

TIPOS DE ARQUITECTURA DE REDES NEURONALES ARTIFICIALES

Hecho por: Juan Pablo Ocampo Santana, Juan Esteban Mora Diaz y Oscar Andres Tepud Jacome.

REDES DENSAS

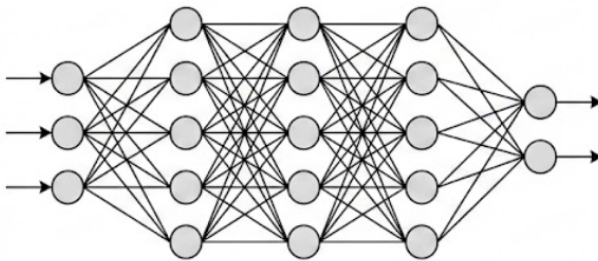
Cada neurona en una capa está conectada a todas las neuronas de la siguiente capa. No asumen estructura espacial ni temporal en los datos.

Modelo Matemático: La salida de una neurona j es una combinación lineal de las entradas seguida de una función de activación no lineal (**sigma**).

$$y = \sigma(Wx + b)$$

- Donde:
- W : Matriz de pesos.
- x : Vector de entrada.
- b : Vector de sesgo (bias).

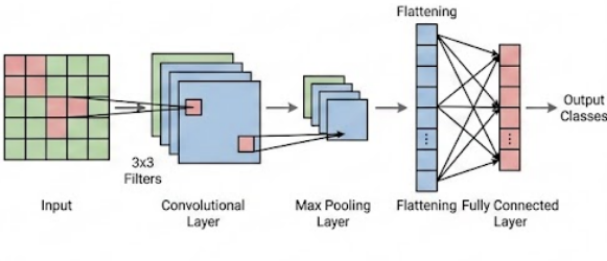
Red Densa (MLP)



Aplicaciones

- Datos tabulares
- Regresión
- Clasificación simple

Red Convolucional (CNN)



Aplicaciones

- Reconocimiento de imágenes.
- Visión por Computadora (Detección de objetos, etc).
- Análisis de señales de audio (espectrogramas).

REDES CONVOLUCIONALES

Utilizan filtros (kernels) que se desplazan sobre la entrada para extraer características locales (bordes, texturas) y preservar la relación espacial. Tienen invarianza a la traslación.

Modelo Matemático: La operación central es la convolución discreta. Para una imagen I y un filtro K :

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n) K(i - m, j - n)$$

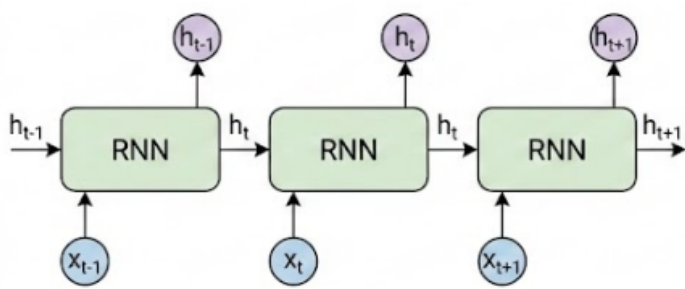
REDES RECURRENTE

Poseen "memoria". La salida actual depende de la entrada actual y del estado oculto anterior (lo que aprendió en el paso de tiempo previo).

Modelo Matemático (RNN Simple): El estado oculto h_t se actualiza en cada paso de tiempo t , Donde h_{t-1} es la memoria del paso anterior:

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

Red Recurrente (RNN)



Aplicaciones

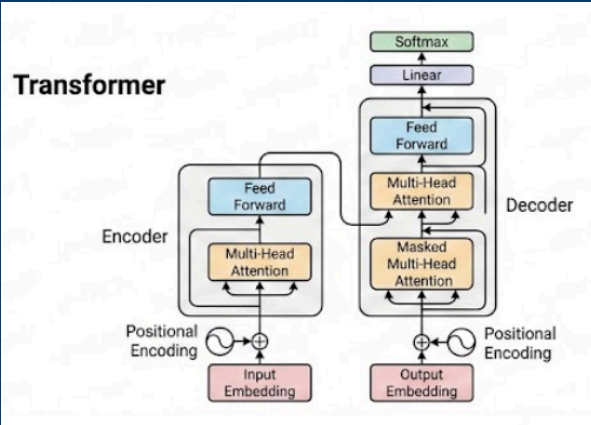
- Series de tiempo.
- Procesamiento de lenguaje natural (antes de transformers).
- Reconocimiento de voz.
- Modelos de secuencia a secuencia.

REDES TRANSFORMERS (ATENCIÓN)

Desechan la recurrencia secuencial. Utilizan el mecanismo de Self-Attention para ponderar la importancia de cada parte de la entrada en relación con las demás, todo en paralelo.

Modelo Matemático (Scaled Dot-Product Attention): Calcula la relevancia entre Queries (Q), Keys (K) y Values (V), Donde $\sqrt{d_k}$ es un factor de escala para estabilidad :

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



Aplicaciones:

- Grandes Modelos de Lenguaje (LLMs como GPT, BERT, Claude).
- Traducción automática de alta calidad.
- Vision Transformers (ViT) para imágenes.