

Multimodal models

CS 4804: Introduction to AI

Fall 2025

<https://tuvllms.github.io/ai-fall-2025/>

Tu Vu



Logistics

- Class canceled this Thursday
- Homework 1 & Quiz 1 will be released today
 - Homework 1 due in 3 weeks
 - Quiz 1 due in 1 week
- Final Project proposal (due 10/7)
 - Project research ideas
 - Check out [GPT-5 System Card](#) & [Gemini 2.5 Pro Technical Report](#)
 - New benchmarks
 - New prompting methods
 - Improving small models

...

Deep Research

What are you researching?

Get a detailed report

+ Research Sources ▾

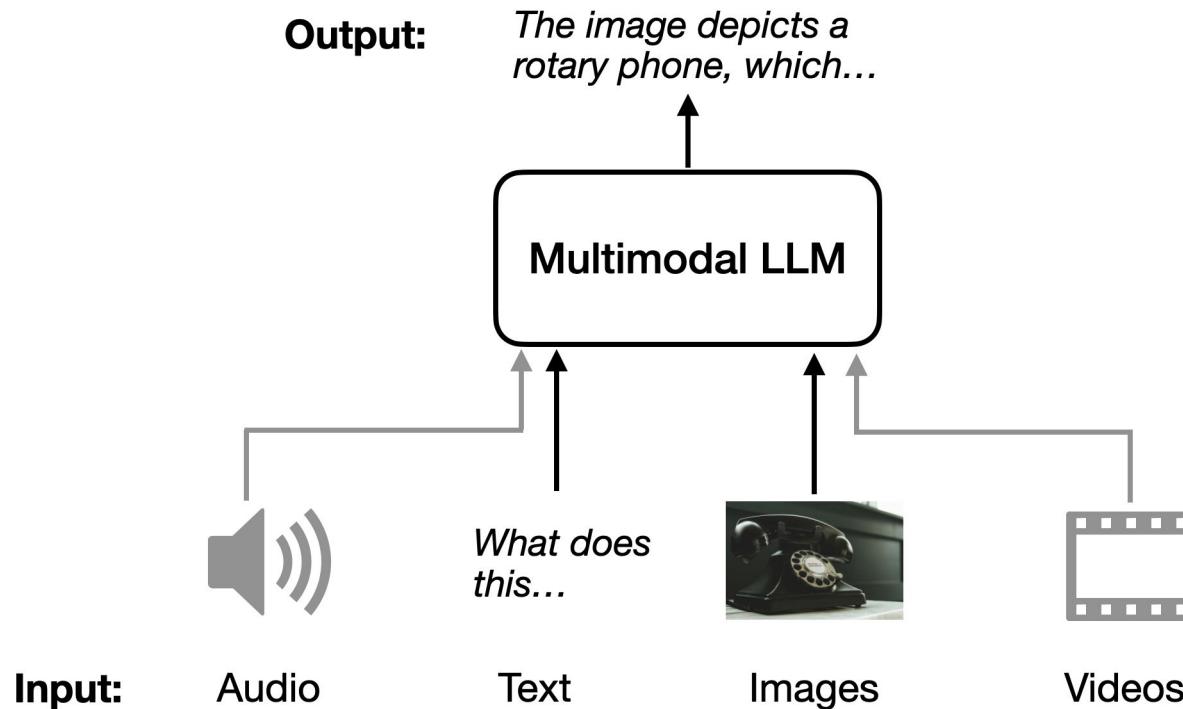
+

- Add photos & files
- Deep research
- Create image
- Agent mode
- Use connectors
- ... More >

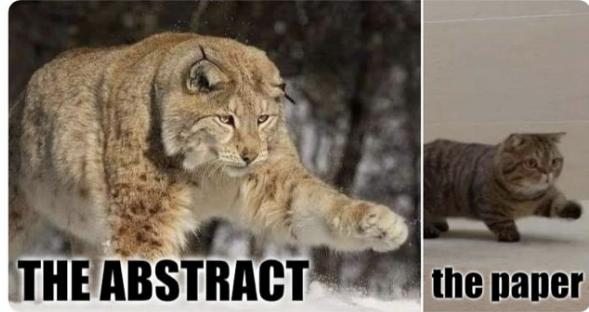
0

Help

A multimodal LLM can accept inputs in multiple modalities and emit text outputs



Use cases of multimodal LLMs



Explain the meme

This meme humorously captures the common experience many researchers and readers encounter when reading academic papers. It shows a powerful, fierce lynx labeled "THE ABSTRACT" on the left and a smaller, cautious-looking domestic cat labeled "the paper" on the right.

The joke here is that an abstract (the brief summary at the beginning of a paper) often makes the research sound exciting, groundbreaking, or impactful, like a "fierce lynx." However, the actual content of the paper may feel less impressive, or overly complex, dense, or even disappointing by comparison, represented by the "scaredy-cat" on the right. This is a playful take on how expectations can differ from reality in academic writing.

Use cases of multimodal LLMs (cont'd)

<p>(a) User:</p>  <p><code>{ "smartphone": 1, "teddy bear": 1 }</code></p>  <p><code>{ "cat": 3 }</code></p>  <p><code>{ "book": 3, "vase": 1, "glass": 1 }</code></p> 	MM1-30B (Ours): <code>{ "dog": 2, "frisbee": 1 }</code>
<p>(b) User:</p>  <p>Red circle: "no parking anytime"</p>  <p>Red circle: "Raffaello"</p>  <p>Red circle: "Rue Saint-Paul"</p>  <p>Red circle: "Hyde Park"</p>	MM1-30B (Ours): <code>"Hyde Park"</code>
<p>(c) User:</p>  <p>furniture: bed frame, weight: 50 and 150 pounds (23 to 68 kg)</p>  <p>furniture: sofa, weight: 100 to 200 pounds (45 to 91 kg)</p>  <p>furniture: stove, weight: 150 to 300 pounds (68 to 136 kg)</p>  <p>furniture: refrigerator, weight: 200 to 300 pounds (91 to 136 kg)</p>	MM1-30B (Ours): <code>refrigerator, weight: 200 to 300 pounds (91 to 136 kg)</code>
<p>(d) User:</p>   <p>total: $1 + 3 = 4$</p>   <p>total: $6 + 4 = 10$</p>   <p>total: $4 + 1 = 5$</p>	MM1-30B (Ours): <code>total: 4 + 1 = 5</code>

Fig. 1: MM1 can perform in-context predictions thanks to its large-scale multimodal pre-training. This allows MM1 to (a) count objects and follow custom formatting, (b) refer to parts of the images and perform OCR, (c) demonstrate common-sense and word knowledge about everyday objects, and (d) perform basic math functions. Images are from the COCO 2014 validation set [72].

Use cases of multimodal LLMs (cont'd)

User:

Photo: <img1> Menu: <img2> How much should I pay for all the beer on the table according to the price on the menu?



Wine & Beers			
	White	Rose	Sparkling
White			
Principessa Gava, Cava	15.48		
Sherman Chardonnay, Sicily	17.43	Coppice rosanera, Campania	16.89
Bucci Verdicchio Del Celere D'Isoli	15.09		
Reds			
San Felice, Chianti Classico, Tuscany	19.50		
Edo di Jesi, Piceno, Marche	16.42	Medalla	5
Sherman, Nero D'Avola, Sicily	17.43	Milagro	6
			Presidente
			6
		Pane	

MM1-30B-Chat (Ours): 12

Emu-Chat-37B: According to the menu, the price for all the beer on the table is 15.99.

LLaVA-NeXT-34B: The image shows a menu with beer prices, but it's not entirely clear which specific beers are on the table. However, I can provide you with an estimate based on the visible prices. From the menu, it appears that the prices for beer are as follows: Magna: 5, Medalla: 5, Heineken: 6 Presidente: 6

Assuming that the beers on the table are Magna, Medalla, Heineken, and Presidente, and there are two of each, you would pay:

$2 \times \text{Magna: } 2 \times 5 = 10, 2 \times \text{Medalla: } 2 \times 5 = 10, 2 \times \text{Heineken: } 2 \times 6 = 12, 2 \times \text{Presidente: } 2 \times 6 = 12$ Total: $10 + 10 + 12 + 12 = 44$

Please note that this is an estimate based on the visible prices and the assumption that there are two of each type of beer. The actual cost may vary if the beers on the table are not the ones listed on the menu or if there are different quantities.

User: Explain why.

MM1-30B-Chat (Ours): There are 2 beers on the table, and each beer costs 6 according to the price on the menu. So, $2 \times 6 = 12$.

Emu-Chat-37B: The price for all the beer on the table is 15.99 because each beer has a price of 10.99, and there are a total of five beers on the table. When you multiply the number of beers (5) by the price per beer (10.99), you get a total cost of 54.95 (rounded up to 15.99).

LLaVA-NeXT-34B: The image shows a menu with beer prices, but it's not entirely clear which specific beers are on the table. However, I can provide you with an estimate based on the visible prices. From the menu, it appears that the prices for beer are as follows: (...)

Fig. 2: MM1 can follow instructions and reason across images. Example and images from VILA [71]; VILA answers correctly when prompted with chain-of-thought.

The resolution curse

384x384

*What type of
shop is in the
image?*



The resolution curse

384x384

*What type of
shop is in the
image?*



Grayscale images



0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

Color images

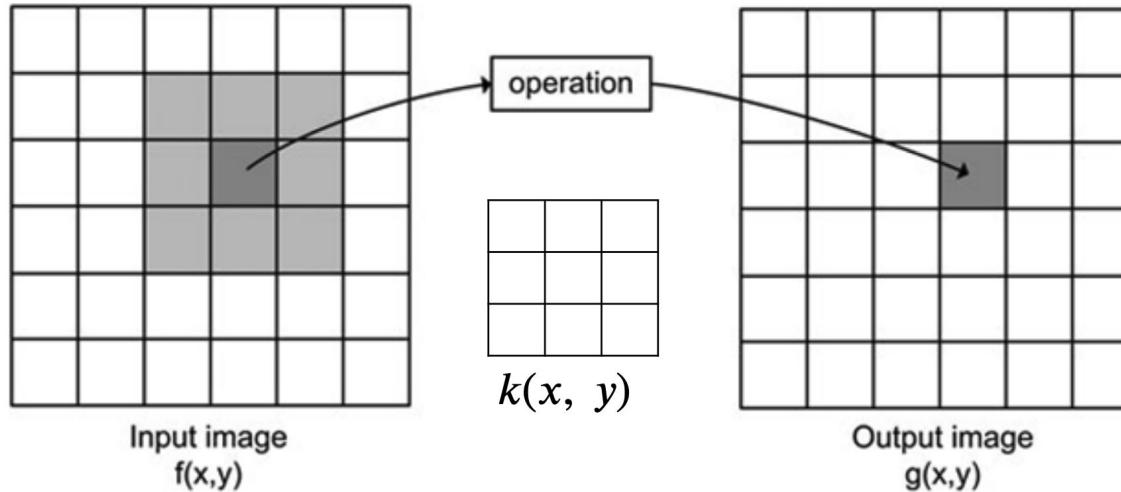


0	2	0	5	1	7	6	0	8
3	2	0	5	4	7	6	9	8
2	3	0	1	2	3	4	5	6
5	2	1	0	3	2	5	4	7
4	5	2	3	0	1	2	3	4
7	4	3	2	1	0	3	2	5
6	7	4	5	2	3	0	1	2
9	6	5	4	3	2	1	0	3
8	9	6	7	4	5	2	3	0
	8	7	6	5	4	3	2	1

channel x height x width

Channels are usually RGB: Red, Green, and Blue

Convolution operator

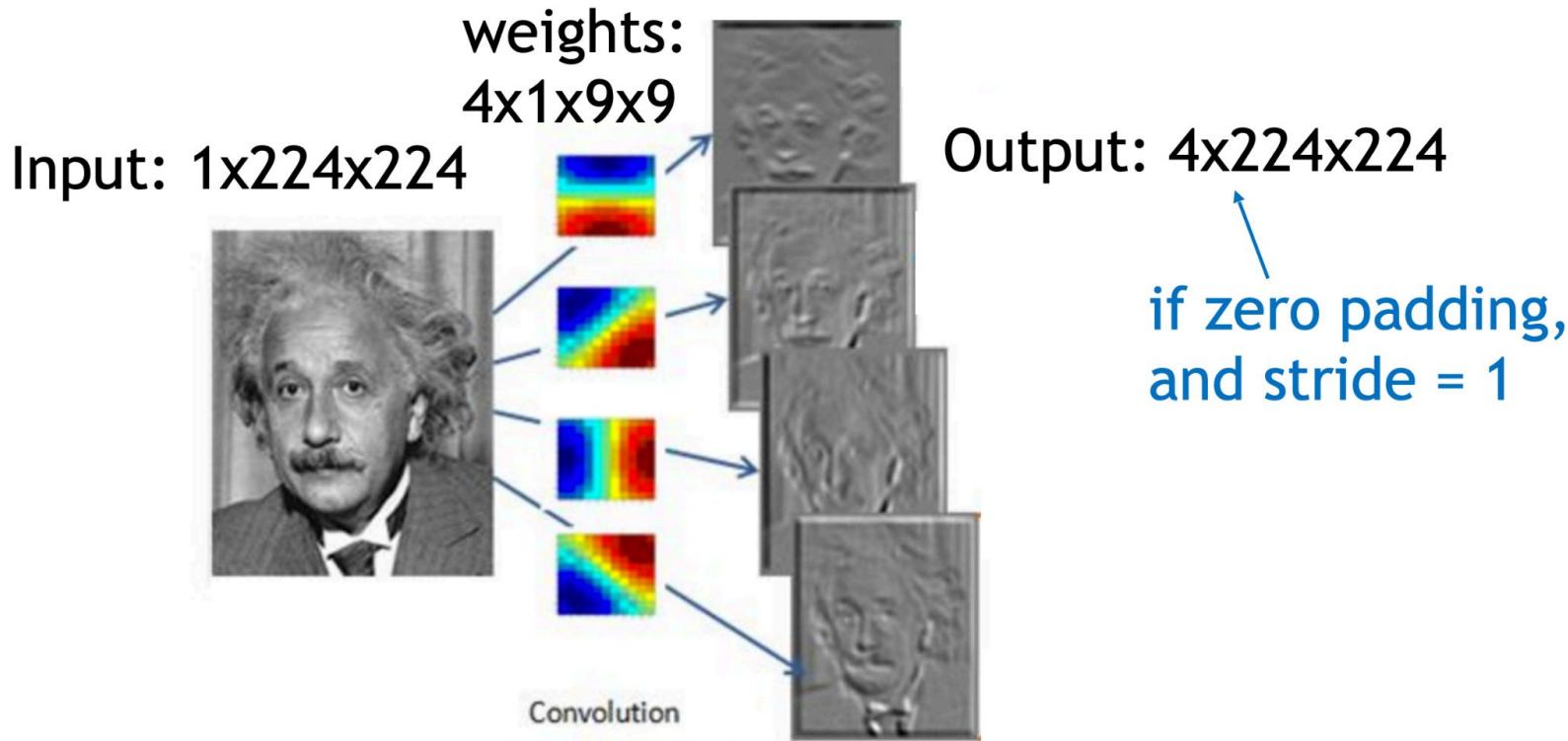


$$g(x, y) = \sum_v \sum_u k(u, v) f(x - u, y - v)$$

Demo

- <https://setosa.io/ev/image-kernels/>

Convolutional layer (with 4 filters)



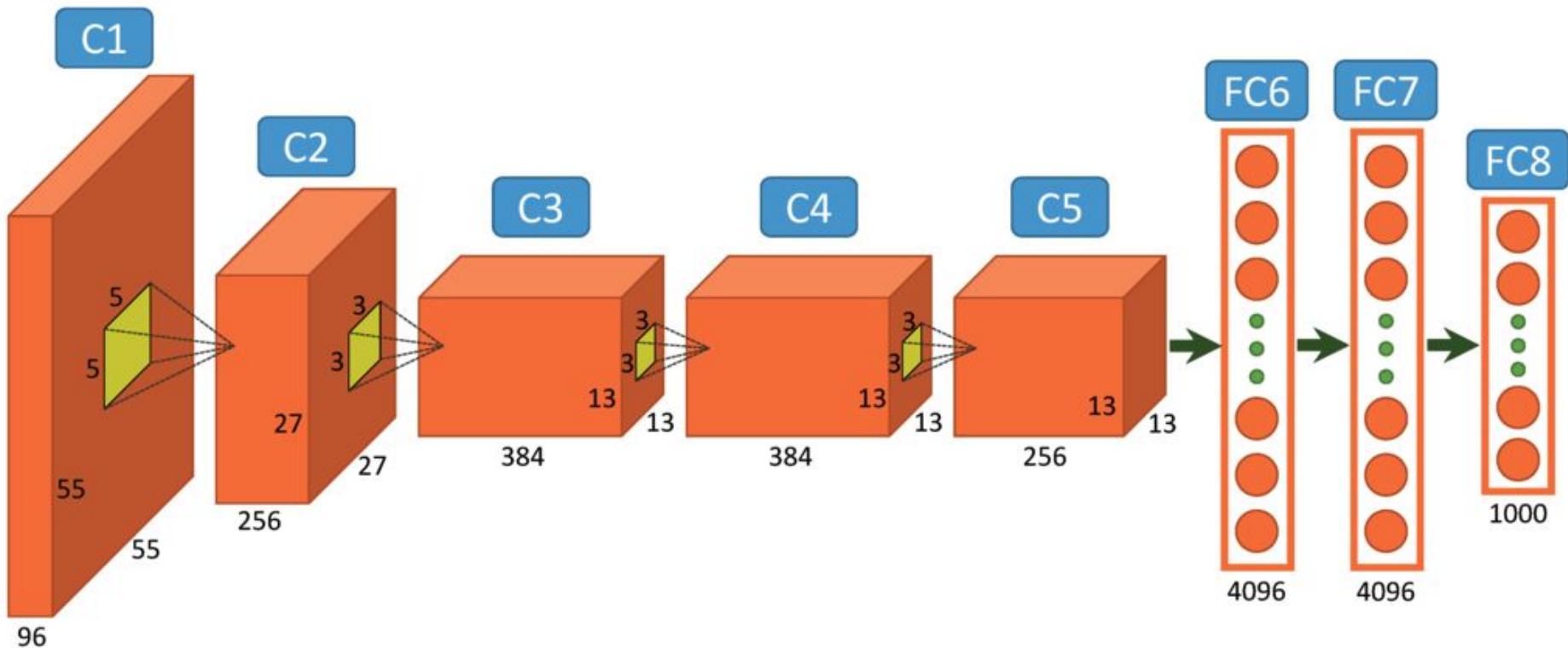
ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

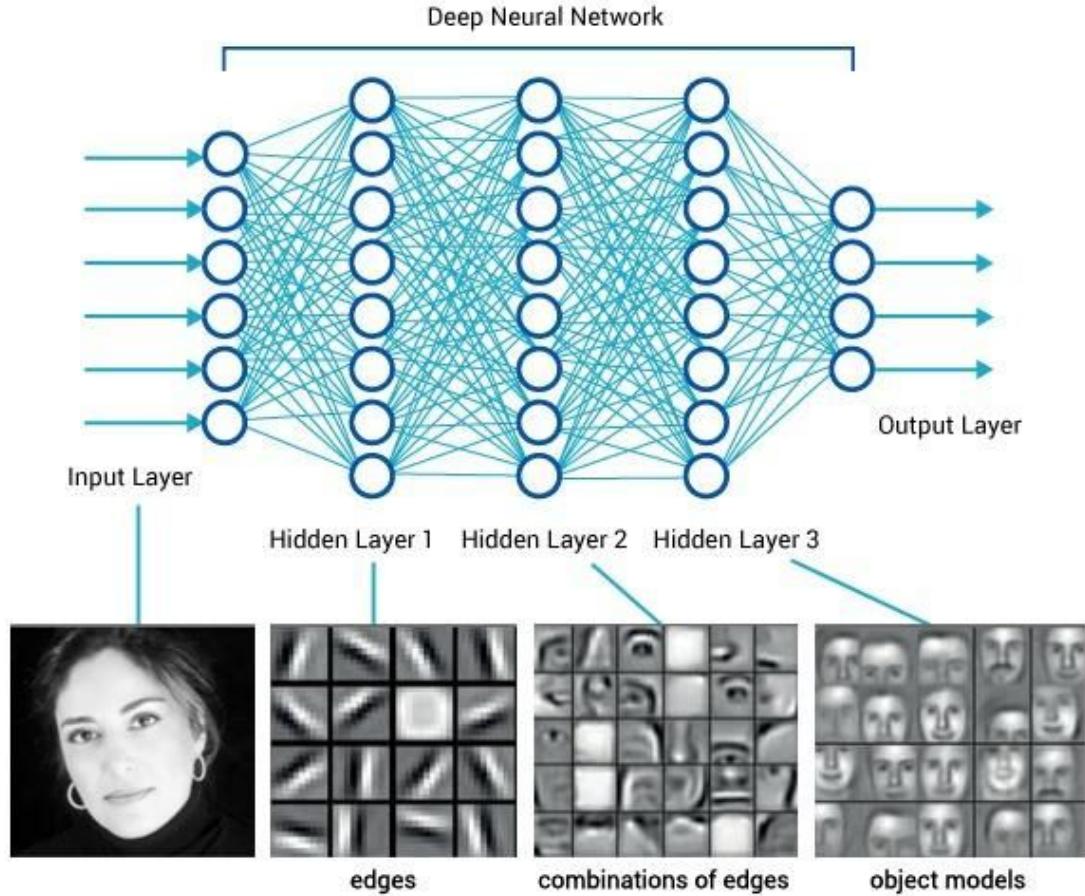
Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

AlexNet



AlexNet (cont'd)



AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

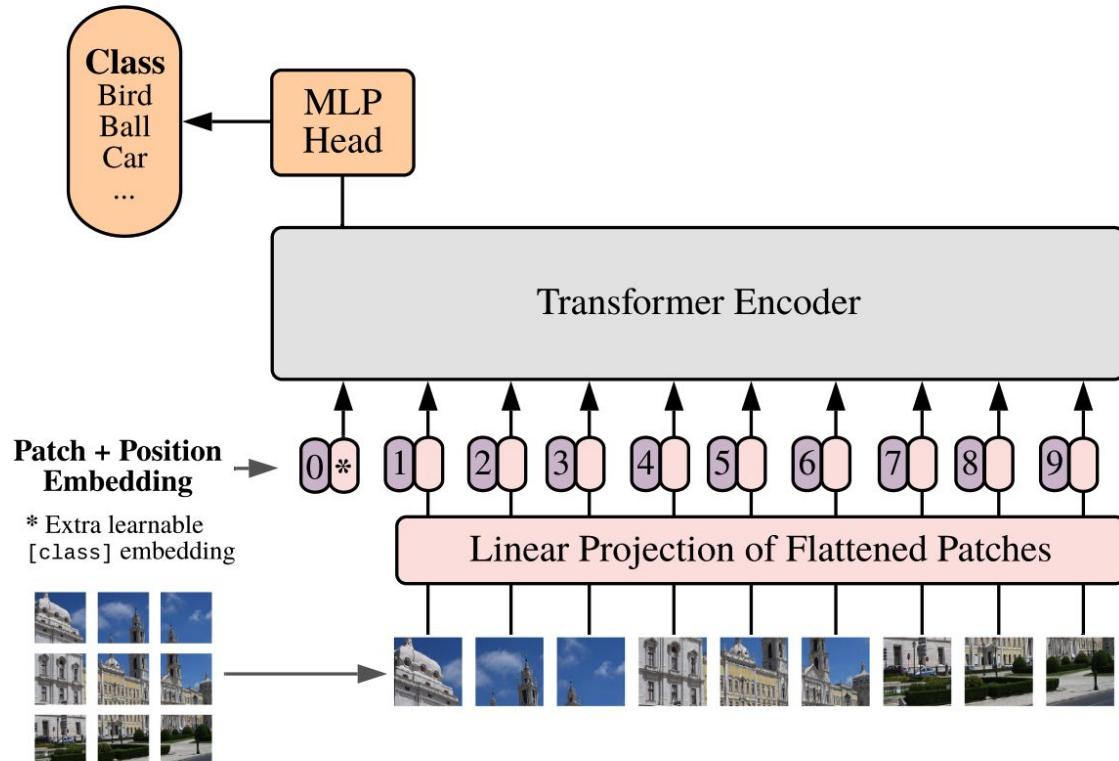
**Alexey Dosovitskiy^{*,†}, Lucas Beyer^{*}, Alexander Kolesnikov^{*}, Dirk Weissenborn^{*},
Xiaohua Zhai^{*}, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer,
Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby^{*,†}**

*equal technical contribution, †equal advising

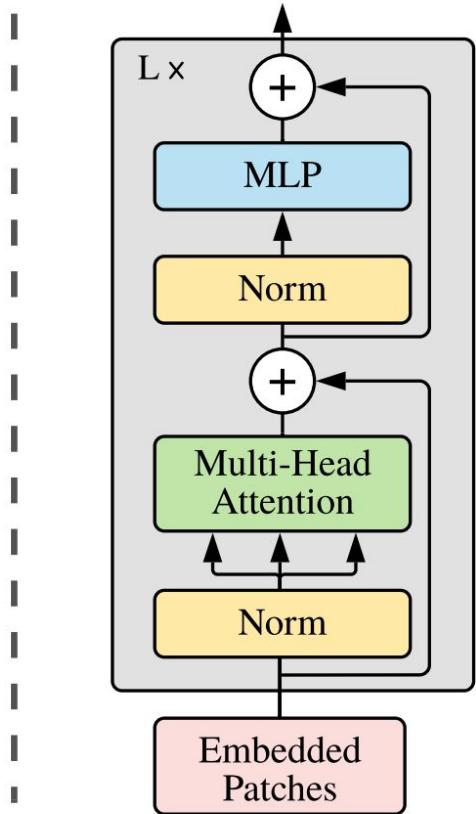
Google Research, Brain Team

{adosovitskiy, neilhoulsby}@google.com

Vision Transformer (ViT)



Transformer Encoder



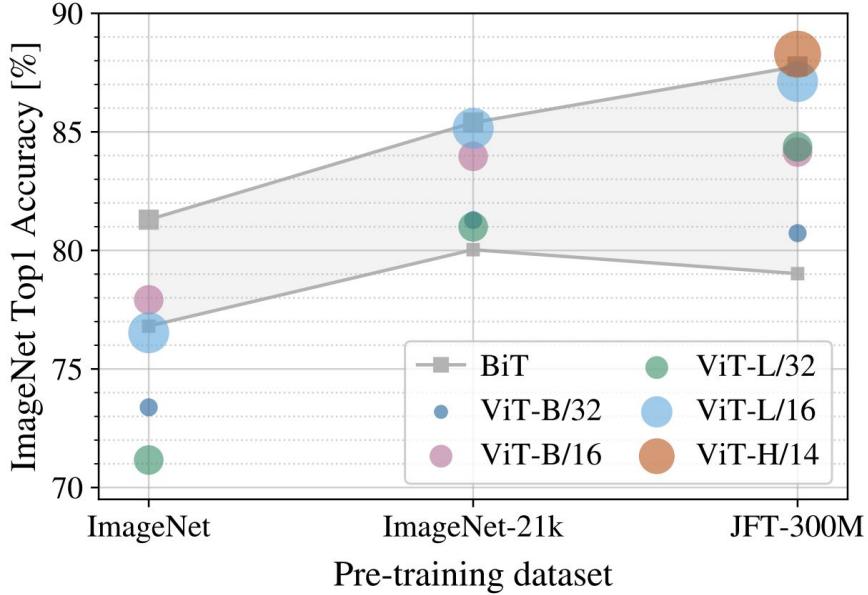
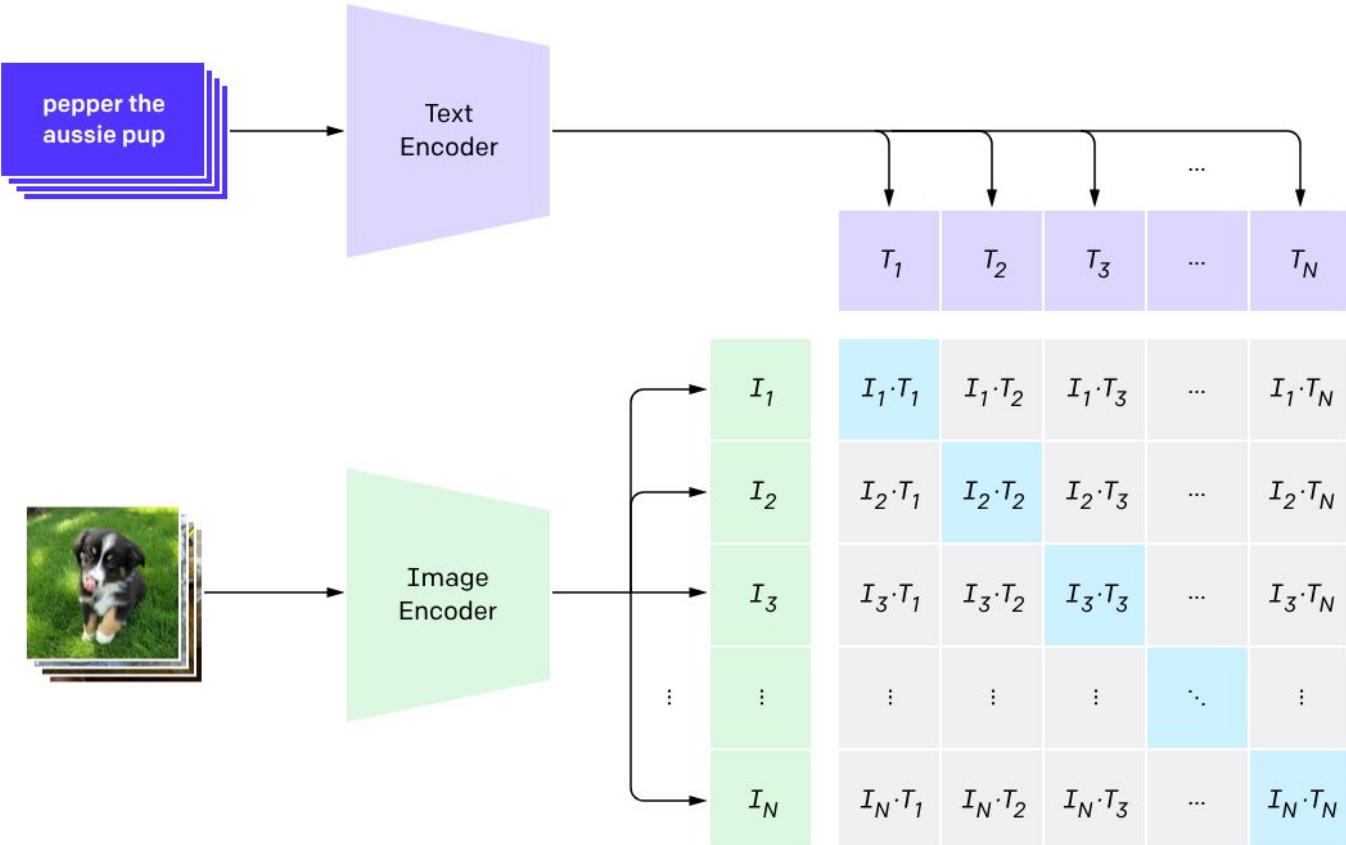


Figure 3: Transfer to ImageNet. While large ViT models perform worse than BiT ResNets (shaded area) when pre-trained on small datasets, they shine when pre-trained on larger datasets. Similarly, larger ViT variants overtake smaller ones as the dataset grows.

OpenAI's CLIP

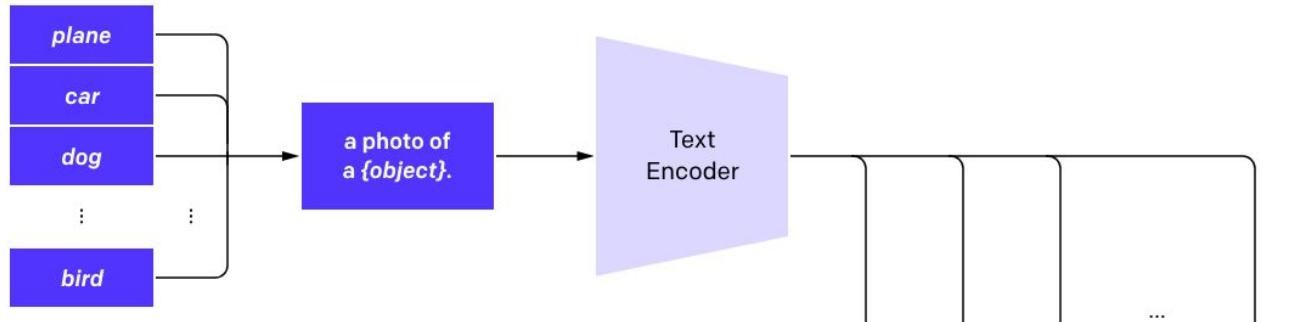
1. Contrastive pre-training



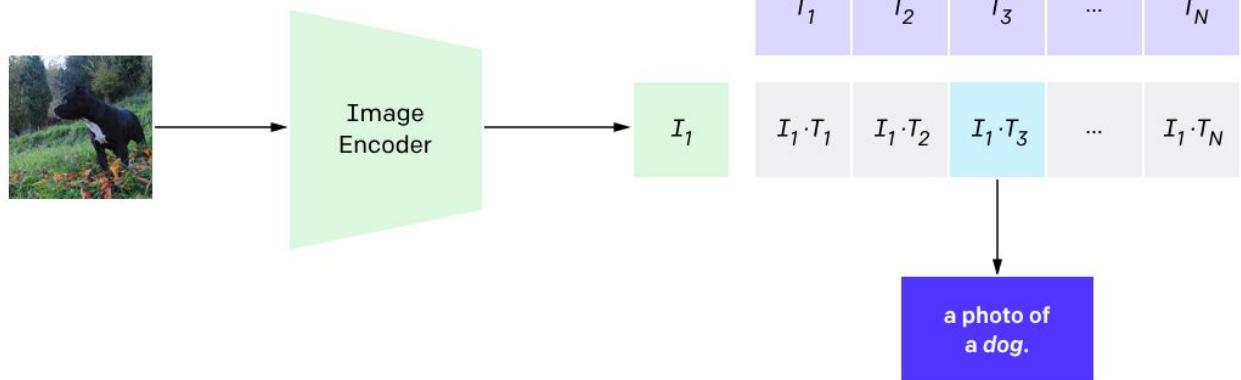
<https://openai.com/index/clip/>

OpenAI's CLIP (cont'd)

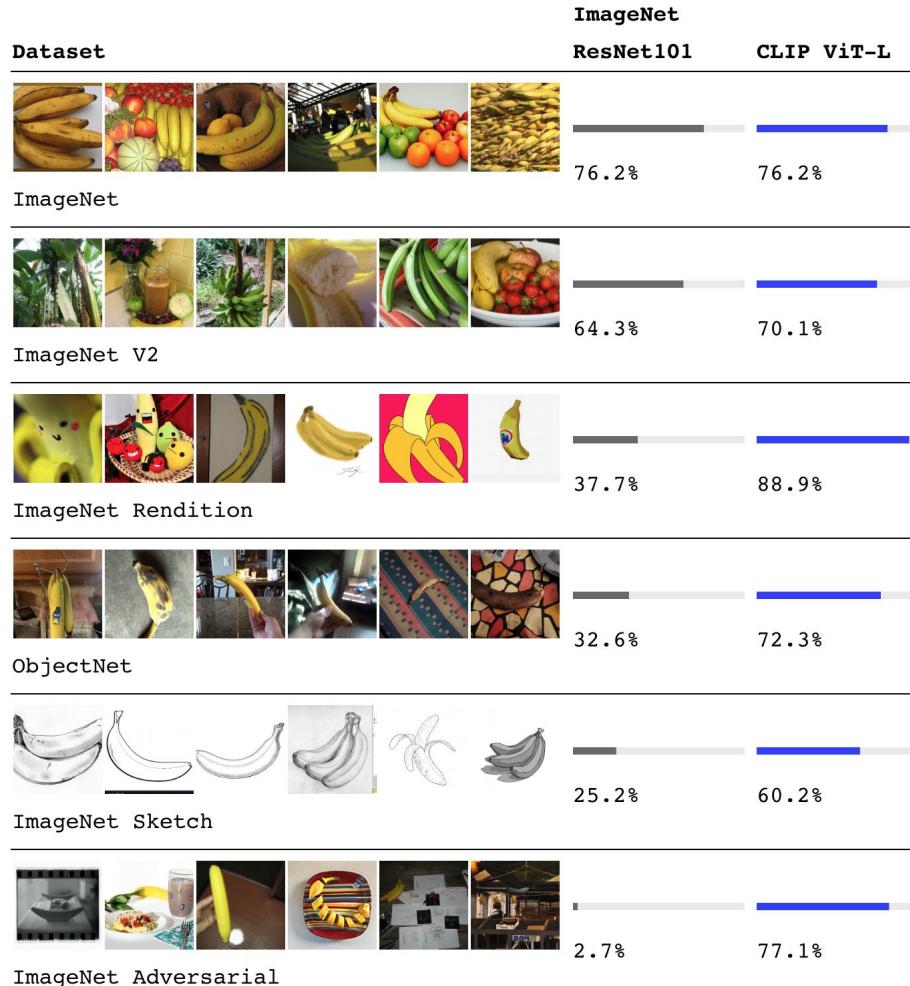
2. Create dataset classifier from label text



3. Use for zero-shot prediction



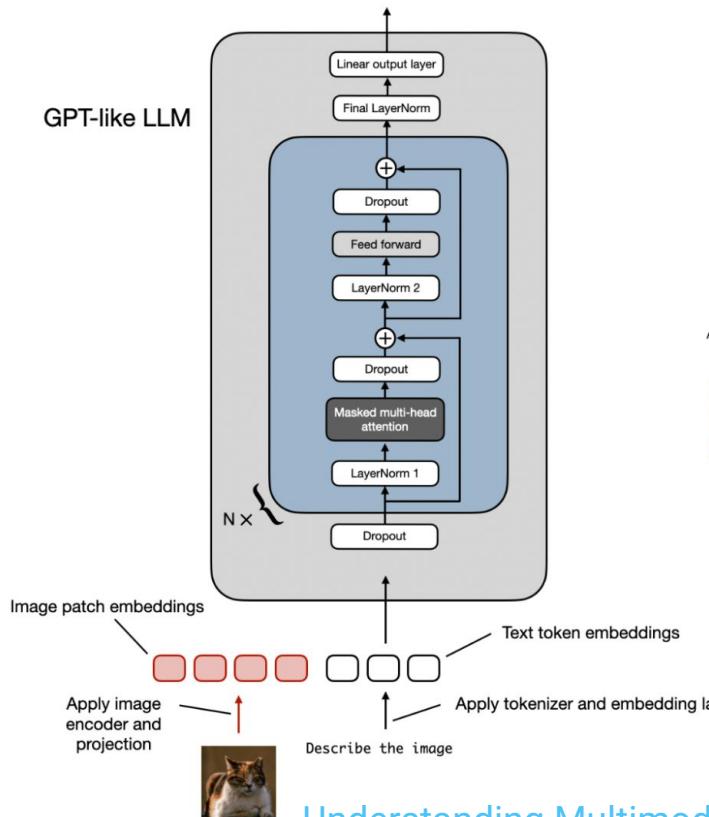
OpenAI's CLIP (cont'd)



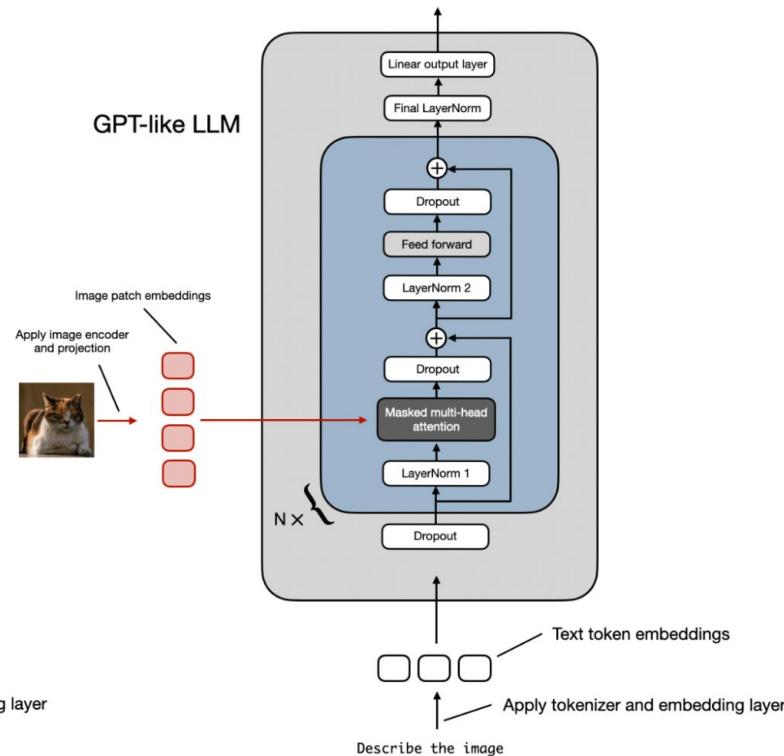
<https://openai.com/index/clip/>

Common approaches to building multimodal models

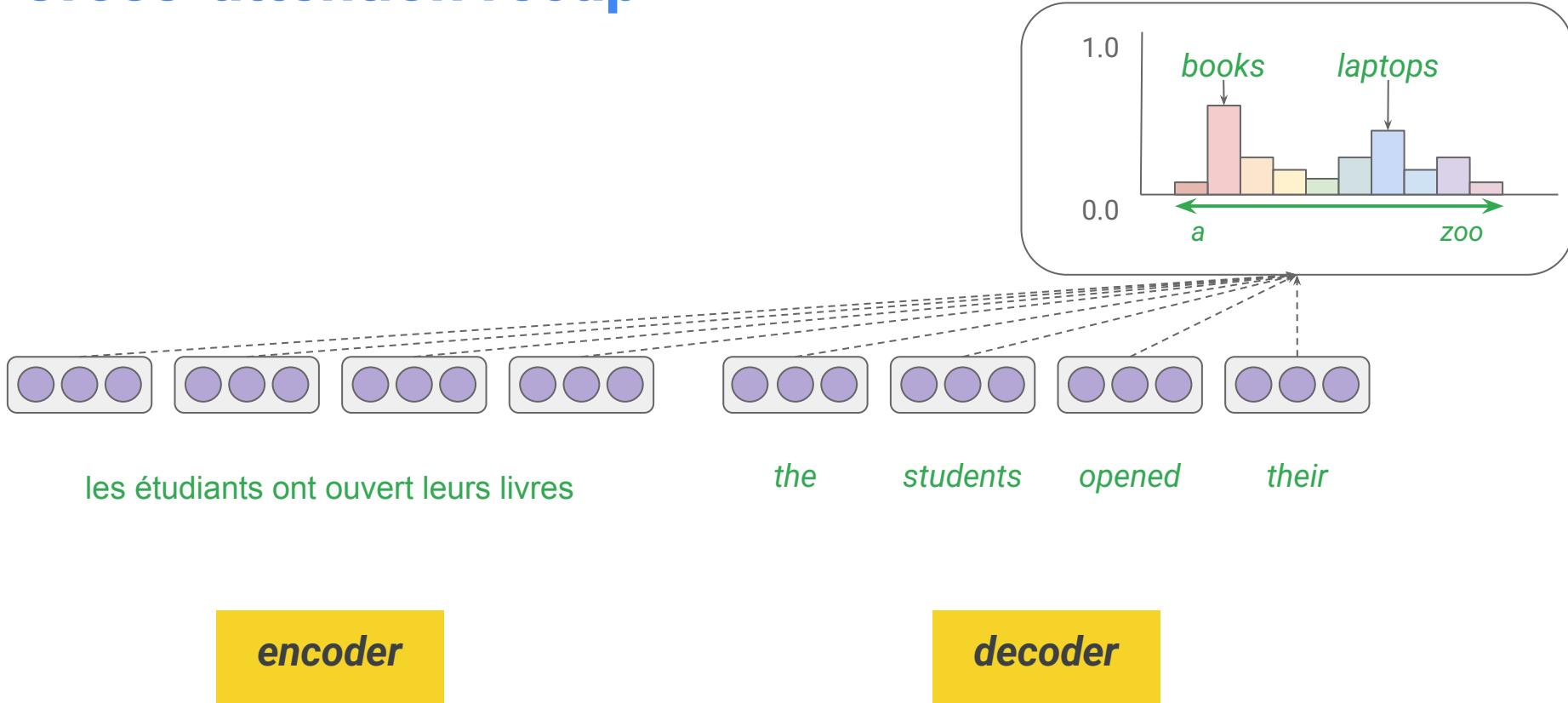
Method A: Unified Embedding Decoder Architecture



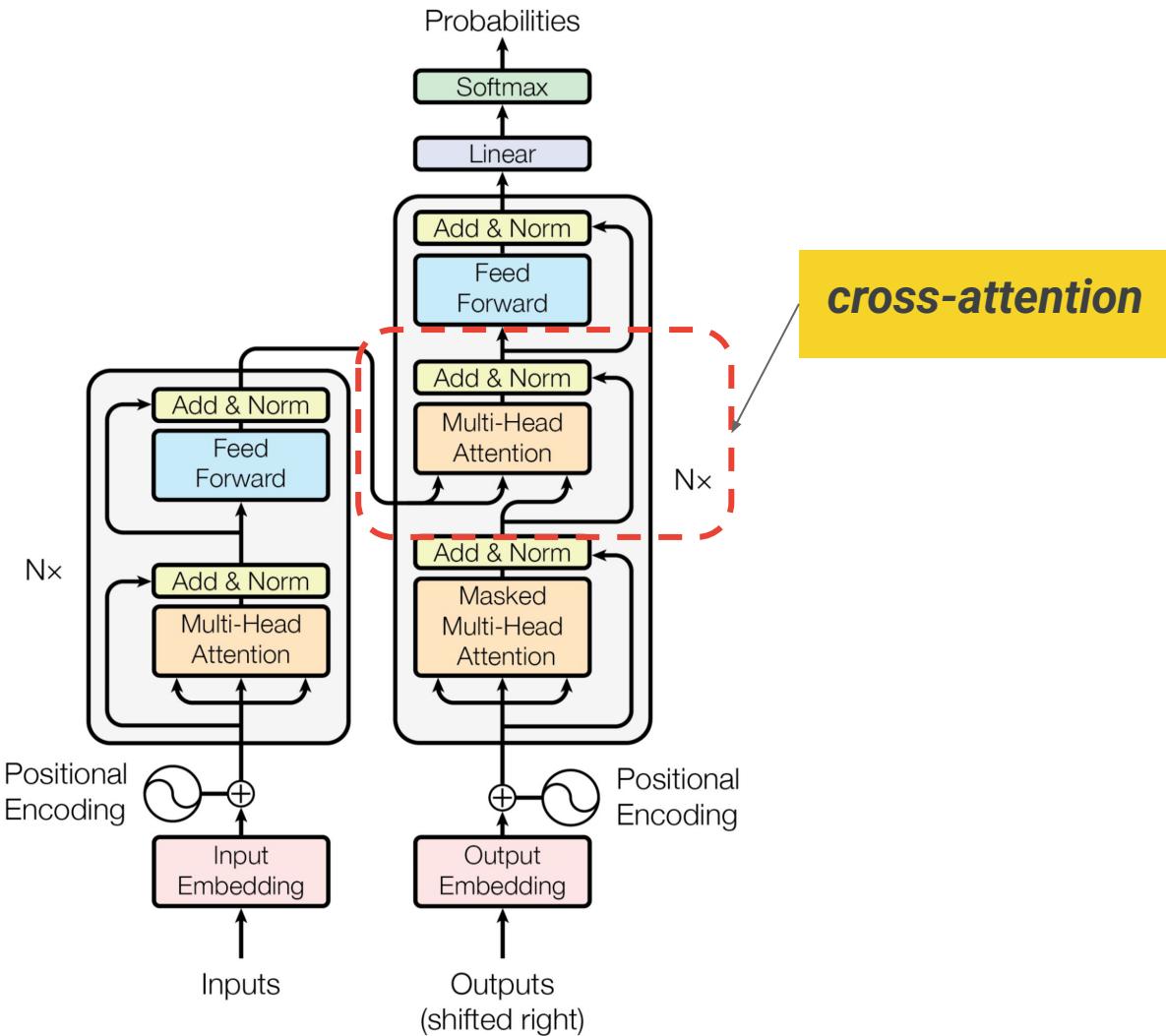
Method B: Cross-Modality Attention Architecture



Cross-attention recap

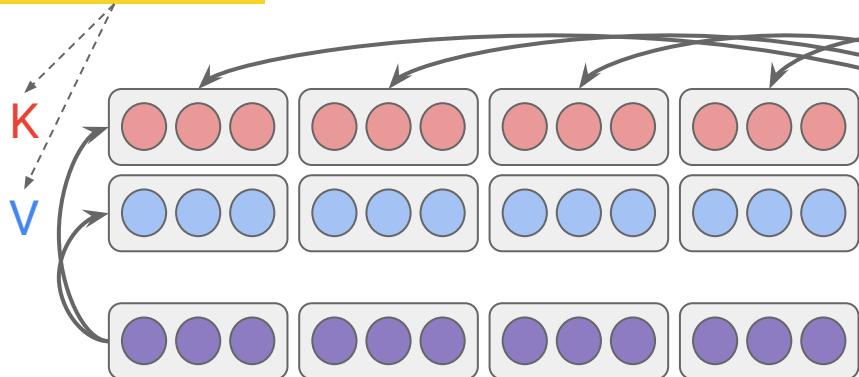


Cross-attention in the decoder



Cross-attention in the decoder

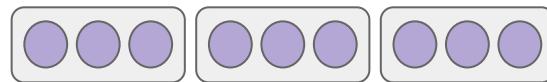
linear
projections



Multi-head Attention
(unmasked)



encoder



Multi-head cross-attention
(unmasked)



Multi-head Attention
(masked)

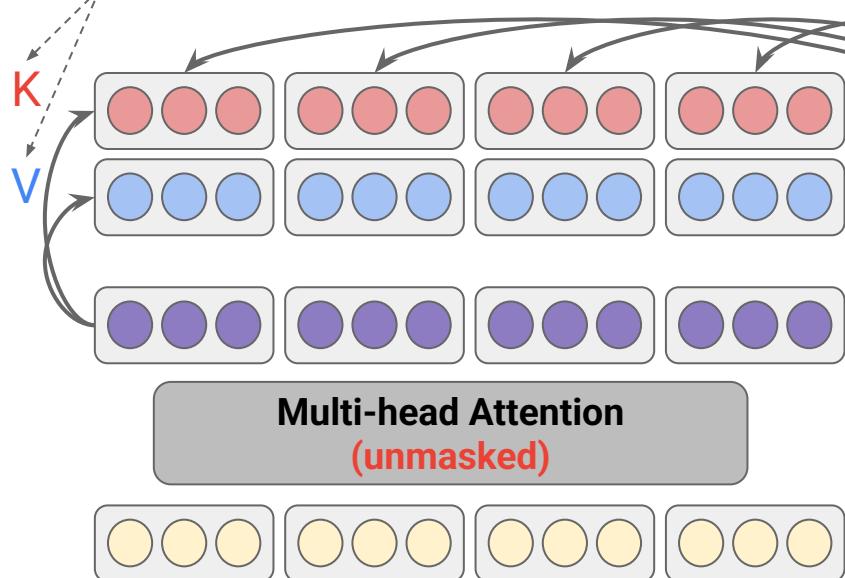


decoder

Cross-attention in the decoder (cont'd)

residual connections

linear projections

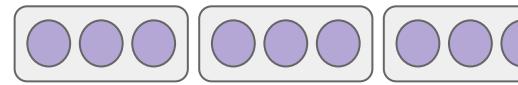


Multi-head Attention
(unmasked)

encoder

Multi-head cross-attention
(unmasked)

Q

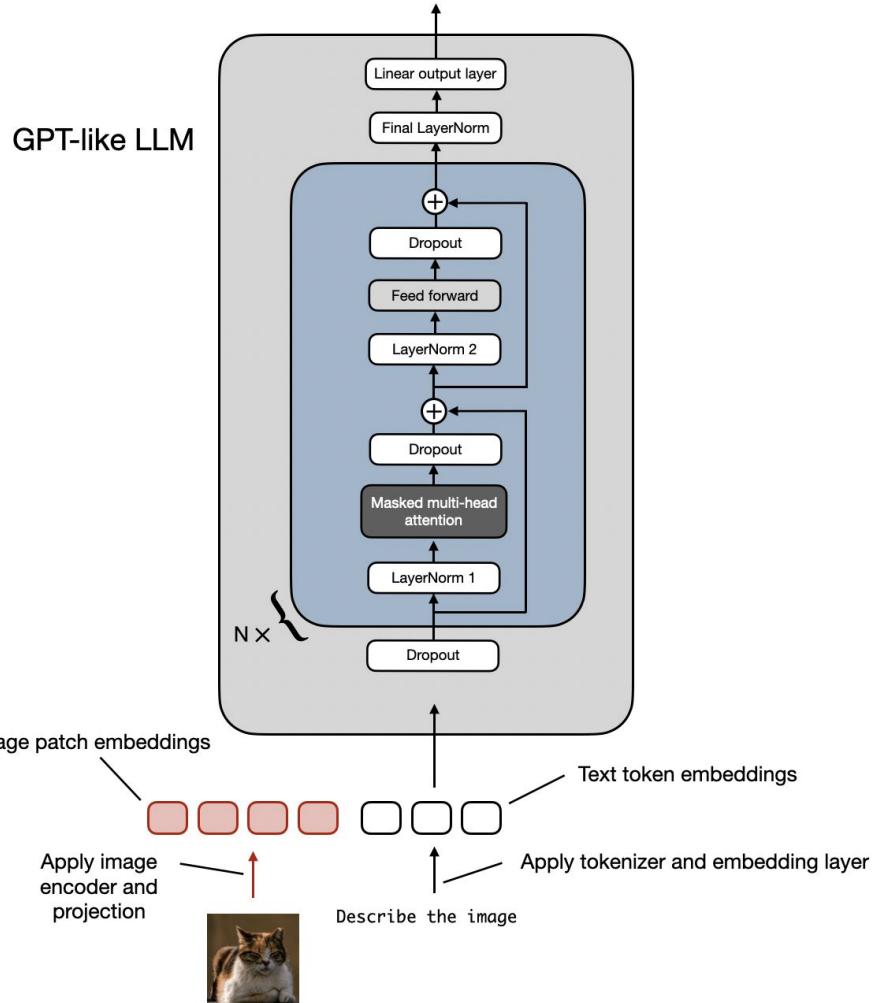


Multi-head Attention
(masked)

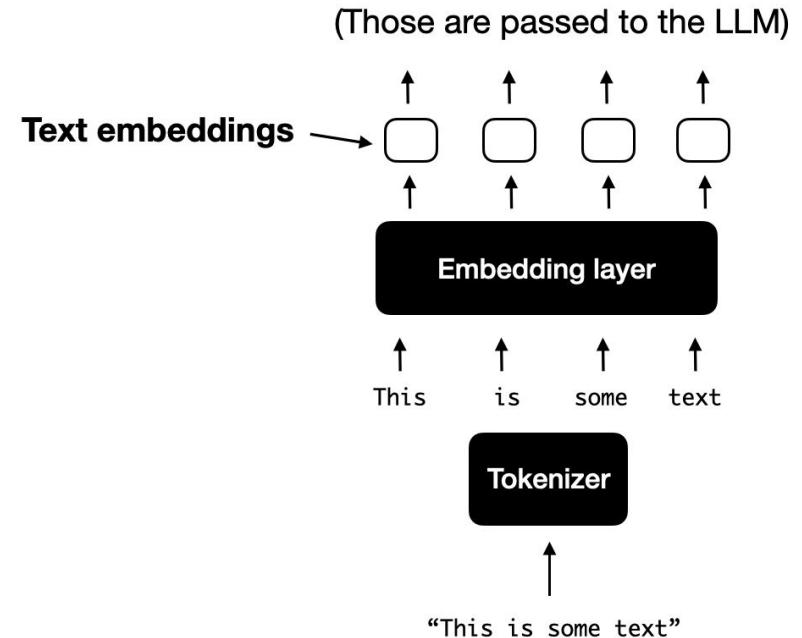
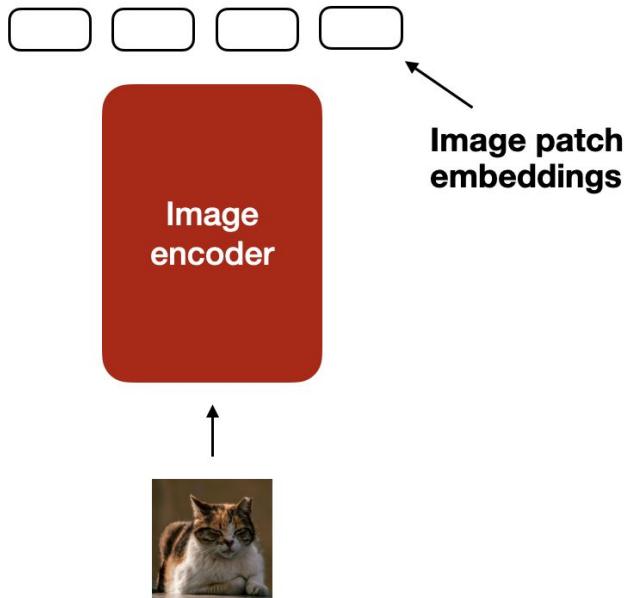
decoder

residual connections

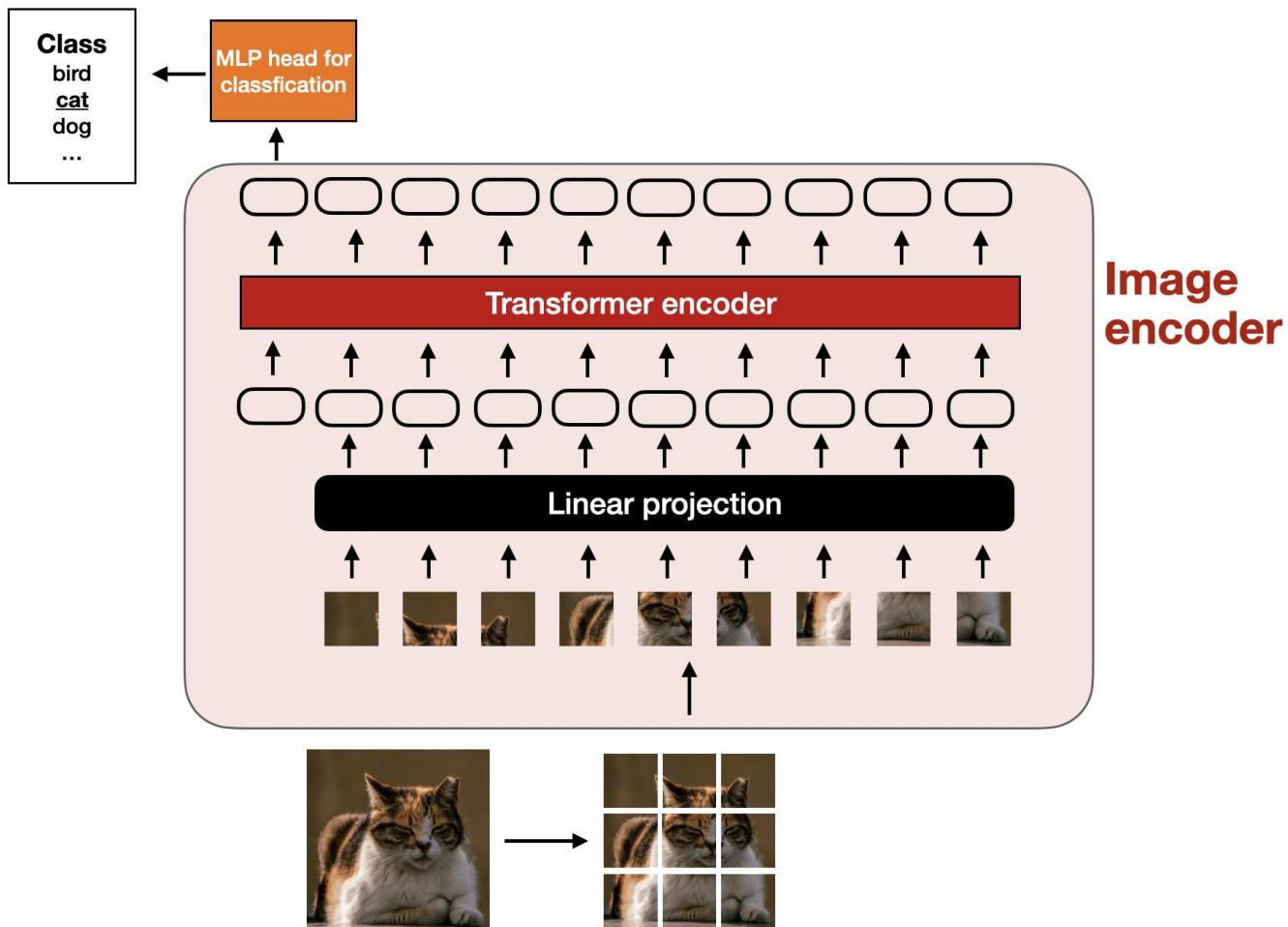
Method A: Unified embedding decoder architecture



Understanding image encoders



Understanding image encoder (cont'd)



BERT recap

softmax

linear



Image created by
Gemini

[CLS]



Multi-head Self-attention
(unmasked)



[CLS]

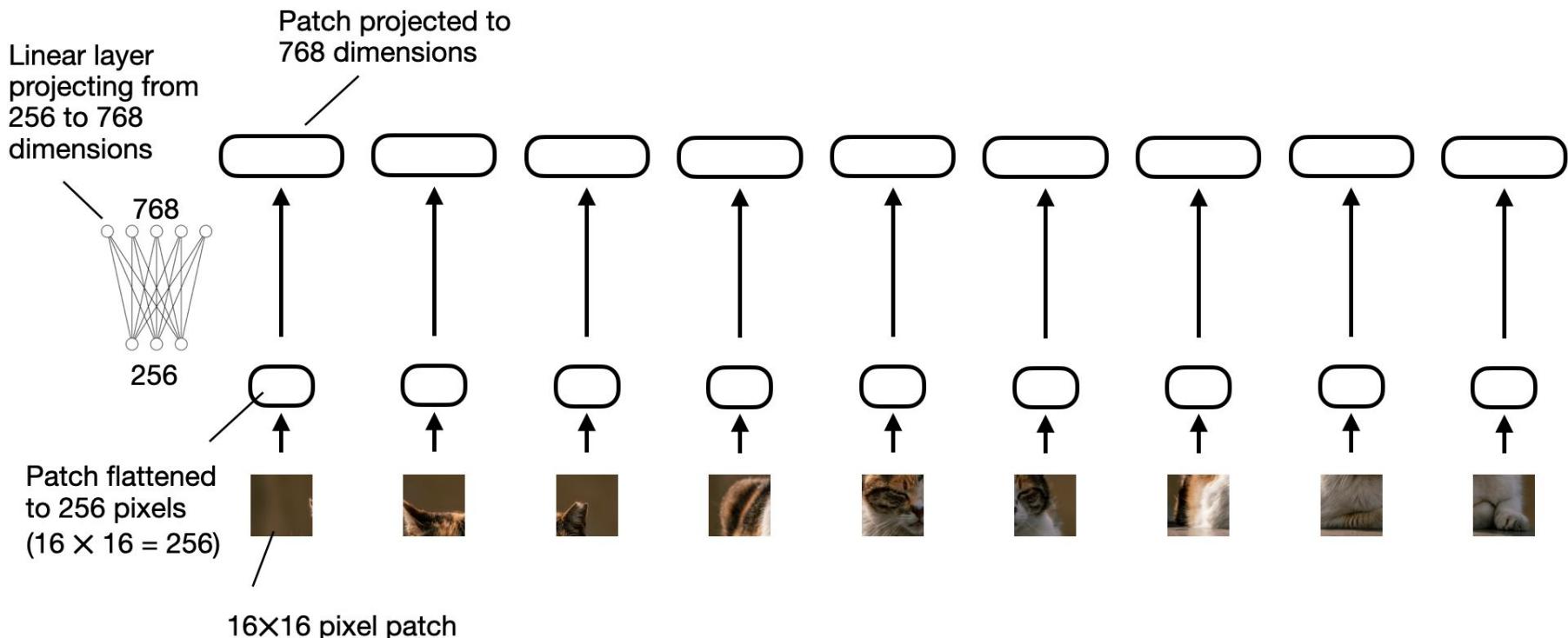
the

movie

was

good

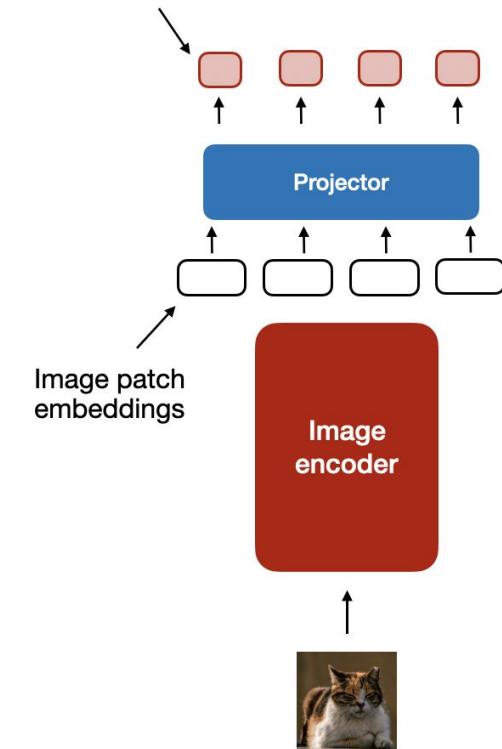
The role of the linear projection module



Text and image tokenization and embedding

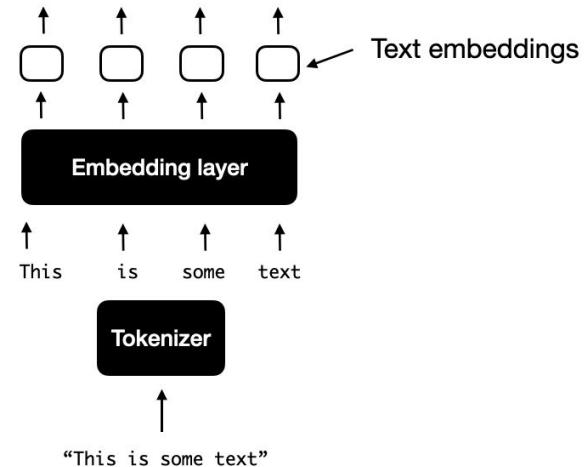
Image tokenization

Image patch embeddings rescaled to match the text embedding dimension

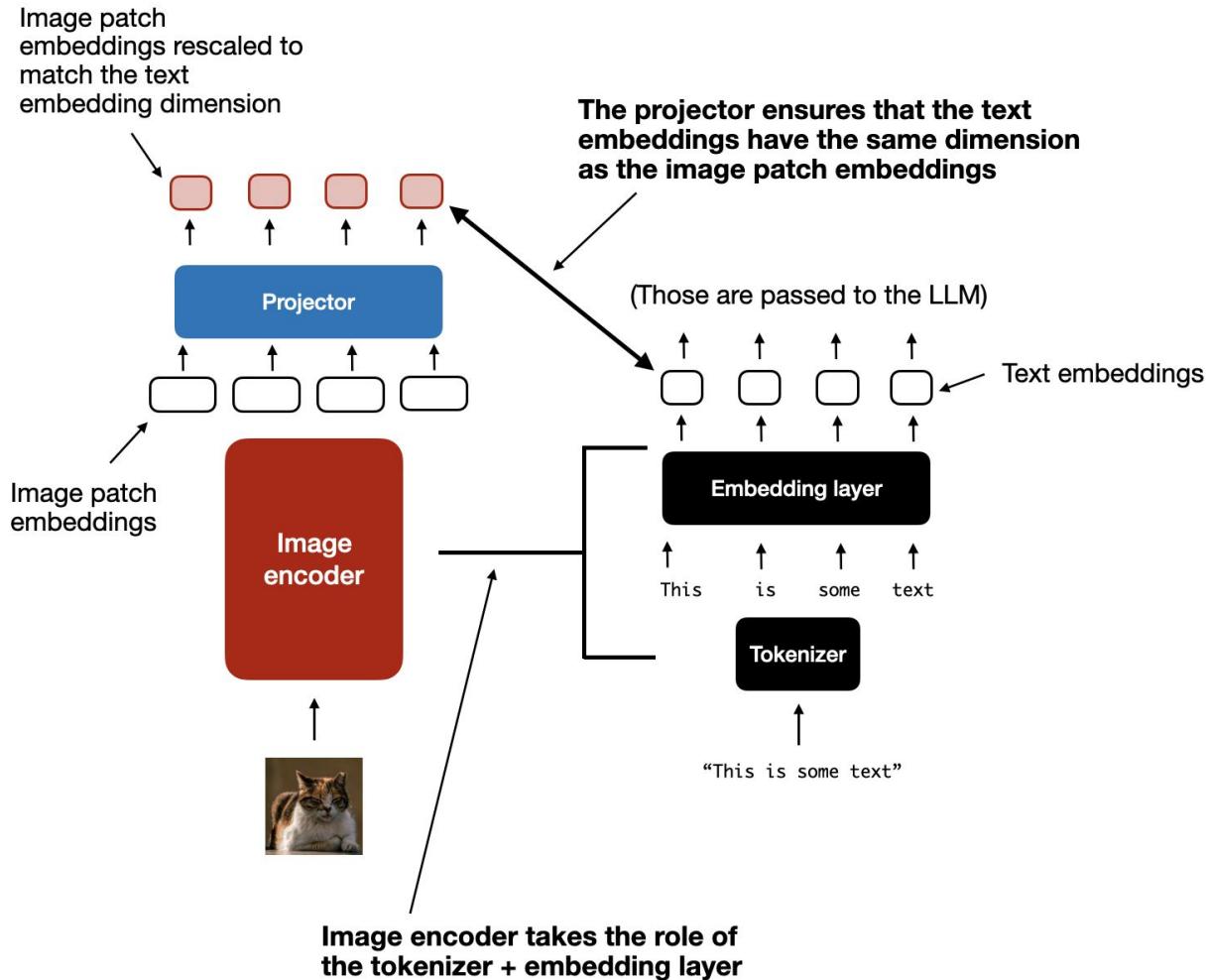


Text tokenization

(Those are passed to the LLM)

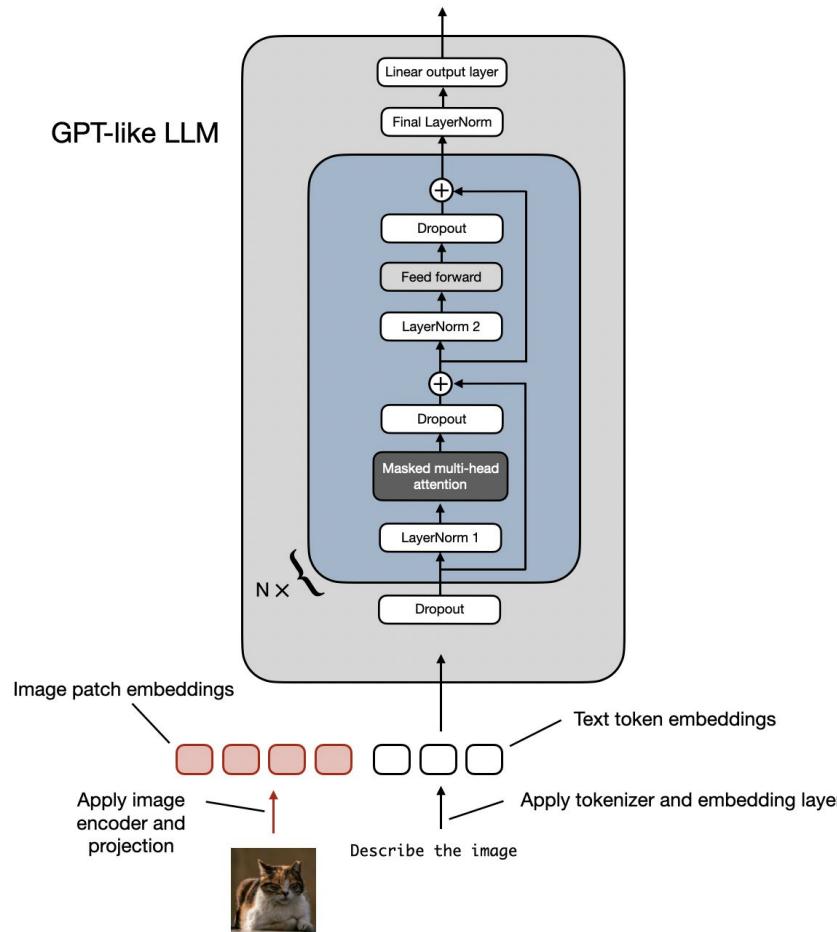


The role of the projector is to match the text token embedding dimensions

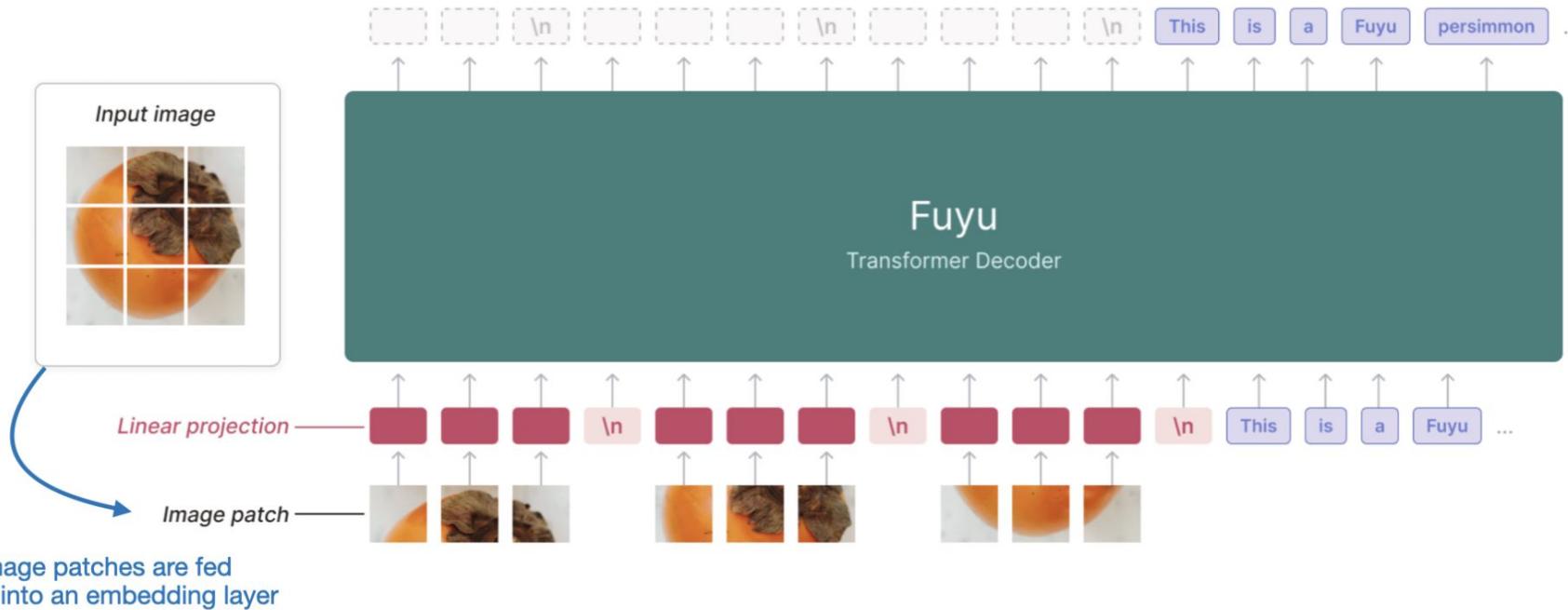


Method A: Unified Embedding Decoder Architecture

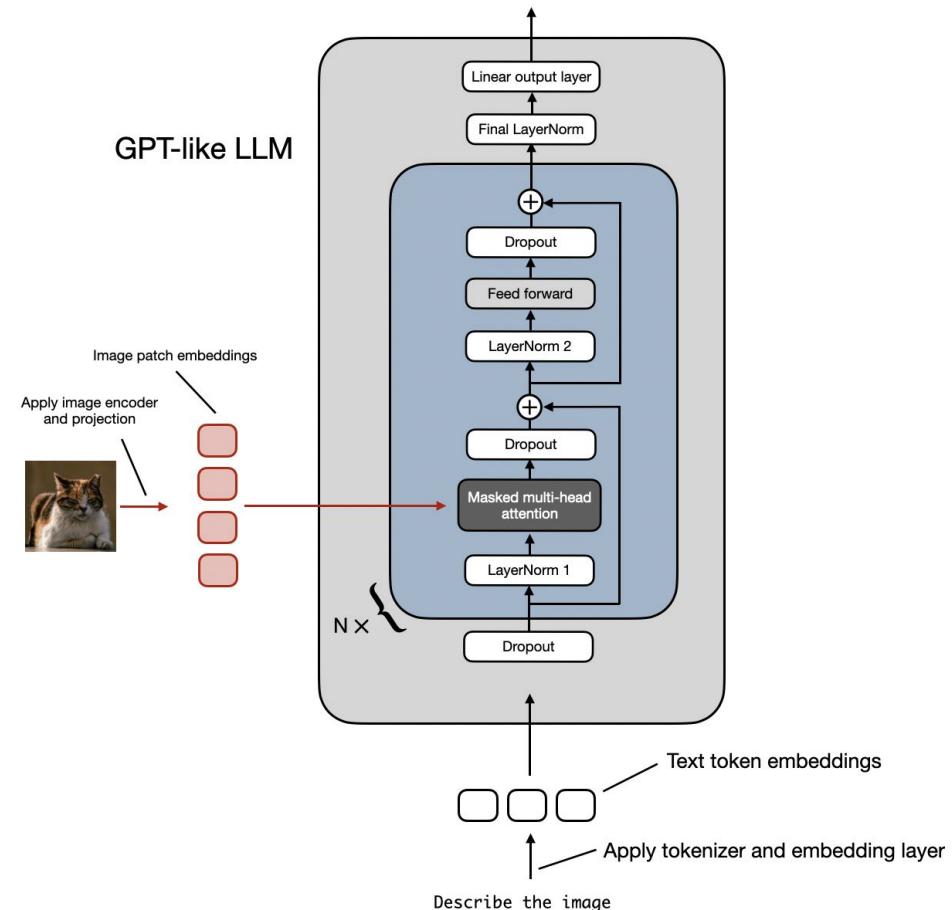
We can simply concatenate image and text embeddings



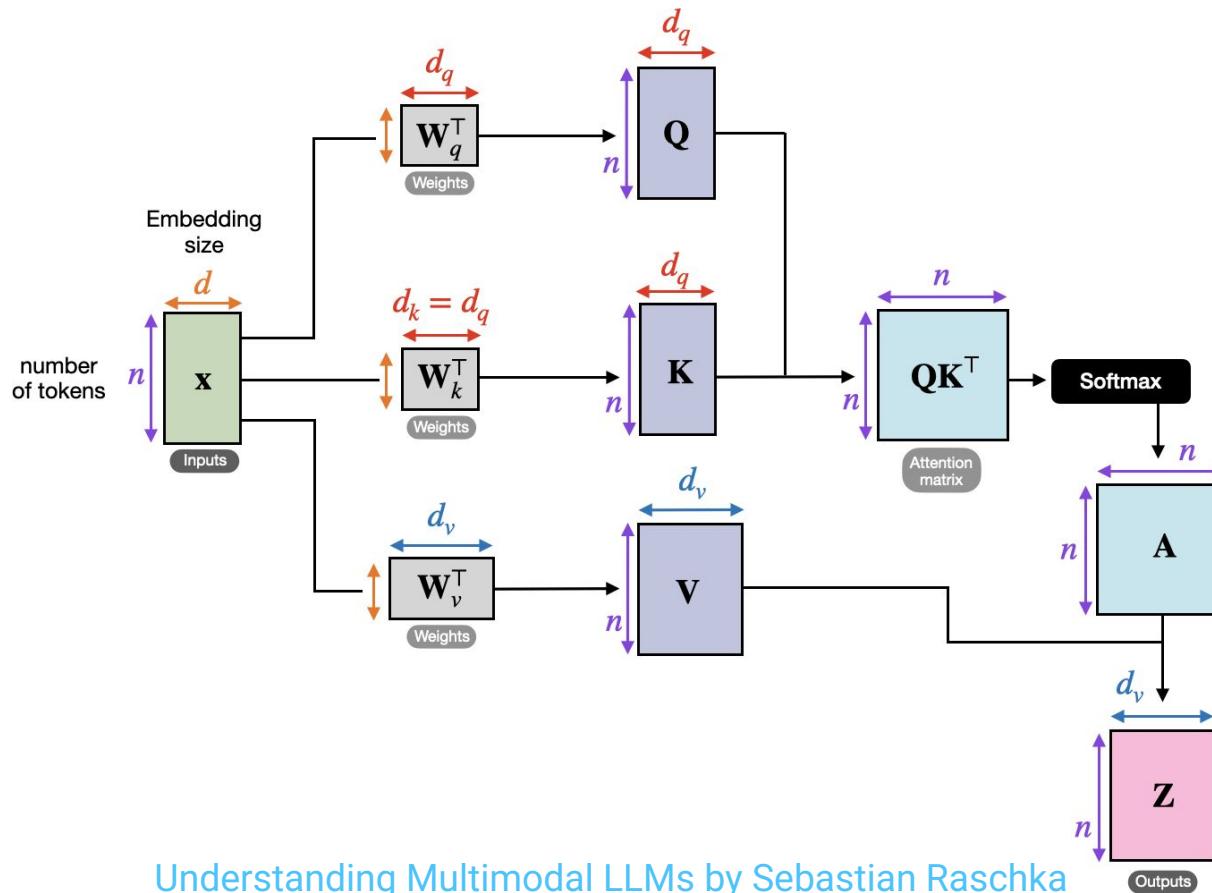
Versions of Method A that operate directly on patches



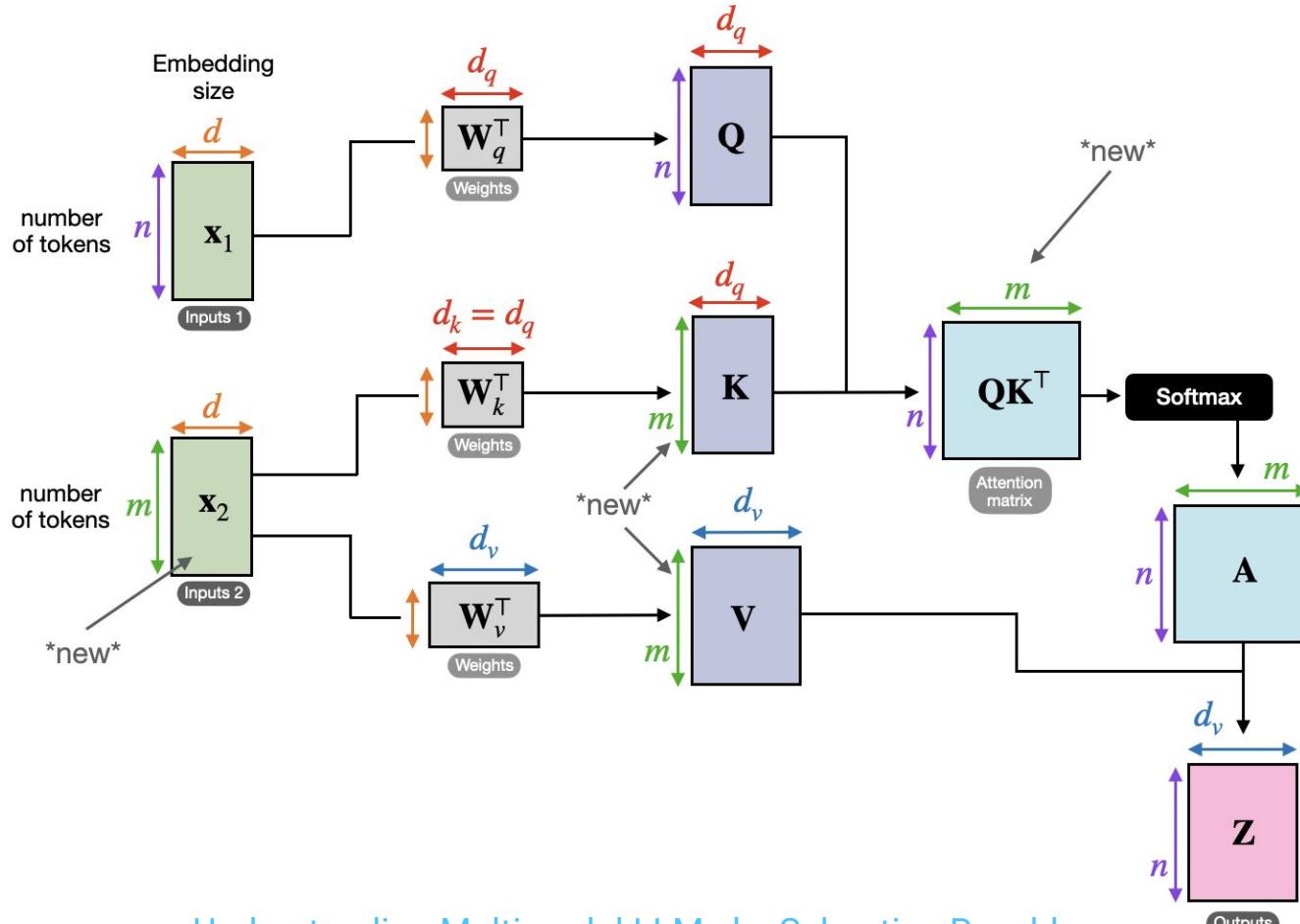
Method B: Cross-modality attention architecture



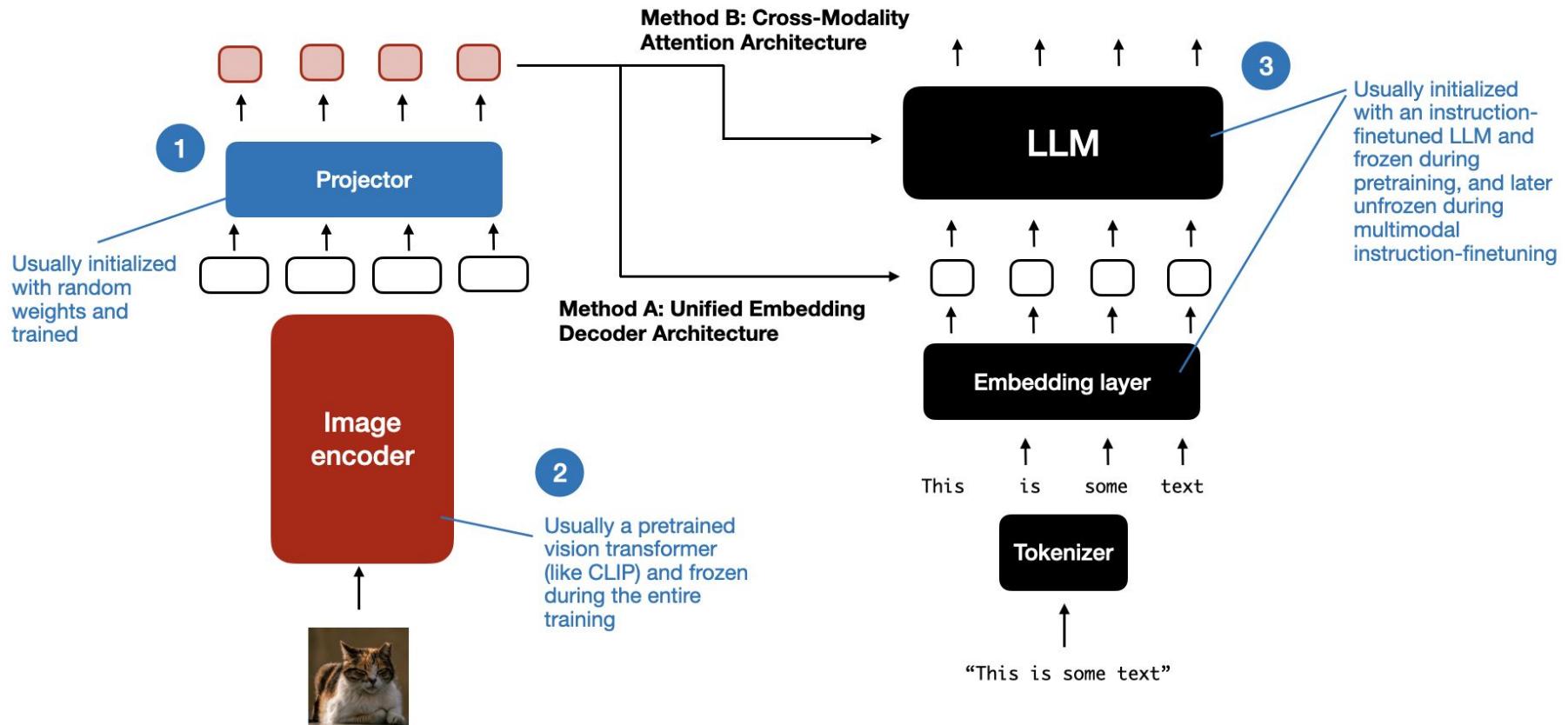
Regular self-attention



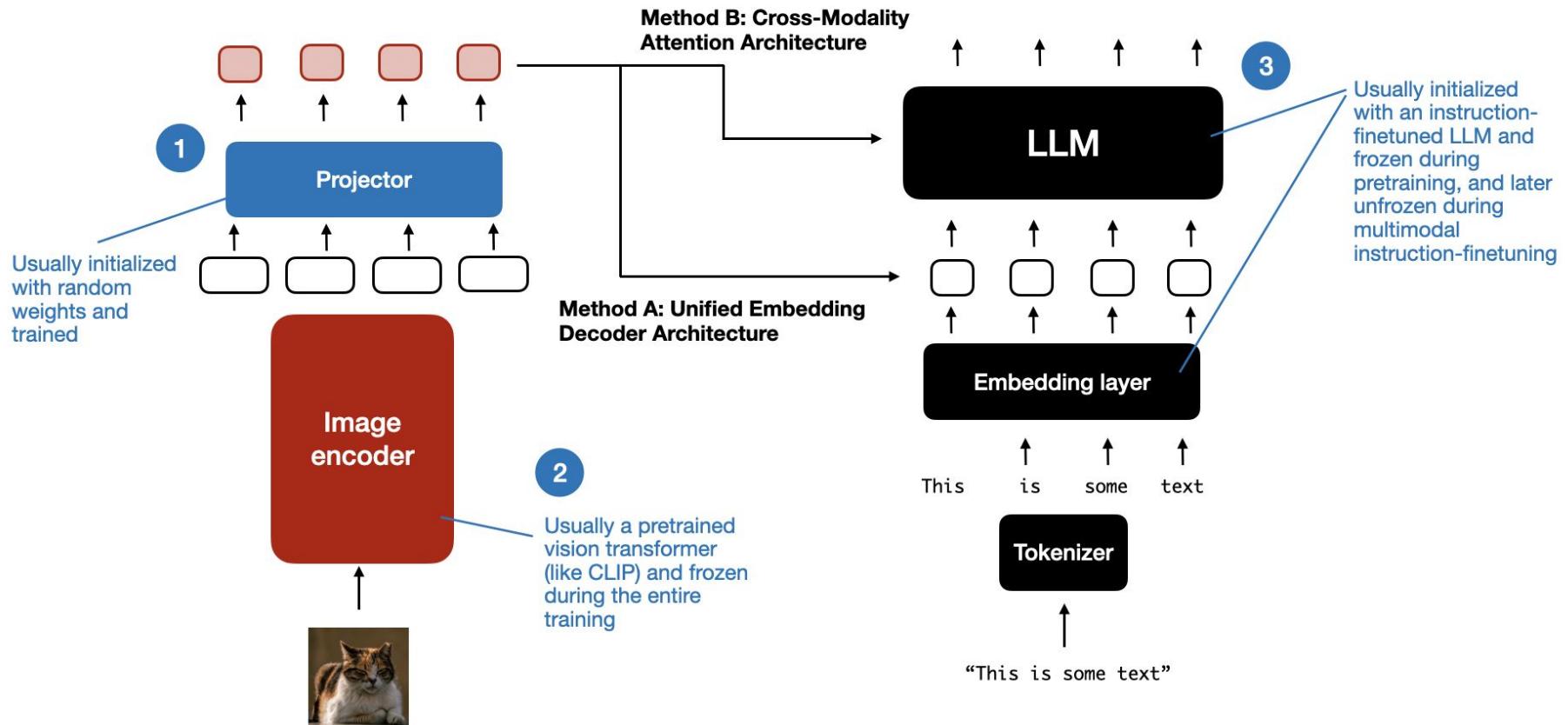
Cross-attention



Unified decoder and cross-attention model training

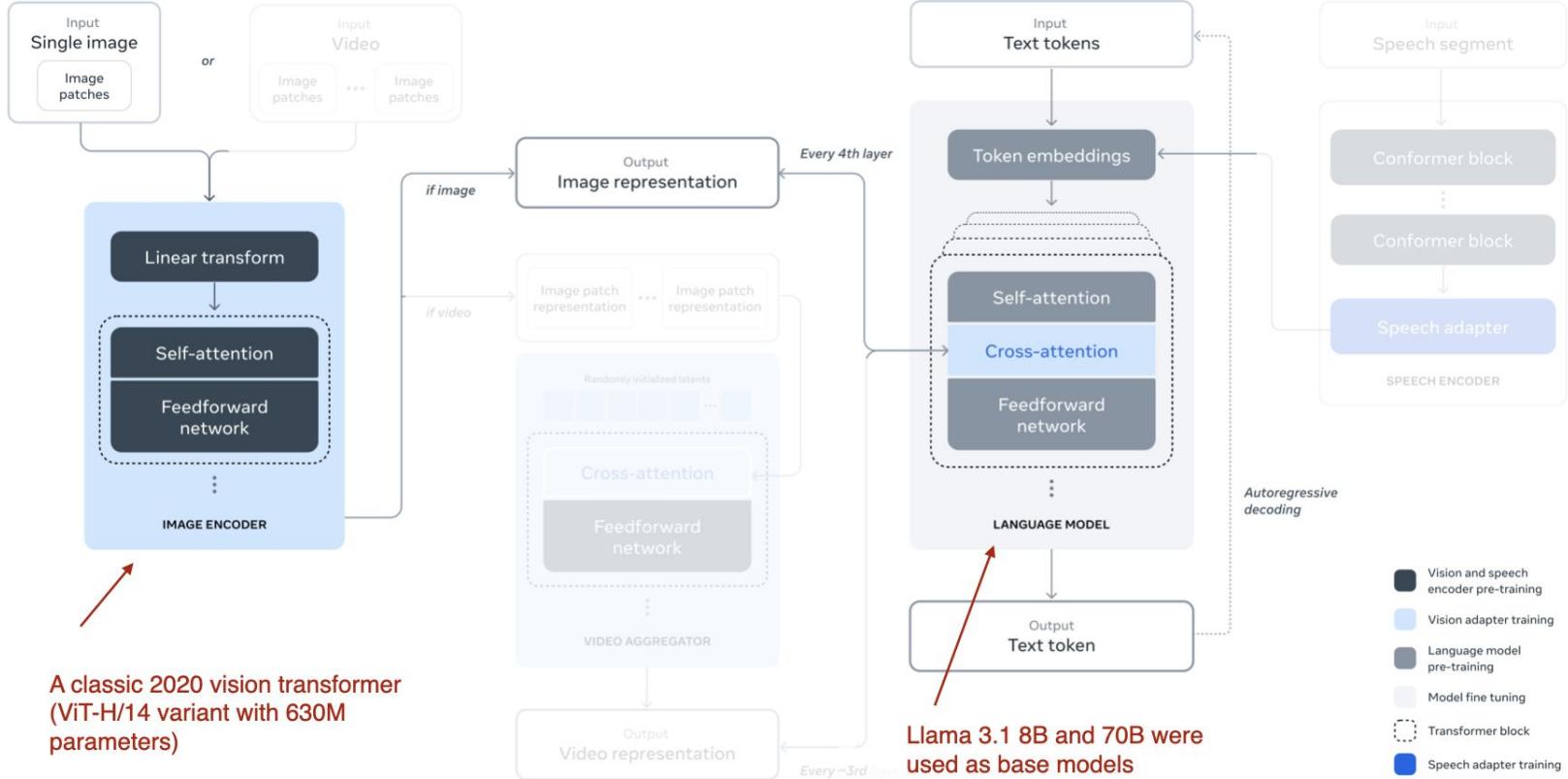


Unified decoder and cross-attention model training



Recent multimodal models and methods

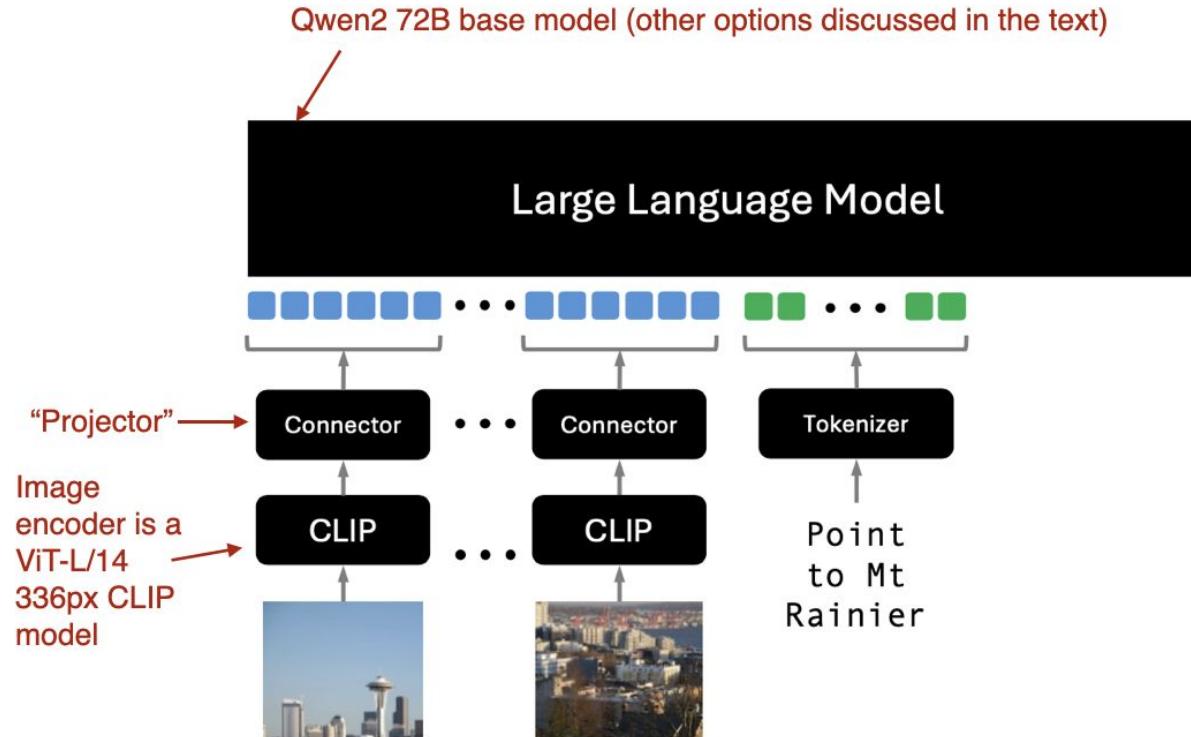
The Llama 3 herd of models



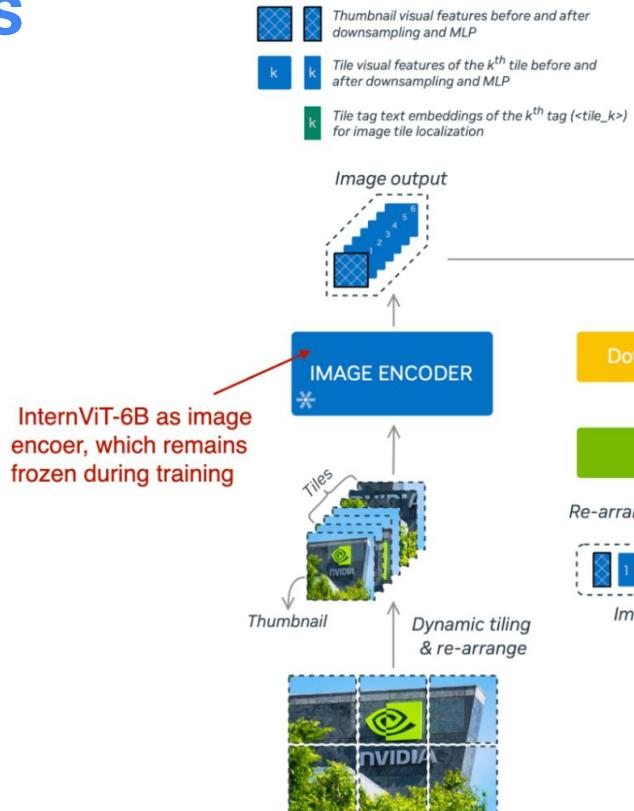
<https://arxiv.org/abs/2407.21783>

Understanding Multimodal LLMs by Sebastian Raschka

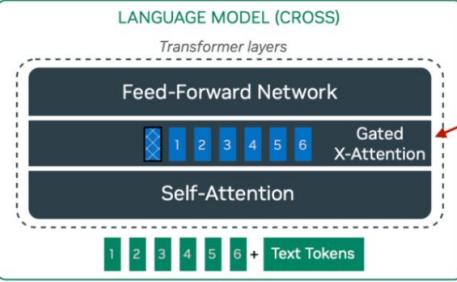
AI2's Molmo and PixMo



NVIDIA's NVLM



Method B: Cross-attention based (NVLM-X)



Cross-attention layers are trained

Hybrid method (NVLM-H)

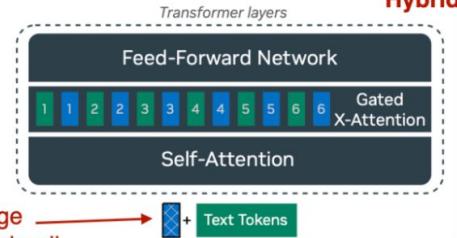
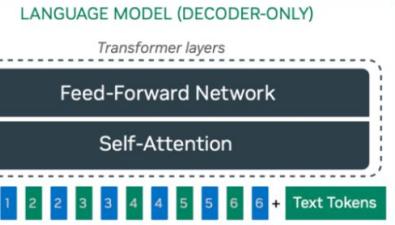


Image thumbnail → **+ Text Tokens**



Method A: Decoder-only (NVLM-D)

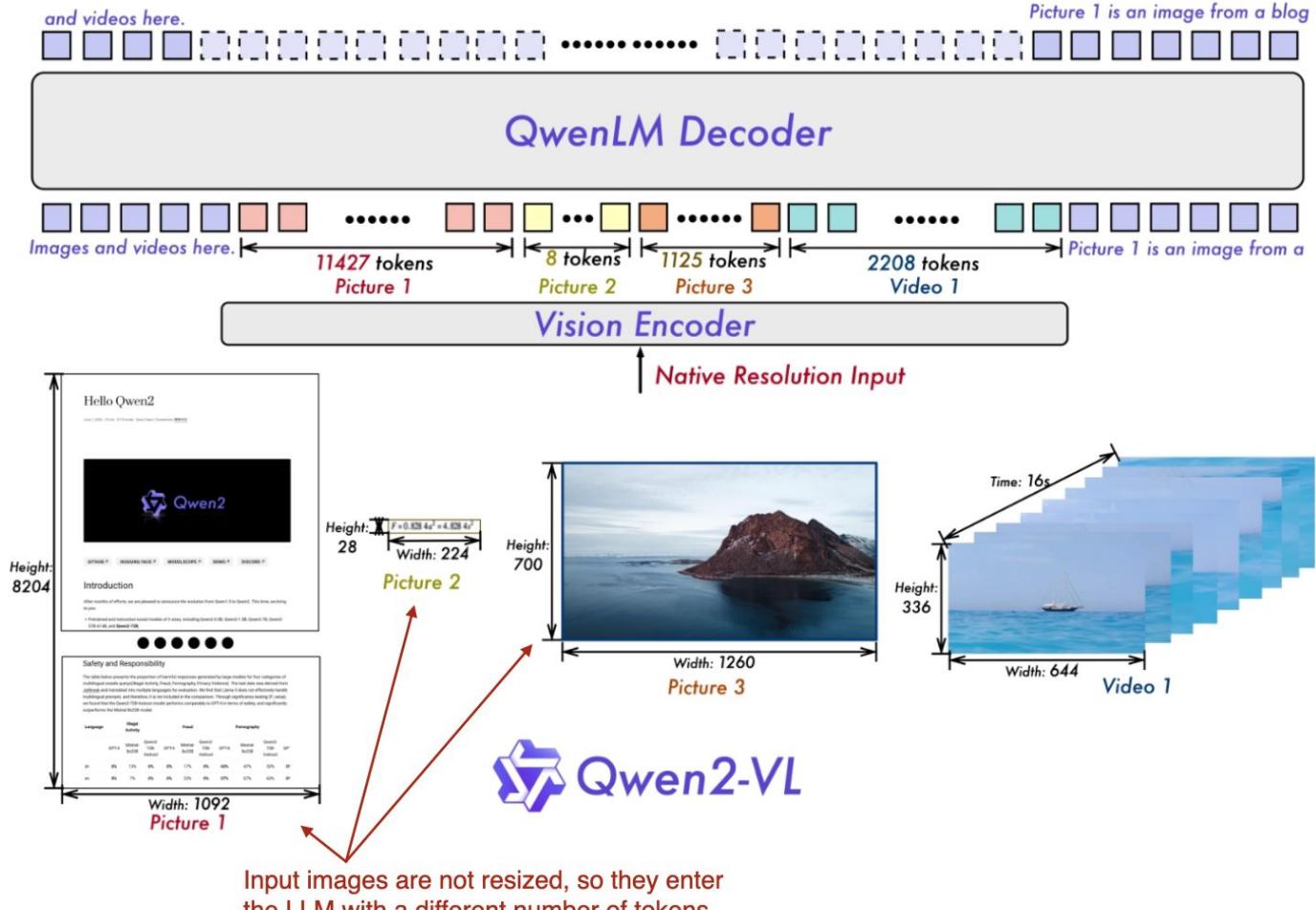
NVIDIA's NVLM

- **NVLM-X** (cross-attention) demonstrates superior computational efficiency for high-resolution images.
- **NVLM-D** (unified embedding) achieves higher accuracy in OCR-related tasks.
- **NVLM-H** combines the advantages of both methods.

<https://arxiv.org/abs/2409.11402>

[Understanding Multimodal LLMs by Sebastian Raschka](#)

Qwen2-VL: Enhancing vision- language model's perception of the world at any resolution

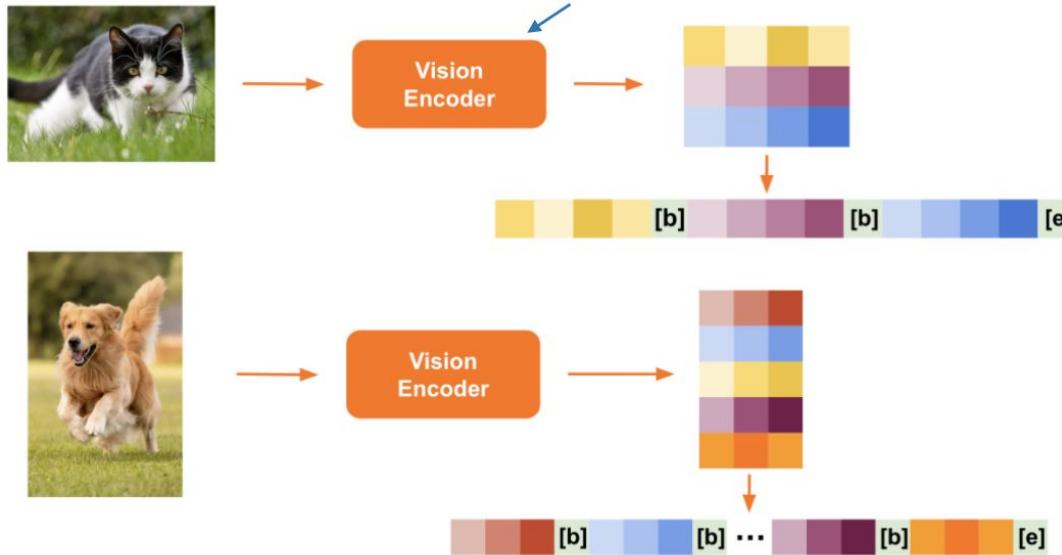


<https://arxiv.org/abs/2409.12191>

Understanding Multimodal LLMs by Sebastian Raschka

Pixtral 12B

Pixtral uses a 400M parameter image encoder, trained from scratch, that supports different image sizes natively



MM1: Methods, analysis & insights

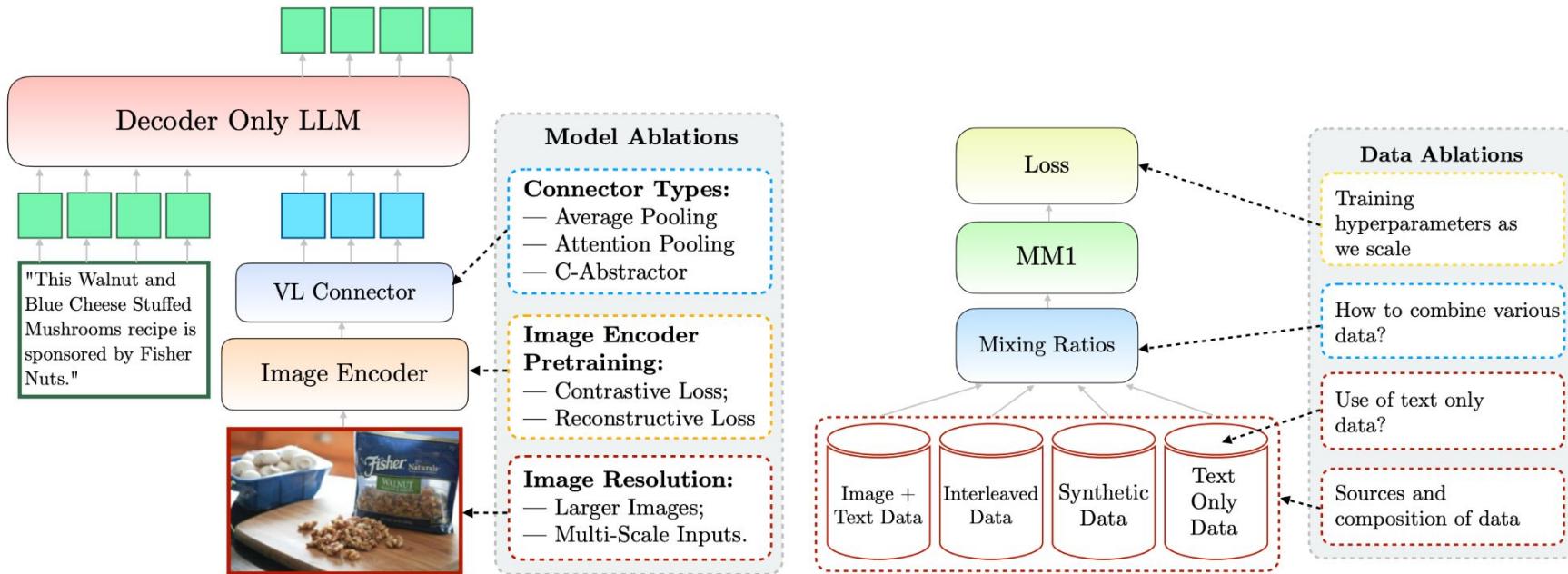
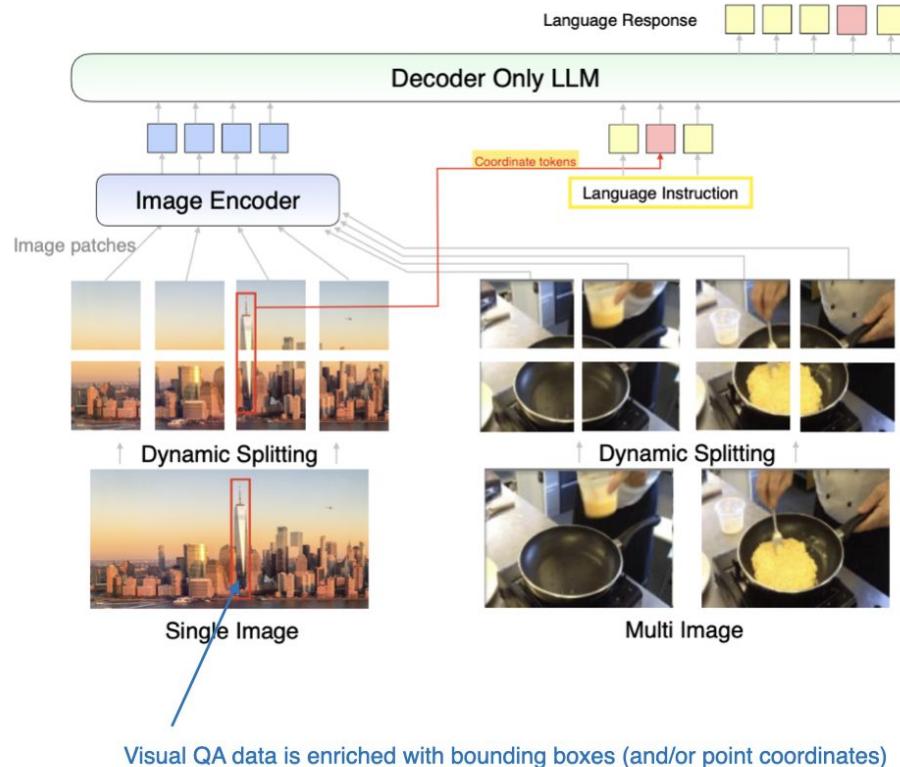


Fig. 3: *Left:* Model ablations: what visual encoder to use, how to feed rich visual data, and how to connect the visual representation to the LLM. *Right:* Data ablations: type of data, and their mixture.

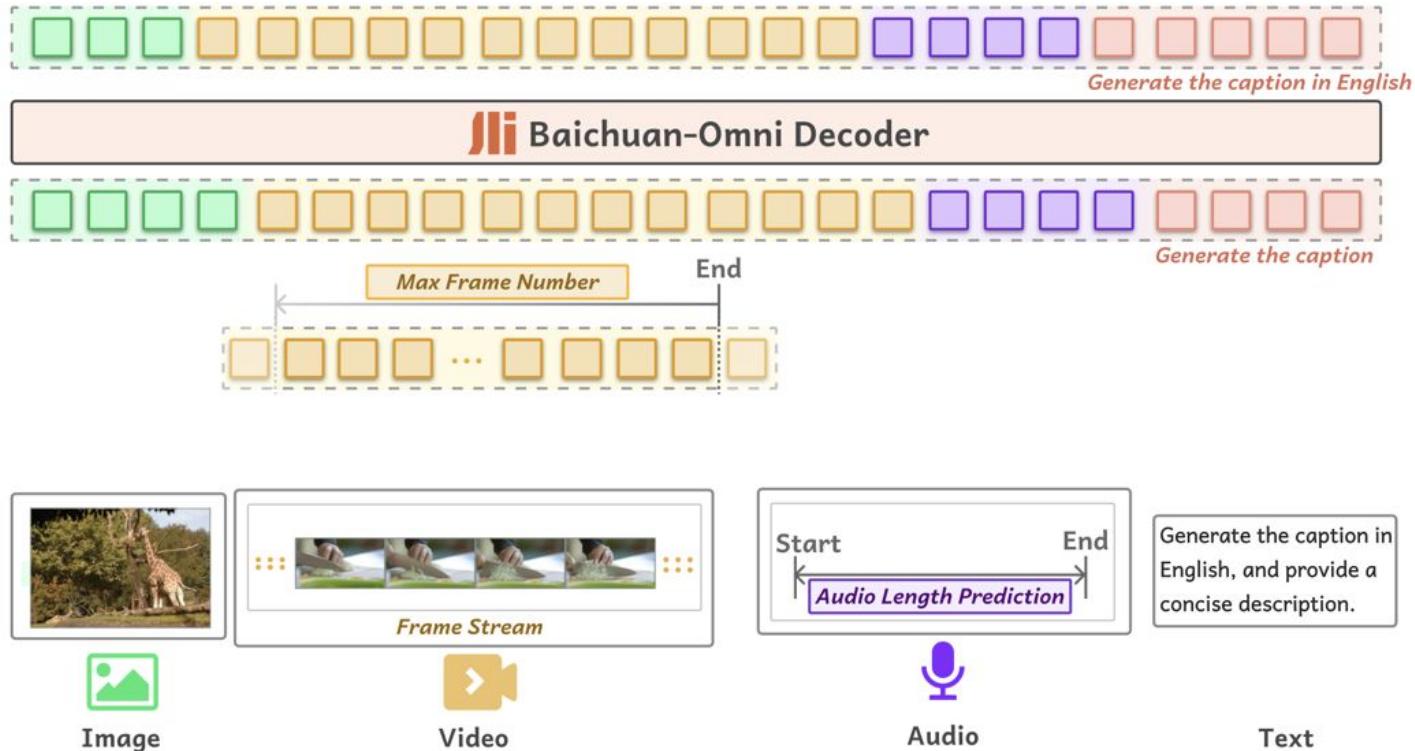
MM1.5: Methods, analysis & insights



<https://arxiv.org/abs/2409.20566>
Understanding Multimodal LLMs by Sebastian Raschka

Baichuan -Omni

Baichuan-Omni uses the Unified Embedding Decoder Architecture setup



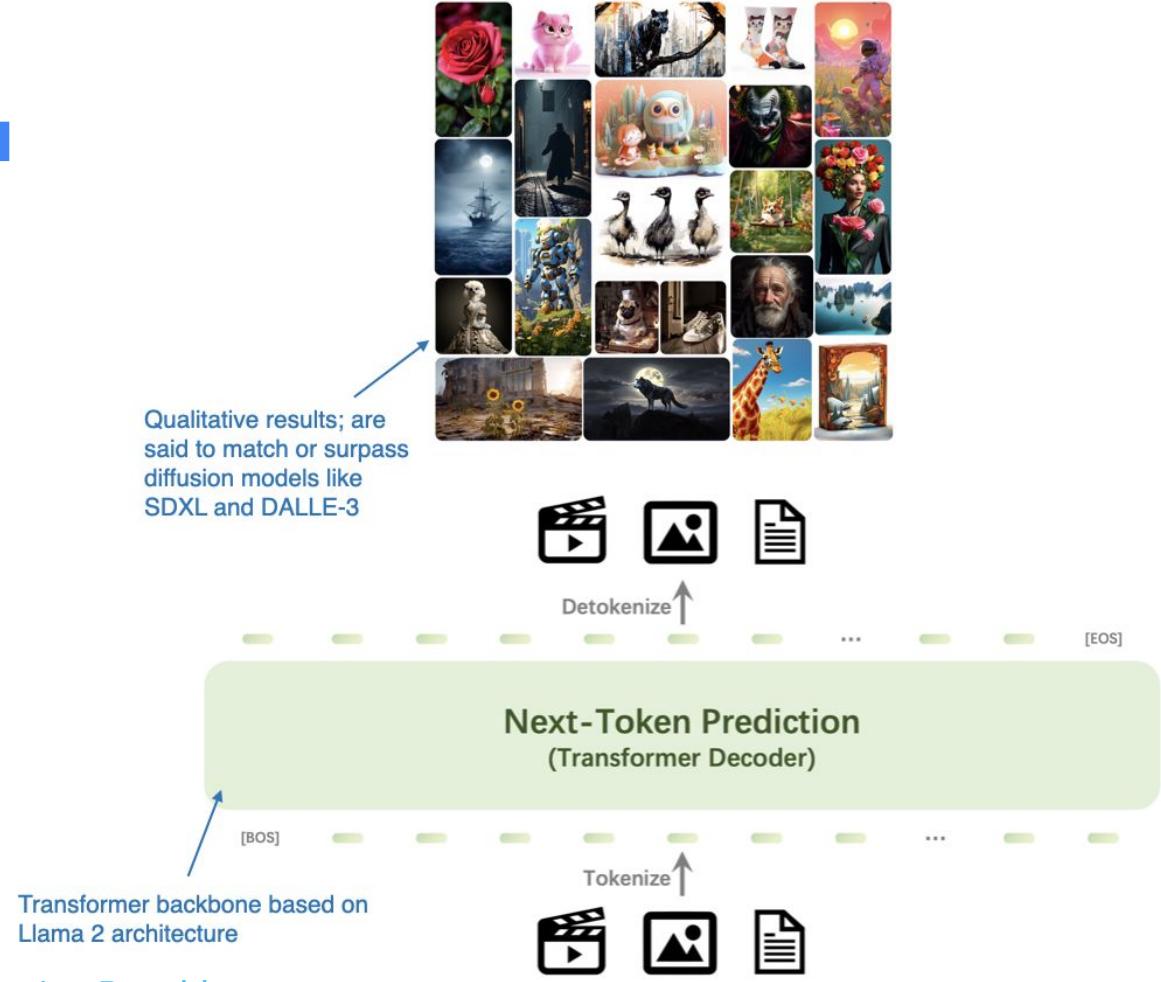
Baichuan-Omni (cont'd)

- **Projector training:** Initially, only the projector is trained, while both the vision encoder and the language model (LLM) remain frozen.
- **Vision encoder training:** Next, the vision encoder is unfrozen and trained, with the LLM still frozen.
- **Full model training:** Finally, the LLM is unfrozen, allowing the entire model to be trained end-to-end.

<https://arxiv.org/abs/2410.08565>

[Understanding Multimodal LLMs by Sebastian Raschka](#)

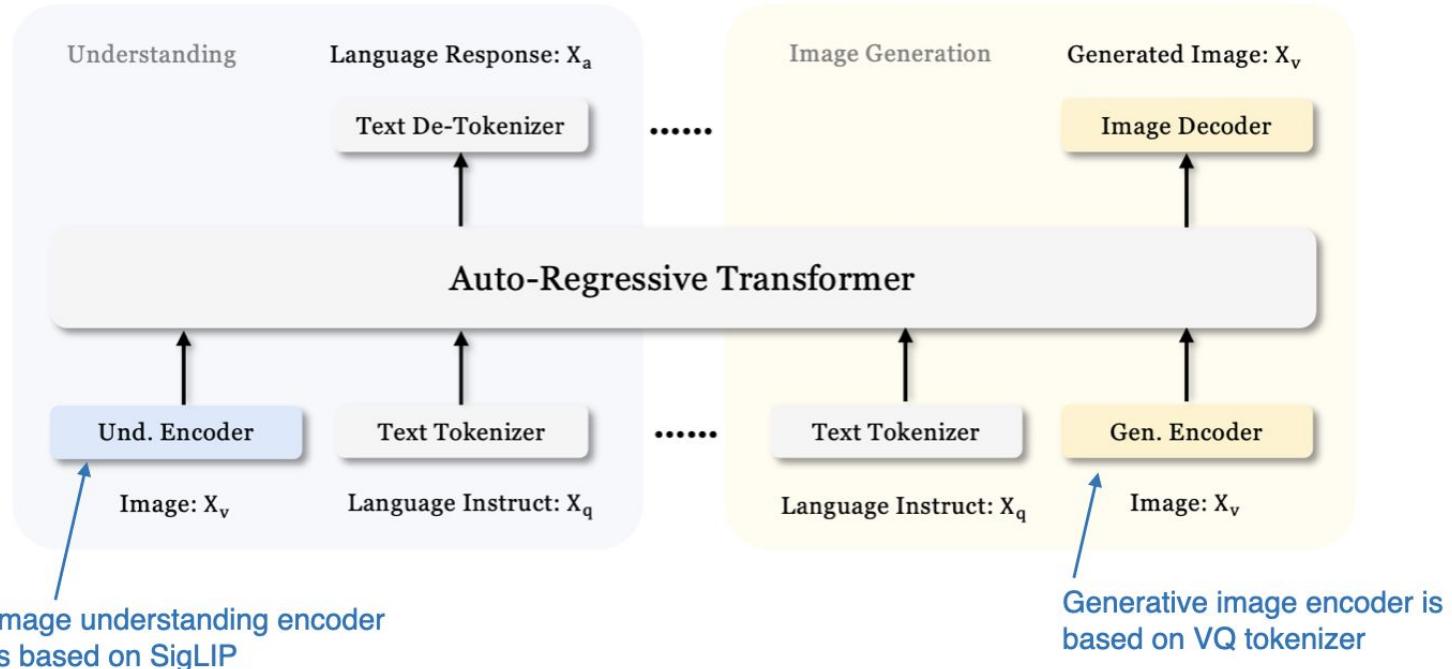
Emu3: Next-token prediction is all you need



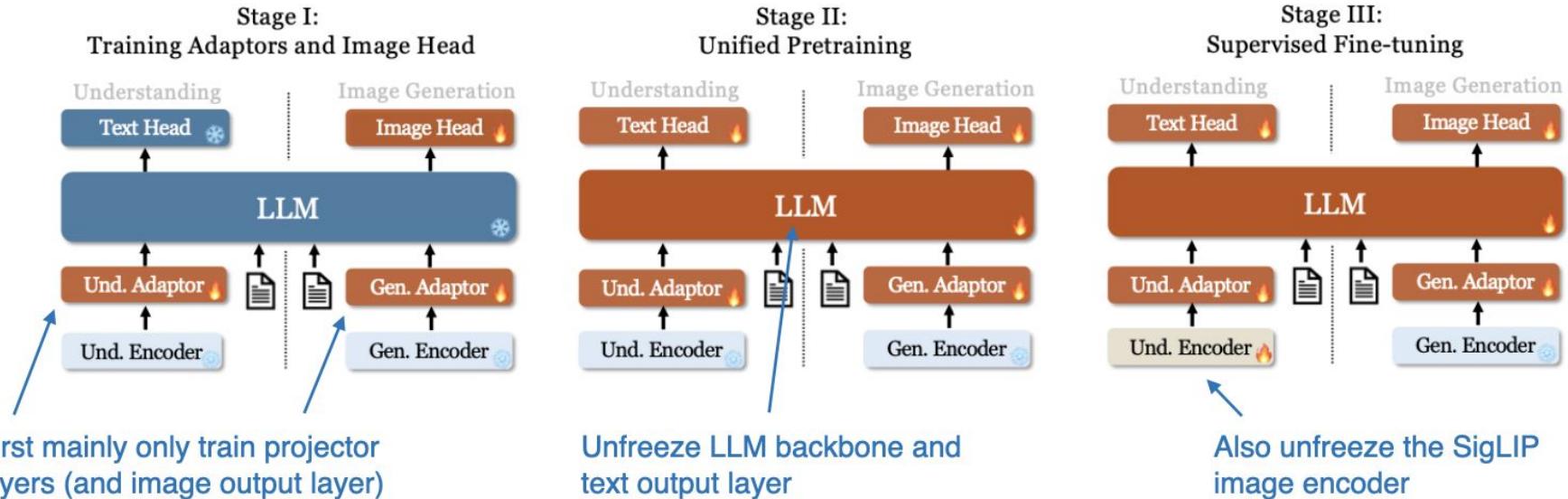
<https://arxiv.org/abs/2409.18869>

Understanding Multimodal LLMs by Sebastian Raschka

Janus: Decoupling Visual Encoding for Unified Multimodal Understanding and Generation



Janus: Decoupling Visual Encoding for Unified Multimodal Understanding and Generation



Thank you!