

Sistemas Recomendadores

IIC-3633

Deep Learning en Sistemas Recomendadores
Parte 1

Esta clase

1. Modelos de Lenguaje
2. Deep Learning para recomendación (Modelos de Lenguaje)

Recomendación de contenido de texto hasta ahora...

$$TF(t, d) = \frac{\text{number of times } t \text{ appears in } d}{\text{total number of terms in } d}$$

$$IDF(t) = \log \frac{N}{1 + df}$$

$$TF - IDF(t, d) = TF(t, d) * IDF(t)$$

¿Qué ventajas / desventajas tiene TF-IDF?

Ventajas

- Se adapta al corpus porque se basa en frecuencia de términos
- No necesita entrenar un modelo

Desventajas

- Tiene muy alta dimensionalidad por vector y muy “sparsed”
- No aprende información semántica del texto
- No entiende palabras que aparecen en distintos contextos

Modelos de lenguaje

WORD2VEC

RNN

BERT

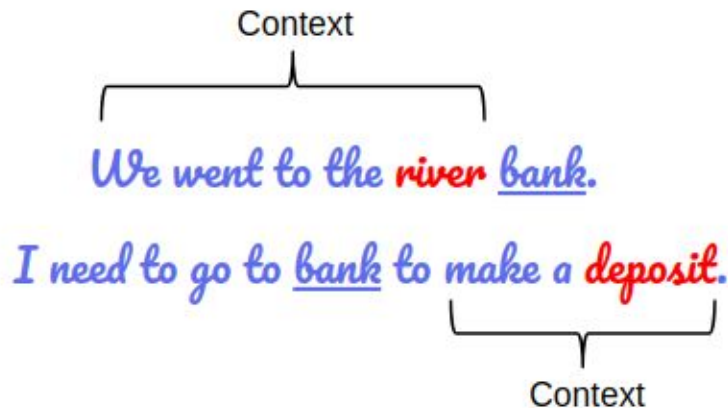
MODELOS GENERATIVOS

Formulación de modelo de lenguaje

- Un modelo de lenguaje permite calcular la probabilidad de una palabra (o n-grama) dada una serie de “eventos” (palabras o n-gramas) observados:

$$\begin{aligned} P(w_{1:n}) &= P(w_1)P(w_2|w_1)P(w_3|w_{1:2}) \dots P(w_n|w_{1:n-1}) \\ &= \prod_{k=1}^n P(w_k|w_{1:k-1}) \end{aligned}$$

Vectorización de texto



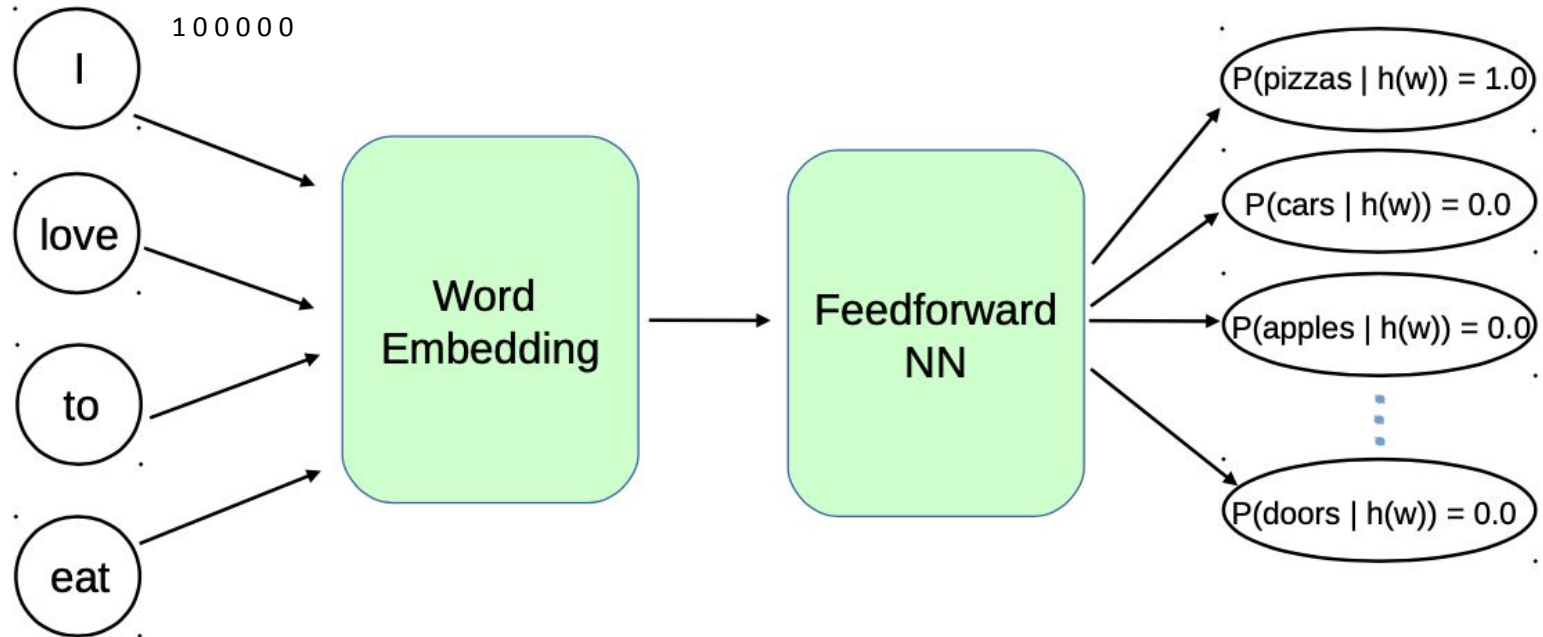
Idea clave: Palabras en contextos similares tendrán una representación similar.

Vectorización de texto

$x = \text{"I love to eat"} \rightarrow \text{CONTEXTO } h(w)$

$y = \text{"pizzas"} \rightarrow \text{SIGUIENTE PALABRA}$

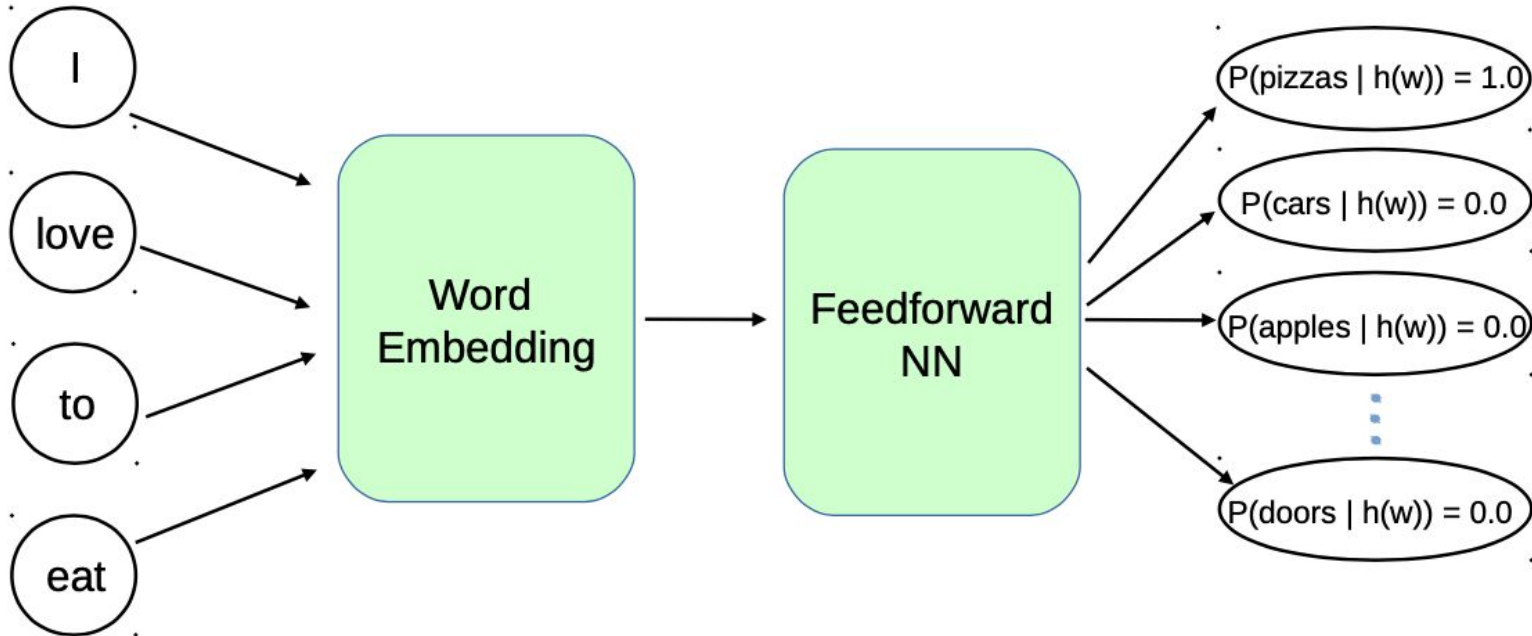
Tarea: predecir la palabra siguiente más probable de ocurrir dado un contexto.



Vectorización de texto

$x = \text{"I love to eat"} \rightarrow \text{CONTEXTO } h(w)$

$y = ? \rightarrow \text{SIGUIENTE PALABRA}$



Otras técnicas de Word Vectors

Otras técnicas para vectorizar palabras (word embeddings):

- GloVe
- FastText

DOC = “El bus rojo ”

El = [0.45, 0.66, 0.12 N=300]

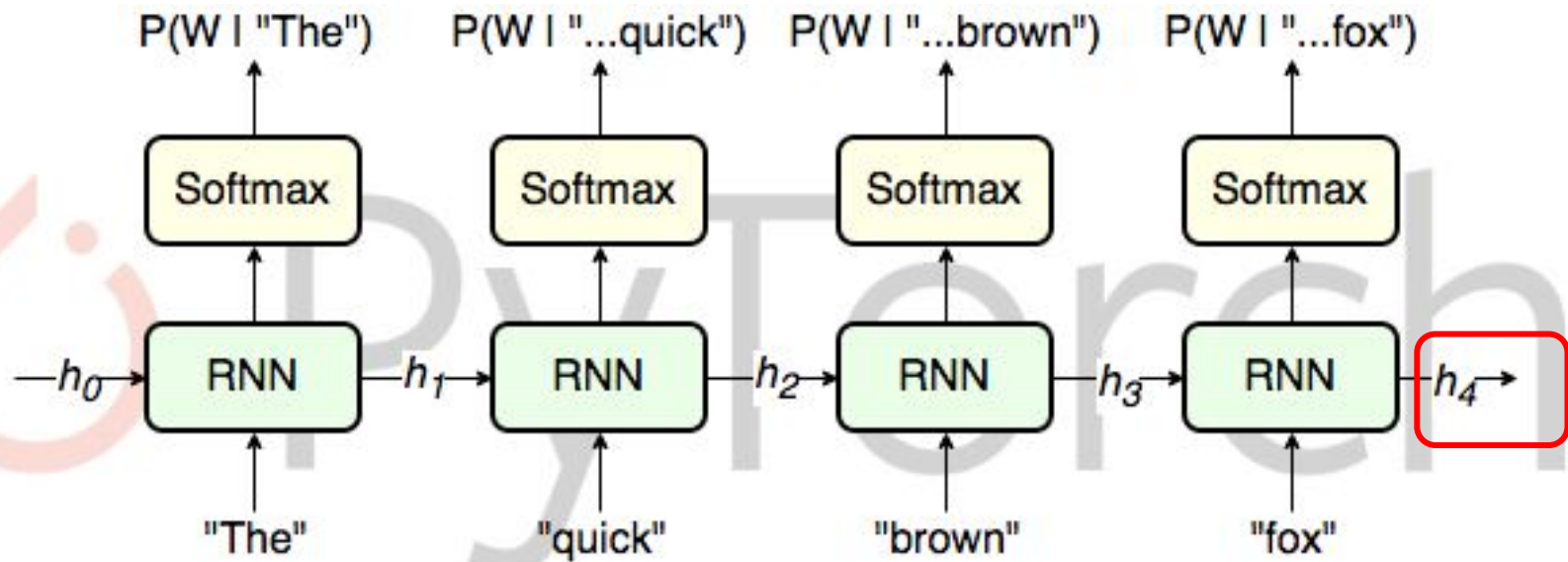
bus = [0.23, 0.34, 0.55 N = 300]

rojo = [0.46, 0.76 N = 300]

Limitaciones:

- Tenemos vectores por cada palabra pero necesitamos agregarlas para representar un texto.
- No sabe lidiar con **palabras que están fuera del vocabulario**.
- No escala a **nuevos idiomas** (esp, africano, frances) .

Red Neuronal Recurrente (RNN)



TASK: Predecir la palabra siguiente con mayor probabilidad

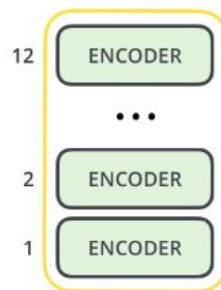
BERT



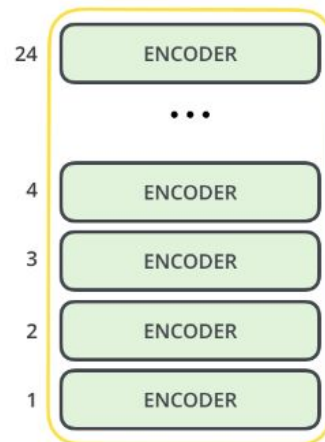
BERT_{BASE}



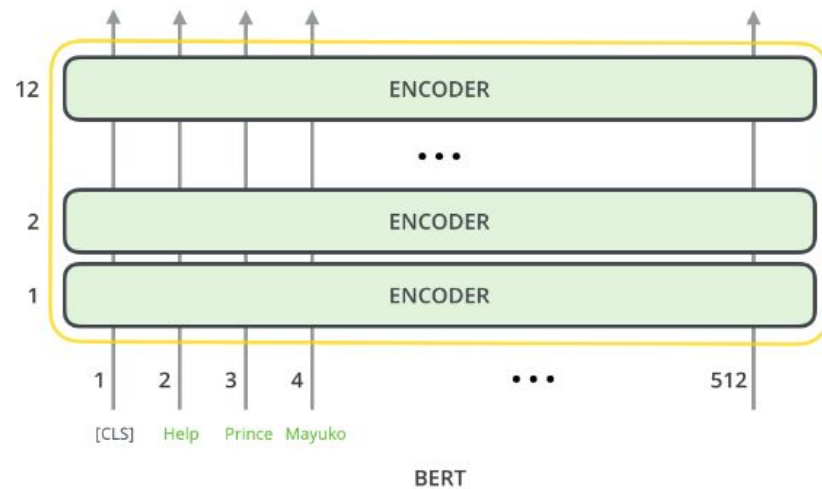
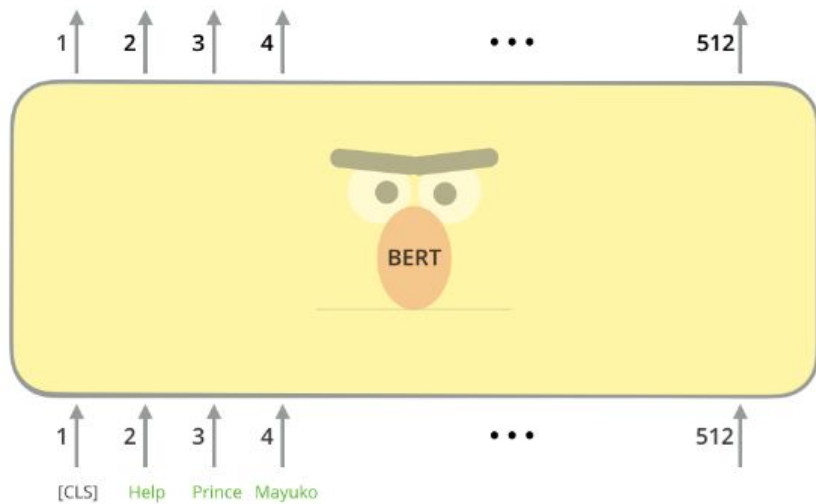
BERT_{LARGE}

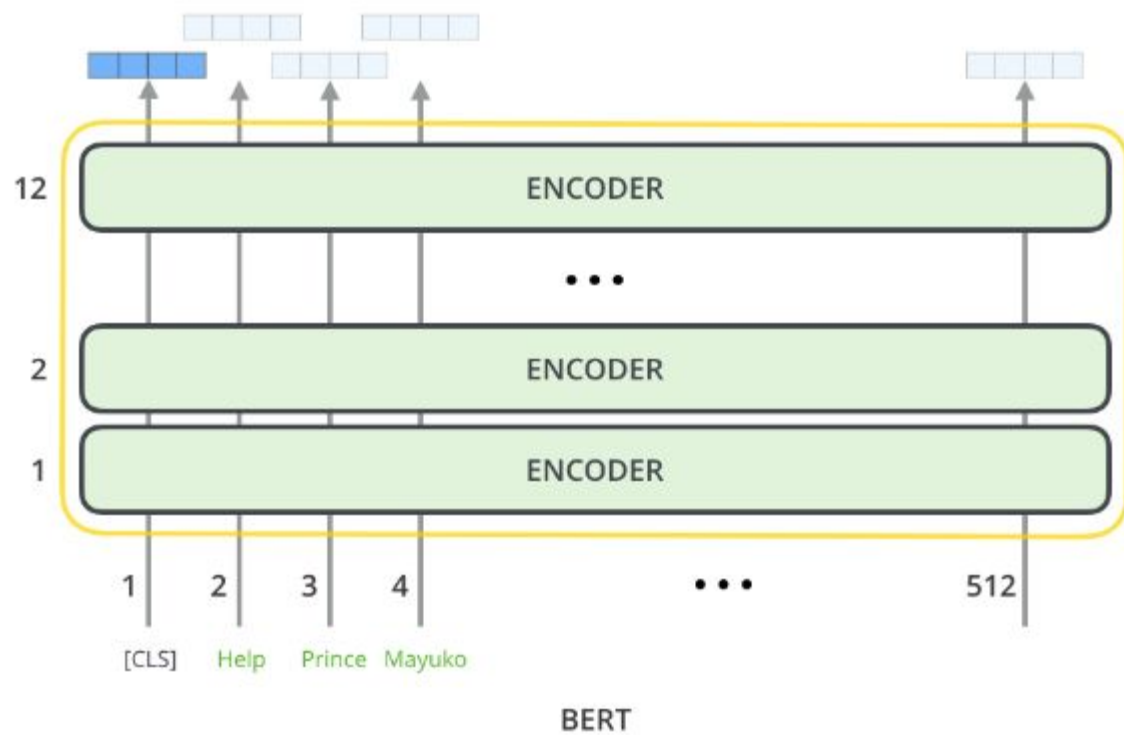


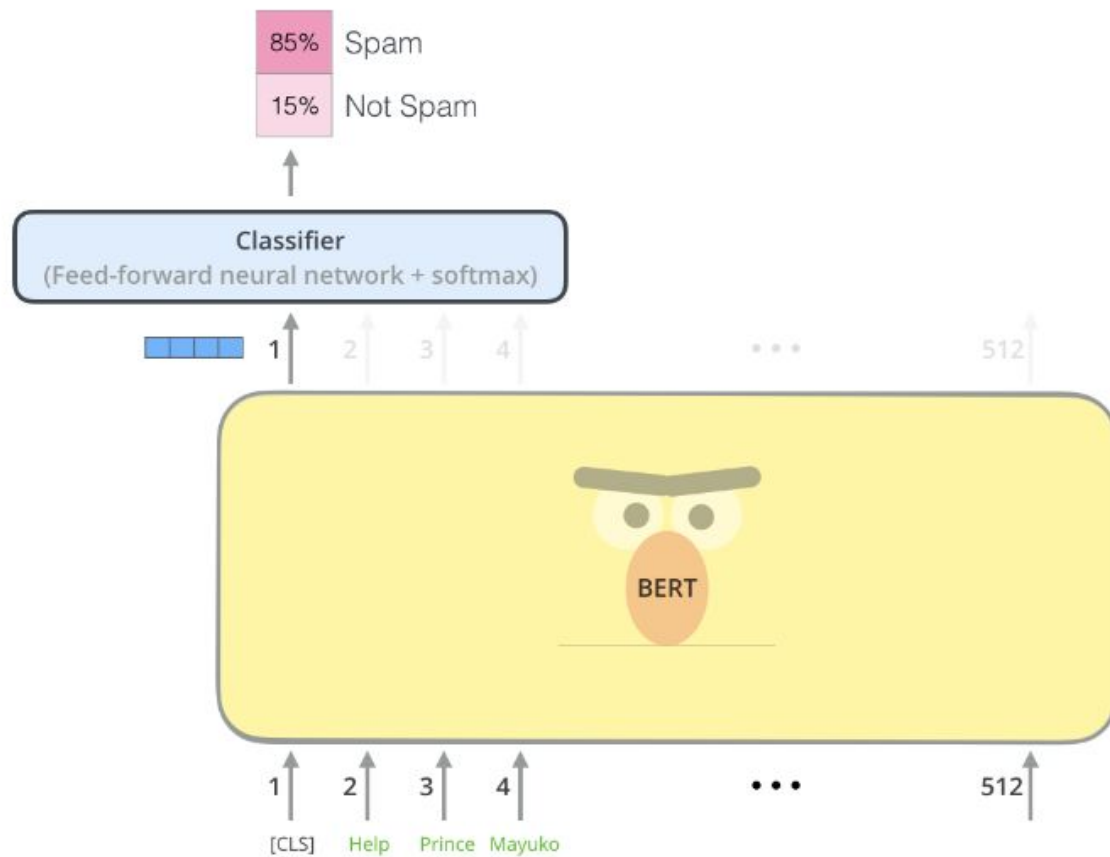
BERT_{BASE}



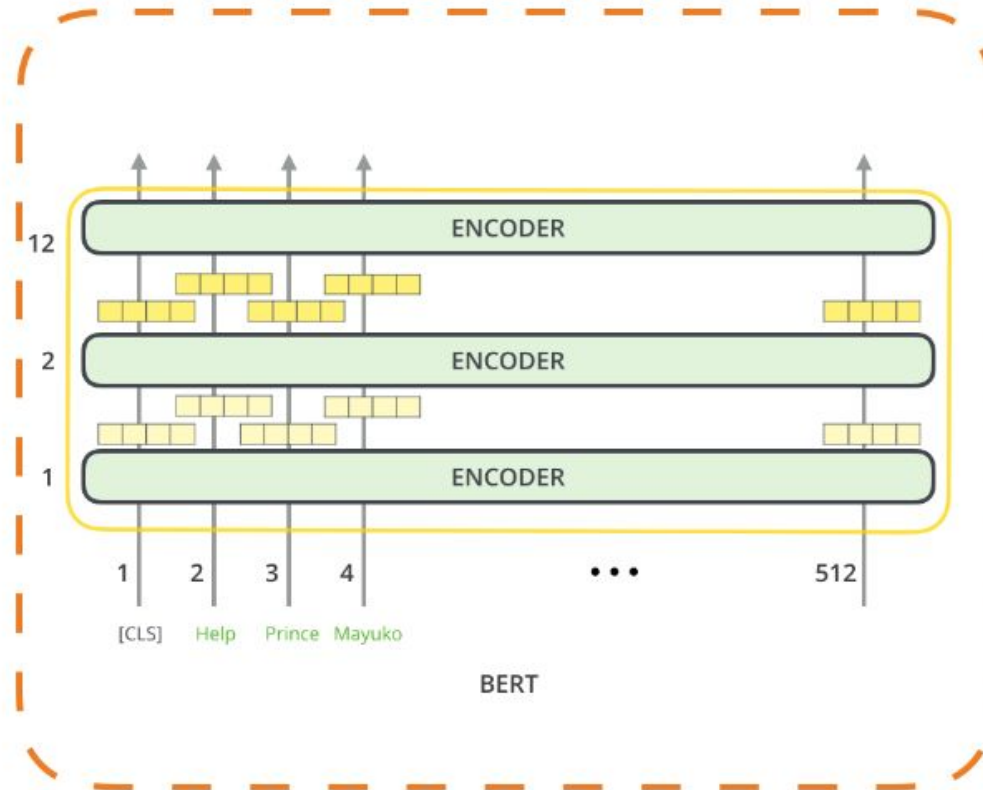
BERT_{LARGE}



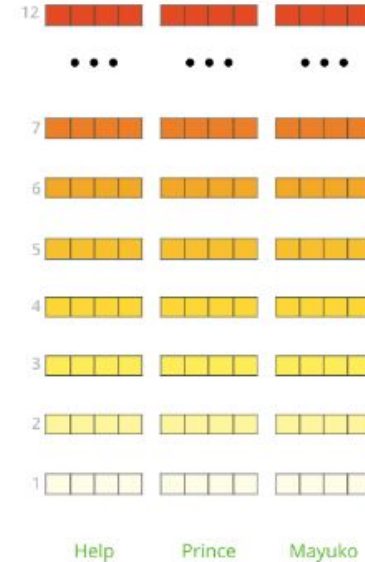




Generate Contextualized Embeddings

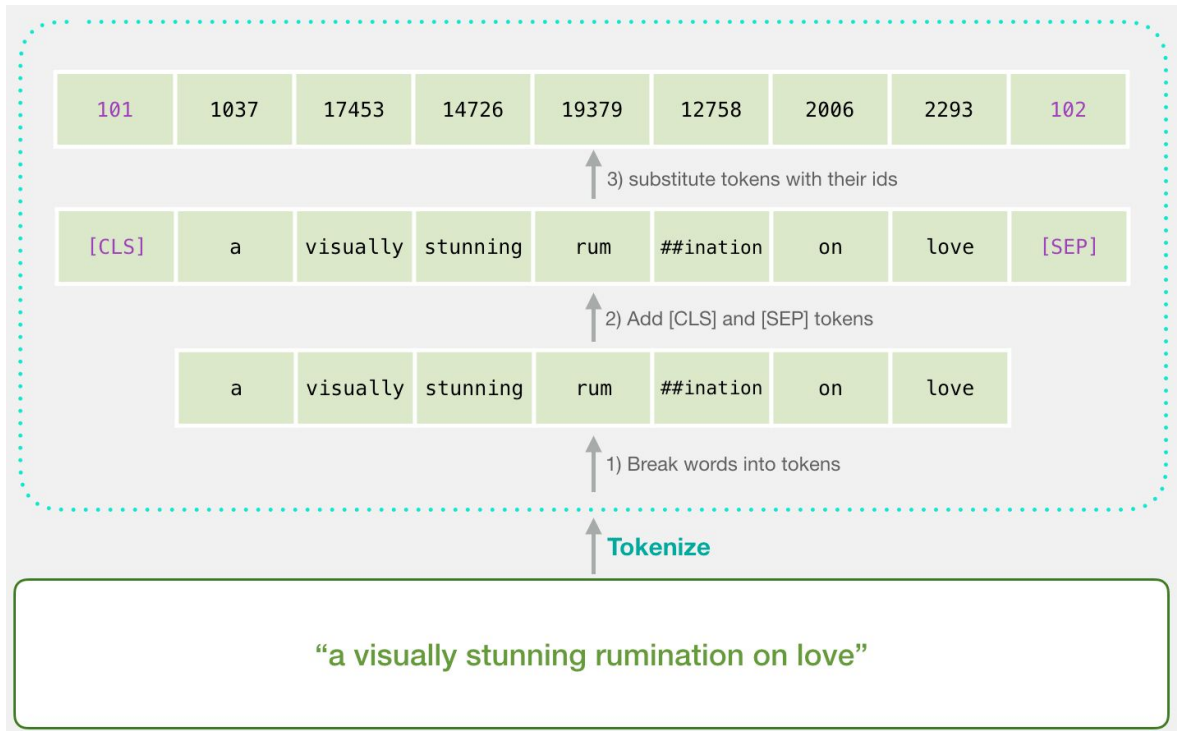


The output of each encoder layer along each token's path can be used as a feature representing that token.



But which one should we use?

BERT tokenizer

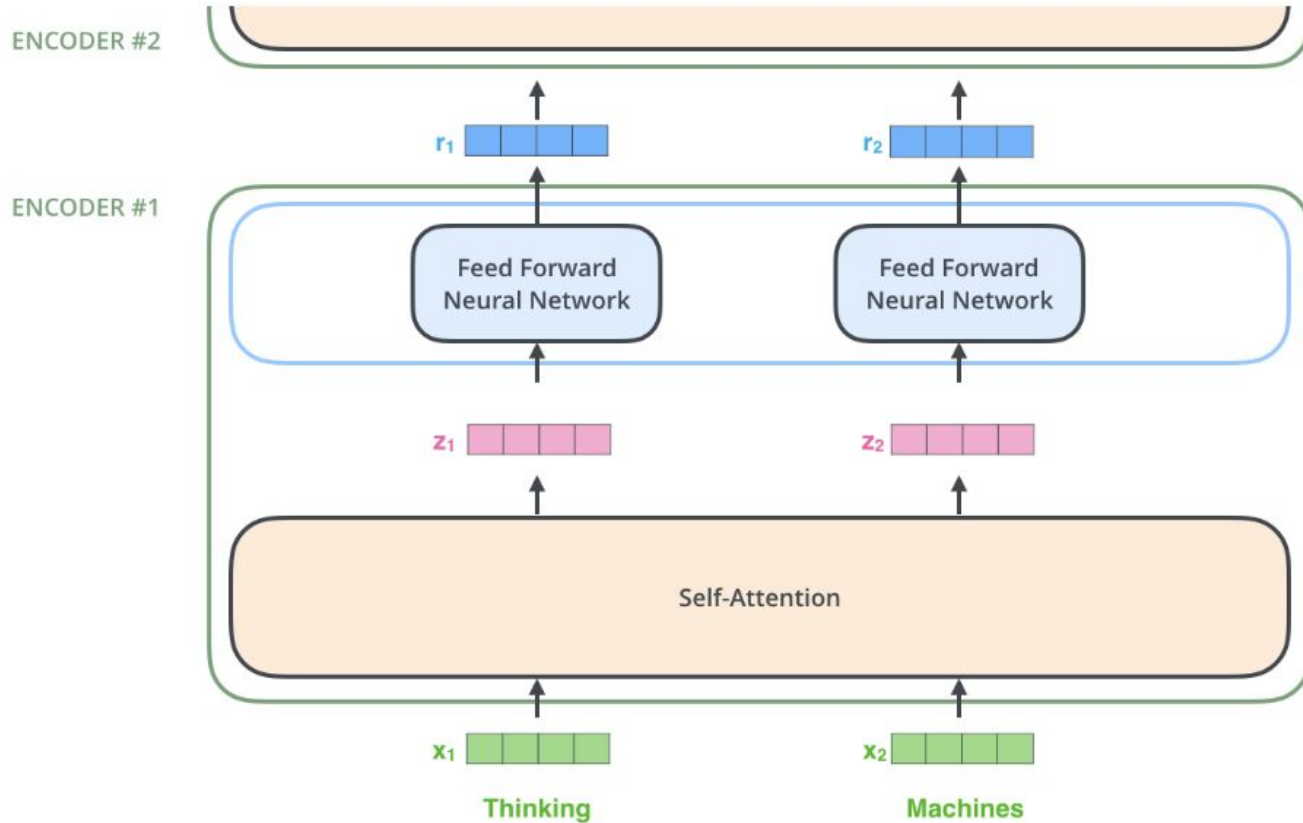


BERT trae incorporado un tokenizer que:

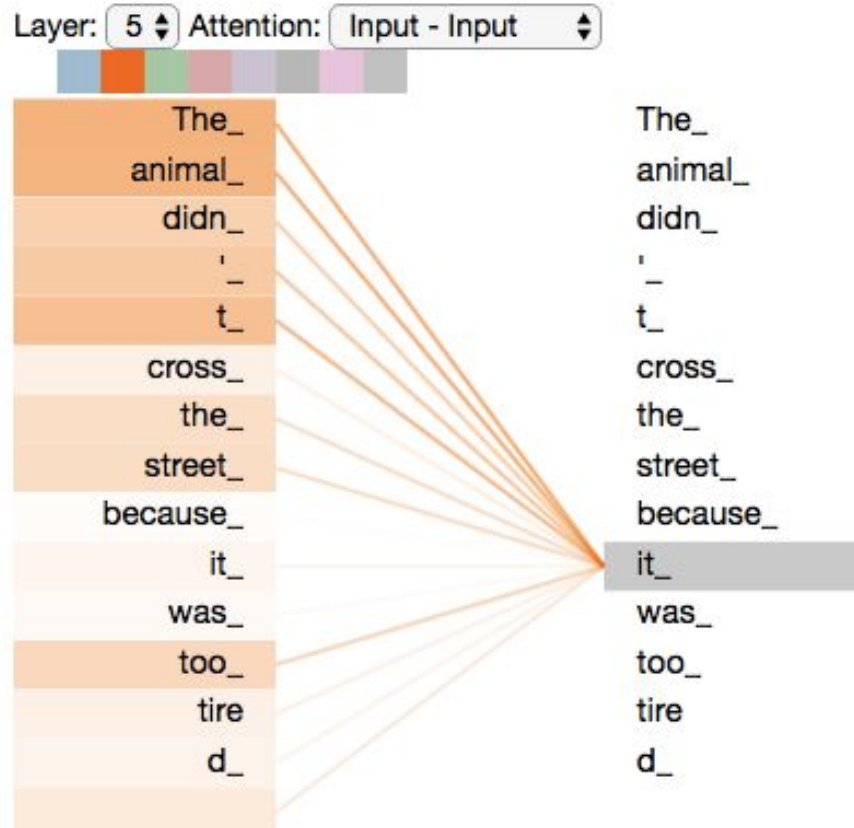
1. Divide palabras en dos partes de manera que la segunda se pueda utilizar de nuevo.
(ej. ama + #dos
da +
1. Agregar tokens especiales [CLS] que representa un texto completo y [SEP] para denotar separación si tiene más de una oración.
2. Convertir cada token al índice en el vocabulario.

¿Cómo funciona cada encoder?

¿Cómo funciona cada encoder?



Concepto de auto-atención (self attention)



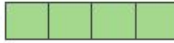
Cada palabra tiene un embedding y vectores Q , K y V.
Se aprenden modificando matrices de pesos.

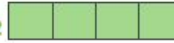
Input

Thinking

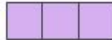
Machines

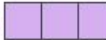
Embedding

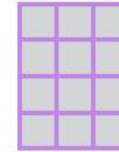
x_1 

x_2 

Queries

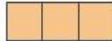
q_1 

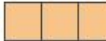
q_2 

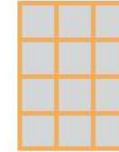


W^Q

Keys

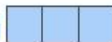
k_1 

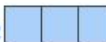
k_2 



W^K

Values

v_1 

v_2 



W^V

Multiplying x_1 by the W^Q weight matrix produces q_1 , the "query" vector associated with that word. We end up creating a "query", a "key", and a "value" projection of each word in the input sentence.

Input

Embedding

Queries

Keys

Values

Score

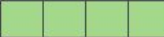
Divide by 8 ($\sqrt{d_k}$)

Softmax

Softmax
X
Value

Sum

Thinking

x_1 

q_1 

k_1 

v_1 

$q_1 \cdot k_1 = 112$

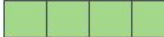
14

0.88

v_1 

z_1 

Machines

x_2 

q_2 

k_2 

v_2 

$q_1 \cdot k_2 = 96$

12

0.12

v_2 

z_2 

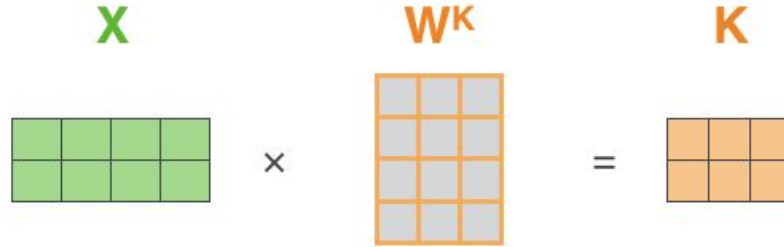
Cada palabra se
compara con
otras palabras de
una oración.

Objetivo.
Aprender
relaciones entre
palabras.

Para obtener
vectores **Q**, **K** y **V**

Se tiene que
aprender una
matriz de pesos
para cada una:

W_q
W_k
W_v



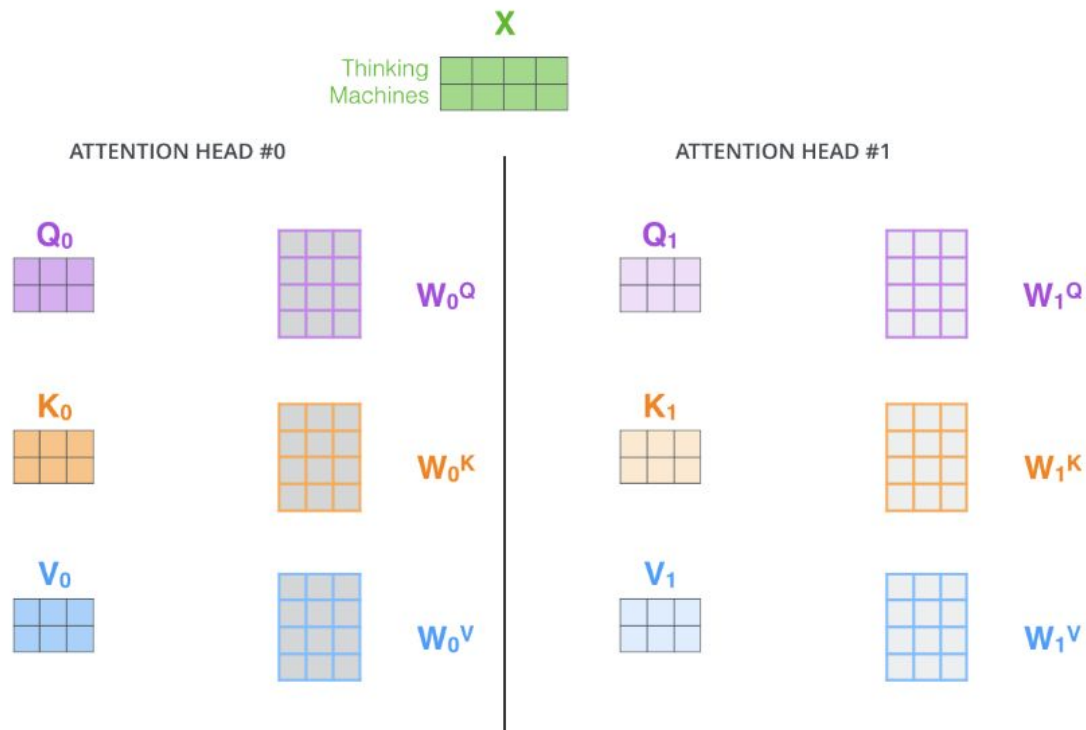
Cálculo del vector Z con matrices para paralelizar operaciones.

$$\text{softmax}\left(\frac{\begin{matrix} \text{Q} \\ \begin{array}{|c|c|c|} \hline \square & \square & \square \\ \hline \square & \square & \square \\ \hline \end{array} \end{matrix} \times \begin{matrix} \text{K}^T \\ \begin{array}{|c|c|} \hline \square & \square \\ \hline \square & \square \\ \hline \square & \square \\ \hline \end{array} \end{matrix}}{\sqrt{d_k}}\right) \begin{matrix} \text{V} \\ \begin{array}{|c|c|c|} \hline \square & \square & \square \\ \hline \square & \square & \square \\ \hline \end{array} \end{matrix}$$

$$= \begin{matrix} \text{Z} \\ \begin{array}{|c|c|c|} \hline \square & \square & \square \\ \hline \square & \square & \square \\ \hline \end{array} \end{matrix}$$

The self-attention calculation in matrix form


Repite el mismo proceso para múltiples cabezales (heads) , el N heads es un meta-parámetro



With multi-headed attention, we maintain separate Q/K/V weight matrices for each head resulting in different Q/K/V matrices. As we did before, we multiply X by the $WQ/WK/WV$ matrices to produce Q/K/V matrices.

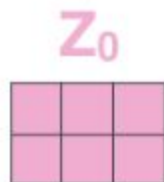


Calculating attention separately in
eight different attention heads

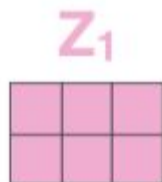


A diagram showing a downward-pointing arrow, indicating a process or flow from the input matrix to the attention heads.

ATTENTION
HEAD #0

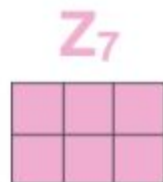


ATTENTION
HEAD #1



...

ATTENTION
HEAD #7



Esto se repite dependiendo de cuantos encoders
tenga el modelo....

Vectorización de texto: BERT (Transformer)

Limitaciones de BERT:

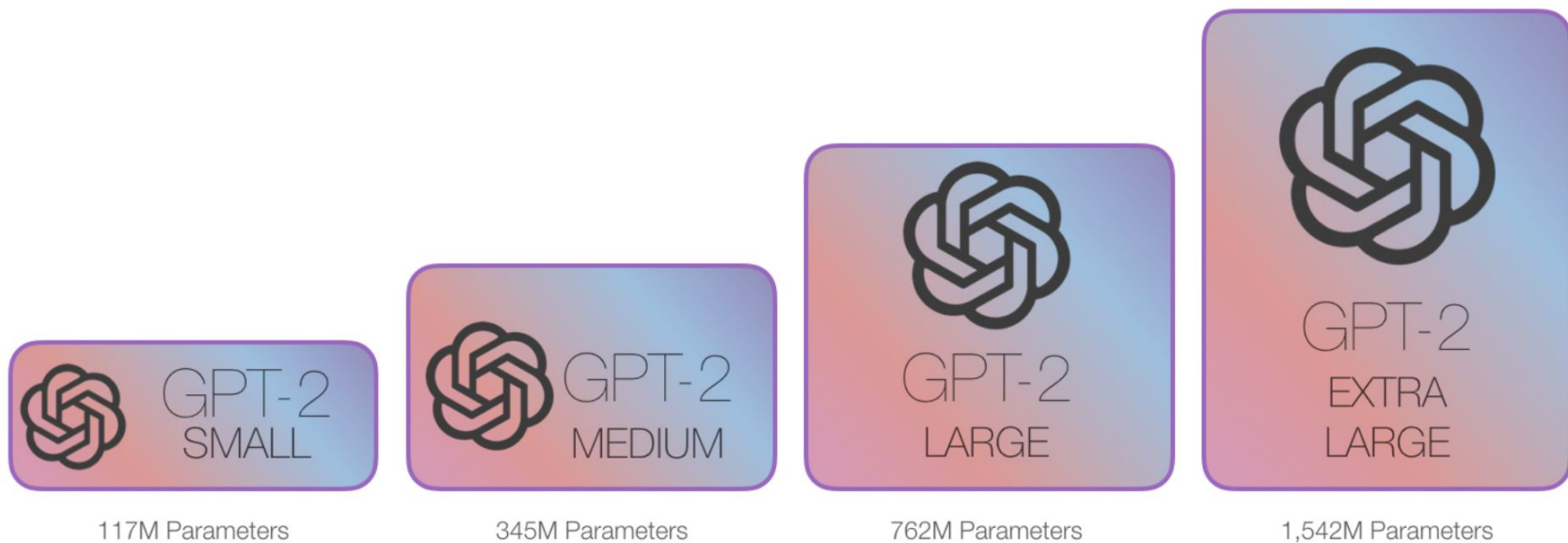
- Complejidad computacional para entrenar en nuevo corpus.
- Se limita largo de los textos a 512 tokens.

Alternativas:

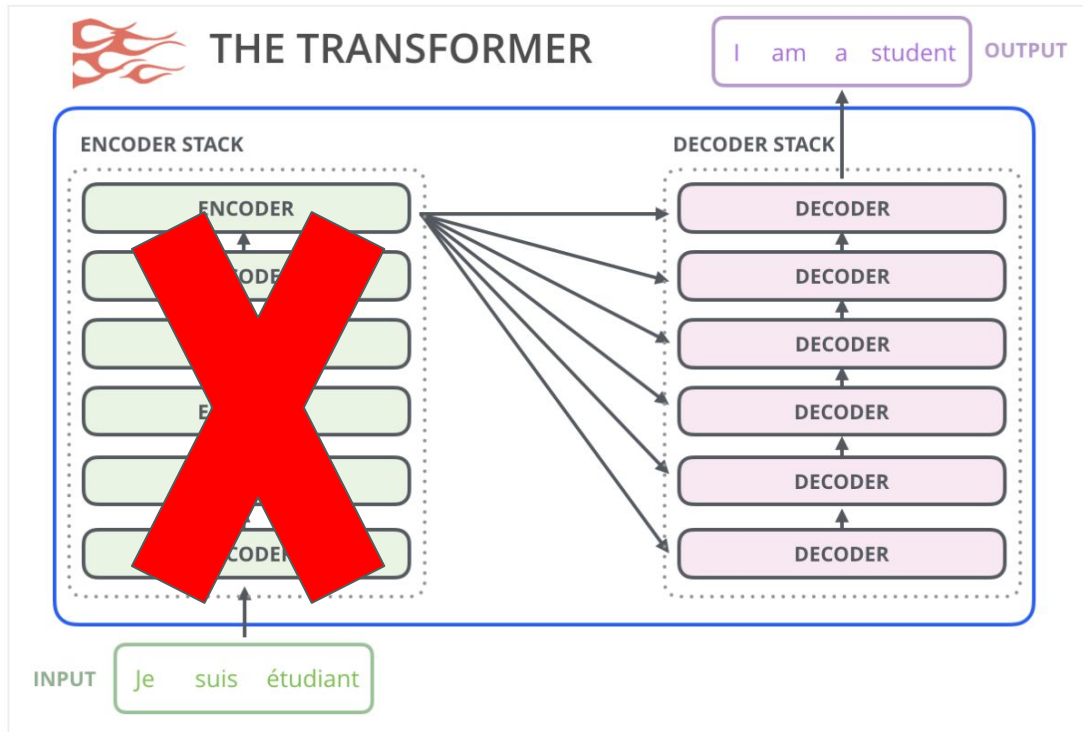
- XLNET
- RoBERTA
- Distill-BERT
- BigBird
- ULMFit
- GPT-2
- GPT-3

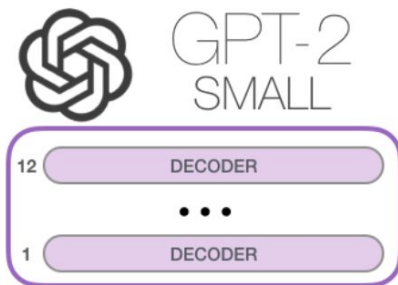
Modelos generativos

Modelos generativos de lenguaje



Transformer

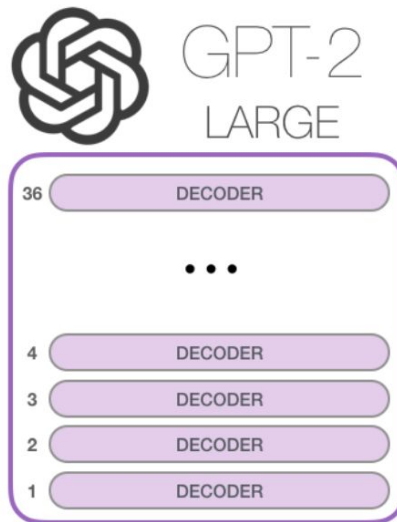




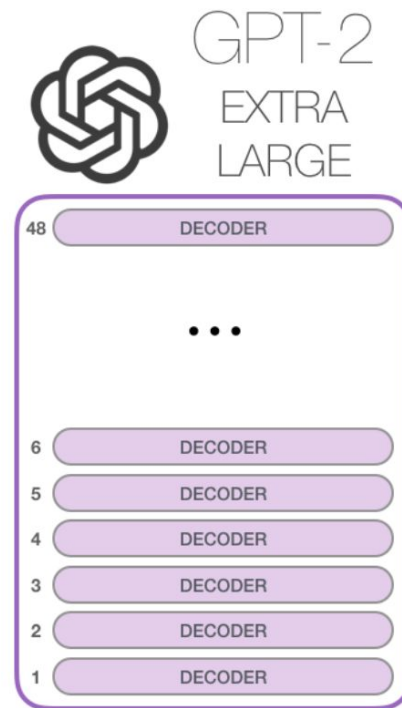
Model Dimensionality: 768



Model Dimensionality: 1024



Model Dimensionality: 1280



Model Dimensionality: 1600

Output

A	robot	may	not					
---	-------	-----	-----	--	--	--	--	--

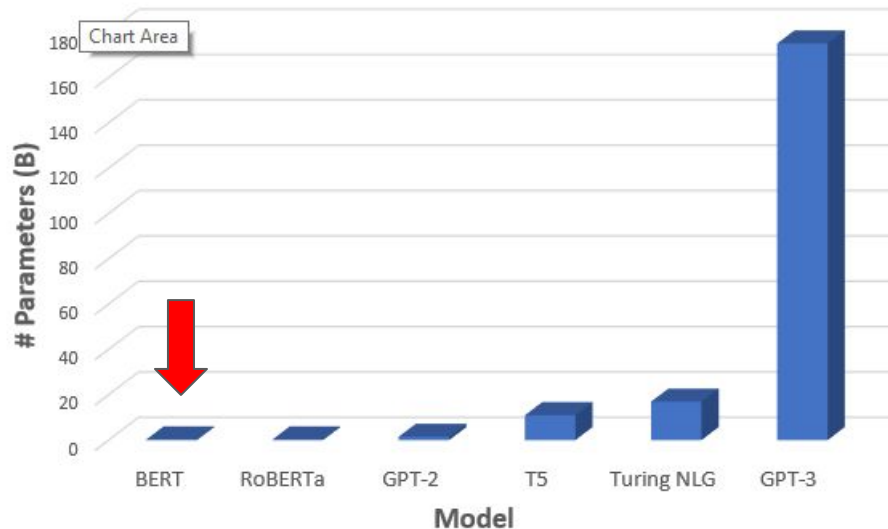


Input

recite	the	first	law	\$	A	robot	may	not						
--------	-----	-------	-----	----	---	-------	-----	-----	--	--	--	--	--	--

Input Sentence

El desempeño de modelos de lenguaje depende mucho de la cantidad de parámetros a entrenar.



Supera a todos los modelos anteriores en tareas de:

- Text Summarization
- Question Answering
- Language understanding

entre otros

3. Ejemplos de approach de recomendación basada en contenido

Embedding-based News Recommendation for Millions of Users

Shumpei Okura

Yahoo Japan Corporation

Tokyo, Japan

sokura@yahoo-corp.jp

Shingo Ono

Yahoo Japan Corporation

Tokyo, Japan

shiono@yahoo-corp.jp

Yukihiro Tagami

Yahoo Japan Corporation

Tokyo, Japan

yutagami@yahoo-corp.jp

Akira Tajima

Yahoo Japan Corporation

Tokyo, Japan

atajima@yahoo-corp.jp

Pasos

Distributed Representations of Articles

- Este paso implica generar una representación de los artículos para capturar sus características y rasgos.

Generación de User Representations

- Utiliza una Red Neuronal Recurrente (RNN) para este paso.
- Las secuencias de entrada para la RNN serán los historiales de navegación de los usuarios.
- El objetivo es crear una representación de los usuarios basada en su interacción con los artículos.

Matching y Listing de Artículos

- Realiza operaciones de matching y listing de artículos y usuarios basadas en operaciones de producto interno (inner-product operations).
- El objetivo es proporcionar a los usuarios artículos relevantes basados en sus preferencias e interacciones representadas.



Solving the Sparsity Problem in Recommendations via Cross-Domain Item Embedding Based on Co-Clustering

Yaqing Wang¹, Chunyan Feng^{1,2}, Caili Guo^{1,2}, Yunfei Chu^{1,2} and Jenq-Neng Hwang³

¹Beijing Key Laboratory of Network System Architecture and Convergence,
School of Information and Communication Engineering,

Beijing University of Posts and Telecommunications, Beijing, China

²Beijing Laboratory of Advanced Information Networks, Beijing, China

³Department of Electrical Engineering, University of Washington, Seattle, USA
{wangyq,cyfeng,guocaili,yfchu}@bupt.edu.cn,hwang@uw.edu

- Si un usuario escucha canciones de películas y luego pasa a ver películas relacionadas, hay una correlación entre los dominios de música y películas.
- Proponen utilizar información de diferentes dominios para aprender más sobre los intereses del usuario y generar mejores recomendaciones.
- El método identifica relaciones a nivel de clúster entre ítems de diferentes dominios, lo que ayuda a filtrar el ruido y a descubrir patrones útiles.

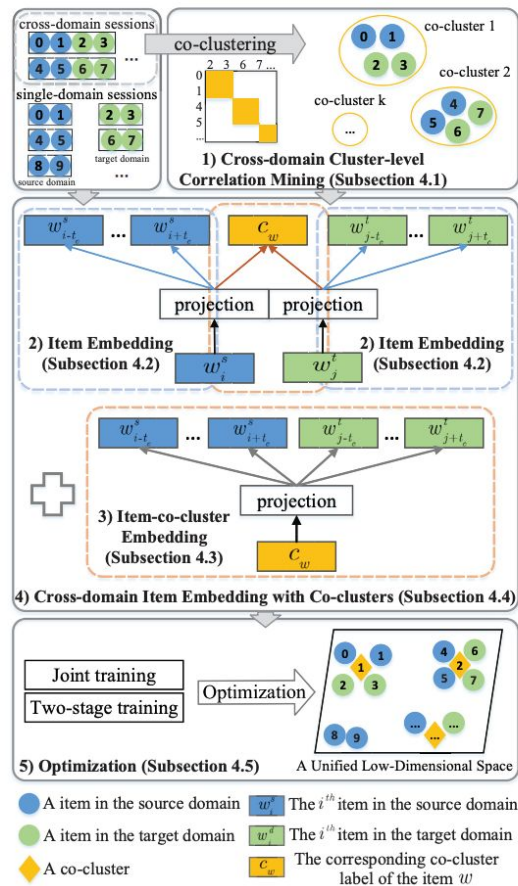


Figure 1: Illustration of the CDIE-C framework.

Gracias!