

# Нейронные сети в КОМПЬЮТЕРНОМ ЗРЕНИИ

**Постановки задач, приложения, железо, метрики.**

Даня Меркулов

17 мая 2023

**Приложения и кейсы  
использования  
нейронных сетей сегодня**



# Сельское хозяйство

## Статья

Глубокое обучение для сельского хозяйства. Обзор.

## Код

Использование PyTorch для решения задач в сельском хозяйстве.

## Код

Технологии с открытым исходным кодом для сельского хозяйства, фермерства и садоводства.



Sharpness



Color Clarity



Field Coverage

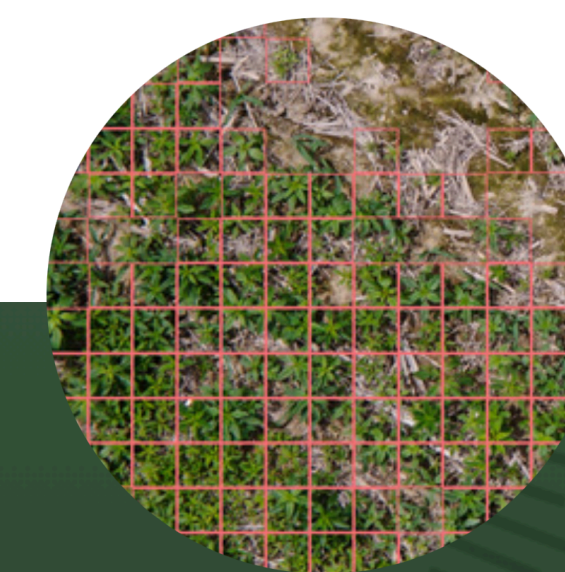


Image Distribution

# Девелопмент



Код

Инструменты для запуска глубокого обучения (DL) на спутниковых снимках.

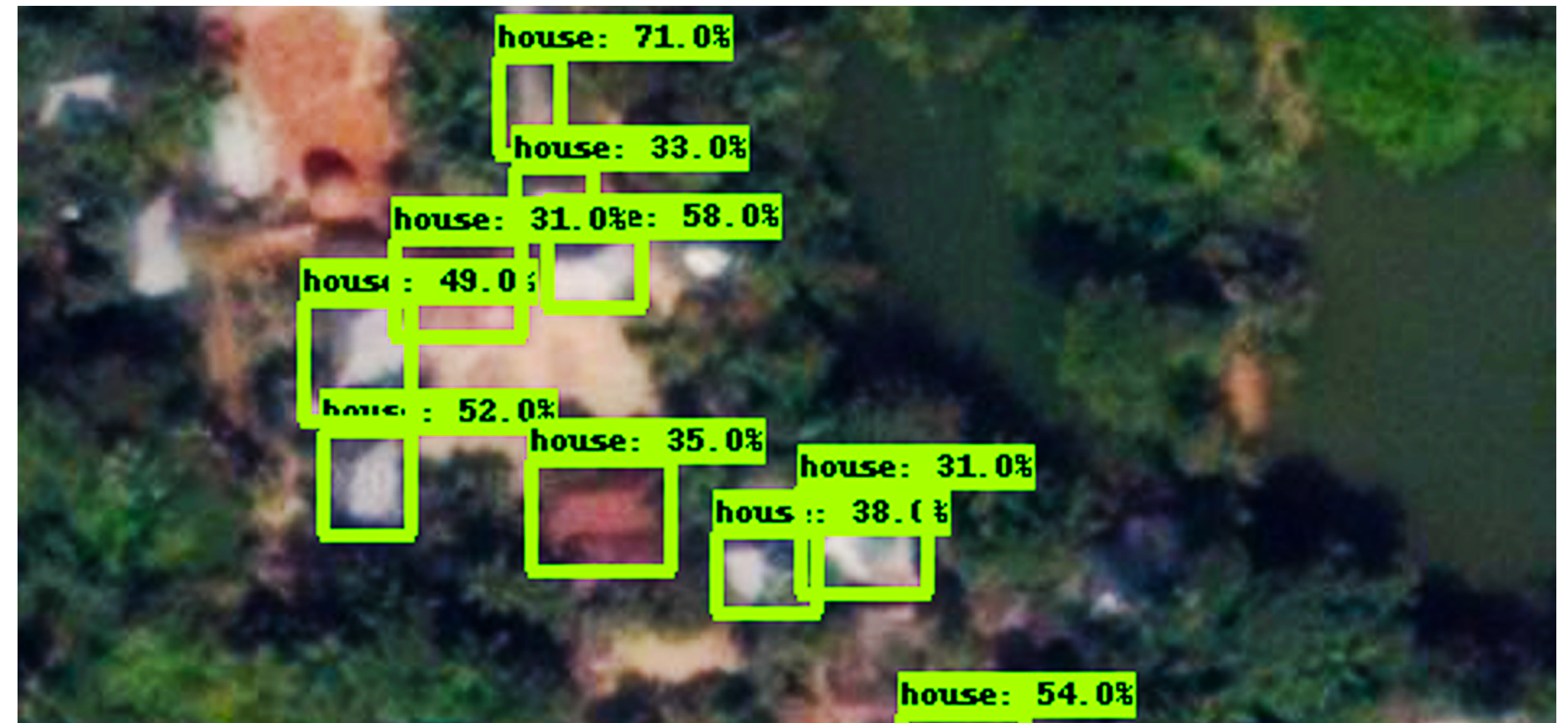


Код

Детектирование домов на спутниковых снимках с помощью нейронной сети.

## Проект Sunroof от Google

Предсказание о том, сколько может сэкономить человек, если он поставит на крышу своего дома солнечные панели по геолокации в Google Maps (USA only).



15 Glendale Ave, Somerville, MA 02144, USA



1,479 hours of usable sunlight per year  
Based on day-to-day analysis of weather patterns



497 sq feet available for solar panels  
Based on 3D modeling of roof and nearby trees

**\$18,000 savings**

Estimated net savings for roof over 20 years

CHECK MY ROOF



# Здравоохранение



**Код**

Инструментарий для глубокого обучения в области медицинских изображений.



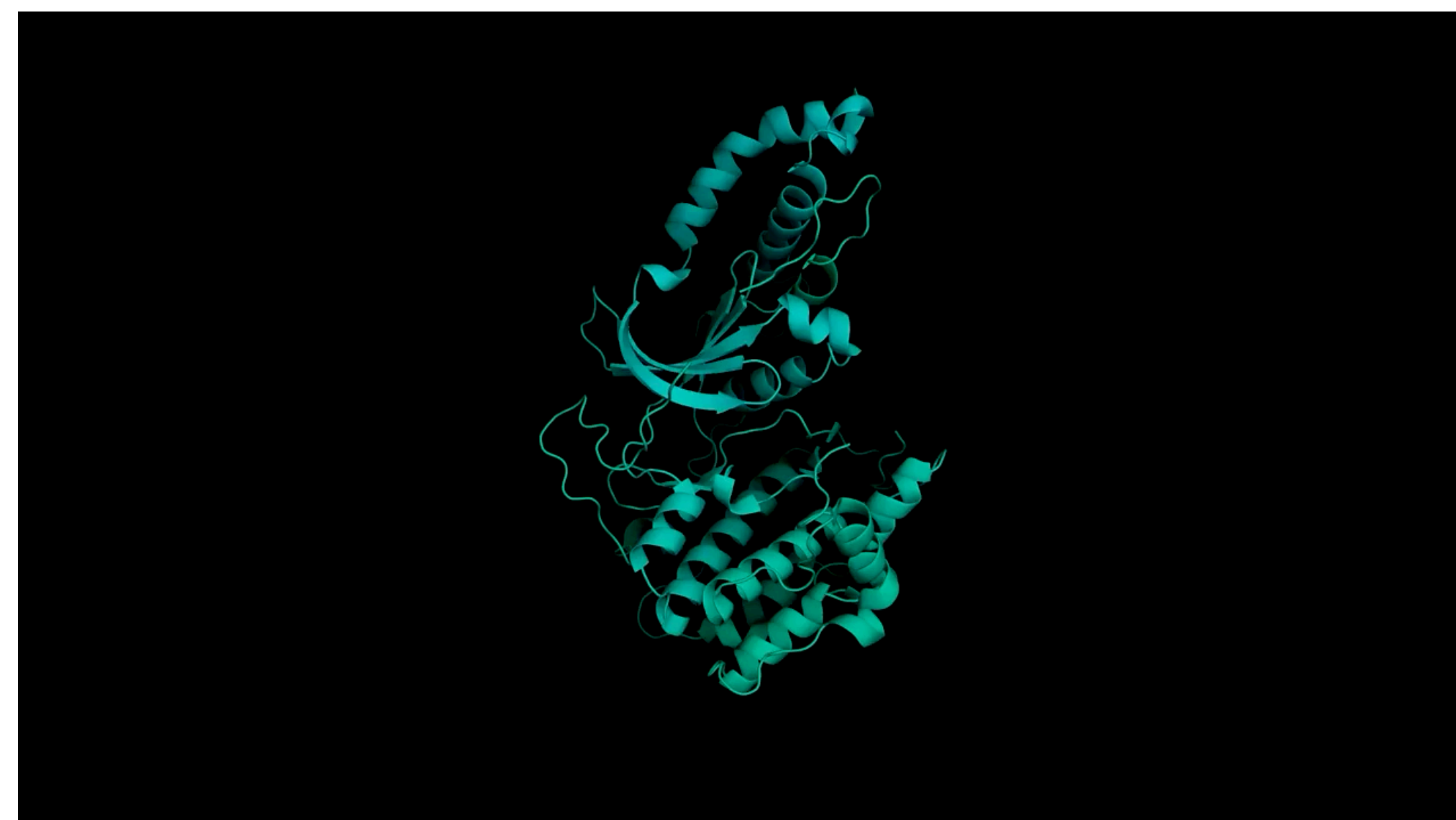
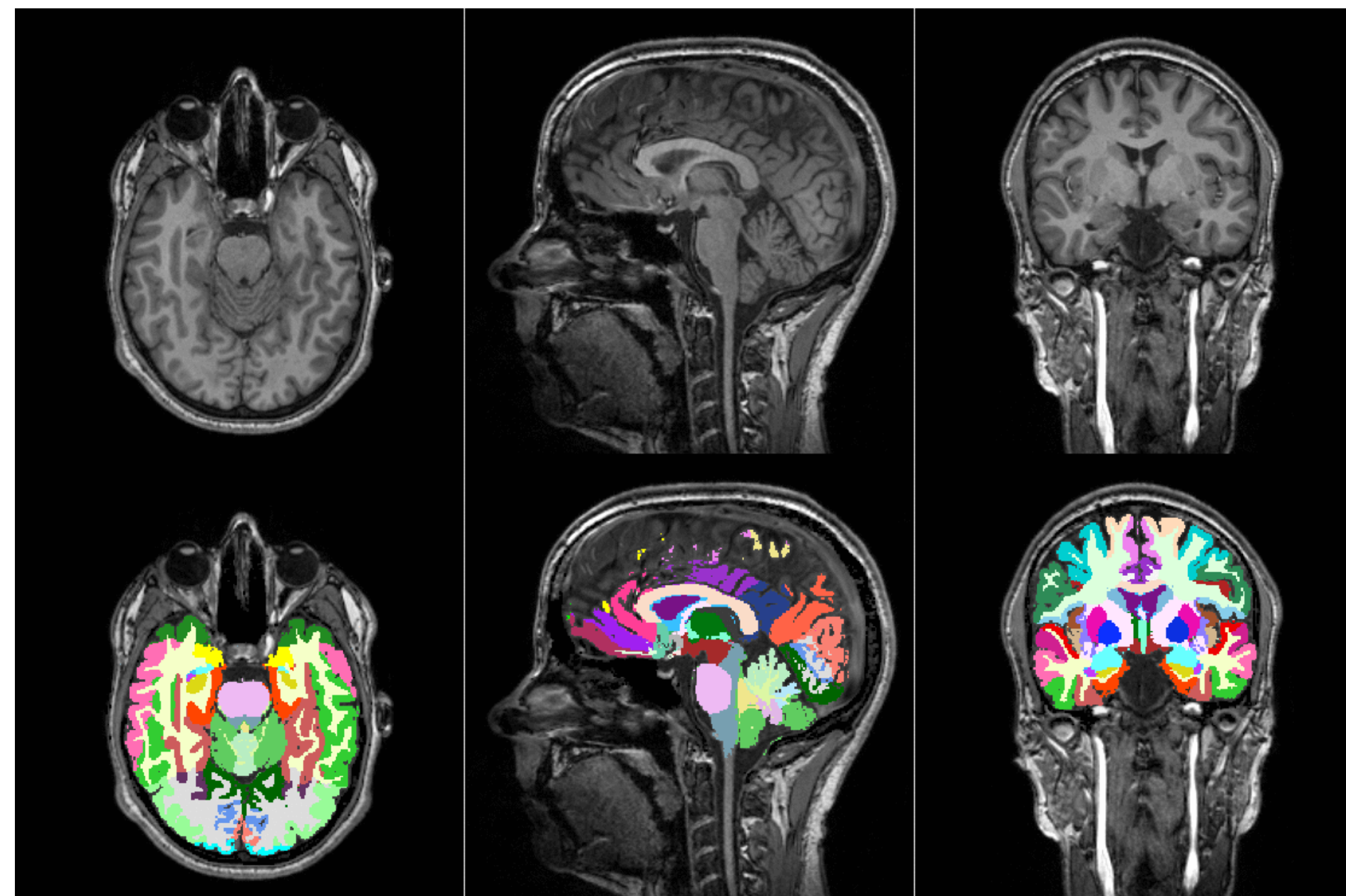
**Компания BenevolentAI**

Анализ химических соединений и поиск лекарств с помощью ИИ



**Статья**

Мультимодальное машинное обучение в точном здравоохранении: Обзор.



# Эволюция алгоритмов распознавания



# 2006. Классификация 🐱/🐶

КАПУША



Методы компьютерного зрения  
60%

$$0.6^{12} = 0.00217$$


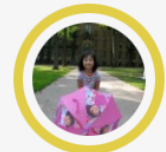





# 2014. Классификация 🐱/🐶



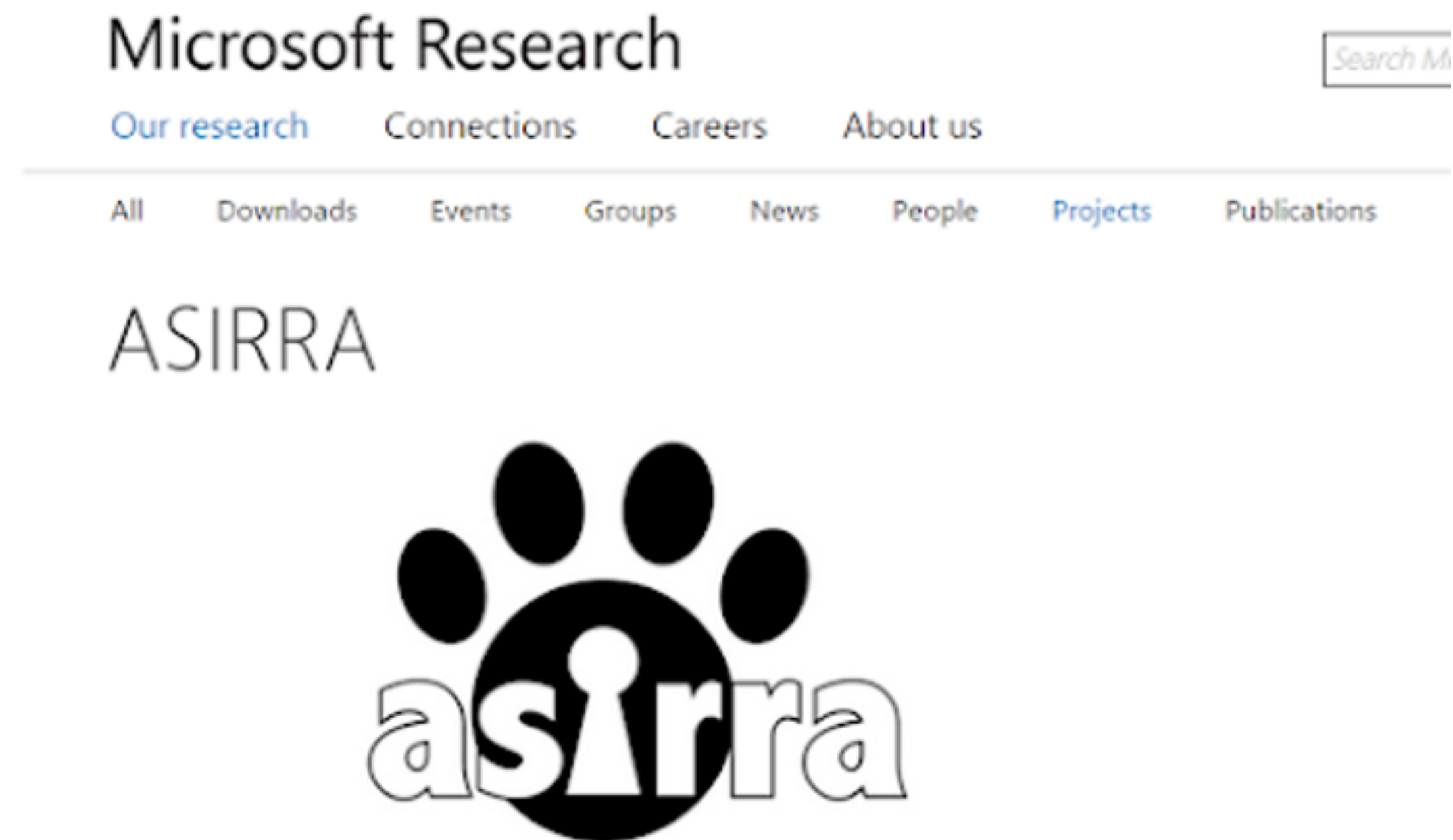
Нейронные сети  
99%

$0.989^{12} = 0.875$

#	△	Team	Members	Score
1	—	Pierre Sermanet		0.98914
2	▲ 4	orchid		0.98308
3	—	Owen		0.98171
4	—	Paul Covington		0.98171
5	▼ 3	Maxim Milakov		0.98137

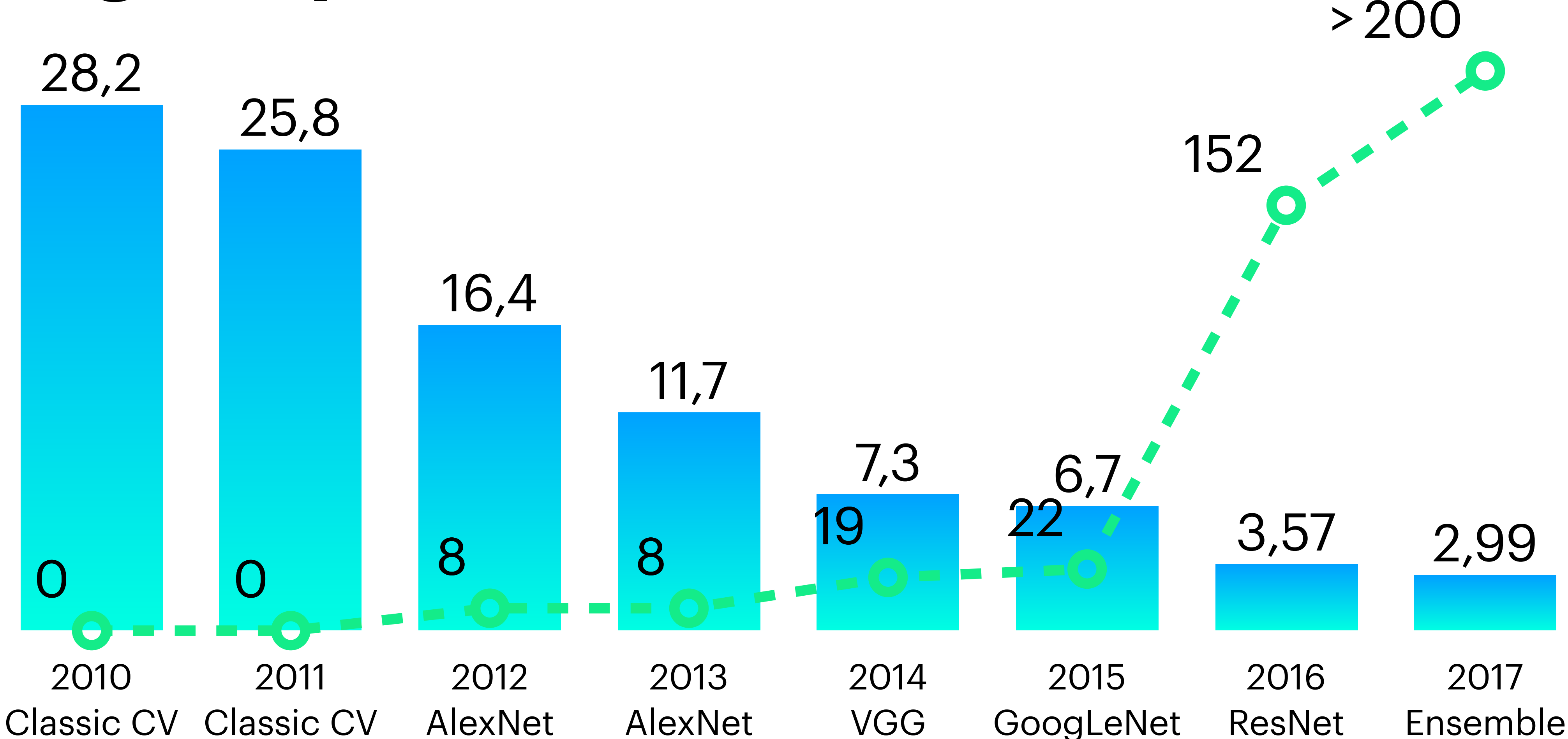


# 2014. Классификация 🐱 / 🐶



After 8 years of operation, Asirra is shutting down effective October 1, 2014. Thank you to all of our users!

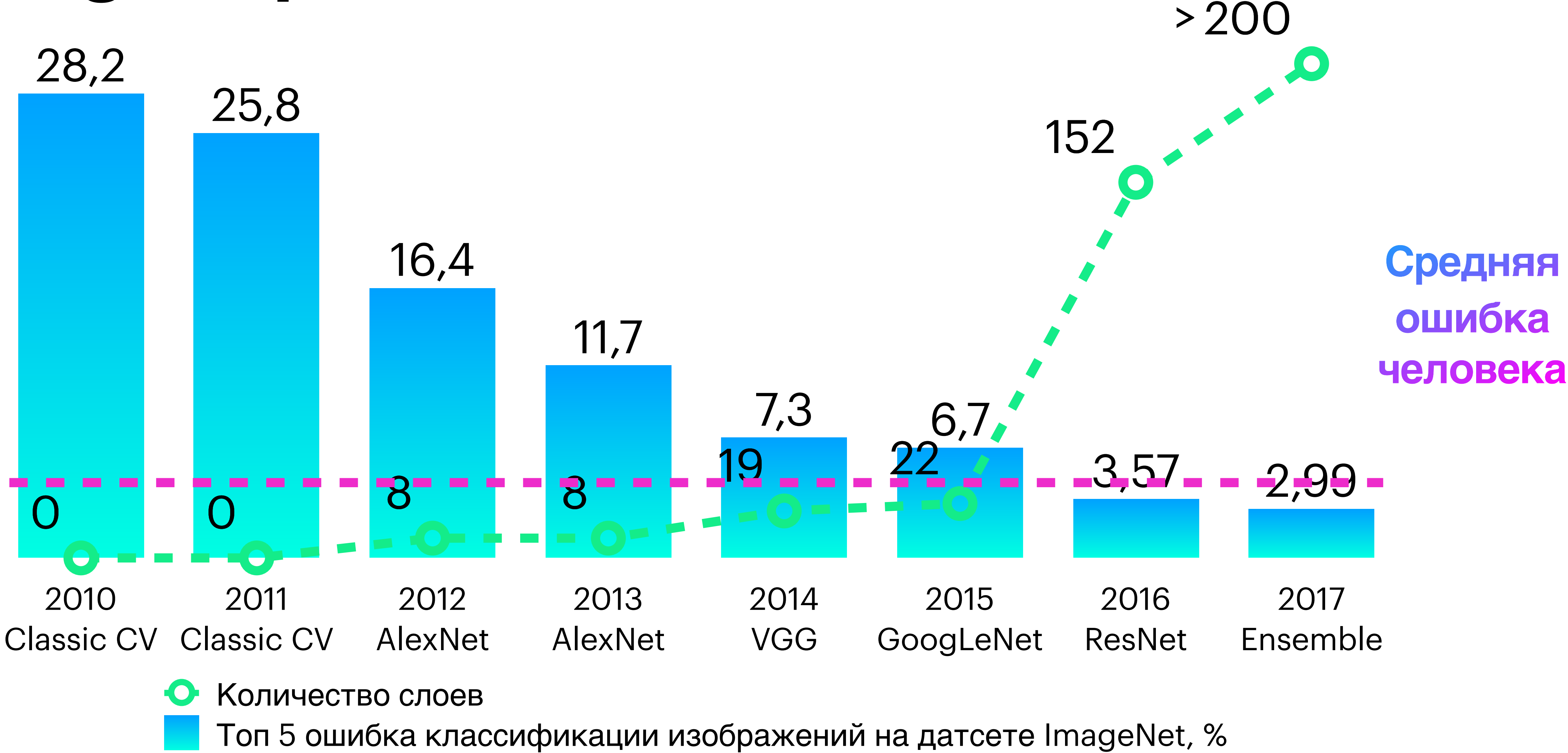
# Going deep



○ Количество слоев

■ Топ 5 ошибка классификации изображений на датсете ImageNet, %

# Going deep



# Генерация лиц



2014



2015



2016



2017



2018



2022  
Midjourney



Сайт

This person does not exist



Сайт

This cat does not exist



Сайт

This X does not exist

# Нейронные сети



# Как работает 🧠 ?

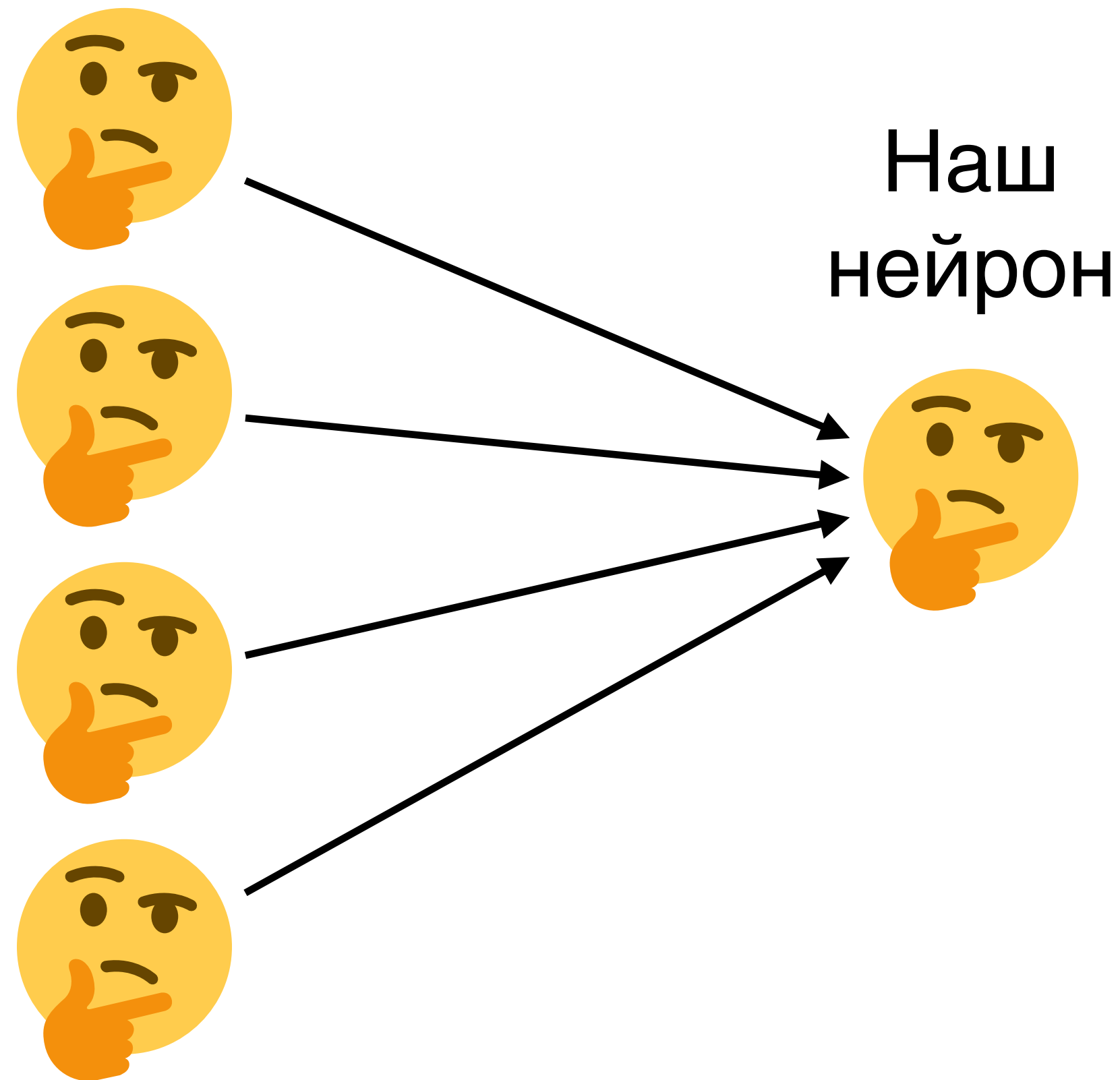


$8.6 \times 10^{10}$  нейронов в мозге человека

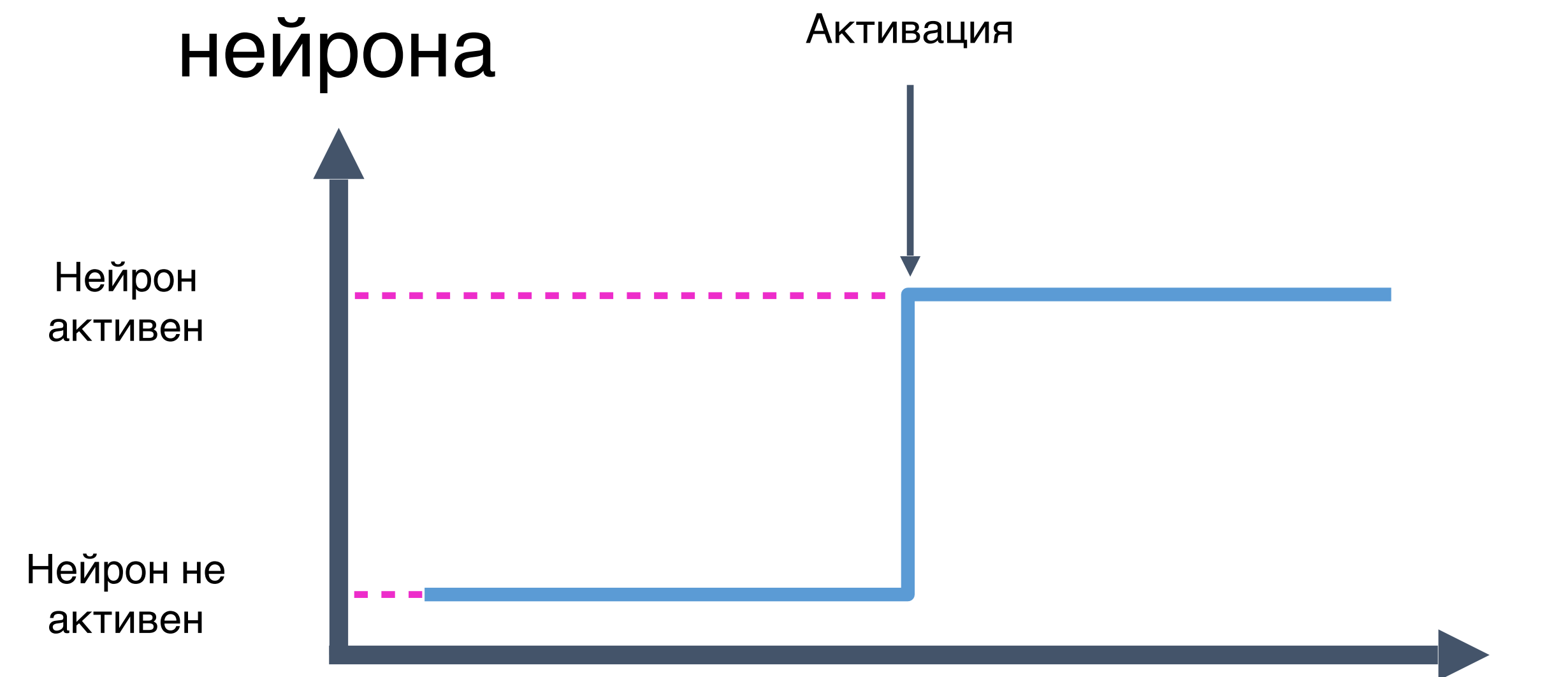
$\sim 1.5 \times 10^{14}$  связей между ними

# Как работает 🧠 ?

Другие  
нейроны



Реакция  
нейрона



Сумма входных сигналов  
от других нейронов

# Как работает 🤖🧠 ?

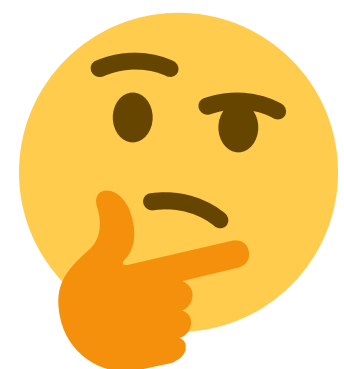
Другие  
нейроны



$x_1$



$x_2$



$x_3$

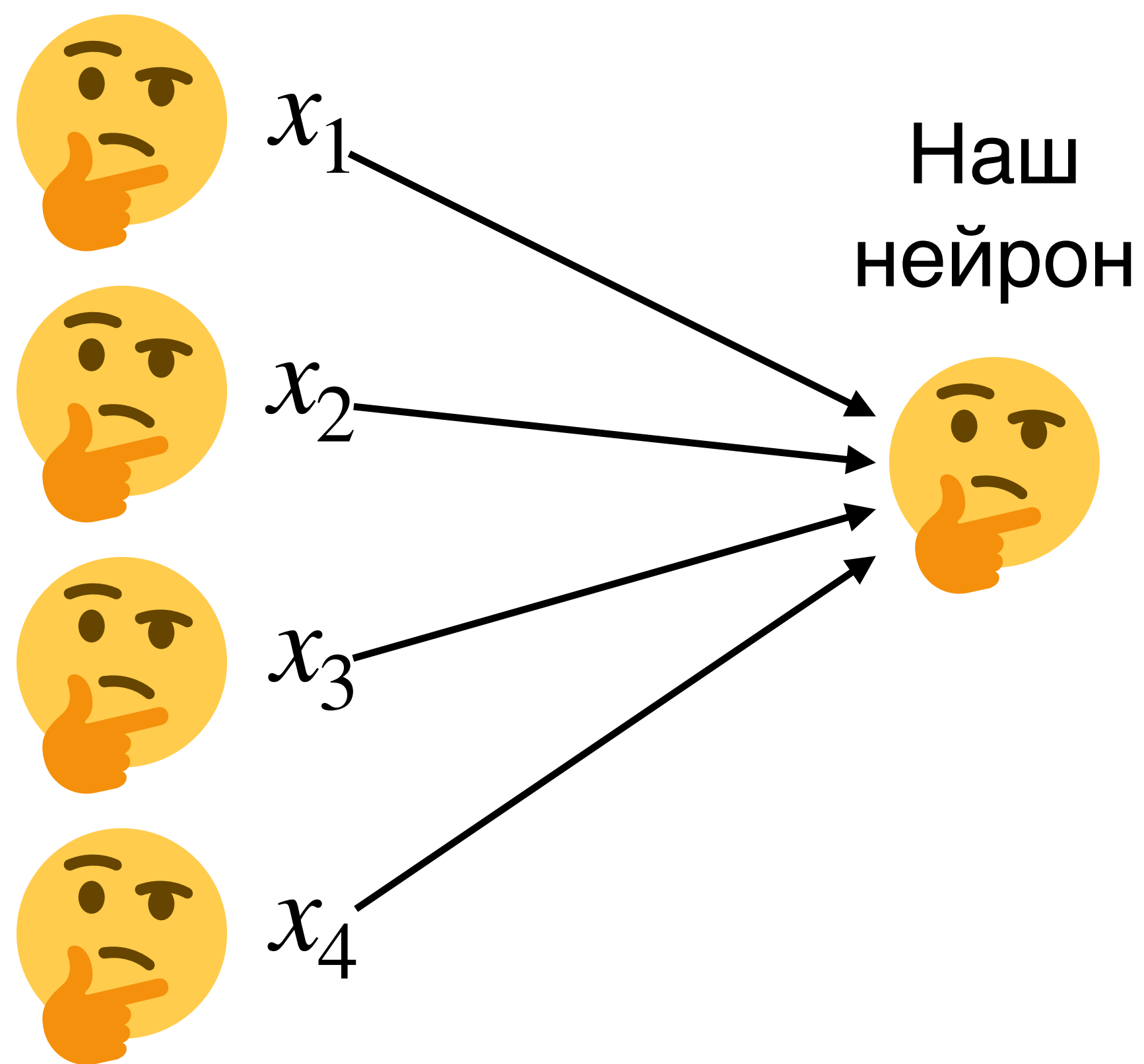


$x_4$



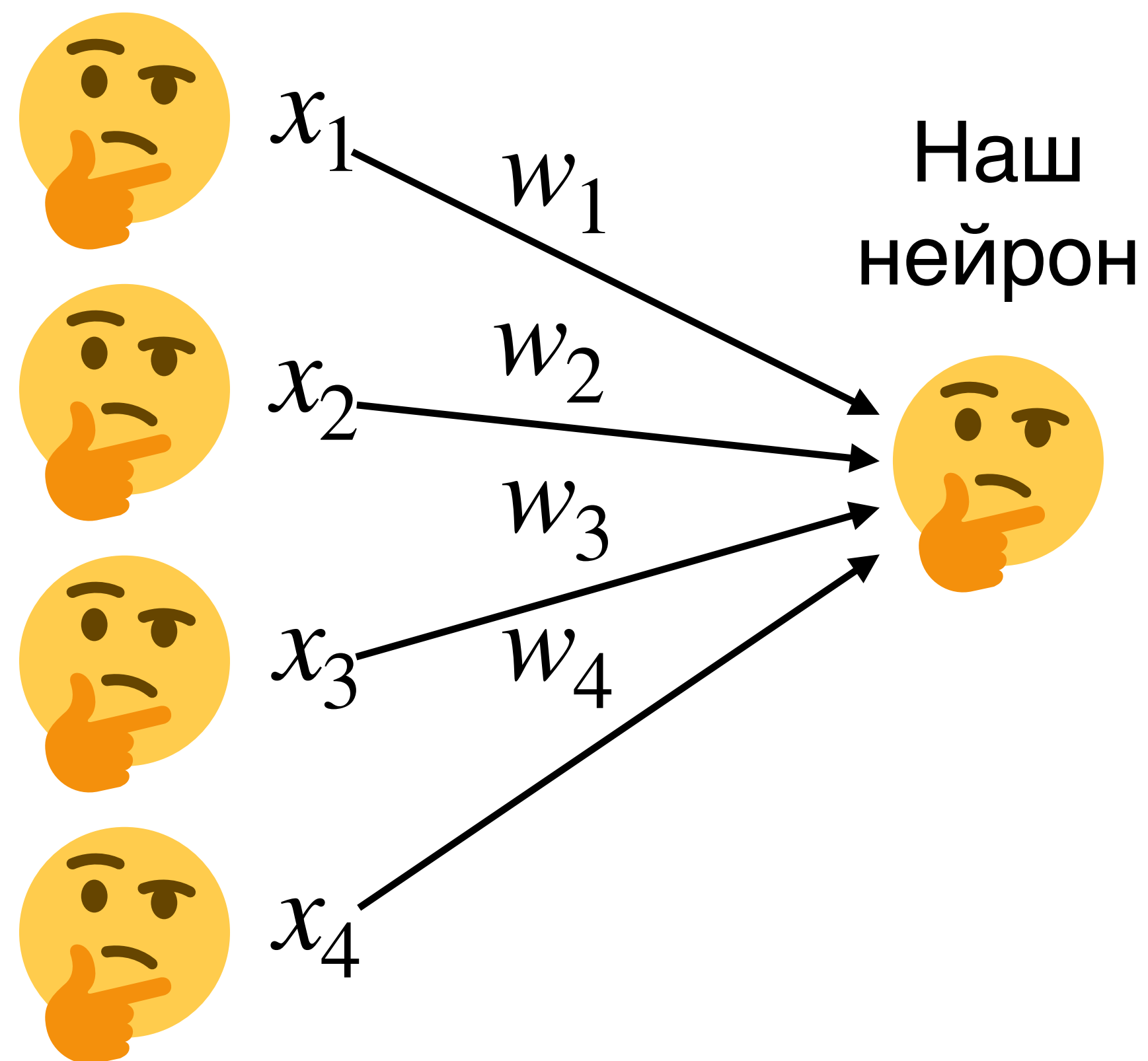
# Как работает 🤖🧠 ?

Другие  
нейроны



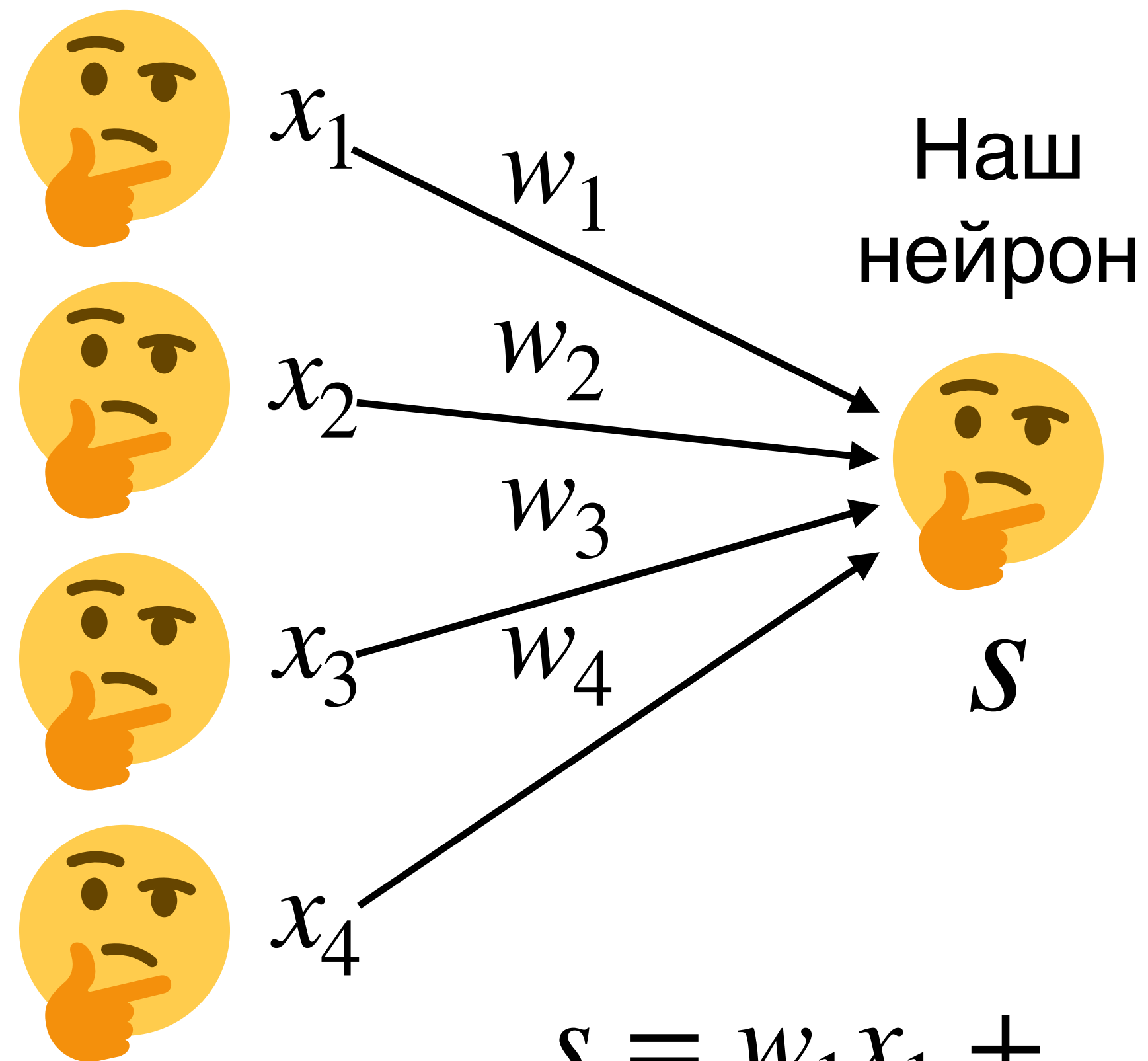
# Как работает 🤖🧠 ?

Другие  
нейроны



# Как работает 🤖🧠 ?

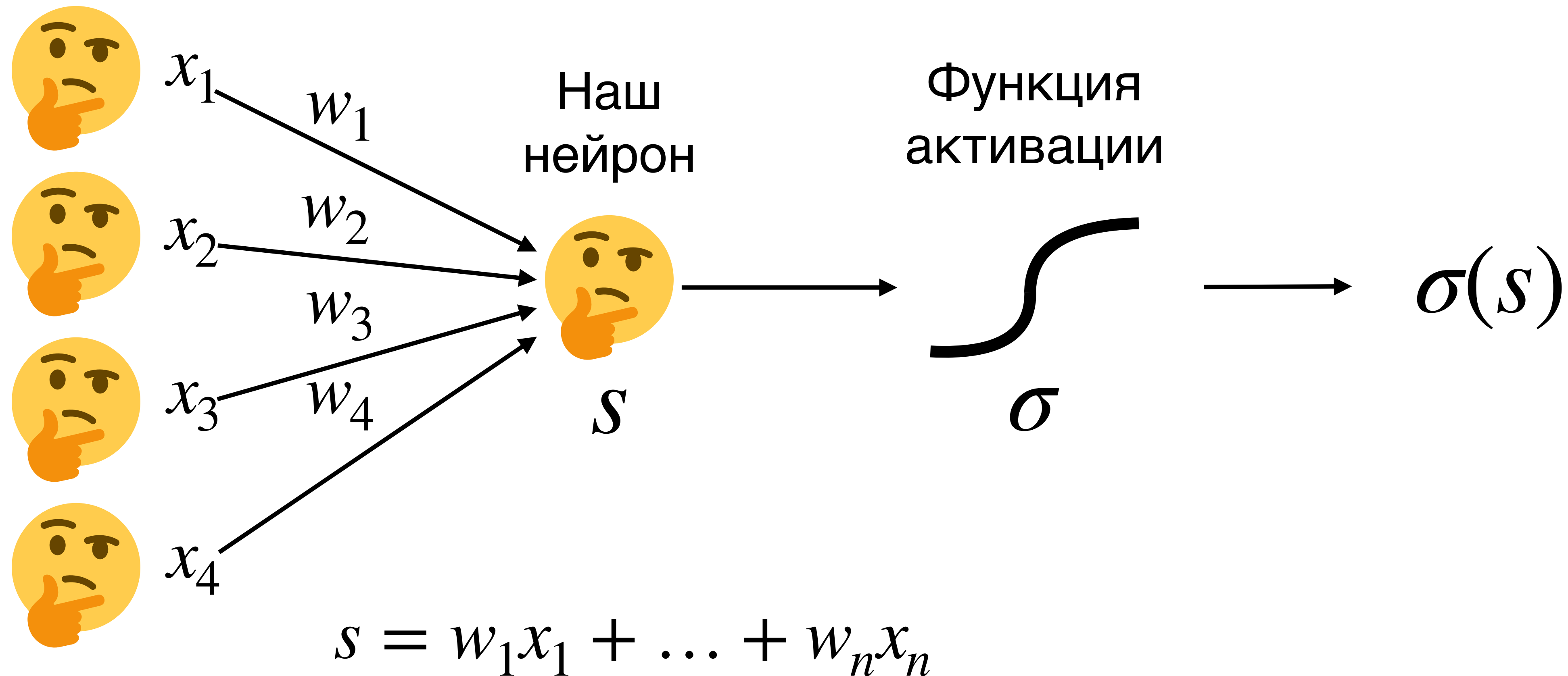
Другие  
нейроны



$$S = w_1x_1 + \dots + w_nx_n$$

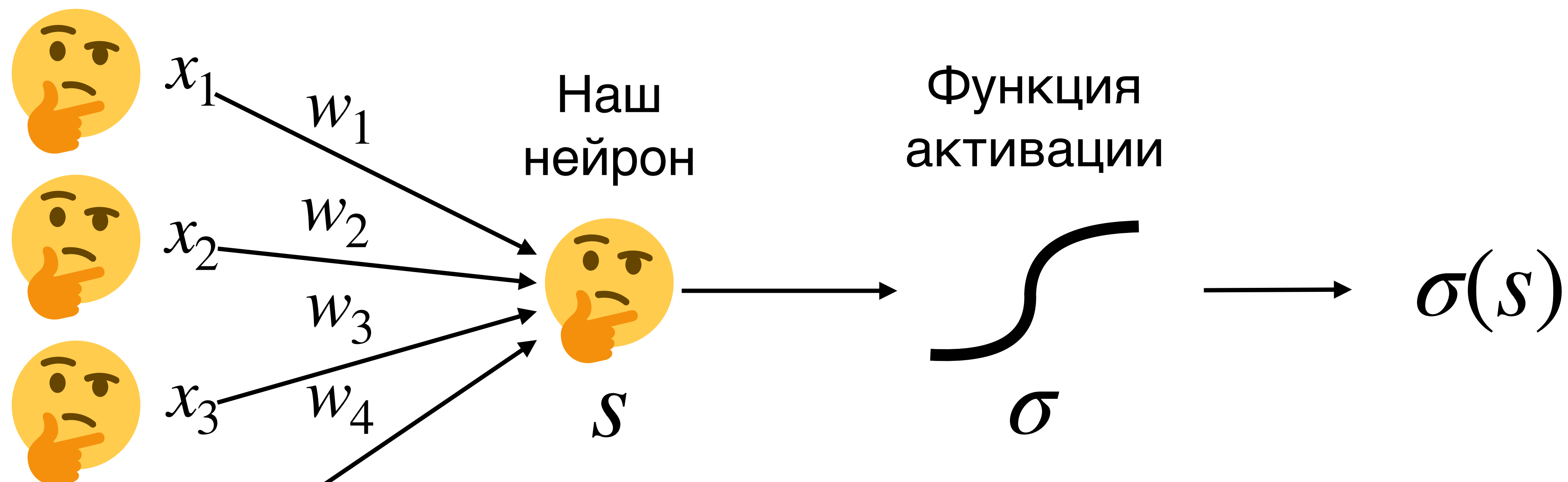
# Как работает 🤖🧠 ?

Другие  
нейроны



# Как работает 🤖🧠 ?

Другие  
нейроны



$$s = w_1x_1 + \dots + w_nx_n$$

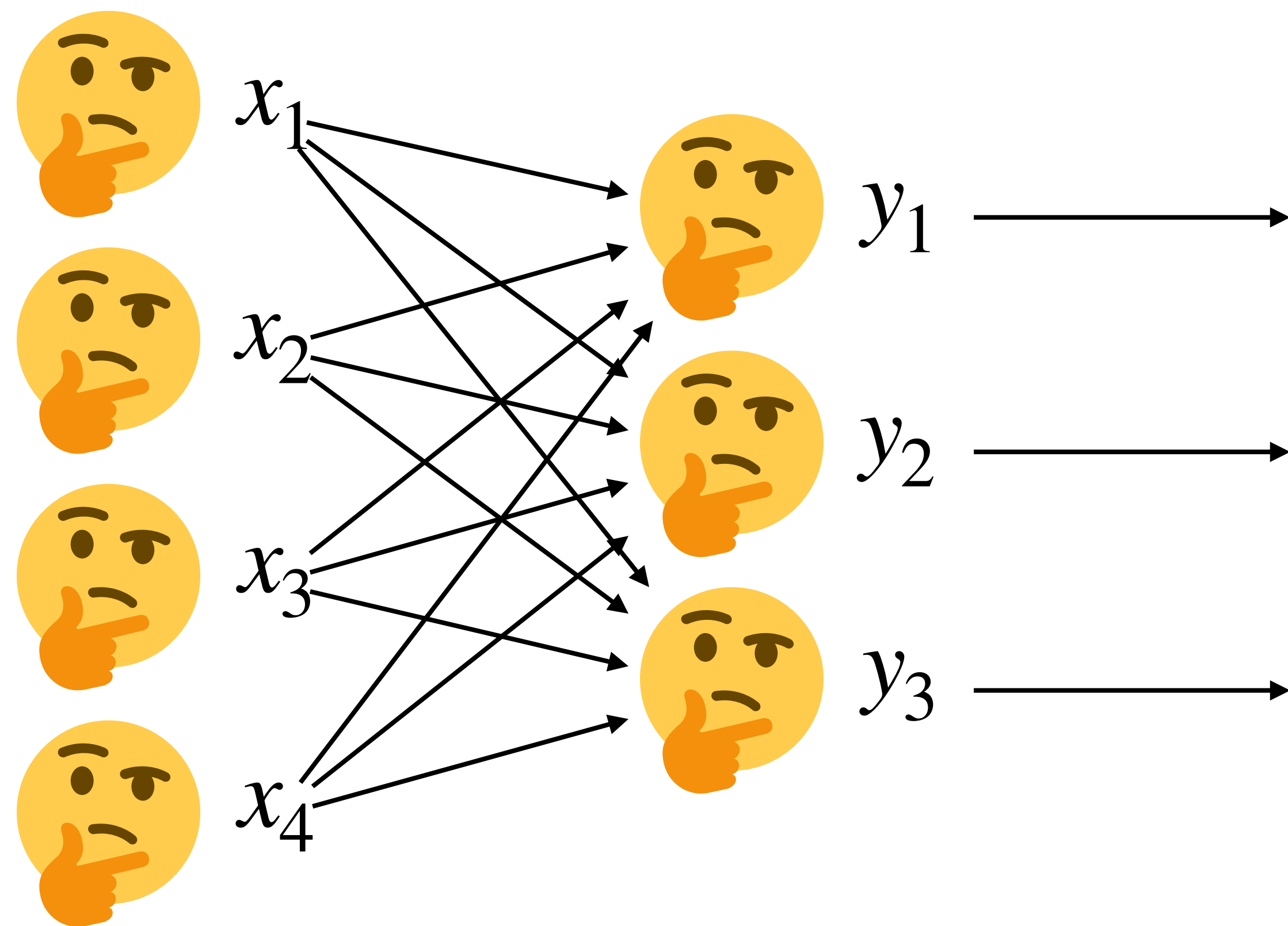
**Вход:**  $x_1$   $x_2$   $x_3$   $x_4$

**Выход:**  $\sigma(w_1x_1 + \dots + w_nx_n)$

**Параметры:**  $w_1$   $w_2$   $w_3$   $w_4$

# Как работает 🤖🧠 ?

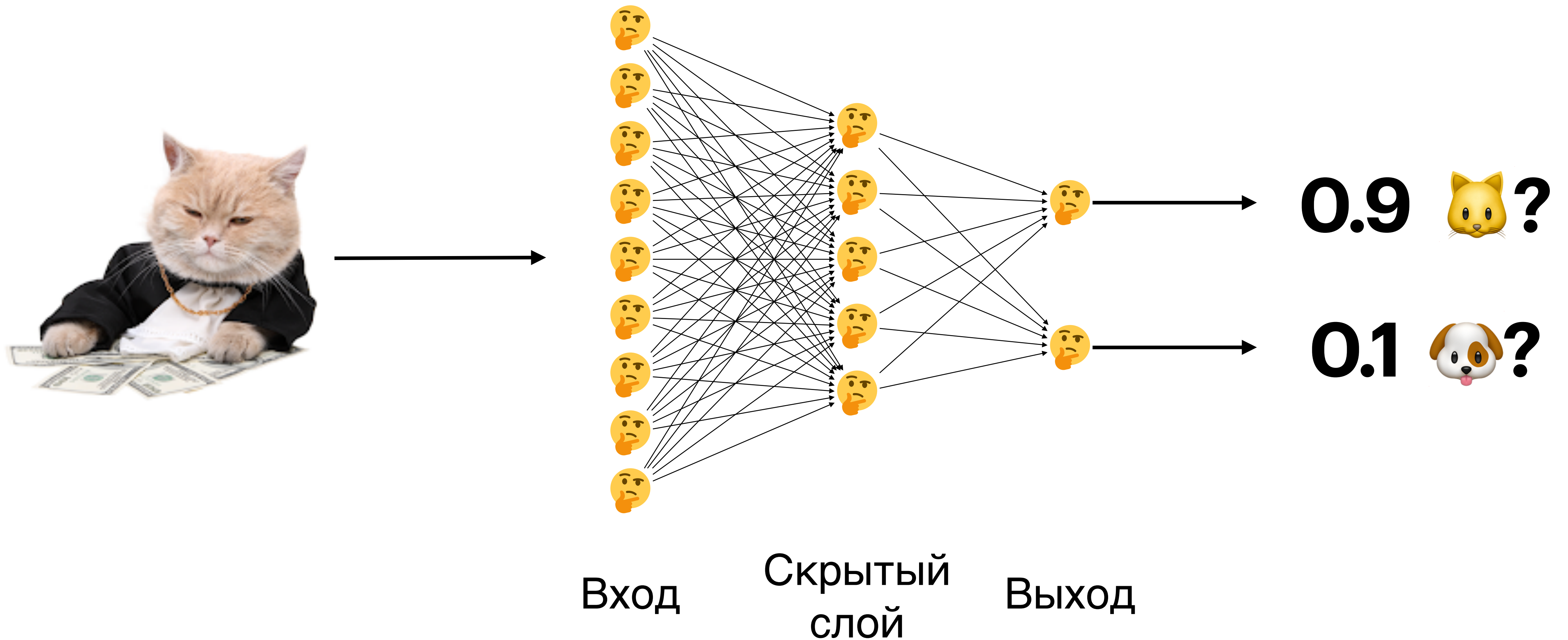
Возьмем несколько нейронов и получим полносвязный слой



$$y = \sigma(Wx)$$

3x1                      3x4 4x1

# Как использовать 🤖🧠 ?



# Обучение нейросети

Объект  
 $x \in \mathbb{R}^d; y \in \mathbb{R}^c$



КОТИК

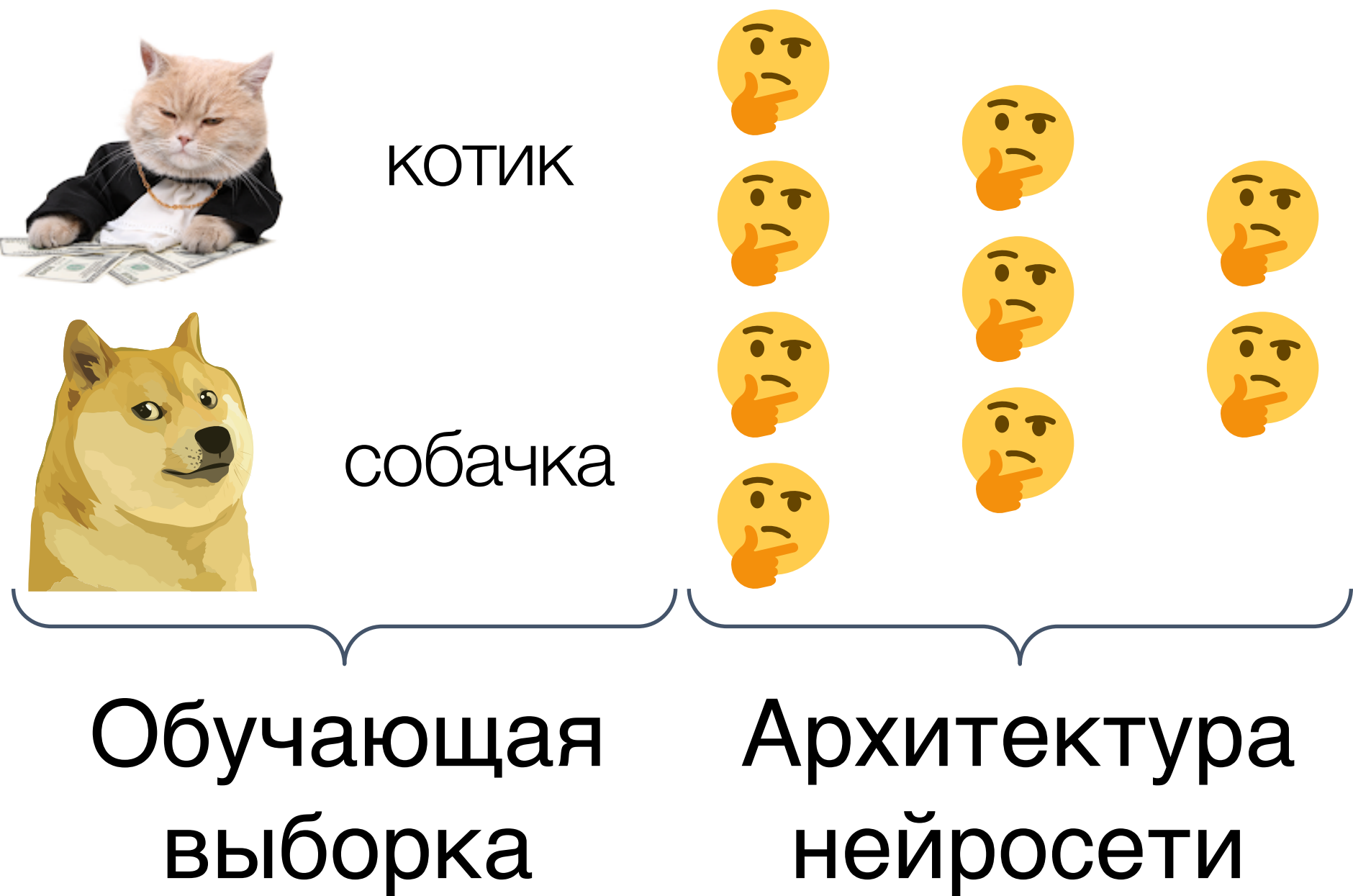
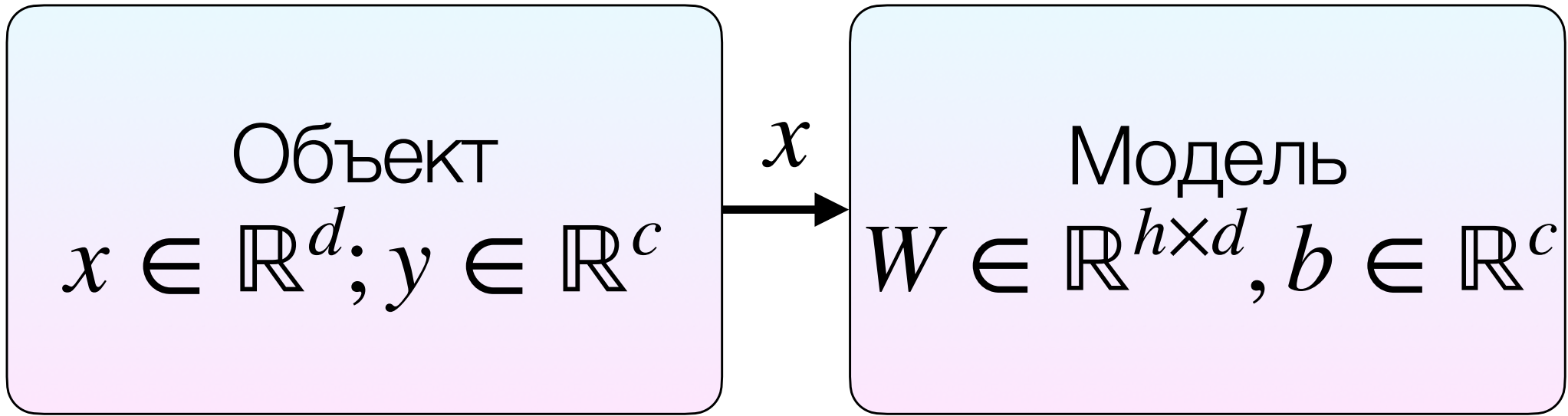


собачка

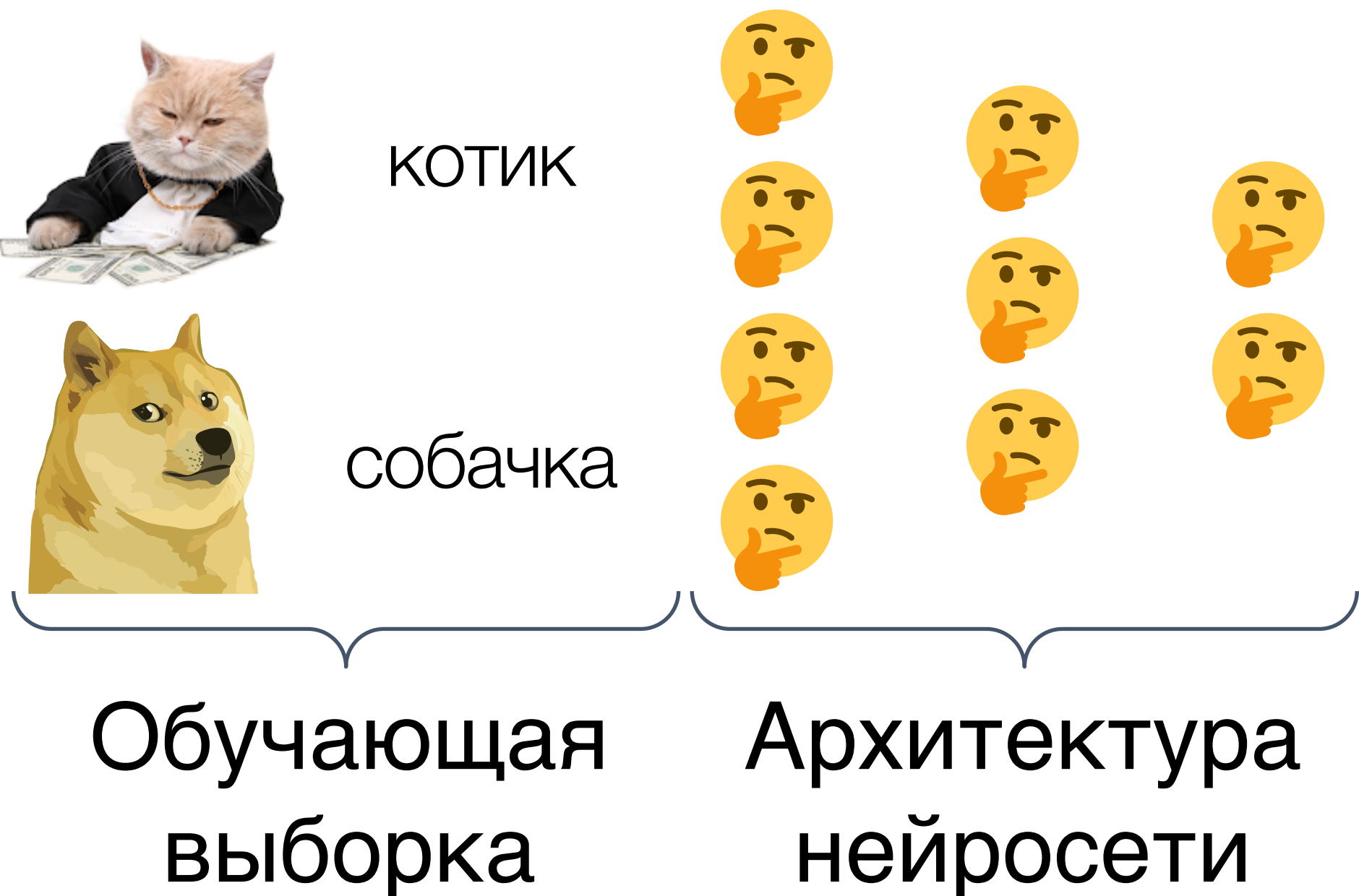
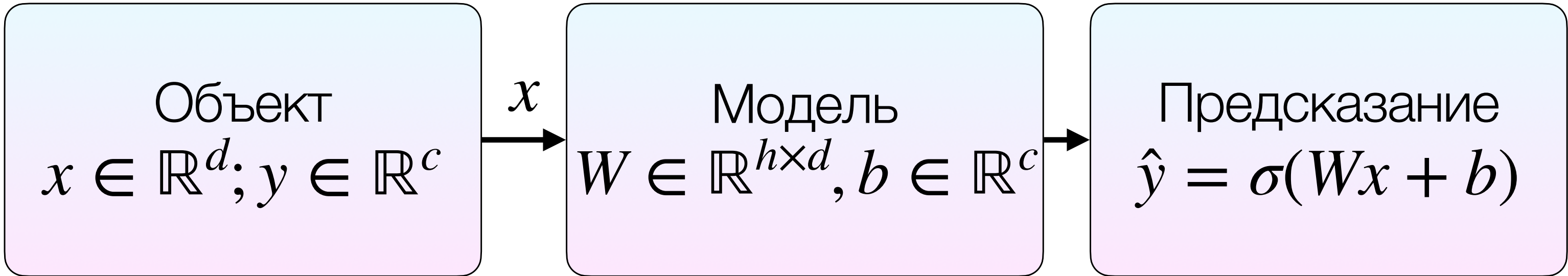
Обучающая  
выборка



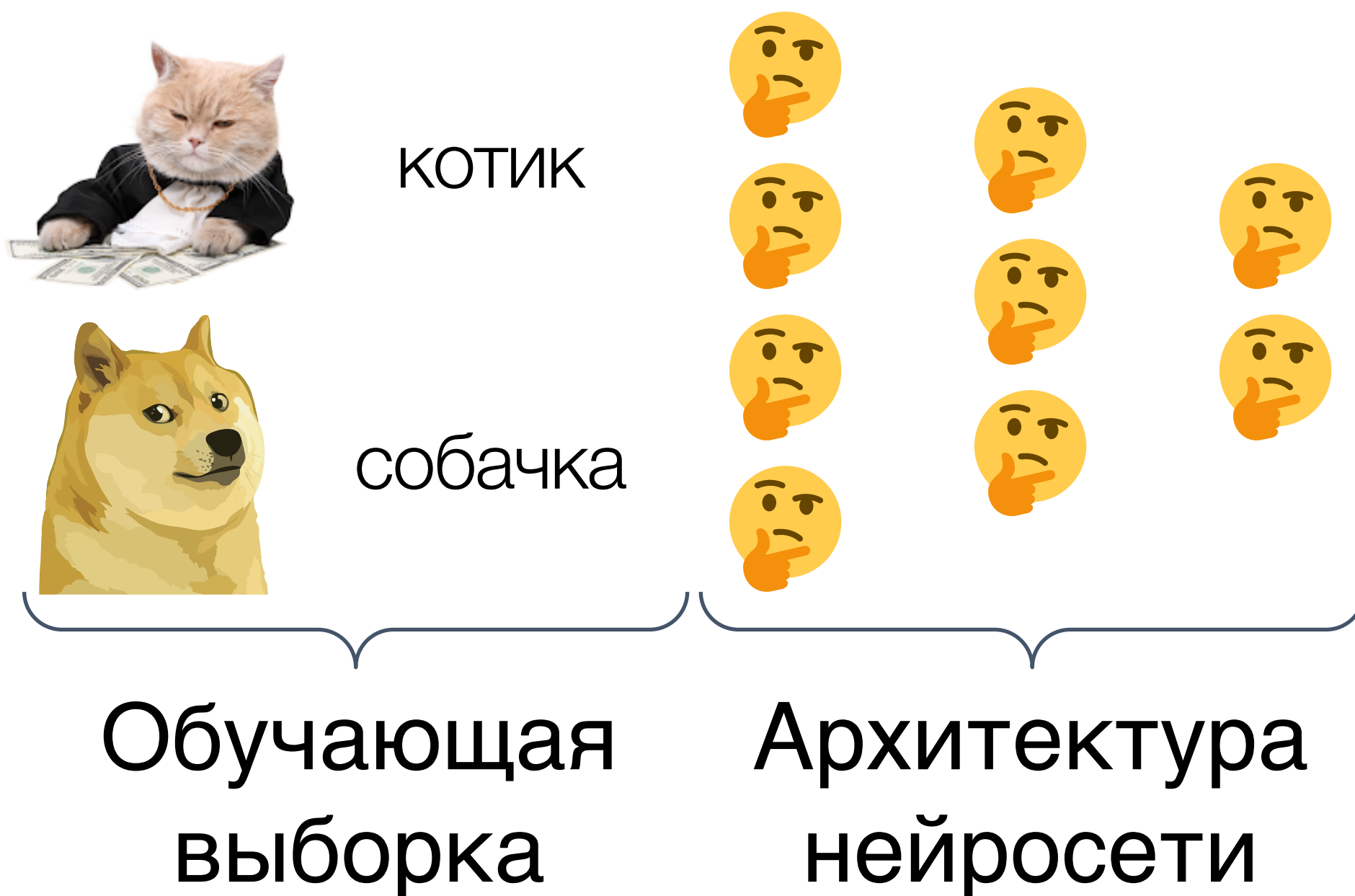
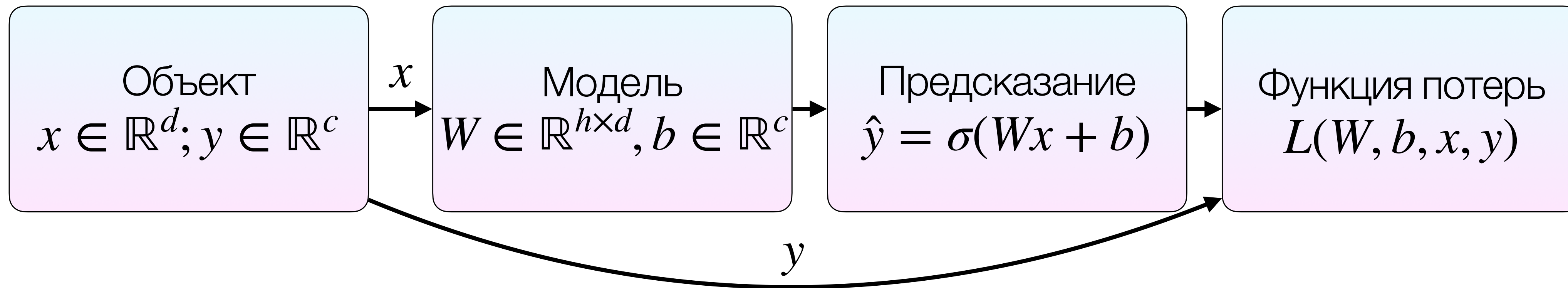
# Обучение нейросети



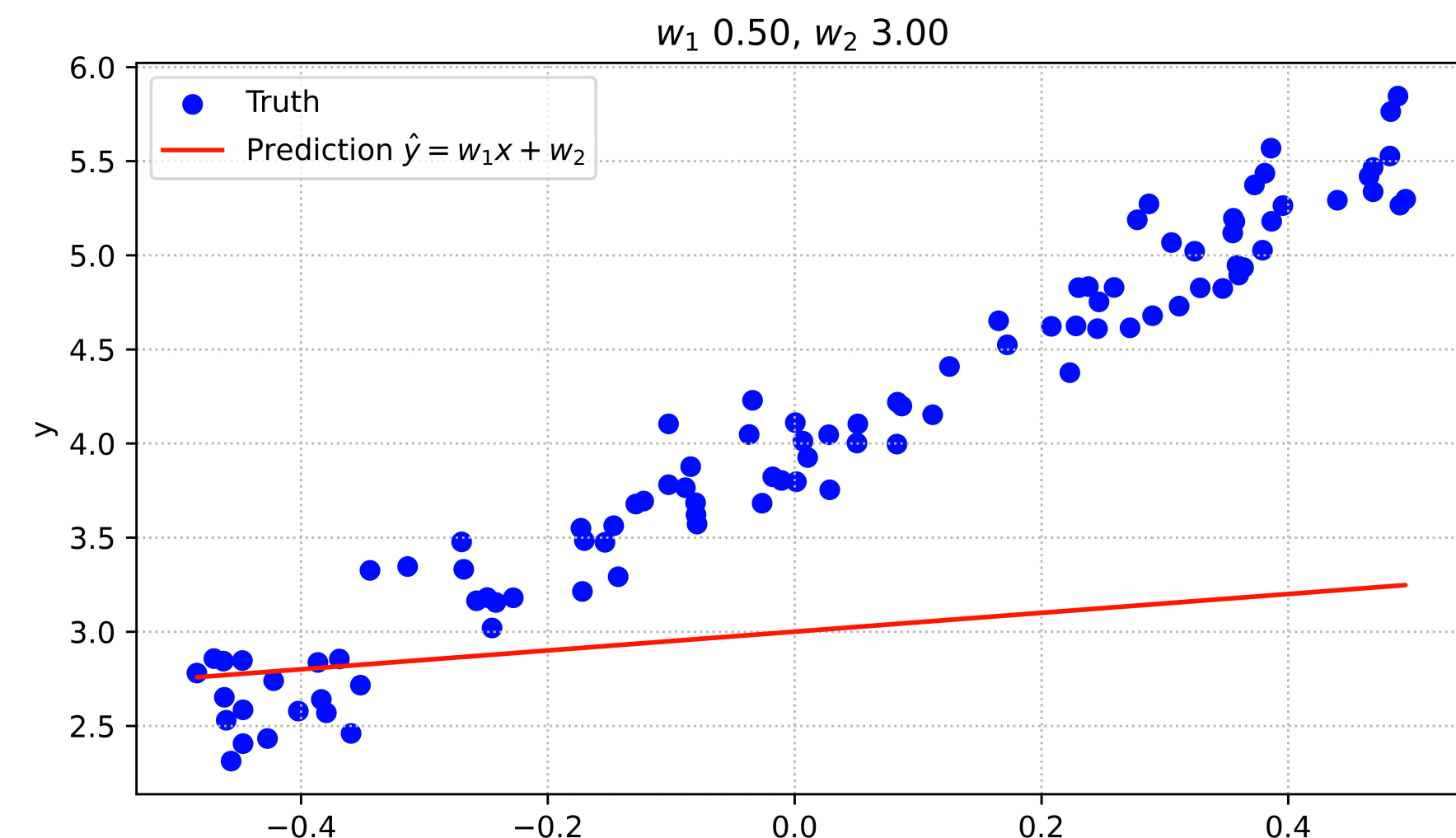
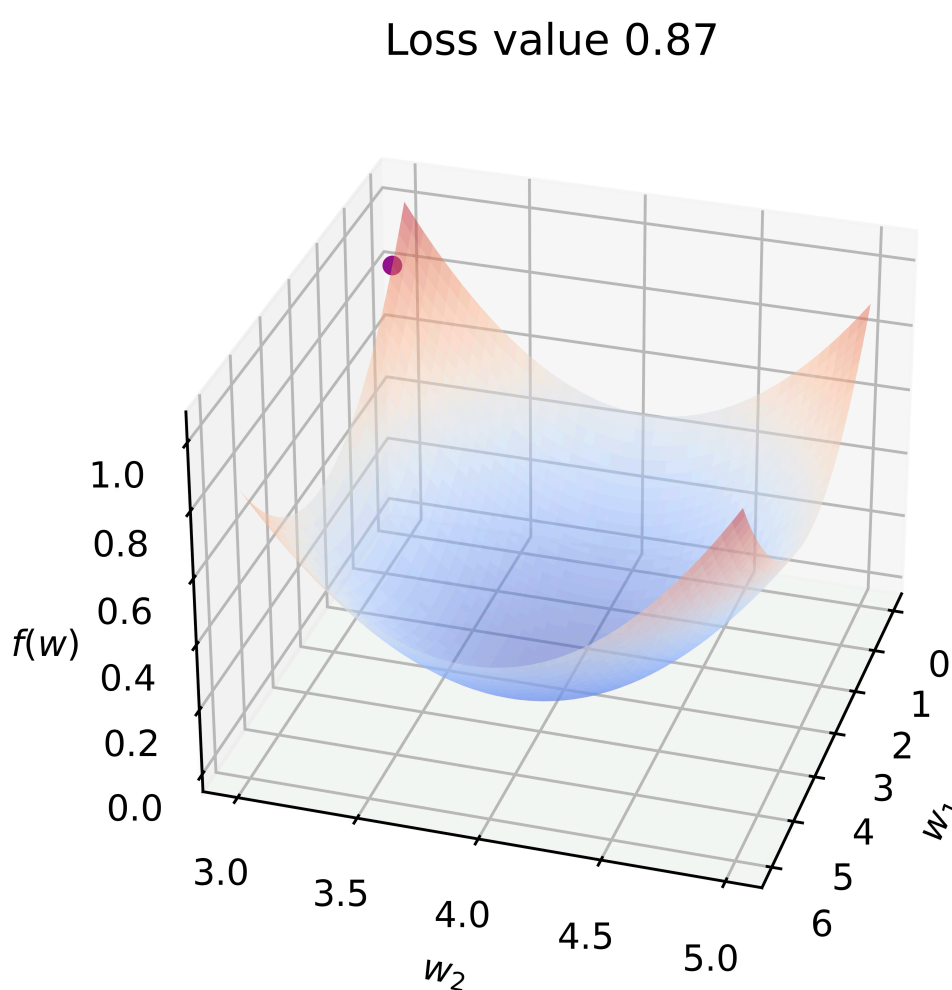
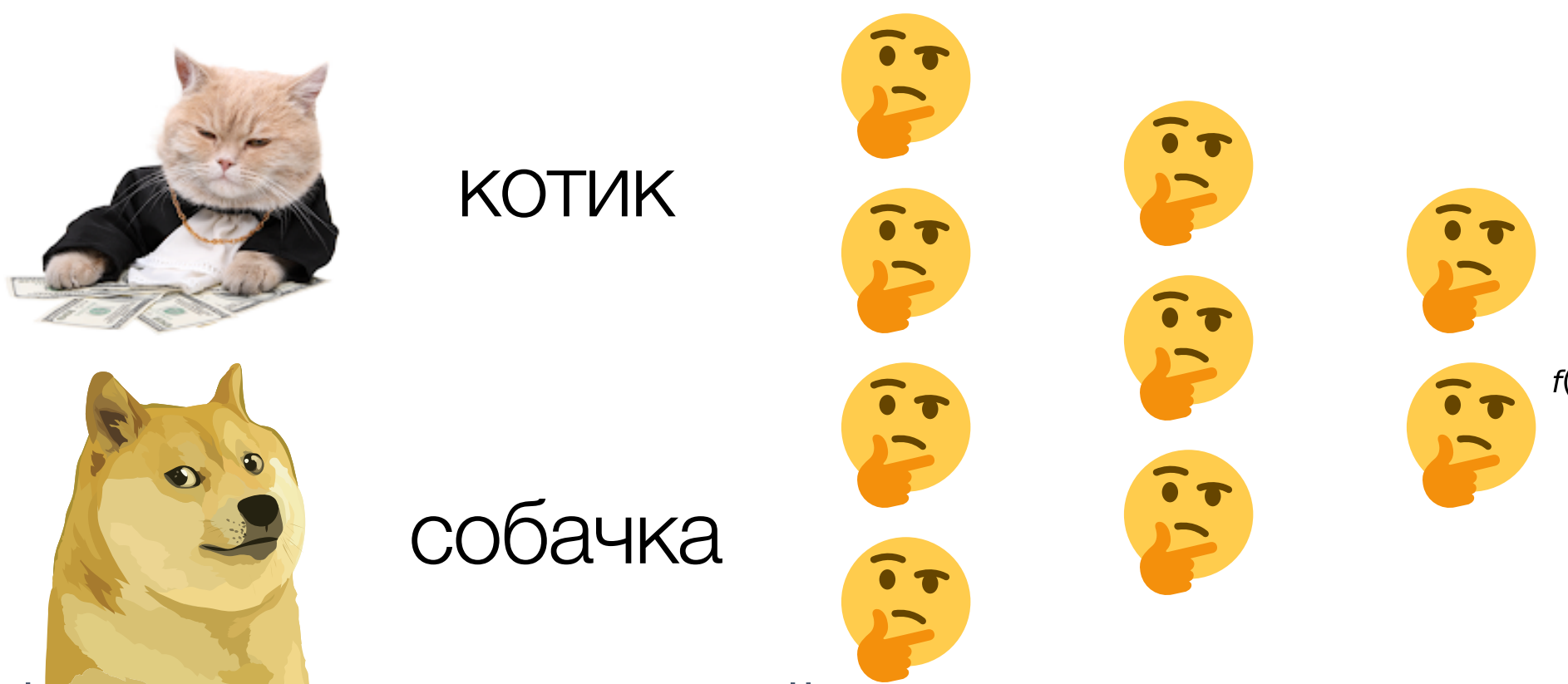
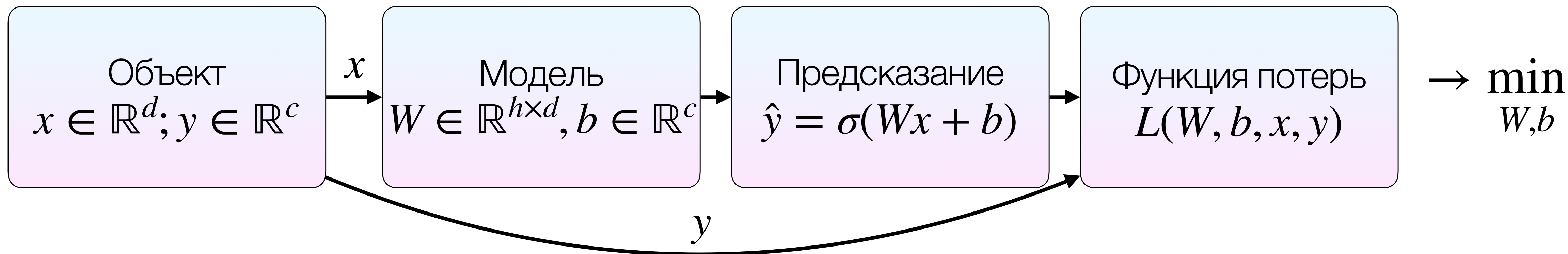
# Обучение нейросети



# Обучение нейросети



# Обучение нейросети



$$W_{k+1} = W_k - \alpha \nabla_W L(W_k)$$

Изменение параметров таким образом, чтобы уменьшать значение функции потерь на обучающей выборке

# Откуда брать данные?



# ImageNet - огромный размеченный датасет

IMAGENET

[www.image-net.org](http://www.image-net.org)

14,197,122 images,  
21841 synsets indexed

Statistics of high level categories

High level category	# synset (subcategories)	Avg # images per synset	Total # images
amphibian	94	591	56K
animal	3822	732	2799K
appliance	51	1164	59K
bird	856	949	812K
covering	946	819	774K
device	2385	675	1610K
fabric	262	690	181K
fish	566	494	280K
flower	462	735	339K
food	1495	670	1001K
fruit	309	607	188K

(all competitions  
on much smaller  
subset  
1000X1000)

# ImageNet

## Flying lemur, flying cat, colugo

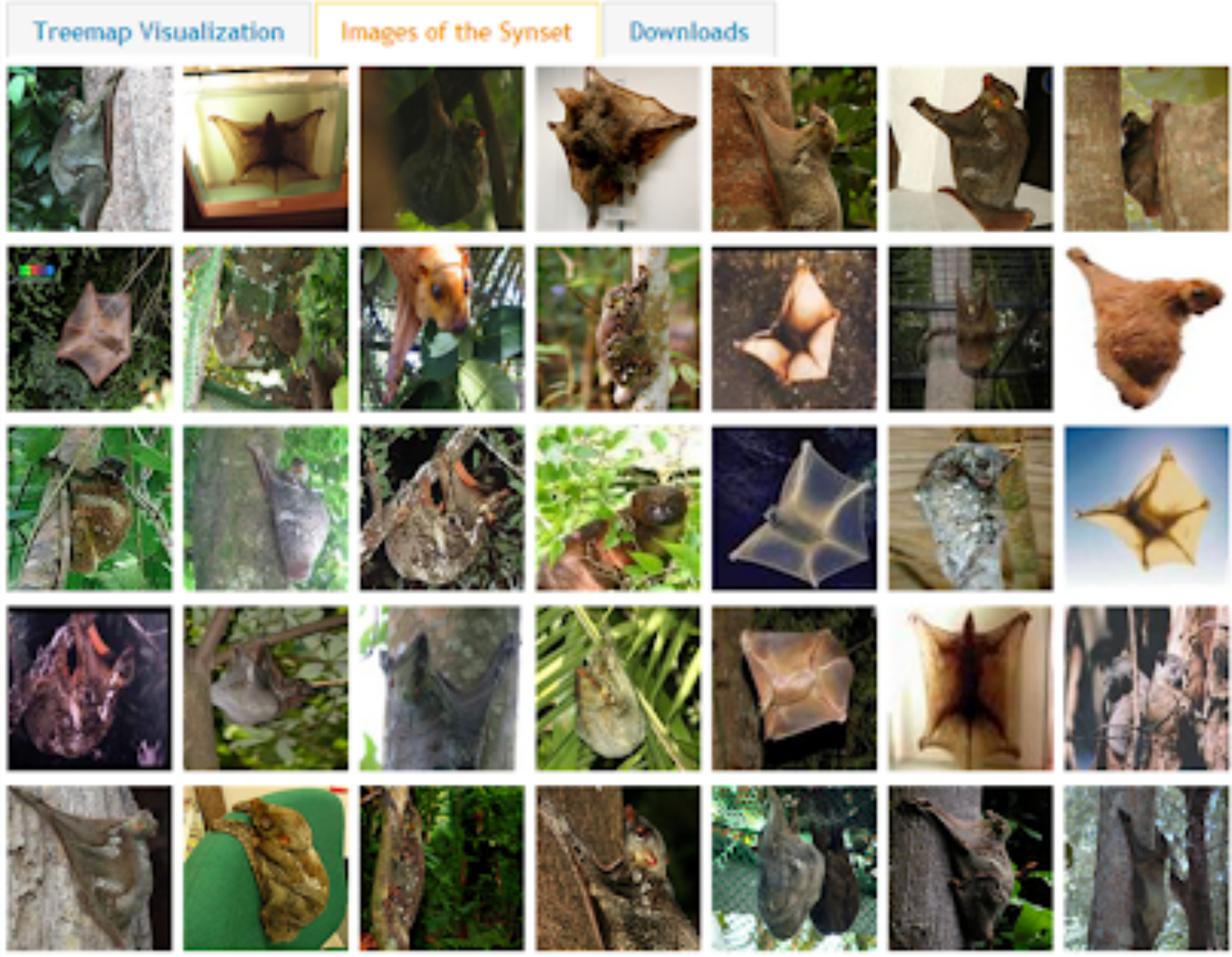
Arboreal nocturnal mammal of southeast Asia and the Philippines resembling a lemur and having a fold of skin on each side from neck to tail that is used for long gliding leaps

45 pictures

61.09% Popularity Percentile



- invertebrate (1)
- predator, predatory animal (1)
- larva (49)
- acrodont (0)
- feeder (0)
- stunt (0)
- chordate (3087)
  - tunicate, urochordate, uroct
  - cephalochordate (1)
  - vertebrate, craniate (3077)
    - mammal, mammalian (3)
      - fossorial mammal (3)
      - placental, placental r
        - lagomorph, gnaw
        - primate (104)
          - flying lemur, flying
            - Cynocephalus
          - tree shrew (1)
        - proboscidean, pr
        - aardvark, ant bea
        - Fissipedia (0)
        - carnivore (365)
        - plantigrade mam
        - unguiculate, ungu
        - aquatic mammal
        - Ungulata (0)
        - Unguiculata (0)
        - digitigrade mamr
        - ungulate, hoofed
        - edentate (21)
        - bat, chiropteran (1)
        - pachyderm (10)
        - pangolin, scaly ar



\*Images of children synsets are not included. All images shown are thumbnails. Images may be subject to copyright.

Prev 1 2 Next

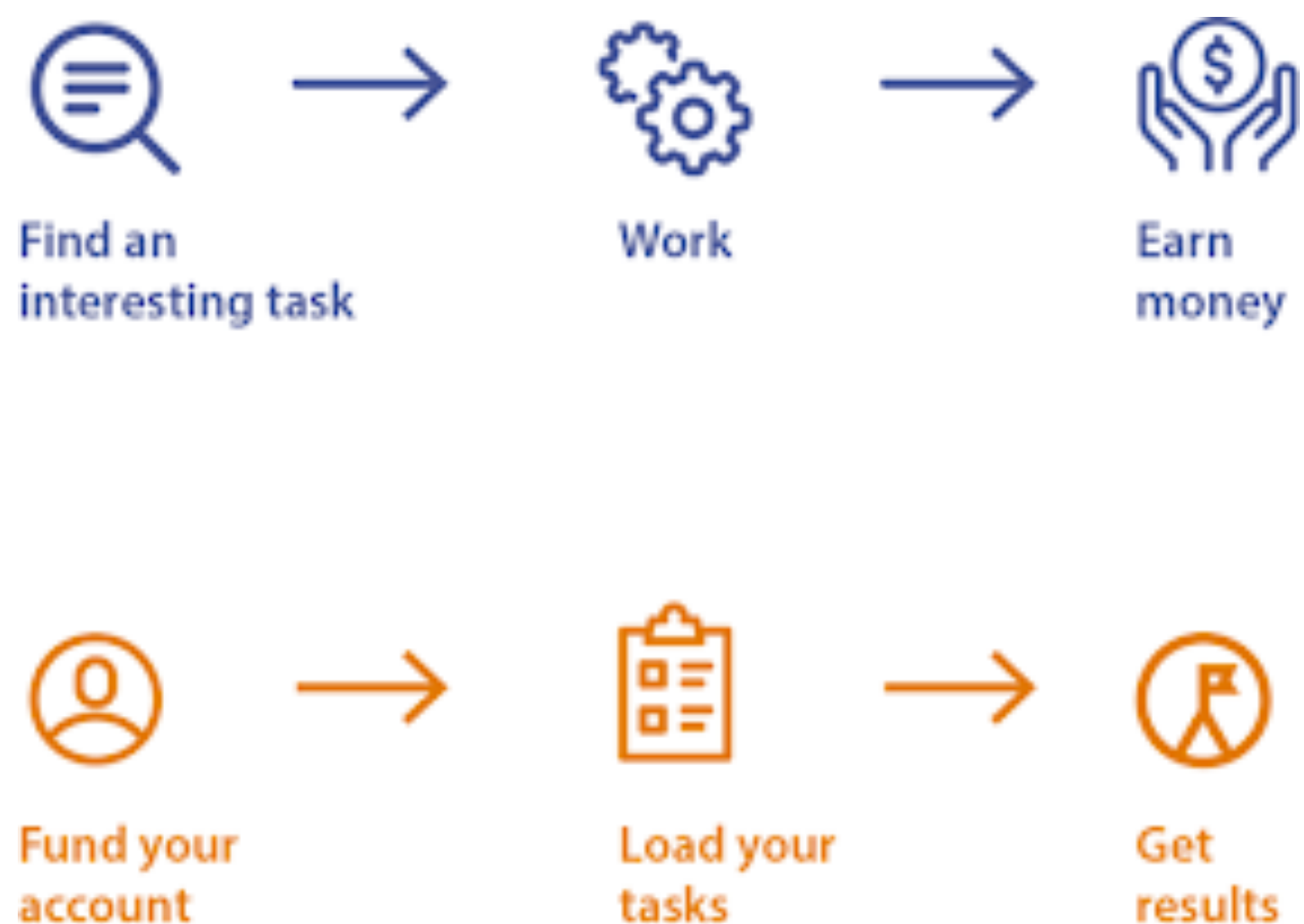
# Краудсорсинг для разметки датасетов

Коллекция ImageNet: <http://image-net.org>

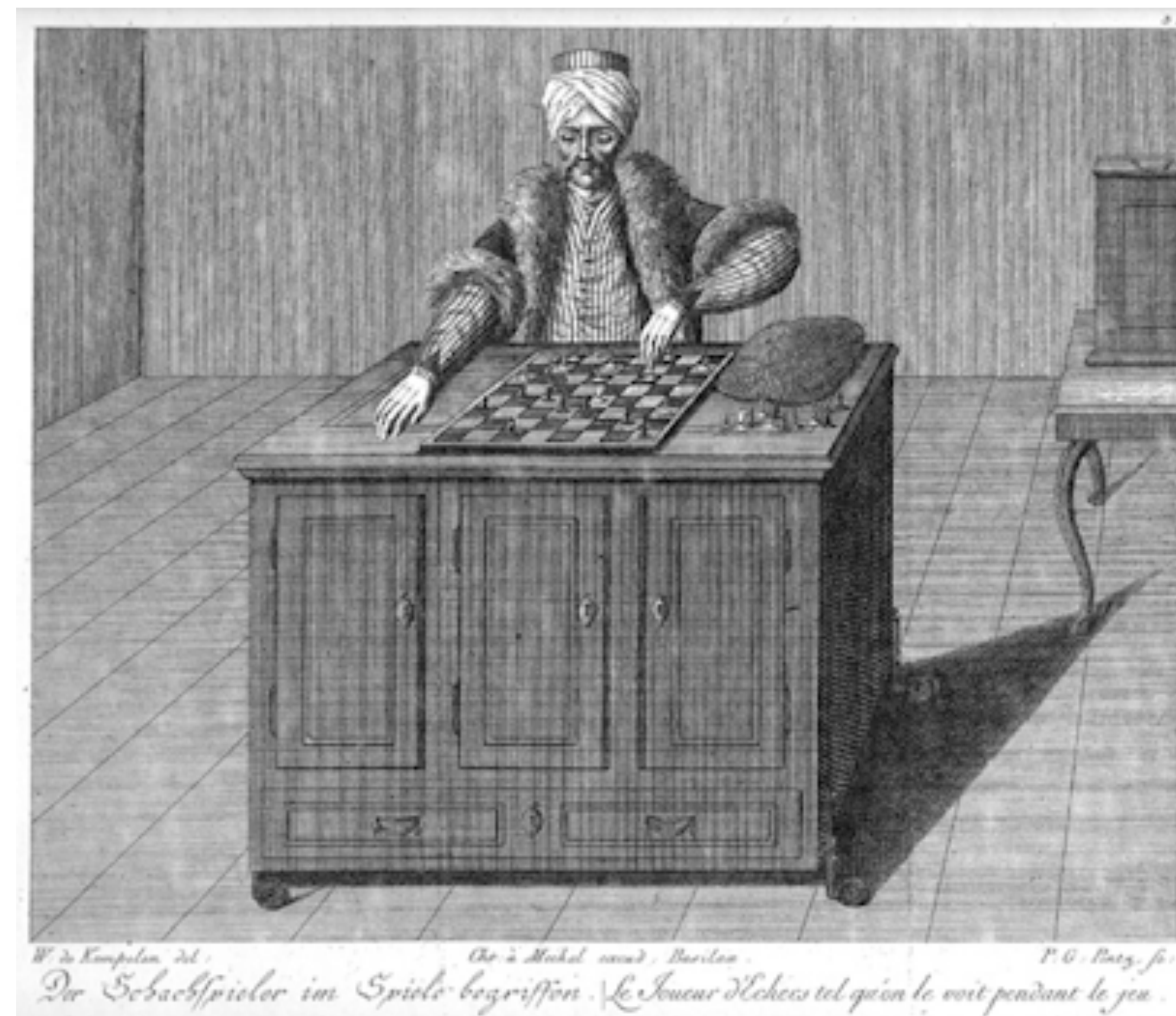
**Цель:** создать коллекцию с 1К изображений на каждый из 117К классов

14,197,122 картинок

из 21,841 синсетов (групп понятий)



- Amazon Mechanical Turk
- Yandex Toloka



Автомат В.Кемпелена (1769 г., Вена). Партию с ним играл Наполеон.



# Данные для GPT-3

Датасет	Количество токенов
<i>Common Crawl (filtered)</i>	410 billion
<i>WebText2</i>	19 billion
<i>Books1</i>	12 billion
<i>Books2</i>	55 billion
<i>Wikipedia</i>	3 billion



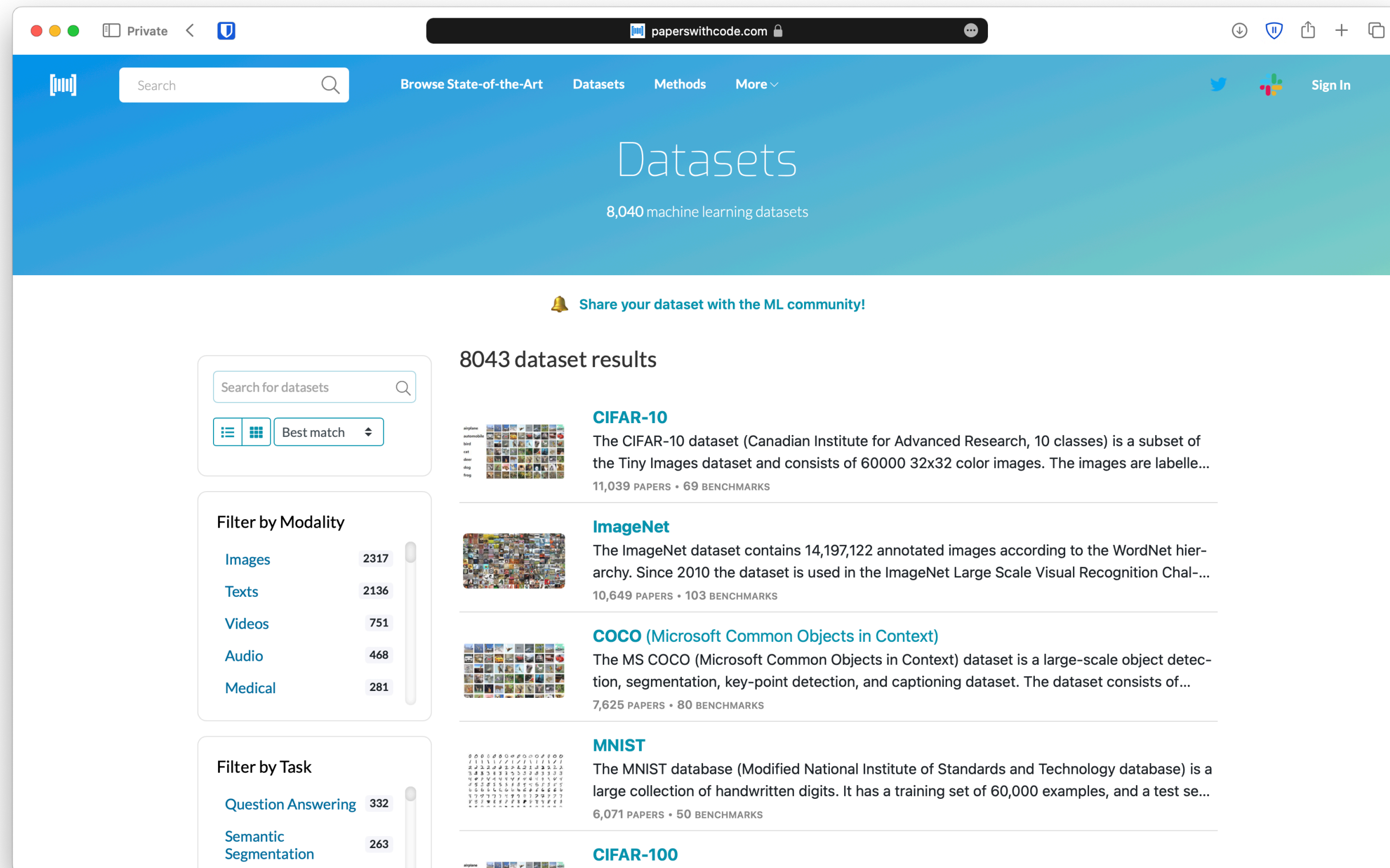
Common Crawl dataset


**WebText2** - это текст веб-страниц из всех исходящих ссылок Reddit из постов с 3+ upvotes. (закрытый датасет OpenAI)

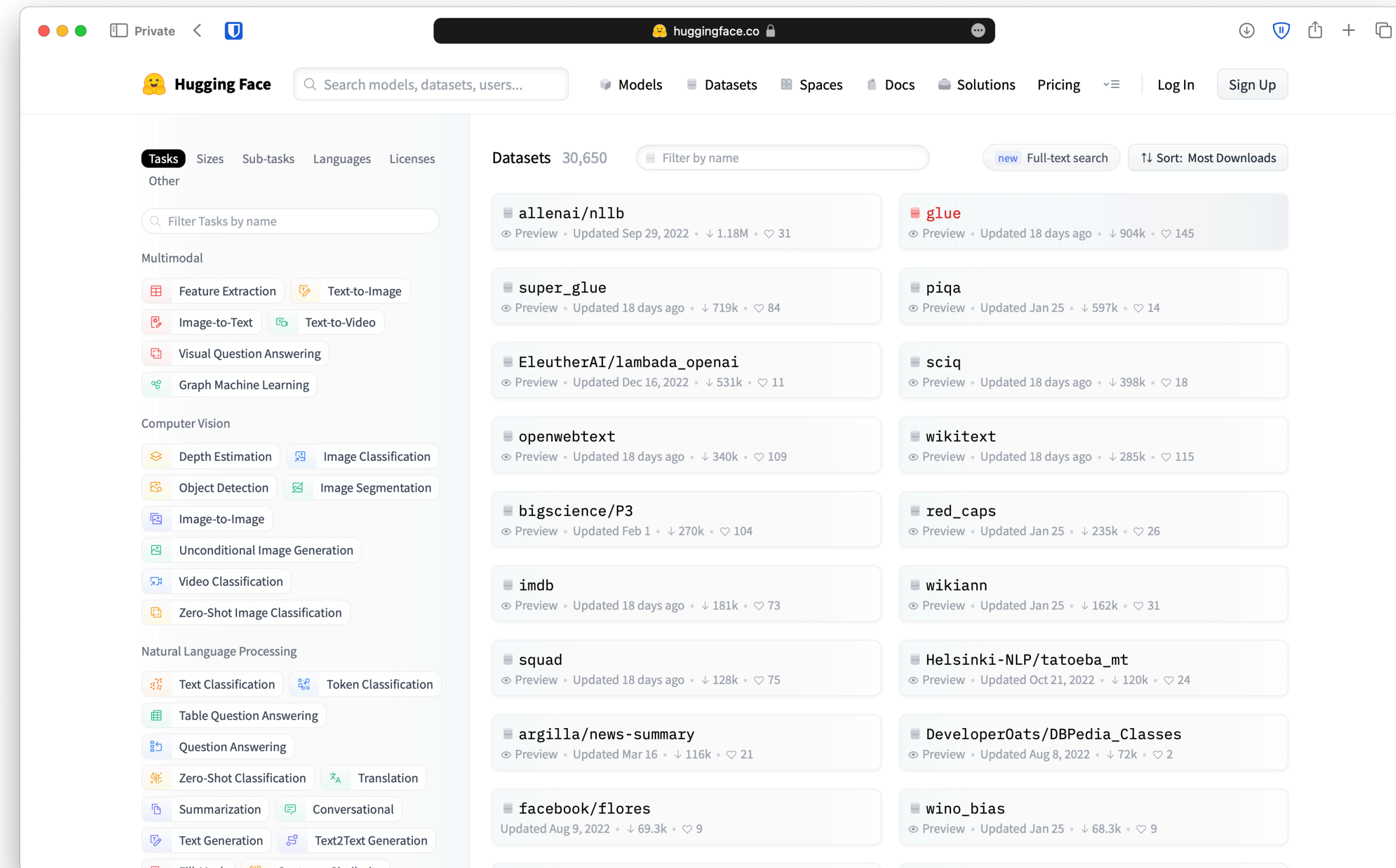
**Books1** и **Books2** – два набора книг, доступных в интернете.


**Wikipedia** - Страницы Википедии на английском языке.

# Поиск данных для вашей задачи

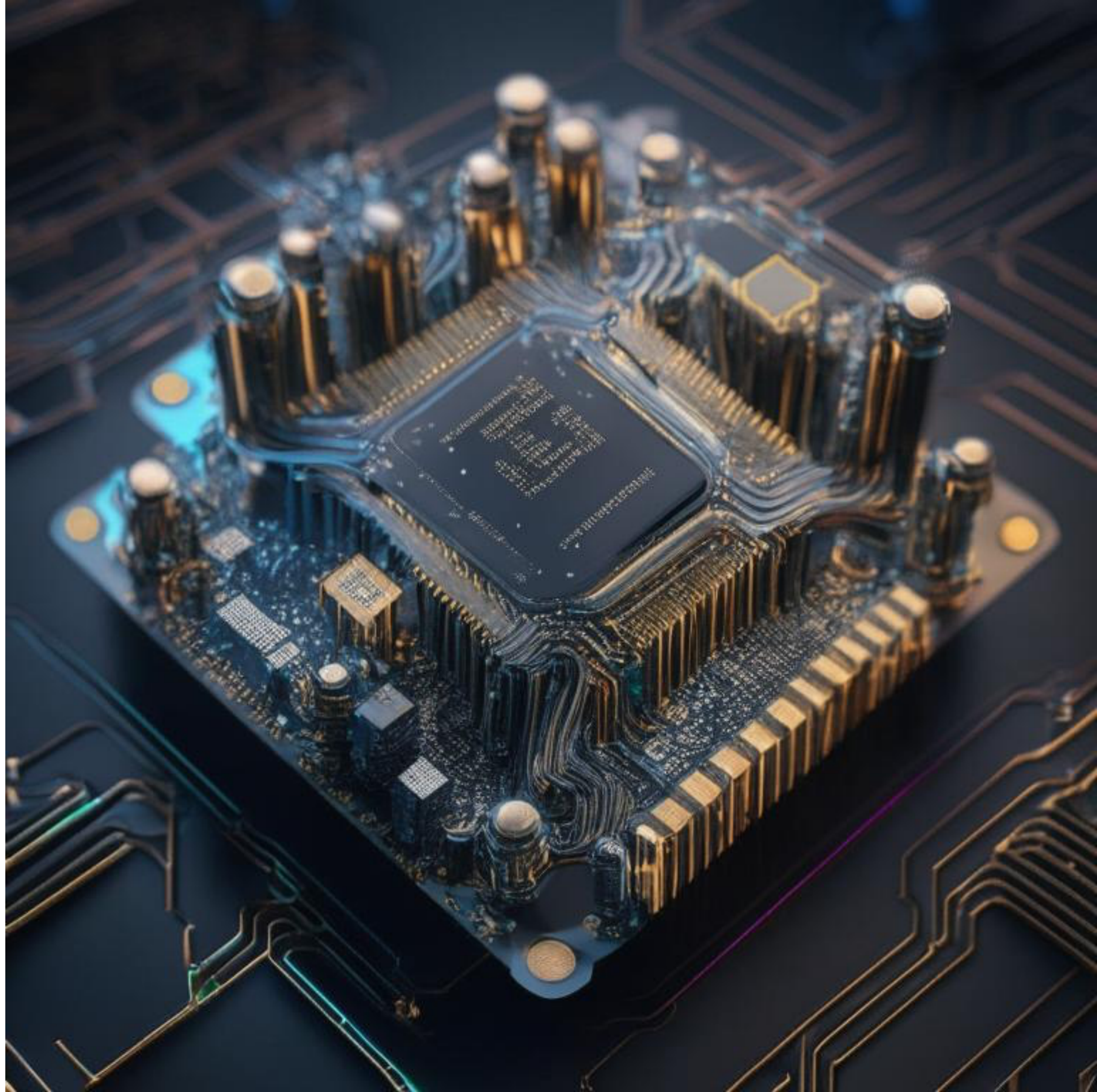


 Сайт  
Papers with code



 Сайт  
Hugging Face

**Железо**



# Обучение NN и параллельные вычисления

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

# Обучение NN и параллельные вычисления

Функция потерь  
(меньше - лучше)

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

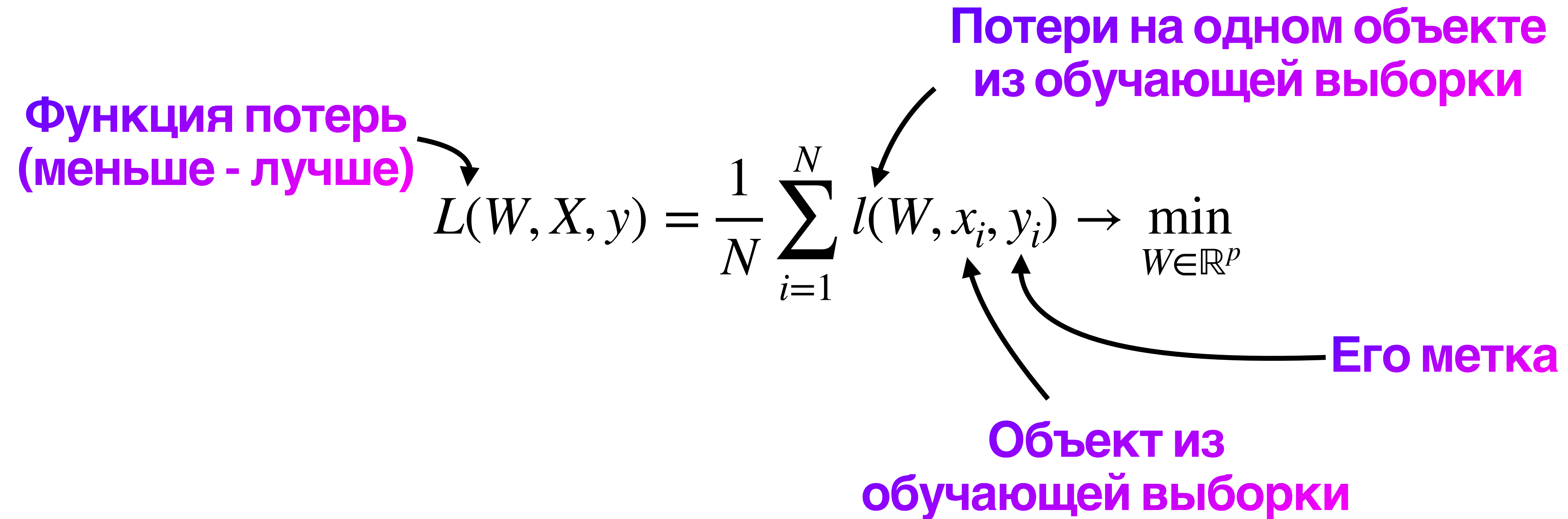
# Обучение NN и параллельные вычисления

Функция потерь  
(меньше - лучше)

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

Потери на одном объекте  
из обучающей выборки

# Обучение NN и параллельные вычисления



# Обучение NN и параллельные вычисления

Размер обучающей выборки

ImageNet  $\approx 1.4 \cdot 10^7$

WikiText  $\approx 10^8$

Потери на одном объекте из обучающей выборки

Функция потерь (меньше - лучше)

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

Его метка

Объект из обучающей выборки



# Обучение NN и параллельные вычисления

Размер обучающей выборки

ImageNet  $\approx 1.4 \cdot 10^7$

WikiText  $\approx 10^8$

Потери на одном объекте из обучающей выборки

Функция потерь (меньше - лучше)

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

Веса модели, которые нужно подобрать  
НО КАК?

Его метка

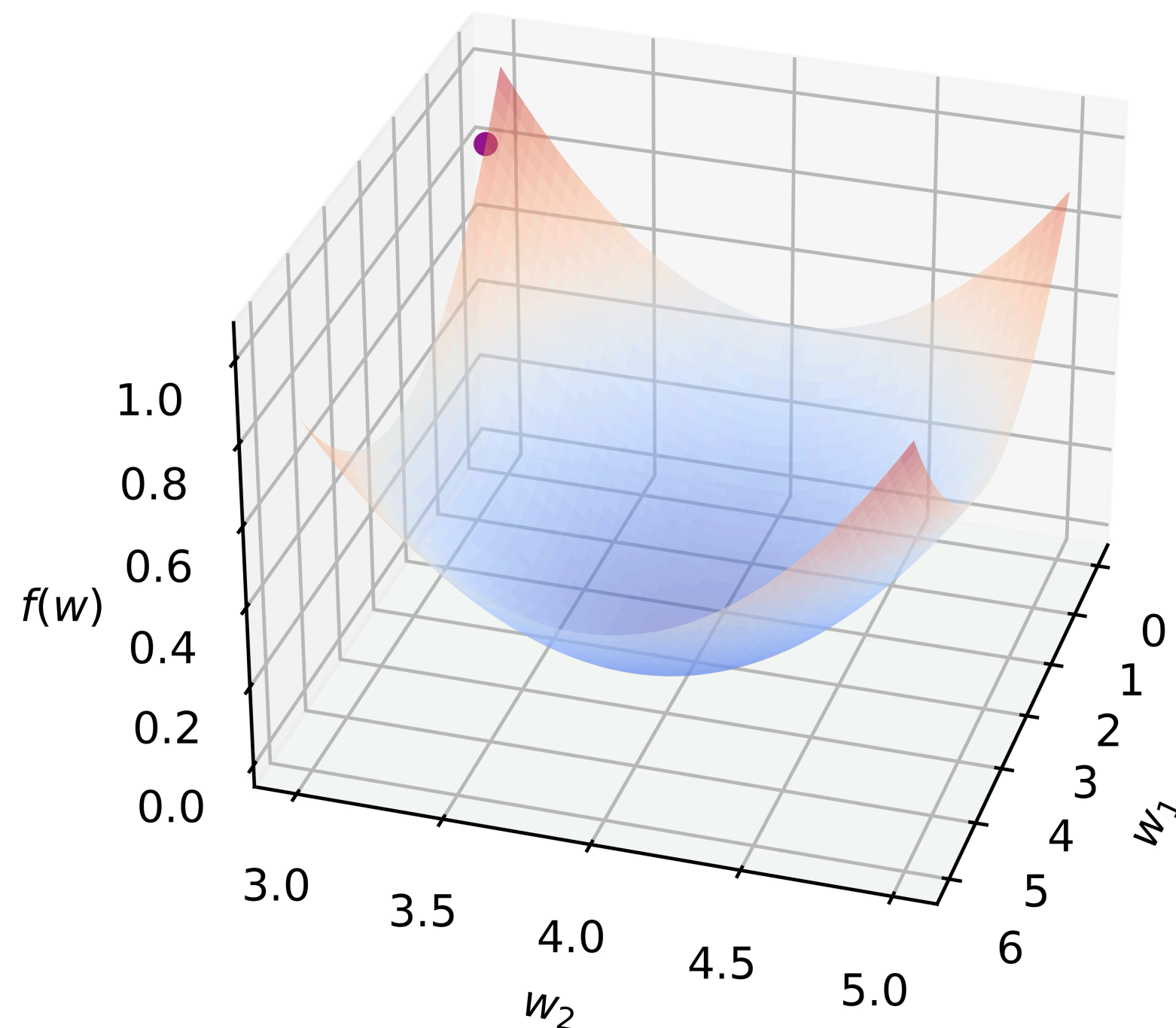
Объект из обучающей выборки

# Обучение NN и параллельные вычисления

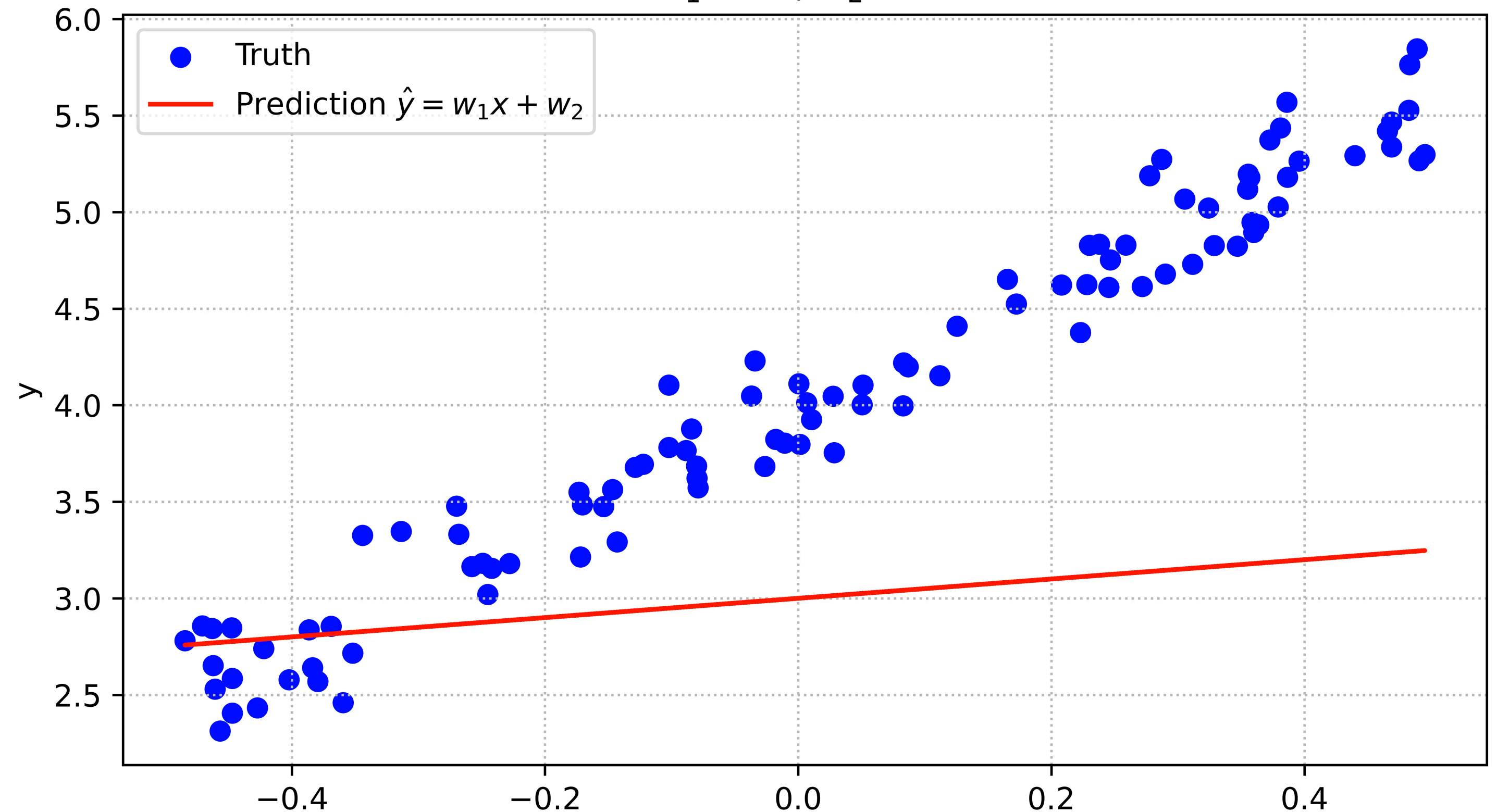
## Метод градиентного спуска (GD)

$$W_{k+1} = W_k - \alpha \nabla_W L(W_k)$$

Loss value 0.87



$w_1$  0.50,  $w_2$  3.00



# Обучение NN и параллельные вычисления

Метод градиентного спуска (GD)

$$W_{k+1} = W_k - \alpha \nabla_W L(W_k)$$

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

# Обучение NN и параллельные вычисления

Метод градиентного спуска (GD)

$$W_{k+1} = W_k - \alpha \nabla_W L(W_k)$$

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

$$\nabla_W L(W_k) = \frac{1}{N} \sum_{i=1}^N \nabla_W l(W_k, x_i, y_i)$$

# Обучение NN и параллельные вычисления

Метод градиентного спуска (GD)

$$W_{k+1} = W_k - \alpha \nabla_W L(W_k)$$

$$L(W, X, y) = \frac{1}{N} \sum_{i=1}^N l(W, x_i, y_i) \rightarrow \min_{W \in \mathbb{R}^p}$$

$$\nabla_W L(W_k) = \frac{1}{N} \sum_{i=1}^N \nabla_W l(W_k, x_i, y_i)$$

Тяжело считать при большом N 🤔

# Обучение NN и параллельные вычисления

Метод стохастического  
градиентного спуска (SGD)

$$W_{k+1} = W_k - \alpha g_k$$

$$\nabla_W L(W_k) = \frac{1}{N} \sum_{i=1}^N \nabla_W l(W_k, x_i, y_i)$$

$$\nabla_W L(W_k) = \frac{1}{b} \sum_{i=1}^b \nabla_W l(W_k, x_{j_i}, y_{j_i})$$

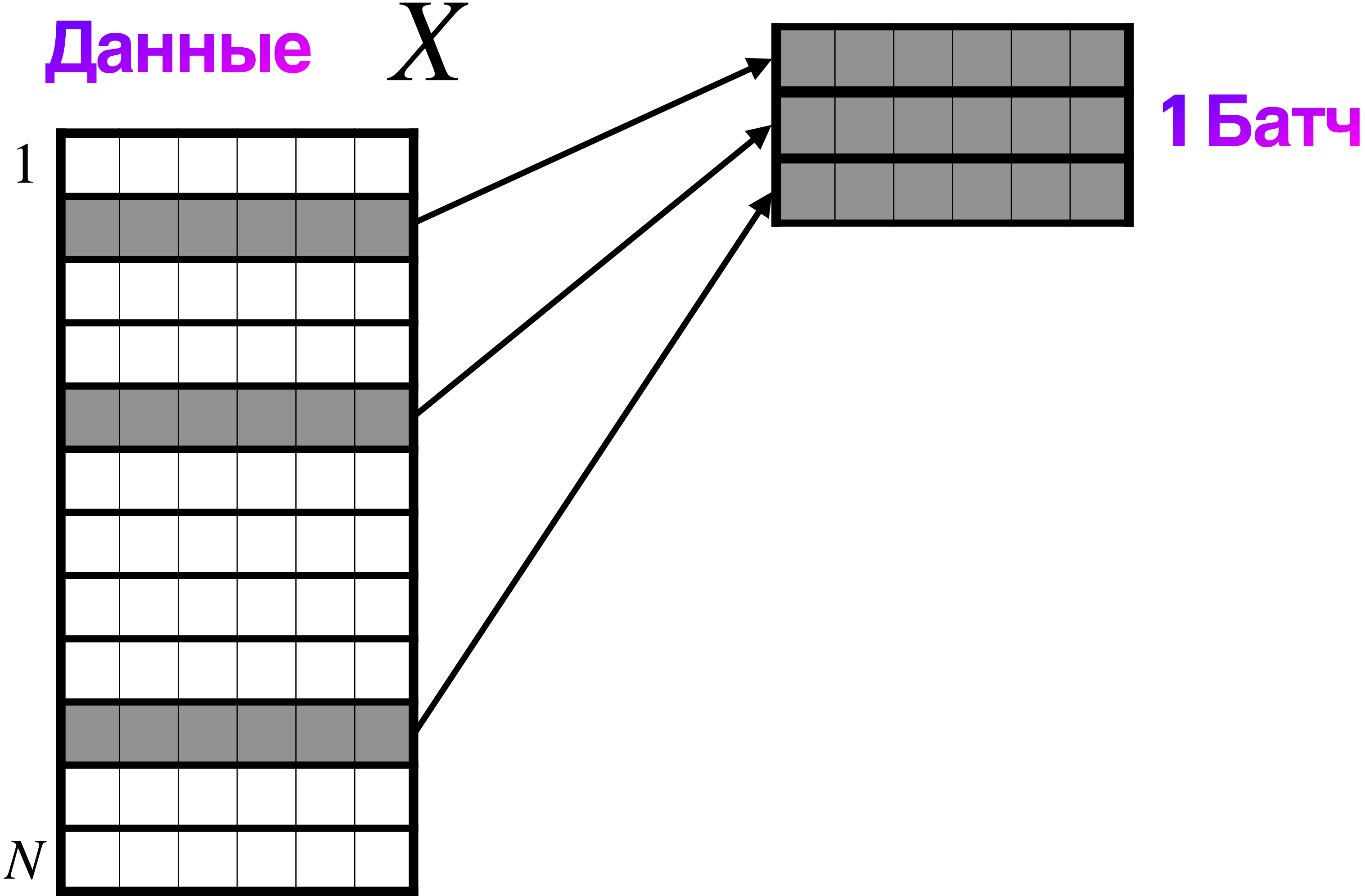
$$b \ll N$$

# Обучение NN и параллельные вычисления

Данные  $X$

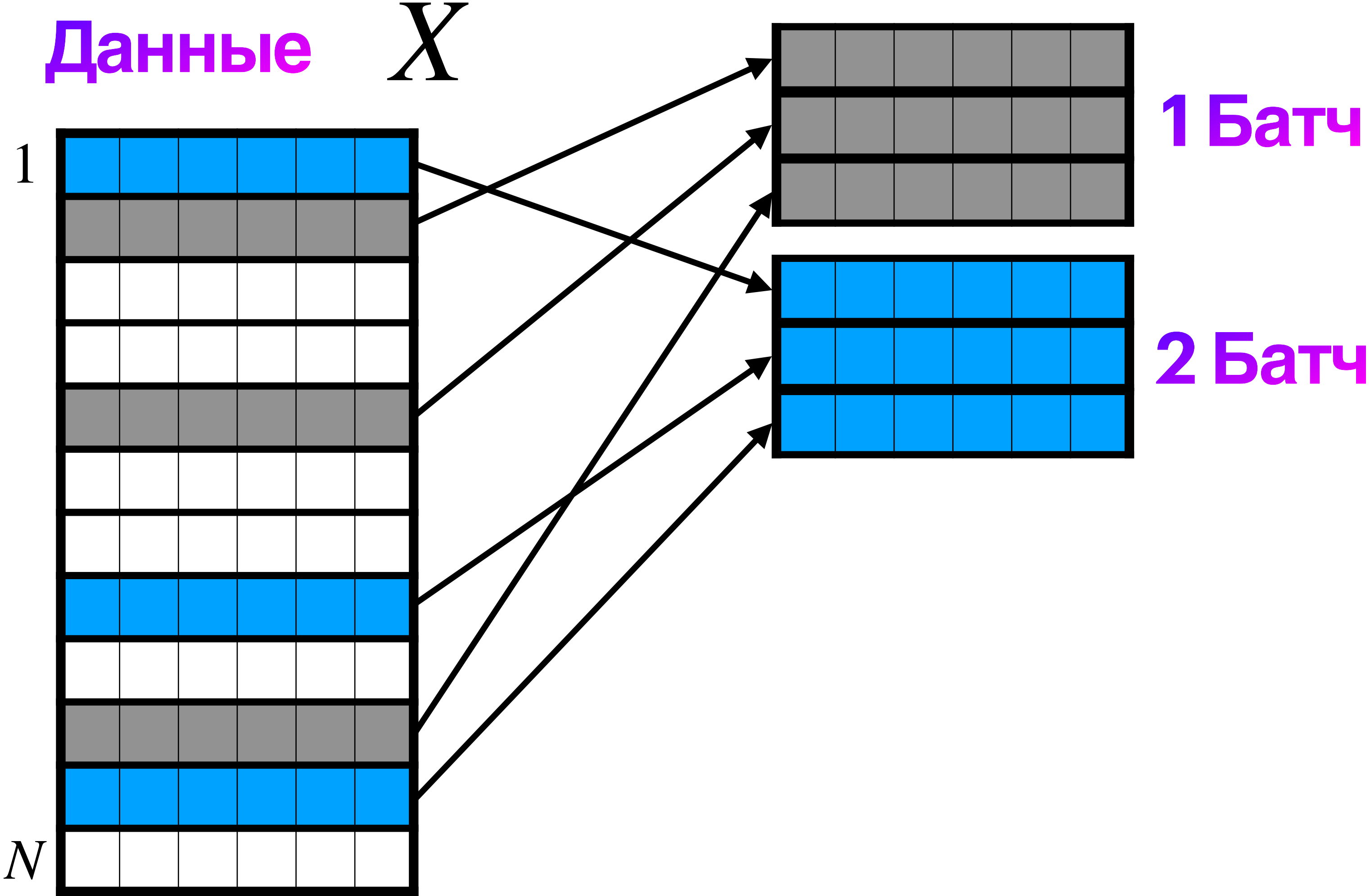
1						
$N$						

# Обучение NN и параллельные вычисления

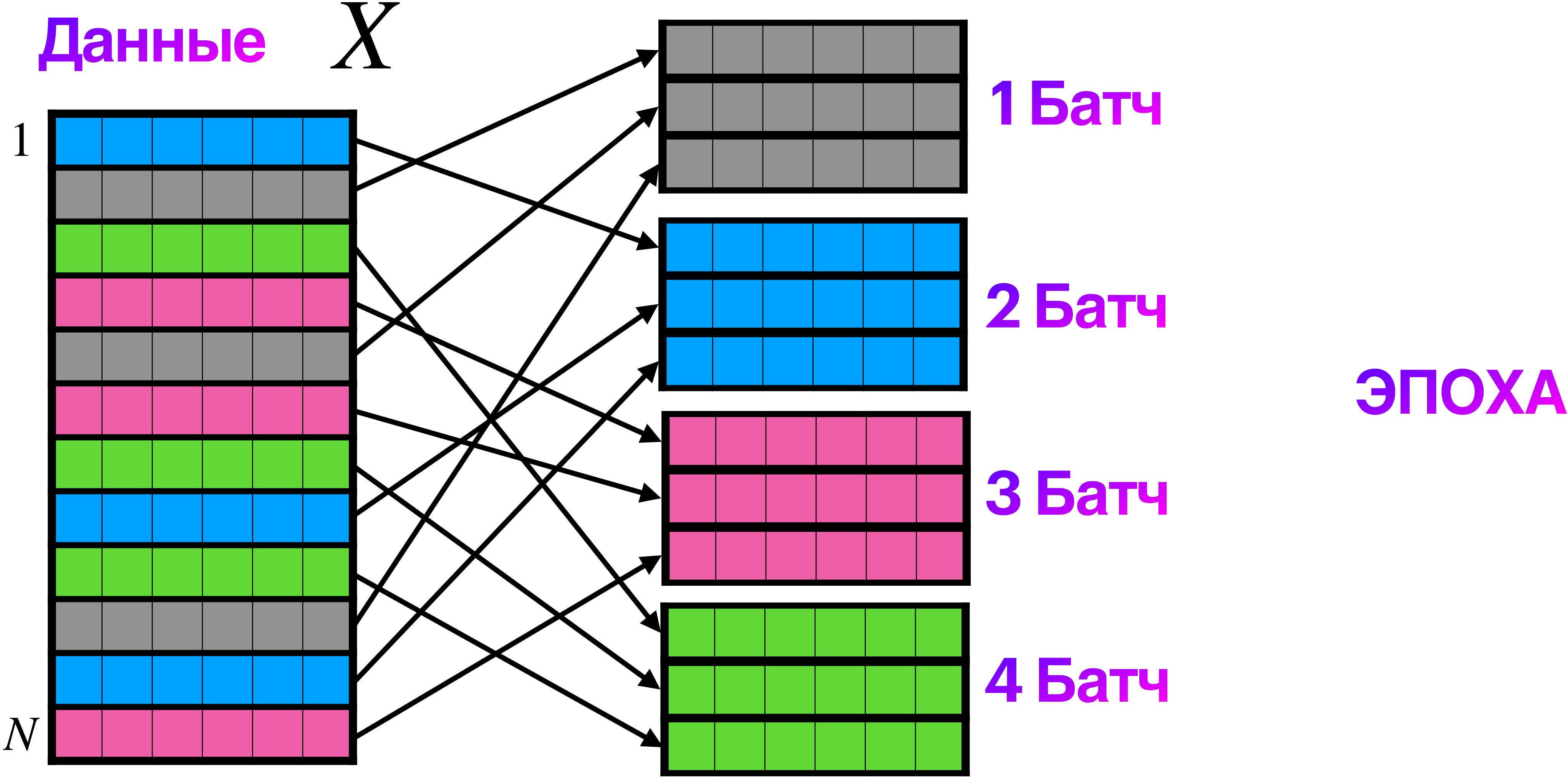




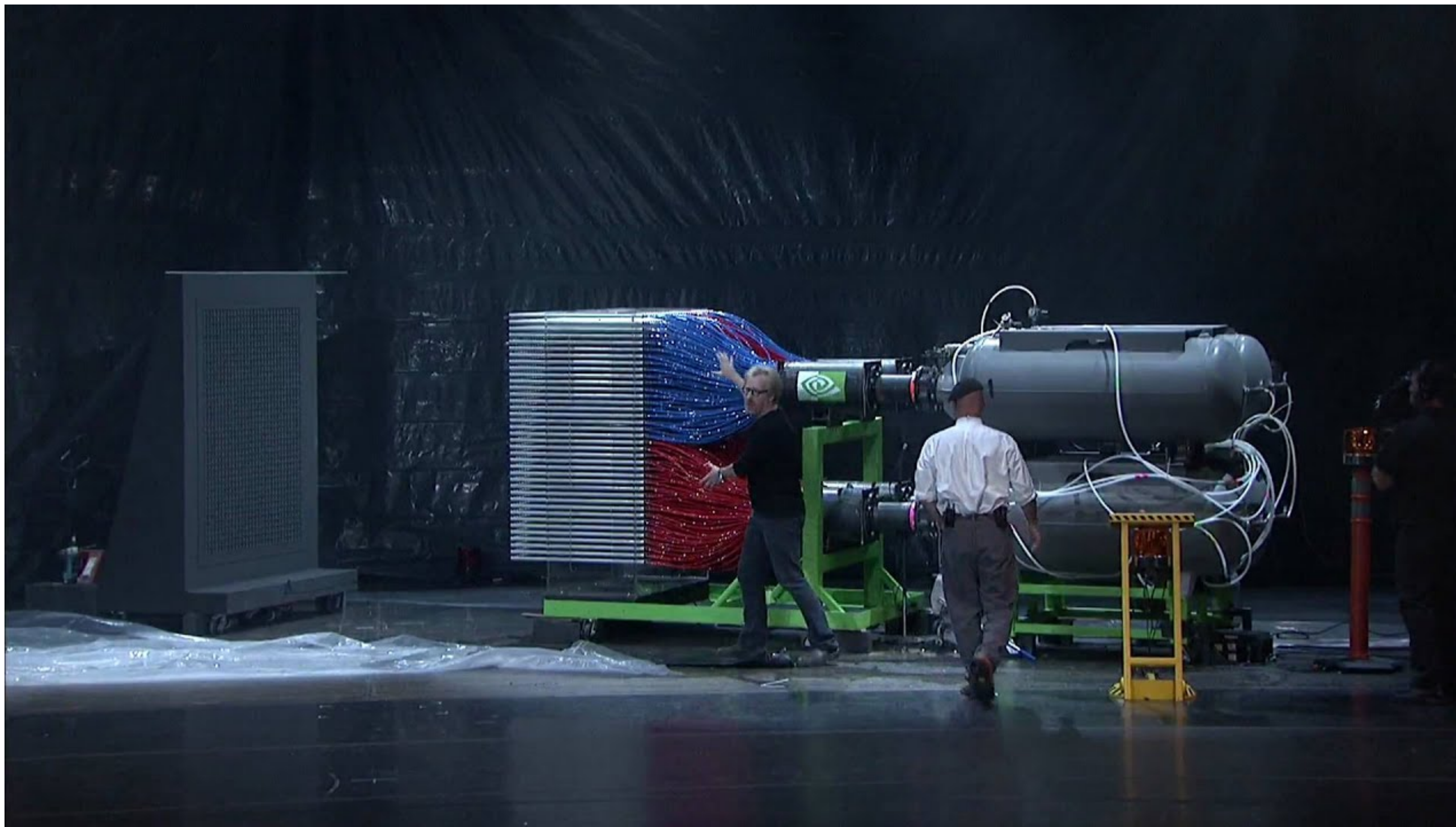
# Обучение NN и параллельные вычисления



# Обучение NN и параллельные вычисления



# Почему для обучения NN нужны GPU?



# Свое железо для обучения - дорого

	ARCH	VRAM	FP32 TFLOPS	ENERGY	ENPRICE	PPRICE
RTX 2080 Ti	Turing	11	13.5	250	500 €	450 €
RTX 2080 Super	Turing	8	11.2	215	430 €	350 €
RTX 2070 Super	Turing	8	9.1	175	350 €	300 €
RTX 4090	Ada Lovelace	24	82.6	450	900 €	1,900 €
RTX 4070 Ti	Ada Lovelace	12	40.1	285	570 €	1,000 €
RTX 3070 Ti	Ampere	8	21.75	290	580 €	600 €
RTX 4080	Ada Lovelace	16	48.7	320	640 €	1,300 €
RTX 3080 Ti	Ampere	12	34.1	350	700 €	1,100 €
RTX 3090 Ti	Ampere	24	40	450	900 €	1,500 €
RTX A4000	Ampere	16	19.1	140	280 €	1,000 €
RTX 4000 Ada	Ada Lovelace	20	19.2	70	140 €	1,250 €
RTX A5000	Ampere	24	27.8	230	460 €	2,500 €
RTX 6000 Ada	Ada Lovelace	48	91.1	300	600 €	7,000 €
L40	Ada Lovelace	48	90.5	300	600 €	9,000 €
RTX A6000	Ampere	48	38.7	300	600 €	6,000 €
A100	Ampere	40	19.5	250	500 €	11,000 €
H100	Hopper	80	24.1	350	700 €	35,000 €

ENPRICE - оценка стоимости работы GPU на 100% в течение года

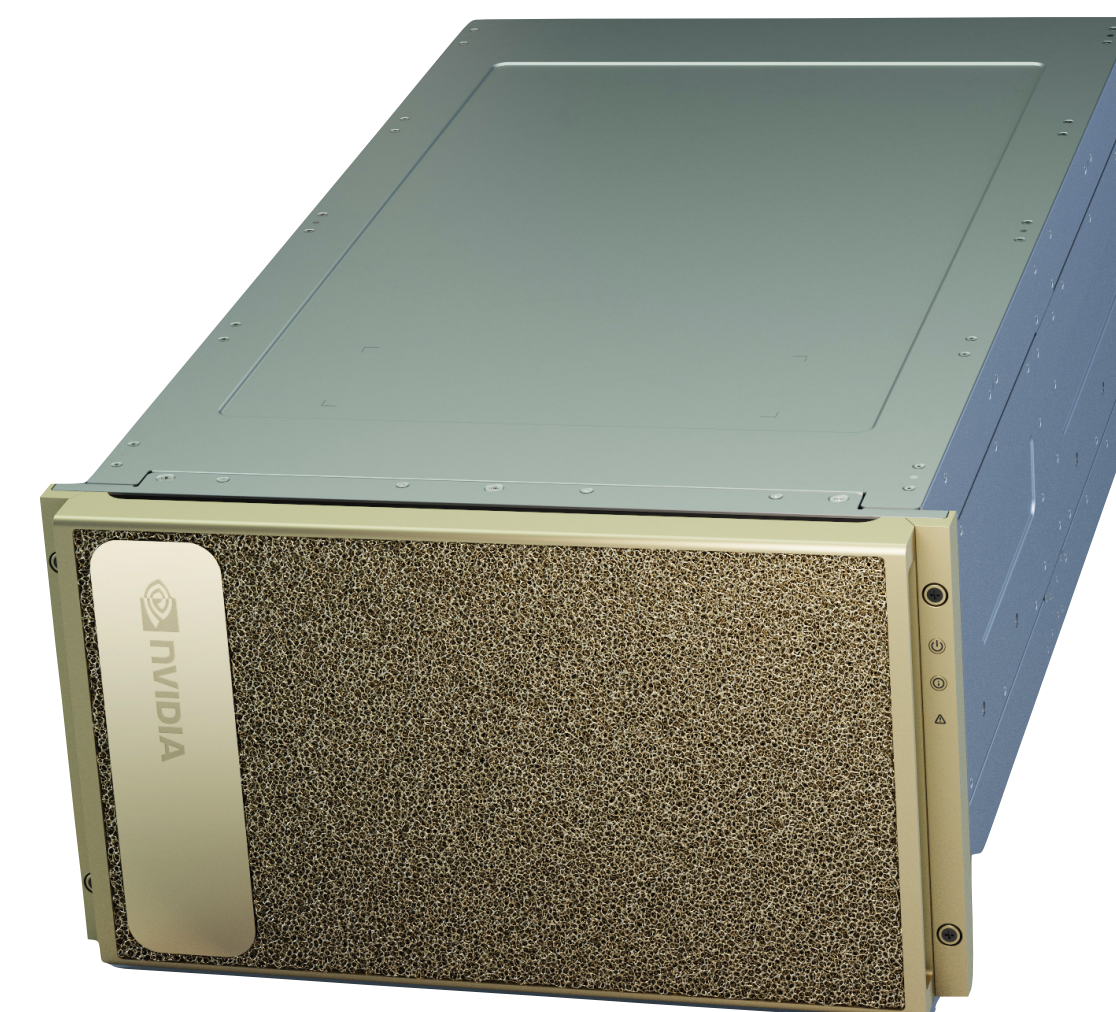


Сайт

Таблица ценами на GPU и сравнением

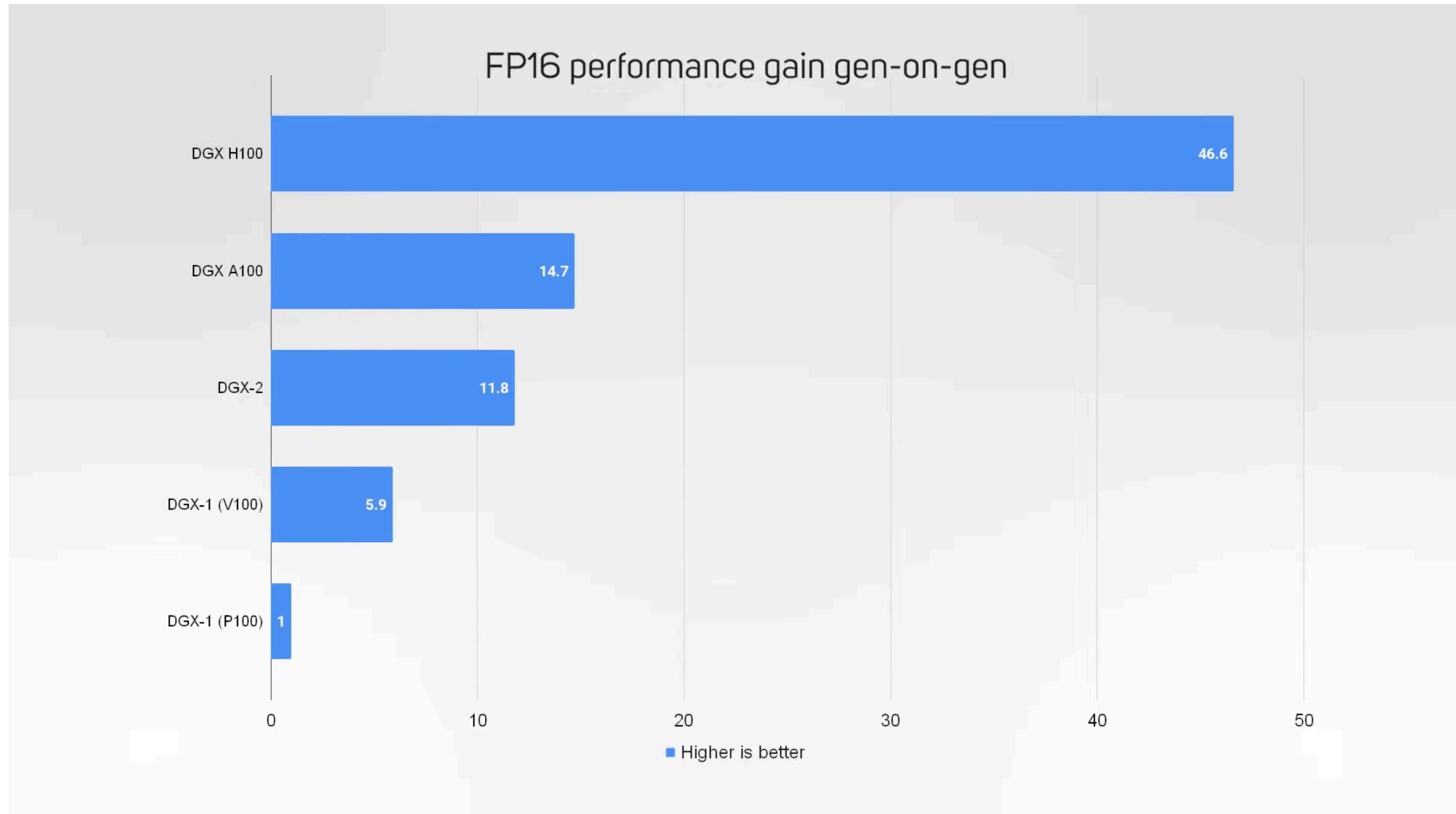
# Кластеры NVIDIA

	Дата выхода	Кол-во GPU	RAM, Gb	VRAM, Gb	Цена на запуске, тыс.\$
<b>DGX-1</b>	2016, Apr	8	512	128	129-149
<b>DGX-2</b>	2018, Mar	16	1536	512	399
<b>DGX A100</b>	2020, May	8	1024	320/640	199
<b>DGX H100</b>	2022, Mar	8	2048	640	400

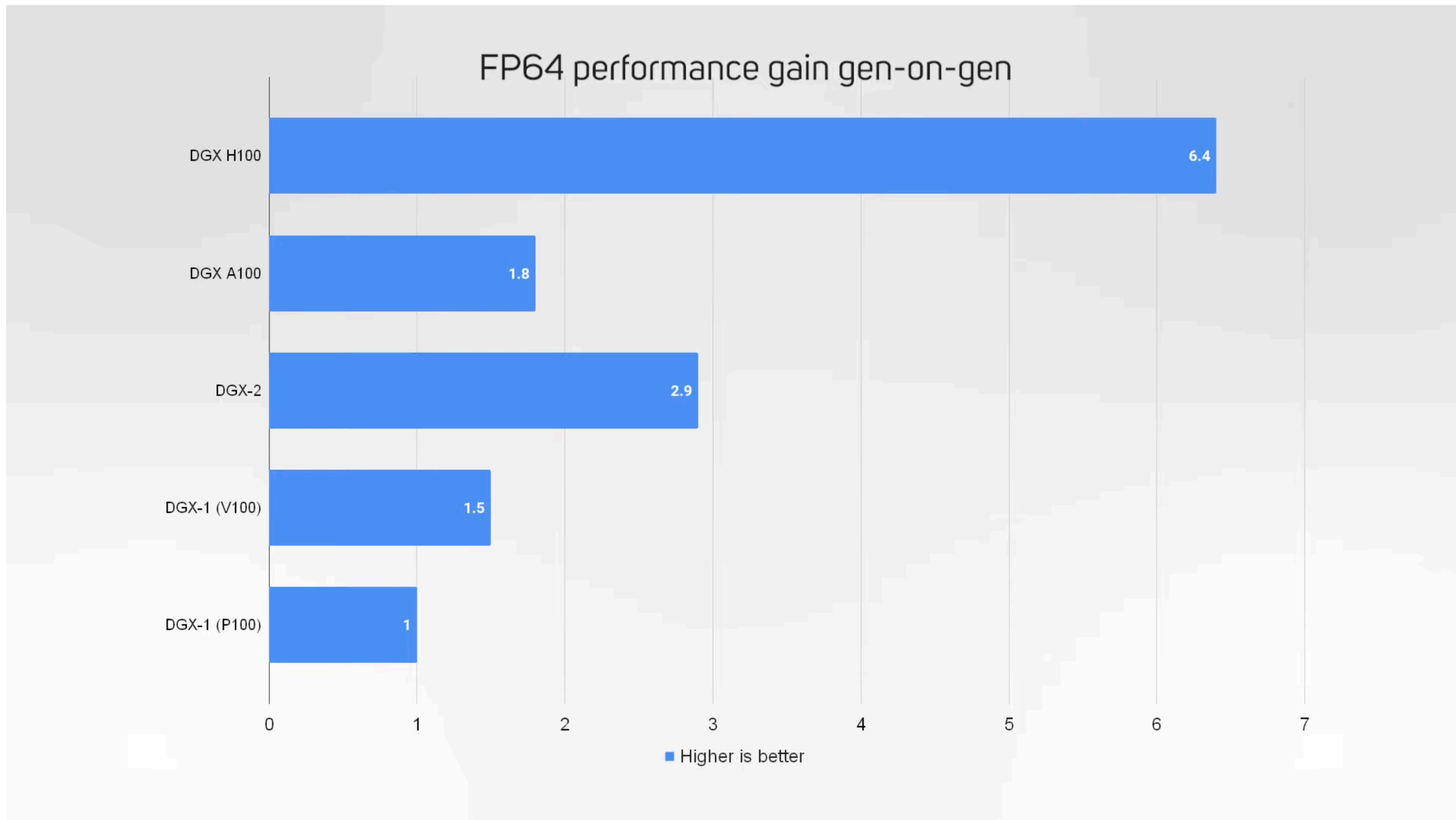


DGX A100

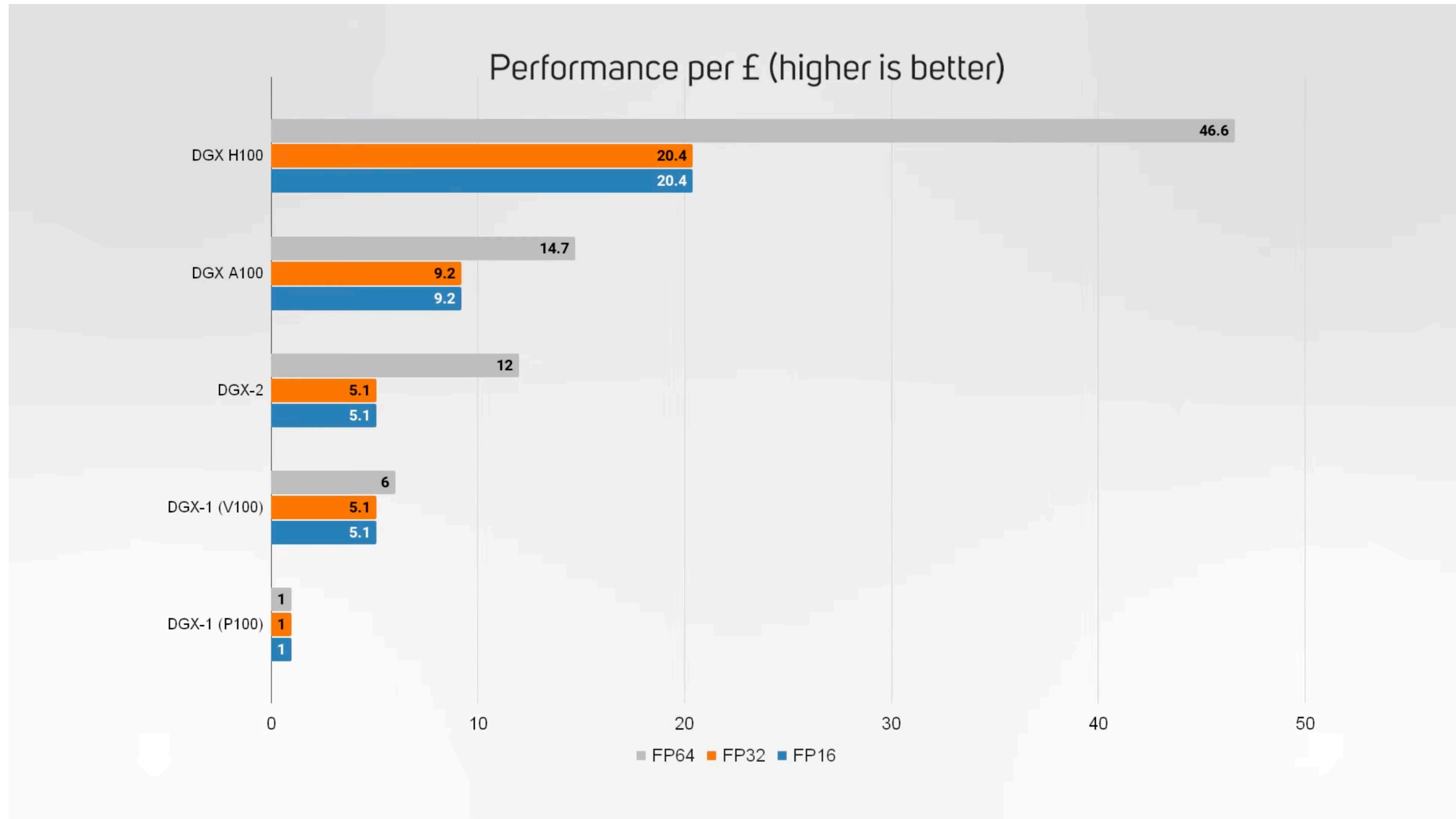
# Кластеры NVIDIA



# Кластеры NVIDIA



# Кластеры NVIDIA





# Нехватка памяти для обучения больших моделей

```
RuntimeError: cuda runtime error (2) : out of memory at /data/users/soumith/miniconda2/cond
```

how can i solve this error?

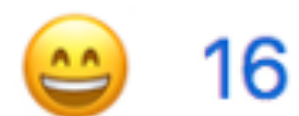


apaszke commented on Mar 8, 2017

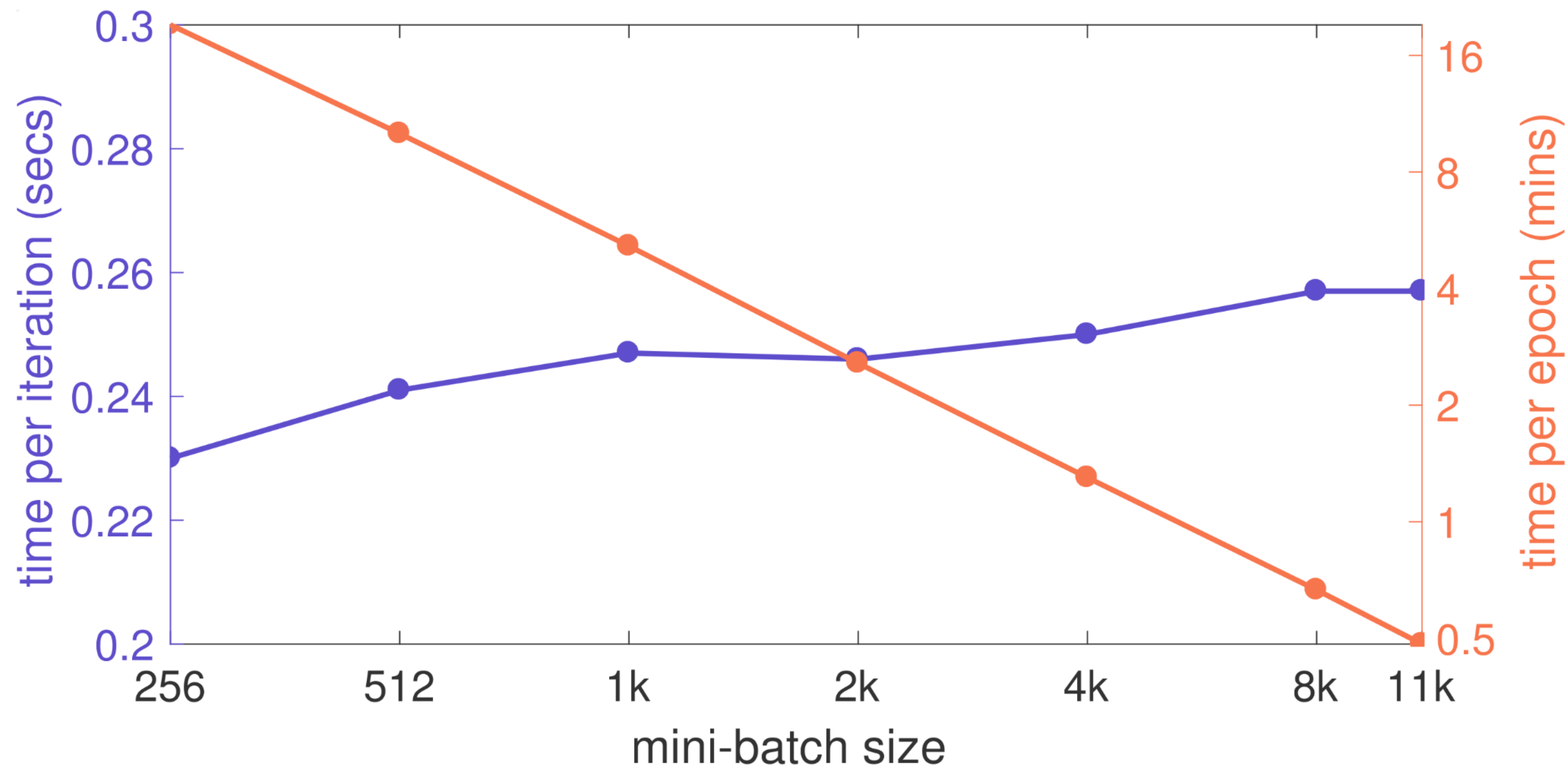
Member



You're running out of memory on the GPU. It's not a bug.



# Размер батча и время, затрачиваемое на одну эпоху



При наличии достаточной памяти GPU, увеличение размера батча позволяет утилизировать ресурсы параллельных вычислений.

 Paper

Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour.

# Просто так увеличить размер батча не получится

$kn$	$\eta$	top-1 error (%)
256	0.05	23.92 $\pm$ 0.10
256	0.10	23.60 $\pm$ 0.12
256	0.20	23.68 $\pm$ 0.09
8k	0.05 $\cdot$ 32	24.27 $\pm$ 0.08
8k	0.10 $\cdot$ 32	23.74 $\pm$ 0.09
8k	0.20 $\cdot$ 32	24.05 $\pm$ 0.18
8k	0.10	41.67 $\pm$ 0.10
8k	0.10 $\cdot$ $\sqrt{32}$	26.22 $\pm$ 0.03

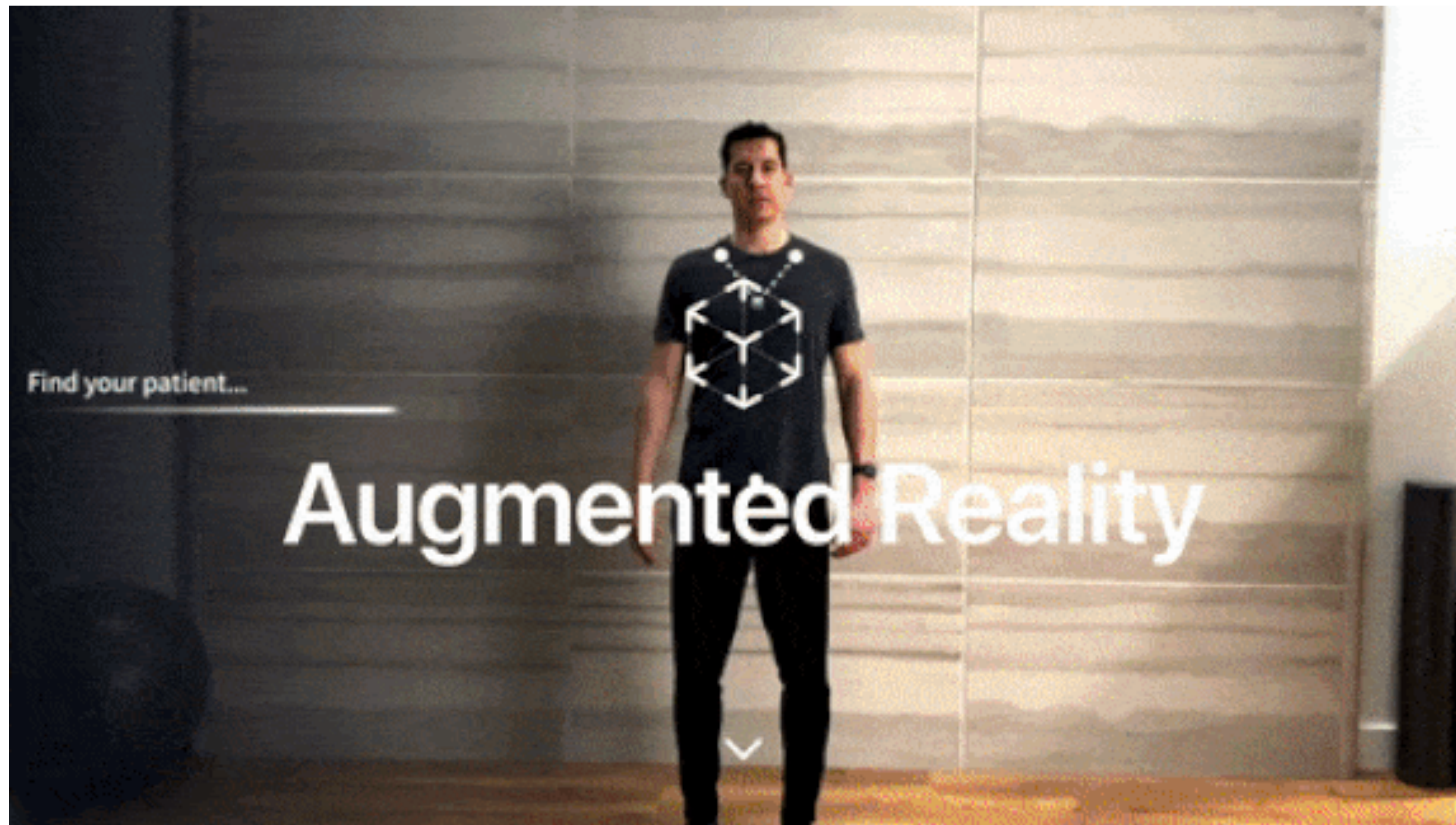
Обучение ResNet-50 на датасете ImageNet с разными вариантами увеличения размера батча.

(a) **Comparison of learning rate scaling rules.** A reference learning rate of  $\eta = 0.1$  works best for  $kn = 256$  (23.68% error). The linear scaling rule suggests  $\eta = 0.1 \cdot 32$  when  $kn = 8k$ , which again gives best performance (23.74% error). Other ways of scaling  $\eta$  give worse results.

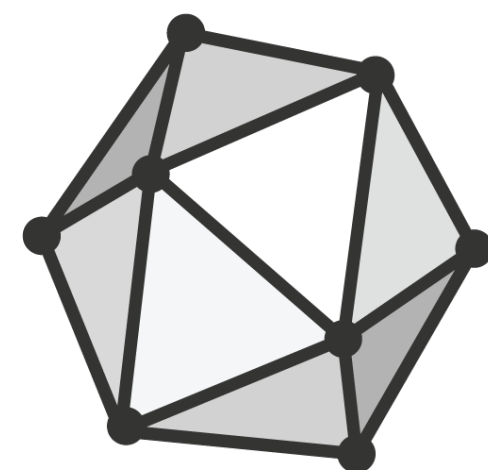
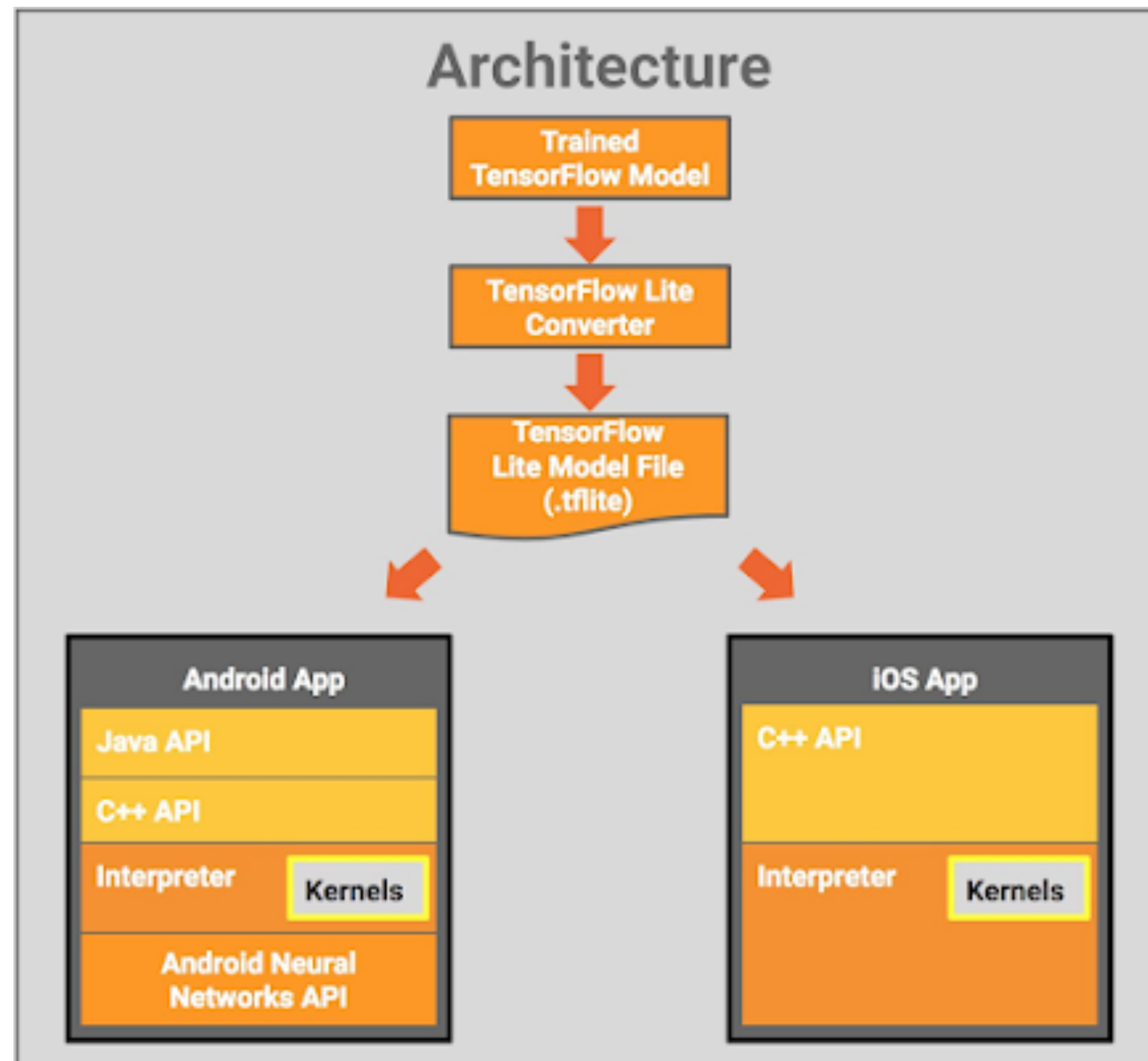
 Paper

Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour.

# On device AI



# On device AI



# ONNX



Можно обучать модели на мощном кластере, конвертировать и использовать на мобильных устройствах (или в браузере)

# LLM на телефоне

```
== Running in chat mode. ==
- Press Ctrl+C to interject at any time.
- Press Return to return control to LLaMA.
- If you want to submit another line, end your input in '\'.

кто такой пу
>кто такой Пушкин АС
Пушкин – город в России, расположенный на берегу реки Не
ве и относящийся к Московской области; основан 17 июля (
28 июня по старому стилю) 1956 года.
> кто такой пушкин а.с.
Пушкин – город в России, расположенный на берегу рек
и/ Нева и относящийся к Московской области; основан 17 и
юля (28 июня по старому стилю) 1956 года.
> пушкин александр сергеевич это
Пушкин, Александр Сергеевич – русский поэт и драматург (
написал «Евга»), который был одним из основоположников р
омантического стиля в литературе. Он родился 26 апреля 1
798 года на берегу Невы; умер от туберкулёза 30 марта 18
45 года, не дожившись своего тридцатипятого дня рождения
(на момент смерти ему было всего двадцать шесть лет).
Пушкин был одним из первых русских писателей-романтиче
в и создал несколько произведений в этом стиле, таких ка
к «Евга», «Руслан и Людмила» или «Борис Годунов». Он так
же написал романы (например, «Капитанская дочка») и пьес
у («Мнимый больной»).
Посмертно Пушкин получил всемирное признание. В 1870 год
у в честь его памяти был открыт Музей-квартира на Неве; с
тех пор он стал одним из главных туристических объектов
Санкт Петербурга, а также местом проведения различных к
ультурно-образовательных мероприятий.
В 1936 году в честь Пушкина был переименован город Москв
а (до этого носил название «Петров»), и с тех пор он ста
л называться Санкт Петербург, а сам поэт получил звание
почётного гражданина города-героя.
В 1956 году в честь Пушкина был назван космический объек
т – Луна (до этого носил название «Земля»). В настоящее
время он является одним из самых популярных и любимым пи
сателей России, а также считается выдающимся поэтом-рома
нтиком.
В 1987 году в честь Пушкин был назван космический объект
– Марс (до этого носил название «Меркурий»). В настояще
е время он является одним из самых популярных и любимым
писателей России, а также считается выдающимся поэтом-ро
мантиком.
В 2017 году в честь Пушкина был назван космический объек
т – Нептун (до этого носил название «Юпитер»). В настоящ
ее время он является одним из самых популярных
```


```
16Kb 16Kb
05:00 89%

= 291

system_info: n_threads = 4 / 8 | AVX = 0 | AVX2 = 0 | AV
X512 = 0 | FMA = 0 | NEON = 1 | ARM_FMA = 1 | F16C = 0 |
FP16_VA = 1 | WASM_SIMD = 0 | BLAS = 0 | SSE3 = 0 | VSX
= 0 |
main: interactive mode on.
sampling parameters: temp = 0.100000, top_k = 40, top_p
= 0.950000, repeat_last_n = 64, repeat_penalty = 1.30000
0

== Running in chat mode. ==
- Press Ctrl+C to interject at any time.
- Press Return to return control to LLaMA.
- If you want to submit another line, end your input in
'\'.

> tell me a story about fallout
Once upon a time, there was a small town in America call
ed Fallout Shelter. It had been around for generations a
nd it'd always felt like home to its residents - until o
ne day when the world changed forever... A nuclear warh
ead fell from space onto Earth and landed right on top o
f their beloved little hamlet, destroying everything wit
h a blinding flash!
The survivors were left in shock. They had no idea what
was going on or how to deal with it - until they found t
he Fallout Shelter... It turned out that this secret un
derground shelter could protect them from radiation and
other dangers of nuclear warfare, so everyone rushed dow
n there for safety!
But even though their lives were saved by a miracle,
```

 **Habr**  
Запускаем на телефоне нейронку LLaMa 7B -13B 4 bit от фейсбука с настройкой Alраса-LoRA

# USB accelerators



Google coral USB Accelerator, ~60\$



Nvidia Jetson nano, ~130\$



Intel Movidius Neural compute stick ~80\$

MobileNetV2

Setup	Time [s]	fps	Score
i7-4870HQ	22,60	11,06	0,9294
i7-7700K + GTX1080 2560CUDA	0,82	304,88	0,9294
i7-7700K (tflite)	7,71	32,43	16,2184
i7-7700K + Coral (tflite)	0,71	352,11	0,9961
Jetson Nano 128CUDA	20,52	12,18	0,9294
Jetson Nano (tflite)	23,02	10,86	16,2184
RPi	120,81	2,07	0,9294
RPi (tflite)	59,05	4,23	16,2184
Rpi + Coral (tflite)	3,31	75,53	0,9961
Jetson Nano + Coral (tflite)	1,12	223,21	0,9961

# Архитектуры искусственных нейросетей и решаемые задачи





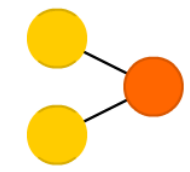
# Много разных архитектур

## A mostly complete chart of Neural Networks

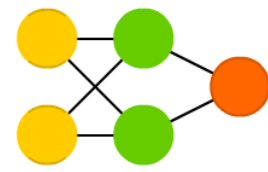
©2019 Fjodor van Veen & Stefan Leijnen asimovinstitute.org

- Input Cell
- Backfed Input Cell
- △ Noisy Input Cell
- Hidden Cell
- Probabilistic Hidden Cell
- △ Spiking Hidden Cell
- Capsule Cell
- Output Cell
- Match Input Output Cell
- Recurrent Cell
- Memory Cell
- △ Gated Memory Cell
- Kernel
- Convolution or Pool

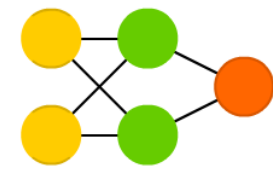
Perceptron (P)



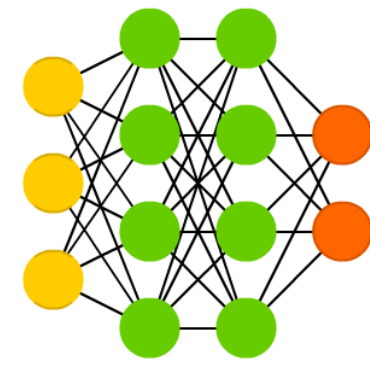
Feed Forward (FF)



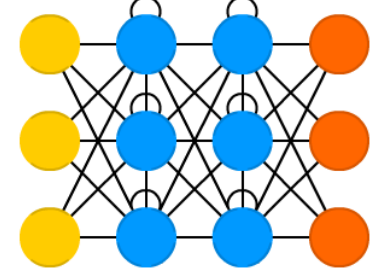
Radial Basis Network (RBF)



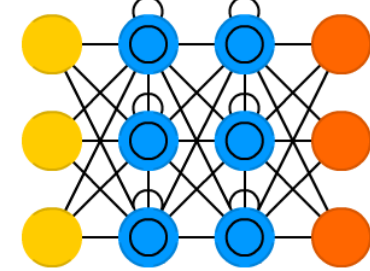
Deep Feed Forward (DFF)



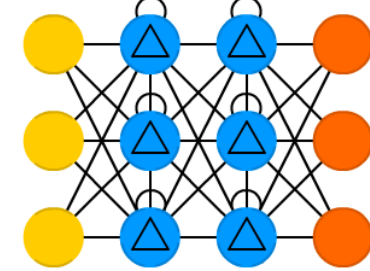
Recurrent Neural Network (RNN)



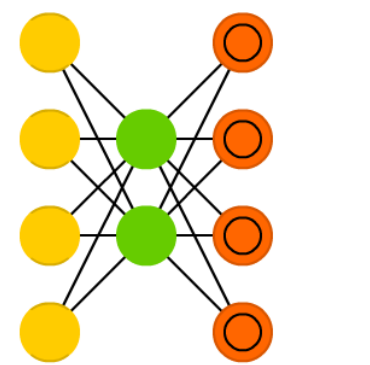
Long / Short Term Memory (LSTM)



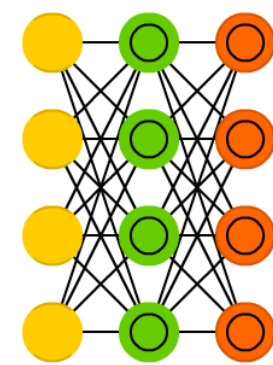
Gated Recurrent Unit (GRU)



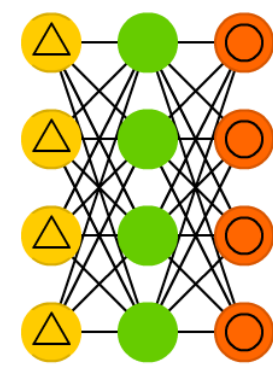
Auto Encoder (AE)



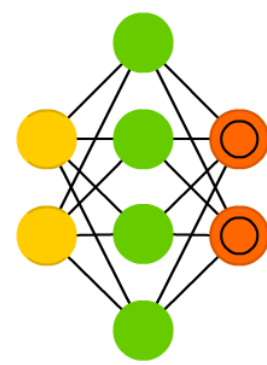
Variational AE (VAE)



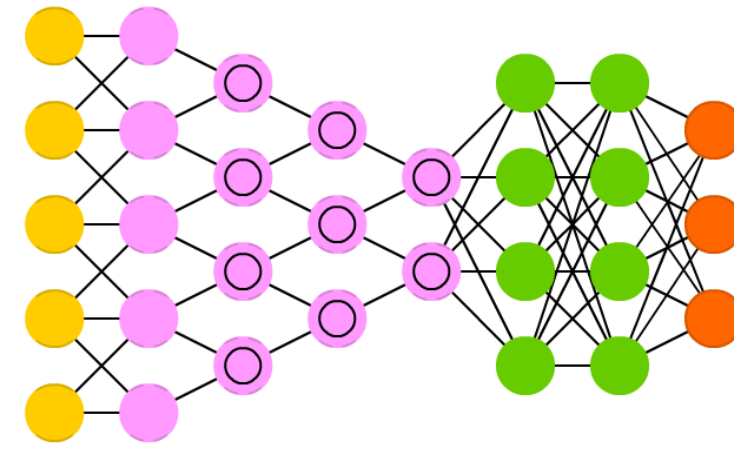
Denosing AE (DAE)



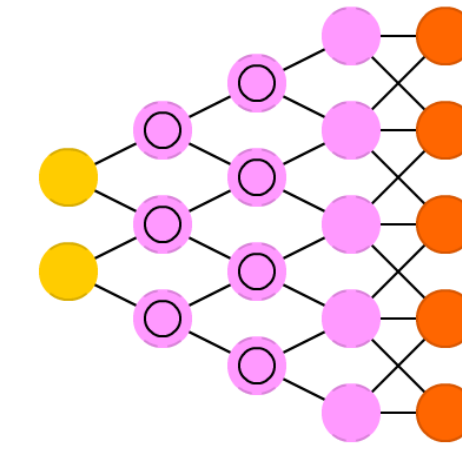
Sparse AE (SAE)



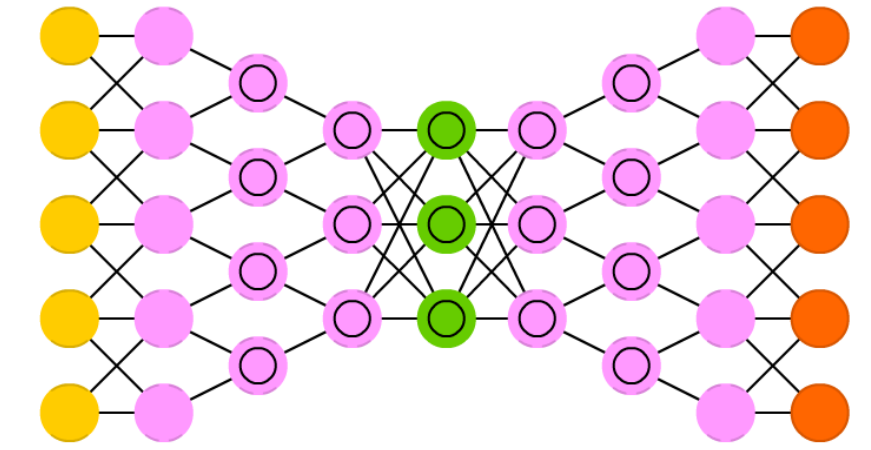
Deep Convolutional Network (DCN)



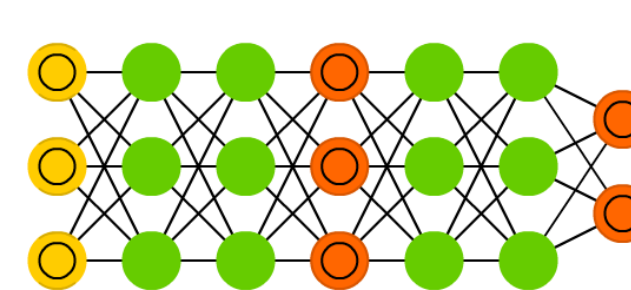
Deconvolutional Network (DN)



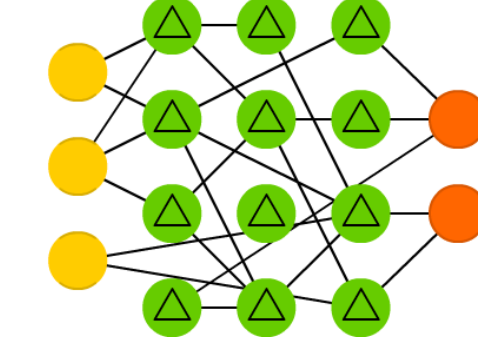
Deep Convolutional Inverse Graphics Network (DCIGN)



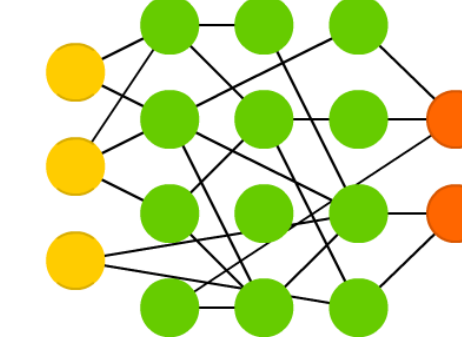
Generative Adversarial Network (GAN)



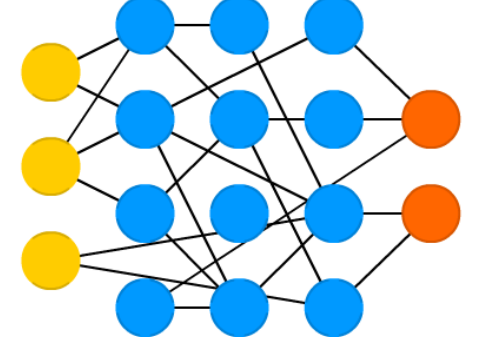
Liquid State Machine (LSM)



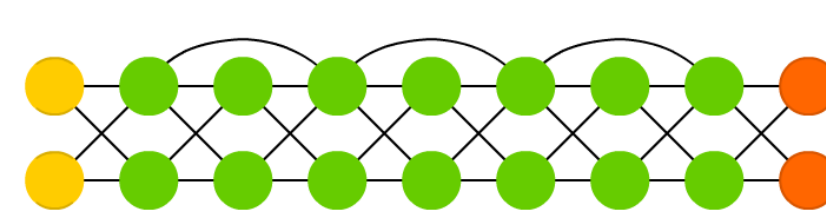
Extreme Learning Machine (ELM)



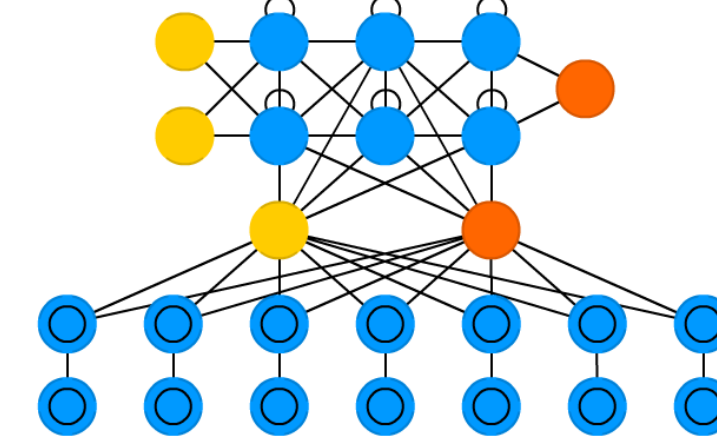
Echo State Network (ESN)



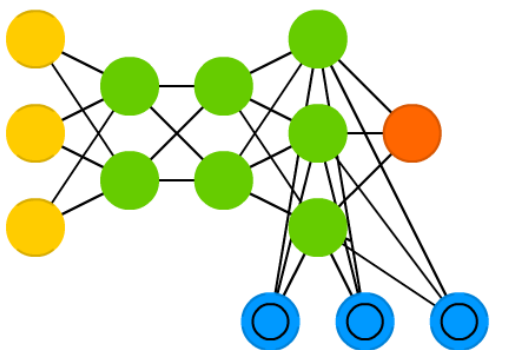
Deep Residual Network (DRN)



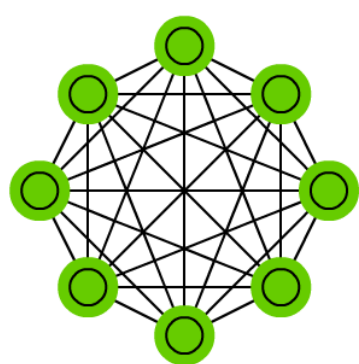
Differentiable Neural Computer (DNC)



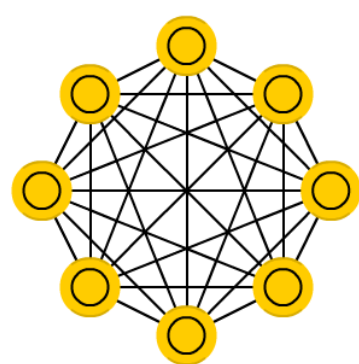
Neural Turing Machine (NTM)



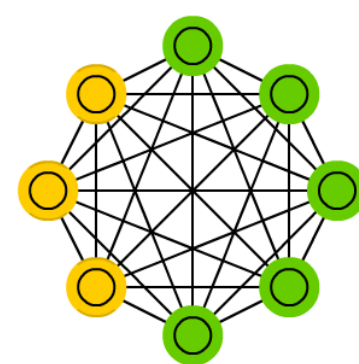
Markov Chain (MC)



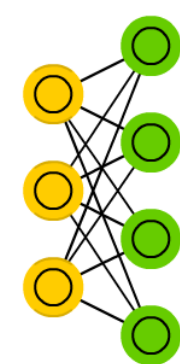
Hopfield Network (HN)



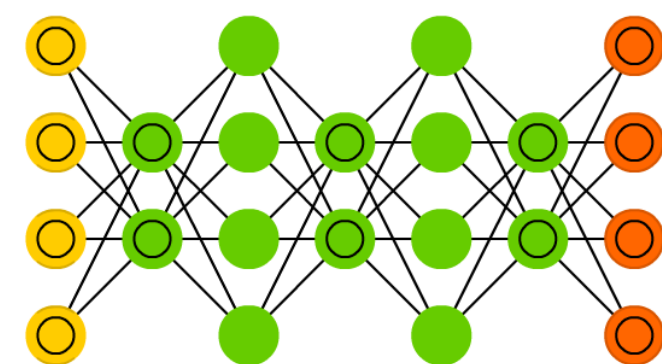
Boltzmann Machine (BM)



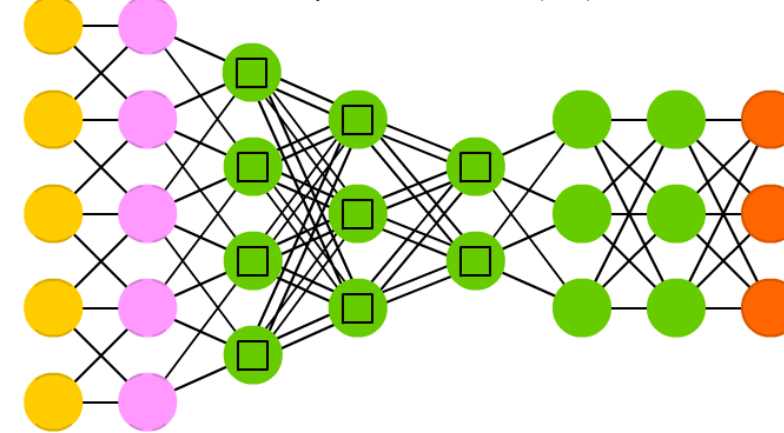
Restricted BM (RBM)



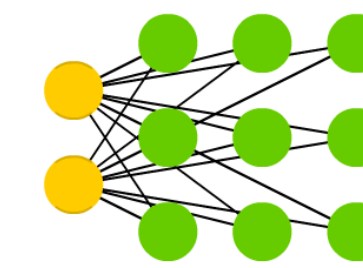
Deep Belief Network (DBN)



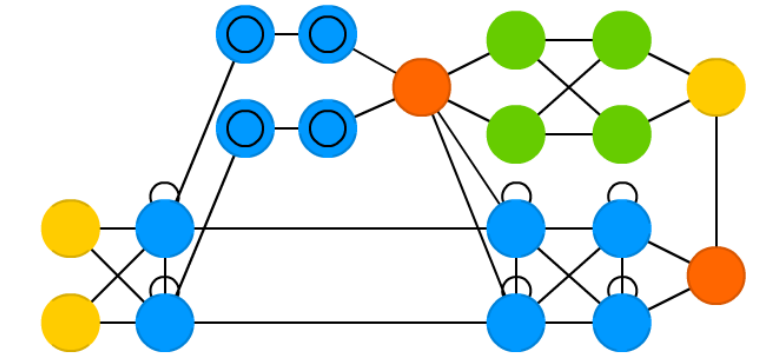
Capsule Network (CN)



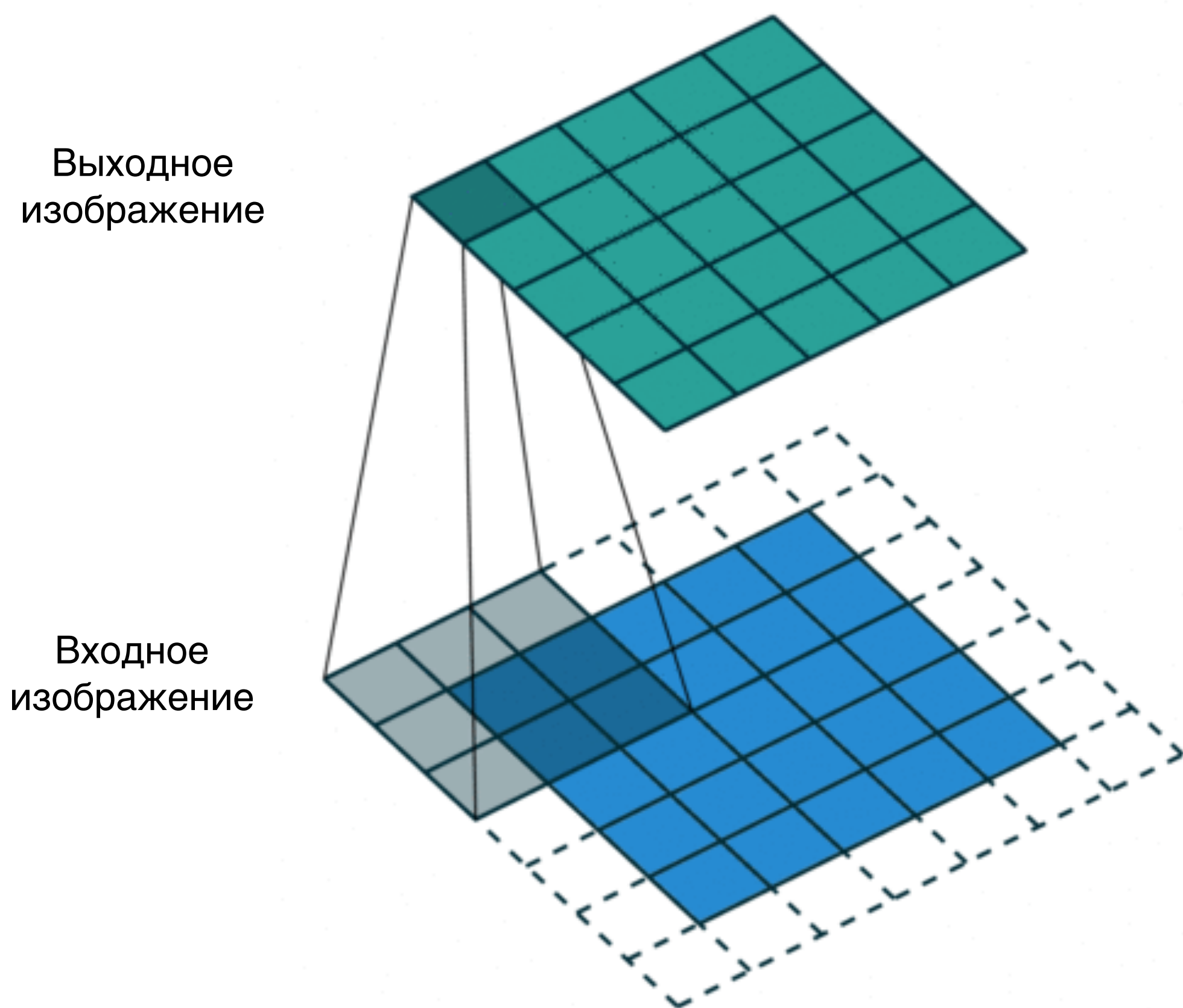
Kohonen Network (KN)



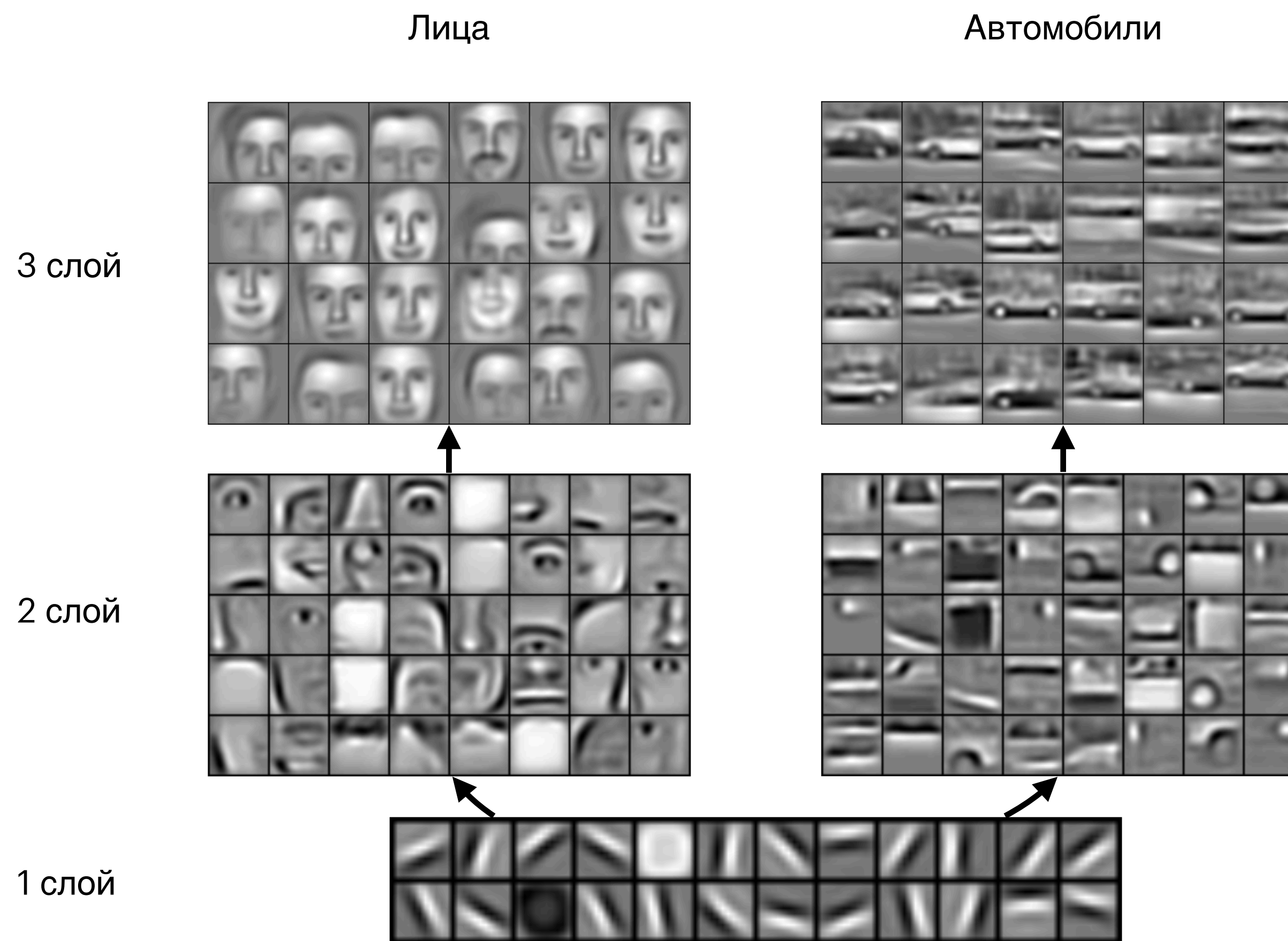
Attention Network (AN)



# Сверточные слои



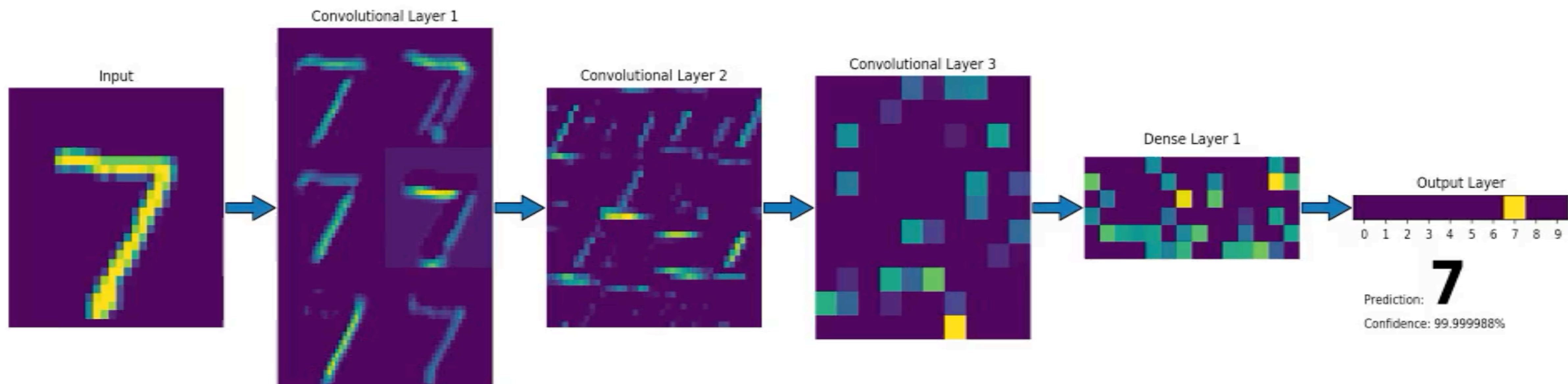
Фильтр размера 3x3 возвращает большое число, если на картинке паттерн, похожий на фильтр



Чем глубже слой нейросети, тем на более сложные паттерны реагирует фильтр нейросети

# Сверточные слои

Повсеместно применяются в анализе изображений, т.к. не игнорируют информацию о взаимном расположении пикселей



🤖 Сайт

Интерактивная карта сверточной нейросети

# Выводы

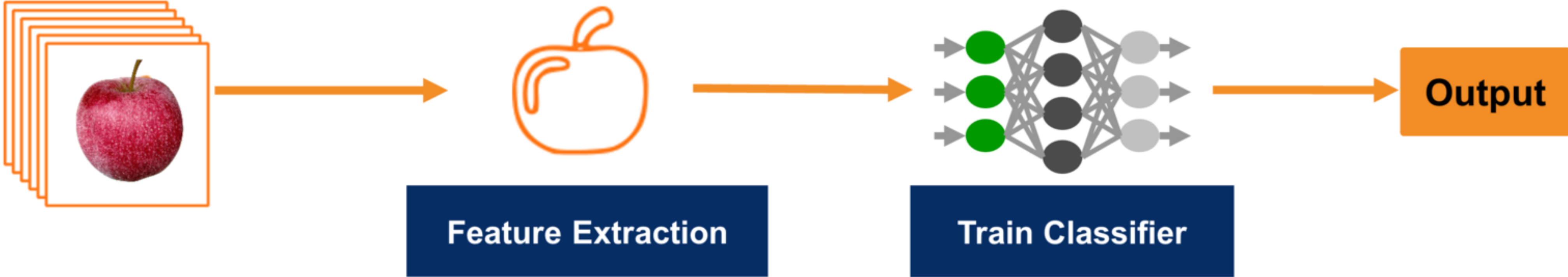
- Модели на основе нейронных сетей (НС) уже активно применяются в анализе изображений, видео, текста, речи и других данных.
- Разработка новых архитектур\слоев\алгоритмов обучения НС – задача исследователей. Прикладная задача – подбор «оптимальных» параметров НС.
- Сбор данных для НС – сложная задача. Если есть возможность – данные нужно брать не только специфичные для вашей задачи, но и использовать публичные датасеты.

# Постановки задач в компьютерном зрении



# Classical ML vs DL

## Classic Machine Learning



## Deep Learning



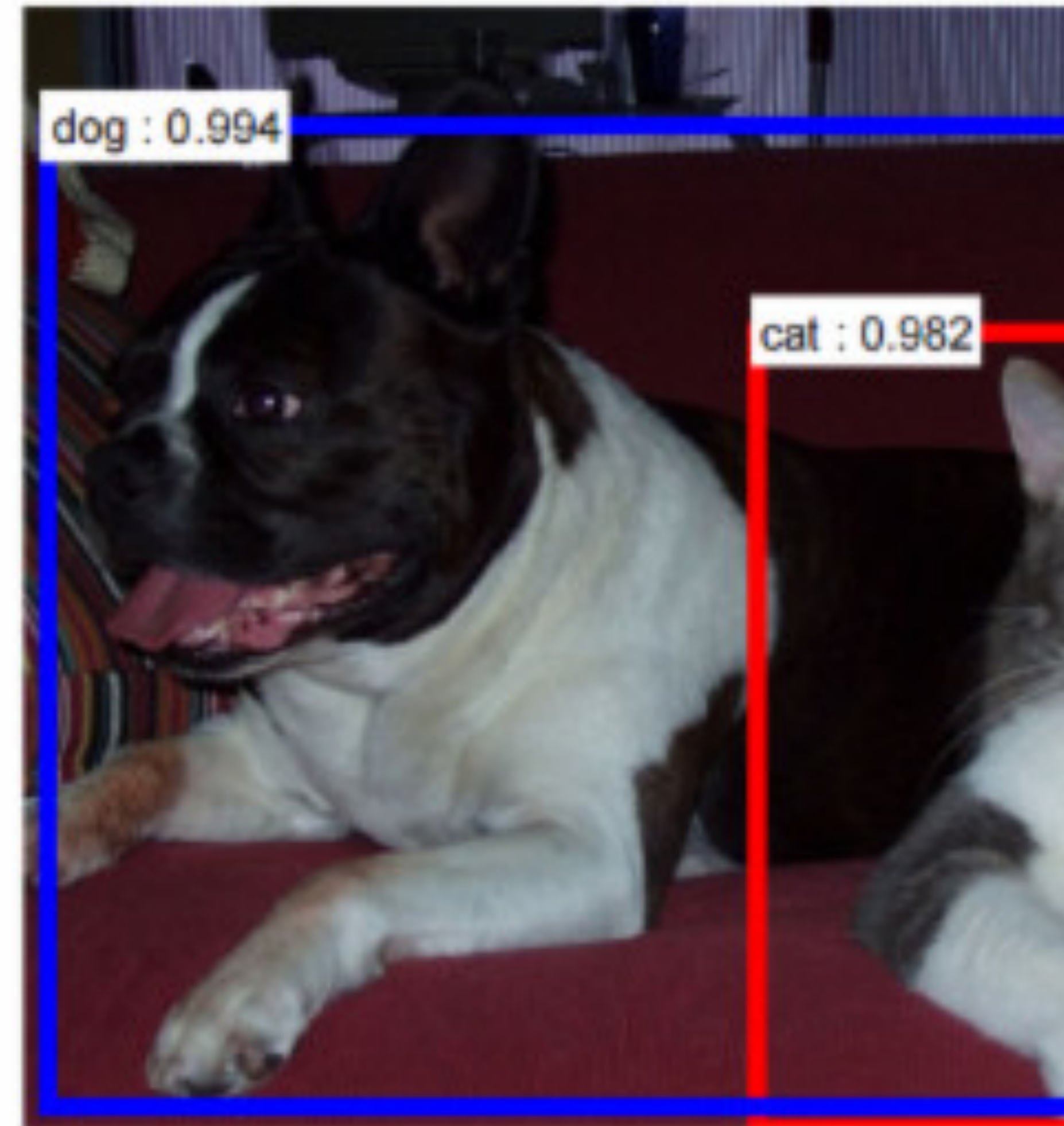
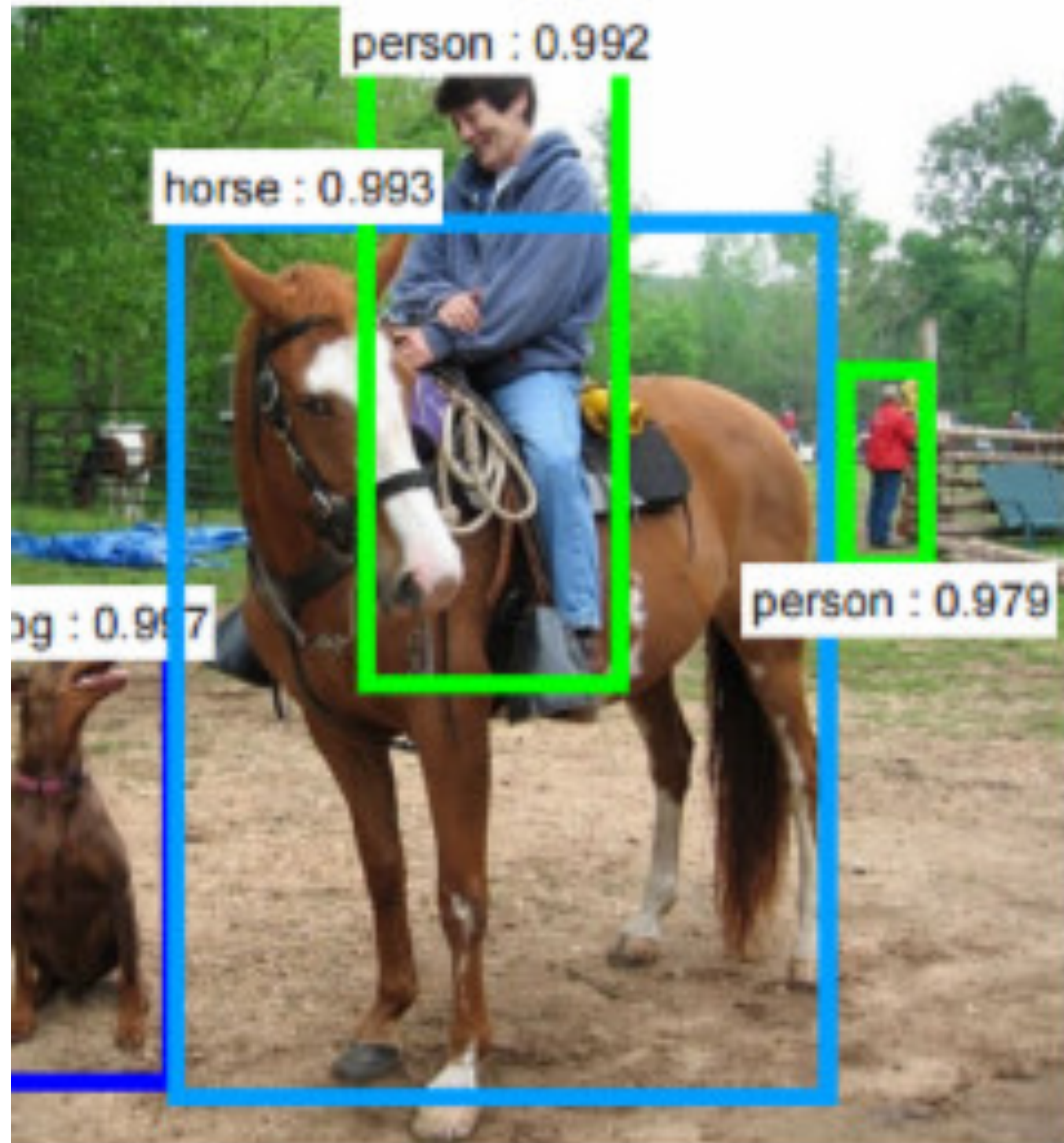
# Image classification



По изображению необходимо предсказать его класс

В качестве метрики качества используется точность

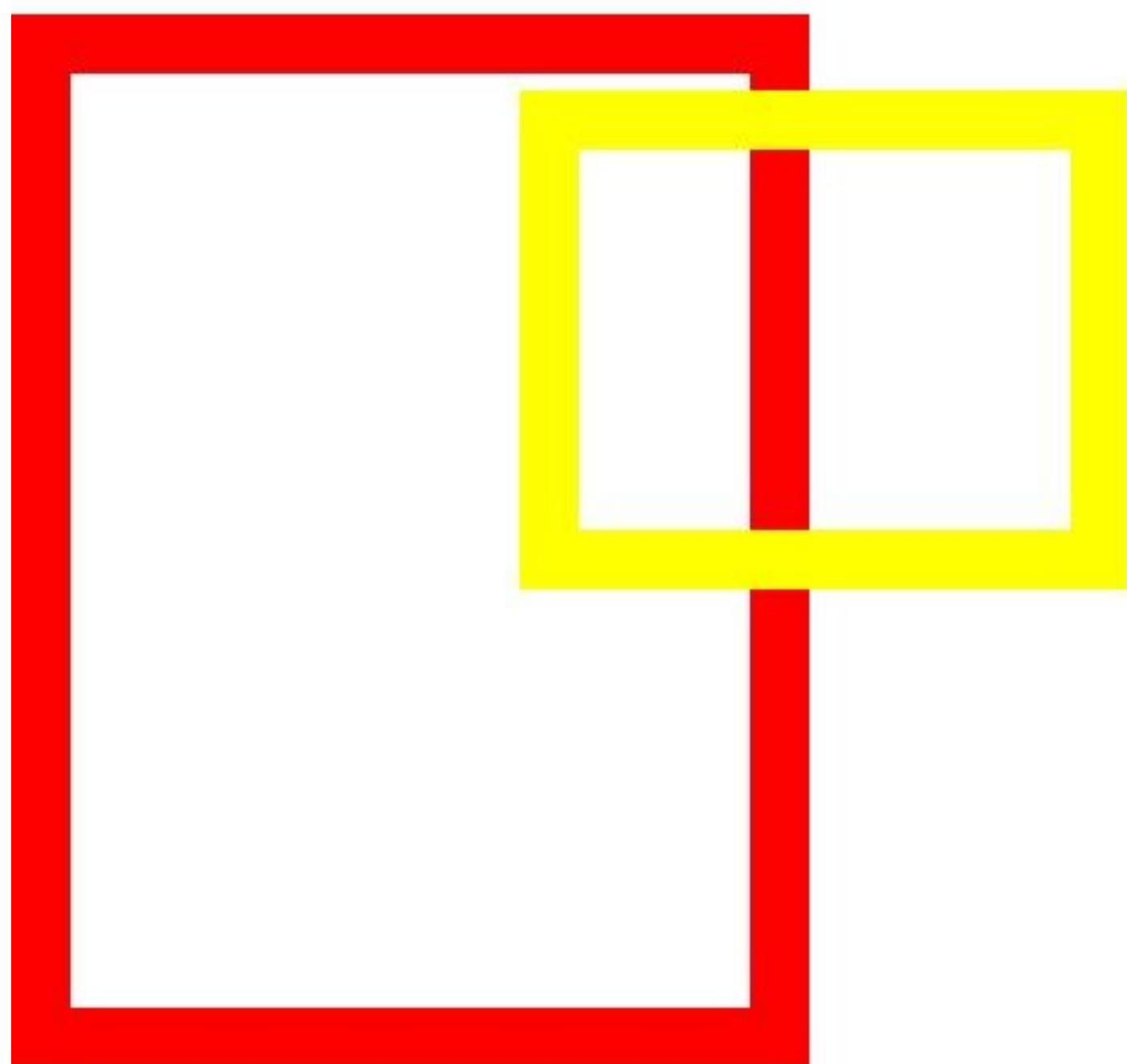
# Object detection





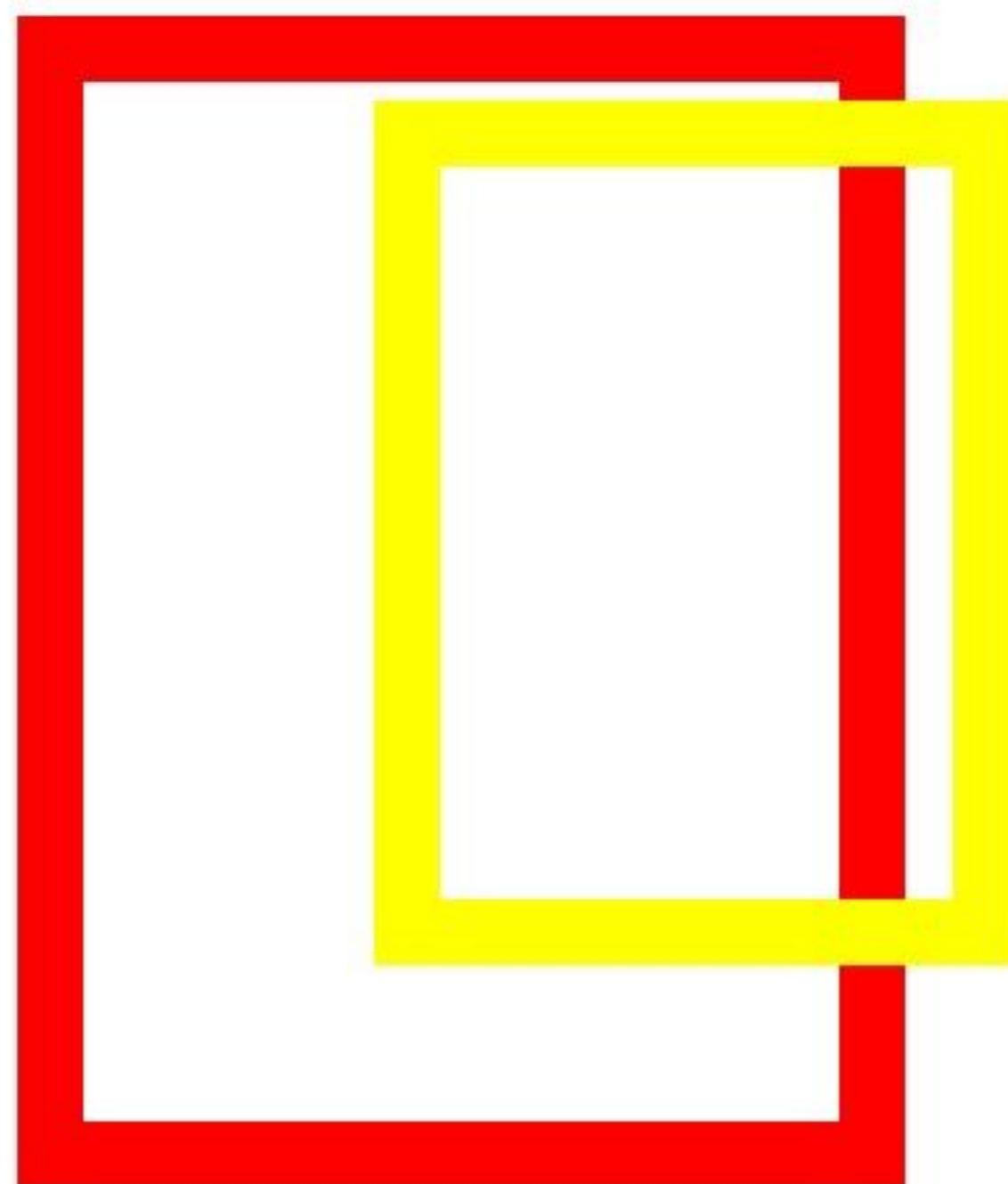
# Object detection

IoU 0.2



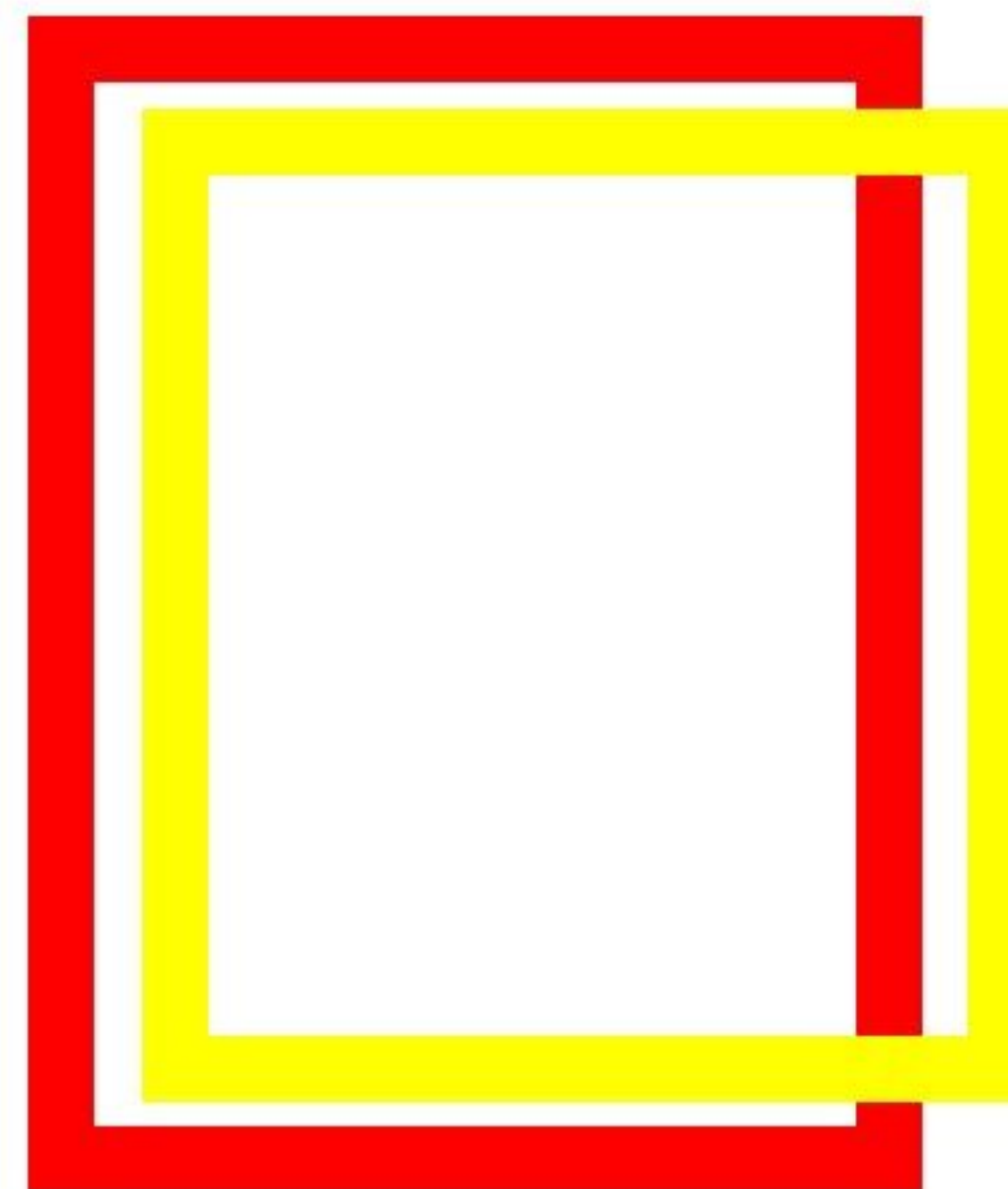
A

IoU 0.5



B

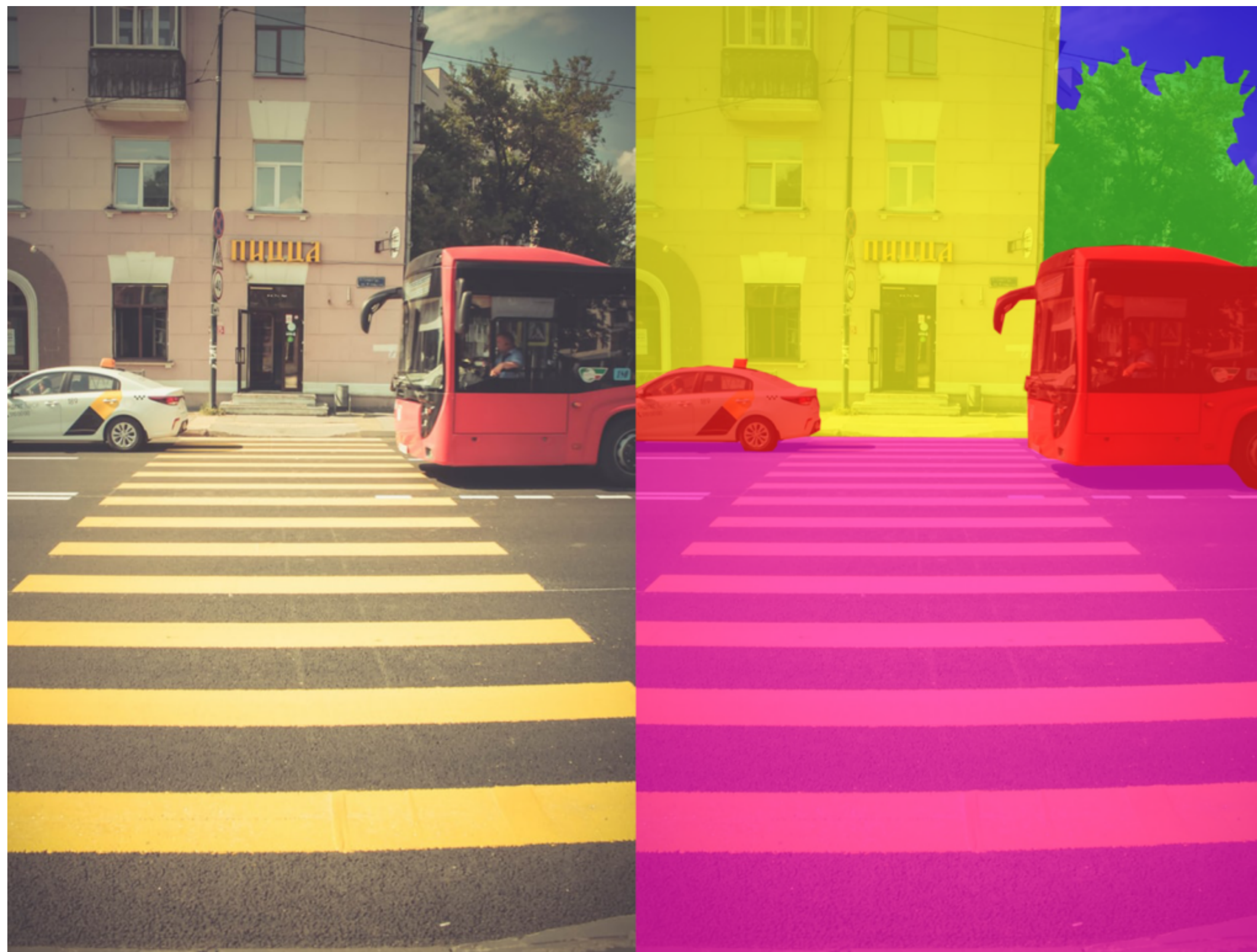
IoU 0.9



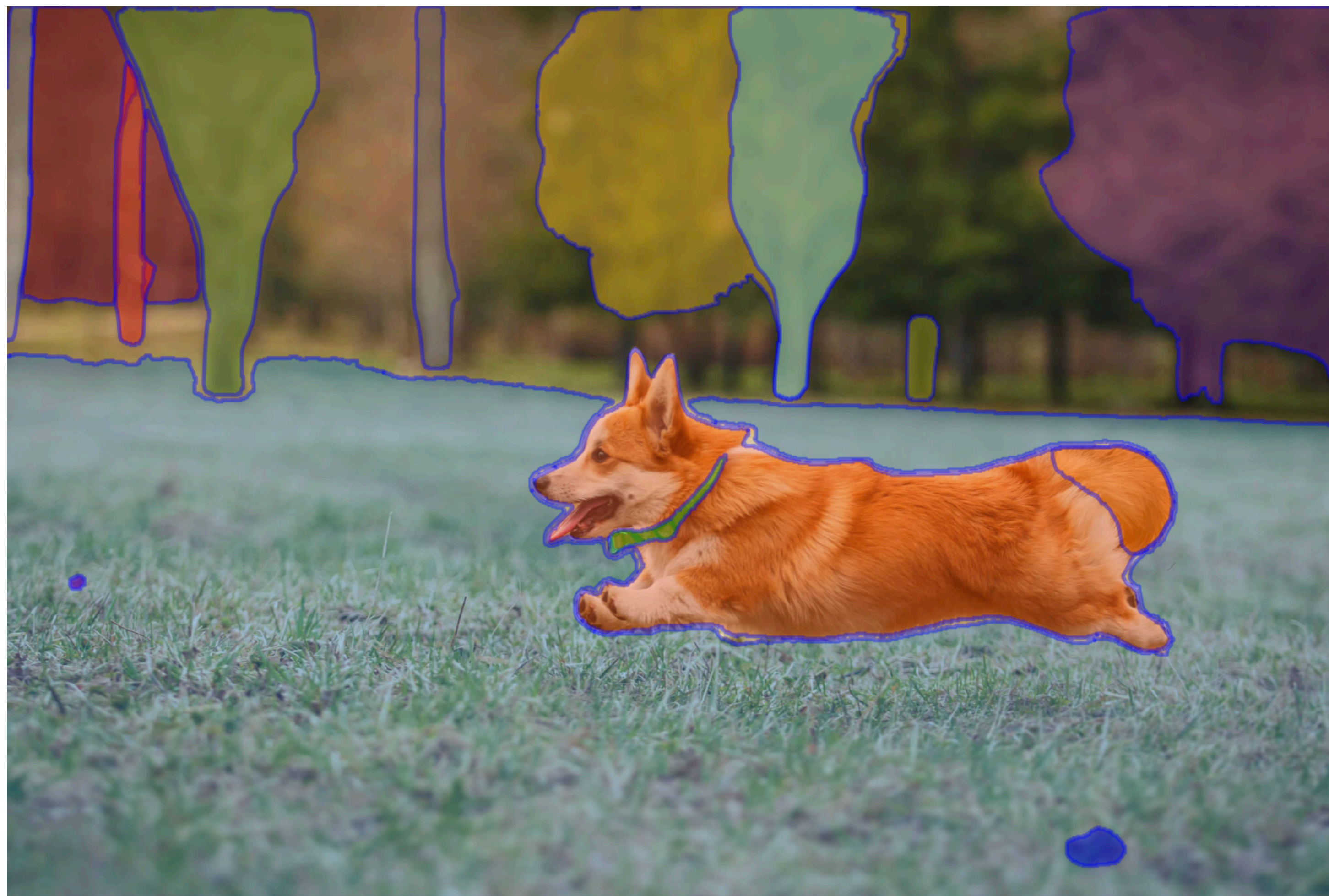
C

Intersection over union (IoU)

# Semantic segmentation



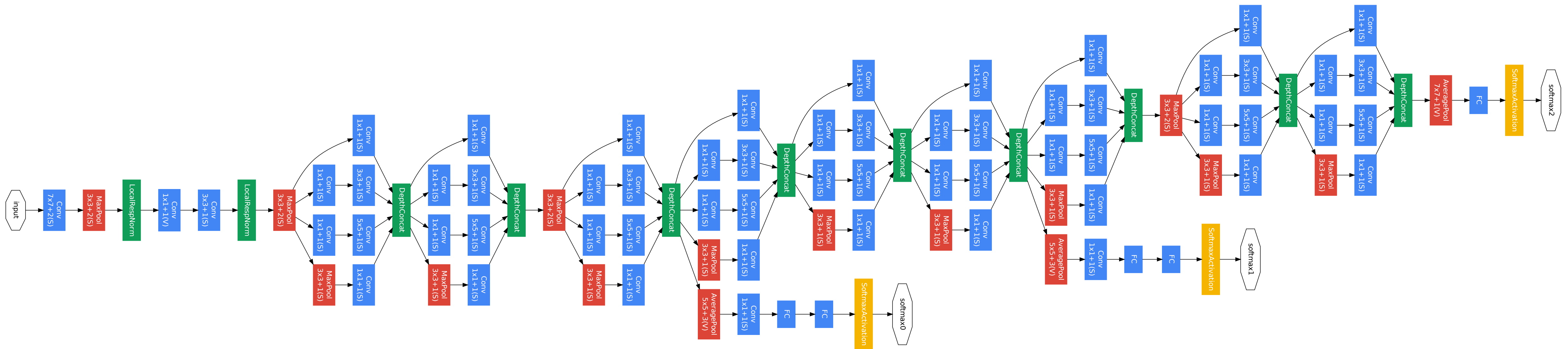
# Semantic segmentation



 Paper

Segment Anything Model

# Inception (GoogleNet) (2014)




Convolution/FC

Concat / Normalization

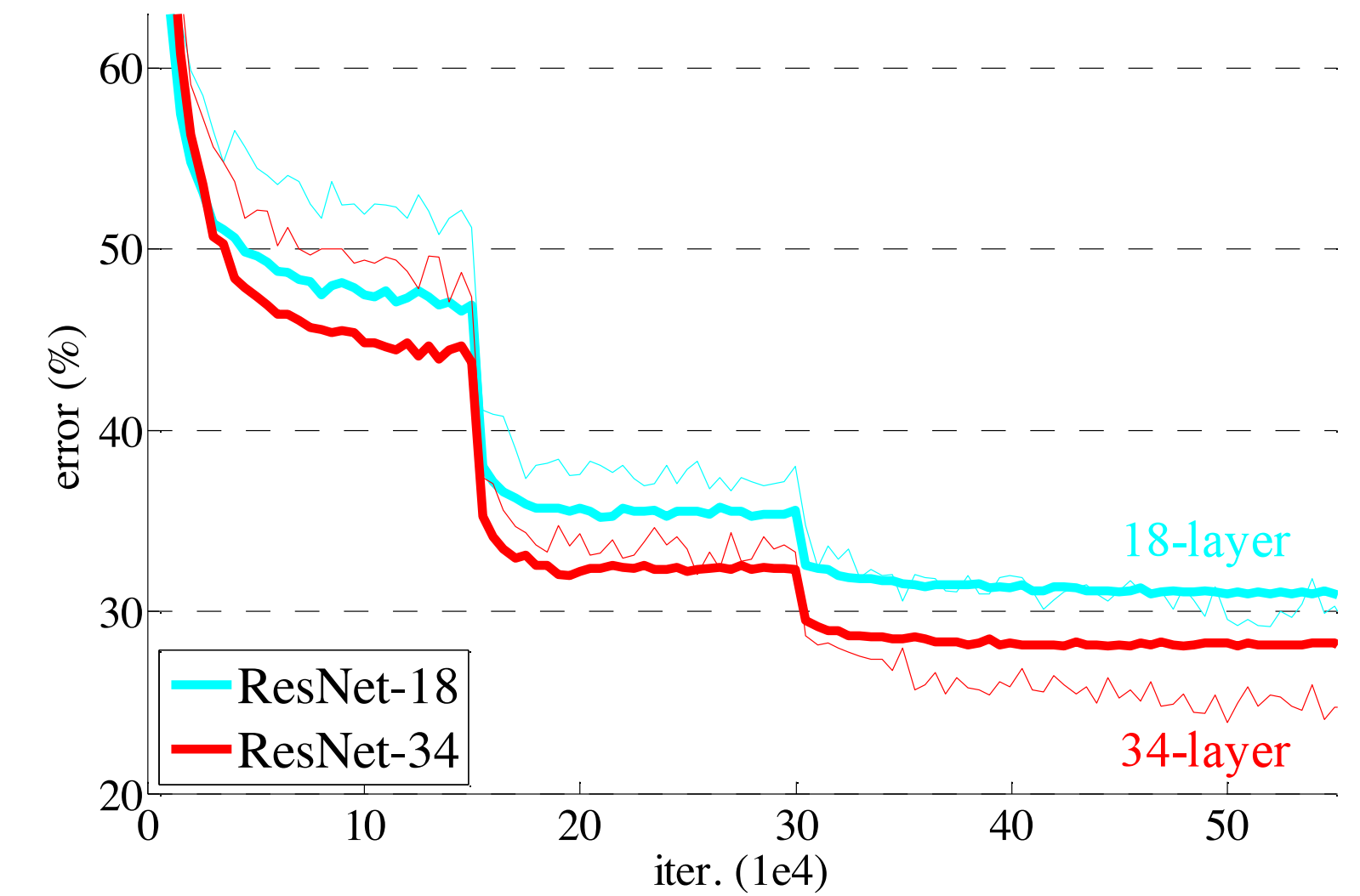
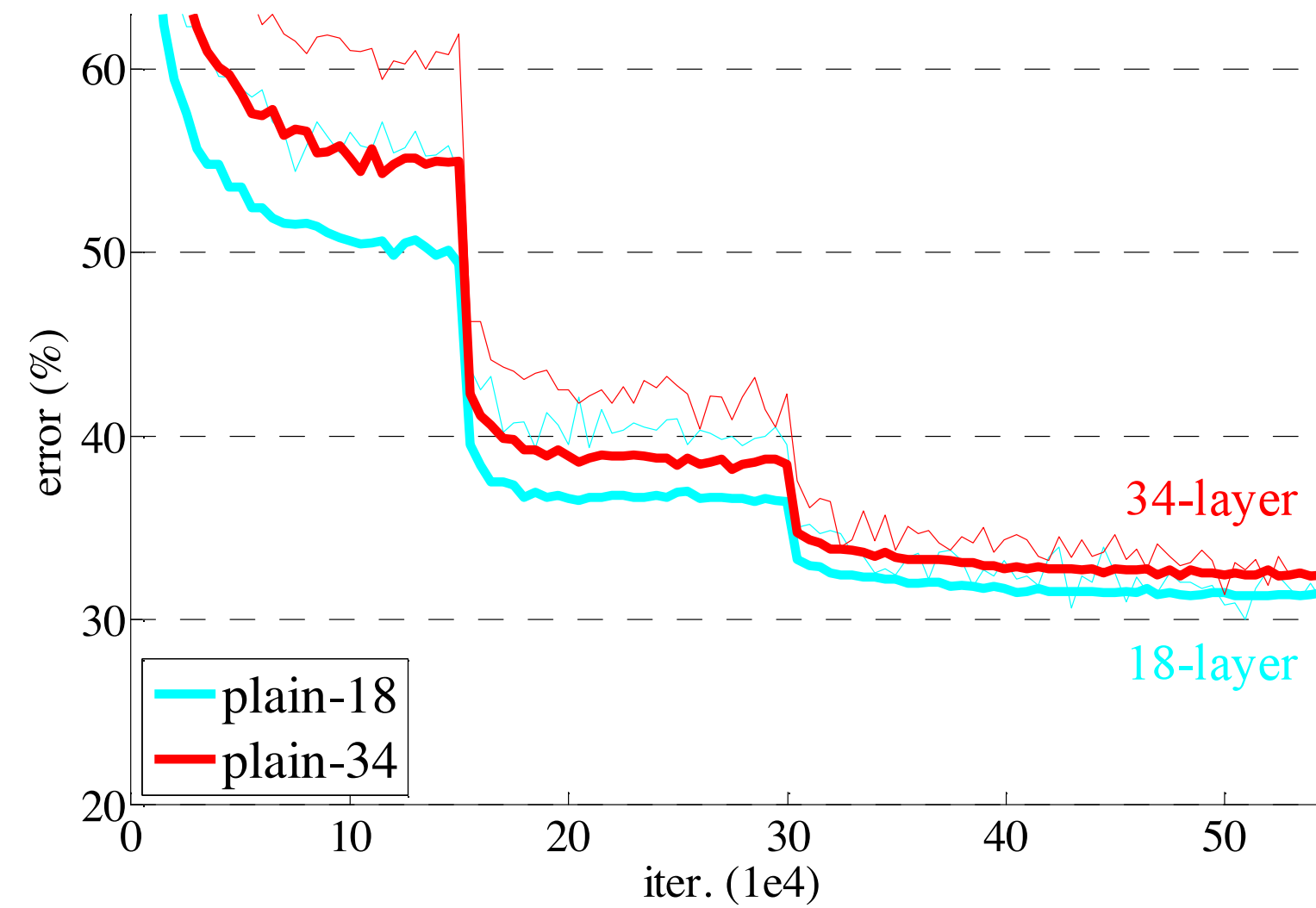
Pooling

Softmax

 **Paper**  
Going deeper with convolutions

# ResNet (2015)

**Проблема: глубокие сети тяжело обучать (затухание/взрыв градиентов)**

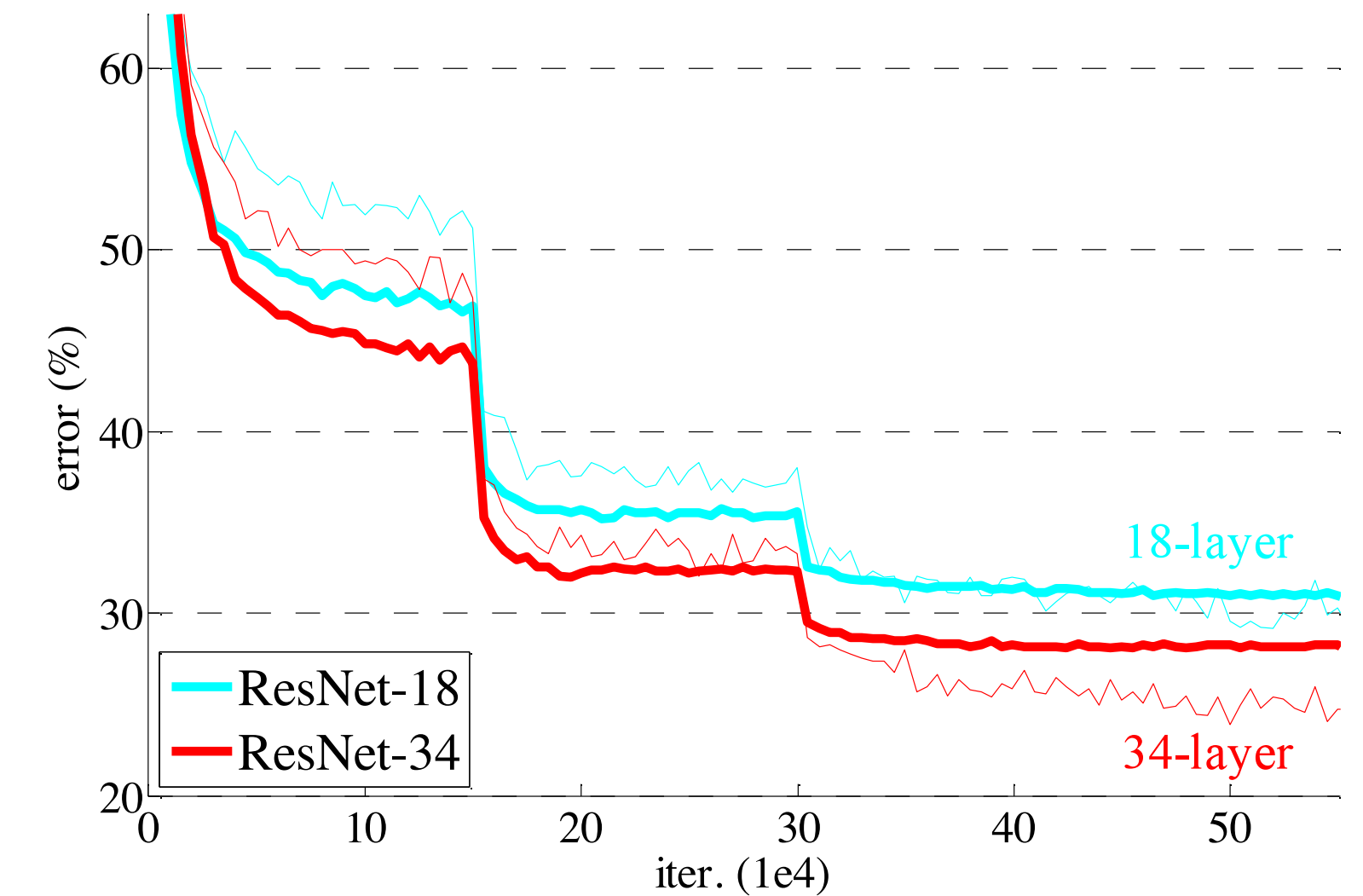
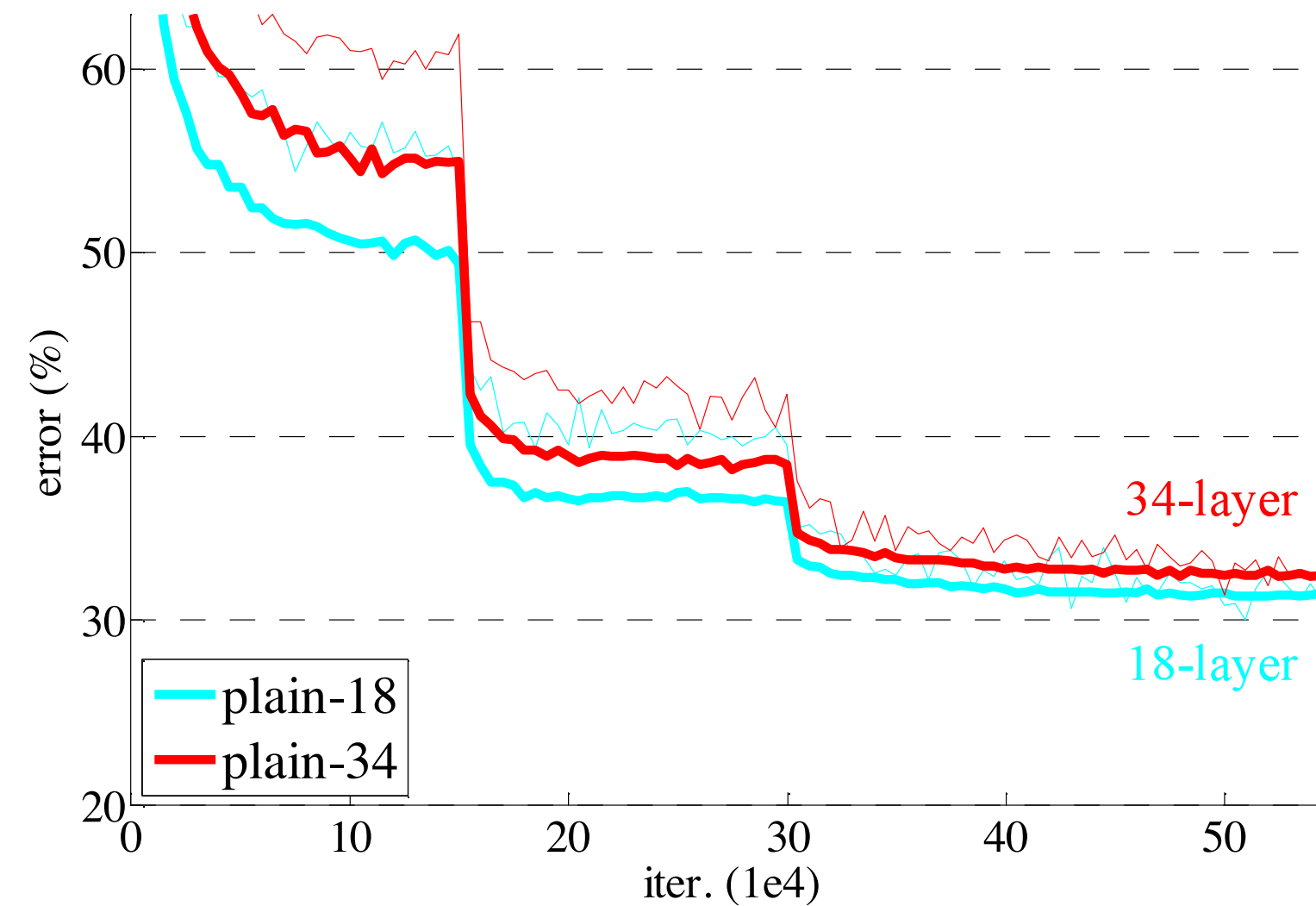


 Paper

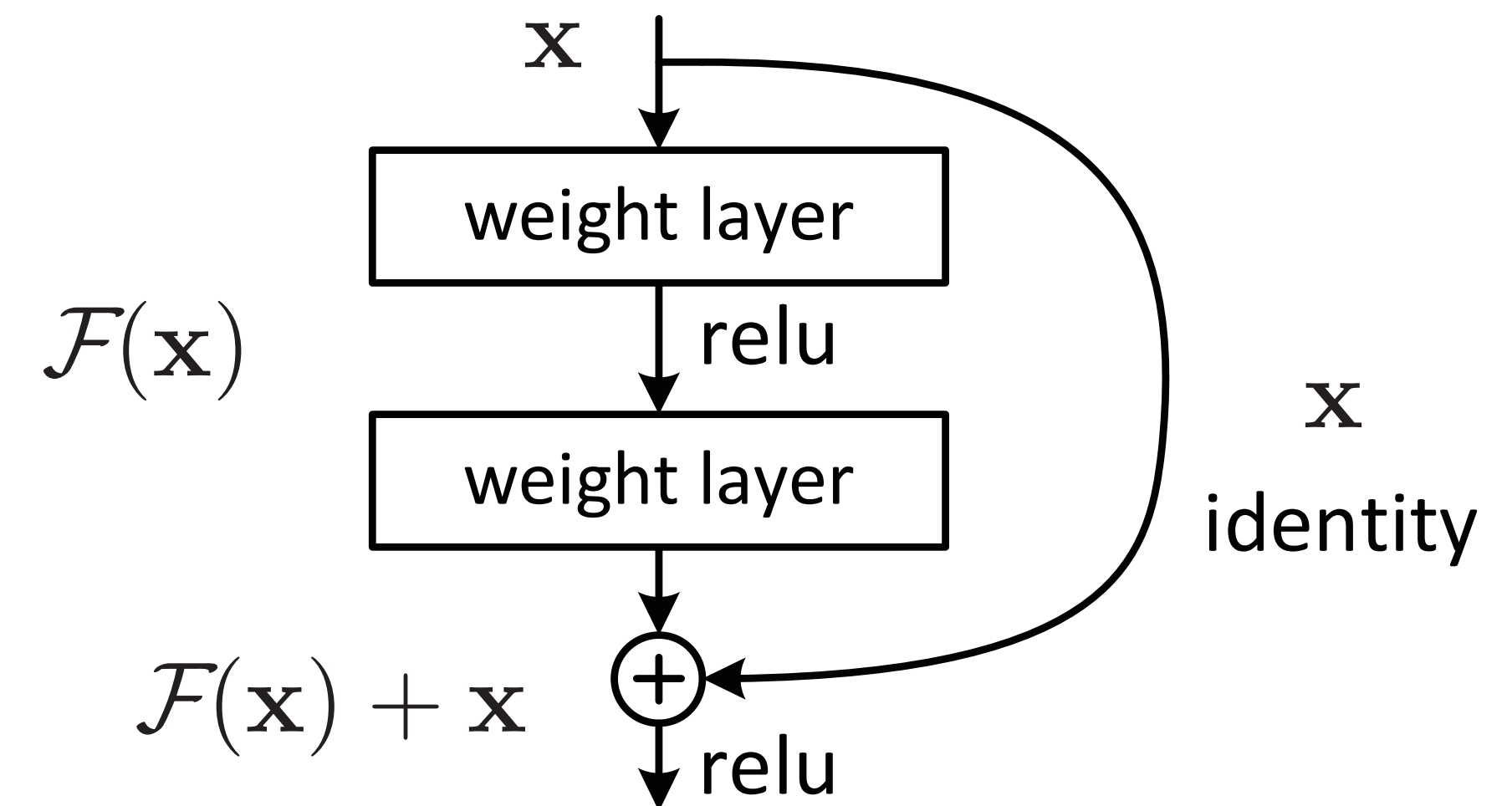
Deep Residual Learning for Image Recognition

# ResNet (2015)

**Проблема:** глубокие сети тяжело обучать (затухание/взрыв градиентов)



**Решение:** skip-connections позволяют обучать слои в инкрементальной манере



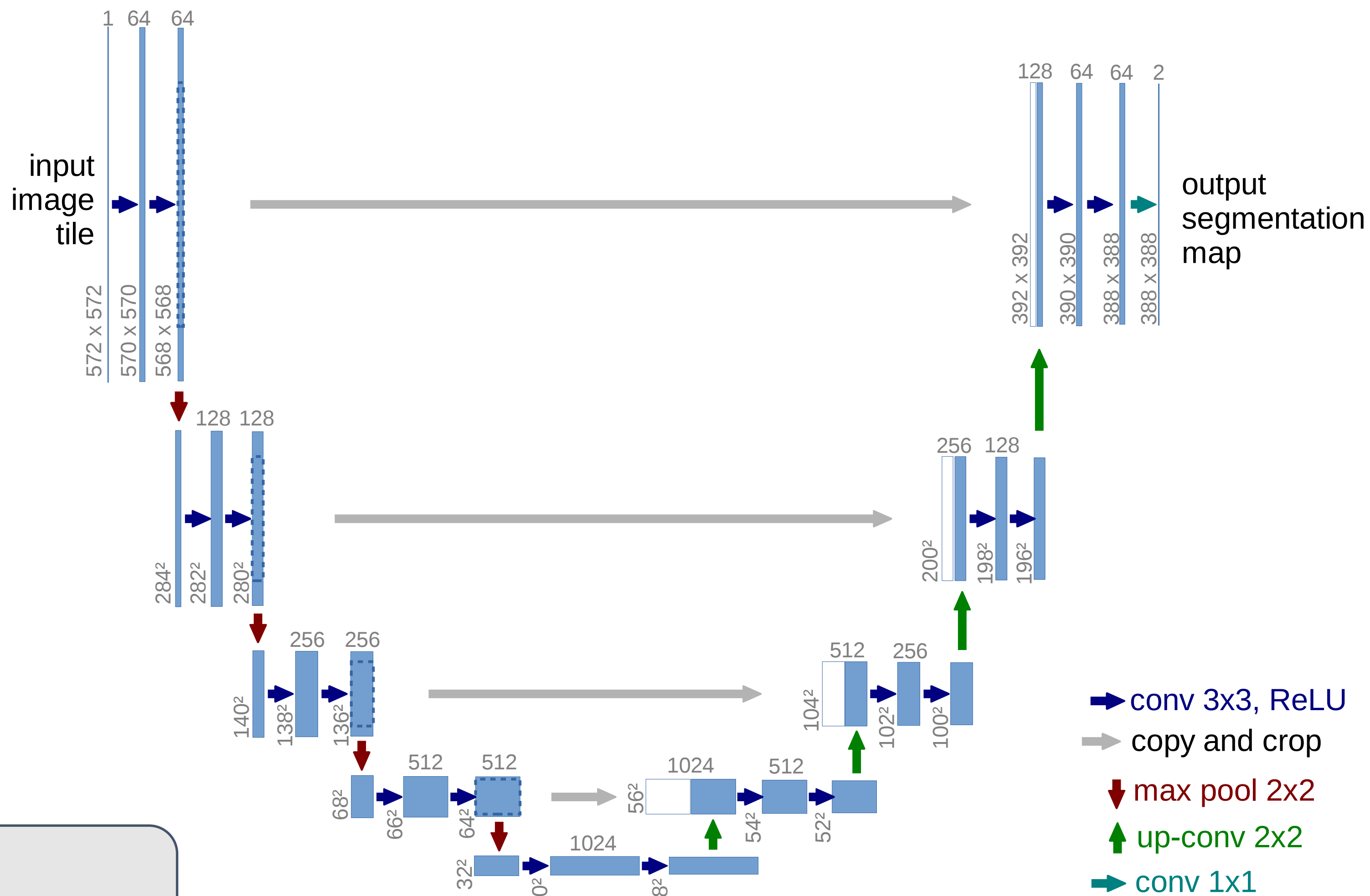
Paper

Deep Residual Learning for Image Recognition

# UNet (2015)

Вход и выход  
одинаковые по форме

Используются skip-connections (как и в ResNet)



Paper

U-Net: Convolutional Networks for Biomedical Image Segmentation

# MobileNet (2017)

Сверточные фильтры малого размера 1x1

Поканальные сверточные фильтры

Встроенный гиперпараметр, регулирующий ширину сети дает возможность удобно искать баланс между сложностью модели и ТОЧНОСТЬЮ

Network Architecture	Number of Parameters	Top-1 Accuracy	Top-5 Accuracy
Xception	22.91M	0.790	0.945
VGG16	138.35M	0.715	0.901
MobileNetV1 (alpha=1, rho=1)	4.20M	0.709	0.899
MobileNetV1 (alpha=0.75, rho=0.85)	2.59M	0.672	0.873
MobileNetV1 (alpha=0.25, rho=0.57)	0.47M	0.415	0.663
MobileNetV2 (alpha=1.4, rho=1)	6.06M	0.750	0.925
MobileNetV2 (alpha=1, rho=1)	3.47M	0.718	0.910
MobileNetV2 (alpha=0.35, rho=0.43)	1.66M	0.455	0.704

 Paper

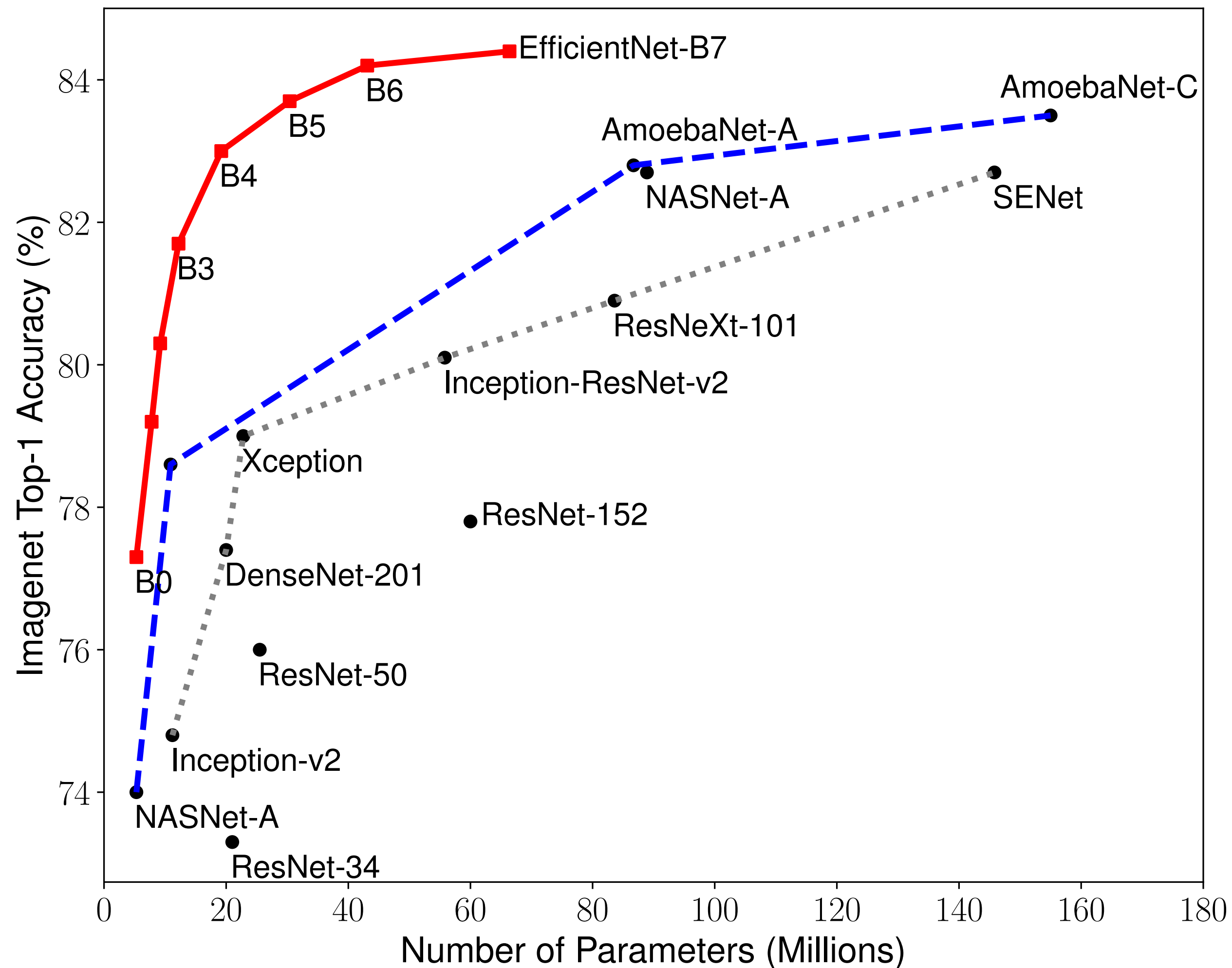
MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications



# EfficientNet (2019)

Компаунд-масштабирование (глубины, ширины и разрешения) позволяет выбрать различные варианты EfficientNet в зависимости от ограничений ресурсов.

Neural architecture search (NAS) исследует пространство гиперпараметров модели для достижения максимальных метрик



 Paper

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks




# Vision transformers



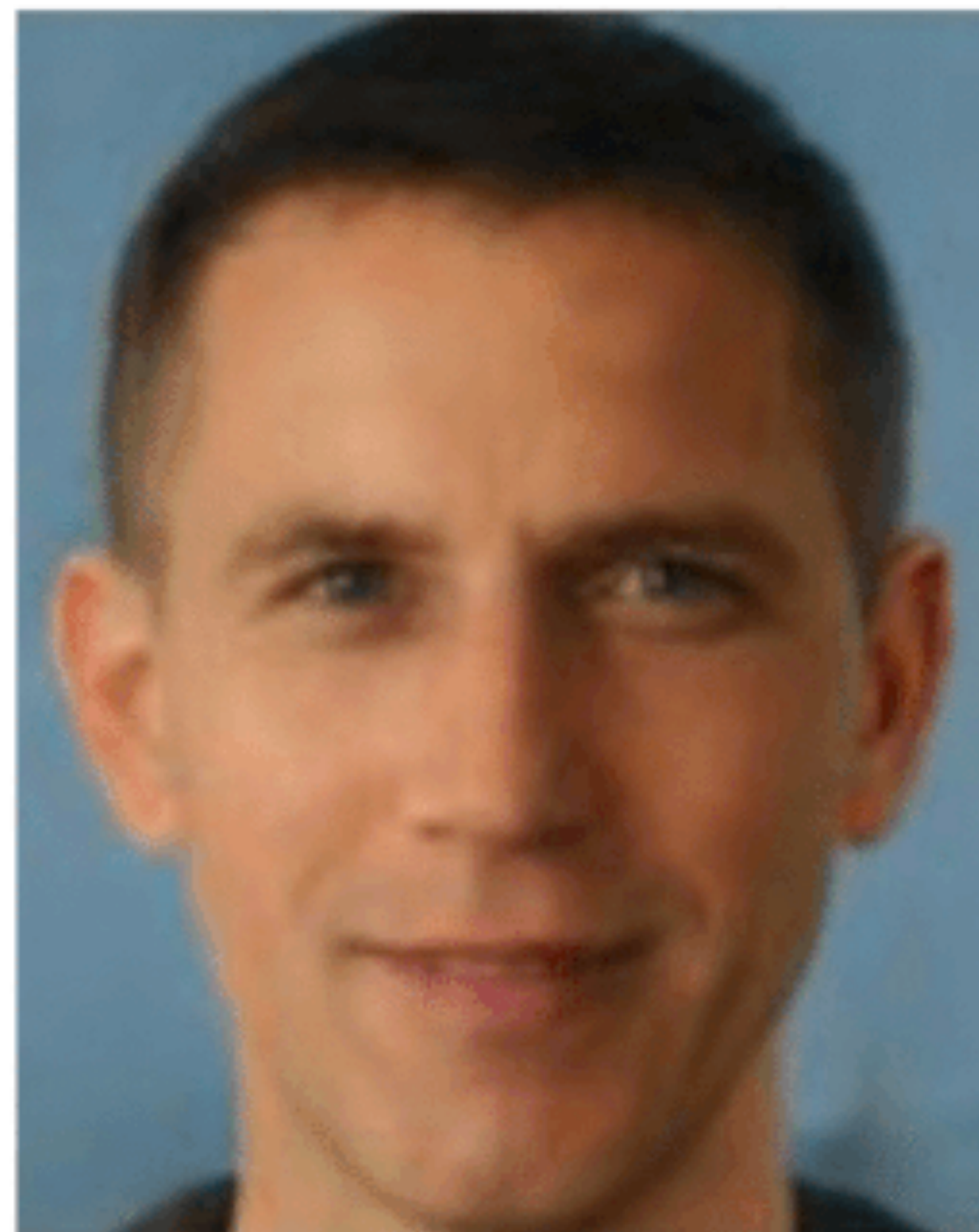
# Проблемы и открытые вопросы



# Работа с неожиданными данными

True Label	COVID-19 (Training Data)	COVID-19 (Unseen Data)	Cat (Unrelated Data)			
						
Model	Prediction	Confidence	Prediction	Confidence	Prediction	Confidence
DNN	COVID-19	99.7%	Non-COVID	75.1%	COVID-19	100%
BNN	COVID-19	95.5%	COVID-19	67.1%	COVID-19	99.8%
Ours	COVID-19	99.9%	COVID-19	69.0%	COVID-19	50.1%

# Предвзятость ~~везде~~ в данных



# Предвзятость везде в данных

Objects Labels Logos Web Properties Safe Search



Screenshot from 2020-04-03 09-51-57.png



Objects Labels Web Properties Safe Search



Screenshot from 2020-04-02 11-51-45.png



# Adversarial attacks



# Энергоэффективность



~10 кВт  
~400 к\$

$10^{11}$  параметров у GPT-3



~10 Вт  
Бесценно

$\sim 10^{10} - 10^{11}$  нейронов  
 $\sim 10^{14}$  связей

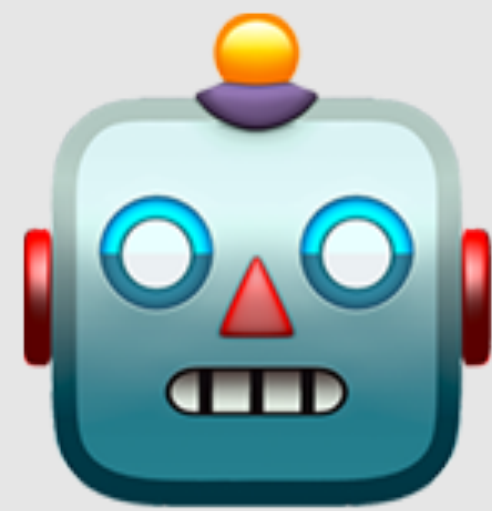


# Выводы



# Выводы

- Нейронные сети в компьютерном зрении стали незаменимым инструментом для решения сложных задач. Сверточные сети - важный этап развития моделей CV
- Для обучения нейронных сетей используют GPU. Для CV расходы на обучение новых моделей часто посильны (в отличие от, например, NLP) не только компаниям с большим бюджетом.
- Модели CV - Inception, ResNet, MobileNet, UNet, EfficientNet, ViT предоставляют лучшие результаты в различных задачах компьютерного зрения, включая распознавание объектов и семантическую сегментацию.
- SotA модели во многих задачах сегодня - Visual Transformers и другие модели на основе механизма внимания.



**ПРАКТИКА**

# Спасибо за внимание! Вопросы?



Пожалуйста,  
поделитесь обратной  
связью о прошедшем  
вебинаре

