


A decorative graphic in the top-left corner featuring a network of interconnected nodes and lines. Some nodes are solid blue circles, while others are grey circles with concentric outlines. The lines are thin and grey.

Inteligencia Artificial

IA-03.

Principales arquitecturas de IA generativa y
Uso de varias Herramientas de IA Generativa

A decorative graphic in the bottom-right corner, similar to the one in the top-left, showing a network of nodes and lines with some blue and grey nodes.

Principales Arquitecturas de IA Generativa

La arquitectura de la inteligencia artificial generativa (IAG) es el conjunto de técnicas y algoritmos que permiten crear contenidos digitales a partir de datos, como imágenes, textos, sonidos o vídeos. La IAG se basa en el aprendizaje automático, que es la capacidad de las máquinas de aprender de los datos sin necesidad de programación explícita. Una de las formas más comunes de IAG es el uso de redes neuronales artificiales, que son modelos matemáticos inspirados en el funcionamiento del cerebro humano.

Las redes neuronales artificiales se componen de capas de unidades de procesamiento llamadas neuronas, que se conectan entre sí mediante pesos. Cada neurona recibe una entrada, la procesa y genera una salida. La salida de una capa se convierte en la entrada de la siguiente, hasta llegar a la capa final, que produce el resultado deseado. Las redes neuronales se entrenan mediante un proceso de retroalimentación, en el que se comparan los resultados obtenidos con los esperados y se ajustan los pesos para minimizar el error.

Generative adversarial networks (GANs):

- ⊙ Los GANs son una de las arquitecturas más populares en IA generativa. Consiste en dos redes neuronales, el generador y el discriminador que se entrenan de manera adversarial.
- ⊙ Por ejemplo, supongamos que tienes dos compañeros uno es el generador y el otro el discriminador. El generador se encarga de inventar cosas que se parezcan mucho a la realidad, mientras que el discriminador se dedica a distinguir entre las cosas reales y lo que inventa el creador.
- ⊙ Este proceso de competencia impulsa a la red a mejorar constantemente sus habilidades generativas.

Redes neuronales recurrentes (RNNs) y long short-term memory (LSTM):

- ◎ Son arquitecturas clave en el campo de la IA generativa, diseñadas para trabajar con secuencias de datos, como texto o audio.
- ◎ Los RNNs tienen la habilidad especial para recordar cosas de etapas anteriores en la secuencia. Lo que hace que puedan entender contextos de datos que siguen un orden.
- ◎ Los LSTMs, por otro lado, son como los RNNs mejorados. Pueden capturar relaciones a largo plazo en las secuencias, como esos detalles importantes que se extienden en el tiempo.

Transformers:

- ⊙ El modelo de IA generativa basado en transformadores se basa en los conceptos de codificador y decodificador de los VAE. Los modelos basados en transformadores añaden más capas al codificador para mejorar el rendimiento en las tareas basadas en texto, como la comprensión, la traducción y la escritura creativa.
- ⊙ Los modelos basados en transformadores utilizan un mecanismo de autoatención. Sopesan la importancia de las diferentes partes de una secuencia de entrada al procesar cada elemento de la secuencia.
- ⊙ Otra característica clave es que estos modelos de IA implementan incrustaciones contextuales. La codificación de un elemento de secuencia depende no solo del elemento en sí, sino también de su contexto dentro de la secuencia.

Variational Autoencoders (VAEs):



- ⦿ Los VAEs son redes neuronales que aprenden a representar y generar datos a través de un espacio latente probabilístico.
- ⦿ Se componen de un codificador que mapea los datos de entrada al espacio latente y un decodificador que mapea muestras del espacio latente de vuelta a datos de salida.
- ⦿ Los VAEs se utilizan en aplicaciones como la generación de imágenes y la interpolación de datos
- ⦿ Los VAEs son modelos probabilísticos que aprenden representaciones latentes de datos.
- ⦿ Cuentan con dos partes clave: un codificador y un decodificador. El codificador toma la entrada y la convierte de forma especial en un secreto. Luego, el decodificador toma este secreto y lo transforma de vuelta en la entrada original.

Autoregressive Models:

- ⦿ Los modelos autorregresivos, dentro del ámbito de la IA generativa, se basan en lo que han hecho antes, imagina que estás dibujando y cada trazo se basa en el anterior.
- ⦿ Son generadores de datos uno a la vez, cada nueva parte se construye sobre la anterior, por ejemplo son geniales para predecir cosas en secuencias.
- ⦿ Estos modelos modelan la distribución de probabilidad de los datos como un producto de distribuciones condicionales univariadas.
- ⦿ Generan datos secuenciales o de alta dimensión utilizando un proceso de muestreo autoregresivo.
- ⦿ Ejemplos incluyen PixelCNN y WaveNet.

Kandinsky-2

Un innovador modelo de código abierto convierte texto en imágenes

 — Text embedding
 — Image embedding



CLIP-ViT-G
Text Encoder



Latent Diffusion
U-Net

MOVQ
Decoder



variation



CLIP-ViT-G
Image Encoder



Latent Diffusion
U-Net

MOVQ
Decoder



image fusion

Neon
lights



CLIP-ViT-G
Text Encoder



Diffusion
Mapping

CLIP-ViT-G
Image Encoder



Latent Diffusion
U-Net

MOVQ
Decoder



fusion with text


```
de kandinsky2 importar get_kandinsky2  
model = get_kandinsky2( 'cuda' , task_type= 'text2img' , model_version= '2.1' , use_flash_attention= False )  
image_cat = model.generate_text2img(  
    "gato rojo, foto 4k" ,  
    num_steps= 100 ,  
    lote_size= 1 ,  
    Guia_escala= 4 ,  
    h= 768 , w= 768 ,  
    muestra= 'p_sampler' ,  
    prior_cf_scale= 4 ,  
    prior_steps= "5"  
)[ 0]
```

Aquí se pueden elegir diferentes versiones de modelos: 2.0, 2.1, 2.2.



```
image_tiger = model.generate_text2img(  
    "tigre" ,  
    num_steps= 100 ,  
    tamaño de lote = 1 ,  
    escala_orientación = 4 ,  
    h = 768 ,  
    w = 768 ,  
    muestra = 'p_sampler' ,  
    prior_cf_scale = 4 ,  
    prior_steps = "5"  
) [ 0 ]
```



Y ahora combinémoslo usando el mix_imagemodelo 🐱+🐯 = ? :

```
image_mixed = model.mix_images(  
    [image_cat, image_tiger], [ 0.5 , 0.5 ],  
    num_steps= 100 ,  
    tamaño_de_lote = 1 ,  
    escala_de_orientación = 4 ,  
    h = 768 ,  
    w = 768 ,  
    muestra = 'p_sampler' ,  
    escala_cf_antes = 4 ,  
    pasos_piorales = "5" ,  
)[ 0 ]
```



¿y entonces, cual debo usar?

- © Estas son algunas de las principales arquitecturas utilizadas en IA generativa, cada una con sus propias características y aplicaciones específicas. La elección de la arquitectura depende del tipo de datos que se esté generando y las necesidades del problema en cuestión..