IESTI05 – Edge AI

Machine Learning System Engineering

17. SLMs with Ollama Review

UNIFEI

Marcelo Rovai

Professor Honoris Causa

Transformers to LLMs and SLMs

Open - China

KIMI | Qwen | DeepSeek

Open

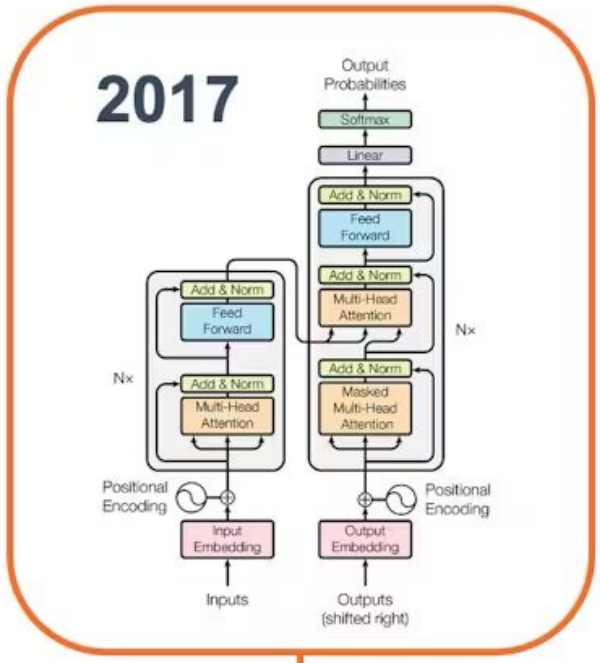
Phi | Gemma | LlaMa

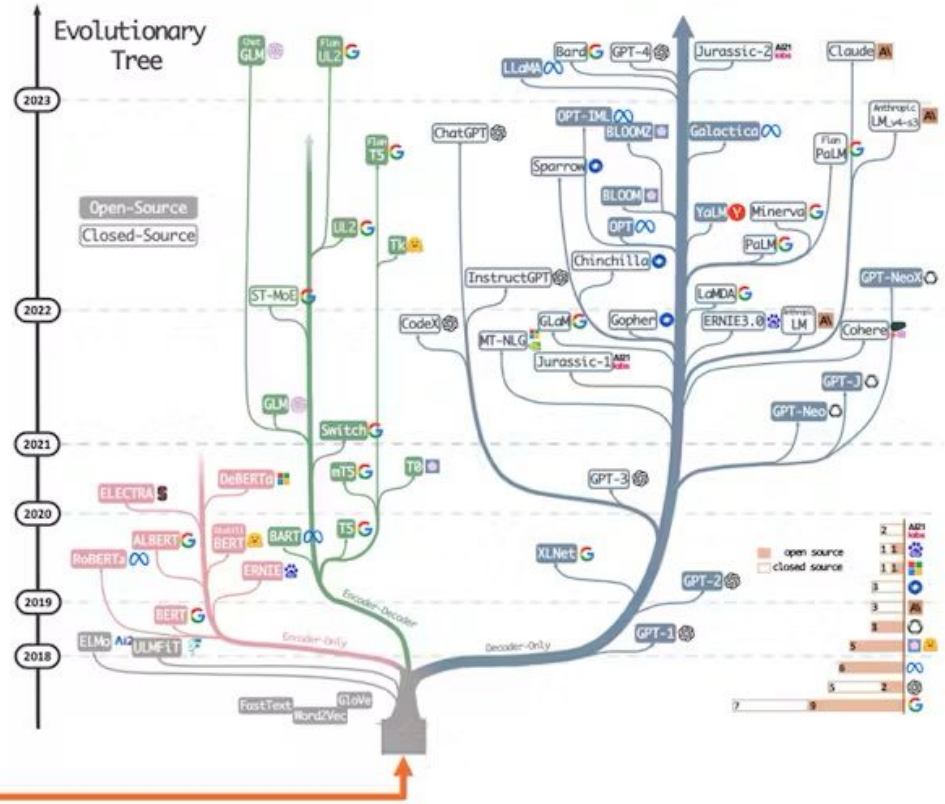
Closed

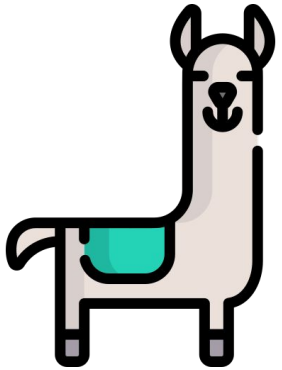
Gemini | GPT | Claude

2025

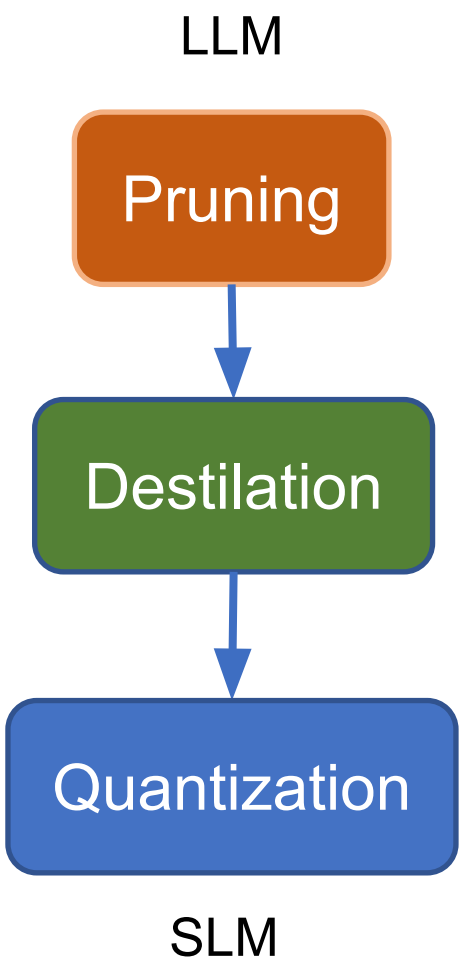
Grok

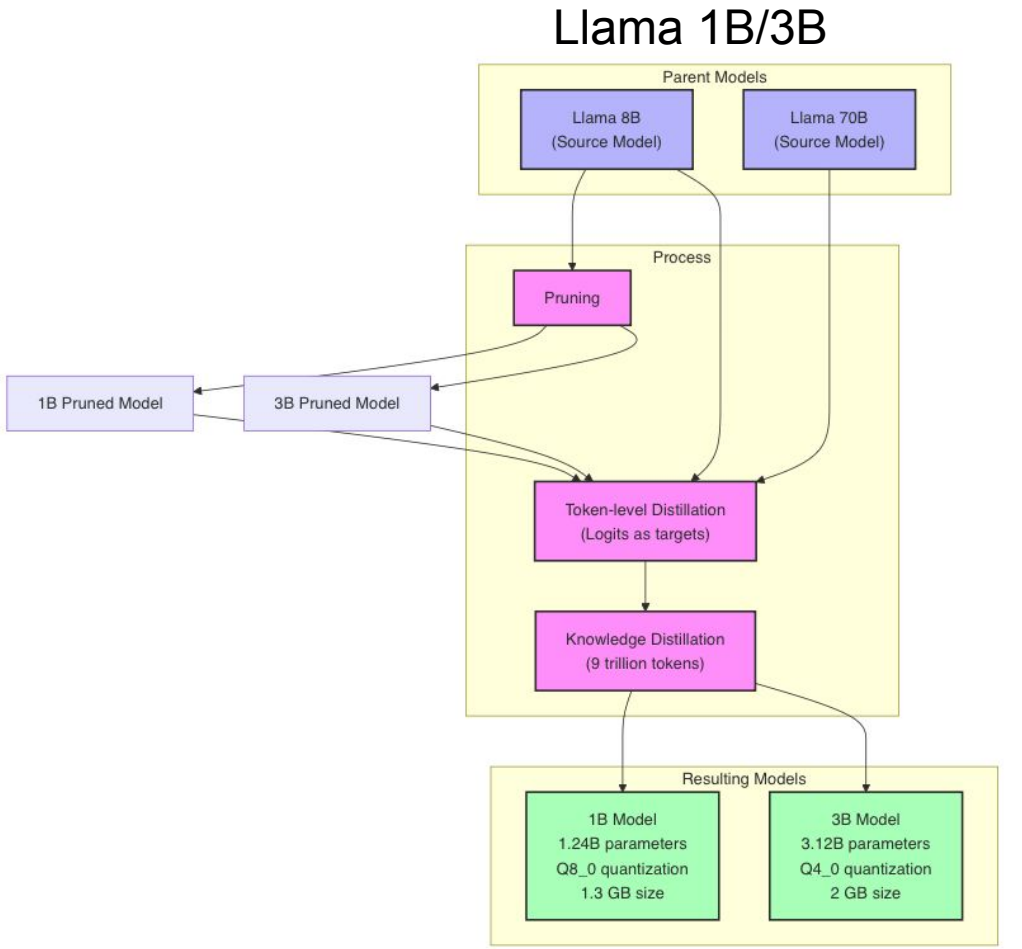


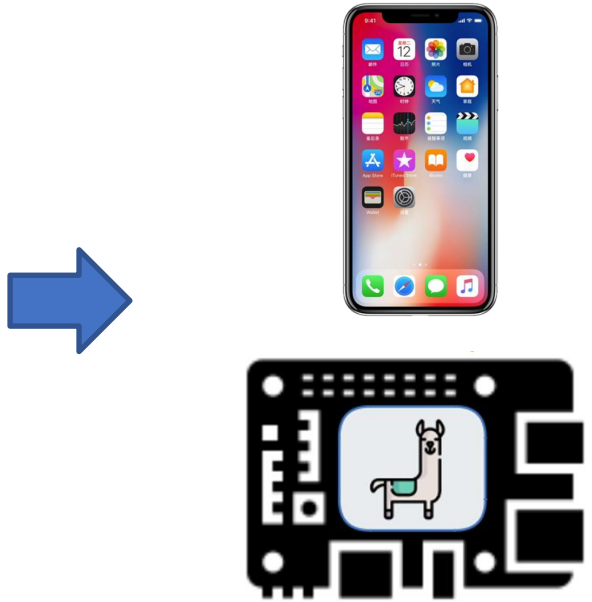




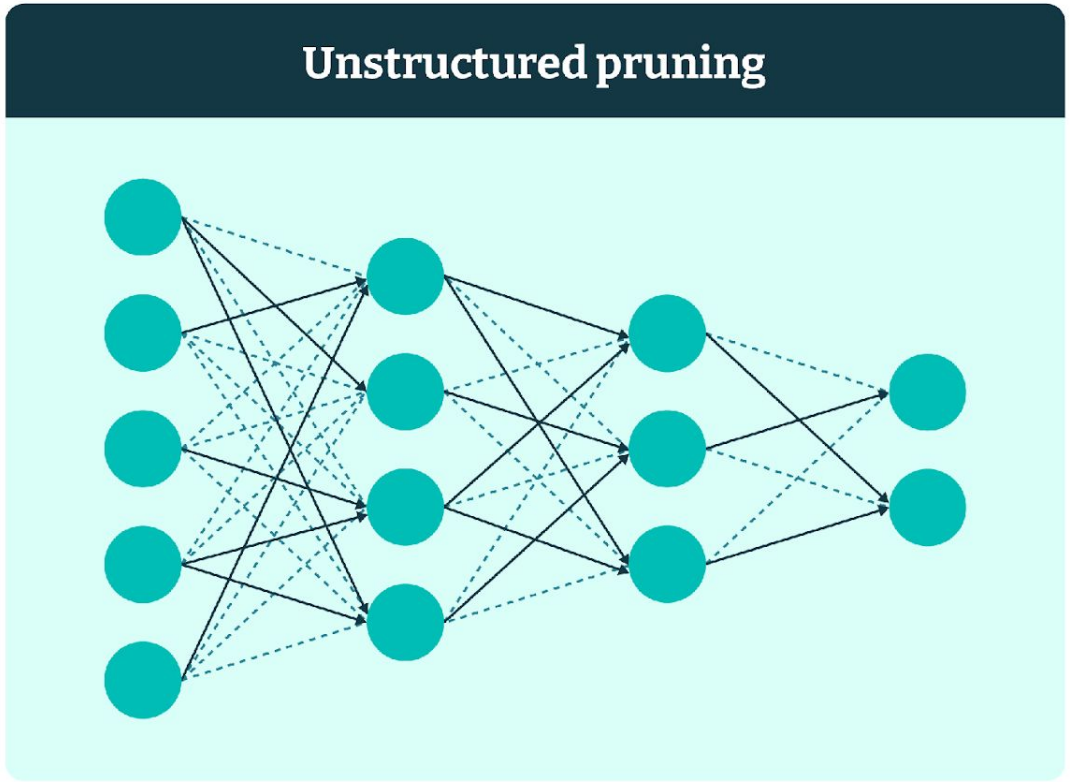
Small Language Models (SLMs)

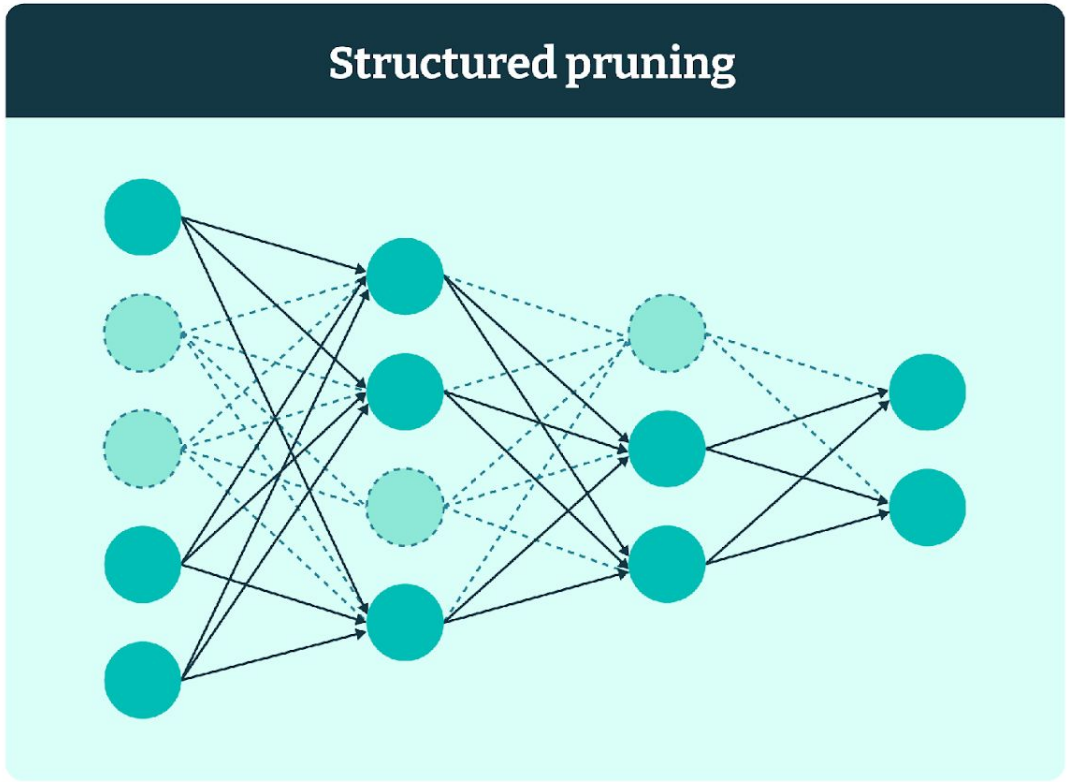




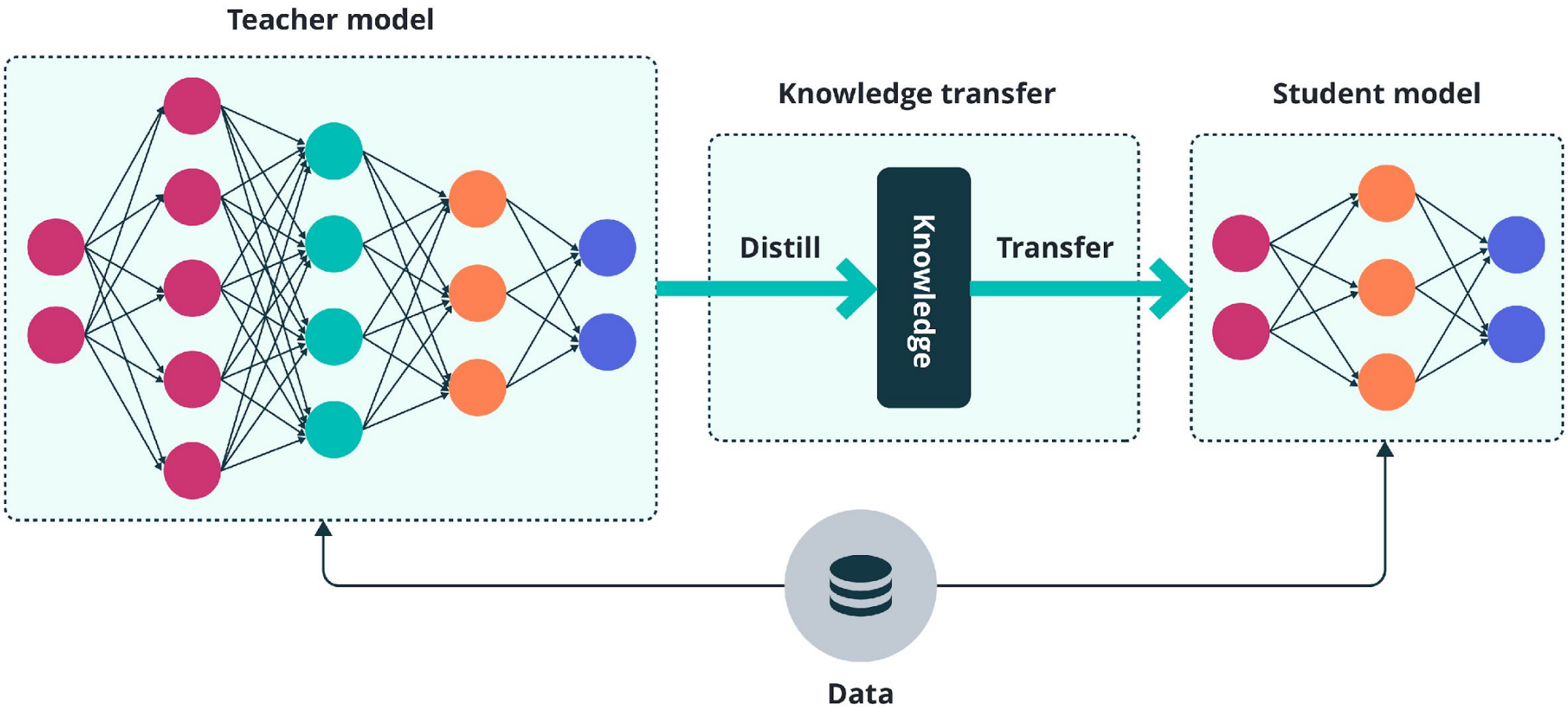


Pruning



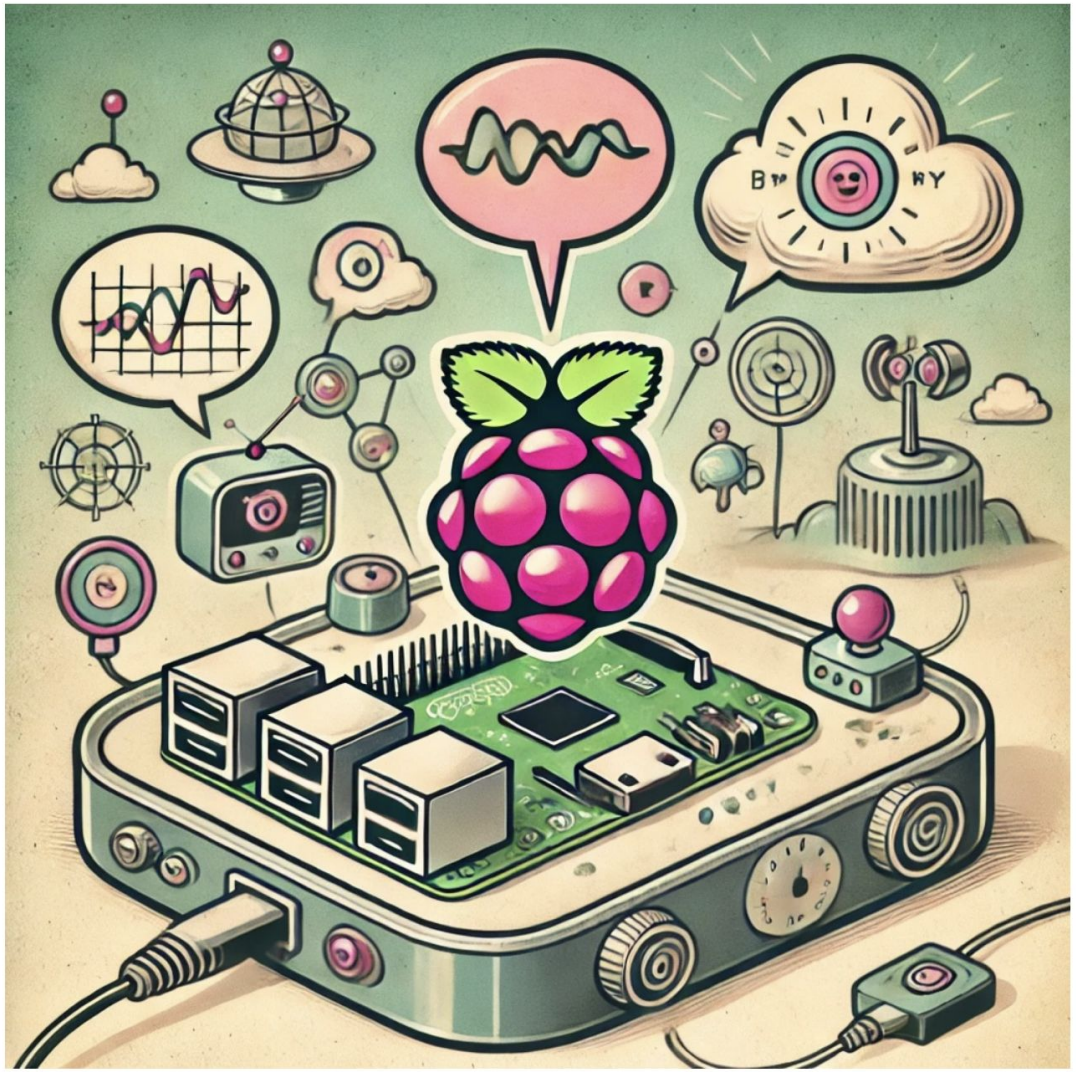


Knowledge Distillation (KD)



https://mjrovai.github.io/EdgeML_Made_Ease_ebook/raspi/kd_intro/kd_intro.html

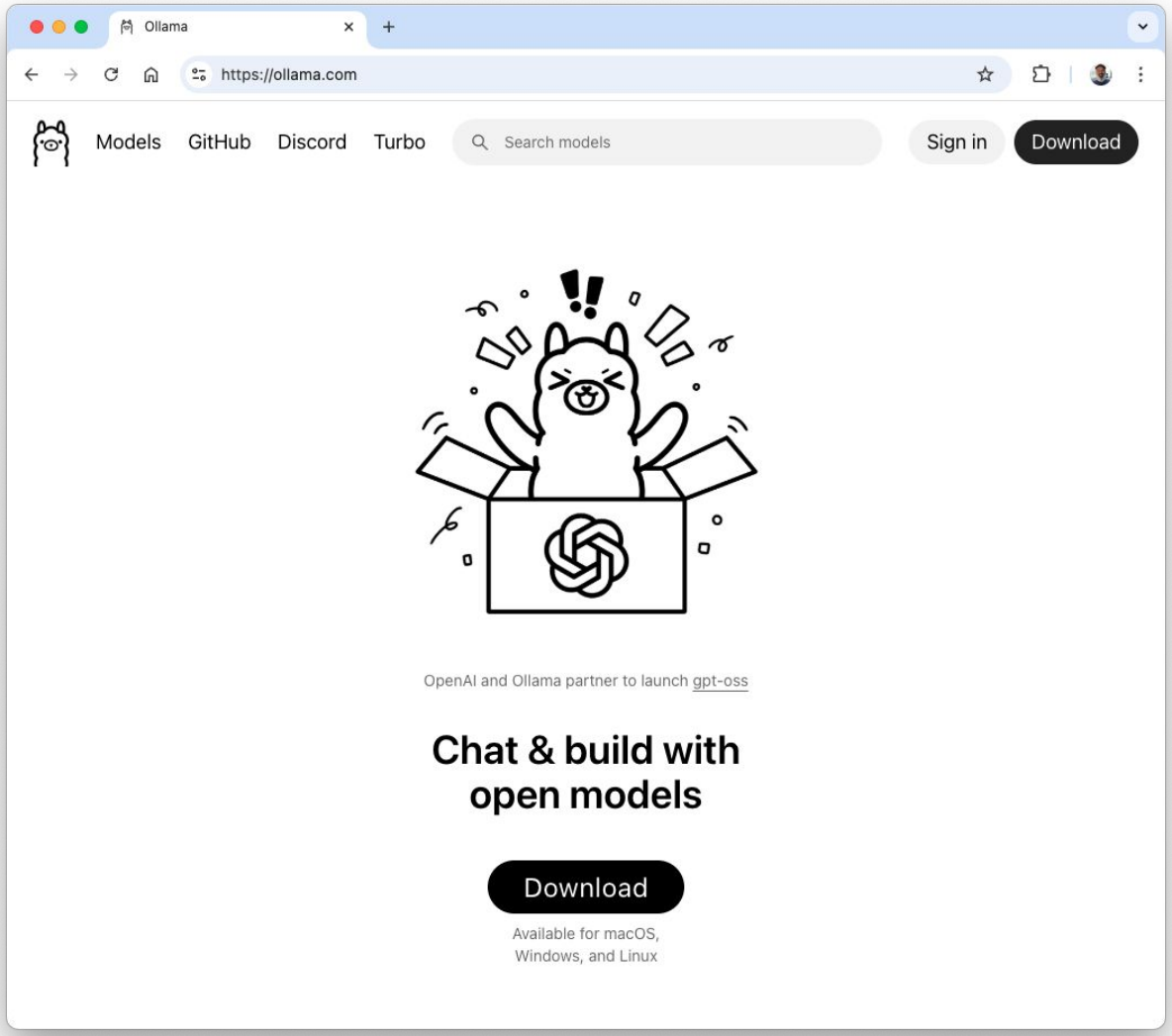
Small Language Models (SLM)



https://mjrovai.github.io/EdgeML_Made_Ease_ebook/raspi/llm/llm.html

Ollama

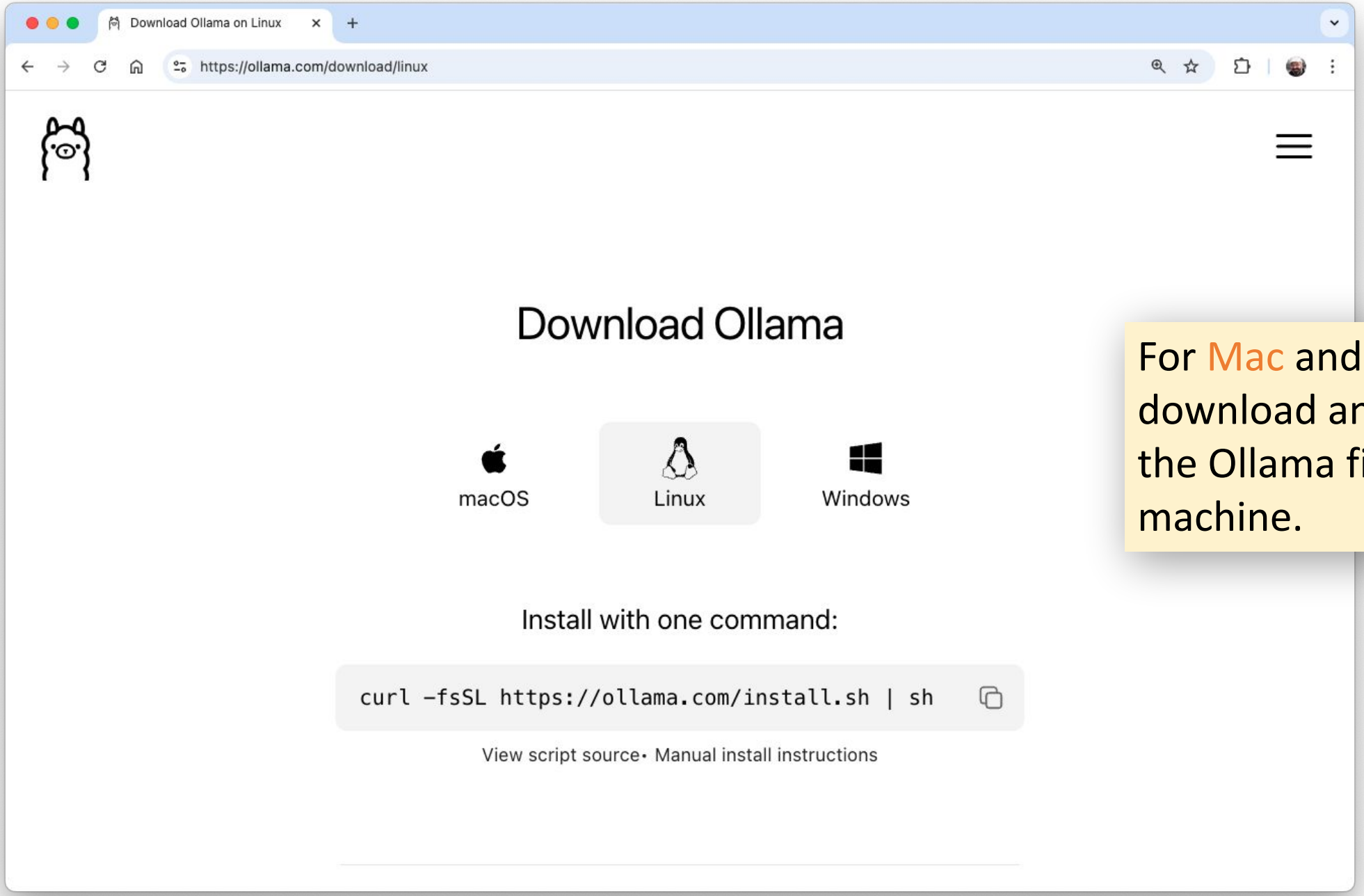
Running SLMs at the Edge



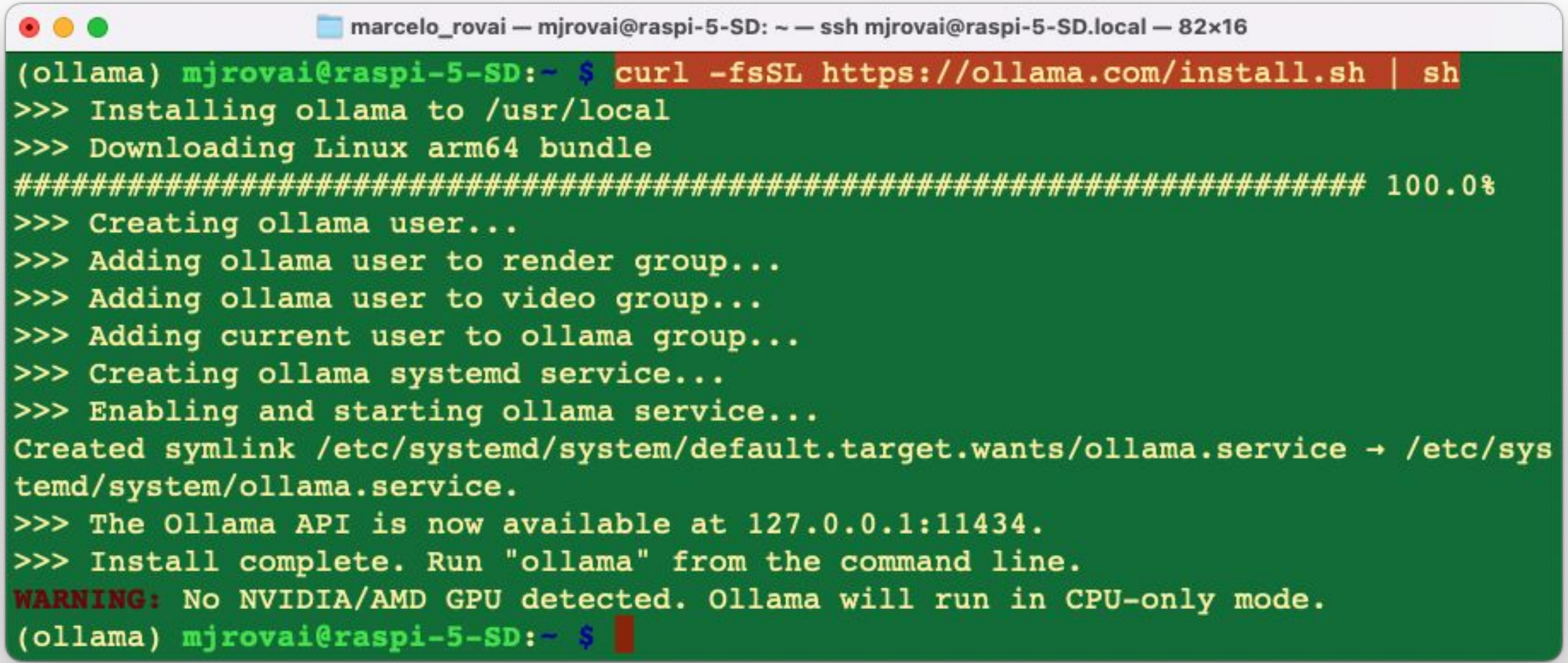
The primary and most user-friendly tool for running Small Language Models (SLMs) directly on a Raspberry Pi, especially the Pi 5, is Ollama. It is an open-source framework that allows you to install, manage, and run various SLMs (such as TinyLlama, smollm, Microsoft Phi, Google Gemma, Meta Llama, and LLaVa) locally on your Raspberry Pi for tasks like text generation and translation.

Alternatives options:

llama.cpp and the Hugging Face **Transformers** library are well-supported for running SLMs on Raspberry Pi. llama.cpp is particularly efficient for running quantized models natively, while Hugging Face Transformers offers a broader range of models and tasks, best suited for smaller architectures due to hardware limitations.



For Mac and Windows, download and execute the Ollama file on the machine.


marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 82×16
(ollama) mjrovai@raspi-5-SD:~ $ curl -fsSL https://ollama.com/install.sh | sh
>>> Installing ollama to /usr/local
>>> Downloading Linux arm64 bundle
100.0%
>>> Creating ollama user...
>>> Adding ollama user to render group...
>>> Adding ollama user to video group...
>>> Adding current user to ollama group...
>>> Creating ollama systemd service...
>>> Enabling and starting ollama service...
Created symlink /etc/systemd/system/default.target.wants/ollama.service → /etc/systemd/system/ollama.service.
>>> The Ollama API is now available at 127.0.0.1:11434.
>>> Install complete. Run "ollama" from the command line.
WARNING: No NVIDIA/AMD GPU detected. Ollama will run in CPU-only mode.
(ollama) mjrovai@raspi-5-SD:~ $

Llama 3.2:3B Architecture

3.2B parameters | 131K context | Q4_K_M quantized

1. Text Input

User prompt or conversation

```
Example: "Explain quantum computing"
```

2. Tokenization

Text → Token IDs using Llama tokenizer

- Byte-Pair Encoding (BPE)
- Vocabulary: ~128K tokens
- Output: [seq_len] token IDs

3. Token Embeddings

Convert tokens to dense vectors

```
Token IDs → Embedding lookup
Shape: [seq_len, 3072]
Embedding dimension: 3072
```

4. Transformer Decoder Layers

28 stacked decoder layers

Each layer contains:

- Multi-Head Self-Attention (GQA)
  - 24 query heads
  - 8 key-value heads (grouped)
  - Head dim: 128
- RMSNorm (pre-normalization)
- SwiGLU Feed-Forward Network
  - Intermediate dim: ~8192
- Residual connections

Output: [seq_len, 3072]

Key Features

- **RoPE** (Rotary Position Embeddings)
- **GQA** (Grouped Query Attention) for efficiency
- **RMSNorm** instead of LayerNorm
- **SwiGLU** activation in FFN
- **Causal masking** (autoregressive)

5. Final RMSNorm

Normalize final hidden states

```
[seq_len, 3072] → normalized
```

6. Language Modeling Head

Project to vocabulary & generate next token

```
Linear: [3072] → [~128K vocab]
Softmax → probability distribution
Sampling (temp, top-p, top-k)
→ "Quantum computing uses..."
```

⚡ Performance

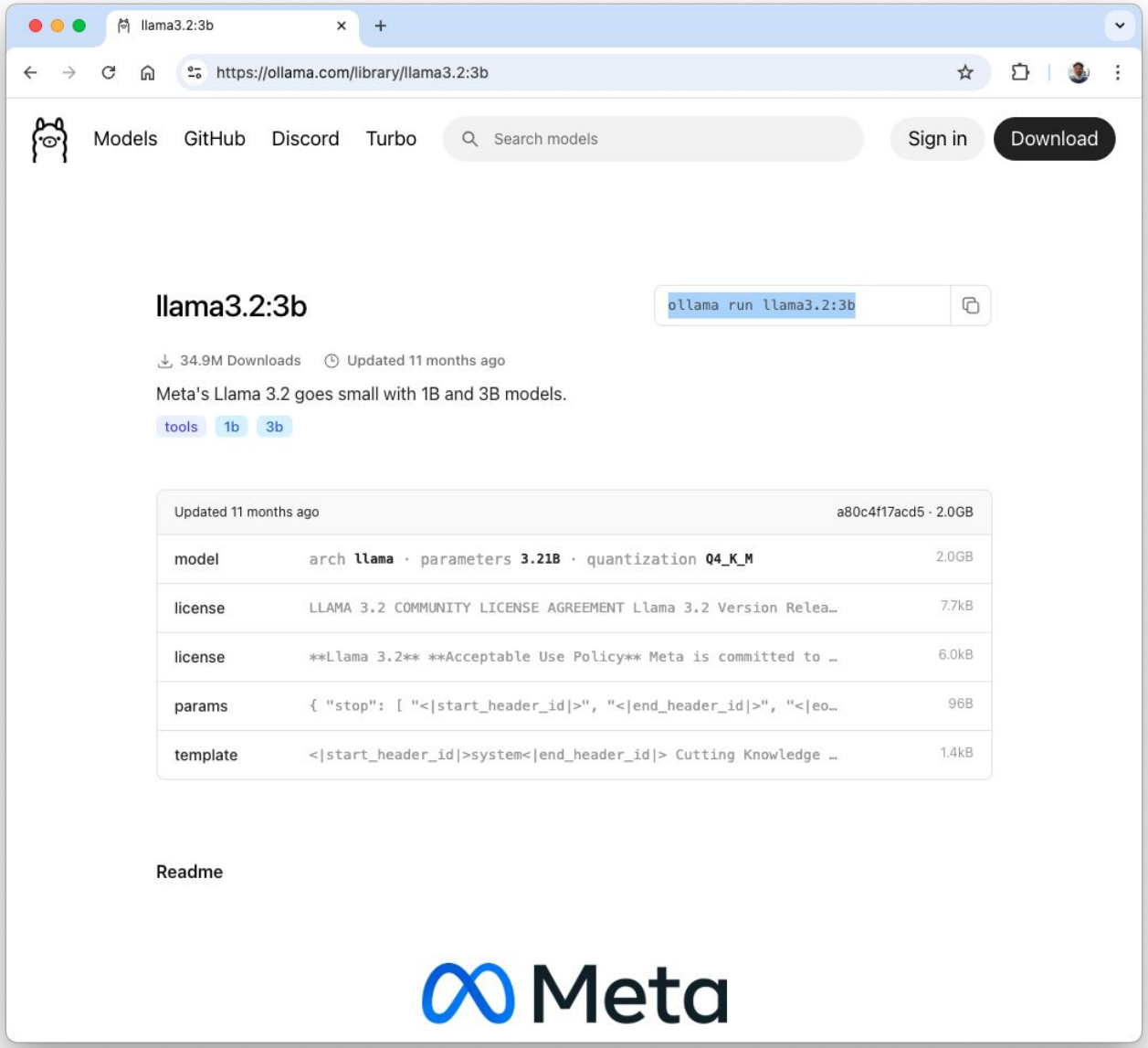
- **Context:** 131,072 tokens
- **Quantization:** Q4_K_M
- **Size:** ~2GB on disk

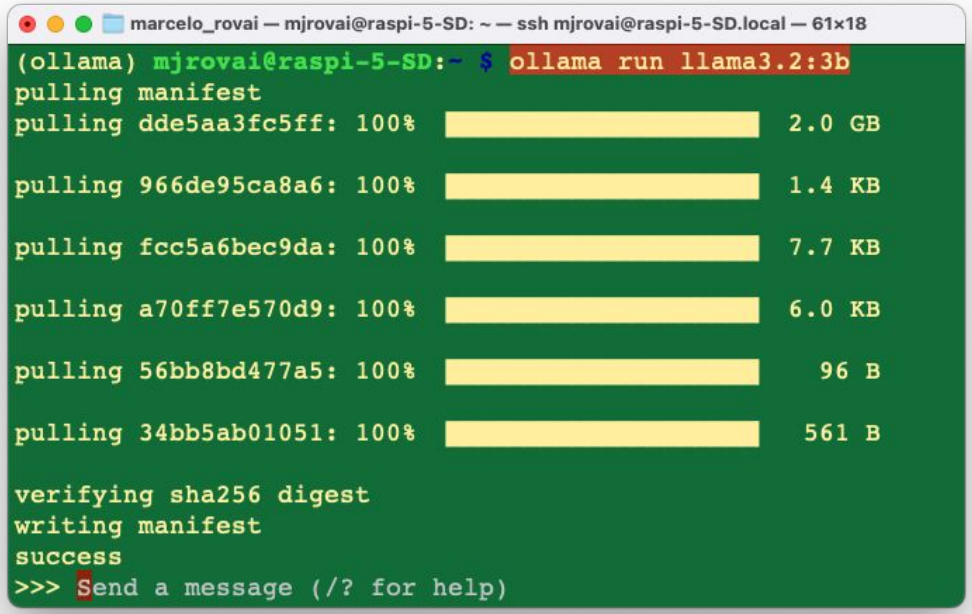
🎯 Capabilities

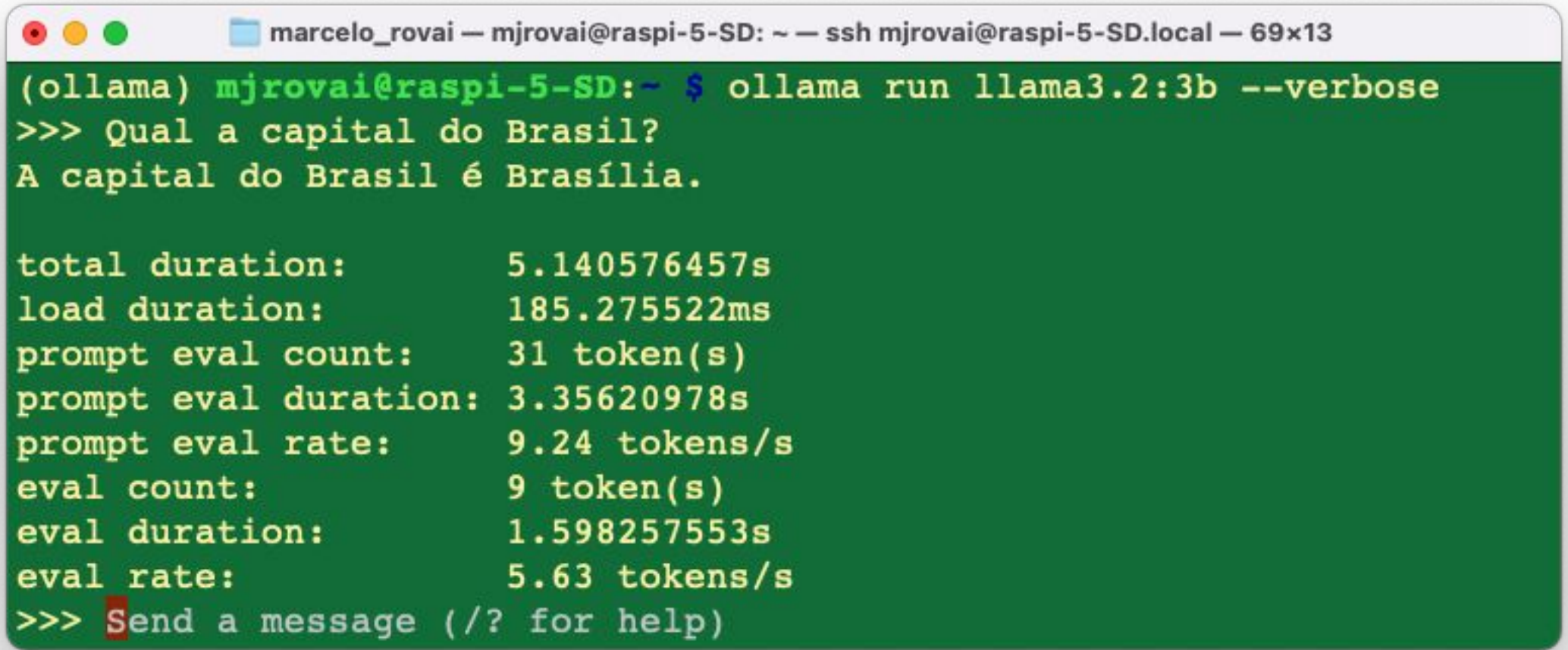
- Text completion
- Tool/function calling
- Long context reasoning

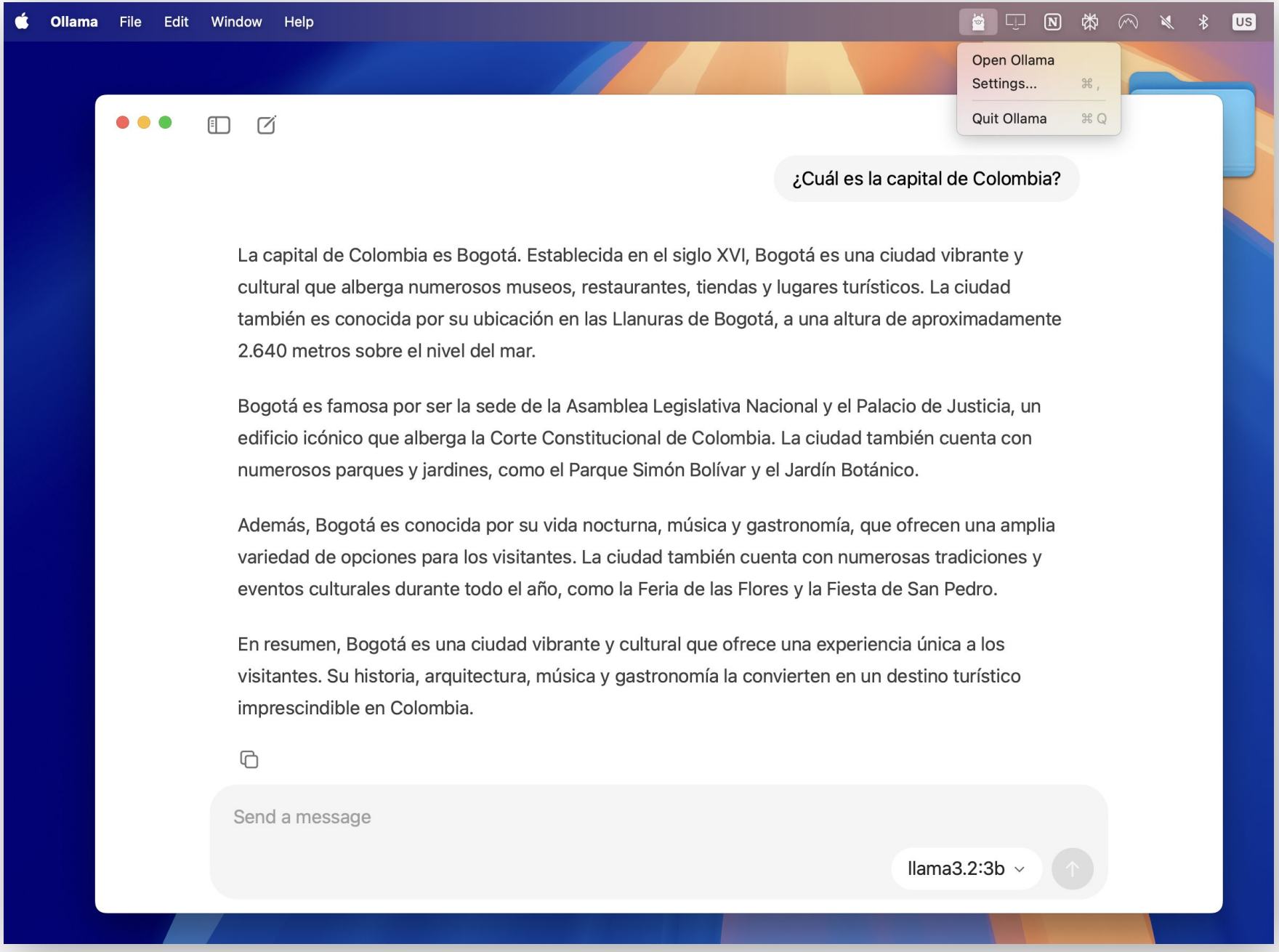
🔑 Key Insight

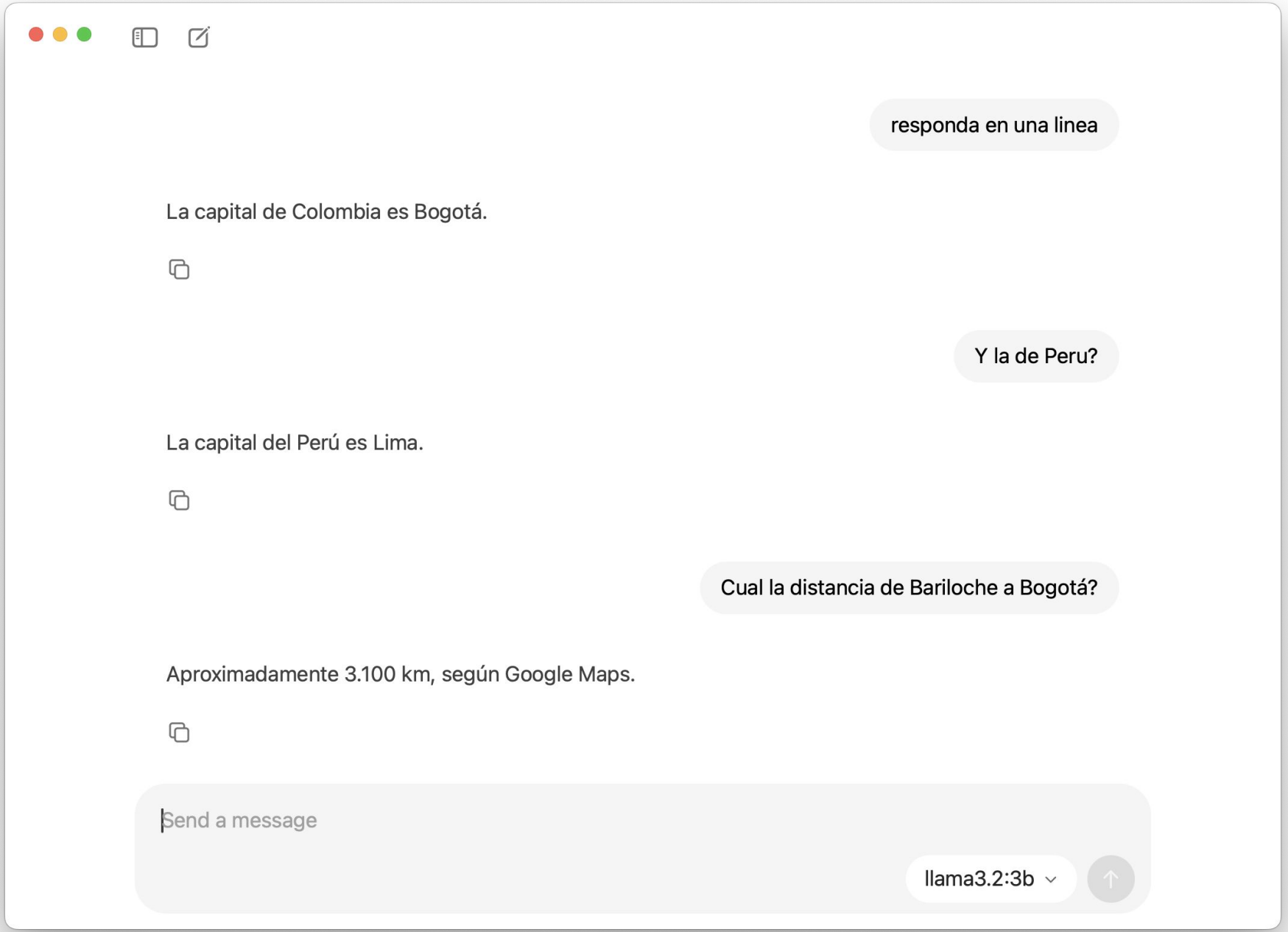
Llama 3.2:3B is a **text-only decoder model** using modern techniques like **Grouped Query Attention** and **SwiGLU** for efficiency. Unlike MoonDream, it has **no vision encoder** - it's pure language modeling.


llama3.2:3b
https://ollama.com/library/llama3.2:3b
Models
GitHub
Discord
Turbo
Search models
Sign in
Download
llama3.2:3b
ollama run llama3.2:3b
34.9M Downloads
Updated 11 months ago
Meta's Llama 3.2 goes small with 1B and 3B models.
tools
1b
3b
Updated 11 months ago
a80c4f17acd5 · 2.0GB
model
arch llama · parameters 3.21B · quantization Q4_K_M
2.0GB
license
LLAMA 3.2 COMMUNITY LICENSE AGREEMENT Llama 3.2 Version Relea…
7.7kB
license
Llama 3.2 **Acceptable Use Policy** Meta is committed to …
6.0kB
params
{ "stop": ["<|start_header_id|>", "<|end_header_id|>", "<|eo…
96B
template
<|start_header_id|>system<|end_header_id|> Cutting Knowledge …
1.4kB
Readme
Meta

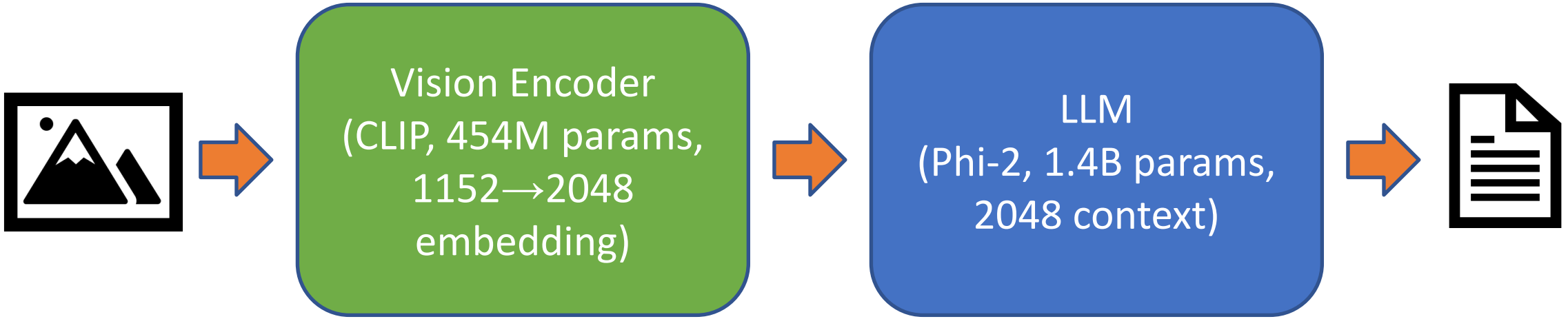

marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 61×18
(ollama) mjrovai@raspi-5-SD:~ $ ollama run llama3.2:3b
pulling manifest
pulling dde5aa3fc5ff: 100% 2.0 GB
pulling 966de95ca8a6: 100% 1.4 KB
pulling fcc5a6bec9da: 100% 7.7 KB
pulling a70ff7e570d9: 100% 6.0 KB
pulling 56bb8bd477a5: 100% 96 B
pulling 34bb5ab01051: 100% 561 B
verifying sha256 digest
writing manifest
success
>>> Send a message (/? for help)


marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 69×13
(ollama) mjrovai@raspi-5-SD:~ $ ollama run llama3.2:3b --verbose
>>> Qual a capital do Brasil?
A capital do Brasil é Brasília.
total duration: 5.140576457s
load duration: 185.275522ms
prompt eval count: 31 token(s)
prompt eval duration: 3.35620978s
prompt eval rate: 9.24 tokens/s
eval count: 9 token(s)
eval duration: 1.598257553s
eval rate: 5.63 tokens/s
>>> Send a message (/? for help)


Ollama File Edit Window Help
US
Open Ollama
Settings... ⌘ ,
Quit Ollama ⌘ Q
¿Cuál es la capital de Colombia?
La capital de Colombia es Bogotá. Establecida en el siglo XVI, Bogotá es una ciudad vibrante y cultural que alberga numerosos museos, restaurantes, tiendas y lugares turísticos. La ciudad también es conocida por su ubicación en las Llanuras de Bogotá, a una altura de aproximadamente 2.640 metros sobre el nivel del mar.
Bogotá es famosa por ser la sede de la Asamblea Legislativa Nacional y el Palacio de Justicia, un edificio icónico que alberga la Corte Constitucional de Colombia. La ciudad también cuenta con numerosos parques y jardines, como el Parque Simón Bolívar y el Jardín Botánico.
Además, Bogotá es conocida por su vida nocturna, música y gastronomía, que ofrecen una amplia variedad de opciones para los visitantes. La ciudad también cuenta con numerosas tradiciones y eventos culturales durante todo el año, como la Feria de las Flores y la Fiesta de San Pedro.
En resumen, Bogotá es una ciudad vibrante y cultural que ofrece una experiencia única a los visitantes. Su historia, arquitectura, música y gastronomía la convierten en un destino turístico imprescindible en Colombia.
Send a message
llama3.2:3b


responda en una linea
La capital de Colombia es Bogotá.
Y la de Peru?
La capital del Perú es Lima.
Cual la distancia de Bariloche a Bogotá?
Aproximadamente 3.100 km, según Google Maps.
Send a message
llama3.2:3b

MoonDream


Vision Encoder
(CLIP, 454M params,
1152→2048
embedding)
LLM
(Phi-2, 1.4B params,
2048 context)

MoonDream Vision-Language Pipeline

CLIP (454.45M) → Projection → Phi-2 (1.4B) | Q4_0 Quantized

1. Image Input

Raw pixel data

```
RGB Image: 378×378×3
```

2. CLIP Vision Encoder

Pre-trained frozen weights (454.45M params)

- Patch embedding (16×16 patches)
- Vision Transformer layers
- Output: [num_patches, 1152 dims]

3. Projection Layer

Trainable adapter (1152 → 2048 dims)

```
Linear projection:
[num_patches, 1152] → [num_patches, 2048]
```

Visual Tokens

```
[num_patches, 2048]
```

Text Tokens

```
[N, 2048]
```

Concatenate: [num_patches+N tokens, 2048 dims]

4. Phi-2 Transformer (1.4B params)

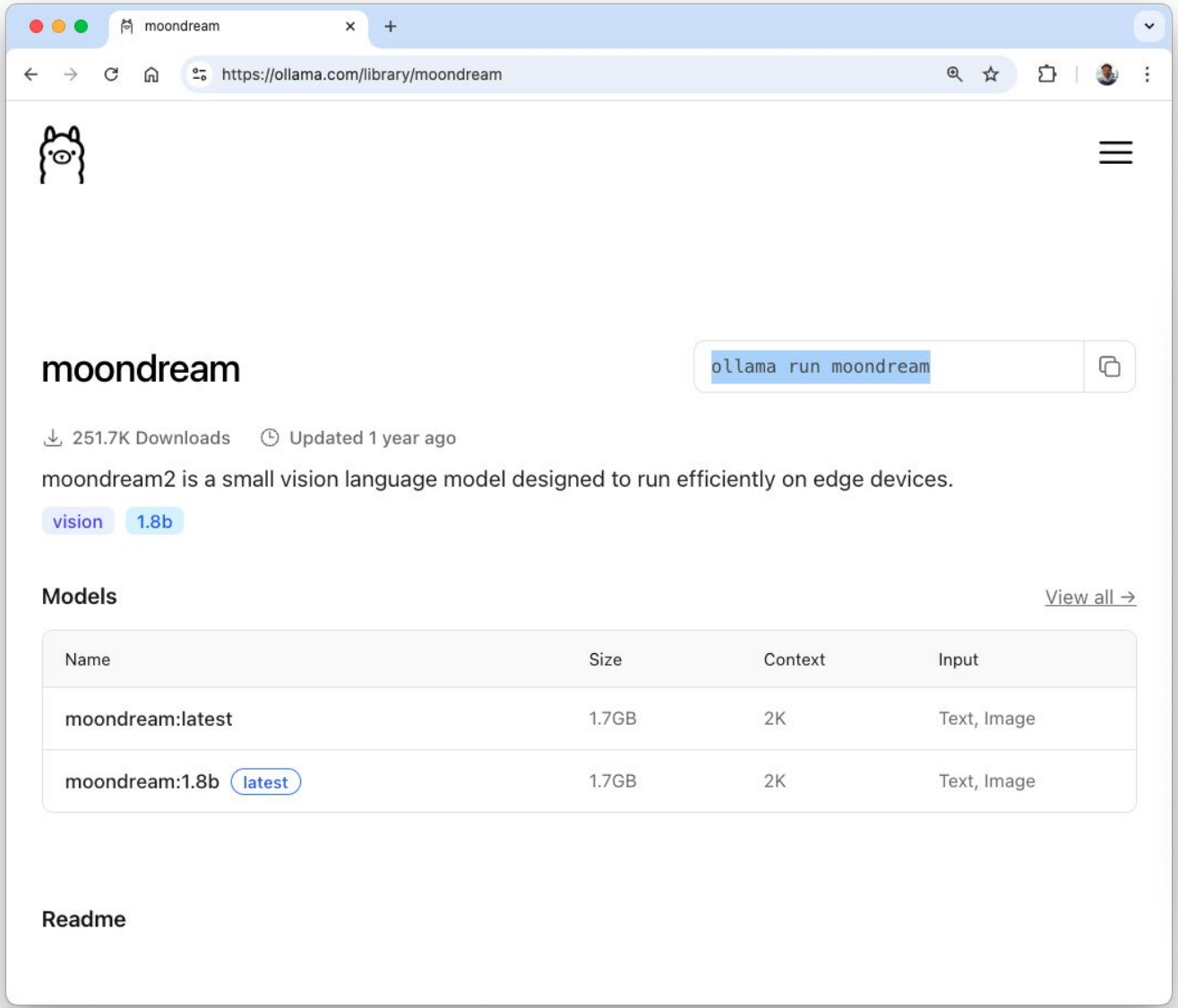
32 layers of self-attention

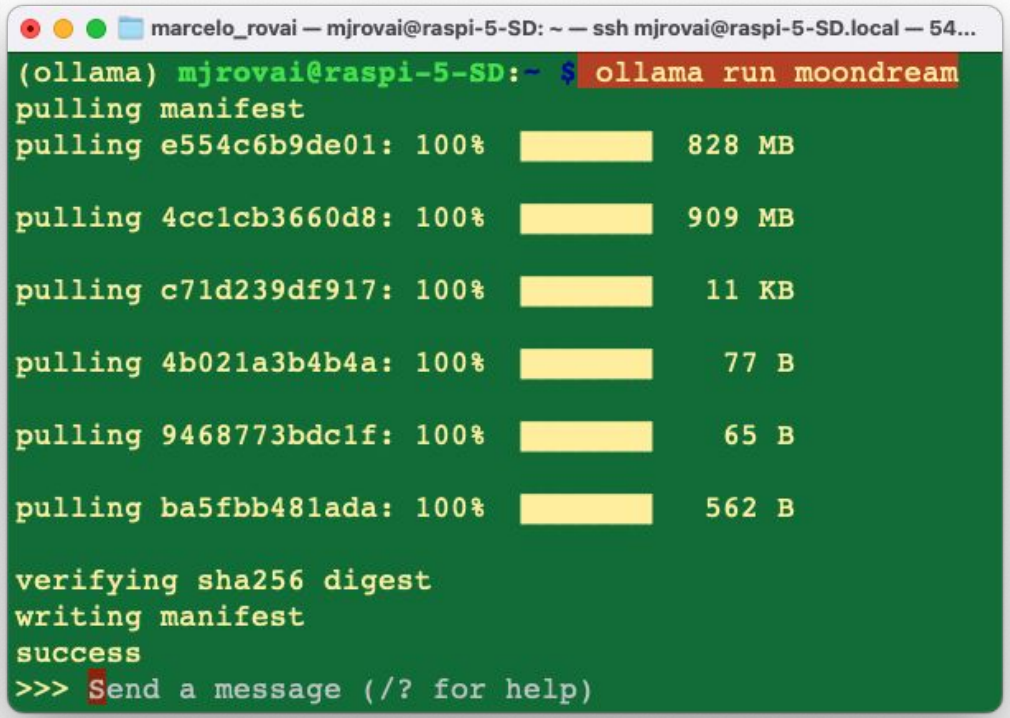
- Text attends to visual tokens
- Cross-modal reasoning
- Context: 2048 tokens
- Output: [num_patches+N, 2048]

5. Language Modeling Head

Generate answer tokens

```
Logits → Sampling → Tokens
"A cat sitting on a couch"
```


moondream
https://ollama.com/library/moondream
moondream
ollama run moondream
251.7K Downloads
Updated 1 year ago
moondream2 is a small vision language model designed to run efficiently on edge devices.
vision
1.8b
Models
View all →
Name
Size
Context
Input
moondream:latest
1.7GB
2K
Text, Image
moondream:1.8b
latest
1.7GB
2K
Text, Image
Readme

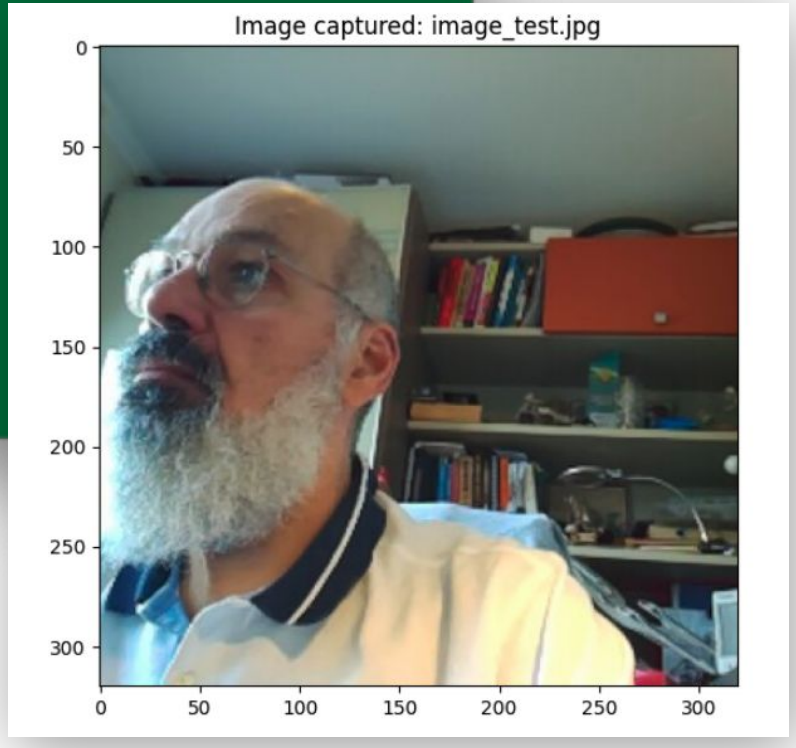

marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 54...
(ollama) mjrovai@raspi-5-SD:~ $ ollama run moondream
pulling manifest
pulling e554c6b9de01: 100% 828 MB
pulling 4cc1cb3660d8: 100% 909 MB
pulling c71d239df917: 100% 11 KB
pulling 4b021a3b4b4a: 100% 77 B
pulling 9468773bdc1f: 100% 65 B
pulling ba5fbb481ada: 100% 562 B
verifying sha256 digest
writing manifest
success
>>> Send a message (/? for help)

marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 87×18

```
(ollama) mjrovai@raspi-5-SD:~ $ ollama run moondream --verbose
>>> describe the image: /home/mjrovai/Documents/OLLAMA/image_test.jpg"
Added image '/home/mjrovai/Documents/OLLAMA/image_test.jpg'

The image features a man with glasses and a white beard, who is wearing a
collared shirt. He appears to be looking off into the distance or at something
out of frame. The room has several bookshelves filled with numerous books,
indicating that he might have an interest in reading or studying.

total duration:       54.36227837s
load duration:        63.257535ms
prompt eval count:    743 token(s)
prompt eval duration: 48.361595541s
prompt eval rate:     15.36 tokens/s
eval count:           62 token(s)
eval duration:        5.935988132s
eval rate:            10.44 tokens/s
>>> Send a message (/? for help)
```

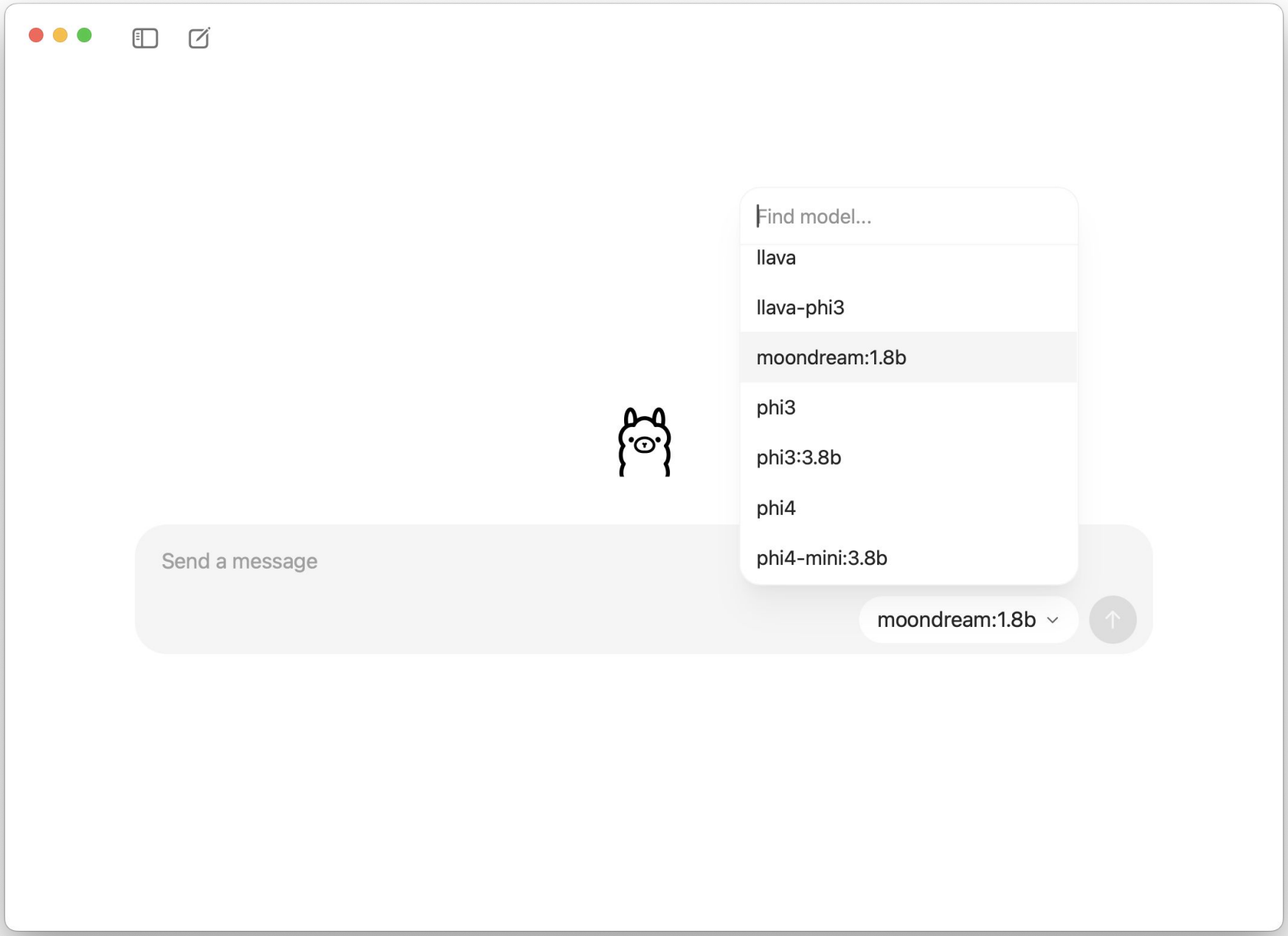


marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.loca...

```
>>> what is the color of the shirt?

The man's shirt is white.

total duration:       1m15.308297242s
```


Find model...
llava
llava-phi3
moondream:1.8b
phi3
phi3:3.8b
phi4
phi4-mini:3.8b
Send a message
moondream:1.8b

Describe the image

The image features a large group of people standing together, with some smiling and others waving. There are at least 13 individuals in the scene, all gathered closely to each other. They appear to be posing for a photo or enjoying an event outdoors. The group is positioned on a street, possibly near a park or a public area.

In addition to the people, there are several backpacks and handbags scattered around the scene, indicating that these individuals might have brought their belongings with them. A car can also be seen in the background, parked further away from the gathering of people.

What is the color of the shirt the guy with sunglasses is wearing?

The guy with sunglasses and a red jacket is wearing a blue shirt.

Send a message

moondream:1.8b

Command	Description	Example Usage
`ollama serve`	Starts the Ollama server for local API access and inference	`ollama serve`
`ollama run <model>`	Runs a specified model interactively or with input text	`ollama run smollm2:135m`
`ollama pull <model>`	Downloads a model from the Ollama registry	`ollama pull phi`
`ollama list`	Lists all models currently downloaded and available	`ollama list`
`ollama show <model>`	Displays detailed information about a specific model	`ollama show phi`
`ollama create <new_model> -f <file>`	Creates a new model from a Modelfile (customization)	`ollama create mymodel -f Modelfile`
`ollama rm <model>`	Removes a specified model from your system	`ollama rm phi`

Command	Description	Example Usage
`ollama ps`	Shows currently running Ollama processes or sessions	`ollama ps`
`ollama stop <model>`	Stops a running model session or process	`ollama stop phi`
`ollama cp <source> <target>`	Copies an existing model to a new name	`ollama cp phi mybackup`
`ollama push <model>`	Uploads a model to the Ollama registry (requires account & API key)	`ollama push mymodel`
`ollama --help`	Lists all available commands and their descriptions	`ollama --help`
`ollama --version`	Shows the current version of Ollama installed	`ollama --version`
`ollama <model> --verbose`	Run the model, showing statistics at the end	`ollama run phi --verbose`

marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 77×5

```
>>> /bye
(ollama) mjrovai@raspi-5-SD:~ $ ollama list
NAME                    ID              SIZE      MODIFIED
moondream:latest        55fc3abd3867    1.7 GB    10 minutes ago
llama3.2:3b             a80c4f17acd5    2.0 GB    21 minutes ago
```

marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 57×23

```
(ollama) mjrovai@raspi-5-SD:~ $ ollama show llama3.2:3b
  Model
    architecture        llama
    parameters          3.2B
    context length      131072
    embedding length    3072
    quantization        Q4_K_M

  Capabilities
    completion
    tools

  Parameters
    stop    "<|start_header_id|>"
    stop    "<|end_header_id|>"
    stop    "<|eot_id|>"

  License
    LLAMA 3.2 COMMUNITY LICENSE AGREEMENT

    Llama 3.2 Version Release Date: September 25, 2024

    ...
```

marcelo_rovai — mjrovai@raspi-5-SD: ~ — ssh mjrovai@raspi-5-SD.local — 54...

```
(ollama) mjrovai@raspi-5-SD:~ $ ollama show moondream
  Model
    architecture        phi2
    parameters          1.4B
    context length      2048
    embedding length    2048
    quantization        Q4_0

  Capabilities
    completion
    vision

  Projector
    architecture        clip
    parameters          454.45M
    embedding length    1152
    dimensions          2048

  Parameters
    stop            "<|endoftext|>"
    stop            "Question:"
    temperature     0

  License
    Apache License
    Version 2.0, January 2004
    ...
```

Questions?

Prof. Marcelo J. Rovai

rovai@unifei.edu.br

UNIFEI