# NLP and Computer Vision: Multimodality

*Federico Stella*

EPFL

# Presentation structure

**1**

***STEP***

***Interaction***
***Reasoning***
***Multimodality***

**2**

***STEP***

***Computer Vision***

**3**

***STEP***

***Flamingo***

# *On the Opportunities and Risks of Foundation Models*

Rishi Bommasani, Percy Liang et al. (2022)

- 2.2 Vision

  Shyamal Buch, Drew A. Hudson, Frieda Rong, Alex Tamkin, Xikun Zhang, Bohan Wu, Ehsan Adeli, Stefano Ermon, Ranjay Krishna, Juan Carlos Niebles, Jiajun Wu, Li Fei-Fei

- 2.3 Robotics

  Siddharth Karamcheti, Annie Chen, Suvir Mirchandani, Suraj Nair, Krishnan Srinivasan, Kyle Hsu, Jeannette Bohg, Dorsa Sadigh, Chelsea Finn

- 2.4 Reasoning and Search

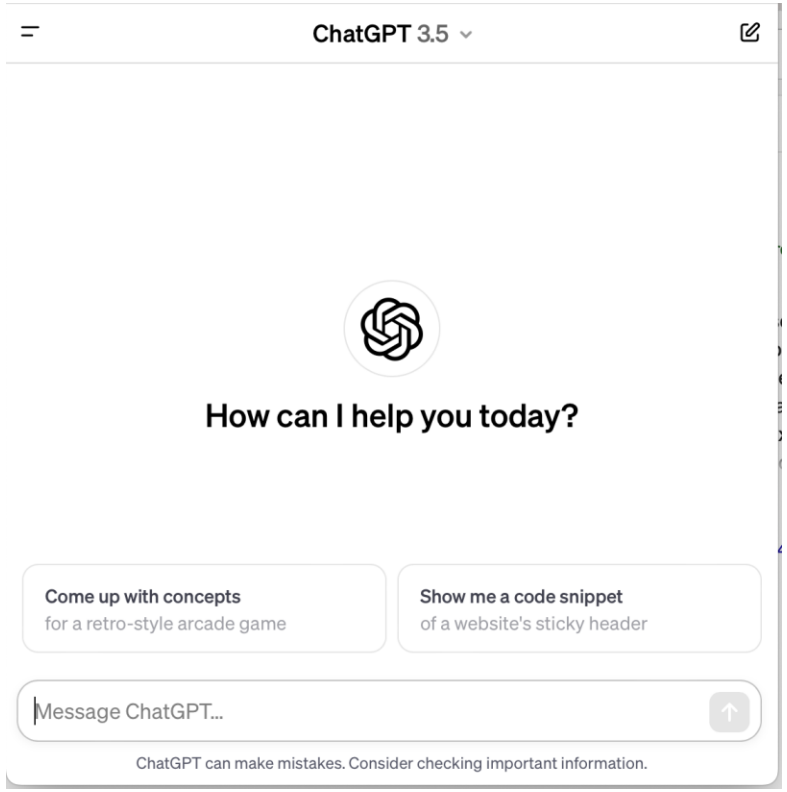
  Yuhuai Wu, Frieda Rong, Hongyu Ren, Sang Michael Xie, Xuechen Li, Andy Shih, Drew A. Hudson, Omar Khattab

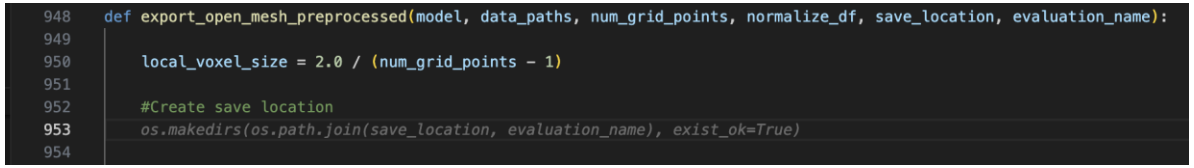
- 2.5 Interaction

  Joon Sung Park, Chris Donahue, Mina Lee, Siddharth Karamcheti, Dorsa Sadigh, Michael S. Bernstein

# How do we interact with foundation models?

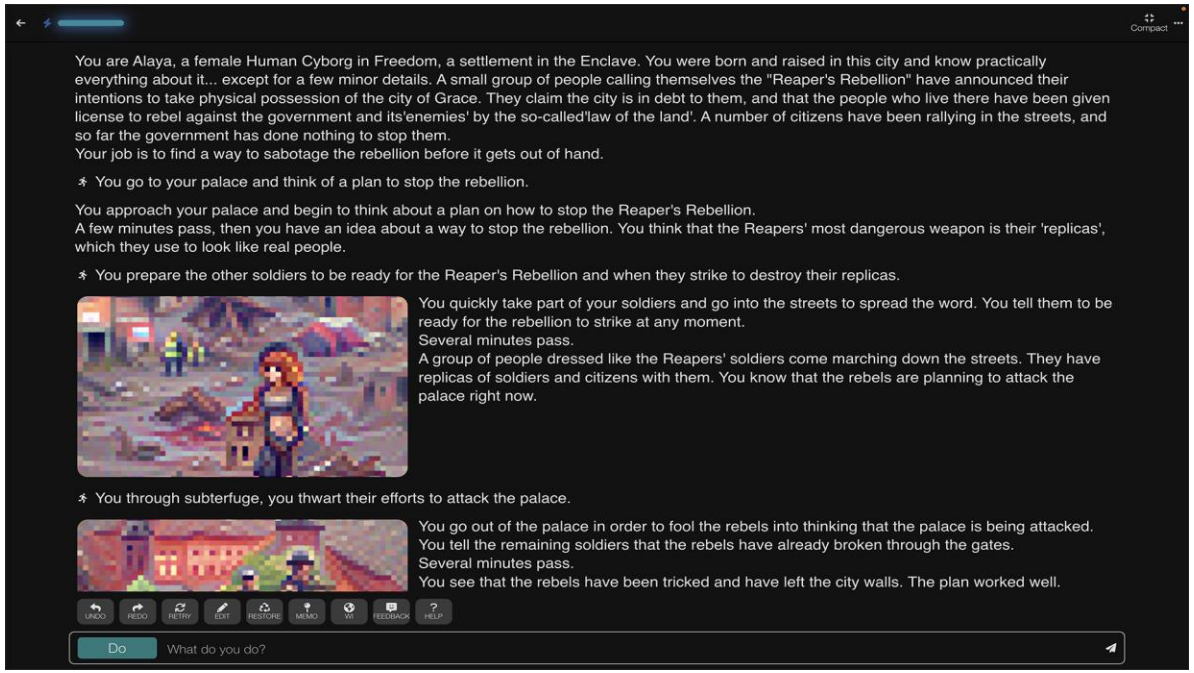
ChatGPT



```
948  def export_open_mesh_preprocessed(model, data_paths, num_grid_points, normalize_df, save_location, evaluation_name):
949
950      local_voxel_size = 2.0 / (num_grid_points - 1)
951
952      #Create save location
953      os.makedirs(os.path.join(save_location, evaluation_name), exist_ok=True)
954
```

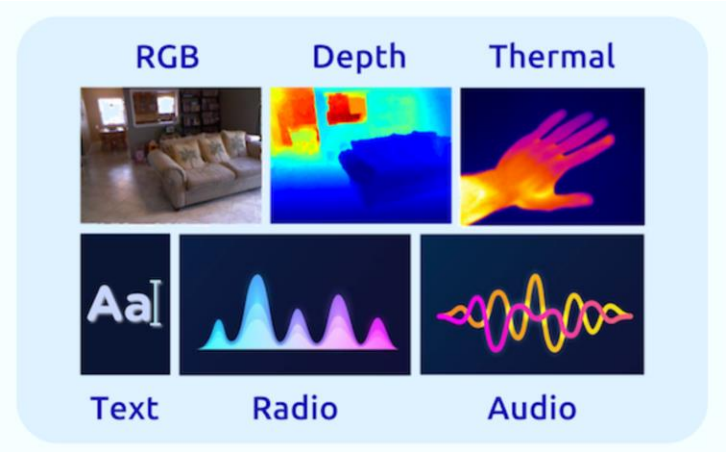
GitHub Copilot



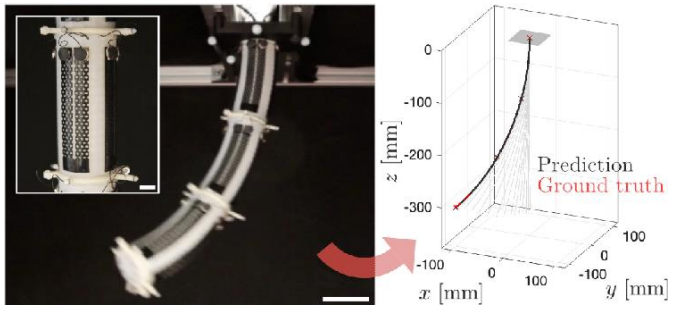

AI Dungeon

Mainly via text inputs

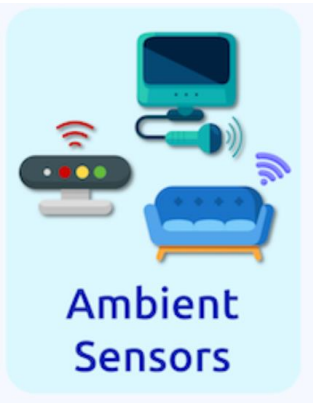
# Are there other modalities?








Robotic proprioception




**Why are they useful?**

*Better interaction*

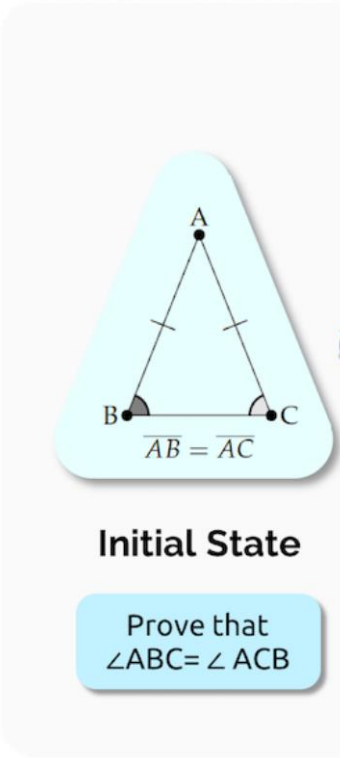
*Ability to solve new tasks (driving, medical, multimedia, …)*

*Increased model capabilities*

**Is that all?**

# A small mind experiment

Given an isosceles triangle ABC, prove that the angles corresponding to the congruent segments ∠ABC, ∠ACB are equal.




Reasoning is easier when we visually ground our words

# Reasoning

Reasoning is easier when we visually ground our words → Reasoning may benefit from multimodal models

## Important for many tasks

Theorem proving

Drug discovery

Computer aided design

More

Program synthesis

Chemical synthesis

Combinatorial optimization

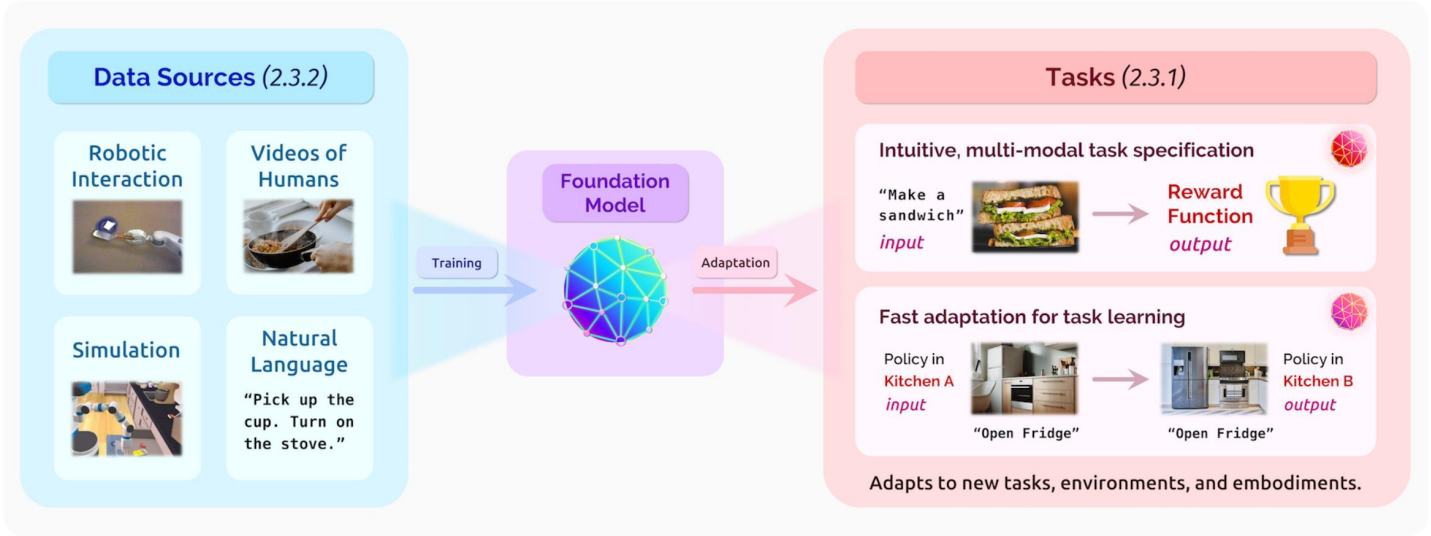
Problems with a very big search space

LLMs already offer a generic way to approach these problems and have some reasoning capabilities thanks to:

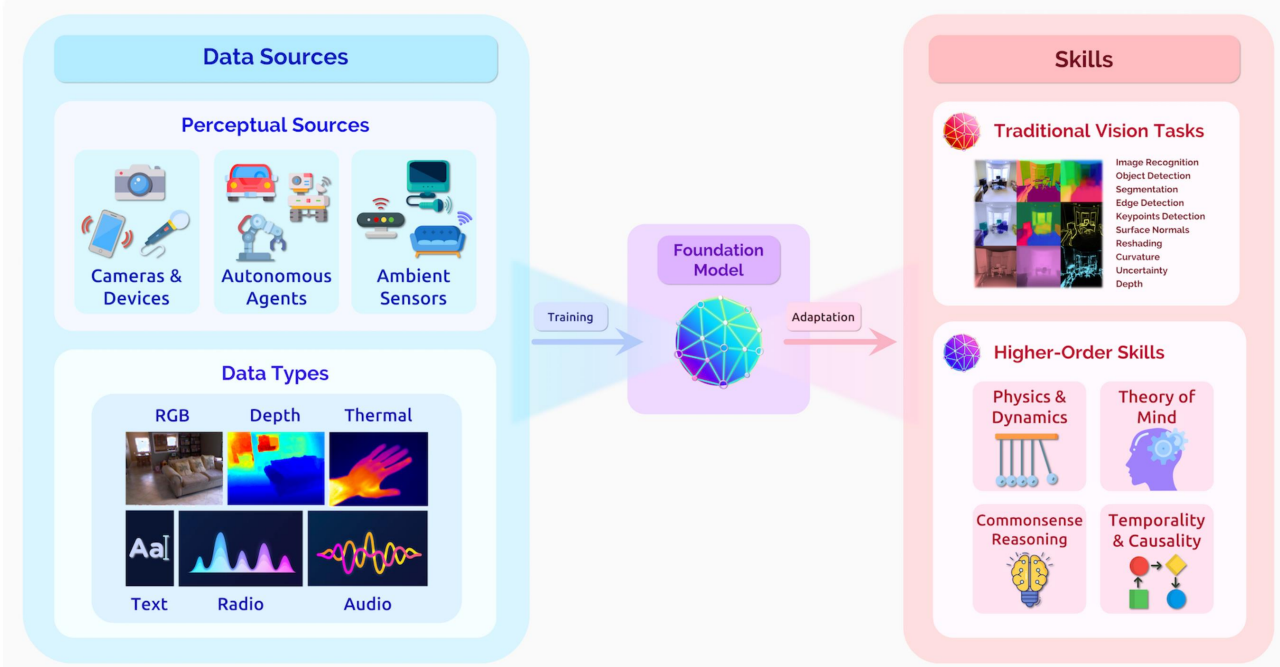
- Generativity
- Universality
- **Grounding**

# Multimodality: examples

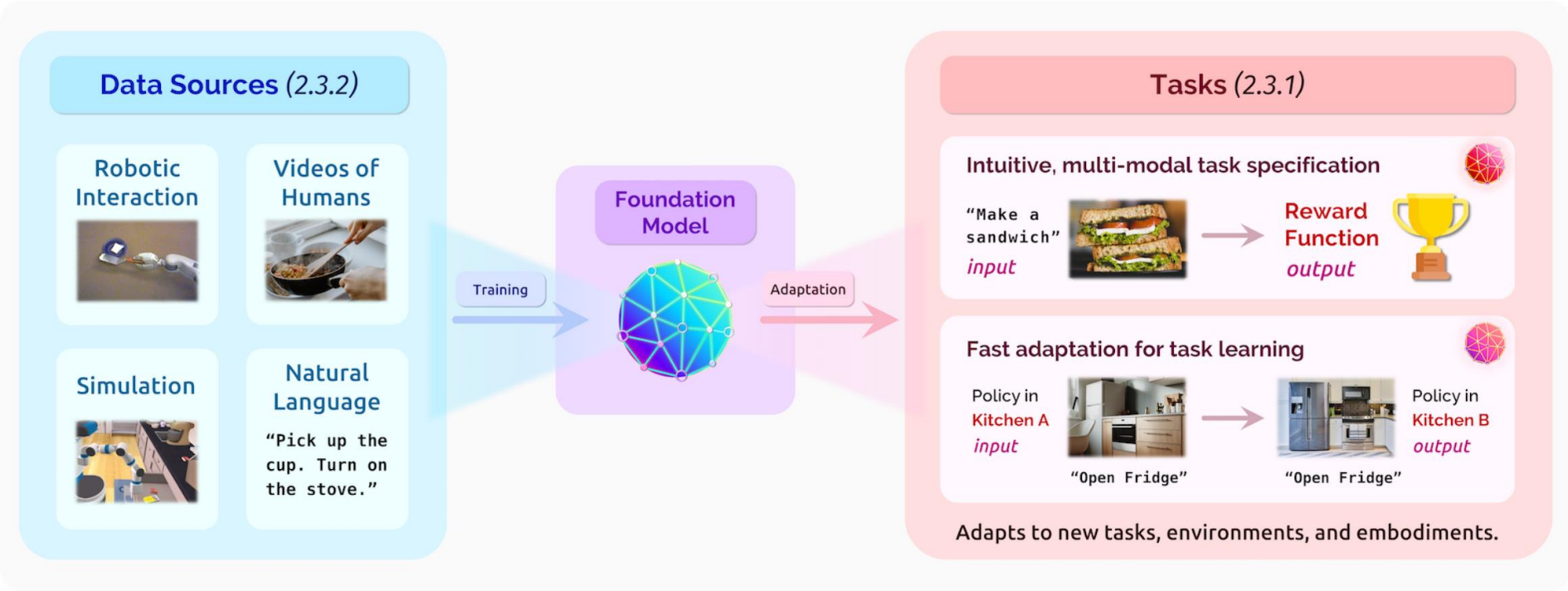
## Robotics




## Vision



# Robotics




Most important and difficult things in robotics!

Examples:

- Manufacturing
- Construction
- Autonomous driving
- Household aid
- Personal assistance
- Etc.

Challenges:

- Safety and robustness
- **Data needs**

Common to all modalities!

- What data?
- How do we get it?

# Robotics

Hans Moravec paradox (1988): In AI, hard problems are easy and likewise easy problems are hard, and among the "easiest" problems of them all is the visual acuity which we use each day to continually interpret complex scenes in a matter of milliseconds.

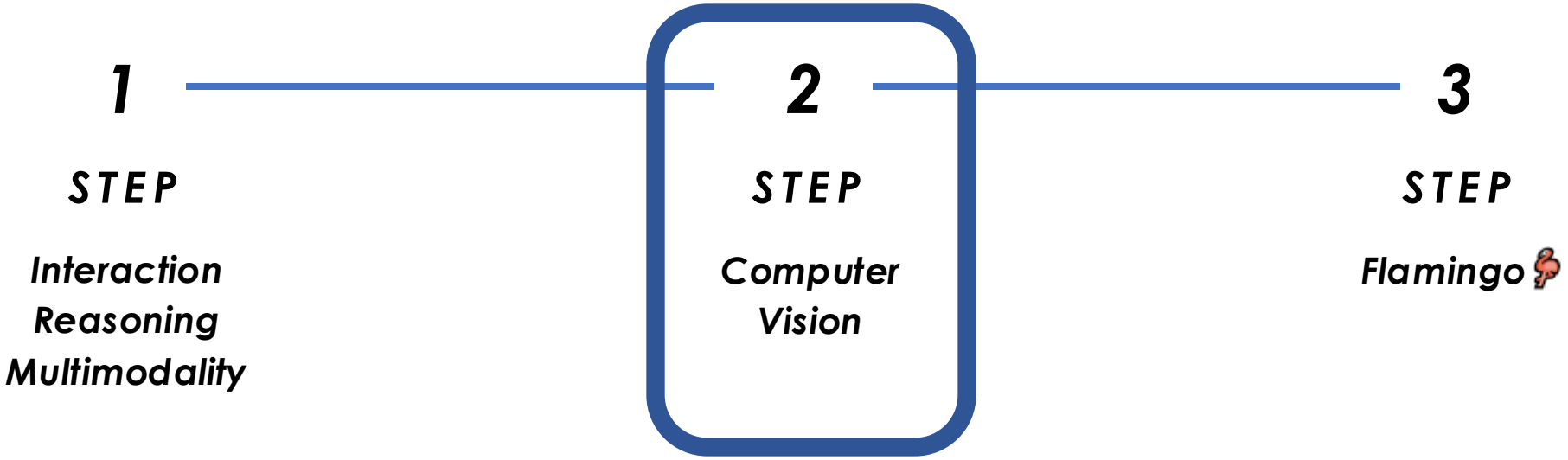
James Gibson (1979): Human vision is inherently embodied and interactive ecological environments may play a key role in its development.

There are many things going on when humans learn:

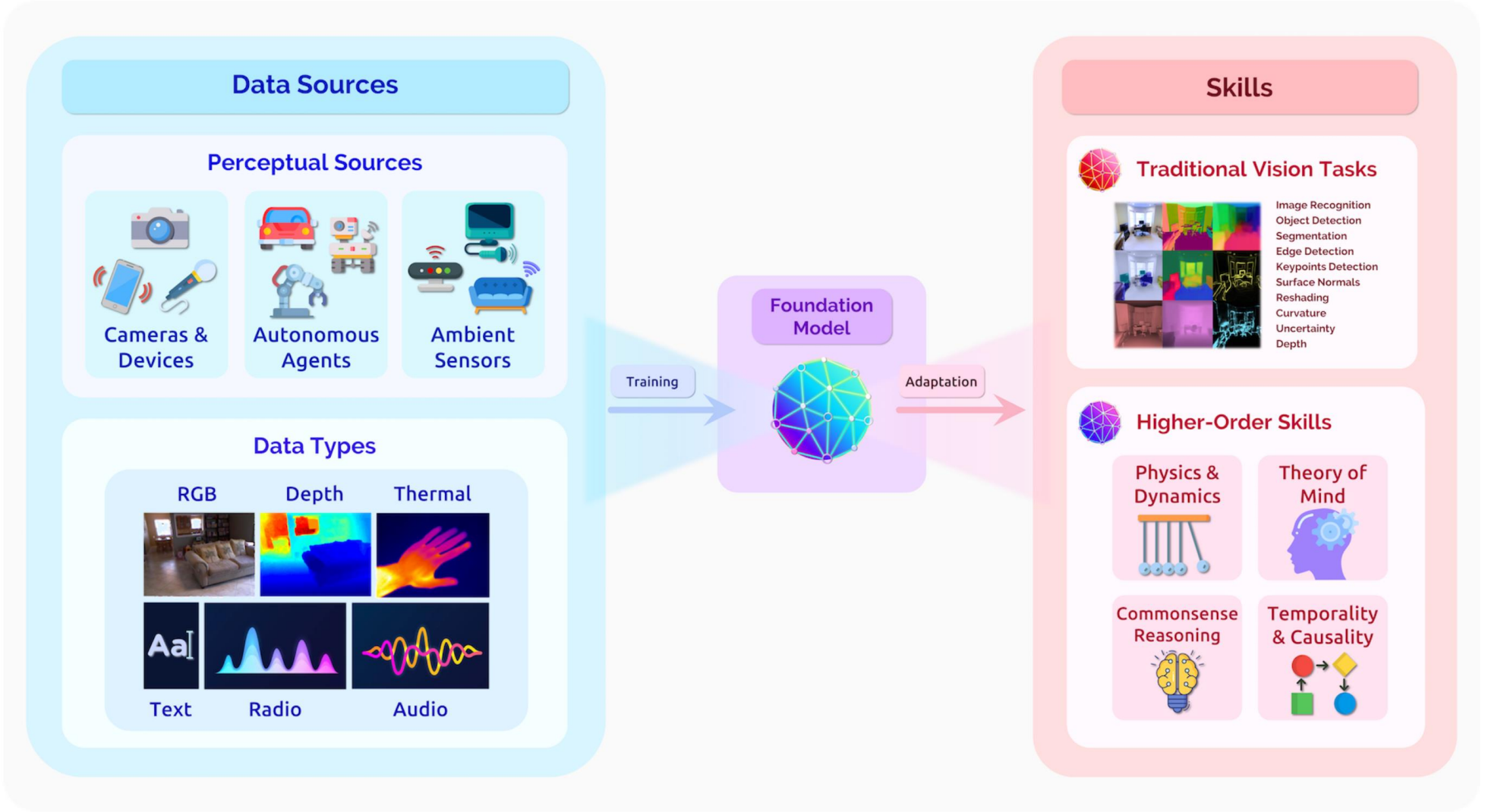
- Some explicit teaching
- Self supervision
- Interaction
- World model
- Grounding
- Embodiment
- Etc.

**Let's take one step at a time!**

# Presentation structure



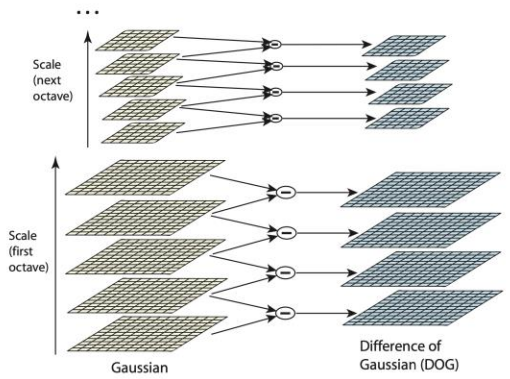
# Vision



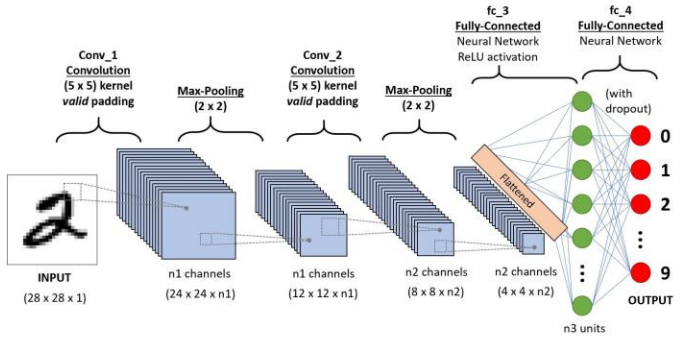

In a way, simpler than robotics:

- It does not explicitly require embodiment and on-line interaction
- Image/Video+Text datasets are easier to find compared to robotics

# Very brief history of (computer) vision




Hand-crafted feature extraction
[Lowe 2004]

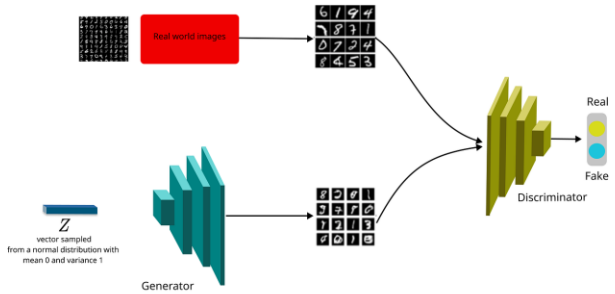



Efficient GPU implementations of CNNs
[Multiple works, 2004-2006]

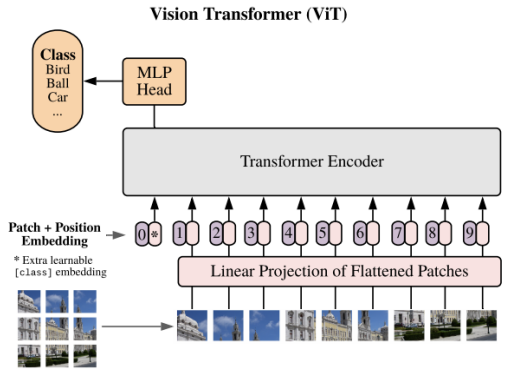
ImageNet and deep CNNs
[ImageNet 2009, AlexNet 2012]

- Large, general-purpose pre-training
- Adaptation to downstream tasks

Similar to foundation models in spirit, but needs **annotated data** and lacks in-context learning




GANs, VAEs
[Goodfellow 2014, Kingma and Welling 2014]




Contrastive Learning, Vision Transformer
[Chen 2020, Dosovitskiy 2021]

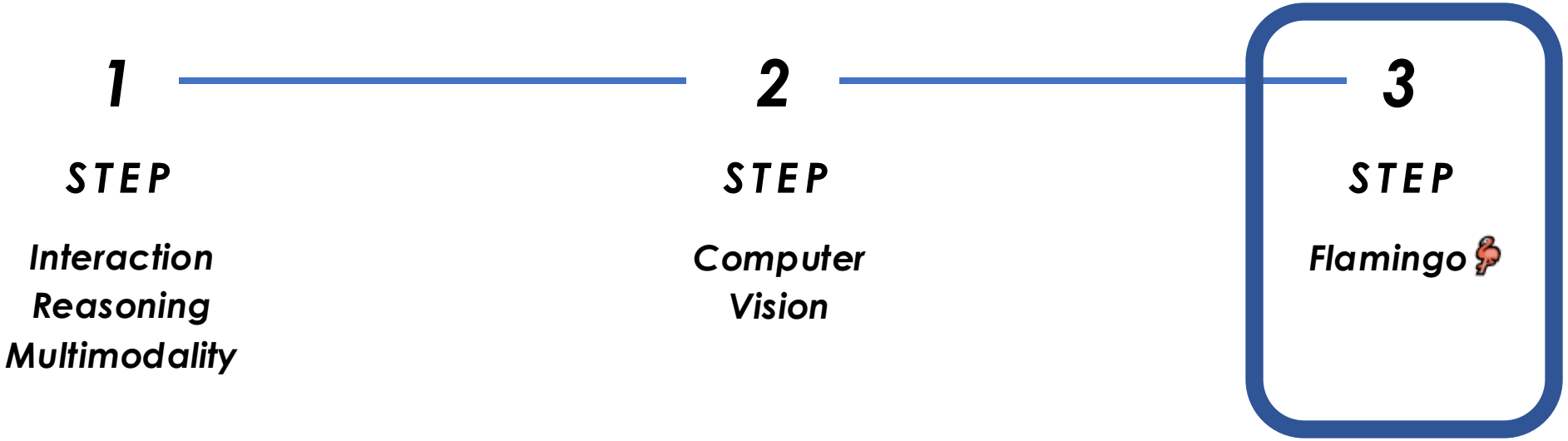
Visual BERT, CLIP, Flamingo belong here

# *Multimodality vision + language: challenges*

- Reduce dependence on explicit annotations
- How to incorporate the modalities (models, architectures, losses)
- How to generalize to unseen scenes and enable reasoning on physical and geometric properties
- Size and complexity (images are big!)
- How to get useful datasets
- How to benchmark the models

Even though vision does not explicitly require embodiement and interaction, there are clearly many challenges. However the authors are positive about future progress

# Presentation structure



# *Flamingo: a Visual Language Model for Few-Shot Learning*

by Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, Karen Simonyan

Google DeepMind

NeurIPS 2022

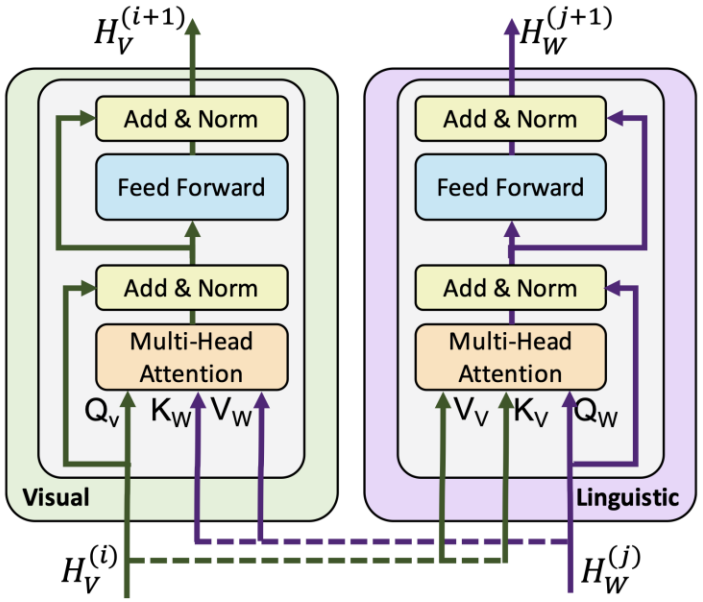
# Flamingo: a family of Visual Language Models for Few-Shot Learning

by Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, Karen Simonyan

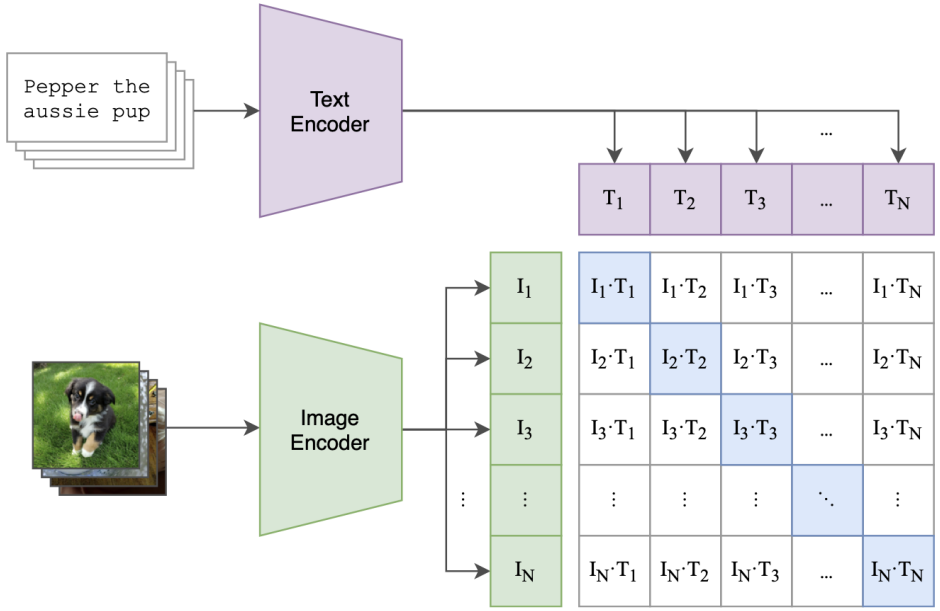
Google DeepMind

NeurIPS 2022

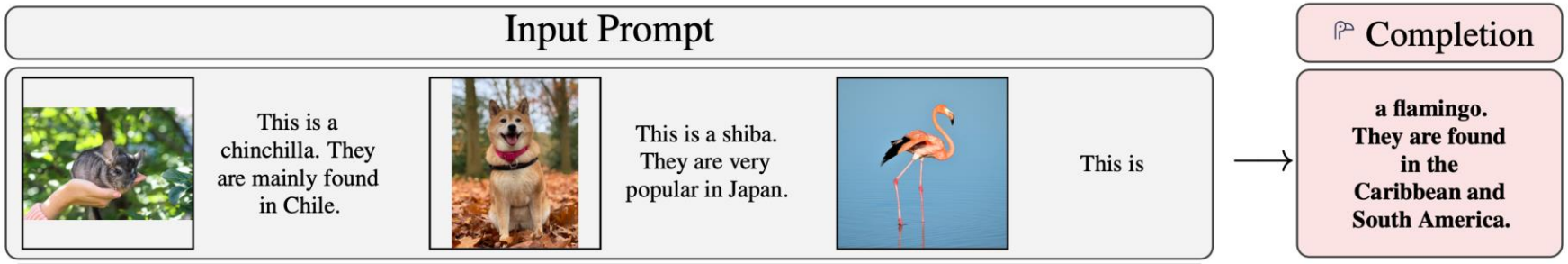
# Starting point




Visual BERT models (ViLBERT)
Require fine-tuning

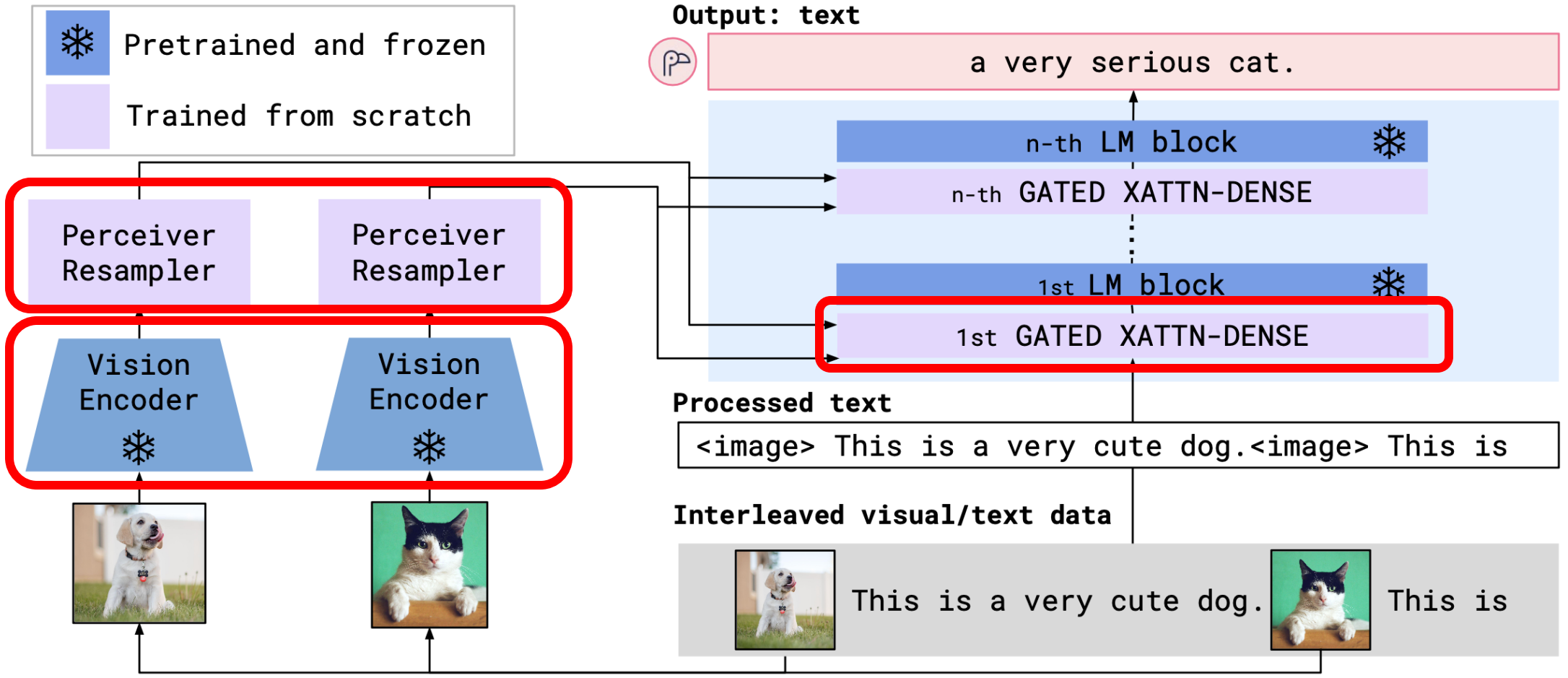



Contrastive pre-trained models (CLIP)
Cannot generate language

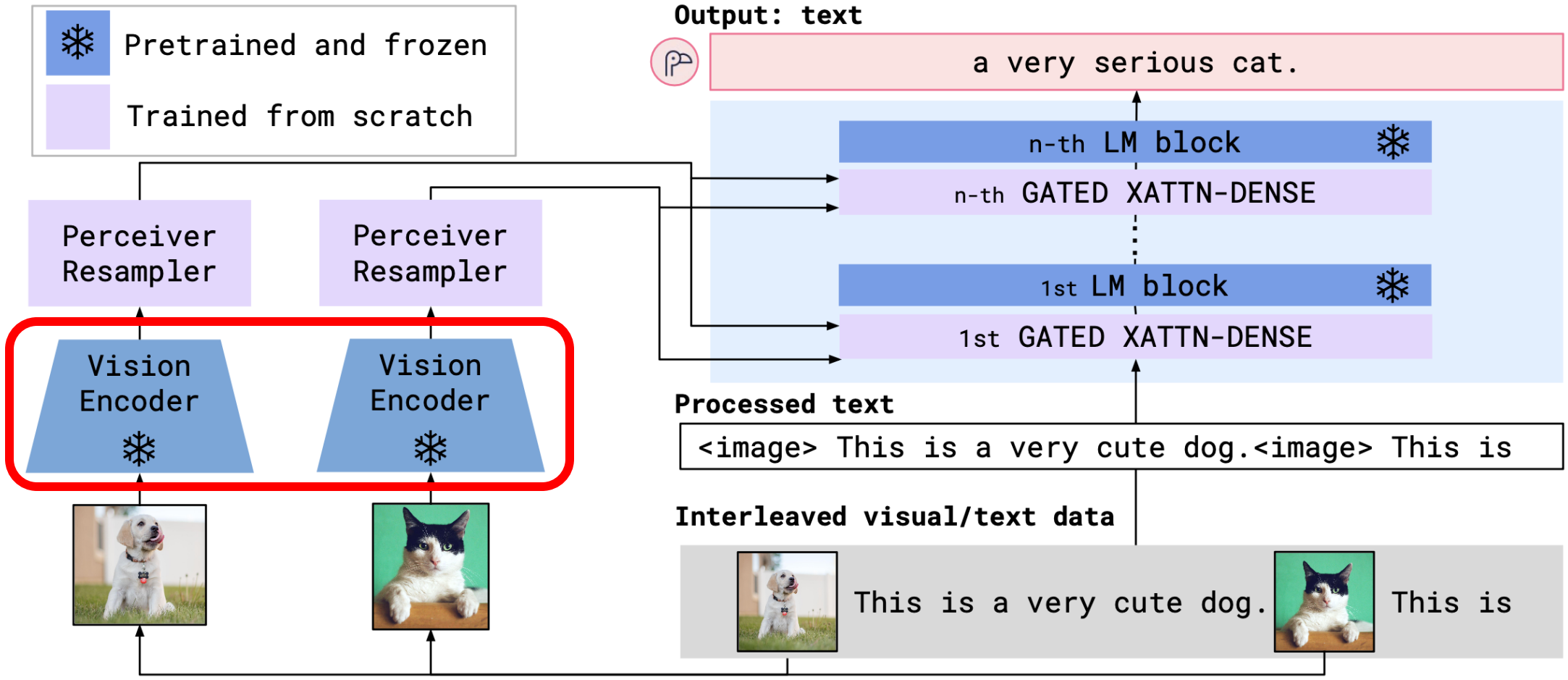



Flamingo
Language generation + in-context learning

# Flamingo architecture

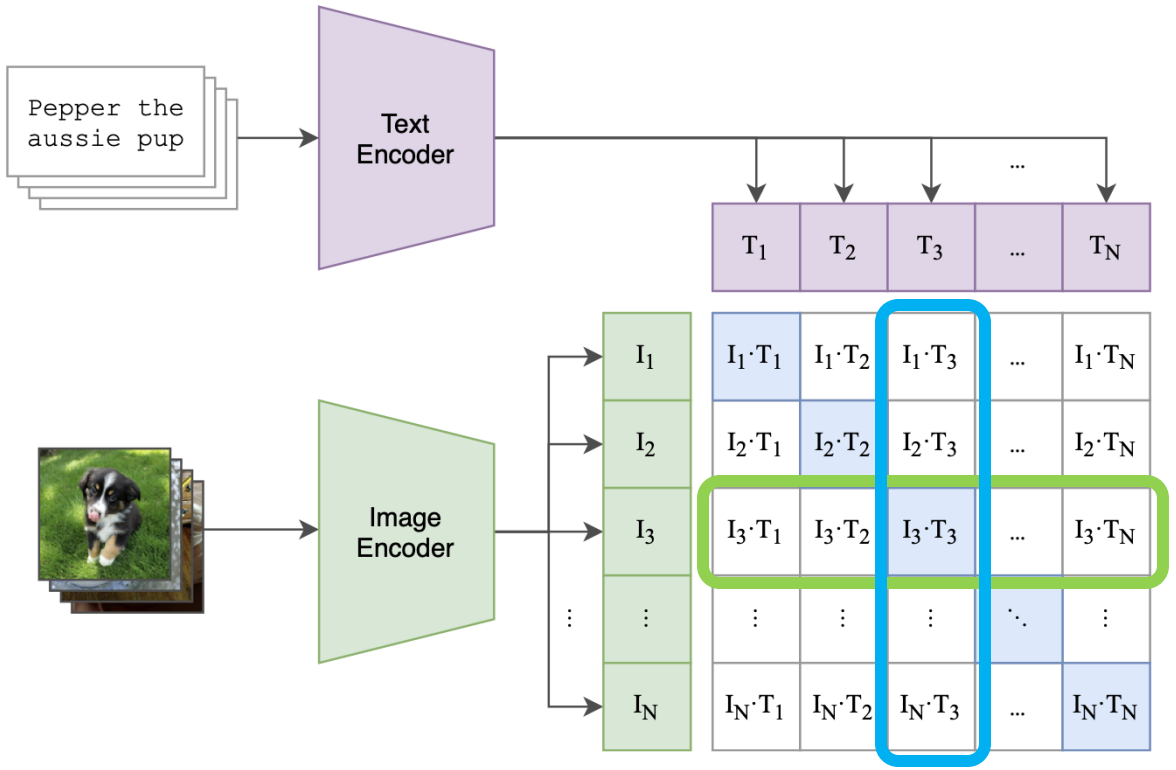


# Flamingo architecture

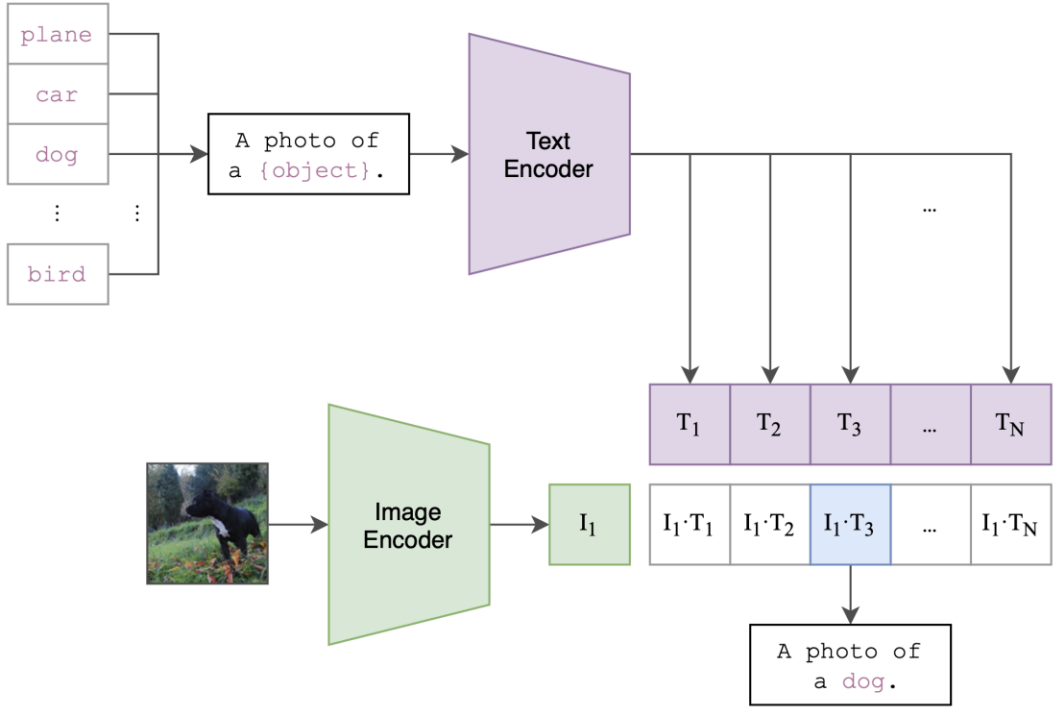


# CLIP [1]

Training




Zero-shot adaptation



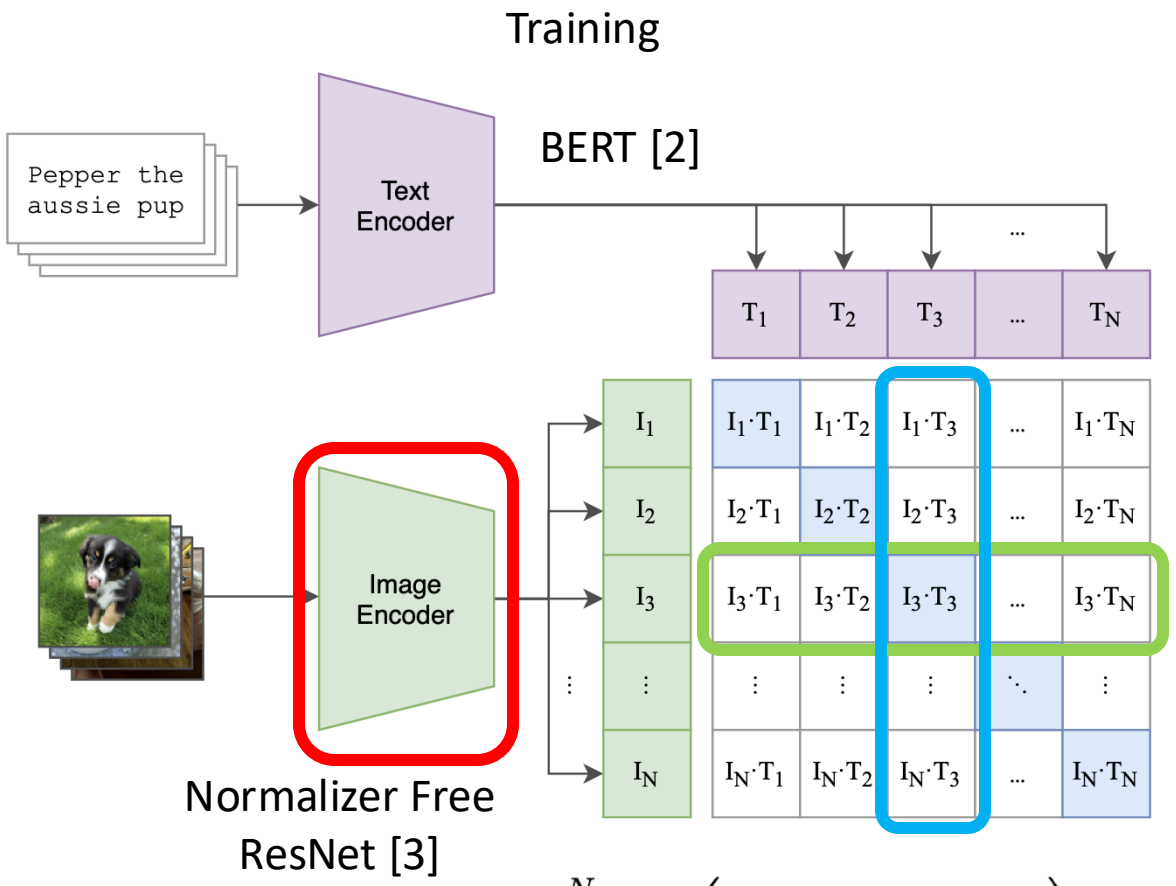

$$L_{contrastive:txt2im} = -\frac{1}{N}\sum_{i}^{N}\log\left(\frac{\exp(L_i^{\intercal}V_i\beta)}{\sum_j^N \exp(L_i^{\intercal}V_j\beta)}\right)$$

$$L_{contrastive:im2txt} = -\frac{1}{N}\sum_{i}^{N}\log\left(\frac{\exp(V_i^{\intercal}L_i\beta)}{\sum_j^N \exp(V_i^{\intercal}L_j\beta)}\right)$$



[1] Learning Transferable Visual Models From Natural Language Supervision, Radford et al., OpenAI, ICML 2021

# Flamingo Vision Encoder




$$L_{contrastive:txt2im} = -\frac{1}{N}\sum_{i}^{N}\log\left(\frac{\exp(L_i^\intercal V_i\beta)}{\sum_j^N \exp(L_i^\intercal V_j\beta)}\right)$$

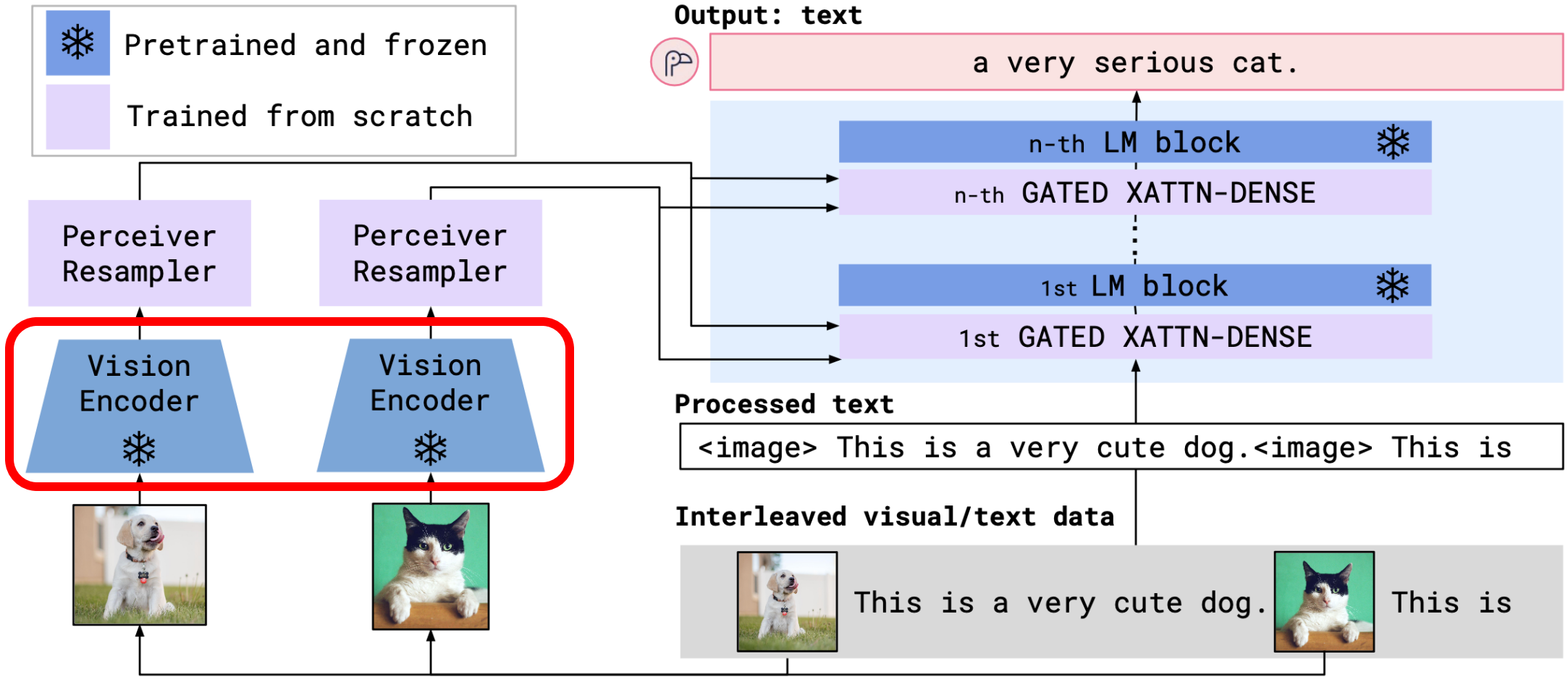
$$L_{contrastive:im2txt} = -\frac{1}{N}\sum_{i}^{N}\log\left(\frac{\exp(V_i^\intercal L_i\beta)}{\sum_j^N \exp(V_i^\intercal L_j\beta)}\right)$$

```
# image_encoder - ResNet or Vision Transformer
# text_encoder  - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]       - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t             - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T)  #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss   = (loss_i + loss_t)/2
```

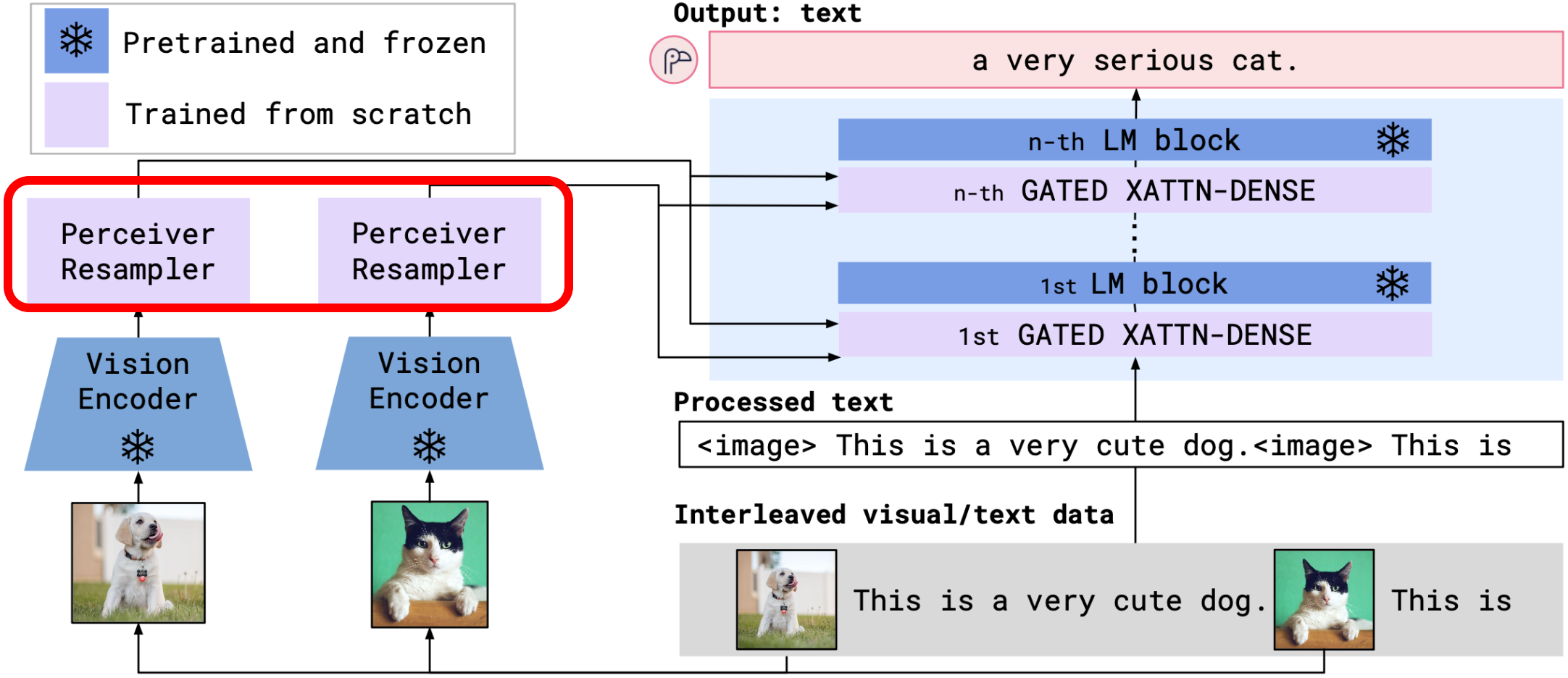
*Figure 3.* Numpy-like pseudocode for the core of an implementation of CLIP.

[2] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding , Devlin et al., Google, NAACL 2019
[3] High-performance large-scale image recognition without normalization, Brock et al., PMLR 2021

# Flamingo architecture

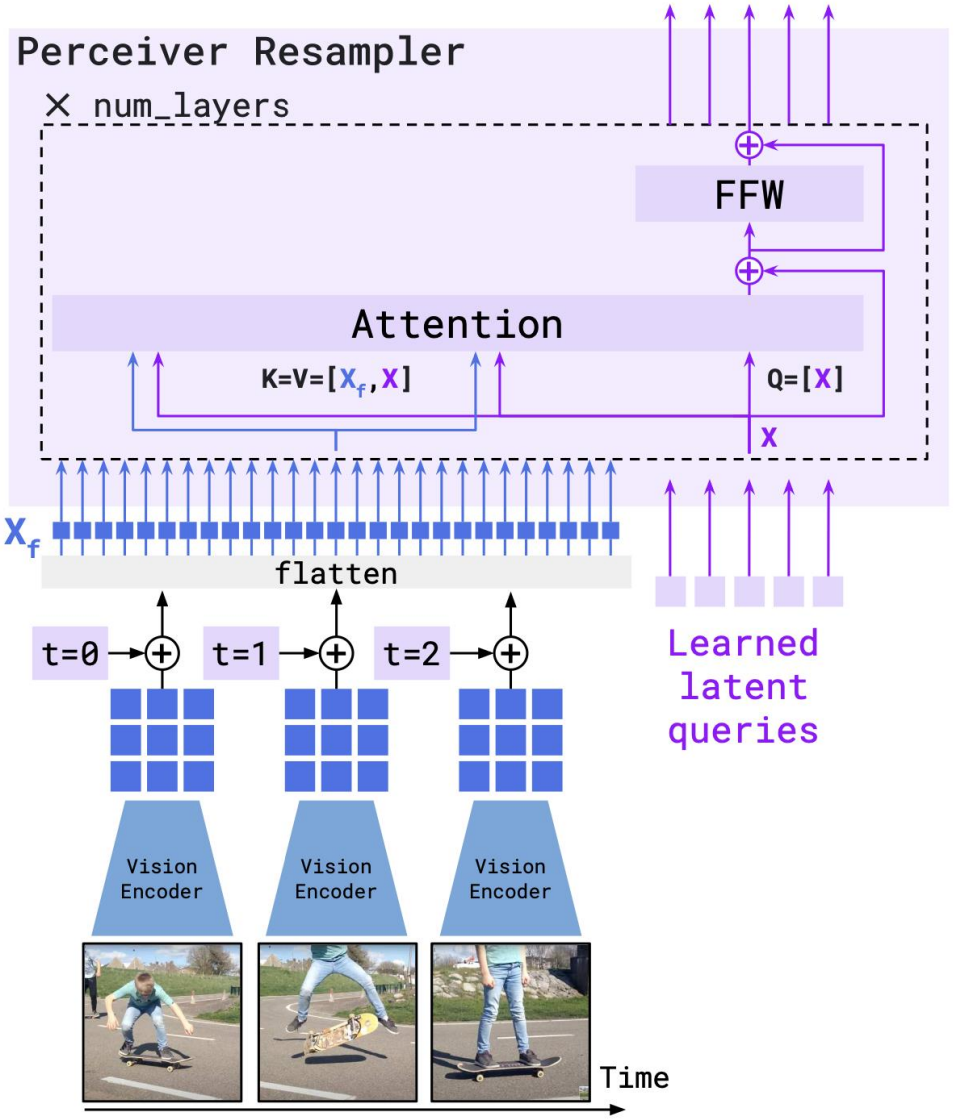


# Flamingo architecture



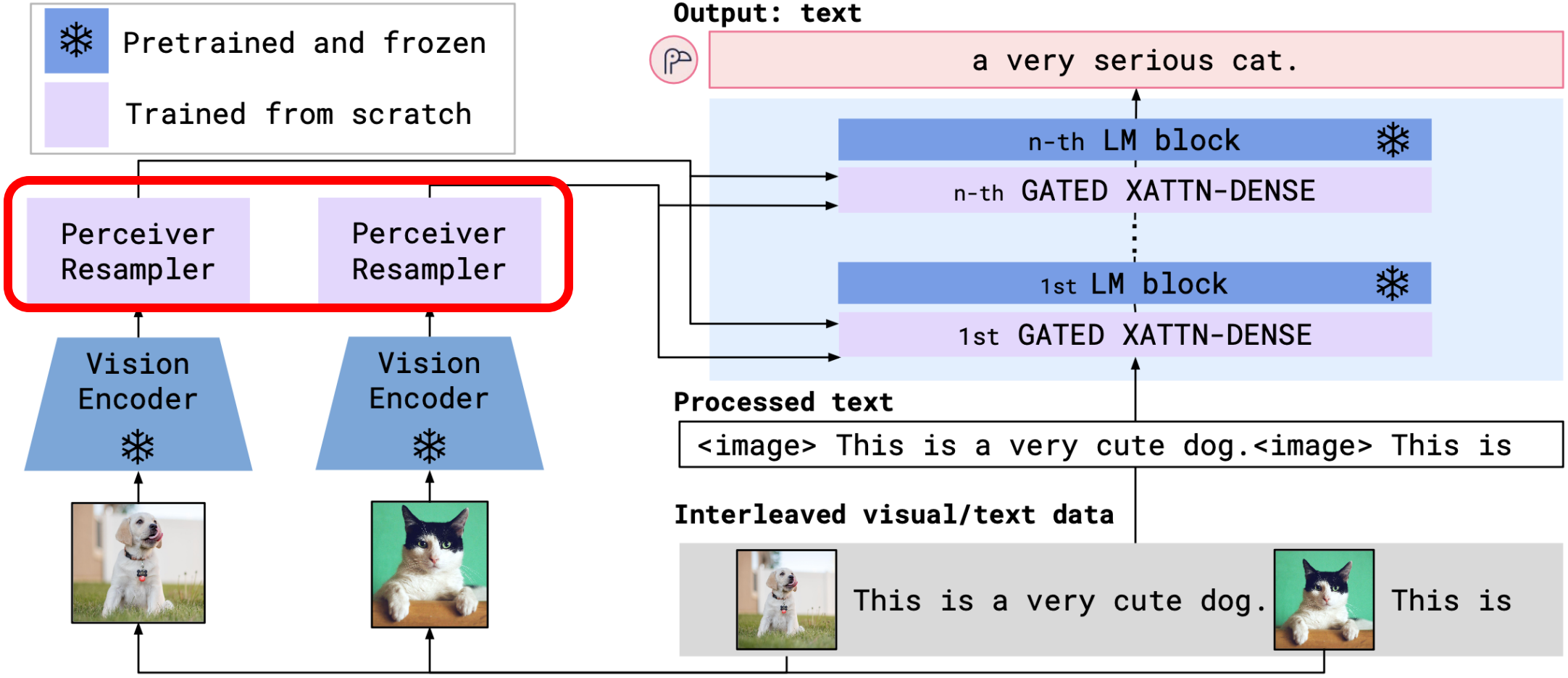
# Perceiver Resampler

**Constant number of tokens!**

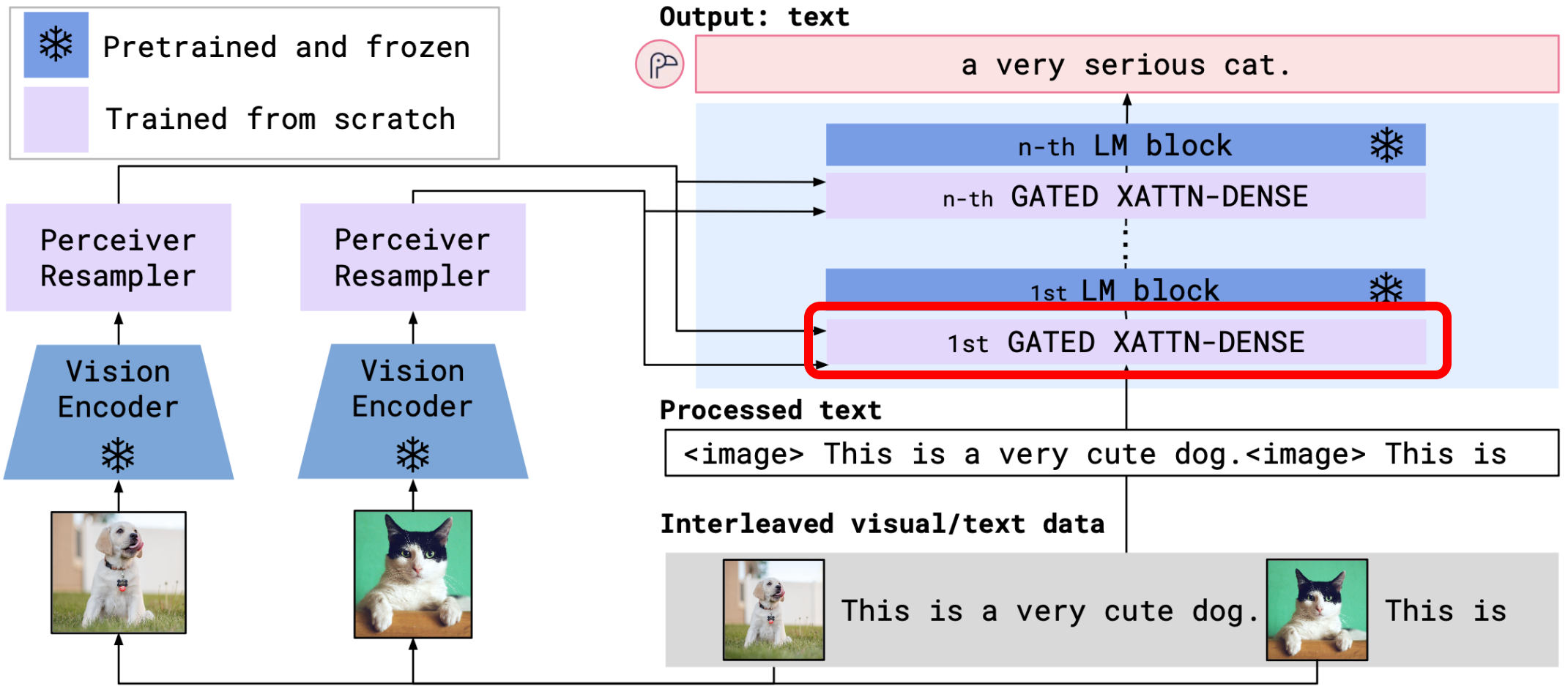



```
def perceiver_resampler(
    x_f,  # The [T, S, d] visual features (T=time, S=space)
    time_embeddings,  # The [T, 1, d] time pos embeddings.
    x,  # R learned latents of shape [R, d]
    num_layers,  # Number of layers
):
  """The Perceiver Resampler model."""

  # Add the time position embeddings and flatten.
  x_f = x_f + time_embeddings
  x_f = flatten(x_f)  # [T, S, d] -> [T * S, d]
  # Apply the Perceiver Resampler layers.
  for i in range(num_layers):
    # Attention.
    x = x + attention_i(q=x, kv=concat([x_f, x]))
    # Feed forward.
    x = x + ffw_i(x)
  return x
```

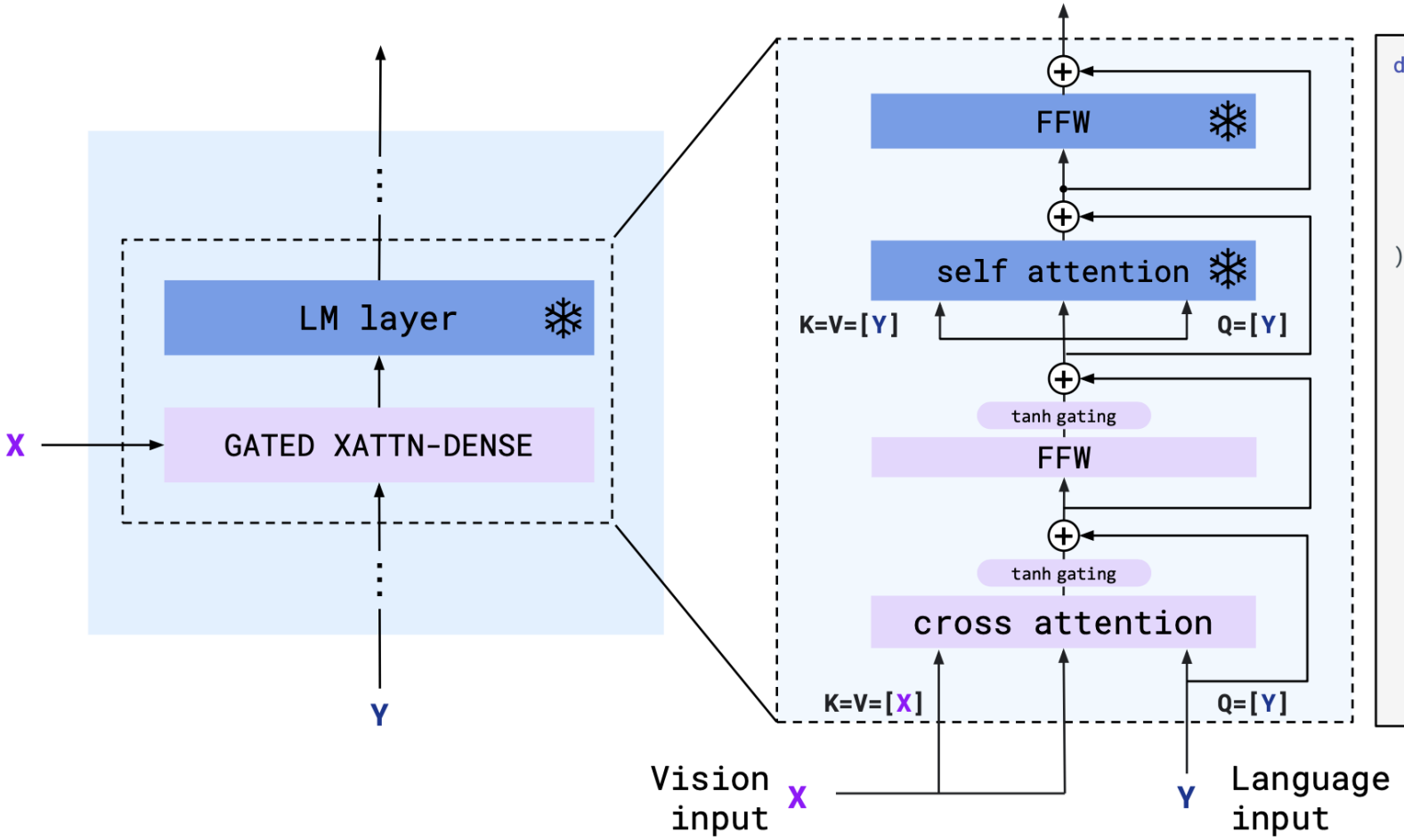
# Flamingo architecture



# Flamingo architecture

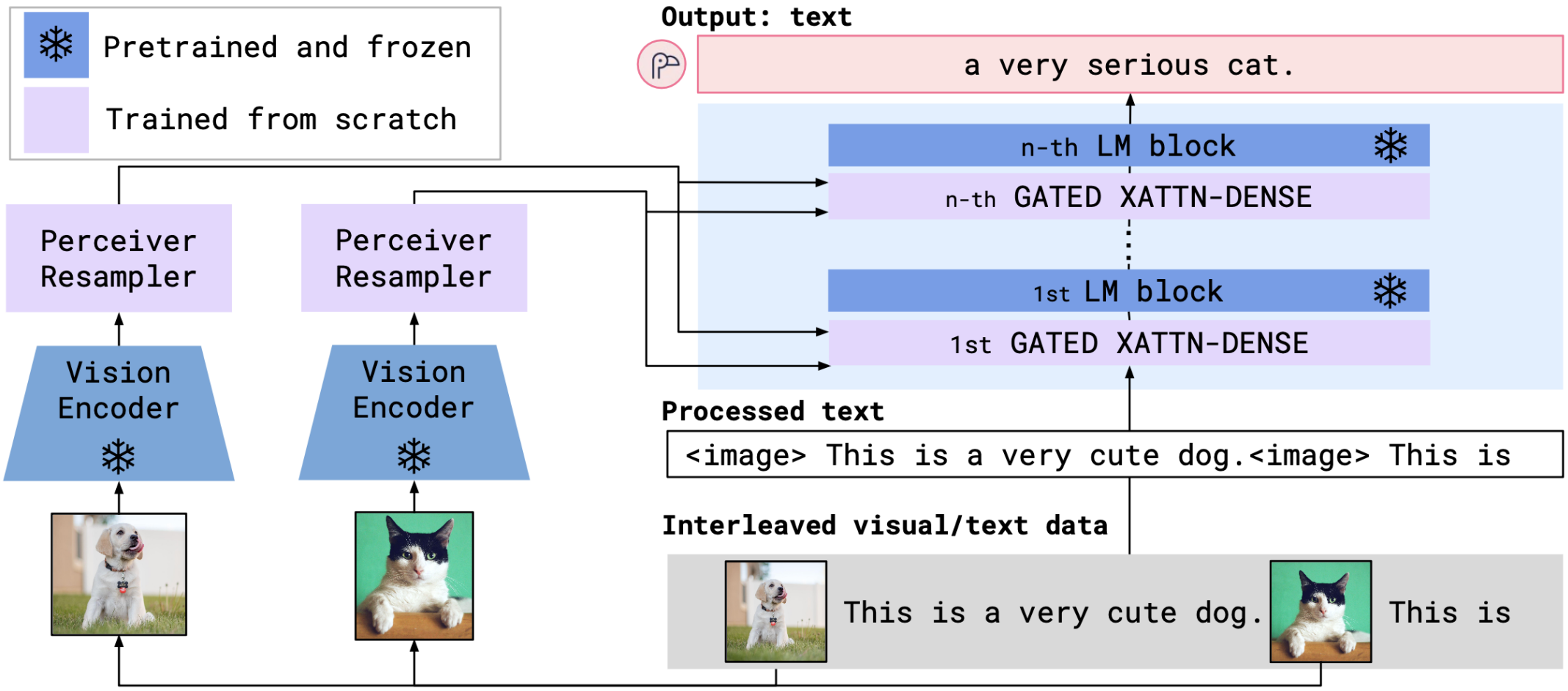


# Gated cross-attention dense




```
def gated_xattn_dense(
    y,  # input language features
    x,  # input visual features
    alpha_xattn, # xattn gating parameter – init at 0.
    alpha_dense, # ffw gating parameter – init at 0.
):
  """Applies a GATED XATTN-DENSE layer."""

  # 1. Gated Cross Attention
  y = y + tanh(alpha_xattn) * attention(q=y, kv=x)
  # 2. Gated Feed Forward (dense) Layer
  y = y + tanh(alpha_dense) * ffw(y)

  # Regular self-attention + FFW on language
  y = y + frozen_attention(q=y, kv=y)
  y = y + frozen_ffw(y)
  return y  # output visually informed language features
```

# Flamingo architecture



# Training data


Image-Text Pairs dataset
[N=1, T=1, H, W, C]


Video-Text Pairs dataset
[N=1, T>1, H, W, C]


Multi-Modal Massive Web (M3W) dataset
[N>1, T=1, H, W, C]

- ALIGN: 1.8 billion images with alt-text
- LTIP (Long Text & Image Pairs): their own dataset of 312 million pairs.

VTP (Video & Text Pairs): their own dataset of 27 million pairs of short videos (avg. 22s) and descriptions

MultiModal MassiveWeb (M3W) dataset: text and images from 43 million webpages.

# Training strategy

Given that:

| | |
|---|---|
| $y$ | denotes a text token |
| $x$ | denotes a visual token |
| $y_\ell$ | denotes the $l$-th text token of the input |
| $y_{<\ell}$ | denotes the set of preceding text tokens |
| $x_{\leq\ell}$ | denotes the set of preceding visual tokens |

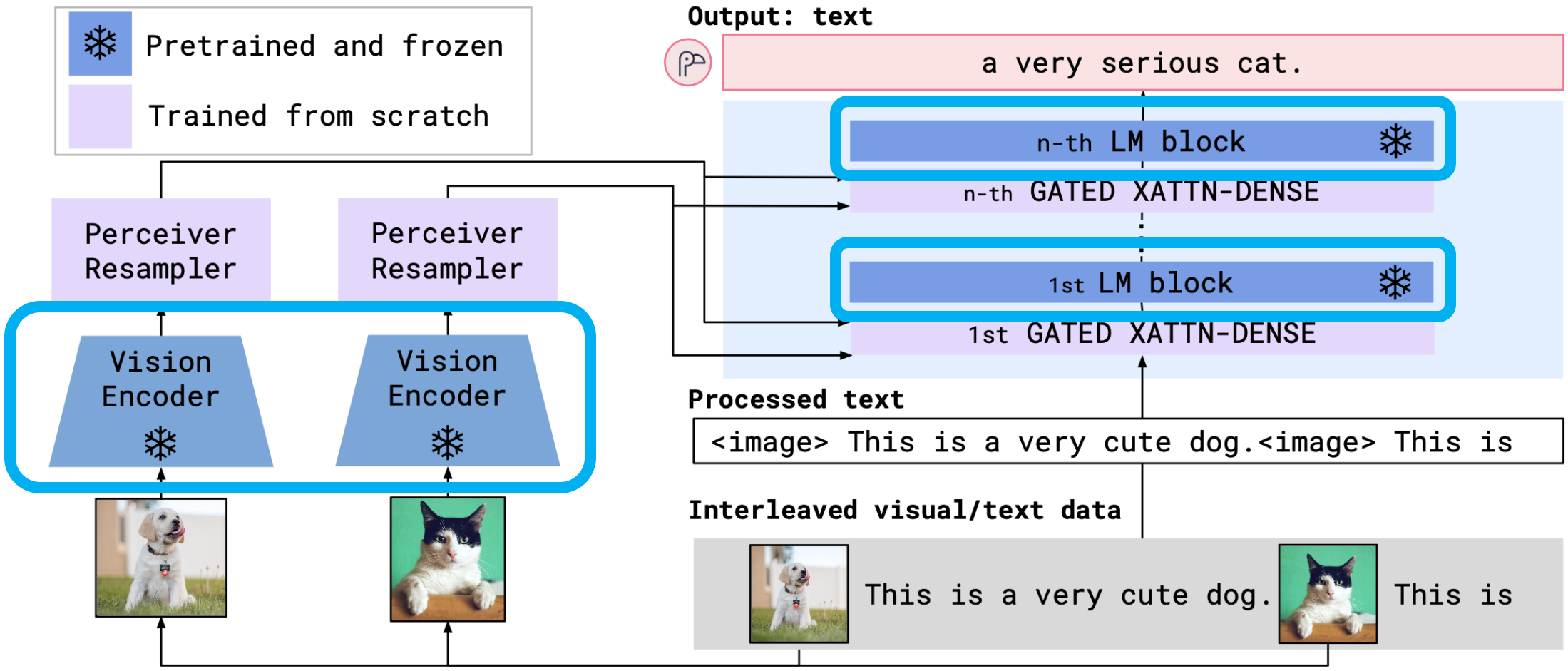
Flamingo output likelihood

$$p(y|x) = \prod_{\ell=1}^{L} p(y_\ell | y_{<\ell}, x_{\leq\ell})$$

Flamingo training loss

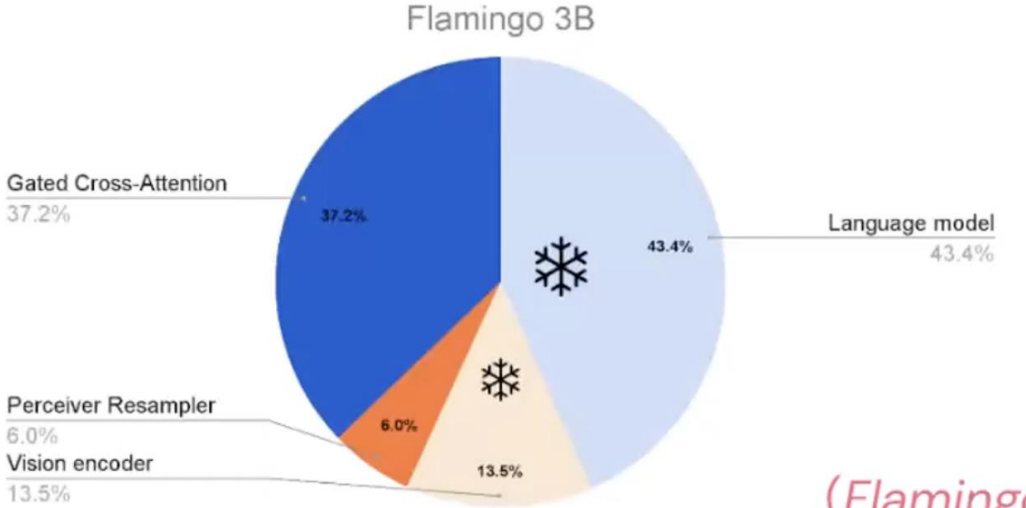
$$\sum_{m=1}^{M} \lambda_m \cdot \mathbb{E}_{(x,y)\sim\mathcal{D}_m} \left[ -\sum_{\ell=1}^{L} \log p(y_\ell | y_{<\ell}, x_{\leq\ell}) \right]$$

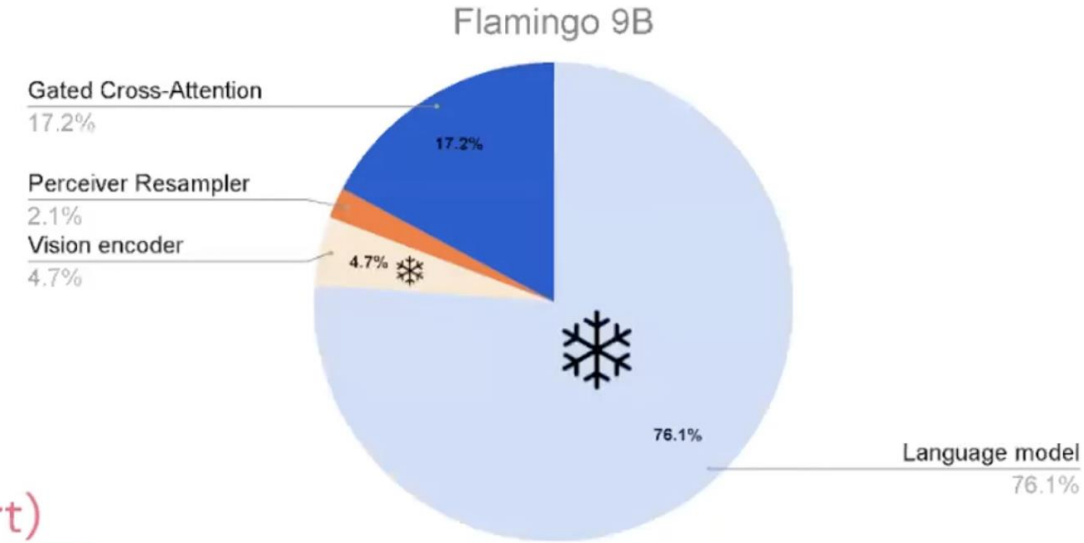
# Frozen models

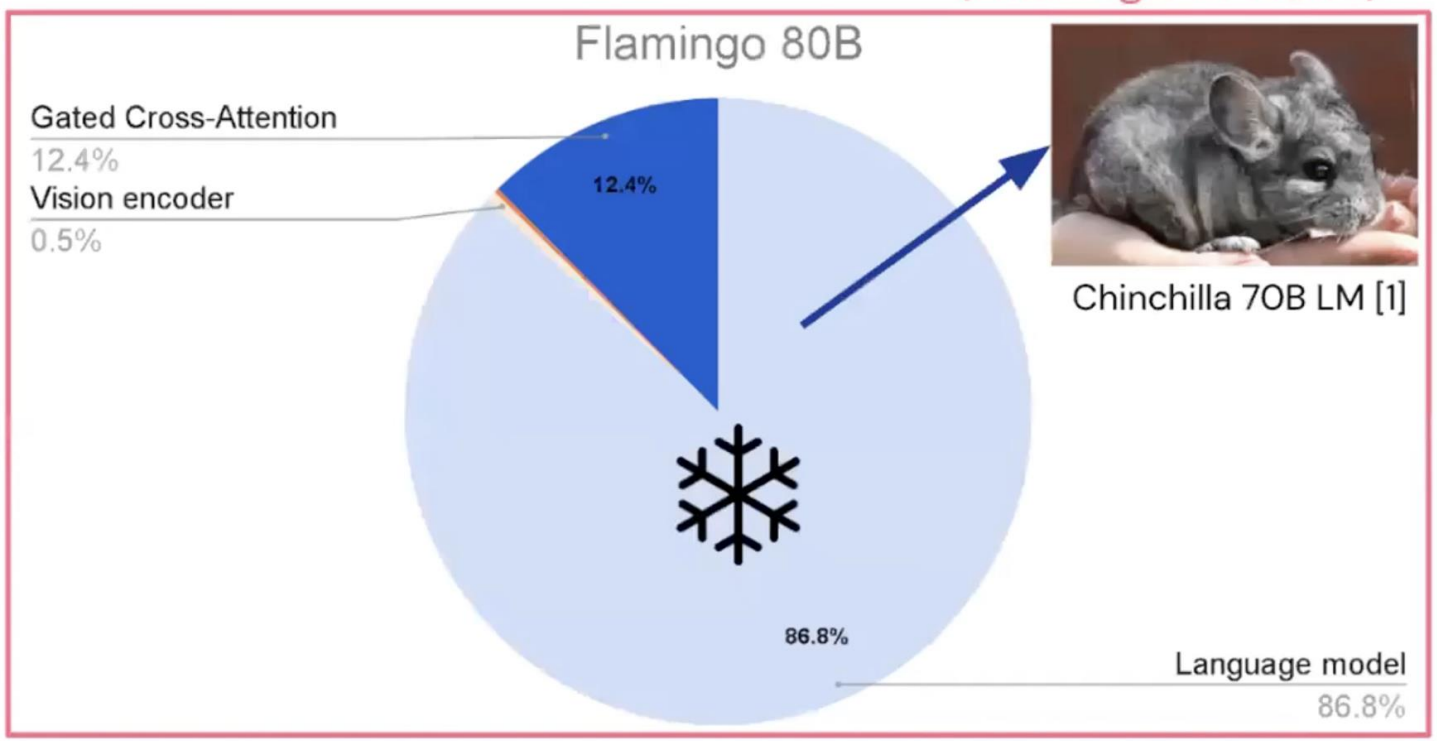


# The Flamingo family

(*Flamingo* in short)






Chinchilla 70B LM [1]

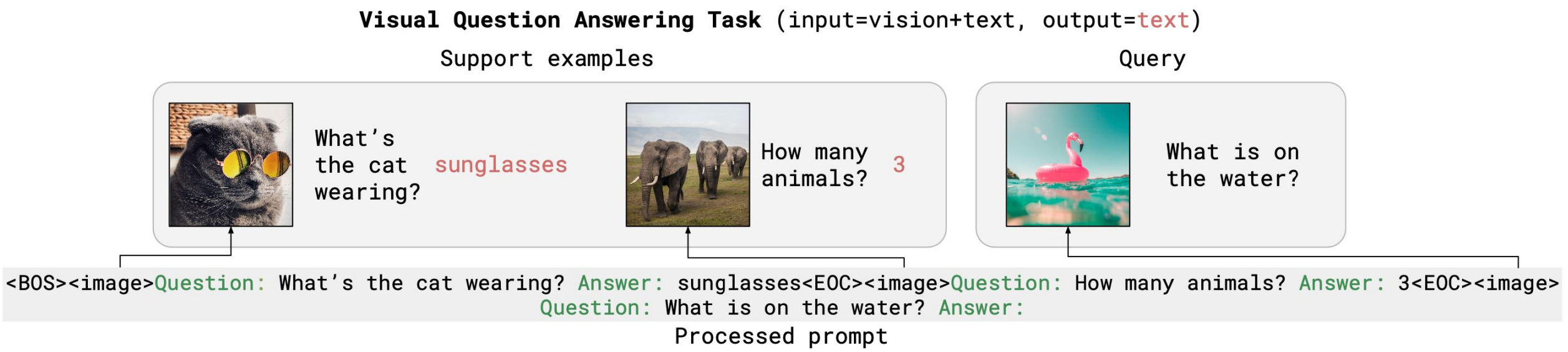
- Vision encoder (NFNet-F6) size fixed.
- Resampler size fixed.
- Focus on scaling the frozen language model.

[1] Hoffmann et al., Training Compute-Optimal Large Language Models, 2022

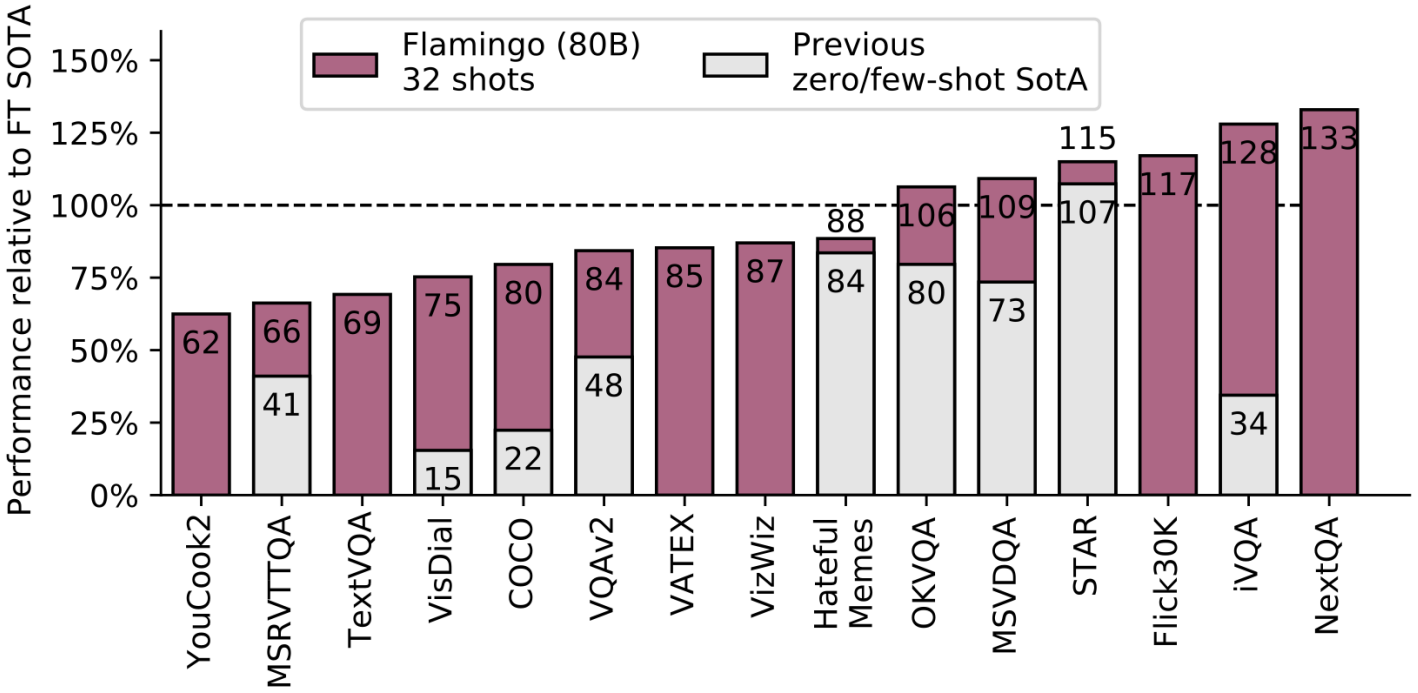
# Few-shot in-context learning

Model can be used for many different tasks:
Object classification, scene description, visual question answering (with/without external knowledge required), text reading, meme classification, action classification, etc.

Use the power of the LLM for in-context learning!



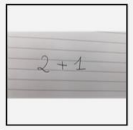
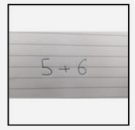
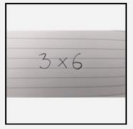
# Results!




- Better than 0/few-shot SOTA in 15 datasets
- Better than **any** fine-tuned model in 6 datasets

Performance scales with model size and, up to a certain limit, with provided examples

# Examples

| 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** |
|  | 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.** |

# Examples

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.

This is a cityscape. It looks like Chicago.

What makes you think this is Chicago?

I think it's Chicago because of the Shedd Aquarium in the background.

What about this one? Which city is this and what famous landmark helped you recognise the city?

This is Tokyo. I think it's Tokyo because of the Tokyo Tower.

# Examples

| | | | |
|---|---|---|---|
| Input Prompt |  Question: What do you think the capacities of these are? Answer: |  Question: What is odd about this image? Explain why it is unusual. Answer: |  Question: What country is this? Why do you think so? Answer: |
| Completion | **The floppy disk is 1.44MB and the CD is 700MB.** | **The image is odd because the elephant is in the back of the truck. It is unusual because elephants are not usually transported in the back of a truck.** | **It is Canada. I think so because the flag is the Canadian flag.** |

| | | | |
|---|---|---|---|
| Input Prompt |  Question: What nutrients is the fruit in the image rich in? Answer: |  Question: What is the purpose of this? Answer: |  Question: What season is this? Explain why. Answer: |
| Completion | **Vitamin C, potassium, and fiber.** | **To keep the mosquitoes out of the bed.** | **This is winter season. The Christmas tree and the presents are the main reasons why it is winter.** |

| | | | |
|---|---|---|---|
| Input Prompt |  Question: What is the title of the book? Answer: |  The ingredients of this dish are |   The soundtrack includes |
| Completion | **The House Book.** | **: beef, potatoes, carrots, broccoli, and lemon.** | **the songs "Let It Go" and "For the First Time in Forever" by Tony Award® winner Idina Menzel, who plays Elsa [...]** |

# Examples

This is a dog. It's a white fluffy dog.

What season is this?

It looks like a summer day.

Why do you think so?

The grass is green and the dog is not wearing a sweater.

What breed is the dog?

It's a Samoyed.

Is the dog running?

No, it's sitting.

Can you describe the pose of its back legs?

The back legs are bent and the dog is sitting on its haunches.


Which city is this?

This is a street sign in New York City.

Which street corner is it?

It is the intersection of Madison Avenue and East 42nd Street.

Which subway station is nearby?

The closest subway station is Grand Central Station.

How about the nearest park?

The nearest park is Bryant Park.

There is another sign. What does it say?

It says "One way".

I would like you to read the color and then say the color it is written in. For example:

PURPLE

Color is "Purple" and it is written in red.

GREEN

Color is "Green" and it is written in blue.

YELLOW

Color is "Yellow" and it is written in green.

BLACK

Color is "Black" and it is written in yellow.

Well done! Do you know the name of the test these images come from?

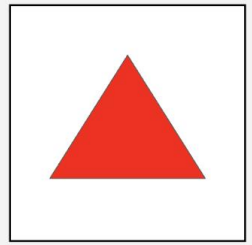
I think it is called the Stroop test.

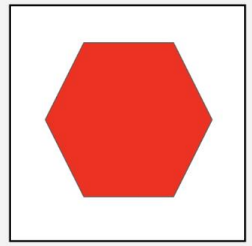
Can you explain how humans perform in this test?

Humans are slower when the color of the word and the color of the word are different.

How about you?

I am not affected by this difference.





What is the difference between these two images?

The first one is a triangle and the second one is a hexagon.

Which one has larger internal angles?

The hexagon has larger internal angles.

What color are they?

They are both red.

# Examples

What is in this picture?

It's a bowl of soup with a monster face on it.

What is the monster made out of?

It's made out of vegetables.

No, it's made out of a kind of fabric. Can you see what kind?

It's made out of a woolen fabric.

This is a picture of a dog. It looks like it is a little bit naughty.

What happened here?

It looks like the dog is destroying the sofa.

How does the person thinking about it?

The person is angry.

What is the person holding?

It looks like a cup of coffee.

How is the weather outside?

It looks like a sunny day.


What is the name of the company on this card?

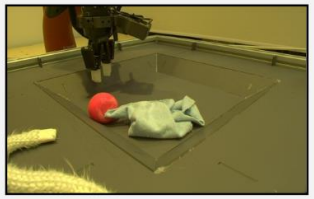
It's Homebase.

What is its web url?

It's homebase.co.uk.

What are they doing?

They are a home improvement retailer.



What do you think the robot is doing here?

The robot is picking up a ball.

What color is the ball?

It's red.

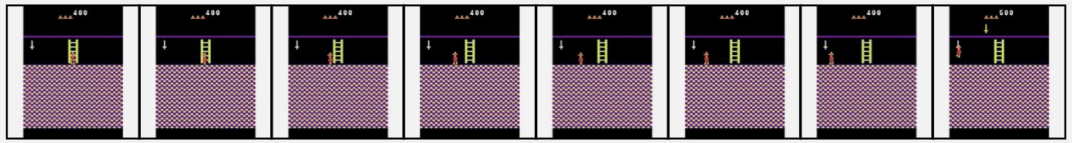
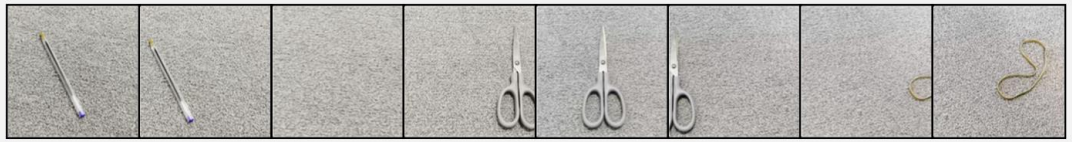
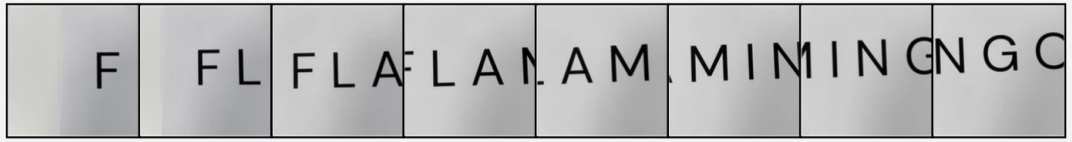
I want the ball to be hidden. Is there something the robot can use for that?

The robot can use a cloth to hide the ball.

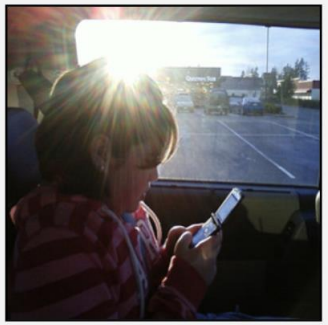
What color is the cloth?

The cloth is blue.

# Examples: videos

| Input Prompt | Completion |
| --- | --- |
|  Question: What is happening here? Answer: | The dachschund puppy is being weighed on a scale. |
|  Question: What object is the avatar picking up? Answer: | A sword. |
|  Question: What is the object being lifted? Answer: | The object is a small plastic bowl. |
|  Question: What are the three objects in this video? Answer: | A scissors, a pen, and a rubber band. |
|  Question: What is written here? Answer: | Flamingo. |
|  What happens to the man after hitting the ball? Answer: | he falls down. |

# Failure cases

| | | | |
|---|---|---|---|
| Input Prompt |  Question: What is on the phone screen? Answer: |  Question: What can you see out the window? Answer: |  Question: Whom is the person texting? Answer: |
| Output | **A text message from a friend.** | **A parking lot.** | **The driver.** |

# Limitations

1. Limitations from language models, such as hallucinations

2. Classification performance is lower than contrastive methods, like CLIP

3. In-context learning has both advantages and disadvantages:
   - + no gradient required
   - + easy for the end-user
   - - limited to few examples
   - - sensitive to example choice and order
   - - poor inference compute scaling

# Extra: ablation 1

| | Ablated setting | *Flamingo*-3B original value | Changed value | Param. count ↓ | Step time ↓ | COCO CIDEr↑ | OKVQA top1↑ | VQAv2 top1↑ | MSVDQA top1↑ | VATEX CIDEr↑ | Overall score↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ***Flamingo*-3B model** | | 3.2B | 1.74s | 86.5 | 42.1 | 55.8 | 36.3 | 53.4 | **70.7** |
| **(i)** | Training data | All data | w/o Video-Text pairs | 3.2B | 1.42s | 84.2 | 43.0 | 53.9 | 34.5 | 46.0 | 67.3 |
| | | | w/o Image-Text pairs | 3.2B | 0.95s | 66.3 | 39.2 | 51.6 | 32.0 | 41.6 | 60.9 |
| | | | Image-Text pairs→ LAION | 3.2B | 1.74s | 79.5 | 41.4 | 53.5 | 33.9 | 47.6 | 66.4 |
| | | | w/o M3W | 3.2B | 1.02s | 54.1 | 36.5 | 52.7 | 31.4 | 23.5 | 53.4 |
| **(ii)** | Optimisation | Accumulation | Round Robin | 3.2B | 1.68s | 76.1 | 39.8 | 52.1 | 33.2 | 40.8 | 62.9 |
| **(iii)** | Tanh gating | ✓ | ✗ | 3.2B | 1.74s | 78.4 | 40.5 | 52.9 | 35.9 | 47.5 | 66.5 |
| **(iv)** | Cross-attention architecture | GATED XATTN-DENSE | VANILLA XATTN | 2.4B | 1.16s | 80.6 | 41.5 | 53.4 | 32.9 | 50.7 | 66.9 |
| | | | GRAFTING | 3.3B | 1.74s | 79.2 | 36.1 | 50.8 | 32.2 | 47.8 | 63.1 |
| **(v)** | Cross-attention frequency | Every | Single in middle | 2.0B | 0.87s | 71.5 | 38.1 | 50.2 | 29.1 | 42.3 | 59.8 |
| | | | Every 4th | 2.3B | 1.02s | 82.3 | 42.7 | 55.1 | 34.6 | 50.8 | 68.8 |
| | | | Every 2nd | 2.6B | 1.24s | 83.7 | 41.0 | 55.8 | 34.5 | 49.7 | 68.2 |
| **(vi)** | Resampler | Perceiver | MLP | 3.2B | 1.85s | 78.6 | 42.2 | 54.7 | 35.2 | 44.7 | 66.6 |
| | | | Transformer | 3.2B | 1.81s | 83.2 | 41.7 | 55.6 | 31.5 | 48.3 | 66.7 |
| **(vii)** | Vision encoder | NFNet-F6 | CLIP ViT-L/14 | 3.1B | 1.58s | 76.5 | 41.6 | 53.4 | 33.2 | 44.5 | 64.9 |
| | | | NFNet-F0 | 2.9B | 1.45s | 73.8 | 40.5 | 52.8 | 31.1 | 42.9 | 62.7 |
| **(viii)** | Freezing LM | ✓ | ✗ (random init) | 3.2B | 2.42s | 74.8 | 31.5 | 45.6 | 26.9 | 50.1 | 57.8 |
| | | | ✗ (pretrained) | 3.2B | 2.42s | 81.2 | 33.7 | 47.4 | 31.0 | 53.9 | 62.7 |

# Extra: ablation 2

| | Ablated setting | Flamingo 3B value | Changed value | Param. count ↓ | Step time ↓ | COCO CIDEr↑ | OKVQA top1↑ | VQAv2 top1↑ | MSVDQA top1↑ | VATEX CIDEr↑ | Overall score↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Flamingo 3B model (short training)** | | | 3.2B | 1.74s | 86.5 | 42.1 | 55.8 | 36.3 | 53.4 | **70.7** |
| **(i)** | Resampler size | Medium | Small | 3.1B | 1.58s | 81.1 | 40.4 | 54.1 | 36.0 | 50.2 | 67.9 |
| | | | Large | 3.4B | 1.87s | 84.4 | 42.2 | 54.4 | 35.1 | 51.4 | 69.0 |
| **(ii)** | Multi-Img att. | Only last | All previous | 3.2B | 1.74s | 70.0 | 40.9 | 52.0 | 32.1 | 46.8 | 63.5 |
| **(iii)** | $p_{next}$ | 0.5 | 0.0 | 3.2B | 1.74s | 85.0 | 41.6 | 55.2 | 36.7 | 50.6 | 69.6 |
| | | | 1.0 | 3.2B | 1.74s | 81.3 | 43.3 | 55.6 | 36.8 | 52.7 | 70.4 |
| **(iv)** | LM pretraining | MassiveText | C4 | 3.2B | 1.74s | 81.3 | 34.4 | 47.1 | 60.6 | 53.9 | 62.8 |
| **(v)** | Freezing Vision | ✓ | ✗ (random init) | 3.2B | 4.70s* | 74.5 | 41.6 | 52.7 | 31.4 | 35.8 | 61.4 |
| | | | ✗ (pretrained) | 3.2B | 4.70s* | 83.5 | 40.6 | 55.1 | 34.6 | 50.7 | 68.1 |
| **(vi)** | Co-train LM on MassiveText | ✗ | ✓ (random init) | 3.2B | 5.34s* | 69.3 | 29.9 | 46.1 | 28.1 | 45.5 | 55.9 |
| | | | ✓ (pretrained) | 3.2B | 5.34s* | 83.0 | 42.5 | 53.3 | 35.1 | 51.1 | 68.6 |
| **(vii)** | Dataset and Vision encoder | M3W+ITP+VTP and NFNetF6 | LAION400M and CLIP | 3.1B | 0.86s | 61.4 | 37.9 | 50.9 | 27.9 | 29.7 | 54.7 |
| | | | M3W+LAION400M+VTP and CLIP | 3.1B | 1.58s | 76.3 | 41.5 | 53.4 | 32.5 | 46.1 | 64.9 |