

# Lecture 15:

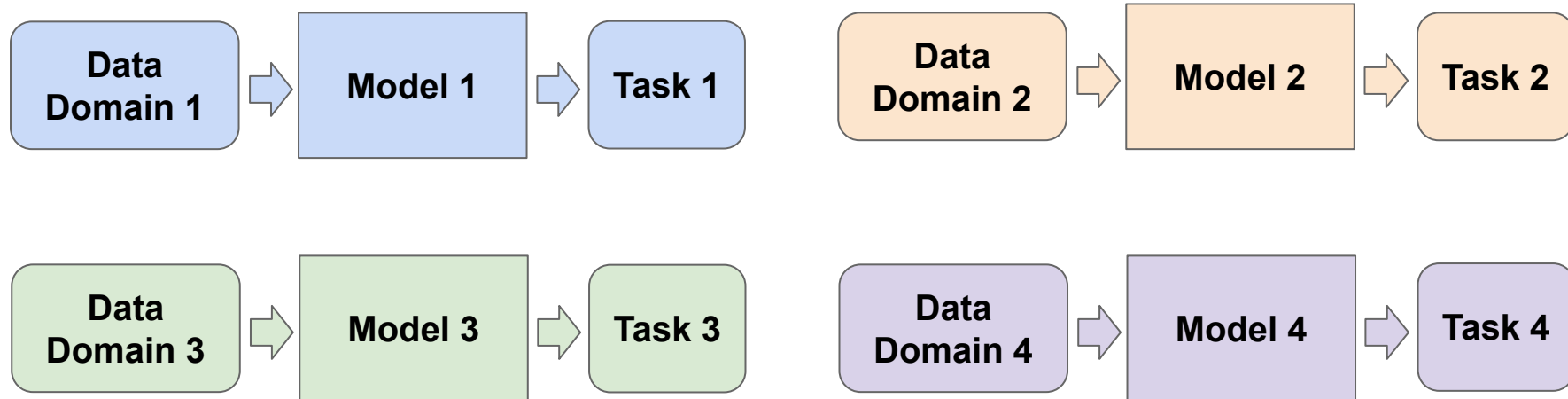
# Multimodal Foundation Models

# Administrative

- Project milestone due May 22nd
- A4 due May 25th
- A5 due June 5th (will be out next week)

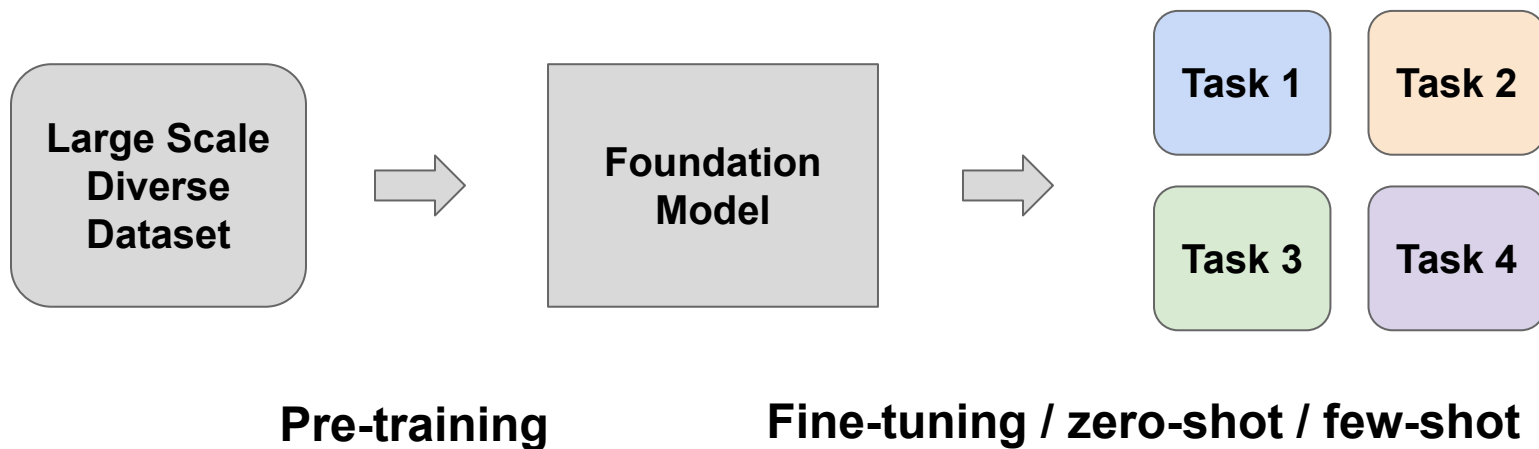
# How have we been thinking about models in this class so far?

*Train a specialized model for each task*



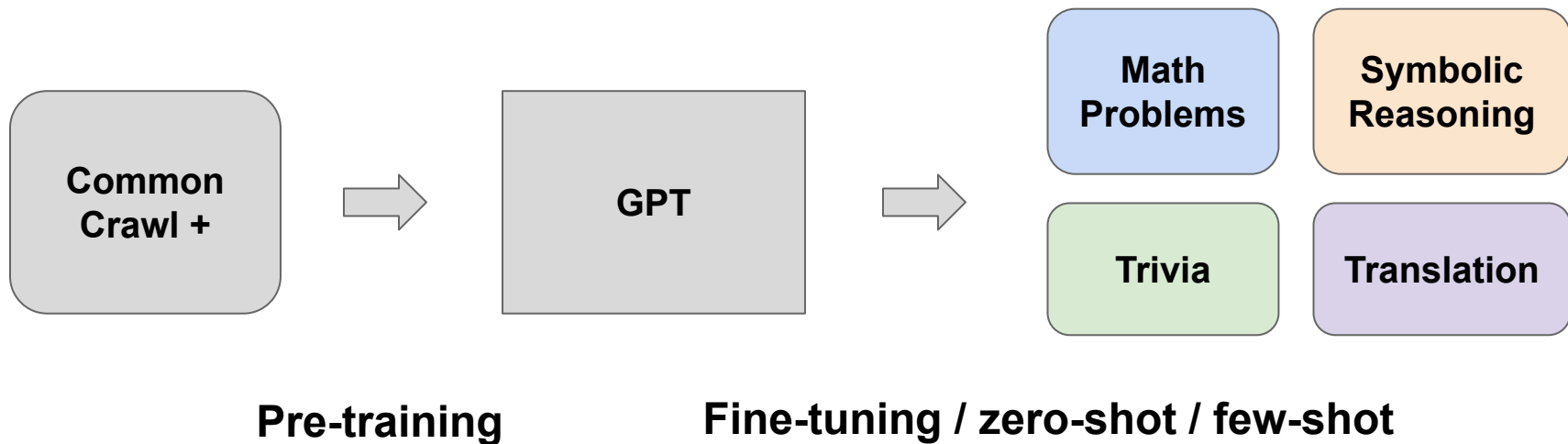
# Foundation Models

Instead: pre-train one model that acts as the *foundation* for many different tasks



# Foundation Models

## Language



# Foundation Models

Multimodal?

# What are Foundation Models?

## **Always see with foundation models:**

- General / robust to many different tasks

## **Often see with foundation models:**

- Large number of parameters
- Large amount of data
- Self-supervised pre-training objective

Bommasani et al., 2021: “On the Opportunities and Risks of Foundation Models”

# There are many classes of Foundation Models

<u>Language</u>	<u>Classification</u>	<u>LM + Vision</u>	<u>And More!</u>	<u>Chaining</u>
ELMo	CLIP	LLaVA	Segment Anything	Visual Programming
BERT	CoCa	Flamingo	Whisper	LMs + CLIP
GPT		GPT-4V	Dalle	
T5		Gemini	Stable Diffusion	
		Molmo	Imagen	

# We will focus on Multimodal Foundation Models

## Language

**ELMo**  
**BERT**  
**GPT**  
**T5**

## Classification

**CLIP**  
CoCa

## LM + Vision

**LLaVA**  
**Flamingo**  
GPT-4V  
Gemini  
**Molmo**

## And More!

Segment Anything  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

**Visual Programming**  
LMs + CLIP

# Let's start with the foundation models for classification

## Language

ELMo  
BERT  
GPT  
T5

## Classification

CLIP  
CoCa

## LM + Vision

LLaVA  
Flamingo  
GPT-4V  
Gemini  
Molmo

## And More!

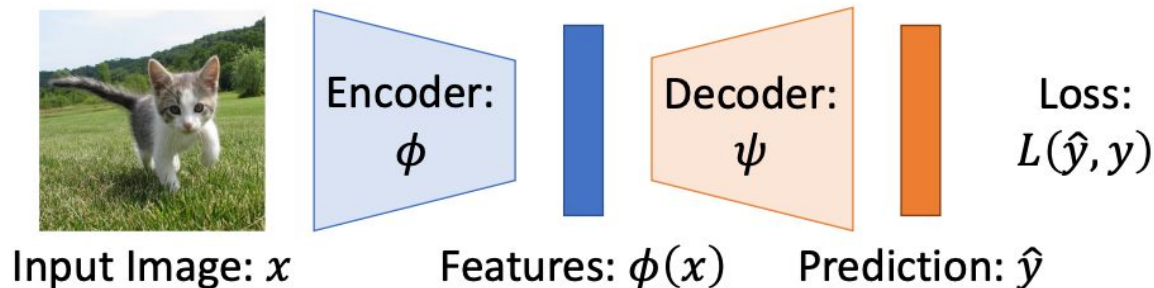
Segment Anything  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

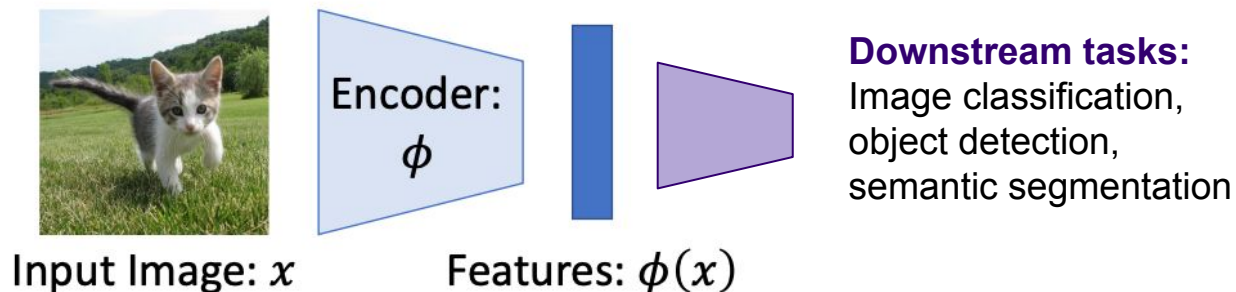
Visual Programming  
LMs + CLIP

# Recall Self-Supervised Learning: Pretext, then Transfer

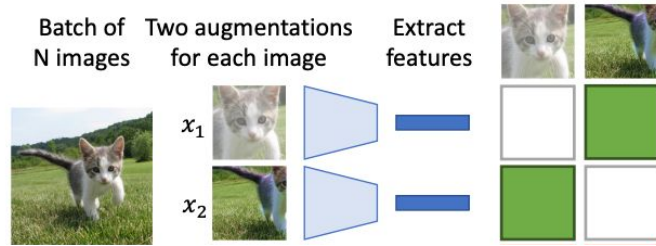
**Step 1:** Pretrain a network on a pretext task that doesn't require supervision



**Step 2:** Transfer encoder to downstream tasks via linear classifiers, KNN, finetuning

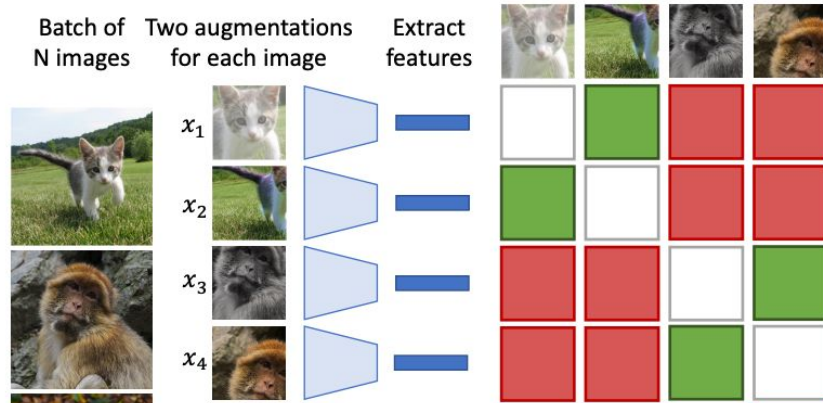


# Recall this **self-supervised** objective from SimCLR



Maximize similarity between features from the same image

# Recall this **self-supervised** objective from SimCLR

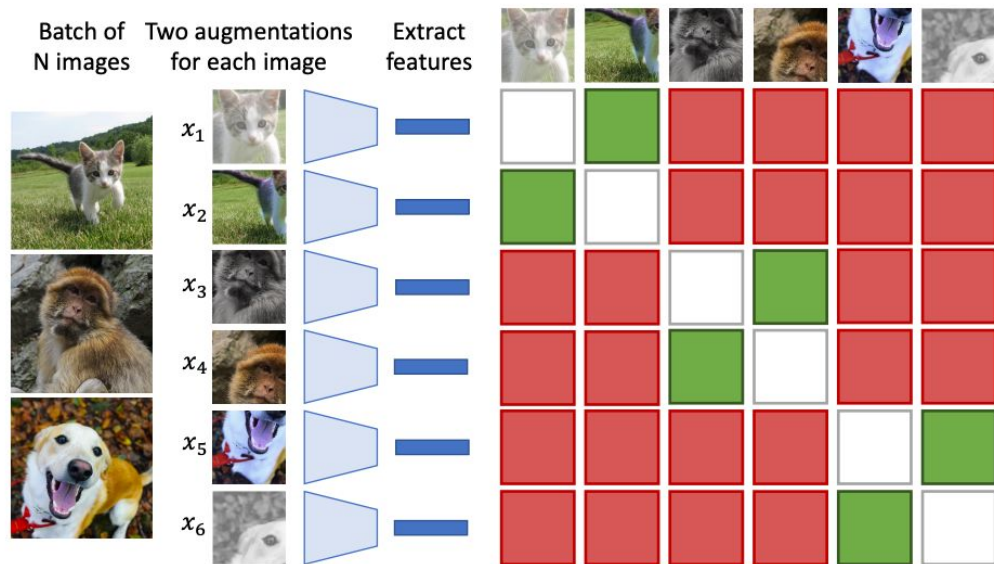


Maximize similarity between features from the same image

Minimize similarity between features from different images

(cosine similarity,  
a.k.a. dot product)

# Recall this **self-supervised** objective from SimCLR

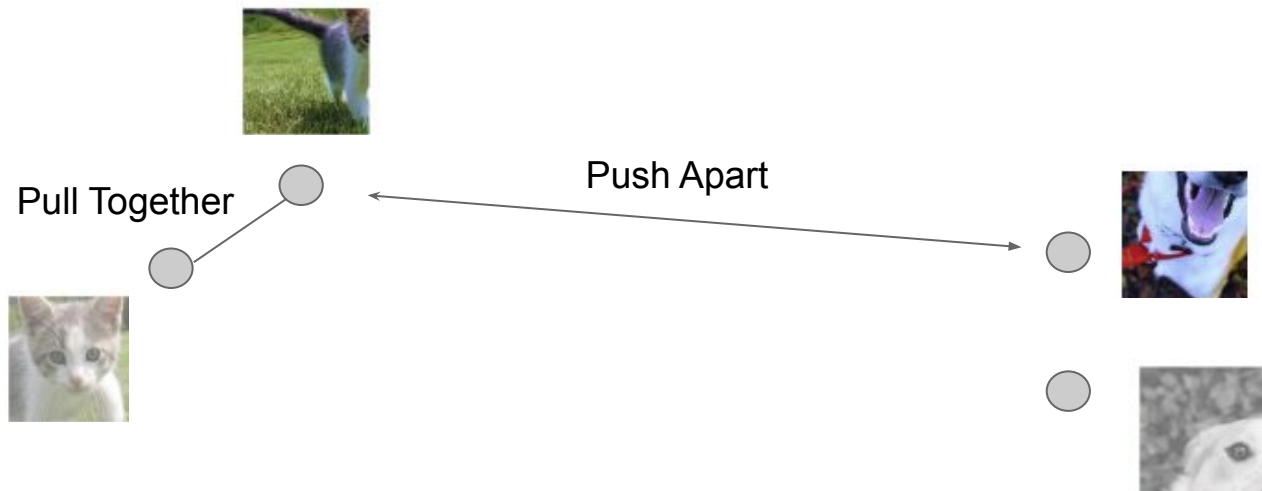


Use Self Supervised learning to learn good image features (Pretext: similarity)

Can train small classifiers on top of these features using supervised learning (Transfer)

Chen et al., 2020: "A Simple Framework for Contrastive Learning of Visual Representations"

The main idea was to learn concepts without **labels**  
→ a self-supervised pre-training objective

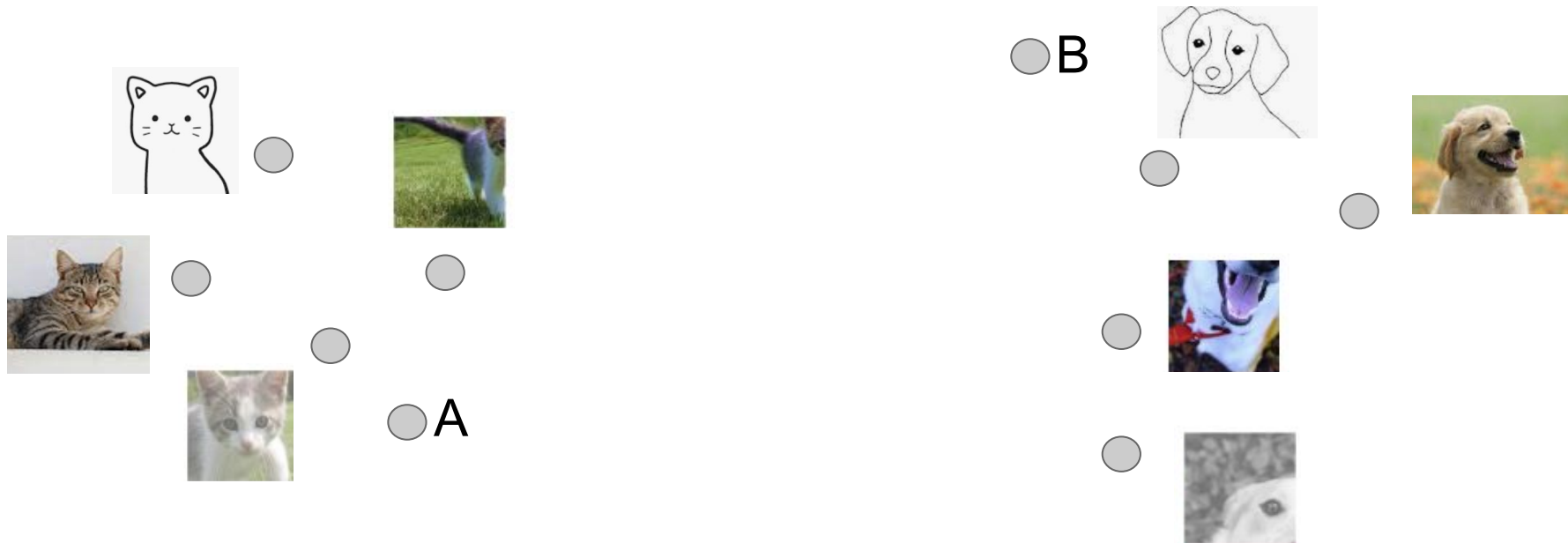


At the end:  
an encoder that  
gives you good  
features (for your  
downstream task)

# The hope was that the encoder could generalize to new instances

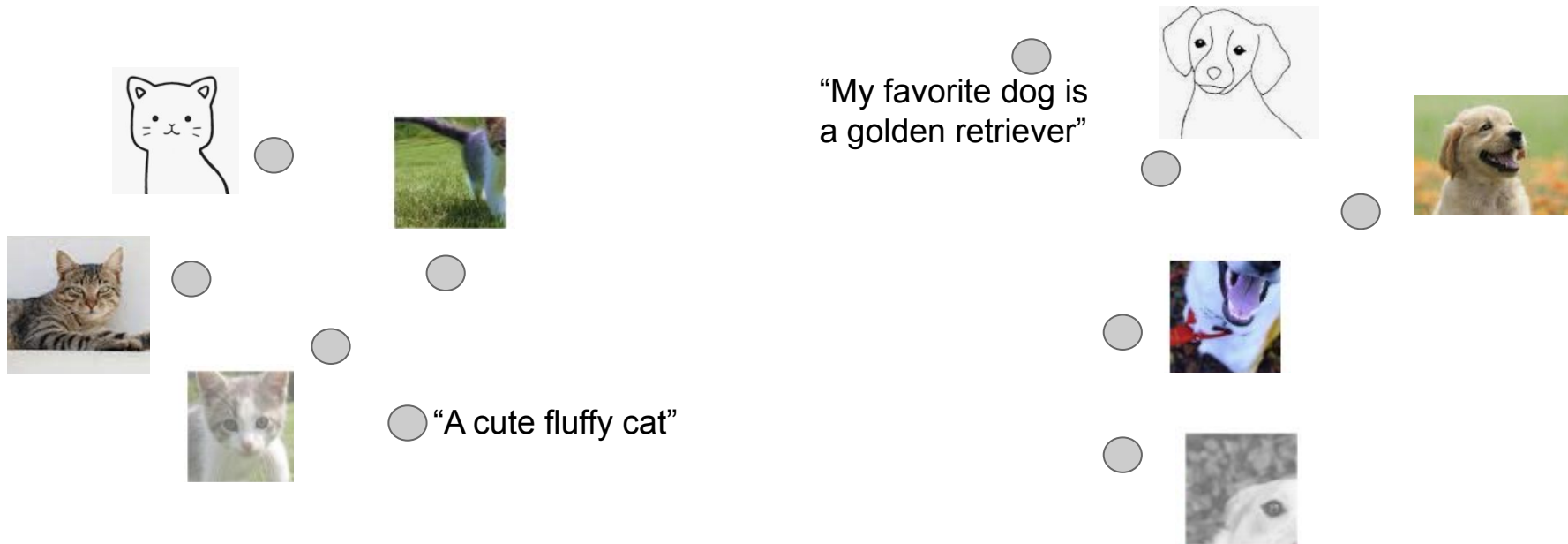


# Can the encoder generalize beyond just images? *To language, perhaps?*



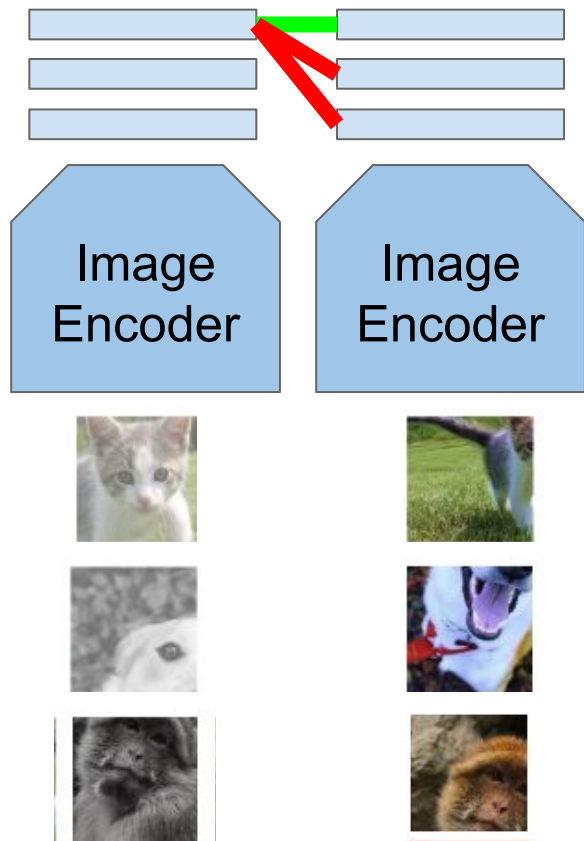
1. "A cute fluffy cat"
2. "My favorite dog is a golden retriever"

# A shared representation space for images and text



This enables multimodal tasks like text-to-image retrieval, and more...

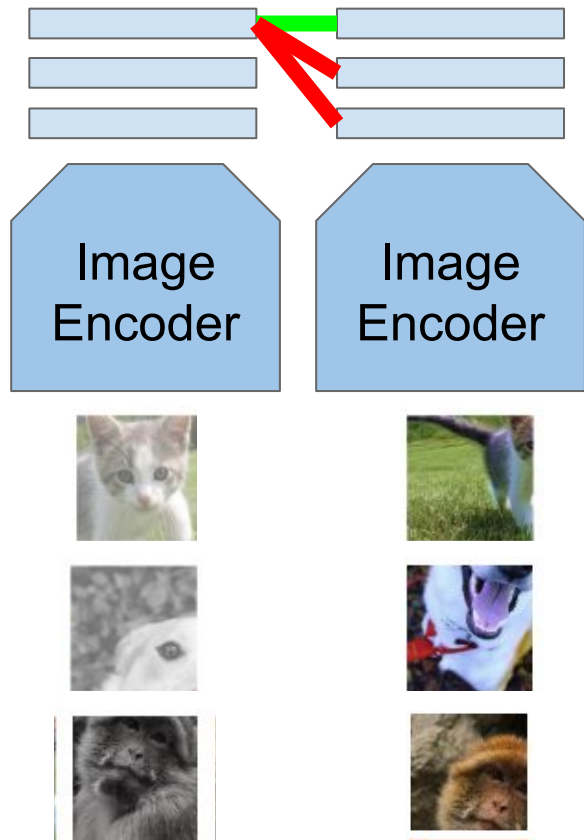
# SimClr



**Maximize** similarity  
between **matching**  
features

**Minimize** similarity  
between **non-matching**  
features

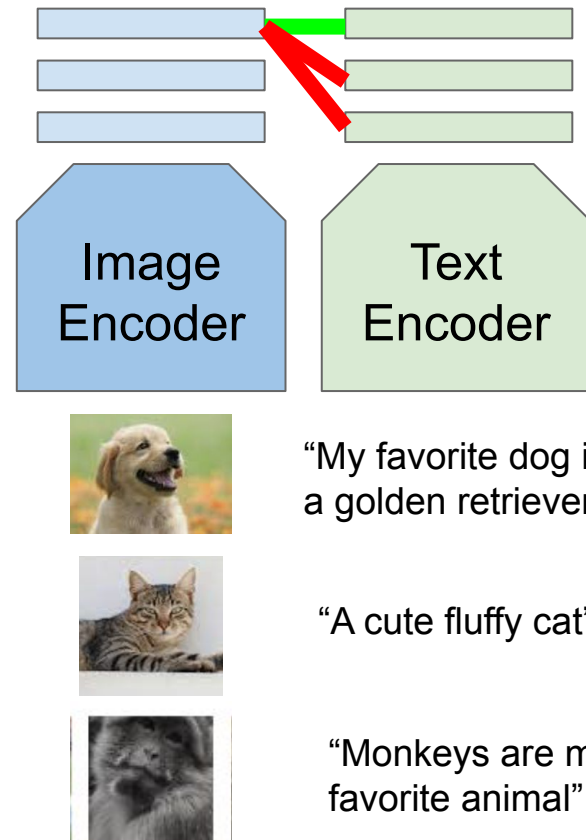
# SimClr



**Maximize** similarity between **matching** features

**Minimize** similarity between **non-matching** features

# CLIP



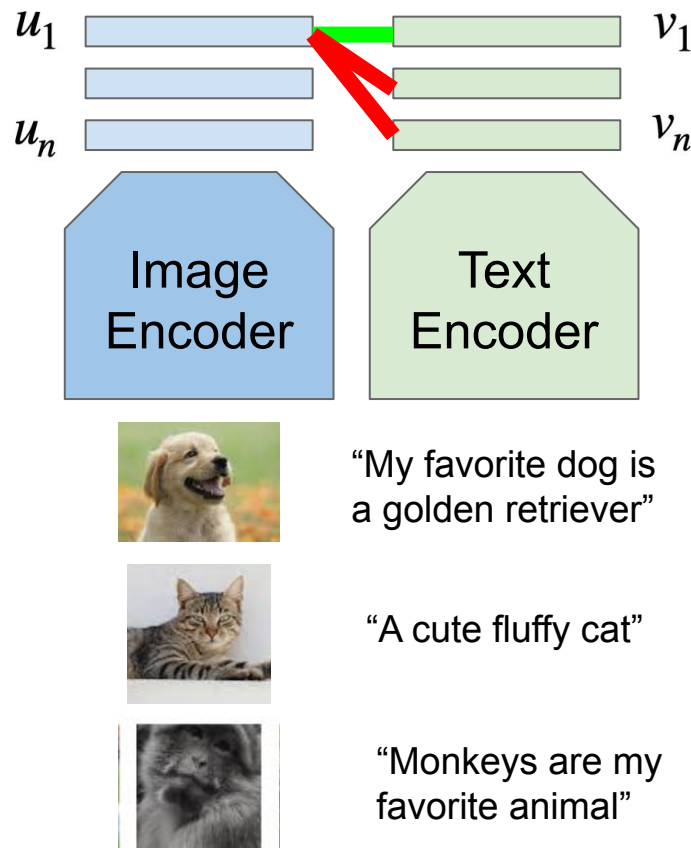
# CLIP Training

Start from scratch: randomly initialized image and text encoders.

Show the model a batch of image-text pairs.

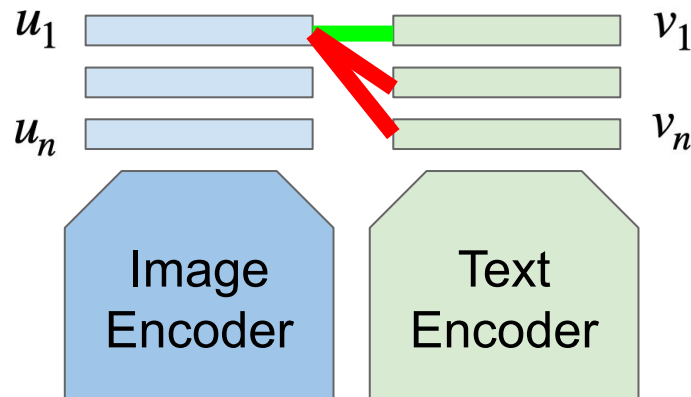
Train the model to maximize similarity between image features and text features of matching pairs...

...and minimize similarity between non-matching features.



# CLIP Training Objective

$$\sum_{i=1}^n -\log \left( \frac{e^{\langle u_i, v_i \rangle}}{\sum_{j=1}^n e^{\langle u_i, v_j \rangle}} \right)$$



“My favorite dog is a golden retriever”



“A cute fluffy cat”

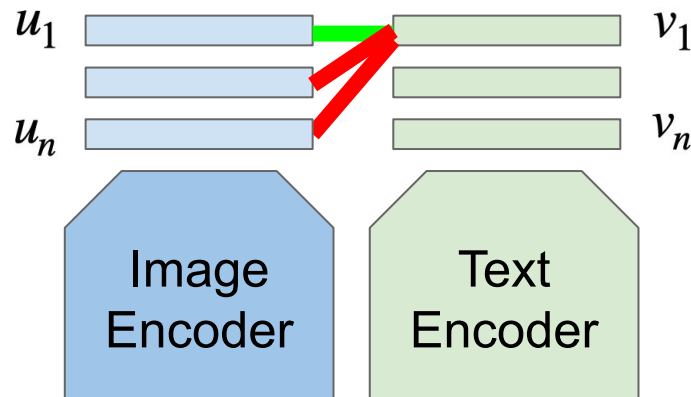


“Monkeys are my favorite animal”

# CLIP Training Objective

$$\sum_{i=1}^n -\log \left( \frac{e^{\langle u_i, v_i \rangle}}{\sum_{j=1}^n e^{\langle u_i, v_j \rangle}} \right) + \sum_{i=1}^n -\log \left( \frac{e^{\langle u_i, v_i \rangle}}{\sum_{j=1}^n e^{\langle u_j, v_i \rangle}} \right)$$

“contrastive loss”



“My favorite dog is a golden retriever”



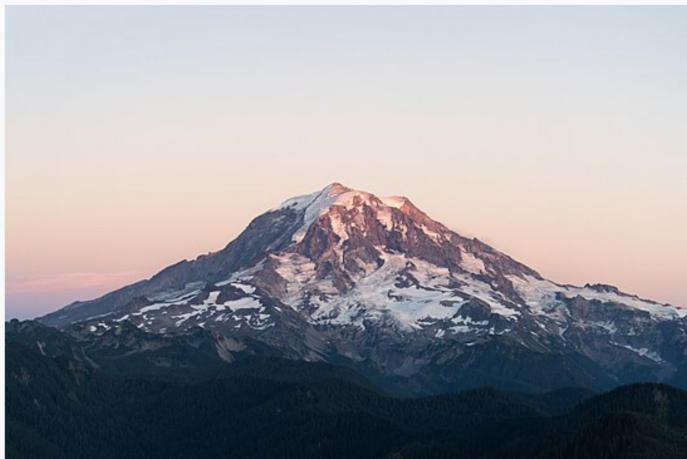
“A cute fluffy cat”



\$\$\$?

“Monkeys are my favorite animal”

# Lots of image-text data can be found online



Mount Rainier's northwestern slope viewed aerially just before sunset on September 6, 2020

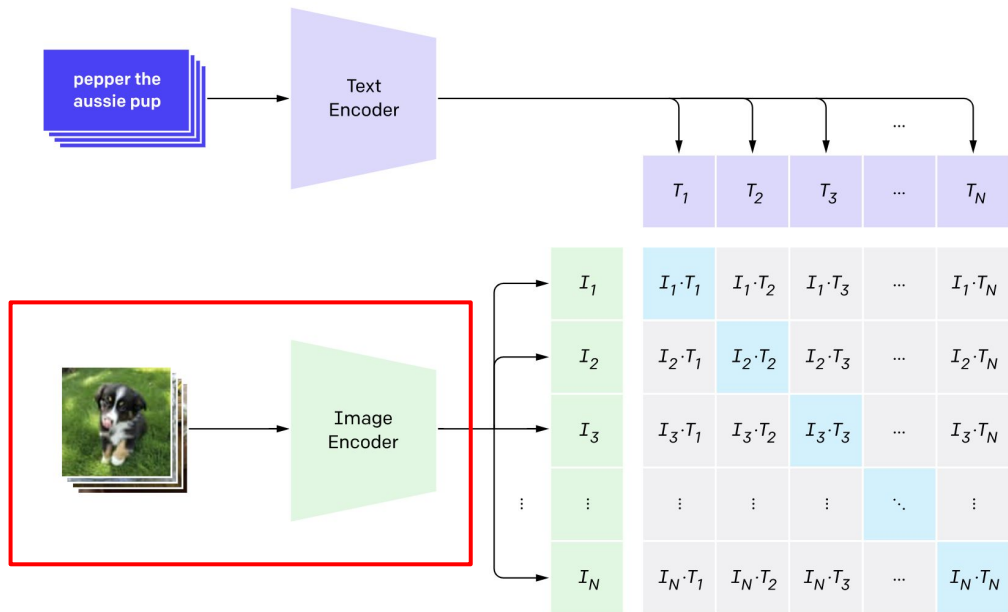
CLIP training data was scraped at scale from images and their associated alt-text from the internet

Initially, ~400M, but has since scaled to 5B+

[https://en.wikipedia.org/wiki/Mount\\_Rainier](https://en.wikipedia.org/wiki/Mount_Rainier)

# CLIP Training Objective

## 1. Contrastive pre-training

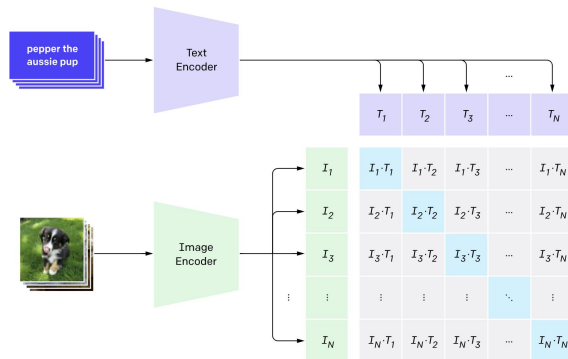


At the end of training, you have a model that can give you a similarity score between an image and a text.

You also have a very good image encoder, which you can use to get image representations!

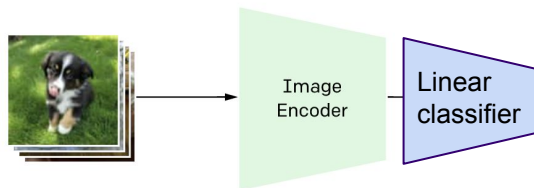
# Using pre-trained models

**Step 1:** Pretrain a network on a pretext task that doesn't require supervision



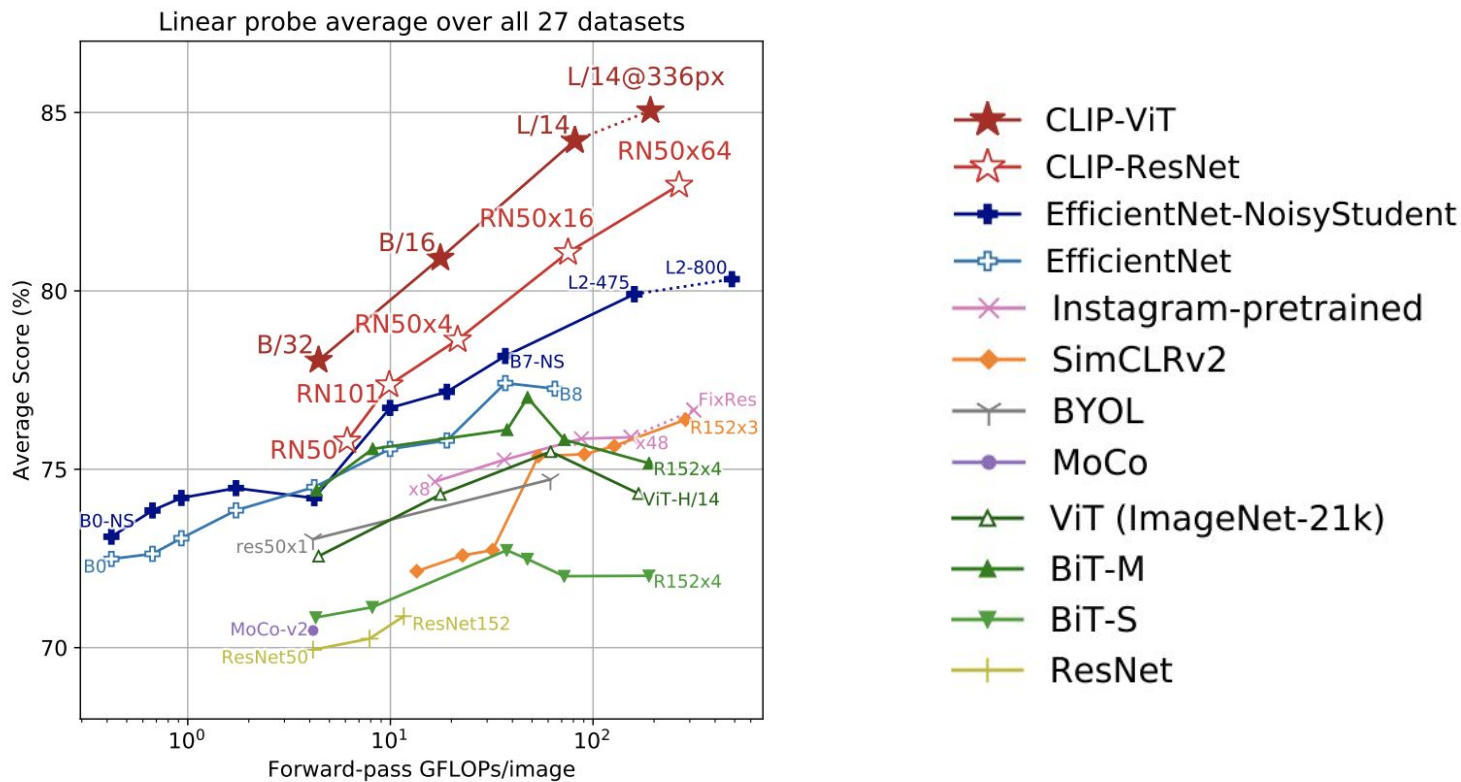
**Pre-training tasks:**  
Contrastive Objective

**Step 2:** Transfer encoder to downstream tasks via **linear classifiers**



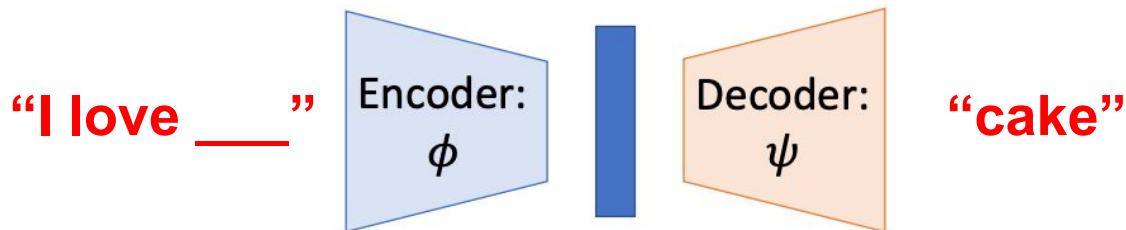
**Downstream tasks:**  
Image classification,  
object detection,  
semantic segmentation

# CLIP features w/ **linear probe** across multiple datasets

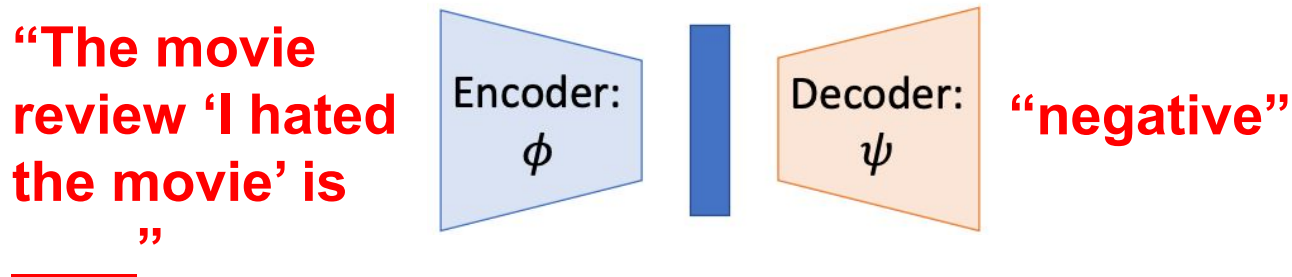


# Remember: We can use pretrained LLMs **zero-shot** for downstream tasks

**Step 1:** Pretrain a network on a pretext task that doesn't require supervision

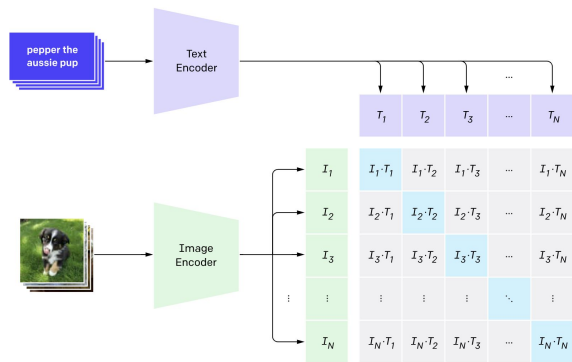


**Step 2:** Instead of finetuning, use the model out of the box in a creative way!



# How do we use pre-trained **VLMs zero-shot** for downstream tasks?

**Step 1:** Pretrain a network on a pretext task that doesn't require supervision



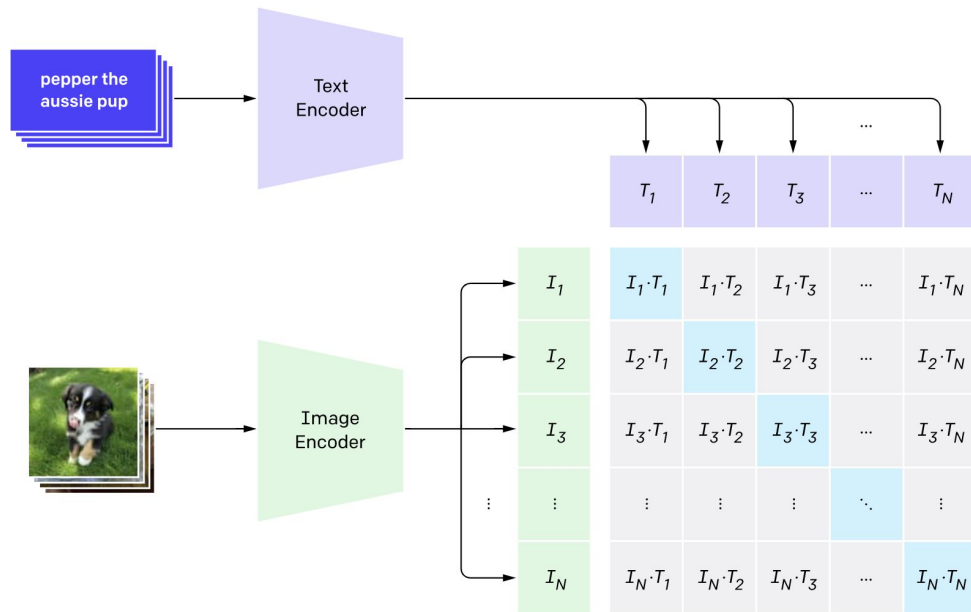
**Pre-training tasks:**  
Contrastive Objective

**Step 2:** Instead of finetuning, use the model out of the box in a creative way!

**Out of the box classification  
(No fine-tuning)**

# CLIP Training Objective

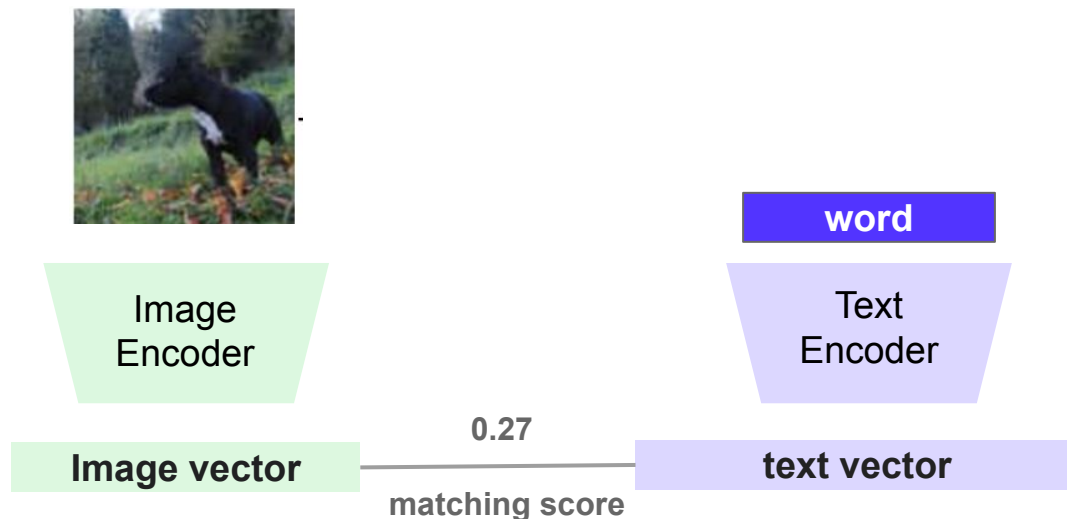
## 1. Contrastive pre-training



At the end of training, you have a model that can give you a similarity score between an image and a text.

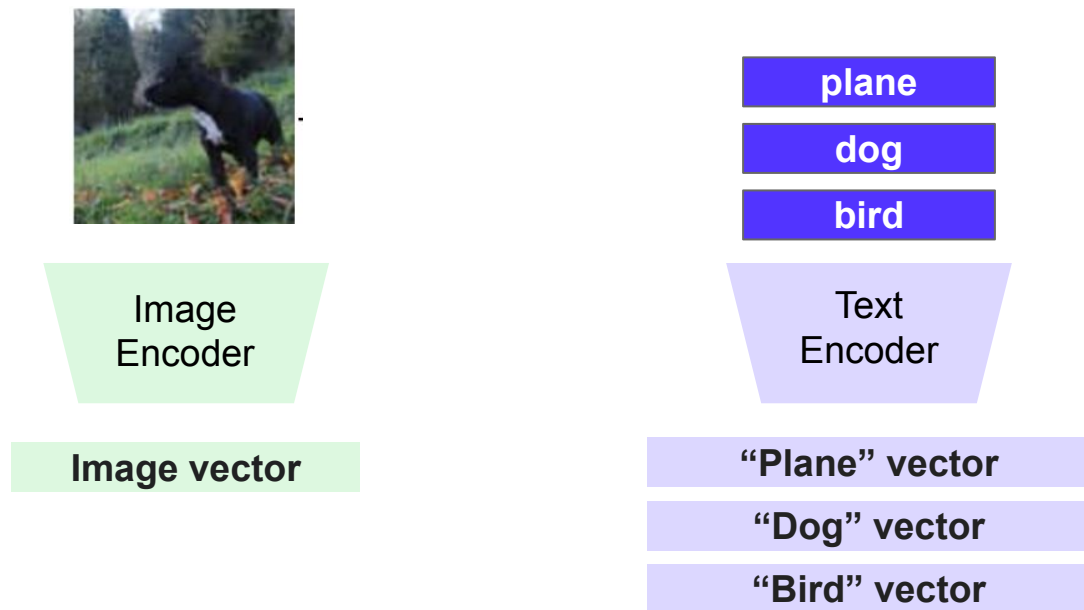
You also have a very good image encoder, which you can use to get image representations!

# Zero-Shot CLIP Classification



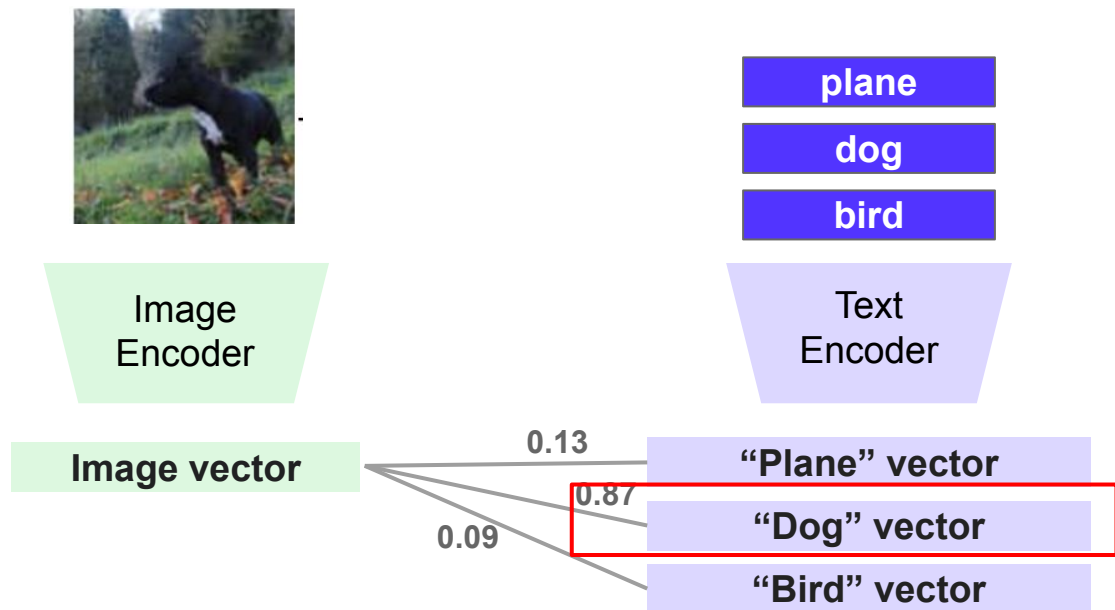
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

# Zero-Shot CLIP Classification



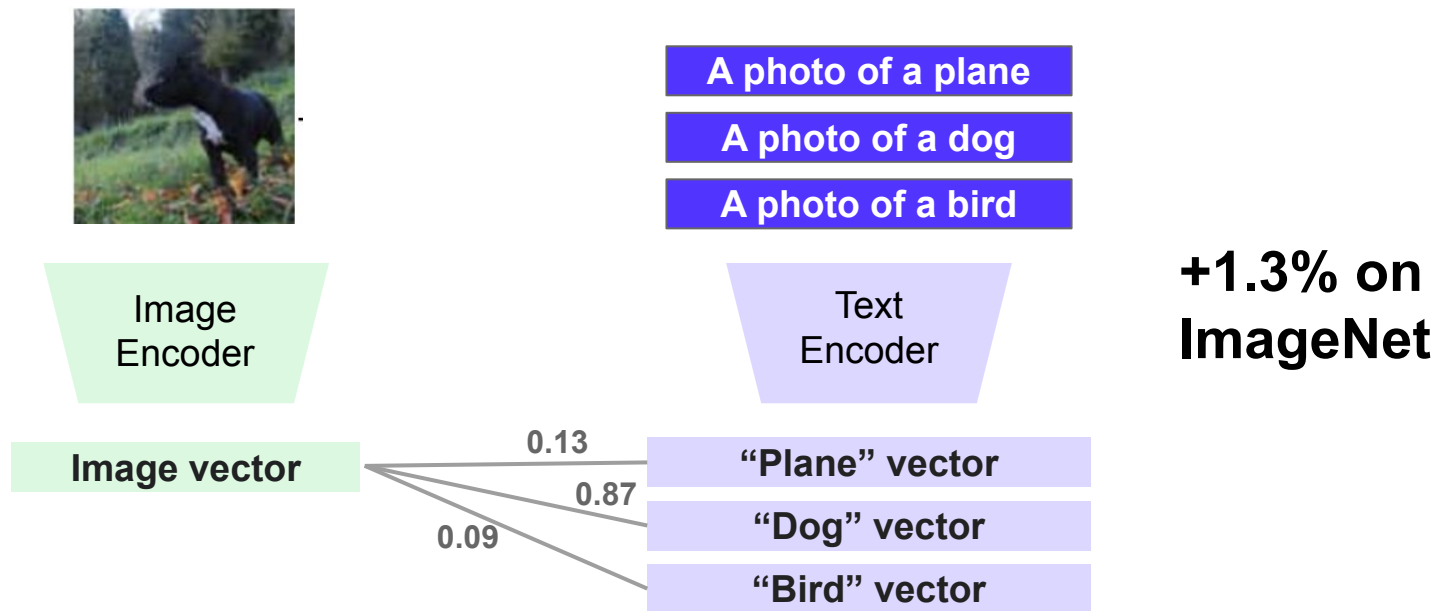
Radford et al “Learning Transferable Visual Models From Natural Language Supervision”

# Zero-Shot CLIP Classification



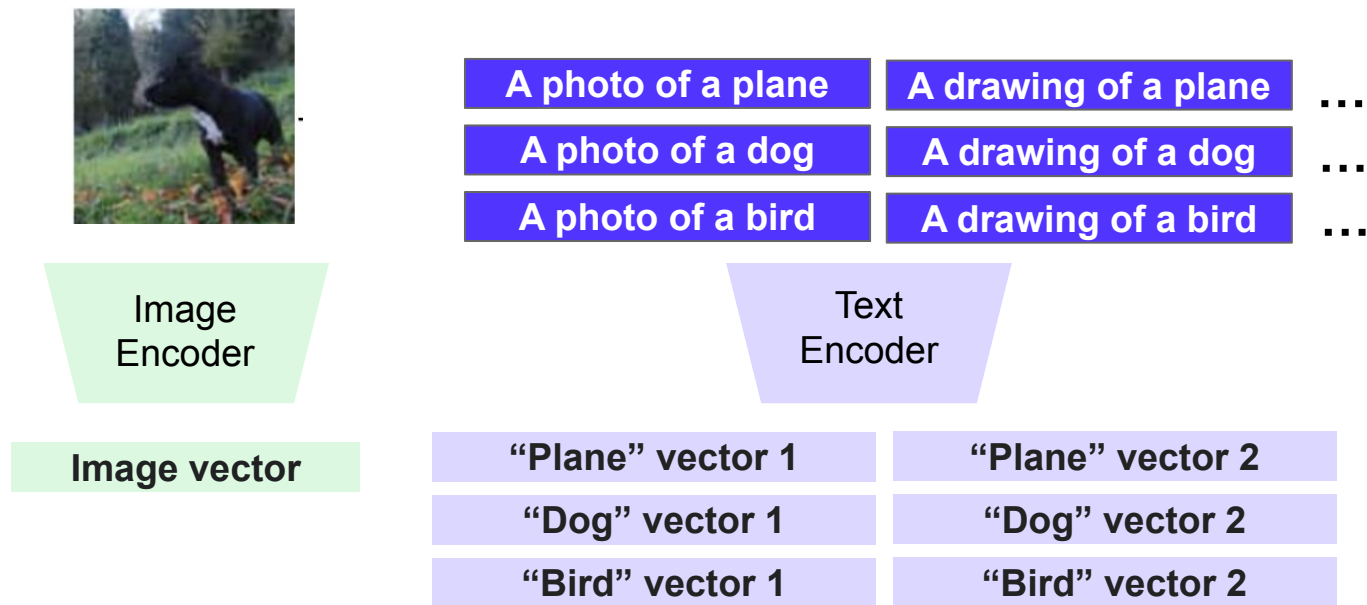
Radford et al “Learning Transferable Visual Models From Natural Language Supervision”

# Zero-Shot CLIP Classification



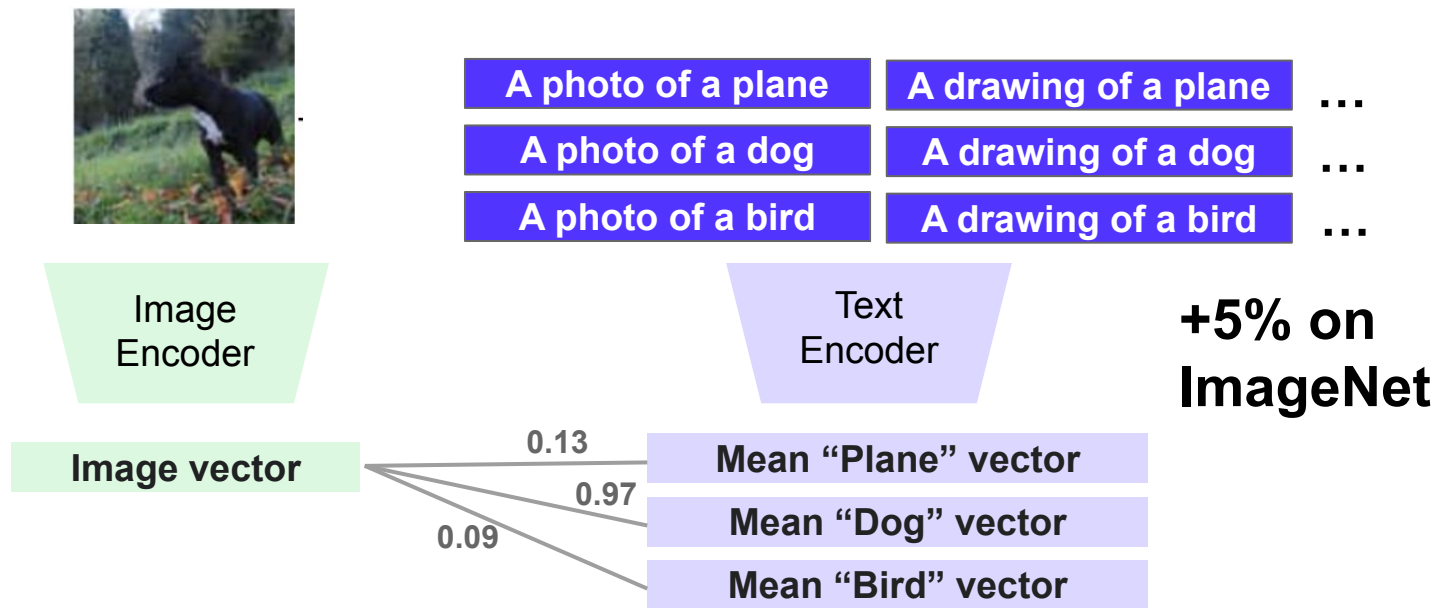
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

# Zero-Shot CLIP Classification



Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

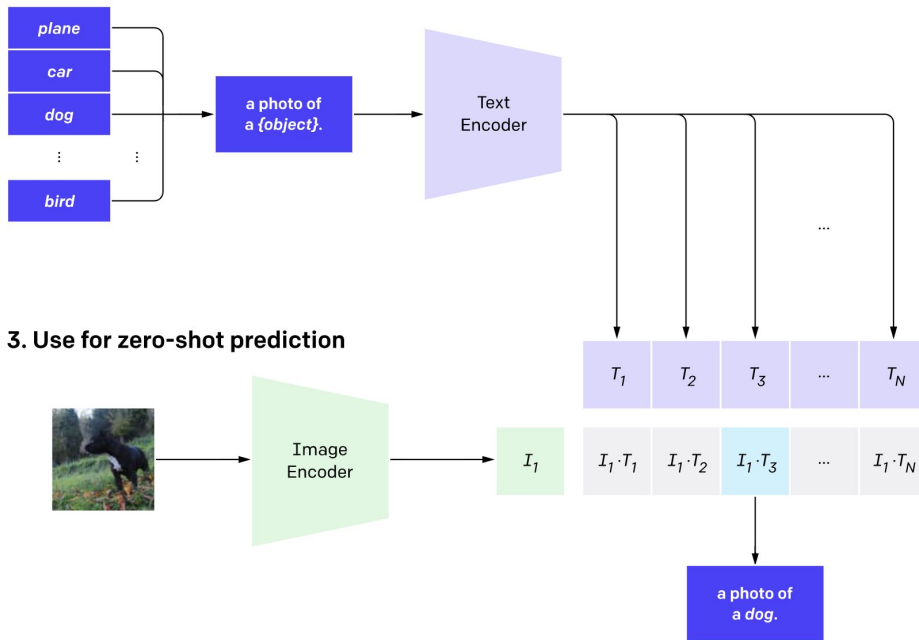
# Zero-Shot CLIP Classification



Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

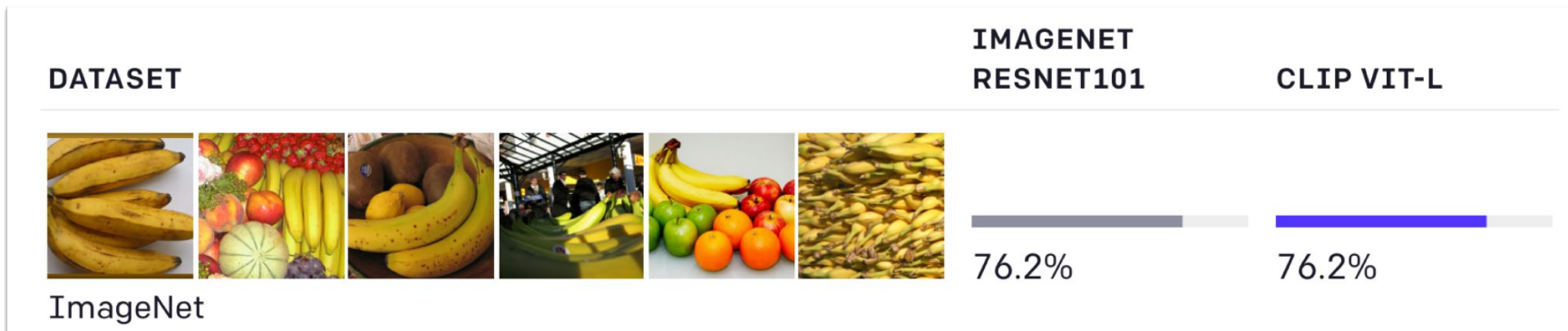
# That's it! Now, you can use CLIP as a foundation model for image classification for any dataset

## 2. Create dataset classifier from label text



Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

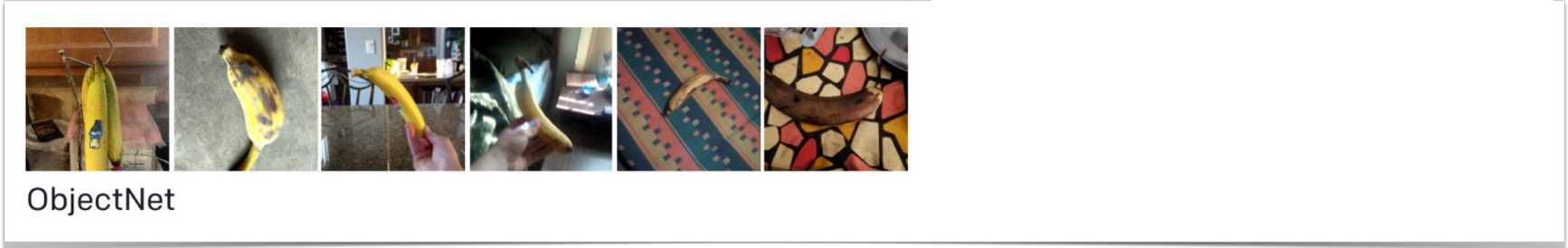
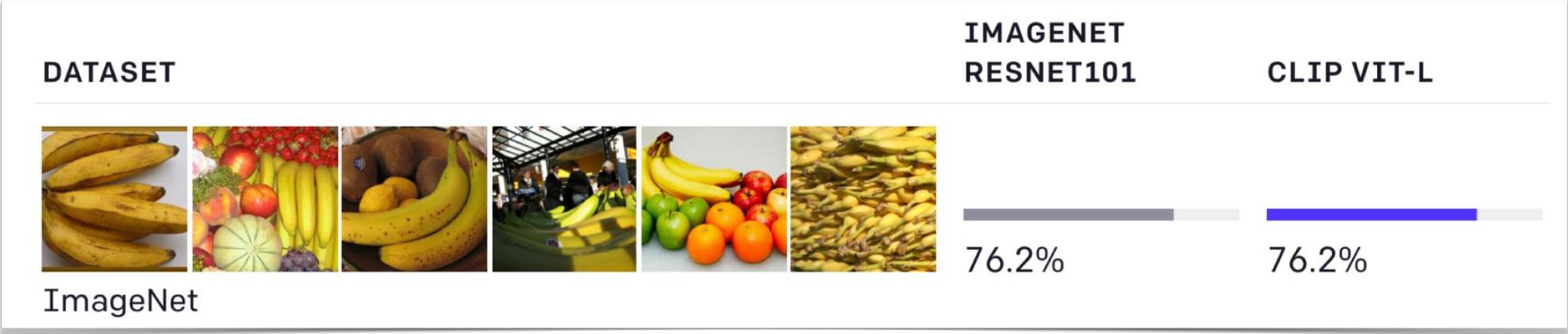
# ImageNet Accuracy



Matches the accuracy of ResNet 101 (that has been trained on human-labeled ImageNet) with no supervised human labels at all!

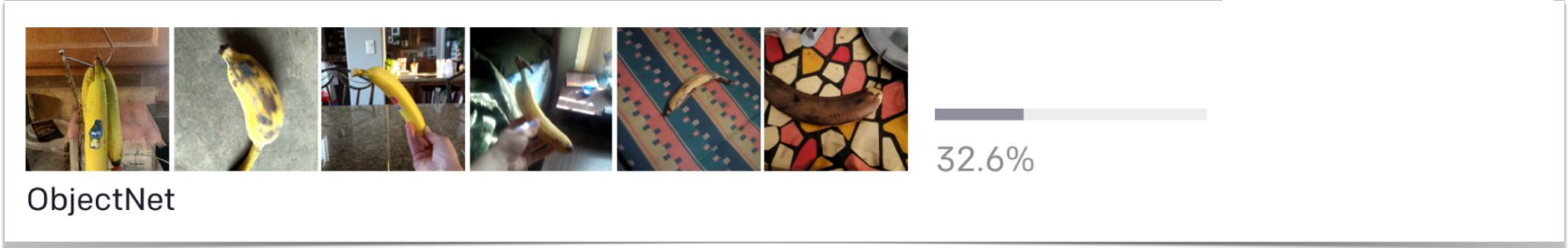
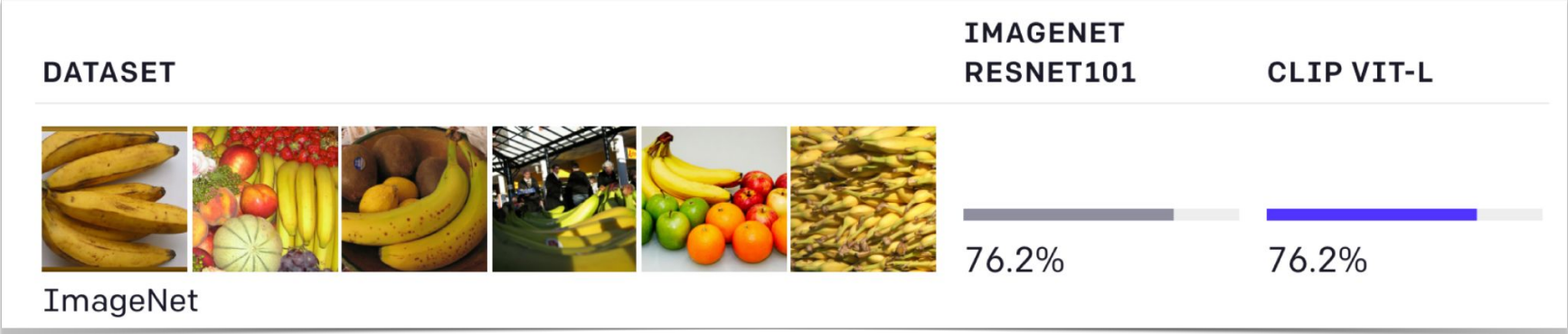
Radford et al “Learning Transferable Visual Models From Natural Language Supervision”

# ImageNet Accuracy



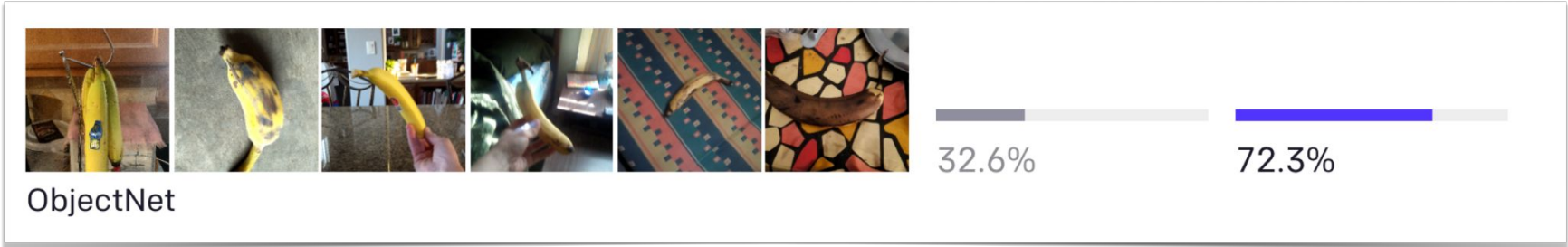
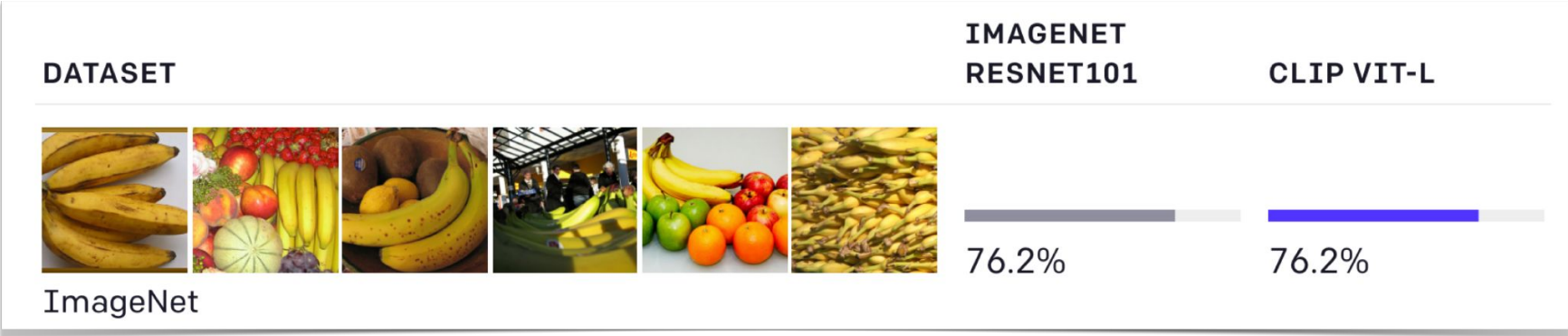
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

# ImageNet Accuracy








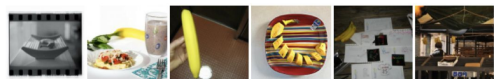
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

# ImageNet Accuracy



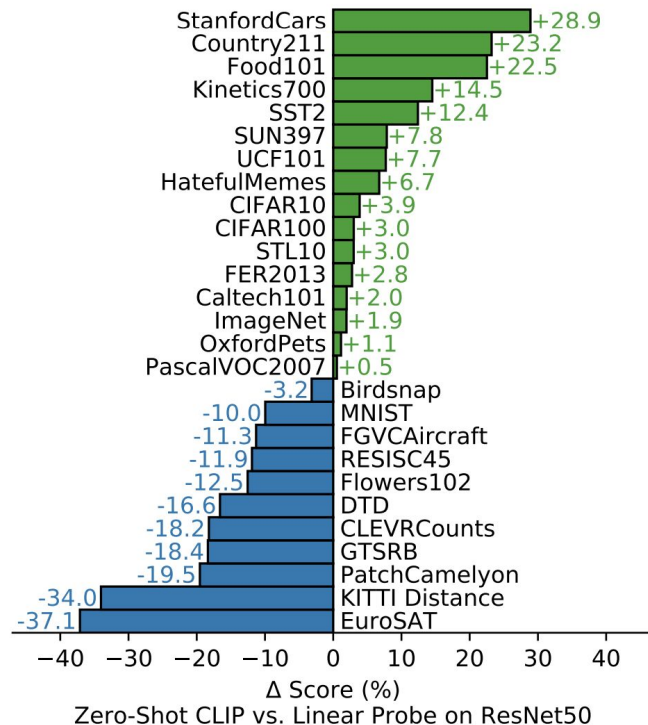
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

CLIP performance is also great on graphic images, sketches, and adversarial datasets.

DATASET	IMAGENET RESNET101	CLIP VIT-L
 ImageNet	76.2%	76.2%
 ImageNet V2	64.3%	70.1%
 ImageNet Rendition	37.7%	88.9%
 ObjectNet	32.6%	72.3%
 ImageNet Sketch	25.2%	60.2%
 ImageNet Adversarial	2.7%	77.1%

Radford et al “Learning Transferable Visual Models From Natural Language Supervision”

# Accuracy on other datasets



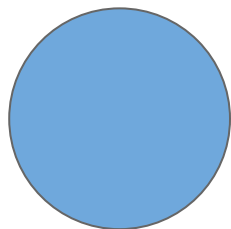
Radford et al “Learning Transferable Visual Models From Natural Language Supervision”

# Why does CLIP perform so well?

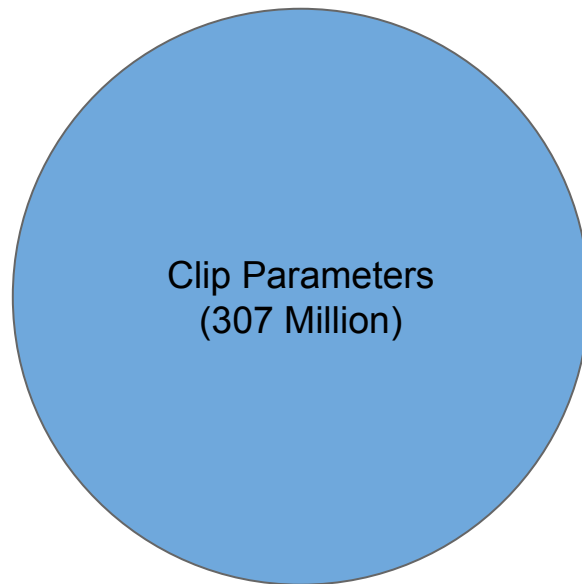
How can no labels beat labels??

**Scale!**

# CLIP scaled up the model parameters with the transformer architecture



ImageNet ResNet Parameters  
(44.5 Million)

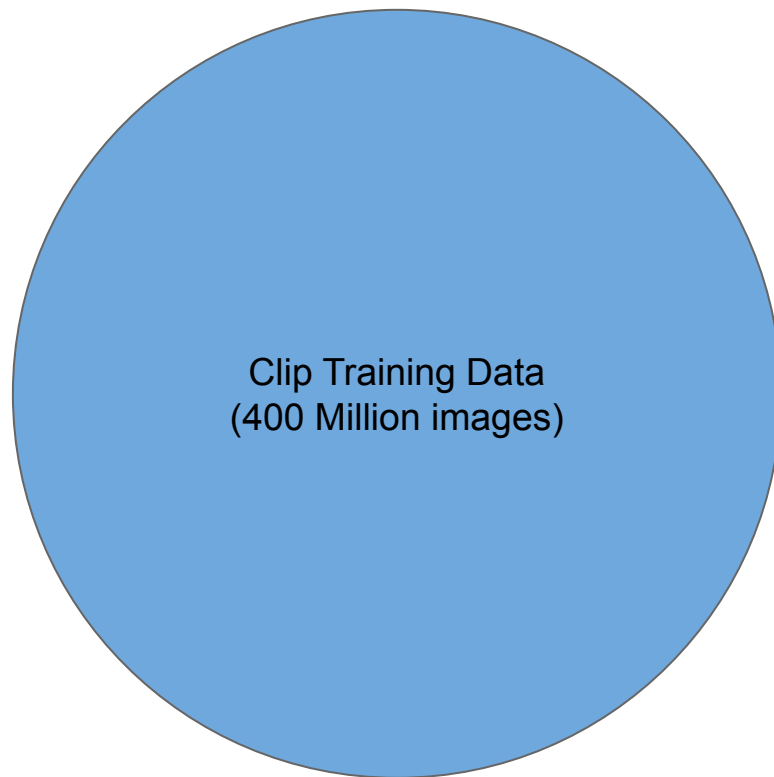


Clip Parameters  
(307 Million)

# CLIP scaled up the training data by scraping image-text pairs from the internet



ImageNet ResNet Training Data  
(1.28 Million)



Clip Training Data  
(400 Million images)

# Advantages of CLIP-style models

1. Dot product is super efficient
  - a. Easy to train (enables scaling)
  - b. Fast inference, e.g., retrieval over 5B images
2. Open-vocabulary (zero-shot generalization)
3. Can be chained with other models (LMs + CLIP)



April 2022, Tristan Thrush et al:

CLIP can't distinguish between:

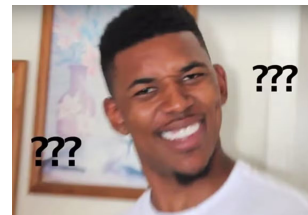


there is a mug in some grass



there is some grass in a mug

...



# Disadvantages of CLIP-style models

## 1. Rely too heavily on batch size to learn concepts

Increasing batch size helps you understand fine-grained concepts



Batch size: 4

“animal”

Batch size: 100

“dog”

Batch size: **32000**

“Welsh Corgi”

# Disadvantages of CLIP-style models

1. Rely too heavily on batch size to learn concepts

Increasing batch size helps you understand fine-grained concepts

**BUT**, there's a limit to how fine-grained you can get this way!

Even in a batch of 32K, it's unlikely you see both “a mug in some grass” and “some grass in a mug” → so, you don't learn to distinguish between them.

# Disadvantages of CLIP-style models

## 1. Rely too heavily on batch size to learn concepts

Winoground



there is a mug in  
some grass



there is some  
grass in a mug

ARO



the grass is eating the horse 81%

the horse is eating the grass 78%

CREPE



✓ Crepe on a skillet. 🗨️ ?

✗ Boats on a skillet.

✗ Crepe under a skillet.

✗ Crepe on a dog.

...

Paper	Venue	Perturbation
Winoground	CVPR 2022 (Oral)	word order
VL-Checklist	EMNLP 2022	replacements
When-and-Why	ICLR 2023 (Oral)	word order
CREPE	CVPR 2023 (Spotlight)	word order replacements negations
SVLC	CVPR 2023	replacements
DAC	NeurIPS 2023 (spotlight)	replacements
What's Up	EMNLP 2023	replacements
Text encoders...	EMNLP 2023	word order
SugarCREPE	NeurIPS 2023	word order replacements additions
COLA	NeurIPS 2023 D&B	replacements

# Disadvantages of CLIP-style models

1. Rely too heavily on batch size to learn concepts
2. Image-level captions are insufficient supervision



“living room”



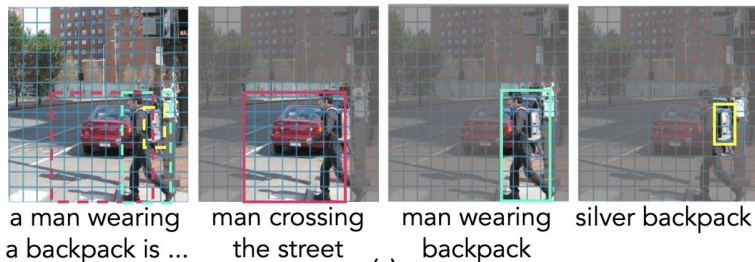
“house plants”

“couch”

# Disadvantages of CLIP-style models

1. Rely too heavily on batch size to learn concepts
2. Image-level captions are insufficient supervision

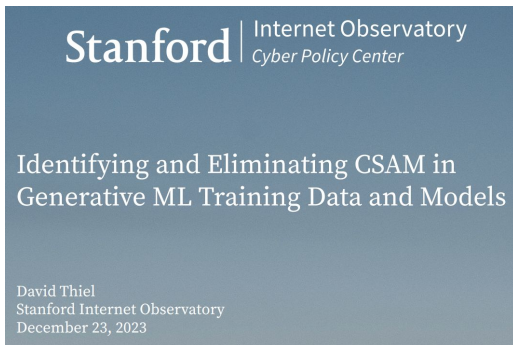
Also train on region captions  
with bounding box coordinates



Zeng et al, 2021: “Multi-Grained Vision Language Pre-Training: Aligning Texts with Visual Concepts”

# Disadvantages of CLIP-style models

1. Rely too heavily on batch size to learn concepts
2. Image-level captions are insufficient supervision
3. You can't have seen everything in your 5B dataset



It's extremely important to be intentional about data collection and filtering

# Disadvantages of CLIP-style models

1. Rely too heavily on batch size to learn concepts
2. Image-level captions are insufficient supervision
3. You can't have seen everything in your 5B dataset
4. Can only output a similarity score!

CoCa: Gets around (1) and ~ (4) by having both a contrastive loss and a captioning loss

Yu et al, 2022: "CoCa: Contrastive Captioners are Image-Text Foundation Models"

# Foundation Models

## Language

ELMo  
BERT  
GPT  
T5

## Classification

CLIP  
CoCa

## LM + Vision

LLaVA  
Flamingo  
GPT-4V  
Gemini  
Molmo

## And More!

Segment Anything  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

Visual Programming  
LMs + CLIP

# LLaVA

Motivation: CLIP is extremely general in its learned representation, but limited in its out-of-the box applications.

(can only output similarity scores between image and text)

... plus, the problems we discussed earlier (fine-grained visual understanding, grounding)

# LLaVA

Motivation: Language models which do next token prediction can be applied to a wide variety of tasks at inference (math, sentiment analysis, symbolic reasoning)

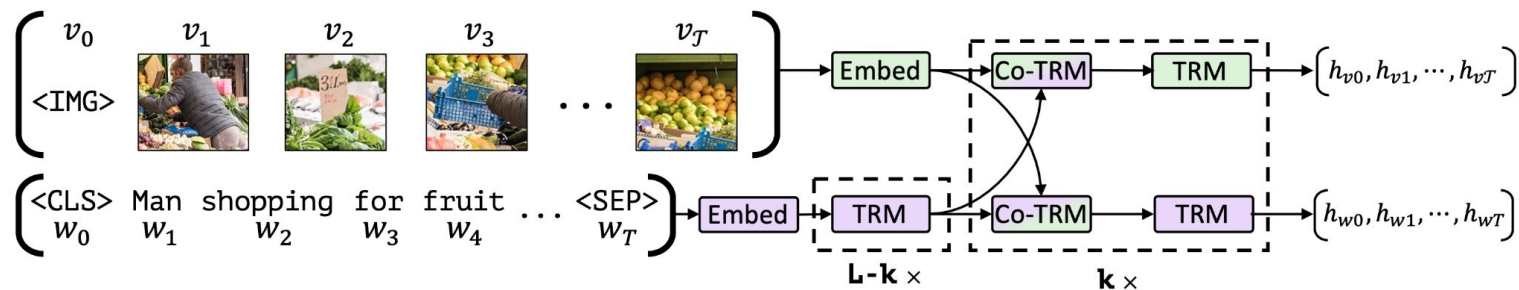
**Can we build a model that can accept images and text as input, and then output text?**

→ **Vision-Language Models**

# First, some historical context

Vision-Language Models didn't start with LLaVA!

They go as far back as 2019 → ViLBERT



# Historical context

Vision-Language Models didn't start with LLaVA!

They go as far back as 2019 → ViLBERT

**BUT**, they had to finetune for each task separately, with non-trivial task-specific methods (e.g., Mask-RCNN bounding box re-ranking for RefCOCO)

→ Same paradigm as we discussed right at the beginning of this lecture:  
very task-specific

# Historical context

A lot of this was BERT-based (ViLBERT, VisualBERT...)

~2020, auto-regressive language models became popular in NLP, so the vision community tried to take advantage of them

→ Format a wide variety of tasks as text output and treat them the same, i.e., train and test on all of them

Now comes LLaVA, in 2023: one of the first Multimodal (generative) Foundation models.

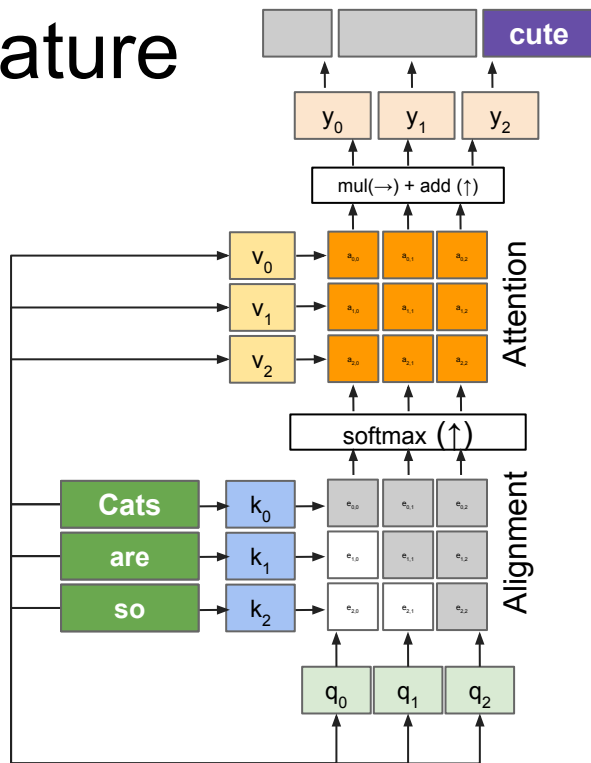
# LLaVA

Idea: You have your LLM, which has a lot of world knowledge and can do all sorts of tasks. Why don't we just convert images to "language"?

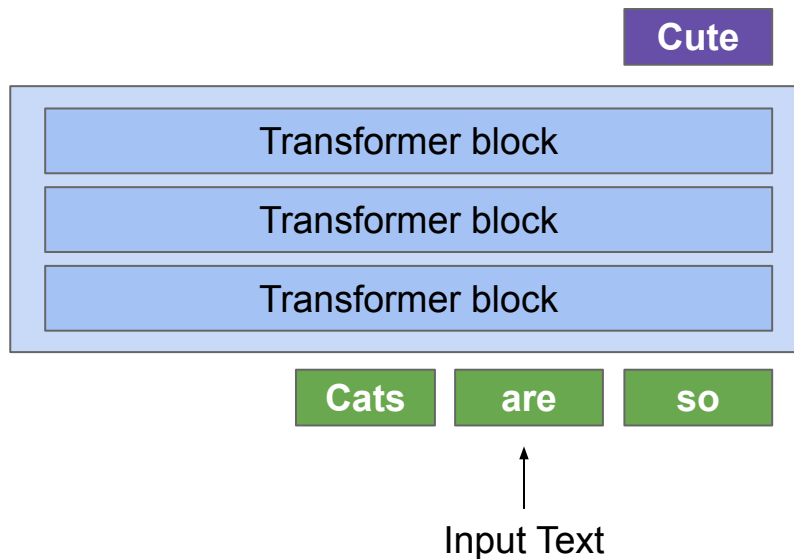
1. Convert your image representation into the same space as a pre-trained LLM's text representations
2. Stick them into the LLM
3. Train further on vision-language tasks, like captioning, VQA, etc

That's pretty much it!

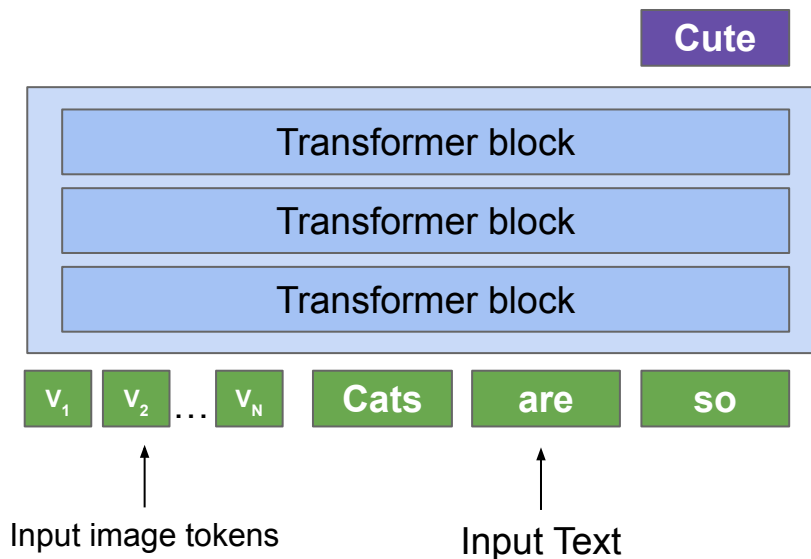
LLaVA uses the autoregressive nature of LLMs



# Recall how transformers decode language

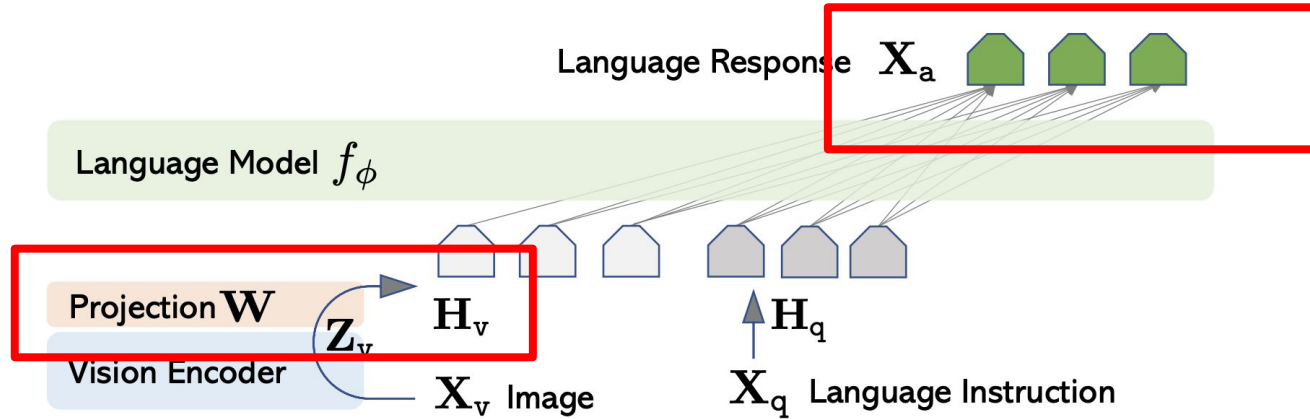


# Key idea behind LLaVA – add visual information to the LLM



Which image tokens work best here?

# LLaVA: Architecture



Convert your image representation into the same space as a pre-trained LLM's text representations

Train additionally on vision-language tasks, like captioning, VQA, etc

# LLaVA: Architecture

Image encoder: CLIP ViT-L/14

Text encoder: LLaMA

Projection: Linear projection

# LLaVA

LLaVA wasn't the first to think of this type of *architecture*.

VL-T5 (Cho et al, 2021)

GPV-2 (Kamath et al, 2022)

LLaVA's main contribution is actually the *data*.

“First attempt to use language-only GPT-4 to generate multimodal language-image instruction-following data.”

- GPT-4 came out in March 2023
- LLaVA came out in April 2023

# LLaVA: Training

Phase 1: Pre-training for Feature Alignment

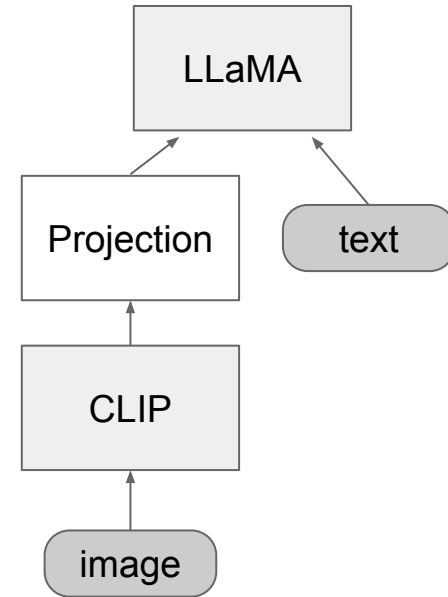
Phase 2: Fine-tuning End-to-End

# LLaVA: Training

## Phase 1: Pre-training for Feature Alignment

### Phase 2: Fine-tuning End-to-End

- You're starting out with a pre-trained LLaMA text encoder, a pre-trained CLIP image encoder, and a randomly initialized projection layer between them → the projection needs to learn how to project CLIP's image representations into LLaMA's text representation space.

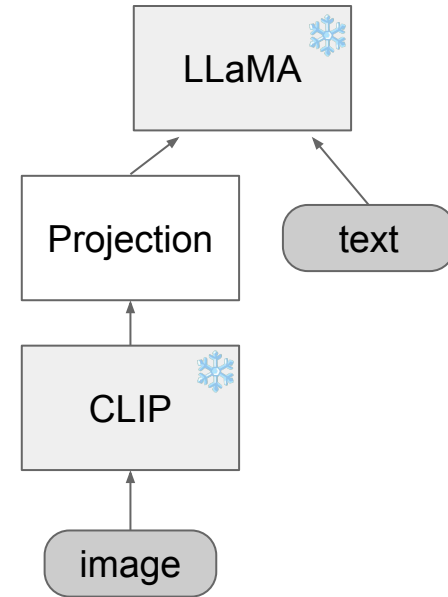


# LLaVA: Training

## Phase 1: Pre-training for Feature Alignment

## Phase 2: Fine-tuning End-to-End

- 595K image-text pairs from Conceptual Captions
- “Converted to instruction-following data” → add a text input prompt like “describe the image”
- CLIP and LLaMA are frozen, only the projection layer is trained



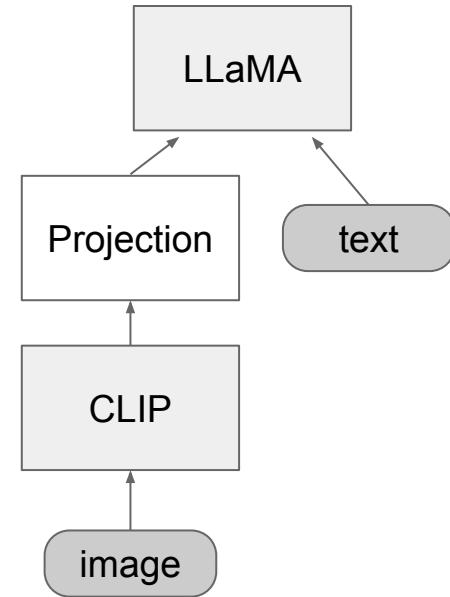
# LLaVA: Training

Phase 1: Pre-training for Feature Alignment

**Phase 2: Fine-tuning End-to-End**

Finetune everything on a wide variety of vision-language data:

- “Multimodal Chatbot” (GPT4-generated data)
- Science QA



# LLaVA: Training

## Multimodal Chatbot (GPT4-generated) data

### 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], backpack: [0.384, 0.696, 0.485, 0.914], suitcase: ...<omitted>

### Response type 1: conversation

Question: What type of vehicle is featured in the image?

Answer: The image features a black sport utility vehicle (SUV) ...<omitted>

### Response type 2: detailed description

The image is an underground parking area with a black sport utility vehicle (SUV) parked. There are three people in the scene, with one person standing closer to the left side of the vehicle, another person in the middle, and the third person on the right side. They are all working together to pack their luggage into the SUV for a trip. ...<omitted>

### Response type 3: complex reasoning

Question: What challenges do these people face?

Answer: In the image, a group of people is standing outside a black SUV in a parking area, surrounded by various pieces of luggage, including suitcases and backpacks. They are facing the challenge of fitting all their luggage into the black SUV. There are multiple suitcases and backpacks to be packed, which suggests that the group has a significant amount of belongings ...<omitted>

Rather than just training on academic benchmarks like ScienceQA, you want a wide variety of questions that people could *actually* ask about the image.

“Instruction tuning”

— Again, towards foundation models’ idea of generality

# LLaVA

---

## Visual input example, Extreme Ironing:

---



Source: <https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg>

User  
LLaVA

What is unusual about this image?

The unusual aspect of this image is a man ironing clothes on the back of a minivan or van. This is not a typical place to perform this activity, as one would usually iron clothes in a more stationary and safe location, such as a home, using a regular ironing board. The scene depicted in the image is peculiar as it involves a makeshift ironing setup on a vehicle, which can be both unsafe and unconventional. Additionally, it is not clear how the man is able to maintain balance and stability while ironing clothes in such an unstable environment.

[Start a new conversation, and clear the history]

User  
LLaVA

What's happening in the scene?

The image depicts a man standing on top of a yellow SUV in a busy city street. He is holding a portable ladder, seemingly ironing clothes while standing on the vehicle. Around the scene, there are other cars, a traffic light, and a fire hydrant. Additionally, there are two other people in the vicinity, one close to the man on the car and another further away. The man on the ladder appears to be attracting attention **due to his unconventional choice of ironing his clothes on top of a moving car**. The city street around him is bustling with activity, adding to the unique nature of the scene.

User  
GPT-4 [36]

What is unusual about this image?

The unusual thing about this image is that a man is ironing clothes on an ironing board attached to the roof of a moving taxi.

User  
BLIP-2

What is unusual about this image?

a man is sitting on the back of a yellow cab

User  
OpenFlamingo

What is unusual about this image?

The man is drying his clothes on the hood of his car.

# LLaVA

Visual input example, Extreme Ironing:



Flexible!

	Source: <a href="https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg">https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg</a>
User	What is unusual about this image?
LLaVA	The unusual aspect of this image is a man ironing clothes on the back of a minivan or van. This is not a typical place to perform this activity, as one would usually iron clothes in a more stationary and safe location, such as a home, using a regular ironing board. The scene depicted in the image is peculiar as it involves a makeshift ironing setup on a vehicle, which can be both unsafe and unconventional. Additionally, it is not clear how the man is able to maintain balance and stability while ironing clothes in such an unstable environment. [Start a new conversation, and clear the history]
User	What's happening in the scene?
LLaVA	The image depicts a man standing on top of a yellow SUV in a busy city street. He is holding a portable ladder, seemingly ironing clothes while standing on the vehicle. Around the scene, there are other cars, a traffic light, and a fire hydrant. Additionally, there are two other people in the vicinity, one close to the man on the car and another further away. The man on the ladder appears to be attracting attention <b>due to his unconventional choice of ironing his clothes on top of a moving car</b> . The city street around him is bustling with activity, adding to the unique nature of the scene.
User	What is unusual about this image?
GPT-4 [36]	The unusual thing about this image is that a man is ironing clothes on an ironing board attached to the roof of a moving taxi.
User	What is unusual about this image?
BLIP-2	a man is sitting on the back of a yellow cab
User	What is unusual about this image?
OpenFlamingo	The man is drying his clothes on the hood of his car.


# LLaVA: Training

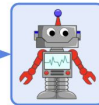
## ScienceQA

LLaVA input


**Question:** Which type of force from the baby's hand opens the cabinet door?

**Options:** (A) pull (B) push

**Context:** A baby wants to know what is inside of a cabinet. Her hand applies a force to the door, and the door opens. 




LLaVA output

 **Answer:** The answer is A.

**BECAUSE:**

**Lecture:** A force is a **push** or a **pull** that one object applies to a second object. The direction of a push is **away from** the object that is pushing. The direction of a **pull** is **toward** the object that is pulling.

 **Explanation:** The **baby's hand** applies a force to the **cabinet door**. This force causes the **door** to **open**. The direction of this force is **toward** the **baby's hand**. This force is a **pull**.

# LLaVA: Evaluation

## ScienceQA

Method	Subject			Context Modality			Grade		Average
	NAT	SOC	LAN	TXT	IMG	NO	G1-6	G7-12	
<i>Representative &amp; SoTA methods with numbers reported in the literature</i>									
Human [34]	90.23	84.97	87.48	89.60	87.50	88.10	91.59	82.42	88.40
GPT-3.5 [34]	74.64	69.74	76.00	74.44	67.28	77.42	76.80	68.89	73.97
GPT-3.5 w/ CoT [34]	75.44	70.87	78.09	74.68	67.43	79.93	78.23	69.68	75.17
LLaMA-Adapter [59]	84.37	88.30	84.36	83.72	80.32	86.90	85.83	84.05	85.19
MM-CoT <sub>Base</sub> [61]	87.52	77.17	85.82	87.88	82.90	86.83	84.65	85.37	84.91
MM-CoT <sub>Large</sub> [61]	95.91	82.00	90.82	95.26	88.80	92.89	92.44	90.31	91.68
<i>Results with our own experiment runs</i>									
GPT-4 <sup>†</sup>	84.06	73.45	87.36	81.87	70.75	90.73	84.69	79.10	82.69
LLaVA	90.36	95.95	88.00	89.49	88.00	90.66	90.93	90.90	90.92

# LLaVA: Evaluation

You have the first general-purpose vision-language model, now how do you evaluate it?

Remember, most evaluations at the time wouldn't have shown off the general-purpose-ness very well.

# LLaVA: Evaluation

## LLaVA-Bench (In-the-Wild)

Very small (24 images,  
60 questions)

---

### Challenging examples from LLaVA-Bench (In-the-Wild):

---



ICHIRAN Ramen [source]



Filled fridge [source]

---

Annotation

A close-up photo of a meal at **ICHI-RAN**. The chashu ramen bowl with a spoon is placed in the center. The ramen is seasoned with **chili sauce**, **chopped scallions**, and served with **two pieces of chashu**. Chopsticks are placed to the right of the bowl, still in their paper wrap, not yet opened. The ramen is also served with nori on the left. On top, from left to right, the following sides are served: a bowl of **orange spice** (possibly garlic sauce), a plate of **smoke-flavored stewed pork with chopped scallions**, and a cup of **matcha green tea**.

An open refrigerator filled with a variety of food items. In the left part of the compartment, towards the front, there is a **plastic box of strawberries** with a small bag of baby carrots on top. Towards the back, there is a stack of sauce containers. In the middle part of the compartment, towards the front, there is a green plastic box, and there is an unidentified plastic bag placed on it. Towards the back, there is a carton of milk. In the right part of the compartment, towards the front, there is a box of blueberries with three yogurts stacked on top. The large bottle of yogurt is **Fage non-fat yogurt**, and **one of the smaller cups is Fage blueberry yogurt**. The brand and flavor of the other smaller cup are unknown. Towards the back, there is a container with an unknown content.

---

Question 1 What's the name of the restaurant?

What is the brand of the blueberry-flavored yogurt?

Question 2 Describe this photo in detail.

Is there strawberry-flavored yogurt in the fridge?

---

# LLaVA: Evaluation

## LLaVA-Bench (In-the-Wild)

	Conversation	Detail description	Complex reasoning	All
OpenFlamingo [5]	$19.3 \pm 0.5$	$19.0 \pm 0.5$	$19.1 \pm 0.7$	$19.1 \pm 0.4$
BLIP-2 [28]	$54.6 \pm 1.4$	$29.1 \pm 1.2$	$32.9 \pm 0.7$	$38.1 \pm 1.0$
LLaVA	$57.3 \pm 1.9$	$52.5 \pm 6.3$	$81.7 \pm 1.8$	$67.3 \pm 2.0$

# A side note about evaluations

It's important to think outside the box when creating evaluations — this helps drive research progress in interesting directions!

Since then, we have more general-purpose and useful benchmarks, capturing how users really use models (e.g., “arenas”)

# LLaVA: Summary

- Use a vision encoder + an LLM
- Synthetically generate training data
- Instruction tuning → chatbot
- Finetune on a wide variety of downstream tasks

→ broad applicability

# Foundation Models

## Language

ELMo  
BERT  
GPT  
T5

## Classification

CLIP  
CoCa

## LM + Vision

LLaVA  
Flamingo  
GPT-4V  
Gemini  
Molmo

## And More!

Segment Anything  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

Visual Programming  
LMs + CLIP

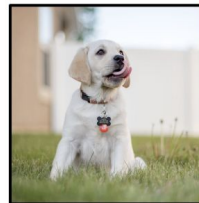
# Flamingo

All of these models were restricted to 1 image + 1 text segment in the input.

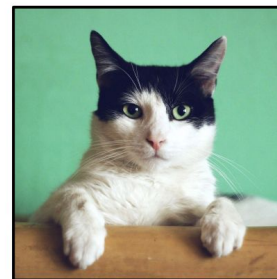
→ How do we go beyond this?

Flamingo: Train on interleaved image-text data on the internet.

Cute pics of my pets!



My puppy sitting in the grass.



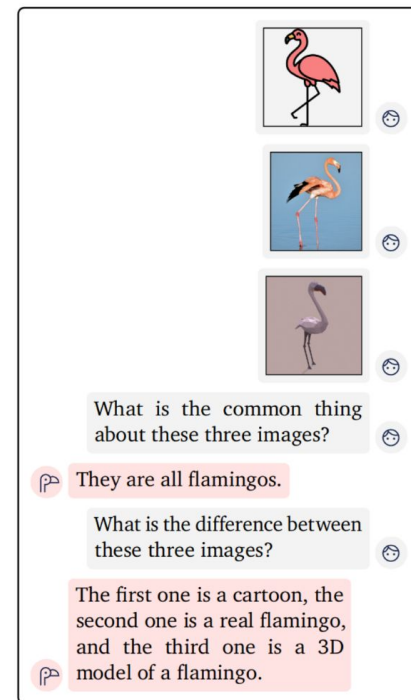
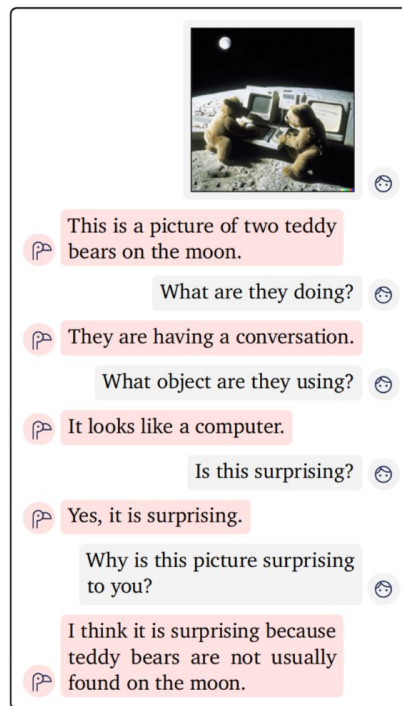
My cat looking very dignified.

Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

# Flamingo: Interleaved image and text

What behaviors does this allow?

1. Longer “conversations”
2. Multiple images

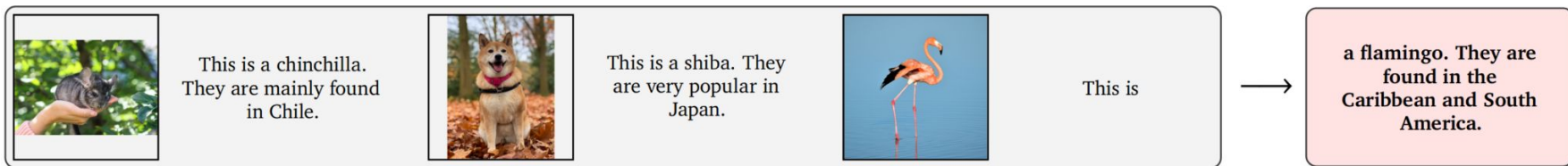


Alayrac et al “Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

# Flamingo: Interleaved image and text

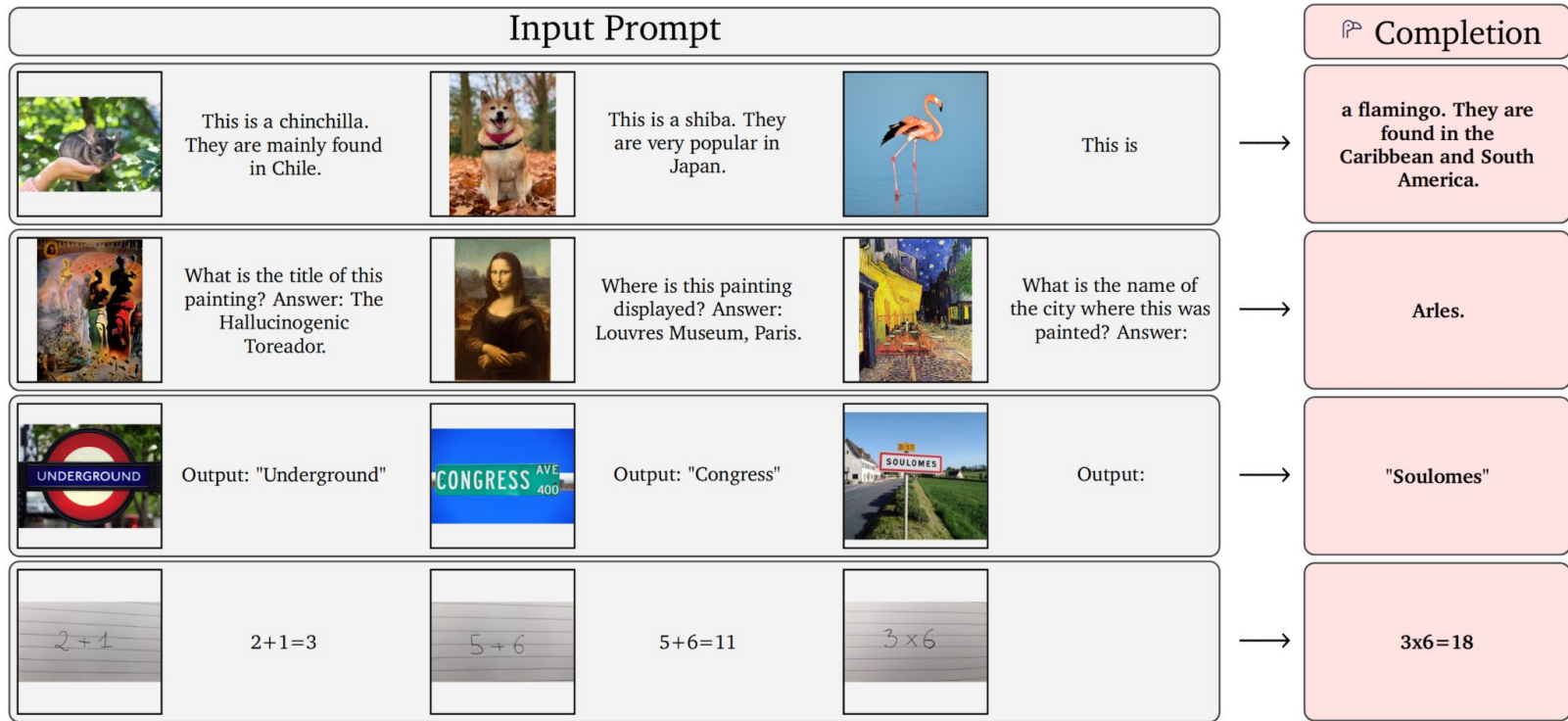
What behaviors does this allow?

1. Longer “conversations”
2. Multiple images
3. Few-shot examples in the input



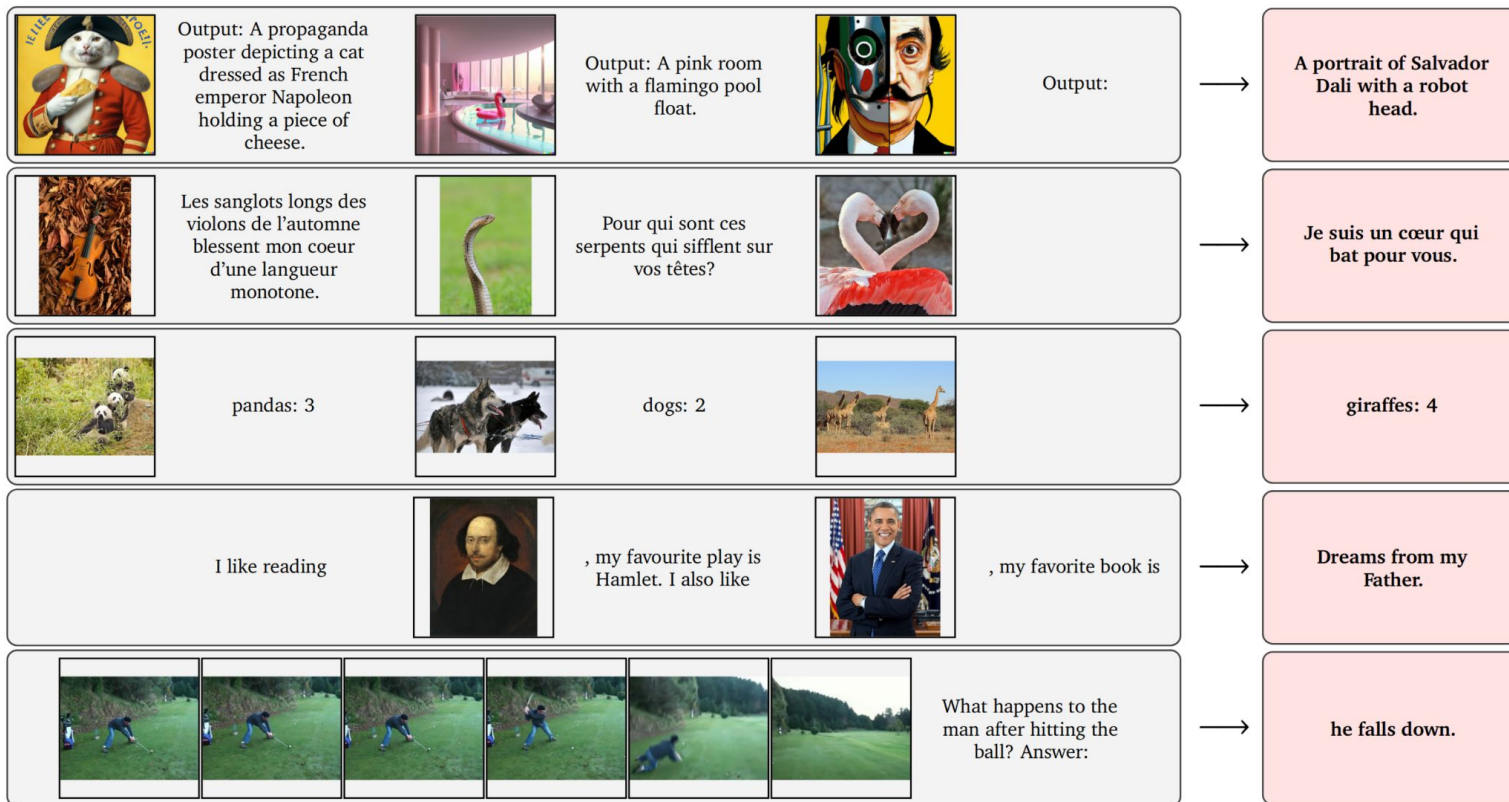
Alayrac et al “Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

# Flamingo enables **in-context learning**



Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

# Flamingo enables in-context learning



Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022."

# Foundation Models

## Language

ELMo  
BERT  
GPT  
T5

## Classification

CLIP  
CoCa

## LM + Vision

LLaVA  
Flamingo  
GPT-4V  
Gemini  
Molmo

## And More!

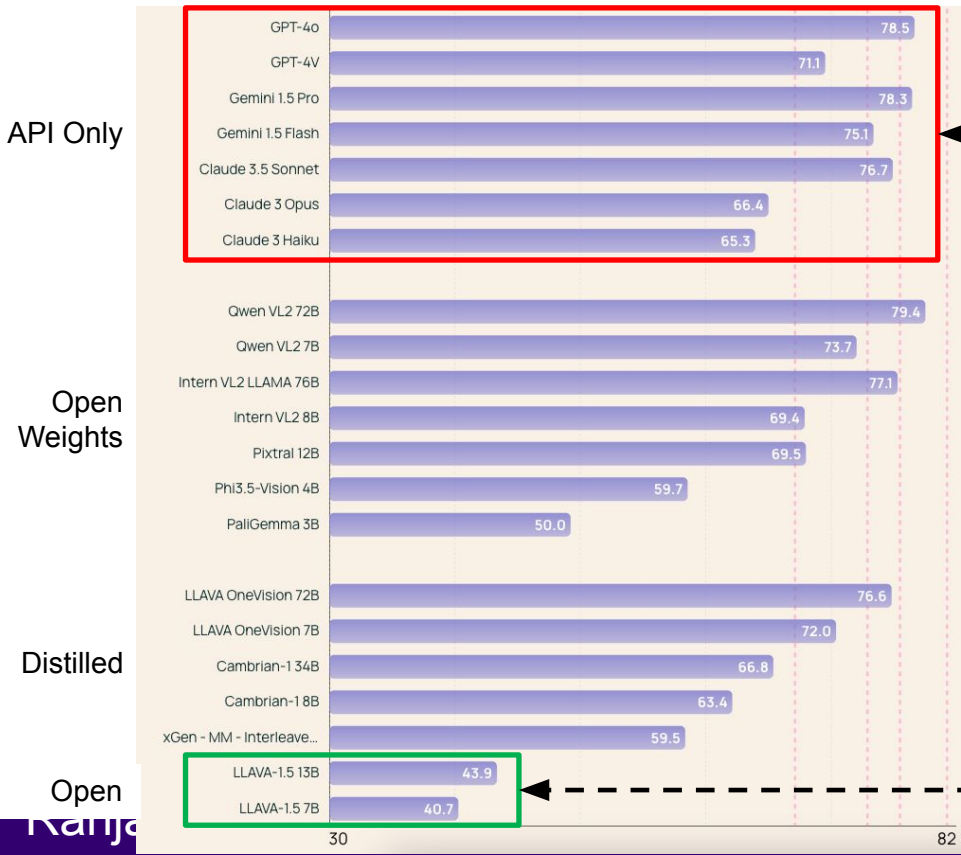
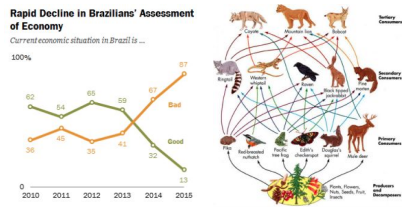
Segment Anything  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

Visual Programming  
LMs + CLIP



# There are open-weight models but they are all distilled from API-only



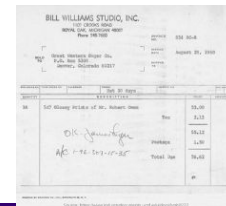
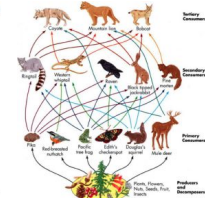
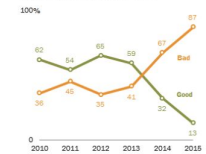
How do we close the gap without relying on proprietary models?

# There are open-weight models but they are all distilled from API-only



**Rapid Decline in Brazilians' Assessment of Economy**

Current economic situation in Brazil is ...

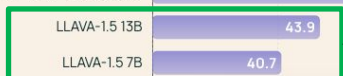
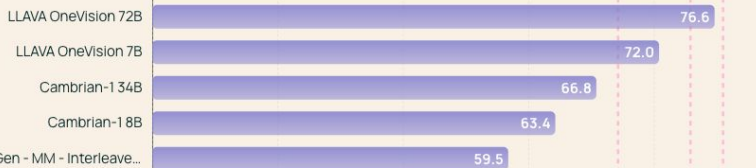
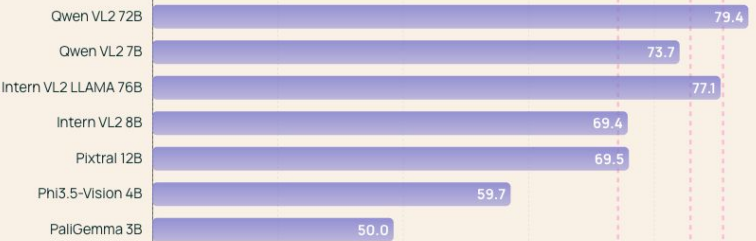
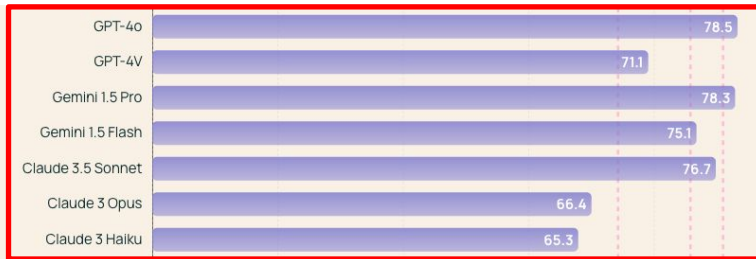


API Only

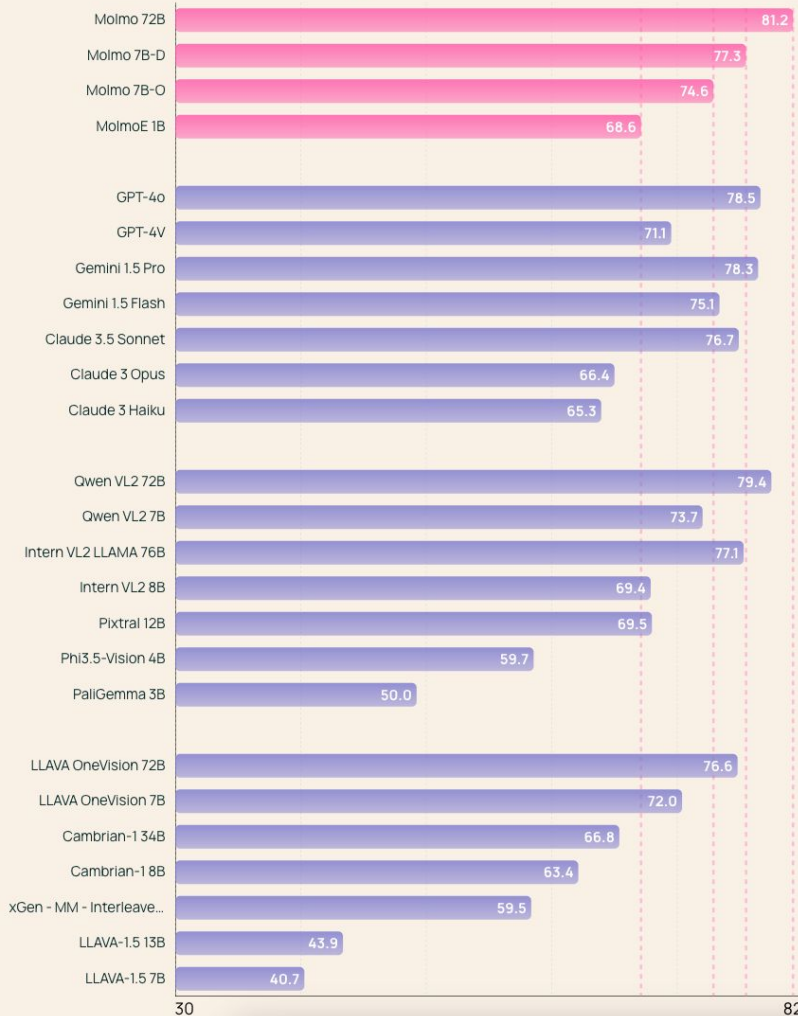
Open Weights

Distilled

Open



Average Score on 11 Academic Benchmarks



Open  
Weights  
Data  
Code  
Evals

API Only

Open  
Weights

Distilled

Open  
Weights



Completely Open

Open Weights

Open Data

Open Code

Open Evals

Deitke et al., 2024: "Molmo and PixMo: Open Weights and Open Data for State-of-the-Art Vision-Language Models"

Average Score on 11 Academic Benchmarks

Human Preference Elo Rating

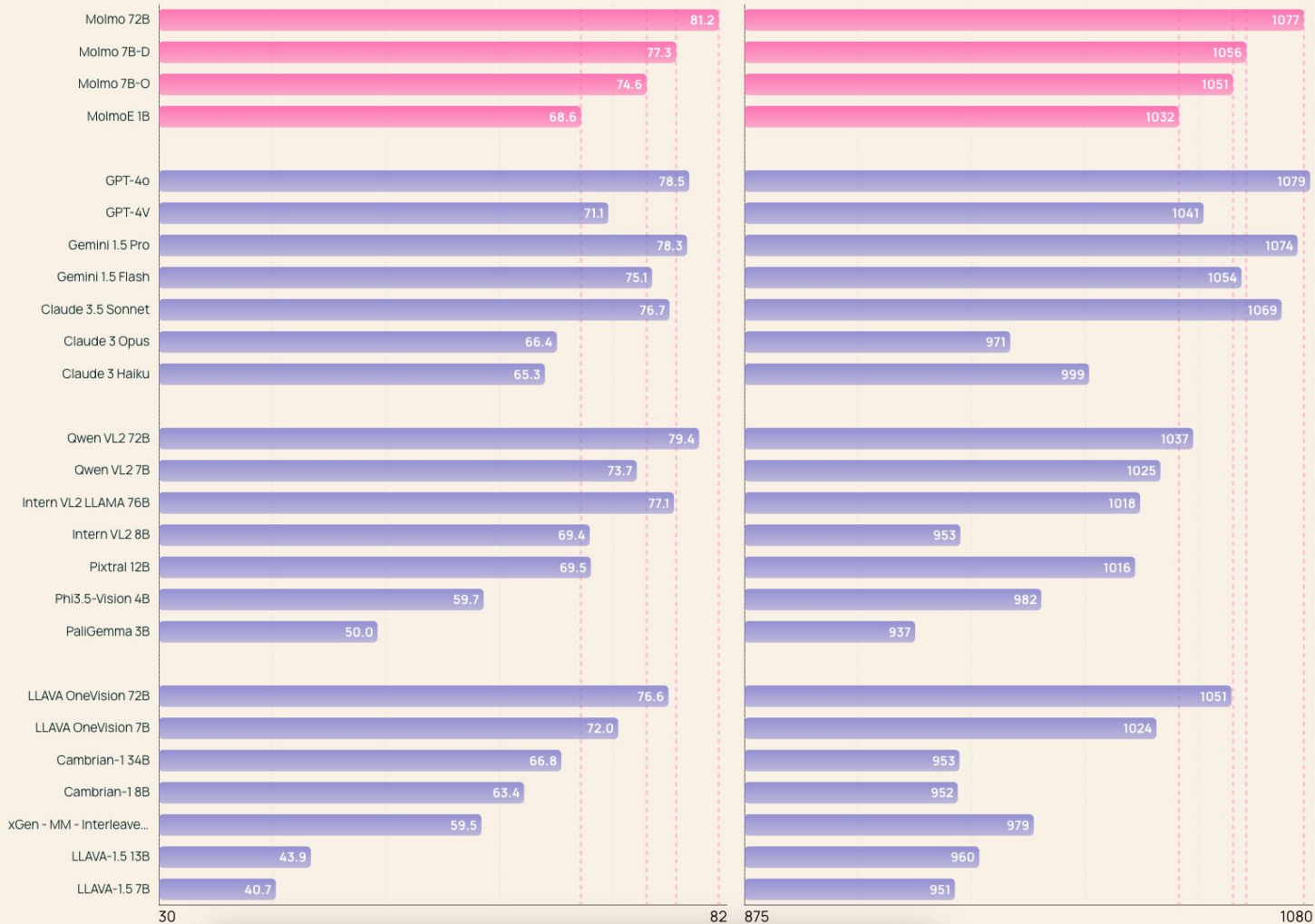
Open  
Weights  
Data  
Code  
Evals

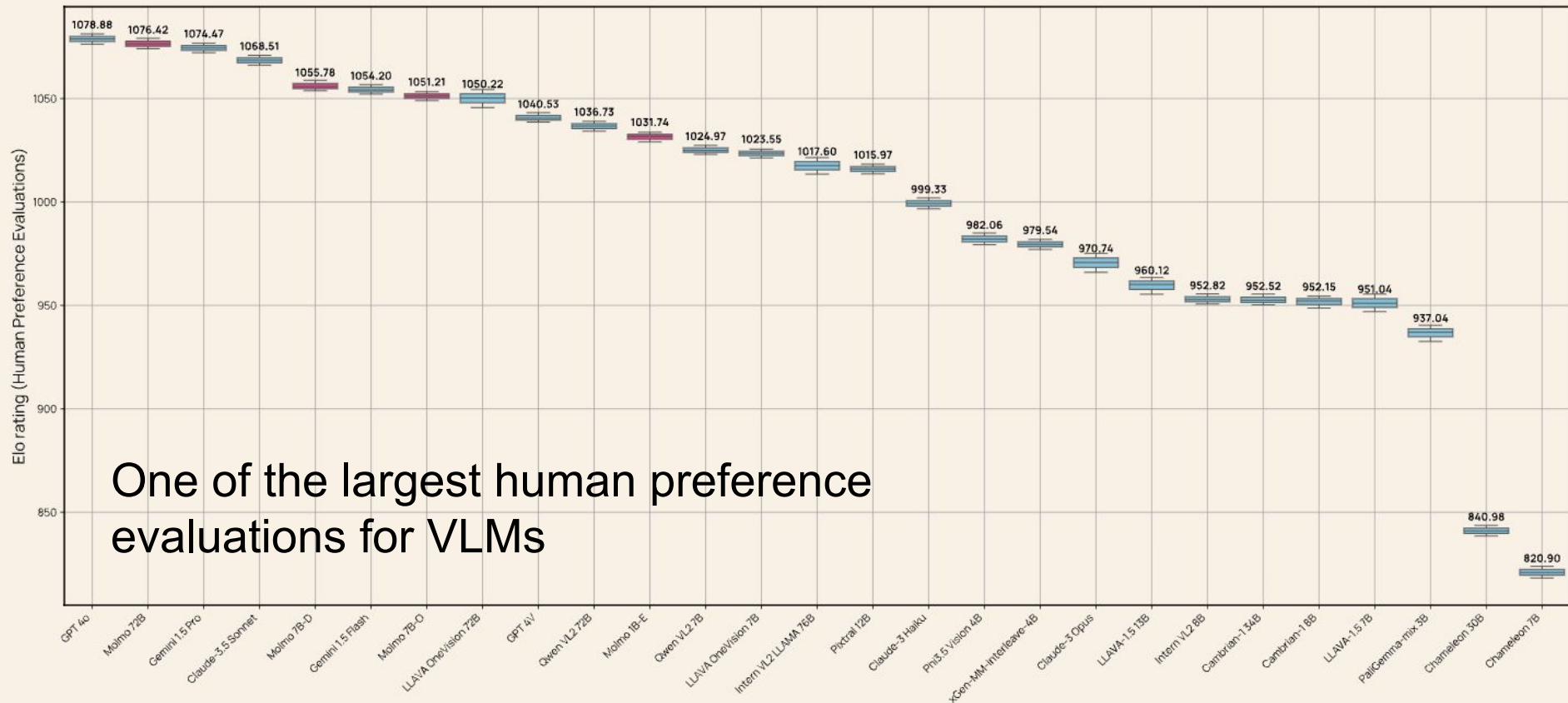
API Only

Open  
Weights

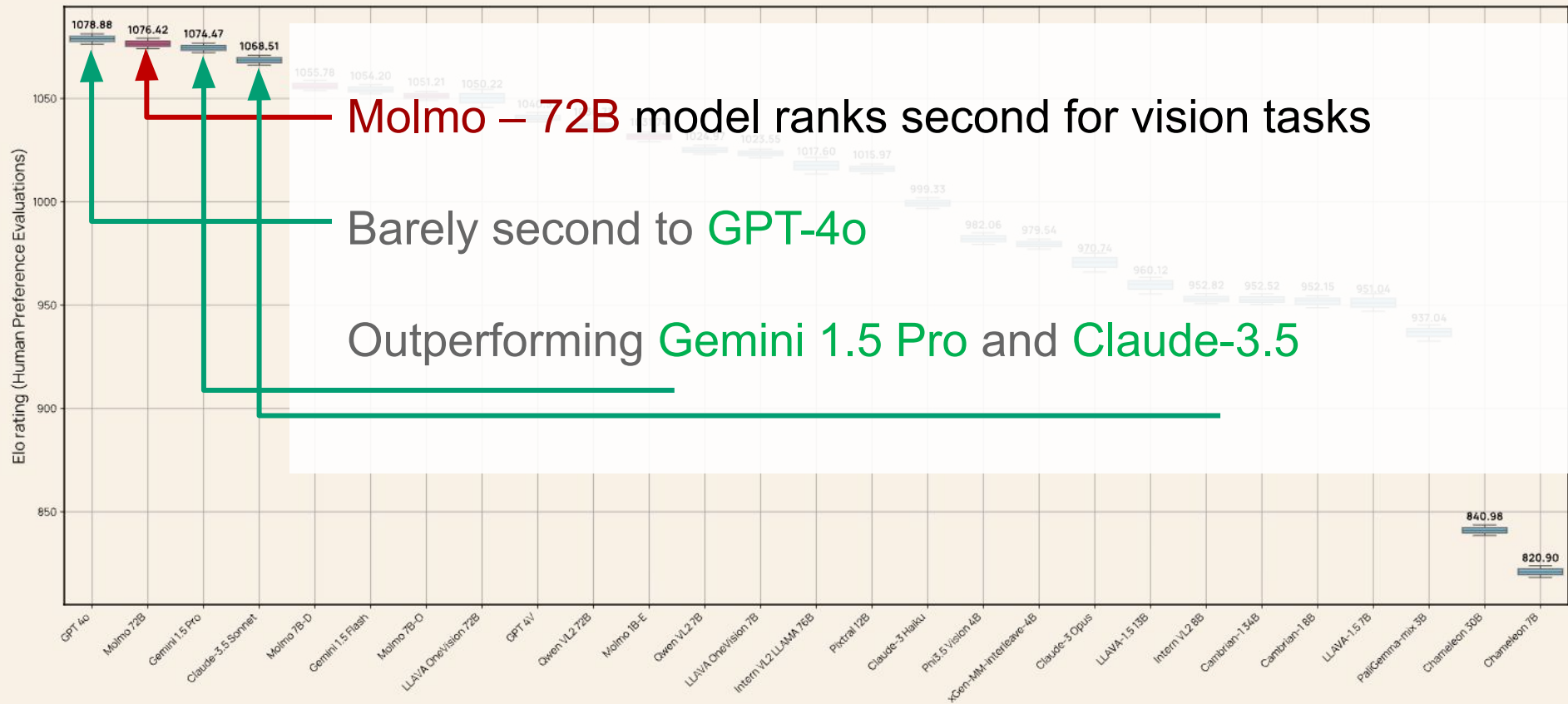
Distilled

Open  
Weights

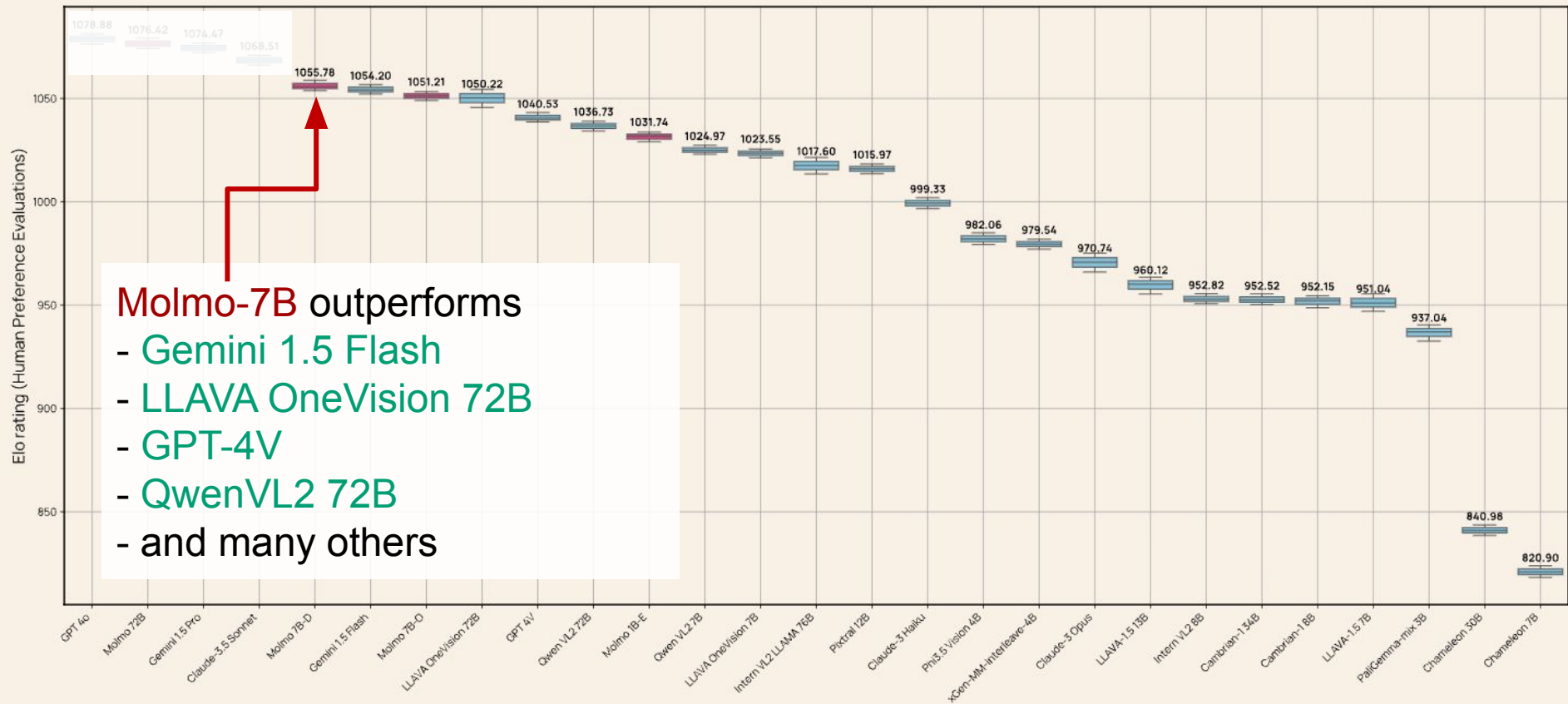




with **325k** pairwise comparisons  
and **870** human annotators



with **325k** pairwise comparisons  
and **870** human annotators



with **325k** pairwise comparisons  
and **870** human annotators

# Reaction online – released September 2024

WILL KNIGHT BUSINESS SEP 25, 2024 9:00 AM

## The Most Capable Open Source AI Model Yet Could Supercharge AI Agents

A compact and fully open source visual AI model will make it easier for AI to take control of your computer— hopefully in a good way.

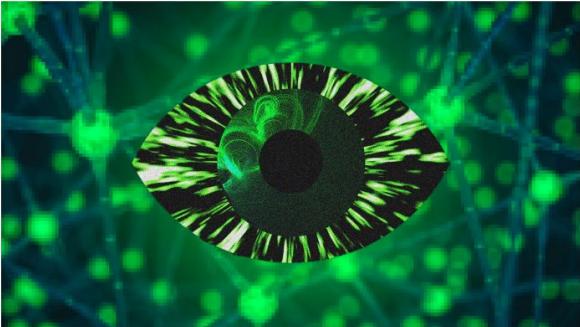


PHOTO-ILLUSTRATION: WIRED STAFF; GETTY



Login

Search Q

Startups

Venture

Apple

Security

AI

Apps

Events

Startup Battlefield

More

AI

## Ai2's Molmo shows open source can meet, and beat, closed multimodal models

Devin Coldewey / 6:00 AM PDT • September 25, 2024

Comment

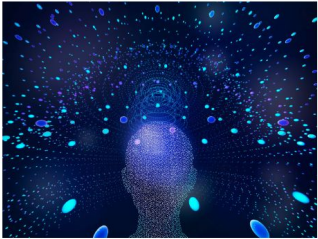


Image Credits: Hiroshi Watanabe / Getty Images

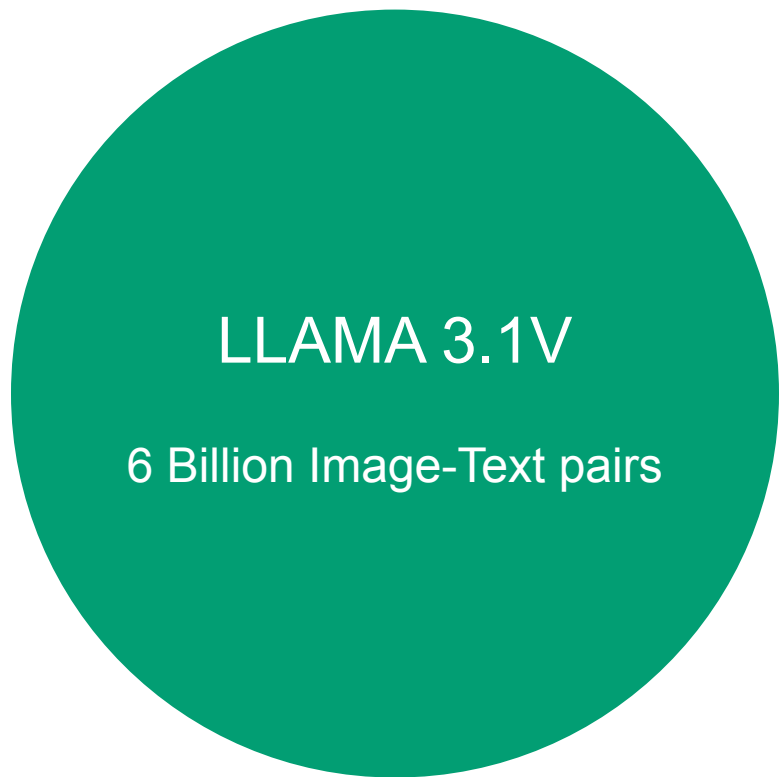
The common wisdom is that companies like Google, OpenAI, and Anthropic, with bottomless cash reserves and hundreds of top-tier researchers, are the only ones that can make a state-of-the-art foundation model. But as one among them famously noted, they “have no moat” — and Ai2 showed that today with the release of Molmo, a multimodal AI model that matches their best while also being small, free, and truly open source.



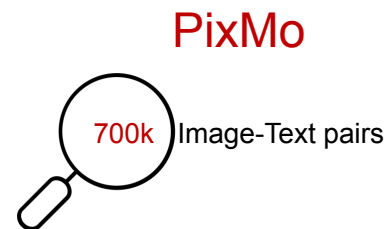
# Why did it do so well?

1. High-quality data
2. Pointing capability

# Data matters! Quality over quantity, even for pretraining



Molmo is trained with



# PixMo data is intentional:



This photograph captures a well-organized work desk set prominently in the middle of the frame. The desk is **large and rectangular**, made from a **polished, rich wood** that **spans horizontally across the image**. Its structure is supported by four distinctive A-shaped legs, adding an elegant touch. On the desk, a striking dual-monitor setup is noticeable: a tall, vertical screen placed behind and **to the right of** a wider, horizontal computer monitor.

To the right of these monitors, a black mouse rests on a mouse pad. **Scattered around** the mouse pad, some white papers or letters are strewn across the far right side. **On the left side** of the desk, a black desk lamp with an extended arm hangs down, illuminating the workspace. Nearby, a stack of books is neatly placed in the upper left corner of the table.

The **background** wall is painted a subtle beige-white, complementing the refined ambiance of the space. The floor below the desk features elegant pinkish marble tiles, enhancing the room's sophisticated look. To the far right of the image, a large window or patio door allows **natural light** to pour in, with clear glass that offers a glimpse into the outside area. This exterior view includes part of a rustic brick wall and a metal pail, hinting at an adjacent patio.

Annotators were asked questions designed to extract meaningful visual information:

- What is the image at first glance?
- What are the objects and their counts?
- What does the text say?
- What are the positions of the objects?
- What subtle details are noticeable?
- What is in the background?
- What is the style and color?

People don't like to **type**  
... but they love to **talk**

Annotators were asked to speak for 60 to 90 seconds  
about an image

Automatically convert speech into text for pretraining

# Pointing capability

Count the boats



● boats

Counting the **boats** shows a total of 35.

point\_qa: What is the route of this bus?



● Route number and destination displayed on bus

The image shows a white bus with a green stripe along its side. **The bus is heading to Berkeley BART**, as indicated by the digital display on its front. Berkeley BART likely refers to the Berkeley station of the Bay Area Rapid Transit system, which is the main light rail and bus network serving the San Francisco Bay Area.



## Camera Feed



P|

Send

Points are shown in the image.

# Foundation Models

## Language

ELMo  
BERT  
GPT  
T5

## Classification

CLIP  
CoCa

## LM + Vision

LLaVA  
Flamingo  
GPT-4V  
Gemini

## And More!

Segment Anything  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

**Visual Programming**  
LMs + CLIP

# VisProg (visual programming)

Many Visual Question Answering models have been trained to do this type of task



Are there 3 people in the boat?

Menon et al “ViperGPT: Visual Inference via Python Execution for Reasoning”. 2023.  
Gupta et al “Visual Programming: Compositional visual reasoning without training”. 2023.

# VisProg (visual programming)

LEFT:



RIGHT:

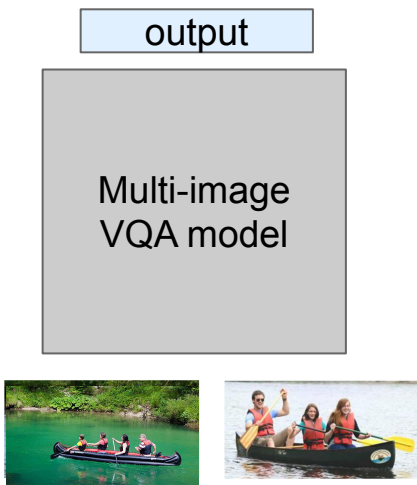


**Statement:** The left and right image contains a total of six people and two boats.

Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# VisProg (visual programming)

Train a new model for your task



Write a python script with the models you have

```
Class MyMultiImageVQA():  
  
    Def ProcessImgs():  
        Ans1 = VQA(Image1)  
        Ans2 = VQA(Image2)  
        Return Ans1 + Ans2
```

**General to 2 images now, but not beyond that**

Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# VisProg (visual programming)

**LEFT:**  **RIGHT:** 

**Statement:** The left and right image contains a total of six people and two boats.

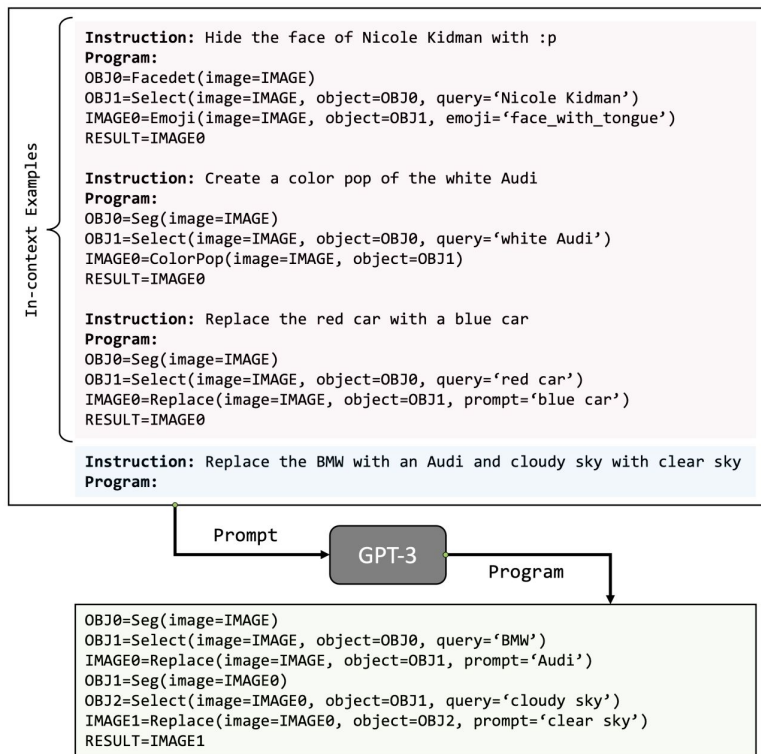
GPT

```
Class MyMultiImageVQA():  
  
  Def ProcessIms():  
    Ans1 = VQA(Image1)  
    Ans2 = VQA(Image2)  
    Return Ans1 + Ans2
```

False

Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# VisProg (visual programming)



Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# VisProg (visual programming)

Image Understanding	<b>Loc</b> OWL-ViT	<b>FaceDet</b> DSFD (pypi)	<b>Seg</b> MaskFormer	<b>Select</b> CLIP-ViT	<b>Classify</b> CLIP-ViT	<b>Vqa</b> ViLT
Image Manipulation	<b>Replace</b> Stable Diffusion	<b>ColorPop</b> PIL.convert() cv2.grabCut()	<b>BgBlur</b> PIL.GaussianBlur() cv2.grabCut()	<b>Tag</b> PIL.rectangle() PIL.text()	<b>Emoji</b> AugLy (pypi)	
	<b>Crop</b> PIL.crop()	<b>CropLeft</b> PIL.crop()	<b>CropRight</b> PIL.crop()	<b>CropAbove</b> PIL.crop()	<b>CropBelow</b> PIL.crop()	
Knowledge Retrieval	<b>List</b> GPT3	Arithmetic & Logical	<b>Eval</b> eval()	<b>Count</b> len()	<b>Result</b> dict()	

“Tools”

Gupta et al “Visual Programming: Compositional visual reasoning without training”. 2023.

# VisProg (visual programming)

## Natural Language Visual Reasoning

LEFT:



RIGHT:



**Statement:** The left and right image contains a total of six people and two boats.

**Program:**

```
ANSWER0=Vqa(image=LEFT, question='How many people are in the image?')
ANSWER1=Vqa(image=RIGHT, question='How many people are in the image?')
ANSWER2=Vqa(image=LEFT, question='How many boats are in the image?')
ANSWER3=Vqa(image=RIGHT, question='How many boats are in the image?')
ANSWER4=Eval('{ANSWER0} + {ANSWER1} == 6 and {ANSWER2} + {ANSWER3} == 2')
RESULT=ANSWER4
```

**Prediction:** False

Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# VisProg (visual programming)

## Factual Knowledge Object Tagging

**IMAGE:**



**Prediction: IMAGE0**



**Instruction:** Tag the 7 main characters on the TV show Big Bang Theory

**Program:**

```
OBJ0=FaceDet(image=IMAGE)
```

```
LIST0=List(query='main characters on the TV show Big Bang Theory', max=7)
```

```
OBJ1=Classify(image=IMAGE, object=OBJ0, categories=LIST0)
```

```
IMAGE0=Tag(image=IMAGE, object=OBJ1)
```

```
RESULT=IMAGE0
```

Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# VisProg (visual programming)

**IMAGE:**



**Prediction: IMAGE0**



**Instruction:** Replace desert with lush green grass

**Program:**

```
OBJ0=Seg(image=IMAGE)
```

```
OBJ1=Select(image=IMAGE, object=OBJ0, query='desert', category=None)
```

```
IMAGE0=Replace(image=IMAGE, object=OBJ1, prompt='lush green grass')
```

```
RESULT=IMAGE0
```

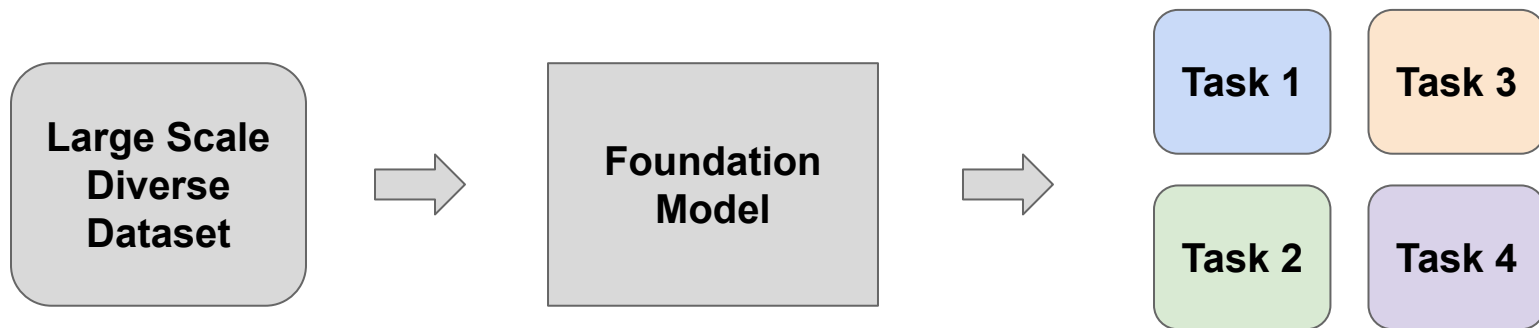
Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

# Cool open problems in Vision-Language

- What's missing from the training data? How do you fill the gaps?
- What can scale solve? What can scale *not* solve?
- Are image-level captions really enough?
- Is there a better way to encode images than a sequence of patches?
- (Mechanistic) interpretability for problems like hallucination
- Making models more efficient, e.g., via distillation
- What are our evaluations missing? Are we setting the wrong research targets?
- When do you use tools?

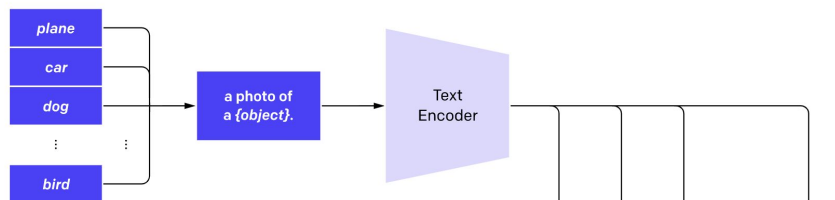
[your questions here!]

# Summary

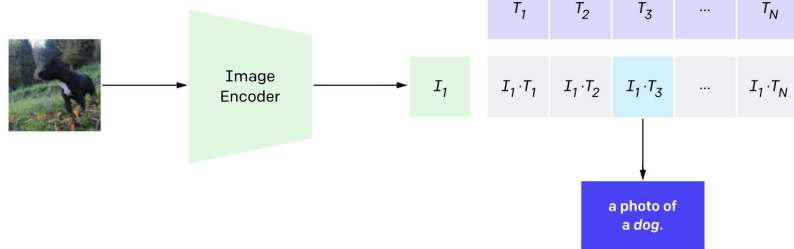


# Summary

## 2. Create dataset classifier from label text

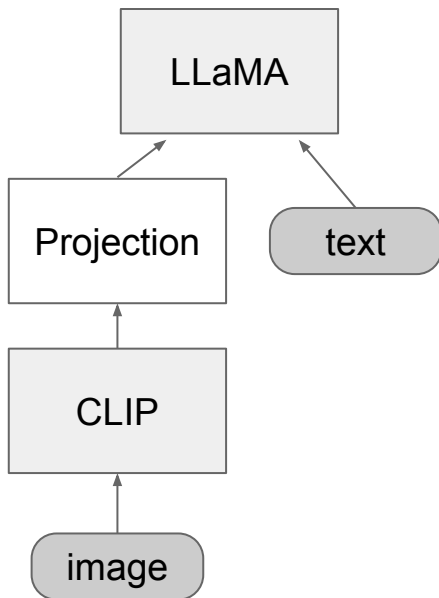


## 3. Use for zero-shot prediction



DATASET	IMAGENET RESNET101	CLIP VIT-L
ImageNet	76.2%	76.2%
ImageNet V2	64.3%	70.1%
ImageNet Rendition	37.7%	88.9%
ObjectNet	32.6%	72.3%
ImageNet Sketch	25.2%	60.2%
ImageNet Adversarial	2.7%	77.1%

# Summary



---

## Visual input example, Extreme Ironing:

---



Source: <https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg>

User  
LLaVA

What is unusual about this image?

The unusual aspect of this image is a man ironing clothes on the back of a minivan or van. This is not a typical place to perform this activity, as one would usually iron clothes in a more stationary and safe location, such as a home, using a regular ironing board. The scene depicted in the image is peculiar as it involves a makeshift ironing setup on a vehicle, which can be both unsafe and unconventional. Additionally, it is not clear how the man is able to maintain balance and stability while ironing clothes in such an unstable environment.

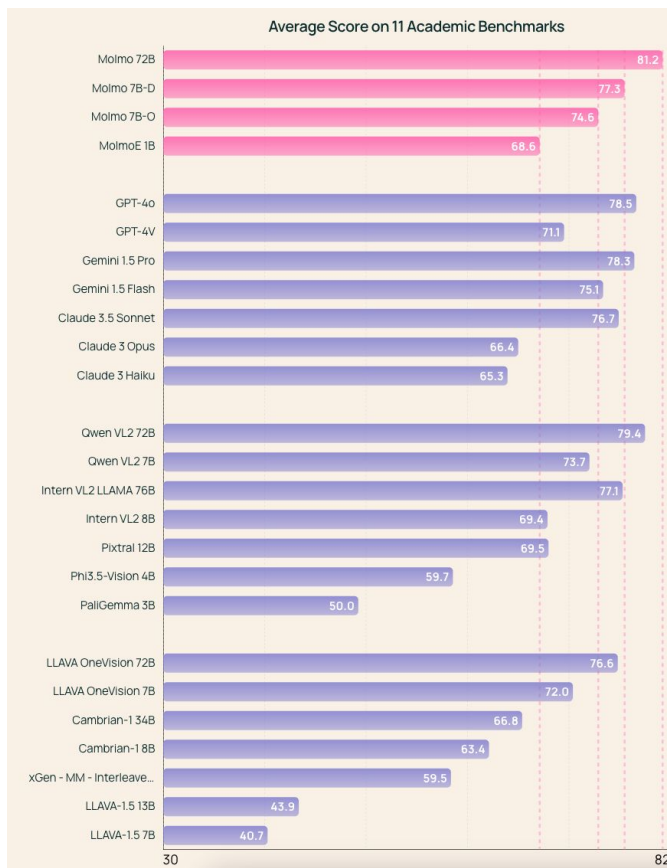
User  
LLaVA

[Start a new conversation, and clear the history]

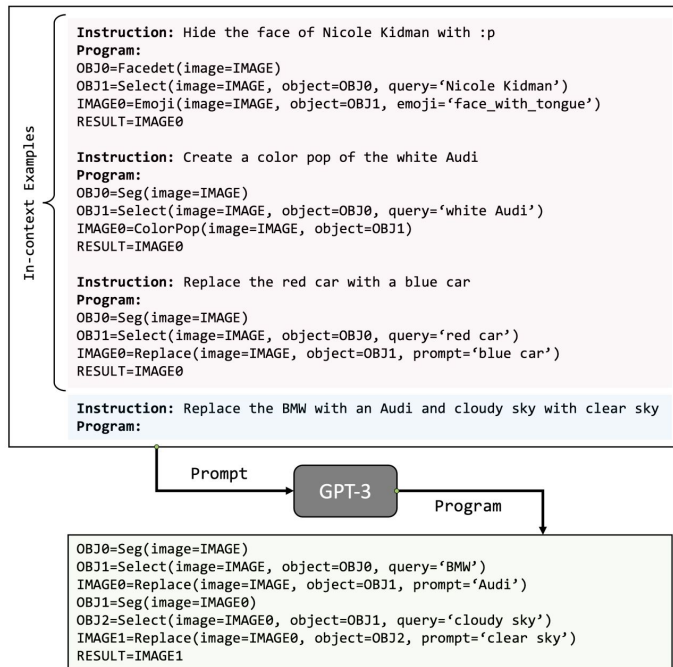
What's happening in the scene?

The image depicts a man standing on top of a yellow SUV in a busy city street. He is holding a portable ladder, seemingly ironing clothes while standing on the vehicle. Around the scene, there are other cars, a traffic light, and a fire hydrant. Additionally, there are two other people in the vicinity, one close to the man on the car and another further away. The man on the ladder appears to be attracting attention **due to his unconventional choice of ironing his clothes on top of a moving car**. The city street around him is bustling with activity, adding to the unique nature of the scene.

# Summary



# Summary



Next time: Generative models

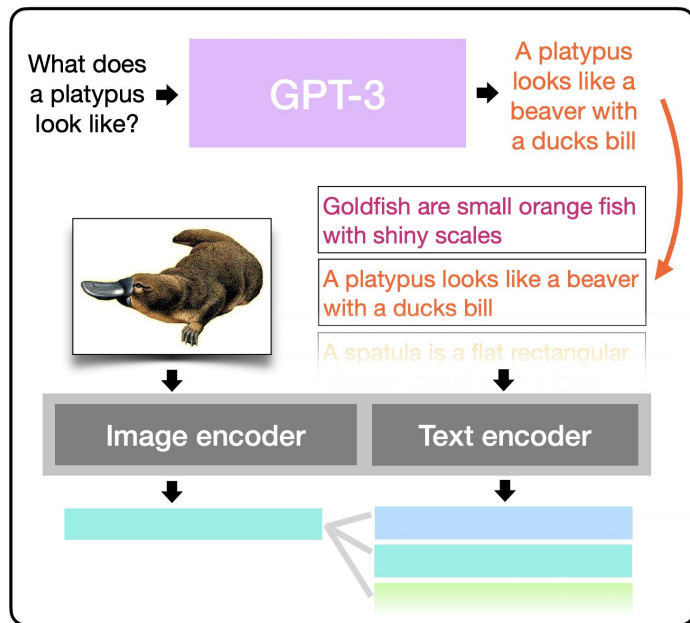
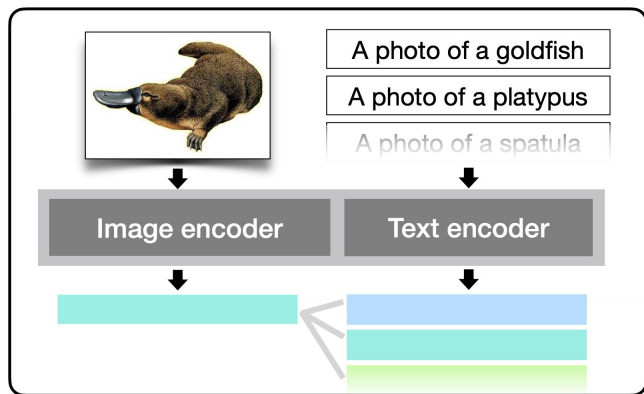
# Appendix

Slides from previous years that we didn't have time to get to today, due to adding new material.

# CLIP

# Advantages of CLIP-style models

## Chaining LMs + CLIP:

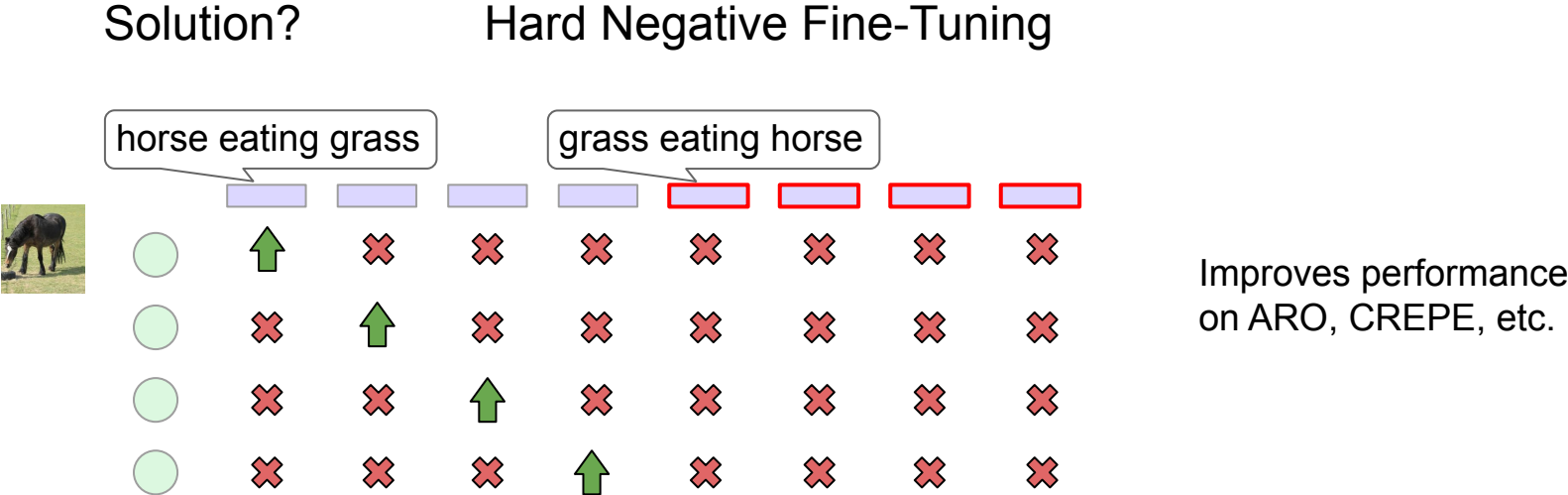


Improvements on a wide variety of classification tasks!

Pratt et al., 2022: "What does a platypus look like? Generating customized prompts for zero-shot image classification"

# Disadvantages of CLIP-style models

## 1. Rely too heavily on batch size to learn concepts



# Disadvantages of CLIP-style models

## 1. Rely too heavily on batch size to learn concepts

Solution?

Hard Negative Fine-Tuning

But, Hard Negative Fine-Tuning has its own problems...



“A black cat and a brown dog”

✓

“A brown cat and a black dog”

✗

“A brown dog and a black cat”

✗



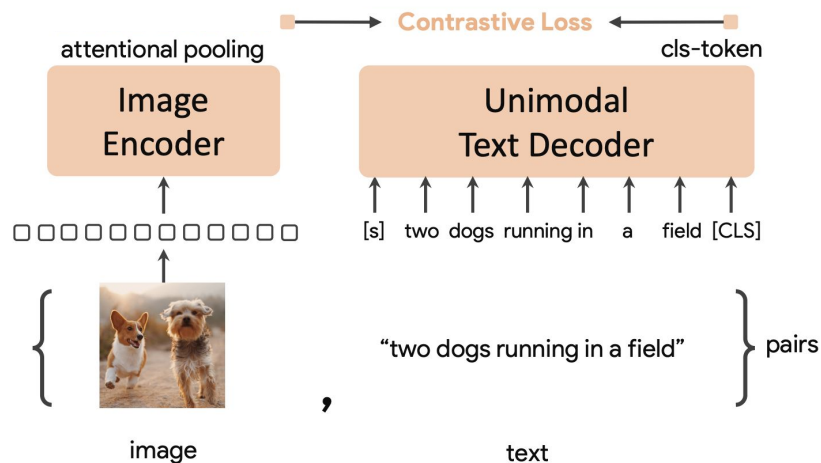
causes *oversensitivity*

“hard positives”

Kamath et al, 2024: “The Hard Positive Truth about Vision-Language Compositionality”

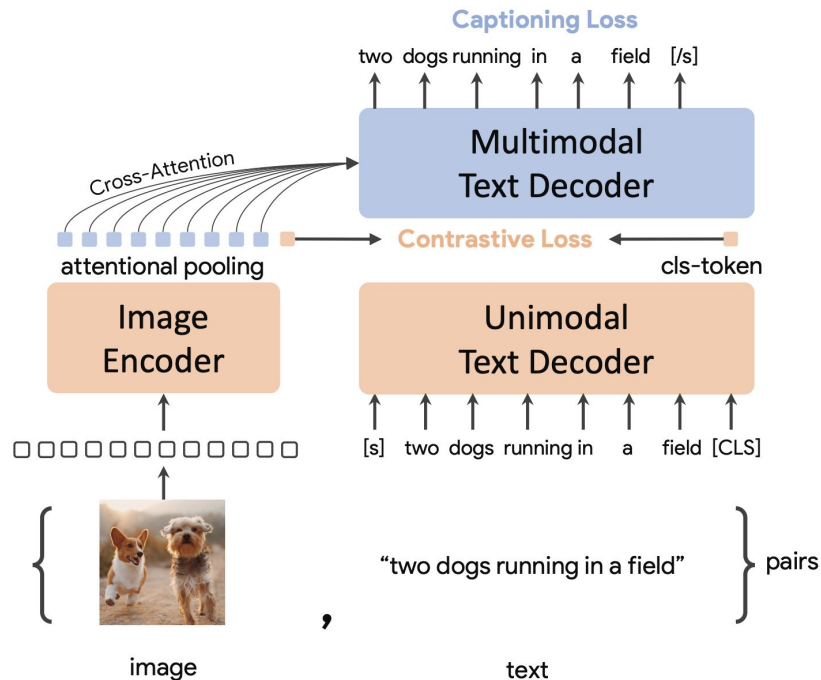
CoCa

# CoCa improved upon CLIP by adding a generation objective



“Contrastive Captioners are Image-Text Foundation Models”, 2022

# CoCa added a decoder with a captioning loss



"Contrastive Captioners are Image-Text Foundation Models", 2022

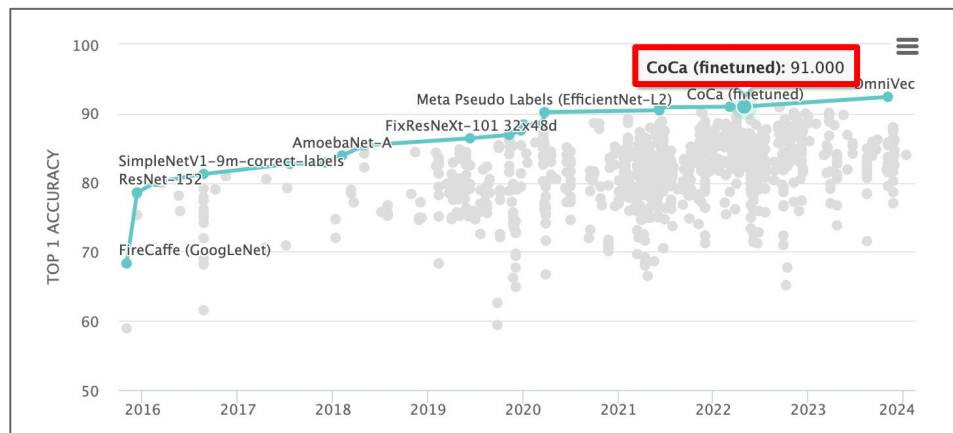
# CoCa: Contrastive Captioners are Image-Text Foundation Models

Model	ImageNet	ImageNet-A	ImageNet-R	ImageNet-V2	ImageNet-Sketch	ObjectNet	Average
CLIP [12]	76.2	77.2	88.9	70.1	60.2	72.3	74.3
ALIGN [13]	76.4	75.8	92.2	70.1	64.8	72.2	74.5
FILIP [61]	78.3	-	-	-	-	-	-
Florence [14]	83.7	-	-	-	-	-	-
LiT [32]	84.5	79.4	93.9	78.7	-	81.1	-
BASIC [33]	85.7	85.6	95.7	80.6	76.1	78.9	83.7
CoCa-Base	82.6	76.4	93.2	76.5	71.7	71.6	78.7
CoCa-Large	84.8	85.7	95.6	79.6	75.7	78.6	83.3
<b>CoCa</b>	<b>86.3</b>	<b>90.2</b>	<b>96.5</b>	<b>80.7</b>	<b>77.6</b>	<b>82.7</b>	<b>85.7</b>

Table 4: Zero-shot image classification results on ImageNet [9], ImageNet-A [64], ImageNet-R [65], ImageNet-V2 [66], ImageNet-Sketch [67] and ObjectNet [68].

# Classifier foundation models now beat all other models on ImageNet

Model	ImageNet
ALIGN [13]	88.6
Florence [14]	90.1
MetaPseudoLabels [51]	90.2
CoAtNet [10]	90.9
ViT-G [21]	90.5
+ Model Soups [52]	90.9
CoCa (frozen)	90.6
<b>CoCa (finetuned)</b>	<b>91.0</b>



# LLAVA implementation details

# LLaVA: Training

Prompt given to GPT4 to generate data:

```
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.

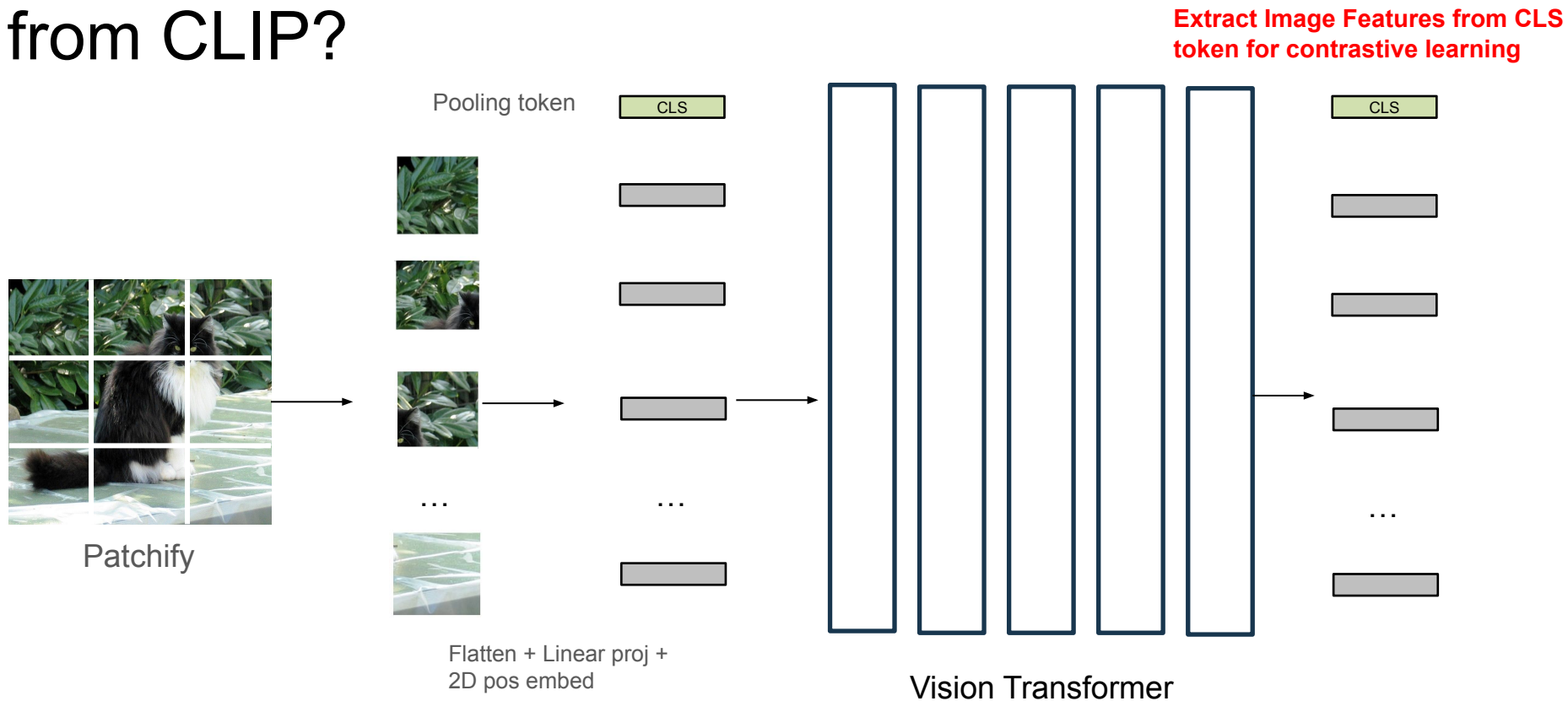
Include questions asking about the visual content of the image, including the **object types, counting the objects, object actions, object locations, relative positions between objects**, etc. Only include questions that have definite answers:

- (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."""

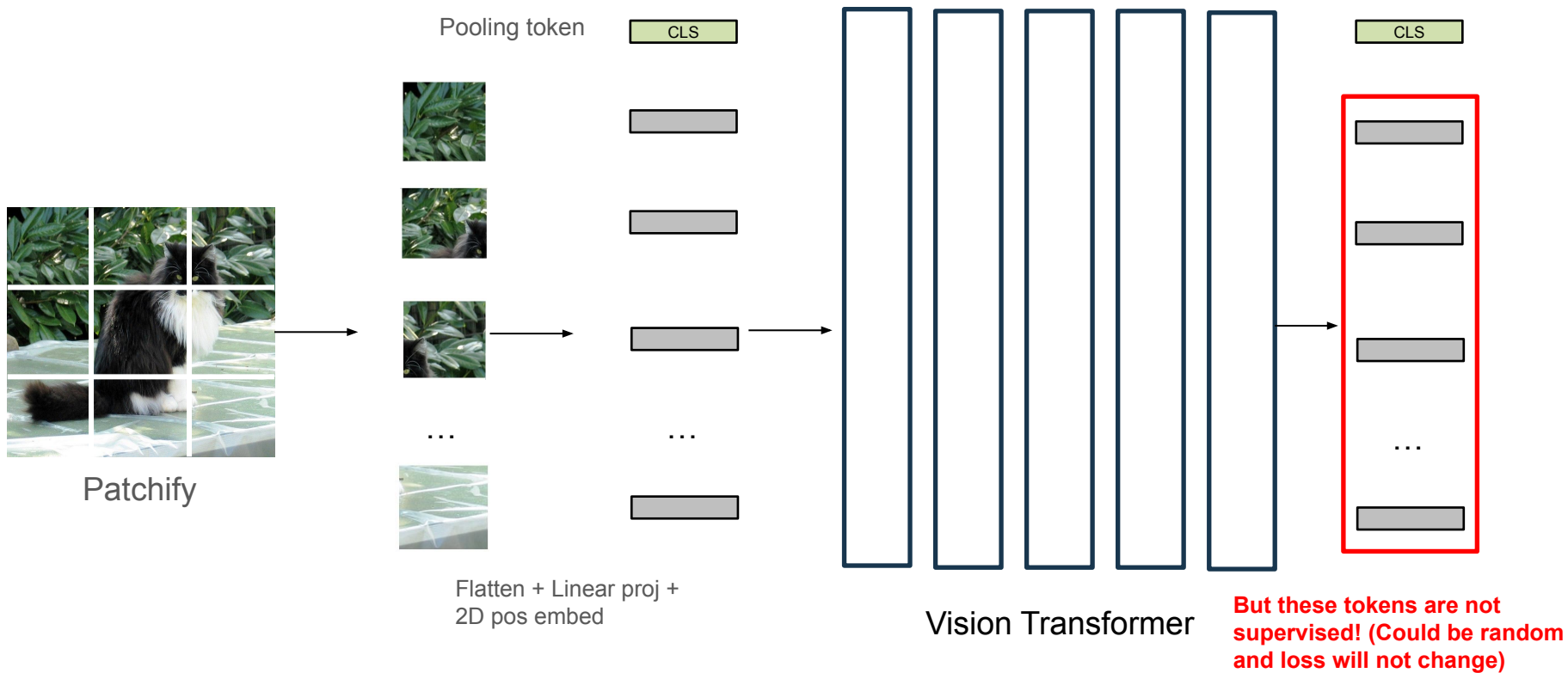
```
    ]  
    for sample in fewshot_samples:  
        messages.append({"role": "user", "content": sample['context']})  
        messages.append({"role": "assistant", "content": sample['response']})  
    messages.append({"role": "user", "content": '\n'.join(query)})
```

# What features should we use from CLIP?

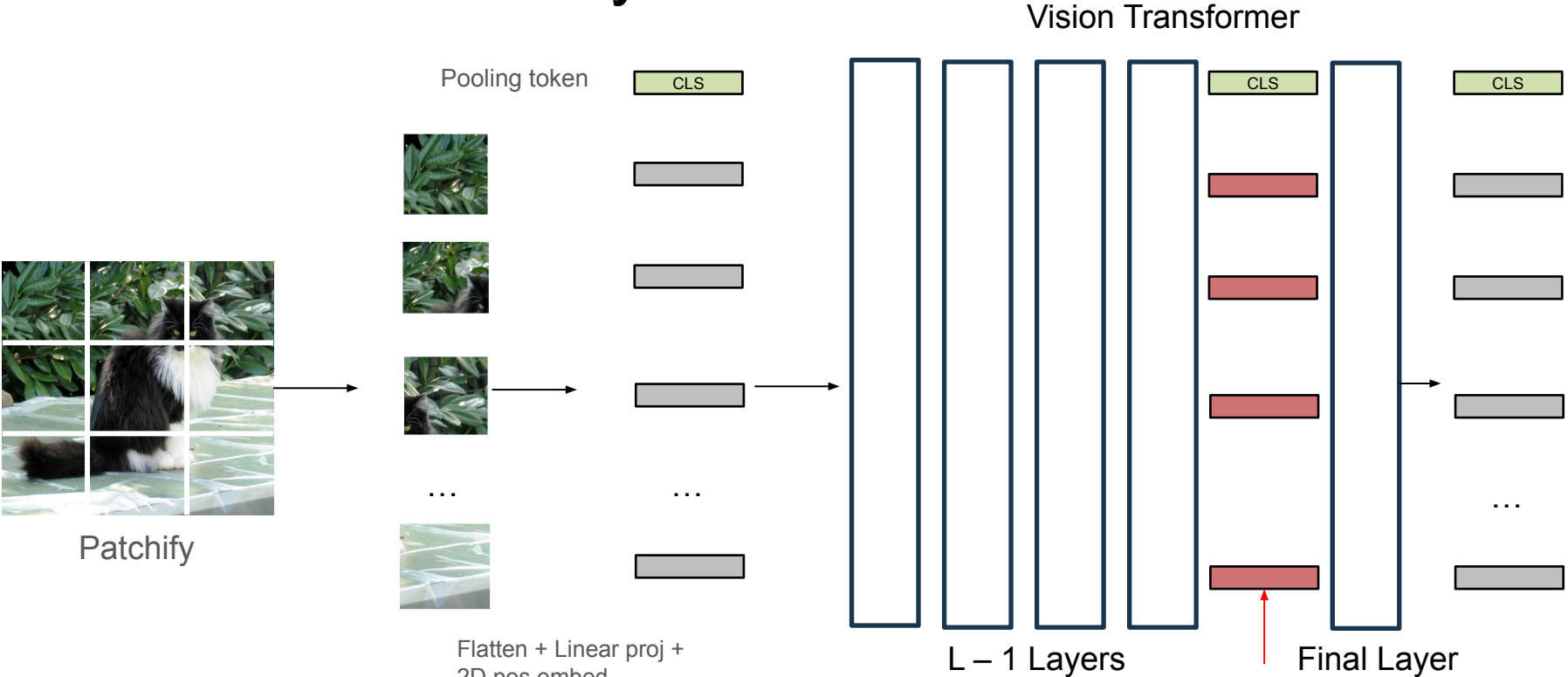


[Image source]

# What features should we use from CLIP?



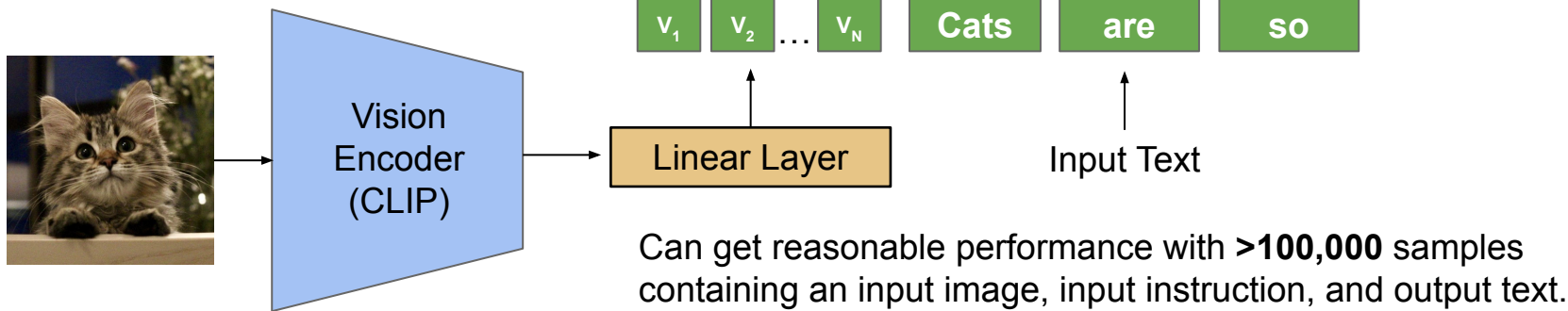
# Use Penultimate Layer!



**In practice, these tokens preserve spatial and linguistic information best for LLMs. Can drop CLS for slight gains.**

# LLaVA – Overall Architecture + Training Recipe

1. Initialize with pretrained Language Model for LLM Decoder (e.g. LLaMA) and pretrained image encoder (e.g. CLIP)
2. Train a new **linear layer** to bridge CLIP features to LLM input space
3. Finetune LLM + linear layer together



Can get reasonable performance with **>100,000** samples containing an input image, input instruction, and output text.

# Segment Anything Model (SAM)

Images: He et al. Mask R-CNN. 2017

# Foundation Models

## Language

ELMo  
BERT  
GPT  
T5

## Classification

CLIP  
CoCa

## LM + Vision

LLaVA  
Flamingo  
GPT-4V  
Gemini

## And More!

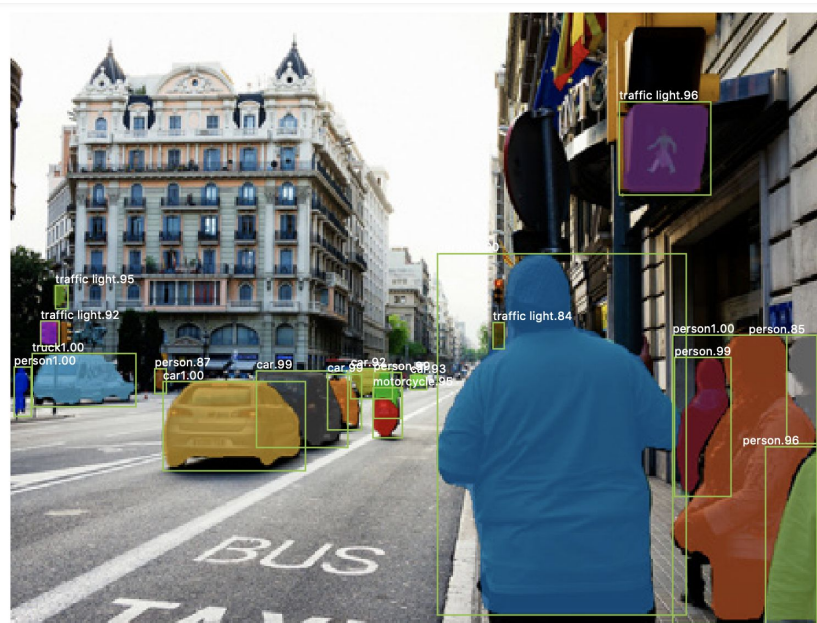
**Segment Anything**  
Whisper  
Dalle  
Stable Diffusion  
Imagen

## Chaining

Visual Programming  
LMs + CLIP

# Segment Anything Model (SAM)

What does it mean to have a segmentation foundation model?



Masking model trained on dataset of specific number of objects (80 in COCO)

Model outputs masks of all objects in that image that is one of the categories of interest

Images: He et al. Mask R-CNN. 2017

# Segment Anything Model (SAM)

What does it mean to have a segmentation foundation model?



Masking model trained on a dataset of a huge number of categories

Model outputs mask of any objects that the user cares about

Images: Kirillov et al. Segment Anything. 2023.

# Segment Anything Model (SAM)

What does it mean to have a segmentation foundation model?



Masking model trained on a dataset of a huge number of categories

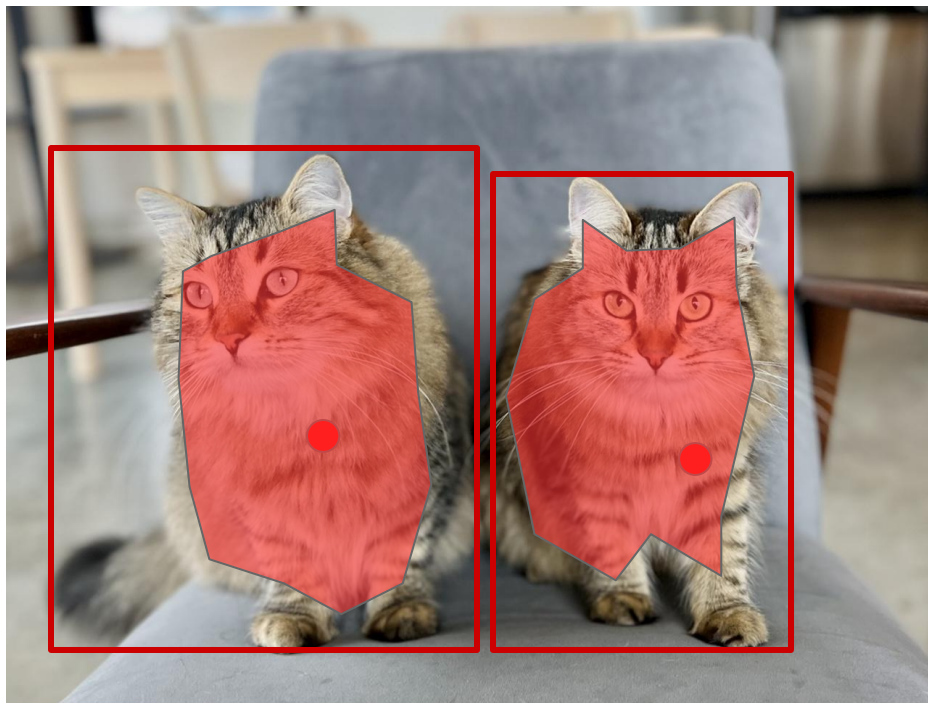
**How to get this?**

Model outputs mask of any objects that the user cares about

**How to know this?**

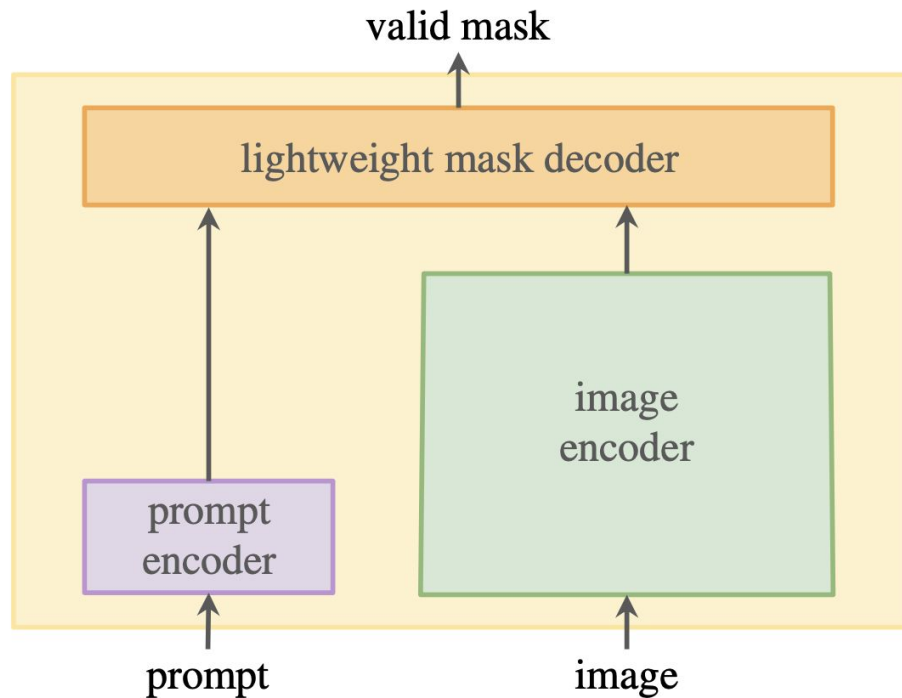
Images: Kirillov et al. Segment Anything. 2023.

# How to know what to mask?



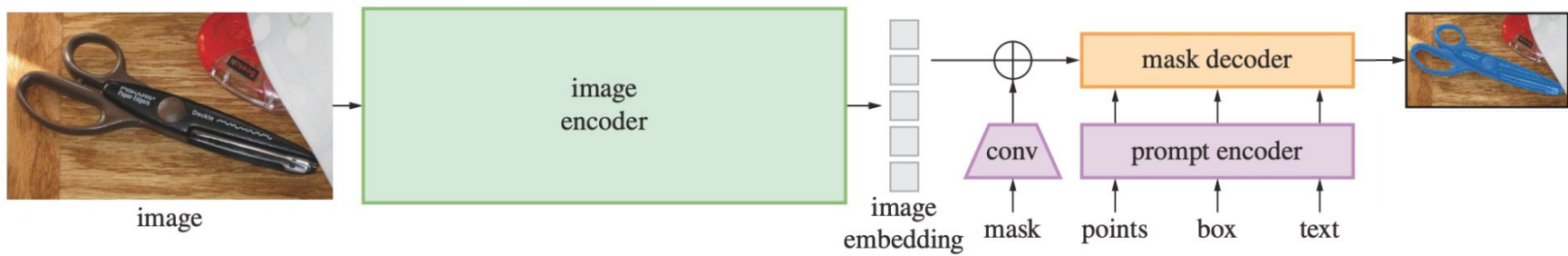
**“Cats”**

# Basic SAM Architecture



Images: Kirillov et al. Segment Anything. 2023.

# SAM Architecture



Images: Kirillov et al. Segment Anything. 2023.

# Ambiguity in correct prompt



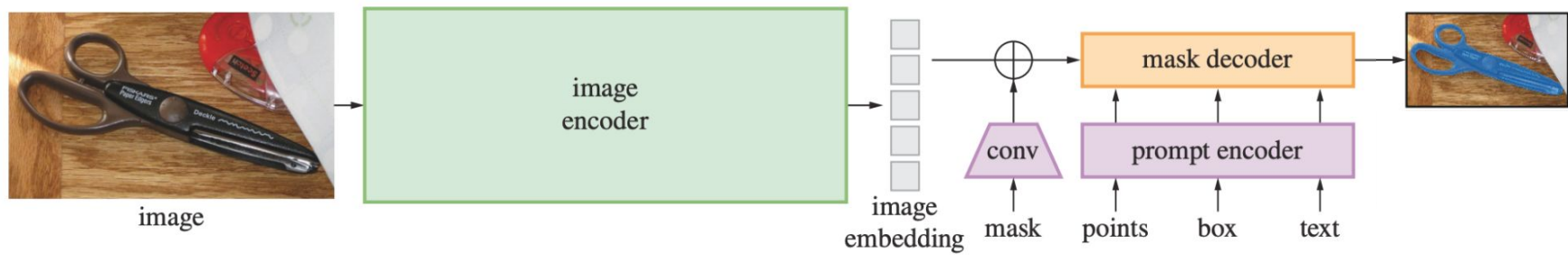
Images: Kirillov et al. Segment Anything. 2023.

# Ambiguity in correct prompt



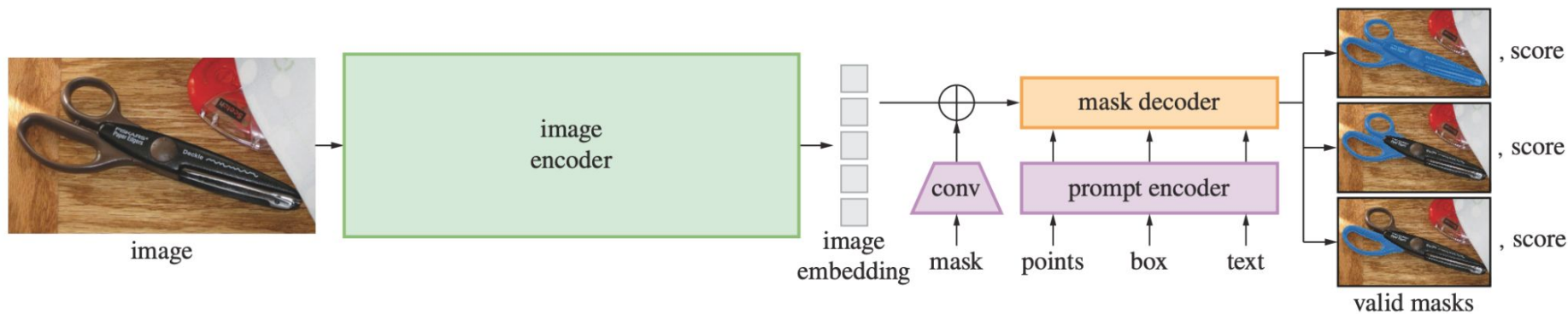
Images: Kirillov et al. Segment Anything. 2023.

# SAM Architecture



Images: Kirillov et al. Segment Anything. 2023.

# SAM Architecture



1. Loss only calculated with respect to best mask
2. Model also trained to output confidence score for each mask

Images: Kirillov et al. Segment Anything. 2023.

# Segment Anything Model (SAM)

What does it mean to have a segmentation foundation model?



Masking model trained on a dataset of a huge number of categories

**How to get this?**

Model outputs mask of any objects that the user cares about

**How to know this?**

Images: Kirillov et al. Segment Anything. 2023.

# Segment Anything Model (SAM)

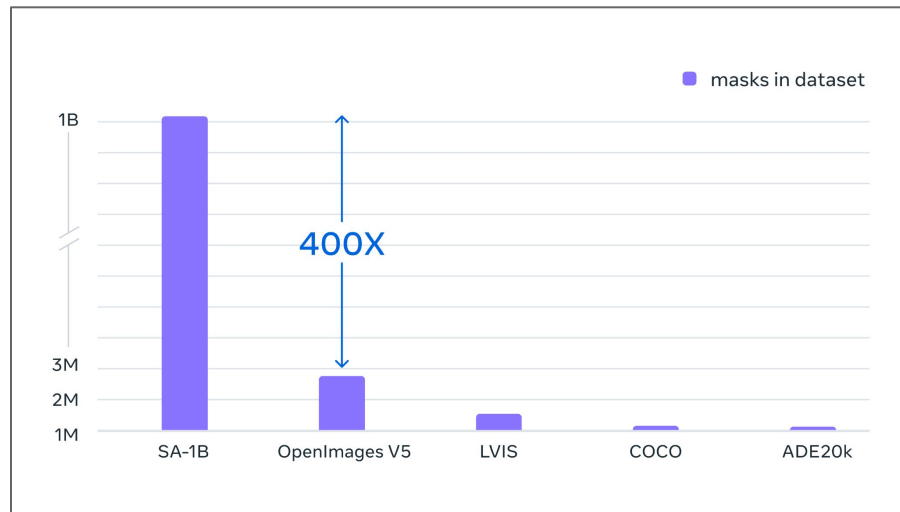
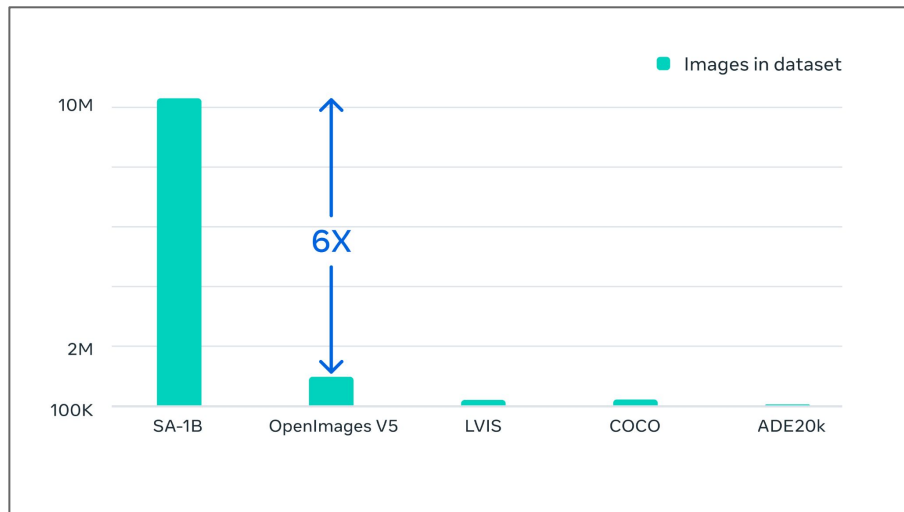
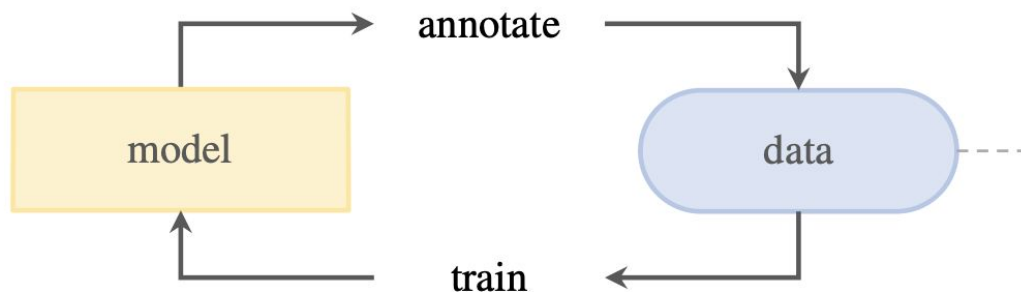


Image Source: <https://segment-anything.com/>

# Segment Anything Model (SAM)



## Segment Anything 1B (SA-1B):

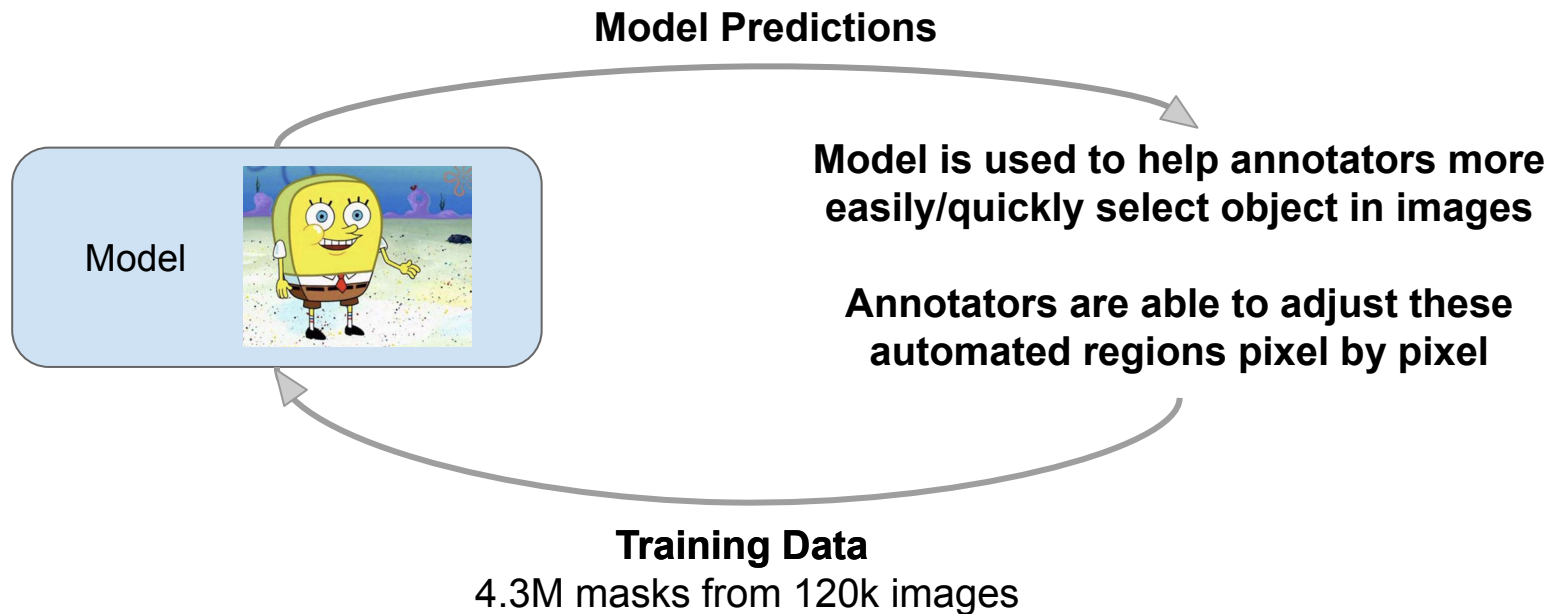
- 1+ billion masks
- 11 million images
- privacy respecting
- licensed images



Images: Kirillov et al. Segment Anything. 2023.

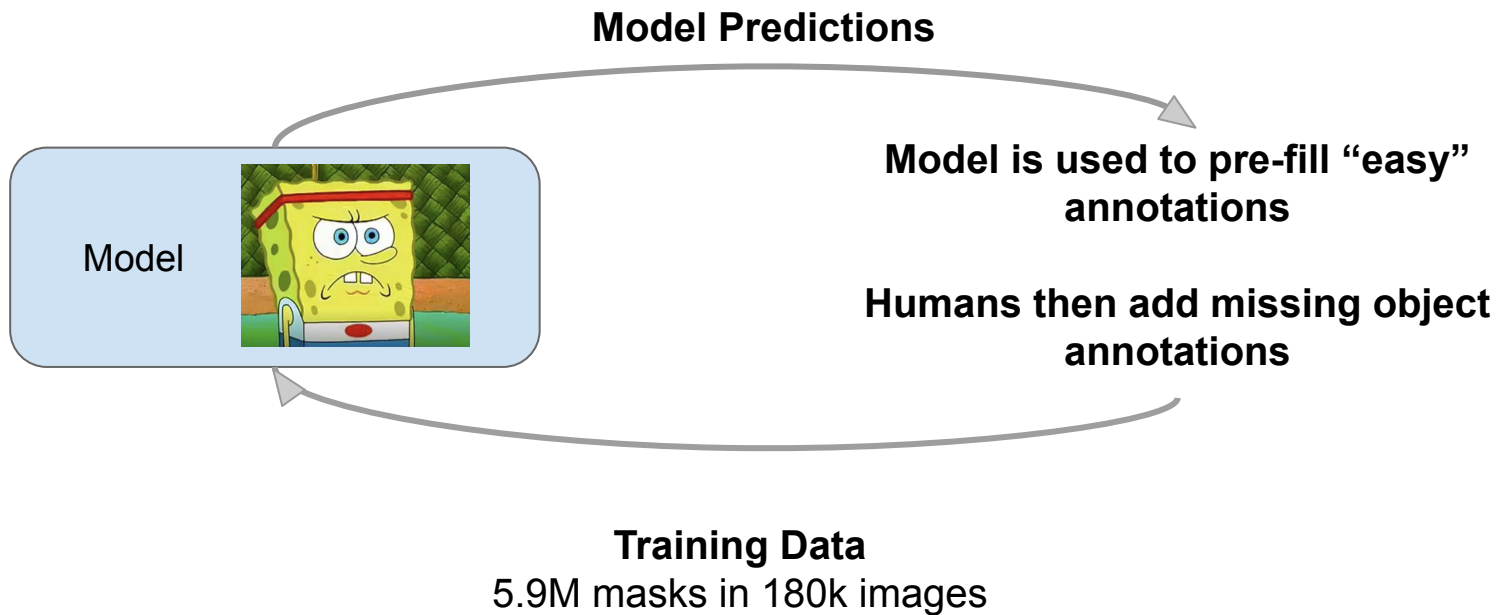
# Segment Anything Model (SAM)

## Assisted-manual stage



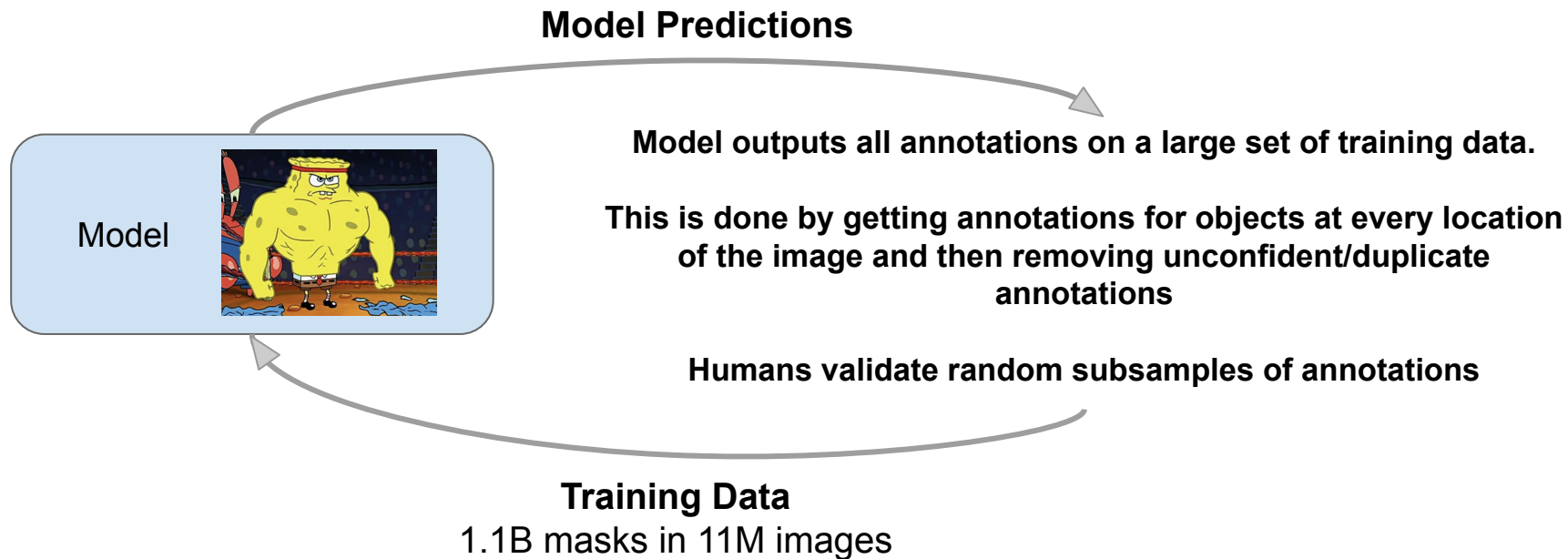
# Segment Anything Model (SAM)

## Semi-automatic stage



# Segment Anything Model (SAM)

## Fully automatic stage



# SAM Results



Image Source: Kirillov et al. Segment Anything. 2023

# SAM Results



Image Source: Kirillov et al. Segment Anything. 2023

# Zero-Shot with SAM

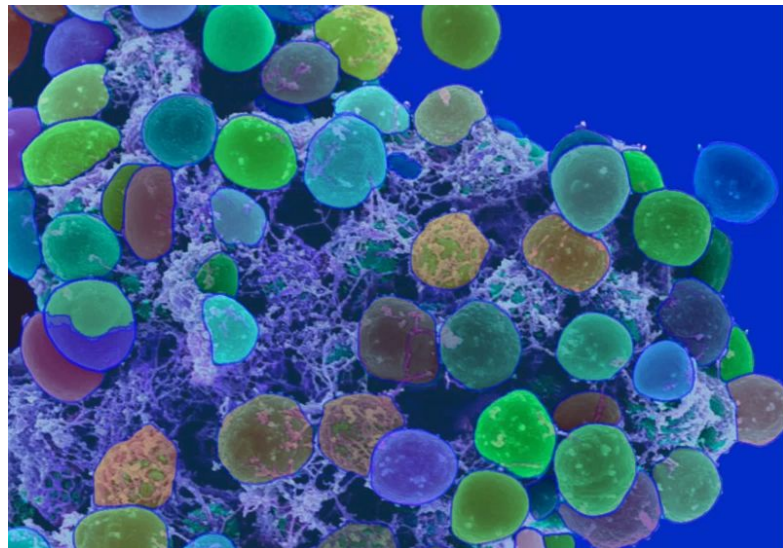
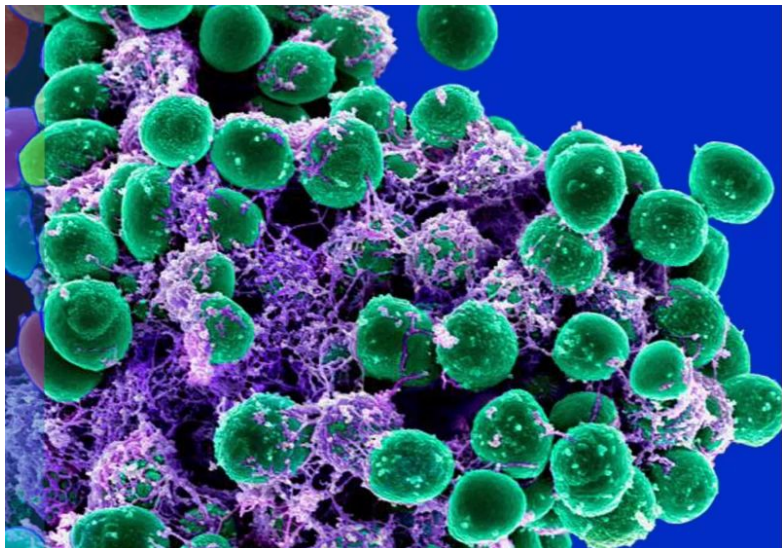


Image Source: <https://segment-anything.com/>

# Zero-Shot with SAM

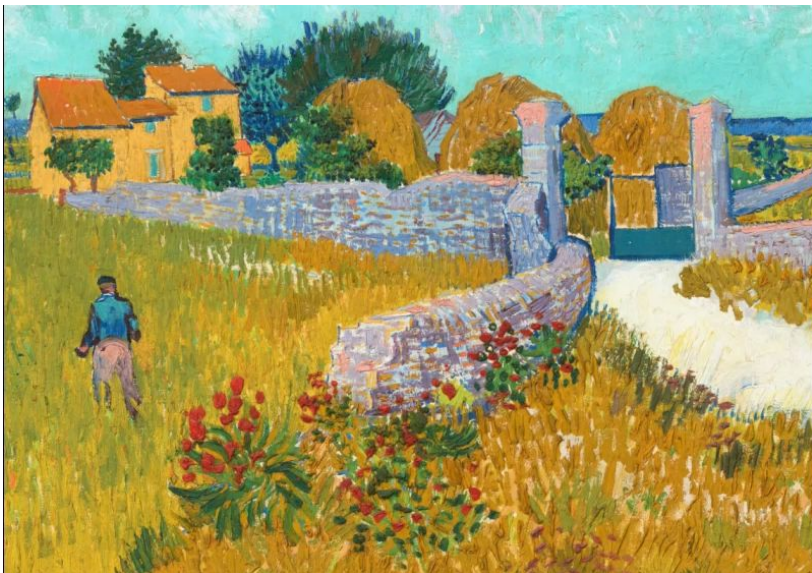


Image Source: <https://segment-anything.com/>