Visión Computacional

Ivan Sipiran

CLIP (OpenAI, 2021)

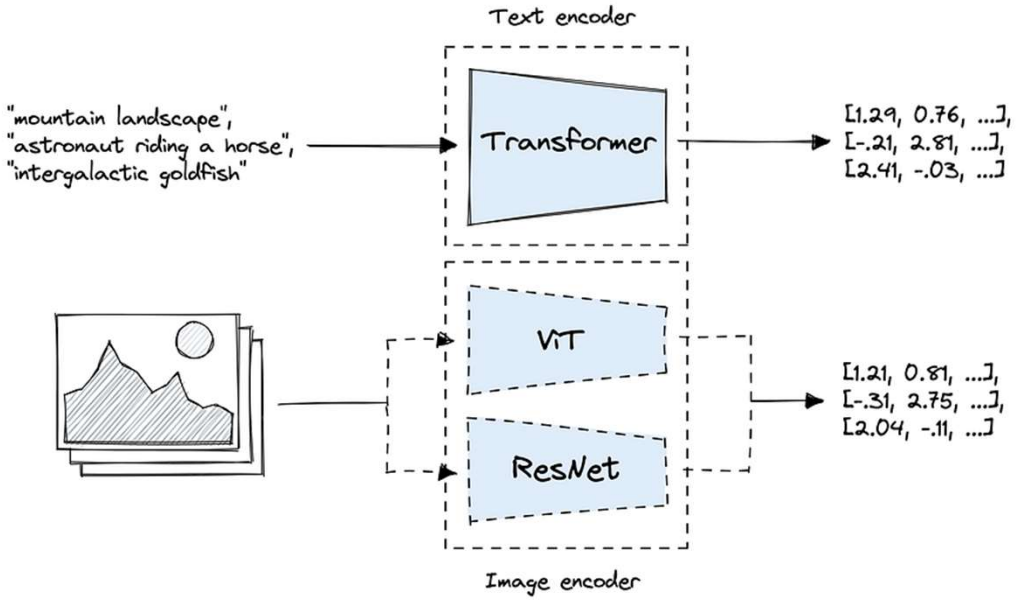
- Relaciona imágenes con texto utilizando aprendizaje contrastivo
- Objetivo: aprender representación conjunta entre imágenes y texto
- Utilidad en tareas de clasificación y búsqueda

- Entrenado en 400M de pares imágenes/texto
- No requiere etiquetas, aprende desde el texto
- Generaliza bien a zero-shot learning

CLIP – Arquitectura General

- Visual encoder: ResNet-50/ViT. Convierte imágenes a vectores
- Text encoder: Transformer. Convierte texto a vectores

- Encoders transforman los datos al mismo espacio latente
- No hay concatenación ni fusión temprana

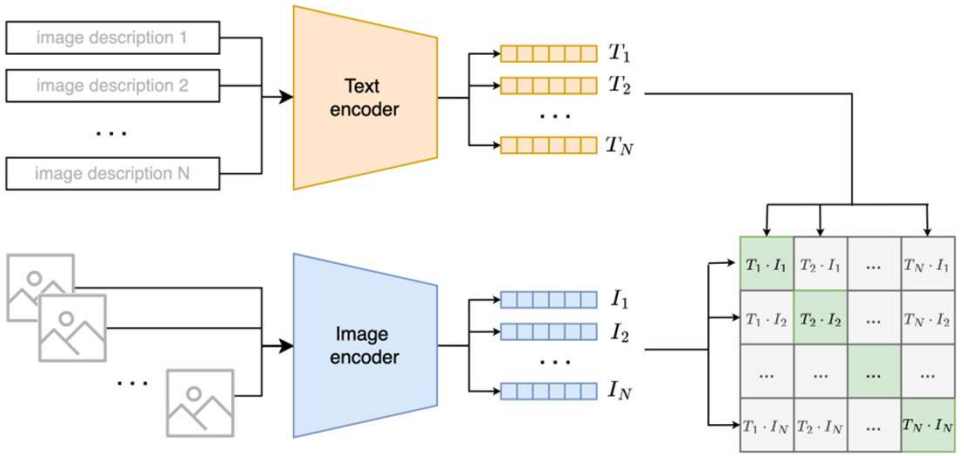


CLIP – Aprendizaje contrastivo

- Objetivo: maximizar la similitud coseno entre una imagen y su descripción textual correspondiente.
- Minimizar la similitud entre elementos del mismo batch

$$\text{Sim}(I_i, T_j) = \frac{I_i \cdot T_j}{\|I_i\| \|T_j\|}$$

- Batch de tamaño N
- Se calculan $N \times N$ similitudes y se aplica cross-entropy

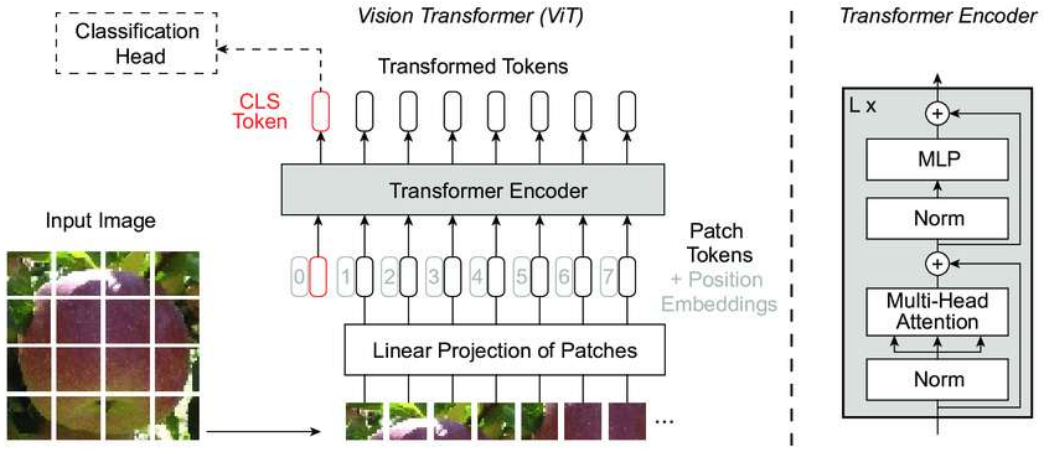


$$\mathcal{L} = \frac{1}{2N} \sum_{i=1}^{N} [\text{CE}(\text{Sim}(I_i, T_1, ..., T_N), i) + \text{CE}(\text{Sim}(T_i, I_1, ..., I_N), i)]$$

CLIP – Encoder visual

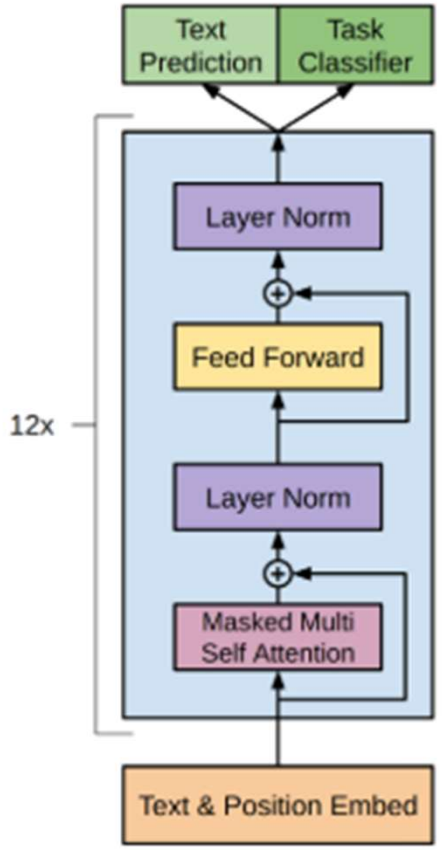
- Se entrenó con dos encoders visuales

- ResNet-50 modificada
  - Se reemplaza el pooling por attention pooling
  - Se eliminan ciertas capas para mejorar el alineamiento con texto
- Vision Transformer
  - Imagen → parches → embeddings + positional encodings
  - Pasa por capas de Transformer → se toma el token [CLS] como embedding visual



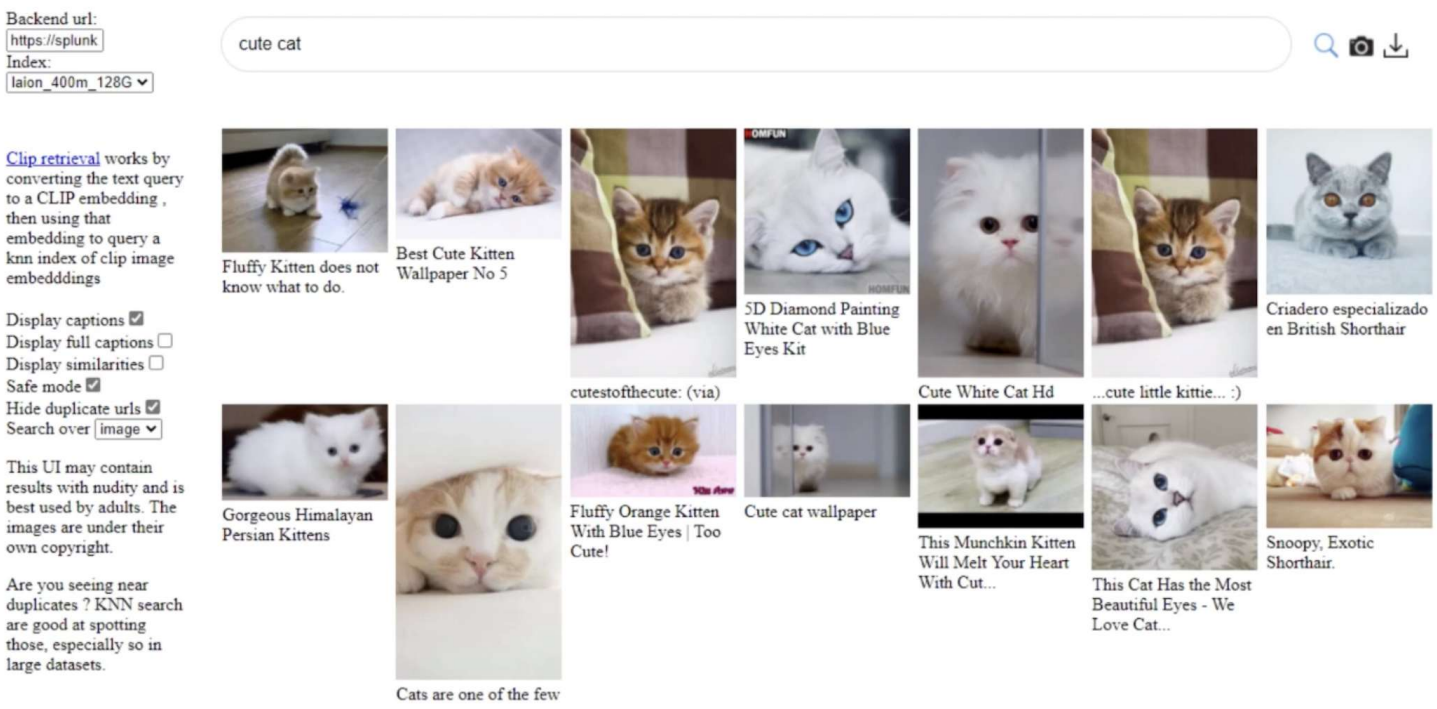
CLIP – Encoder textual

- Basado en Transformer de 12 capas – similar a GPT2
  - Tokenización de GPT2
  - Agrega un token especial de final [EOS], cuyo embedding final representa al texto



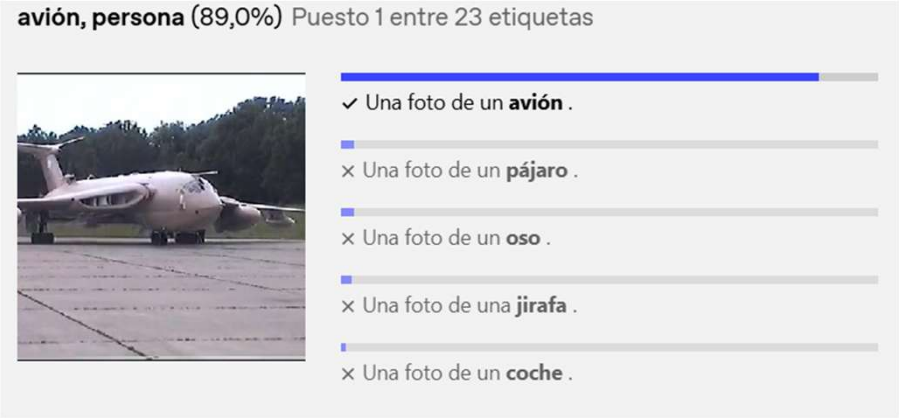
CLIP – Dataset y entrenamiento

- 400M de pares imagen/texto recolectados de la web, sin curación manual
- Entrenamiento por contrastive learning durante semanas en GPUs
- Optimización
  - AdamW
  - Batch size enorme (32K)



CLIP – Evaluación

- Tareas de clasificación sin finetuning, usando prompts textuales como clases
- Ejemplo de Zero-shot (incluir figura demostrativa)



- Resultados
  - Desempeño competitivo en >30 datasets de visión (ImageNet, CIFAR, EuroSAT, etc)
  - En algunos casos, supera modelos entrenados supervisadamente

CLIP – Comentarios finales

- Fortalezas de CLIP
  - Generaliza bien a tareas no vistas (zero-shot)
  - Usa lenguaje natural como interfaz de interacción
  - Sirve como inspiración para otros modelos como DALL-E, Flamingo y BLIP

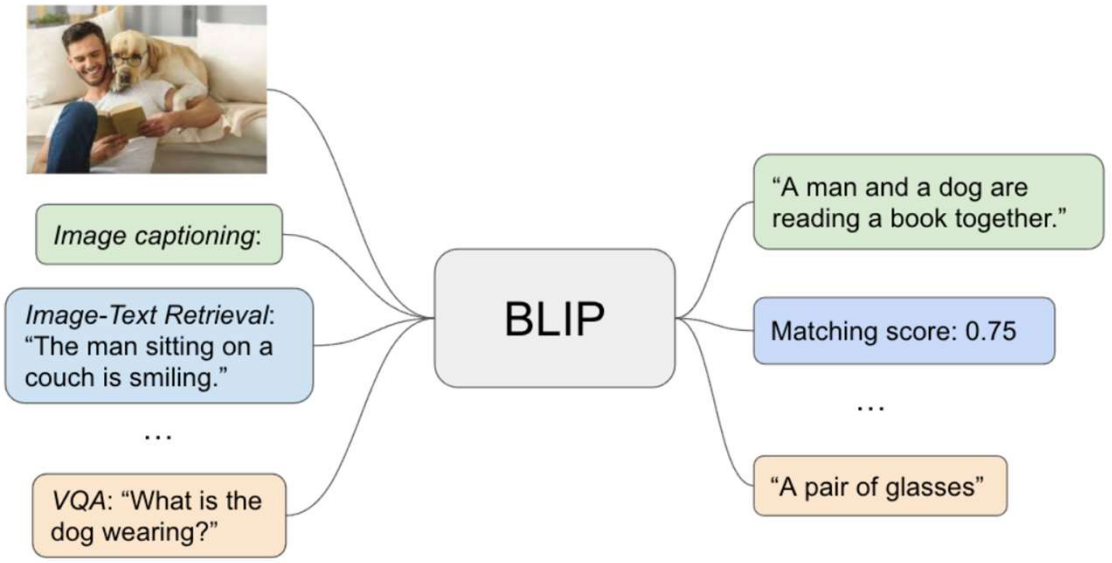
- Limitaciones
  - Sesgos presentes en los datos web → pueden amplificarse
  - No hace generación de texto ni análisis contextual
  - No se entrena para tareas estructuradas como VQA o captioning

- Se usa como backbone en otras propuestas
  - Imagen – text retrieval (ALIGN)
  - Text-to-image (DALL-E)
  - Multimodal QA (Flamingo, BLIP-2)

BLIP (Salesforce Research, 2022)

- Combina visión y lenguaje usando arquitectura multimodal unificada
- Tres tareas de pre-entrenamiento

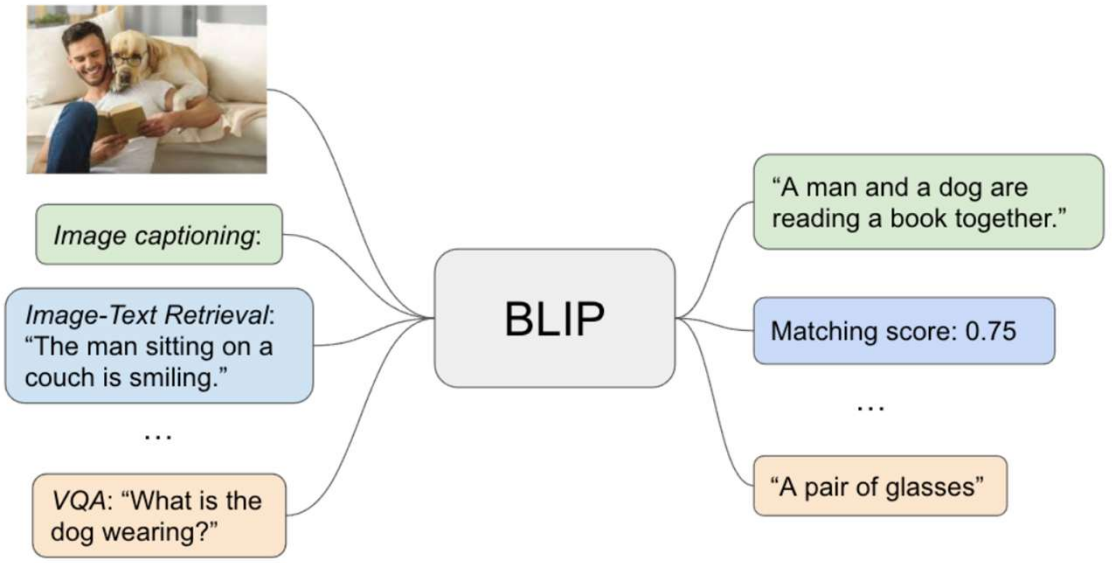
- Entrenado en datasets sin curación (Conceptual Captions, LAION)
- Aprendizaje multitarea contrastivo + generativo
- Se puede usar con zero-shot y finetuning



BLIP (Salesforce Research, 2022)

- Combina visión y lenguaje usando arquitectura multimodal unificada
- Tres tareas de pre-entrenamiento

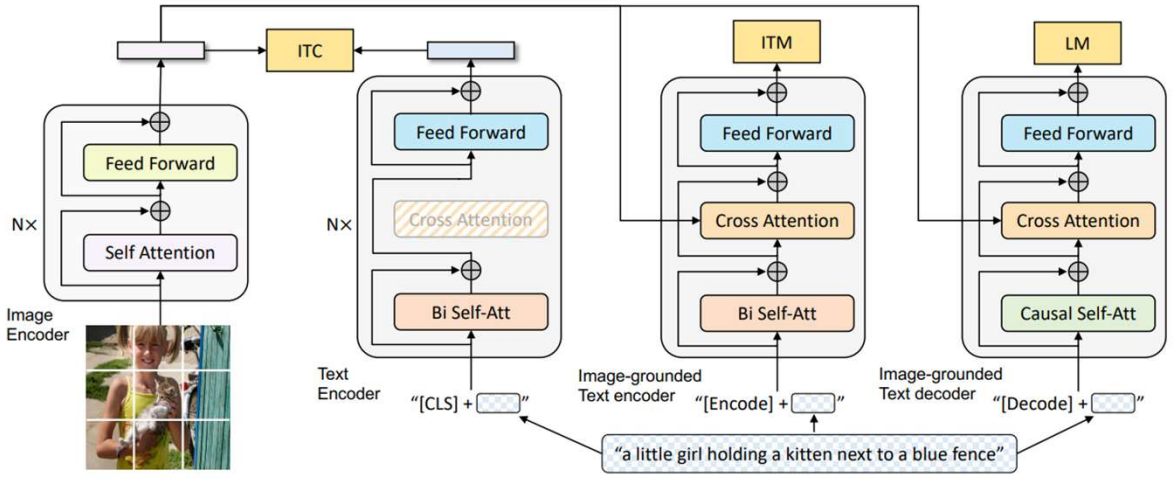
- Entrenado en datasets sin curación (Conceptual Captions, LAION)
- Aprendizaje multitarea contrastivo + generativo
- Se puede usar con zero-shot y finetuning



BLIP – Arquitectura general

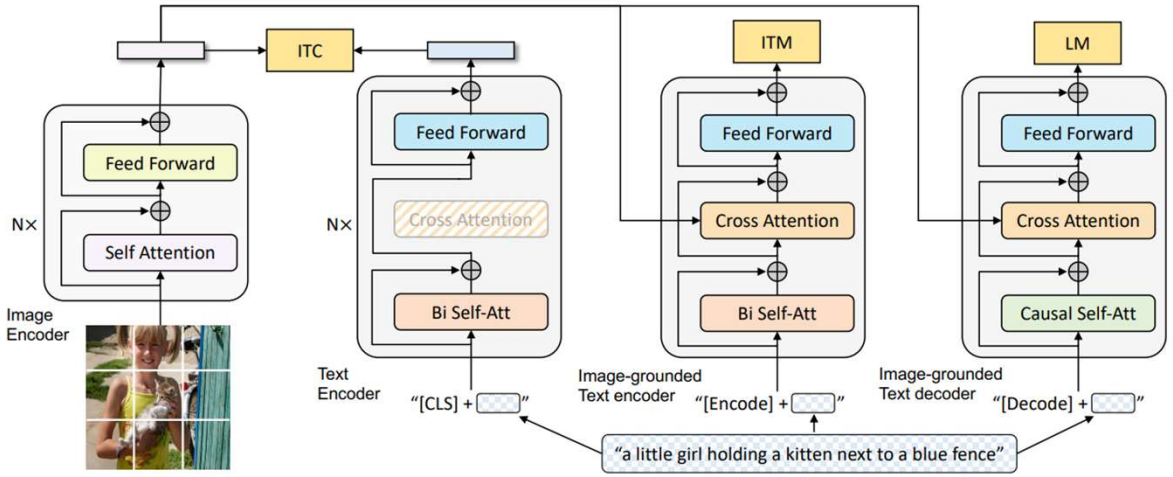
- Encoder visual → ViT
- Encoder textual → Transformer
- Fusionador → Transformer Cross-attention

- Tres modos de operación:
  - Encoder-decoder: captioning, generation
  - Image-text contrastive: retrieval
  - Image-text matching: razonamiento



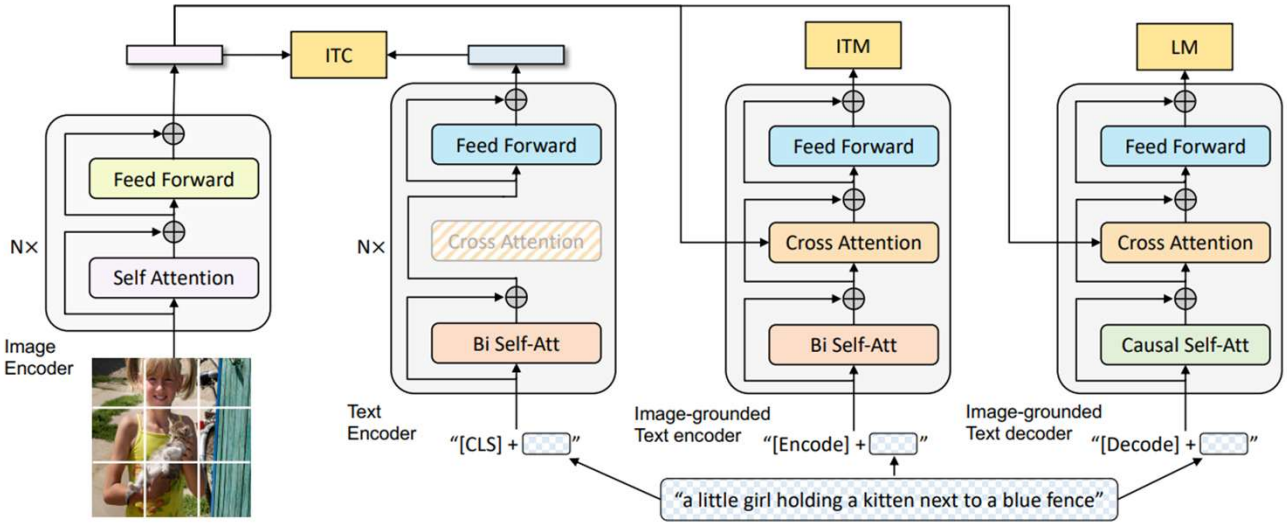
BLIP – Módulos

- Encoder visual
  - Vision Transformer: ViT-B/16 o ViT-L/16 preentrenado
  - Produce embeddings de parche + CLS
- Text
  - BERT-base (12 capas)
  - Puede actuar como encoder o decoder



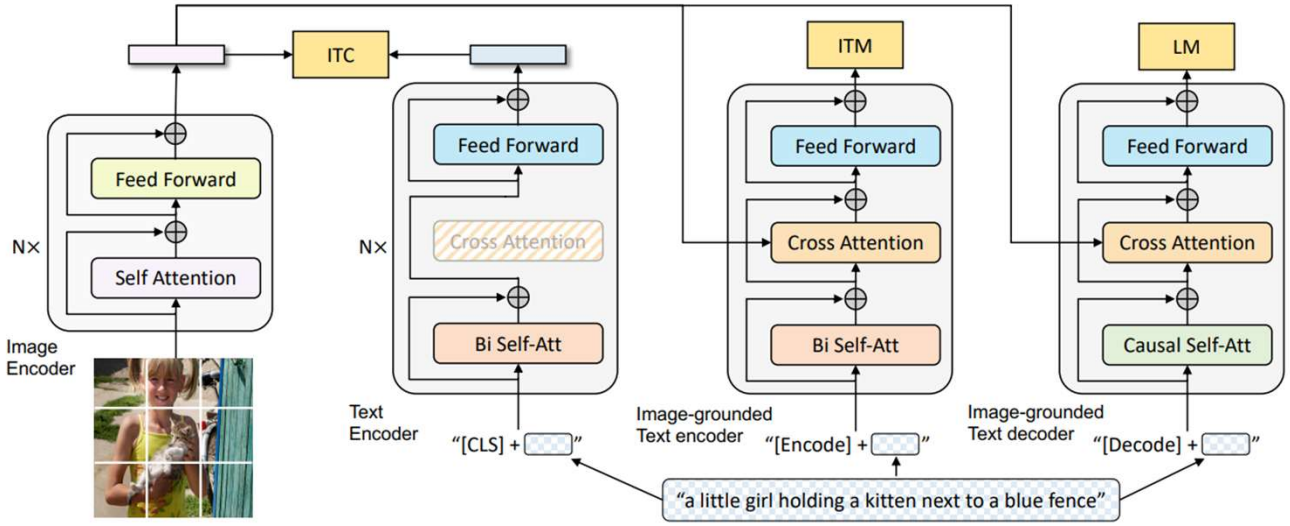
BLIP – Pre-entrenamiento

- Image-text contrastive (ITC)
  - Similar a CLIP
  - Imagen y texto se proyectan a mismo espacio latente y se alinean con los contrastiva
- Image-text matching (ITM)
  - Un clasificador binario predice si una imagen y un texto corresponde
  - Se alimenta con ejemplos positivos y negativos
- Language Modeling (LM)
  - El decodificador genera texto (caption/respuesta)
  - Entrenamiento autoregresivo



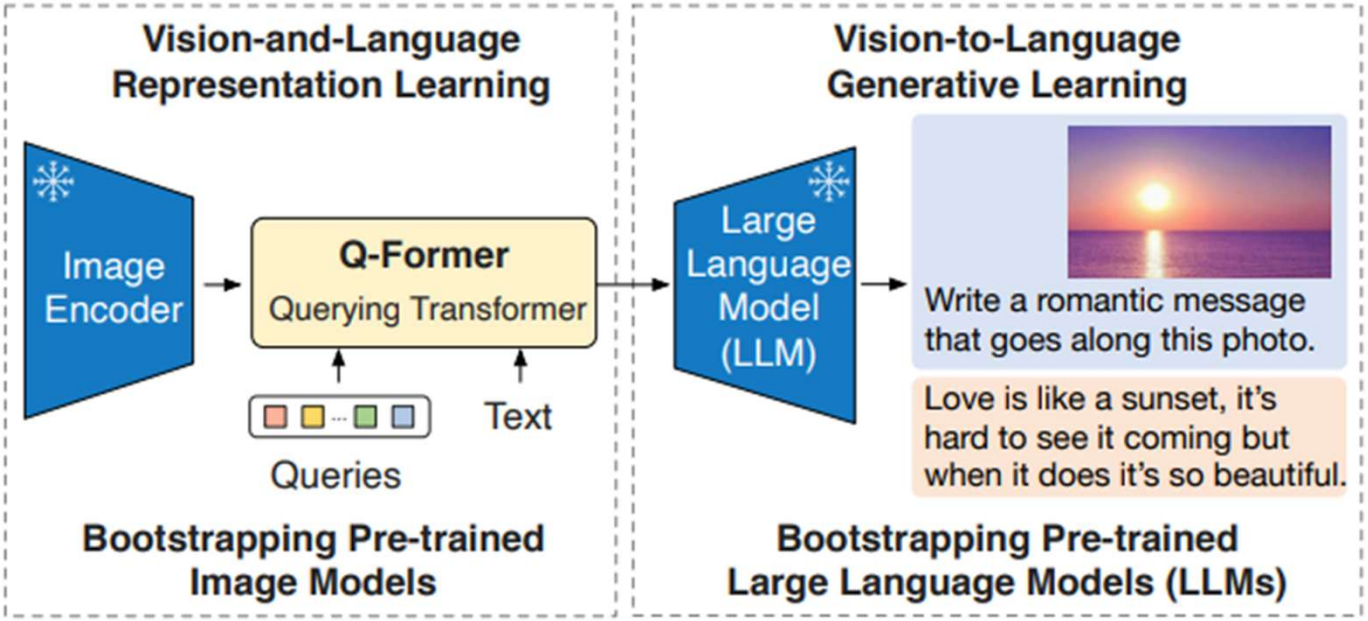
BLIP – Bootstrapping

- Se entrena un modelo base con datos depurados (CC3M, COCO, Visual Genome)
- Se usa el modelo para filtrar pares de buena calidad de datasets grandes (LAION, CC12M)
- El modelo se reentrena sobre el conjunto filtrado



BLIP-2

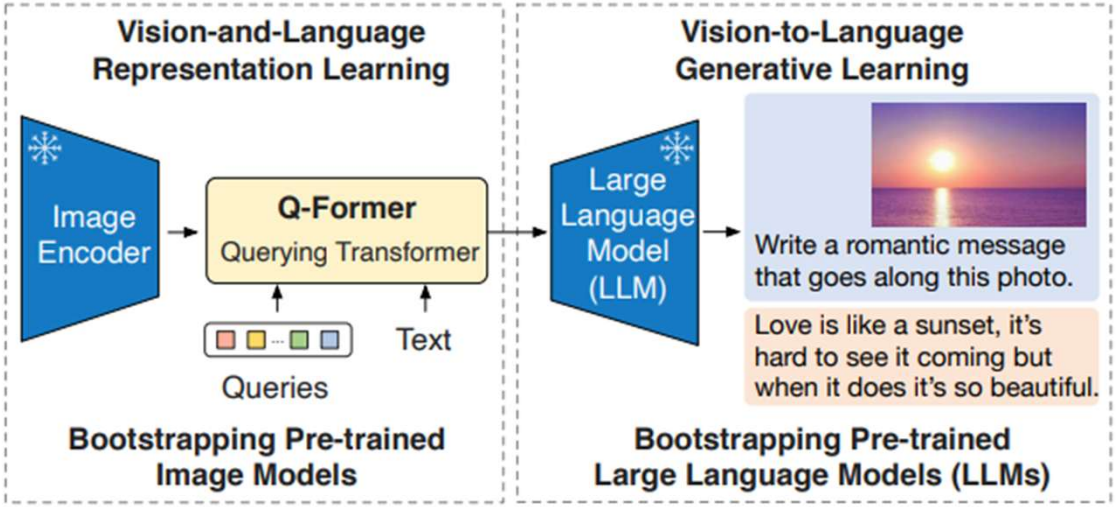
- Extensión de BLIP con tres ideas nuevas
  - Características visuales fijas
  - Mapear características visuales a espacio de lenguaje (Q-Former)
  - Usar un modelo de lenguaje (Vicuña) para responder, generar, razonar



BLIP-2 - Arquitectura

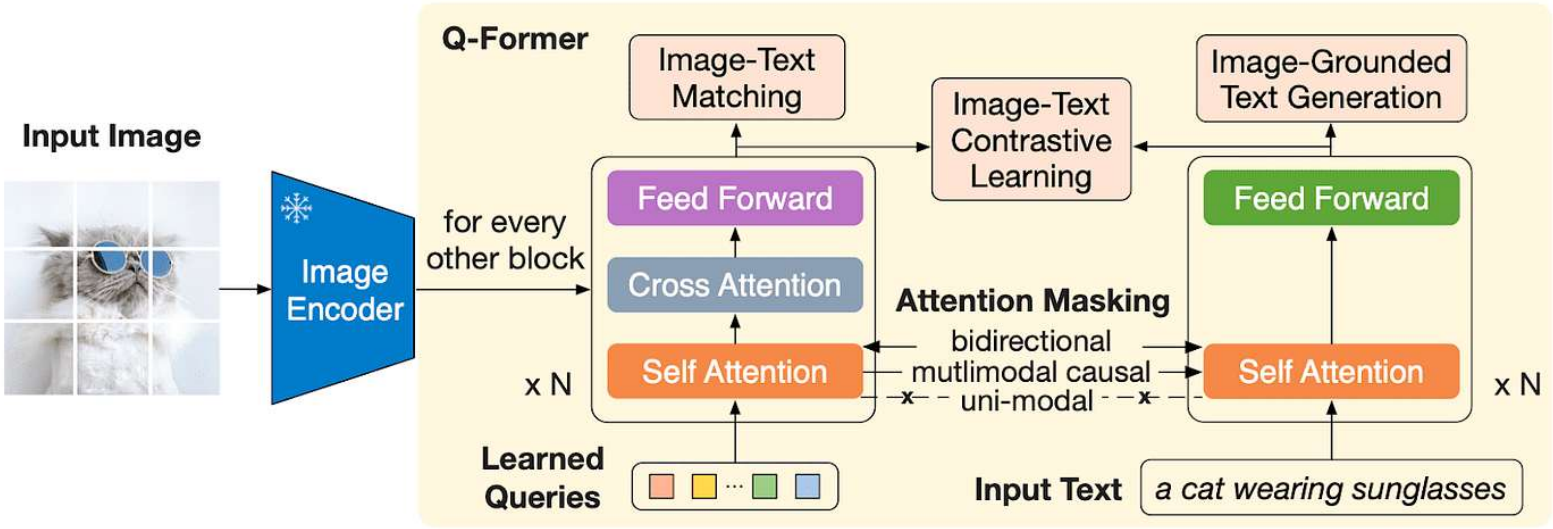
- Componentes
  - ViT Encoder
  - Q-Former: Transformer ligero + queries aprendibles
  - LLM (Flan-T5, OPT, Vicuña)

- ViT extrae características visuales
- Q-Former selecciona y adapta features visuales
- LLM genera texto y responde preguntas



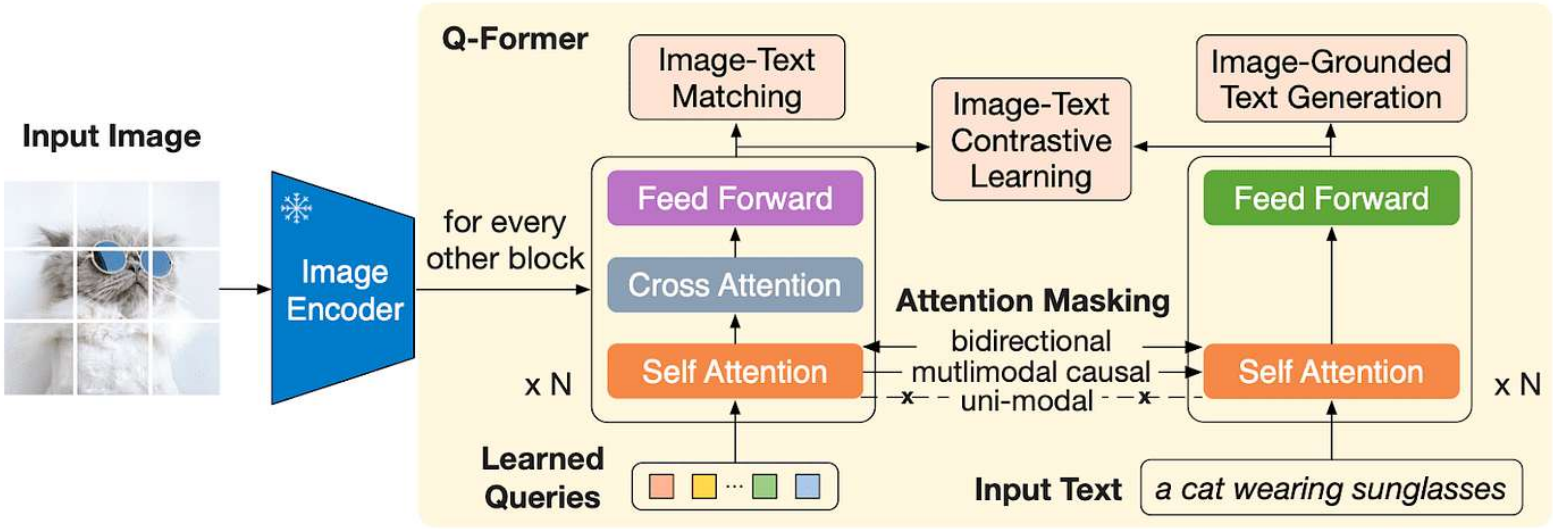
BLIP-2 – Q-Former

- Transformer con queries aprendibles
- Interactúan con las features de imagen usando atención cruzada
- Output: embeddings de texto que puedan pasar a un LLM
- Ventajas:
  - Reduce el tamaño del input al LLM
  - Aprende a seleccionar solo lo relevante visualmente



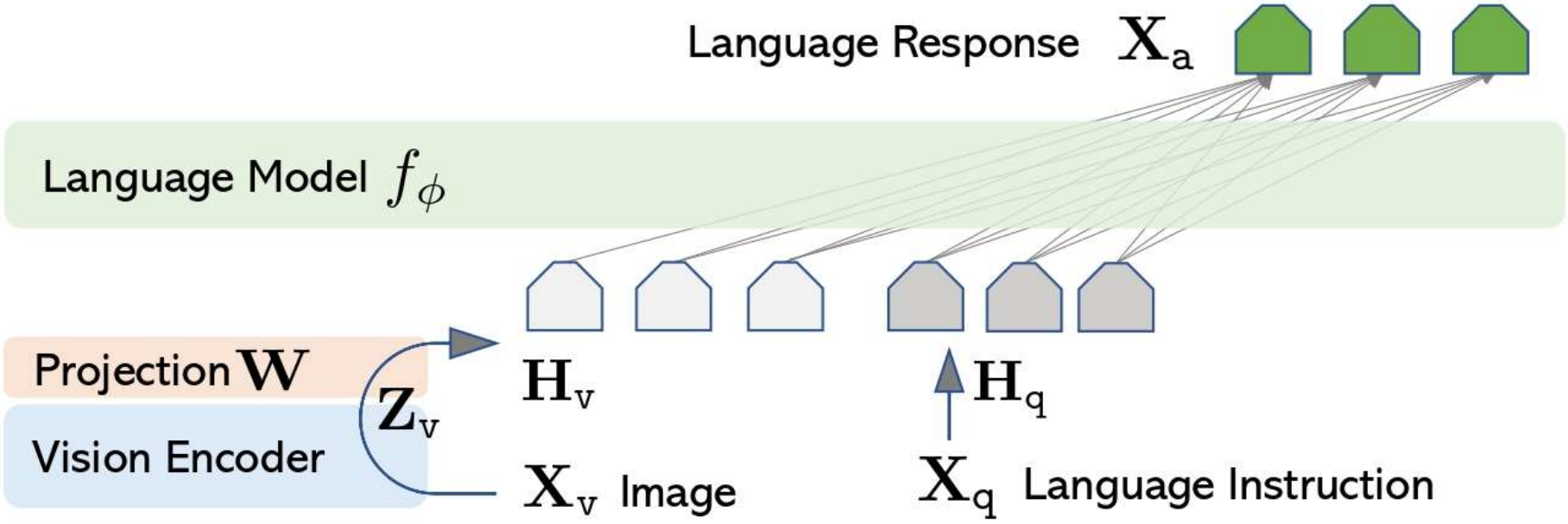
BLIP-2 – Entrenamiento

1) Pretrain del Q-Former
    1) ITC, ITM, LM, como en BLIP
2) Entrenamiento de Q-Former para alinear con LLM
    1) Usando texto generado por el LLM
3) Prompt tuning del LLM
    1) Sin modificar pesos del LLM (congelado)



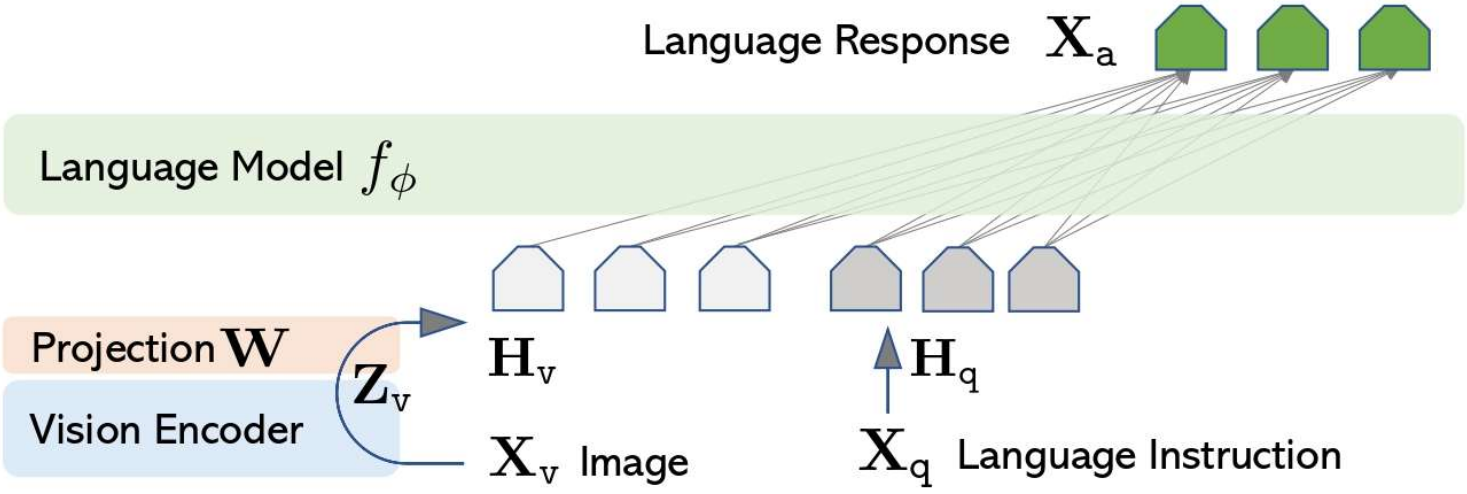
LLaVA (Large Language and Vision Assistant)

- Microsoft y UC Berkeley, 2023
- Integra encoder visual (CLIP-ViT) y LLM LLaMA
- Razonamiento conversacional: VQA, captioning, etc



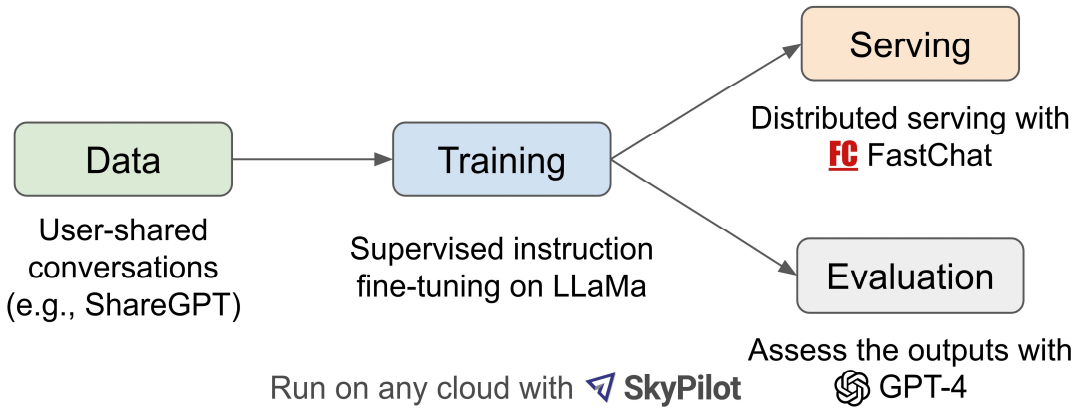
LLaVA – Arquitectura General

- Vision encoder: CLIP ViT L/14: produce embeddings por parches
- Linear Projection: Mapea los embeddings visuales al espacio de tokens del LLM
- Large Language Model LLaMA
  - LLM autoregresivo (LLaMA 7B o 13B)
  - Genera texto condicionado a los tokens visuales
- Input tokens: prompt textual + tokens visuales proyectados



LLaVA – Entrenamiento

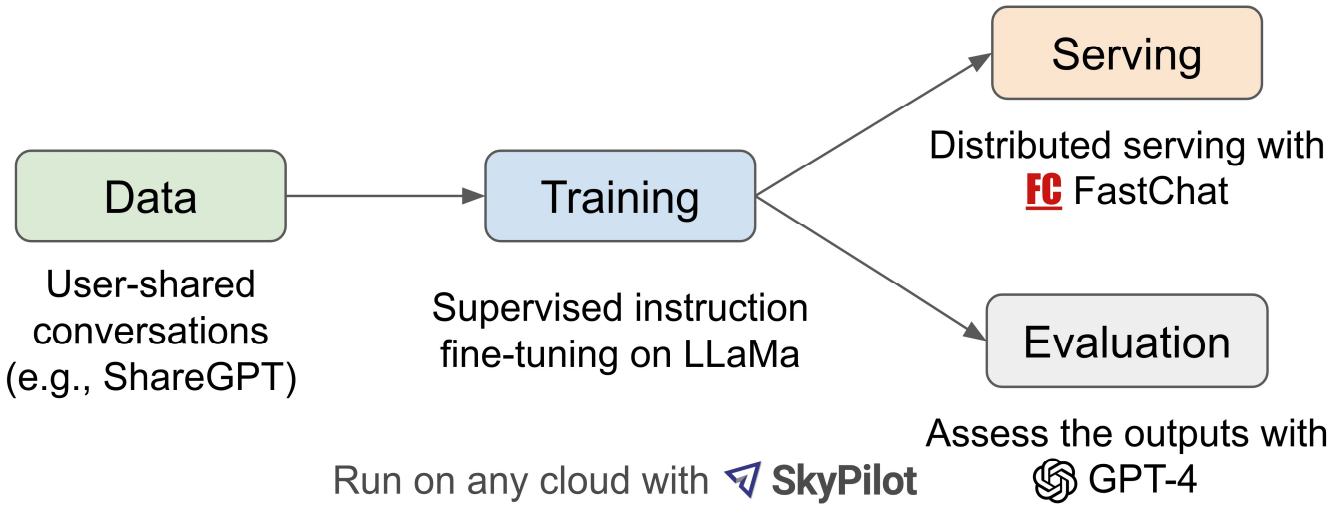
- Pre-align
  - Adaptar la proyección visual al espacio del LLM
  - Datos: 158k pares imagen-texto generados por BLIP-2
  - Minimizar pérdida de generación textual (causal loss)
- Conversational Finetuning
  - Datos sintéticos generados por GPT-4 usando imágenes y preguntas
  - 80K muestras de diálogo imagen-texto
  - Adaptar el modelo a una conversación multimodal



LLaVA – Datos

- Imagen: COCO, CC3M, Visual Genome
- Texto: generado automáticamente con GPT-4 a partir de instrucciones
- Instrucción: what would a helpful assistant say when shown this image and this prompt?

- Esto permite entrenar sin humanos, usando GPT-4 como supervisor

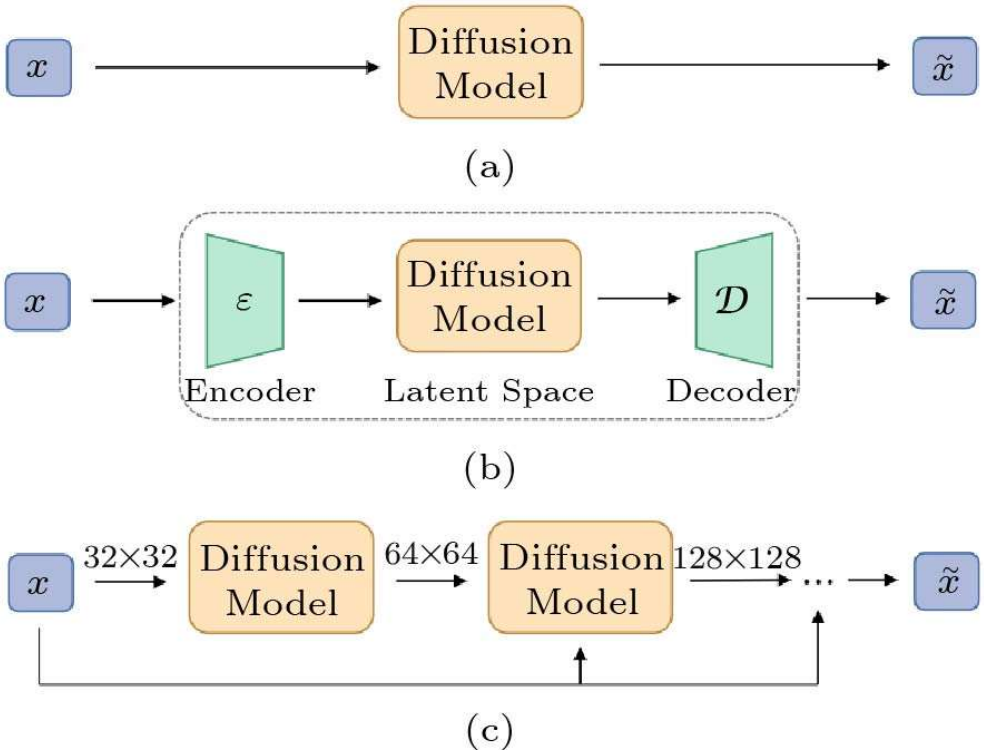


LLaVA

Modelo	Imagen encoder	LLM	Proyección	Razonamiento
CLIP	CLIP-ViT	No aplica	-	No
BLIP-2	ViT + QFormer	Flan-T5	Q-Former	Sí
LLaVA	CLIP-ViT	LLaMA	Linear	Básico
MiniGPT-4	CLIP-ViT	Vicuna	Linear	Sí (limitado)

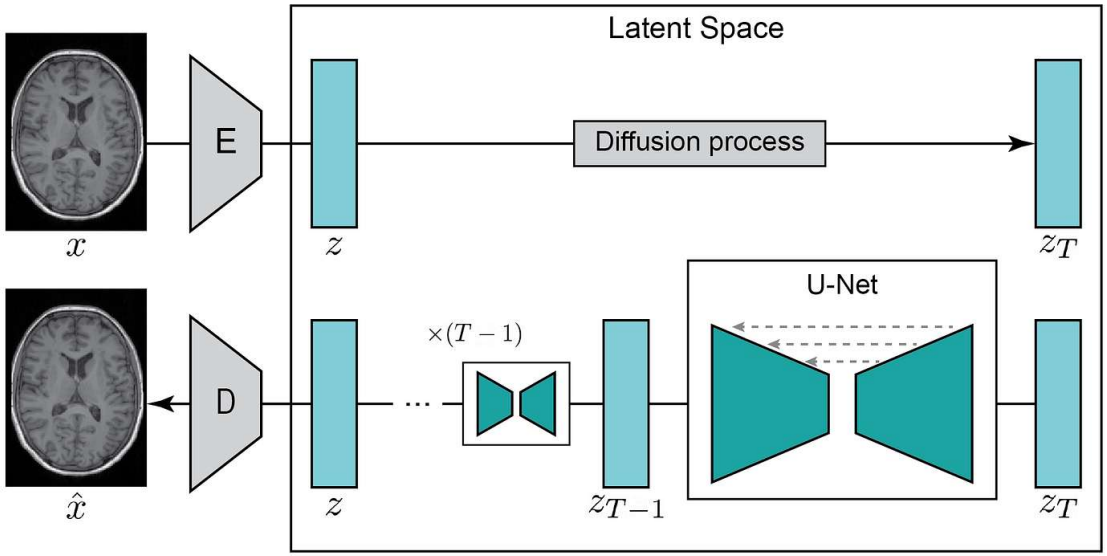
Latent Diffusion Models

Modelo generativo para aprender a sintetizar datos mediante un proceso de difusión, pero no en el espacio de píxeles, sino en un espacio latente comprimido



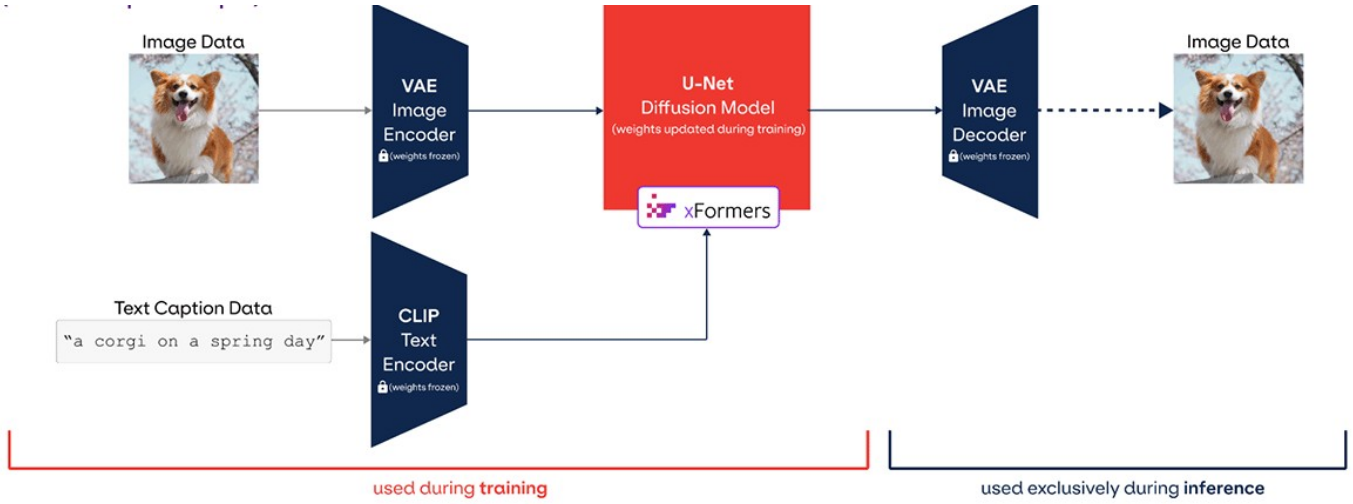
Latent Diffusion Models - Arquitectura

1. Autoencoder (VAE)
    1. Imagen $x \in \mathbb{R}^{H \times W \times 3}$ → espacio latente $z \in \mathbb{R}^{h \times w \times c}$
2. Modelo de difusión (U-Nnet)
    1. Entrenado para hacer denoising sobre z
3. Condicionamiento opcional
    1. Texto, clase, imagen, guía → usado como condición



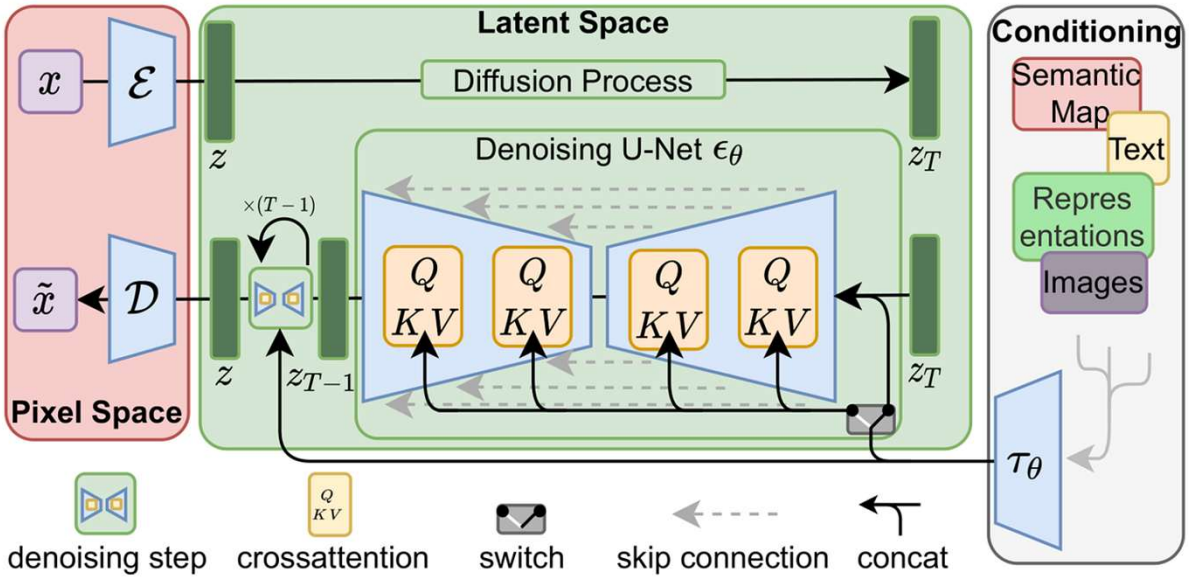
Stable Diffusion

- Modelo de difusión latente que genera imágenes desde texto
- Propuesto por StabilityAI (2022)
- Entrenado en LAION – 5B
- Código y pesos disponibles públicamente
- Tareas: Text-to-image, Image-to-image, Inpainting, Style transfer



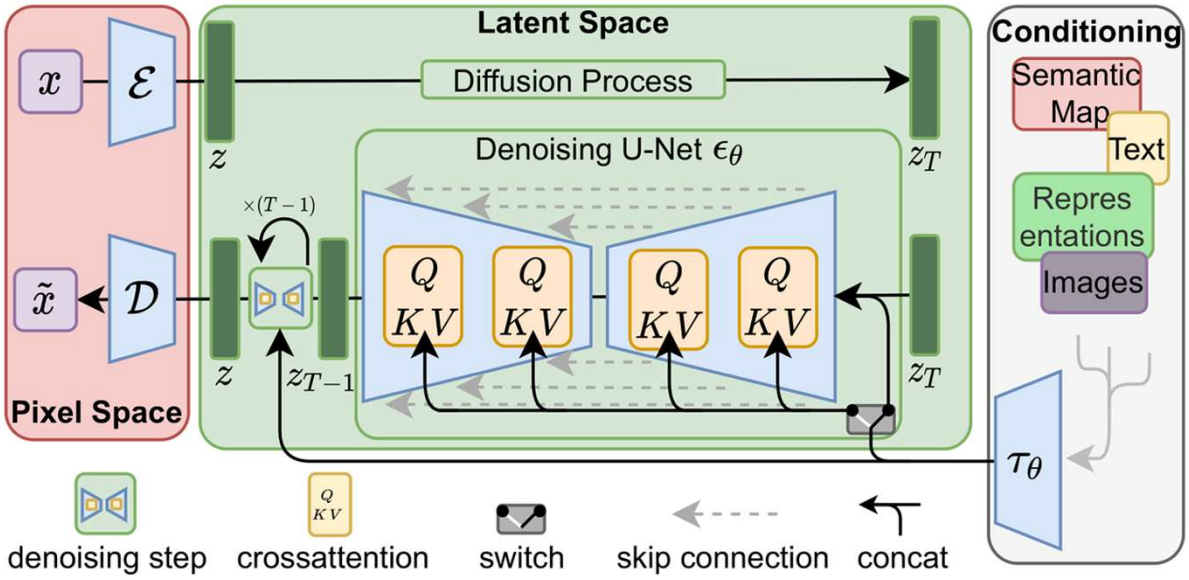
Stable Diffusion – Arquitectura General

- Autoencoder latente (VAE)
  - Codifica imágenes al espacio latente z
  - Decodifica z a imágenes completas
  - Reduce costos computacionales
- U-Net Denoiser
  - Modelo de difusión que aprende a remover ruido del espacio latente
  - Entrenado para invertir el proceso de degradación
- Condicionamiento textual (CLIP-T5)
  - Un codificador textual (CLIP o T5)
  - Embedding del texto guía la generación durante la difusión



Stable Diffusion – Entrenamiento

- Entrenamiento
  - Dataset: LAION
  - Imágenes de 512 x 512
- Optimización
  - Reconstrucción de ruido
  - Texto condicional
- Pasos
  - Auto-encoder pre-entrenado
  - U-NET condicionado
  - Encoder de texto congelado



Stable Diffusion – Variantes

Versión	Mejoras
Stable Diffusion v1.4	Modelo base público (512×512)
Stable Diffusion v1.5	Mejor rendimiento en caras y detalles
v2.0 y v2.1	Resolución 768×768, SDXL en desarrollo
SDXL	Mejor estilo, más detalles, prompts complejos
ControlNet, DreamBooth	Técnicas avanzadas para control y personalización