Meta

DeepLearning.AI

One Pager on Llama 3 (&2)

	Llama 2.0 (7B, 13B, 70B)	Llama 3.0 (8B, 70B)	Llama 3.1 (8B, 70B, 405B)	Llama 3.2 Multimodal (11B & 90B)	Llama 3.2 Lightweight Text Only (1B & 3B)
Release Date	July 18, 2023	April 18, 2024	July 23, 2024	Sep 25, 2024	Sep 25, 2024
Context Window	4K	8K	128K	128K	128K
Vocabulary Size	32K	128K	128K	128K	128K
Official Multilingual	English Only	English Only	8 Languages	8 Languages	8 Languages
Tool Calling	No	No	Yes	Yes	Yes
Knowledge Cutoff	Sep 2022	2023, Mar (8B) Dec (70B)	Dec 2023	Dec 2023	Dec 2023

Meta

DeepLearning.AI

Llama 3.1 models

	Finetuned	Tool use	Multilingual	Multimodal	Release
Llama 3.1 8B	No	No	Yes	No	July 2024
Llama 3.1 70B	No	No	Yes	No	July 2024
Llama 3.1 405B	No	No	Yes	No	July 2024
Llama 3.1 8B Instruct	Yes	Yes	Yes	No	July 2024
Llama 3.1 70B Instruct	Yes	Yes	Yes	No	July 2024
Llama 3.1 405B Instruct	Yes	Yes	Yes	No	July 2024

Meta

DeepLearning.AI

What is new in 3.1 and 3.2?

- **Tokenizer:** A new tokenizer with a vocabulary of 128k tokens.
- **Context window:** A larger context window of 128k tokens.
- **Languages:** Native support of 8 languages: English, German, French, Italian, Portuguese, Hindi, Spanish, and Thai.

Meta

DeepLearning.AI

What is new in 3.1 and 3.2?

- **Tokenizer:** A new tokenizer with a vocabulary of 128k tokens.
- **Context window:** A larger context window of 128k tokens.
- **Languages:** Native support of 8 languages: English, German, French, Italian, Portuguese, Hindi, Spanish, and Thai.
- **Tool calling:** Native support of tool calling.
- **Llama Stack:** A set of APIs and CLI for the entire Llama lifecycle, and the API providers and distributions.

Meta

DeepLearning.AI

What's new with Llama 3.2

1) Multimodal input in 11B and 90B models

- Image (objects, scenes, drawing) and OCR understanding
- Captioning and QA
- Visual reasoning (equations, charts, documents)

2) Smaller sizes in 1B and 3B text only models

- New SLM (Small Language Model) use cases:
  - on-device summarization
  - writing, translation
  - QA in multiple languages

Meta

DeepLearning.AI

Vision

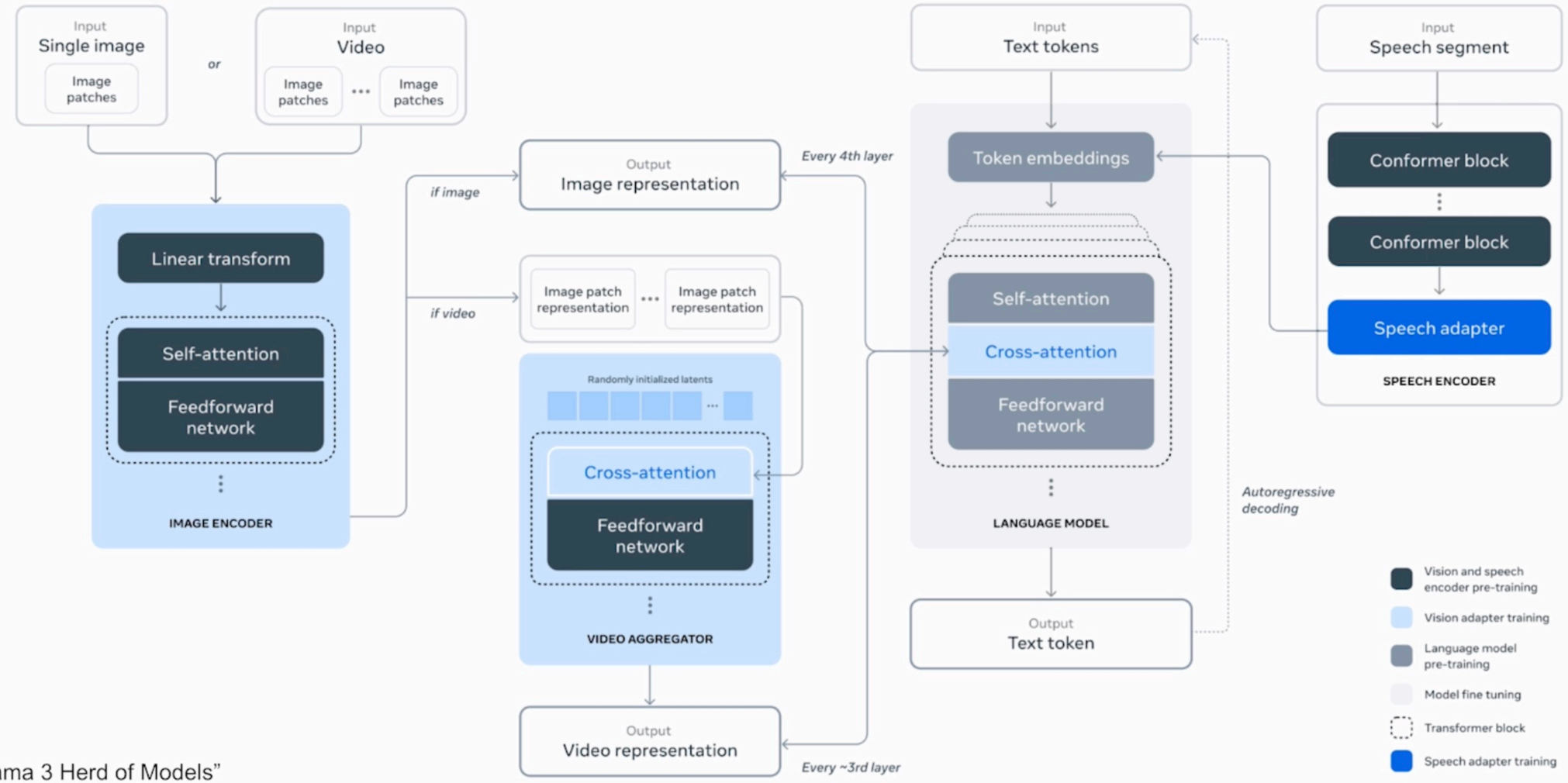


Image from “The Llama 3 Herd of Models”
https://ai.meta.com/research/publications/the-llama-3-herd-of-models/

Meta

DeepLearning.AI

Vision

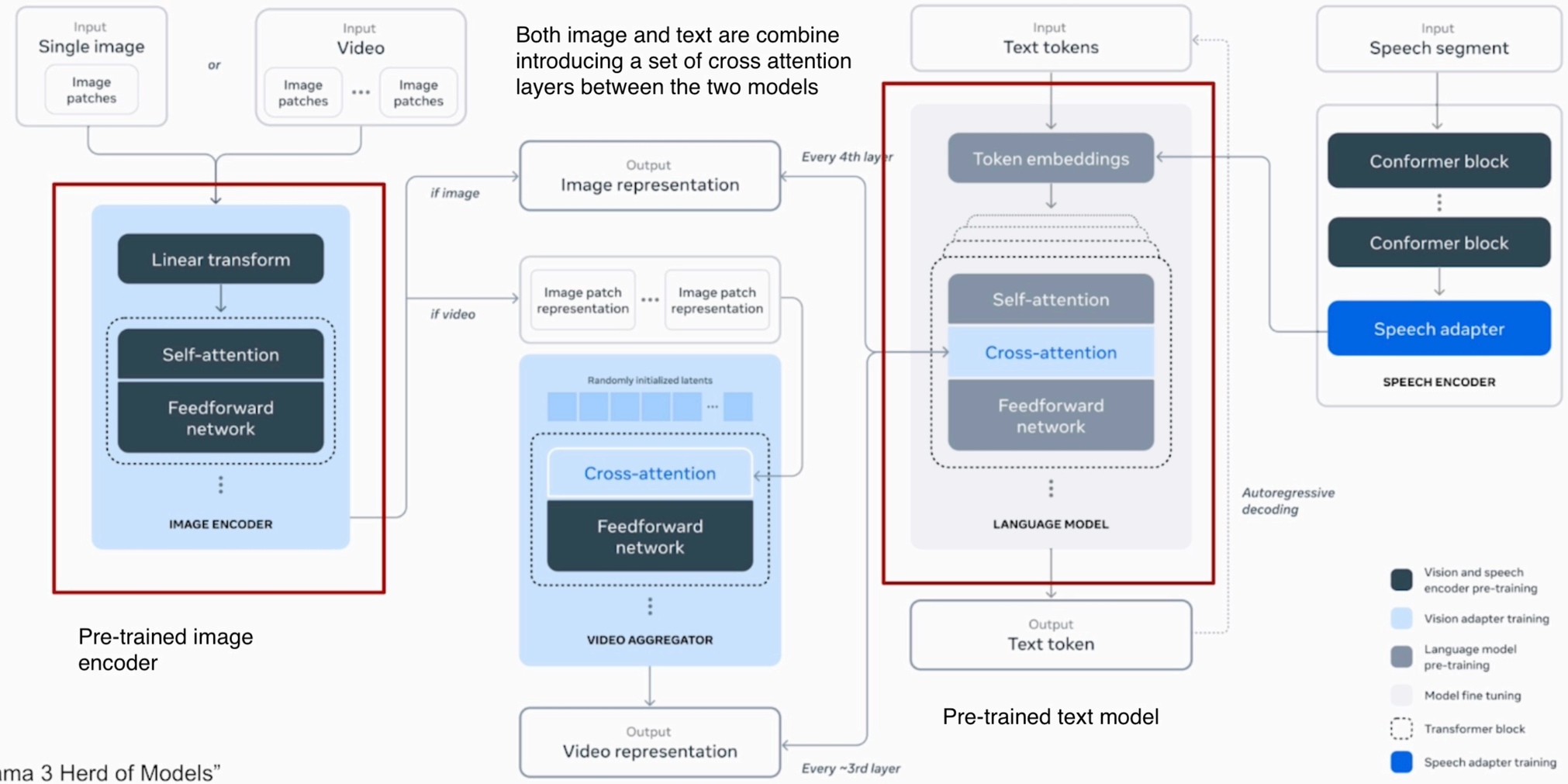


Image from “The Llama 3 Herd of Models”
https://ai.meta.com/research/publications/the-llama-3-herd-of-models/

Meta

DeepLearning.AI

Vision

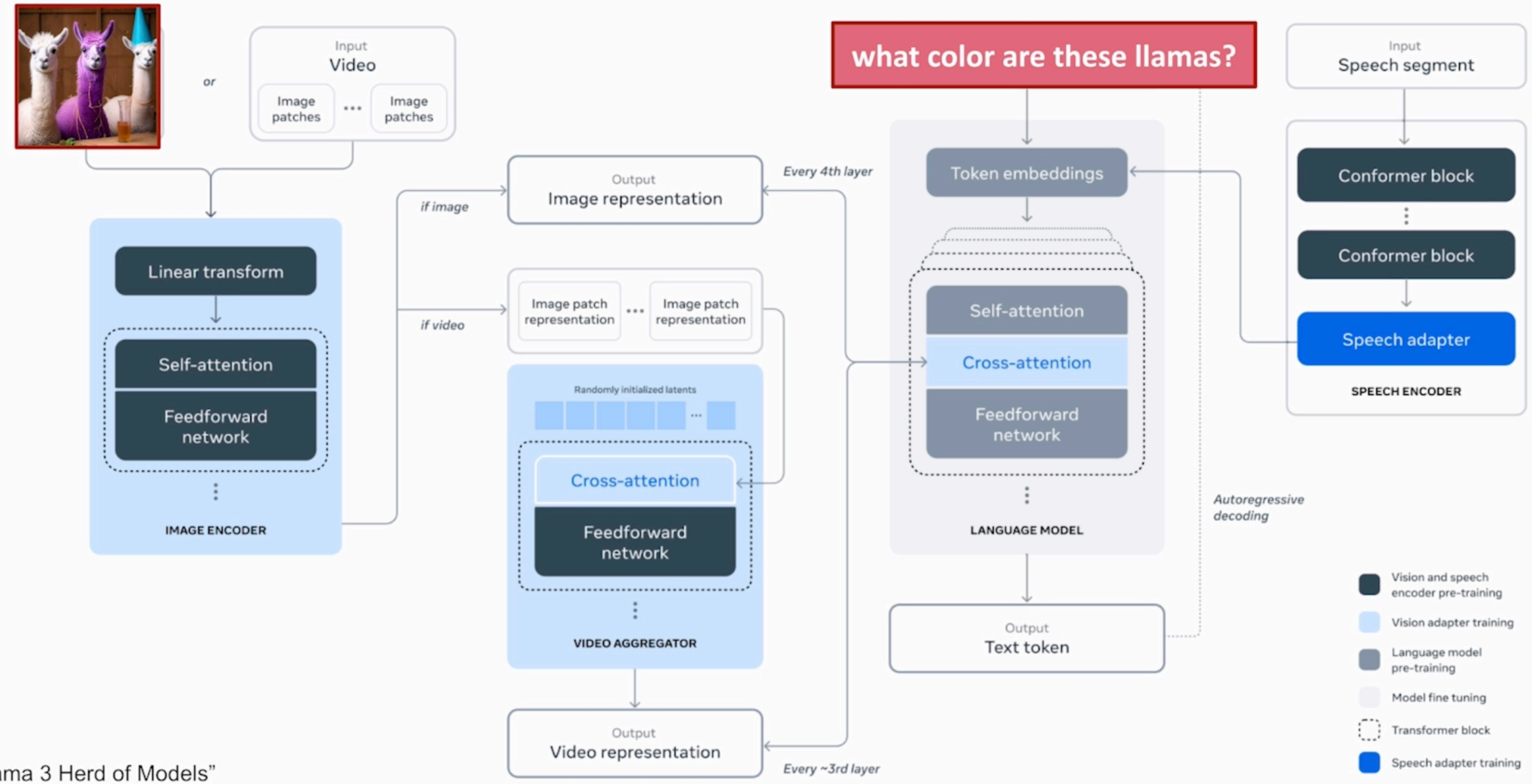


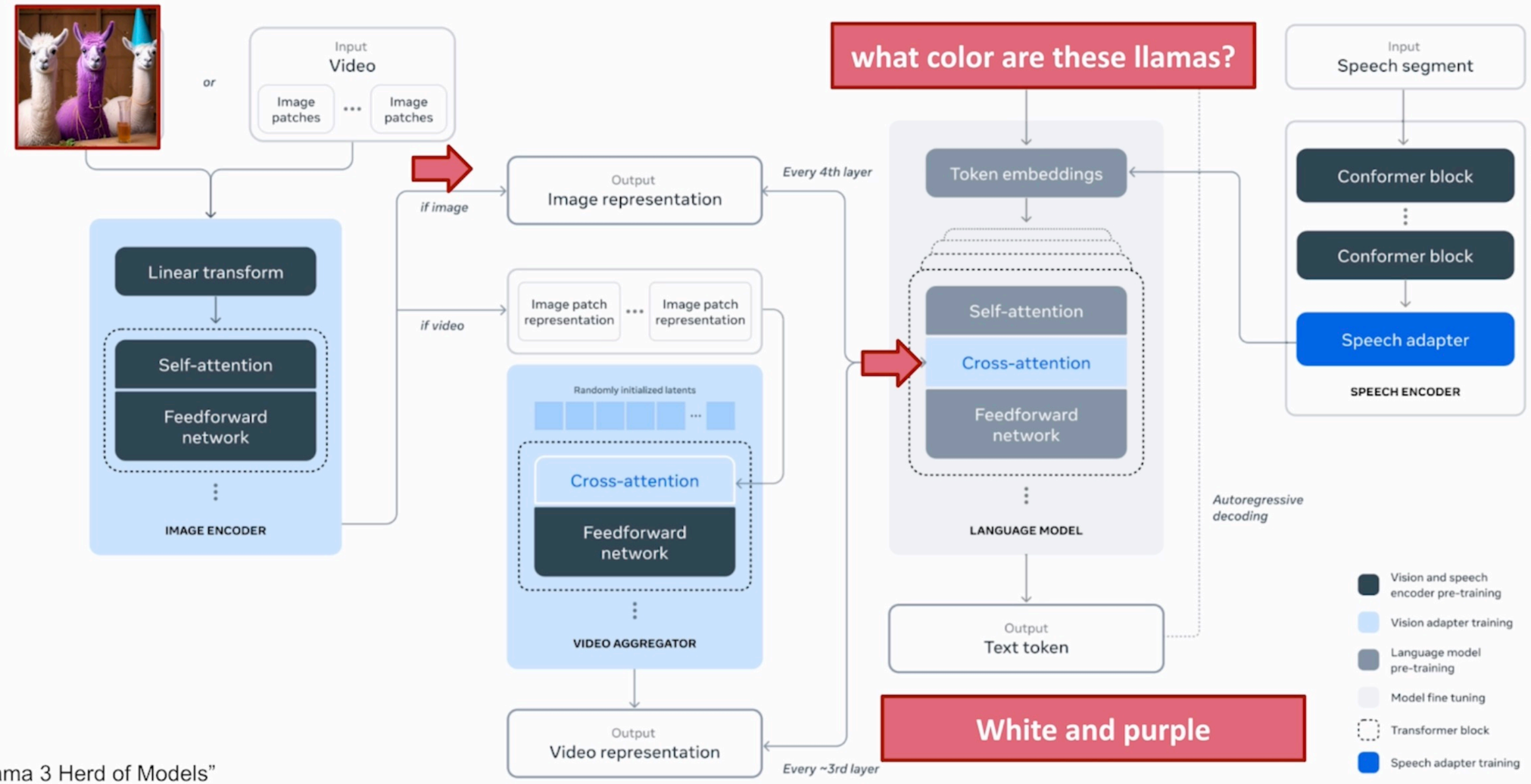
Image from “The Llama 3 Herd of Models”
https://ai.meta.com/research/publications/the-llama-3-herd-of-models/

Meta

DeepLearning.AI

Vision

During inference both image and text are provided to respective models
Image information is conveyed to the language model via cross attention and languge model produce the text response



Speech to text can be further added to the model via speech to text attention

Image from “The Llama 3 Herd of Models”
https://ai.meta.com/research/publications/the-llama-3-herd-of-models/

Meta

DeepLearning.AI

Llama 3.2 models

		Finetuned	Tool use	Multilingual	Multimodal	Release
Llama 3.2 **Base**	**MM 11B**	No	No	Yes	Yes (Image + Text input; Text Output)	Sep 2024
	MM 90B	No	No	Yes	Yes	Sep 2024
Llama 3.2 **Instruct**	**MM 11B**	Yes	Yes	Yes	Yes	Sep 2024
	MM 90B	Yes	Yes	Yes	Yes	Sep 2024
	1B	Yes	Yes	Yes	No	Sep 2024
	3B	Yes	Yes	Yes	No	Sep 2024

Meta

DeepLearning.AI

Llama 3.2 Benchmark Evals

Category	Benchmark	Llama 3.2 11B Base	Llama 3.2 90B Base
Image Understanding	VQAv2 (test-dev, 30k)	66.83	73.64
	Text VQA (val)	73.14	73.52
	DocVQA (val, unseen)	62.26	70.65
Visual Reasoning	MMMU (val, 0-shot)	41.67	49.33
	ChartQA (test)	39.4	54.16
	InfographicsQA (val, unseen)	43.21	56.79
	AI2 Diagram (test)	62.37	75.26

Meta

DeepLearning.AI

Llama 3.2 Benchmark Evals (cont.)

Category	Benchmark	Llama 3.2 **11B Instruct**	Llama 3.2 **90B Instruct**
College-level Problems and Mathematical Reasoning	MMMU (val, CoT)	50.7	60.2
	MMMU-Pro, Standard (10 opts, test)	33.0	45.2
	MMMU-Pro, Vision (test)	24.4	32.6
	MathVista (testmini)	51.5	57.9
Charts and Diagram Understanding	ChartQA (test, CoT)	83.4	86.4
	AI2 Diagram (test)	90.6	92.5
	DocVQA (test)	88.4	90.8
General Visual Question Answering	VQAv2 (test)	75.1	75.6

Meta

DeepLearning.AI

Llama 3.1 & 3.2 Benchmark Evals

	General MMLU (CoT)	General IFEval	Code HumanEval	Math GSM8K	ARC Challenge	Tool Use BFCL	Multi-lingual
Llama 3.1 8B	73.0	80.4	72.6	84.5	83.4	76.1	68.9
Gemma 2 9B	72.3	73.6	54.3	76.7	87.6	-	53.2
Llama 3.1 70B	86.0	87.5	80.5	95.1	94.8	84.8	86.9
GPT 3.5 Turbo	69.8	69.9	68.0	81.6	83.7	85.9	51.4
Llama 3.1 405B	88.6	88.6	88.6	96.8	96.9	88.5	91.6
GPT-4o	88.7	85.6	85.6	96.1	96.7	80.5	90.5

Meta

DeepLearning.AI

Running Llama Everywhere

In the cloud

AWS/Azure/GCP

Ollama

Locally

On-premise

On-device

Small 1b models

Meta

DeepLearning.AI

Llama 3.2 vision models

Pretrained:

- Llama-3.2 1B (text only)
- Llama-3.2 3B (text only)
- Llama-3.2 11B-Vision (text+image)
- Llama-3.2 90B-Vision (text+image)

Pre-trained can be used for image reasoning tasks

Instruction-tuned:

- Llama-3.2 1B-Instruct (text only)
- Llama-3.2 3B-Instruct (text only)
- Llama-3.2 11B-Vision-Instruct (text+image)
- Llama-3.2 90B-Vision-Instruct (text+image)

Insturtion-tuned models can used for Visual recognition, Image reasoning and answering general question about an image

Meta

DeepLearning.AI

Llama 3.2 text model

- Built on top of Llama 3.1 text-only models.
- The same tokenizer as Llama 3.1
- The same 128k context window.

Text capabilities:

- Llama 3.2 **11B** ⇔ Llama 3.1 **8B**
- Llama 3.2 **90B** ⇔ Llama 3.1 **70B**

Supported languages:

- The same as Llama 3.1.
- **For text only tasks:** English, German, French, Italian, Portuguese, Hindi, Spanish, and Thai.
- **For image+text applications:** Only English.

Example

<|begin_of_text|>

<|start_header_id|>

user

<|end_header_id|>

<|image|>Describe this image in two sentences.

<|eot_id|>

<|start_header_id|>

assistant

<|end_header_id|>

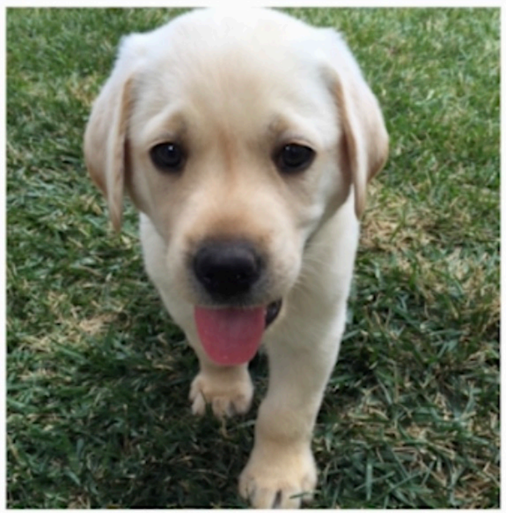
Use case

Notebook or hands-on for the below use cases

Counting the llamas

Plant recognition

Tire pressure warning

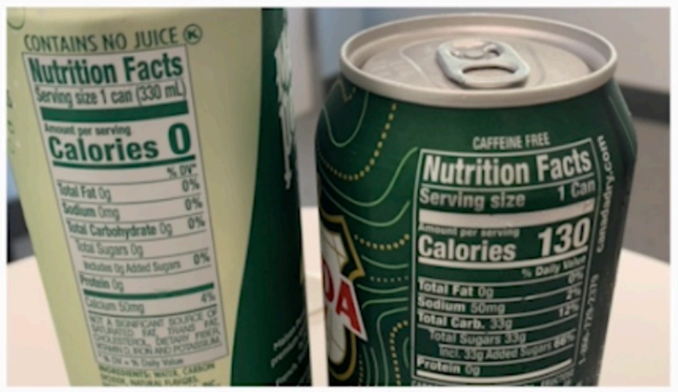


Dog breed recognition

Use case

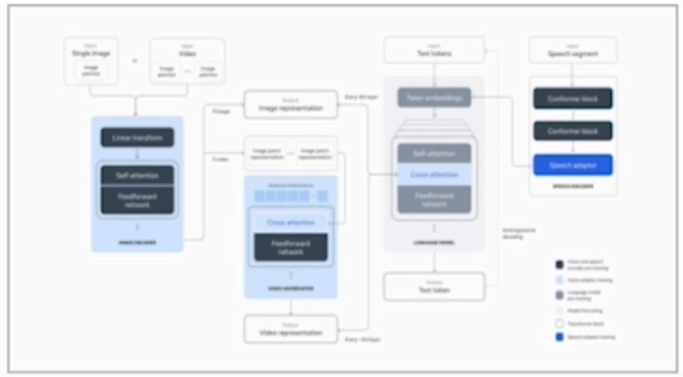
Analyzing multiple receipt images

Interior design assistant

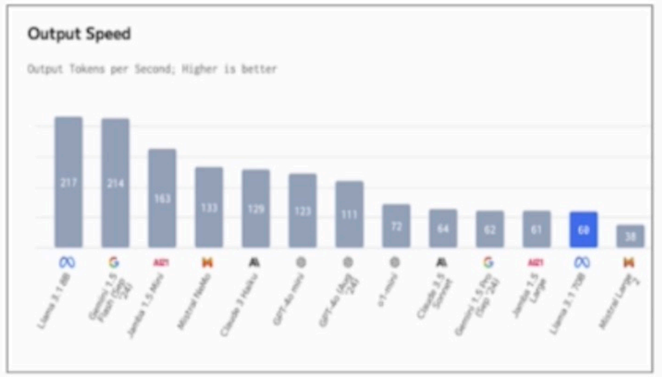


Nutrition facts to JSON

Use case

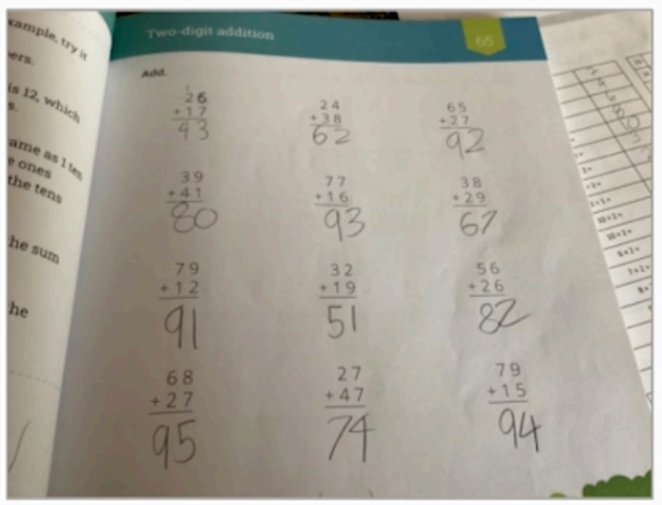


Convert diagram to code



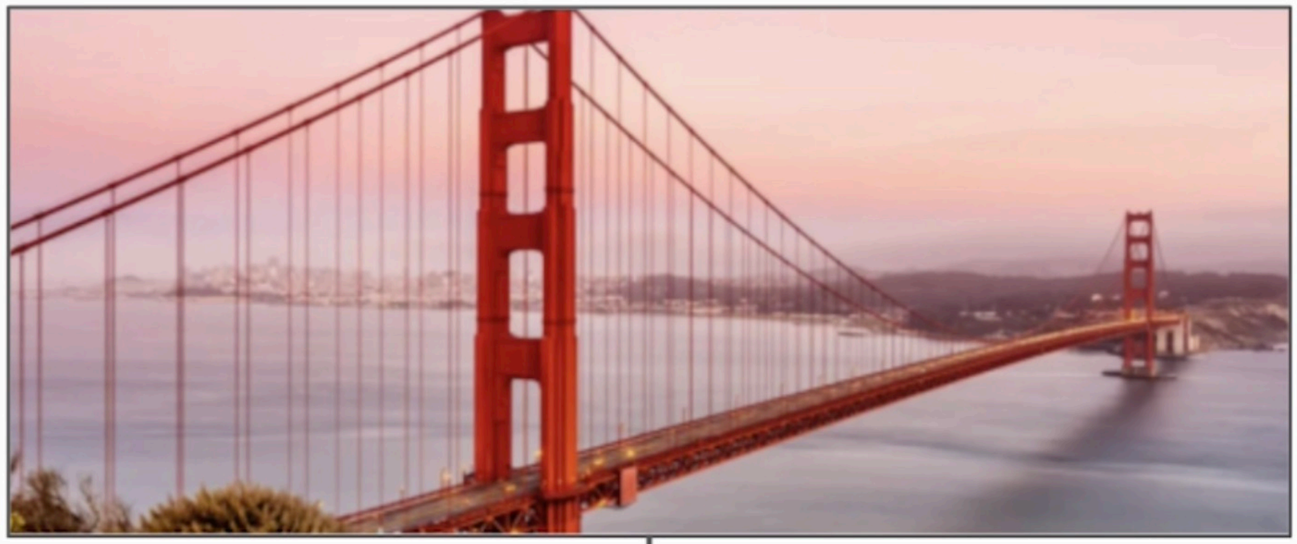
Convert plot to table

Analyze fridge content



Grade math homework

Use case: image to tool call



Where is this place?

What is the current weather at this place

```
<|python_tag|>
brave_search.call(
query="current weather in San Francisco")
```

Meta

DeepLearning.AI

The Llama 3.1 & 3.2 supported roles

system - Sets the context in which to interact with the AI model. It typically includes rules, guidelines, or necessary information that helps the model respond effectively.

user - Represents the human interacting with the model. It includes the inputs, commands, and questions to the model.

ipython - A new role introduced in Llama 3.1. Semantically, this role means "tool". This role is used to mark messages with the output of a tool call when sent back to the model from the executor.

assistant - Represents the response generated by the AI model based on the context provided in the 'system', 'ipython' and 'user' prompts.

Each role is set between the special tokens **<|start_header_id|>** and **<|end_header_id|>**.

Example

<|begin_of_text|>

<|start_header_id|>

user

<|end_header_id|>

Who wrote the book Charlotte's Web?

<|eot_id|>

<|start_header_id|>

<|end_header_id|>

Special tokens for single-turn and multi-turn chat

1. **<|begin_of_text|>**: Start of a prompt.

2. **<|start_header_id|>**: Start of a role for a particular message. Possible roles are: system, user, assistant and ipython.

3. **<|end_header_id|>**: End of the role for a particular message.

4. **<|eot_id|>**: End of a turn, which can be the end of the model's interaction with the user or a tool.

Special tokens for tool calling

5. **<|eom_id|>**: End of Message. A message represents a possible stopping point where the model can inform the execution environment that a tool call needs to be made.

6. **<|python_tag|>**: A special tag used in the model's response to signify a tool call.

Special tokens for fine-tuning and base model

7. **<|finetune_right_pad_id|>**: Used for padding text sequences in a batch to the same length.

8. **<|end_of_text|>**: Model will cease to generate more tokens after this. This token is generated only by the base models.

Using the "user" and "assistant" roles

```
question = "Who wrote the book Charlotte's Web?"

prompt = (
    "<|begin_of_text|>"                                  # start of prompt
    "<|start_header_id|>user<|end_header_id|>"           # user header
    f"{question}"                                        # user input
    "<|eot_id|>"                                         #end of turn
    "<|start_header_id|>assistant<|end_header_id|>"      #assistant header
)

print(prompt)
```

```
from  utils import llama31
response = llama31(prompt,8)
print(response)
```

```
from utils import cprint
response = llama31(prompt,8, raw=True)

cprint(response)
```

Meta

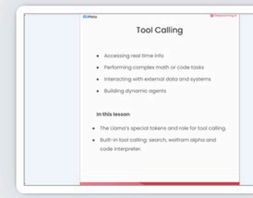
DeepLearning.AI

Tool Calling

- Accessing real time info
- Performing complex math or code tasks
- Interacting with external data and systems
- Building dynamic agents

In this lesson

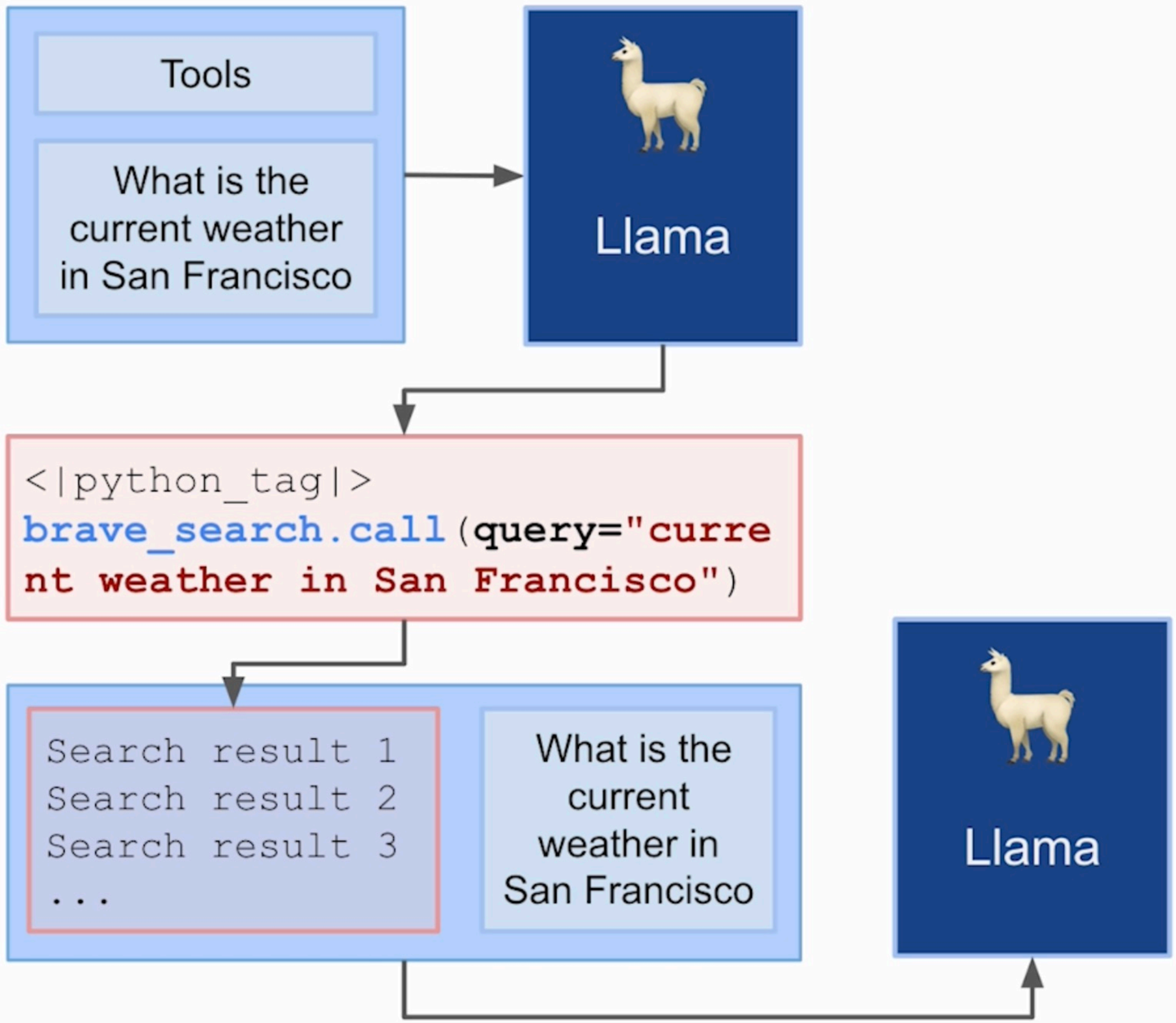
- The Llama's special tokens and role for tool calling.
- Built-in tool calling: search, wolfram alpha and code interpreter.
- Custom tool calling



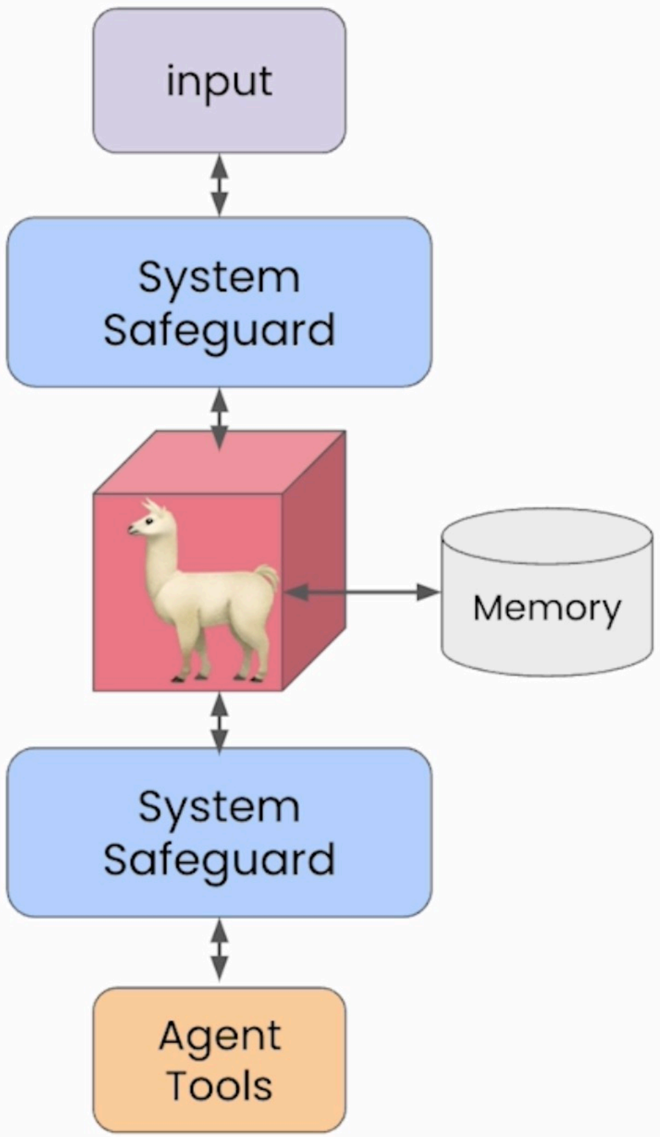
Meta

DeepLearning.AI

Use case: image to tool call



Models work in a system



AI System
- Multilingual safety models,
- a prompt injection filter
- Cybersecurity Evaluation Suite

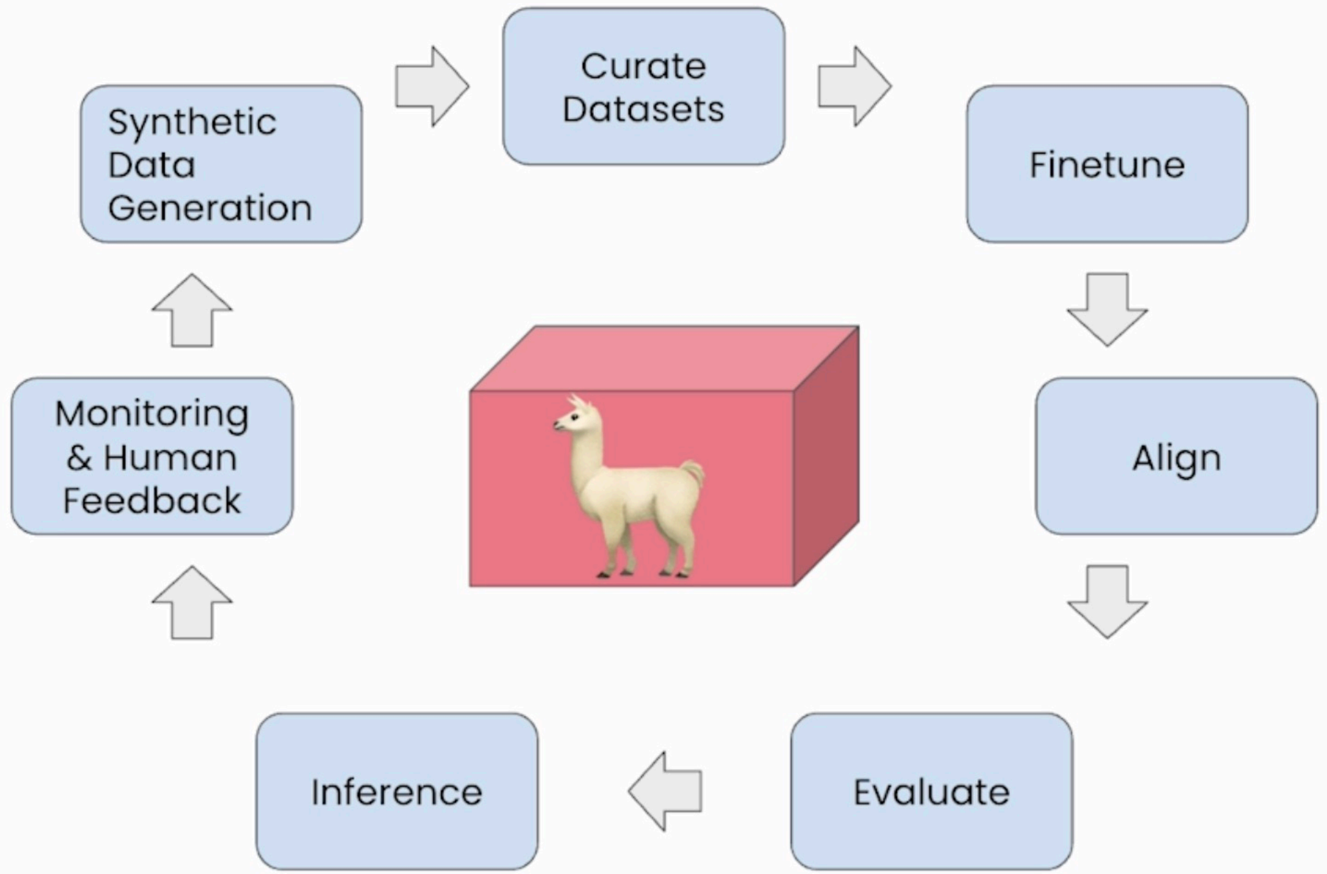
Agentic systems require
- external tool use
- memory

LLM reference System

Meta

DeepLearning.AI

Model Lifecycle



Meta

DeepLearning.AI

Llama Stack APIs

Agentic Apps
End applications

Agentic System API
System component orchestration

- PromptStore
- Assistant
- Shields
- Memory
- Orchestrator

Model Toolchain API
Model development & production tools

- Batch Inference
- Realtime Inference
- Quantized Inference
- Continual Pretraining
- Evals
- Fine Tuning
- Pretraining
- Reward Scoring
- Synthetic Data Gen

Data
Pretraining, preference, post training

Models
Core, safety, customized

Hardware
GPUs, accelerators, storage