

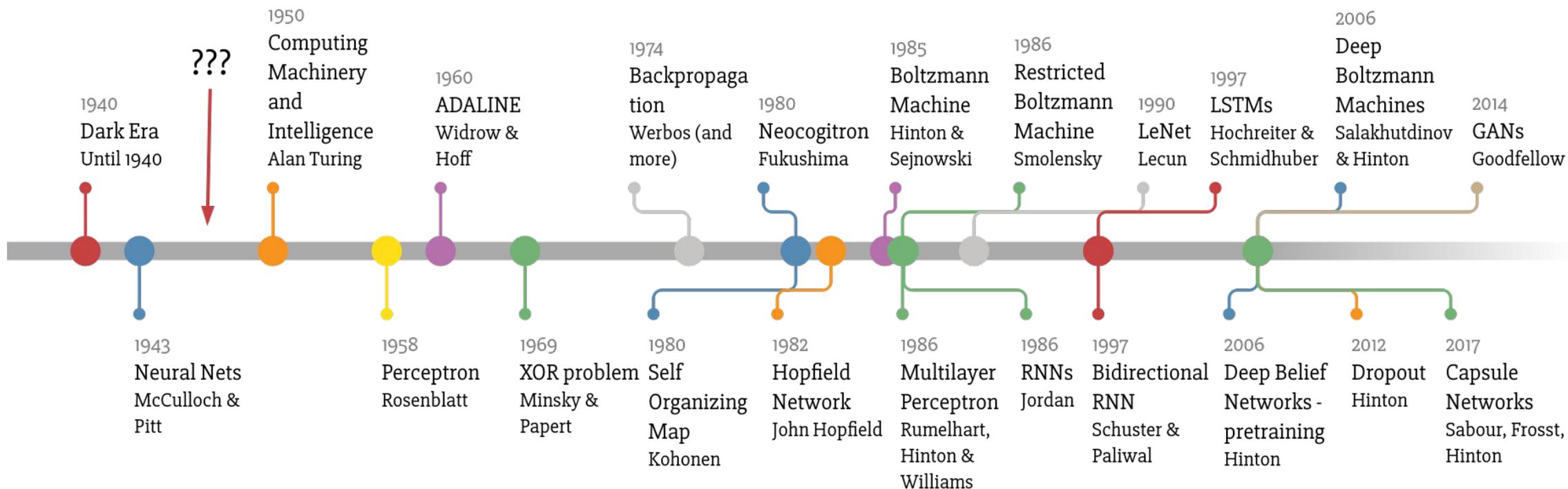


Нейронные сети

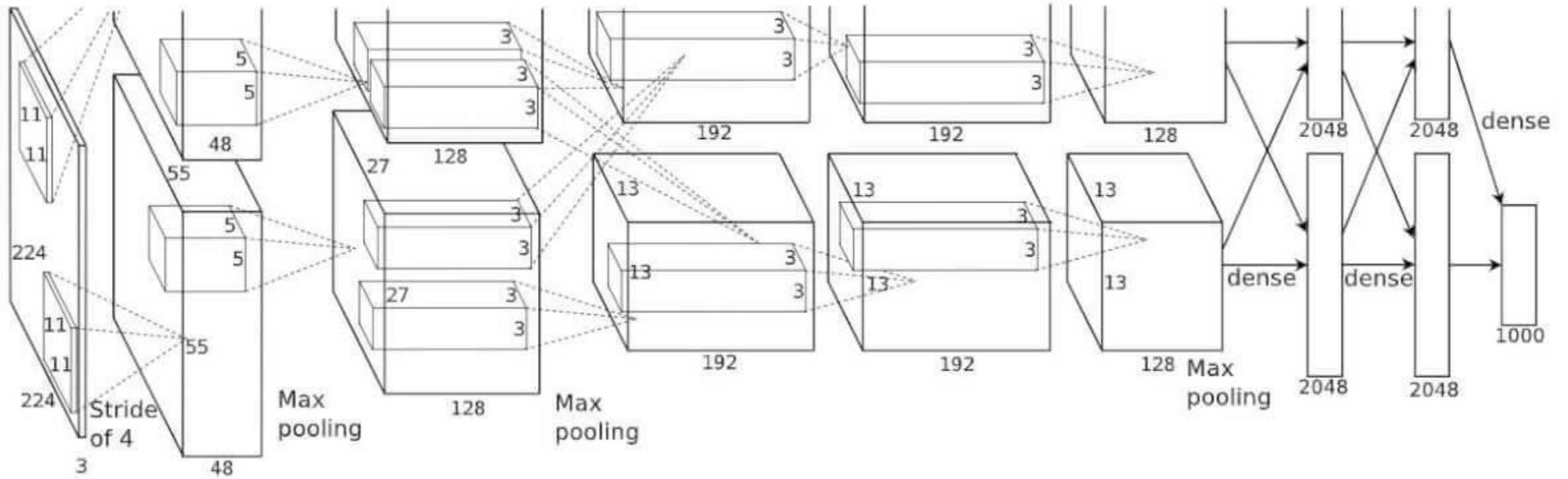


История

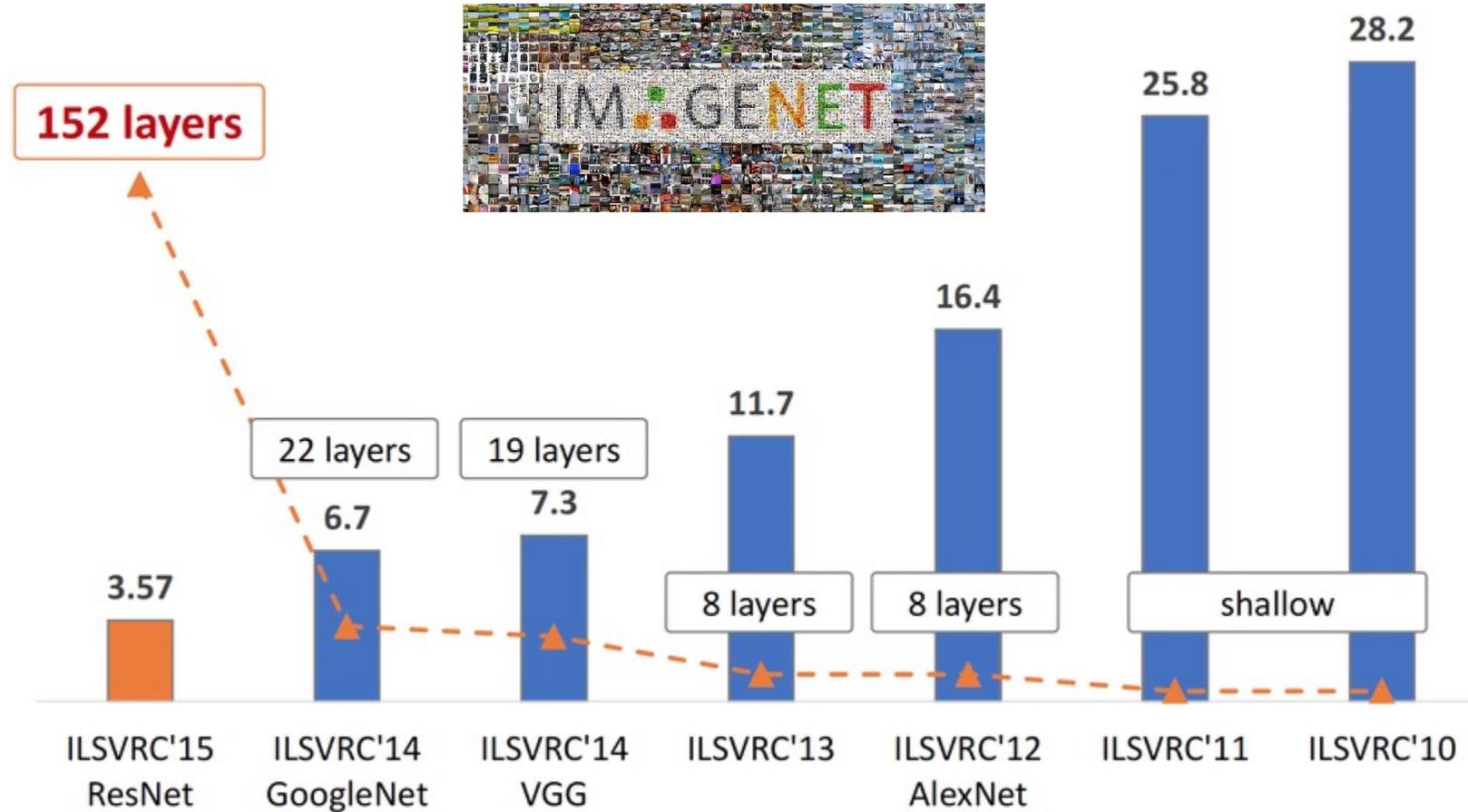
Deep Learning Timeline



AlexNet (2012)



ImageNet



[The evolution of the winning entries on the ImageNet Large Scale Visual Recognition Challenge from 2010 to 2015](#)



Глубокое обучение

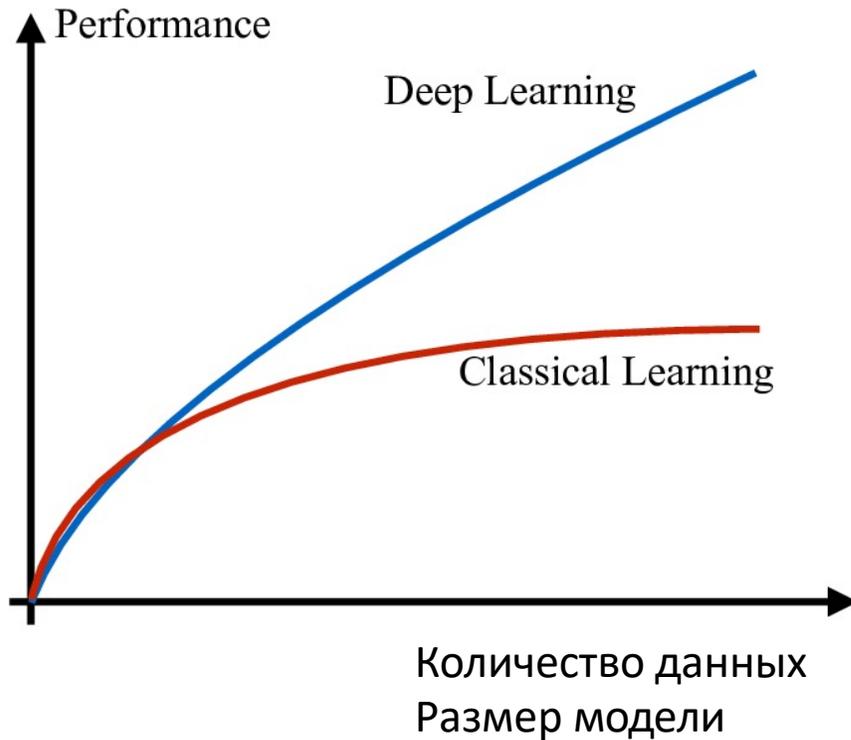
Почему сейчас?

Увеличение объема
доступных данных

Рост вычислительных
мощностей

Улучшение алгоритмов

Много данных



COCO
328K Images



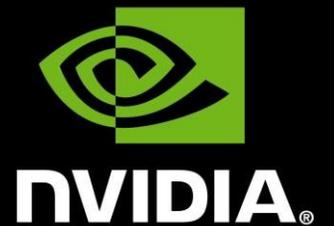
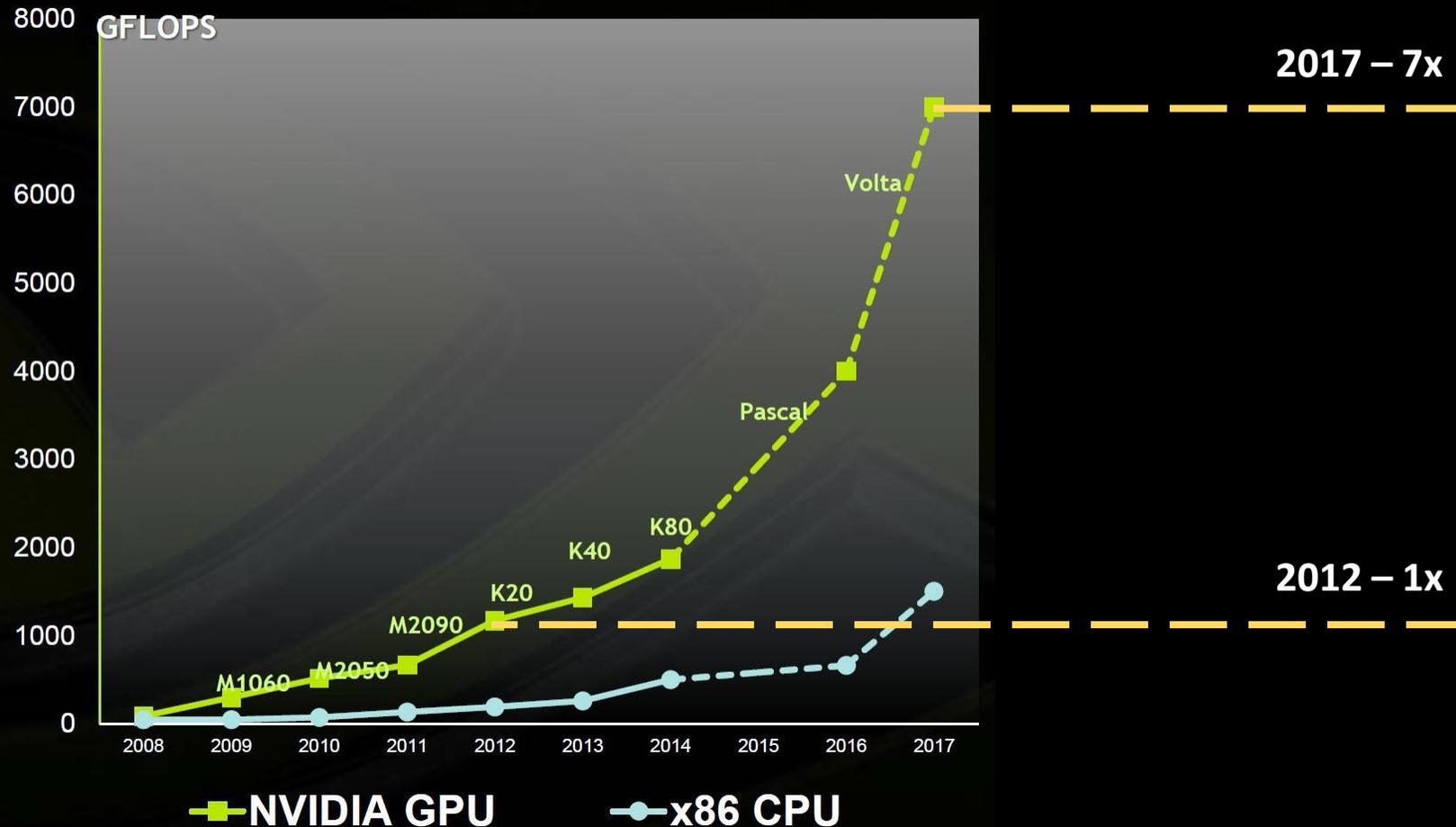
ImageNet
14M Images

Open Images v6+

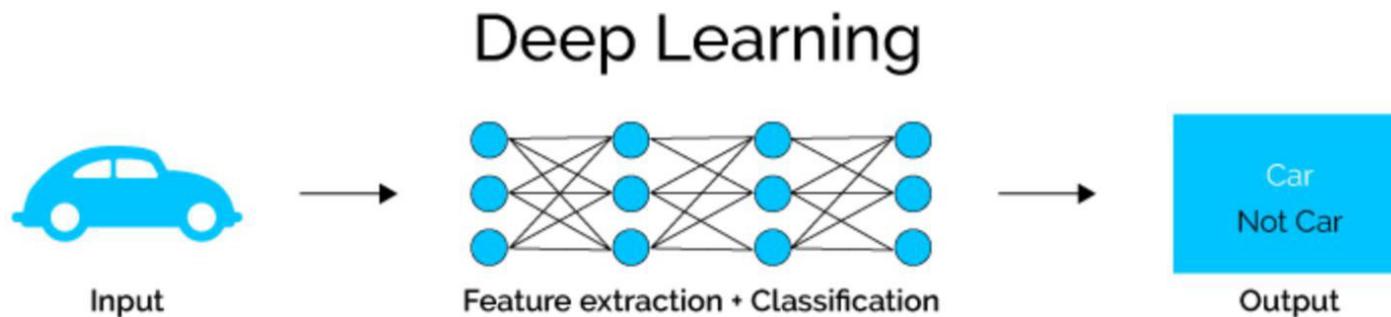
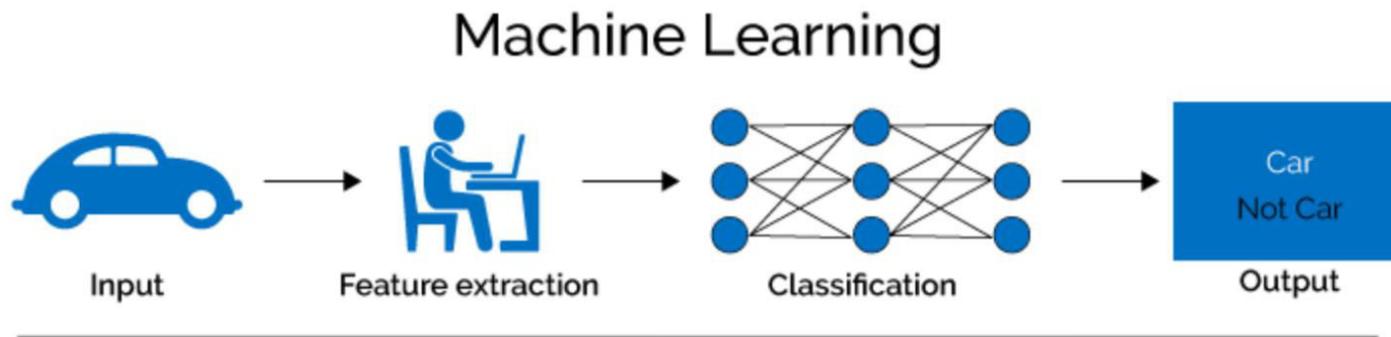
Open Images v6 +
59M Images

Рост вычислительных мощностей - GPU

Peak Double Precision FLOPS



Отличие глубокого обучения от классического машинного обучения



Отличие глубокого обучения от классического машинного обучения

Классификация изображений

$$f(x, w) \rightarrow \{p_1, \dots, p_K\}, x \in \mathbb{R}^n,$$

где p_1, p_2, \dots, p_K – значения уверенности алгоритма о принадлежности изображения тому или иному классу, причем

$$\sum_{k=1}^K p_k = 1, p_k > 0, k = 1, \dots, K.$$

Необходимо самостоятельно построить отображение

$$\phi: \mathbb{R}^{n \times n \times 3} \rightarrow \mathbb{R}^{\varphi}$$

для вычисления значимых характеристик изображения.

На основе полученных характеристик может быть обучен исходный алгоритм:

$$f(\phi(x), w) \rightarrow \{p_1, \dots, p_K\}$$

Методы извлечения признаков



HoG



SIFT



**Optical flow
(Bruhn)**



Задачи

Задачи компьютерного зрения



Классификация +
Локализация



Object Detection

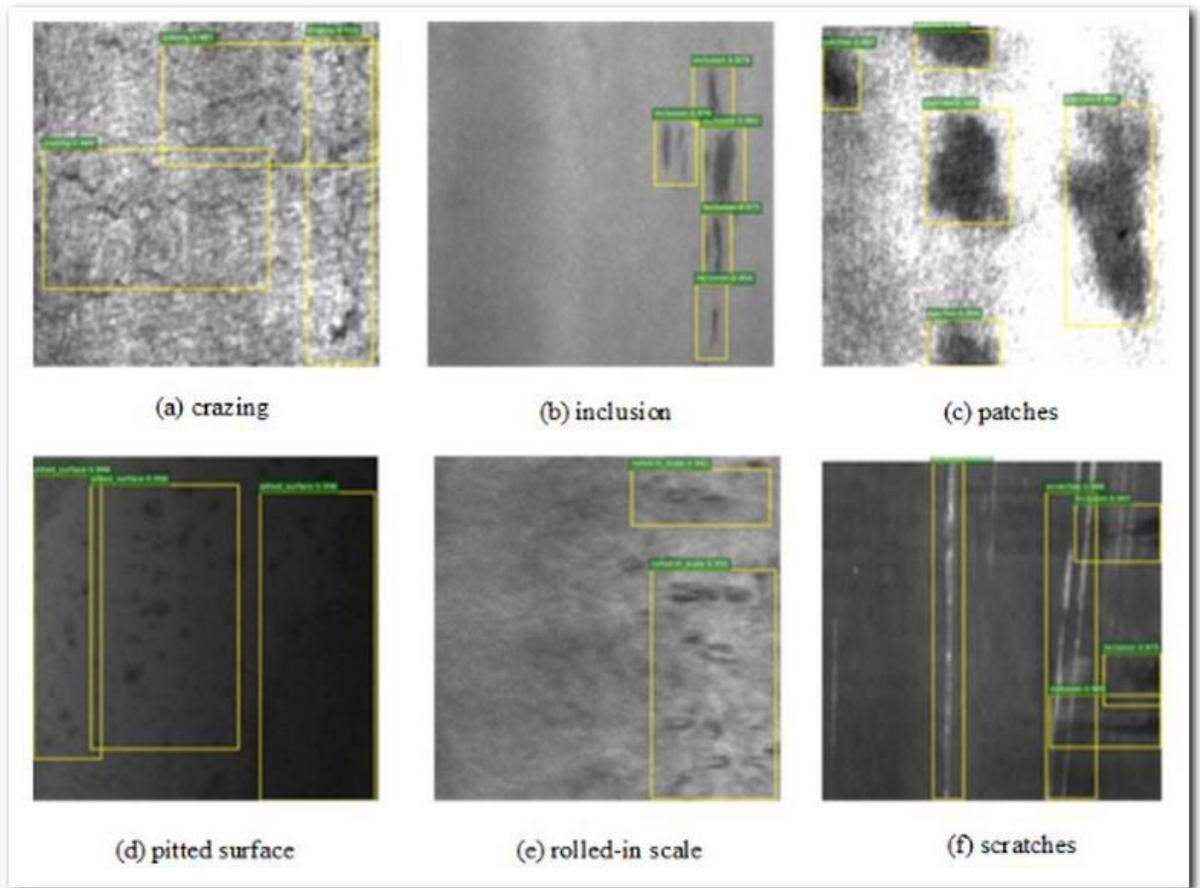
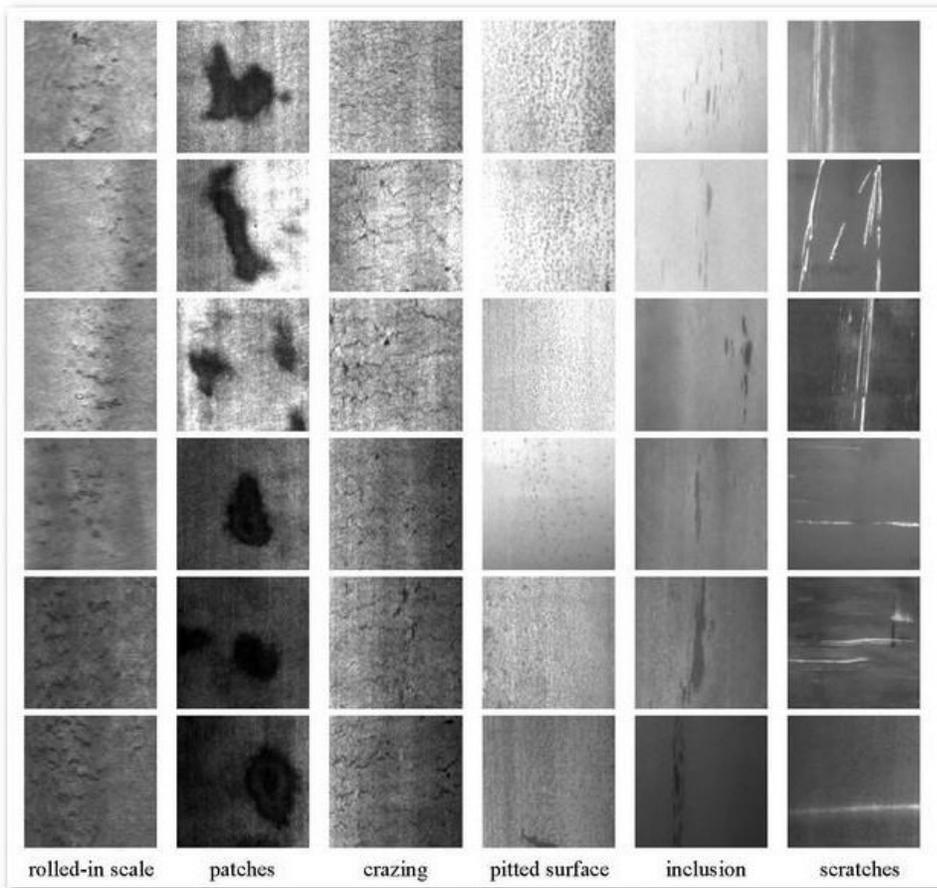


Semantic
Segmentation



Semantic Instance
Segmentation

Оценка качества на производстве



Solving Rubik's Cube with Robot Hand



Автономное вождение



Компьютерное зрение

Classification



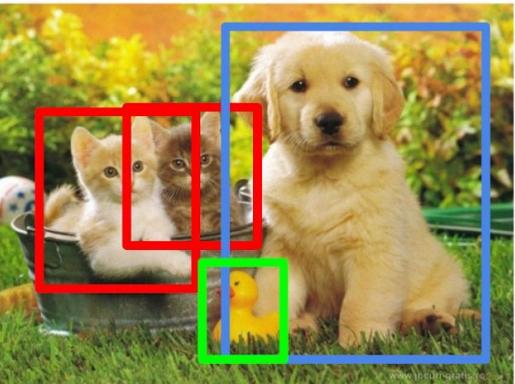
CAT

**Classification
+ Localization**



CAT

Object Detection



CAT, DOG, DUCK

Instance Segmentation

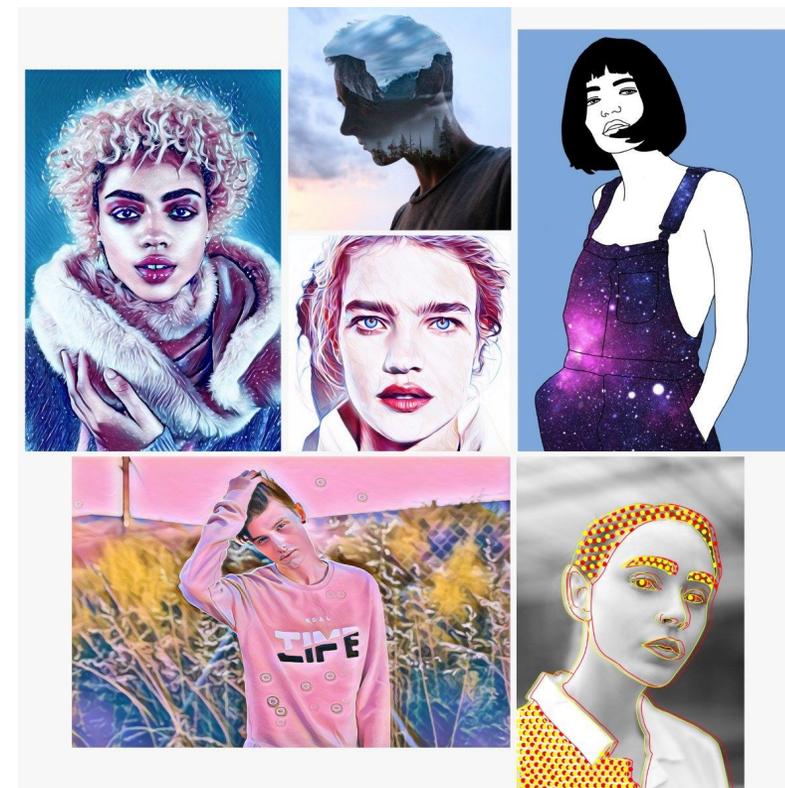
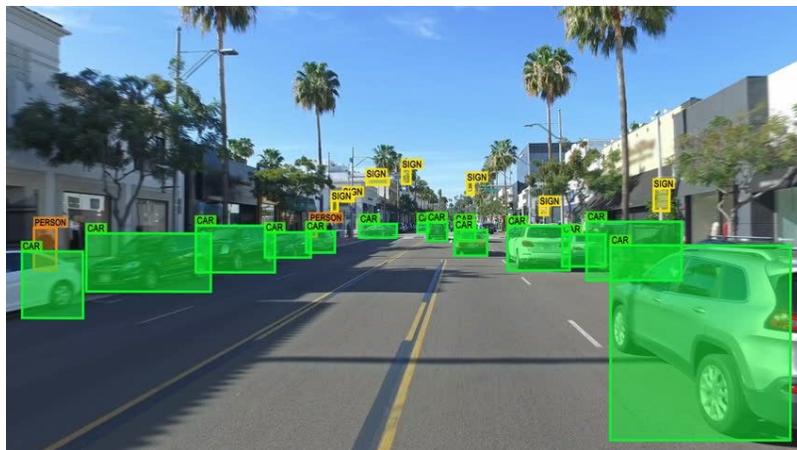


CAT, DOG, DUCK

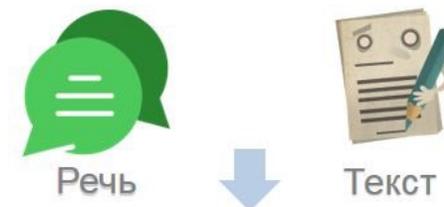
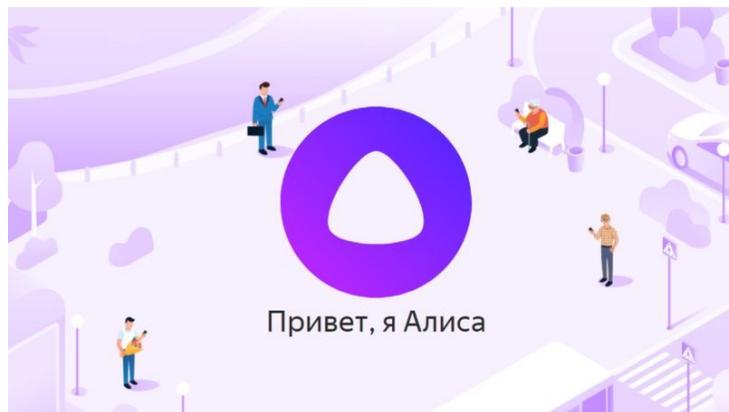
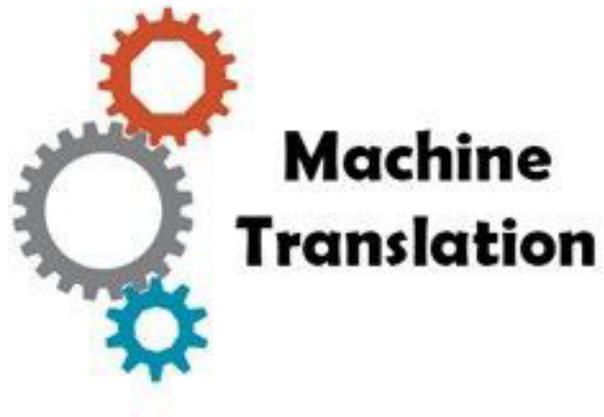
Single object

Multiple objects

Компьютерное зрение



Обработка естественного языка и речи



Первичная обработка

Морфологический анализ

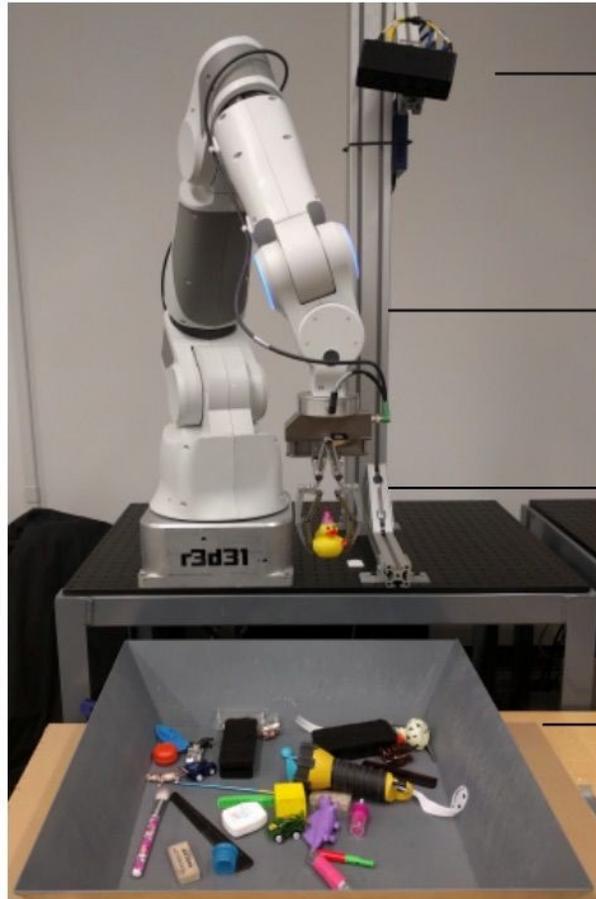
Синтаксический анализ

Семантический анализ

Анализ дискурса

Прагматический анализ

Управление: робототехика



monocular RGB camera

7 DoF robotic manipulator

2-finger gripper

object bin



Управление: игры

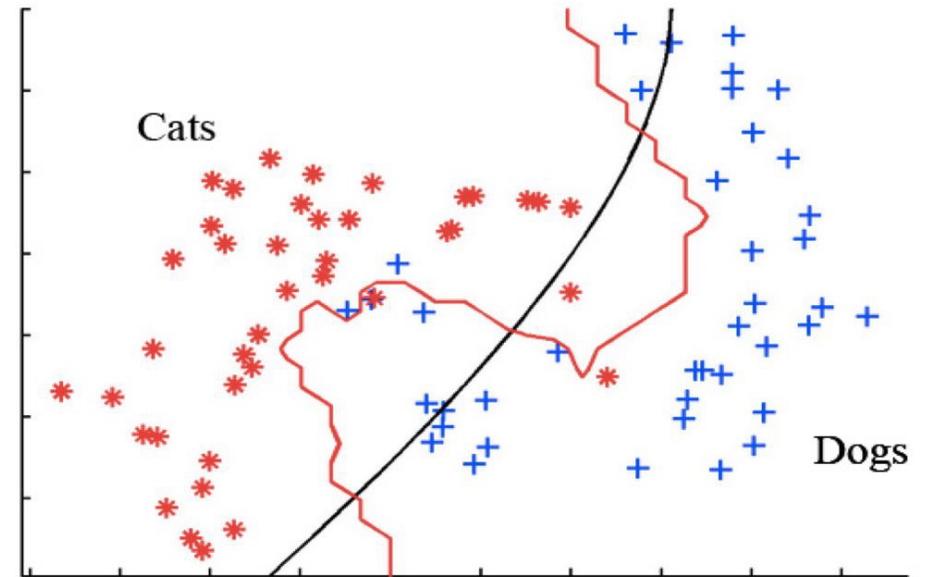
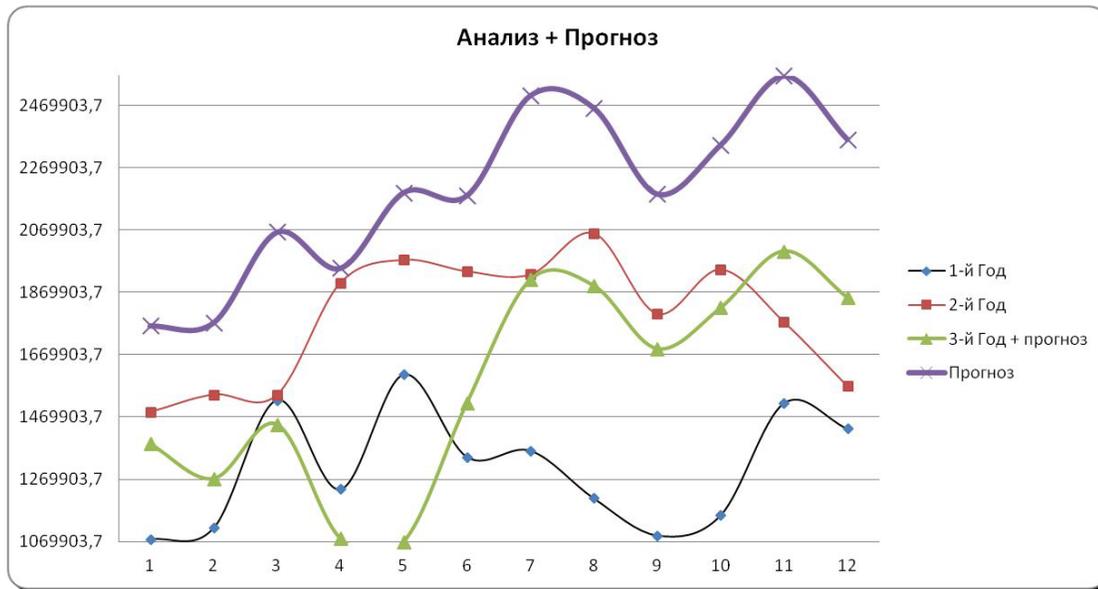


Breakout and Space Invaders, 2 of the 49 Atari games used in the paper



Задачи на структурированных данных

- Все те задачи, которые решают с помощью не нейросетевых ML-алгоритмов



Минус: интерпретируемость



Минус: интерпретируемость

Understanding Neural Networks Through Deep Visualization

Jason Yosinski
Cornell University

YOSINSKI@CS.CORNELL.EDU

Jeff Clune
Anh Nguyen
University of Wyoming

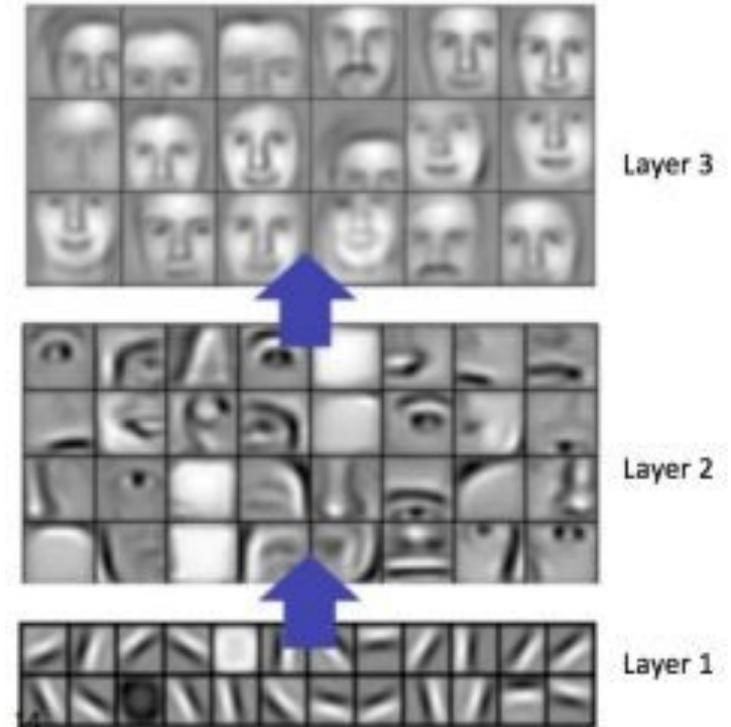
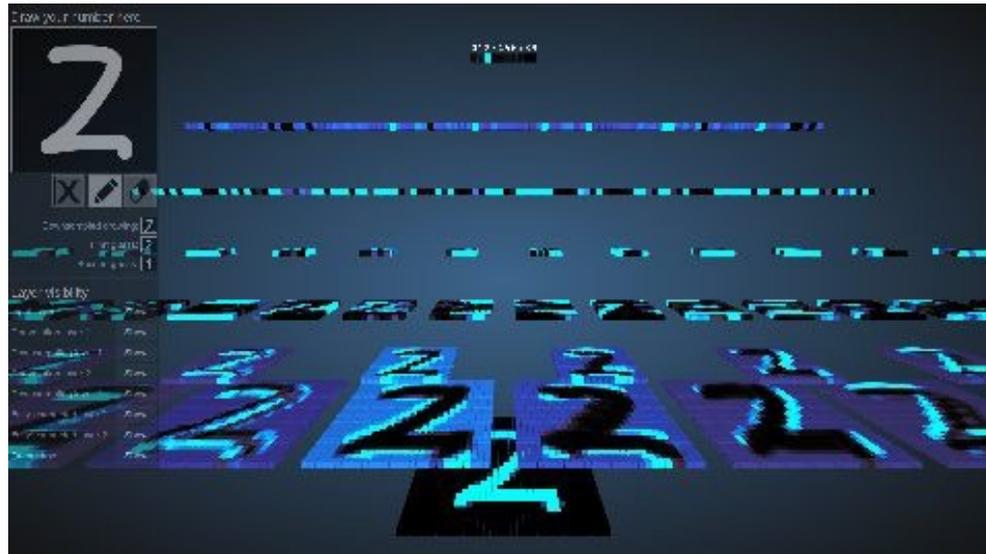
JEFFCLUNE@UWYO.EDU
ANGUYEN8@UWYO.EDU

Thomas Fuchs
Jet Propulsion Laboratory, California Institute of Technology

FUCHS@CALTECH.EDU

Hod Lipson
Cornell University

HOD.LIPSON@CORNELL.EDU

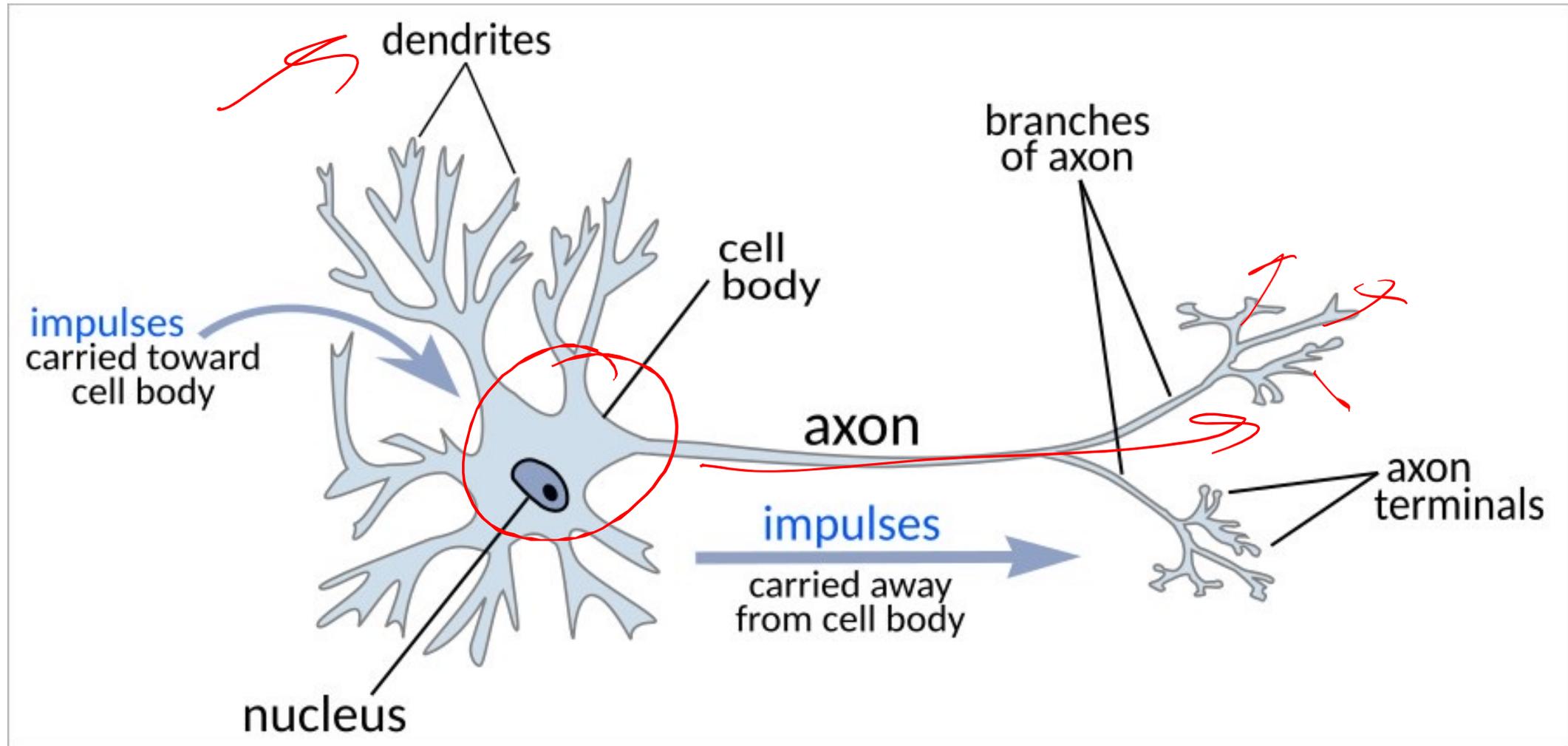


<http://scs.ryerson.ca/~aharley/vis>

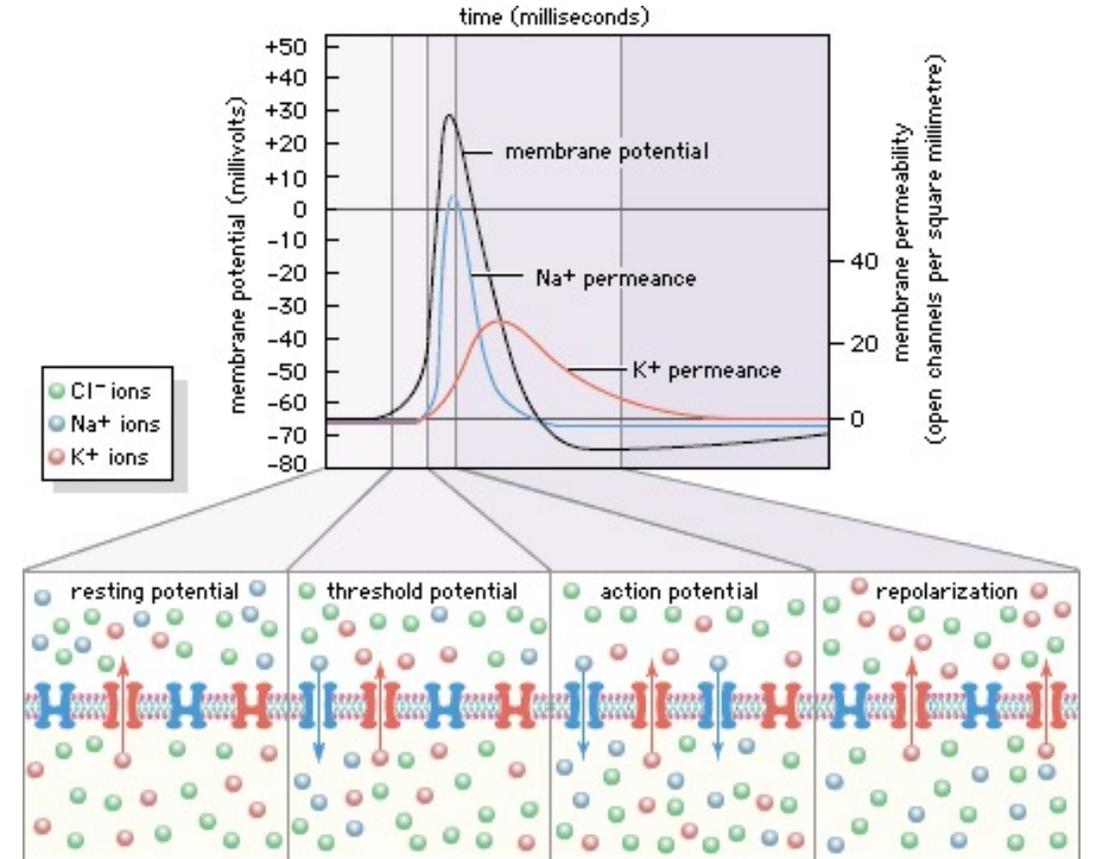
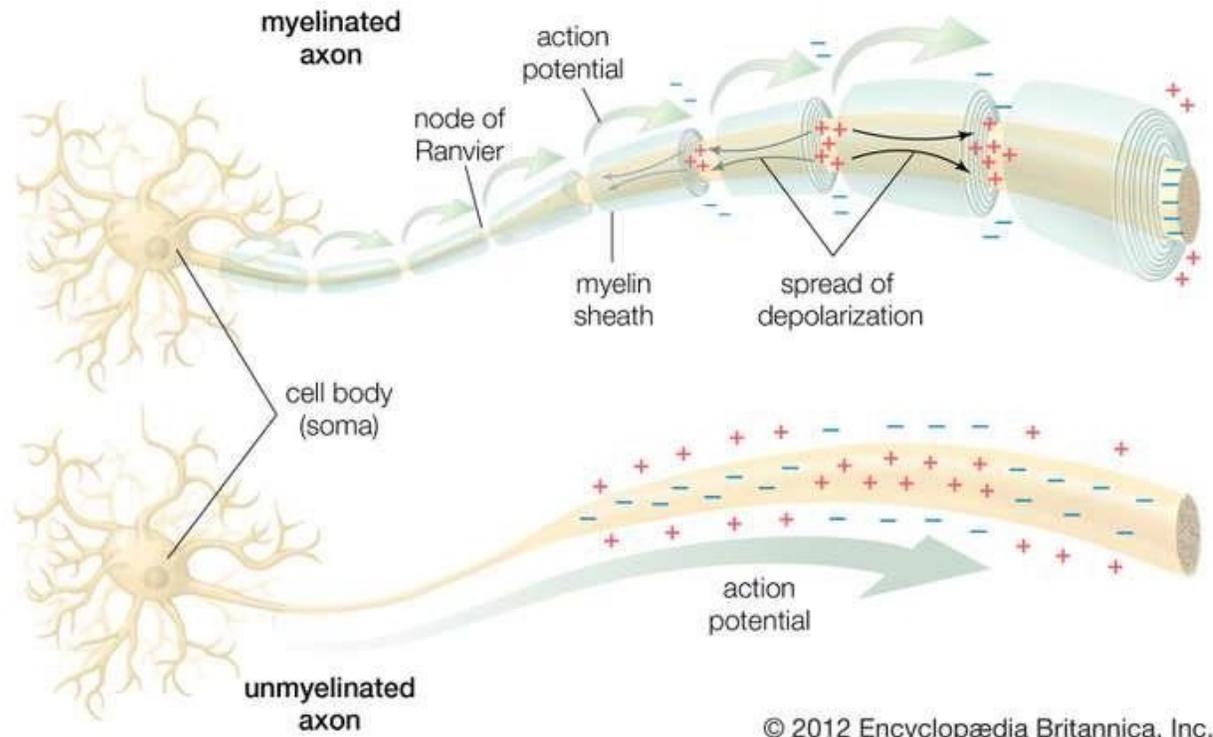


Нейрон

Биологический нейрон



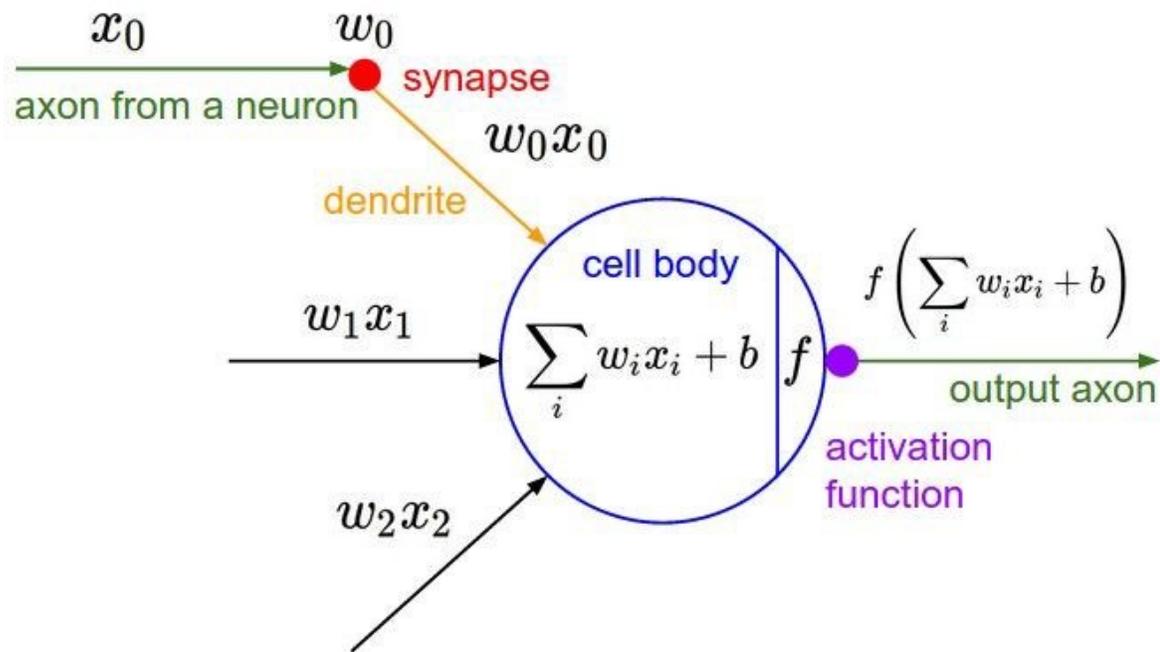
Биологический нейрон



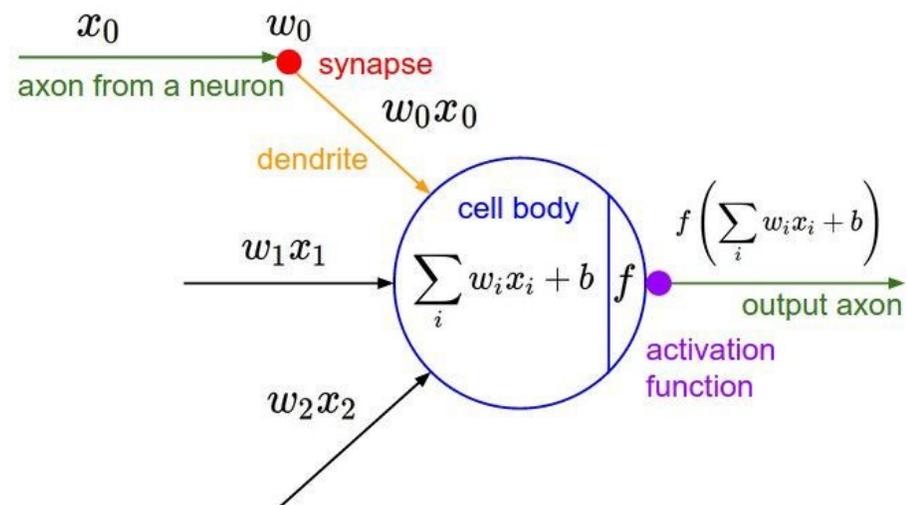
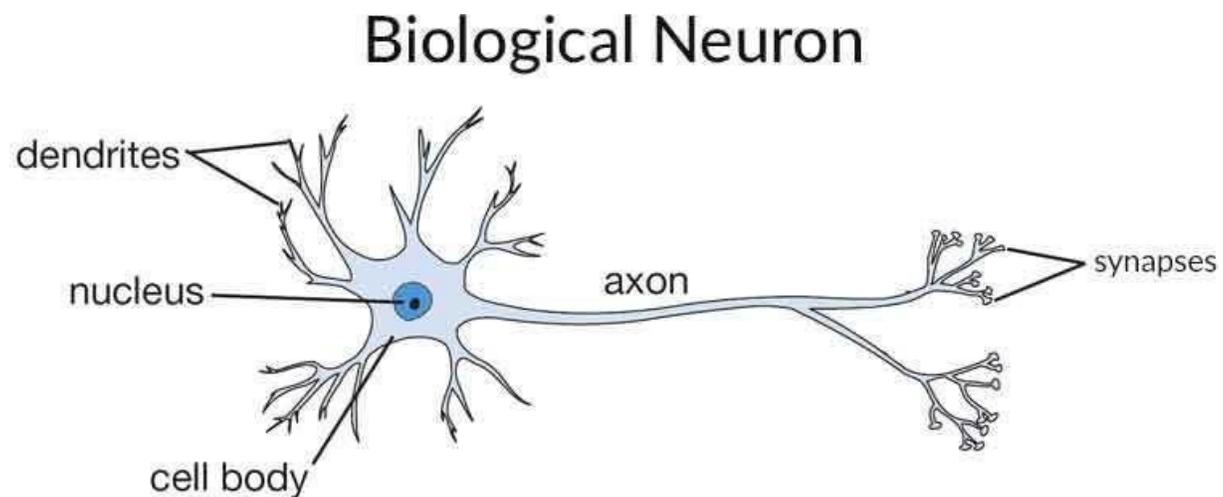
<https://www.britannica.com/science/action-potential>

Искусственный нейрон

- Вход: вектор чисел \mathbf{x} размера $(k, 1)$
- Веса нейрона:
 - вектор чисел \mathbf{w} размера $(k, 1)$
 - скаляр \mathbf{b} (свободный член)
- Функция активации: \mathbf{f}
- Выход: метка класса \mathbf{y} (0 или 1)



Один нейрон

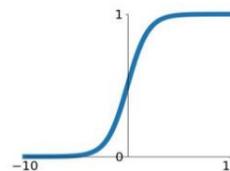


Нейрон

- Модель та же, что и у перцептрона
- Другие функции активации

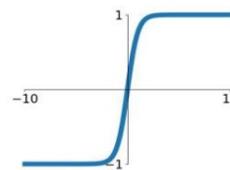
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



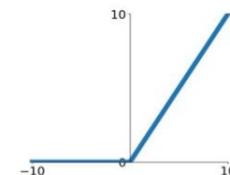
tanh

$$\tanh(x)$$



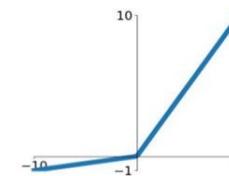
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

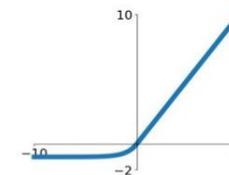


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



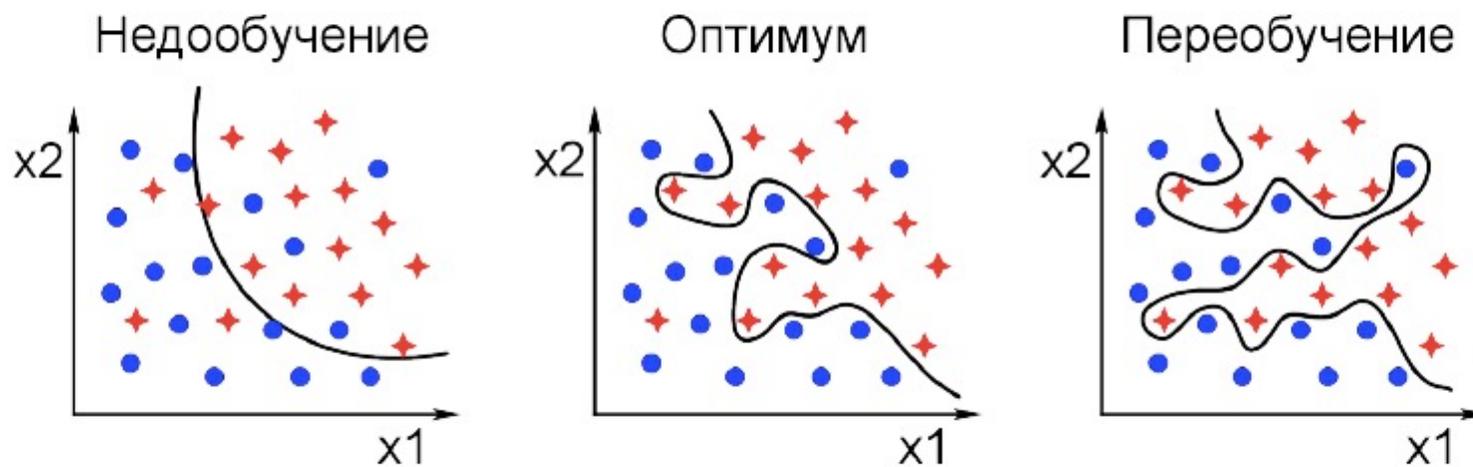
<https://arxiv.org/pdf/1710.05941.pdf>

Нейрон: обучение

- Задан Loss: функция потерь
- Задача обучения с учителем: есть ответы на каждом из объектов
- Оптимизируем градиентным спуском

$$w^{j+1} = w^j - \alpha \frac{\partial \text{Loss}}{\partial w}(w^j)$$

Нейрон: обучение

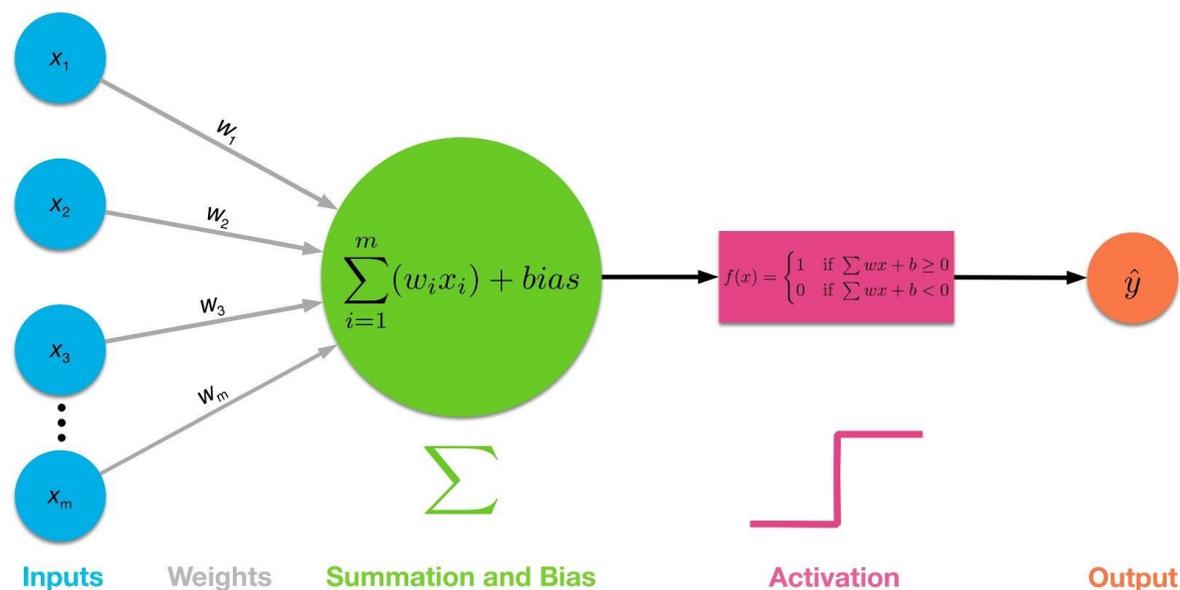




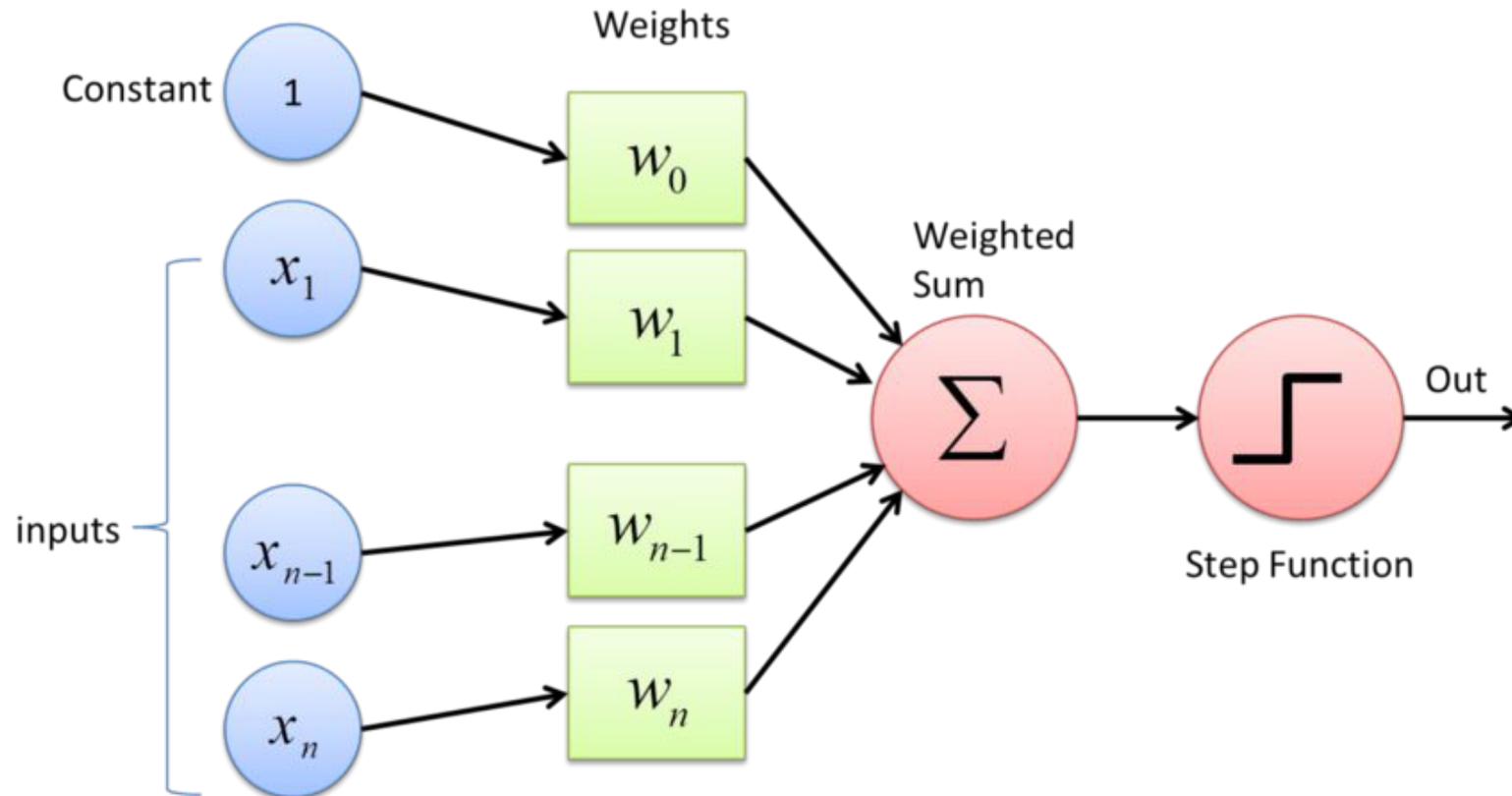
Персептрон

Перцептрон Розенблатта: обучение

- [Метод коррекции ошибки](#) : метод обновления весов
- “Двигаем веса в соответствие с ошибками”
- Ранняя версия градиентного спуска для частного случая
- [Теорема о сходимости перцептрона](#)



Перцептрон (Rosenblat, 1958). Модель.

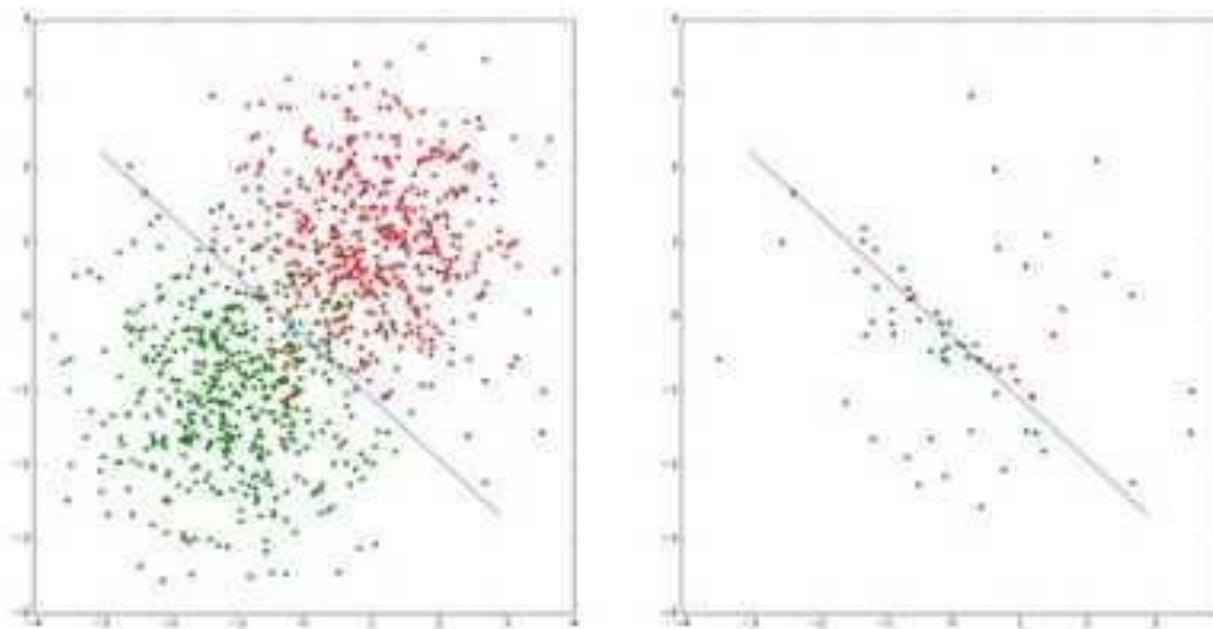


$$\text{pred} = f\left(\sum_{i=1}^n \mathbf{w}_i x_i + \mathbf{w}_0\right), f(s) = \begin{cases} 0 & s \leq 0 \\ 1 & s > 0 \end{cases}$$

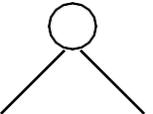
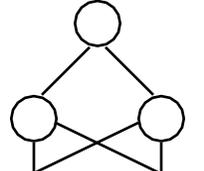
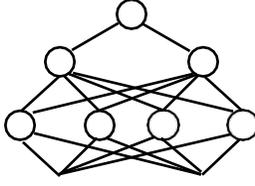
Перцептрон (Rosenblat, 1958). Обучение.

```
while !convergence do
  Pick random  $\mathbf{x} \in P \cup N$  ;
  if  $\mathbf{x} \in P$  and  $\mathbf{w} \cdot \mathbf{x} < 0$  then
     $\mathbf{w} = \mathbf{w} + \mathbf{x}$  ;
  end
  if  $\mathbf{x} \in N$  and  $\mathbf{w} \cdot \mathbf{x} \geq 0$  then
     $\mathbf{w} = \mathbf{w} - \mathbf{x}$  ;
  end
end
```

Перцептрон (Rosenblatt, 1958). Обучение.



Возможности персептронов

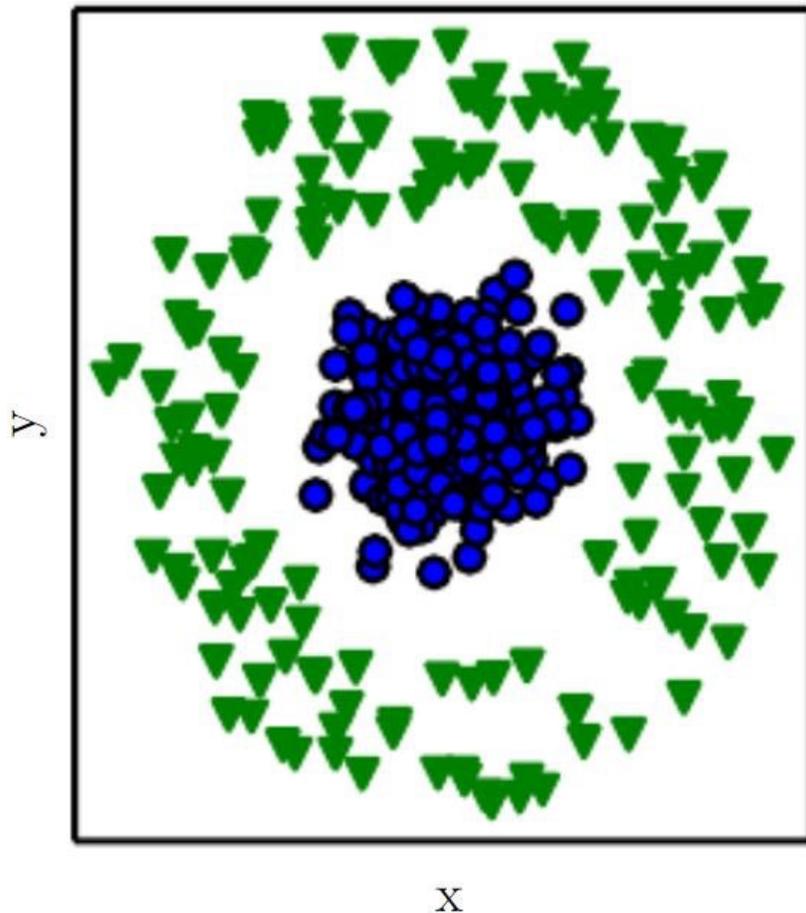
| Архитектура сети | Тип разделяющего правила | XOR задача | Получаемые области | Самый общий возможный вид |
|--|---|------------|--------------------|---------------------------|
| <p>Только выход</p>  | Линейная гиперплоскость | | | |
| <p>однослойный</p>  | Выпуклые открытые области | | | |
| <p>двухслойный</p>  | Произвольные области (сложность ограничена числом нейронов) | | | |

[Multilayer Feedforward Networks are Universal Approximators](#), Hornik, 1988

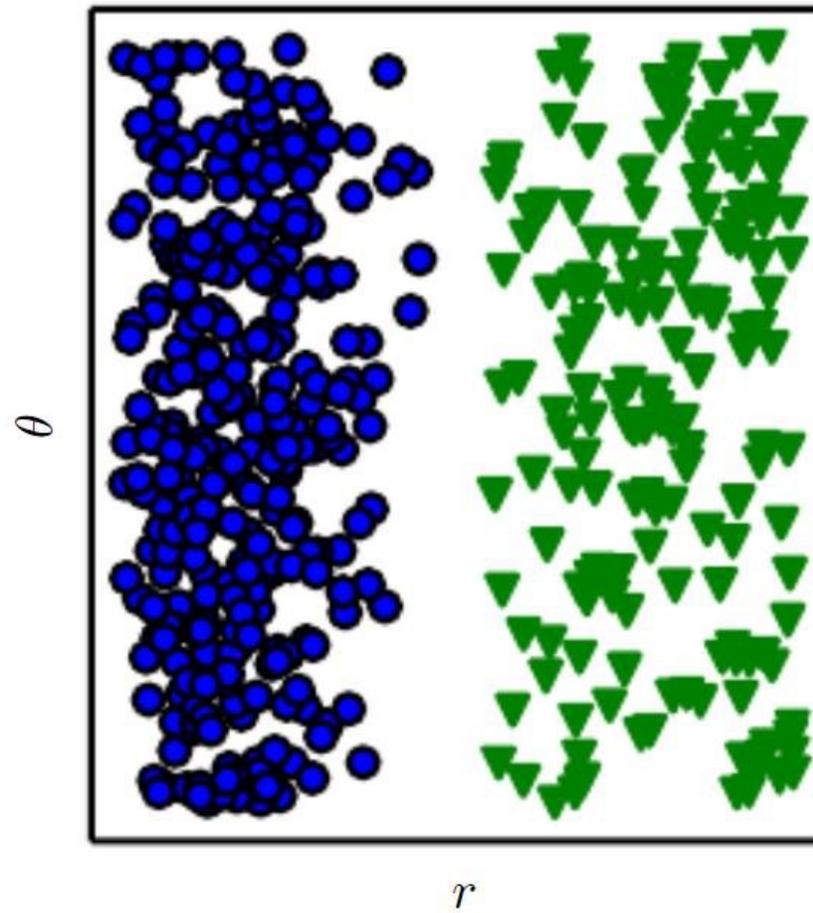
[Approximation capabilities of multilayer feedforward networks](#), Hornik, 1991

Важность правильных представлений

Cartesian coordinates



Polar coordinates

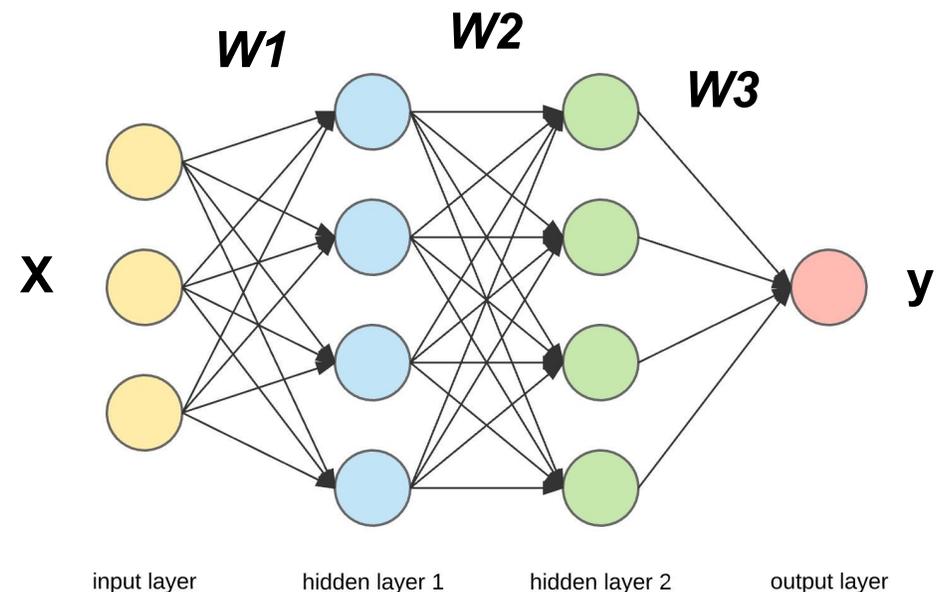




Полносвязные сети

Полносвязная нейросеть

- Вход: числовая матрица X
- Внутри: матрицы параметров (веса нейросети)
- Выход: вектор ответов y
 - метки классов (классификация)
 - вещественные числа (регрессия)



Типы нейронных сетей

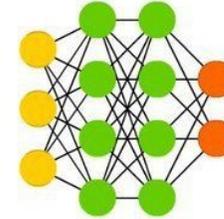
A mostly complete chart of

Neural Networks

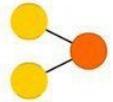
©2016 Fjodor van Veen - asimovinstitute.org

- Backfed Input Cell
- Input Cell
- △ Noisy Input Cell
- Hidden Cell
- Probablistic Hidden Cell
- △ Spiking Hidden Cell
- Output Cell
- Match Input Output Cell
- Recurrent Cell
- Memory Cell
- △ Different Memory Cell
- Kernel
- Convolution or Pool

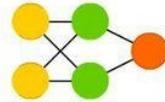
Deep Feed Forward (DFF)



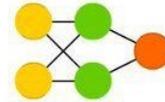
Perceptron (P)



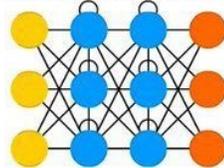
Feed Forward (FF)



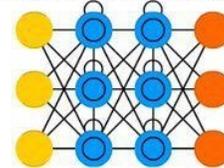
Radial Basis Network (RBF)



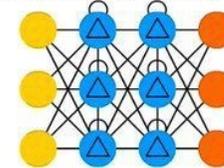
Recurrent Neural Network (RNN)



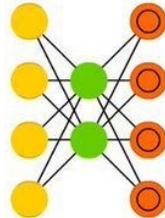
Long / Short Term Memory (LSTM)



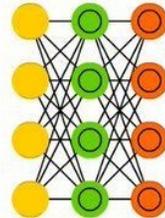
Gated Recurrent Unit (GRU)



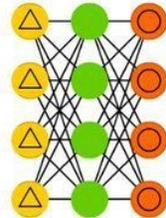
Auto Encoder (AE)



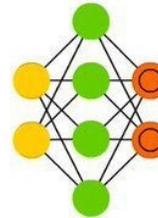
Variational AE (VAE)



Denosing AE (DAE)



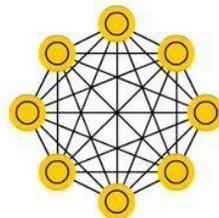
Sparse AE (SAE)



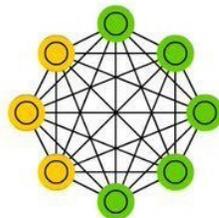
Markov Chain (MC)



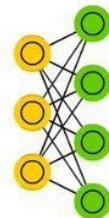
Hopfield Network (HN)



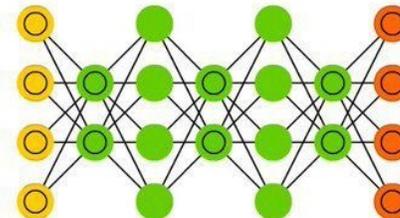
Boltzmann Machine (BM)



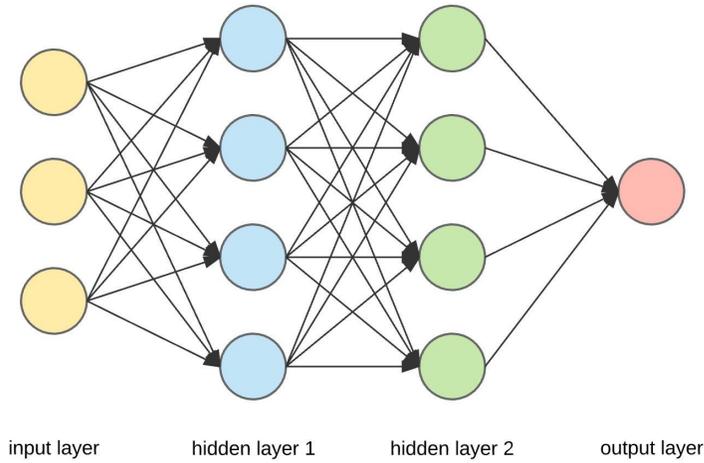
Restricted BM (RBM)



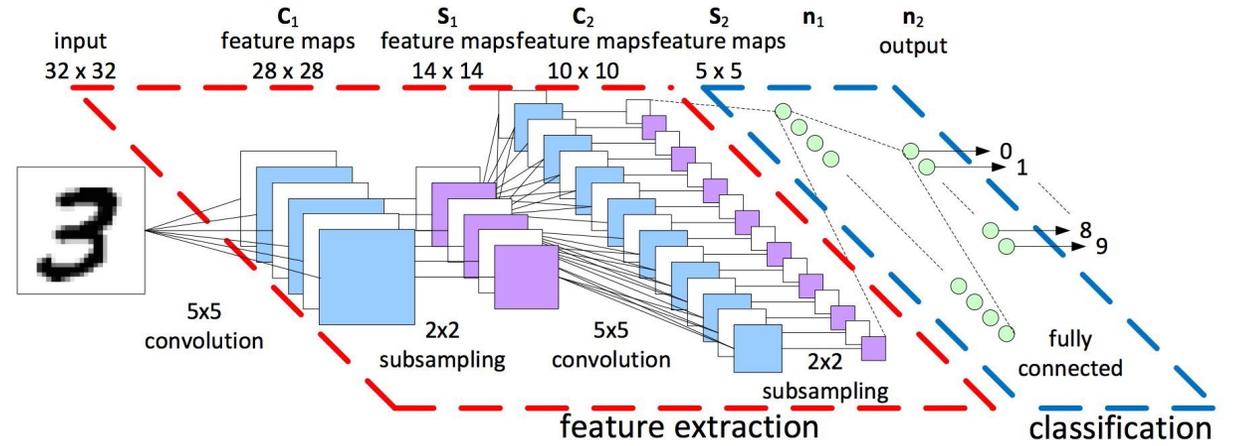
Deep Belief Network (DBN)



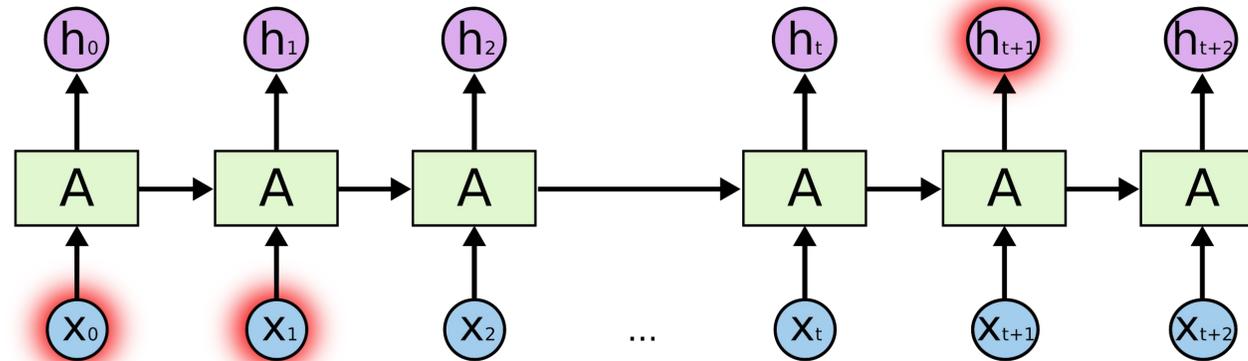
Типы нейронных сетей



Dense =
Fully Connected =
Multi Layered Perceptron



Convolutional Neural Networks

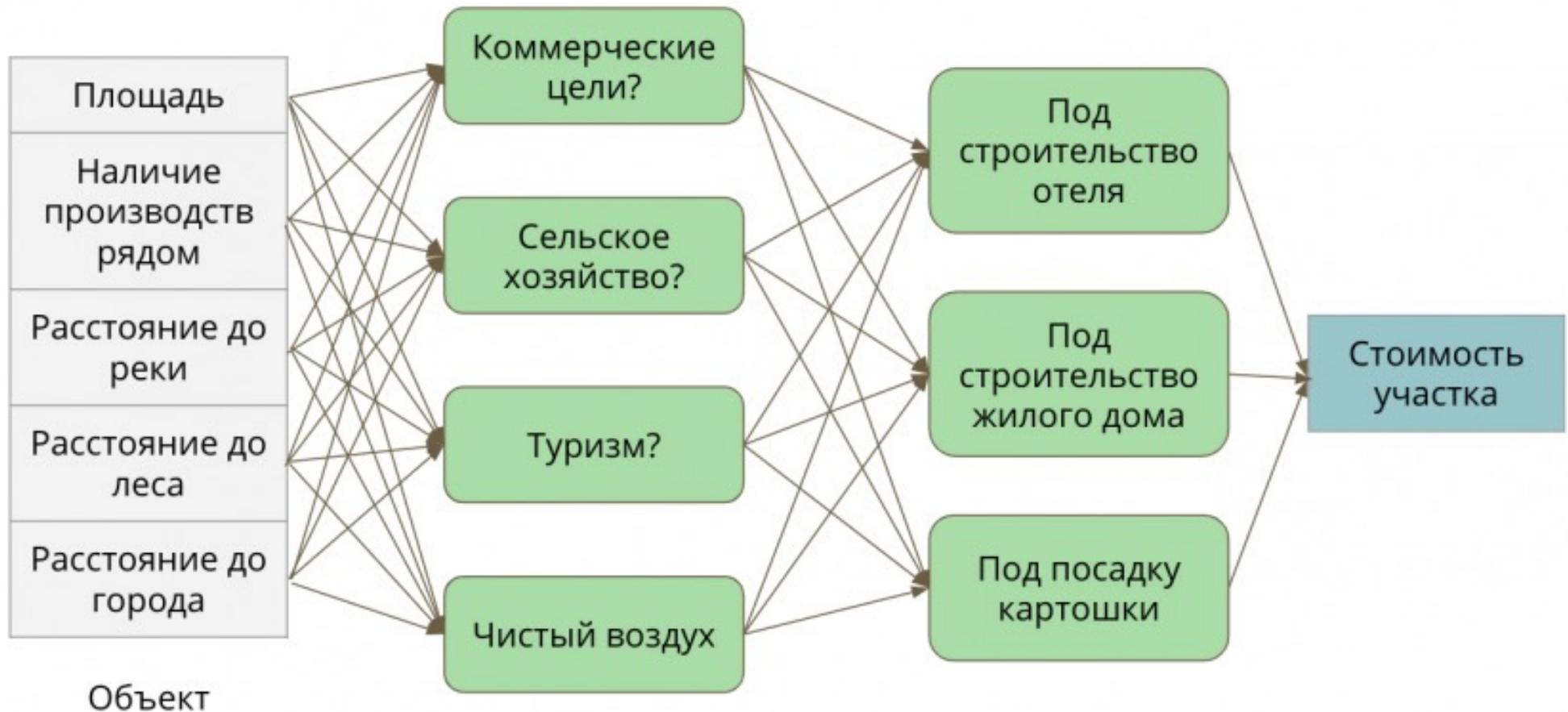


Recurrent Neural Networks

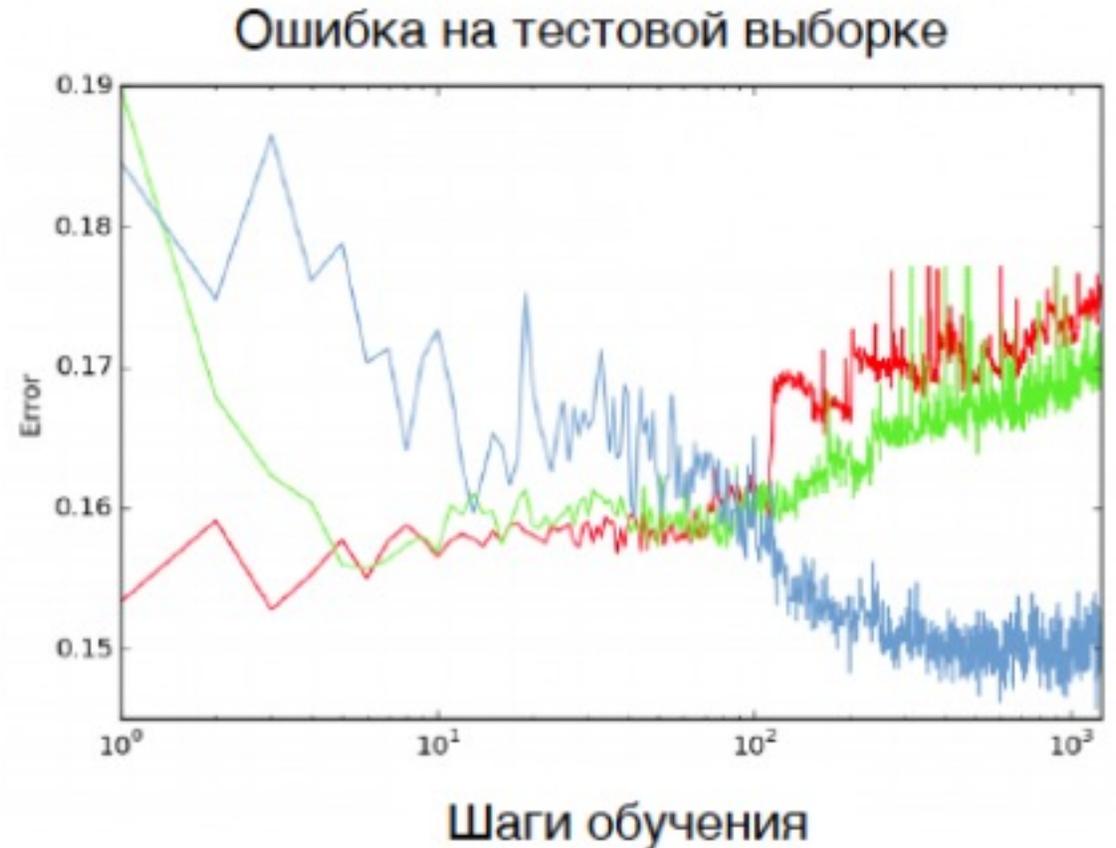
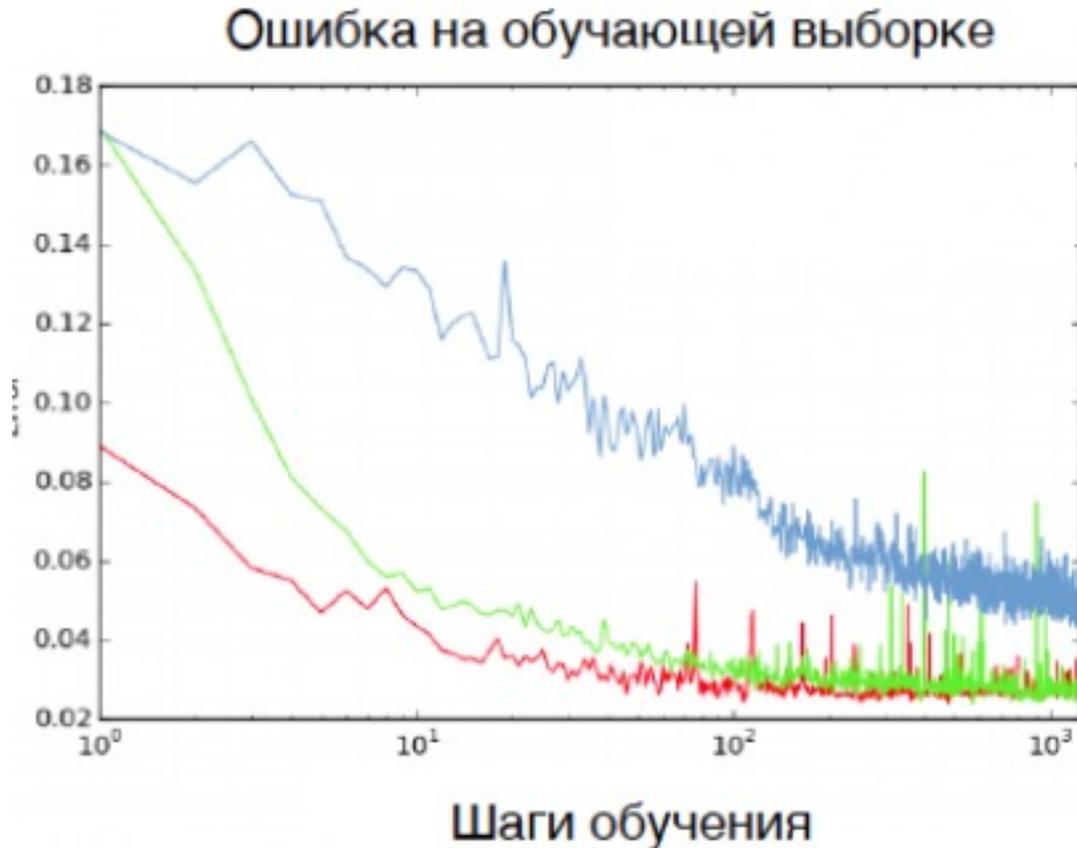
Линейная модель

Цена участка $\approx 2,5 \cdot \text{площадь} + 4,1 \cdot (\text{есть ли рядом река}) + 10 \cdot (\text{есть ли удобная дорога})$

Трехслойная нейронная сеть

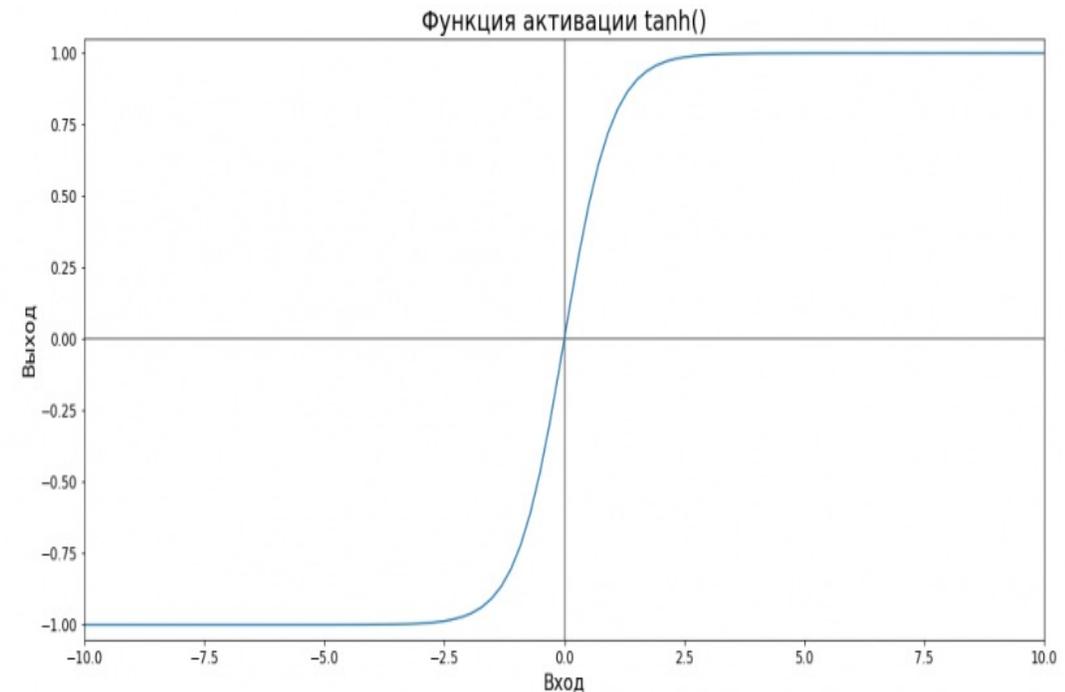
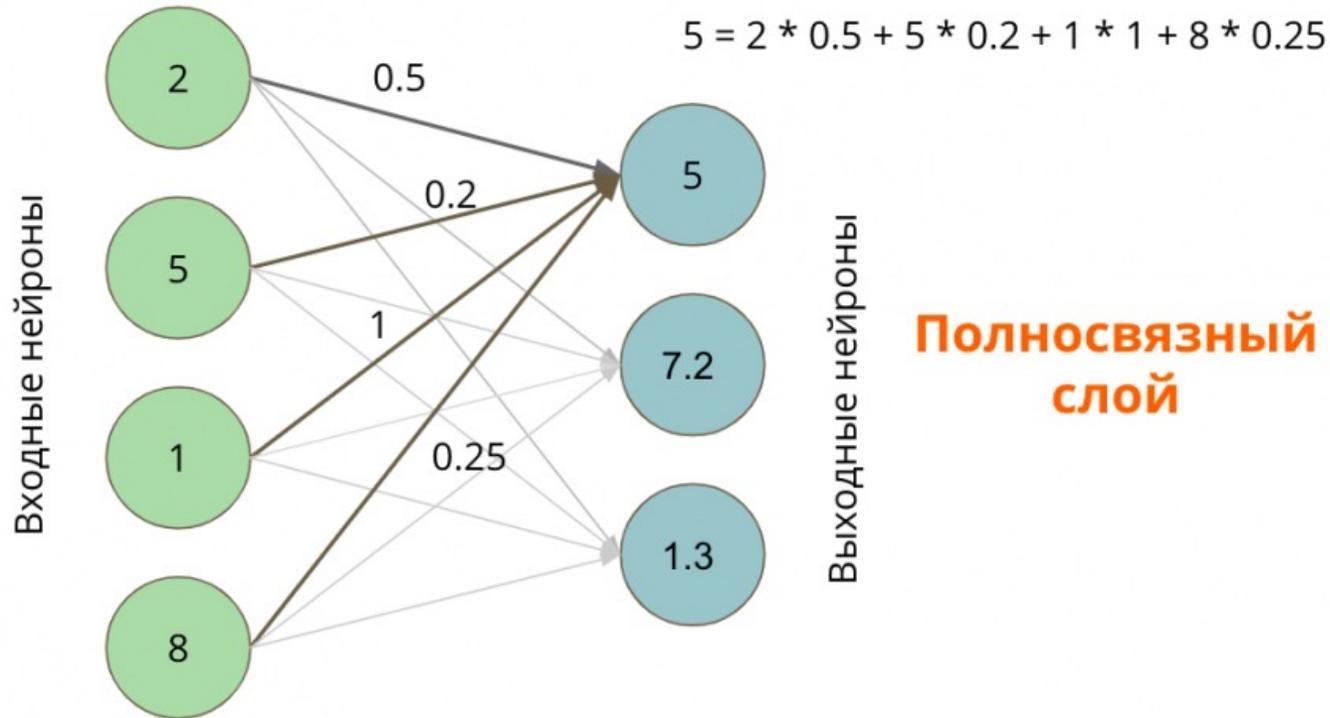


Обучение нейронной сети



На графике приведен пример изменения ошибки в процессе обучения трех различных нейросетей в задаче регрессии: **по оси x** — шаги обновления весов, **по оси y** — ошибка.

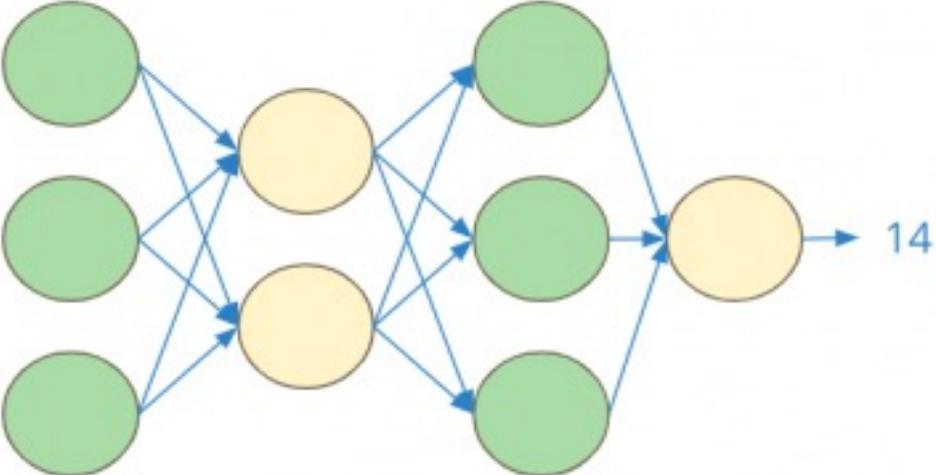
Полносвязный слой и слой нелинейности



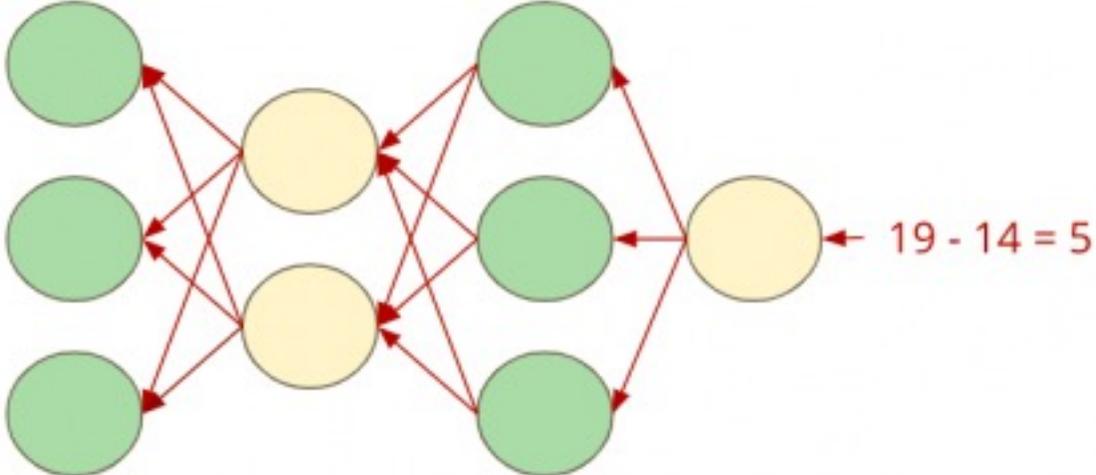
В каждом нейроне хранится число. Каждый нейрон справа (выходной нейрон) вычисляется как сумма всех нейронов слева (входных нейронов), умноженных на веса (линейная модель).

На практике чаще всего используют нелинейность под названием ReLU: у нее всего один излом.

Алгоритм обратного распространения ошибки

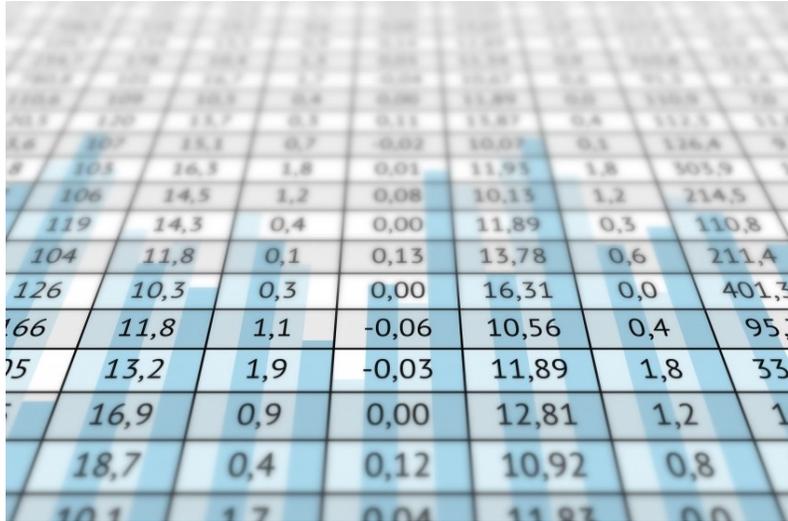


Проход вперед

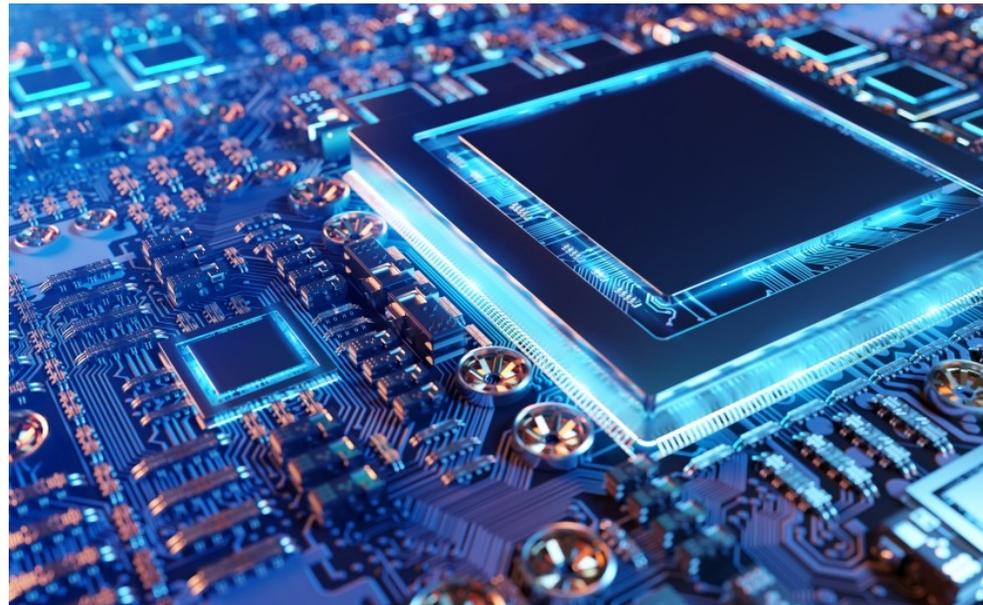


Проход назад

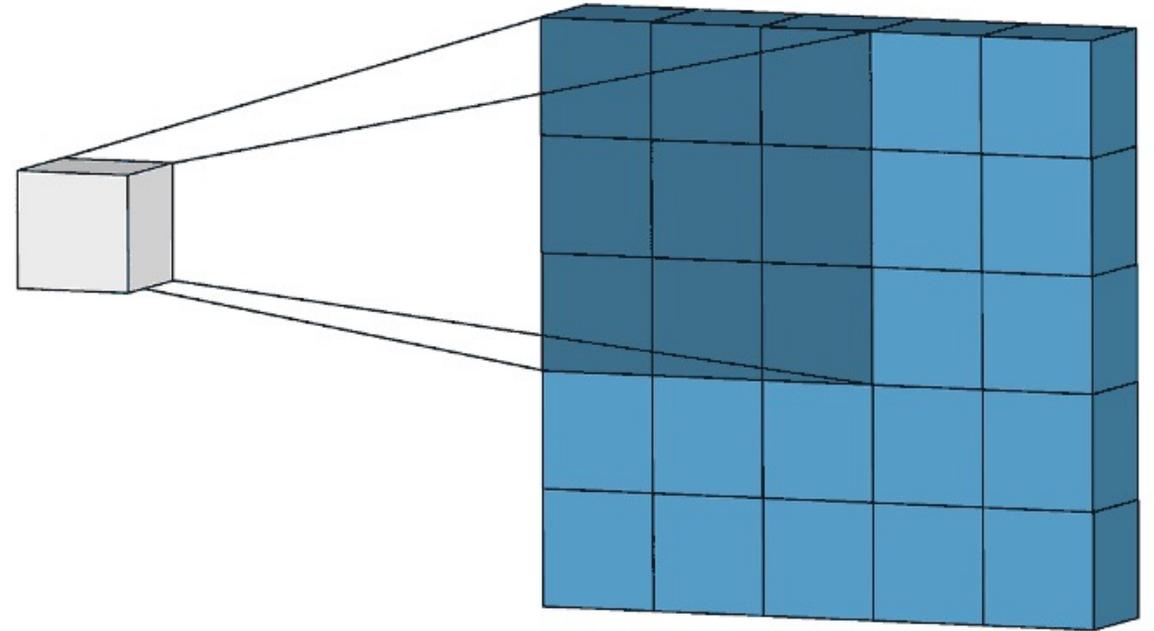
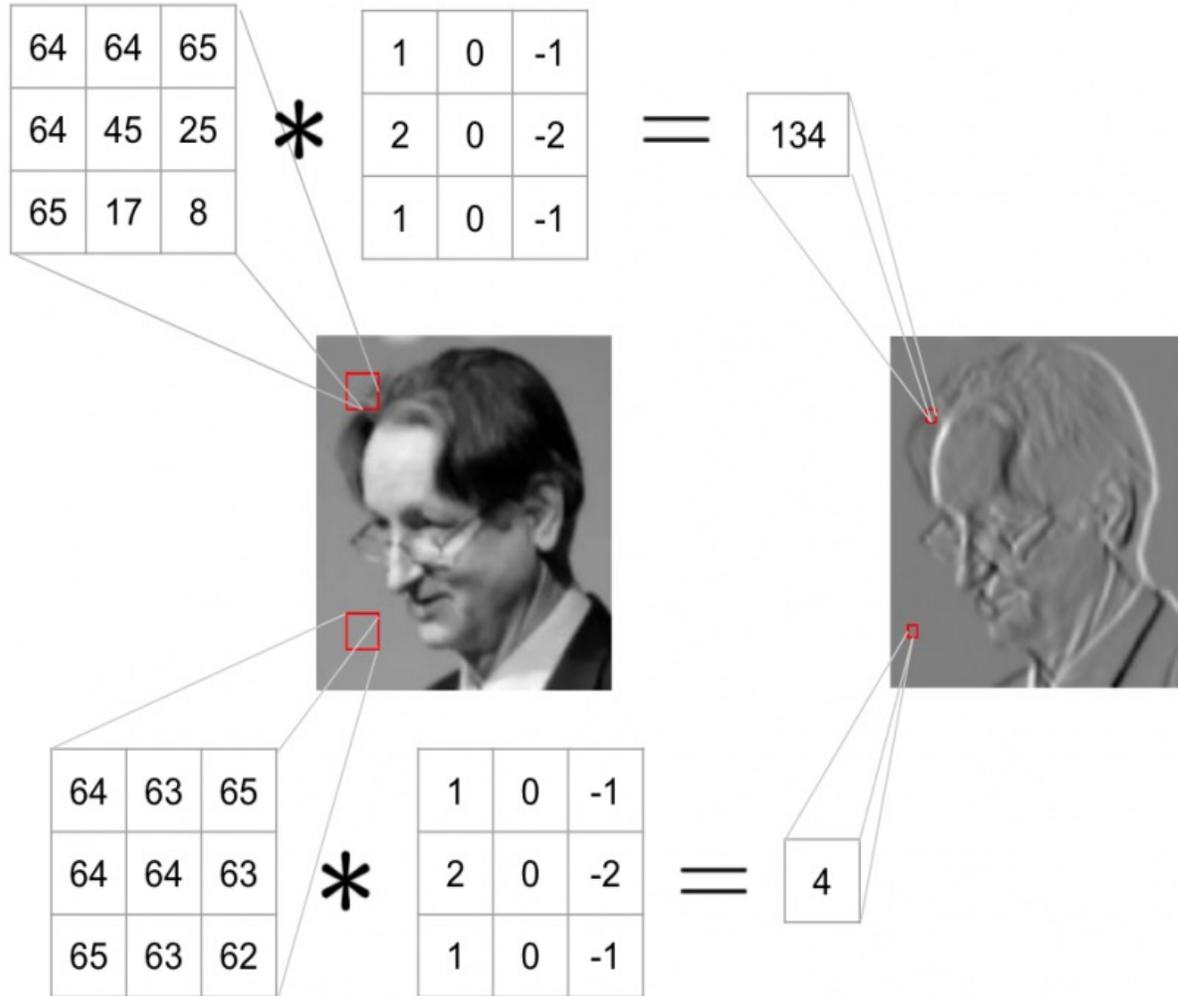
Нейронные сети в задачах машинного обучения



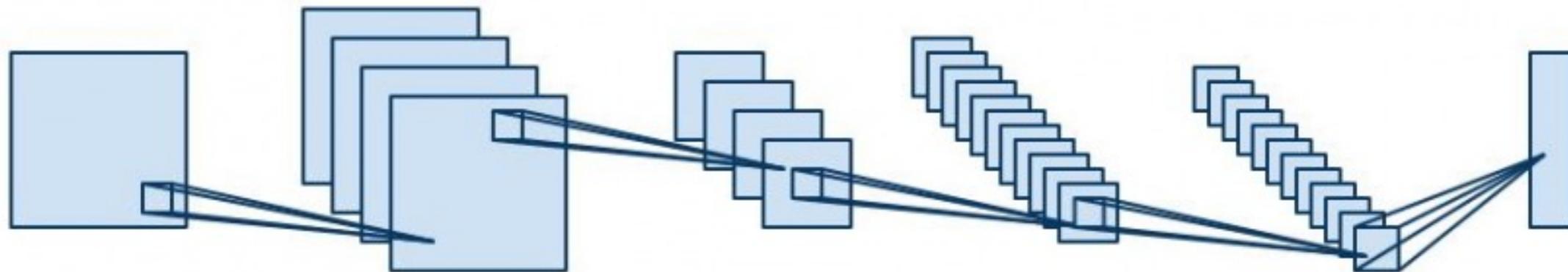
| | | | | | | |
|-----|------|-----|-------|-------|-----|-------|
| 104 | 11,8 | 0,1 | 0,13 | 13,78 | 0,6 | 211,4 |
| 126 | 10,3 | 0,3 | 0,00 | 16,31 | 0,0 | 401,3 |
| 166 | 11,8 | 1,1 | -0,06 | 10,56 | 0,4 | 95,7 |
| 175 | 13,2 | 1,9 | -0,03 | 11,89 | 1,8 | 33,5 |
| 177 | 16,9 | 0,9 | 0,00 | 12,81 | 1,2 | 1,1 |
| 187 | 18,7 | 0,4 | 0,12 | 10,92 | 0,8 | 0,0 |



Сверточные нейронные сети. Фильтр Собеля

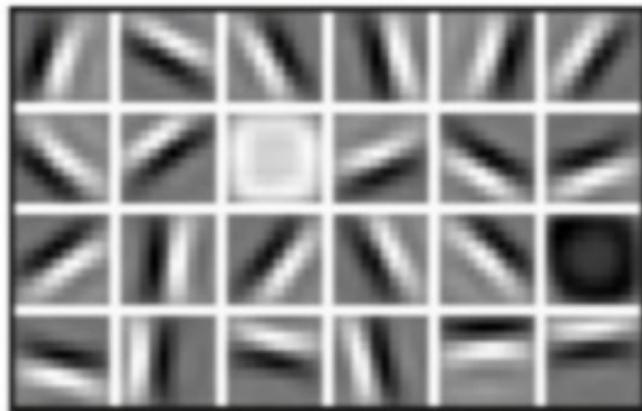


Сверточные нейронные сети. Конструкция



Пример сверточной нейронной сети, состоящей из четырех сверточных слоев и одного полносвязного слоя

Низкоуровневые признаки



Линии и границы

Признаки средних уровней



Глаза, носы, уши

Высокоуровневые признаки

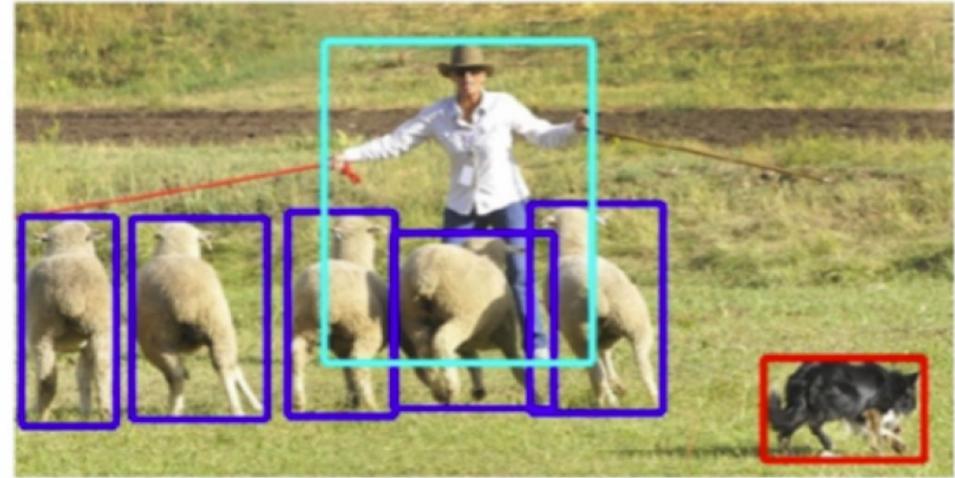


Строение лица

Задачи анализа изображений



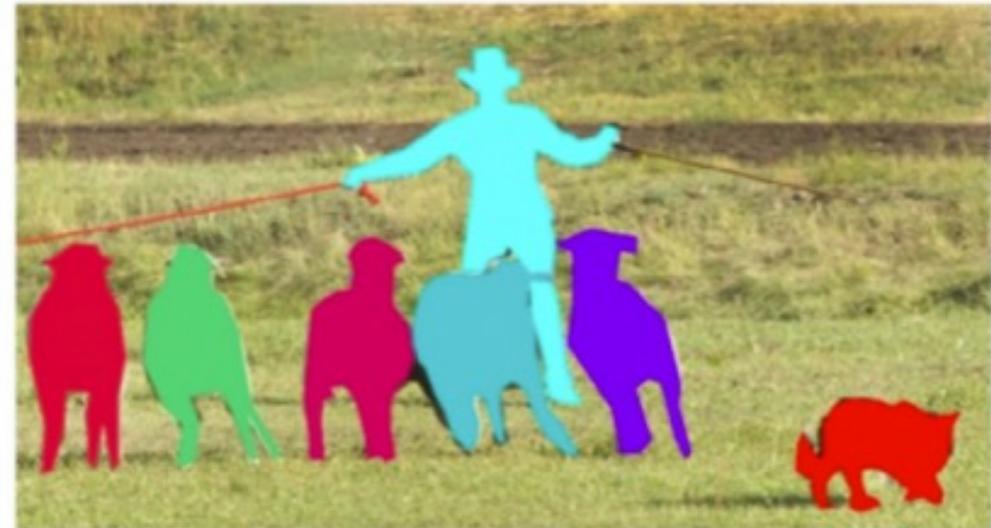
Классификация изображения



Детекция



Семантическая сегментация



Задачи генерации изображений



Примеры лиц знаменитостей, сгенерированных нейросетями.

<https://quickdraw.withgoogle.com/>

<https://deepart.io/>

<https://affinelayer.com/pixsrv/>

<https://habr.com/ru/post/513680/>

<https://www.artlebedev.ru/ironov/>

<https://vc.ru/ml/194200-dall-e-revoljuciya-v-generacii-izobrazheniy-ot-openai>

Анализ текстов

Предобработка текстов:

- Очистка данных
- Сборка словаря

Лето закончится через девятнадцать дней

2 6 5 7 5

Словарь:

| | |
|--------------|-------|
| я | 1 |
| | |
| лето | 2 |
| | |
| мы | 3 |
| | |
| через | 4 |
| | |
| дней | 5 |
| | |
| закончится | 6 |
| | |
| девятнадцать | 7 |

Векторные представления слов

отличное



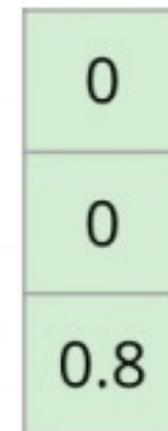
замечательное



мобильное



приложение



Эмбеддинги

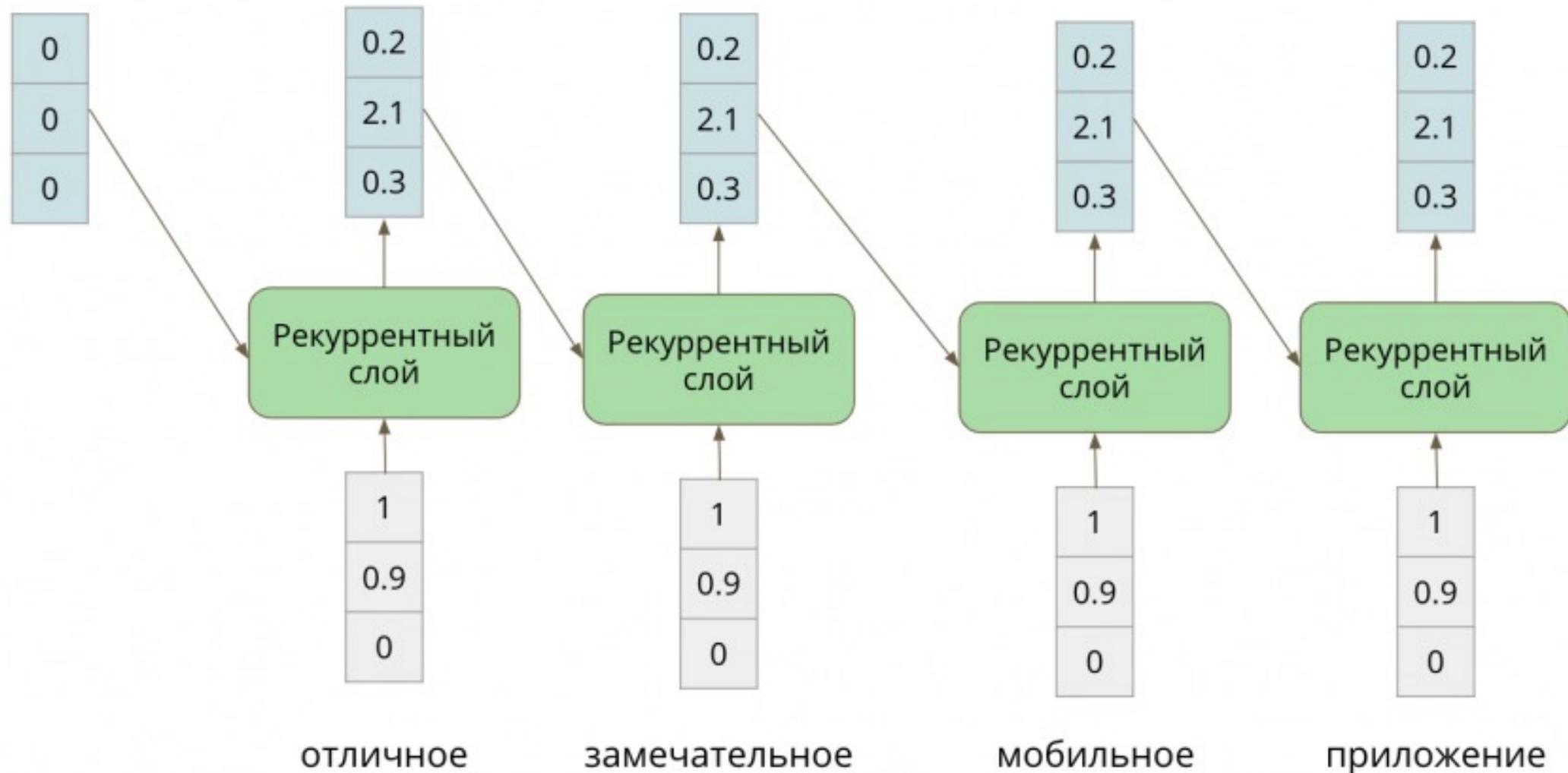
Векторные представления слов



Пример визуализации для задачи анализа эмоционального окраса отзывов

Архитектуры нейросетей для анализа текстов

Рекуррентная нейронная сеть



Задачи анализа и генерации текстов

- Предсказание заработной платы
- Модерация сообщений
- Тегирование
- Выделение именованных сущностей
- Seq2seq

Голосовым помощником решается целый ряд отдельных задач:

- 1.Перевод звука в текст (задача seq2seq).
- 2.Распознавание запроса по тексту для заранее определенных действий
- 3.Озвучка текста (задача генерации звука).

<https://www.kaggle.com/c/job-salary-prediction>

<https://rb.ru/story/ai-moderator/>

<https://habr.com/ru/company/yandex/blog/205198/>

<https://porfirevich.ru/>

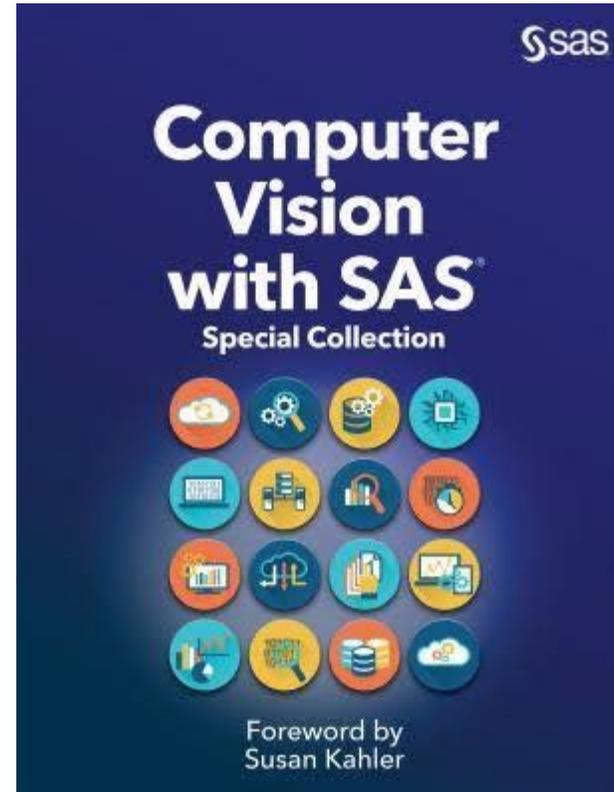
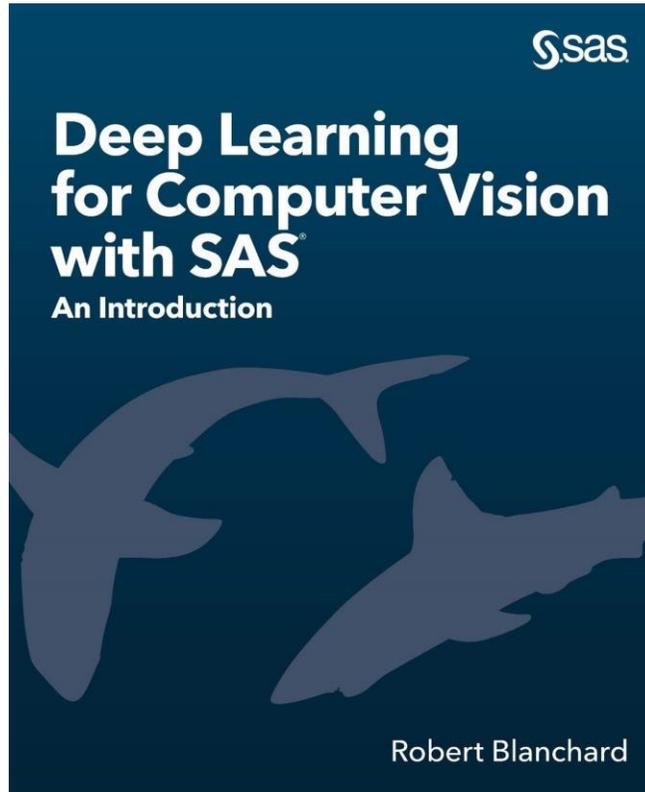
<https://milhidaka.github.io/chainer-image-caption/>

<https://texterra.ru/blog/chemu-nauchilis-neyroseti-i-kogda-oni-zamenyat-kopirayterov.html>

Фреймворки для нейронных сетей:



Литература



Документация DLPy - sassoftware.github.io/python-dlpy

Документация SWAT - sassoftware.github.io/python-swat

Документация SAS - documentation.sas.com