2024

Machine Learning running smooth

Aleksei (Alex) Kharinskii

Bio

Alex Kharinskii

- Linux engineer since 2010
- Cloud engineer since 2019
- Launched a public cloud platform in 2020
- Solopreneur since 2022

linkedin.com/in/kharinskiy/

Launched Mealtune

ML-driven nutrition adviser

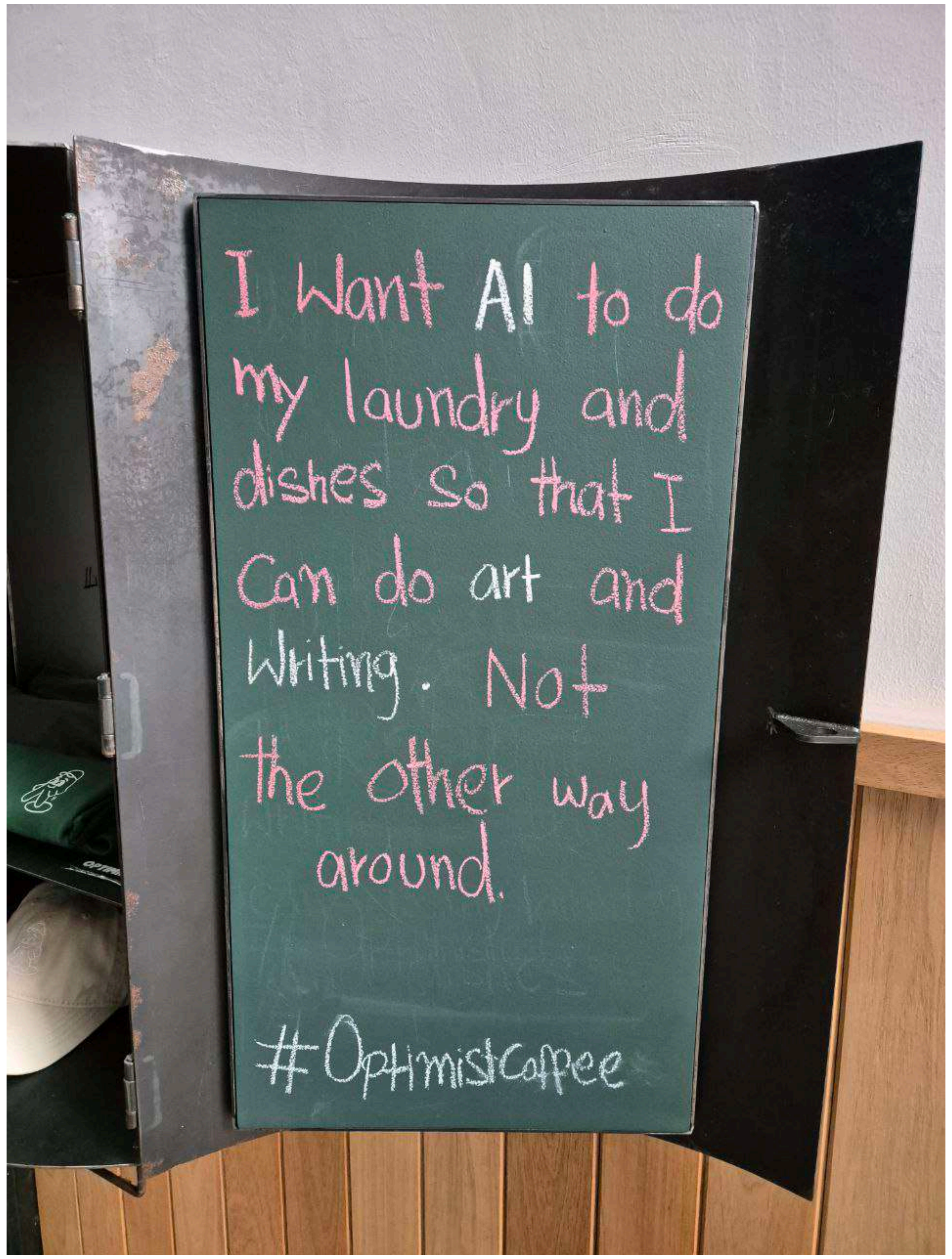
Why to do it by hand

1 Total control

Would you allow ChatGPT to drive your car or rock a kid?

Control the flow of your own data.

Choose a tool for a task, not the other way around.


I Want AI to do my laundry and dishes so that I Can do art and Writing. Not the other way around.
#OptimistCoffee

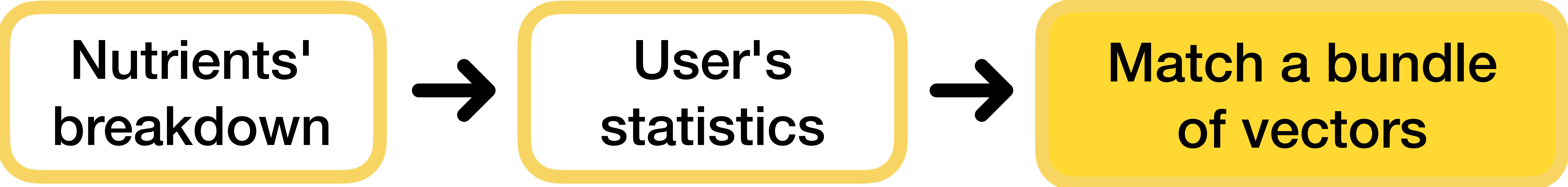
OptimistCoffee, KL

Why to do it by hand

Don’t follow rules, not a bad idea at all

2 Reduce a code complexity

Before

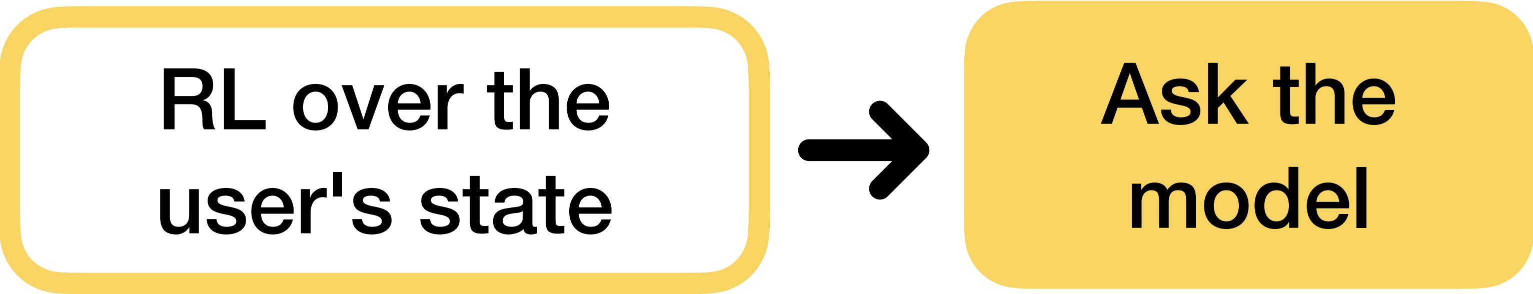


Nutrients' breakdown → User's statistics → Match a bundle of vectors

Fuzzy search → Vector’s embeddings

Regex / unstructured data parsing → Text labeling

After



RL over the user's state → Ask the model

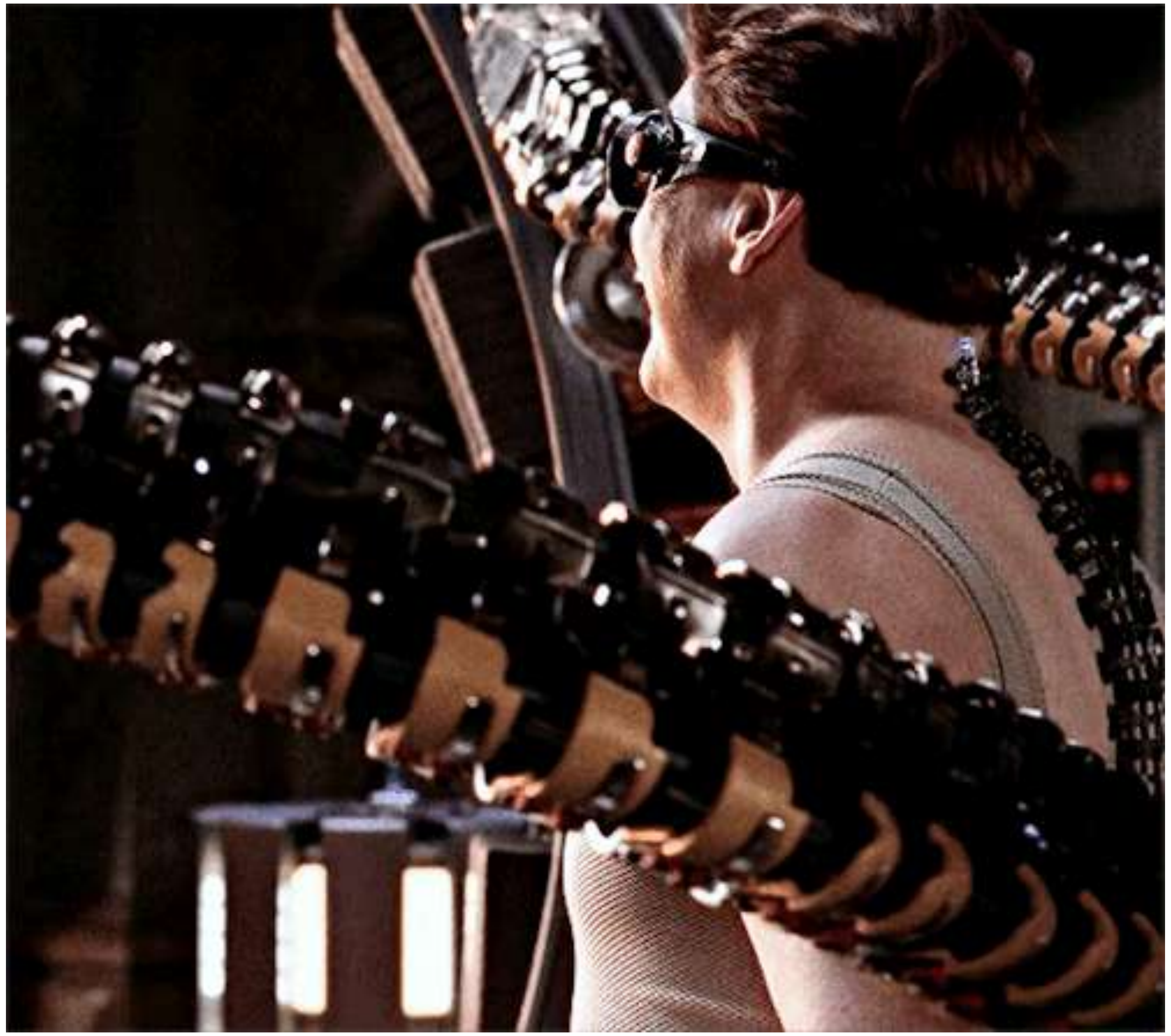
Reinforcement learning → find the best decision within the environment's state

Why to do it by hand

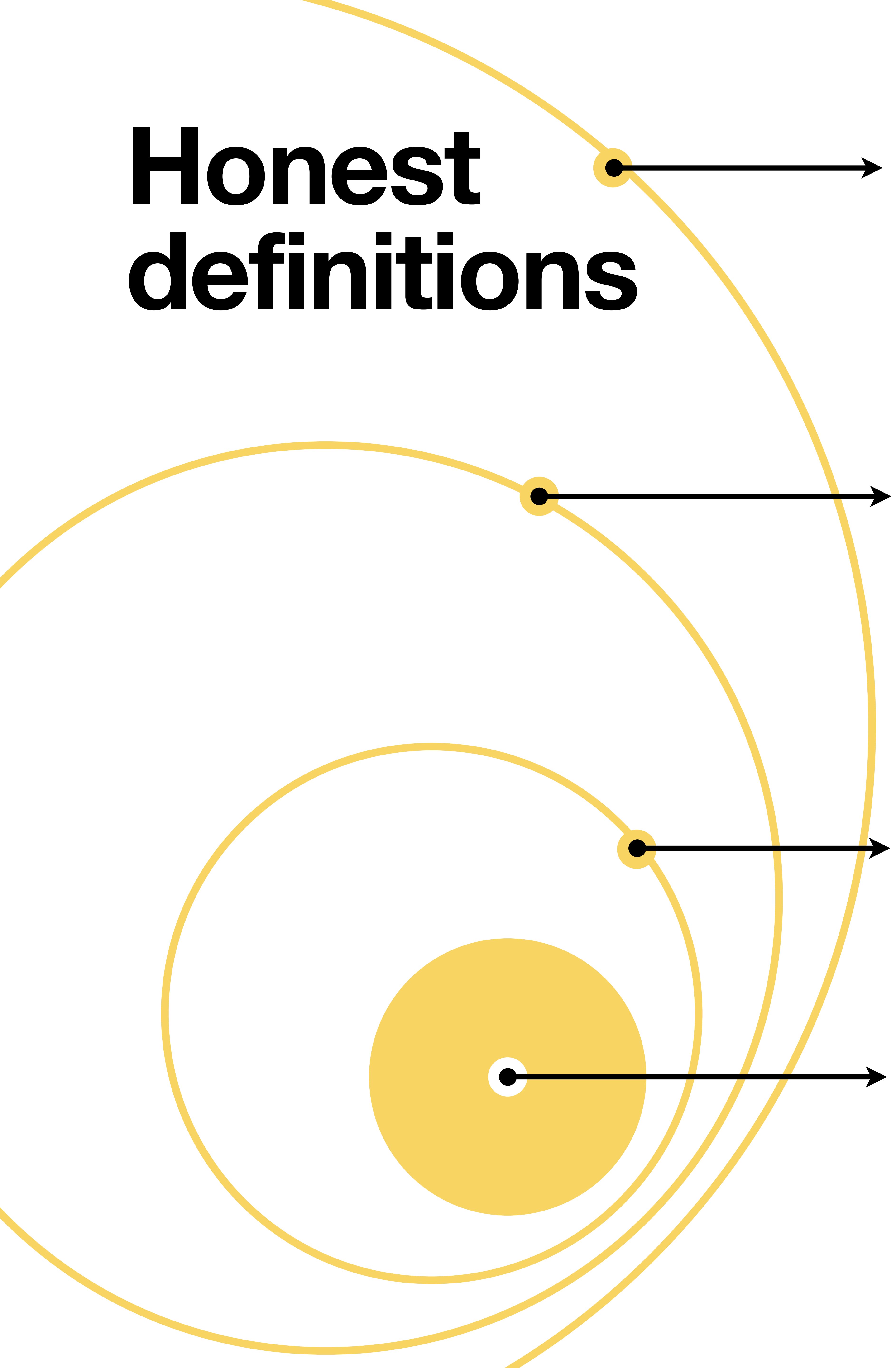
3 For fun!

Understand how to launch it anywhere.

Cool robotic stuff for cosplay or social good!



Honest definitions



Machine Learning:

- Subfield of the AI field of research (Wikipedia).
- Could be "Trained". Works on the "Inference" phase.
- Using data to imitate the way that humans learn, improving its accuracy.

Deep Learning:

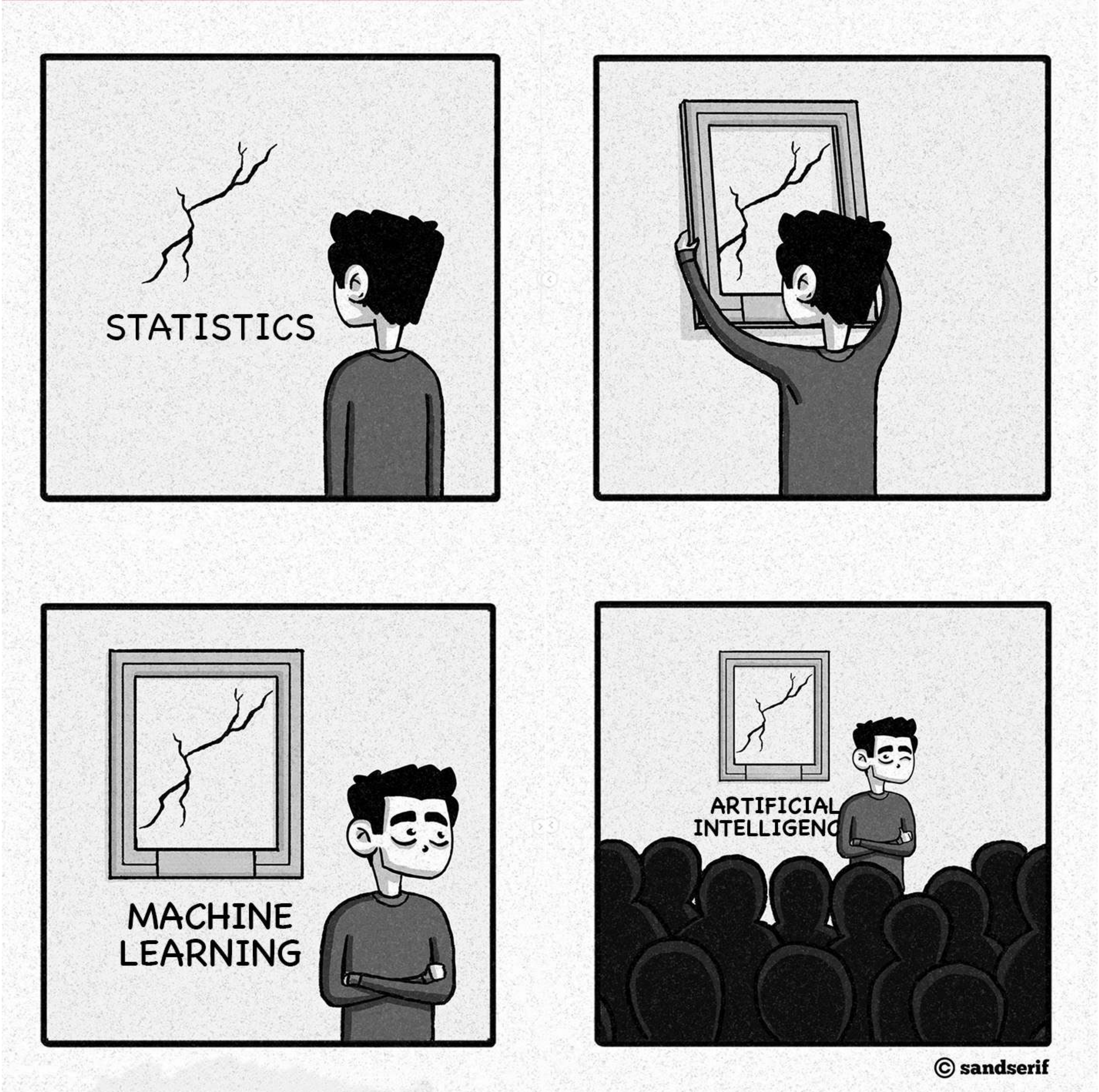
- Subfield of machine learning methods based on neural networks.
- Neural networks is about **tensor's operation.**
- Tensor — multi dimensional array
- Deep — means multiple layers inside the model.

Generative models:

- Part of statistical models.
- Could generate new data based on probability.

LLM (Large Language Models):

- Could understand natural language and process it: summarize, for example.
- Works not only with languages but other sequences.
- First really “valuable” public model was BERT (October 2018).

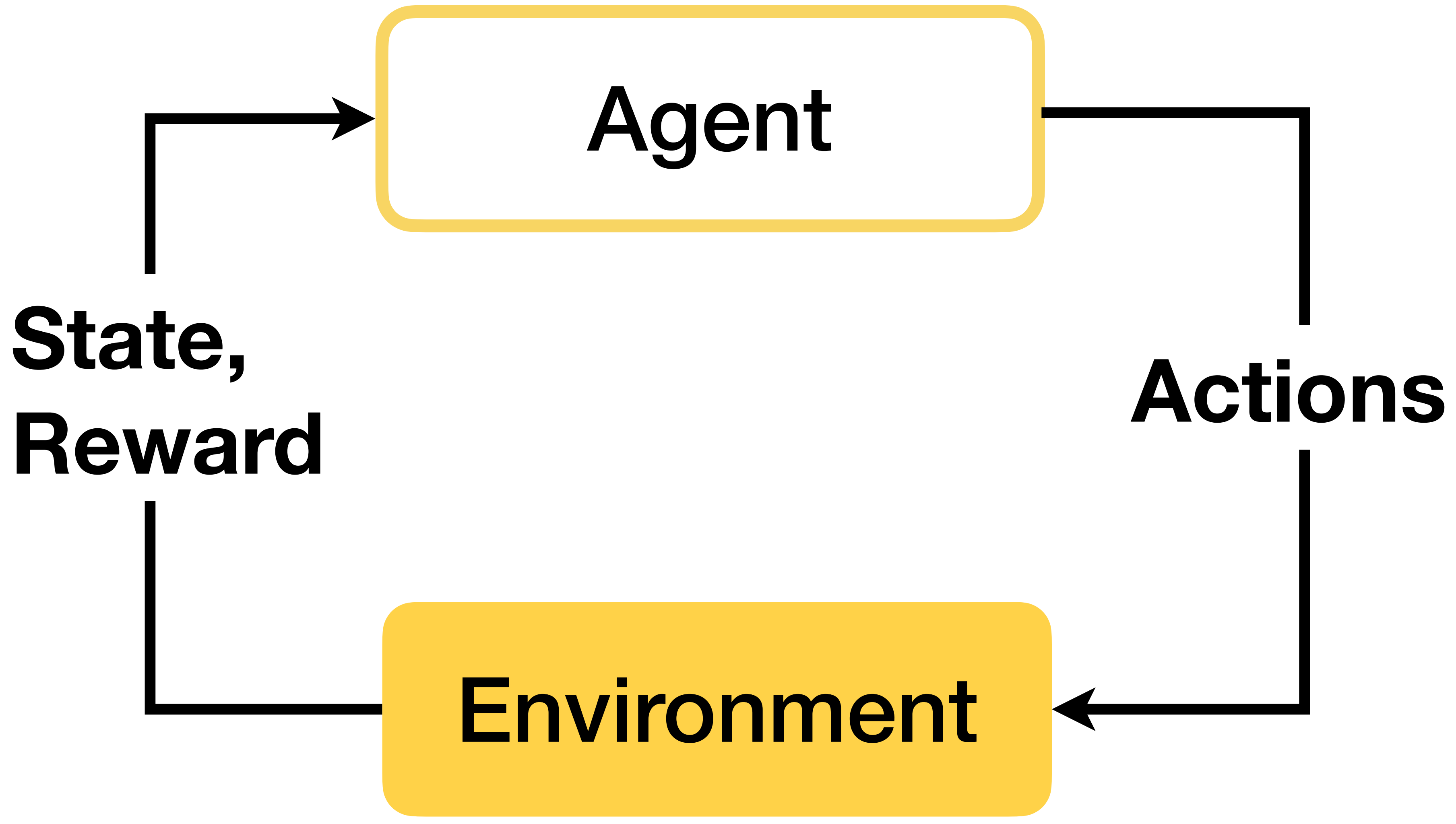


sandserifcomics

What are we going to discuss?

- Reinforcement learning
- Transformers
- Stable Diffusion
- How to run all of these

(Deep) Reinforcement Learning



The general idea

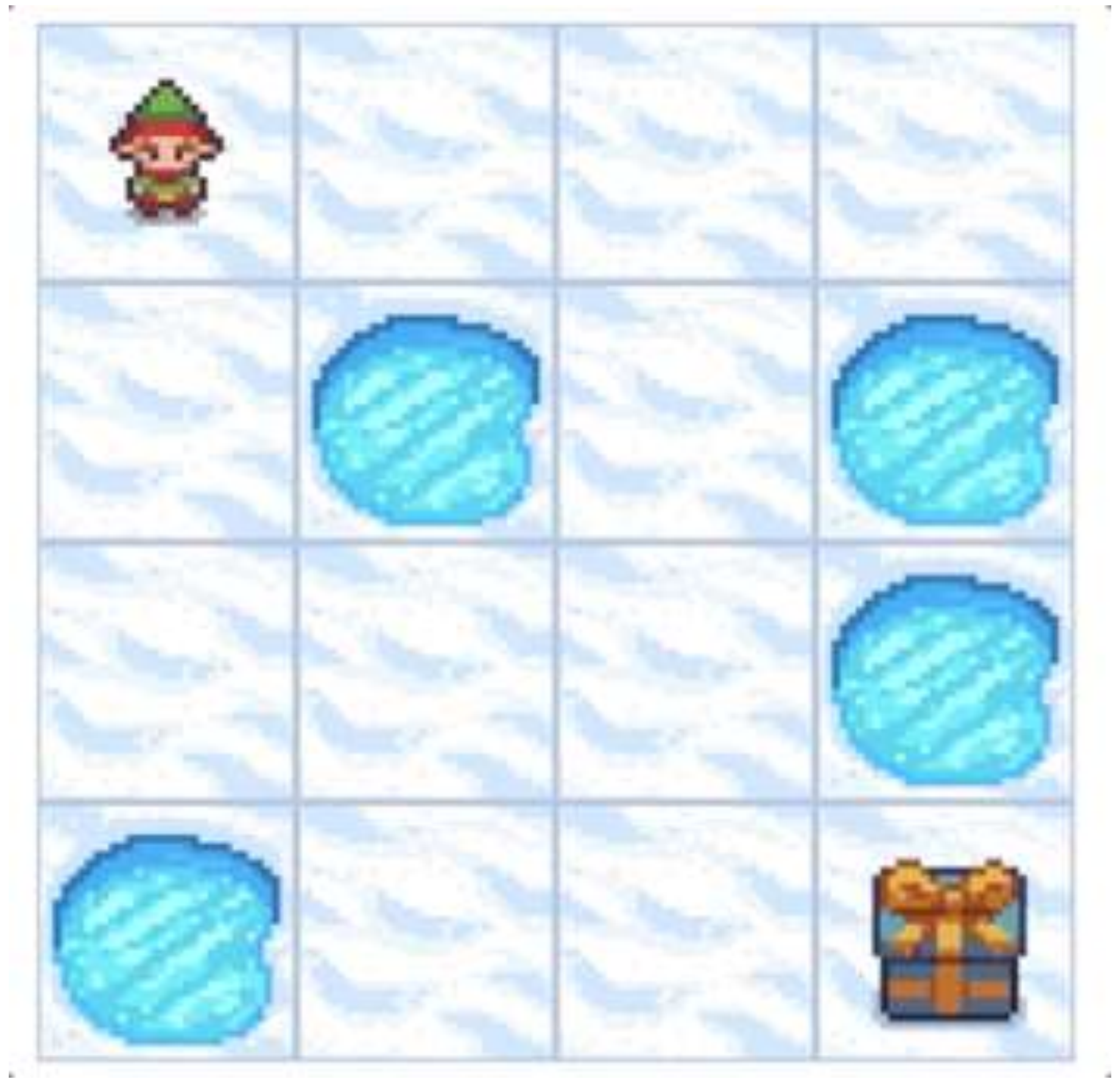
- Make an Action from the set (A) based on the State (S) of the Environment (E) — evaluate the reward. **Approximate a function that describes the environment within a certain level of probability.**
- Keep a reasonable balance between exploiting (pick the reward) and exploring (discover the environment).
- The reward could be discounted over time to motivate the model to "think" quickly.

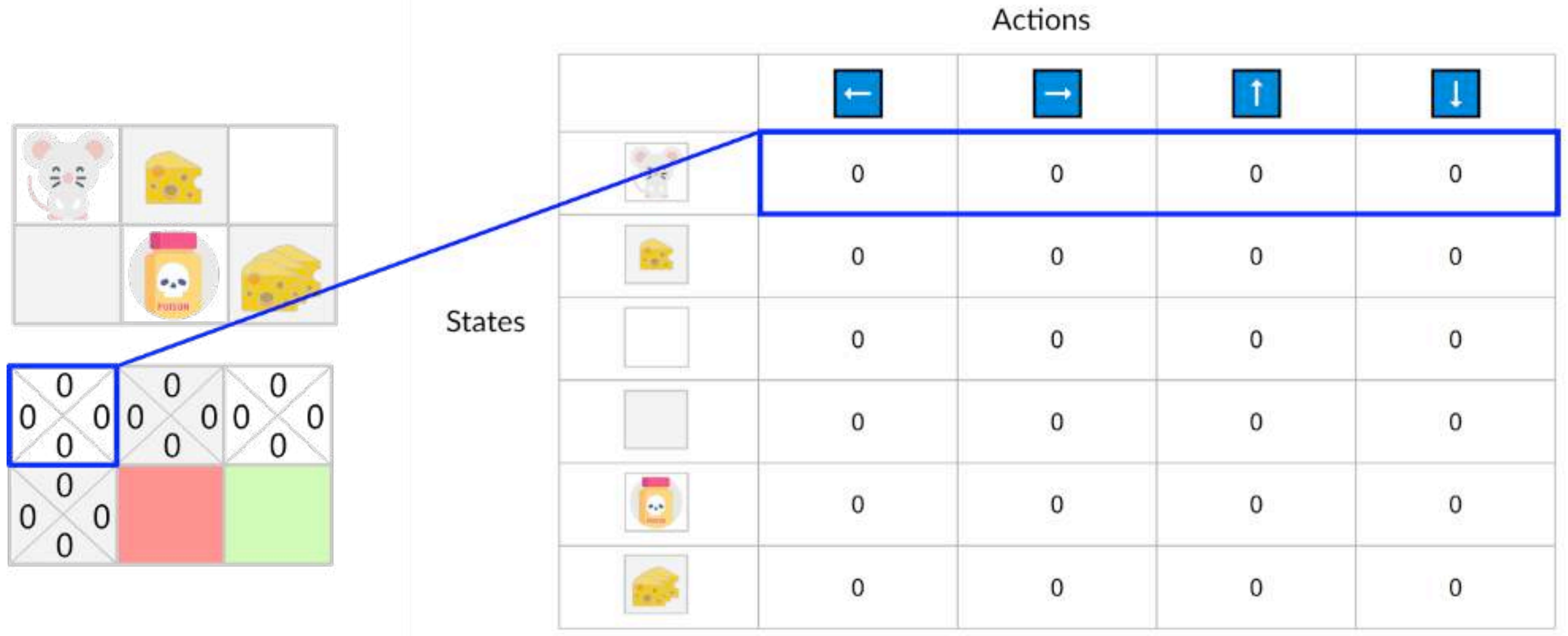
Examples:

- automotive robots
- playing video games
- recommendation systems

Simple RL model

- Simple RL based on the table — is something close to the backtracking algorithm.
- Greedy policy — take the best action to achieve the highest reward.
- Explore the environment at random, no matter what rewards are around.
- Algorithm relies on the Q-Table.



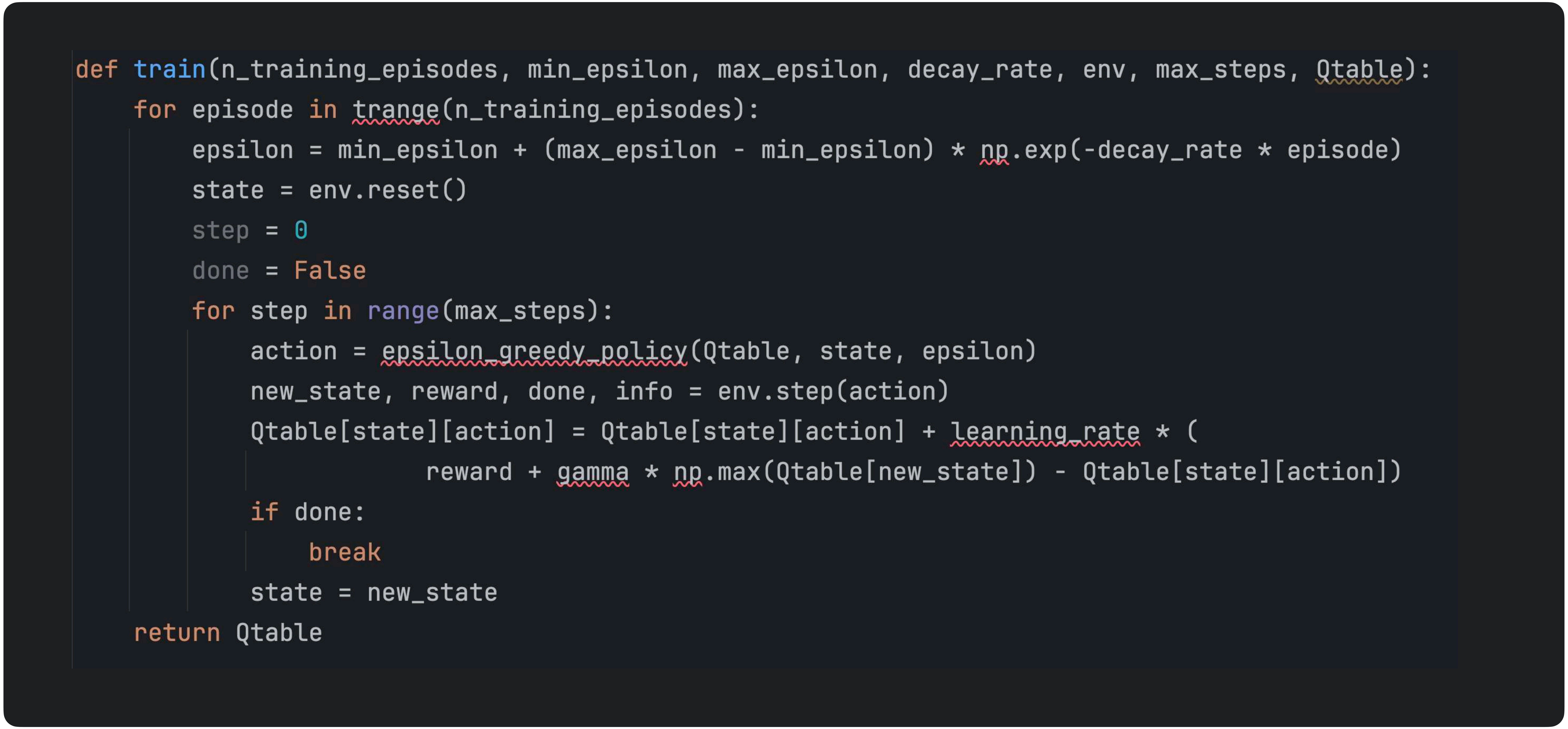


Actions

States	←	→	↑	↓
🐭	0	0	0	0
🧀	0	0	0	0
	0	0	0	0
	0	0	0	0
☠	0	0	0	0
🧀	0	0	0	0

Taken from the Hugging Face website

Simple RL model



```
def train(n_training_episodes, min_epsilon, max_epsilon, decay_rate, env, max_steps, Qtable):
    for episode in trange(n_training_episodes):
        epsilon = min_epsilon + (max_epsilon - min_epsilon) * np.exp(-decay_rate * episode)
        state = env.reset()
        step = 0
        done = False
        for step in range(max_steps):
            action = epsilon_greedy_policy(Qtable, state, epsilon)
            new_state, reward, done, info = env.step(action)
            Qtable[state][action] = Qtable[state][action] + learning_rate * (
                        reward + gamma * np.max(Qtable[new_state]) - Qtable[state][action])
            if done:
                break
            state = new_state
    return Qtable
```

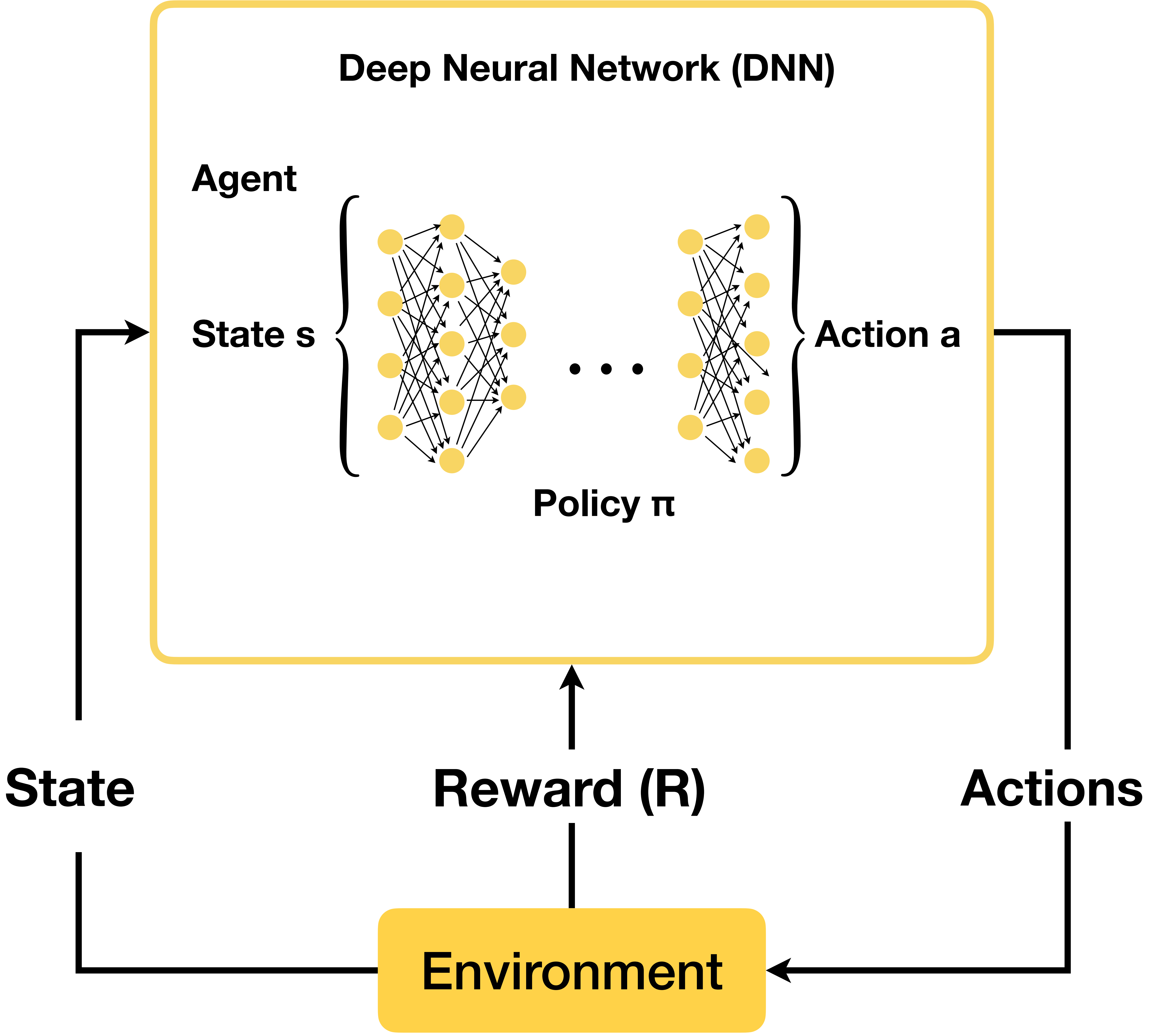
Simple RL model

Main characteristics

- No datasets needed. Learn from the environment.
- Simple cases could be learned and launched on a calc. Just need a memory to maintain the Q-Table and the logic computation device (CPU).
- The memory requirements grow within state/action's numbers.

Deep RL model

- Deep Neural Network as an agent.
- The agent adjusts its weights based on the reward it gets from the environment.
- The agent approximates a function that describes the environment.



Deep RL model

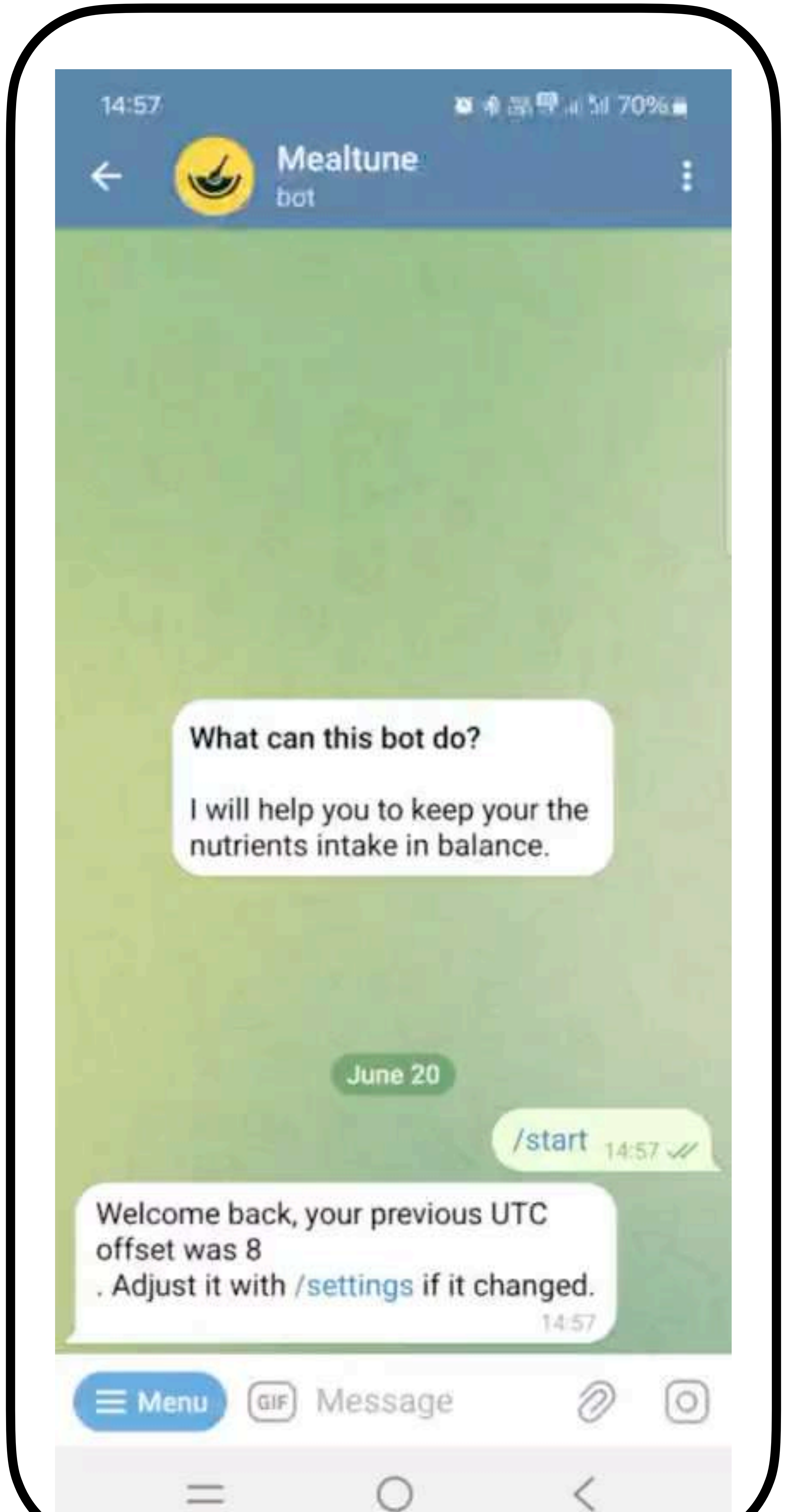
Main characteristics

- No datasets needed. Learn from the environment.
- Relatively "small" models, perform nice on CPU.
- Deep NN as an agent == **matrix multiplications**. It's expensive.
- Can be used to “train” an LLM. TRL - Transformer Reinforcement Learning.
- Approximates a complex function with many different environmental states.

Deep RL model

Mealtune uses one to suggest the best meal option that matches the current user state.

Thanks to the "exploration" modifier it suggests variety of options.



What can this bot do?

I will help you to keep your the nutrients intake in balance.

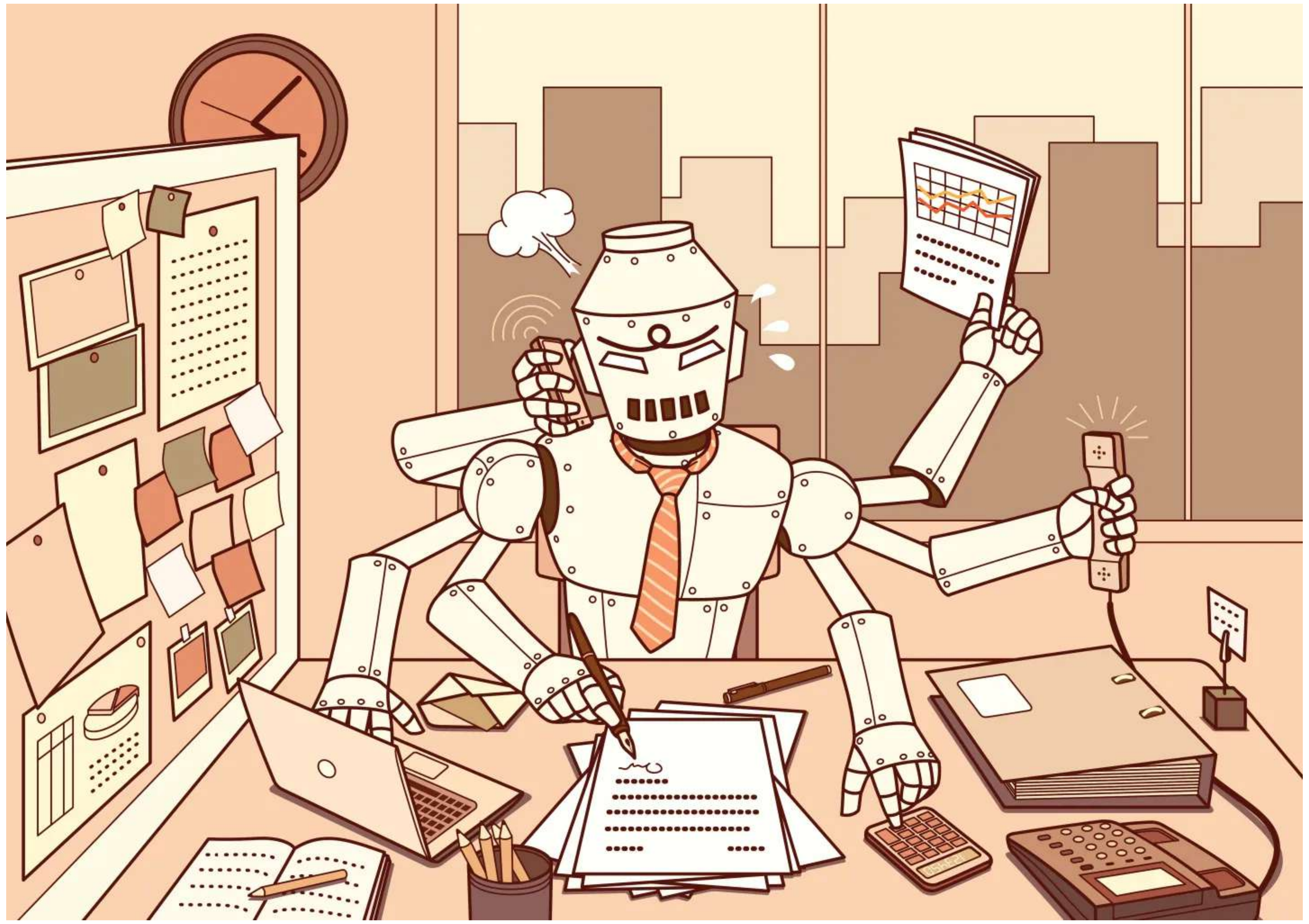
/start 14:57

Welcome back, your previous UTC offset was 8
. Adjust it with /settings if it changed.
14:57

Transformers

The general idea

- It is the **statistical model** that tries to predict the next token from the vocabulary.
- Uses the **Attention mechanism** to keep valuable parts of a sequence.
- The idea scales perfectly on related areas: Visual Transformers (ViT) — classifying images, splitting it by patches and leveraging the “attention” mechanism.
- Tasks: sequence’s generation, classification, summarization.

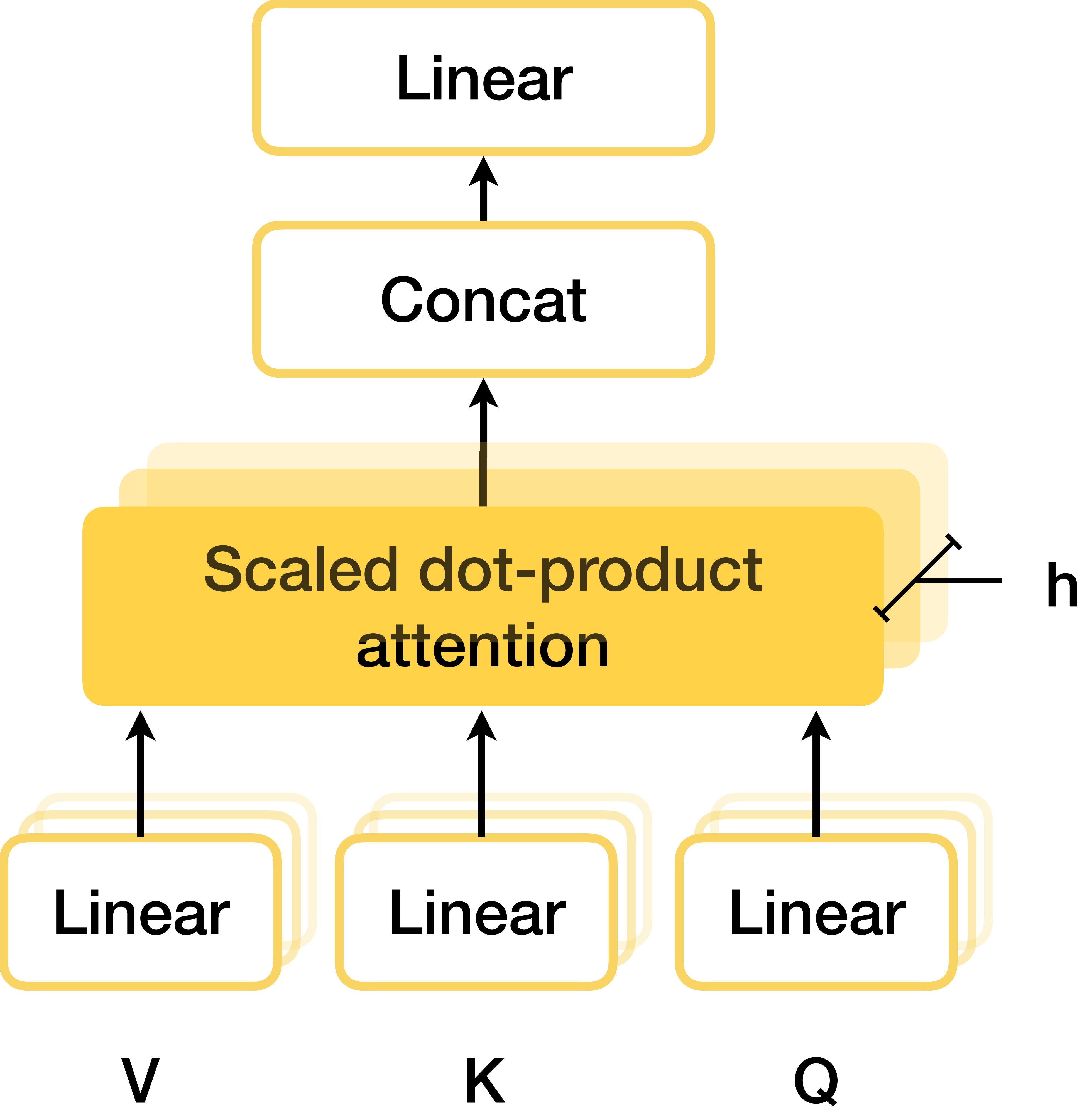


VICHANCHAIRAT — GETTY IMAGES

Transformers

Architecture and working steps

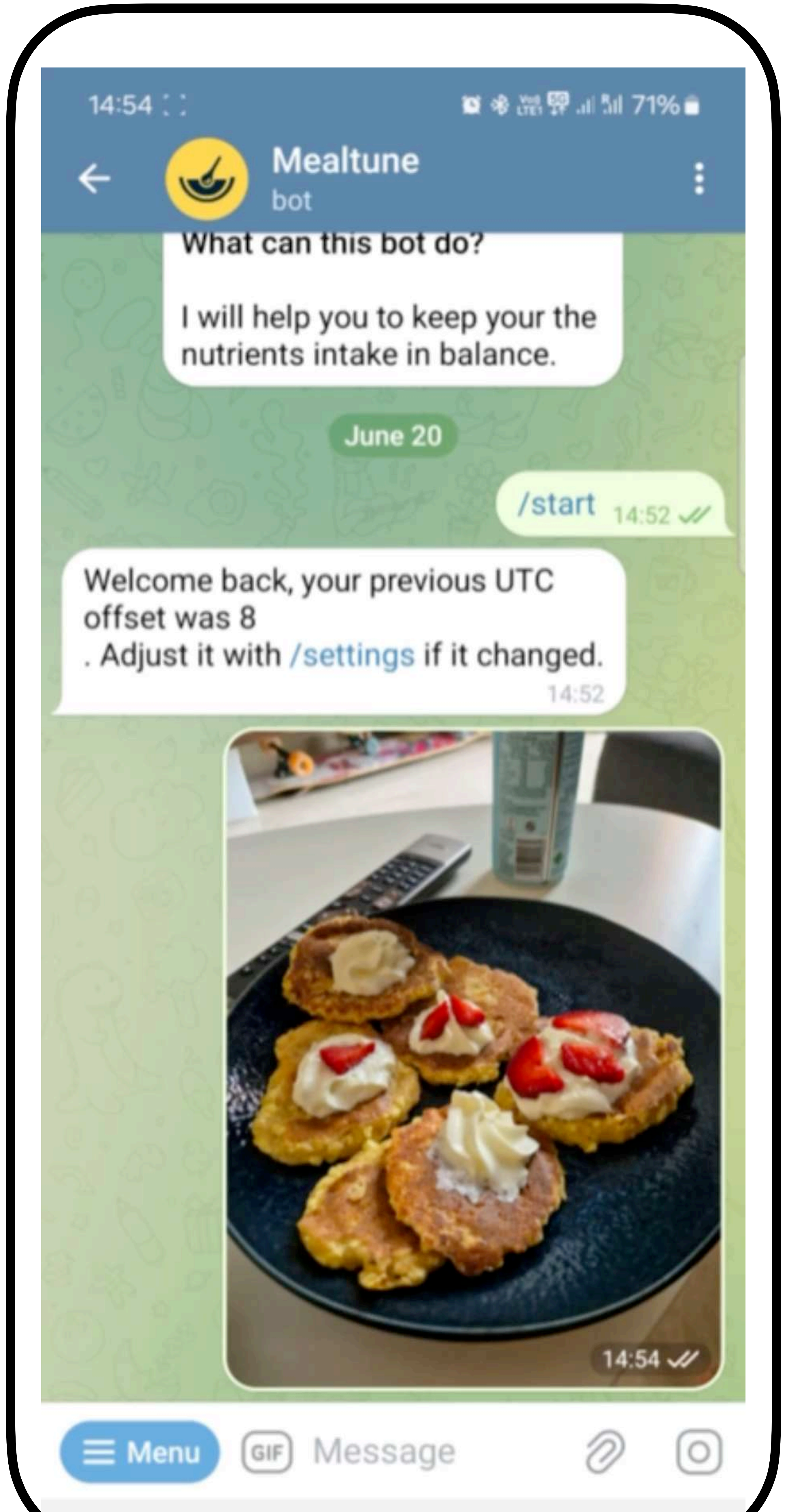
- Attention allows the model to focus on meaningful data. It relies on the Query-Key-Value mechanism (QKV).
- QKV — **the dot product** between tensors to get the "attention weights" for the part of the sequence.
- QKV could be cached by the model.
- A tokenizer builds a word embedding based on its "attention weights".



Multi-Head attention
"Attention Is All You Need" paper (2017)

Transformers’ Workload

Mealtune uses ViT model for the image classification process



What can this bot do?

I will help you to keep your the nutrients intake in balance.

/start 14:52

Welcome back, your previous UTC offset was 8
. Adjust it with /settings if it changed.
14:52

Transformers’ Workload

Main characteristics

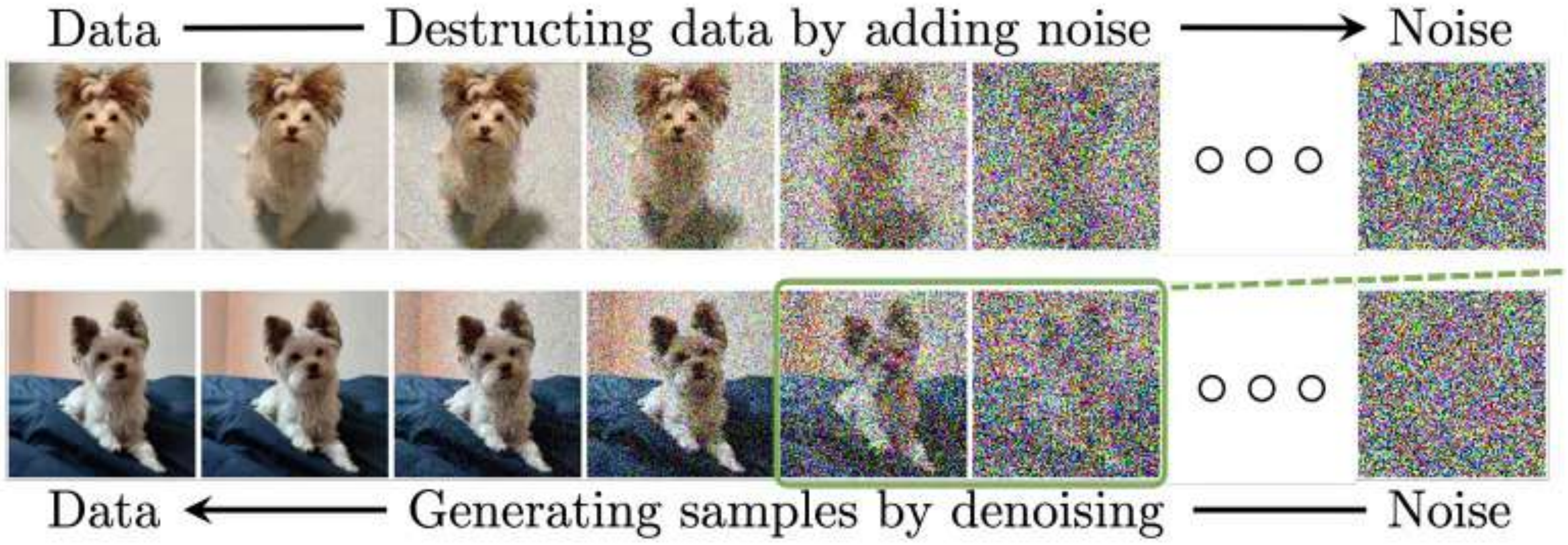
- The main idea — the attention mechanism — transfers greatly to different tasks and languages.
- Huge memory requirements: weights, biases, optimizer’s parameters — should be loaded into memory.
- Matrix multiplication is a heavy operation best fitted for GPU.

Diffusion models’ Workload

The general idea

- Learn to noise and denoise a picture to “grasp” the initial pixel map and the result described by prompt.

- Gradually "enhance" the initial picture from the noised one.


Data
Destructing data by adding noise
Noise
Data
Generating samples by denoising
Noise

Diffusion Models: A Comprehensive Survey of Methods and Applications, 2022

Sounds familiar?

Zoom and Enhance!
(CSI)

Summary

Neural networks are universal function approximators (#PyTorch)

Adjust weights in memory to approximate a function that describes a process.

Key notes

- **Matrix multiplications** during the learning process — adjusting weights.
- **Matrix multiplications** during the inference process — depending on the NN's architecture.
- Matrix manipulations — one of the most expensive computational tasks.
- Parameters — weights and biases, should be loaded into memory.
- Some tasks required to work within complex tensor-like data: images or video.

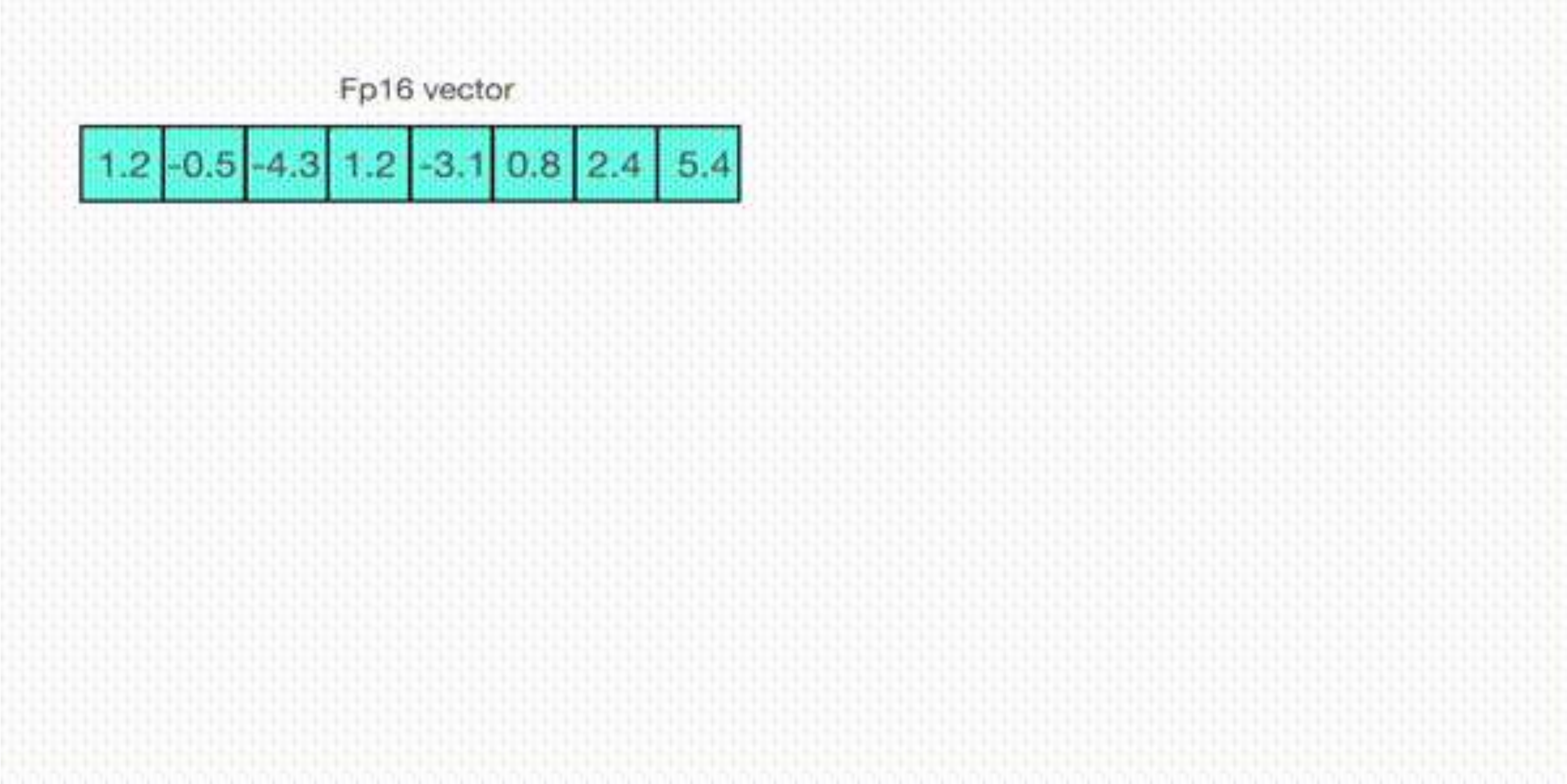
How to manage it?

Memory Consumption Optimization

Quantization

Parameters keeping as **floating number,** let's reduce its precision to safe memory.

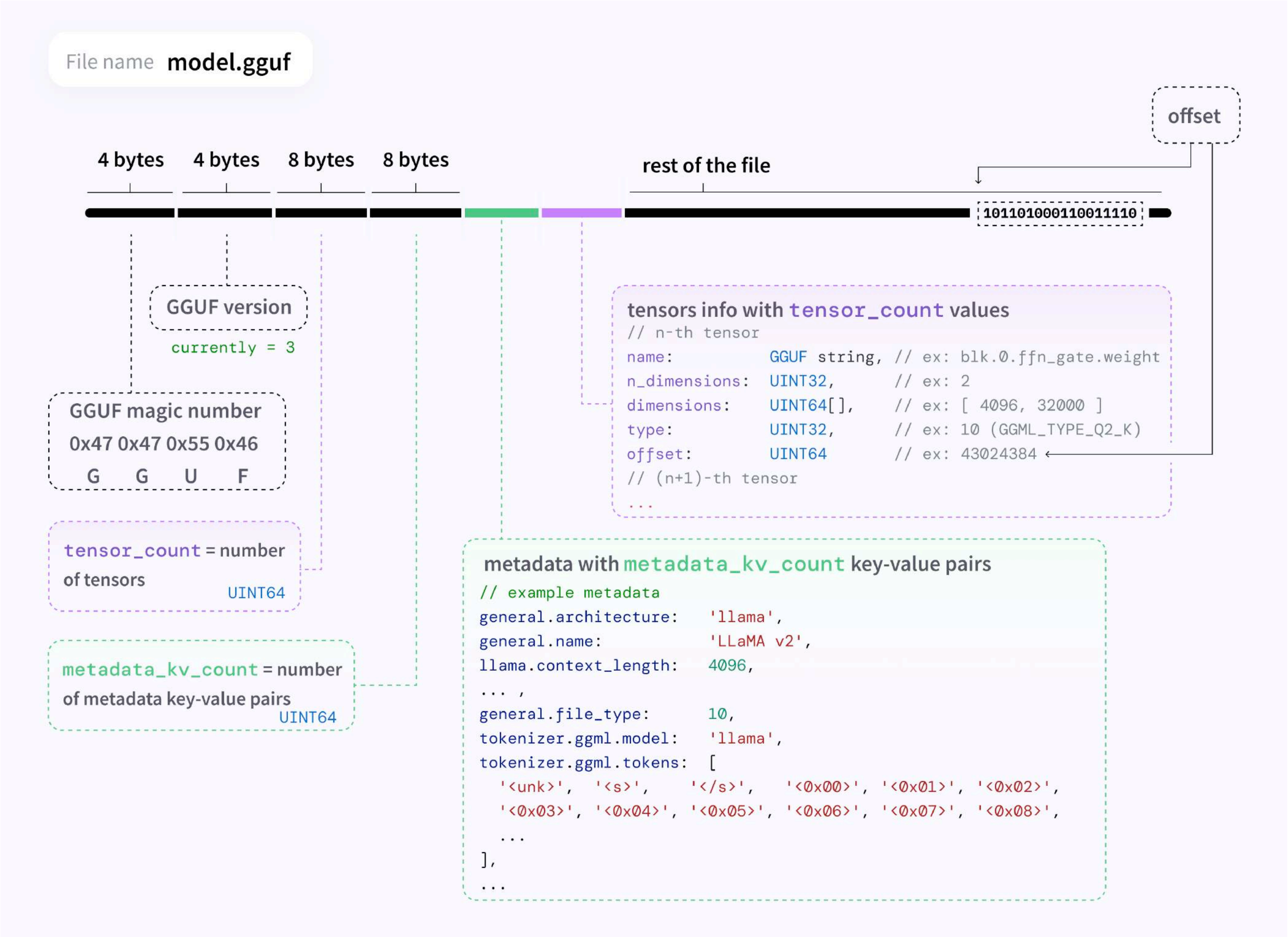
- Hugging Face store many models with different types of quantization.
- Check the model's card for the best quality.
- Evaluate the quantified model before the production use. **Keep in mind the quality.**
- Some ARM devices do not support floating arithmetics. Int-based quantization allows you to launch models on it.



Gif from the Hugging Face blog

LLAMA.CPP/GGUF

- Georgi Gerganov — talented bulgarian engineer presented file formats: GGUF (former GGML).
- LLAMA.CPP — a framework that binds GGUF-formatted models with some APIs.
- Projects like Ollama, GPT4All, Whisper.cpp, StableDiffusion.cpp — all relied heavily on the GGUF format.



tensors info with tensor_count values

```
// n-th tensor
name:          GGUF string, // ex: blk.0.ffn_gate.weight
n_dimensions:  UINT32,      // ex: 2
dimensions:    UINT64[],    // ex: [ 4096, 32000 ]
type:          UINT32,      // ex: 10 (GGML_TYPE_Q2_K)
offset:        UINT64       // ex: 43024384
// (n+1)-th tensor
...
```

metadata with metadata_kv_count key-value pairs

```
// example metadata
general.architecture:   'llama',
general.name:           'LLaMA v2',
llama.context_length:   4096,
... ,
general.file_type:      10,
tokenizer.ggml.model:   'llama',
tokenizer.ggml.tokens:  [
  '<unk>',  '<s>',    '</s>',   '<0x00>', '<0x01>', '<0x02>',
  '<0x03>', '<0x04>', '<0x05>', '<0x06>', '<0x07>', '<0x08>',
  ...
],
...
```

"GGUF file format" from the Hugging Face's blog post

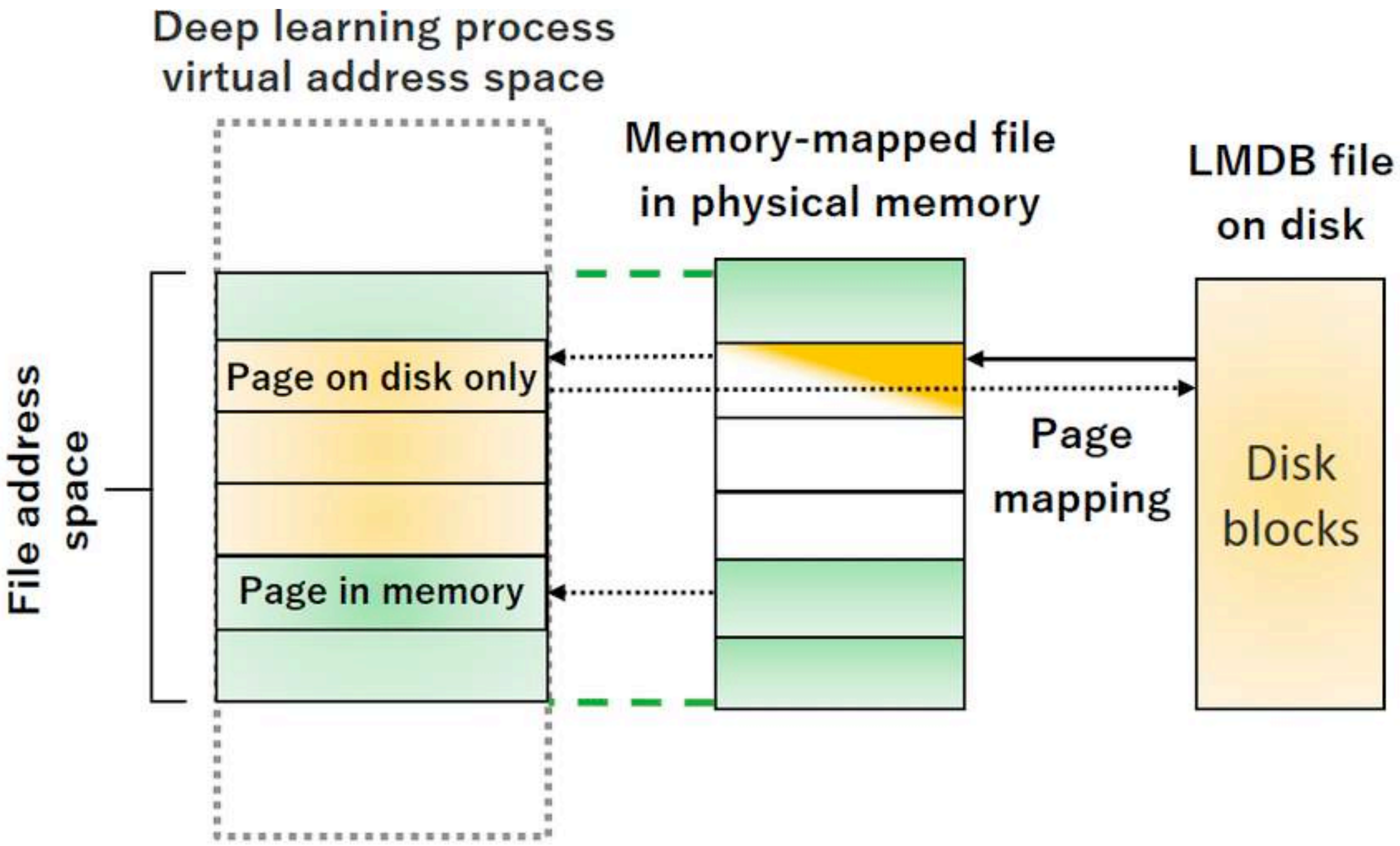
LLAMA.CPP/GGUF

Why is LLAMA.CPP a big deal?

mmap() file right into memory.

KV-cache quantization features.
Save even more memory.

Read the article by Justine Tunney — one of the most famous women hackers.



Design of a data supply mechanism for distributed deep learning

Amir Haderbache, 2017

LLAMA.CPP/GGUF

Why is LLAMA.CPP a big deal?

- Written in C++. The author is keeping it as lean as possible.
- Torch models could be converted to GGUF.
- Many easy-to-read examples for different problems: parallel execution, fine-tuning (LORA), even a Dockerfile for a smooth launch.

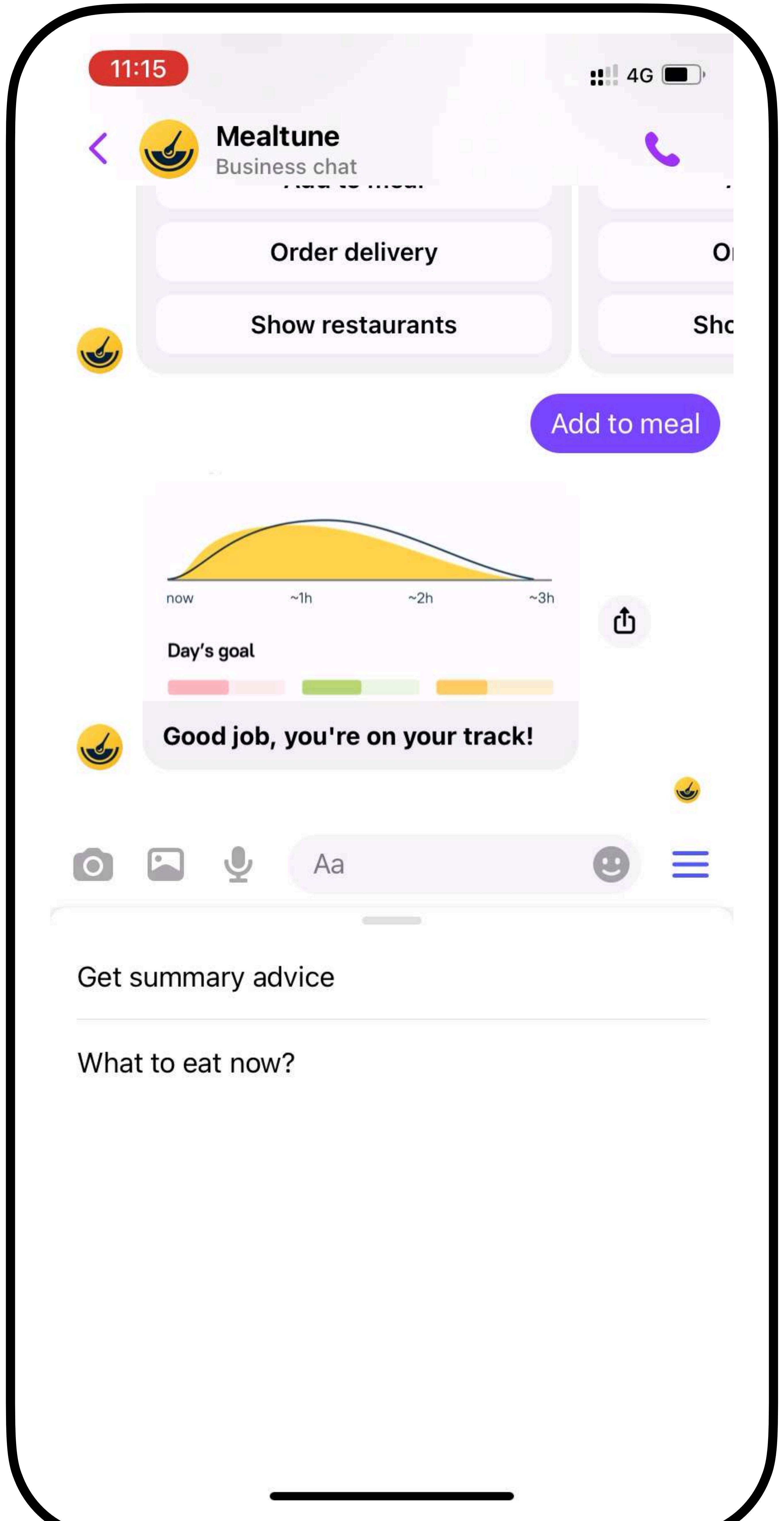
⚠

Don’t put it in production without a container!

Critical security issues, code execution.

LLAMA.CPP/GGUF

Mealtune uses LLAMA with llama.cpp for a daily advice generation.


11:15

4G

Mealtune

Business chat

Order delivery

Show restaurants

Add to meal

Day's goal

Good job, you're on your track!

Aa

Get summary advice

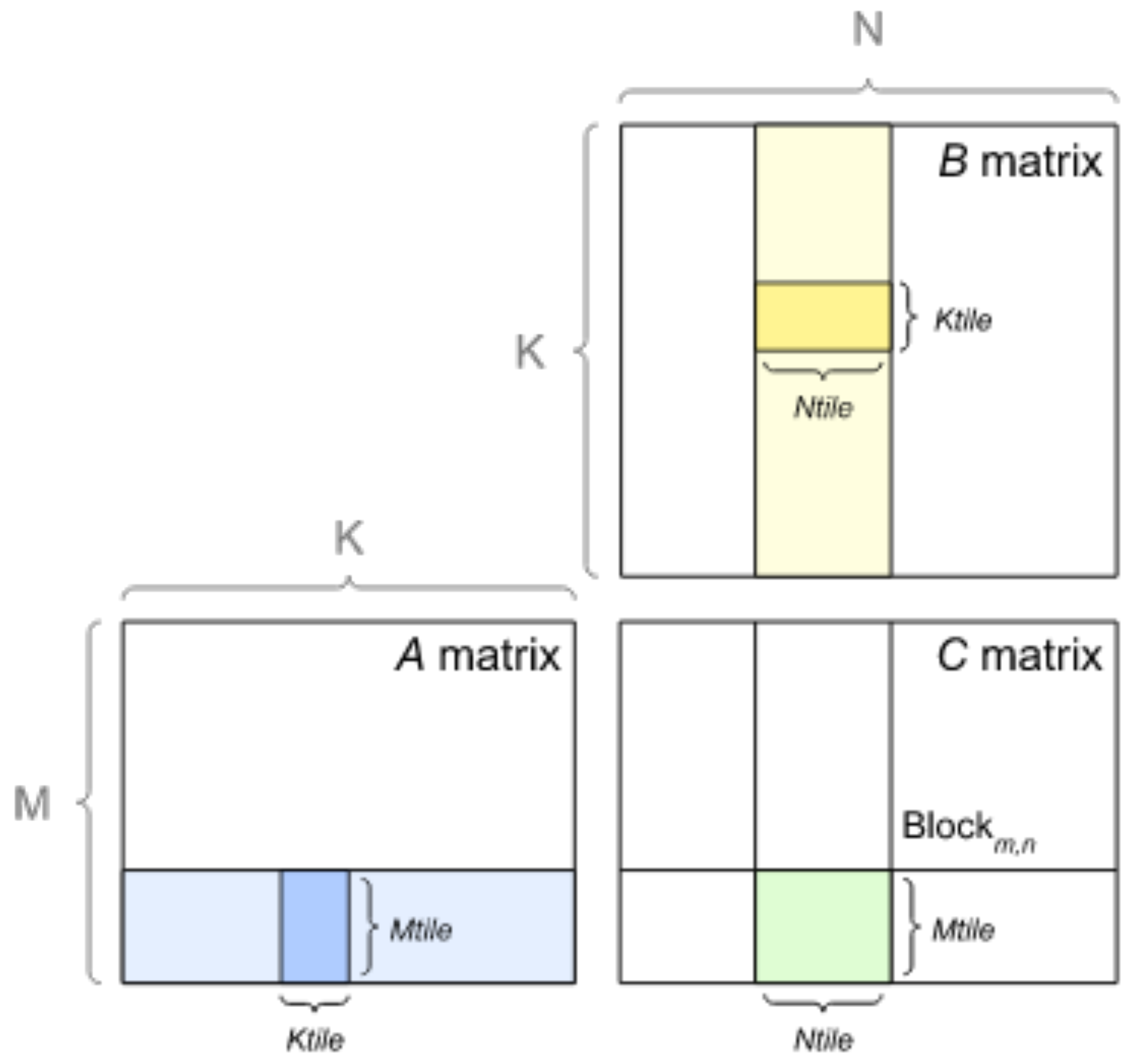
What to eat now?

Hardware layer: GPU

Nvidia GPUs — standard de-facto.

You should not have any problems with it.

GPUs allow partitioning the matrix computation by many thread blocks — architectural units of the device.



“Matrix multiplication background user’s guide”
NVIDIA developers blog

Hardware layer: GPU

Compute Unified Device Architecture (CUDA) is a proprietary parallel computing platform and application programming.

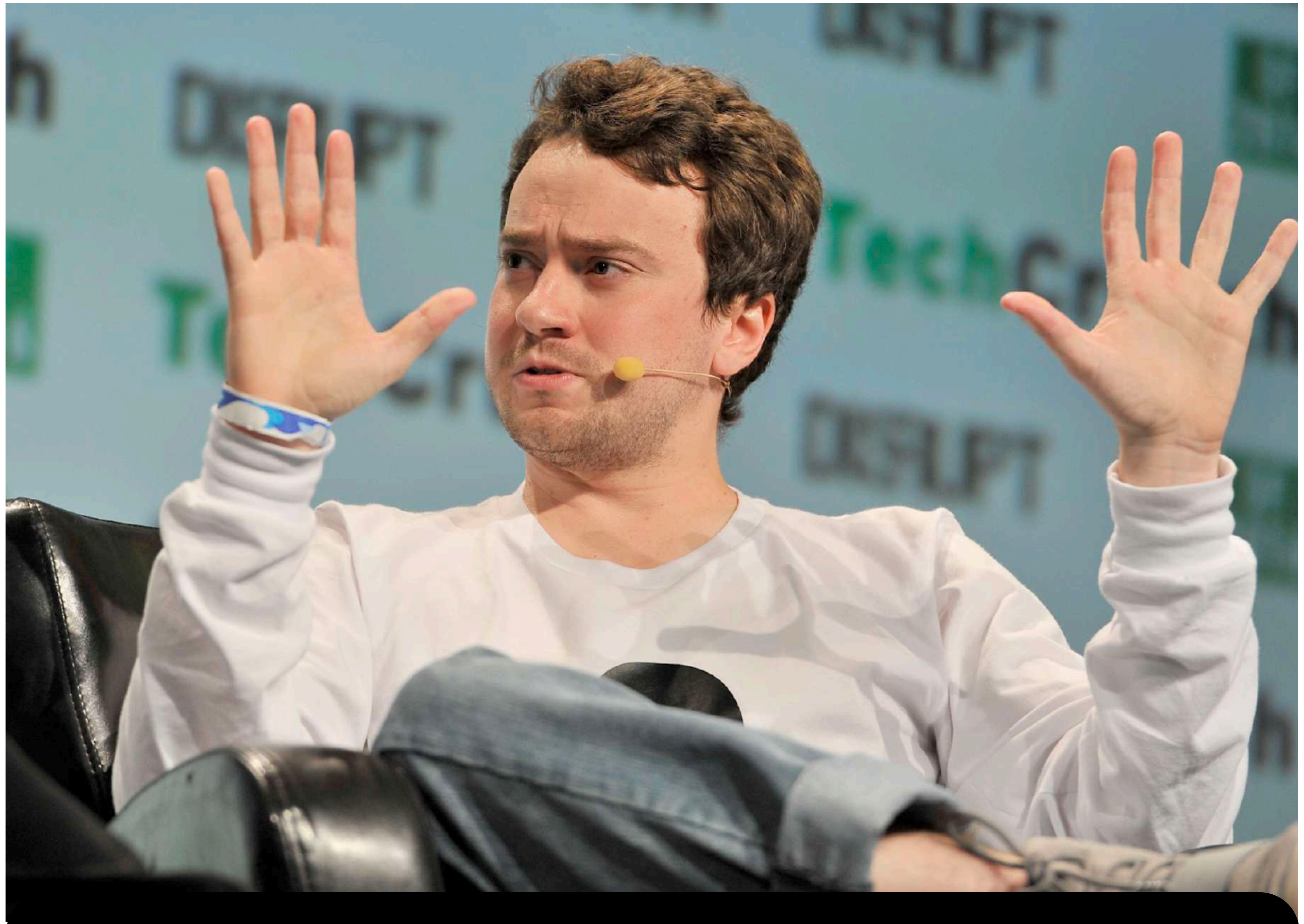
The whole logic follows the convention of Basic Linear Algebra Subprograms (BLAS) that is also implemented for CPU-based libraries.

USE-CASES	**CONSUMER INTERNET** Speeh Translate Recommender	**INDUSTRIAL APPLICATIONS** Healthcare Manufacturing Finance	**SUPERCOMPUTING** Molecular Simulation Weather forecasting Seismic Mapping
APPS & FRAMEWORKS	TensorFlow mxnet Rapids Pytorch Chainer ONNX	Amber NAMD +600 Applications	CATIA AUTODESK 3DS MAX Windows Adobe
CUDA-X LIBRARIES	**MACHINE LEARNING** CuDF CUML CUGRAPH	**DL / HPC** CUDA Math Libraries cuDNN CUTLASS TENSORRT	**LANGUAGES** LLVM Compiler for CUDA Python OpenACC C++
CUDA	**CUDA TOOLKIT** CUDA COMPILER DEVELOPER TOOLS: debuggers, profilers CUDA C++ CORE	**CUDA DRIVER** MEMORY MANAGEMENT WINDOWS & GRAPHICS COMMS LIBRARIES	
OS PLATFORMS	UBUNTU CentOS Windows Server SUSE RED HAT		

Hardware layer: GPU

AMD has its own implementation of
a computational framework — ROCm.
With HIP — the C++ Runtime interface.

But you could have problems with
AMD GPUs stability.



“Sorry, I’m not working with AMD GPUs anymore” — George “Geohot” Hotz.

Image source: Wikipedia
Quote from an AMD's issue on Github.

Hardware layer: GPU

When you need it

- When you deal with Stable Diffusion models

512x512 image generated:

171.70s on an AMD Epyc server with BLAS support, C++ implementation.

costs $300/month

2s for the AMD w7900PRO

costs $2.5k/pcs

Quinoa with salmon.

Generated on StableDiffusion.cpp

Hardware layer: GPU

When you need it

- When you need to train an LLM (billions of parameters) from scratch.
- Lack of memory for a model? Stack of GPU or use quantization.
- LLAMA.cpp supports both ROCm/HIP and CUDA. Define at the building stage.
- Check the StableDiffusion.CPP project.

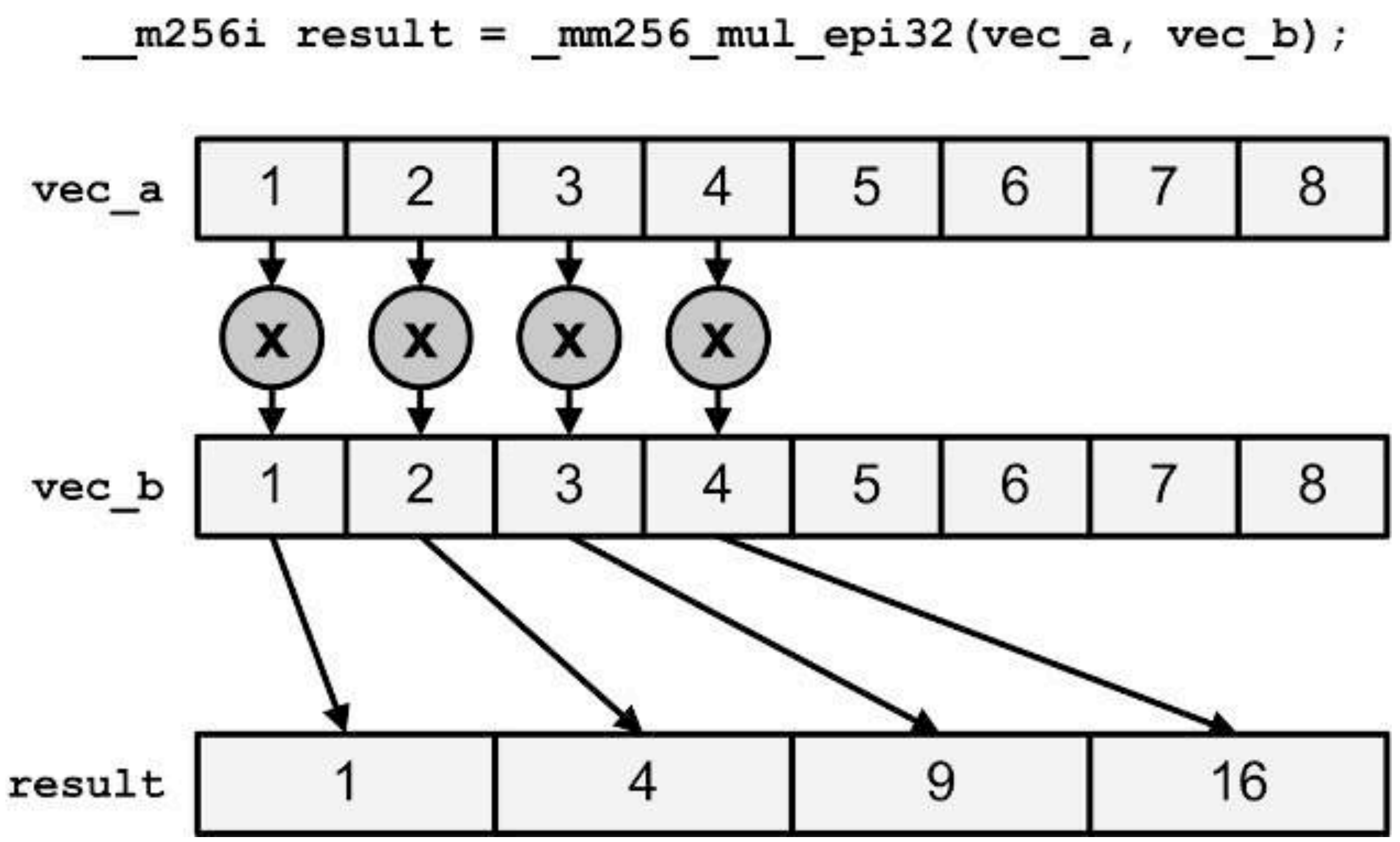
Hardware layer: CPU

Software optimization and hardware tweaks

There are low-level instructions for vector arithmetic.

It's architecture dependent. And **optimized for floating-point arithmetics.**

SIMD (Single Instruction Multiple Data) extensions: AVX, AVX2, AVX512.



```
__m256i result = _mm256_mul_epi32(vec_a, vec_b);
```

Crunching Numbers with AVX and AVX2

Matt Scarpino, codeproject.com

Hardware layer: CPU

Intel kills Alder Lake AVX-512 support for good

PCGamer.com

«I hope AVX-512 dies a painful death, and that Intel starts fixing real problems.» Linus Torvalds

Image: Linus Torvalds/TED/YouTube

Hardware layer: CPU

Linear algebra libraries

BLAS/LAPACK — Linear algebra low-level routines' libraries that are optimized heavily for different architectures.

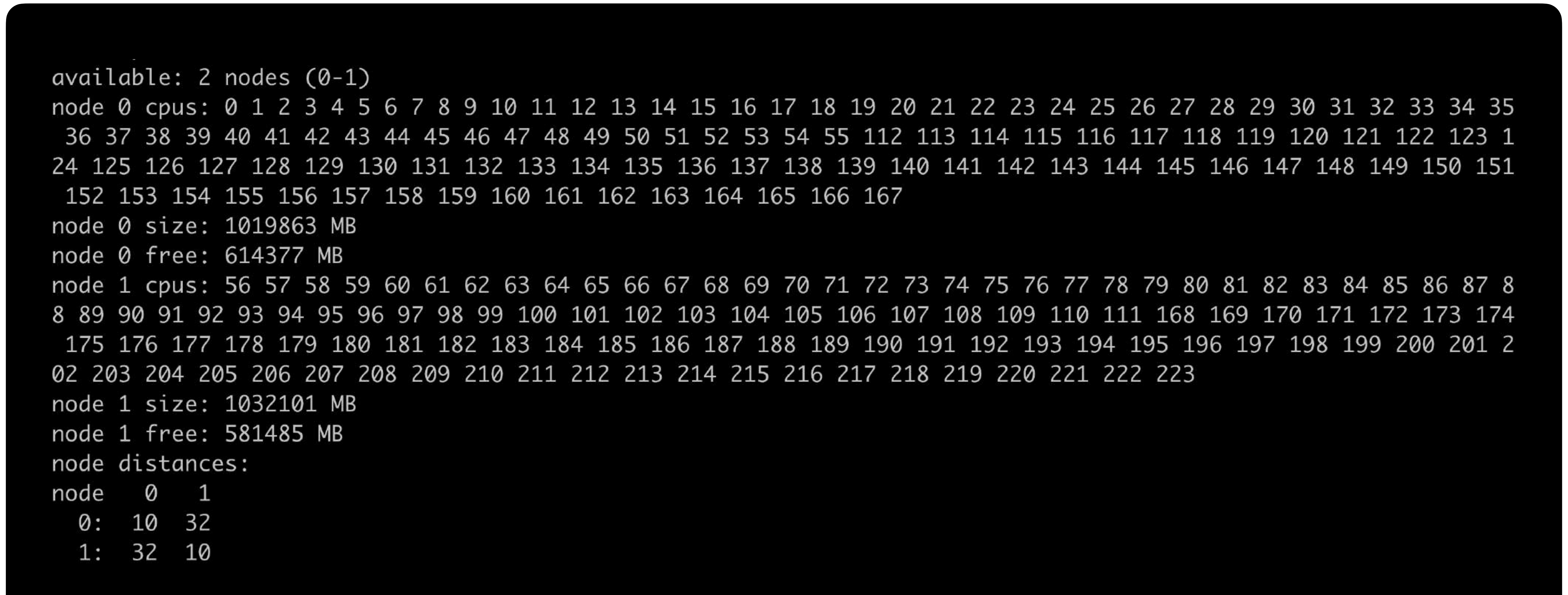
LLAMA.CPP could implement its functions on a bare CPU or over a GPU's driver.

Intel has its oneAPI (oneMKL — Math Kernel Library) — the library that helps develop parallelized algebra operations.

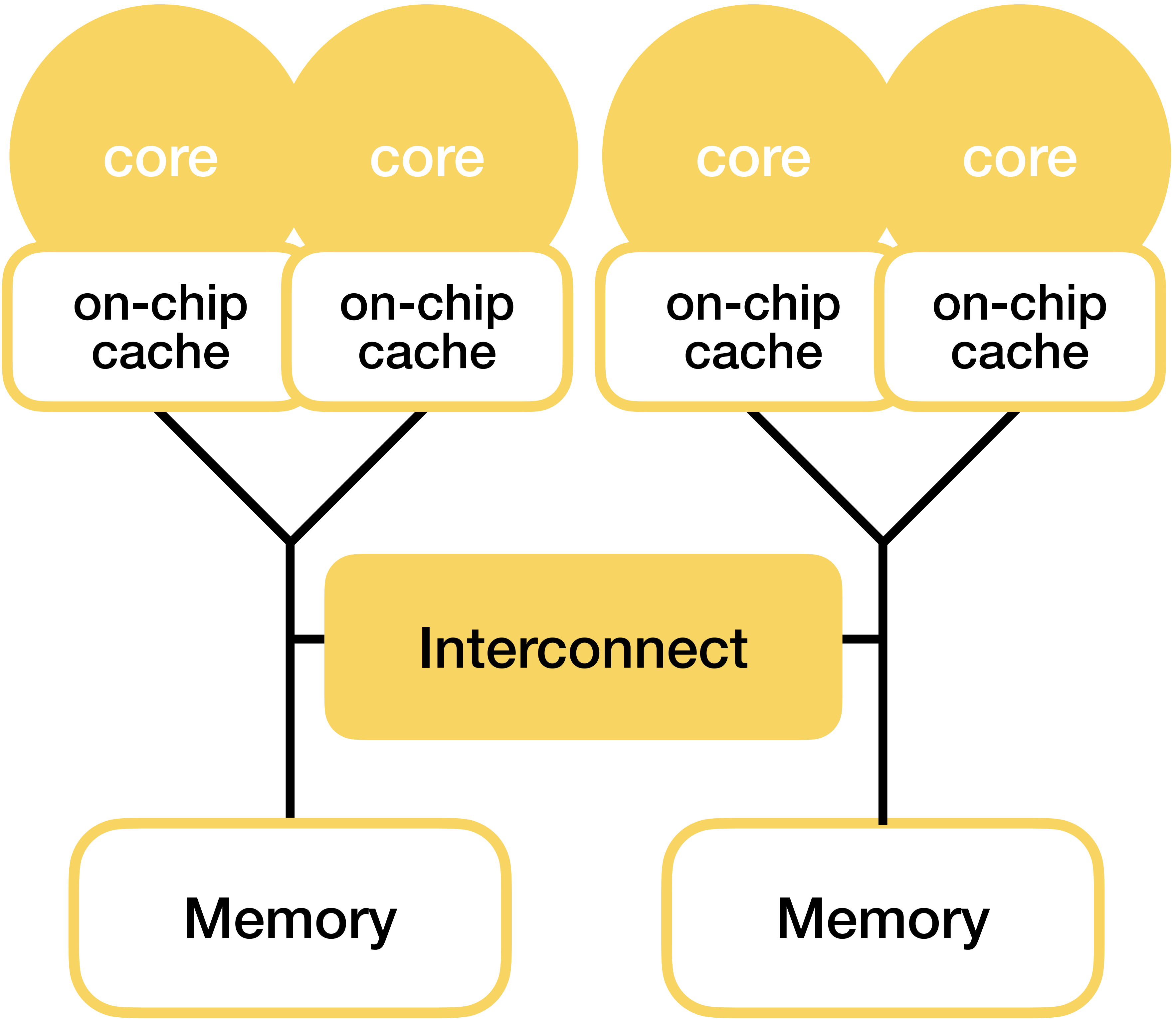
Hardware layer: CPU

NUMA — Non-uniform Memory Access

- Every single processor use its own memory.
- Faster access to NN's parameters in memory.
- You can run any code inside NUMA containers by hand.



```
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35
 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 112 113 114 115 116 117 118 119 120 121 122 123 1
24 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151
 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167
node 0 size: 1019863 MB
node 0 free: 614377 MB
node 1 cpus: 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 8
8 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 168 169 170 171 172 173 174
 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 2
02 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223
node 1 size: 1032101 MB
node 1 free: 581485 MB
node distances:
node   0   1
  0:  10  32
  1:  32  10
```



Taken from HPCWiki

Hardware layer: CPU

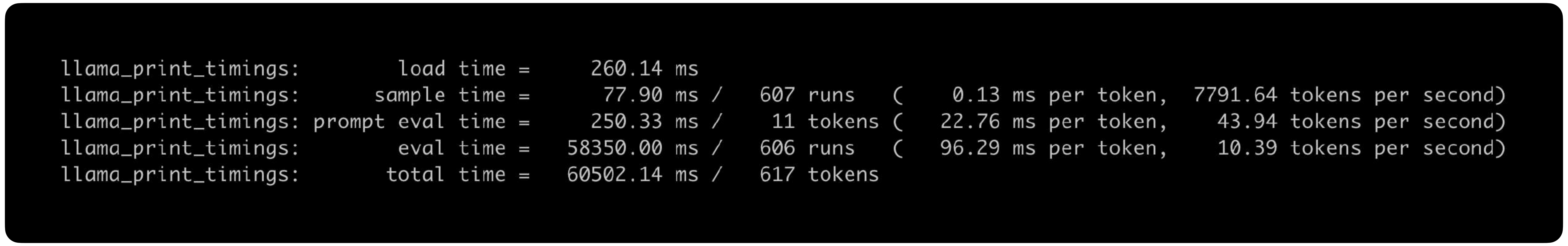
AMD Epyc — the personal choice

- AMD’s 3rd (7k) Gen Epyc has a great balance of price and performance.
- ~300 Eur/month for the dedicated server with 256Gb of DDR5 with 2Tb RAID1 in Hetzner.
- ~12 eur for the cloud based on the Epyc platform.
- It has 64 cores, 128 PCI lanes to connect between processors and 8 dedicated memory channels.

Hardware layer: CPU

AMD w7900 pro

shows x4 per token
($2.5k for just a card)

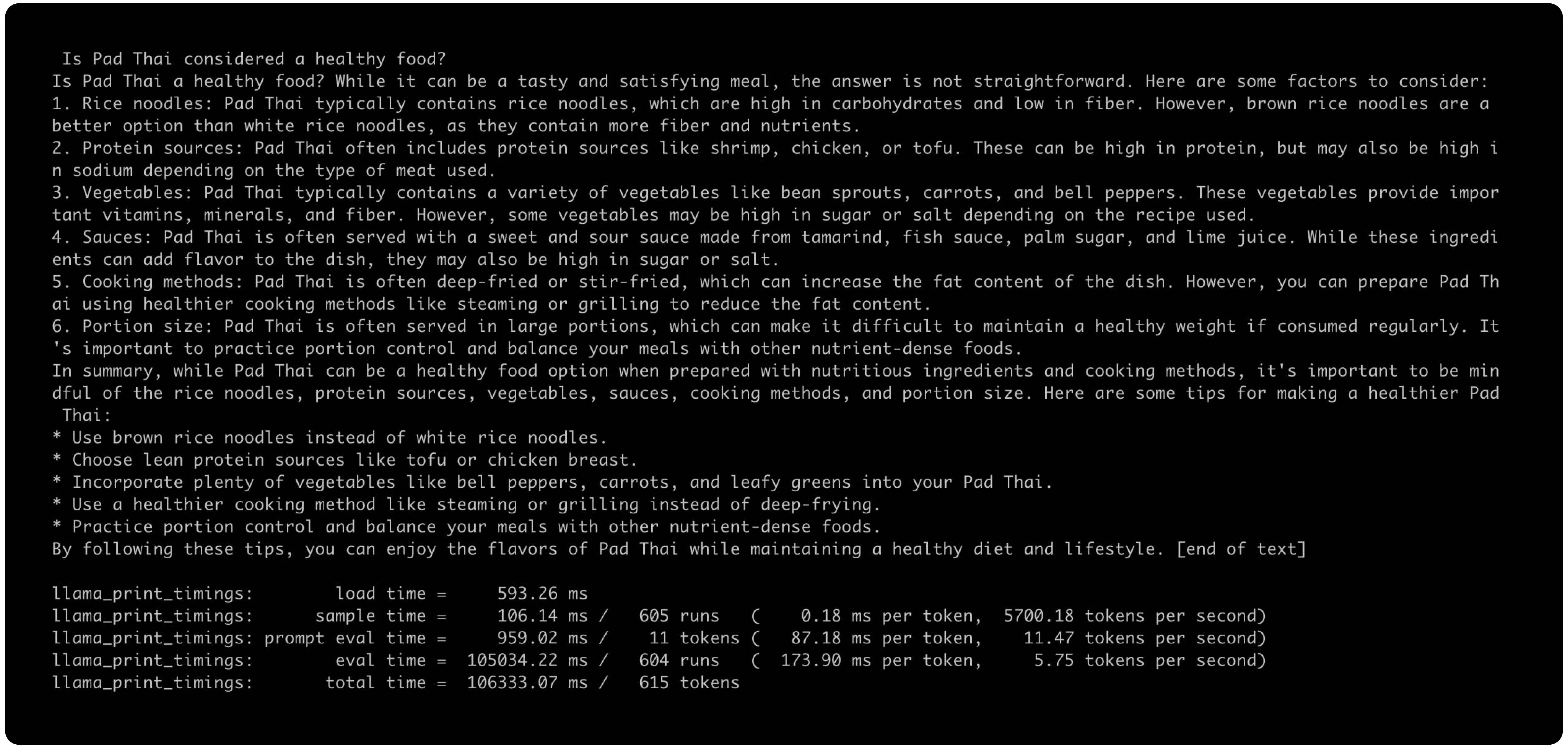


```
llama_print_timings:        load time =     260.14 ms
llama_print_timings:      sample time =      77.90 ms /   607 runs   (    0.13 ms per token,  7791.64 tokens per second)
llama_print_timings: prompt eval time =     250.33 ms /    11 tokens (   22.76 ms per token,    43.94 tokens per second)
llama_print_timings:        eval time =   58350.00 ms /   606 runs   (   96.29 ms per token,    10.39 tokens per second)
llama_print_timings:       total time =   60502.14 ms /   617 tokens
```

LLAMA.CPP

Pure Epyc CPU with BLAS support

~ **40 sec difference** for a bunch of text.

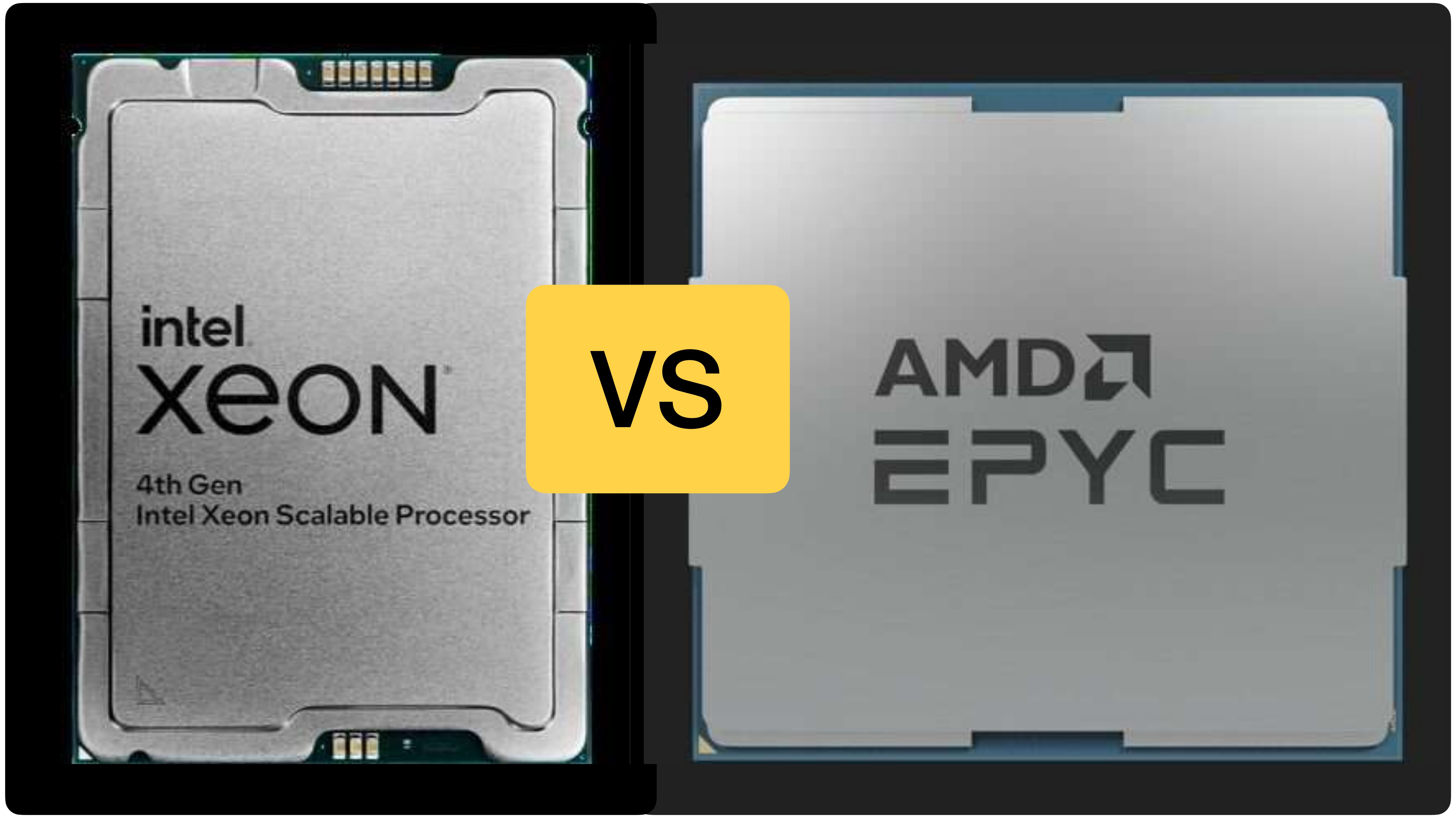


```
 Is Pad Thai considered a healthy food?
Is Pad Thai a healthy food? While it can be a tasty and satisfying meal, the answer is not straightforward. Here are some factors to consider:
1. Rice noodles: Pad Thai typically contains rice noodles, which are high in carbohydrates and low in fiber. However, brown rice noodles are a
better option than white rice noodles, as they contain more fiber and nutrients.
2. Protein sources: Pad Thai often includes protein sources like shrimp, chicken, or tofu. These can be high in protein, but may also be high i
n sodium depending on the type of meat used.
3. Vegetables: Pad Thai typically contains a variety of vegetables like bean sprouts, carrots, and bell peppers. These vegetables provide impor
tant vitamins, minerals, and fiber. However, some vegetables may be high in sugar or salt depending on the recipe used.
4. Sauces: Pad Thai is often served with a sweet and sour sauce made from tamarind, fish sauce, palm sugar, and lime juice. While these ingredi
ents can add flavor to the dish, they may also be high in sugar or salt.
5. Cooking methods: Pad Thai is often deep-fried or stir-fried, which can increase the fat content of the dish. However, you can prepare Pad Th
ai using healthier cooking methods like steaming or grilling to reduce the fat content.
6. Portion size: Pad Thai is often served in large portions, which can make it difficult to maintain a healthy weight if consumed regularly. It
's important to practice portion control and balance your meals with other nutrient-dense foods.
In summary, while Pad Thai can be a healthy food option when prepared with nutritious ingredients and cooking methods, it's important to be min
dful of the rice noodles, protein sources, vegetables, sauces, cooking methods, and portion size. Here are some tips for making a healthier Pad
 Thai:
* Use brown rice noodles instead of white rice noodles.
* Choose lean protein sources like tofu or chicken breast.
* Incorporate plenty of vegetables like bell peppers, carrots, and leafy greens into your Pad Thai.
* Use a healthier cooking method like steaming or grilling instead of deep-frying.
* Practice portion control and balance your meals with other nutrient-dense foods.
By following these tips, you can enjoy the flavors of Pad Thai while maintaining a healthy diet and lifestyle. [end of text]

llama_print_timings:        load time =     593.26 ms
llama_print_timings:      sample time =     106.14 ms /   605 runs   (    0.18 ms per token,  5700.18 tokens per second)
llama_print_timings: prompt eval time =     959.02 ms /    11 tokens (   87.18 ms per token,    11.47 tokens per second)
llama_print_timings:        eval time =  105034.22 ms /   604 runs   (  173.90 ms per token,     5.75 tokens per second)
llama_print_timings:       total time =  106333.07 ms /   615 tokens
```

Hardware layer: CPU

Intel launched “Sapphire Rapids” 4th Gen processors a year ago…



Intel Pits Its “Sapphire Rapids” Xeon SP Against AMD “Genoa” Epycs
TheNextPlatform blog

Hardware layer: CPU

Nowadays CPU manufacturers market...

Hardware layer: CPU

ARM is like…

- THE BRAND NEW AI chips to 2025
- SoftBank is about to use its AI chips (SoftBank holds 90% of stocks).

Chatbot that helps people to achieve goals

Mealtune.com

Links from slides

github.com/cegorah/thai_py_2024

Questions?