NLP

Otras arquitecturas para procesar secuencias: CNN y Attention

Dr. Rodrigo Cardenas Szigety
rodrigo.cardenas.sz@gmail.com

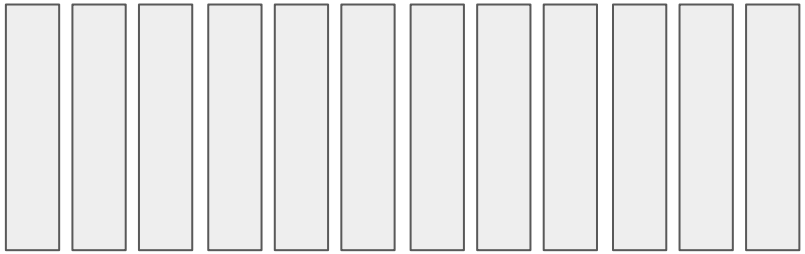
Dr. Nicolas Vattuone
nicolas.vattuone@gmail.com

Programa de la materia

Clase 1: Introducción a NLP, Vectorización de documentos.
Clase 2: Preprocesamiento de texto, librerías de NLP y bots de información.
Clase 3: Word Embeddings, CBOW y SkipGRAM, entrenamiento de embeddings.
Clase 4: Redes recurrentes (RNN), problemas de secuencia y estimación de próxima palabra.
Clase 5: Redes LSTM, análisis de sentimientos.
Clase 6: Modelos Seq2Seq, traductores y bots conversacionales.
Clase 7: Celdas con Attention. Transformers, BERT & ELMo, fine tuning.
Clase 8: Cierre del curso, NLP hoy y futuro, deploy.

*Unidades con desafíos a presentar al finalizar el curso.
*Último desafío y cierre del contenido práctico del curso.

De texto a secuencias de embeddings



Capa de embeddings

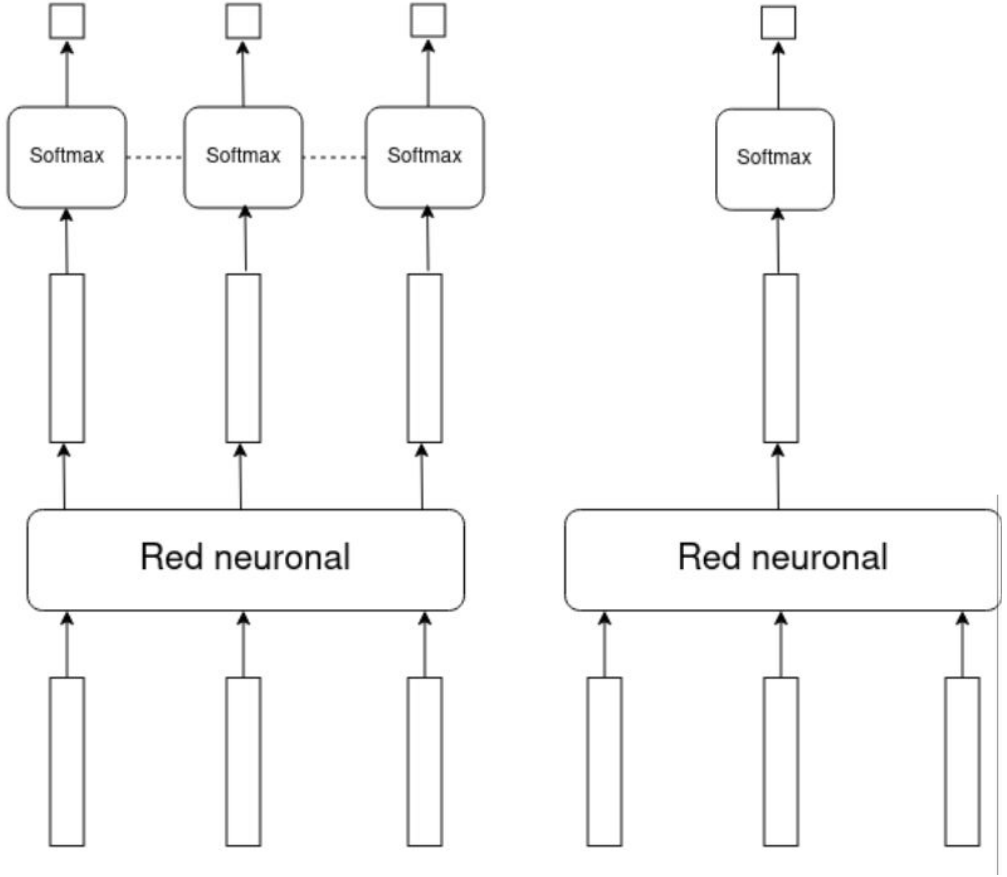
[40, 1213, 1215, 20, 578, 11, 120, 79, 31, 26, 608, 974]

Segmentación, normalización, transformación, filtrado y tokenización.

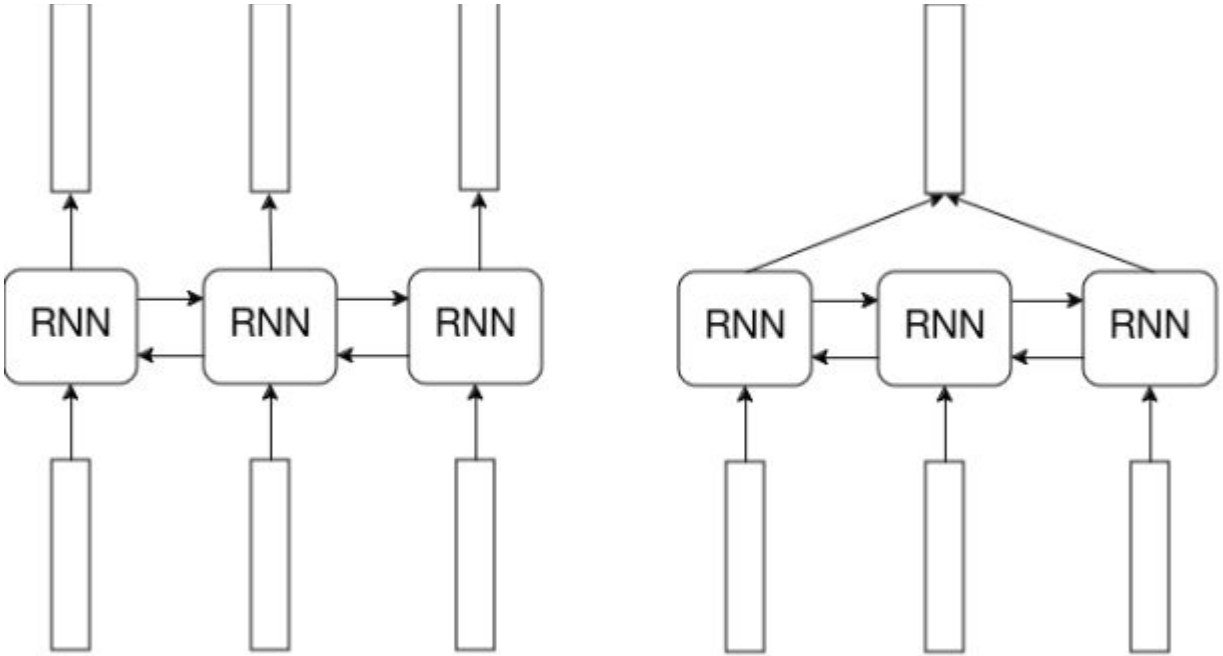
"Cuando Gregorio Samsa se despertó una mañana después de un sueño intranquilo"

Problemas de clasificación en texto a partir de secuencias

- Clasificación en tópicos
- Sentiment analysis
- Modelos de lenguaje
- NER



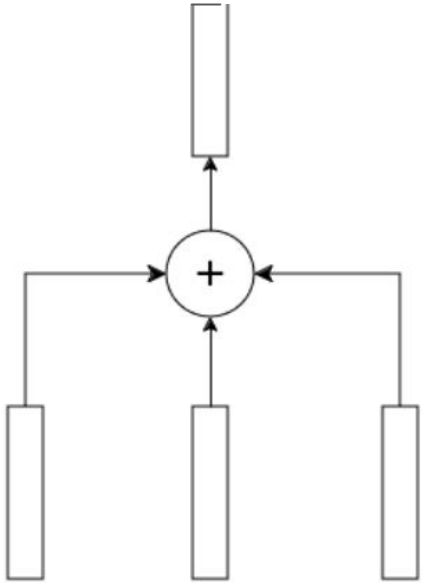
Procesando secuencias con RNNs



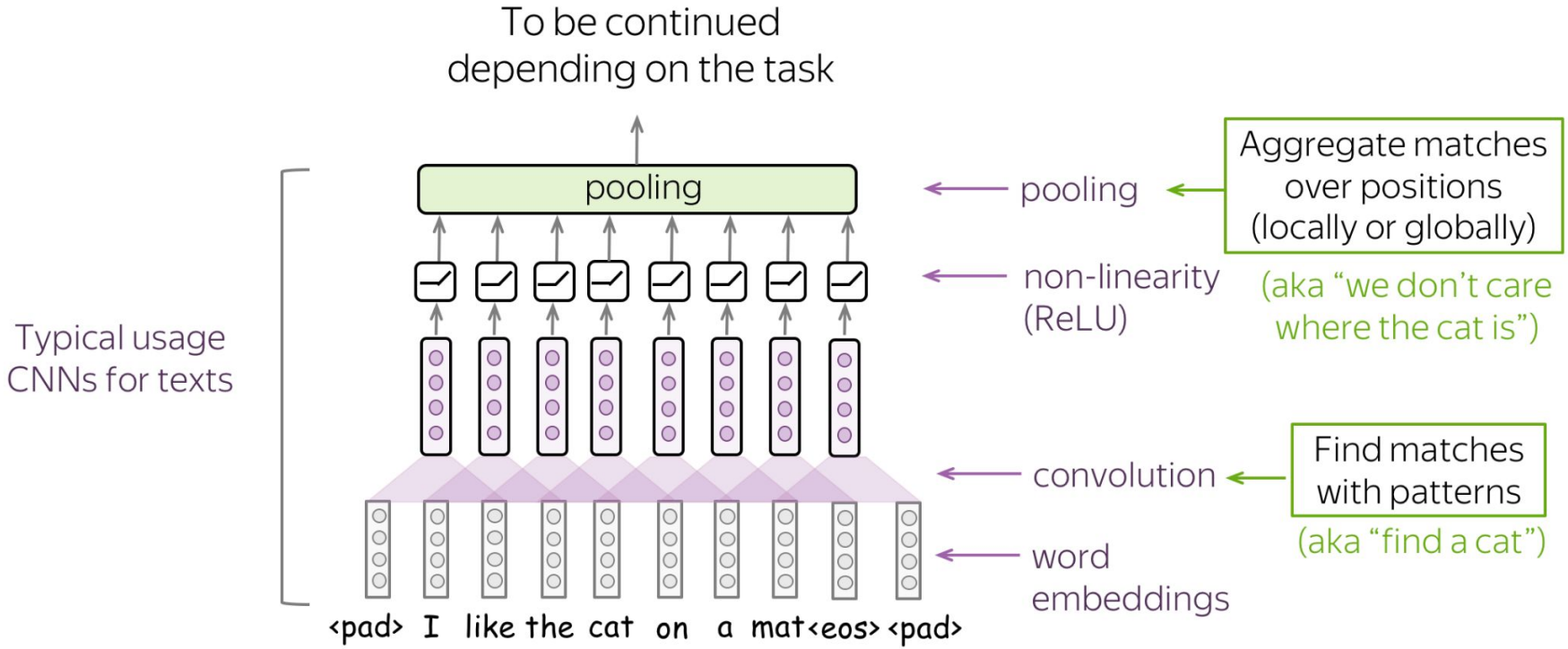
Modelo básico de clasificación a partir de embeddings

Una primera aproximación posible:

Sumar los embeddings



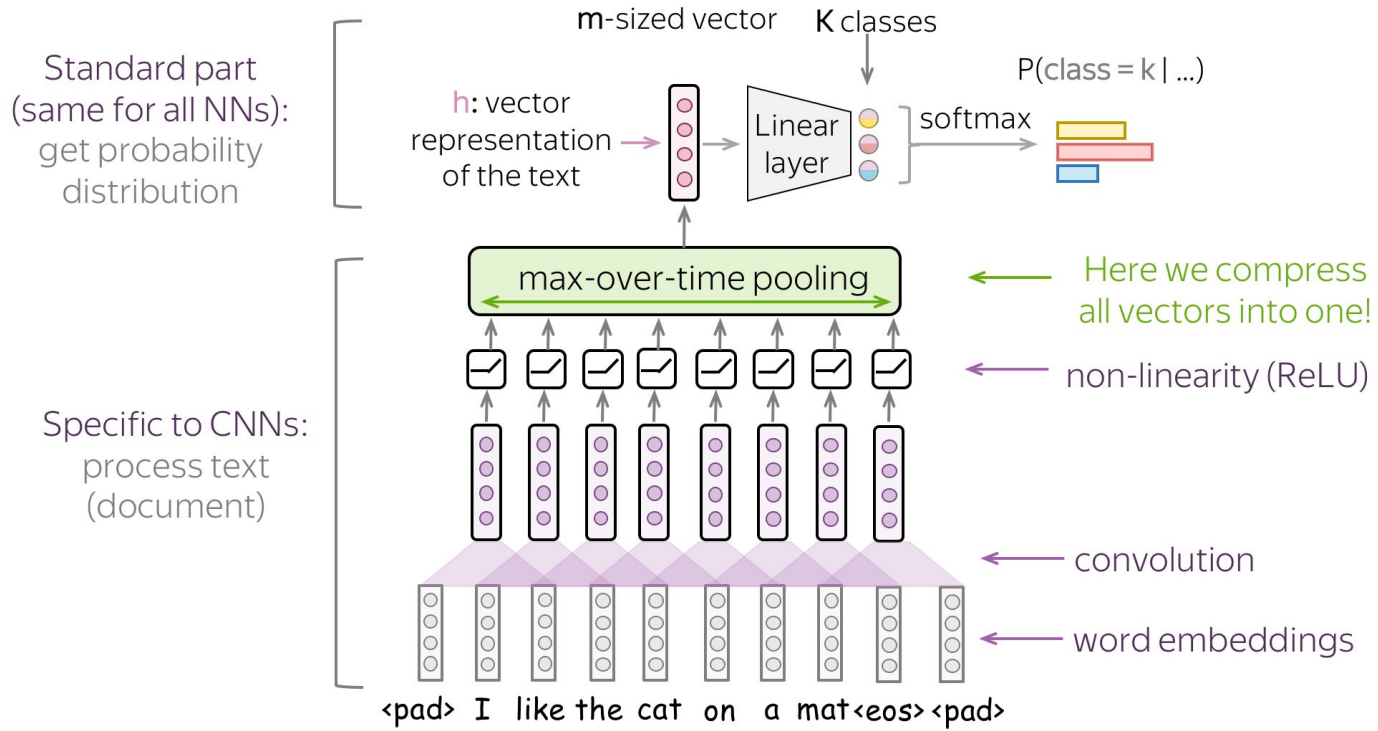
Arquitecturas Convolucionales para procesar secuencias



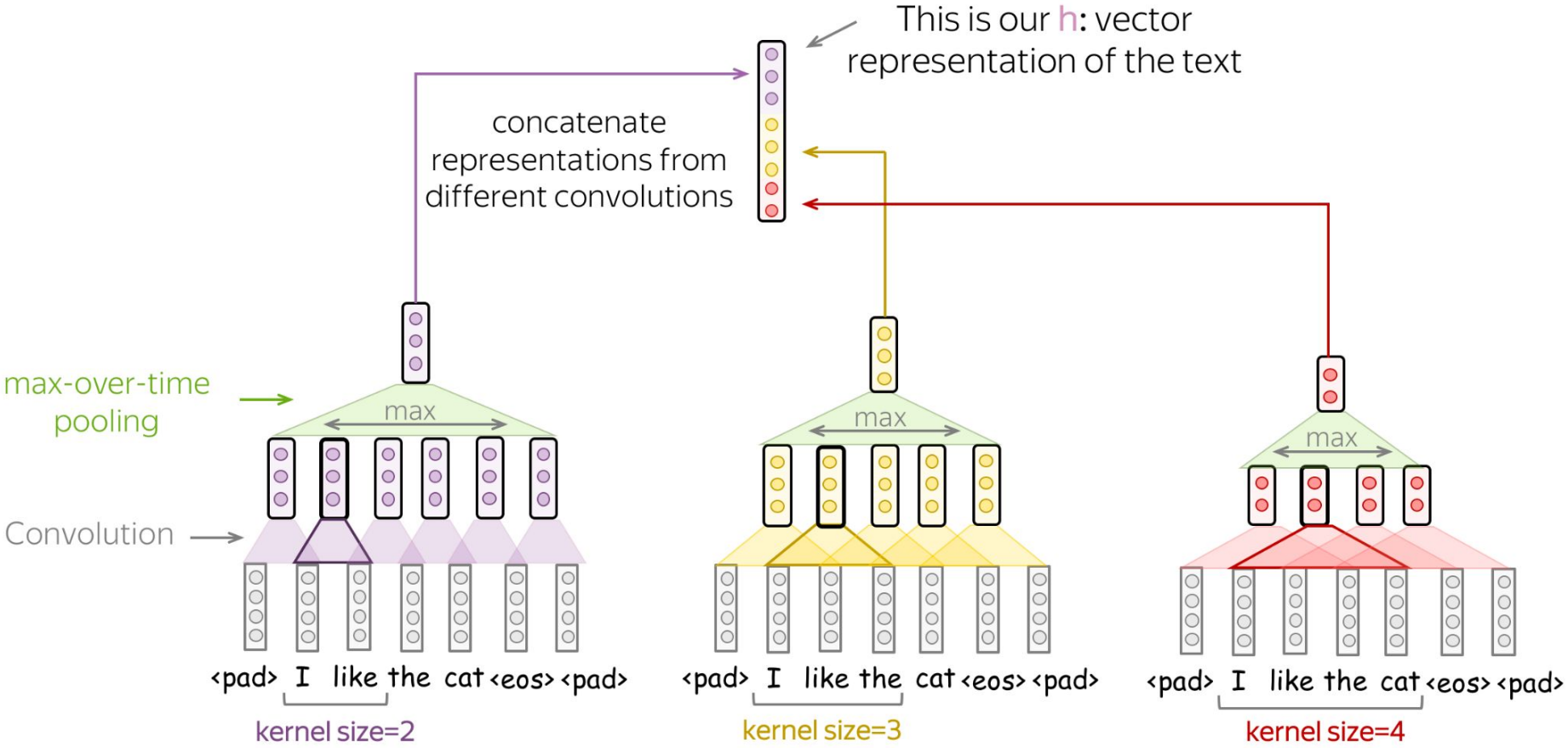
Dimensión de embedding->cantidad de canales

Tamaño de contexto ->tamaño de filtro

Clasificación de textos con CNN



TextCNN



https://arxiv.org/abs/1408.5882
Paper 2014

Mecanismo de atención

En general, un mecanismo de atención es una transformación de secuencias de embeddings a secuencias de embeddings de forma ponderada y paralela.

Existen varios tipos pero las operaciones fundamentales consisten en:

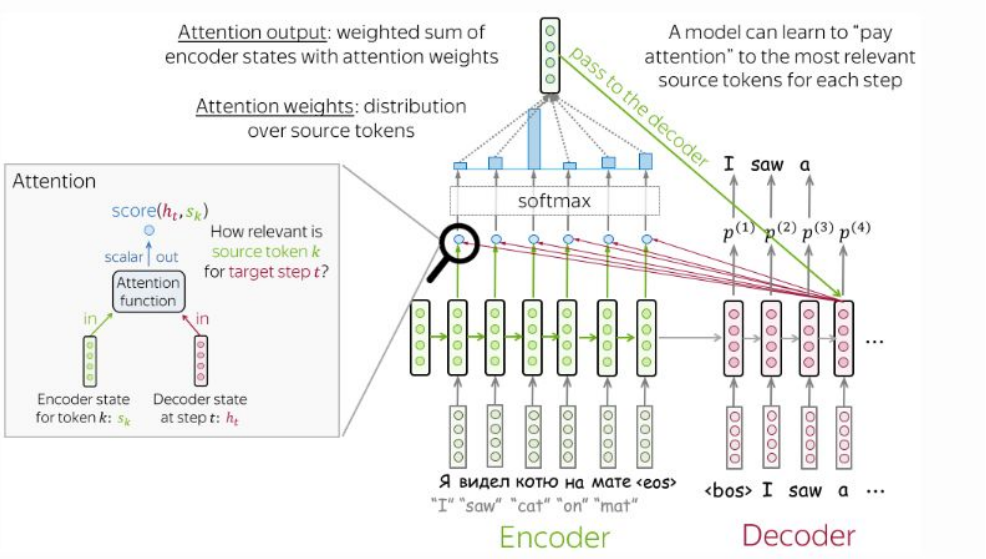
- *El cálculo un vector de pesos/scores de atención .*
- *La construcción de un vector ponderado que es fácilmente paralelizable a lo largo del tamaño de la secuencia.*

¡La degradación del gradiente es independiente del tamaño de la secuencia! Pero se debe fijar el tamaño máximo de secuencia a procesar.

*A futuro veremos que el mecanismo de **self-attention** es el que utilizan las arquitecturas **transformer**.*

Mecanismos de atención

Atención para Encoding-decoding



Self-attention (transformers)

Each vector receives three representations ("roles")

W_Q × = Query: vector **from** which the attention is looking

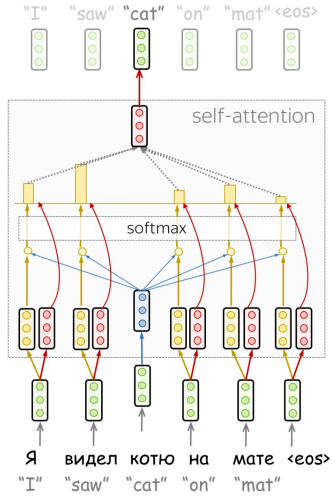
"Hey there, do you have this information?"

W_K × = Key: vector **at** which the query looks to compute weights

"Hi, I have this information – give me a large weight!"

W_V × = Value: their weighted sum is attention output

"Here's the information I have!"



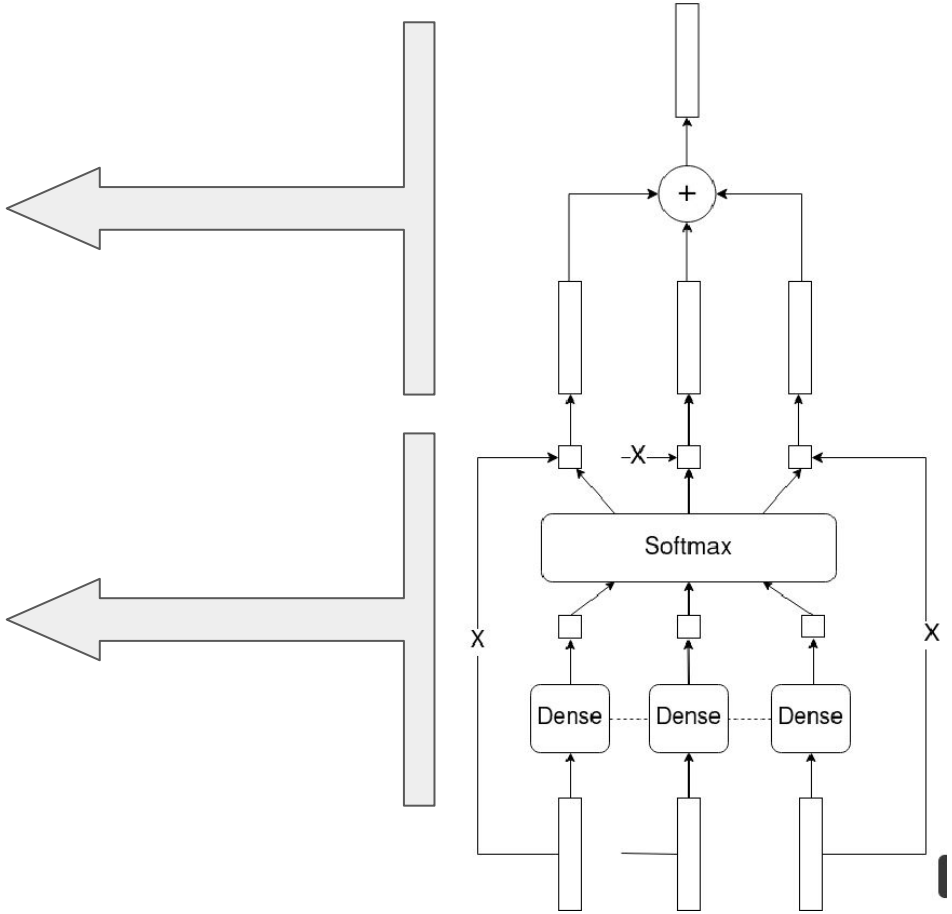
https://lena-voita.github.io/resources/lectures/seq2seq/transformer/encoder_self_attention.mp4

Attention via Feed-Forward Networks

Cálculo de vector ponderado, paralelizable en el tamaño de la secuencia

Cálculo de vector de pesos

¡Es BOW!



Attention via Recurrent Networks

Cálculo de vector ponderado, paralelizable en el tamaño de la secuencia

Cálculo de vector de pesos

¡Es recurrente!