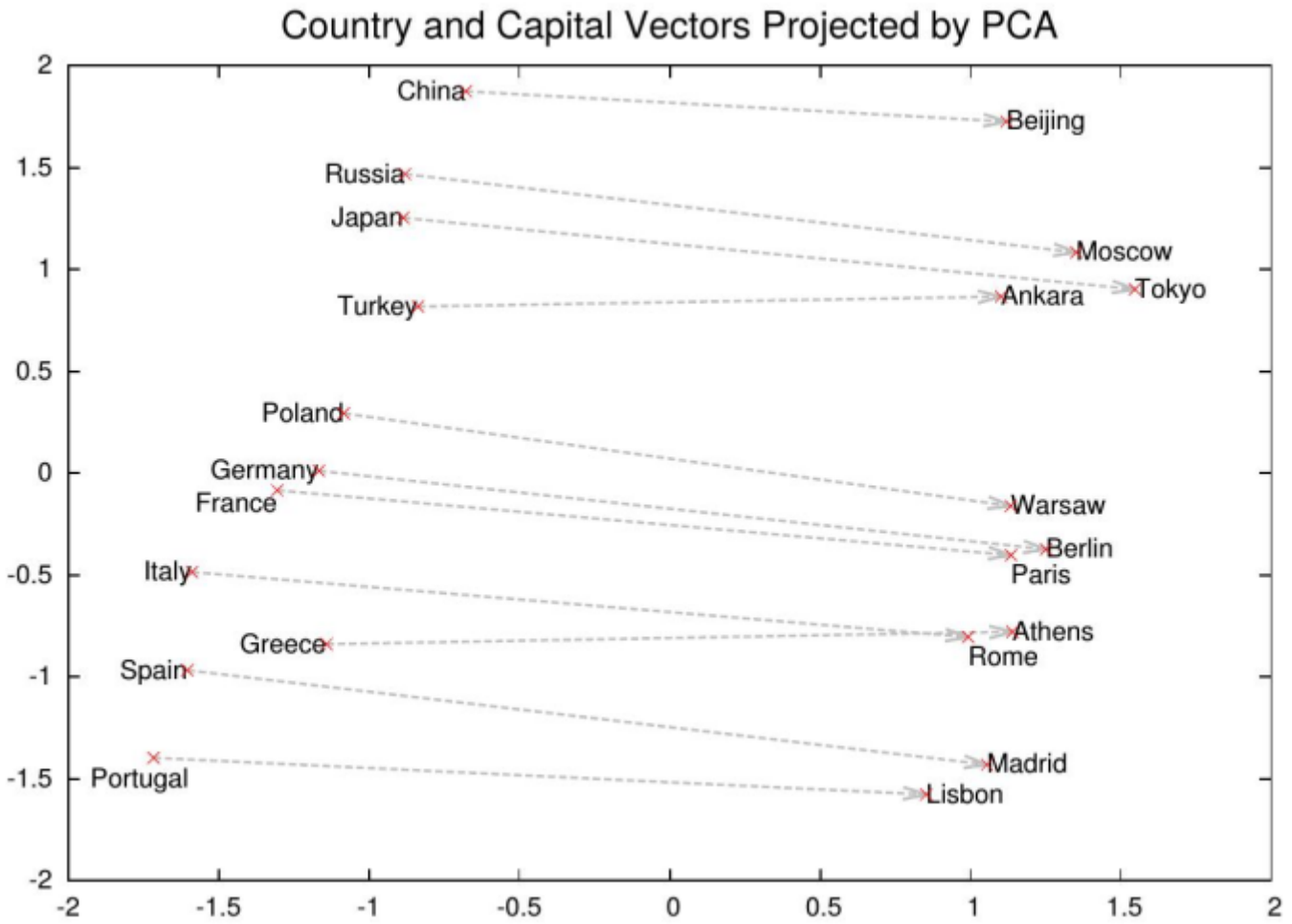
Al terminar el entrenamiento, el producto esencial es la matriz de *embedding*

- La matriz de *embedding* entrega para cada palabra del vocabulario una representación vectorial.
- Dado el entrenamiento conjunto entre *embedding* y clasificador, la representación aprendida es buena para clasificar.
- De manera esperable, al entrenar con muchos datos, esta representación, además de ser buena para clasificar, captura un espacio semántico para las palabras.
- Las técnicas de word embeddings más populares son *word2vec* y *Glove*.

A diferencia de los embeddings categóricos, el espacio de *word embeddings* captura además relaciones a través de operaciones vectoriales

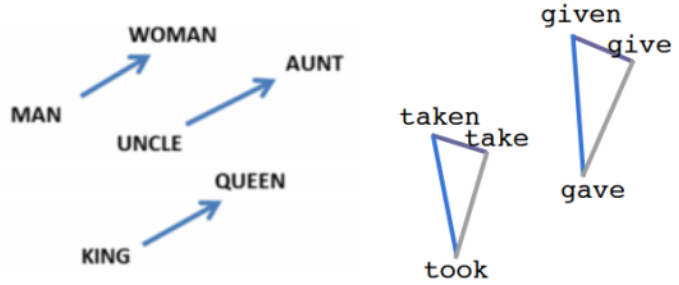
Czech + currency	Vietnam + capital	German + airlines	Russian + river	French + actress
koruna	Hanoi	airline Lufthansa	Moscow	Juliette Binoche
Check crown	Ho Chi Minh City	carrier Lufthansa	Volga River	Vanessa Paradis
Polish zolty	Viet Nam	flag carrier Lufthansa	upriver	Charlotte Gainsbourg
CTK	Vietnamese	Lufthansa	Russia	Cecile De

A diferencia de los embeddings categóricos, el espacio de *word embeddings* captura además relaciones a través de operaciones vectoriales



A diferencia de los embeddings categóricos, el espacio de *word embeddings* captura además relaciones a través de operaciones vectoriales

FRANCE	JESUS	XBOX	REDDISH	SCRATCHED	MEGABITS
AUSTRIA	GOD	AMIGA	GREENISH	NAILED	OCTETS
BELGIUM	SATI	PLAYSTATION	BLUISH	SMASHED	MB/S
GERMANY	CHRIST	MSX	PINKISH	PUNCHED	BIT/S
ITALY	SATAN	IPOD	PURPLISH	POPPED	BAUD
GREECE	KALI	SEGA	BROWNISH	CRIMPED	CARATS
SWEDEN	INDRA	PSNUMBER	GREYISH	SCRAPED	KBIT/S
NORWAY	VISHNU	HD	GRAYISH	SCREWED	MEGAHERTZ
EUROPE	ANANDA	DREAMCAST	WHITISH	SECTIONED	MEGAPIXELS
HUNGARY	PARVATI	GEFORCE	SILVERY	SLASHED	GBIT/S
SWITZERLAND	GRACE	CAPCOM	YELLOWISH	RIPPED	AMPERES



Woman – Man ≈ Aunt - *Uncle*
King - Male + Female ≈ *Queen*
Human - Animal ≈ *Ethics*

¿Qué debilidades tiene este esquema de codificación?

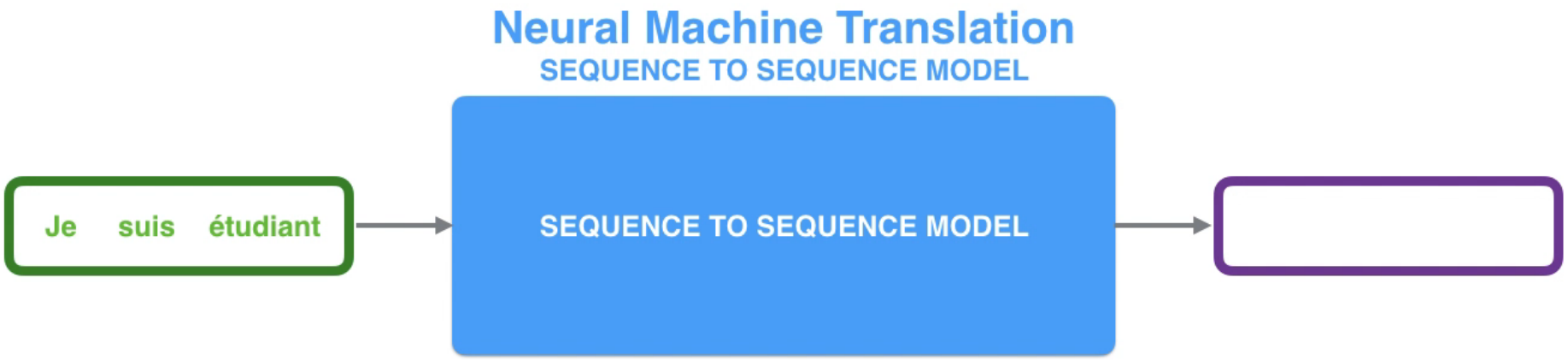
- Al no considerar explícitamente la información secuencial, la matriz de *embedding* genera codificaciones para palabras, pero no para frases.
- Para codificar una frase se requiere un paso extra, como sumar o promediar las representaciones.
- Nuestro segundo enfoque intentará solucionar esto al considerar explícitamente y de manera conjunta el orden de las palabras tanto en la codificación como la decodificación.

Modelos seq2seq al rescate

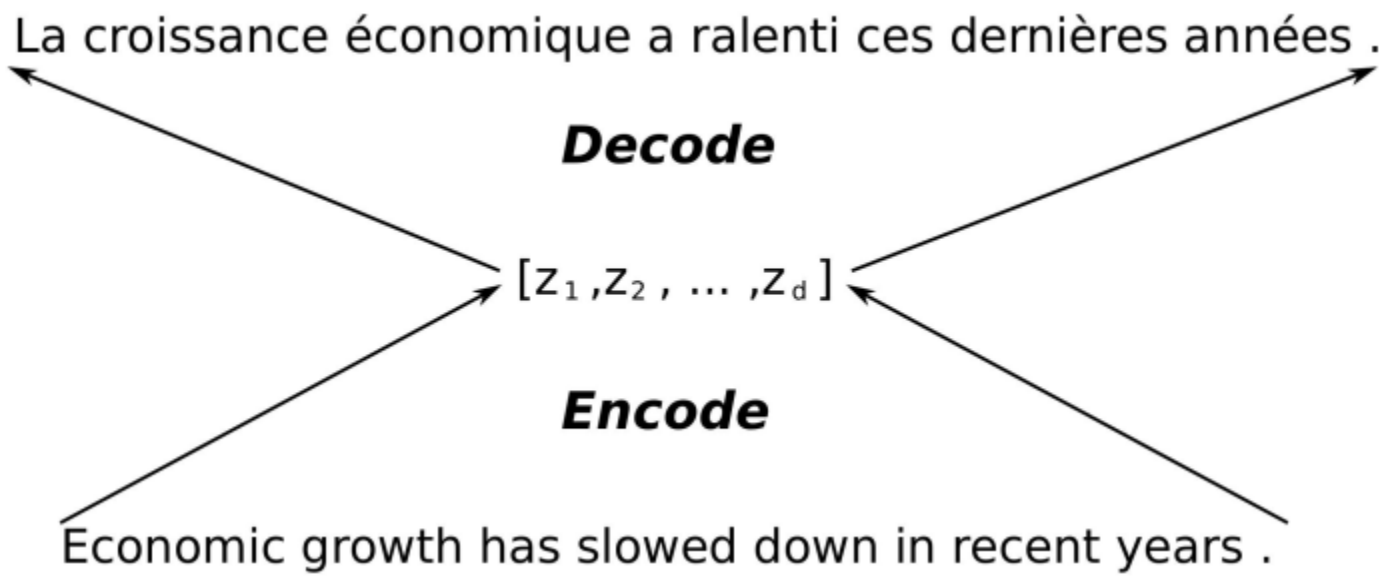
- Con el fin de capturar más información contextual, los modelos seq2seq utilizan redes recurrentes para entrada y salida.
- En general, todos utilizan una arquitectura *encoder-decoder*:
  - Una red recurrente codificadora (*encoder*) procesa la secuencia de entrada y genera un representación vectorial (*estado oculto*) que captura la información relevante.
  - Una red recurrente decodificadora (*decoder*) utiliza esta representación para generar una salida adecuada.
- Ambas redes se entrenan de manera conjunta en este esquema, permitiendo capturar información contextual más rica.
- Este esquema es completamente general, lo que lo hace aplicable a cualquier tipo de secuencia.

Modelos seq2seq al rescate

Modelos seq2seq al rescate

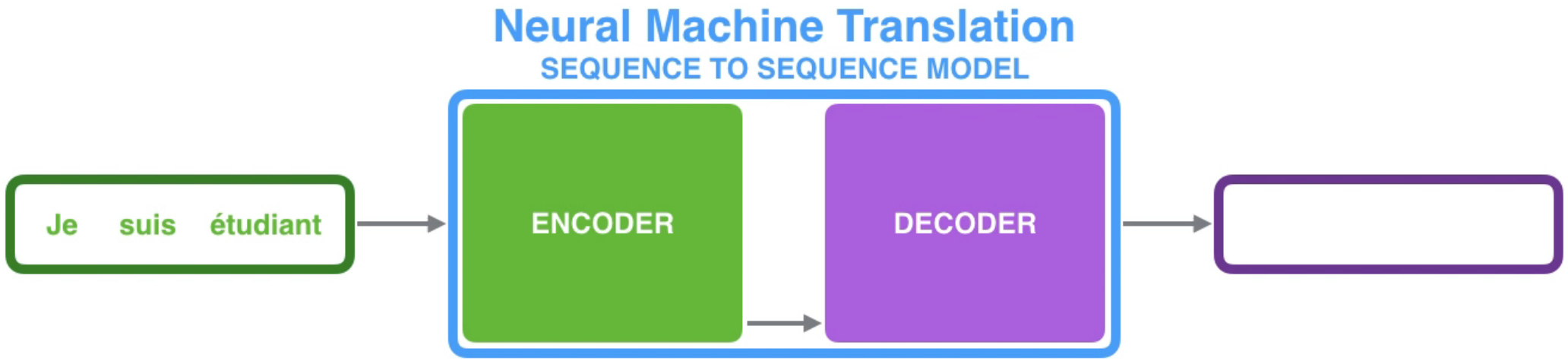


Modelos seq2seq al rescate

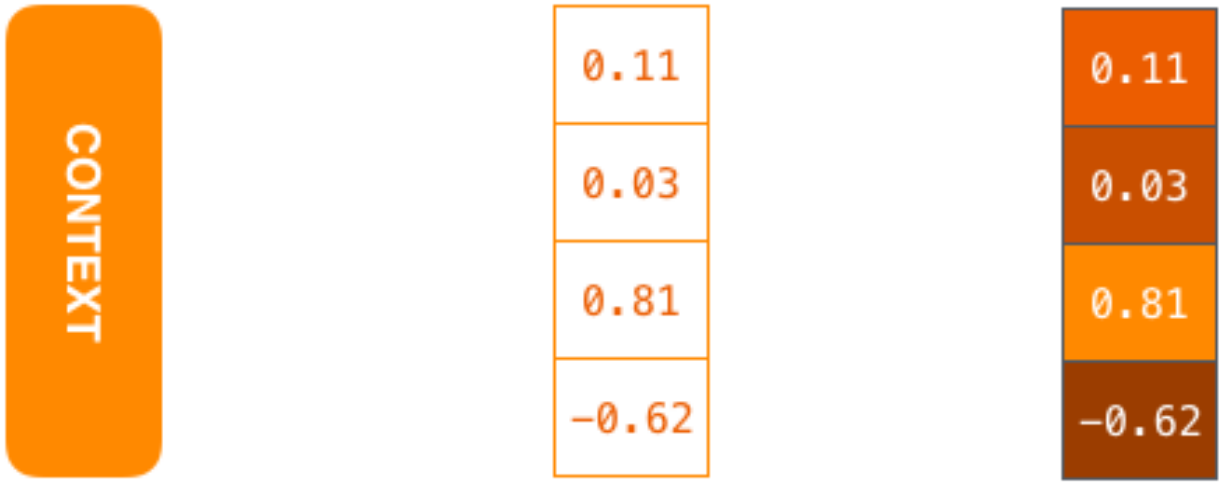


- Al usar una representación latente intermedia, las secuencias de entrada y salida pueden tener largos distintos.
- Esto nos entrega la intuición que esta representación codifica la información relevante de la entrada.
- Además, al usar secuencias, esta representación considera el orden de las palabras, a diferencia de BoW.

El modelo más simple considera un *encoder* que genera un único vector contextual C

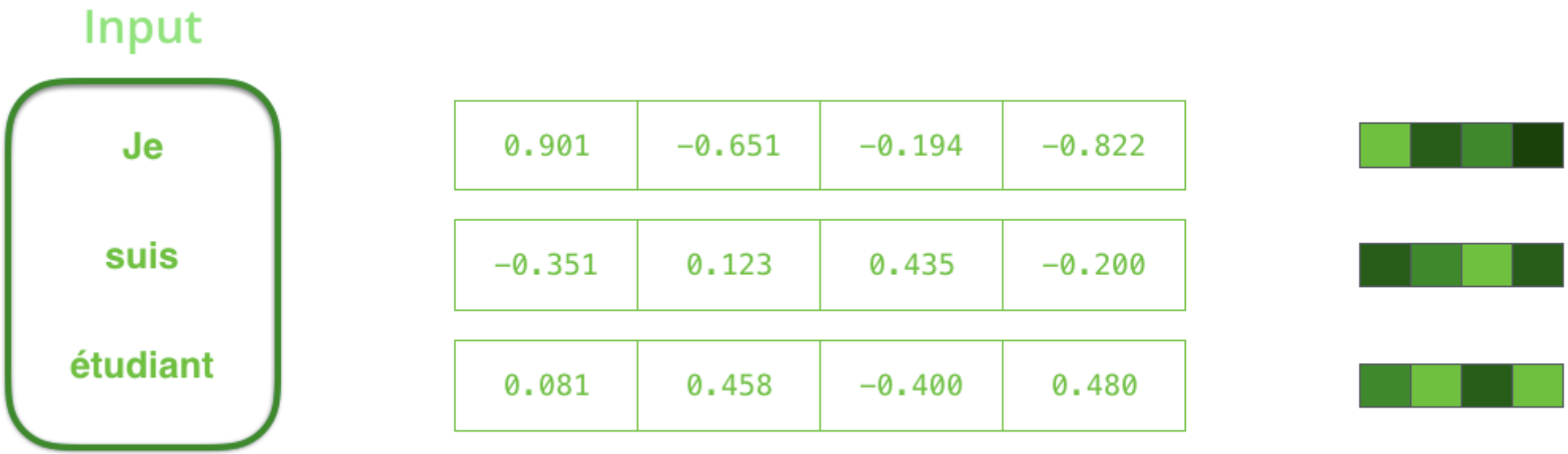


El modelo más simple considera un *encoder* que genera un único vector contextual C



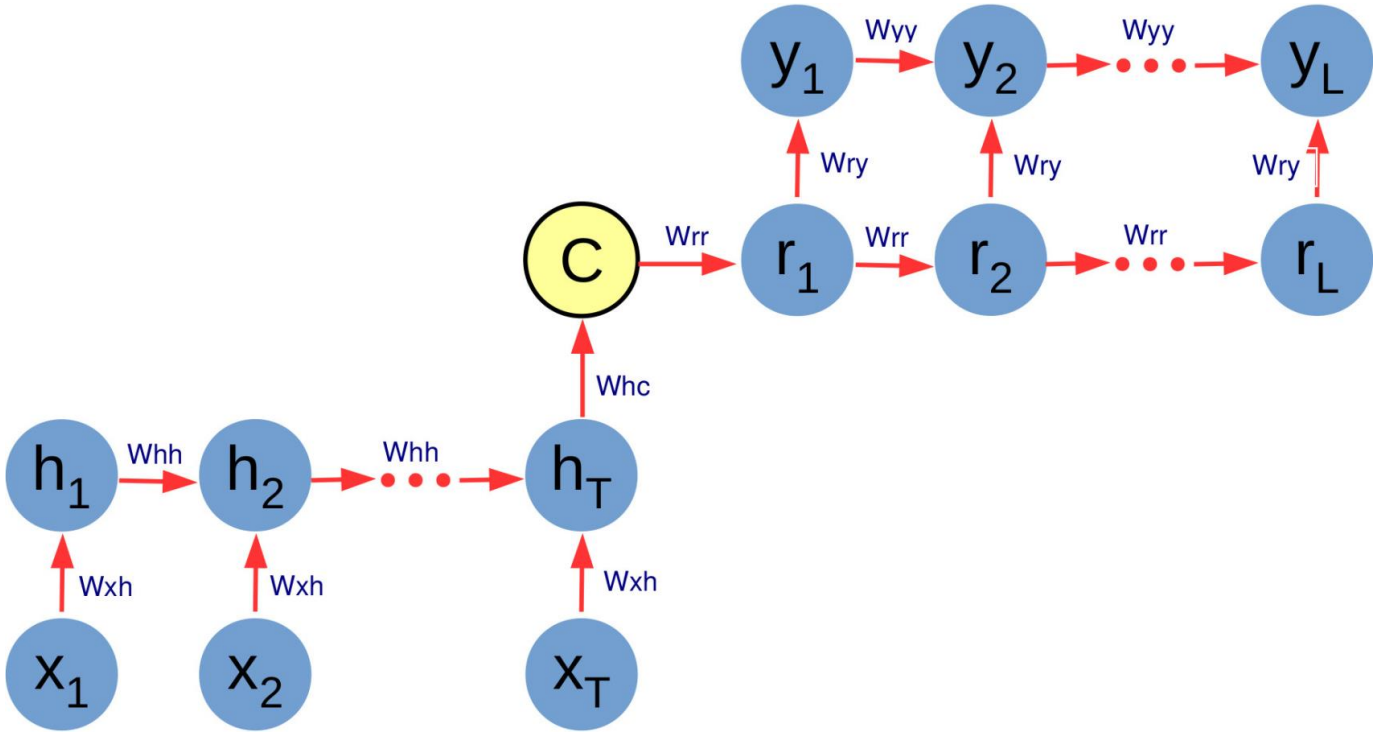
El vector de contexto es un vector de estado oculto como cualquier otro

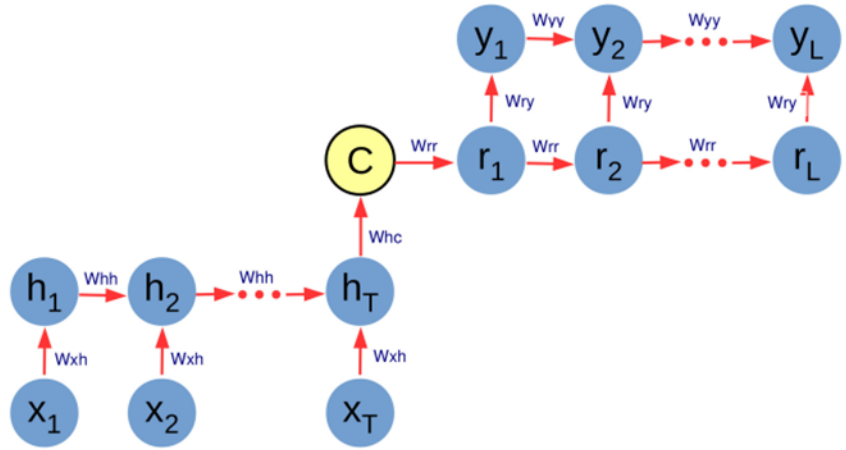
El modelo más simple considera un *encoder* que genera un único vector contextual C



Generalmente, la entrada no son las palabras en sí (o sus representación de BoW), sino que los *embeddings* de cada una de estas

El modelo más simple considera un *encoder* que genera un único vector contextual C





1. Encoder RNN

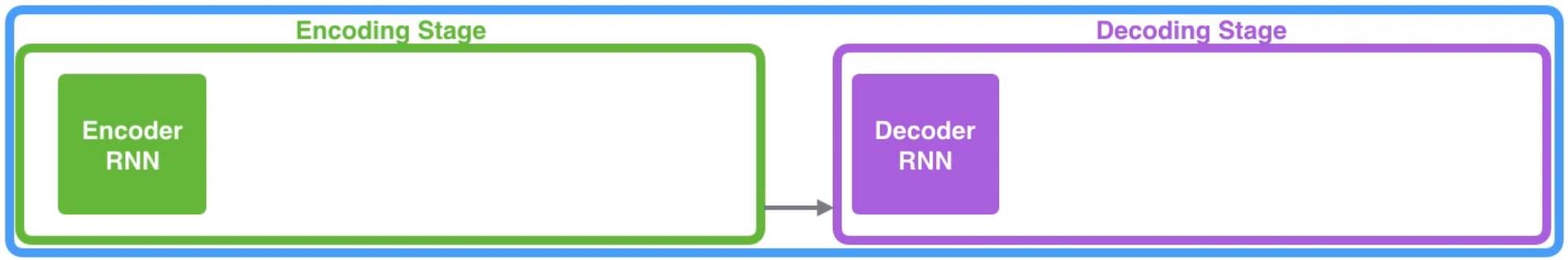
$$h_t = \sigma(W_{xh}x_t + W_{hh}h_{t-1})$$

2. Decoder RNN

$$if(t > 1)$$
$$y_t = \sigma(W_{ry}r_t + W_{yy}y_{t-1})$$
$$r_t = \sigma(W_{rr}r_{t-1})$$
$$if(t = 1)$$
$$y_1 = \sigma(W_{ry}r_1)$$
$$r_1 = \sigma(W_{rr}C)$$
$$C = \sigma(W_{hC}h_T)$$

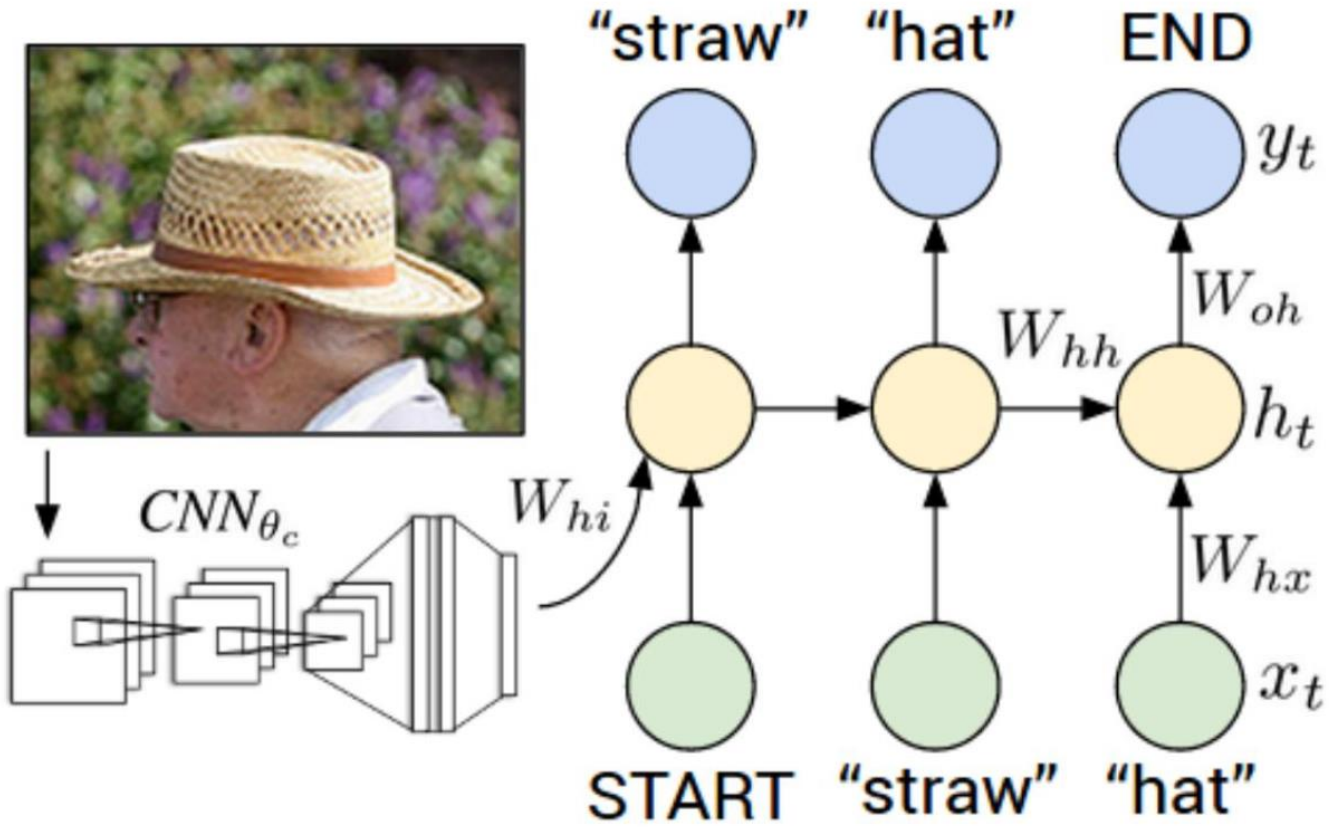
Neural Machine Translation

SEQUENCE TO SEQUENCE MODEL



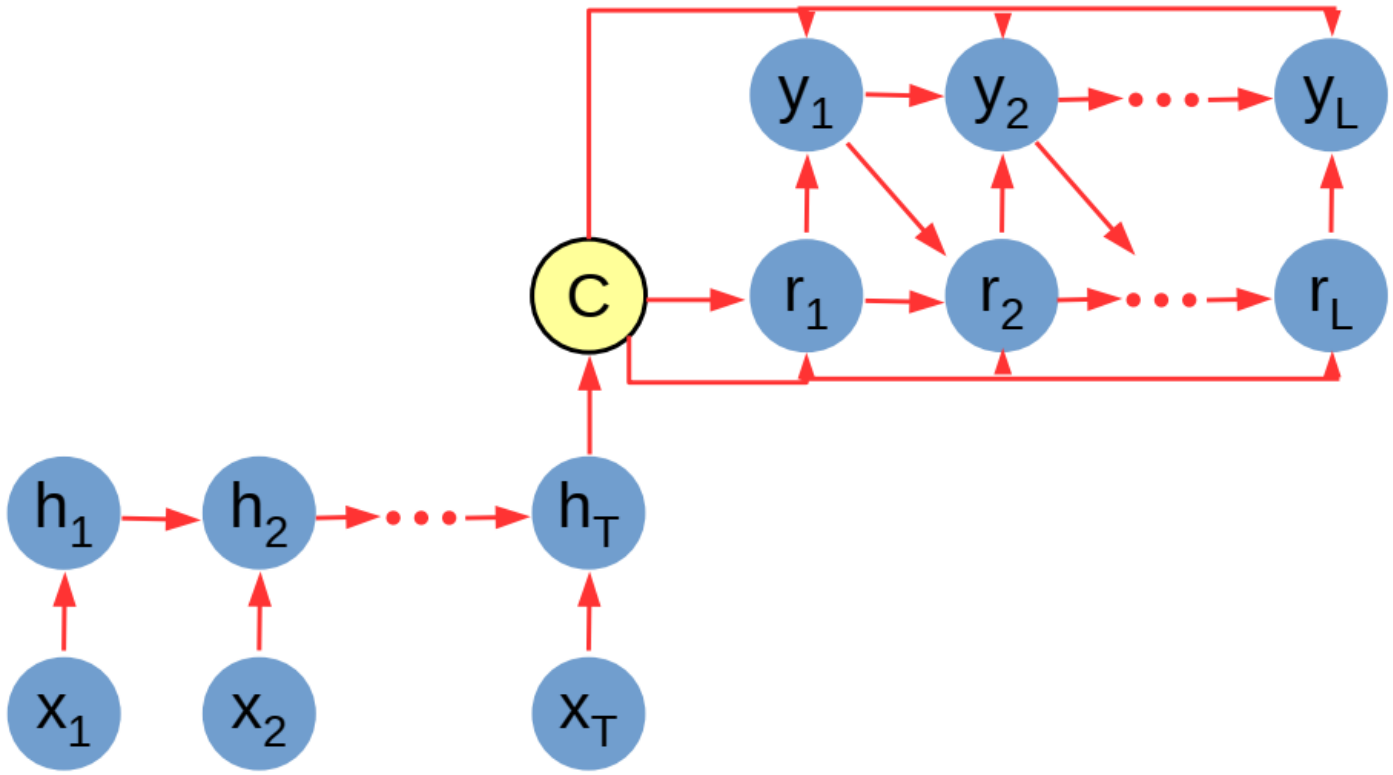
Je suis étudiant

Si lo pensamos bien, el esquema anterior es muy similar al generador de *captions* que vimos anteriormente



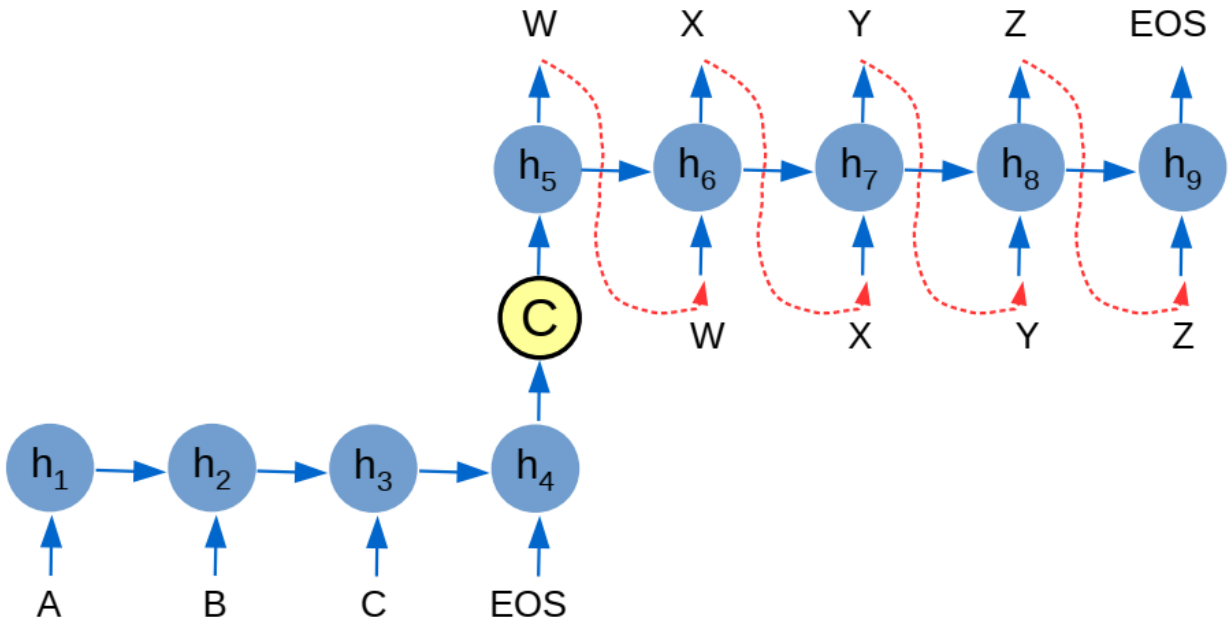
La principal diferencia es que el codificador es acá una CNN. Así, podemos ver este sistema como un traductor del *"lenguaje"* visual al escrito.

También es posible utilizar modelos con configuraciones más complejas

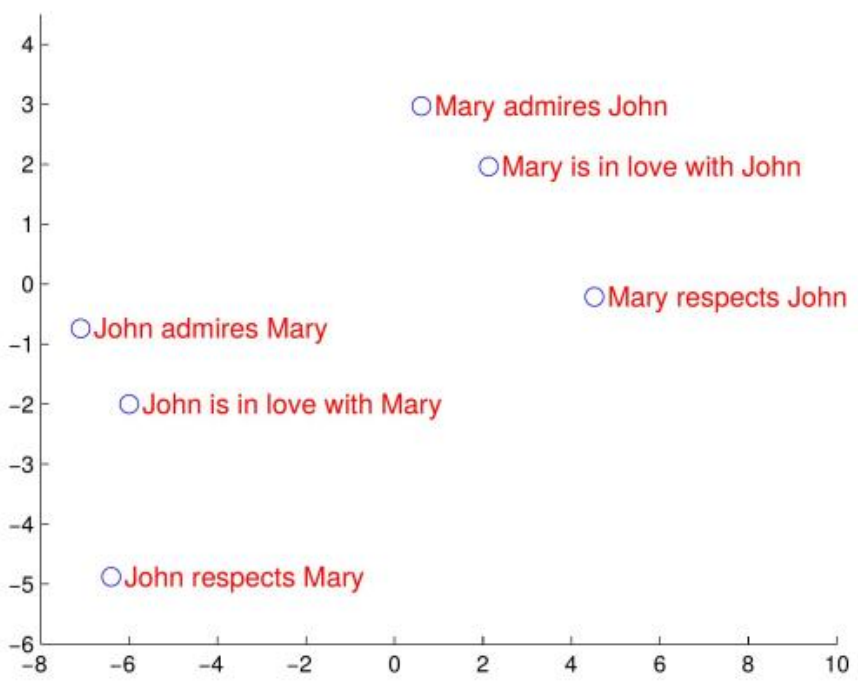


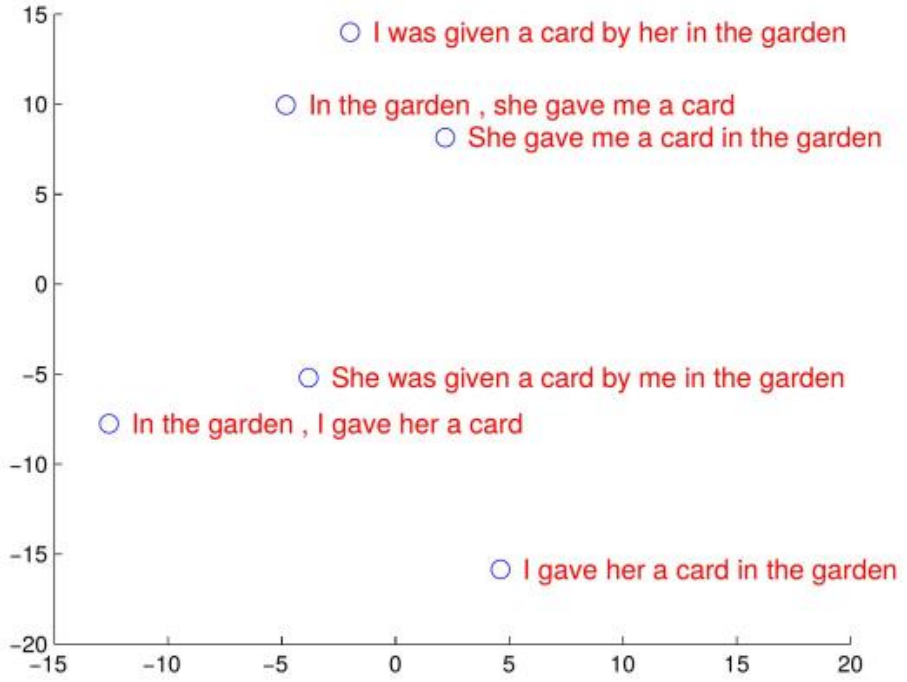
Veamos un ejemplo real: traducción de texto automática (Sutskever et al., 2014)

- Basado en modelo *encoder-decoder*. Ambos son implementados con LSTM de 4 capas.
- El vector de contexto entregado al *decoder* es el último estado oculto del *encoder*.
- Para terminar una secuencia, se utiliza el símbolo especial <EOS>.
- Durante el entrenamiento, los valores reales de la traducción son utilizados como entrada al *decoder*. En test, se utilizan los outputs del paso anterior.



Veamos un ejemplo real: traducción de texto automática (Sutskever et al., 2014)





Al graficar en 2D (PCA) el vector de contexto para distintas frases, vemos que este captura información semántica y que además es sensible al orden de las palabras.

Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ingeniería de Transporte y Logística

Sistemas Urbanos Inteligentes

Modelos de lenguaje

Hans Löbel
Dpto. Ingeniería de Transporte y Logística
Dpto. Ciencia de la Computación