Universidad Nacional de Colombia

Facultad de Ingeniería
Departamento de Ingeniería Eléctrica y Electrónica

Arquitecturas de Deep Learning: Análisis Comparativo

Asignatura: Teoría de Aprendizaje de Máquinas
Tipo de trabajo: Infografía técnica

Estudiantes

Edwin García
Camilo Holguín
Federico Villa

Bogotá D.C., 4 de diciembre de 2025

0. Notación General y Núcleo Común

Notación compartida:

- l : índice de la capa, de 1 (primera capa oculta) a L (capa de salida).
- $\mathbf{x}$: vector de entrada a la red.
- $\mathbf{W}^{(l)}$: matriz de pesos de la capa l.
- $\mathbf{b}^{(l)}$: vector de sesgos (bias) de la capa l.
- $\mathbf{z}^{(l)}$: entrada neta (pre-activación) de la capa l.
- $\mathbf{a}^{(l)}$: activación (salida) de la capa l.
- $f(\cdot)$: función de activación (ReLU, sigmoid, tanh, etc.).
- $L(\cdot)$: función de pérdida (loss).

Forward básico en cualquier arquitectura densa:

$$\mathbf{z}^{(l)} = \mathbf{W}^{(l)}\mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}, \qquad \mathbf{a}^{(l)} = f\big(\mathbf{z}^{(l)}\big)$$

Idea del aprendizaje (Backpropagation):

- Se calcula la predicción $\hat{\mathbf{y}} = \mathbf{a}^{(L)}$.
- Se mide el error con $L(\hat{\mathbf{y}}, \mathbf{y})$.
- Se obtienen gradientes $\dfrac{\partial L}{\partial \mathbf{W}^{(l)}}$ y $\dfrac{\partial L}{\partial \mathbf{b}^{(l)}}$.
- Los parámetros se actualizan, por ejemplo con descenso por gradiente:

$$\mathbf{W}^{(l)} \leftarrow \mathbf{W}^{(l)} - \eta \frac{\partial L}{\partial \mathbf{W}^{(l)}}, \quad \mathbf{b}^{(l)} \leftarrow \mathbf{b}^{(l)} - \eta \frac{\partial L}{\partial \mathbf{b}^{(l)}}$$

1. Redes Densas (MLP)

Concepto general: Arquitectura totalmente conectada (*Fully Connected*) que no impone una estructura espacial o temporal específica sobre los datos. Es el bloque básico para clasificación y regresión.

Capas principales:

- Capa de entrada: vector de características $\mathbf{x}$.
- Capas ocultas densas: combinaciones afines + activación.
- Capa de salida: típicamente lineal, sigmoide o softmax.

(i) Modelo Matemático Fundamental: Para una capa densa:

$$\mathbf{z}^{(l)} = \mathbf{W}^{(l)}\mathbf{a}^{(l-1)} + \mathbf{b}^{(l)}, \qquad \mathbf{a}^{(l)} = f\big(\mathbf{z}^{(l)}\big)$$

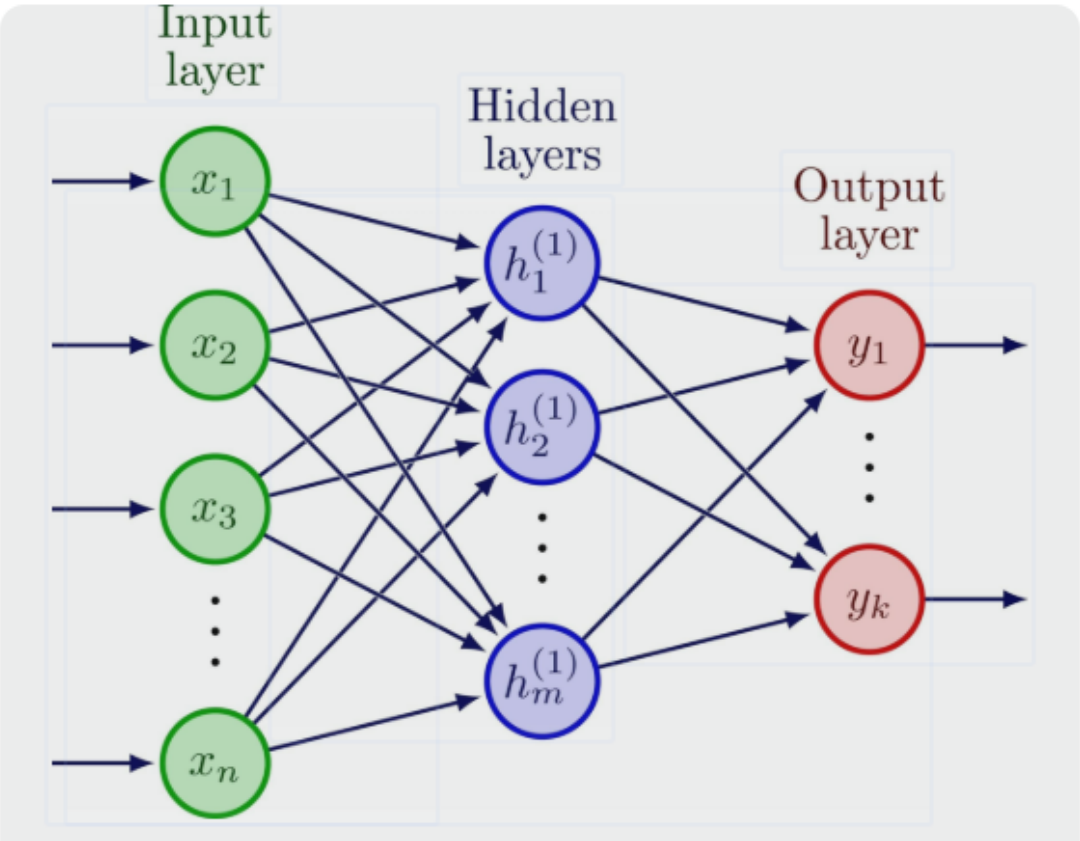
Donde:

- $\mathbf{a}^{(l-1)}$ es la activación de la capa anterior.
- $\mathbf{W}^{(l)} \in \mathbb{R}^{n_l \times n_{l-1}}$ es la matriz de pesos.
- $\mathbf{b}^{(l)} \in \mathbb{R}^{n_l}$ es el vector de sesgos.
- $f(\cdot)$ es la función de activación (ReLU, tanh, etc.).

(ii) Esquema ilustrativo:

$$\text{Input} \rightarrow \text{Hidden}_1 \rightarrow \text{Hidden}_2 \rightarrow \text{Output}$$

[Inserte diagrama de nodos totalmente conectados]



(iii) Aplicaciones representativas:

- Datos tabulares (predicción de demanda, scoring de riesgo, etc.).
- Regresión y clasificación genérica.
- Capas finales de muchas arquitecturas (CNN, RNN, Transformers).

2. Redes Convolucionales (CNN)

Concepto general: Introducen *conexiones locales* y *compartición de pesos* para explotar la estructura espacial (imágenes, mapas de características), con cierto grado de invarianza a traslaciones.

Capas principales:

- Capas convolucionales (Conv): aplican filtros locales.
- Capas de activación (ReLU, etc.).
- Capas de reducción (Pooling).
- Capas densas finales para decisión.

(i) Modelo Matemático Fundamental (Convolución 2D): Para un filtro k en la capa l:

$$z_{i,j,k}^{(l)} = b_k^{(l)} + \sum_{c=1}^{C_{l-1}} \sum_{m=1}^{F} \sum_{n=1}^{F} W_{m,n,c,k}^{(l)} \, a_{i+m,j+n,c}^{(l-1)}$$

Donde:

- $a_{i,j,c}^{(l-1)}$ es el valor de la activación en la posición (i, j) y canal c.
- $W_{m,n,c,k}^{(l)}$ es el peso del filtro k en la posición (m, n) y canal c.
- $F \times F$ es el tamaño espacial del filtro (ej. 3×3).
- C_{l-1} es el número de canales de entrada.
- $b_k^{(l)}$ es el sesgo asociado al filtro k.

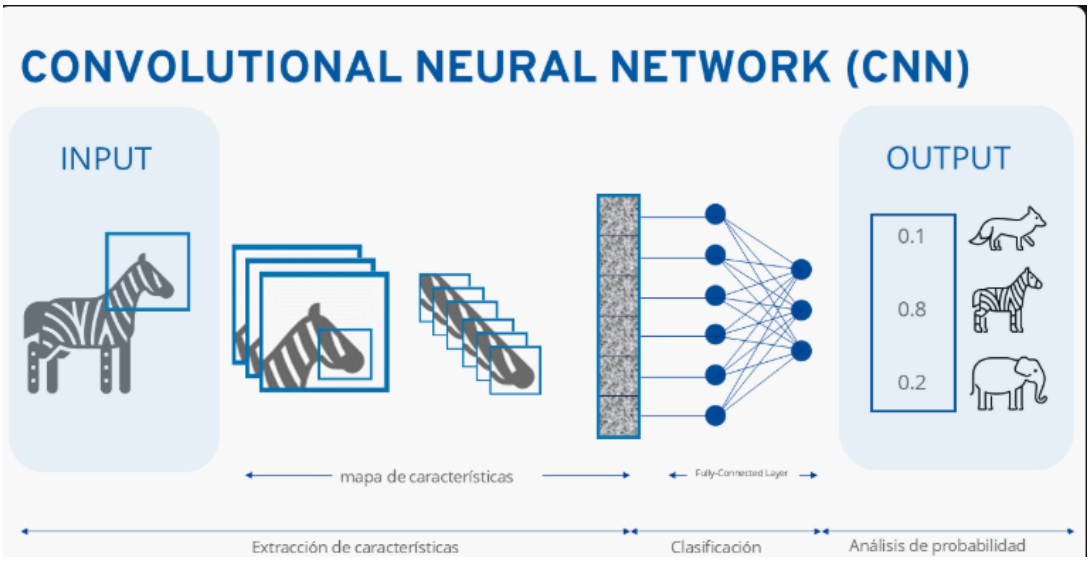
La activación resultante:

$$a_{i,j,k}^{(l)} = f(z_{i,j,k}^{(l)})$$

(ii) Esquema ilustrativo:

Input Image → [Conv + ReLU] → Pooling → FC → Output

[Inserte diagrama de kernel deslizante sobre imagen]



(iii) Aplicaciones representativas:

- Visión por computador (clasificación, detección de objetos, segmentación).
- Análisis médico de imágenes (rayos X, resonancias).
- Procesamiento de audio mediante espectrogramas.

3. Redes Recurrentes (RNN / LSTM)

Concepto general: Procesan secuencias conservando un *estado oculto* que resume información del pasado. Son adecuadas cuando el *orden temporal* importa.

Capas principales:

- Capa de entrada secuencial: $\{x_t\}_{t=1}^{T}$.
- Capa recurrente (RNN simple, LSTM o GRU).
- Capa de salida: puede emitir por paso temporal o solo al final.

(i) Modelo Matemático (RNN simple):

$$h_t = f\big(W_h h_{t-1} + W_x x_t + b\big)$$

Donde:

- x_t es la entrada en el tiempo t.
- h_{t-1} es el estado oculto en el tiempo anterior.
- W_x y W_h son matrices de pesos para entrada y estado.
- b es el vector de sesgos.
- $f(\cdot)$ suele ser tanh o ReLU.

Extensión: Célula LSTM (manejo de memoria): *Puerta de olvido:*

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

Puerta de entrada y contenido candidato:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \qquad \tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C)$$

Actualización del estado de celda:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

Puerta de salida y estado oculto:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \qquad h_t = o_t \odot \tanh(C_t)$$

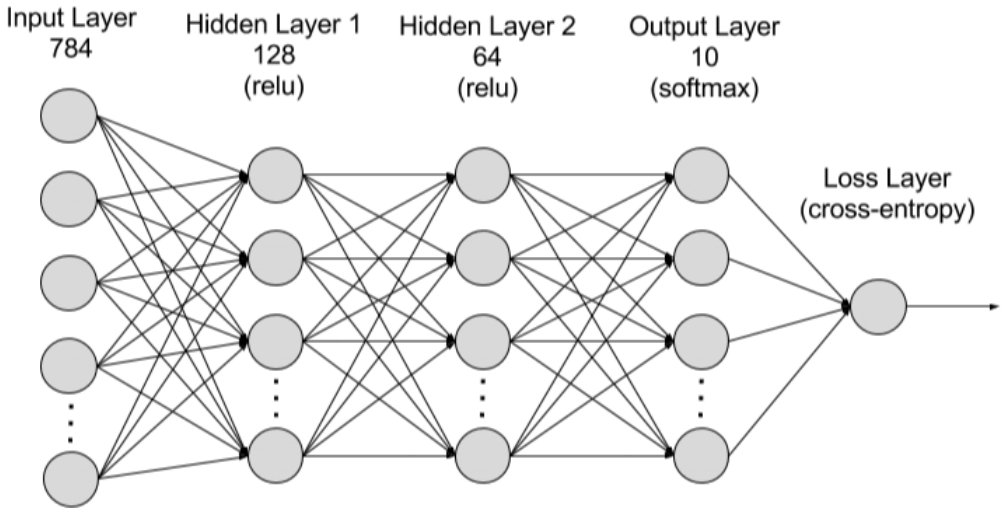
Idea: C_t actúa como una memoria de largo plazo controlada por puertas.

(ii) Esquema ilustrativo:

$$x_1, x_2, \ldots, x_T \text{ pasan por:}$$

$$h_{t-1} \rightarrow \boxed{\text{RNN/LSTM}} \rightarrow h_t \rightarrow h_{t+1}$$

[Inserte diagrama desenrollado en el tiempo]



(iii) Aplicaciones representativas:

- Series de tiempo (energía, finanzas, sensores).
- Reconocimiento de voz y audio secuencial.
- Modelado de lenguaje (antes del dominio de Transformers).

4. Transformers (Self-Attention)

Concepto general: Eliminan la recurrencia y usan *auto-atención* para modelar dependencias globales en la secuencia, permitiendo un entrenamiento altamente paralelo.

Capas principales (bloque encoder típico):

- Capa de embeddings + codificación posicional.
- Multi-Head Self-Attention.
- Bloque feed-forward (MLP) posición a posición.
- Normalización de capa y conexiones residuales.

(i) Modelo Matemático: Atención Escalada A partir de la entrada $X \in \mathbb{R}^{S \times d}$ (longitud S, dimensión d), se definen:

$$Q = XW^Q, \quad K = XW^K, \quad V = XW^V$$

Donde $W^Q, W^K, W^V \in \mathbb{R}^{d \times d_k}$ *son matrices de proyección.*
La atención por producto punto escalado se define como:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V$$

Interpretación:

- $QK^\top$ mide similitud entre posiciones de la secuencia.
- $\sqrt{d_k}$ estabiliza la escala de los productos punto.
- softmax convierte esas similitudes en pesos de atención.

- Multiplicar por V combina la información según esos pesos.

Multi-Head Attention (resumen):

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O$$

Cada cabeza:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

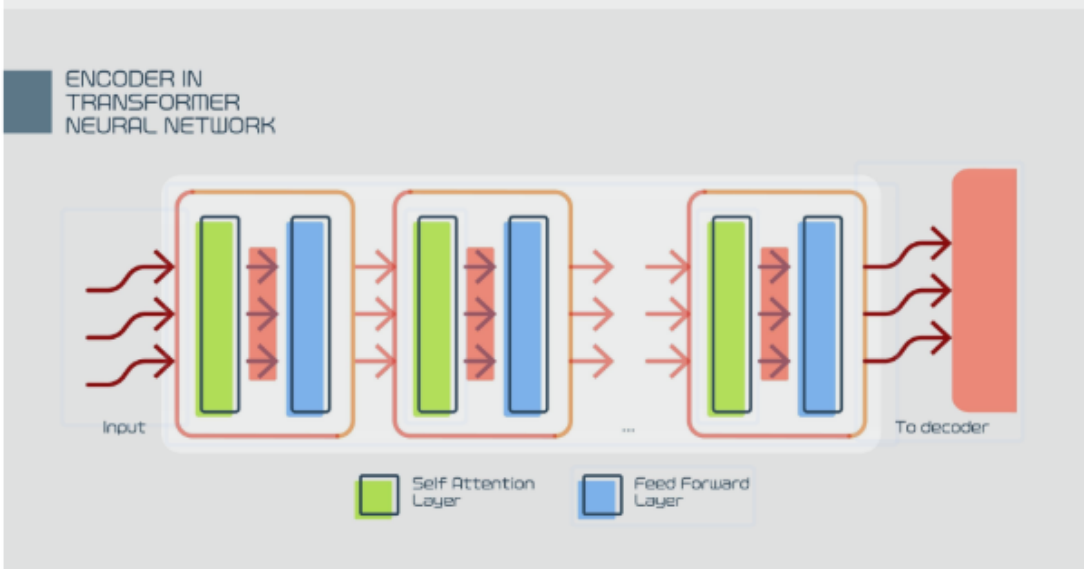
Feed-Forward interno (posición a posición):

$$\text{FFN}(x) = W_2\, f(W_1 x + b_1) + b_2$$

(ii) Esquema ilustrativo:

Tokens → Embedding + Positional Encoding → Multi-Head Self-Attention → FFN → Output

[Inserte diagrama tipo bloque Encoder / Encoder-Decoder]



(iii) Aplicaciones representativas:

- Modelos de lenguaje grandes (GPT, BERT, etc.).
- Vision Transformers (ViT) para imágenes.
- Modelos multimodales (texto + imagen + audio).

Síntesis Comparativa Fundamental

Característica	MLP	CNN	RNN/LSTM	Transformers
Sesgo inductivo	Débil / genérico	Local espacial	Secuencial / temporal	Relacional global
Tipo de entrada	Vectores	Imágenes / mapas	Secuencias	Secuencias (tokens, parches)
Procesamiento	Paralelo	Paralelo	Principalmente secuencial	Altamente paralelo
Memoria	Implícita (profundidad)	Local (receptive field)	Estado oculto y celda	Atención sobre toda la secuencia
Dependencias largas	Limitadas	Difíciles	Posibles pero costosas	Naturales (auto-atención)
Uso típico	Tabular / capa final	Visión, audio 2D	Series, voz, texto	NLP, visión, multimodal