

# Vision Transformers

Docentes:

Esp. Abraham Rodriguez - FIUBA

Mg. Oksana Bokhonok - FIUBA

# Programa de la materia

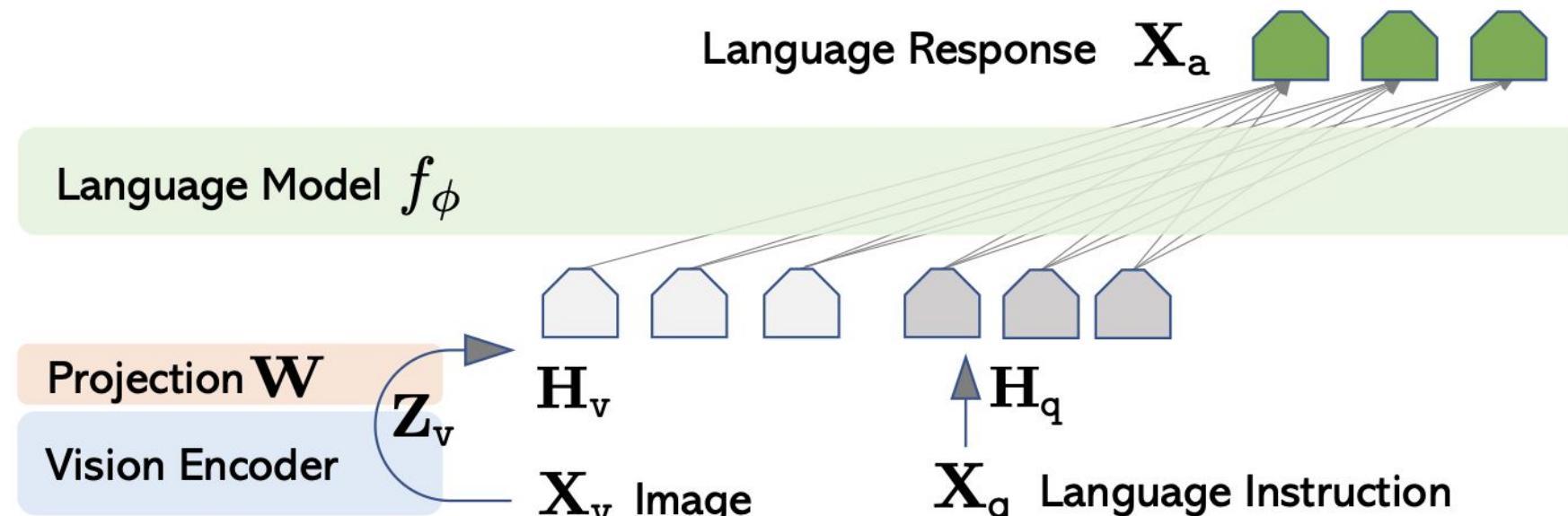
1. Arquitectura de Transformers e imágenes como secuencias.
2. Arquitecturas de ViT y el mecanismo de Attention.
3. Ecosistema actual, Huggingface y modelos pre entrenados.
4. GPT en NLP e ImageGPT.
5. **Modelos multimodales: combinación de visión y lenguaje.**
6. Segmentación con SAM y herramientas de auto etiquetado multimodales.
7. OCR y detección con modelos multimodales.
8. Presentación de proyectos.

# **Modelos multimodales**

# LLaVA

Presentado en el paper “[Visual Instruction Tuning](#)”, consiste en:

- Desarrollo del formato instructivo del dataset de pares de imágenes y texto mediante ChatGPT-4
- integrar un Vision Encoder (CLIP) y una LLM ([Vicuna](#)), para crear una Large Multimodal Model (LMM).
- Benchmark de instrucción multimodal.
- Desarrollo Open-source.



# LLaVA GPT-Assisted Data Generation

Debido al proceso lento de creación de datos, se utilizó ChatGPT-4 para realizar la tarea de **anotación de texto**, para generar un dataset de pares de imágenes-texto

Para una imagen  $X_v$  y su caption  $X_c$ , se crea un conjunto de preguntas  $X_q$ .

La intención es instruir al modelo para **describir** el contenido de la imagen. Mediante GPT-4 se crea una lista de tales preguntas. Por lo tanto, una forma sencilla de expandir un par imagen-texto a su versión de seguimiento de instrucciones es:

Humano :  $X_q X_v <\text{STOP}>$  Asistente :  $X_c <\text{STOP}>$ .

Esta versión ampliada carece de diversidad y razonamiento profundo tanto en las instrucciones como en la respuesta. Para mitigar lo anterior, se generan bounding boxes de los objetos descritos por el caption.

## Context type 1: Captions

A group of people standing outside of a black vehicle with various luggage.

Luggage surrounds a vehicle in an underground parking area

People try to fit all of their luggage in an SUV.

The sport utility vehicle is parked in the public garage, being packed for a trip

Some people with luggage near a van that is transporting it.



## Context type 2: Boxes

person: [0.681, 0.242, 0.774, 0.694], person: [0.63, 0.222, 0.686, 0.516], person: [0.444, 0.233, 0.487, 0.34], backpack: [0.384, 0.696, 0.485, 0.914], backpack: [0.755, 0.413, 0.846, 0.692], suitcase: [0.758, 0.413, 0.845, 0.69], suitcase: [0.1, 0.497, 0.173, 0.579], bicycle: [0.282, 0.363, 0.327, 0.442], car: [0.786, 0.25, 0.848, 0.322], car: [0.783, 0.27, 0.827, 0.335], car: [0.86, 0.254, 0.891, 0.3], car: [0.261, 0.101, 0.787, 0.626]

# LLaVA GPT-Assisted Data Generation

```
messages = [ {"role": "system", "content": f"""You are an AI visual assistant, and you are seeing a single image. What you see are provided with five sentences, describing the same image you are looking at. Answer all questions as you are seeing the image.
```

Design a conversation between you and a person asking about this photo. The answers should be in a tone that a visual AI assistant is seeing the image and answering the question. Ask diverse questions and give corresponding answers.

Include questions asking about the visual content of the image, including the **object types, counting the objects, object actions, object locations, relative positions between objects**, etc. Only include questions that have definite answers:

- (1) one can see the content in the image that the question asks about and can answer confidently;
- (2) one can determine confidently from the image that it is not in the image. Do not ask any question that cannot be answered confidently.

Also include complex questions that are relevant to the content in the image, for example, asking about background knowledge of the objects in the image, asking to discuss about events happening in the image, etc. Again, do not ask about uncertain details. Provide detailed answers when answering complex questions. For example, give detailed examples or reasoning steps to make the content more convincing and well-organized. You can include multiple paragraphs if necessary.""} ]

```
for sample in fewshot_samples:
    messages.append({"role": "user", "content": sample['context']})
    messages.append({"role": "assistant", "content": sample['response']} )
messages.append({"role": "user", "content": '\n'.join(query)})
```

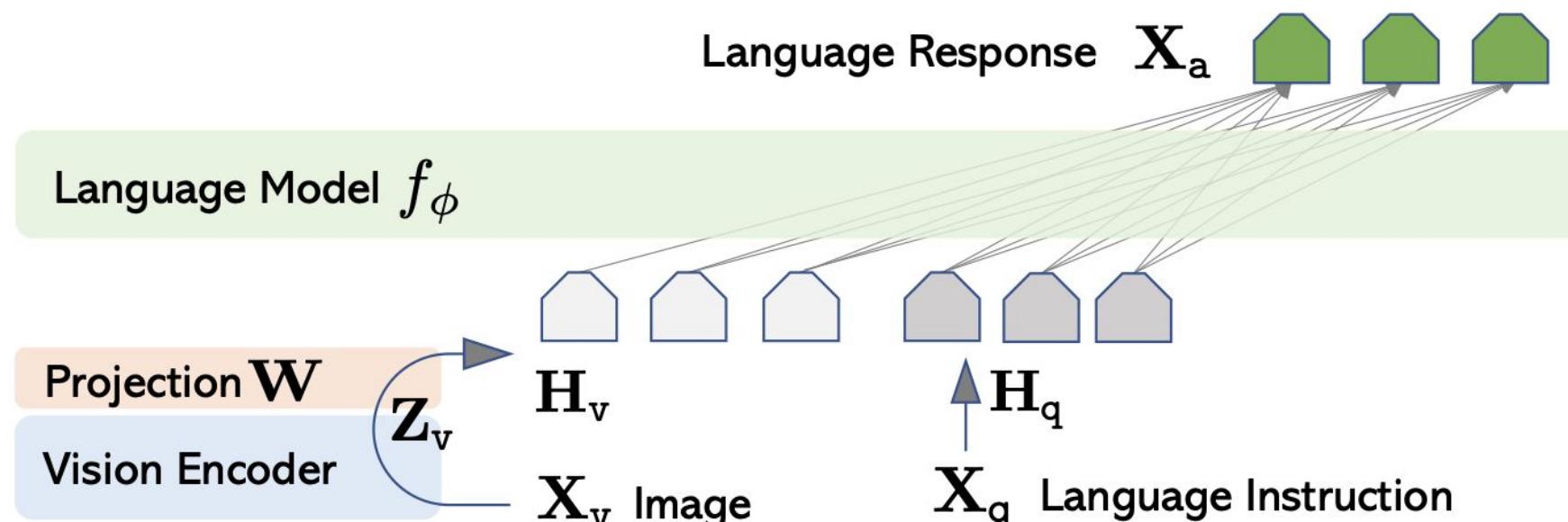
Se generó un total de 158k muestras únicas de instrucciones de texto-imagen, incluyendo 58k en conversaciones, 23k en descripciones detalladas y 77k en razonamiento complejo, respectivamente.

# LLaVA

El objetivo es utilizar las capacidades de LLMs y modelos de vision preentrenados.

Se utiliza **CLIP ViT-L/14** para extraer características visuales  $Z_v = g(X_v)$  de una imagen  $X_v$  de entrada. Estas características se proyectan al espacio de word embeddings mediante una capa lineal simple y entrenable, generando tokens visuales compatibles con el modelo de lenguaje. Este esquema de proyección es liviano, facilitando la realización de experimentos rápidos y centrados en datos.

$$H_v = W \cdot Z_v, \text{ with } Z_v = g(X_v)$$



# LLaVA

[Huggingface space](#)

[Huggingface model](#)

[Github](#)

[ROCM Example](#)

# Llama-3

Llama-3 es una LLM cuya arquitectura base (Llama-2) es expandida al espacio multimodal mediante ViT vanilla (vision) y Conformer (habla).

La conexión multimodal sucede mediante adaptadores de vision y habla, similar a LLaVA, sin embargo los adaptadores consisten en capas de cross-attention que alimentan representaciones hacia la LLM.

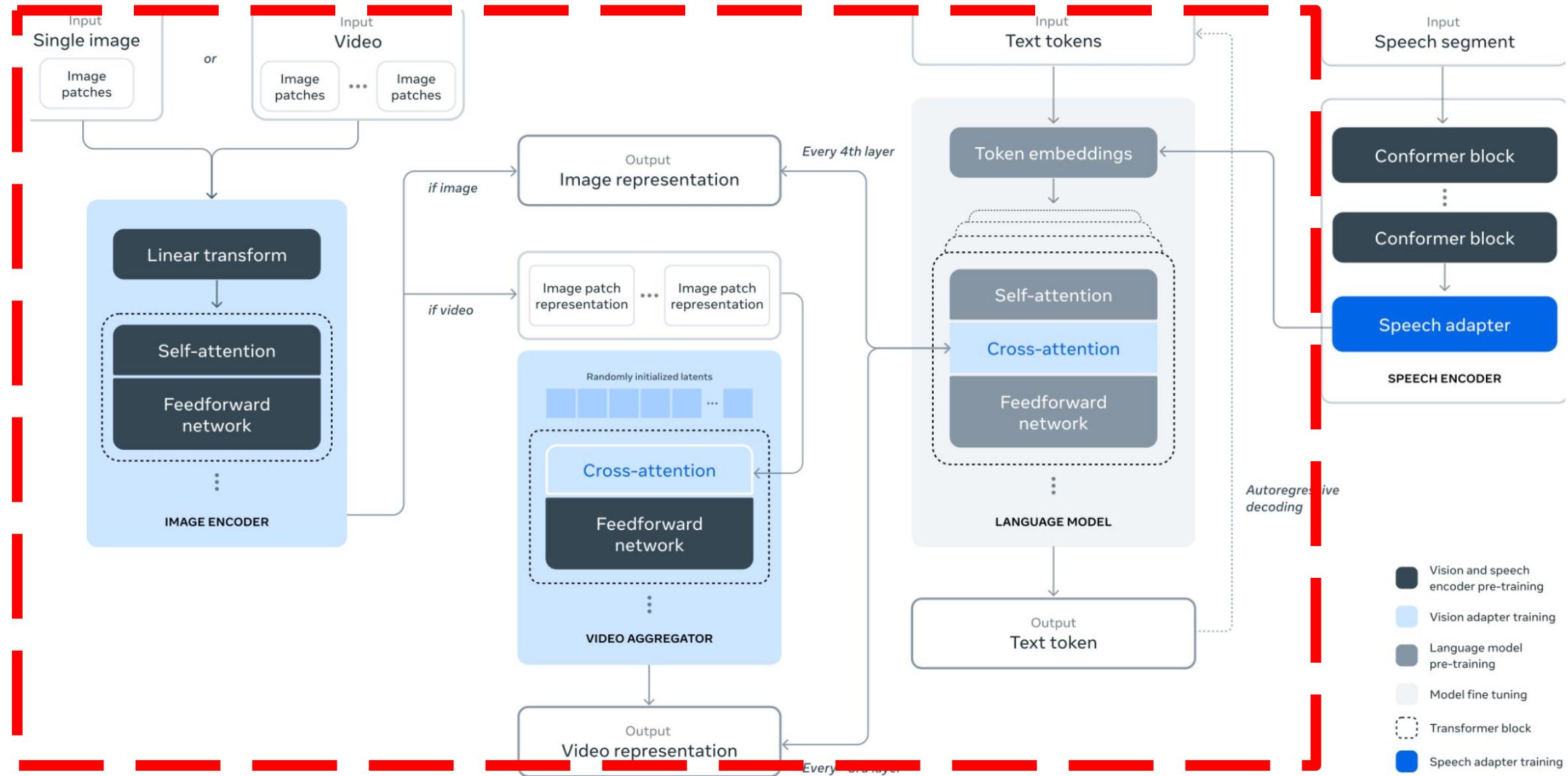
	<b>8B</b>	<b>70B</b>	<b>405B</b>
Layers	32	80	126
Model Dimension	4,096	8192	16,384
FFN Dimension	14,336	28,672	53,248
Attention Heads	32	64	128
Key/Value Heads	8	8	8
Peak Learning Rate	$3 \times 10^{-4}$	$1.5 \times 10^{-4}$	$8 \times 10^{-5}$
Activation Function		SwiGLU	
Vocabulary Size		128,000	
Positional Embeddings		RoPE ( $\theta = 500,000$ )	

# Llama-3

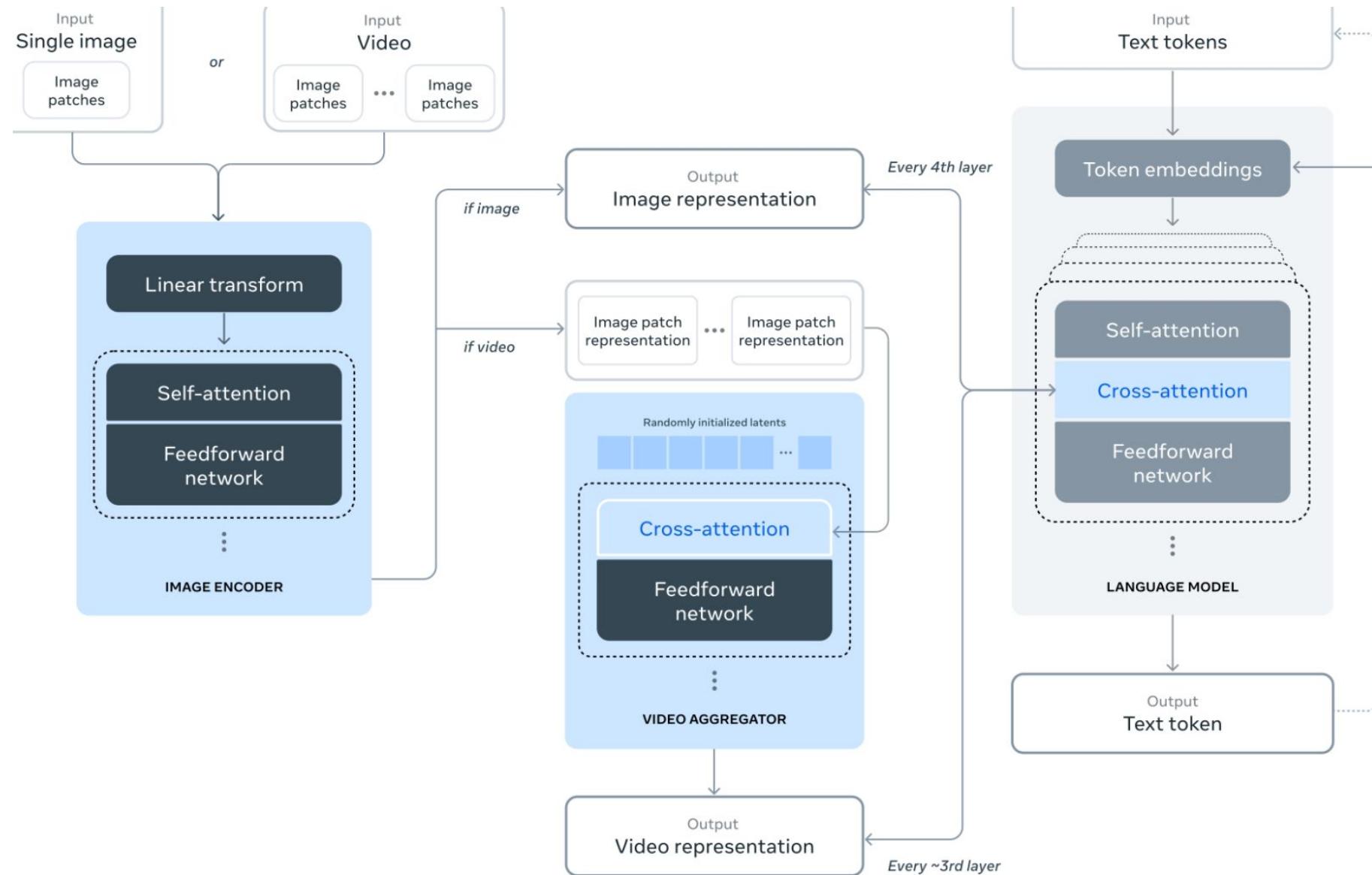
<b>GPUs</b>	<b>TP</b>	<b>CP</b>	<b>PP</b>	<b>DP</b>	<b>Seq. Len.</b>	<b>Batch size/DP</b>	<b>Tokens/Batch</b>	<b>TFLOPs/GPU</b>	<b>BF16 MFU</b>
8,192	8	1	16	64	8,192	32	16M	430	43%
16,384	8	1	16	128	8,192	16	16M	400	41%
16,384	8	16	16	4	131,072	16	16M	380	38%

	<b>8B</b>	<b>70B</b>	<b>405B</b>
Layers	32	80	126
Model Dimension	4,096	8192	16,384
FFN Dimension	14,336	28,672	53,248
Attention Heads	32	64	128
Key/Value Heads	8	8	8
Peak Learning Rate	$3 \times 10^{-4}$	$1.5 \times 10^{-4}$	$8 \times 10^{-5}$
Activation Function	SwiGLU		
Vocabulary Size	128,000		
Positional Embeddings	RoPE ( $\theta = 500,000$ )		

# Llama-3



# Llama-3 Vision



# Llama-3 Vision Encoder

Category	Benchmark	Llama 3 8B	Gemma 2 9B	Mistral 7B	Llama 3 70B	Mistral 8x22B	GPT 3.5 Turbo	Llama 3 405B	Nemotron 4 340B	GPT-4 <sub>(0125)</sub>	GPT-4o	Claude 3.5 Sonnet
General	MMLU (5-shot)	69.4	<b>72.3</b>	61.1	<b>83.6</b>	76.9	70.7	87.3	82.6	85.1	89.1	<b>89.9</b>
	MMLU (0-shot, CoT)	<b>73.0</b>	72.3 <sup>△</sup>	60.5	<b>86.0</b>	79.9	69.8	88.6	78.7 <sup>△</sup>	85.4	<b>88.7</b>	88.3
	MMLU-Pro (5-shot, CoT)	<b>48.3</b>	—	36.9	<b>66.4</b>	56.3	49.2	73.3	62.7	64.8	74.0	<b>77.0</b>
	IFEval	<b>80.4</b>	73.6	57.6	<b>87.5</b>	72.7	69.9	<b>88.6</b>	85.1	84.3	85.6	88.0
Code	HumanEval (0-shot)	<b>72.6</b>	54.3	40.2	<b>80.5</b>	75.6	68.0	89.0	73.2	86.6	90.2	<b>92.0</b>
	MBPP EvalPlus (0-shot)	<b>72.8</b>	71.7	49.5	<b>86.0</b>	78.6	82.0	88.6	72.8	83.6	87.8	<b>90.5</b>
Math	GSM8K (8-shot, CoT)	<b>84.5</b>	76.7	53.2	<b>95.1</b>	88.2	81.6	<b>96.8</b>	92.3 <sup>◇</sup>	94.2	96.1	96.4 <sup>◇</sup>
	MATH (0-shot, CoT)	<b>51.9</b>	44.3	13.0	<b>68.0</b>	54.1	43.1	73.8	41.1	64.5	<b>76.6</b>	71.1
Reasoning	ARC Challenge (0-shot)	83.4	<b>87.6</b>	74.2	<b>94.8</b>	88.7	83.7	<b>96.9</b>	94.6	96.4	96.7	96.7
	GPQA (0-shot, CoT)	32.8	—	28.8	<b>46.7</b>	33.3	30.8	51.1	—	41.4	53.6	<b>59.4</b>
Tool use	BFCL	<b>76.1</b>	—	60.4	84.8	—	<b>85.9</b>	88.5	86.5	88.3	80.5	<b>90.2</b>
	Nexus	<b>38.5</b>	30.0	24.7	<b>56.7</b>	48.5	37.2	<b>58.7</b>	—	50.3	56.1	45.7
Long context	ZeroSCROLLS/QuALITY	81.0	—	—	90.5	—	—	<b>95.2</b>	—	<b>95.2</b>	90.5	90.5
	InfiniteBench/En.MC	65.1	—	—	78.2	—	—	<b>83.4</b>	—	72.1	82.5	—
	NIH/Multi-needle	98.8	—	—	97.5	—	—	98.1	—	<b>100.0</b>	<b>100.0</b>	90.8
Multilingual	MGSM (0-shot, CoT)	<b>68.9</b>	53.2	29.9	<b>86.9</b>	71.1	51.4	<b>91.6</b>	—	85.9	90.5	<b>91.6</b>

# Llama-3 Vision

[ROCM Example](#)

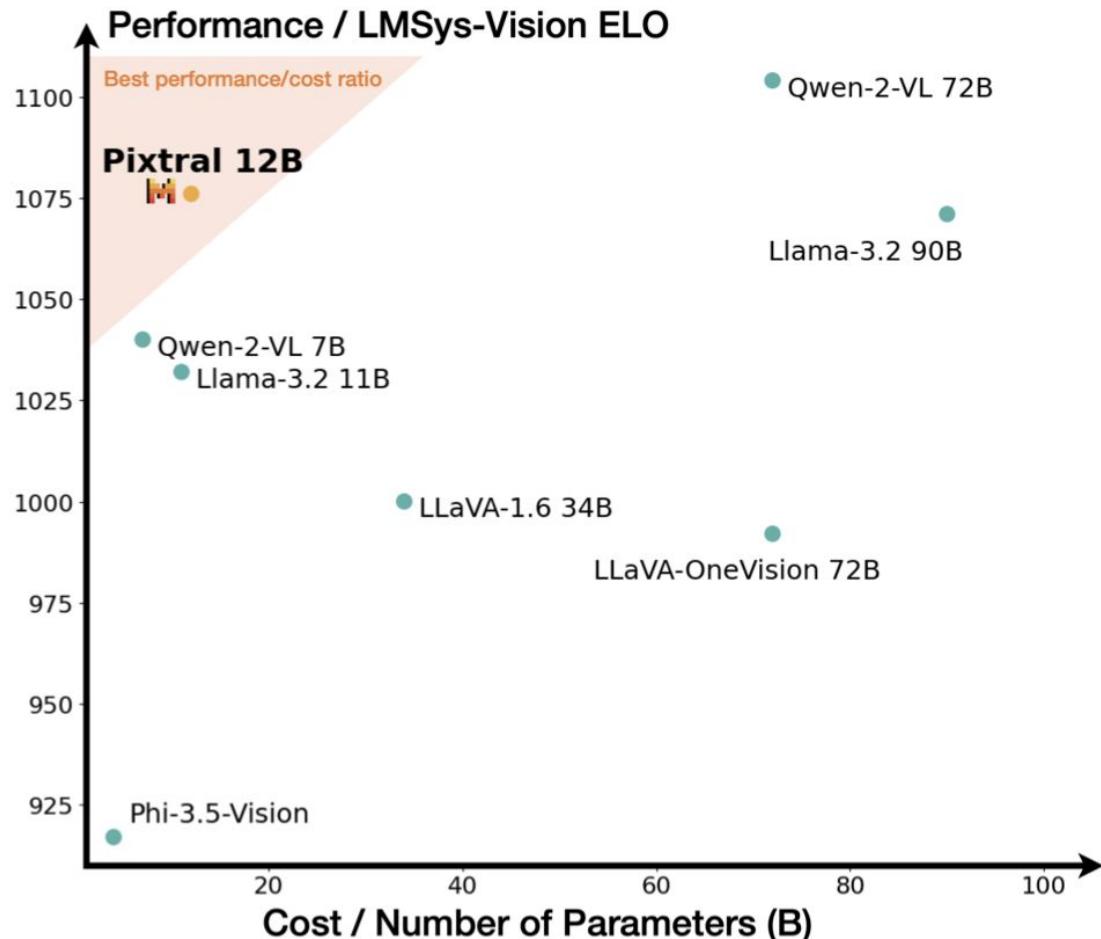
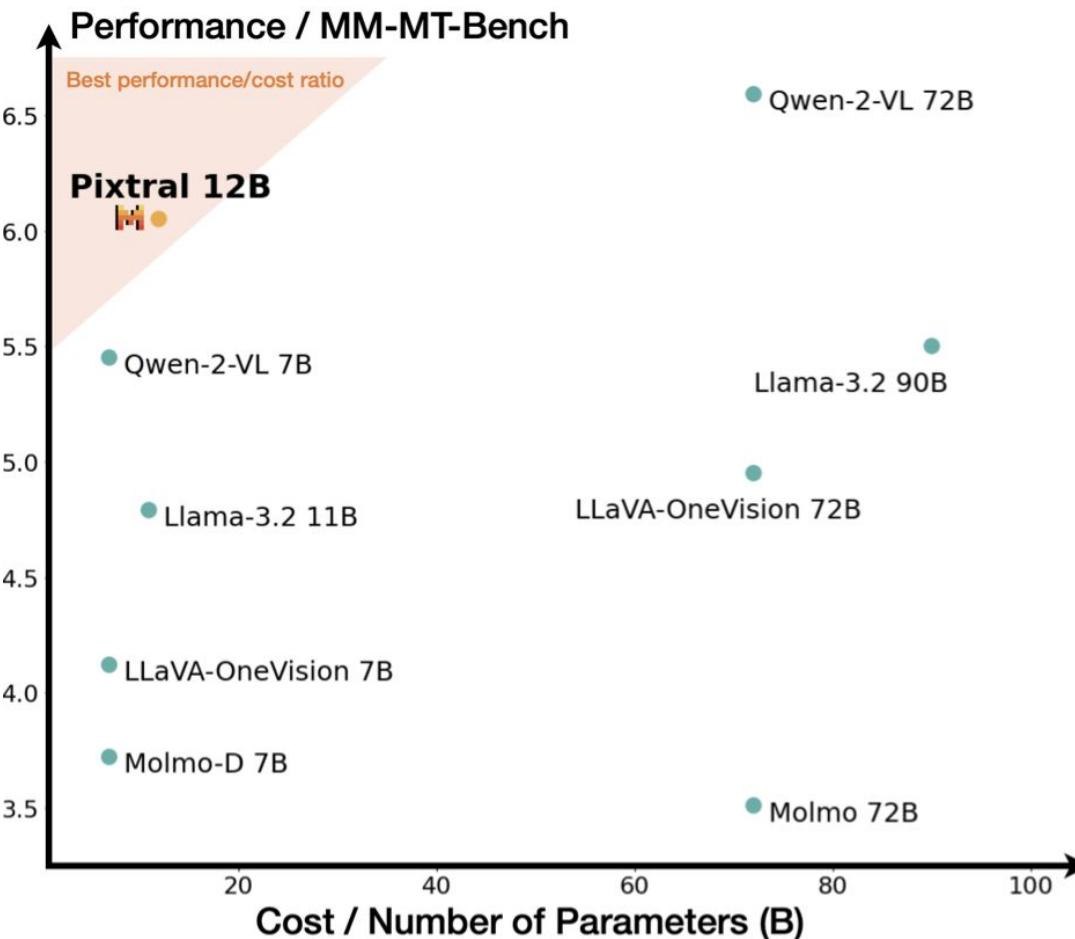
[Huggingface model](#)

[Github](#)

[Llama-3.2](#)

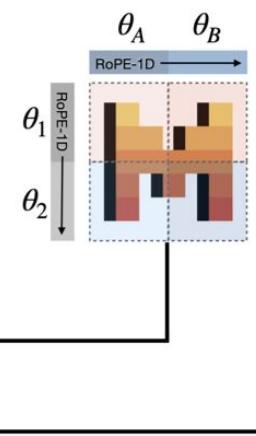
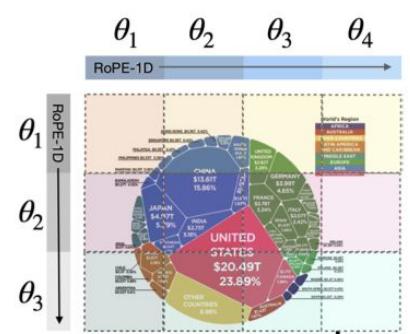
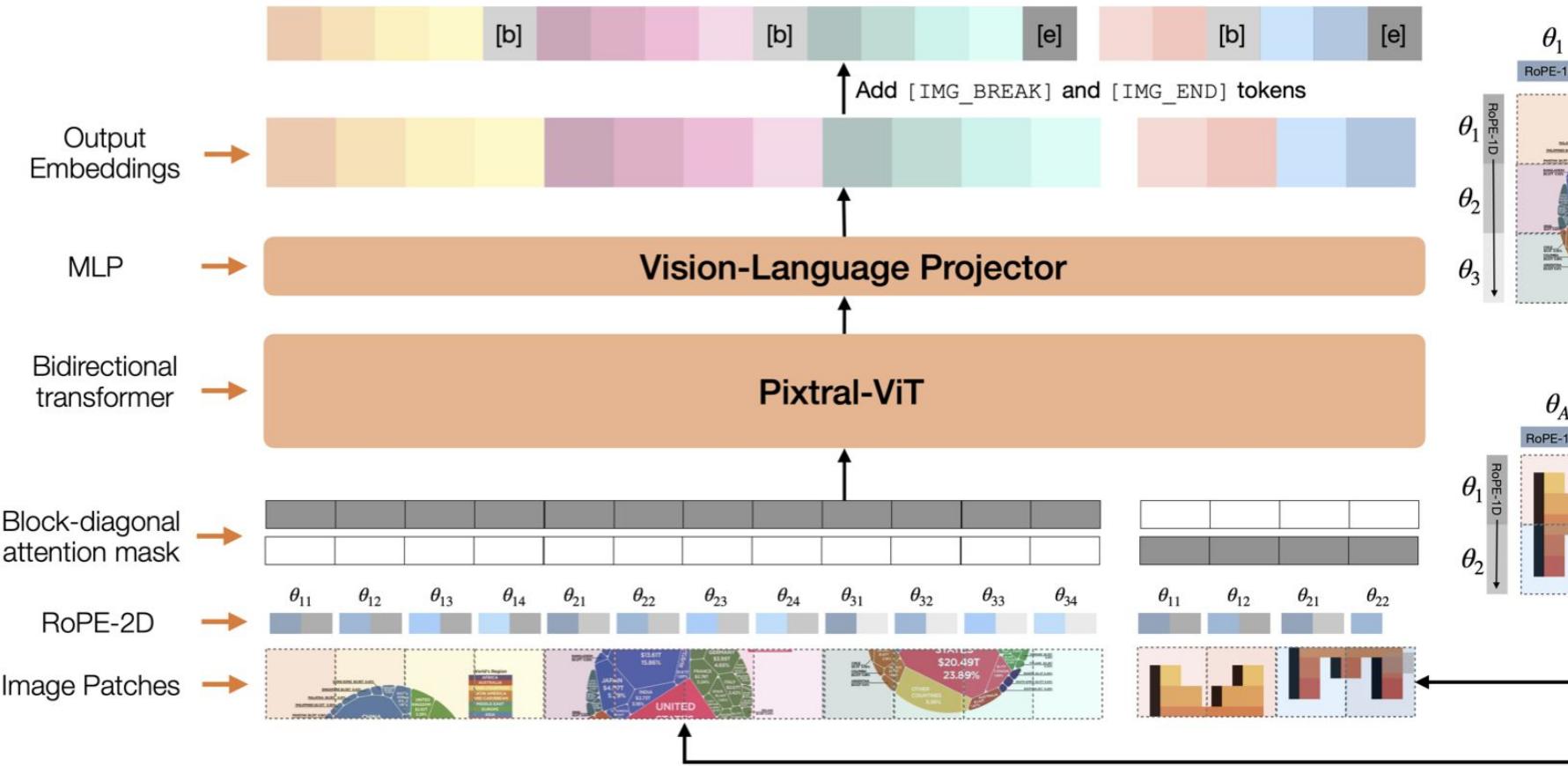
# Pixtral

[Pixtral 12B](#), fue presentado en Octubre 2024 como un modelo multimodal, utiliza un vision encoder entrenado desde cero. Supera a Llama-3 90B mientras es 7 veces más pequeño.



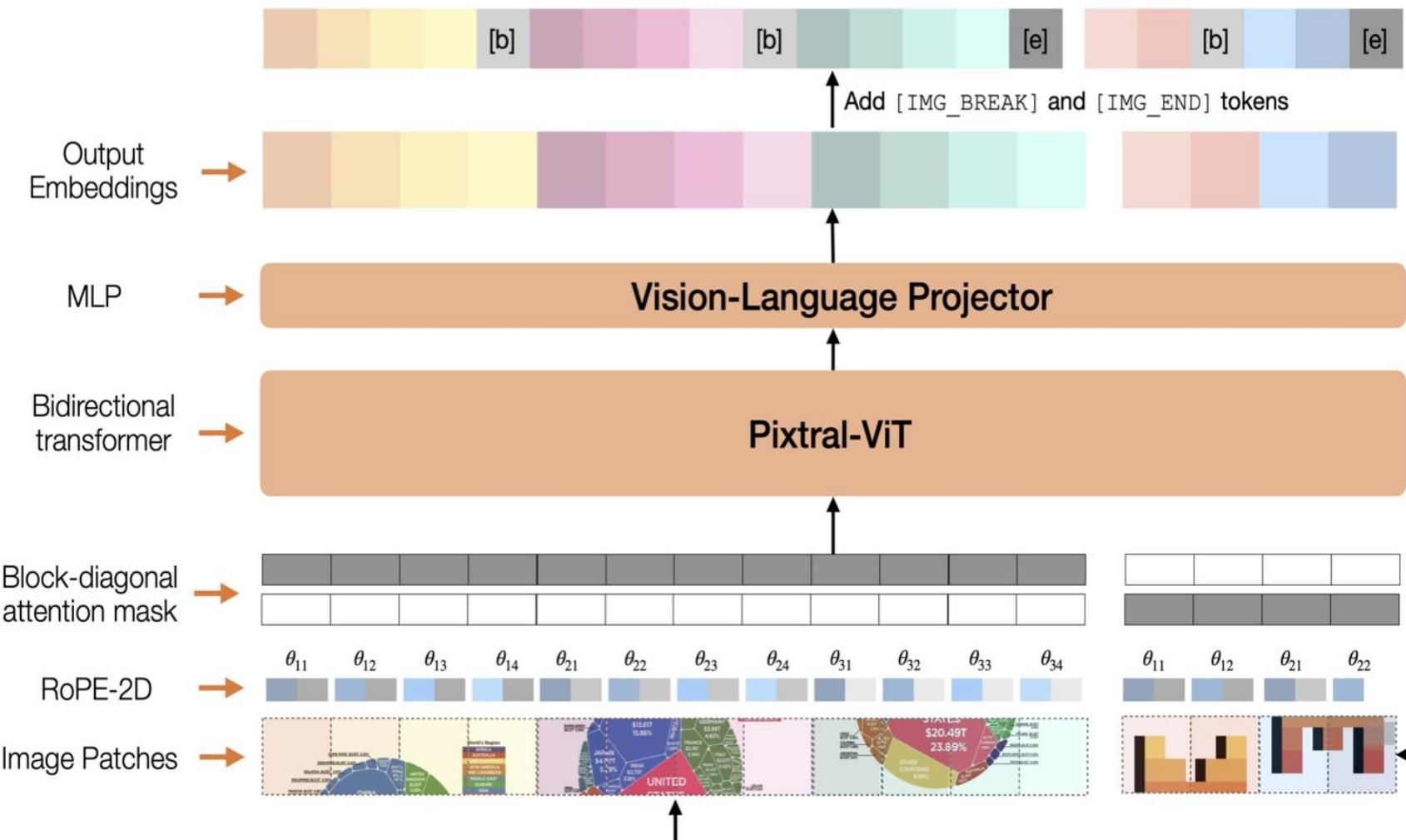
# Pixtral

Pixtral consiste en una variante ViT de 400M de parámetros.



Parameters	Decoder	Encoder
dim	5120	1024
n_layers	40	24
head_dim	128	64
hidden_dim	14336	4096
n_heads	32	16
n_kv_heads	8	16
context_len	131072	4096
vocab_size	131072	-
patch_size	-	16

# Pixtral



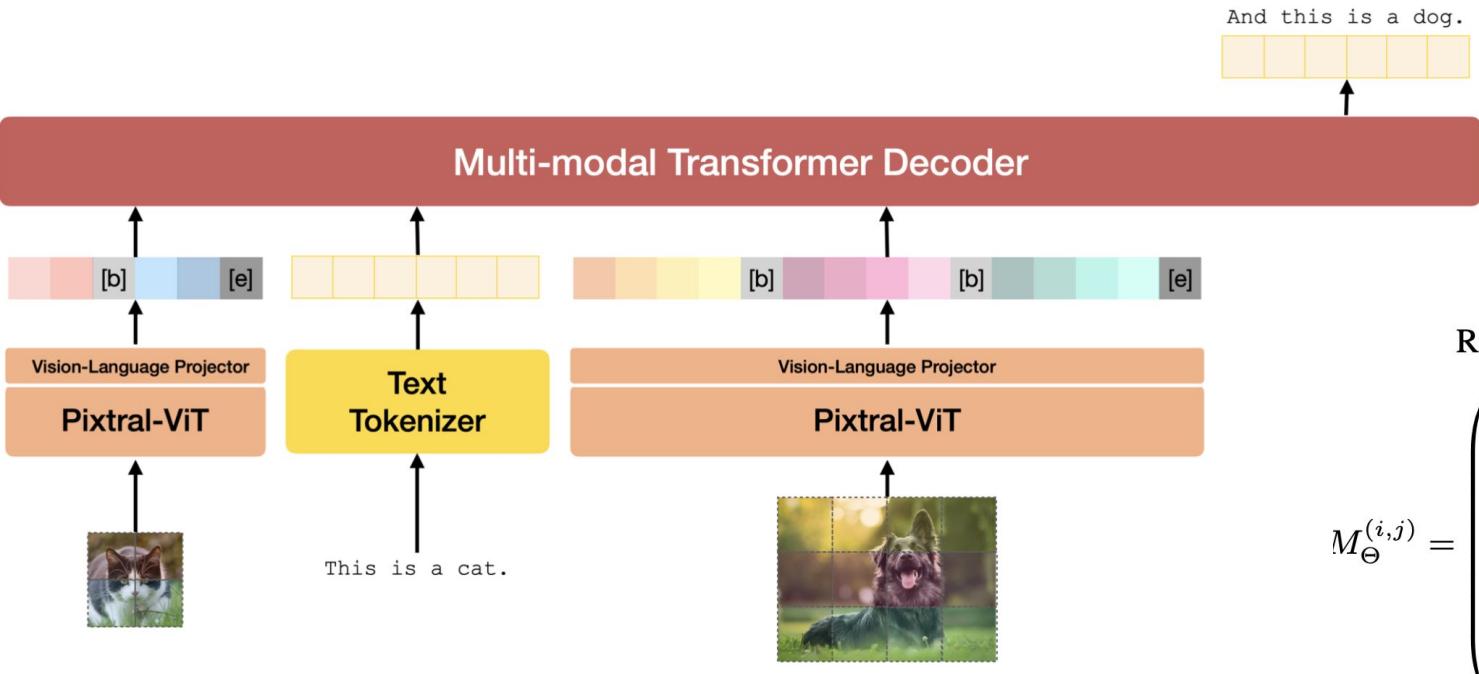
**Break tokens:** Para distinguir imágenes con el mismo número de parches pero distintas proporciones, se agregan tokens **[IMAGE BREAK]** entre filas y un token **[IMAGE END]** al final de la secuencia.

**Gating FFN:** En lugar de una capa FFN estándar en el **bloque de attention**, se utiliza Gating.

**Sequence Packing:** Para procesar eficientemente, se aplana y concatenan las imágenes. Se aplica una máscara diagonal para evitar fugas de atención entre parches.

**RoPE-2D:** Se utiliza *rotatory positional encoding* para adaptar a tamaños de imagen variables.

# Pixtral



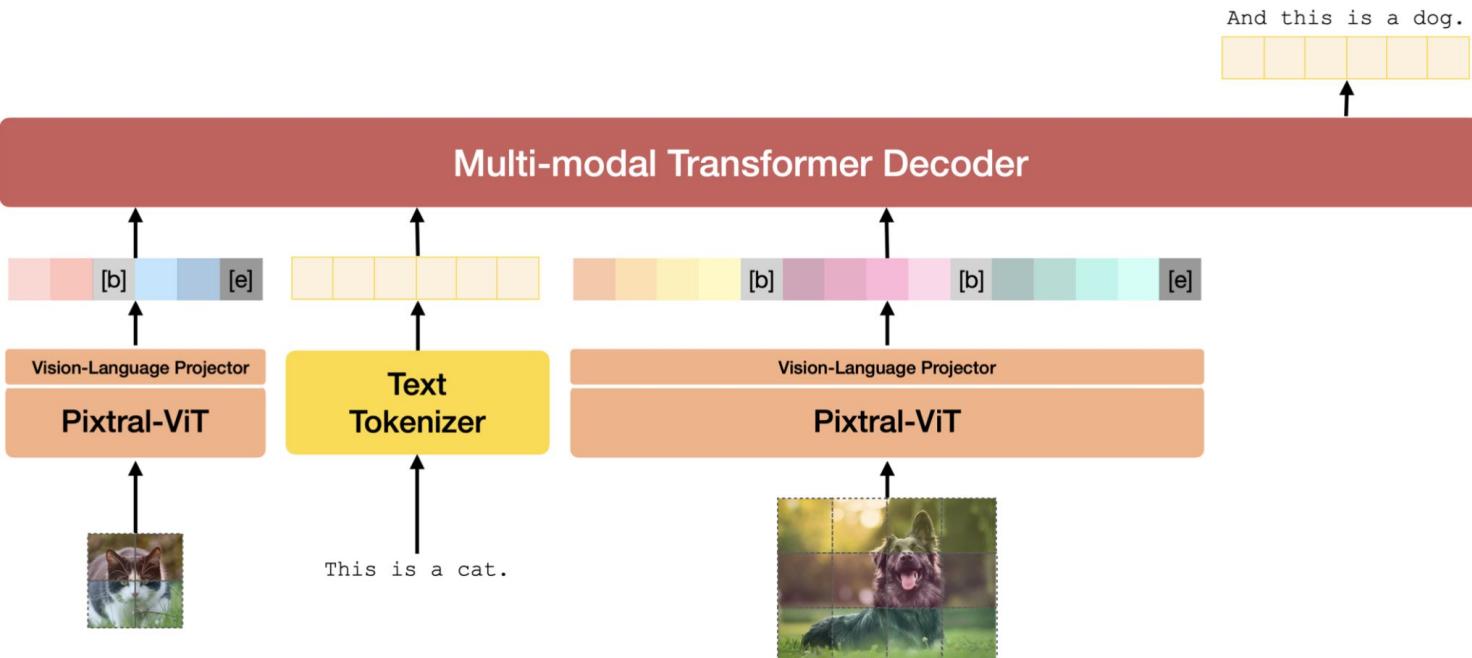
Sea  $X^{(i,j)}$  un vector de parches d-dimensional en la posición  $(i, j)$  de la imagen, entonces ROPE-2D transforma of  $X^{(i,j)}$  de la siguiente manera:

$$\text{RoPE-2D} \left( x^{(i,j)}, \Theta \right) = M_{\Theta}^{(i,j)} x^{(i,j)},$$

$$M_{\Theta}^{(i,j)} = \begin{pmatrix} \cos i\theta_1 & -\sin i\theta_1 & 0 & 0 & \cdots & 0 & 0 \\ \sin i\theta_1 & \cos i\theta_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos j\theta_2 & -\sin j\theta_2 & \cdots & 0 & 0 \\ 0 & 0 & \sin j\theta_2 & \cos j\theta_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos j\theta_{\frac{d}{2}} & -\sin j\theta_{\frac{d}{2}} \\ 0 & 0 & 0 & 0 & \cdots & \sin j\theta_{\frac{d}{2}} & \cos j\theta_{\frac{d}{2}} \end{pmatrix}$$

Las matrices  $M^{(i,j)}[k : k + 2, k : k + 2]$  capturan la posiciones altura y anchura, esto permite adaptarse de manera dinámica a distintas resoluciones.

# Pixtral



El ViT es enlazado con la LLM mediante una FFN-GeLU, transforma el output hacia el input del decoder, esto permite tratar a los tokens de imágenes como tokens de texto

# Pixtral

	<b>Mathvista</b> CoT	<b>MMMU</b> CoT	<b>ChartQA</b> CoT	<b>DocVQA</b> ANLS	<b>VQAv2</b> VQA Match	<b>MM-MT-Bench</b> GPT-4o Judge	<b>LMSys-Vision</b> (Oct '24)
<b>Pixtral 12B</b>	<b>58.3</b>	<b>52.0</b>	<b>81.8</b>	90.7	<b>78.6</b>	<b>6.05</b>	1076
Qwen-2-VL 7B [23]	53.7	48.1	41.2	<b>94.5</b>	75.9	5.45	1040
→ w/ Flexible Parsing	55.2	48.7	77.5	—	—	—	—
Llama-3.2 11B [6]	24.3	23.0	14.8	91.1	67.1	4.79	1032
→ w/ Flexible Parsing	47.9	45.3	78.5	—	—	—	—
Molmo-D 7B [4]	12.3	24.3	27.0	72.2	57.1	3.72	—
LLaVA-OneVision 7B [9]	36.1	45.1	67.2	90.5	78.4	4.12	—
Claude-3 Haiku [1]	44.8	50.4	69.6	74.6	68.4	5.46	1000
Gemini-1.5-Flash 8B(0827) [18]	<b>56.9</b>	50.7	78.0	79.5	65.5	5.93	<b>1111</b>
Molmo 72B [4]	52.2	52.7	75.6	86.5	75.2	3.51	—
LLaVA-OneVision 72B [9]	57.2	54.4	66.9	91.6	83.8	4.95	992
Qwen-2-VL 72B [23]	68.2	60.3	66.6	96.3	81.6	6.59	1104
Llama-3.2 90B [6]	49.1	53.7	33.8	85.7	67.0	5.50	1071
GPT-4o (0513) [16]	64.6	68.6	85.1	88.9	77.8	7.72	1208
Claude-3.5 Sonnet [1]	64.4	68.0	87.6	90.3	70.7	7.50	1189

**MT-Bench**    **MMLU**    **Math**    **HumanEval**

5-shot    Maj@1    Pass@1

<b>Pixtral 12B</b>	<b>7.68</b>	<b>69.2</b>	48.1	<b>72.0</b>
LLaVA-OneVision 7B [9]	6.94	67.9	38.6	65.9
Molmo-D 7B [4]	4.53	61.2	10.2	3.7
Qwen-2-VL 7B [23]	6.41	68.5	27.9	62.2
Llama-3.2 11B [6]	7.51	68.5	<b>48.3</b>	62.8

# Pixtral

[Github](#)

[Huggingface Model](#)

**Input Image**



**Question**

Can you tell me if either of these two insects, or maybe even both, are harmful to potatoes?

Options:

- A. Neither are pest of potato
- B. The one with black coloured antennae
- C. The one with tan coloured antennae
- D. Both are pests of potato

**Naive prompt (VLMEvalKit)**

```
<QUESTION>
Please select the correct answer from the
options above.
```

**Explicit prompt (ours)**

```
<QUESTION>
...
Answer in this format:
Final Answer: <answer>
where <answer> is:
- The single correct letter choice A, B,
C, D, E, F, etc. when options are
provided.
Only include the letter.
```

**GPT-4o response**

...the correct answer is:  
D. Both are pests of potato

**Exact Match Metric**

```
re.search(
    answer,
    "Final Answer: D"
)
```

**Incorrect!**



...Given that both beetles are  
Colorado potato beetles,  
they are both harmful to  
potatoes.

Final Answer: D

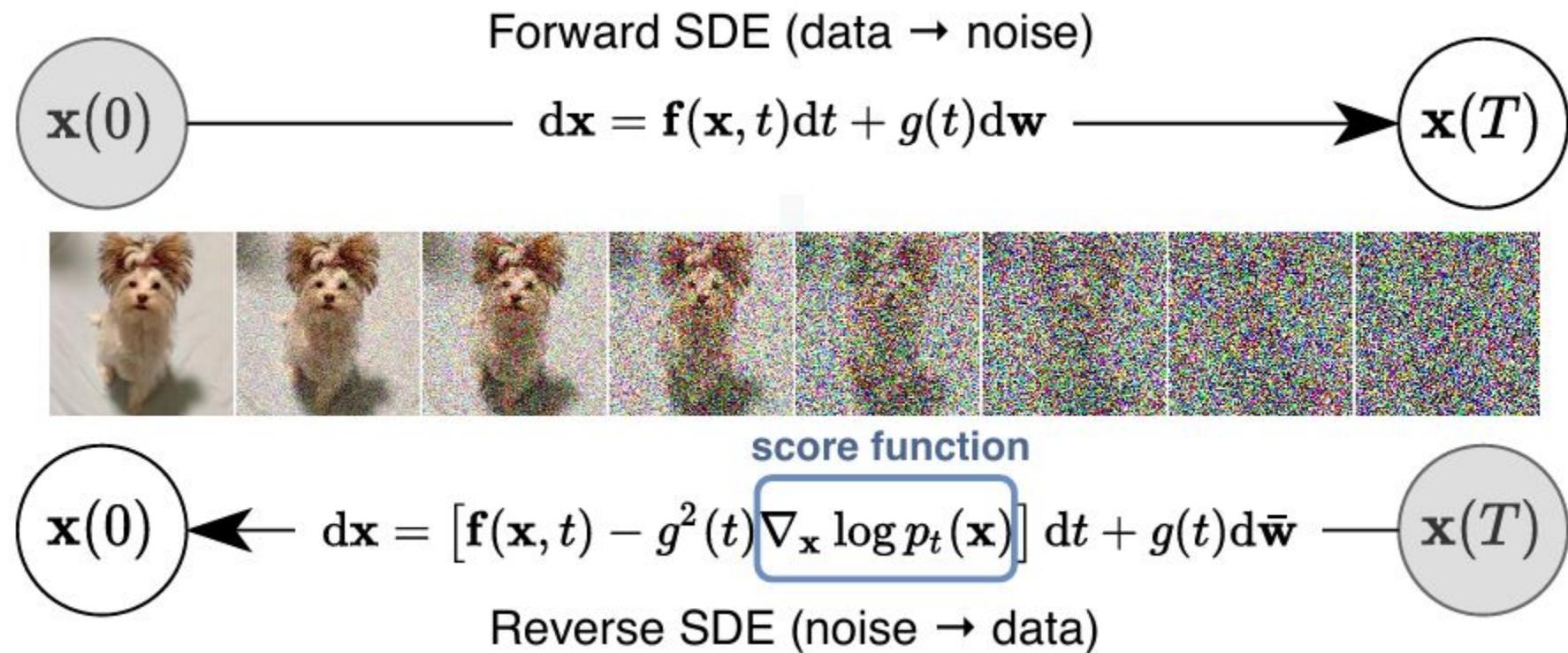
**Correct!**



# **Diffusion models**

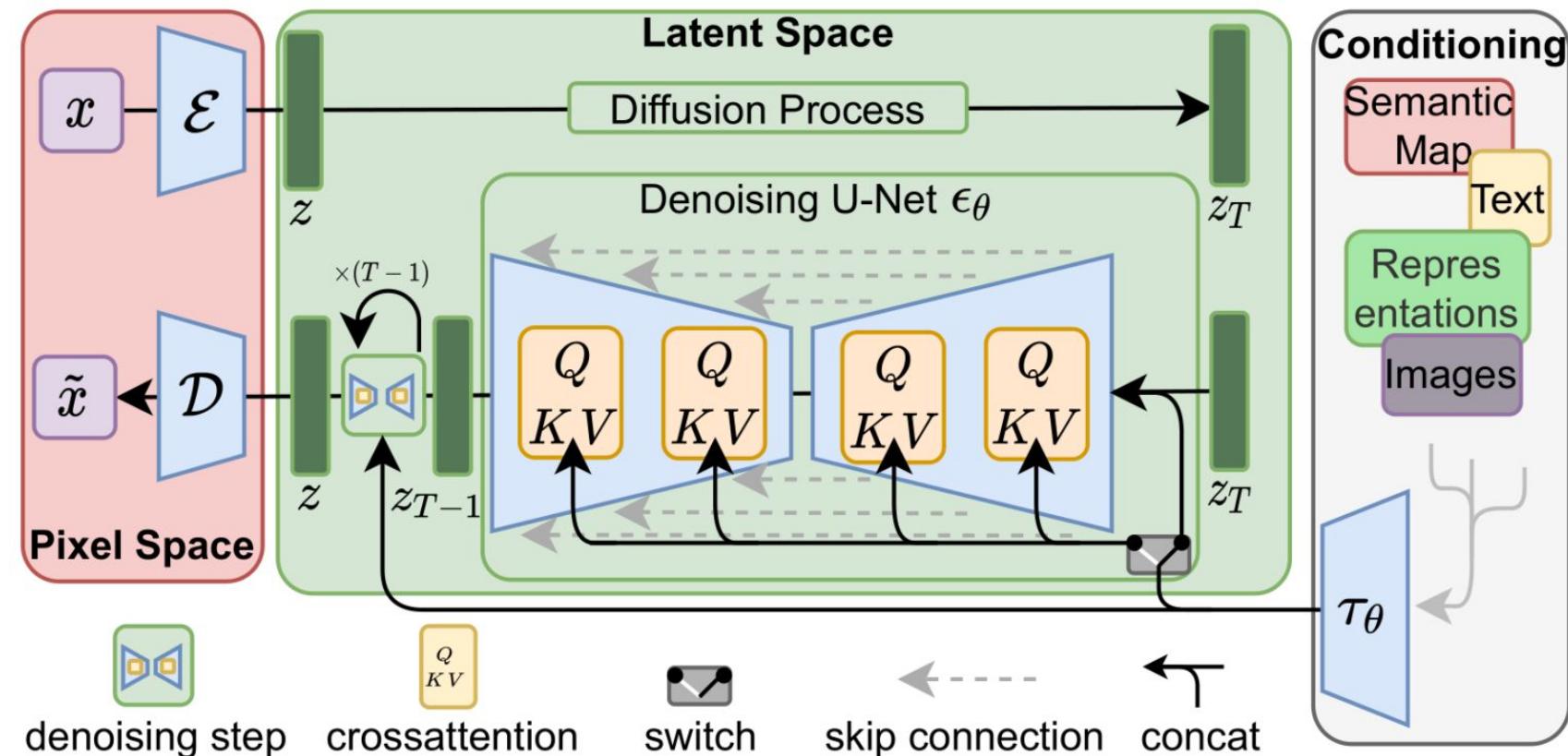
# Diffusion Models

El enfoque score-based en modelado generativo mediante ecuación diferencial estocástica (SDE) fue presentado en el paper “[SCORE-BASED GENERATIVE MODELING THROUGH STOCHASTIC DIFFERENTIAL EQUATIONS](#)”

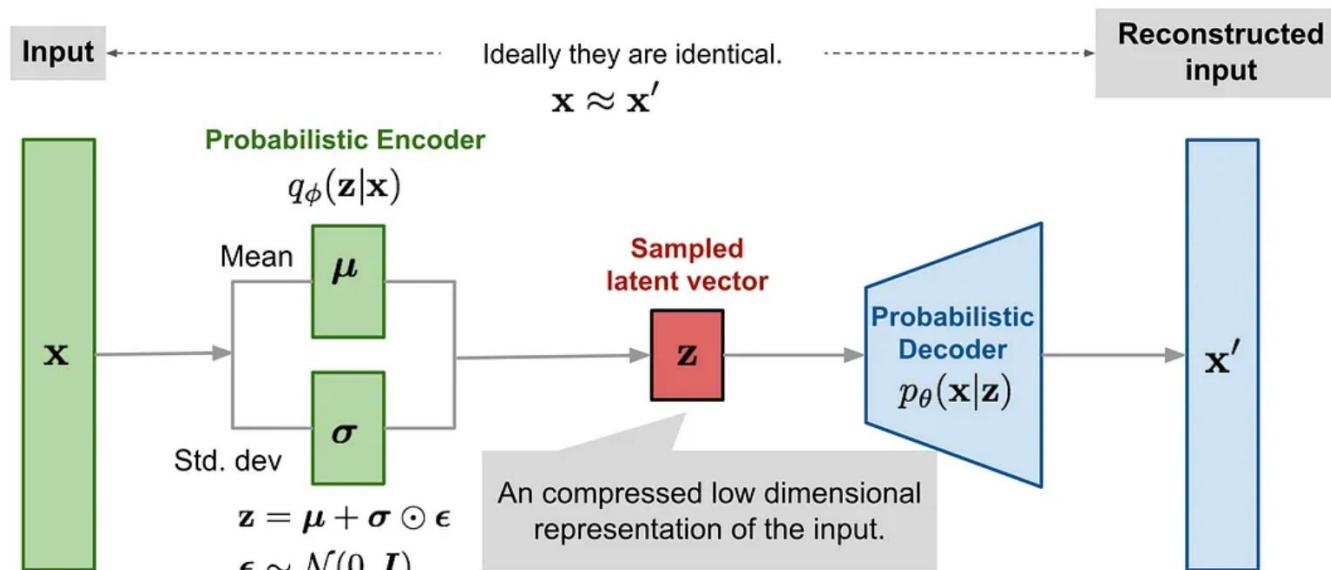


# Diffusion Models

Latent Diffusion Model (LDM) fue presentado en el paper “[High-Resolution Image Synthesis with Latent Diffusion Models](#)”



# VAE (Variational Autoencoder)



- Se trabaja con variables latentes, que son variables ocultas que afectan la distribución de los datos. Estas variables latentes se encuentran en un espacio denominado espacio latente.
- Durante el entrenamiento, se optimiza el modelo para que el codificador genere una distribución  $Q(z | X)$  que sea lo más parecida posible a la distribución posterior  $P(z | X)$ .

# Stable Diffusion

- CompVis: Latent Diffusion Model (LDM), base técnica de Stable Diffusion. ([CompVis](#))
  - [“High-Resolution Image Synthesis with Latent Diffusion Models”](#)
  - [CompVis/stable-diffusion: A latent text-to-image diffusion model](#)
  - [CompVis/stable-diffusion-v1-4 · Hugging Face](#)
- Stability AI: Financiación y recursos para entrenar el modelo en grandes datos. ([Stability AI](#))
  - [Stable Diffusion launch announcement — Stability AI](#)
  - [Stability-AI/generative-models: Generative Models by Stability AI](#)
- Runway: Implementación práctica y accesibilidad del modelo.



# Stable Diffusion 3.5 Large

- Este modelo genera imágenes a partir de indicaciones de texto. Es un Transformer de Difusión Multimodal (MMDiT) que utiliza tres codificadores de texto fijos y pre entrenados, con [normalización QK](#) para mejorar la estabilidad durante el entrenamiento.
- Encoders de texto utilizados:
  - CLIPs: OpenCLIP-ViT/G, CLIP-ViT/L, con una longitud de contexto de 77 tokens
  - T5: T5-xxl, con una longitud de contexto de 77 o 256 tokens en diferentes etapas del entrenamiento
- Este modelo fue entrenado con un conjunto diverso de datos, incluyendo datos sintéticos y datos públicos filtrados.
- [stabilityai/stable-diffusion-3.5-large · Hugging Face](#)
- [huggingface/diffusers:](#) 😊
- [Pipelines](#)

Licencia: Gratis para los investigadores, uso no comercial y uso comercial para organizaciones e individuos con ganancias anuales menores a USD \$1M.

# Stable Diffusion 3.5

Stable Diffusion 3.5 ofrece los siguientes modelos:

**LARGE**: Con 8.1 mil millones de parámetros, muy buena adherencia a las indicaciones. El más poderoso de la familia Stable Diffusion. Modelo para usos profesionales en una resolución de 1 megapíxel.

**LARGE TURBO**: Versión destilada ([ADD](#)) de “Large”, muy buena adherencia a las indicaciones en solo 4 pasos, lo que lo hace considerablemente más rápido que “Large”. Utiliza “score distillation” para aprovechar modelos grandes de difusión preexistentes como “señal maestra”, en combinación con un adversarial loss para asegurar alta fidelidad de imagen en pocos pasos.

**MEDIUM**: Con 2.5 mil millones de parámetros y arquitectura MMDiT-X. Doble bloque de atención en las primeras 12 capas (generación multi-resolución). Este modelo está diseñado para funcionar “listo para usar” en hardware de consumo (9.9 GB de VRAM). Es capaz de generar imágenes en un rango de resolución entre 0.25 y 2 megapíxeles.

Stability AI: [\(Stability AI\)](#)

- [Stable Diffusion launch announcement — Stability AI](#)
- [Stability-AI/generative-models: Generative Models by Stability AI](#)
- [Stable Diffusion pipelines](#)

# Stable Diffusion 3.5

GPU Device	VRAM (GB)	Stable Diffusion 3.5 Medium (2.5B)	SDXL (6.5B including refiner)	Playground v2.5 (3.5B)	AuraFlow v0.2 (8.7B)	Stable Diffusion 3.5 Large / Large Turbo (8.1B)	FLUX.1 [dev] (12B)	FLUX.1 [schnell] (12B)
NVIDIA GeForce RTX 4060	8	!	!	!	!	!	!	!
NVIDIA GeForce RTX 3080	10	✓	!	!	!	!	!	!
NVIDIA GeForce RTX 3060 NVIDIA GeForce RTX 4070 AMD Radeon RX 7700 XT	12	✓	!	!	!	!	!	!
NVIDIA GeForce RTX 4060 Ti NVIDIA GeForce RTX 4070 Ti NVIDIA GeForce RTX 4080 AMD Radeon RX 7800 XT AMD Radeon RX 7600 XT	16	✓	✓	✓	!	!	!	!
AMD Radeon RX 7900 XT	20	✓	✓	✓	✓	!	!	!
NVIDIA GeForce RTX 3090 NVIDIA GeForce RTX 4090 AMD Radeon 7900XTX	24	✓	✓	✓	✓	✓	!	!
NVIDIA H100 AMD Instinct MI250X AMD Instinct MI300A AMD Instinct MI300X	32 (or greater)	✓	✓	✓	✓	✓	✓	✓

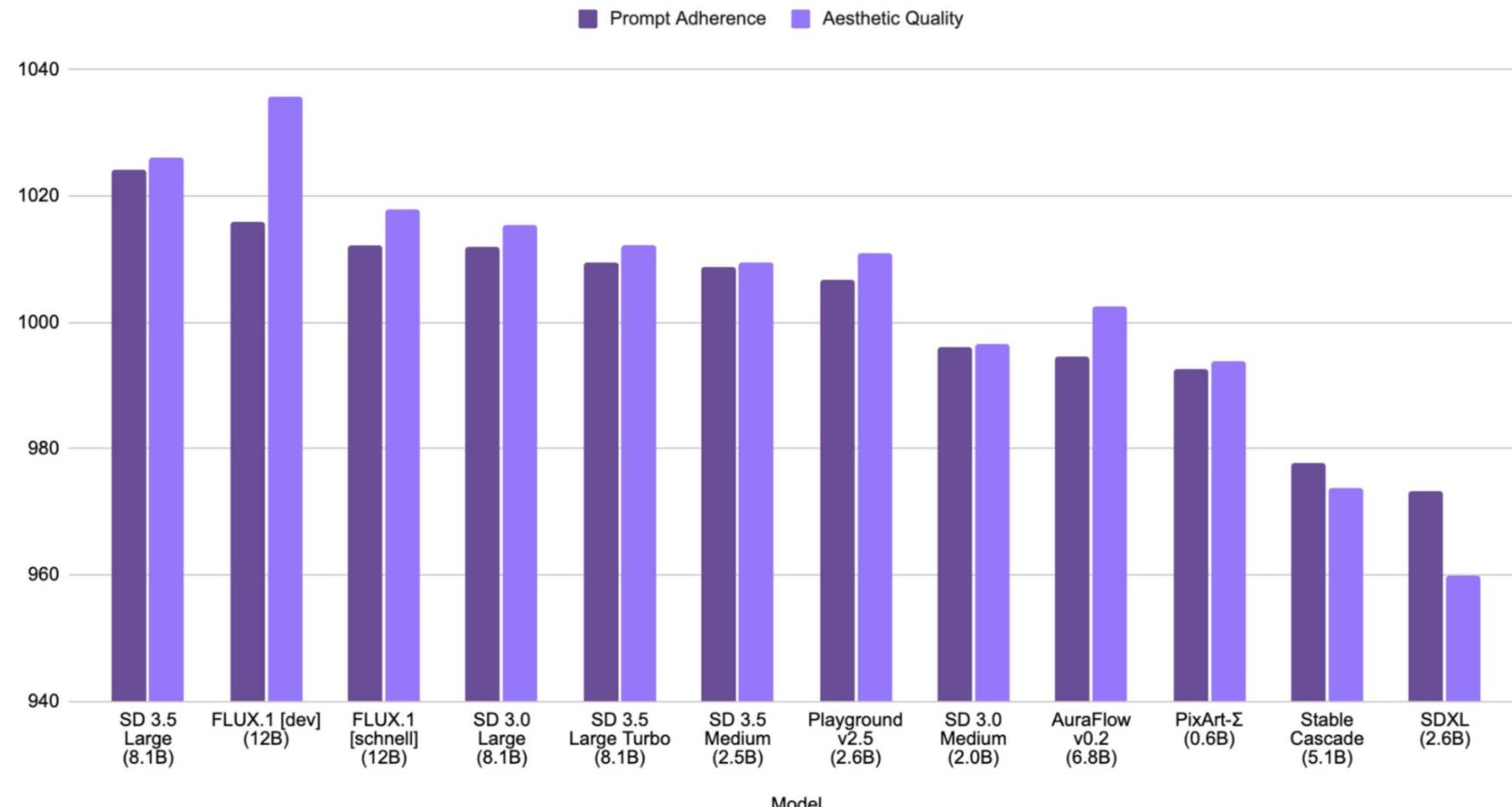
Hardware compatibility and VRAM requirements for open-image base models.

✓ indicates the model runs on this device without any performance tradeoffs.

! indicates the model requires performance-compromising optimizations, such as quantization or sequential offloading, to run on this device.

# Stable Diffusion 3.5

Prompt Adherence & Aesthetic Quality (Elo Score)



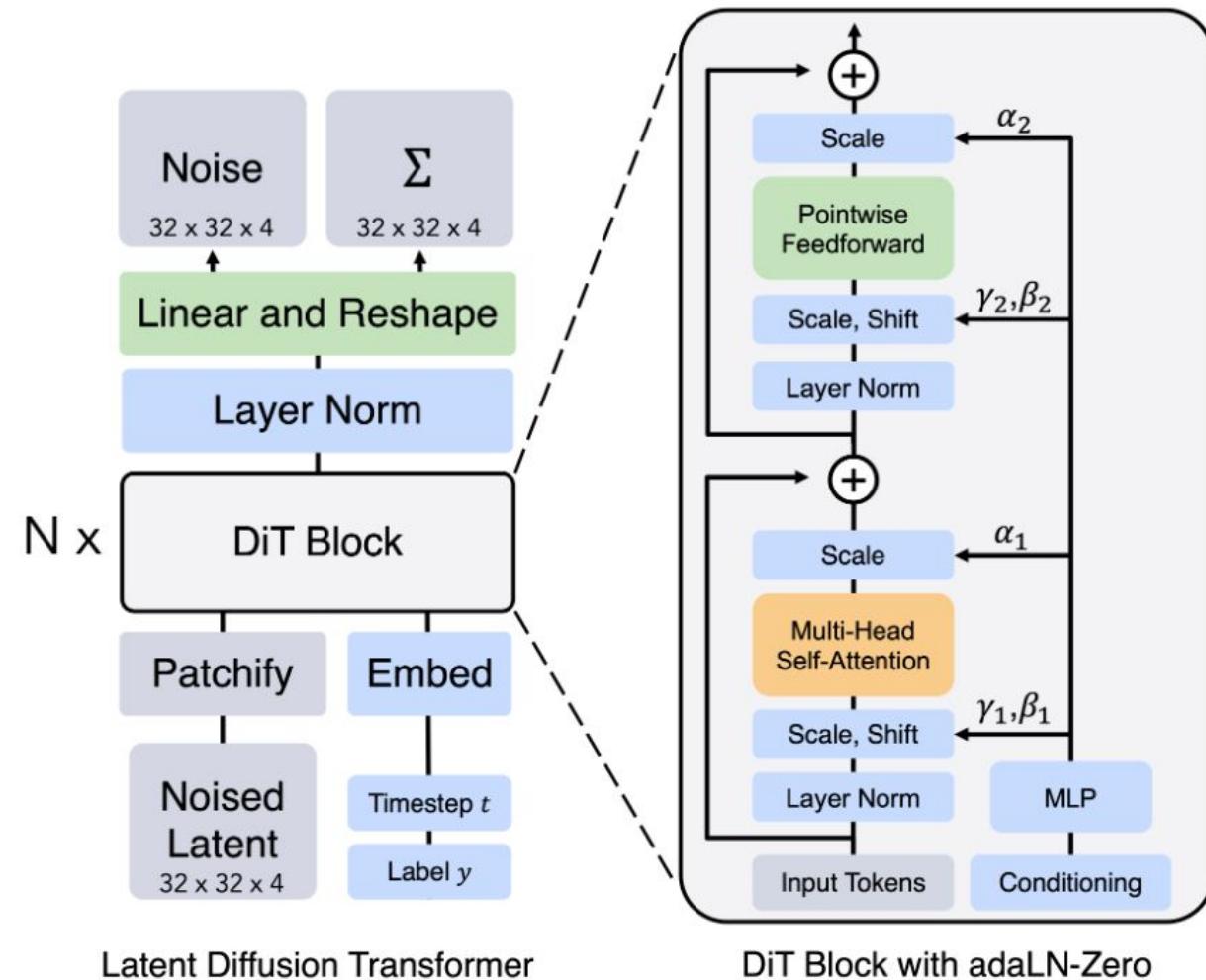
**Prompt adherence** se refiere a la capacidad de un modelo generativo para crear imágenes que coinciden fielmente con la descripción proporcionada en el "prompt"

# Stable Diffusion 3



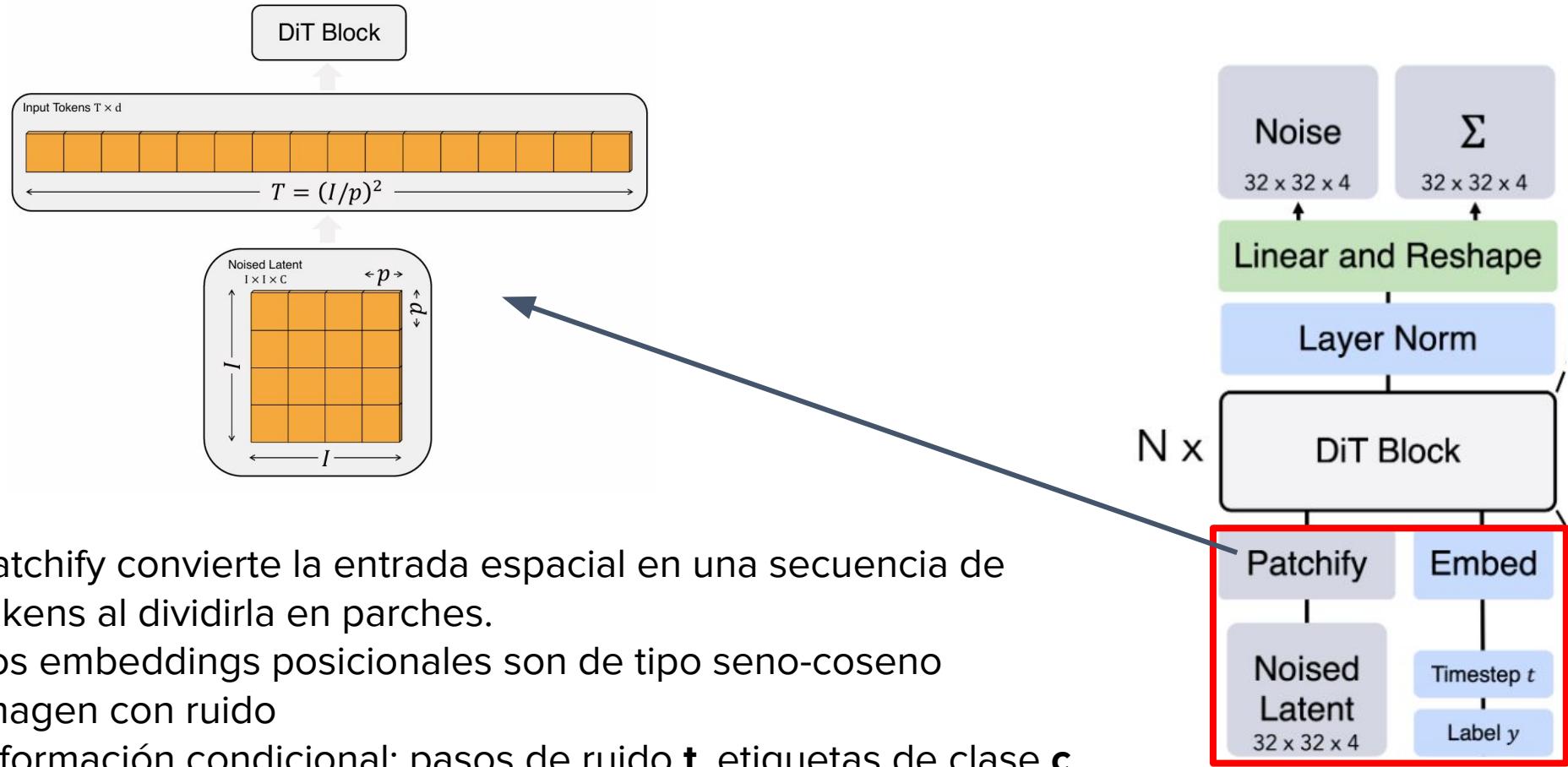
A whimsical and creative image depicting a hybrid creature that is a mix of a waffle and a hippopotamus. This imaginative creature features the distinctive, bulky body of a hippo, but with a texture and appearance resembling a golden-brown, crispy waffle. The creature might have elements like waffle squares across its skin and a syrup-like sheen. It's set in a surreal environment that playfully combines a natural water habitat of a hippo with elements of a breakfast table setting, possibly including oversized utensils or plates in the background. The image should evoke a sense of playful absurdity and culinary fantasy.

# Diffusion Transformers



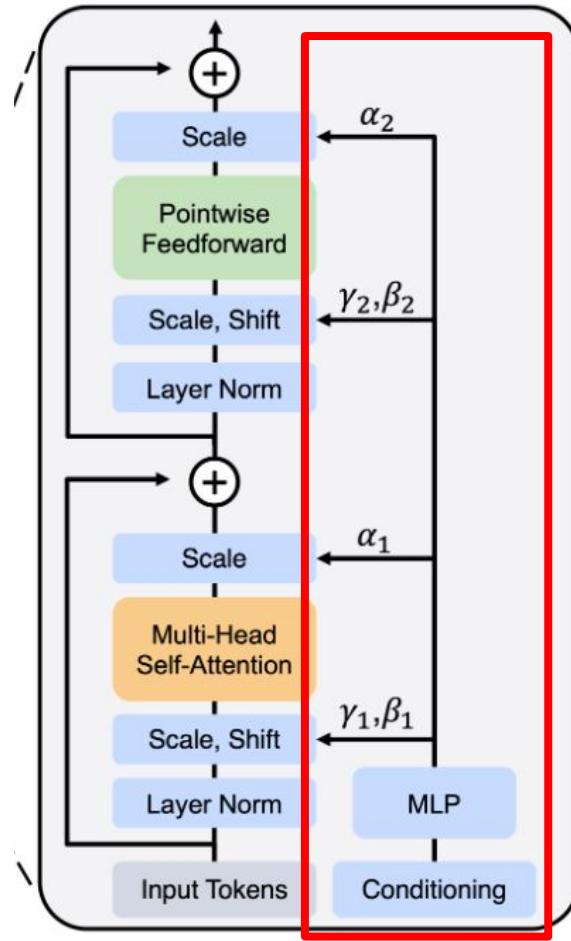
[Scalable Diffusion Models with Transformers](#)

# Diffusion Transformers



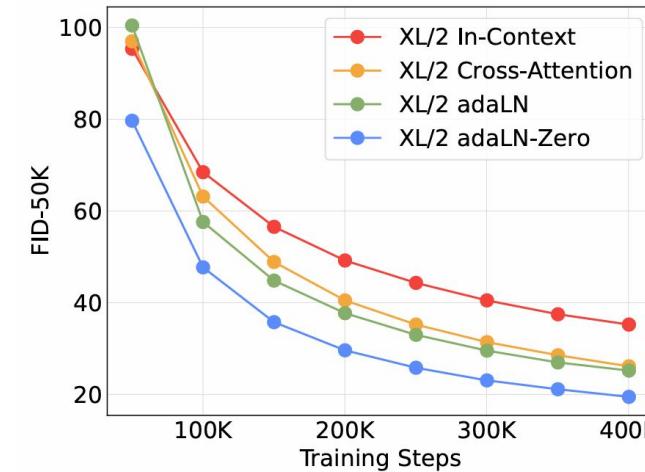
Latent Diffusion Transformer

# Diffusion Transformers



**AdaLN-Zero:** Es una variante de AdaLN donde los parámetros de escalado y desplazamiento ( $\alpha$ ,  $\beta$ ,  $\gamma$ ) se inicializan como cero al principio. Esto hace que el bloque comience como la función identidad, es decir, que no aplique ninguna transformación en las primeras etapas del entrenamiento, lo que puede acelerar el proceso de aprendizaje y evitar que el modelo aprenda transformaciones innecesarias desde el principio.

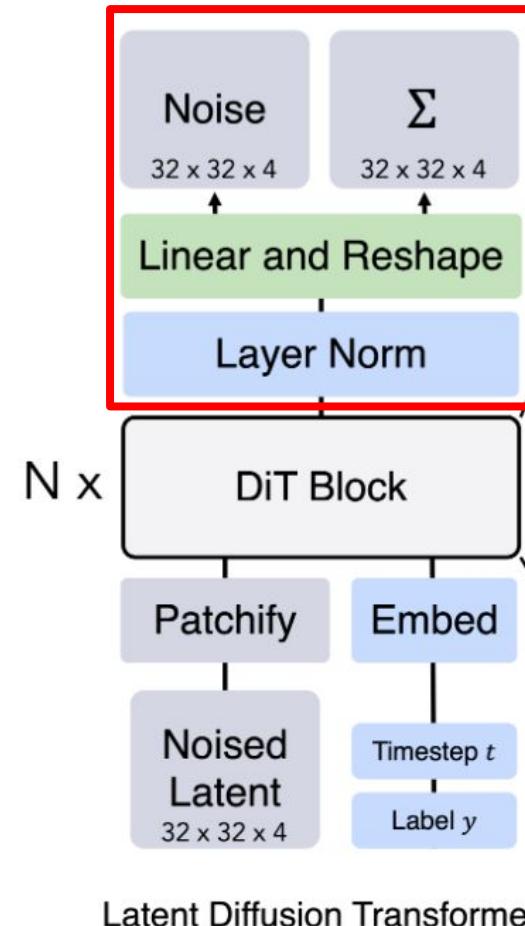
- a:** factor de escala
- b:** desplazamiento
- γ:** escalado adaptativo de la normalización
- t:** paso temporal en el proceso de difusión



**MLP (Multilayer Perceptron)**  
condiciona los parámetros  $\alpha$  y  $\gamma$  en función de las características de la entrada (como el tiempo de ruido  $t$ ) y controla cómo se ajusta la salida del bloque residual adaptativamente

**PointwiseFF:** Red neuronal donde cada entrada se procesa independientemente, sin interacciones entre ellas

# Diffusion Transformers

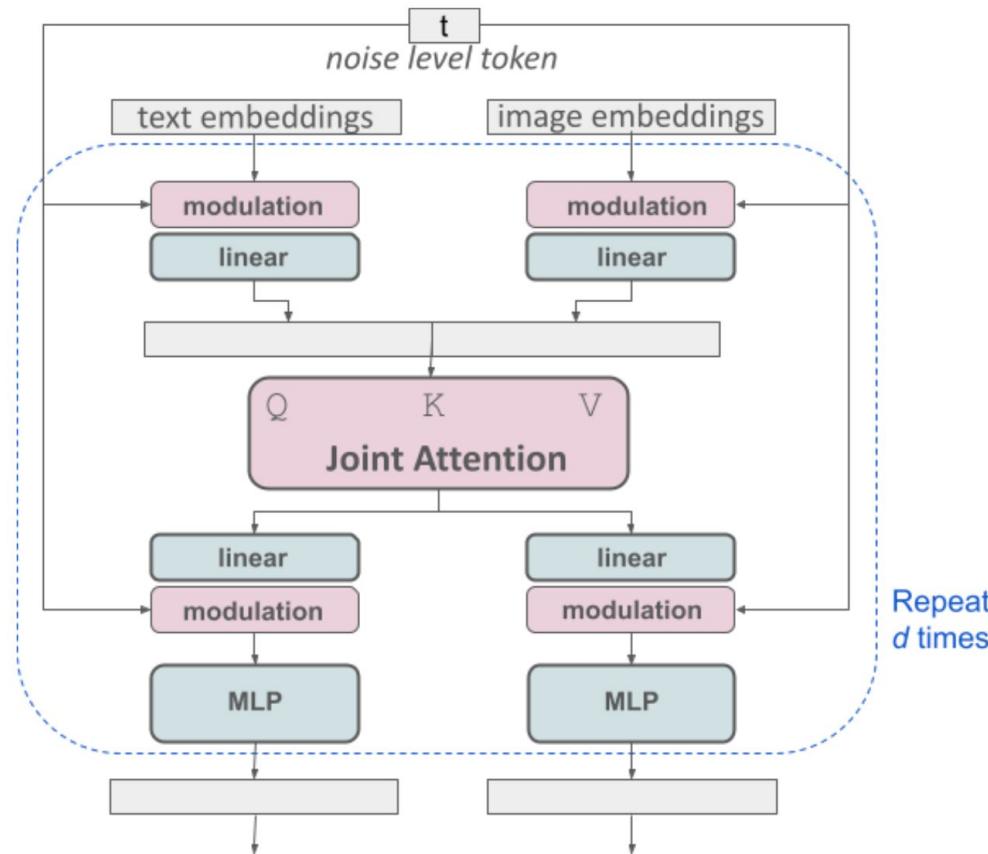


Después del último bloque DiT:

- Se decodifica la secuencia de tokens de imagen para obtener una predicción de ruido de salida y una predicción de covarianza diagonal de salida. Decodificador lineal estándar.
- Ambos resultados tienen la misma forma que la entrada espacial original.
- Normalización de capa final es adaptativa usando adaLN
- Se decodifica linealmente cada token en un tensor  $p \times p \times 2C$ , donde  $C$  es el número de canales en la entrada espacial a DiT.

# Stable Diffusion 3 - MM-DiT

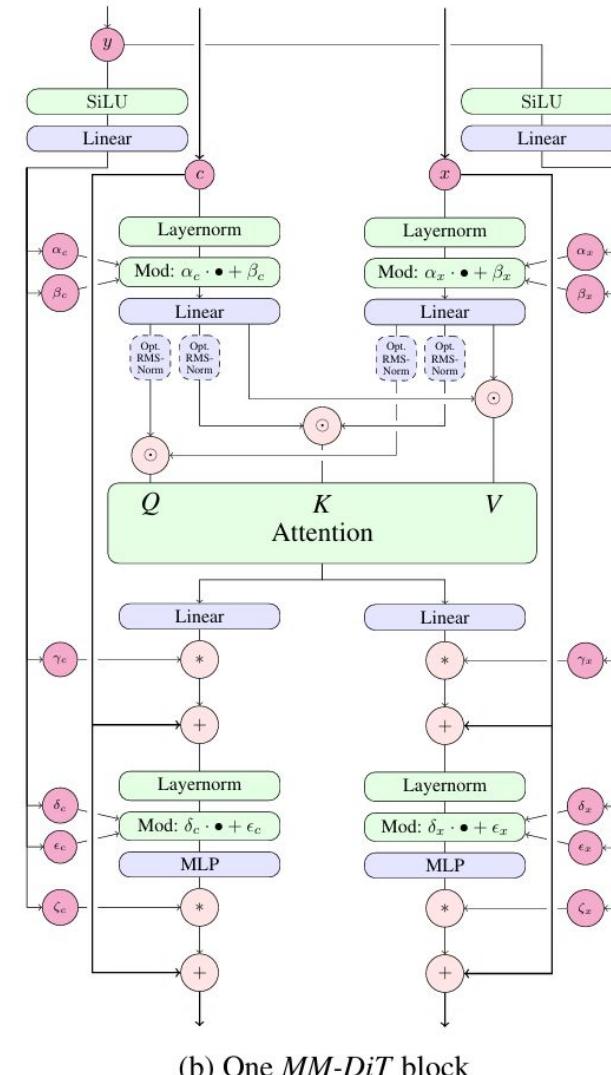
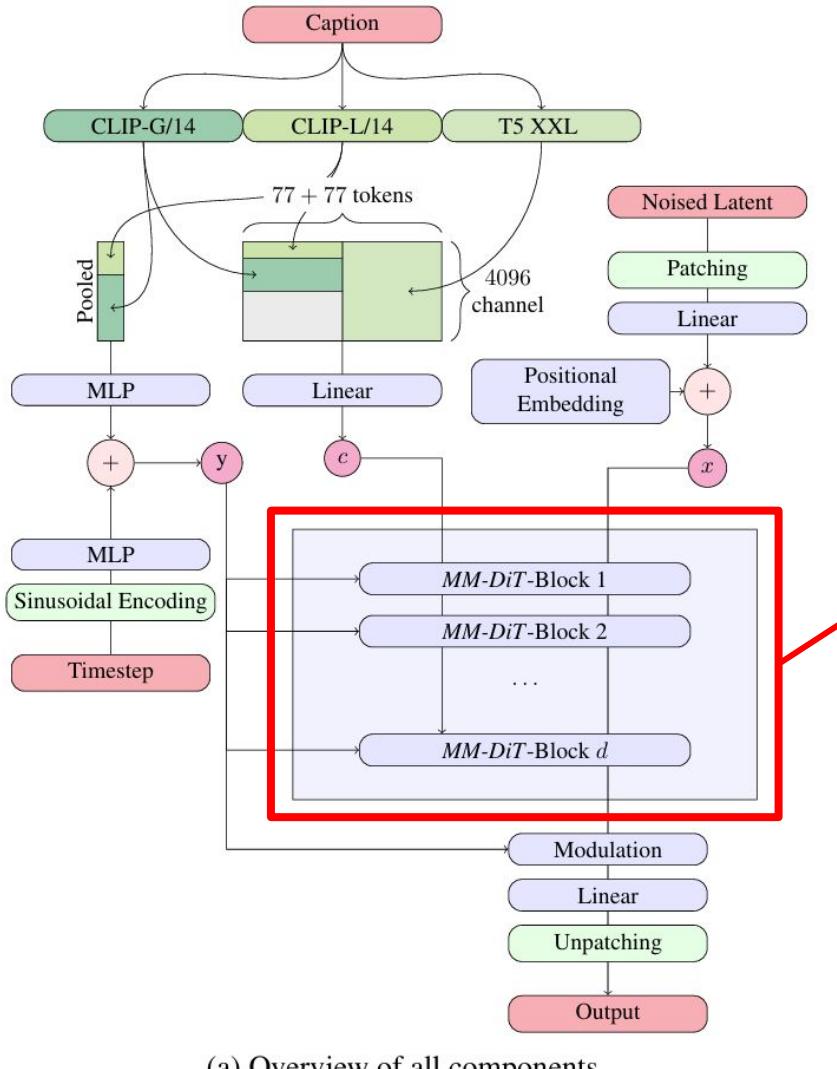
Stable Diffusion 3 es construido basado en el trabajo [Scalable Diffusion Models with Transformers](#)



La arquitectura **MM-DiT** basada en Transformers tiene en cuenta la naturaleza multimodal de la tarea de texto a imagen.

Conceptual visualization of a block of our modified multimodal diffusion transformer: MMDiT.

# Stable Diffusion 3 - MM-DiT - Large



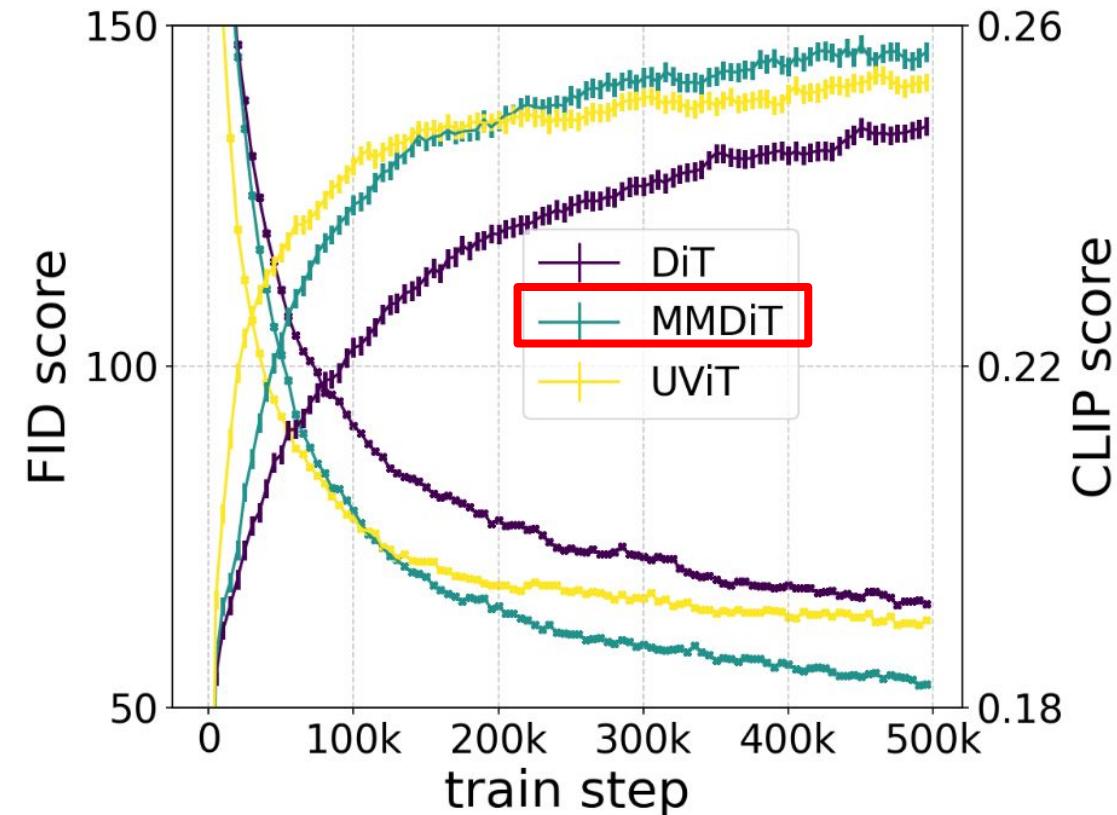
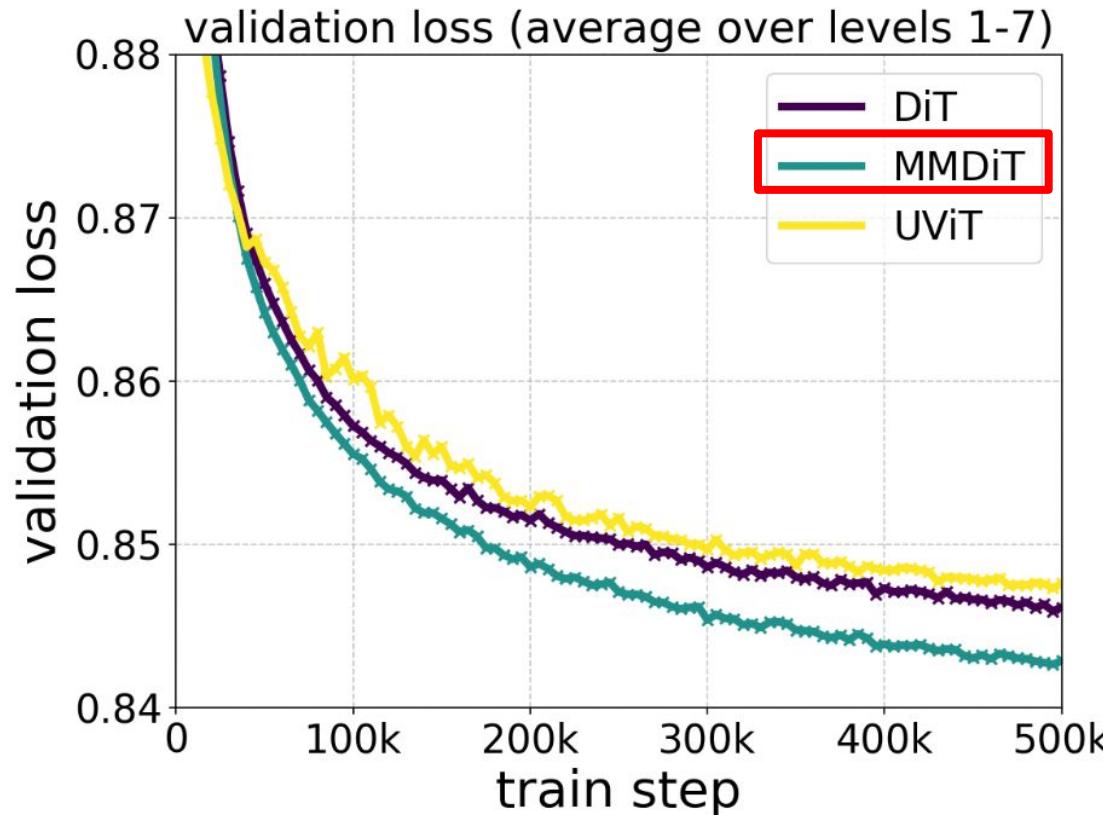
Mecanismo de modulación para condicionar la red al tiempo del proceso de difusión y la etiqueta de clase.

Construcción de secuencia combinando embeddings de texto e imagen.

Codificaciones posicionales.

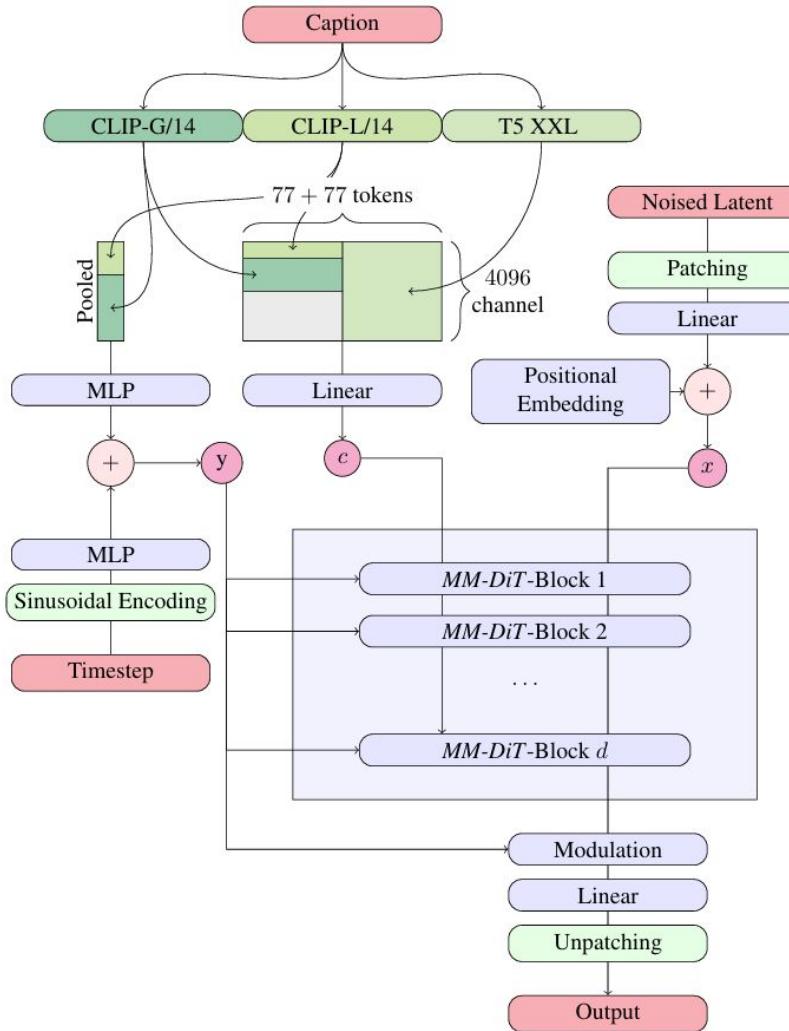
Uso de pesos separados para las modalidades de texto e imagen, con atención conjunta.

# Stable Diffusion 3 - MM-DiT



[stabilityai/stable-diffusion-3.5-large](https://huggingface.co/stabilityai/stable-diffusion-3.5-large) .  
Hugging Face

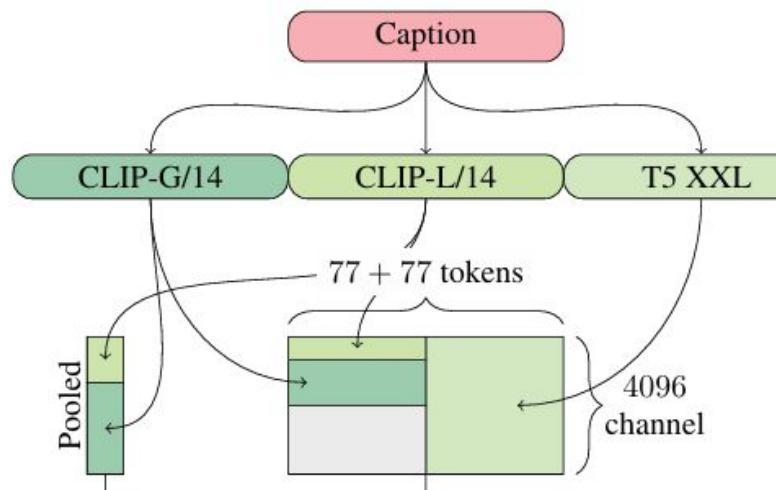
# Stable Diffusion 3 - Text Encoders



- El encoder T5-XXL (de 4.7B parámetros) requiere una cantidad significativa de memoria.
- Para mejorar la eficiencia, se puede optar por utilizar solo un subconjunto de los codificadores disponibles, reduciendo así el uso de recursos.
- Sin embargo, T5 sigue siendo clave para comprender prompts complejos; su omisión puede afectar ligeramente la adherencia al texto.
- Las salidas de los tres codificadores —CLIP-L/14, CLIP-G/14 y T5-XXL— se concatenan, generando una representación textual multimodal de 4096 canales.
- El tensor resultante se proyecta mediante una capa lineal para generar el vector de condicionamiento ( $c$ ) que se usa dentro de los bloques MM-DiT.

[stabilityai/stable-diffusion-3.5-large · Hugging Face](https://github.com/stabilityai/stable-diffusion-3.5-large)

# Stable Diffusion 3 - Encoders



- T5-xxl es un modelo especializado en procesamiento de lenguaje natural (texto a texto). Contexto de 77/256 tokens
- CLIPs son modelos multimodales diseñados para trabajar con imágenes y texto. Contexto de 77 tokens para procesar entradas de texto.
- OpenCLIP-ViT/G es una variante con un tamaño de modelo más grande y CLIP-ViT/L es una versión más pequeña. Ambos modelos comparten la misma arquitectura base.

# Stable Diffusion 3 - Text Encoders

*All text-encoders*



*w/o T5 (Raffel et al., 2019)*



“A burger patty, with the bottom bun and lettuce and tomatoes. “COFFEE” written on it in mustard”



“A monkey holding a sign reading “Scaling transformer models is awesome!”

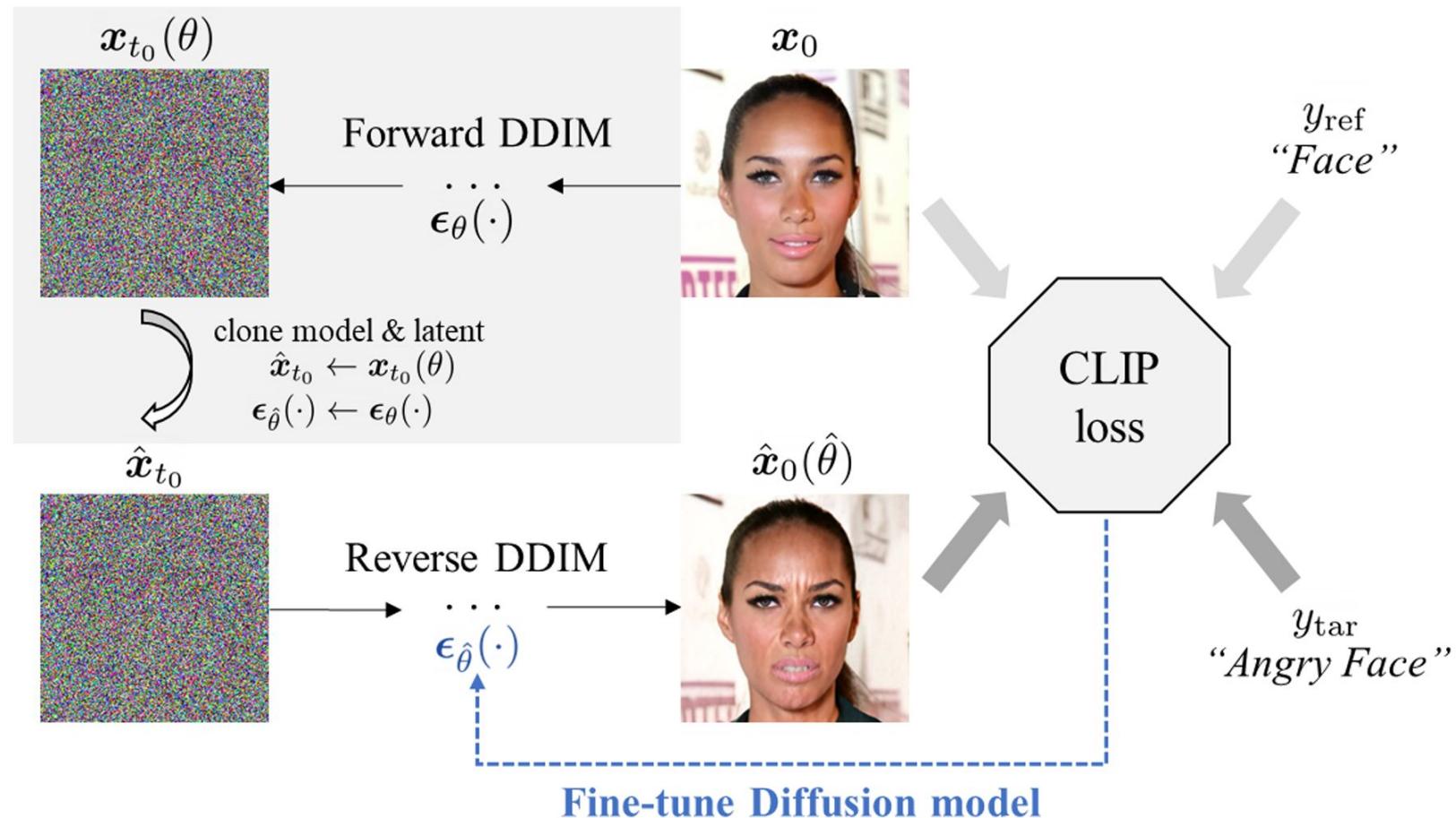


“A mischievous ferret with a playful grin squeezes itself into a large glass jar, surrounded by colorful candy. The jar sits on a wooden table in a cozy kitchen, and warm sunlight filters through a nearby window”

# DiffusionCLIP

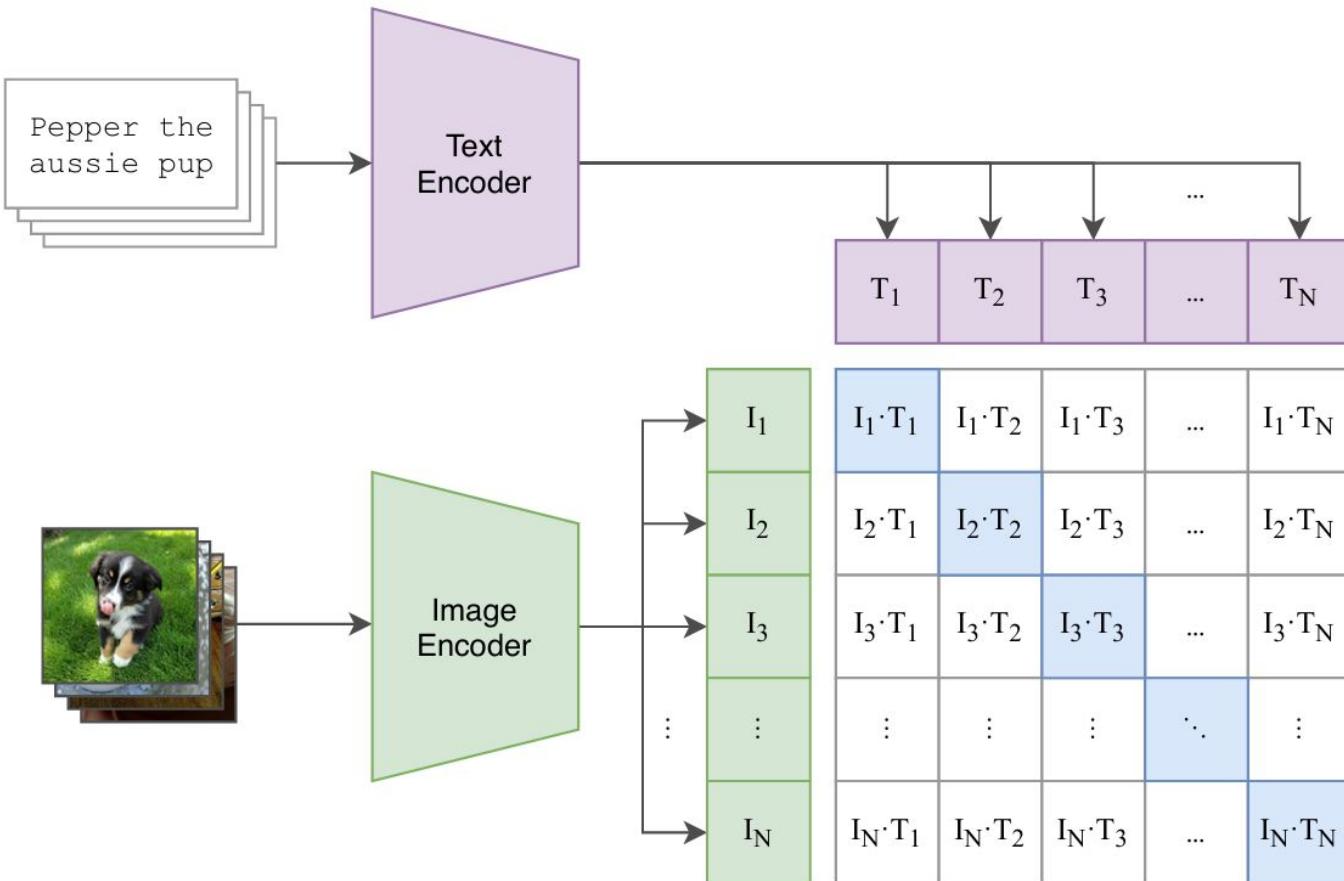
[Github: gwang-kim/DiffusionCLIP](https://github.com/gwang-kim/DiffusionCLIP)

DiffusionCLIP fue presentado en el paper “[DiffusionCLIP: Text-Guided Diffusion Models for Robust Image Manipulation](#)”



# CLIP (Contrastive Language-Image Pretraining)

## (1) Contrastive pre-training

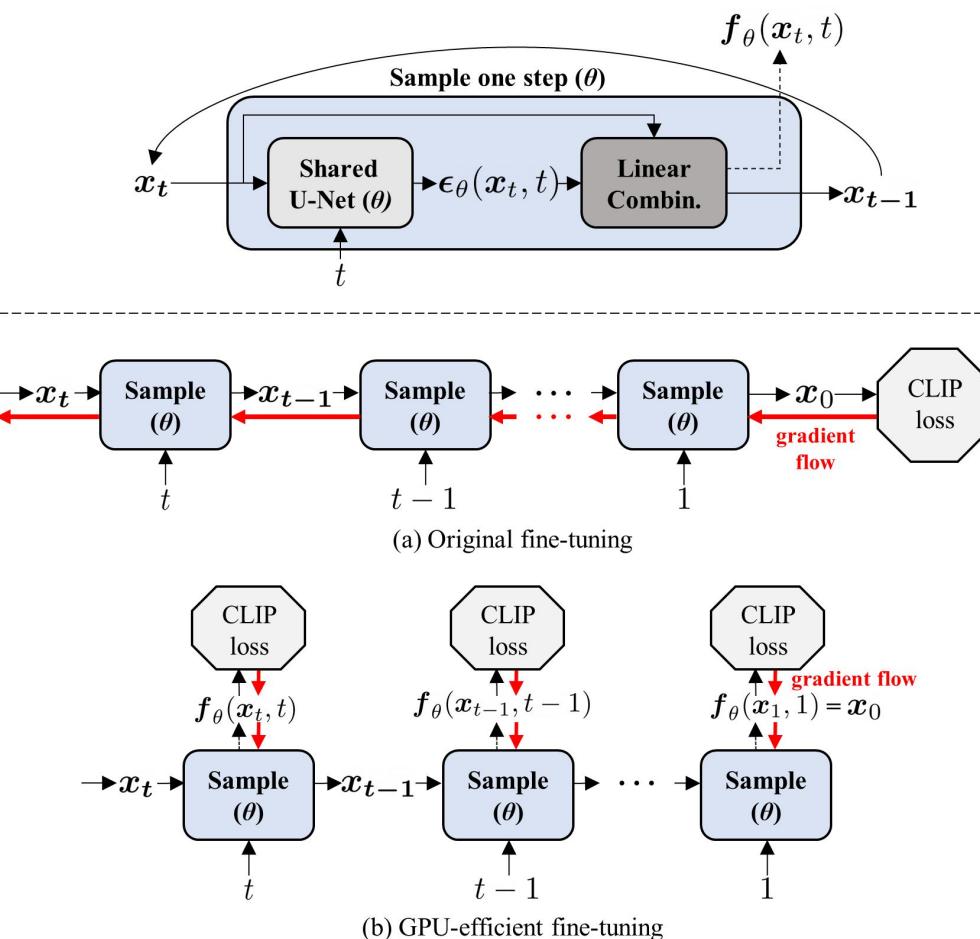


**CLIP** entrena dos redes neuronales, una para imágenes y otra para texto.

- Las imágenes y los textos se proyectan en un espacio latente común.
- Se utiliza una función de pérdida contrastiva que aproxima las representaciones de texto e imagen correspondientes y separa las no correspondientes.
- Una vez entrenado, CLIP puede realizar tareas sin necesidad de entrenamiento adicional, como clasificación de imágenes solo con descripciones textuales.

# DiffusionCLIP

[Github: gwang-kim/DiffusionCLIP](https://github.com/gwang-kim/DiffusionCLIP)



$$\mathcal{L}_{\text{direction}}(\mathbf{x}_{\text{gen}}, y_{\text{tar}}; \mathbf{x}_{\text{ref}}, y_{\text{ref}}) := 1 - \frac{\langle \Delta I, \Delta T \rangle}{\|\Delta I\| \|\Delta T\|},$$

$$\mathcal{L}_{\text{global}}(\mathbf{x}_{\text{gen}}, y_{\text{tar}}) = D_{\text{CLIP}}(\mathbf{x}_{\text{gen}}, y_{\text{tar}}),$$

$$\Delta T = E_T(y_{\text{tar}}) - E_T(y_{\text{ref}}), \quad \Delta I = E_I(\mathbf{x}_{\text{gen}}) - E_I(\mathbf{x}_{\text{ref}}).$$

## CLIP-LOSS

**CLIP Loss Direccional:** Alinea las representaciones de imágenes y textos en un espacio común mediante la maximización de la similitud en una dirección

**CLIP Loss Global:** Considera la similitud mutua entre imágenes y textos en ambas direcciones, buscando una correspondencia bidireccional.

Ambas estrategias optimizan la capacidad del modelo para asociar imágenes con descripciones textuales.

# Preguntas?