Día 09 · Tema 3 — Self-attention, multi-head y el bloque Transformer

Eduardo C. Garrido-Merchán · ecgarrido@comillas.edu Universidad Pontificia Comillas · ICADE

1. Self-attention en cinco pasos

Dada una secuencia de T embeddings $X \in \mathbb{R}^{T\times d}$, la self-attention produce una nueva representación $Z \in \mathbb{R}^{T\times d}$ en cinco operaciones:

$$\begin{aligned} \text{(1) Proyectar:} \quad & Q = XW^Q, \;\; K = XW^K, \;\; V = XW^V, \\ \text{(2) Scores:} \quad & S = QK^\top/\sqrt{d_k}, \\ \text{(3) Softmax:} \quad & A = \mathrm{softmax}(S) \quad \text{(por filas)}, \\ \text{(4) Mezcla:} \quad & Z' = AV, \\ \text{(5) Proyección final:} \quad & Z = Z'W^O. \end{aligned} \tag{1}$$

$X \in \mathbb{R}^{T\times d}$	embeddings de entrada; T tokens, d dimensión
$W^Q, W^K \in \mathbb{R}^{d\times d_k}$	matrices de proyección a query/key
$W^V \in \mathbb{R}^{d\times d_v}$	matriz de proyección a value
$S \in \mathbb{R}^{T\times T}$	matriz de scores; S_{ij} es cuánto i atiende a j
A	pesos de atención normalizados, filas suman 1
W^O	proyección de salida, devuelve a dimensión d

2. Por qué la atención no necesita memoria explícita

A diferencia de la RNN, no hay estado oculto que mantener: cada posición i *accede directamente* a todas las posiciones j a través de A_{ij}. La información de un token lejano viaja en *un paso* en lugar de T pasos secuenciales. La memoria es la propia matriz A.

3. Multi-head attention

Una sola matriz A obliga a la red a comprometerse con un único tipo de relación entre tokens (sintáctica o semántica o de coreferencia). El *multi-head* ejecuta H atenciones en paralelo, cada una con sus propias matrices, y concatena:

$$\mathrm{MHA}(X) \;=\; \mathrm{Concat}\big(\mathrm{head}_1, \ldots, \mathrm{head}_H\big)\, W^O, \qquad \mathrm{head}_h = \mathrm{Att}(X\,W_h^Q,\, X\,W_h^K,\, X\,W_h^V). \tag{2}$$

H	número de cabezas (típicamente 8, 12, 32)
$d_k = d/H$	dimensión por cabeza
W_h^Q, W_h^K, W_h^V	proyecciones específicas de la cabeza h

Cada cabeza puede aprender una relación distinta: una atiende a la palabra siguiente, otra al sujeto del verbo, otra a la entidad de referencia. La expresividad sale gratis: el coste total de las H cabezas es el mismo que una atención de dimensión d.

4. Un bloque Transformer completo

El bloque encoder de Vaswani et al. encadena MHA y MLP, cada uno con *residual connection* y LayerNorm (pre-norm es el estándar moderno):

$$\begin{aligned} X' &= X + \mathrm{MHA}\big(\mathrm{LN}(X)\big), \\ Z &= X' + \mathrm{MLP}\big(\mathrm{LN}(X')\big). \end{aligned} \tag{3}$$

LN	LayerNorm sobre la dimensión de *features*, no del batch
MLP	$\mathrm{Linear}(d, 4d) \to \mathrm{GELU} \to \mathrm{Linear}(4d, d)$
$+X$	conexión residual; conserva información y estabiliza gradientes

La regla de oro es $d_{\mathrm{MLP}} = 4d$. Apilar N bloques (en GPT-3: $N = 96$, $d = 12288$, $H = 96$) construye el Transformer completo.

5. Codificación posicional

La self-attention es *equivariante a permutaciones*: si reordenas los tokens, las salidas se reordenan idénticamente. Para preservar el orden se añade una codificación posicional $P \in \mathbb{R}^{T\times d}$:

$$\tilde{X} \;=\; X + P, \qquad P_{t,2i} = \sin\!\big(t/10000^{2i/d}\big),\;\; P_{t,2i+1} = \cos\!\big(t/10000^{2i/d}\big). \tag{4}$$

Las variantes modernas (RoPE, ALiBi) sustituyen la suma por una rotación o un sesgo lineal: mejor extrapolación a longitudes mayores que las vistas.

6. Encoders, decoders y encoder-decoder

Tres formas del Transformer, según la máscara aplicada a S antes del softmax:

- **Encoder** (BERT-style): atención bidireccional, sin máscara. A_{ij} libre. Casos: clasificación, embeddings, NER.
- **Decoder** (GPT-style): *causal mask*, $S_{ij} = -\infty$ para $j > i$. Predicción autoregresiva: el modelo no ve el futuro.
- **Encoder-decoder** (T5, BART): un encoder bidireccional alimenta a un decoder causal vía *cross-attention*, donde las queries vienen del decoder y los keys/values del encoder.

7. Por qué el Transformer dominó

Cuatro razones técnicas: **(1)** paraleliza en T (training ~10× más rápido que LSTM equivalente); **(2)** cada token accede al resto en distancia 1 (no se desvanecen las dependencias largas); **(3)** escala con datos y parámetros sin saturar (leyes de scaling de Kaplan et al. y Chinchilla); **(4)** arquitectura uniforme aplicable a texto, imagen (ViT), audio (Whisper) y proteínas (AlphaFold). El precio es el coste cuadrático $O(T^2)$, que el Día 10 ataca.