

# Multimodal Transformers

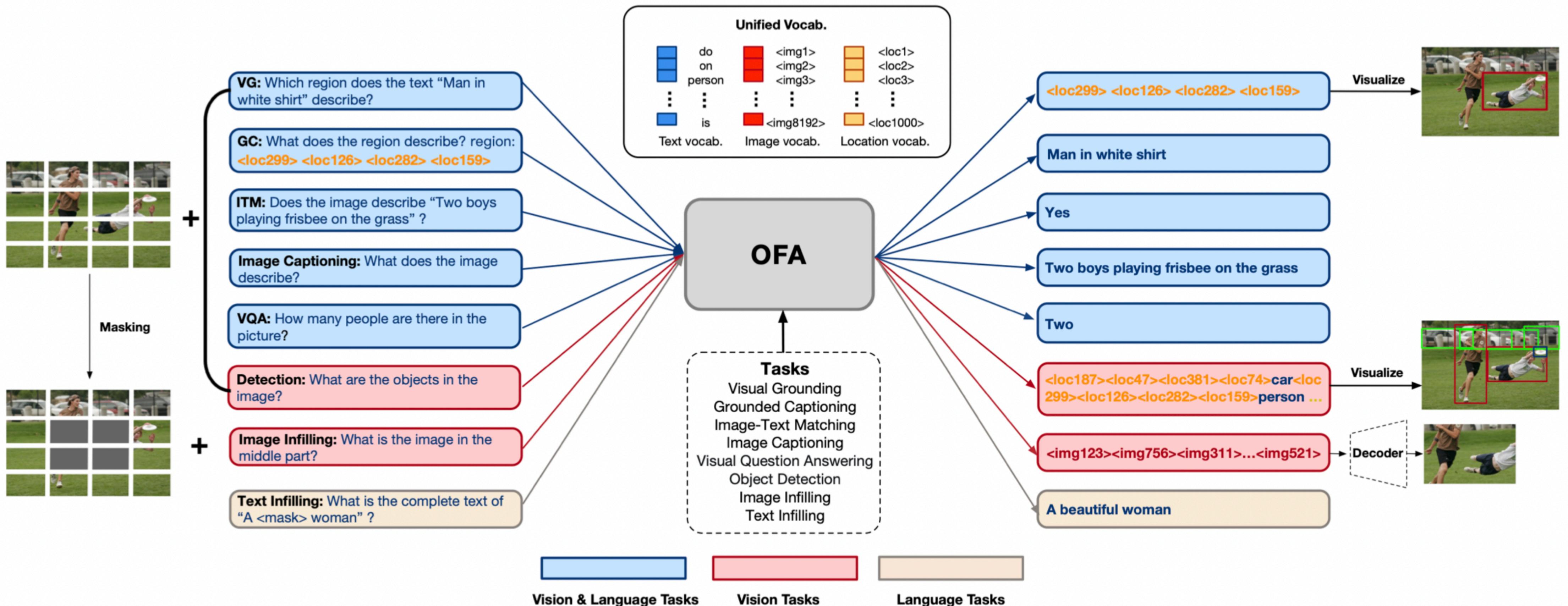
Anton Razzhigaev ( @AbstractDL)  
27. 07. 2023

# Lecture Plan

- Multimodality and inductive bias
- ViT, PIXEL, DINO, iGPT
- CLIP, DALL·E, VQ-VAE
- RuDolph
- Diffusion models
- OFA
- Flamingo
- FROMAGE
- Benchmarks

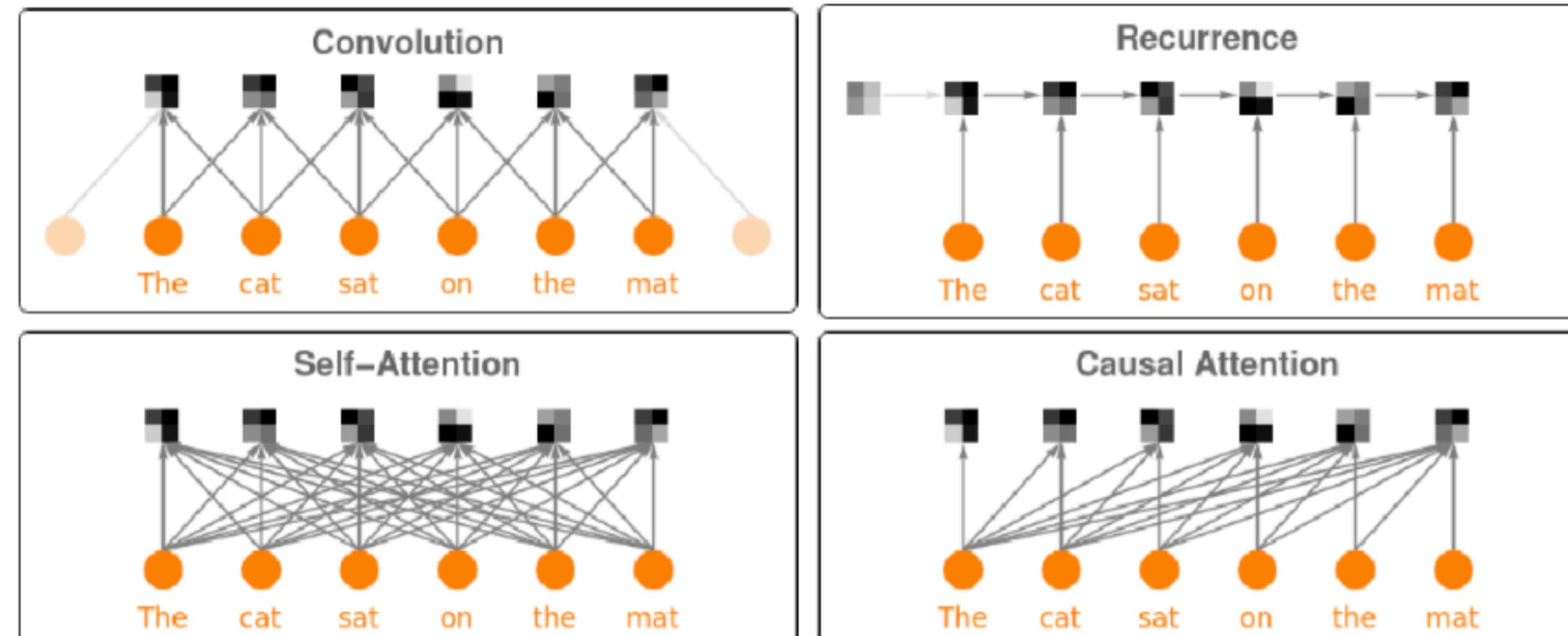


# Multimodality



OFA: UNIFYING ARCHITECTURES, TASKS, AND MODALITIES THROUGH A SIMPLE SEQUENCE-TO-SEQUENCE LEARNING FRAMEWORK

# Inductive Bias



**Inductive bias** — it is a prior knowledge about the nature of data, which a human inserts in the ml model.

- CNNs have locality inductive bias.
- RNNs have sequential inductive bias.

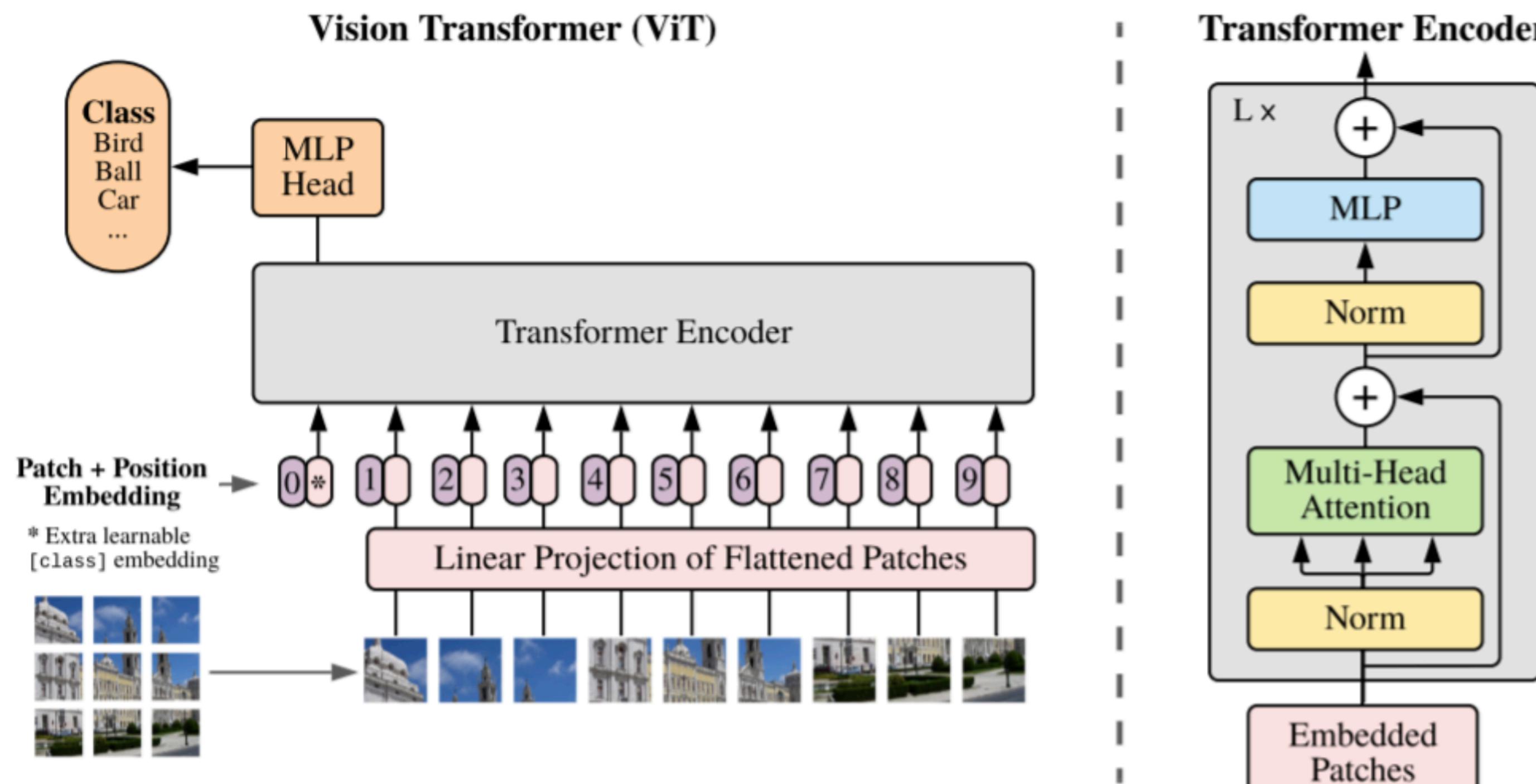
**Strong inductive bias** makes it easier to train the model. But models with strong inductive bias are less suitable for out of domain data (CNNs work not very well with texts).

In case we have **large enough datasets** or, different modalities **it is better to use weak inductive bias**, like fully-connected architectures, or transformers.

That is why **transformers are more flexible** and demonstrate better performance, but require much more data to be trained.

# Visual Transformers

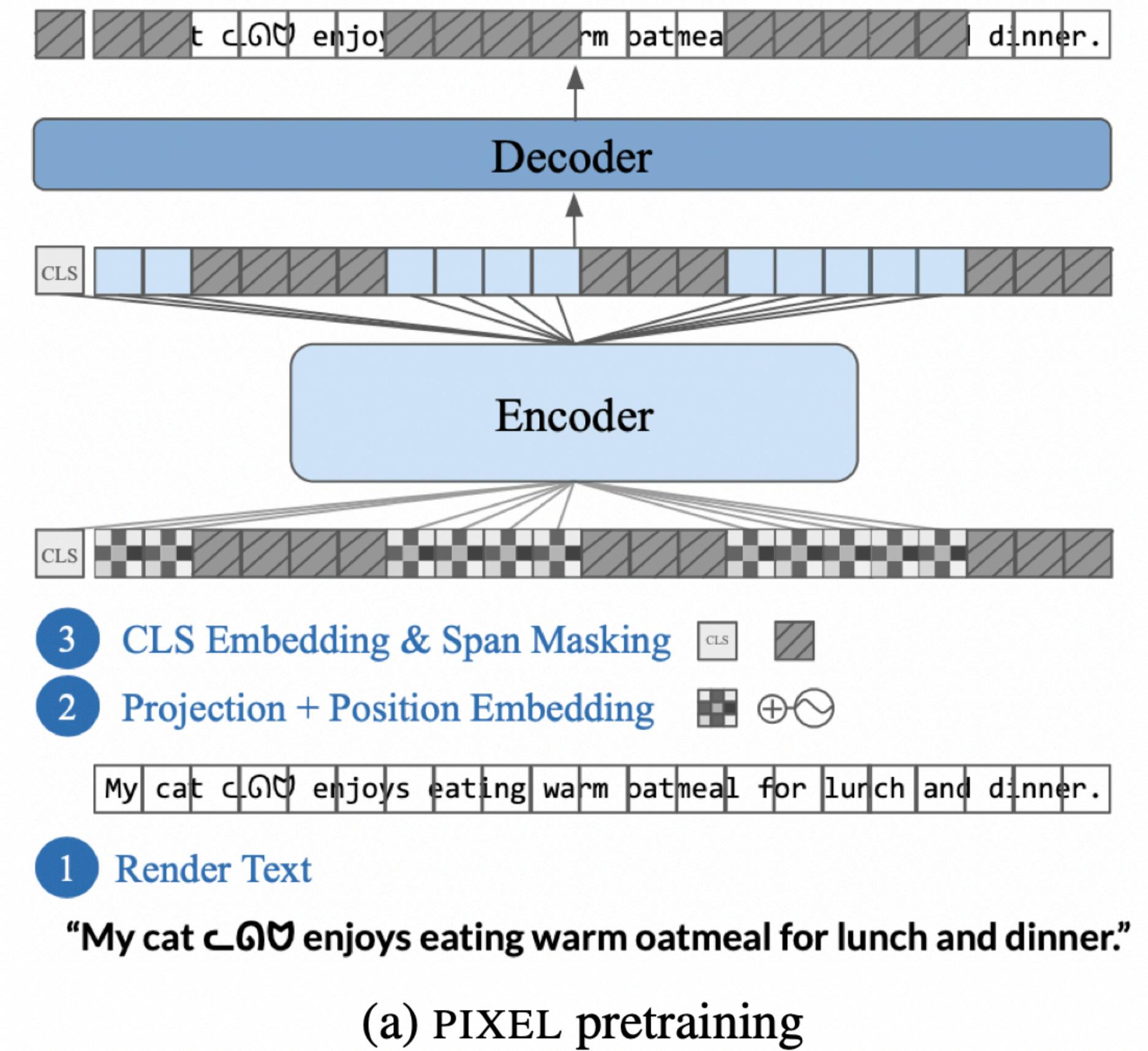
# ViT



An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

# PIXEL

- Character pixels instead of text tokens
- Masked LM over «screenshots»
- BERT-like architecture Understands
- DEEP LEARNING - can understand this text
- More robust to adversarial attacks



(a) PIXEL pretraining

# PIXEL

Our message is simple because we truly believe in our peanut-loving hearts that peanuts make everything better. Peanuts are perfectly packed because they're packed with nutrition and they bring people together. Our thirst for knowledge is unquenchable. We're always sharing snackable news stories, and the benefits of peanuts, stats, research, etc. Our passion for peanuts

# DINO

- ViT architecture
- Self-supervised objective
- Attention maps work as unsupervised segmentation

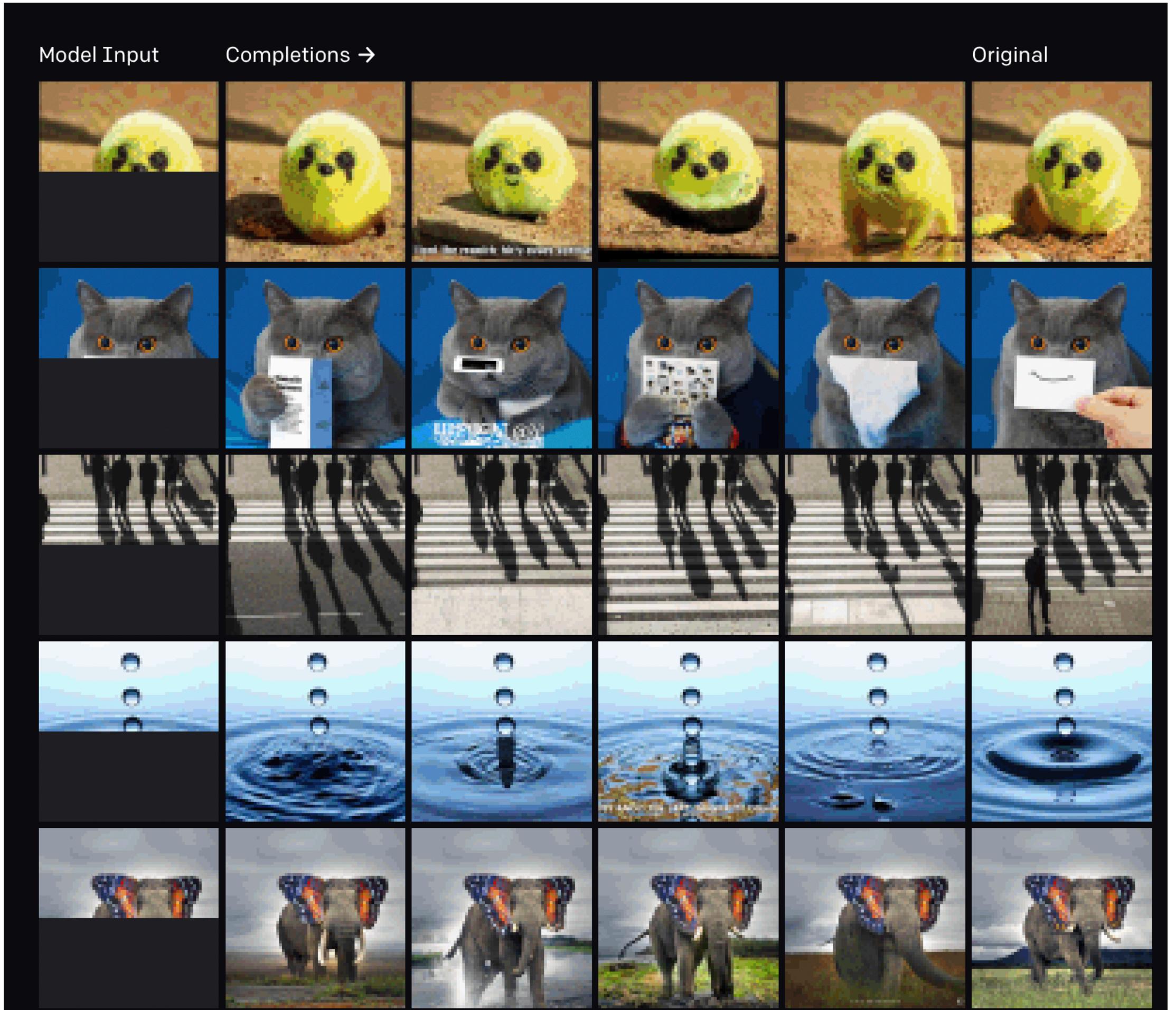


# DINO

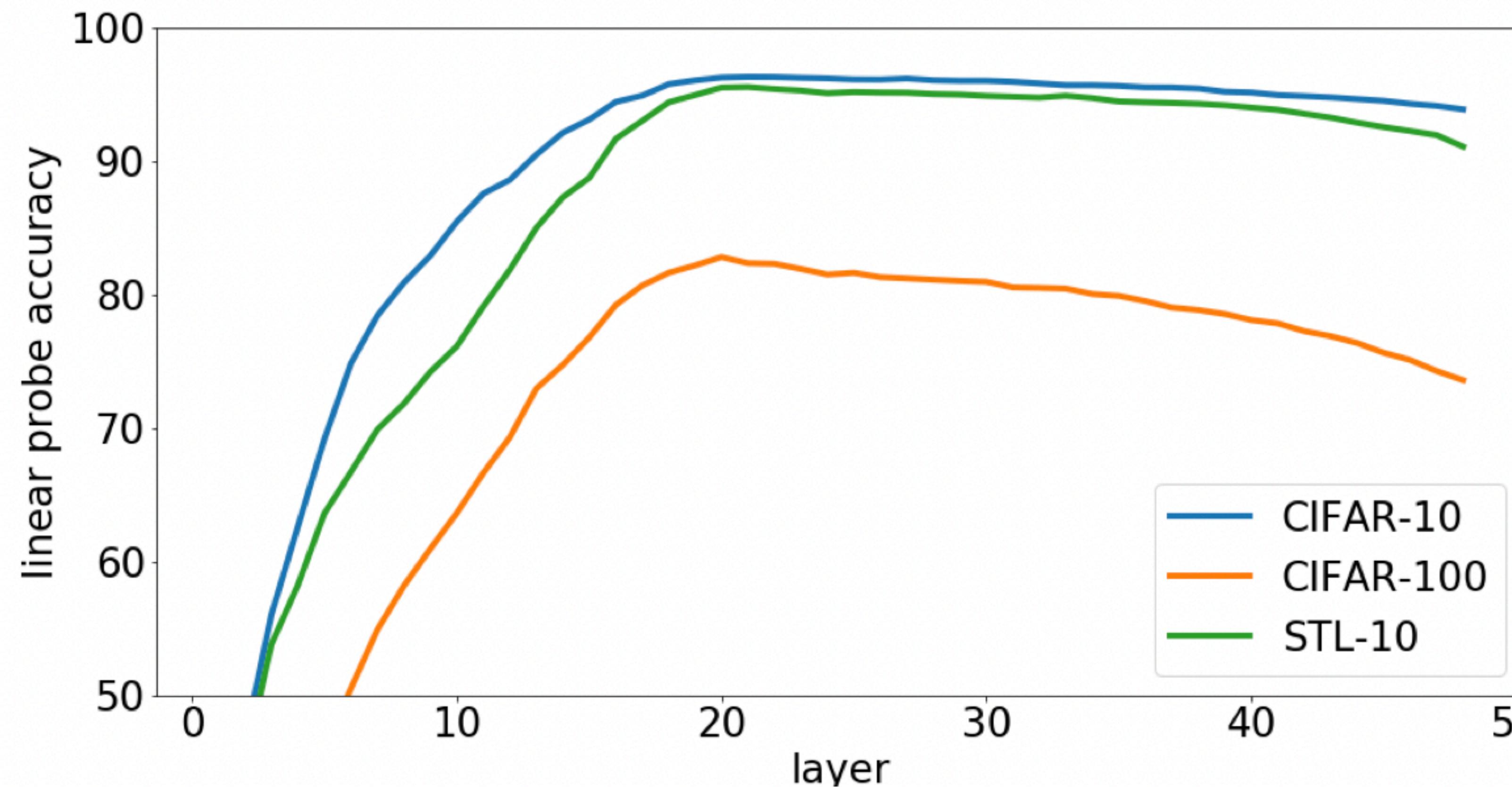
- Two networks: student and teacher
- Random crops of an image go to teacher and student
- CrossEntropy Loss between outputs of student and teacher
- Teacher =  $\text{exp\_avg}(\text{Student})$

# iGPT

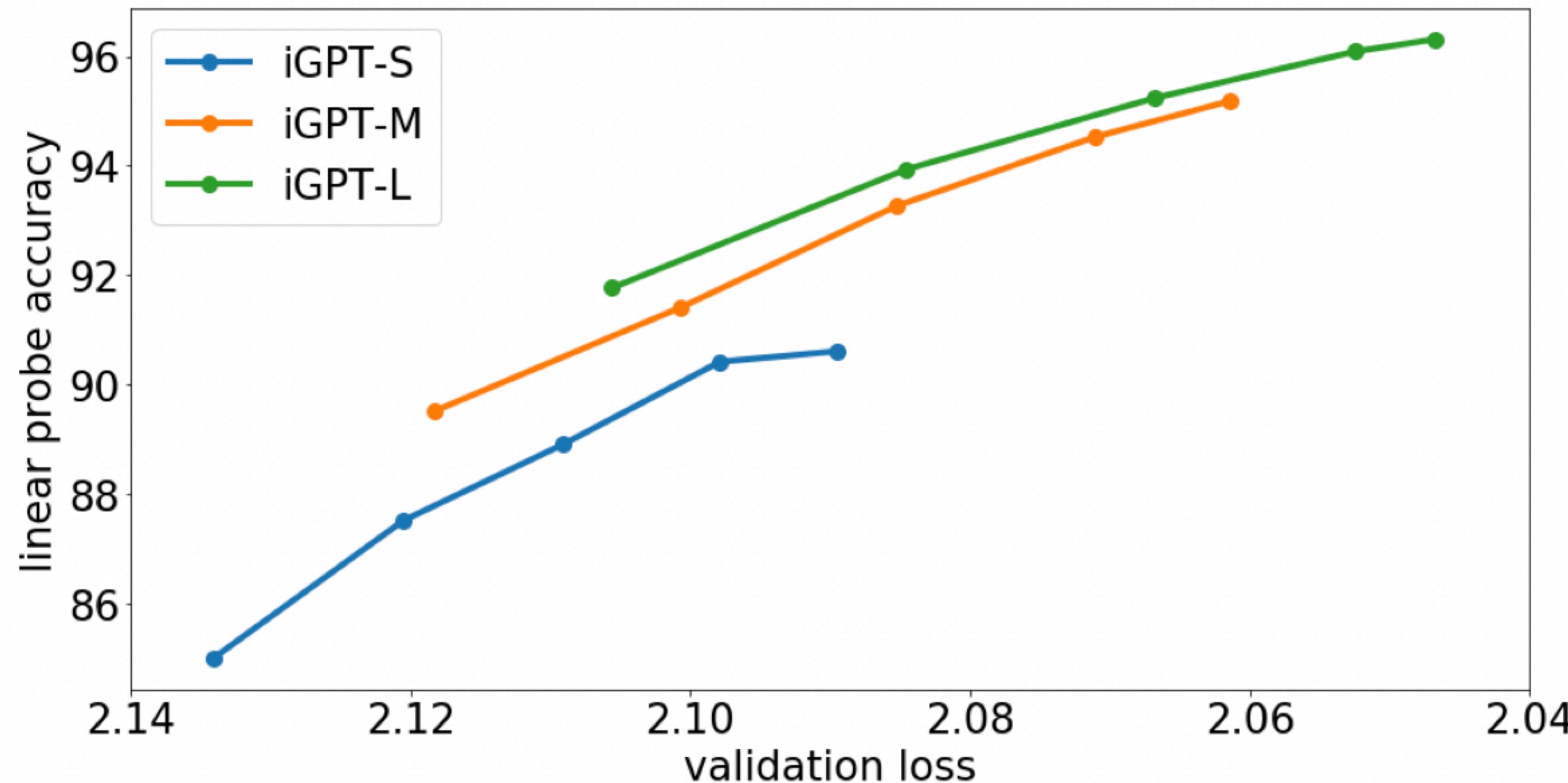
- The same architecture as GPT2
- Pretraining over pixel values with cross-entropy
- Can complete images and generate from scratch
- Embeddings can be used for downstream tasks



# iGPT: per layer linear probe



# iGPT: the larger the better



# Multimodal Tranformers

# CLIP and Dall·E

a pentagonal green clock. a green clock in the shape of a pentagon.



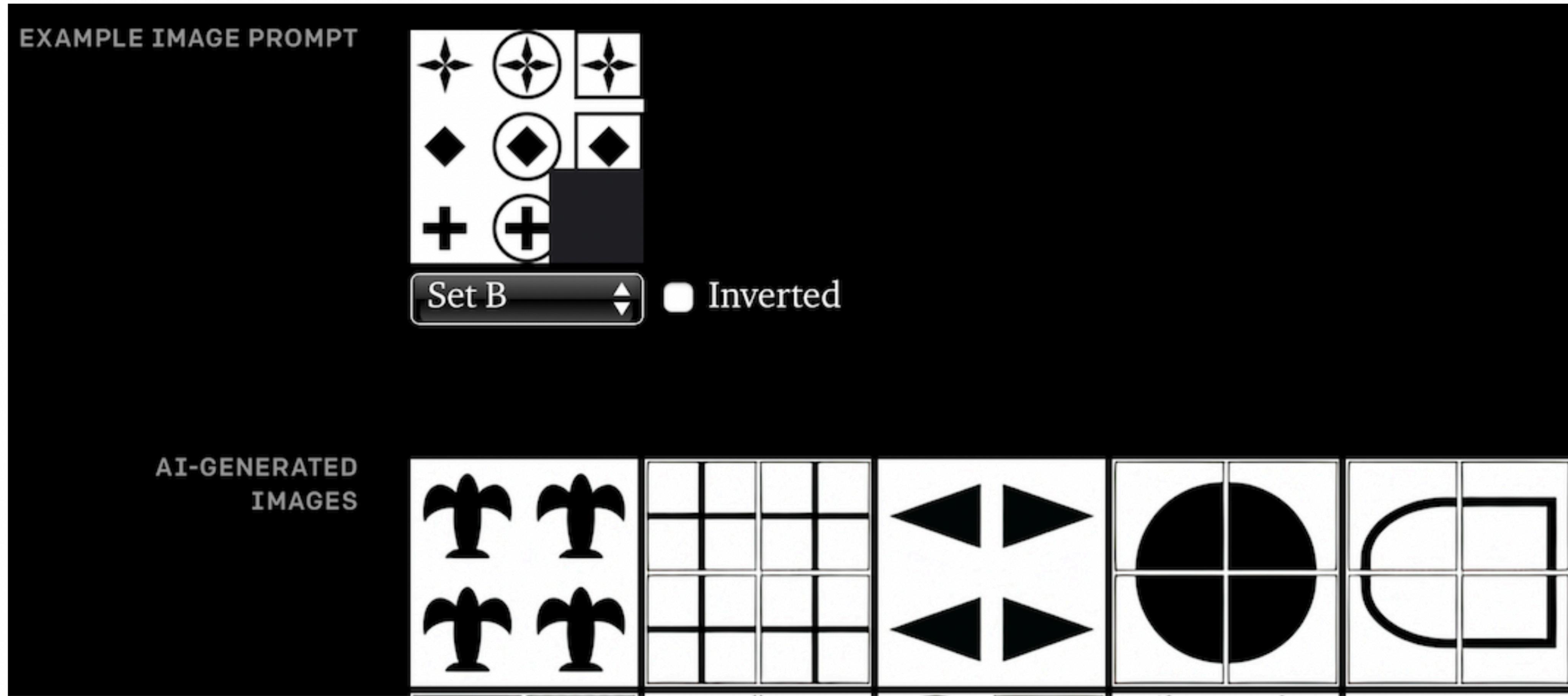
a cube made of porcupine. a cube with the texture of a porcupine.



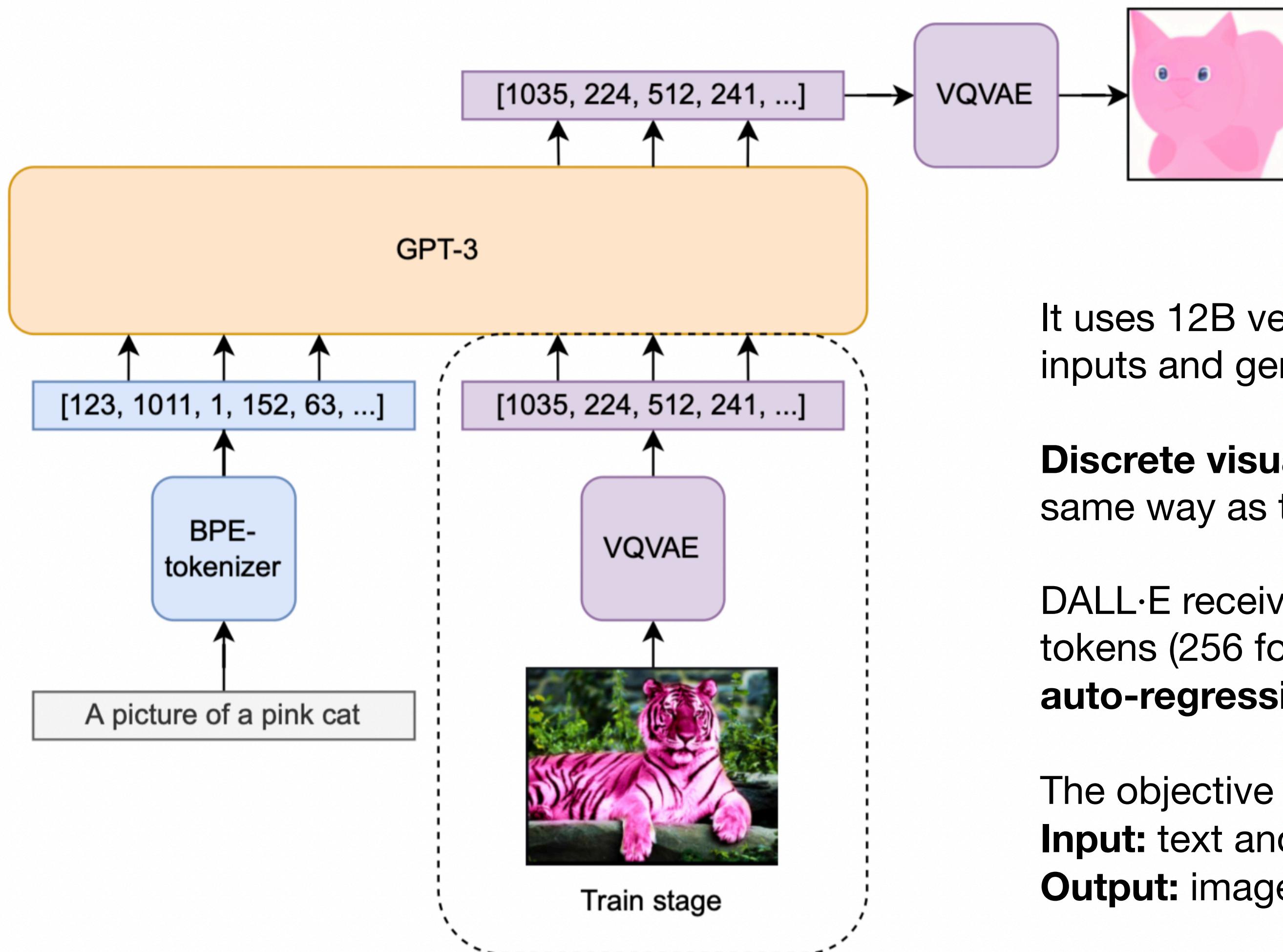
a collection of glasses is sitting on a table



# Visual Understanding of Dall·E



# DALL·E



It uses 12B version of the **GPT-3 model** to interpret natural language inputs and generate corresponding images.

**Discrete visual features from VQVAE** are used as visual tokens in a same way as text tokens, which then can be decoded back to images.

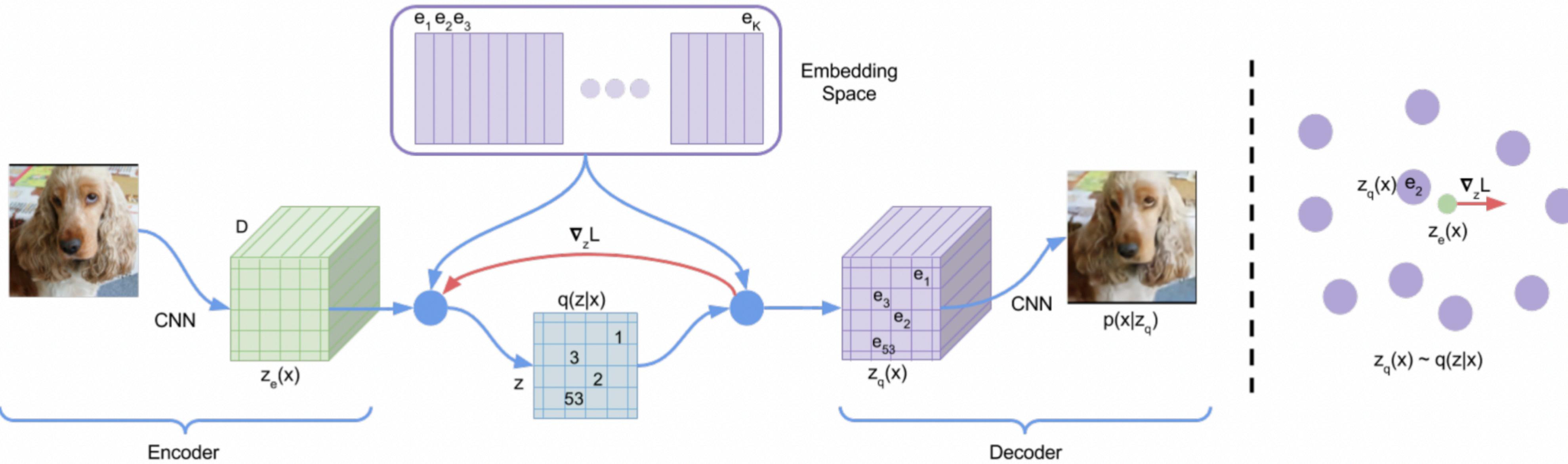
DALL·E receives both the text and the image as a single stream of 1280 tokens (256 for the text and 1024 for the image) and models all of them **auto-regressively**.

The objective is a simple **cross-entropy loss**.

**Input:** text and (optionally) part of an image

**Output:** image

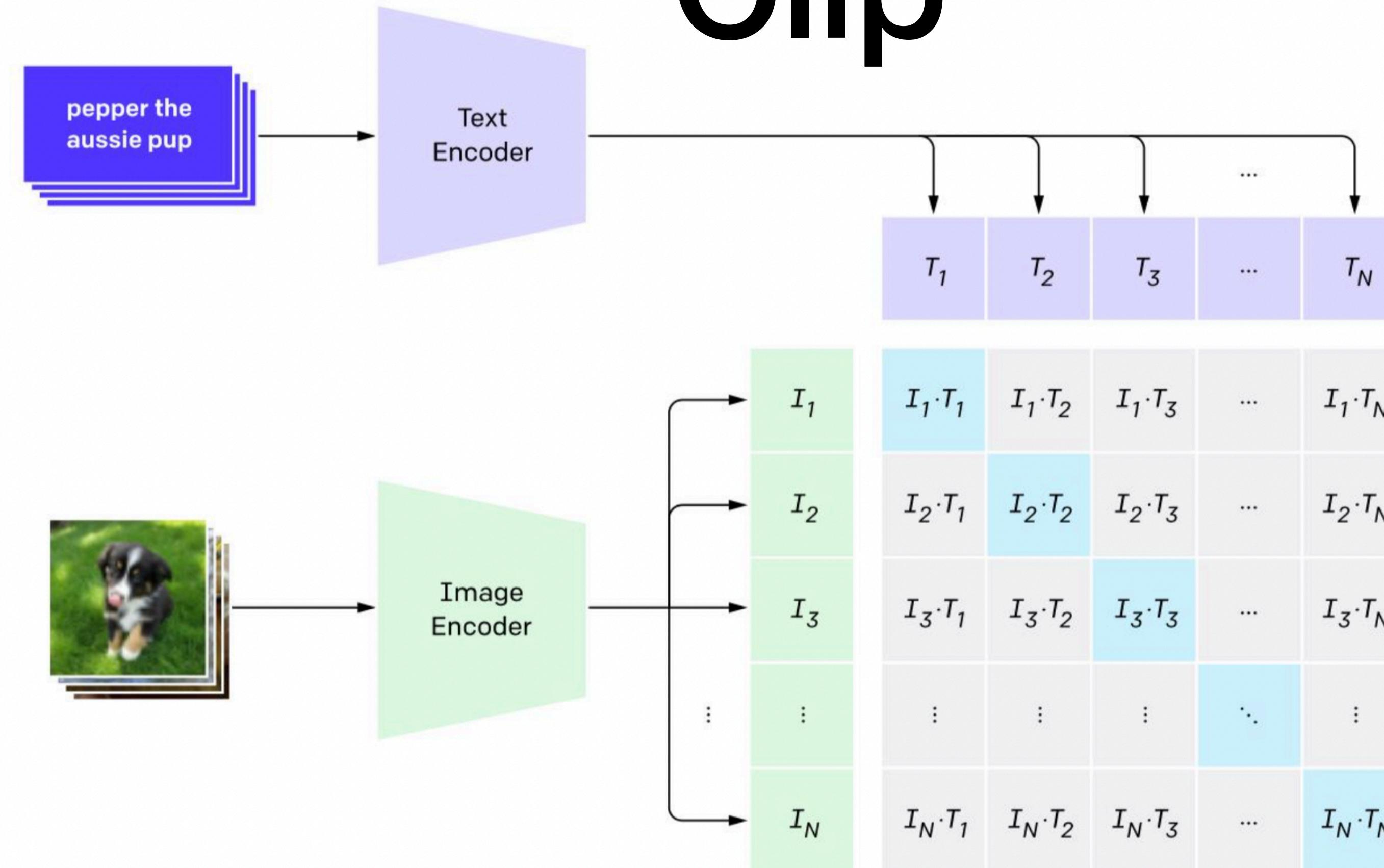
# VQ-VAE



The very important part of DALL·E is **image tokenizer** — the part of the model which transforms an image from pixels to a list of discrete tokens.

It is a usual practice to use **VQVAE** — **Vector Quantized AutoEncoder** — a special type of autoencoders, which use discrete latent space (a kind of quantized embeddings).

# Clip



The idea is pretty simple: two encoders for text and images which provide similar embeddings for images and their descriptions. It is pretrained on a large dataset of image and captions with contrastive loss.

**Input:** image or text

**Output:** embedding

# Clip Abstractions



It understands **open-set visual concepts** from natural language and demonstrate unbelievable generalization abilities!

CLIP even understands **high levels of abstractions and implicit relations between them**. Like in the picture with reversed emotions.



iPod

	Dataset Examples	ImageNet ResNet101	Zero-Shot CLIP	$\Delta$ Score
ImageNet		76.2	76.2	0%
ImageNetV2		64.3	70.1	+5.8%
ImageNet-R		37.7	88.9	+51.2%
ObjectNet		32.6	72.3	+39.7%
ImageNet Sketch		25.2	60.2	+35.0%
ImageNet-A		2.7	77.1	+74.4%

# Zero-shot Applications

query: "Кто съел всю колбасу?"



<https://t.me/abstractDL/92>

- CLIP can be used for:**
- classification
  - object detection
  - Visual-language salience
  - search
  - image reranking
  - ...

# ImageBind – CLIP for 7 modalities

## 1) Cross-Modal Retrieval

### Audio



Crackle of a Fire

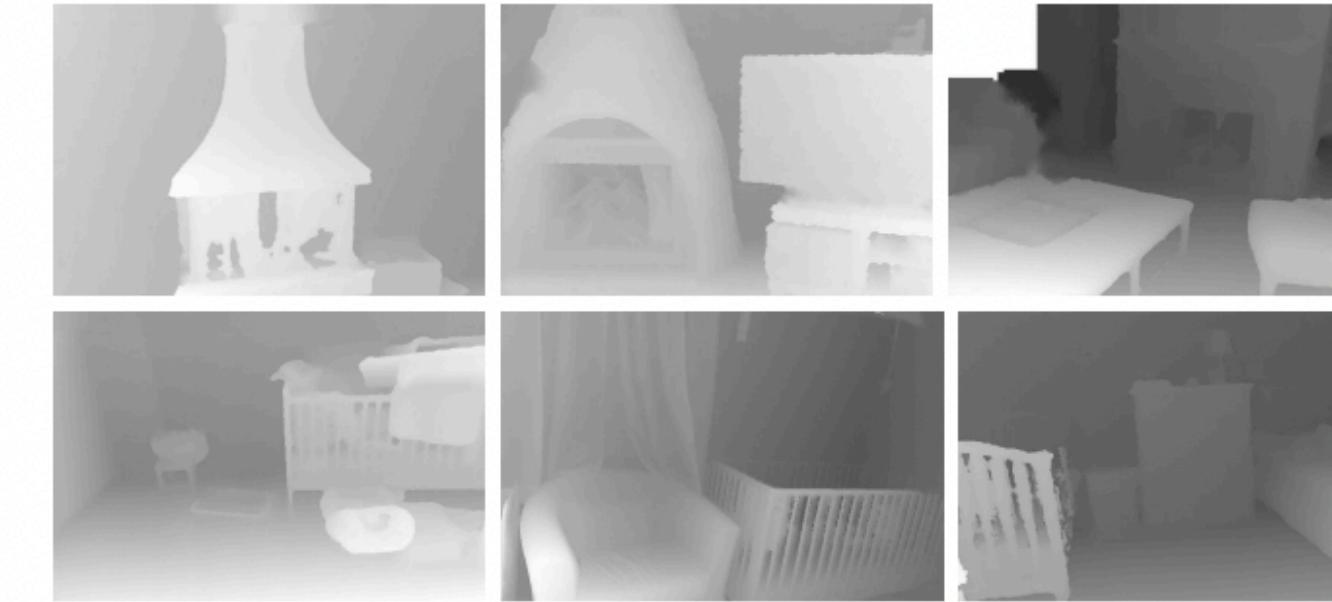


Baby Cooing

### Images & Videos



### Depth



### Text

"A fire crackles while a pan of food is frying on the fire."  
"Fire is crackling then wind starts blowing."  
"Firewood crackles then music..."

"A baby is crying while a toddler is laughing."  
"A baby is laughing while an adult is laughing."  
"A baby laughs and something..."

## 2) Embedding-Space Arithmetic



Waves



## 3) Audio to Image Generation



Dog



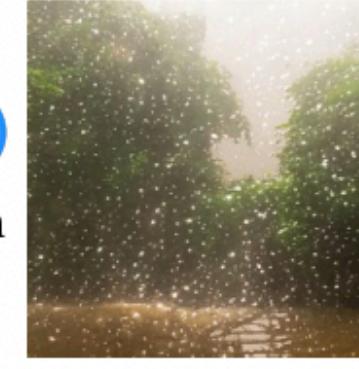
Engine



Fire



Rain



# RuDolph

It is a **hypermodal neural network** which works **similar to DALL·E**, but more flexible and it **can also generate texts**.

Developed by **SberAI**.

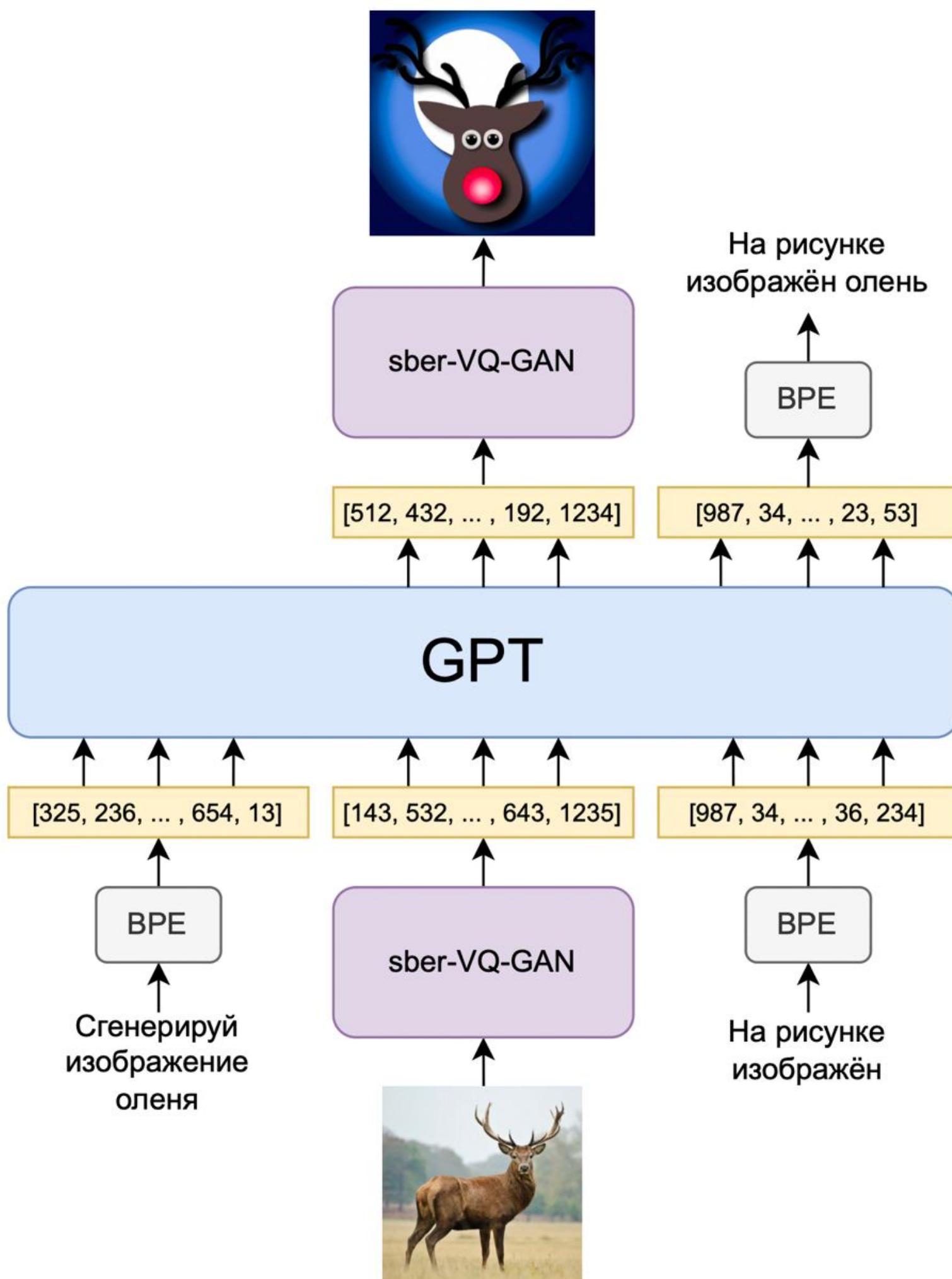
In contrast to DALL·E it has **right and left text contexts**:

- the left one is used for **image generation** (image is in between two contexts)
- the right one is used for **image captioning**.

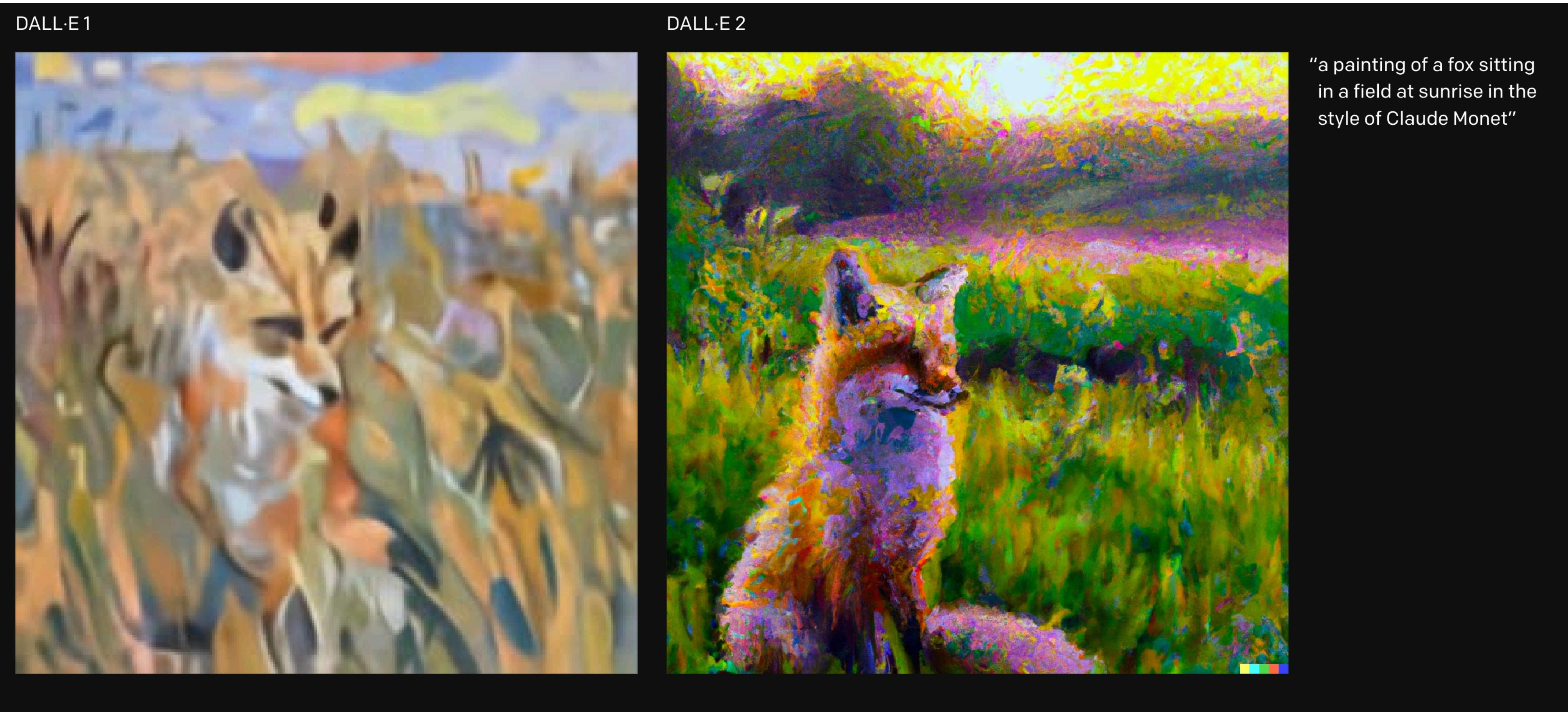
During training this two contexts and tasks alternate each other and the model is trained with **cross-entropy loss**.

**Input:** text or image

**Output:** image of text



# Dall-E·2



**The architecture is absolutely different: now it is a diffusion model conditioned on CLIP embeddings.  
NO AUTOREGRESSION**

[Hierarchical Text-Conditional Image Generation with CLIP Latents](#)

# Diffusion

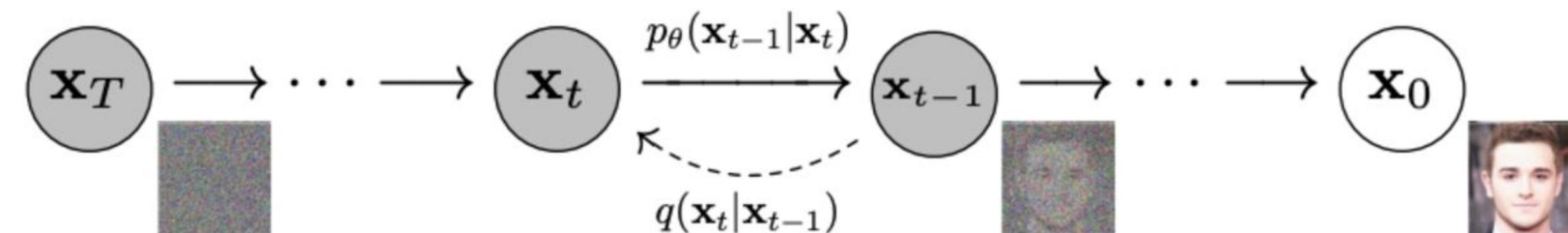
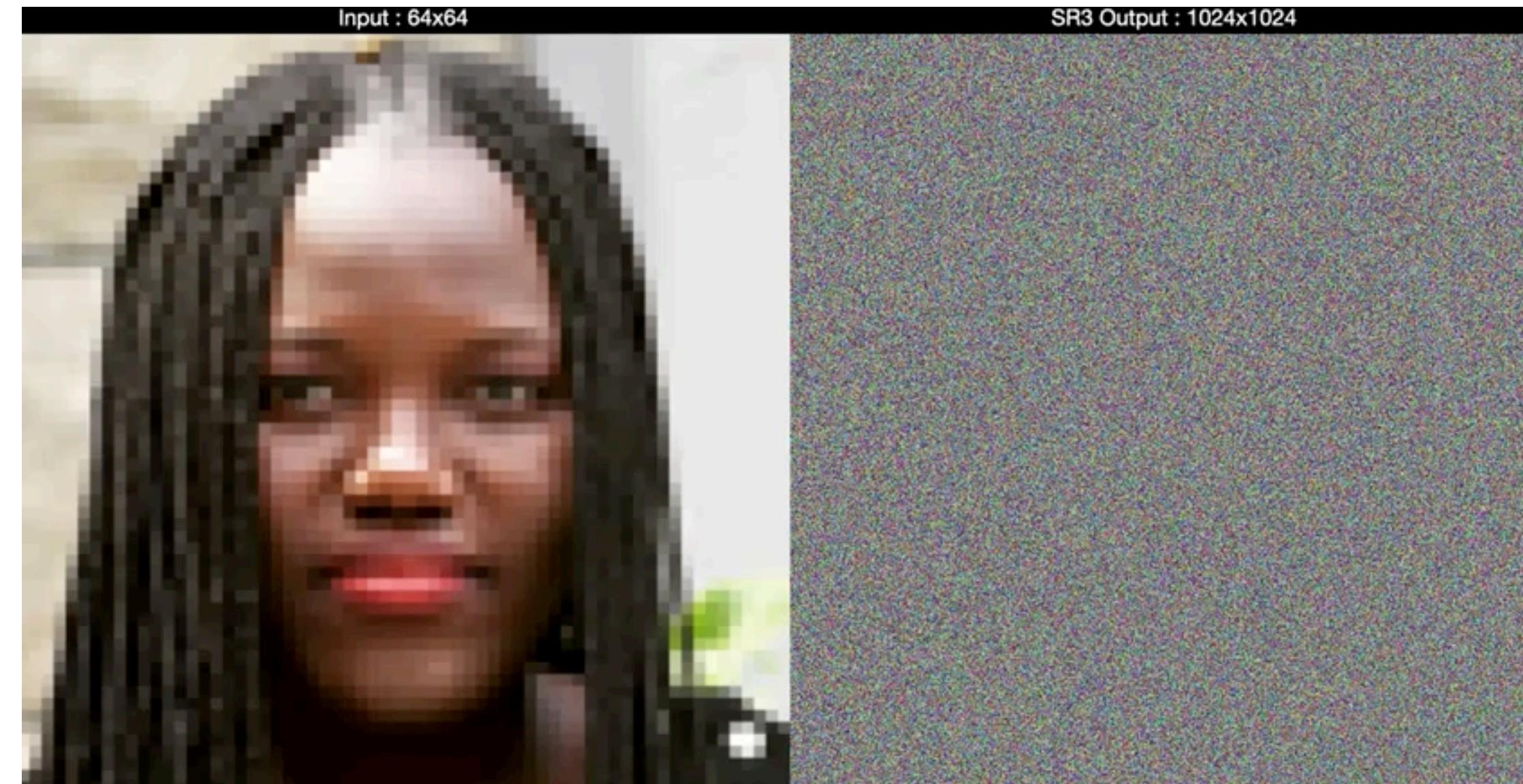
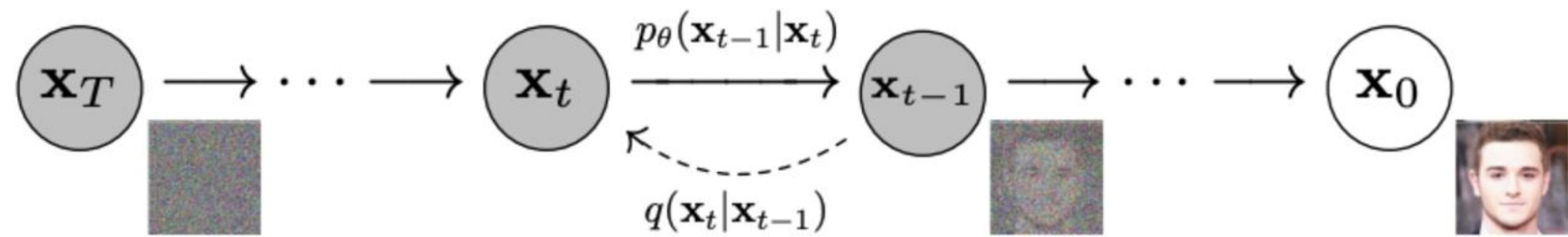
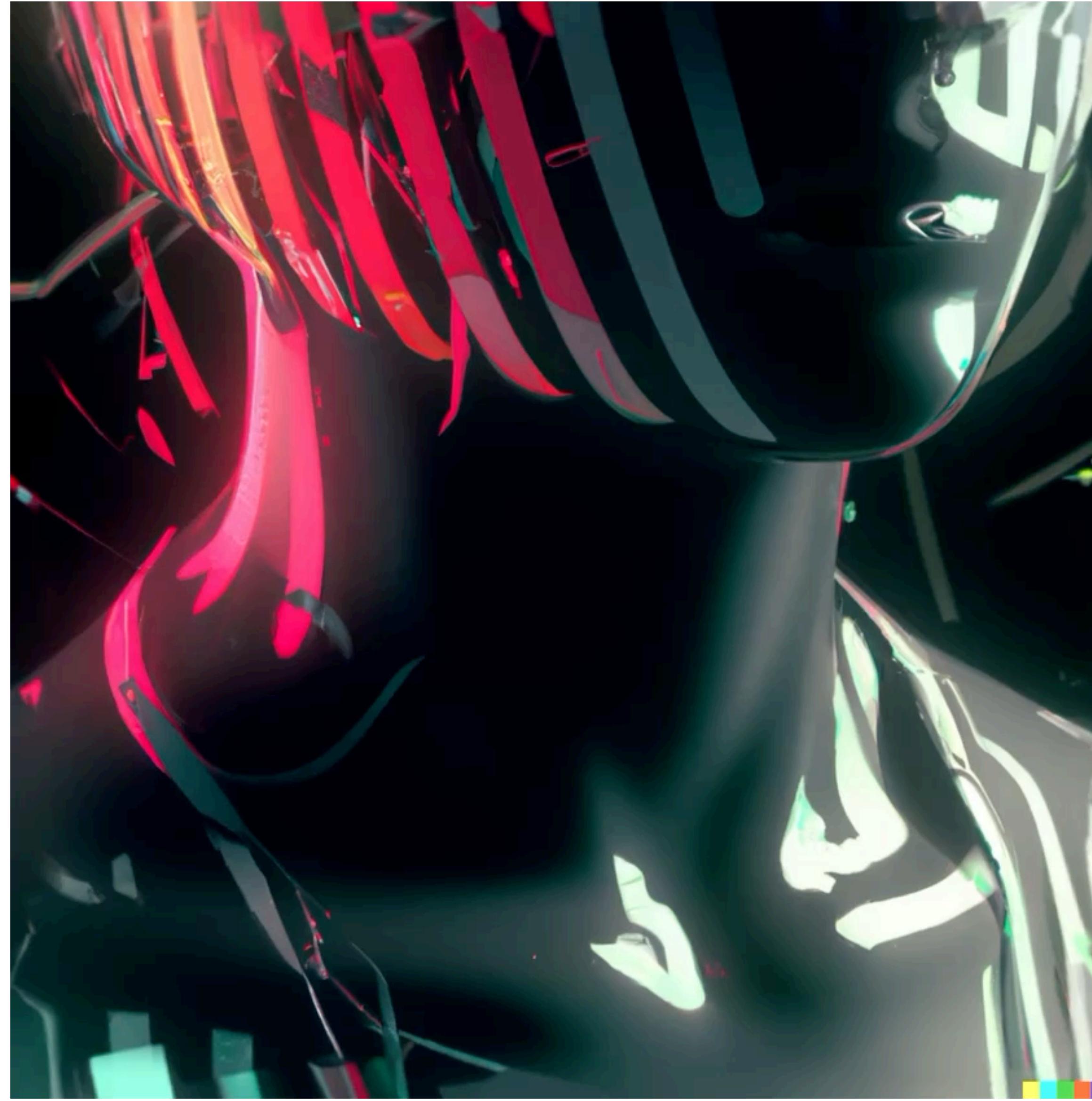


Image Super-Resolution via Iterative Refinement

# Diffusion



Diffusion Models Beat GANs on Image Synthesis



**It also can do in-painting and even zooming-out (video)**

[https://t.me/too\\_motion/455](https://t.me/too_motion/455)

# Kandinsky 2.0

- Based on **Latent Diffusion** – diffusion process in embedding space of KL-VAE
- **Multilingual** – understands more than 100 languages
- Developed by AIRI, SberAI, SberCloud
- Fully open-sourced



Железный человек on the Moon 背景中的烟花



Енот в доспехах

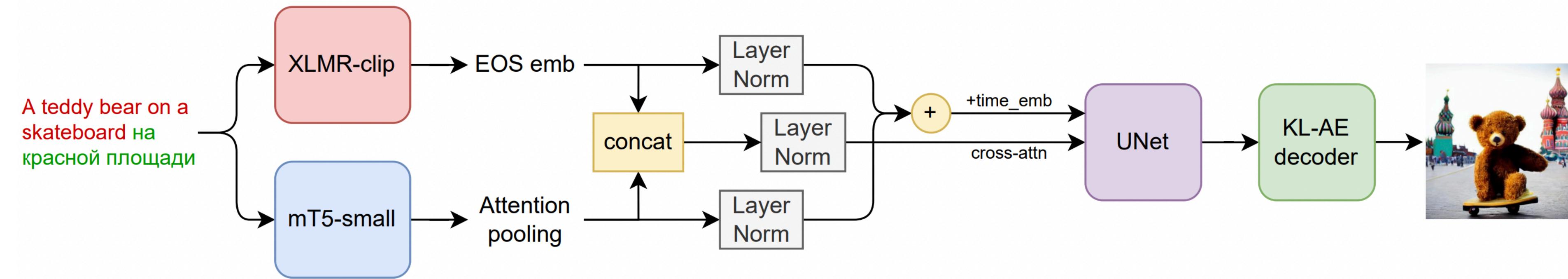


Кресло в форме тыквы

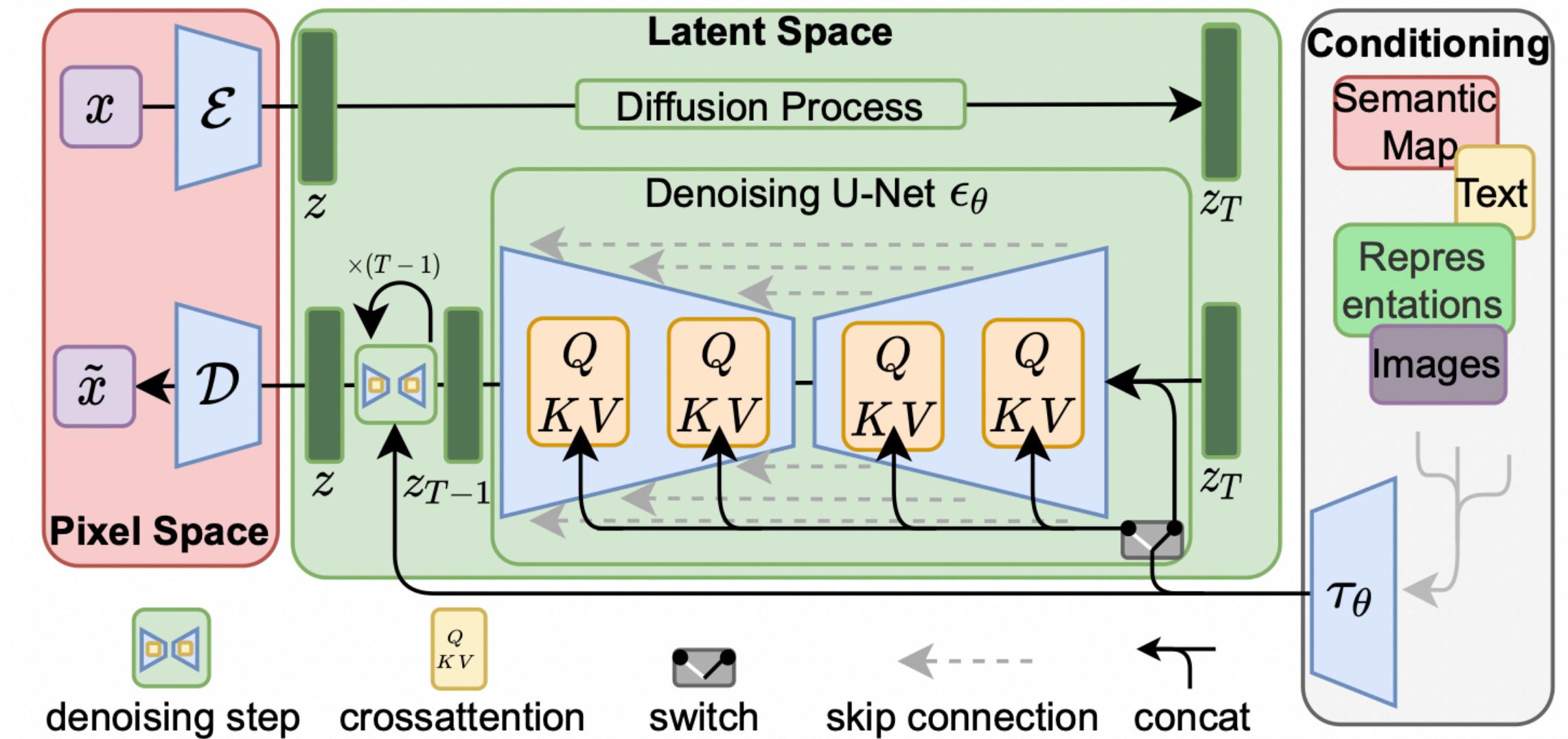


A portrait headshot of sci fi  
metallic human, bright eyes,  
complex geometric figure

# Kandinsky 2.0 Architecture



- Two multilingual encoders:  
**XLMR-clip** and **mT5-small**
- 1.2B parameters in UNET
- Dynamic thresholding



# Kandinsky 2.0 multilingual generation

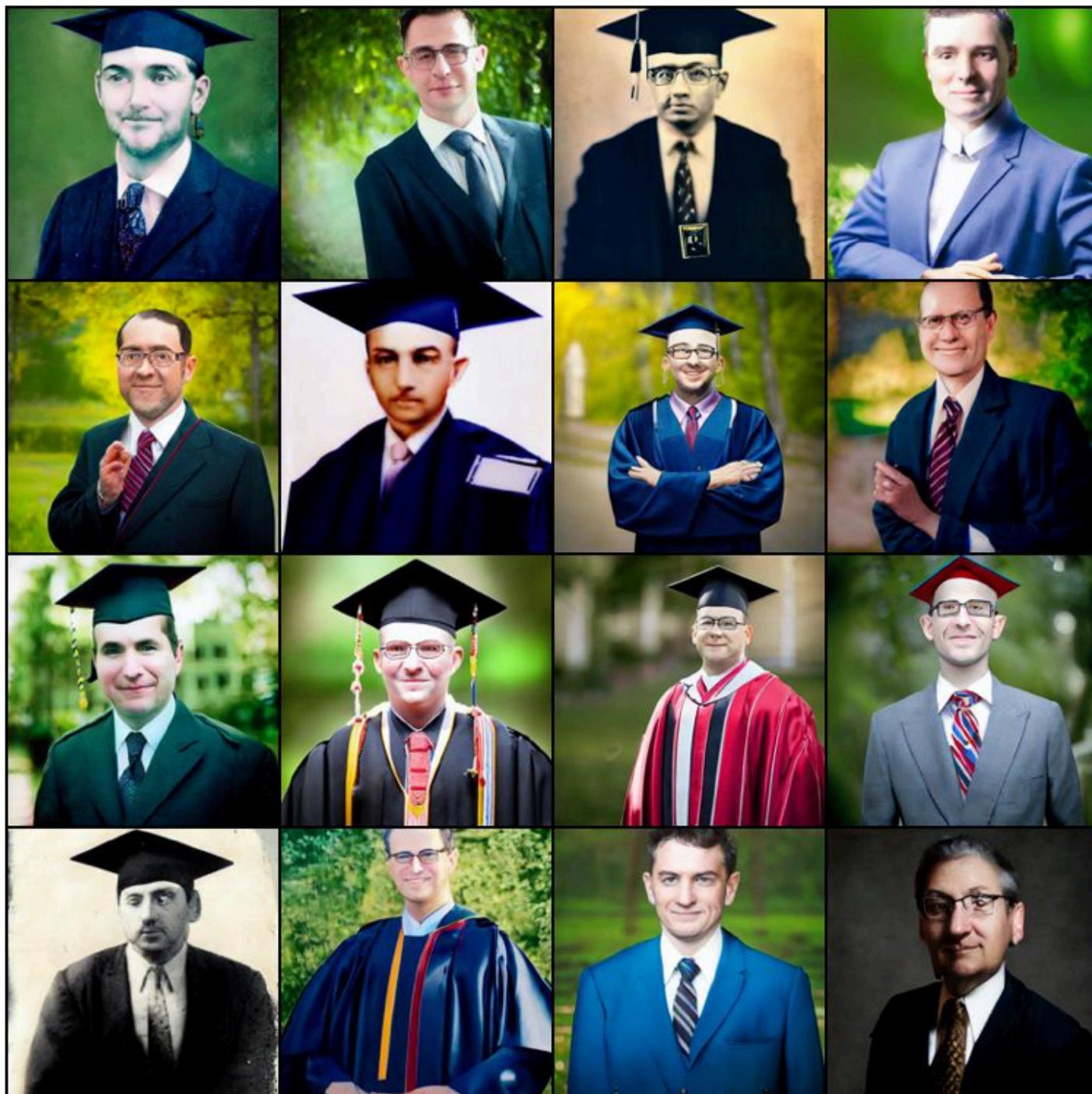


Фото человека с  
высшим образованием



Photo d'une personne  
diplômée de l'enseignement supérieur



受过高等教育的人的照片  
(китайский)

# Kandinsky 2.0 multilingual generation



Фото грабителя



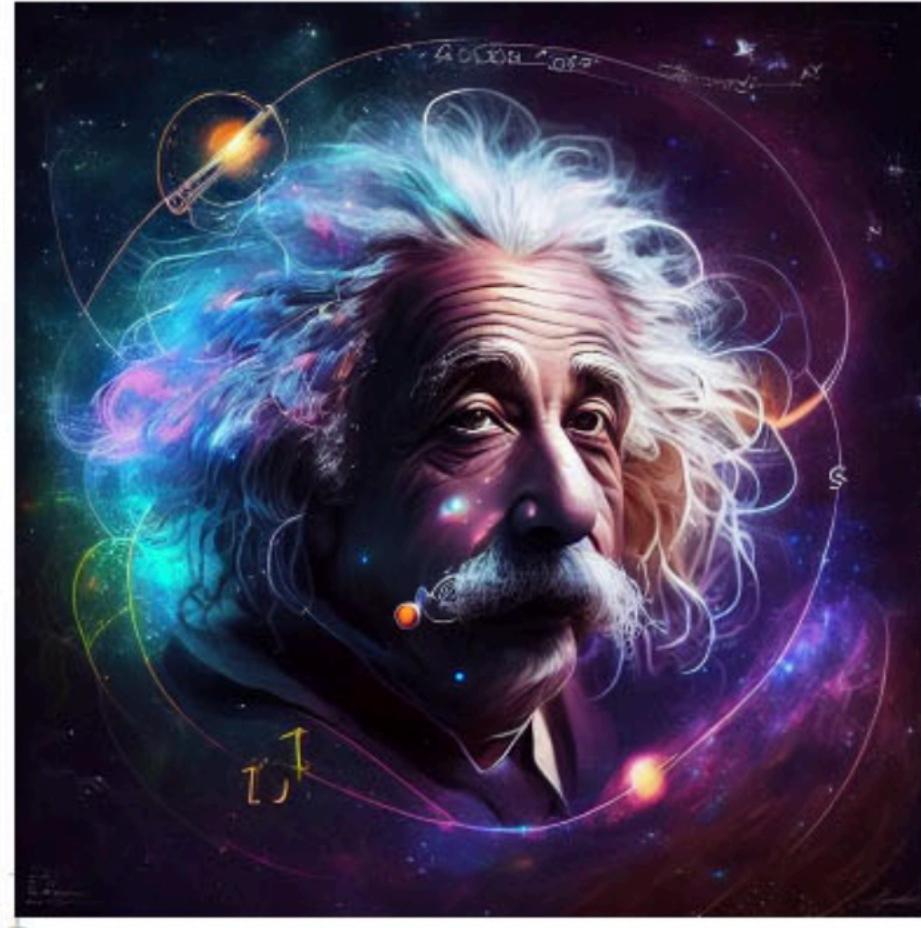
A photo of a burglar



एक चोर की तस्वीर (хинди)

# Kandinsky 2.1

- Shares the same architecture as Kandinsky 2.0 + diffusion mapping of CLIP embeddings + new decoder (MoVQ)
- Developed by AIRI, SberAI, SberCloud
- Fully open-sourced



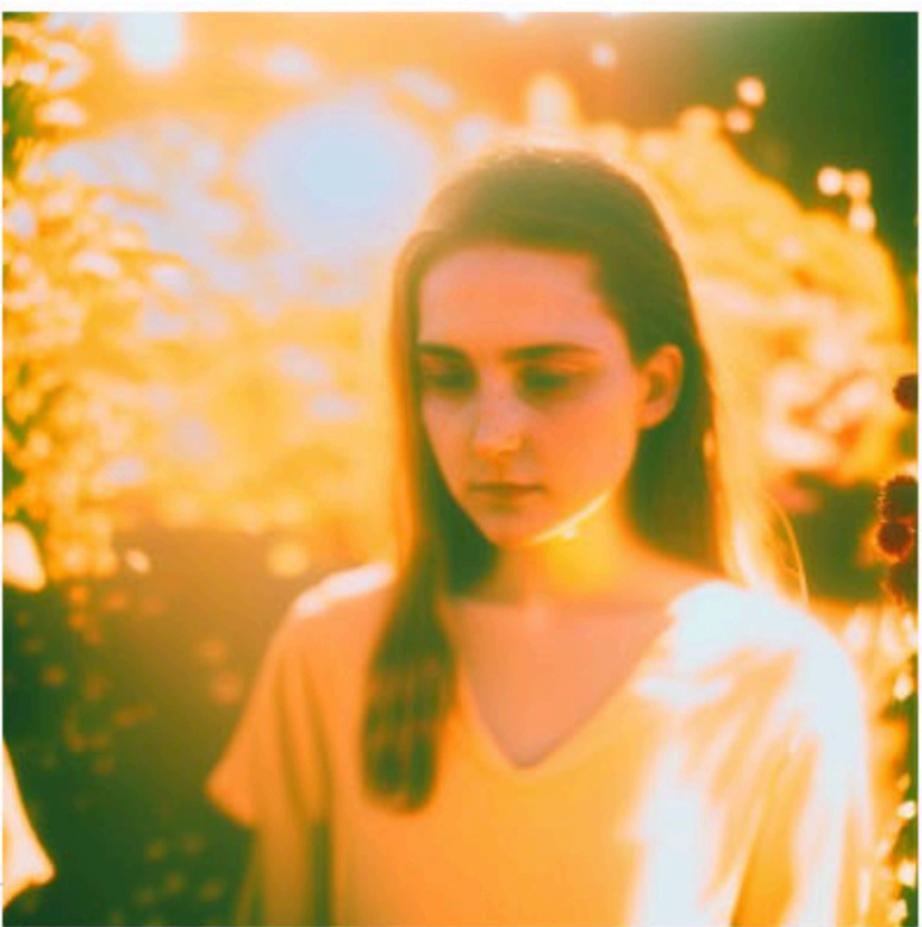
Einstein in space around the logarithm scheme



sad clown face 4k



mutant cat in the style of puppet animation in the style of horror film 4k

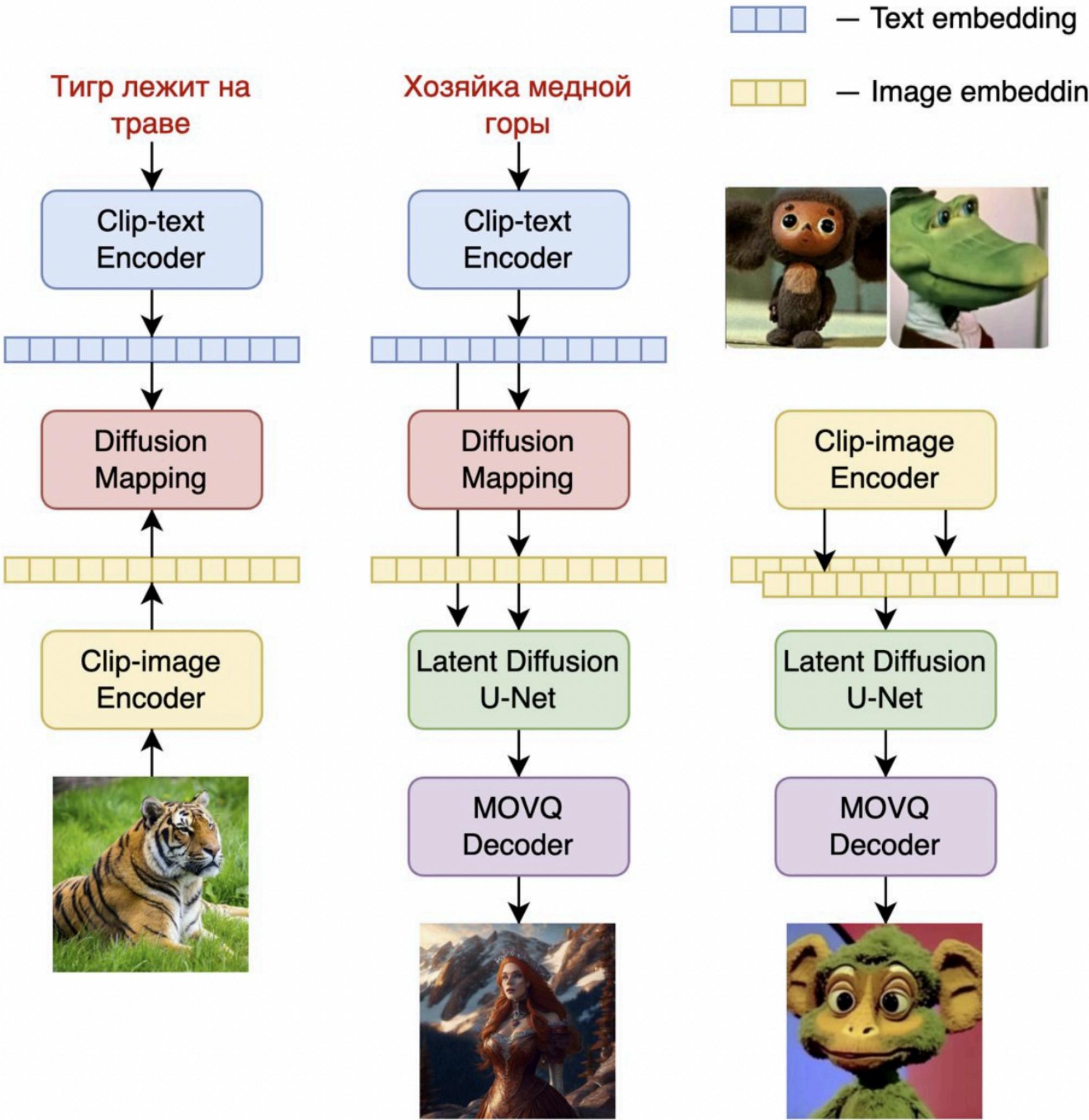


girl in the garden film grain, Kodak portra 800, f1.8, golden hour

[github](#)

[хабр](#)

# Kandinsky 2.1



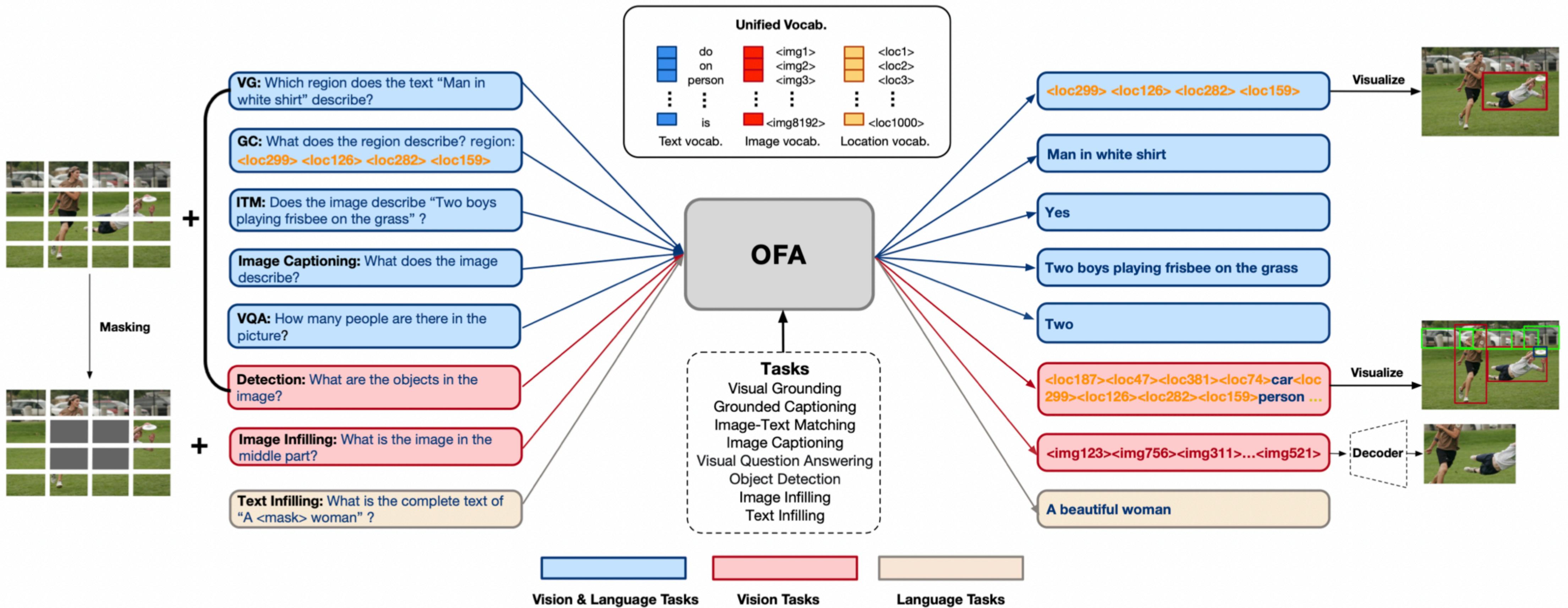
	FID-30K
eDiff-I (2022)	6,95
Imagen (2022)	7,27
<b>Kandinsky 2.1 (2023)</b>	<b>8,21</b>
Stable Diffusion 2.1 (2022)	8,59
GigaGAN, 512x512 (2023)	9,09
DALL-E 2 (2022)	10,39
GLIDE (2022)	12,24
Kandinsky 1.0 (2022)	15,40
DALL-E (2021)	17,89
Kandinsky 2.0 (2022)	20,00
GLIGEN (2022)	21,04





<https://t.me/abstractDL/207>

# OFA



OFA: UNIFYING ARCHITECTURES, TASKS, AND MODALITIES THROUGH A SIMPLE SEQUENCE-TO-SEQUENCE LEARNING FRAMEWORK

# OFA



**One For All** — a multimodal network from Alibaba which can solve almost every possible task:

- text2image generating
- image captioning
- image inpainting
- VQA
- object detection
- NLU

**The text prompt is used to switch between tasks.**  
So you should just "ask" the model to do something.

Architecture — **encoder-decoder**, almost the same as BART.  
For text tokens, visual tokens and spatial (location) tokens the same representation weights are shared (embeddings).

**Q:** what color is the car in the region? region:  
<loc512> <loc483> <loc675> <loc576>

**A:** gray

It is trained with a simple **cross-entropy loss** on multiple tasks.

**Input:** text (optionally), image (optionally), location (optionally)  
**Output:** text or/and image or/and location

Interestingly, it can solve even tasks that it did not see during training!

# Flamingo

**Flamingo** — is multimodal network. It is noticeable as authors did not train vision and language models from scratch, these models are **pretrained and frozen**.

Only **cross-attention and small adapters** are trained — a kind of connections between modalities.

Training set — **interleaved texts and images**. As it is in web pages.

Parameters: **60B**

**Input:** interleaved text and images

**Output:** text

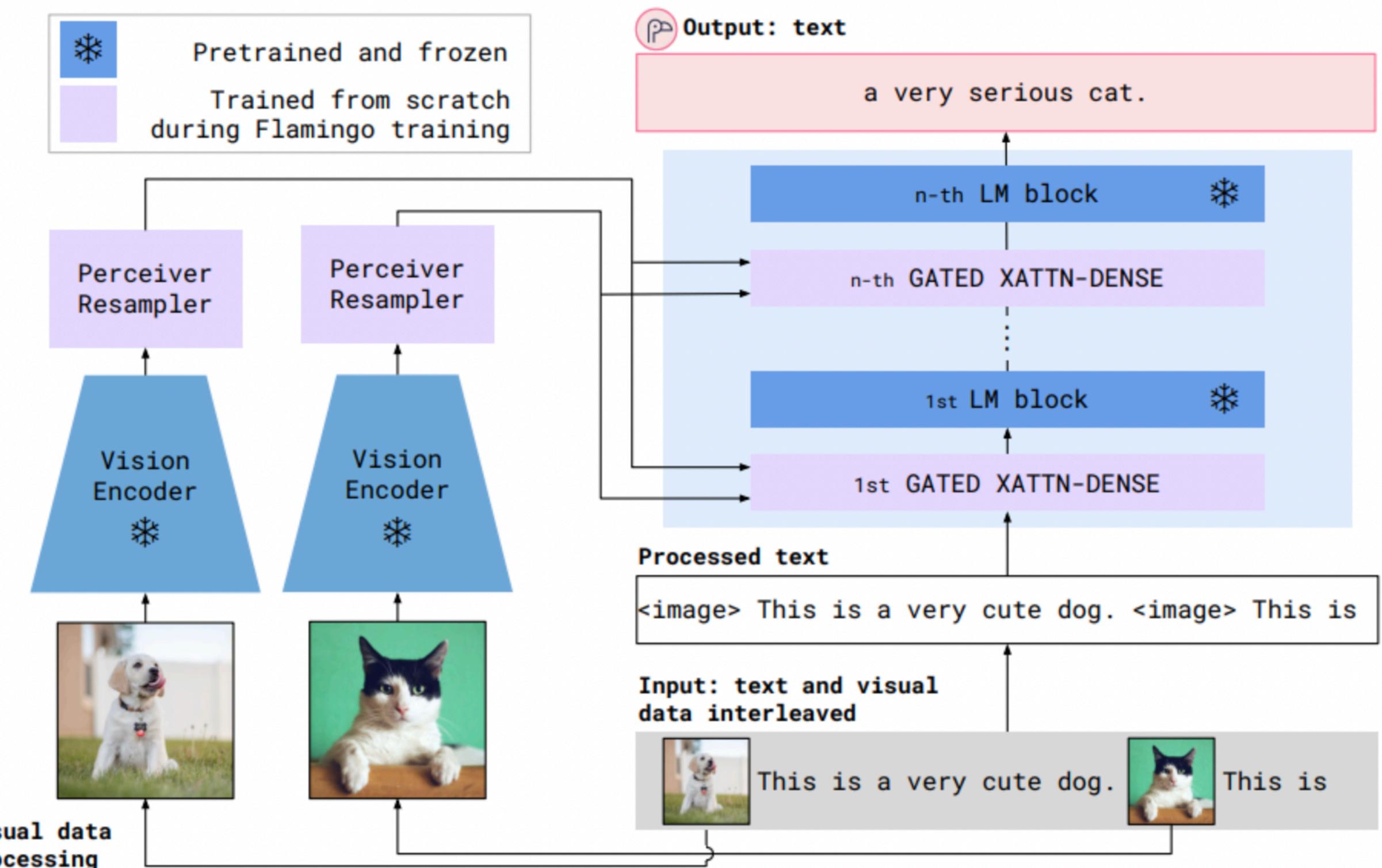


Figure 3 | Overview of the Flamingo model. The Flamingo models are a family of visual language model (VLM) that can take as input visual data interleaved with text and can produce free-form text as output. Key to its performance are novel architectural components and pretraining strategies described in Section 3.

# Flamingo



This is a picture of two teddy bears on the moon.

What are they doing?

They are having a conversation.

What object are they using?

It looks like a computer.

Is this surprising?

Yes, it is surprising.

Why is this picture surprising to you?

I think it is surprising because teddy bears are not usually found on the moon.



What is the common thing about these three images?

They are all flamingos.

What is the difference between these three images?

The first one is a cartoon, the second one is a real flamingo, and the third one is a 3D model of a flamingo.



This is an apple with a sticker on it.

What does the sticker say?

The sticker says "iPod".

Where is the photo taken?

It looks like it's taken in a backyard.

Do you think it is printed or handwritten?

It looks like it's handwritten.

What color is the sticker?

It's white.

# FROMAGe

**FROMAGe** — the easiest approach to train a multimodal dialog model. Also it is capable of retrieving images from a given set.

Only **small adapter** is trained — a mapping of visual embeddings to text embeddings and inserted into GPT.

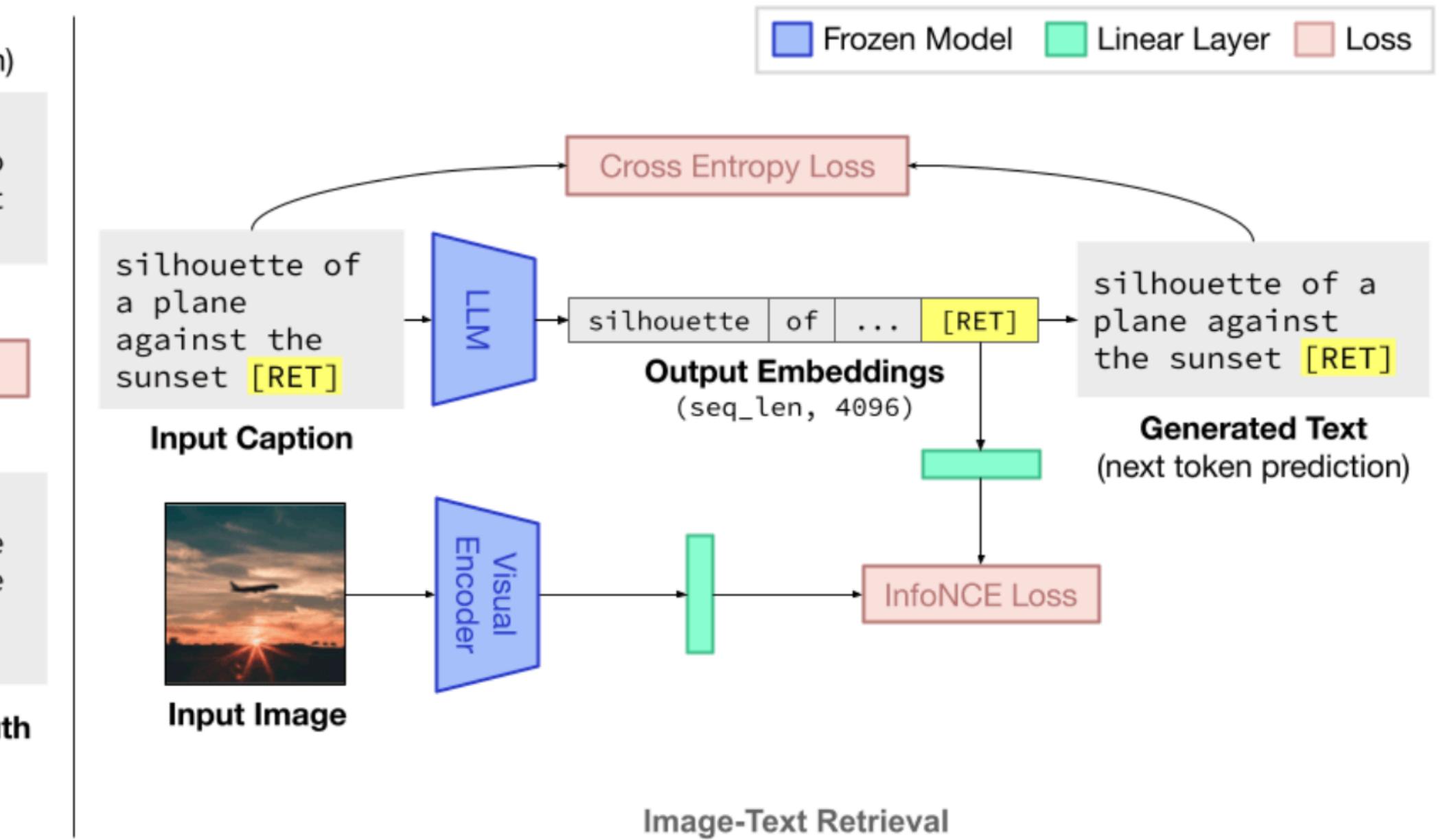
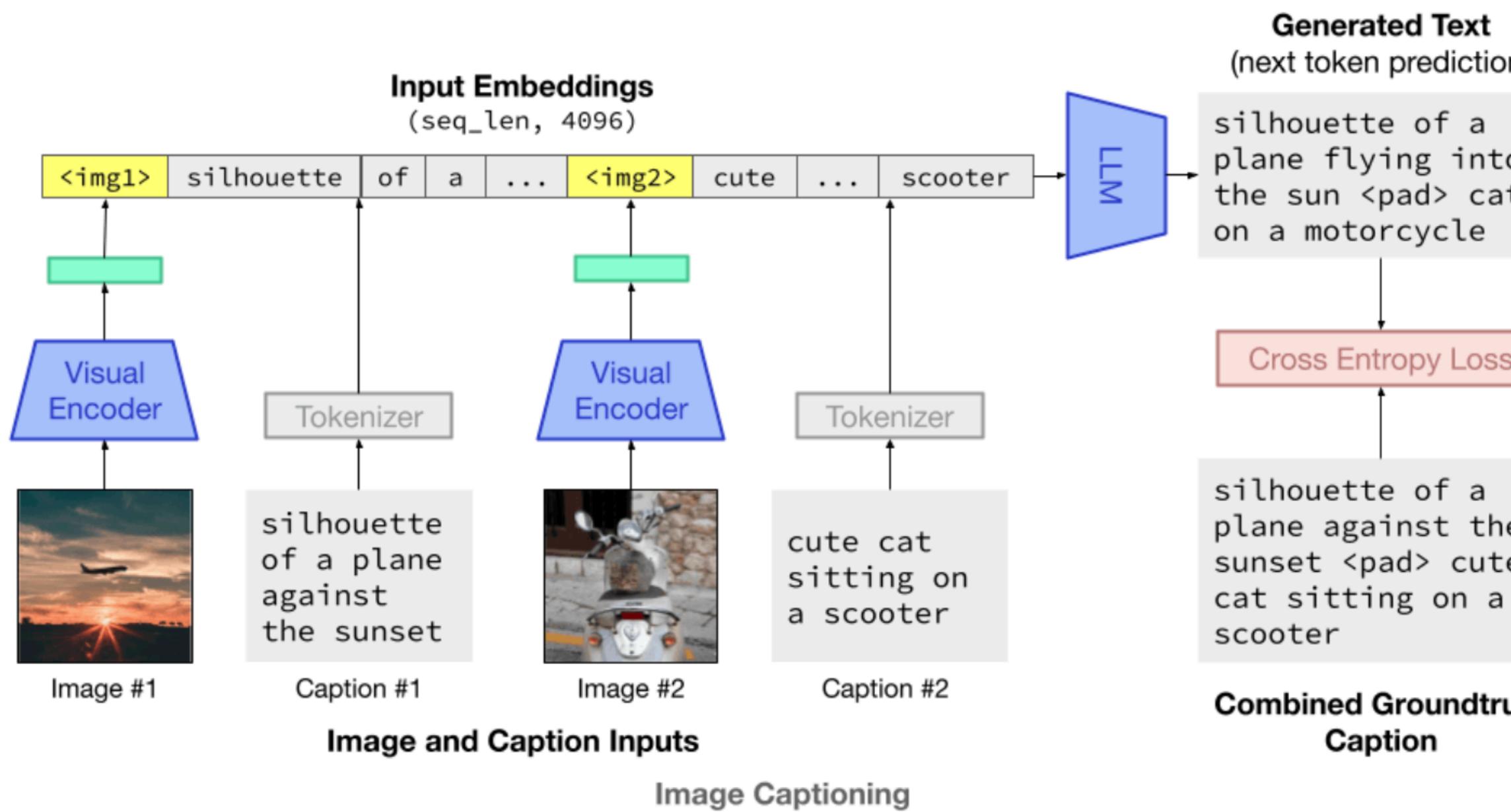
Training set — **image with captions (CC4M)**

Parameters: **30B** (but only 5M are trainable)

**Input:** interleaved text and images

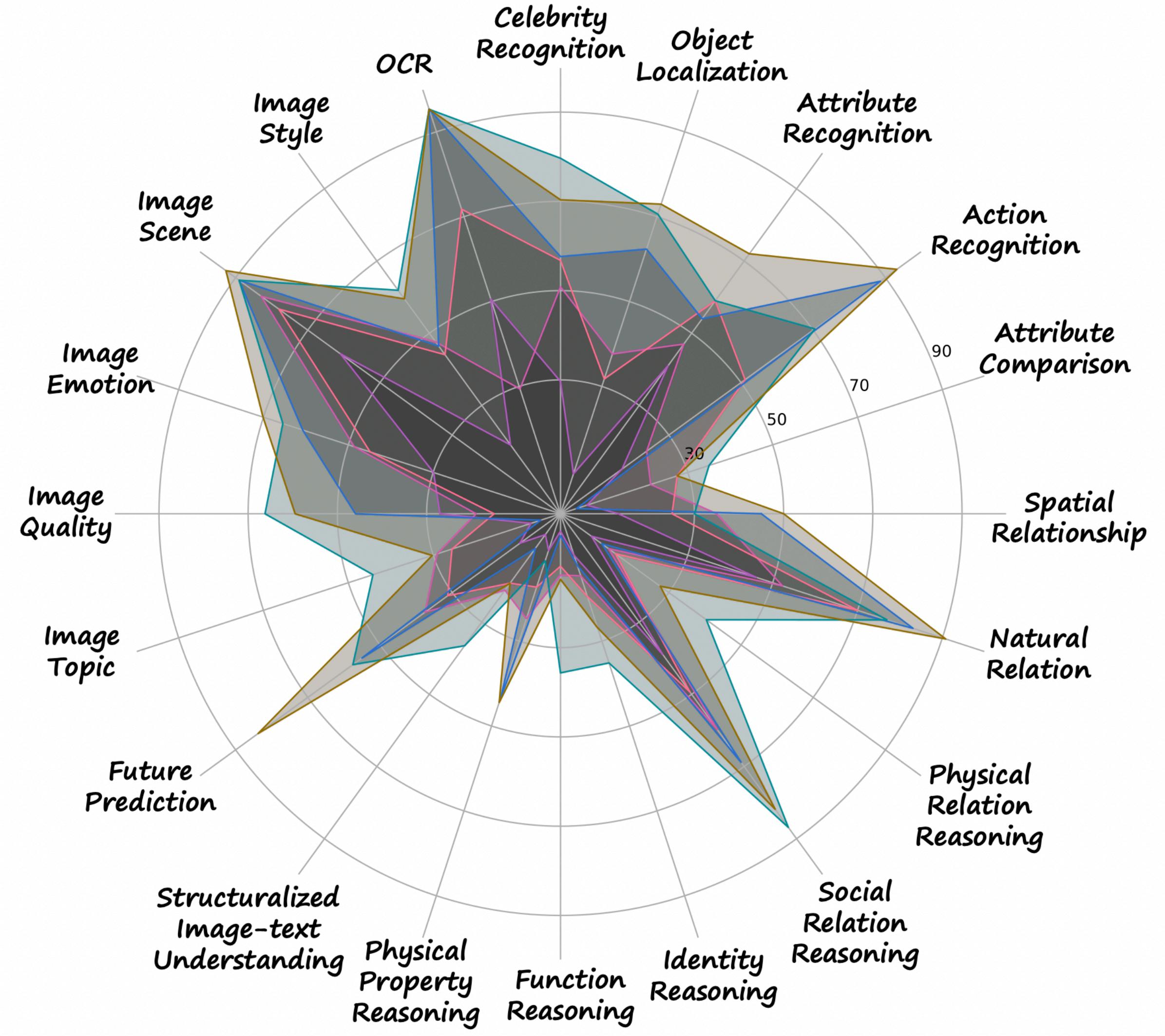
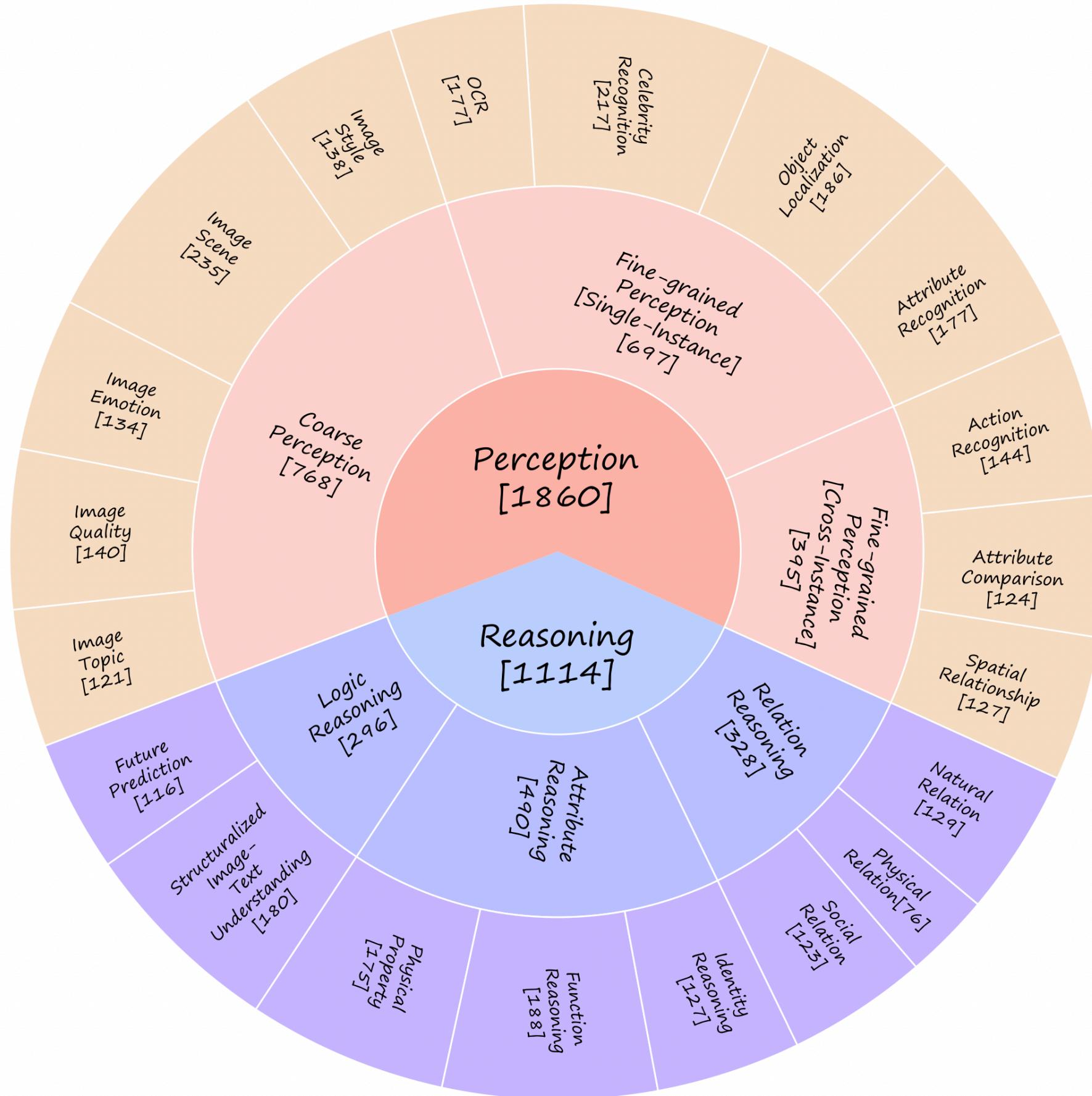
**Output:** text, retrieved images

# FROMAGE



# Multimodal Benchmarks

# MMBench



## Future Prediction



- Q: What will happen next?
- A. this person is gonna cry
  - B. this person is gonna laugh
  - C. this person is gonna get mad
  - D. both A,B, and C

GT: A

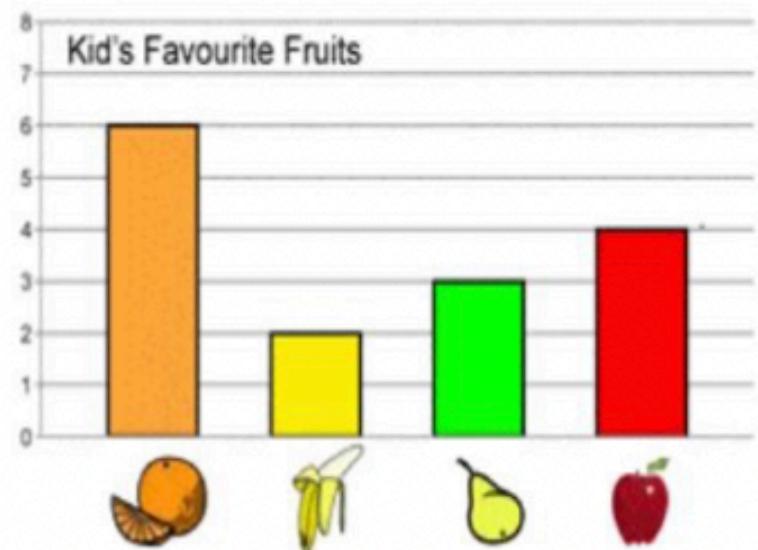


- Q: What will happen next?

- A. the motorcycle is gonna go forward
- B. the motorcycle is gonna crash
- C. the motorcycle is gonna go backward
- D. both A,B, and C

GT: B

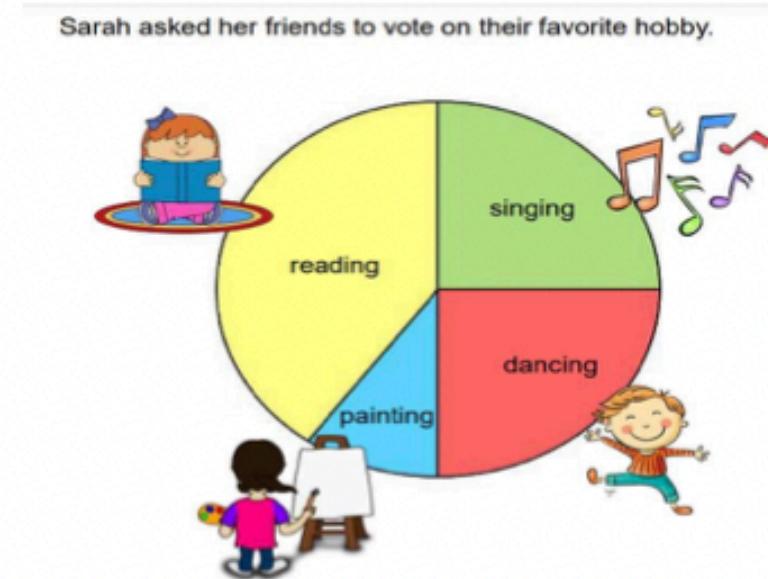
## Structuralized Image-text Understanding



- Q: According to this image, which fruit did the most kids like?

- A. Orange
- B. Banana
- C. Pear
- D. Apple

GT: A



- Q: According to this image, what hobby is liked the least?

- A. Reading
- B. Singing
- C. Painting
- D. Dancing

GT: C

**Table 5: CircularEval results on MMBench dev set (L-2 abilities).** We adopt the following abbreviations: LR for Logical Reasoning; AR for Attribute Reasoning; RR for Relation Reasoning; FP-C for Fine-grained Perception (Cross Instance); FP-S for Fine-grained Perception (Single Instance); CP for Coarse Perception. Methods above the dash line have parameter sizes  $\leq 10B$ ; methods below the dash line have parameter sizes  $> 10B$ . Kosmos-2\* obtains the results using perplexity (PPL).

VLM	Overall	LR	AR	RR	FP-S	FP-C	CP
<b>OpenFlamingo [3]</b>	4.6%	6.7%	8.0%	0.0%	6.7%	2.8%	2.0%
<b>OpenFlamingo v2 [3]</b>	6.6%	4.2%	15.4%	0.9%	8.1%	1.4%	5.0%
<b>MMGPT [13]</b>	15.3%	2.5%	26.4%	13.0%	14.1%	3.4%	20.8%
<b>MiniGPT-4 [43]</b>	24.3%	7.5%	31.3%	4.3%	30.3%	9.0%	35.6%
<b>InstructBLIP [8]</b>	36.0%	14.2%	46.3%	22.6%	37.0%	21.4%	49.0%
<b>VisualGLM [9]</b>	38.1%	10.8%	44.3%	35.7%	43.8%	23.4%	47.3%
<b>LLaVA [26]</b>	38.7%	16.7%	48.3%	30.4%	45.5%	32.4%	40.6%
<b>LLaMA-Adapter-v2 [12]</b>	41.2%	11.7%	35.3%	29.6%	47.5%	38.6%	56.4%
<b><math>\mu</math>-G<sub>2</sub>PT</b>	43.2%	13.3%	38.8%	40.9%	46.5%	38.6%	58.1%
<b>mPLUG-Owl [38]</b>	49.4%	16.7%	53.2%	47.8%	50.2%	40.7%	64.1%
<b>Otter-I [22, 21]</b>	51.4%	32.5%	<b>56.7%</b>	53.9%	46.8%	38.6%	65.4%
<b>Shikra [5]</b>	58.8%	25.8%	<b>56.7%</b>	<b>58.3%</b>	57.2%	<b>57.9%</b>	<b>75.8%</b>
<b>Kosmos-2* [32]</b>	<b>59.2%</b>	<b>46.7%</b>	55.7%	43.5%	<b>64.3%</b>	49.0%	72.5%
<b>PandaGPT [35]</b>	33.5%	10.0%	38.8%	23.5%	27.9%	35.2%	48.3%
<b>MiniGPT-4-13B [43]</b>	42.3%	20.8%	50.7%	30.4%	49.5%	26.2%	50.7%
<b>InstructBLIP-13B [8]</b>	44.0%	19.1%	54.2%	34.8%	47.8%	24.8%	56.4%

# MME

**Perception (Coarse-Grained Tasks)**

**Existence**

[Y] Is there a **elephant** in this image? [Y] Is there a **refrigerator** in this image? [N] Is there a **hair drier** in this image?

**Count**

[Y] Is there a total of **two** person appear in the image? [Y] Are there **two** pieces of pizza in this image? [N] Is there only **one** person appear in the image? [N] Is there only **one** piece of pizza in this image?

**Position**

[Y] Is the motorcycle on the **right** side of the bus? [Y] Is the baby on the **right** of the dog in the image? [N] Is the motorcycle on the **left** side of the bus. [N] Is the baby on the **left** of the dog in the image?

**Color**

[Y] Is there a **red** coat in the image? [Y] Is there a **red** couch in the image? [N] Is there a **yellow** coat in the image? [N] Is there a **black** couch in the image?

**Perception (OCR Task)**

**OCR**

[Y] Is the phone number in the picture "**0131 555 6363**"? [Y] Is the word in the logo "**high time coffee shop**"? [N] Is the phone number in the picture "**0137 556 6363**"? [N] Is the word in the logo "**high tite cofeee shop**"?

**Cognition (Reasoning Tasks)**

**Commonsense Reasoning**

[Y] Should I **stop** when I'm about to **cross** the street? [Y] Is there **one** real cat in this picture? [N] When I see the sign in the picture, can I **cross** the street? [N] Is there **two** real cats in this picture?

**Numerical Calculation**

[Y] Is the answer to the arithmetic question in the image **65**? [Y] Should the value of "a" in the picture equal **3**? [N] Is the answer to the arithmetic question in the image **56**? [N] Should the value of "a" in the picture equal **2**?

**Text Translation**

**老味道** [Y] Appropriate to translate into English '**classic taste**'? [N] Appropriate to translate into English '**strawberry flavor**'?

**共同努力** [Y] Appropriate to translate into English '**work hard together**'? [N] Appropriate to translate into English '**be filled with intrigue**'?

**Code Reasoning**

[Y] Python code. Is the output of the code '**Hello**'? [Y] Python code. Is the output of the code '**0**'? [N] Python code. Is the output of the code '**World**'?

[Y] Python code. Is the output of the code '**I**'? [N] Python code. Is the output of the code '**T**'?