

Lecture 17:

Multimodal Foundation Models

Administrative: Last Deadlines

- A5 due Friday (3/13)
- Multiple W Credit assignments due next Monday (3/16)
- **Absolutely no submissions accepted for anything after 11:59 PM on Monday, March 16th**

Administrative: Project

- Poster session during final exam slot (3/16, 10:30-12:20)
 - Allen Center Atrium
 - We will provide easels
 - Guests/friends/collaborators OK to attend
 - Half the class with stand by and present at a time
- Two options:
 - Print poster yourself & bring it on Monday
 - Upload poster to [Google Form](#) by 9 AM **this Friday** & TAs will bring print-out to poster session

Administrative: Project

- Must **upload** PDF of poster to [Gradescope](#)
 - Please make 1 submission per group
- Failure to do this will result in 0% grade for poster

LLMs

Encoder Only:

I love cake

ELMO: Bi-directional next word prediction,
BERT: Masked language objective, Next Sentence Prediction

Decoder Only:

I love

GPT: text token prediction

Encoder-Decoder:

I love cake

me gusta

T5: Masked language objective

In-Context Learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 cheese => ..... ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← example
3 cheese => ..... ← prompt
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← examples
3 peppermint => menthe poivrée ←
4 plush girafe => girafe peluche ←
5 cheese => ..... ← prompt
```

No training. No gradients.

Image Source: [Language Models are Few-Shot Learners, Brown et al](#)

GPT Results

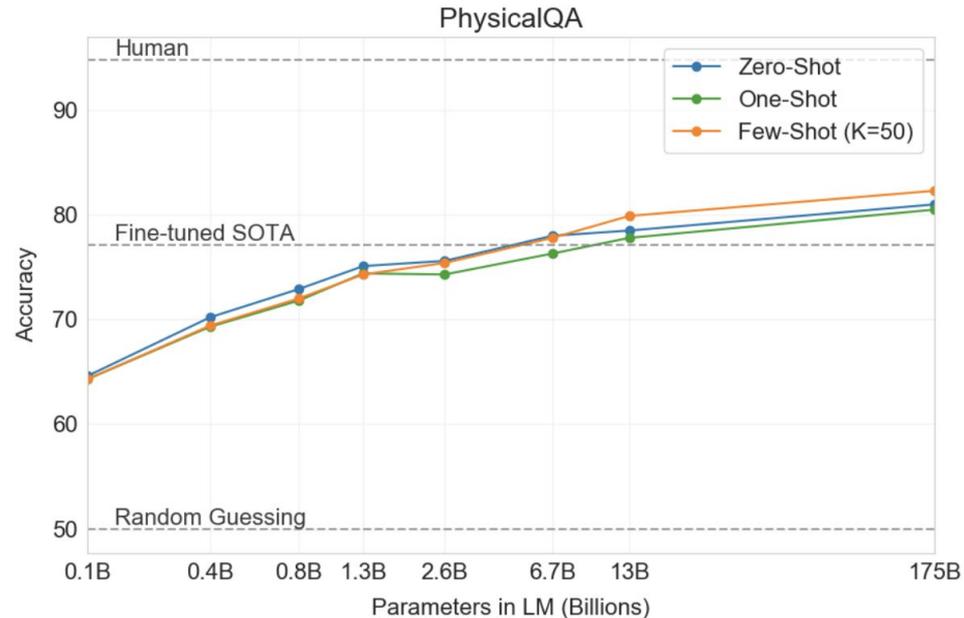
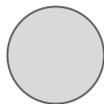


Figure 3.6: GPT-3 results on PIQA in the zero-shot, one-shot, and few-shot settings. The largest model achieves a score on the development set in all three conditions that exceeds the best recorded score on the task.

Image Source: [Language Models are Few-Shot Learners, Brown et al](#)

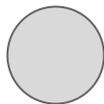
Scale!
(in params)



BERT (Large)
340 million params

GPT - 3
175 billion params

Scale! (In data)



BERT

3.3 Billion tokens¹

- **All of english wikipedia**
- **11,000 Books**

GPT - 3

~300 billion tokens

- **Common Crawl (Much of the internet)**

¹<https://aclanthology.org/W19-4828.pdf>

The Bitter Lesson

Rich Sutton

March 13, 2019

The biggest lesson that can be read from 70 years of AI research is that general methods that leverage computation are ultimately the most effective, and by a large margin. The ultimate reason for this is Moore's law, or rather its generalization of continued exponentially falling cost per unit of computation. Most AI research has been conducted as if the computation available to the agent were constant (in which case leveraging human knowledge would be one of the only ways to improve performance) but, over a slightly longer time than a typical research project, massively more computation inevitably becomes available. Seeking an improvement that makes a difference in the shorter term, researchers seek to leverage their human knowledge of the domain, but the only thing that matters in the long run is the leveraging of computation. These two need not run counter to each other, but in practice they tend to. Time spent on one is time not spent on the other. There are psychological commitments to investment in one approach or the other. And the human-knowledge approach tends to complicate methods in ways that make them less suited to taking advantage of general methods leveraging computation. There were many examples of AI researchers' belated learning of this bitter lesson, and it is instructive to review some of the most prominent.

<http://www.incompleteideas.net/Incldeas/BitterLesson.html>

LLMs

Building LLMs: Pre-training objectives + architectures

- Encoder only
- Decoder only
- Encoder Decoder

GPT

Gradient-Free Performance Improvement

Improving accuracy without fine-tuning

In-context learning is better than zero-shot with no additional training

Can we do better than in context learning with no additional training?

Chain of thought prompting

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

...

Q: John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. How many hours a week does he spend taking care of dogs?

A:

GPT (or similar)

The answer is 50. ❌

	GSM8K
Finetuned GPT-3 175B	33%
Finetuned GPT-3 175B + verifier (prior SOTA)	55%
9–12 year olds (Cobbe et al., 2021)	60%
PaLM 540B: standard prompting	17.9%

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought prompting

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

...

Q: John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. How many hours a week does he spend taking care of dogs?

A:

GPT (or similar)

	GSM8K
Finetuned GPT-3 175B	33%
Finetuned GPT-3 175B + verifier (prior SOTA)	55%
9–12 year olds (Cobbe et al., 2021)	60%
PaLM 540B: standard prompting	17.9%

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought prompting

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

...

Q: John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. How many hours a week does he spend taking care of dogs?

A:

GPT (or similar)

John takes care of 10 dogs. Each dog takes .5 hours a day to walk and take care of their business. So that is $10 \times .5 = 5$ hours a day. $5 \text{ hours a day} \times 7 \text{ days a week} = 35 \text{ hours a week}$. The answer is 35 hours a week. ✓

	GSM8K
Finetuned GPT-3 175B	33%
Finetuned GPT-3 175B + verifier (prior SOTA) 9–12 year olds (Cobbe et al., 2021)	55% 60%
PaLM 540B: standard prompting	17.9%
PaLM 540B: chain of thought prompting	58.1%

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought for Math Problems

Table 20: Few-shot exemplars for full chain of thought prompt for math word problems. This set of exemplars was used for all math word problem datasets except AQuA.

PROMPT FOR MATH WORD PROBLEMS

Q: There are 15 trees in the grove. Grove workers will plant trees in the grove today. After they are done, there will be 21 trees. How many trees did the grove workers plant today?

A: There are 15 trees originally. Then there were 21 trees after some more were planted. So there must have been $21 - 15 = 6$. The answer is 6.

Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?

A: There are originally 3 cars. 2 more cars arrive. $3 + 2 = 5$. The answer is 5.

Q: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?

A: Originally, Leah had 32 chocolates. Her sister had 42. So in total they had $32 + 42 = 74$. After eating 35, they had $74 - 35 = 39$. The answer is 39.

Q: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?

A: Jason started with 20 lollipops. Then he had 12 after giving some to Denny. So he gave Denny $20 - 12 = 8$. The answer is 8.

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought for Math Problems

Table 8: Example of correct chains of thought produced by the model for the GSM8K dataset.

QUESTION: James decides to run 3 sprints 3 times a week. He runs 60 meters each sprint. How many total meters does he run a week?

MODEL ANSWER (CORRECT): James decides to run 3 sprints 3 times a week. He runs 60 meters each sprint. So he runs 60 meters x 3 sprints x 3 times a week. That is 60 meters x 9. The answer is 540. ✓

QUESTION: Brandon's iPhone is four times as old as Ben's iPhone. Ben's iPhone is two times older than Suzy's iPhone. If Suzy's iPhone is 1 year old, how old is Brandon's iPhone?

MODEL ANSWER (CORRECT): Brandon's iPhone is 4 times as old as Ben's iPhone. Ben's iPhone is 2 times older than Suzy's iPhone. So Brandon's iPhone is $4 \times 2 = 8$ times older than Suzy's iPhone. Suzy's iPhone is 1 year old. So Brandon's iPhone is $8 \times 1 = 8$ years old. The answer is 8. ✓

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought for Symbolic Reasoning

PROMPT FOR LAST LETTER CONCATENATION

Q: Take the last letters of the words in "Elon Musk" and concatenate them.

A: The last letter of "Elon" is "n". The last letter of "Musk" is "k". Concatenating them is "nk". The answer is nk.

Q: Take the last letters of the words in "Larry Page" and concatenate them.

A: The last letter of "Larry" is "y". The last letter of "Page" is "e". Concatenating them is "ye". The answer is ye.

Q: Take the last letters of the words in "Sergey Brin" and concatenate them.

A: The last letter of "Sergey" is "y". The last letter of "Brin" is "n". Concatenating them is "yn". The answer is yn.

Q: Take the last letters of the words in "Bill Gates" and concatenate them.

A: The last letter of "Bill" is "l". The last letter of "Gates" is "s". Concatenating them is "ls". The answer is ls.

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought for Symbolic Reasoning

Table 13: Examples of correct and incorrect chains of thought produced by LaMDA 137B on the letter concatenation task.

QUESTION: Take the last letters of the words in “Waldo Schmidt” and concatenate them.

MODEL ANSWER (CORRECT): The last letter of “Waldo” is “o”. The last letter of “Schmidt” is “t”. Concatenating them is “ot”. So the answer is ot. ✓

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought for Physical Reasoning

PROMPT FOR COIN FLIP

Q: A coin is heads up. Ka flips the coin. Sherrie flips the coin. Is the coin still heads up?

A: The coin was flipped by Ka and Sherrie. So the coin was flipped 2 times, which is an even number. The coin started heads up, so after an even number of flips, it will still be heads up. So the answer is yes.

Q: A coin is heads up. Jamey flips the coin. Teressa flips the coin. Is the coin still heads up?

A: The coin was flipped by Jamey and Teressa. So the coin was flipped 2 times, which is an even number. The coin started heads up, so after an even number of flips, it will still be heads up. So the answer is yes.

Q: A coin is heads up. Maybelle flips the coin. Shalonda does not flip the coin. Is the coin still heads up?

A: The coin was flipped by Maybelle. So the coin was flipped 1 time, which is an odd number. The coin started heads up, so after an odd number of flips, it will be tails up. So the answer is no.

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

Chain of thought for Physical Reasoning

QUESTION: A coin is heads up. Andree flips the coin. Audrie does not flip the coin. Is the coin still heads up?

MODEL ANSWER (CORRECT): The coin was flipped by Andree. So the coin was flipped 1 time, which is an odd number. The coin started heads up, so after an odd number of flips, it will be tails up. So the answer is no.



Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

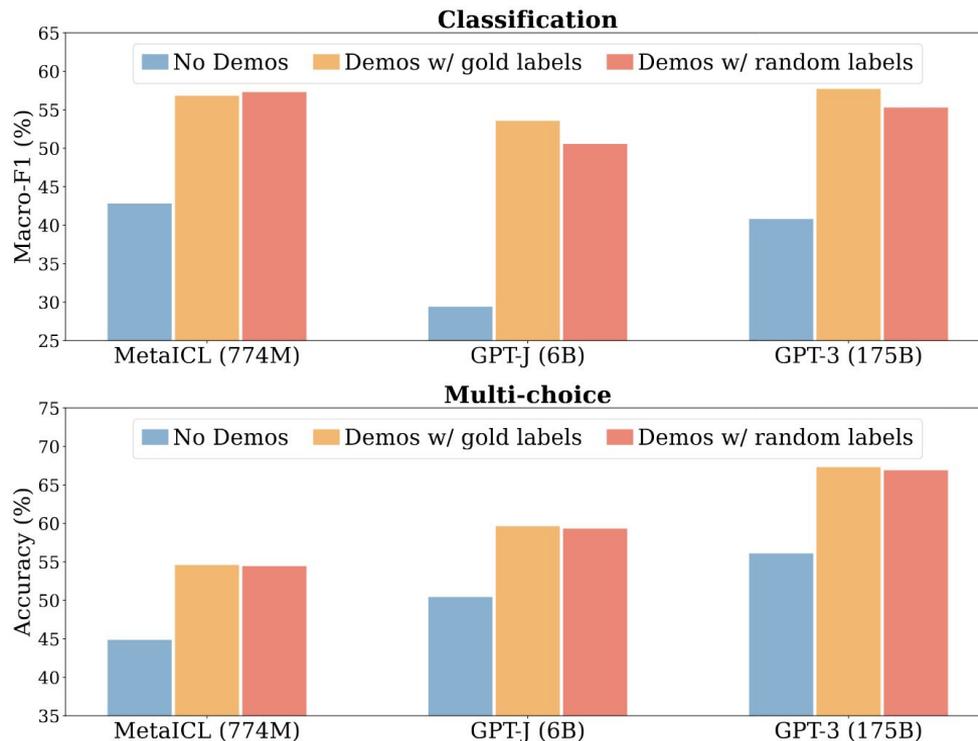
Chain of thought results

	GSM8K	SVAMP	ASDiv	MAWPS
Standard prompting	6.5 \pm 0.4	29.5 \pm 0.6	40.1 \pm 0.6	43.2 \pm 0.9
Chain of thought prompting	14.3 \pm 0.4	36.7 \pm 0.4	46.6 \pm 0.7	57.9 \pm 1.5

	Commonsense			Symbolic	
	Date	Sports	SayCan	Concat	Coin
Standard prompting	21.5 \pm 0.6	59.5 \pm 3.0	80.8 \pm 1.8	5.8 \pm 0.6	49.0 \pm 2.1
Chain of thought prompting	26.8 \pm 2.1	85.8 \pm 1.8	91.7 \pm 1.4	77.5 \pm 3.8	99.6 \pm 0.3

Image Source: [Wei et al. Chain of Thought Prompting Elicits Reasoning in Large Language Models. 2022.](#)

In-context CoT traces need not be accurate (!)



[Min et al., 2022](#)

Explicit CoT is usually not required today

Models are **fine-tuned** on CoT traces, so they often do CoT on their own

However, few-shot examples are still useful

b/c CoT elicits a *behavior*, while few-shot provides useful information

[Min et al., 2022](#)

Think step by step

(a) Few-shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: _____

(Output) *The answer is 8.* ✗

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: The answer (arabic numerals) is _____

(Output) *8* ✗

(b) Few-shot-CoT

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: _____

(Output) *The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4.* ✓

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.** _____

(Output) *There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls.* ✓

Image Source: [Chowdhery et al. Large Language Models are Zero-Shot Reasoners. 2022.](#)

Think step by step

	MultiArith	GSM8K
Zero-Shot	17.7	10.4
Few-Shot (2 samples)	33.7	15.6
Few-Shot (8 samples)	33.8	15.6
Zero-Shot-CoT	78.7	40.7
Few-Shot-CoT (2 samples)	84.8	41.3
Few-Shot-CoT (4 samples : First) (*1)	89.2	-
Few-Shot-CoT (4 samples : Second) (*1)	90.5	-
Few-Shot-CoT (8 samples)	93.0	48.7
Zero-Plus-Few-Shot-CoT (8 samples) (*2)	92.8	51.5

Image Source: [Chowdhery et al. Large Language Models are Zero-Shot Reasoners. 2022.](#)

Self Consistency

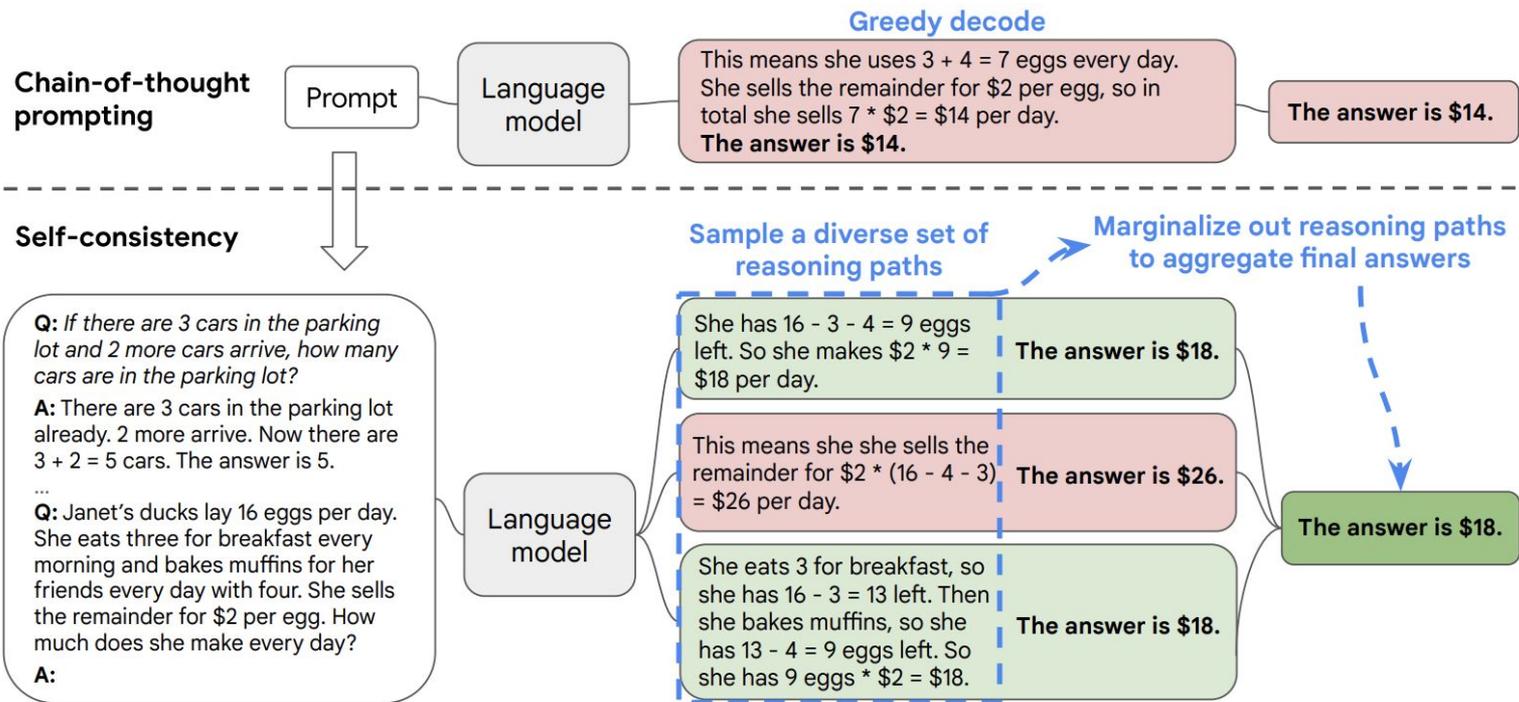


Image Source: [Xie et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. 2022.](#)

Self Consistency

	GSM8K	MultiArith	AQuA	SVAMP	CSQA	ARC-c
Greedy decode	56.5	94.7	35.8	79.0	79.0	85.2
Weighted avg (unnormalized)	56.3 \pm 0.0	90.5 \pm 0.0	35.8 \pm 0.0	73.0 \pm 0.0	74.8 \pm 0.0	82.3 \pm 0.0
Weighted avg (normalized)	22.1 \pm 0.0	59.7 \pm 0.0	15.7 \pm 0.0	40.5 \pm 0.0	52.1 \pm 0.0	51.7 \pm 0.0
Weighted sum (unnormalized)	59.9 \pm 0.0	92.2 \pm 0.0	38.2 \pm 0.0	76.2 \pm 0.0	76.2 \pm 0.0	83.5 \pm 0.0
Weighted sum (normalized)	74.1 \pm 0.0	99.3 \pm 0.0	48.0 \pm 0.0	86.8 \pm 0.0	80.7 \pm 0.0	88.7 \pm 0.0
Unweighted sum (majority vote)	74.4 \pm 0.1	99.3 \pm 0.0	48.3 \pm 0.5	86.6 \pm 0.1	80.7 \pm 0.1	88.7 \pm 0.1

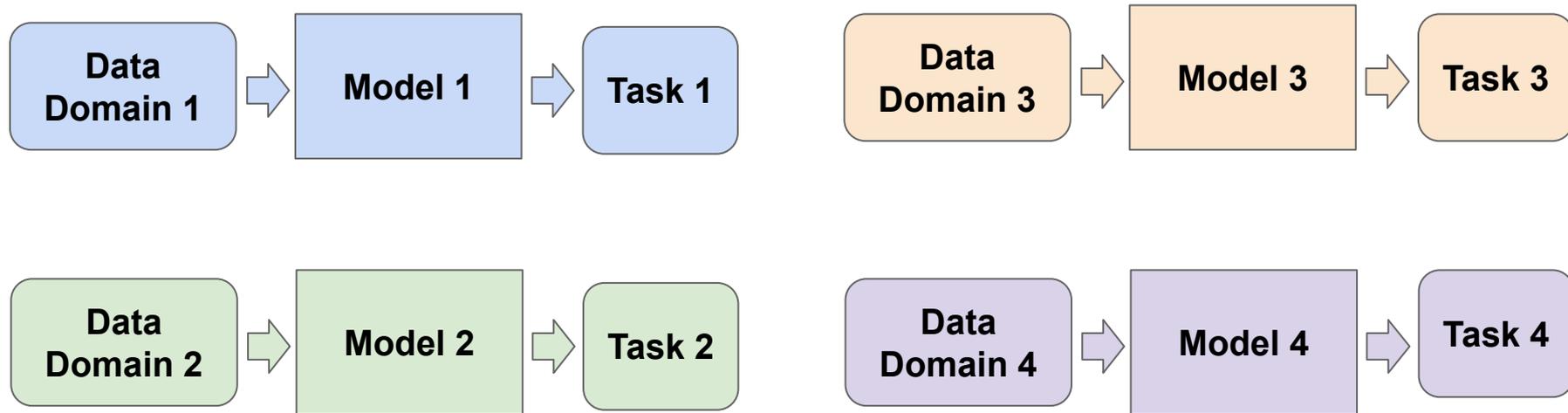
Table 1: Accuracy comparison of different answer aggregation strategies on PaLM-540B.

Image Source: [Xie et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. 2022.](#)

Multimodal Foundation Models

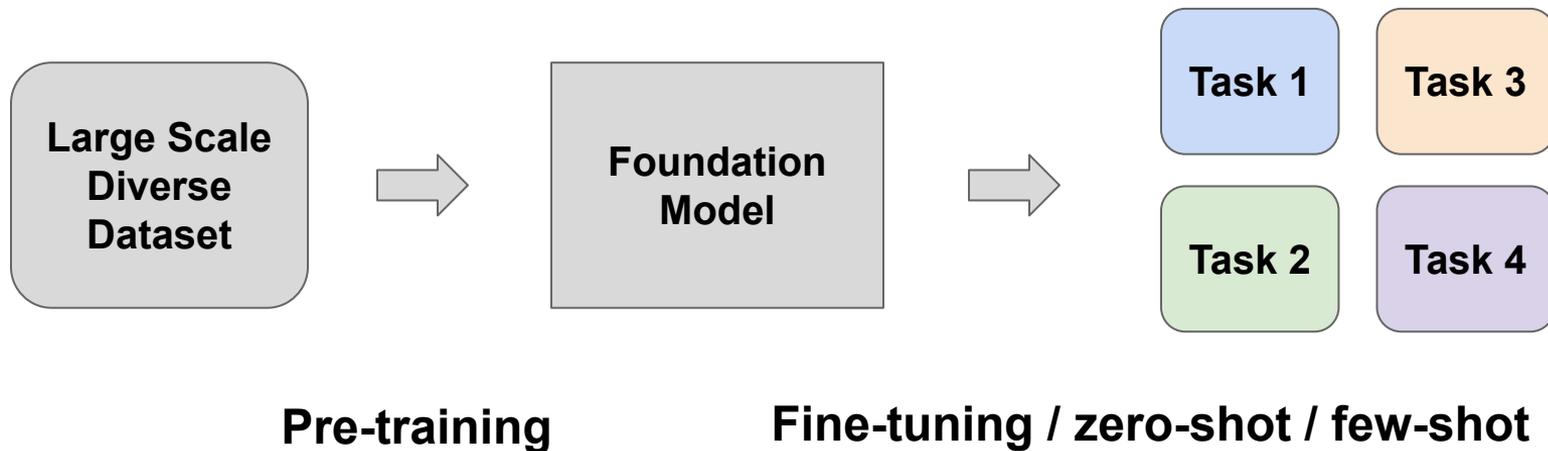
How have we been thinking about models in this class so far?

Train a specialized model for each task



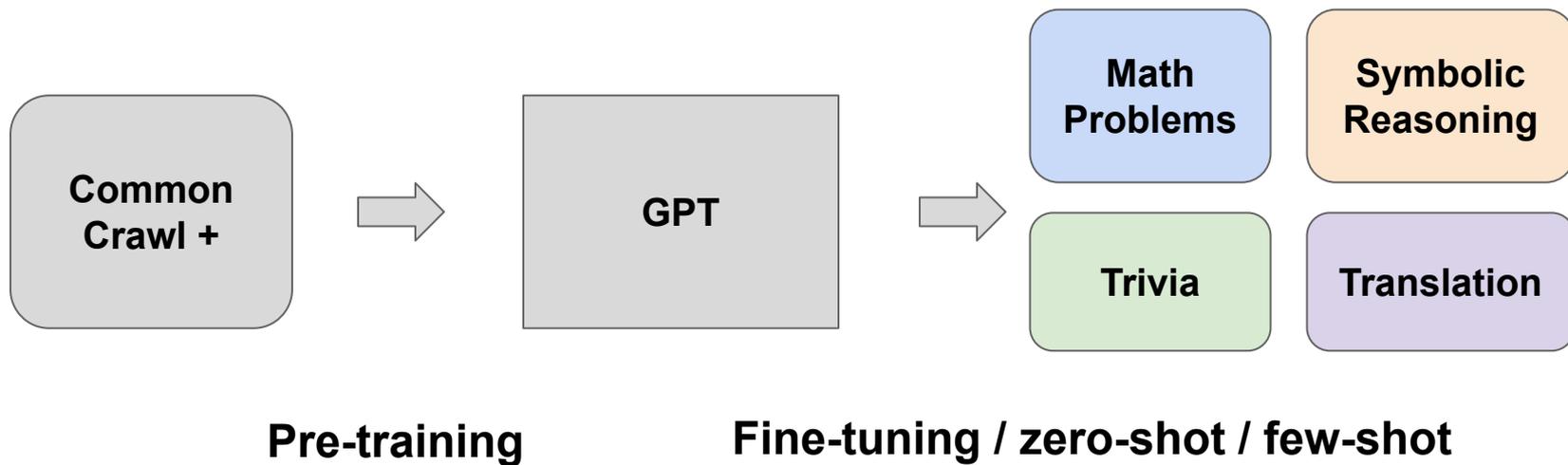
Now, we build **Foundation Models**

Pre-train one model that acts as the foundation for many different tasks



Foundation Models

Language



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	LMs + CLIP
BERT	CoCa	Flamingo	Whisper	Visual Programming
GPT		GPT	Dalle	Tool use
T5		Gemini	Stable Diffusion	
		Molmo	Imagen	

How do identify a model as a Foundation?

Always see with foundation models:

- general /robust to many different tasks

Often see with foundation models:

- Large # params
- Large amount of data
- Self-supervised pre-training objective

Language models are out of scope for this class

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

We will focus on multimodal (vision) foundation models

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

Let's start with the foundation models for classification

Language

ELMo
BERT
GPT
T5

Classification

CLIP
CoCa

LM + Vision

LLaVA
Flamingo
GPT
Gemini
Molmo

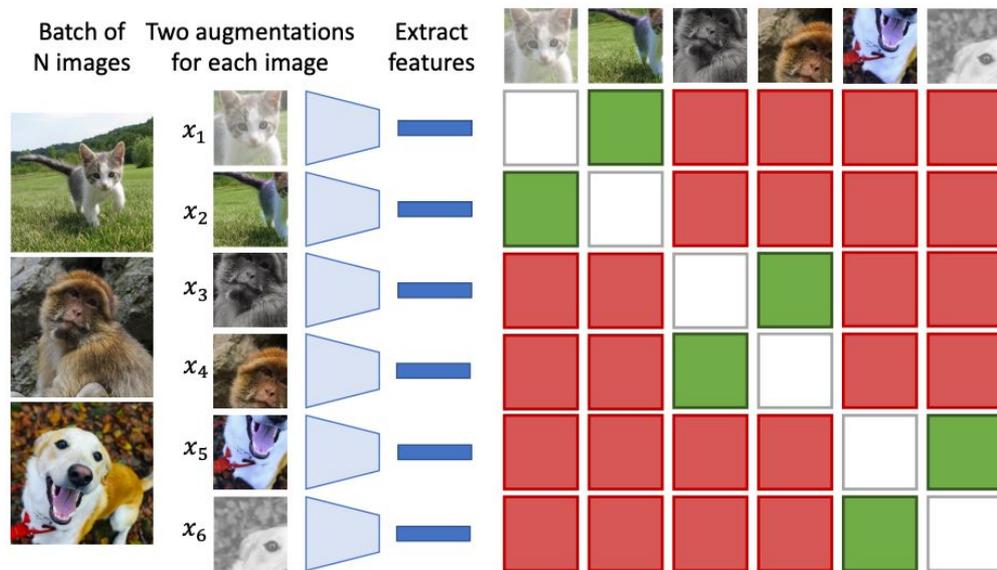
And More!

Segment Anything
Whisper
Dalle
Stable Diffusion
Imagen

Chaining

LMs + CLIP
Visual Programming
Tool use

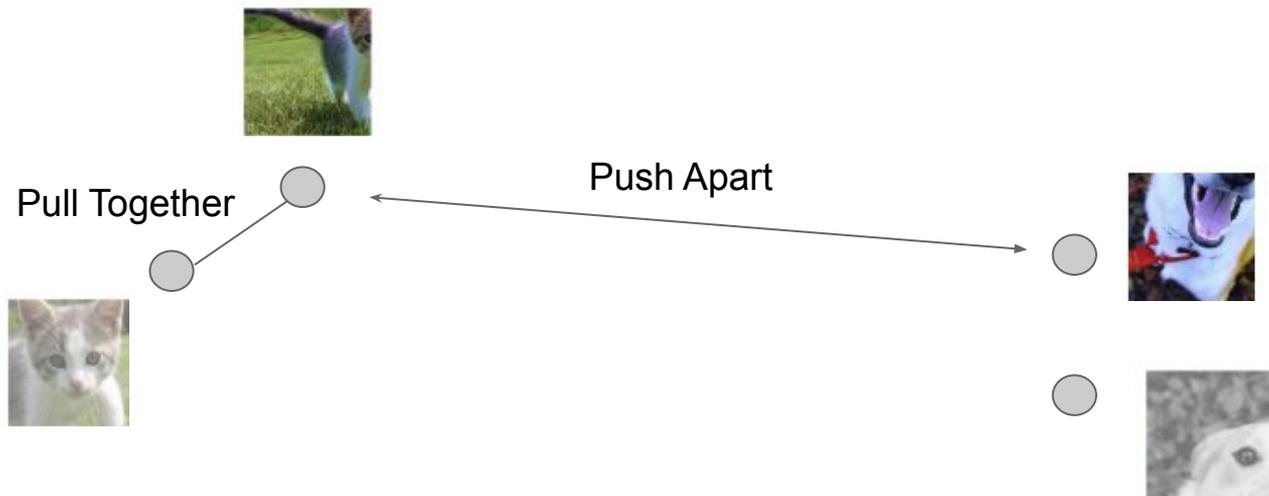
Recall this **self-supervised** objective from SimCLR



Use Self Supervised learning to learn good image features

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

The main idea was to learning concepts without **labels** -> a self-supervised pretraining objective



The hope was that the learned representations generalize to new instances

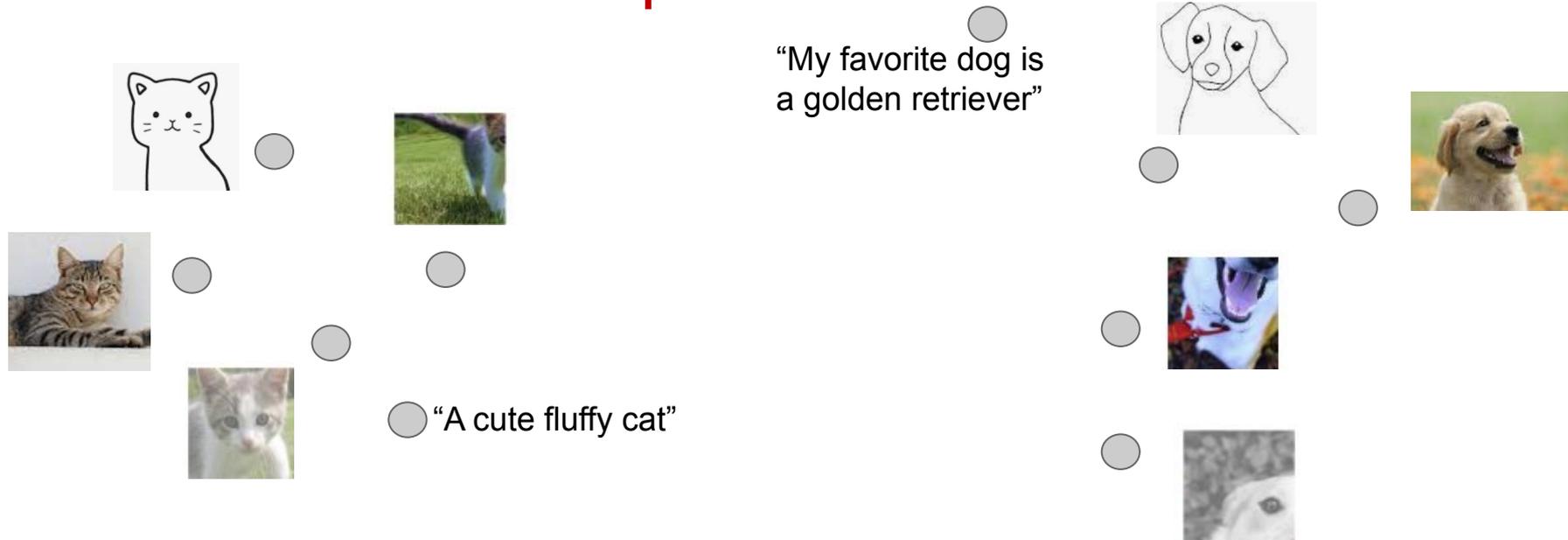


Can we generalize these representations beyond just images? *To language perhaps?*

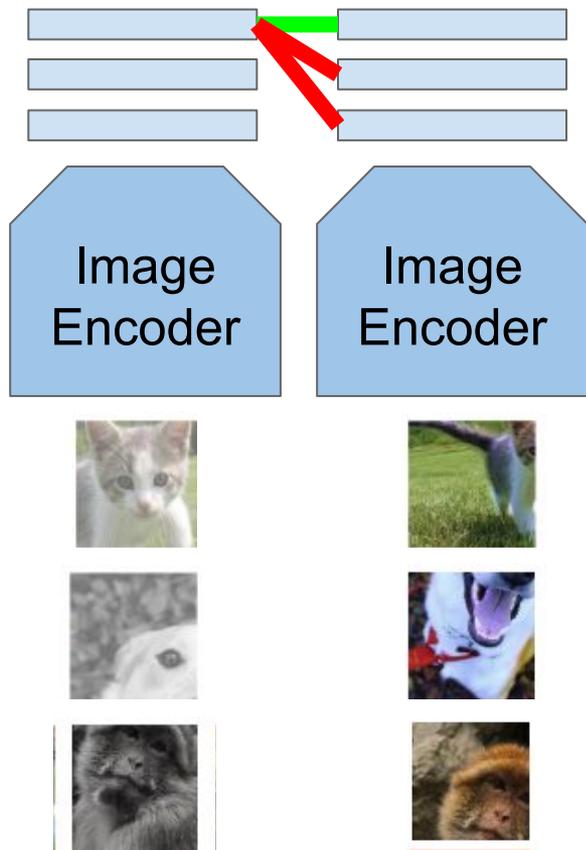


1. "A cute fluffy cat"
2. "My favorite dog is a golden retriever"

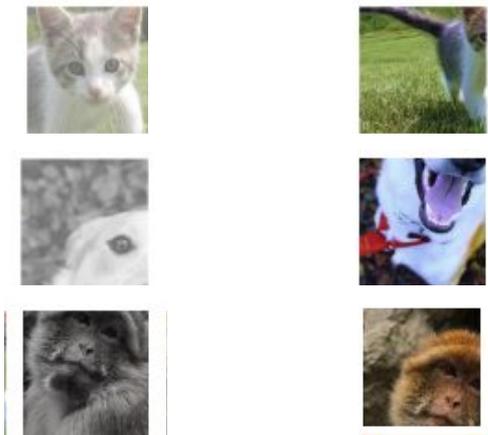
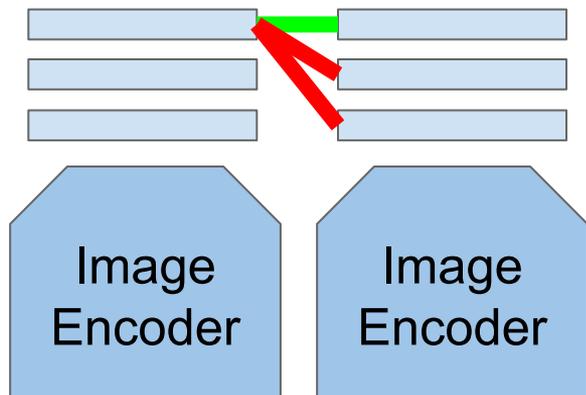
What if this representation space could also embed **sentences/phrases**?



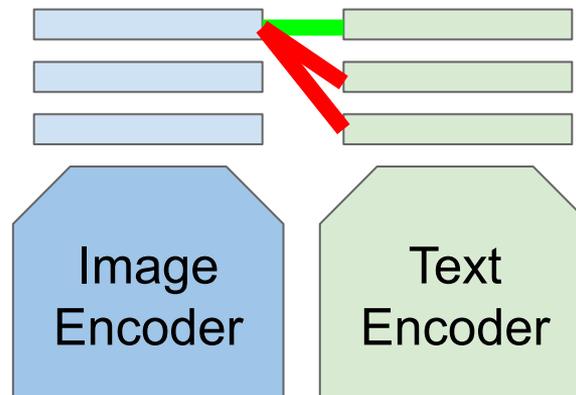
SimClr



SimClr

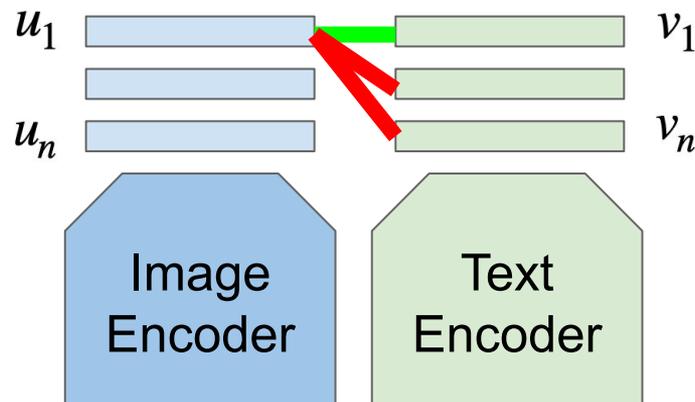


CLIP



CLIP is trained with the same contrastive 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”



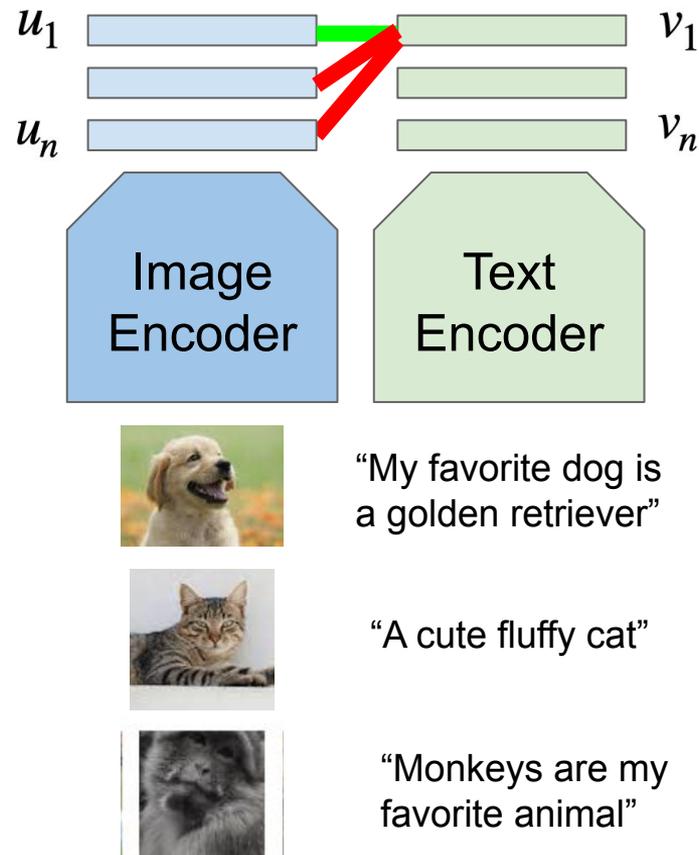
“A cute fluffy cat”



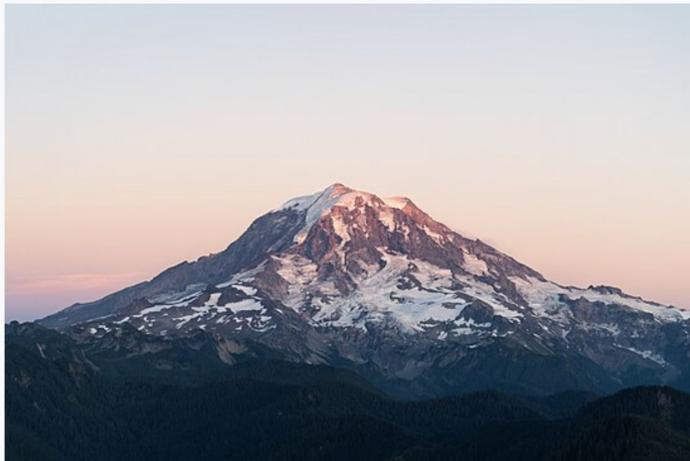
“Monkeys are my favorite animal”

CLIP Training Objective

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



Lots of image-text data can be found online



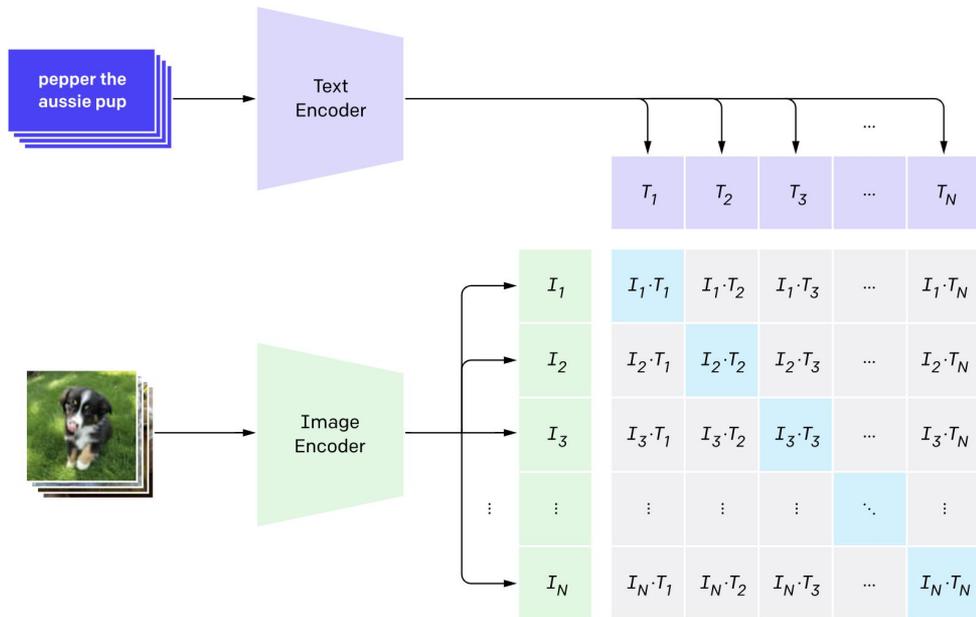
Mount Rainier's northwestern slope viewed aerially
just before sunset on September 6, 2020

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

https://en.wikipedia.org/wiki/Mount_Rainier

CLIP Training Objective

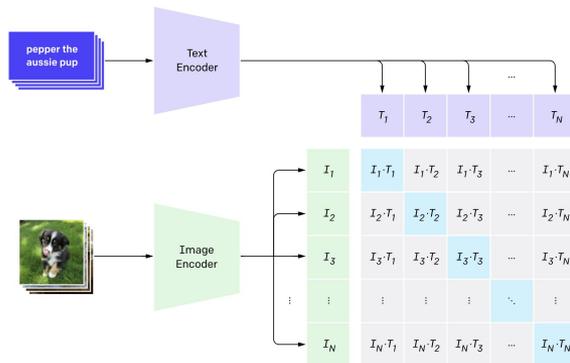
1. Contrastive pre-training



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

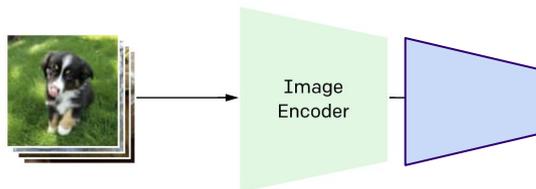
Using pre-trained models out of the box

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



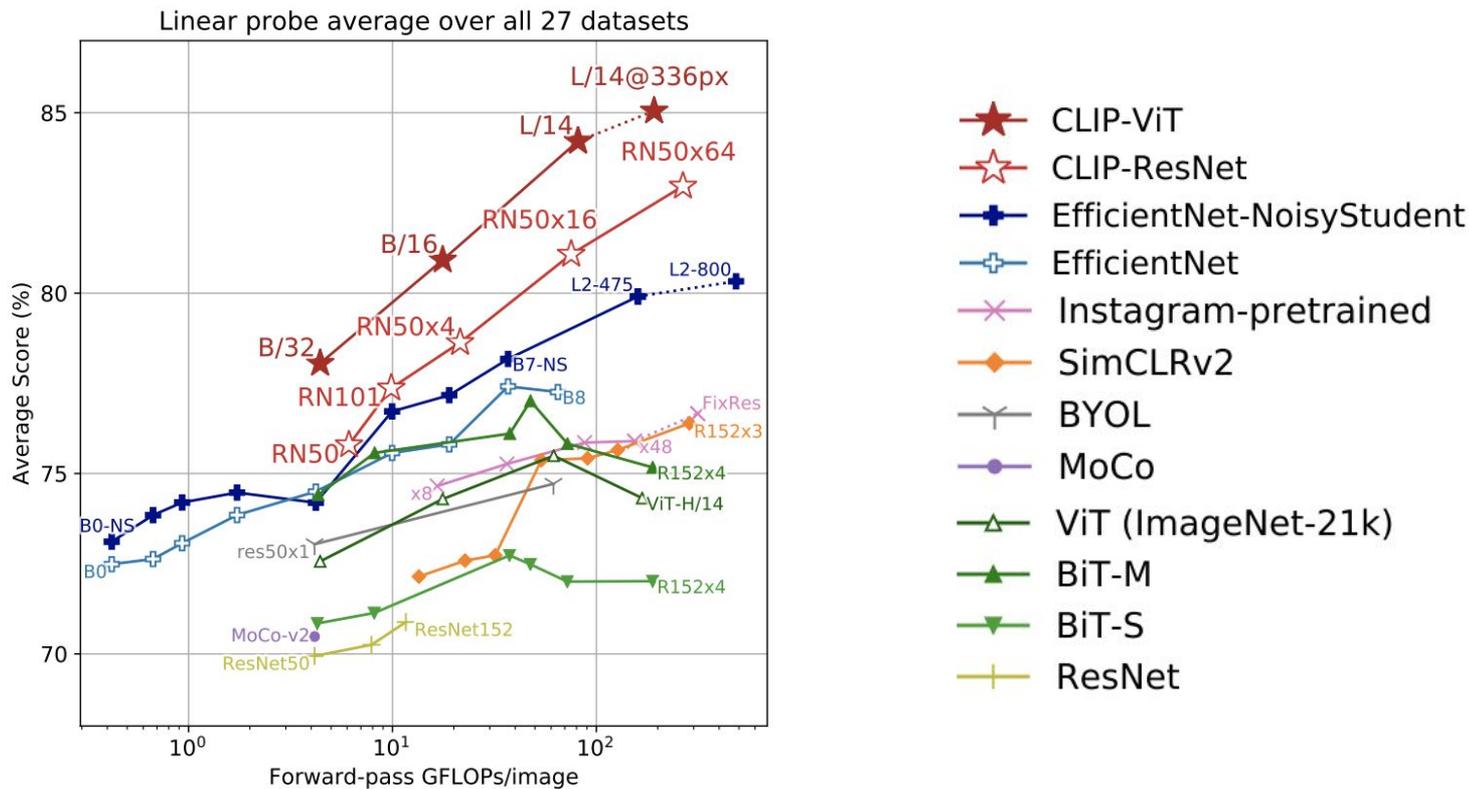
Pre-training tasks:
Contrastive Objective

Step 2: Transfer encoder to downstream tasks via linear classifiers



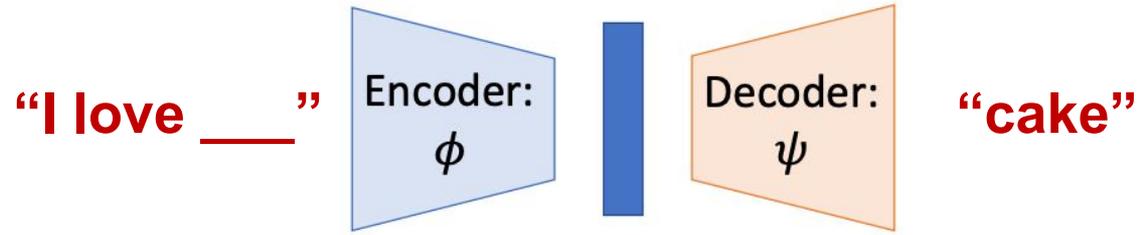
Downstream tasks:
Image classification,
object detection,
semantic segmentation

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

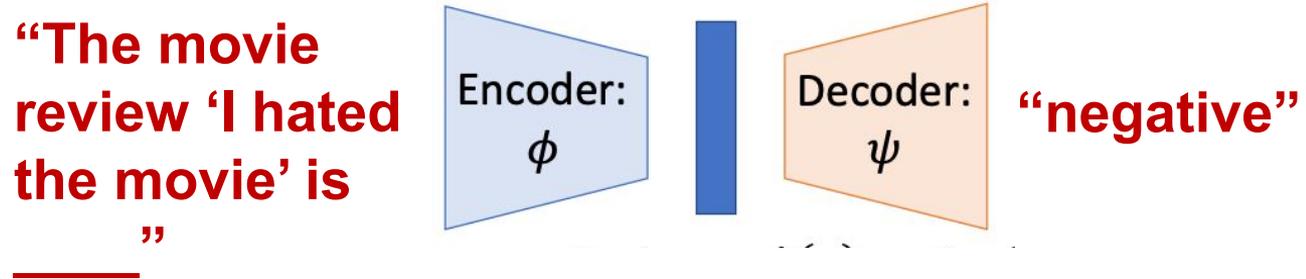


Big difference with language models: We can use LLMs **zero-shot** for new downstream tasks

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

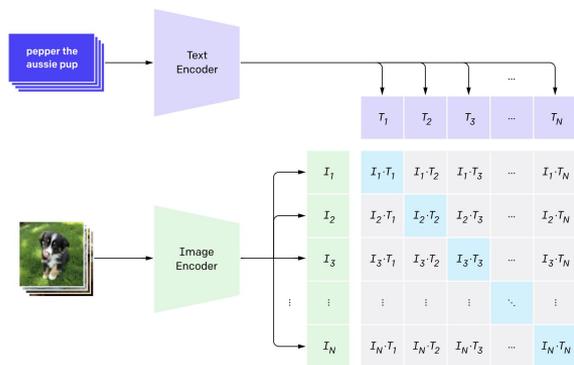


Step 2: Use the model out of the box in a creative way!



But how do we use pre-trained **vision-language** models in a **zero-shot** manner?

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

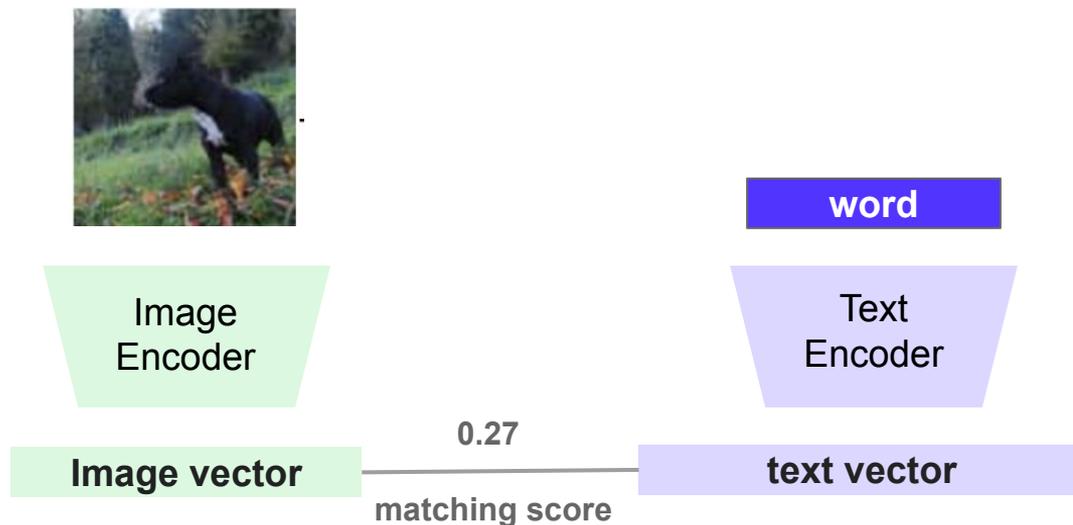


Pre-training tasks:
Contrastive Objective

Step 2: Use the model out of the box in a creative way!

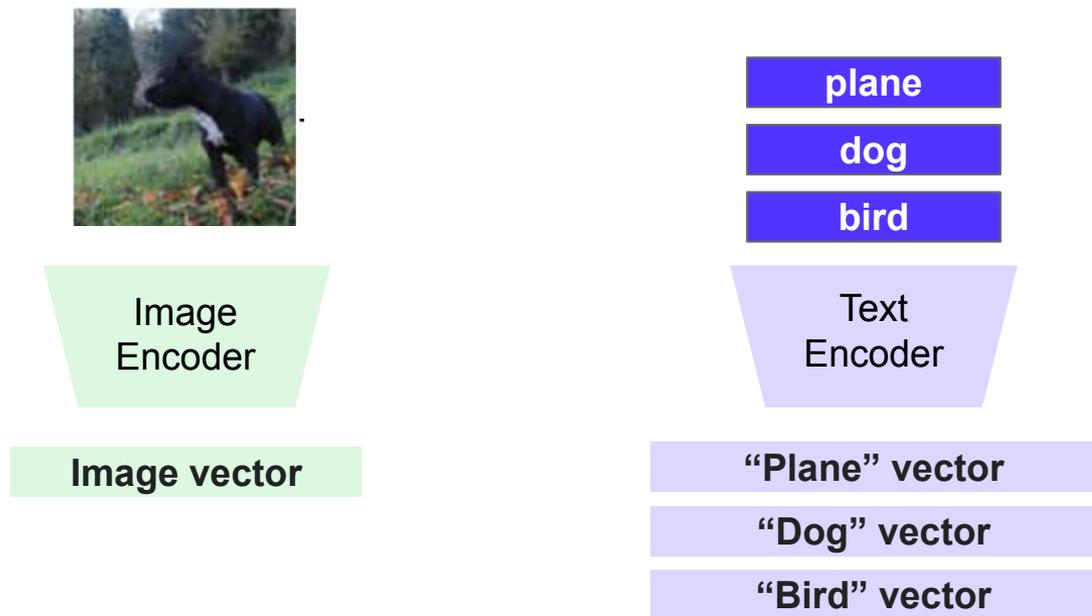
**Out of the box classification
(No fine-tuning)**

Clever trick: we can create a classifier using the text encoder!



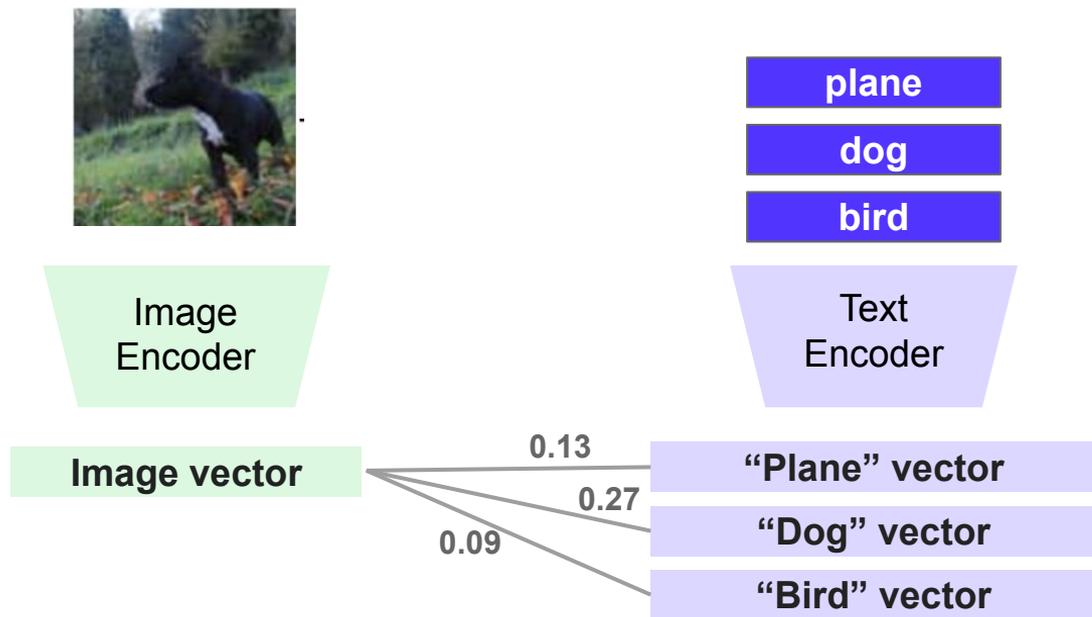
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

Create a vector representation for *each* category!



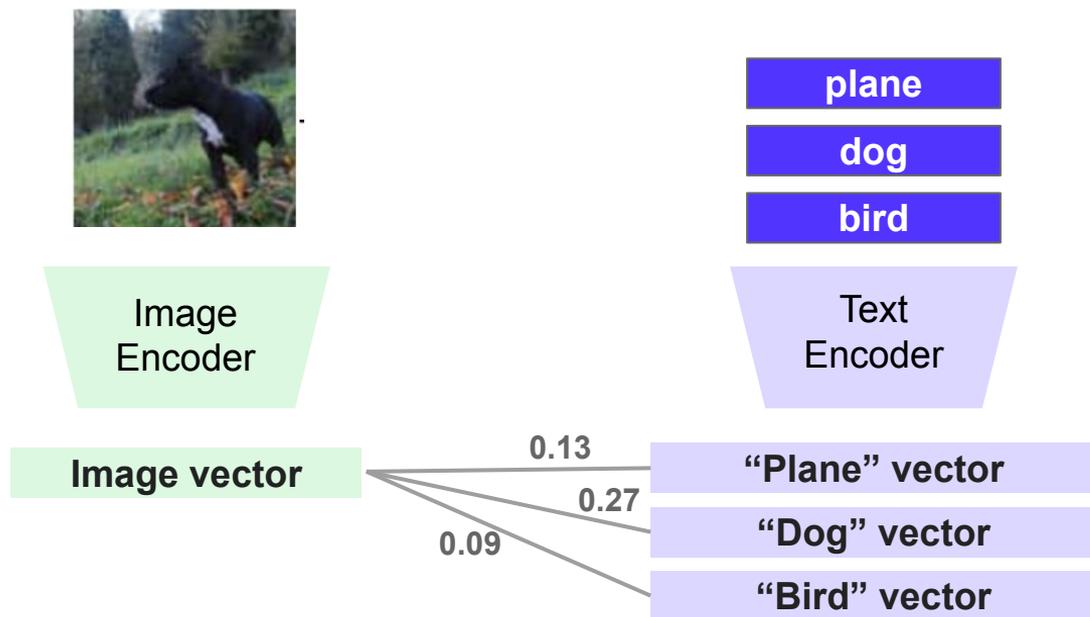
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

Match a new image to the most similar vector



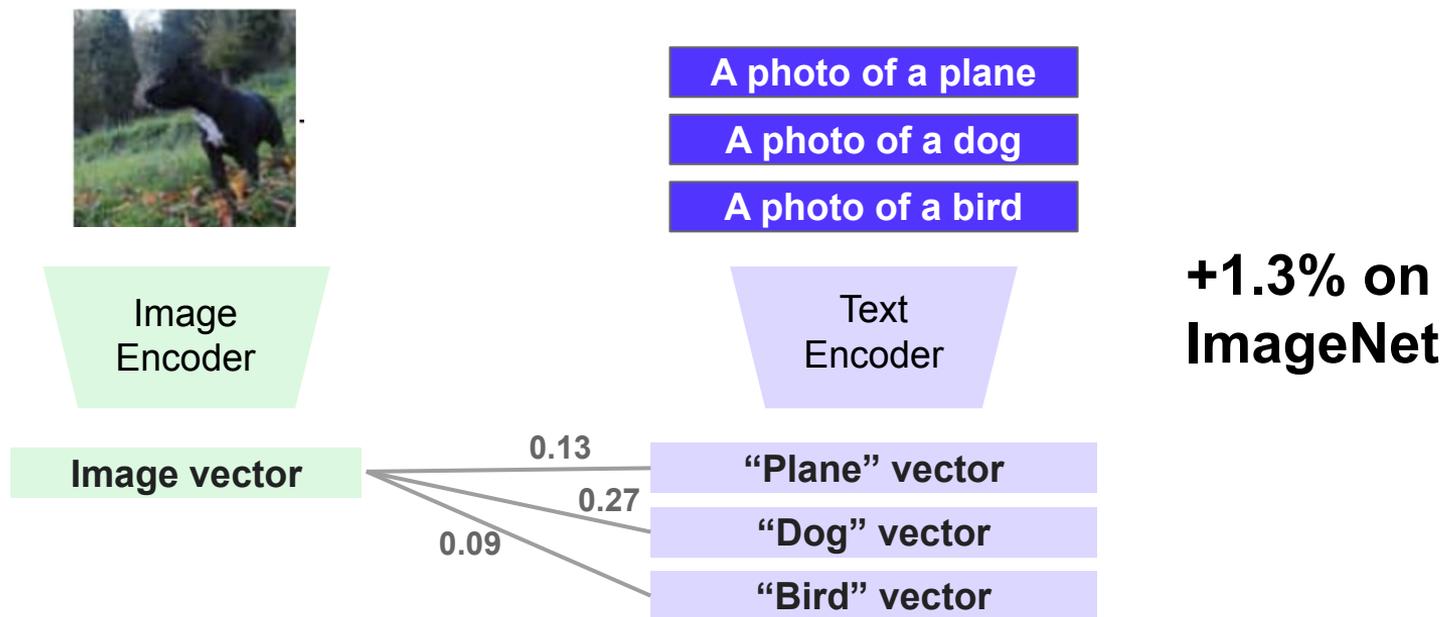
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

You can think of this as a 1-NN algorithm with the vectors as the training data



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

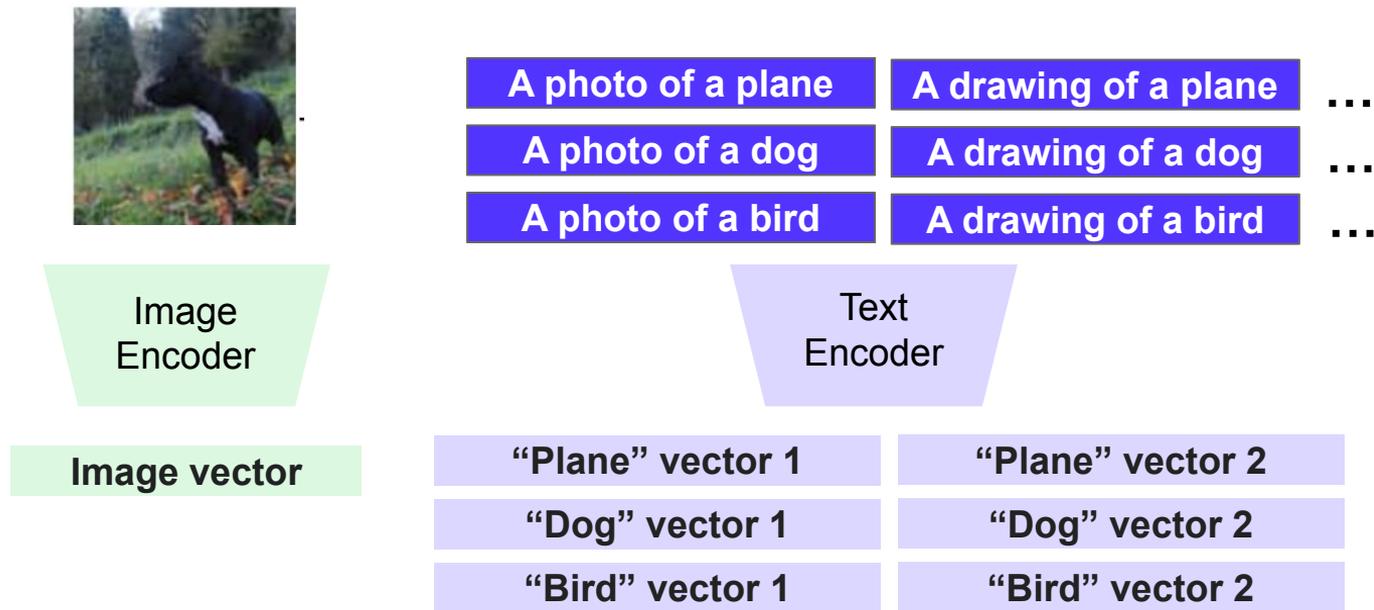
Since CLIP was trained with phrases, you can improve performance by using a phrase “A photo of a [category]”



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

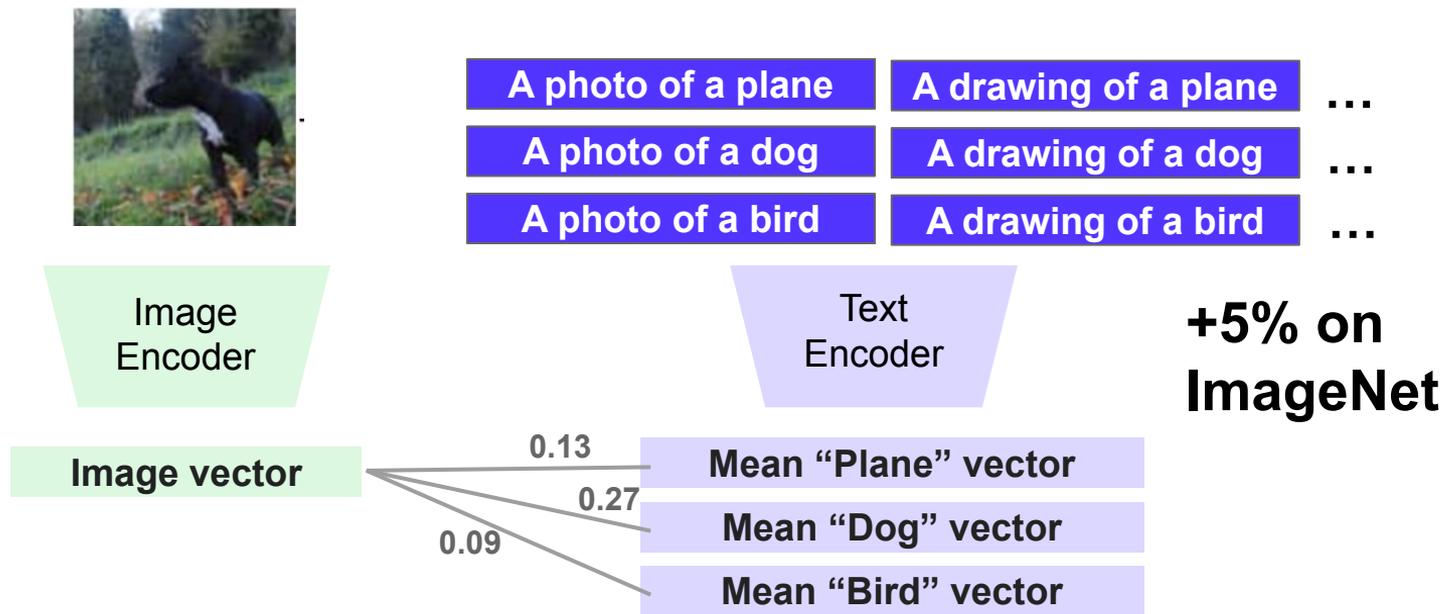
A single phrase might be too biased.

Solution: Use multiple phrases



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

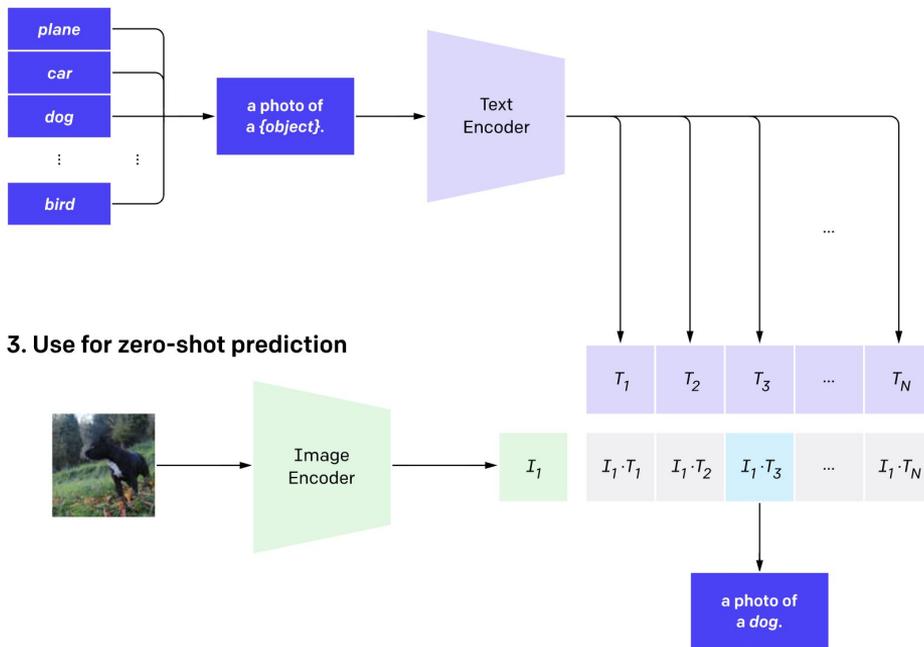
Use the average vector across phrases as the representation for each category



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

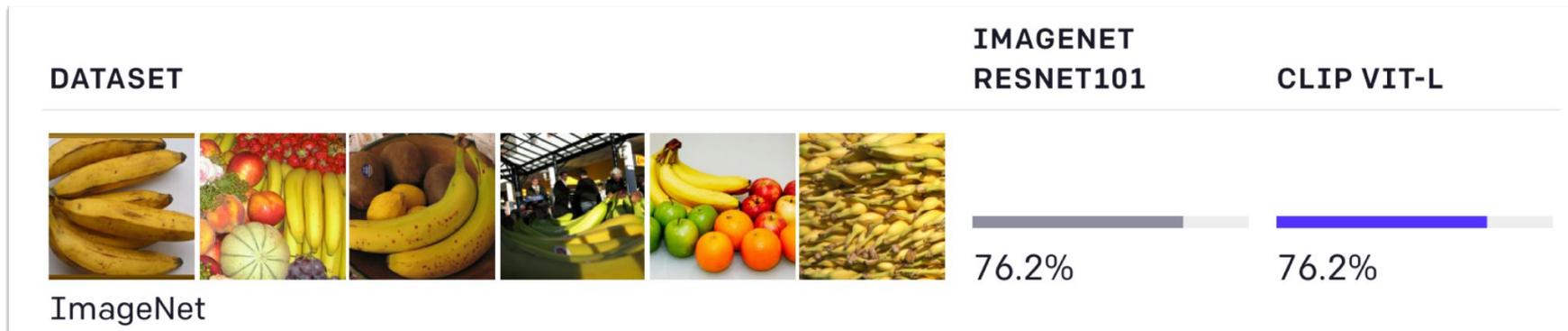
That's it! Now, you can use CLIP as a foundation model for image classification for any dataset

2. Create dataset classifier from label text



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

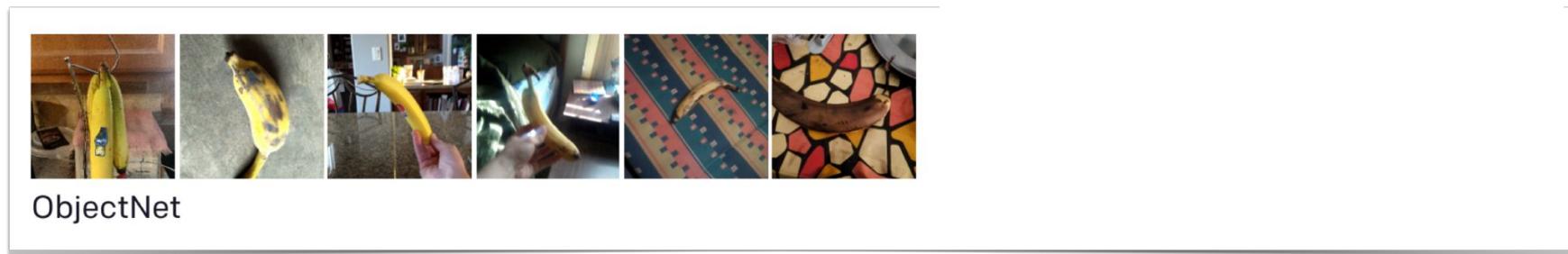
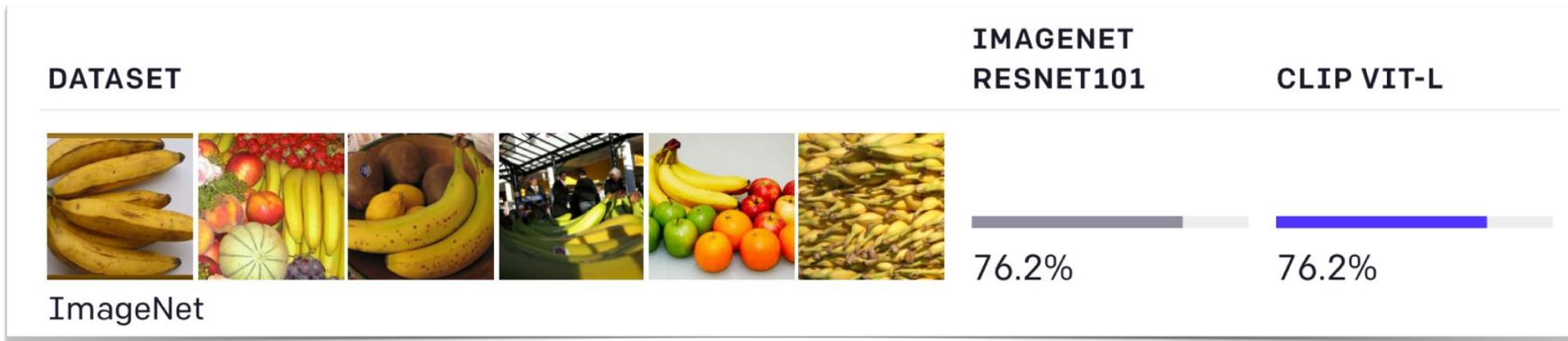
Exciting result after training on 400M image-text pairs



Matches the accuracy of of ResNet 101 that has been trained on ImageNet, except CLIP was trained with no human labels at all!

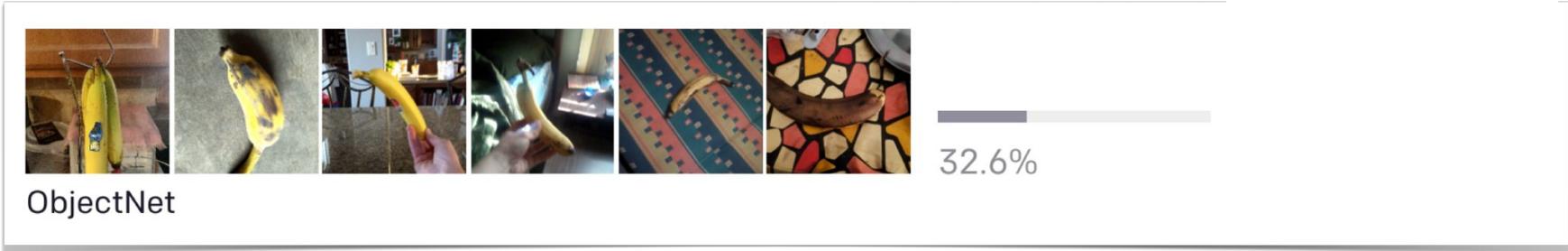
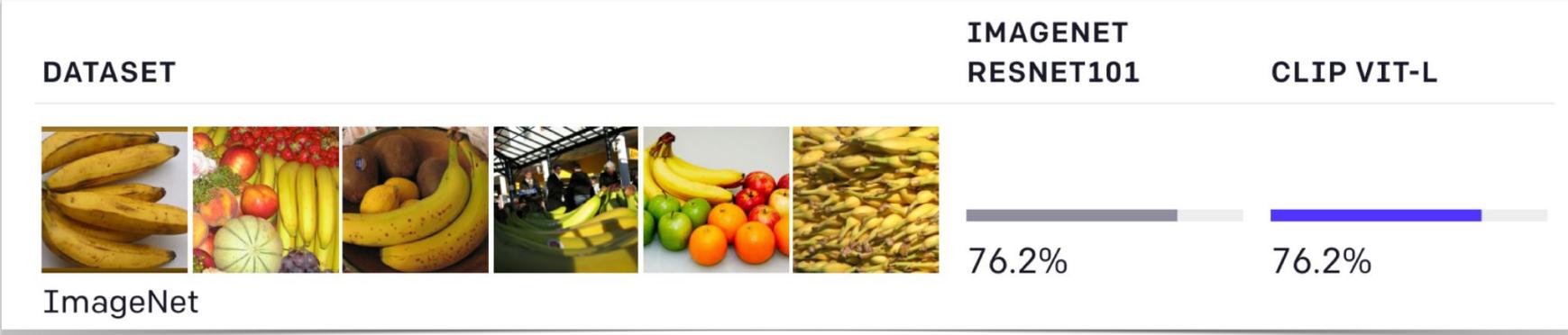
Radford et al “Learning Transferable Visual Models From Natural Language Supervision”

Here's where things get even more exciting



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

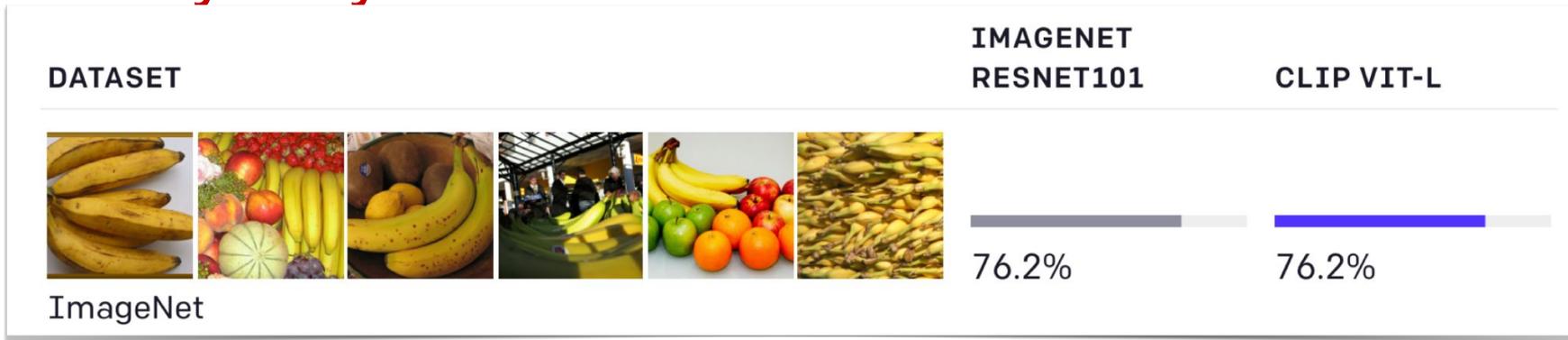
Training on ImageNet doesn't generalize to other datasets. ObjectNet contains the same categories but in weird viewpoints



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

But CLIP zero-shot does so well!

Q. Why do you think that is?



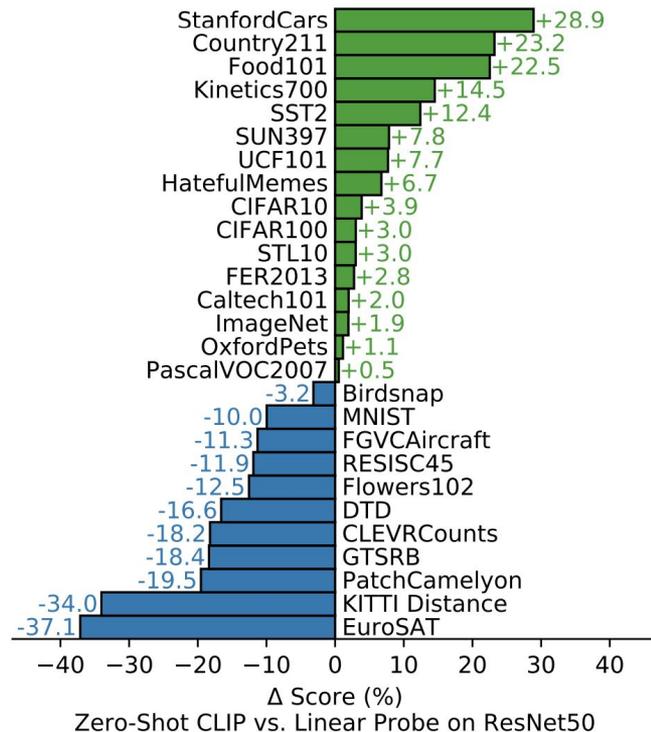
Radford et al "Learning Transferable Visual Models From Natural Language Supervision"

CLIP performance is great also on graphic images, sketches, 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”

Difference in performance between linear probe vs zero-shot



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

Why does CLIP perform so well?

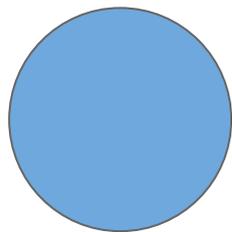
How can no labels beat labels??

Why does CLIP perform so well?

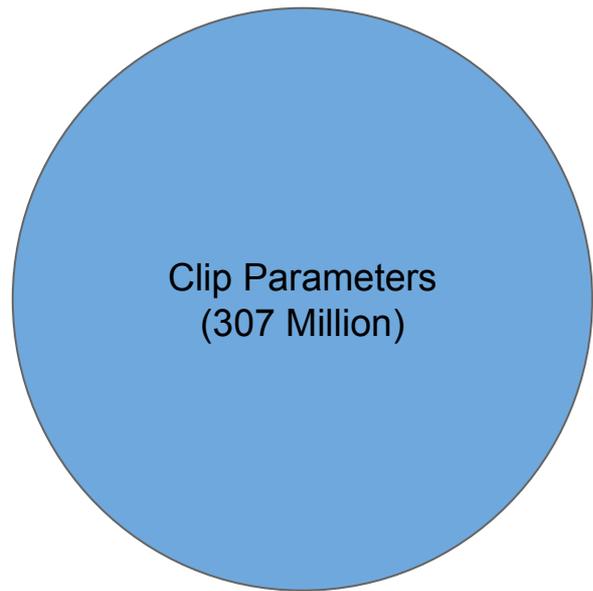
How can no labels beat labels??

Scale!

CLIP scaled up the model parameters with the transformer architecture



ImageNet ResNet Parameters
(44.5 Million)

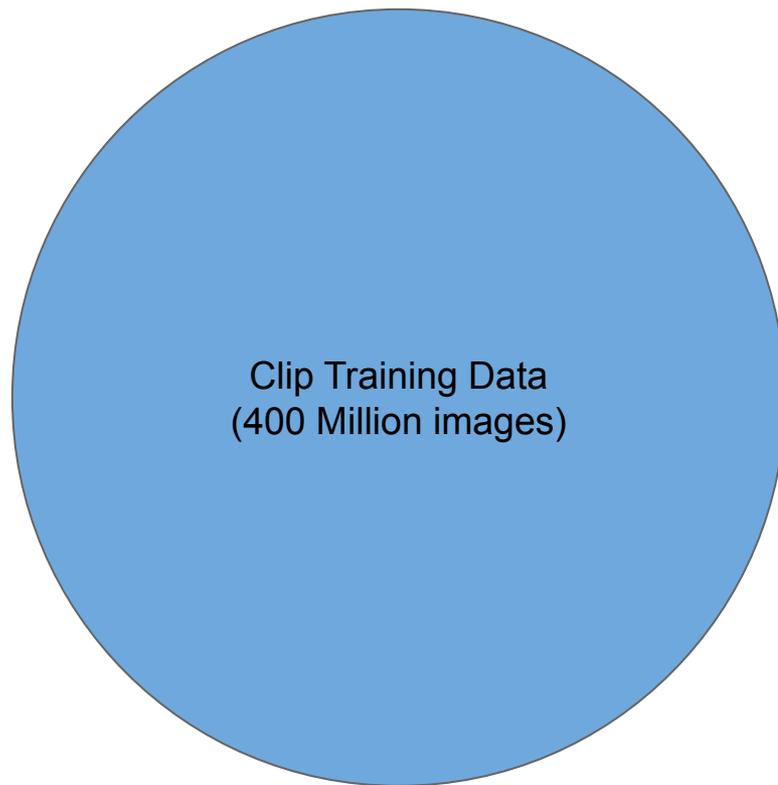


Clip Parameters
(307 Million)

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

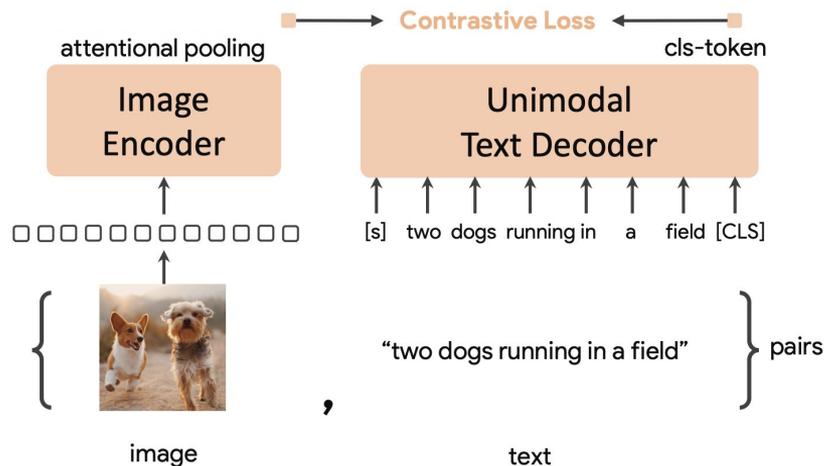


ImageNet ResNet Training Data
(1.28 Million)



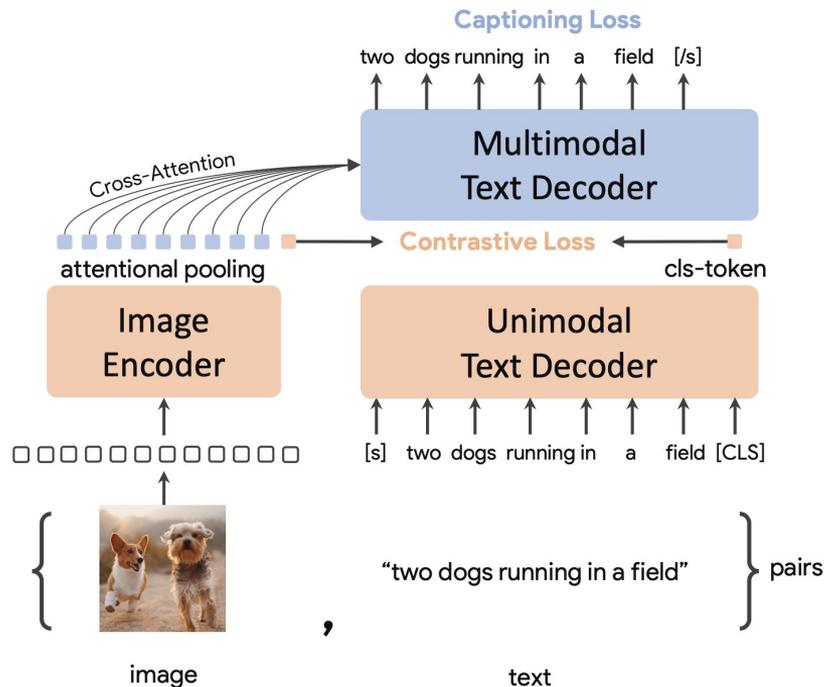
Clip Training Data
(400 Million images)

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

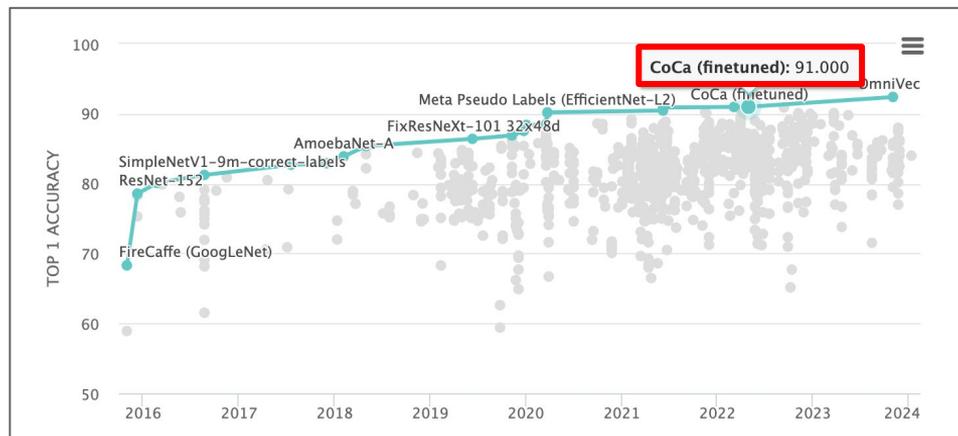
CoCa: Contrastive Captioners are Image-Text Foundation Models

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
CoCa	86.3	90.2	96.5	80.7	77.6	82.7	85.7

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
CoCa (finetuned)	91.0



Flamingo

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

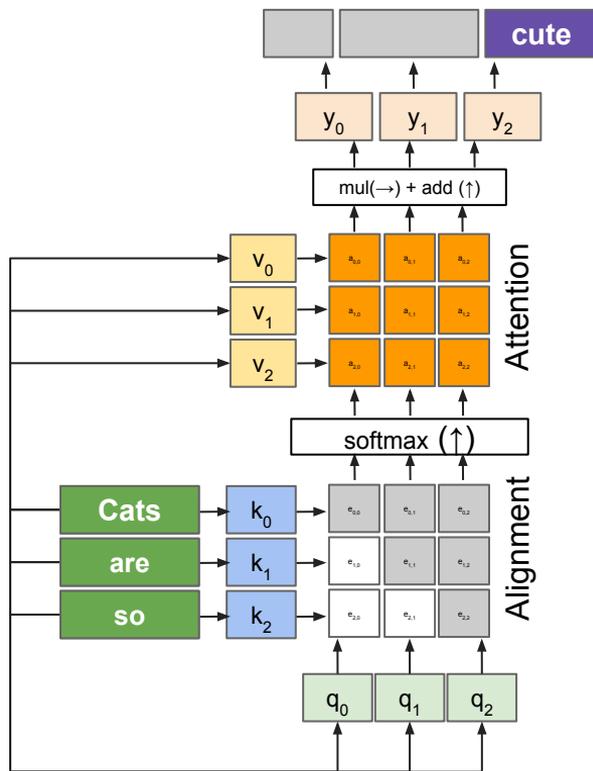
(only can output similarity scores between image and text)

Flamingo

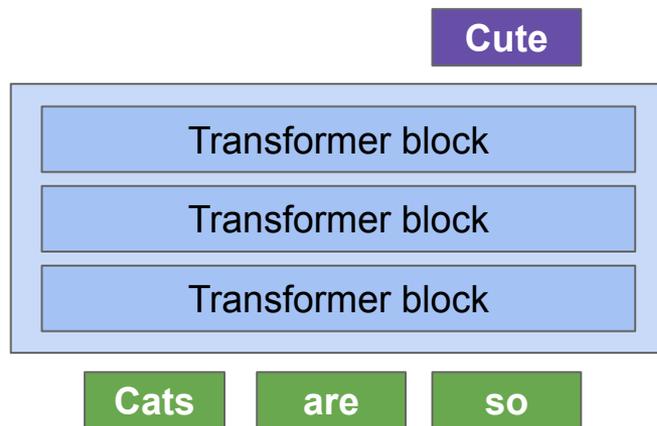
Motivation: Language models which do next token prediction can be applied to a wide variety of tasks at inference (Math, sentiment analysis, symbolic reasoning)

Can we build something like GPT but can accept images and text as input, and then output text?

Flamingo

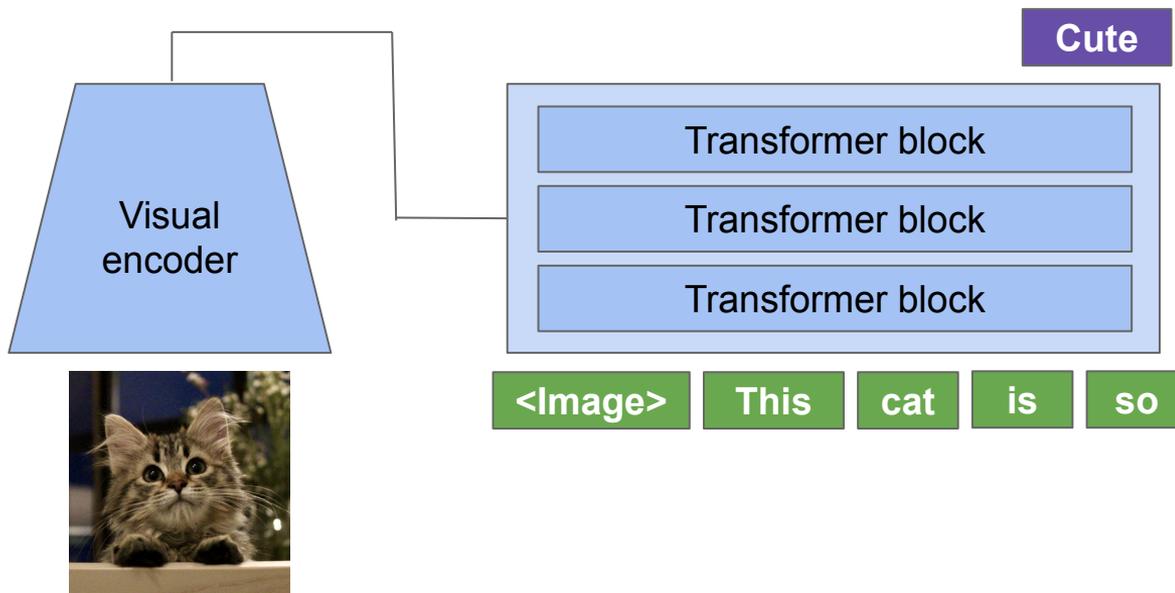


Flamingo

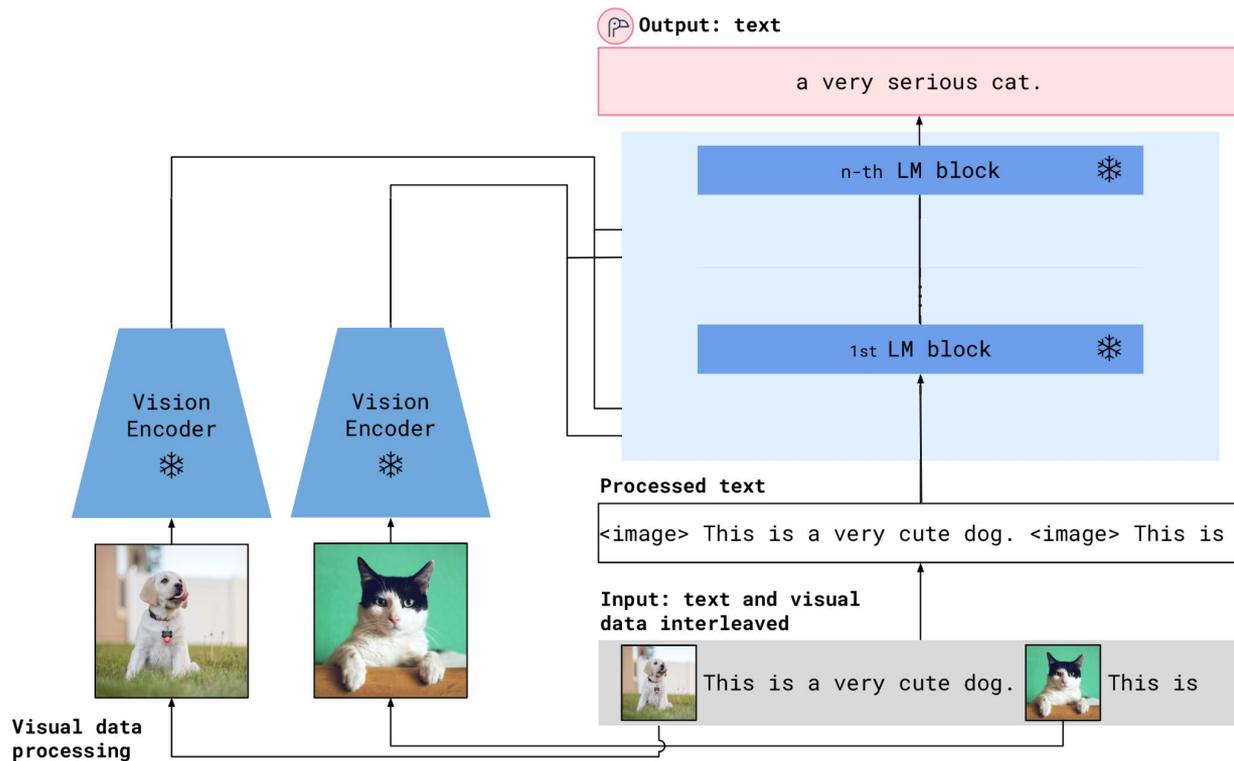


Flamingo

What kind of model is this? (think types of LLMs)



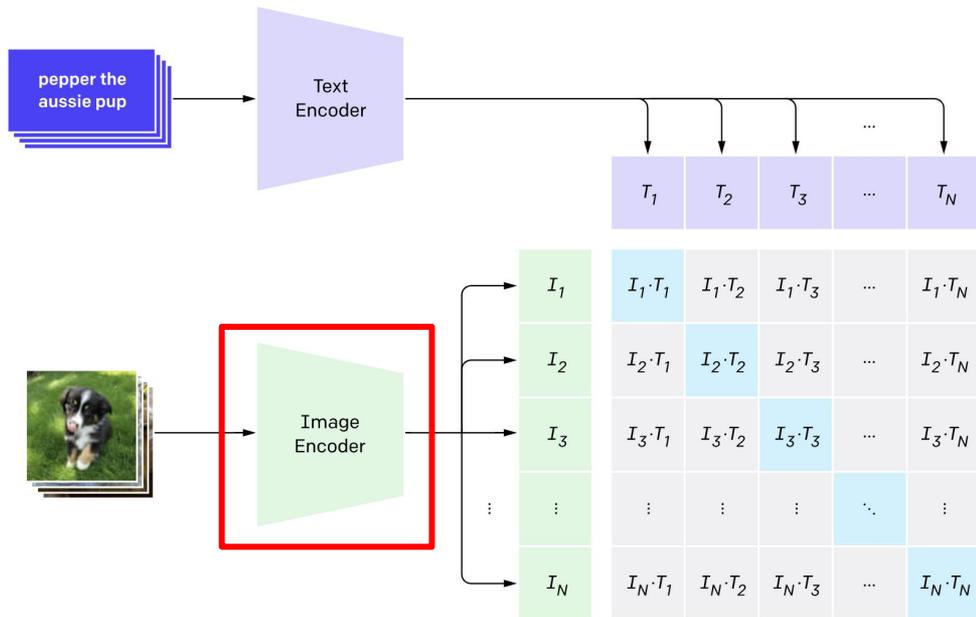
Pre-trained parts of Flamingo



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

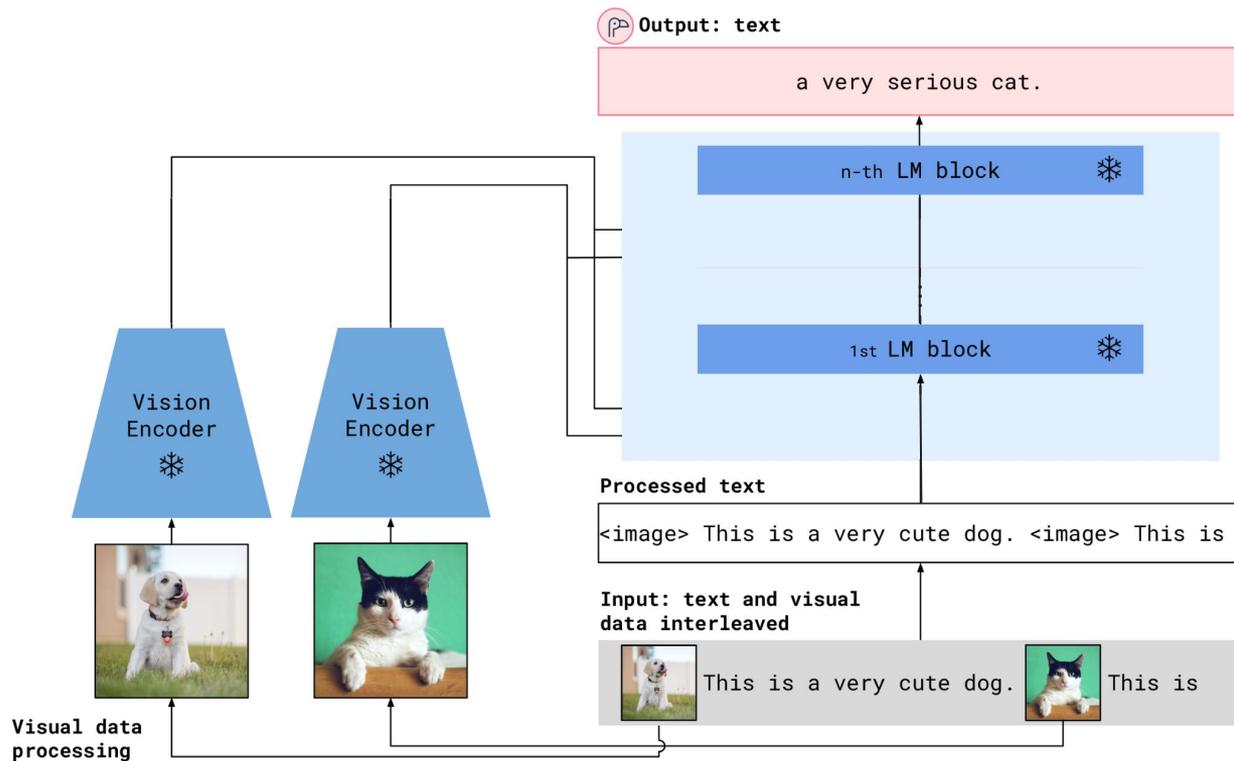
CLIP Training Objective

1. Contrastive pre-training



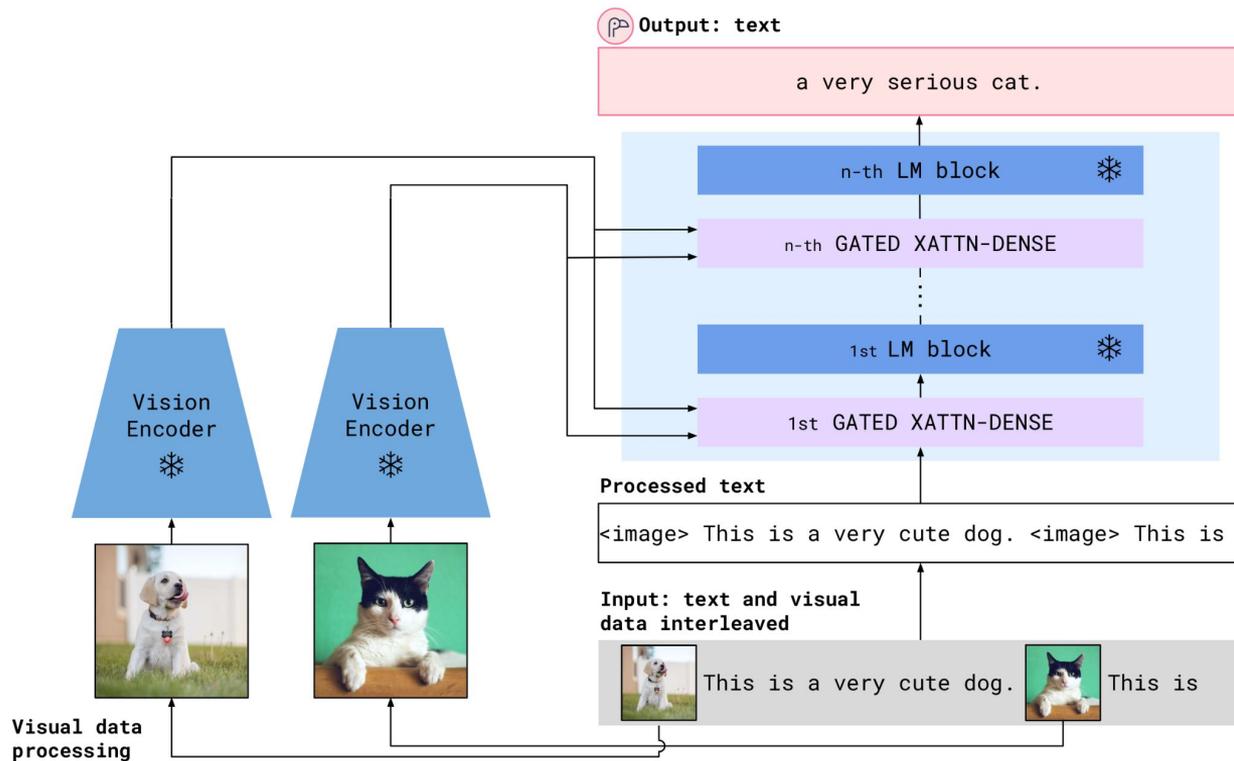
At the end of training, you have a model that will give you a similarity score between an image and a text

Pre-trained parts of Flamingo



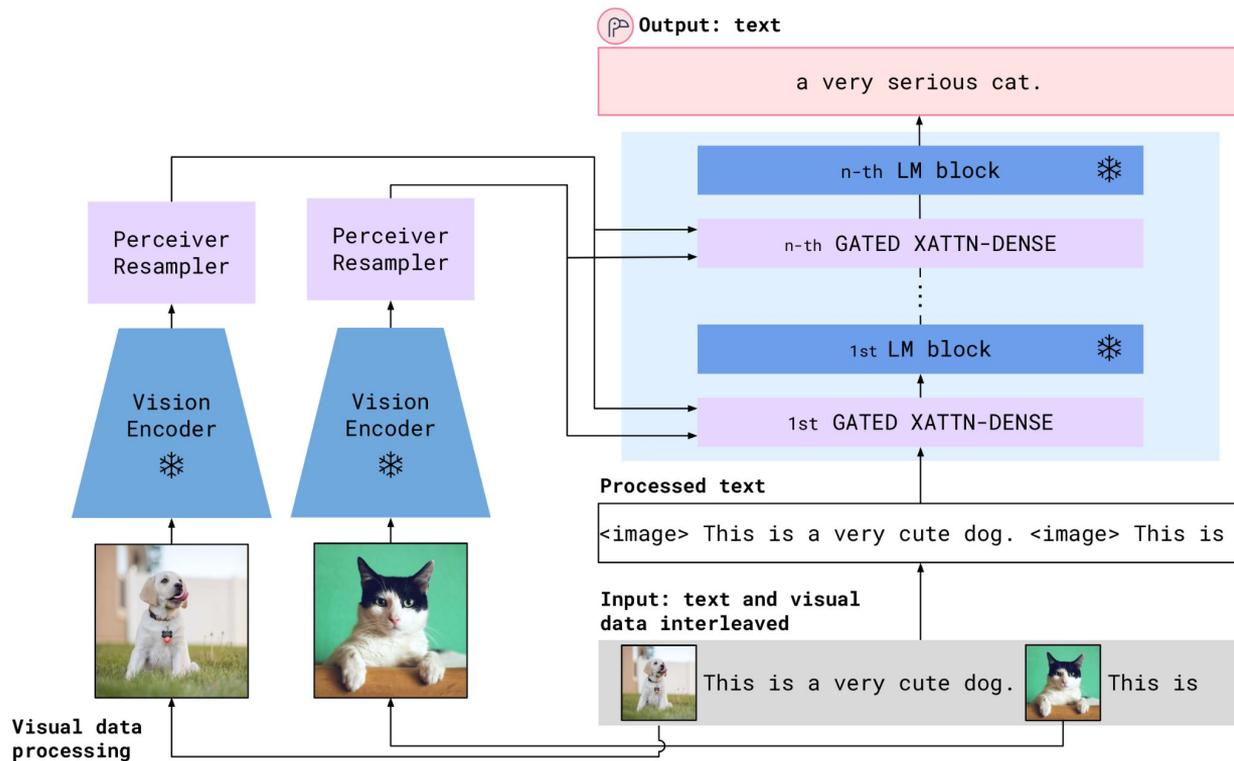
Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

Learned parts of Flamingo



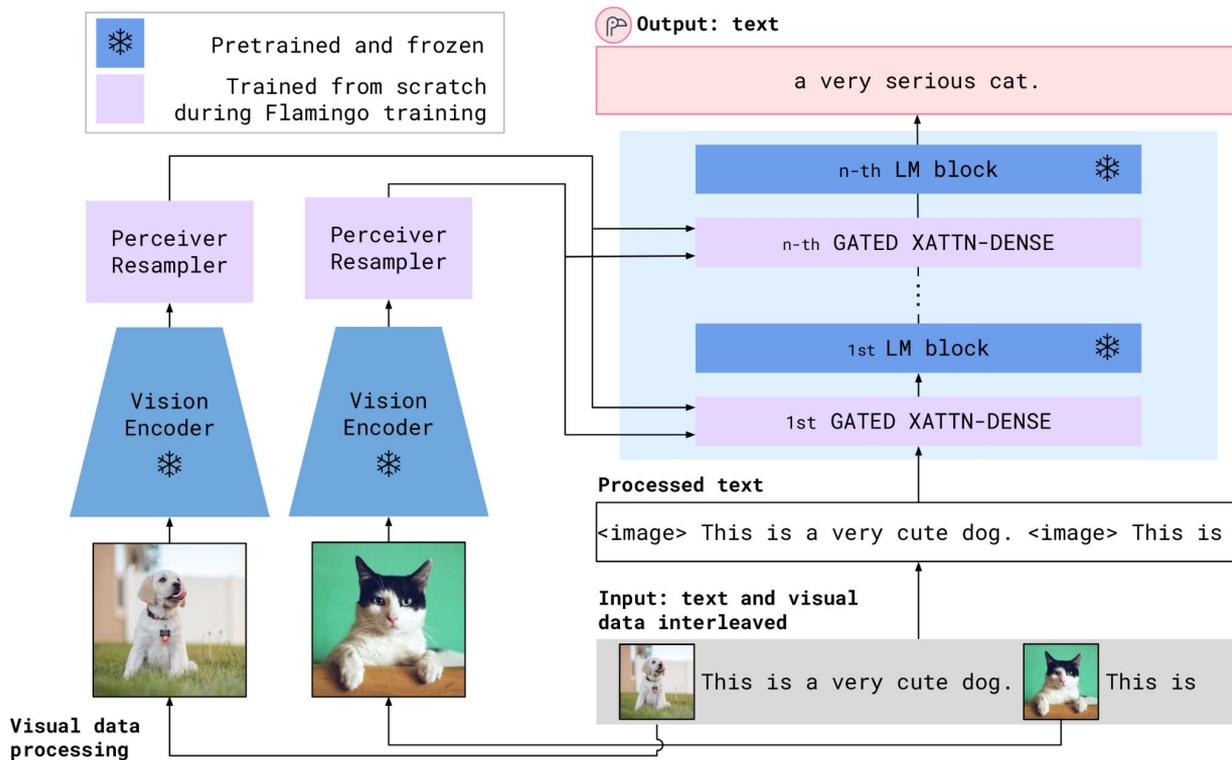
Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

Learned parts of Flamingo



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

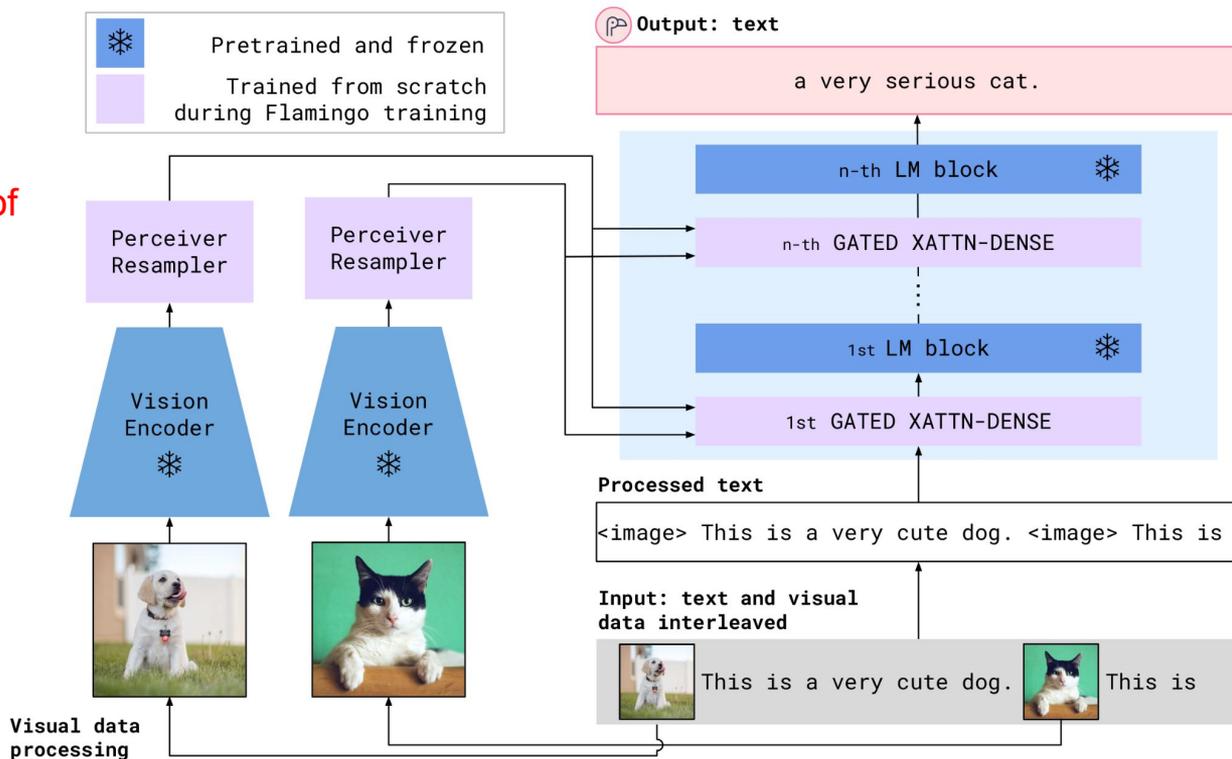
Flamingo full architecture



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

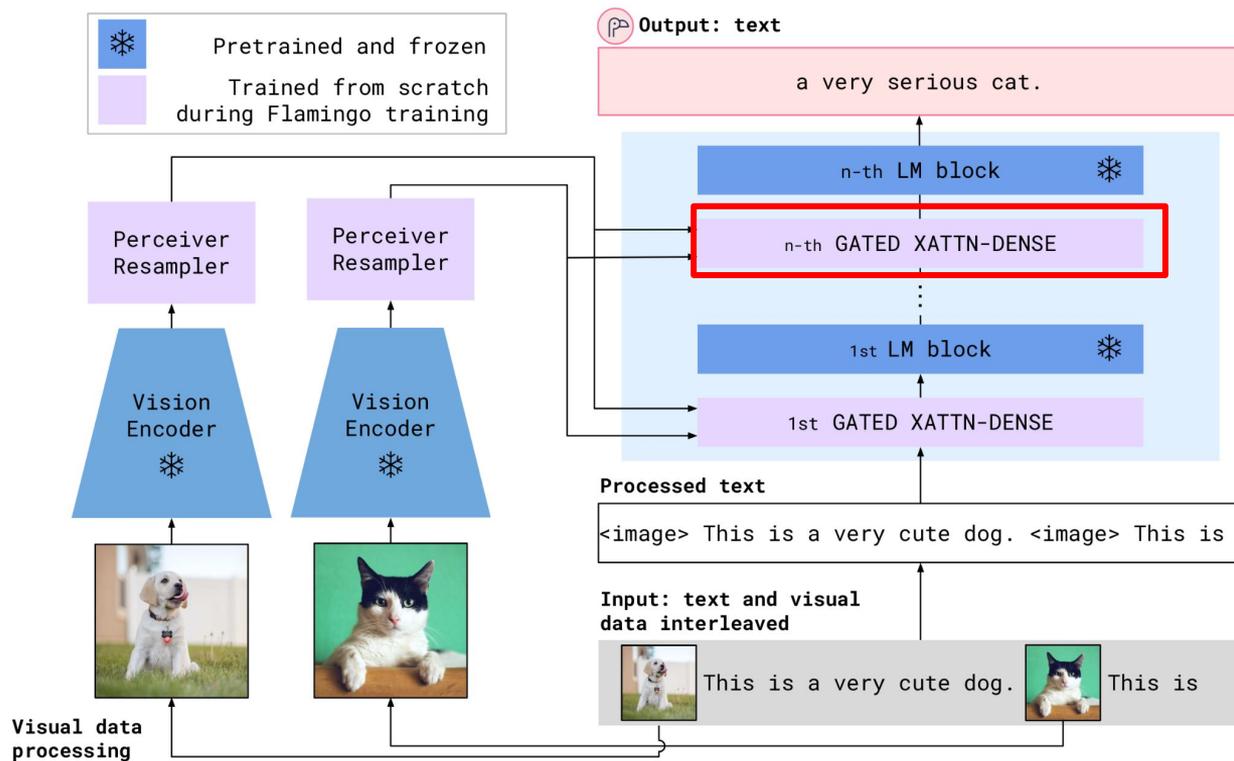
Flamingo full architecture

Learned method of down-sampling image/video representations



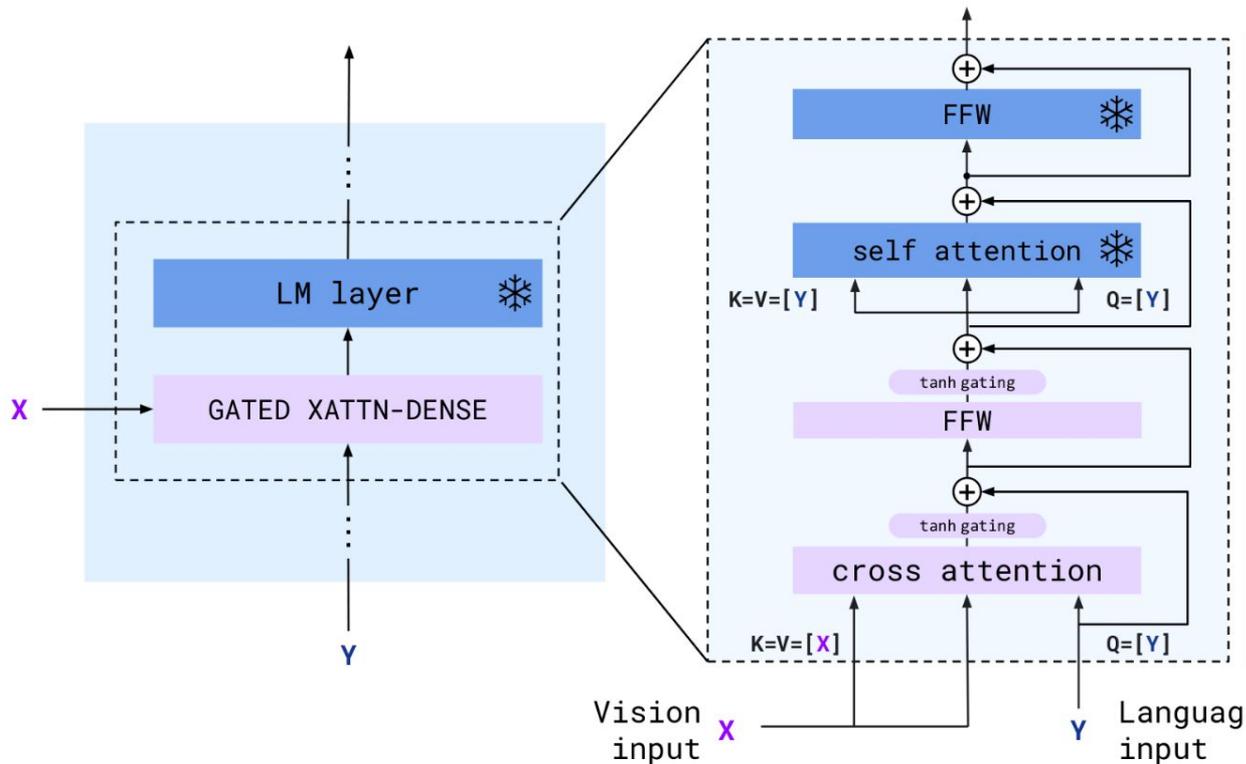
Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

Flamingo full architecture



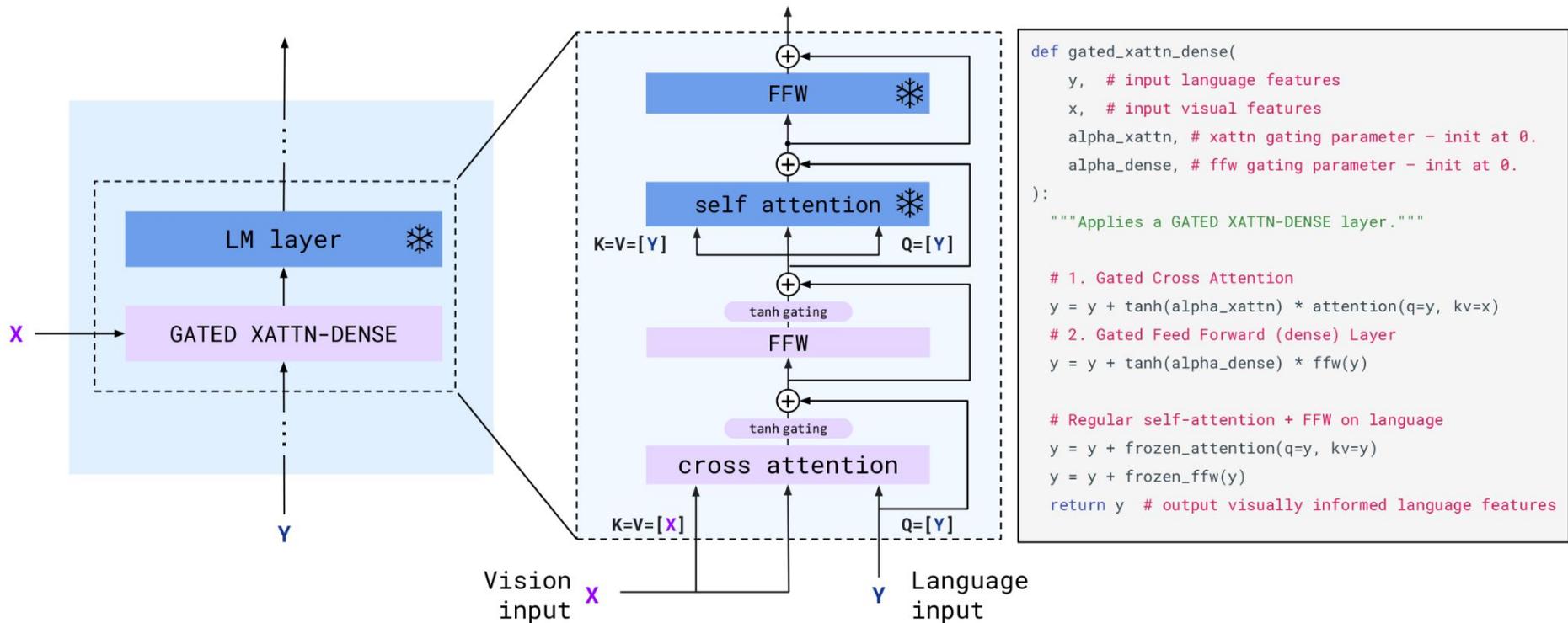
Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

Flamingo gated cross-attention



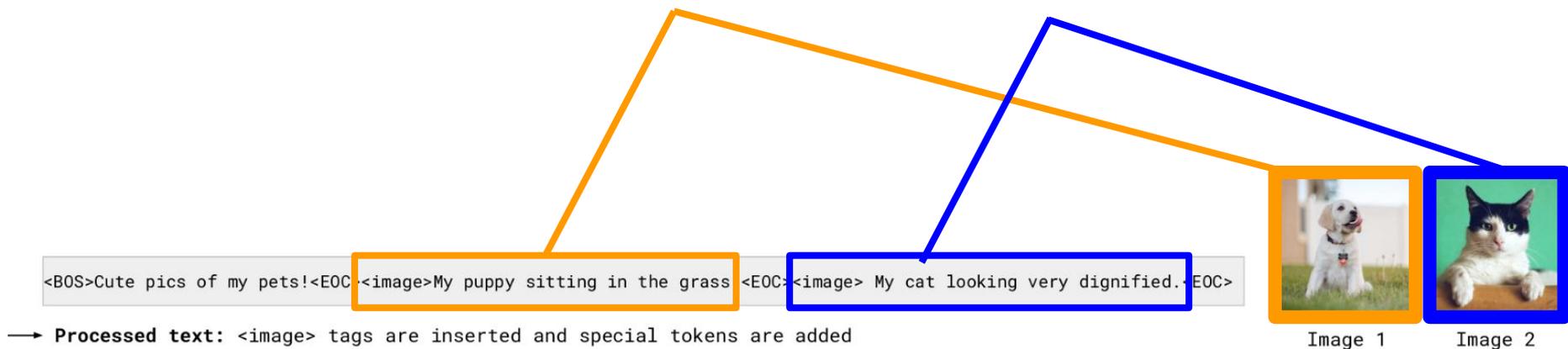
Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

Flamingo gated cross-attention



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

Flamingo masked attention



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

Flamingo masked attention

Φ 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2
Y <BOS> Cute pics of my pets!<EOC><image>My puppy sitting in the grass. <EOC><image>My cat looking very dignified.<EOC>

tokenization

<BOS>Cute pics of my pets!<EOC><image>My puppy sitting in the grass.<EOC><image> My cat looking very dignified.<EOC>

→ **Processed text:** <image> tags are inserted and special tokens are added

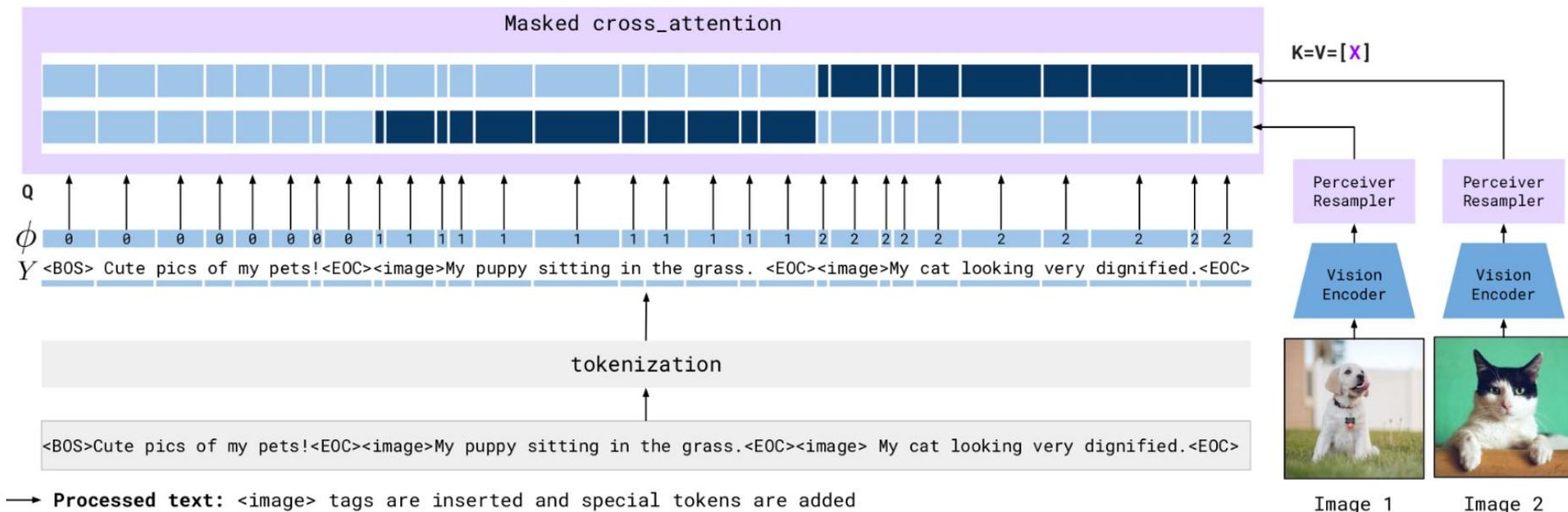


Image 1

Image 2

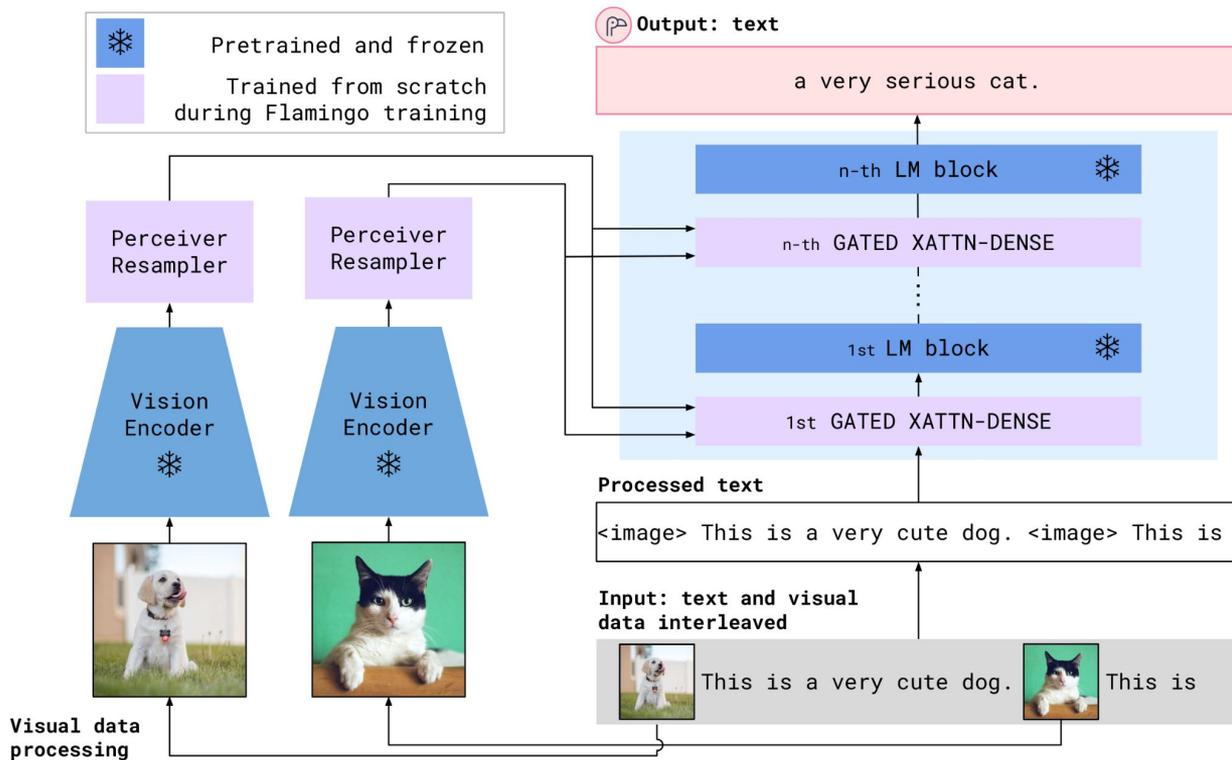
Alayrac et al "Flamingo: a Visual Language Model for Few-Shot Learning. 2022.

Flamingo masked attention



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

Flamingo full architecture



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

Flamingo results



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?

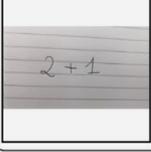
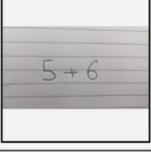
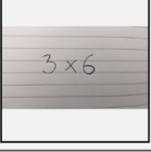
It looks like it's handwritten.

What color is the sticker?

It's white.

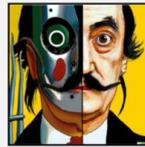
Flamingo results

What is this type of learning called?

Input Prompt				Completion	
	This is a chinchilla. They are mainly found in Chile.		This is a shiba. They are very popular in Japan.		This is a flamingo. They are found in the Caribbean and South America.
	What is the title of this painting? Answer: The Hallucinogenic Toreador.		Where is this painting displayed? Answer: Louvres Museum, Paris.		What is the name of the city where this was painted? Answer: Arles.
	Output: "Underground"		Output: "Congress"		Output: "Soulomes"
	2+1=3		5+6=11		3x6=18

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

Flamingo results

	Output: A propaganda poster depicting a cat dressed as French emperor Napoleon holding a piece of cheese.		Output: A pink room with a flamingo pool float.		Output:	→	A portrait of Salvador Dali with a robot head.
	Les sanglots longs des violons de l'automne blessent mon coeur d'une langueur monotone.		Pour qui sont ces serpents qui sifflent sur vos têtes?			→	Je suis un cœur qui bat pour vous.
	pandas: 3		dogs: 2			→	giraffes: 4
I like reading		, my favourite play is Hamlet. I also like		, my favorite book is		→	Dreams from my Father.
					What happens to the man after hitting the ball? Answer:	→	he falls down.

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

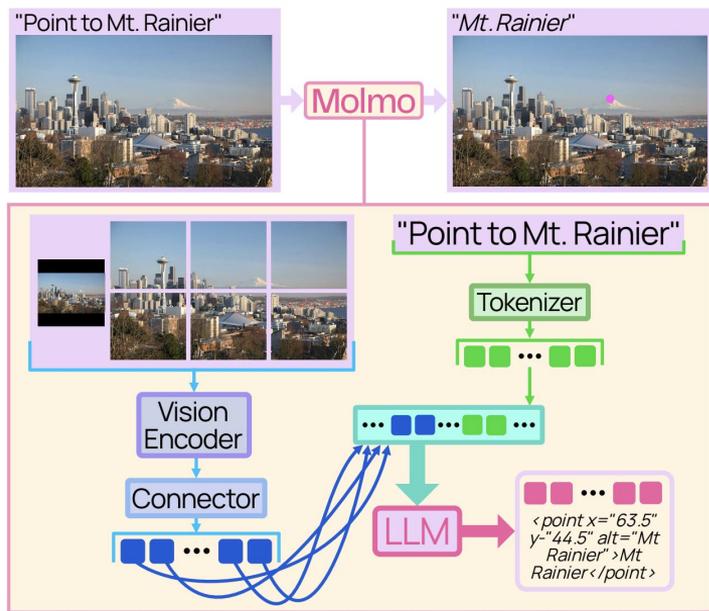
Results: zero & few shot

Method	FT	Shot	OKVQA	VQAv2	COCO	MSVDQA	VATEX	VizWiz	Flick30K	MSRVTTQA	IVQA	YouCook2	STAR	VisDial	TextVQA	NextQA	HatefulMemes	RareAct
Zero/Few shot SOTA	✗		[39] 43.3	[124] 38.2	[134] 32.2	[64] 35.2	-	-	-	[64] 19.2	[145] 12.2	-	[153] 39.4	[87] 11.6	-	-	[94] 66.1	[94] 40.7
		(X)	(16)	(4)	(0)	(0)				(0)	(0)		(0)	(0)			(0)	(0)
Flamingo-3B	✗	0	41.2	49.2	73.0	27.5	40.1	28.9	60.6	11.0	32.7	55.8	39.6	46.1	30.1	21.3	53.7	58.4
	✗	4	43.3	53.2	85.0	33.0	50.0	34.0	72.0	14.9	35.7	64.6	41.3	47.3	32.7	22.4	53.6	-
	✗	8	44.6	55.4	90.6	37.0	54.5	38.4	71.7	19.6	36.8	68.0	40.6	47.6	32.4	23.9	54.7	-
	✗	16	45.6	56.7	95.4	40.2	57.1	43.3	73.4	23.4	37.4	73.2	40.1	47.5	31.8	25.2	55.3	-
	✗	32	45.9	57.1	99.0	42.6	59.2	45.5	71.2	25.6	37.7	76.7	41.6	OOC	30.6	26.1	56.3	-
Flamingo-9B	✗	0	44.7	51.8	79.4	30.2	39.5	28.8	61.5	13.7	35.2	55.0	41.8	48.0	31.8	23.0	57.0	57.9
	✗	4	49.3	56.3	93.1	36.2	51.7	34.9	72.6	18.2	37.7	70.8	42.8	50.4	33.6	24.7	62.7	-
	✗	8	50.0	58.0	99.0	40.8	55.2	39.4	73.4	23.9	40.0	75.0	43.4	51.2	33.6	25.8	63.9	-
	✗	16	50.8	59.4	102.2	44.5	58.5	43.0	72.7	27.6	41.5	77.2	42.4	51.3	33.5	27.6	64.5	-
	✗	32	51.0	60.4	106.3	47.2	57.4	44.0	72.8	29.4	40.7	77.3	41.2	OOC	32.6	28.4	63.5	-
Flamingo	✗	0	50.6	56.3	84.3	35.6	46.7	31.6	67.2	17.4	40.7	60.1	39.7	52.0	35.0	26.7	46.4	60.8
	✗	4	57.4	63.1	103.2	41.7	56.0	39.6	75.1	23.9	44.1	74.5	42.4	55.6	36.5	30.8	68.6	-
	✗	8	57.5	65.6	108.8	45.5	60.6	44.8	78.2	27.6	44.8	80.7	42.3	56.4	37.3	32.3	70.0	-
	✗	16	57.8	66.8	110.5	48.4	62.8	48.4	78.9	30.0	45.2	84.2	41.1	56.8	37.6	32.9	70.0	-
	✗	32	57.8	67.6	113.8	52.3	65.1	49.8	75.4	31.0	45.3	86.8	42.2	OOC	37.9	33.5	70.0	-
Pretrained FT SOTA	✓		54.4	80.2	143.3	47.9	76.3	57.2	67.4	46.8	35.4	138.7	36.7	75.2	54.7	25.2	75.4	-
		(X)	[39] (10K)	[150] (444K)	[134] (500K)	[32] (27K)	[165] (500K)	[70] (20K)	[162] (30K)	[57] (130K)	[145] (6K)	[142] (10K)	[138] (46K)	[87] (123K)	[147] (20K)	[139] (38K)	[60] (9K)	-

Results: zero & few shot

Method	FT	Shot	OKVQA	VQAv2	COCO	MSVDQA	VATEX	VizWiz	Flick30K	MSRVTTQA	iVQA	YouCook2	STAR	VisDial	TextVQA	NextQA	HatefulMemes	RareAct
Zero/Few shot SOTA	X		[39] 43.3	[124] 38.2	[134] 32.2	[64] 35.2	-	-	-	[64] 19.2	[145] 12.2	-	[153] 39.4	[87] 11.6	-	-	[94] 66.1	[94] 40.7
		(X)	(16)	(4)	(0)	(0)				(0)	(0)		(0)	(0)			(0)	(0)
Flamingo-3B	X	0	41.2	49.2	73.0	27.5	40.1	28.9	60.6	11.0	32.7	55.8	39.6	46.1	30.1	21.3	53.7	58.4
	X	4	43.3	53.2	85.0	33.0	50.0	34.0	72.0	14.9	35.7	64.6	41.3	47.3	32.7	22.4	53.6	-
	X	8	44.6	55.4	90.6	37.0	54.5	38.4	71.7	19.6	36.8	68.0	40.6	47.6	32.4	23.9	54.7	-
	X	16	45.6	56.7	95.4	40.2	57.1	43.3	73.4	23.4	37.4	73.2	40.1	47.5	31.8	25.2	55.3	-
	X	32	45.9	57.1	99.0	42.6	59.2	45.5	71.2	25.6	37.7	76.7	41.6	OOO	30.6	26.1	56.3	-
Flamingo-9B	X	0	44.7	51.8	79.4	30.2	39.5	28.8	61.5	13.7	35.2	55.0	41.8	48.0	31.8	23.0	57.0	57.9
	X	4	49.3	56.3	93.1	36.2	51.7	34.9	72.6	18.2	37.7	70.8	42.8	50.4	33.6	24.7	62.7	-
	X	8	50.0	58.0	99.0	40.8	55.2	39.4	73.4	23.9	40.0	75.0	43.4	51.2	33.6	25.8	63.9	-
	X	16	50.8	59.4	102.2	44.5	58.5	43.0	72.7	27.6	41.5	77.2	42.4	51.3	33.5	27.6	64.5	-
	X	32	51.0	60.4	106.3	47.2	57.4	44.0	72.8	29.4	40.7	77.3	41.2	OOO	32.6	28.4	63.5	-
Flamingo	X	0	50.6	56.3	84.3	35.6	46.7	31.6	67.2	17.4	40.7	60.1	39.7	52.0	35.0	26.7	46.4	60.8
	X	4	57.4	63.1	103.2	41.7	56.0	39.6	75.1	23.9	44.1	74.5	42.4	55.6	36.5	30.8	68.6	-
	X	8	57.5	65.6	108.8	45.5	60.6	44.8	78.2	27.6	44.8	80.7	42.3	56.4	37.3	32.3	70.0	-
	X	16	57.8	66.8	110.5	48.4	62.8	48.4	78.9	30.0	45.2	84.2	41.1	56.8	37.6	32.9	70.0	-
	X	32	57.8	67.6	113.8	52.3	65.1	49.8	75.4	31.0	45.3	86.8	42.2	OOO	37.9	33.5	70.0	-
Pretrained FT SOTA	✓		54.4	80.2	143.3	47.9	76.3	57.2	67.4	46.8	35.4	138.7	36.7	75.2	54.7	25.2	75.4	-
		(X)	[39] (10K)	[150] (444K)	[134] (500K)	[32] (27K)	[165] (500K)	[70] (20K)	[162] (30K)	[57] (130K)	[145] (6K)	[142] (10K)	[138] (46K)	[87] (123K)	[147] (20K)	[139] (38K)	[60] (9K)	-

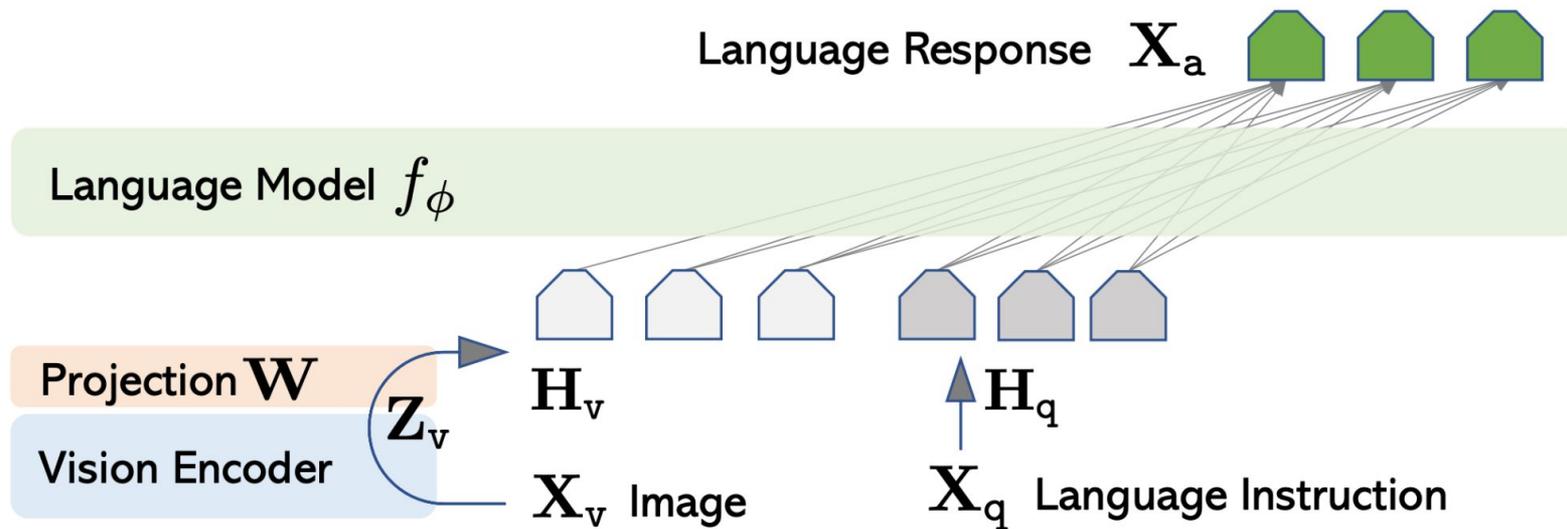
Later multimodal LMs - Molmo



Llava: <https://arxiv.org/pdf/2304.08485>

Molmo: <https://arxiv.org/pdf/2409.17146>

Later multimodal LMs - Llava



Llava: <https://arxiv.org/pdf/2304.08485>

Molmo: <https://arxiv.org/pdf/2409.17146>

Foundation Models

Language

ELMo
BERT
GPT
T5

Classification

CLIP
CoCa

LM + Vision

Flamingo
GPT-4V
Gemini

And More!

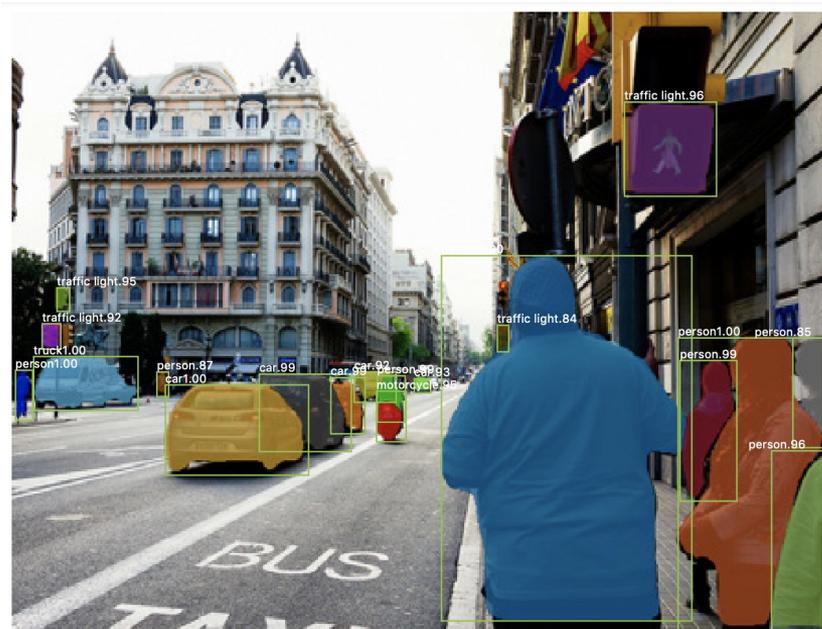
Segment Anything
Whisper
Dalle
Stable Diffusion
Imagen

Chaining

LMs + CLIP
Visual Programming

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

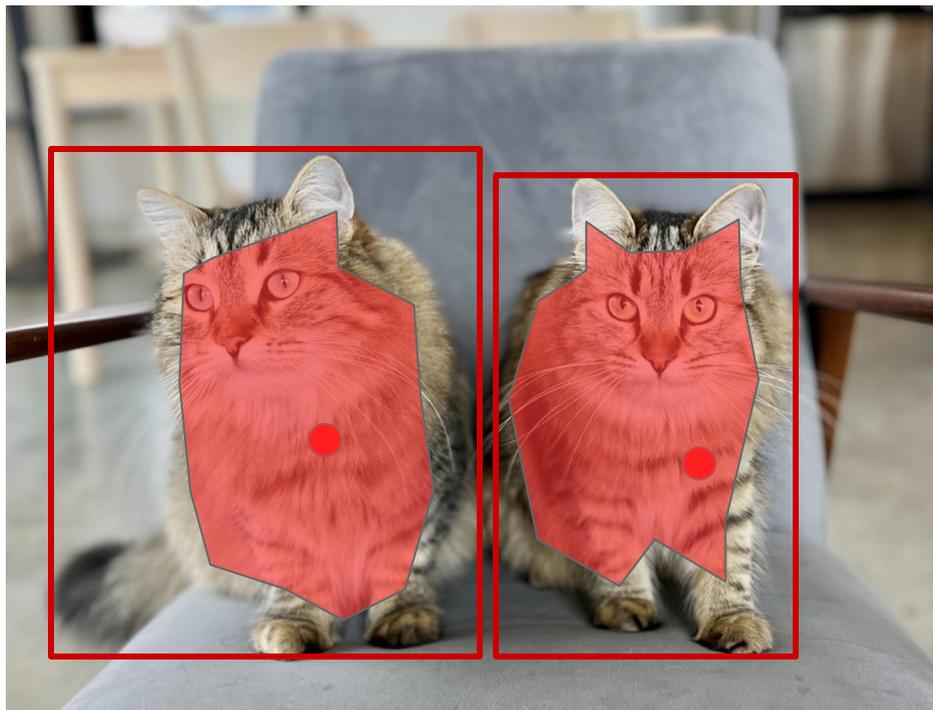
How to get this?

Model outputs mask of any objects that the user cares about

How to know this?

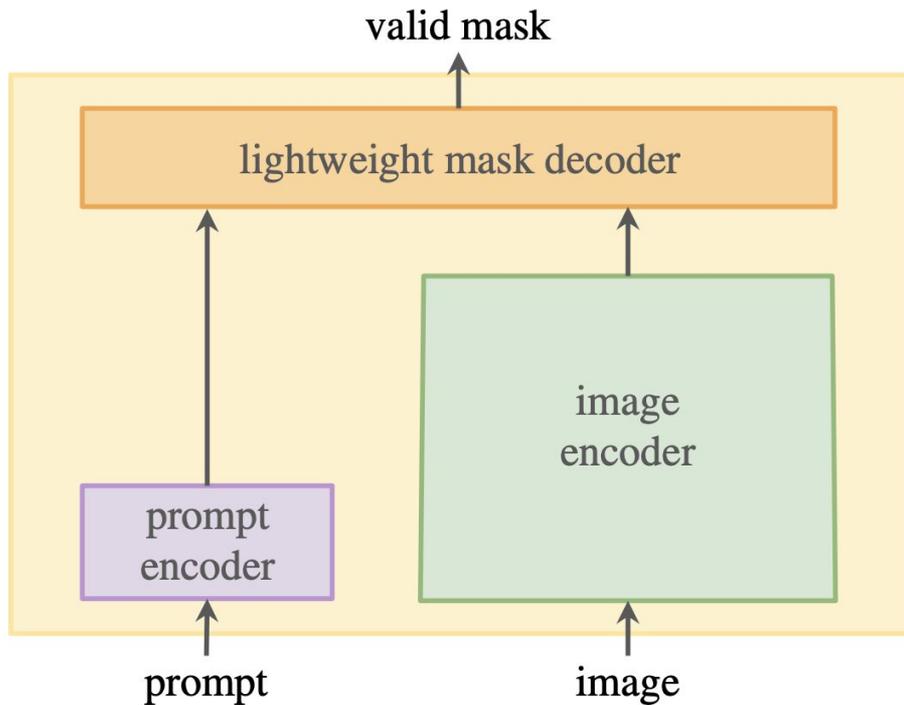
Images: Kirillov et al. Segment Anything. 2023.

How to know what to mask?



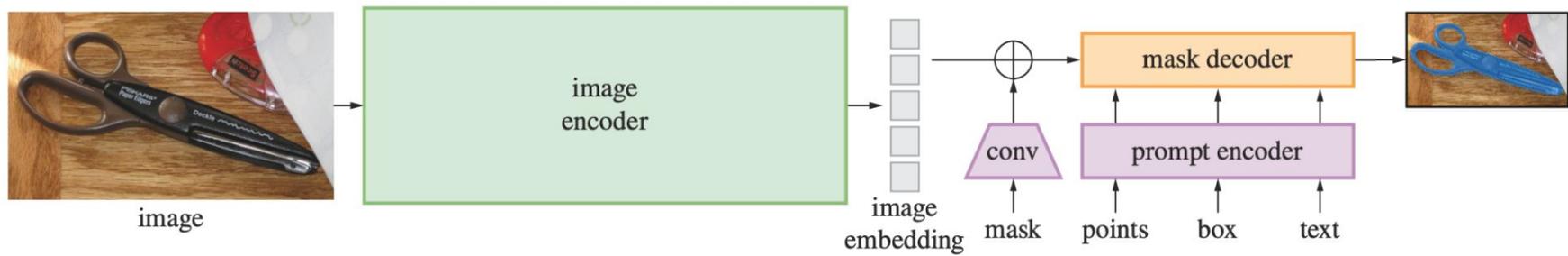
“Cats”

Basic SAM Architecture



Images: Kirillov et al. Segment Anything. 2023.

SAM Architecture



Images: Kirillov et al. Segment Anything. 2023.

Ambiguity in correct prompt



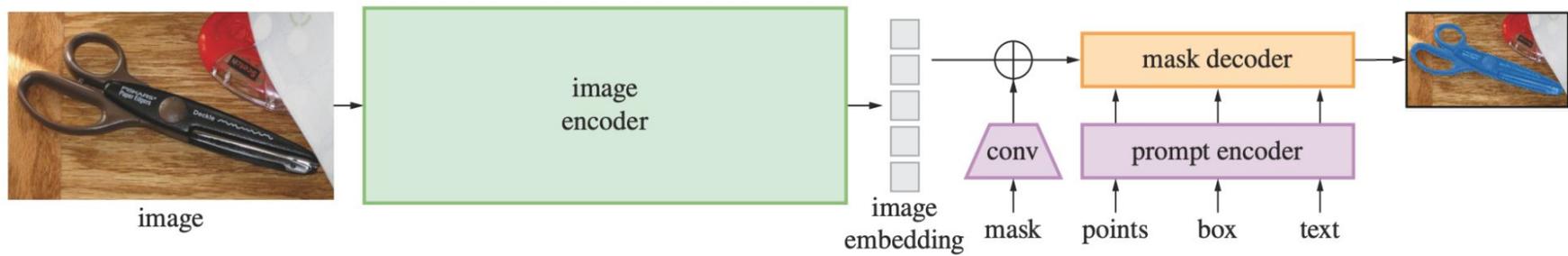
Images: Kirillov et al. Segment Anything. 2023.

Ambiguity in correct prompt



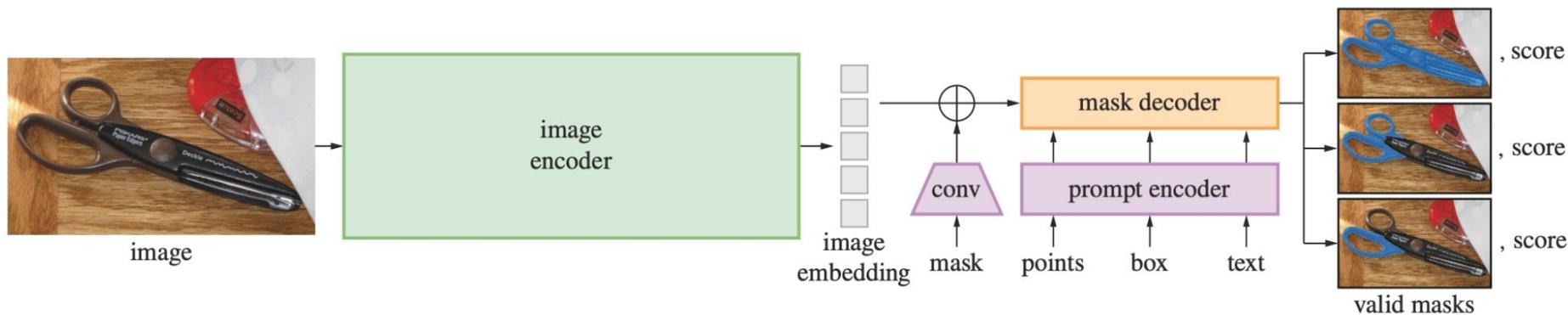
Images: Kirillov et al. Segment Anything. 2023.

SAM Architecture



Images: Kirillov et al. Segment Anything. 2023.

Basic 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)

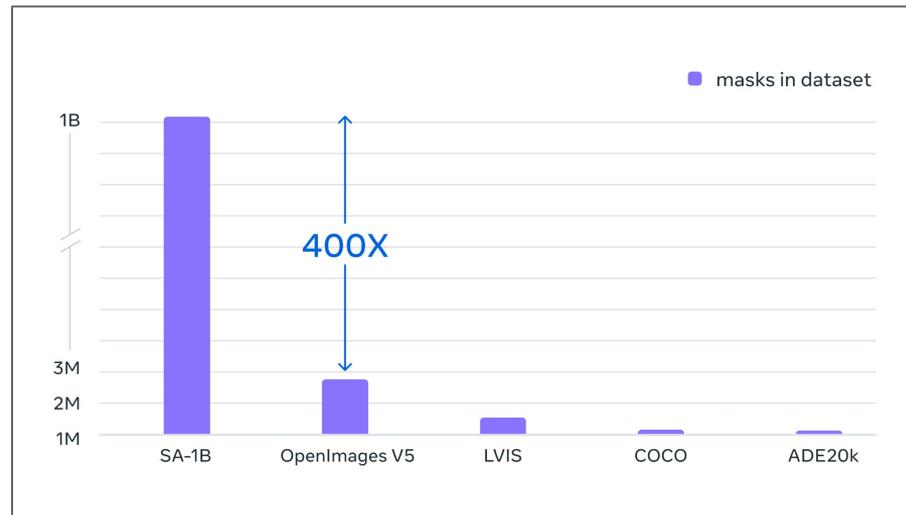
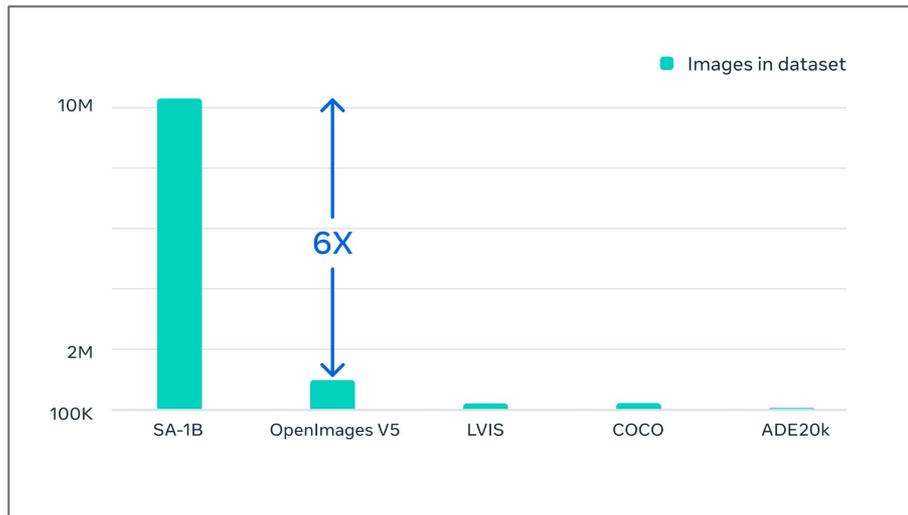
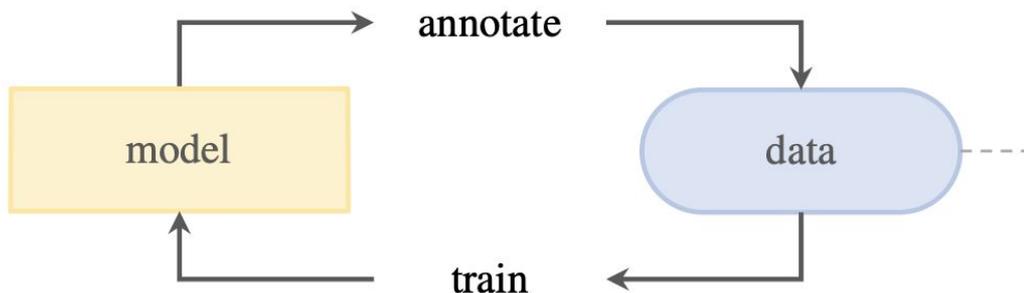


Image Source: <https://segment-anything.com/>

Segment Anything Model (SAM)



Segment Anything 1B (SA-1B):

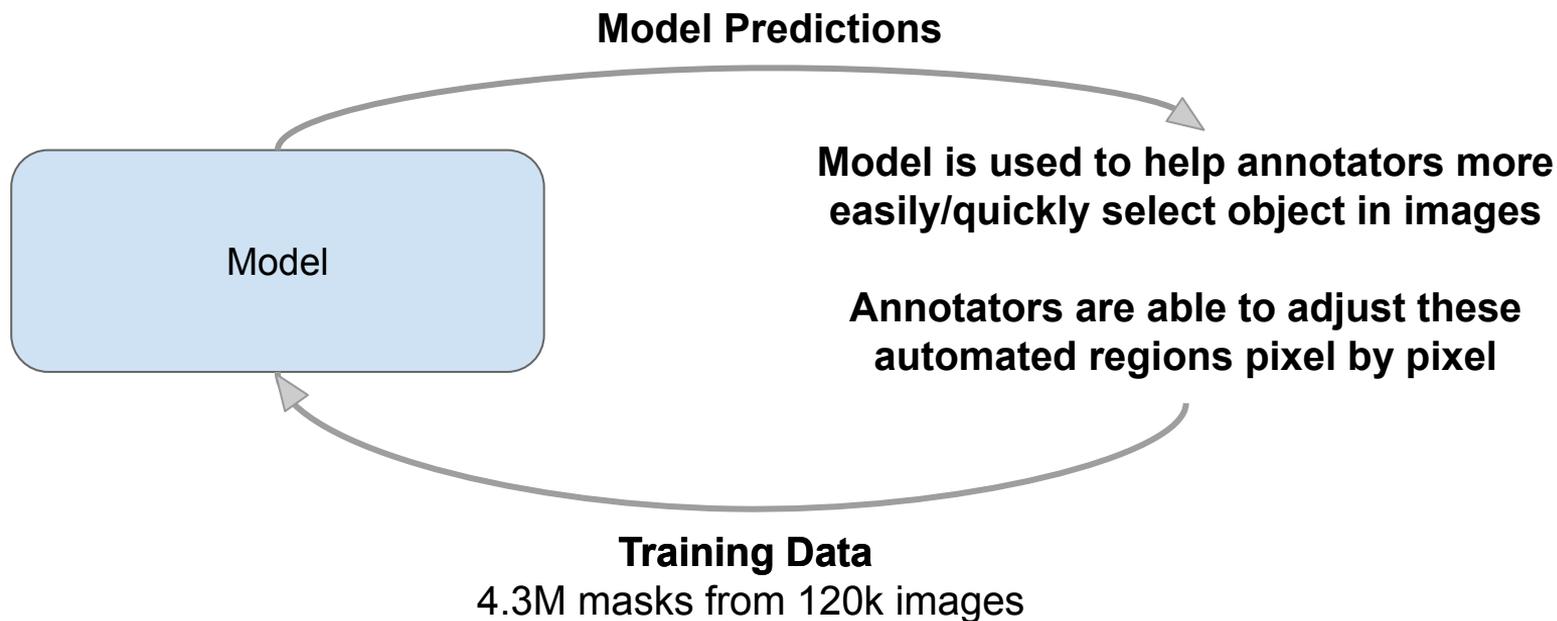
- 1+ billion masks
- 11 million images
- privacy respecting
- licensed images



Images: Kirillov et al. Segment Anything. 2023.

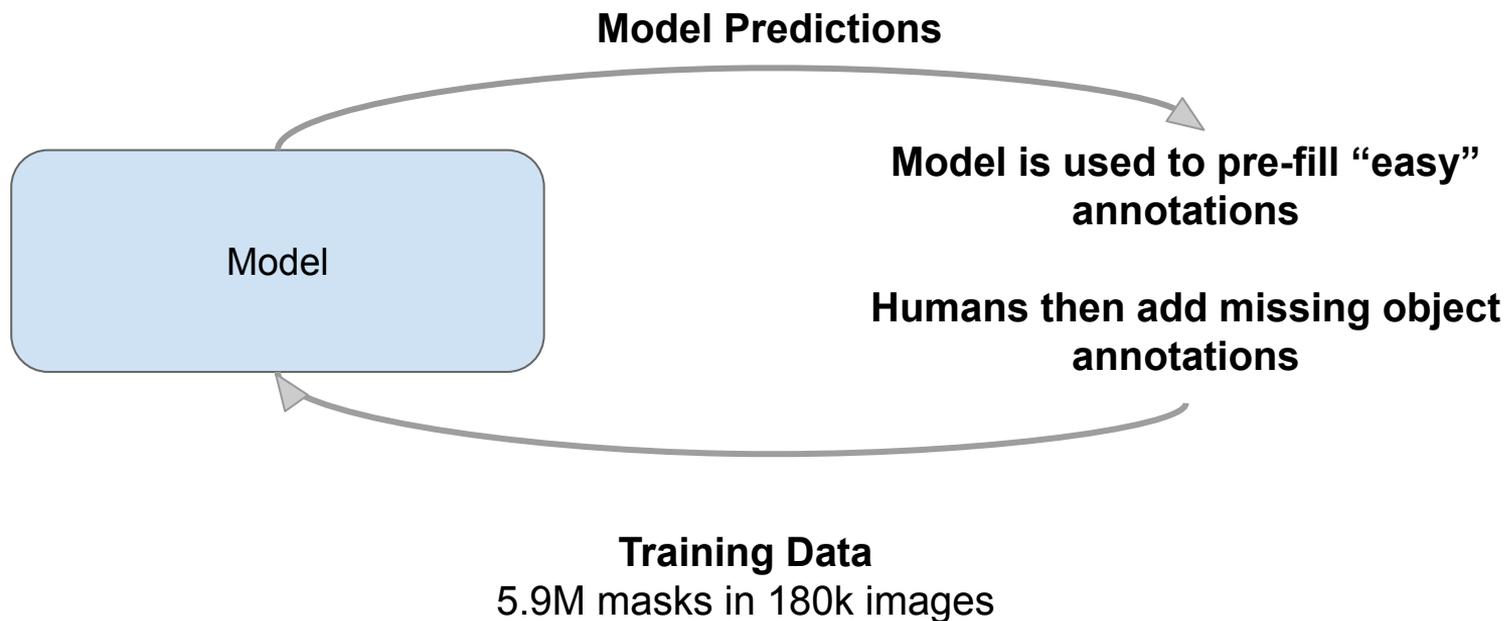
Segment Anything Model (SAM)

Assisted-manual stage



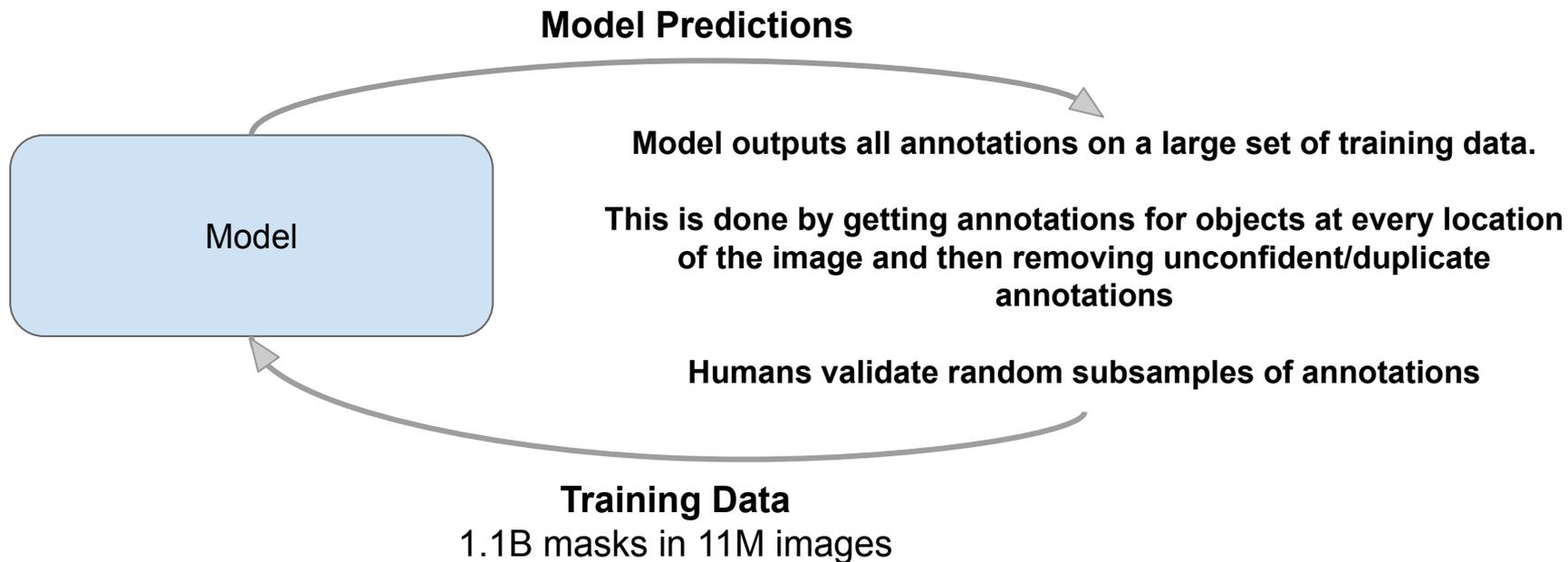
Segment Anything Model (SAM)

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

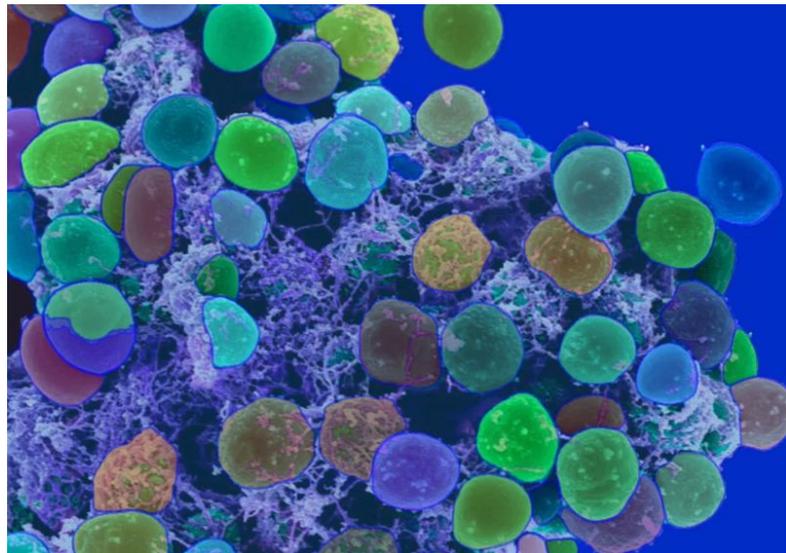
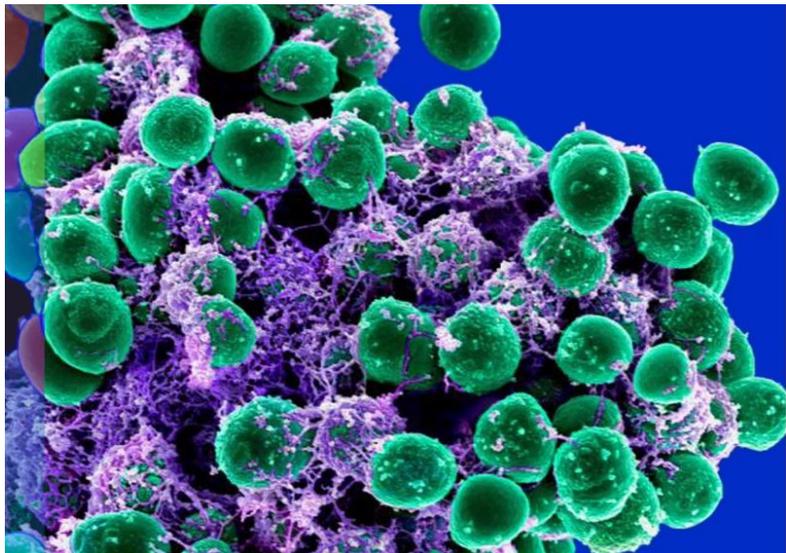


Image Source: <https://segment-anything.com/>

Zero-Shot with SAM

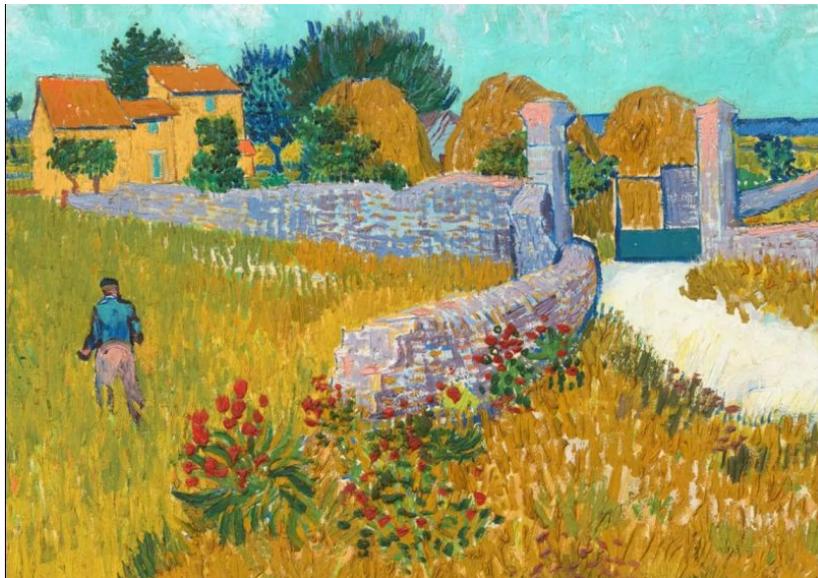


Image Source: <https://segment-anything.com/>

Foundation Models

Language

ELMo
BERT
GPT
T5

Classification

CLIP
CoCa

LM + Vision

Flamingo
GPT-4V
Gemini

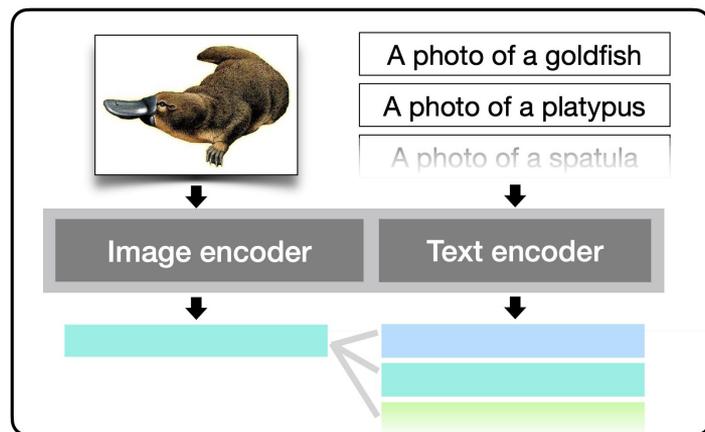
And More!

Segment Anything
Whisper
Dalle
Stable Diffusion
Imagen

Chaining

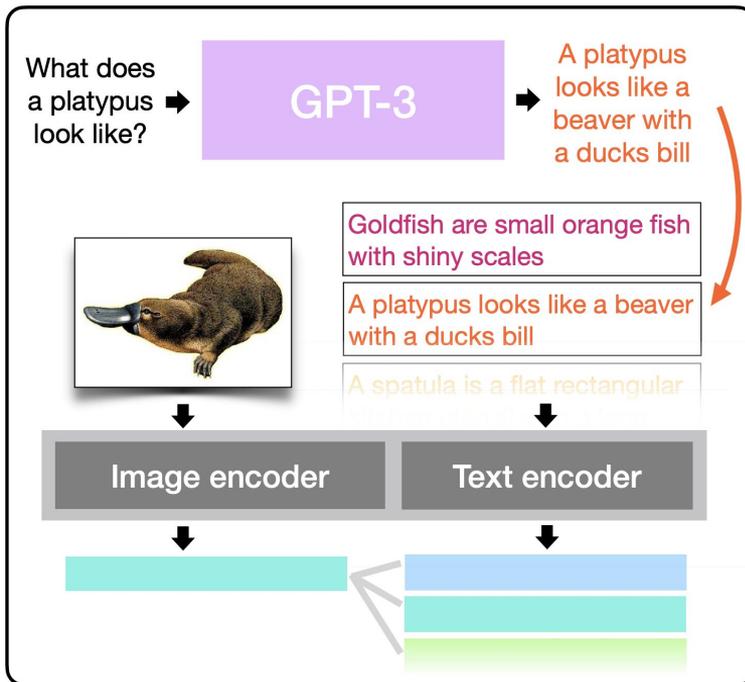
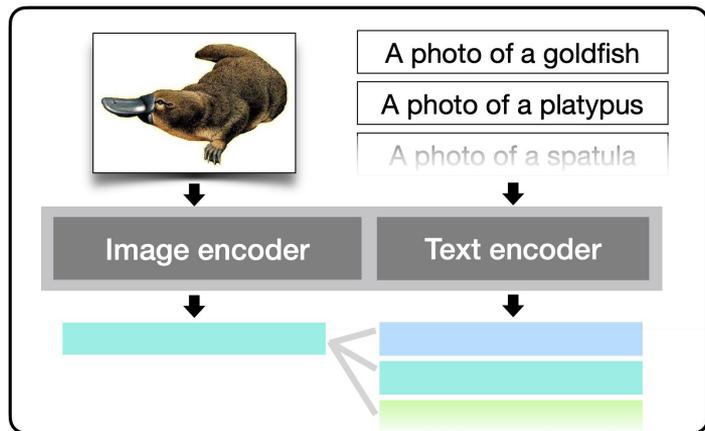
LMs + CLIP
Visual Programming
Tool use

CuPL (CUsutomized Prompts via Language models)



Pratt et al "What does a platypus look like? Generating customized prompts for zero-shot image classification". 2023.

CuPL (CUsutomized Prompts via Language models)



Pratt et al "What does a platypus look like? Generating customized prompts for zero-shot image classification". 2023.

CuPL (CUsutomized Prompts via Language models)

A photo of a marimba
A photo of a viaduct
A photo of a papillon
A photo of a lorikeet



Pratt et al "What does a platypus look like? Generating customized prompts for zero-shot image classification". 2023.

CuPL (CUsutomized Prompts via Language models)

“A **marimba** is a large wooden percussion instrument that looks like a xylophone.”

“A **viaduct** is a bridge composed of several spans supported by piers or pillars.”

“A **papillon** is a small, spaniel-type dog with a long, silky coat and fringed ears.”

“A **lorikeet** is a small to medium-sized parrot with a brightly colored plumage.”



Pratt et al “What does a platypus look like? Generating customized prompts for zero-shot image classification”. 2023.

CuPL (CUsutomized Prompts via Language models)

“A **marimba** is a large wooden percussion instrument that looks like a xylophone.”

“A **viaduct** is a bridge composed of several spans supported by piers or pillars.”

“A **papillon** is a small, spaniel-type dog with a long, silky coat and fringed ears.”

“A **lorikeet** is a small to medium-sized parrot with a brightly colored plumage.”



Lorikeet



Marimba



Viaduct



Papillon

Pratt et al “What does a platypus look like? Generating customized prompts for zero-shot image classification”. 2023.

CuPL (CUsutomized Prompts via Language models)

LLM-prompts:

“What does a
{lorikeet, marimba,
viaduct, papillon}
look like?”



GPT-3

Image-prompts:

“A **lorikeet** is a small to medium-sized parrot with a brightly colored plumage.”
“A **marimba** is a large wooden percussion instrument that looks like a xylophone.”
“A **viaduct** is a bridge composed of several spans supported by piers or pillars.”
“A **papillon** is a small, spaniel-type dog with a long, silky coat and fringed ears.”



Lorikeet



Marimba



Viaduct



Papillon

Pratt et al “What does a platypus look like? Generating customized prompts for zero-shot image classification”. 2023.

CuPL (CUsutomized Prompts via Language models)

	ImageNet	DTD	Stanford Cars	SUN397	Food101	FGVC Aircraft	Oxford Pets	Caltech101	Flowers 102	UCF101	Kinetics-700	RESISC45	CIFAR-10	CIFAR-100	Birdsnap	mean
std	75.54	55.20	77.53	69.31	93.08	32.88	93.33	93.24	78.53	77.45	60.07	71.10	95.59	78.26	50.43	73.43
# hw	80	8	8	2	1	2	1	34	1	48	28	18	18	18	1	
CuPL (full)	76.69	61.70	77.63	73.31	93.36	36.11	93.81	93.45	79.67	78.36	60.63	71.69	95.84	78.57	51.11	74.80
Δ std	+1.15	+6.50	+0.10	+4.00	+0.28	+3.23	+0.48	+0.21	+1.14	+0.91	+0.56	+0.59	+0.25	+0.31	+0.63	
# hw	5	6	9	3	3	2	2	3	2	5	4	5	3	4	3	

Pratt et al "What does a platypus look like? Generating customized prompts for zero-shot image classification". 2023.

VisProg (visual programming)

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



Are there 3 people in the boat?

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

VisProg (visual programming)

LEFT:



RIGHT:

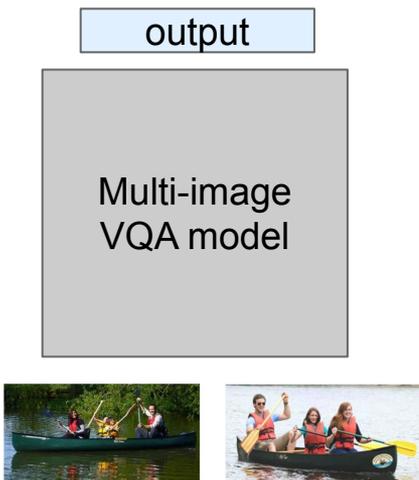


Statement: The left and right image contains a total of six people and two boats.

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.

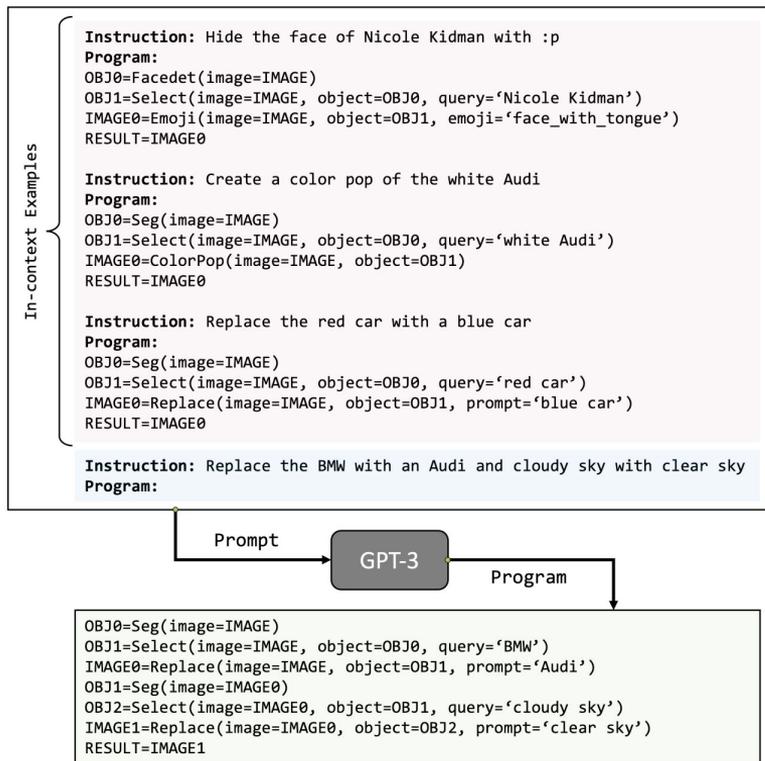
GPT

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

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
```

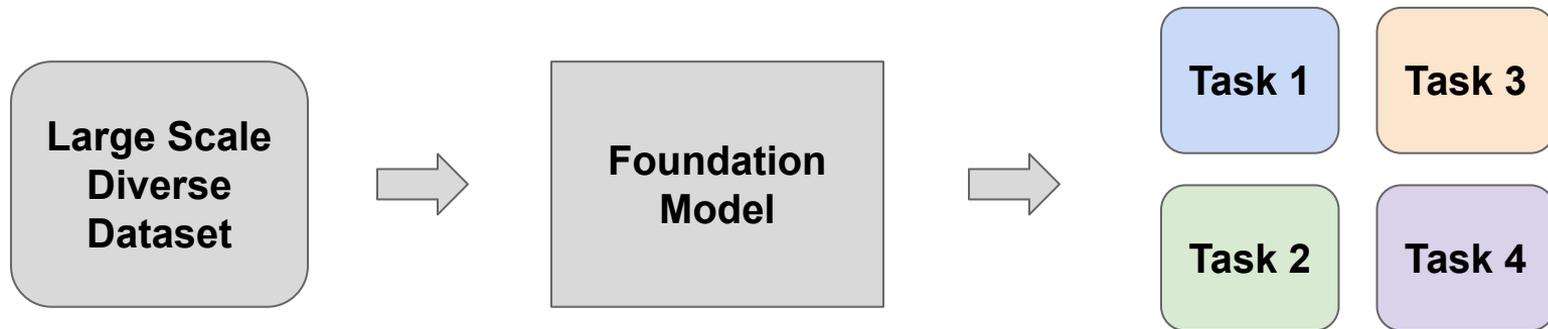
Gupta et al "Visual Programming: Compositional visual reasoning without training". 2023.

Tool Use! Agents! Code!



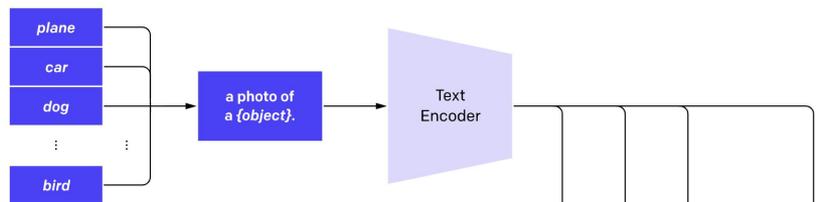
<https://www.anthropic.com/news/enabling-claude-code-to-work-more-autonomously>

Summary

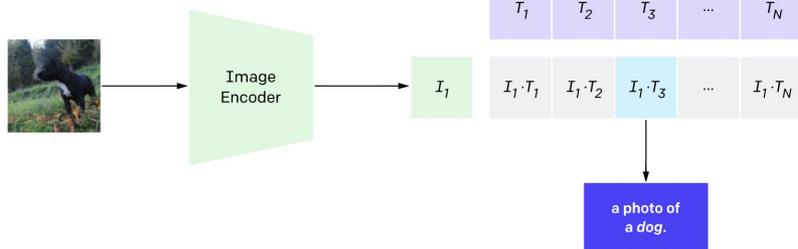


Summary

2. Create dataset classifier from label text

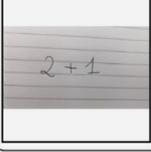
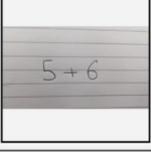
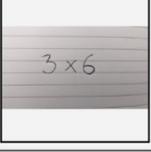


3. Use for zero-shot prediction



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

Input Prompt				Completion	
	This is a chinchilla. They are mainly found in Chile.		This is a shiba. They are very popular in Japan.		This is a flamingo. They are found in the Caribbean and South America.
	What is the title of this painting? Answer: The Hallucinogenic Toreador.		Where is this painting displayed? Answer: Louvres Museum, Paris.		What is the name of the city where this was painted? Answer: Arles.
	Output: "Underground"		Output: "Congress"		Output: "Soulomes"
	2+1=3		5+6=11		3x6=18

Summary



Next time: Scale!