

Введение в предметную область ИИ

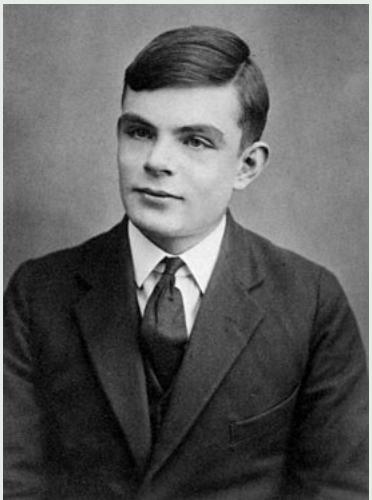
*Артамонов Юрий Николаевич
«Экономика городского хозяйства и жилищного права»*

*Государственное автономное образовательное учреждение высшего образования
«Московский городской университет управления Правительства Москвы имени Ю.М. Лужкова»*



- 1 Основоположники направления ИИ
- 2 Перцепtron
- 3 Функциональное, логическое программирование
- 4 Основные термины и определения
- 5 Задачи машинного обучения
- 6 Современное состояние, тенденции, выводы
- 7 Генеративный искусственный интеллект
- 8 Основные тренды цифровой экономики
- 9 Цифровые проекты и сервисы в городском хозяйстве Москвы

Основоположники направления ИИ



Алан Тьюринг

Джон Маккарти

Фрэнк
блатт

Розен-
Марвин Мински

1950 год:

ИИ будет создан, когда человек общаясь с машиной, не сможет этого распознать.

1956 год:

Проблема состоит в том, что пока мы не можем в целом определить, какие вычислительные процедуры мы хотим называть интеллектуальными. Поэтому под интеллектом понимается только вычислительная составляющая способности достигать целей в мире.

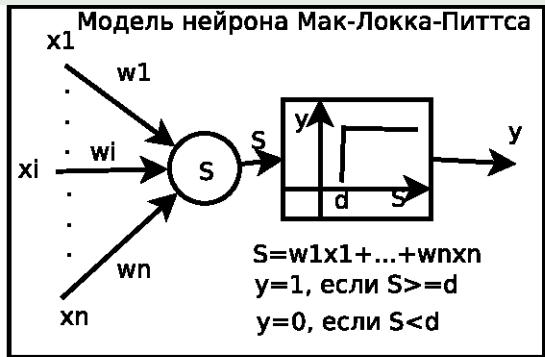
1958—1960:

Нельзя сказать, что мы точно воспроизведим работу человеческого мозга, - но пока перцепtron ближе всего к истине.

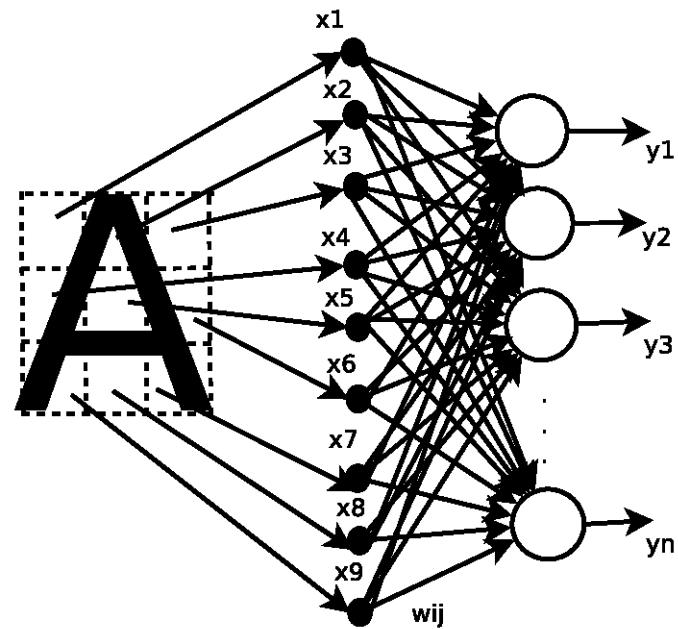
1967 год:

В течение поколения проблема создания ИИ будет практически решена. ... Через три-восемь лет у нас появится машина с интеллектом среднего человека.

Перцептрон



Модель однослойного перцептрана для распознавания букв

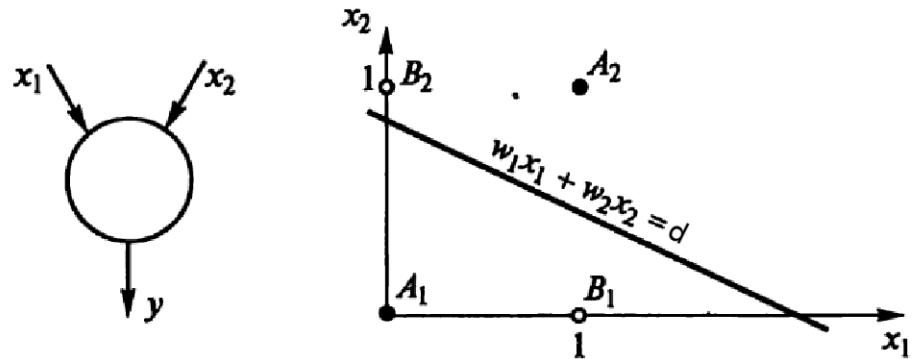


Алгоритм обучения Хебба

если выход y_i неправильный и равен 1, то $w_{ij}(t+1) = w_{ij}(t) - x_j$

если выход y_i неправильный и равен 0, то $w_{ij}(t+1) = w_{ij}(t) + x_j$

Ограничения однослойного перцептрана



Точки	Значение x_1	Значение x_2	Значение y
A_1	0	0	0
A_2	1	1	0
B_1	1	0	1
B_2	0	1	1

Идея рекурсии

Пусть требуется реализовать вычисление

$$sum(n) = 1 + 2 + 3 + \dots + n$$

Реализация без рекурсии

```
1: function sum(n)
2:     result ← 0, i ← 1
3:     while i ≤ n do
4:         result ← result + i
5:         i ← i + 1
6:     end while
7:     return result
8: end function
```

Реализация с рекурсией

```
1: function sum(n)
2:     if n = 0 then
3:         return 0
4:     else
5:         return sum(n - 1) + n
6:     end if
7: end function
```

Идея рекурсии - попытка понять

$$sum(n) = sum(n - 1) + n$$

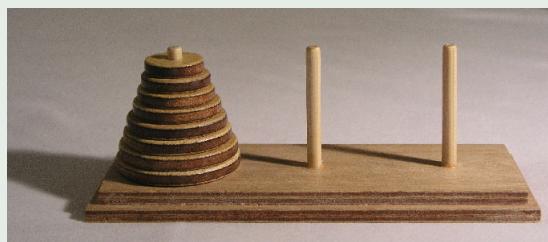
$$sum(n - 1) = sum(n - 2) + (n - 1)$$

...

$$sum(1) = sum(0) + 1$$

$$sum(0) = 0$$

Задача о ханойской Рекурсивный алгоритм башне



Предположим, что мы уже научились перекладывать $n-1$ колец, тогда алгоритм перекладывания n колец такой:

- перекладываем на средний колышек $n-1$ колец
- перекладываем на крайний правый колышек самое большое кольцо
- перекладываем на крайний правый колышек $n-1$ колец
- Последовательно уменьшая n мы приходим к одному кольцу, алгоритм перекладывания которого очевиден.

Функциональное программирование - символьные вычисления

```
emacs@artamonov
File Edit Options Buffers Tools Maxima Complete In/Out Signals Help
Save Undo Cut Copy Paste Find

Maxima 5.42.1 http://maxima.sourceforge.net
using Lisp SBCL 1.4.5.debian
Distributed under the GNU Public License. See the file COPYING.
Dedicated to the memory of William Schelter.
The function bug_report() provides bug reporting information.
(%i1) block(load("/usr/share/emacs/site-lisp/maxima/imaxima.lisp")), linenum:0$

(%i1) expand((x^2+2*x+1)^5);

(%o1)      
$$x^{10} + 10x^9 + 45x^8 + 120x^7 + 210x^6 + 252x^5 + 210x^4 + 120x^3 + 45x^2 + 10x + 1$$


(%i2) integrate(sin(x)*x, x);

(%o2)      
$$\sin x - x \cos x$$


(%i3) diff(x^x,x);

(%o3)      
$$x^x (\log x + 1)$$

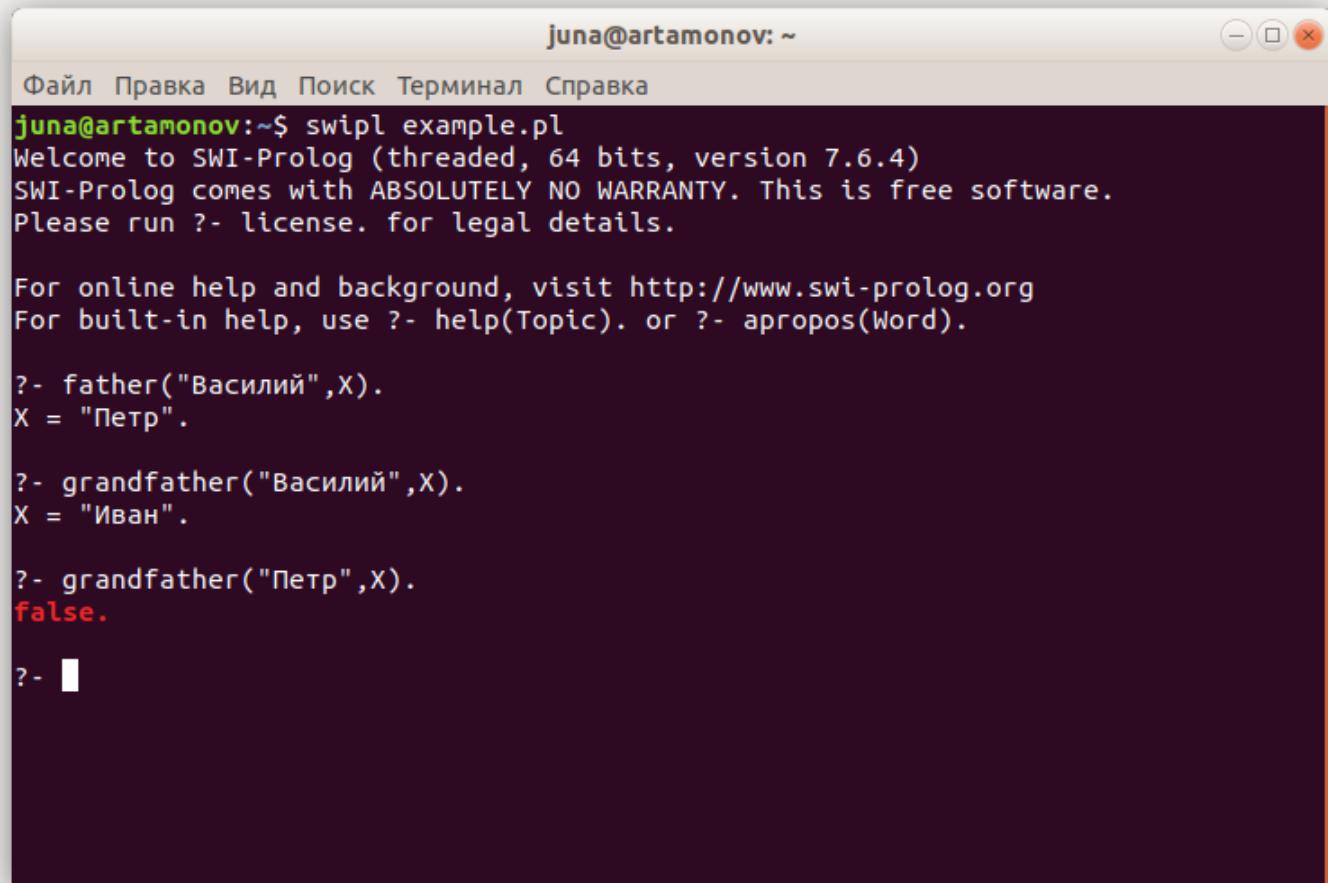

(%i4)

U:**- *maxima* All L17 (Inferior Maxima: run)
Processing Maxima output...done
```

Mathematica — система компьютерной алгебры: <https://www.wolframalpha.com>

Логическое программирование - Пролог

```
father("Василий", "Петр").  
father("Петр", "Иван").  
  
grandfather(X,Y):-father(X,Z),father(Z,Y).
```



The screenshot shows a terminal window titled 'juna@artamonov: ~'. The window contains the following text:

```
Файл Правка Вид Поиск Терминал Справка  
juna@artamonov:~$ swipl example.pl  
Welcome to SWI-Prolog (threaded, 64 bits, version 7.6.4)  
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.  
Please run ?- license. for legal details.  
  
For online help and background, visit http://www.swi-prolog.org  
For built-in help, use ?- help(Topic). or ?- apropos(Word).  
  
?- father("Василий",X).  
X = "Петр".  
  
?- grandfather("Василий",X).  
X = "Иван".  
  
?- grandfather("Петр",X).  
false.  
?-
```

Задача про козу, волка и капусту

Старик должен переправить на другой берег реки волка, козу и капусту. При этом существует два ограничения:

- Его лодка такова, что за один раз может вместе с ним перевезти только кого-то одного: либо волка, либо козу, либо капусту.
- Звери мирно ведут себя только в присутствии старика. Стоит только ему отлучиться, как волк тут же съест козу или коза съест капусту.

Пролог онлайн: https://rextester.com/l/prolog_online_compiler

The screenshot shows a Prolog IDE interface with two windows. The left window contains the Prolog source code for the 'Goat, Wolf, and Cabbage' puzzle. The right window shows the terminal output of the SWI-Prolog interpreter running the code.

Code (t.pl):

```
% W-волк, G-коза, C-капуста, F-фермер
% Состояние: 1-левый берег, 0-правый берег
% конечное состояние - все на правом берегу
route([0,0,0,0],P) :- writeln(P),!.
route(Sost_x,P) :-
    next(Sost_x,Sost_y), % выбор преемника
    not(member(Sost_y,P)), % исключение повторного посещения
    route(Sost_y,[Sost_x|P]). % переход к следующему состоянию
% выбор преемника
next(Sost_x,Sost_y) :- move(Sost_x,Sost_y),
    not(conflict(Sost_y)).
% возможные перемещения
move([1,G,C,1], [0,G,C,0]). % везем вправо волка
move([W,1,C,1], [W,0,C,0]). % везем вправо козу
move([W,G,1,1], [W,G,0,0]). % везем вправо капусту
move([W,G,C,1], [W,G,C,0]). % порожняком идем вправо
move([W,G,C,0], [W,G,C,1]). % порожняком идем влево
move([0,G,C,0], [1,G,C,1]). % везем влево волка
move([W,0,C,0], [W,1,C,1]). % везем влево козу
move([W,G,0,0], [W,G,1,1]). % везем влево капусту
% конфликтные состояния
conflict([1,1,,0]). % фермер-справа, слева - волк, коза
conflict([_,1,1,0]). % фермер-справа, слева - коза, капуста
conflict([0,0,,1]). % фермер-слева, справа - волк, коза
conflict([_,0,0,1]). % фермер-слева, справа - коза, капуста

%Поиск решения
%:- route([1,1,1,1],[]).
```

Terminal Output:

```
juna@artamonov:~$ swipl t.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 7.6.4)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit http://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- route([1,1,1,1],[]).
[[0,1,0,1],[0,1,0,0],[0,1,1,1],[0,0,1,0],[1,0,1,1],[1,0,1,0],[1,1,1,1]]
true .
```

СЛАБЫЙ ИИ

- Решает конкретные узкоспециализированные задачи
- Успешно реализуется сегодняшними технологиями
- Часто превосходит человека



СИЛЬНЫЙ ИИ

- Работает на уровне человека
- Решает множество задач
- Самообучается в условиях неопределенности

В начале 80-х годов получили широкое распространение экспертные системы — программы, имитирующие знания и умения человека. Для создания таких программ проводился подробный опрос экспертов с целью выяснить, каким именно образом они решают задачи, например, по каким правилам психолог отвечает пациенту или врач определяет диагноз. Полученные правила затем реализовывались в виде компьютерной программы. Однако и эти системы быстро достигли своего потолка, положив начало второй «зиме искусственного интеллекта». Тем не менее, экспертные системы по-прежнему используются в некоторых областях, особенно в тех, в которых особое внимание уделяется надежности, например в медицине.

1990 г. - Deep Blue

В конце 90-х годов XX века и начале XXI века драйверами развития искусственного интеллекта стали увеличивающиеся вычислительные мощности, активное применение математических методов, а также упор на решение конкретных задач. Искусственный интеллект начал использоваться в различных областях: в логистике, экономике, медицине. Одним из известных прорывов стала разработка суперкомпьютера Deep Blue, обыгравшего в 1997 году действующего чемпиона мира по шахматам Гарри Каспарова.

2000 г. - Данные

В начале XXI века одним из основных драйверов разработки алгоритмов искусственного интеллекта стали данные, а акценты в сфере искусственного интеллекта сместились в сторону таких областей, как машинное обучение, анализ больших данных и др. В отличие от экспертных систем, формализующих человеческий опыт в виде программы, машинное обучение извлекает знания из большой базы данных, благодаря чему получается более точный алгоритм, корректно работающий в более широком круге случаев.

2010-е: Watson и DeepMind

В 2011 году компания IBM представила систему Watson, способную находить ответы на вопросы, заданные на естественном языке. Для демонстрации навыков система «сыграла» в игру «Jeopardy!» (американский аналог русскоязычной «Своей игры») и обыграла обоих соперников. В основе Watson — алгоритмы информационного поиска, обработки естественного языка, машинного обучения и построения логических цепочек.

В 2015 году британская компания DeepMind (один из крупнейших сегодняшних лидеров в области искусственного интеллекта) представила AlphaGo — систему, обыгравшую чемпиона мира по игре в Го, Ли Седоля. Го намного сложнее шахмат из-за большего количества возможных позиций, что создает сложности в применении алгоритмов поиска ходов, активно используемых в DeepBlue. Основу AlphaGo составляют алгоритмы машинного обучения: программа обучалась на данных 160 тысяч партий профессионалов.

2020: GPT-3, AlphaFold

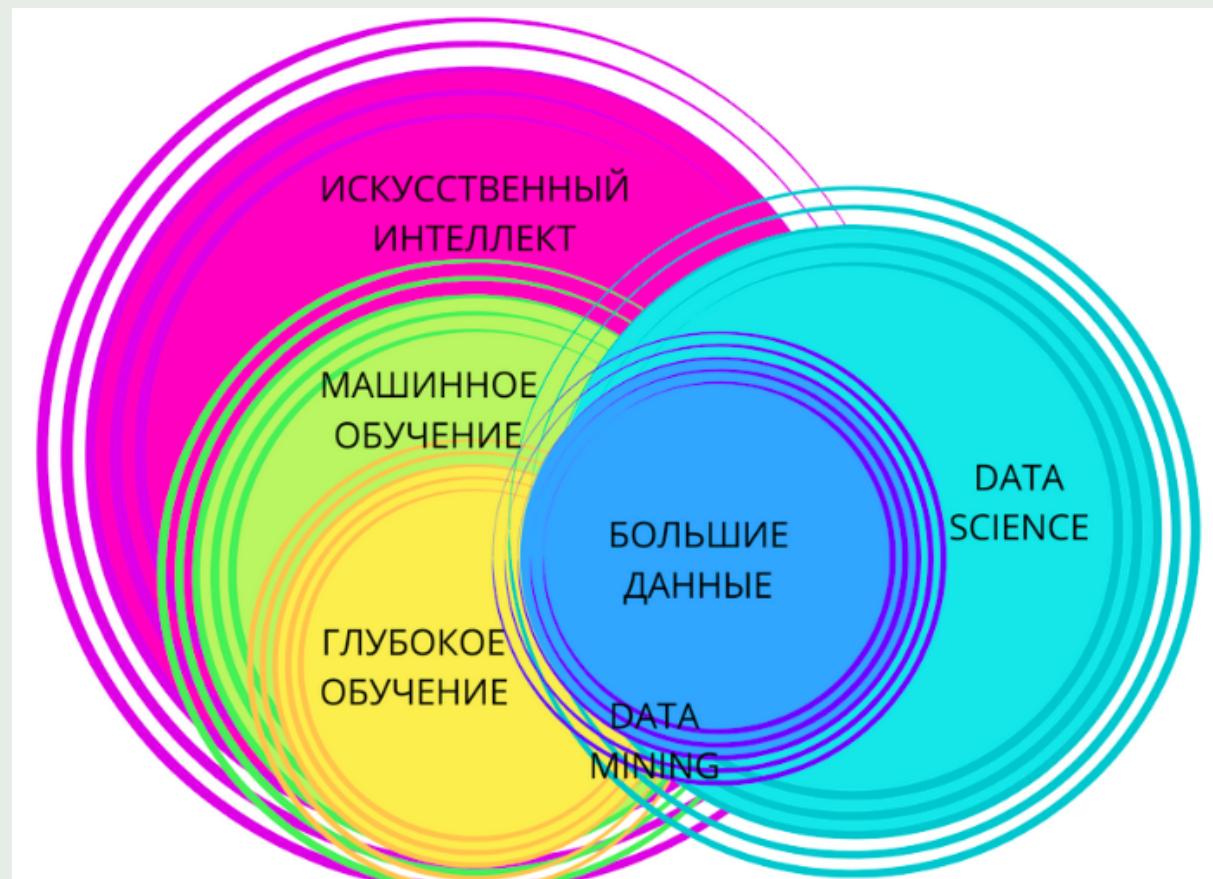
В 2020 году американская компания OpenAI представила систему GPT-3, способную генерировать англоязычные тексты, неотличимые от написанных человеком, с сохранением темы по всему объему текста, грамматической корректностью и т. д.

В том же 2020 году компания DeepMind представила AlphaFold, решающую задачу прогноза третичной структуры белков, которая являлась одной из сложнейших и важнейших задач в современной биологии и решение которой не могли найти около 50 лет.

Основные определения и термины

Искусственный интеллект - это комплекс технологических решений, позволяющий имитировать когнитивные функции человека и получать при выполнении конкретных задач результаты, сопоставимые, как минимум, с результатами интеллектуальной деятельности человека

Искусственный интеллект включает в себя множество областей



Машинное обучение

Класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение в процессе применения решений множества сходных задач.

Глубинное обучение

Иногда называют «глубокое обучение» (от англ. Deep learning). Подобласть машинного обучения, где в качестве алгоритмов используются нейронные сети.

Data Science

Это концепция объединения статистики, анализа данных, машинного обучения и связанных с ними методов для понимания и анализа реальных явлений.

Data Mining

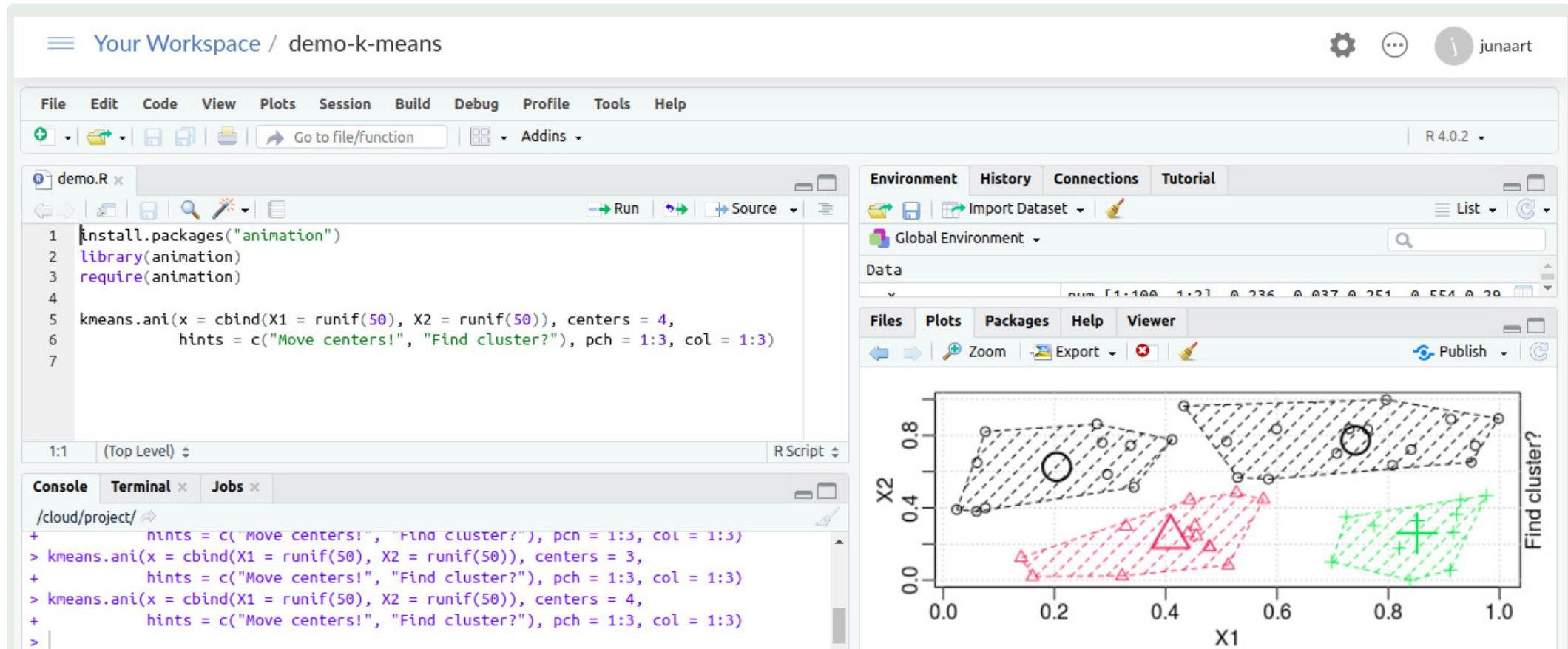
Широкое понятие, означающее извлечение знаний из данных.

Большие данные

Это набор подходов и методов, разработанных для анализа данных огромных размеров.

- Классификация
- Регрессия
- Кластеризация
- Понижение размерности
- Важность признаков
- Ассоциации, рекомендации

Демонстрация метода кластеризации k-means



<https://posit.cloud/content/1732717>

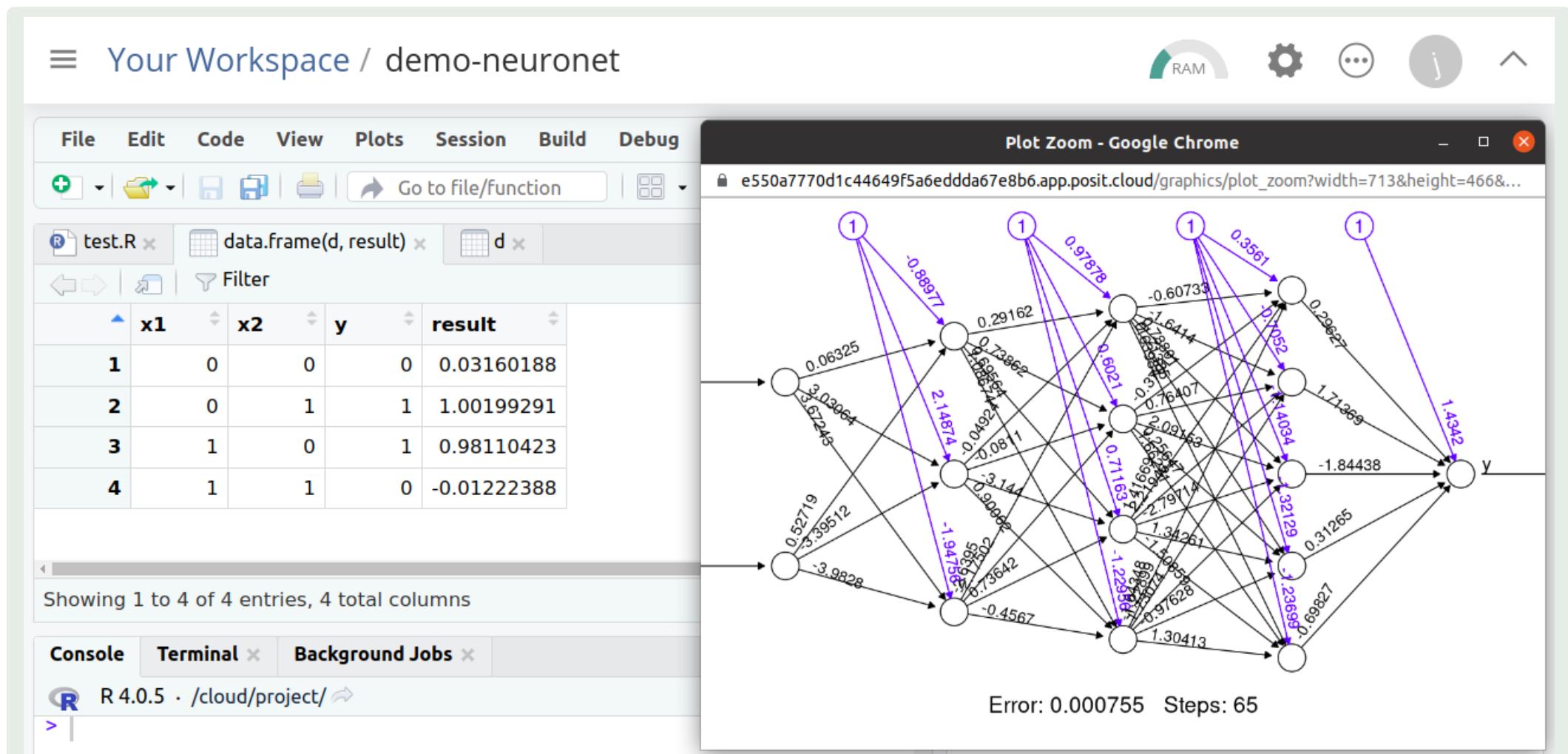
Пример использования нейронных сетей: решение задачи «исключающее или»

```
install.packages("neuralnet")
library(neuralnet)
x1<-c(0,0,1,1)
x2<-c(0,1,0,1)
y<-c(0,1,1,0)

d<-data.frame(x1,x2,y)
View(d)

nn=neuralnet(y~x1+x2, data=d, hidden=c(3,4,5))
plot(nn)
xx<-data.frame(x1,x2)
result<-predict(nn, xx)
View(data.frame(d, result))
```

Пример использования нейронных сетей: решение задачи «ИСКЛЮЧАЮЩЕЕ ИЛИ»



Работа с кодом доступна по ссылке:
<https://posit.cloud/content/1713842>

Пример использования нейронных сетей: решение задачи «Классификация ирисов»

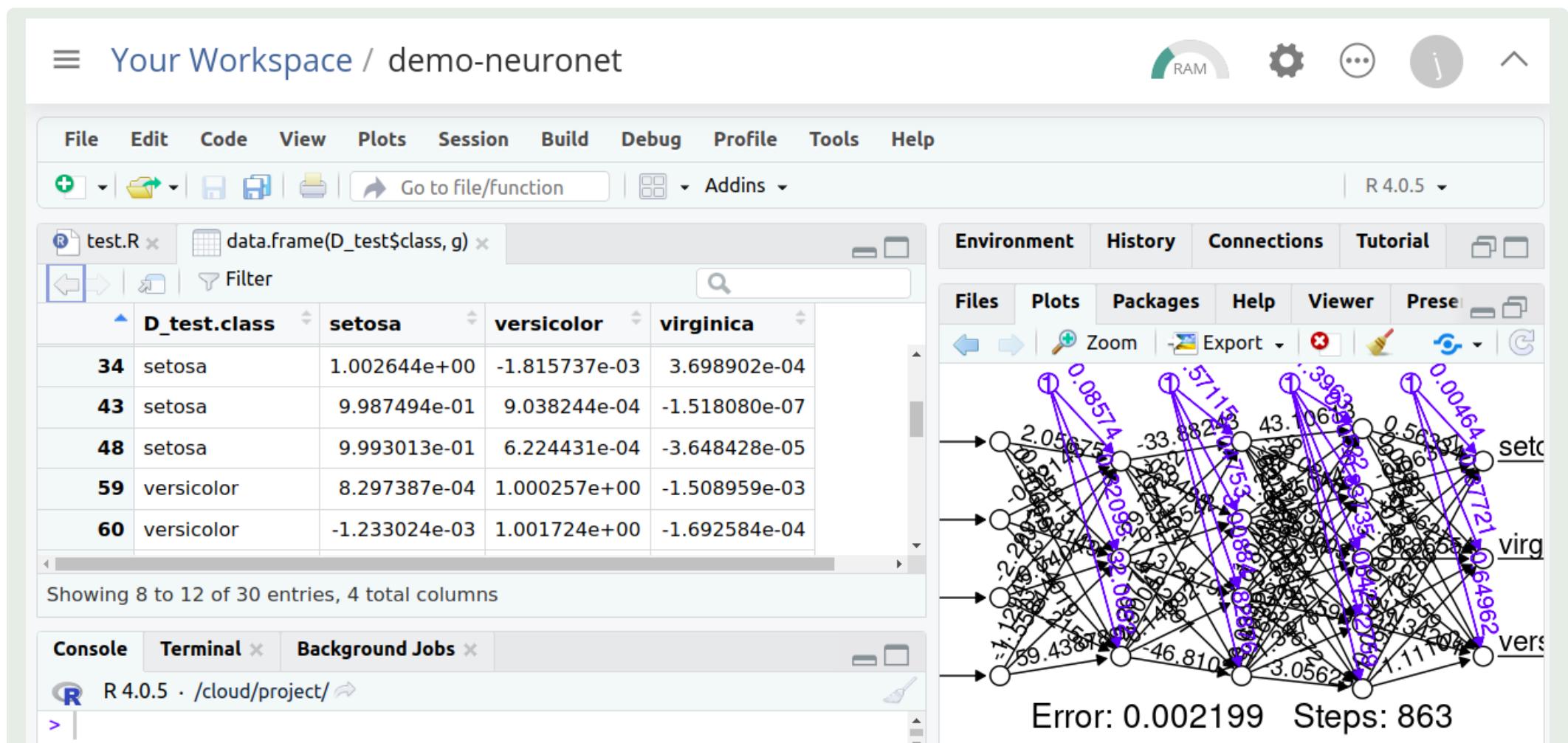
```
x<-c(1:150)
y<-sample(x, floor(150*0.8))
z<-setdiff(x, y)
D_teach<-iris[y,]
D_test<-iris[z,]
nn=neuralnet(class ~ sl + sw + pl + pw, data=
D_teach, hidden=c(3,4,5))
plot(nn)

g<-predict(nn,D_teach)
colnames(g)<-c("setosa","versicolor","virginica")
View(data.frame(D_teach$class,g))

g<-predict(nn,D_test)
colnames(g)<-c("setosa","versicolor","virginica")
View(data.frame(D_test$class,g))
```

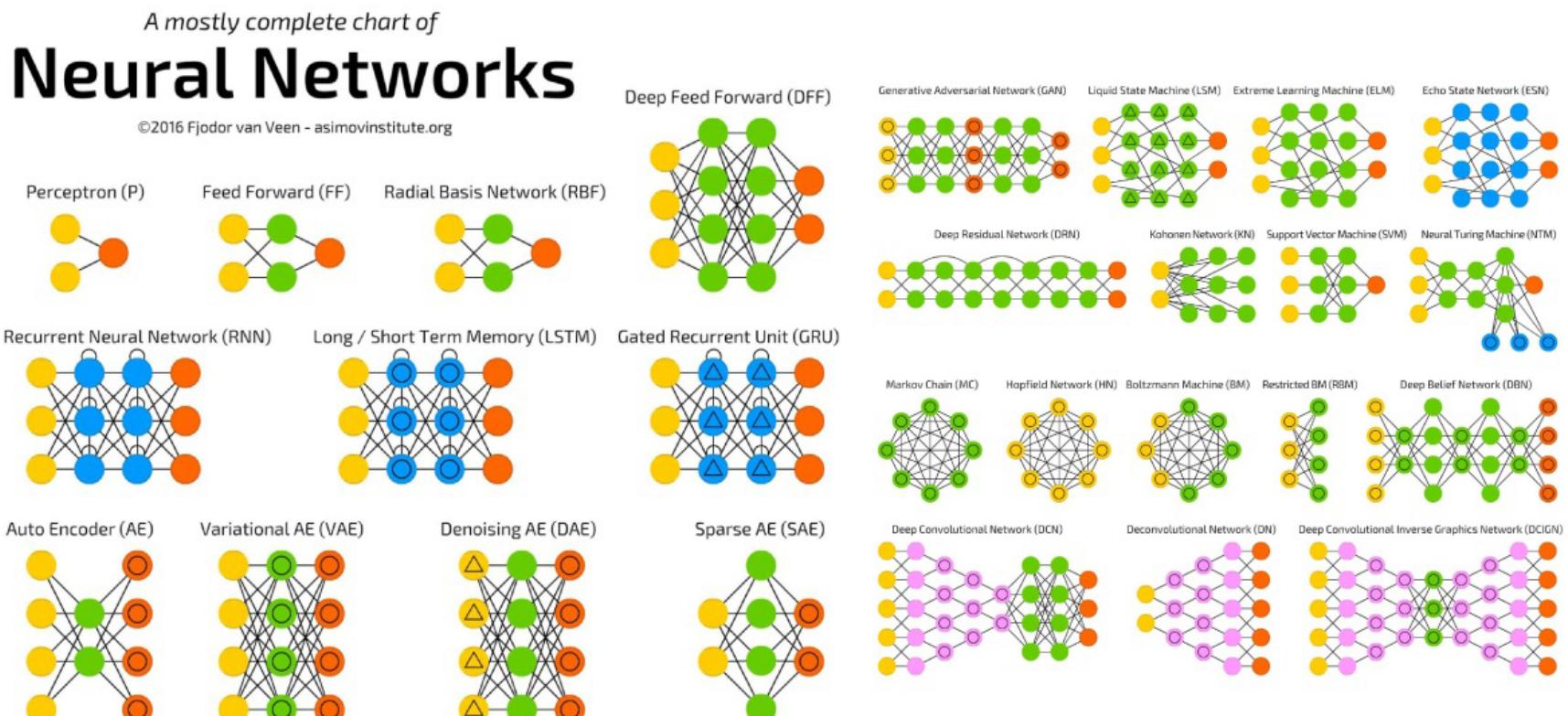
Пример использования нейронных сетей: решение задачи

«Классификация ирисов»



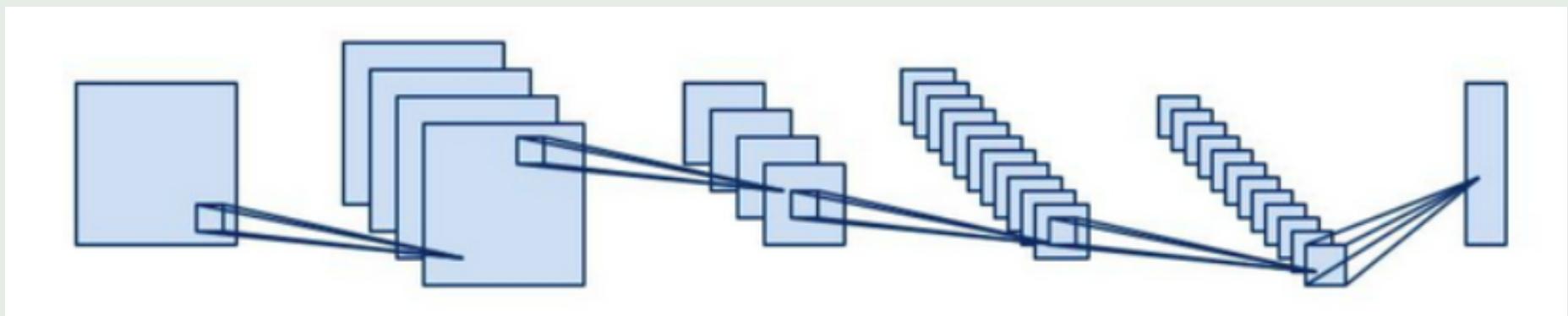
Работа с кодом доступна по ссылке:
<https://posit.cloud/content/1713842>

Различные варианты нейронных сетей



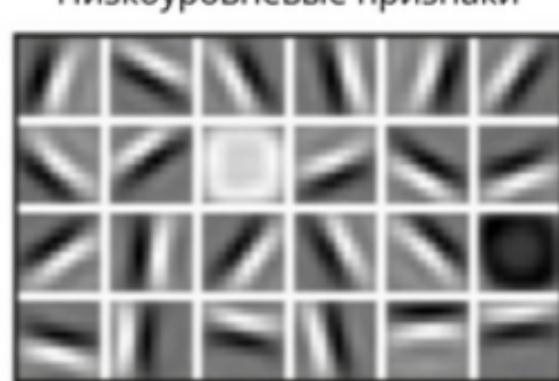
Анализ изображений - сверточные нейронные сети

Первый сверточный слой применяется непосредственно к самому изображению, второй слой — к выходу первого сверточного слоя и т. д. Выход сверточного слоя формально тоже является изображением, но на глубоких слоях нейронной сети это «изображение» уже не будет интерпретироваться человеком. Между сверточными слоями, как и между полно связанными, вставляют слои нелинейности, а в конце сверточной архитектуры обычно вставляют один или несколько полно связанных слоев.



Анализ изображений - сверточные нейронные сети

В сверточных нейросетях при увеличении номера сверточного слоя повышается уровень абстракции. Первые слои распознают простые переливы яркости и отдельные цвета, слои чуть глубже распознают простые геометрические формы, еще более глубокие слои распознают части изображений, например глаза, губы и нос при анализе лиц, а самые глубокие слои отвечают за распознавание целых объектов.



Низкоуровневые признаки
Линии и границы



Признаки средних уровней
Глаза, носы, уши



Высокоуровневые признаки
Строение лица

Задачи анализа изображений



Задачи генерации изображений

За последние несколько лет произошел прорыв в решении обратной задачи — генерации фотoreалистичных изображений и видео. Как и в анализе изображений, в генерации изображений используются сверточные нейронные сети. На вход нейросети подаются характеристики изображения, которое нужно сгенерировать, а на выход нейросеть выдает новое изображение.



Задачи генерации изображений

Перенос стиля работает так: на вход нейросети подаются два изображения — то, которое нужно перерисовать, и то, с которого нужно скопировать стиль, а на выход выдается стилизованное изображение.

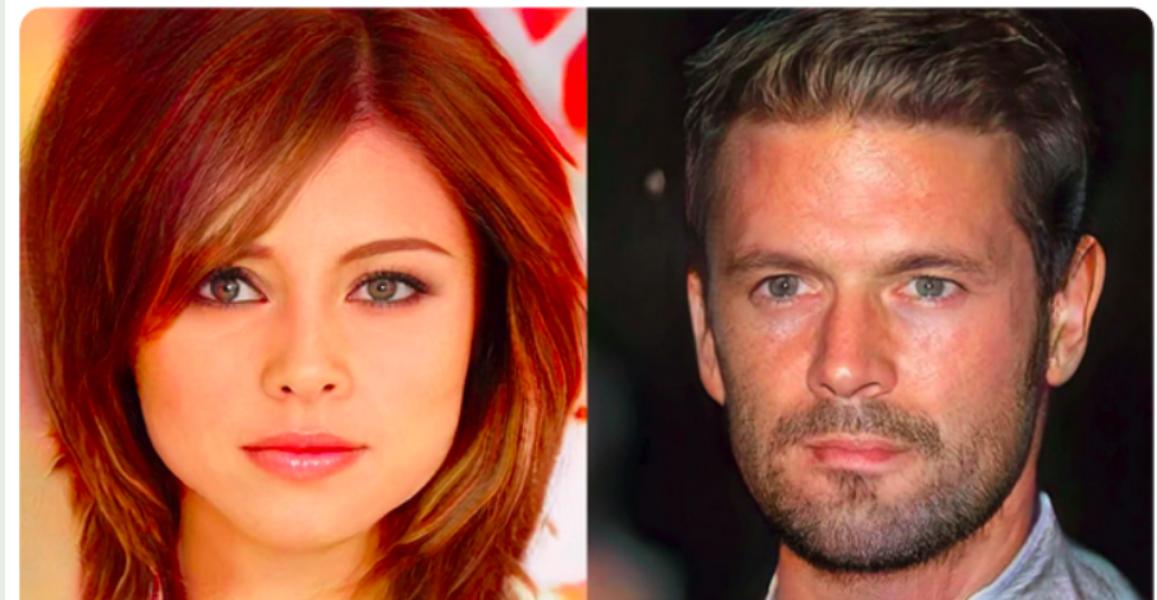


Смена времени дня на фотографиях

Задачи генерации изображений

Чтобы генерировать фотoreалистичные изображения, используют так называемые генеративно-состязательные сети (generative adversarial networks, GAN).

GAN состоит из двух нейросетей: первая генерирует изображение, а вторая решает задачу классификации и учится определять, на входе настоящее оно или сгенерированное. Задача первой нейросети — "обмануть" вторую, и она "вынуждена" генерировать реалистичные кадры.



Лица знаменитостей, сгенерированных нейросетью

Kandinsky - новые возможности генерации изображений



«Альпака в солнечных очках и шляпе, с мороженым, акварельный рисунок»



«Девочка машет рукой, хорошая прорисовка, 4К»



«A cute corgi lives in a house made out of sushi, 4K»

Попробуйте сами!

Kandinsky 2.2 - телеграмм-бот



Анализ текстов - рекуррентные нейронные сети, трансформеры

Область обработки текстов по-английски называется Natural Language processing (NLP).

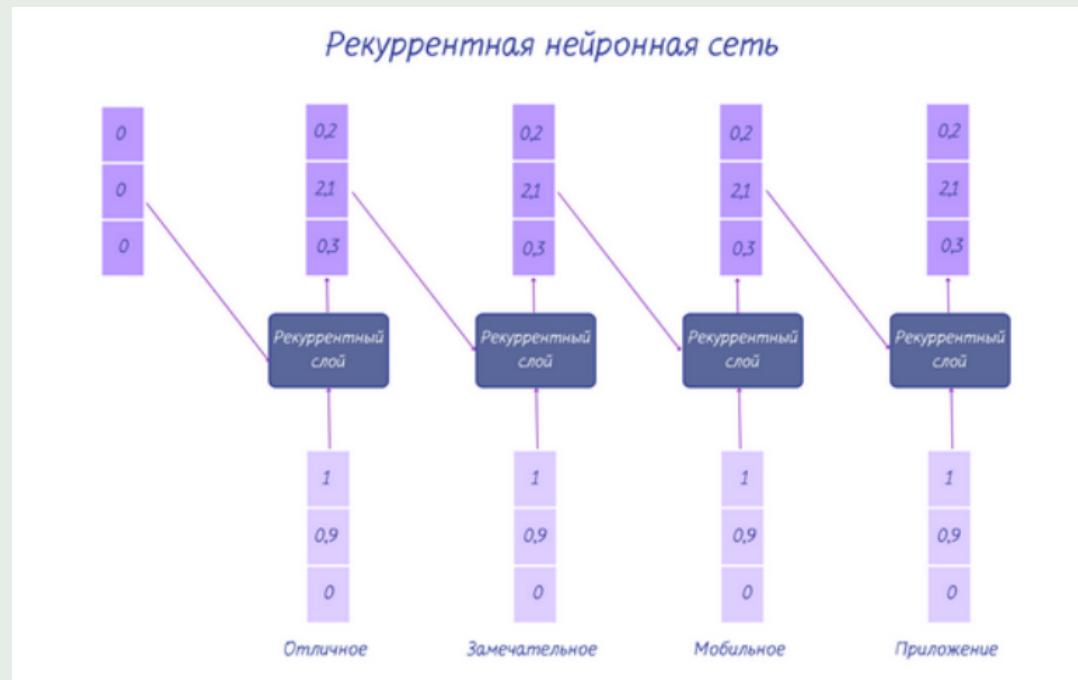
Алгоритм обработки текстов

- Предобработка текстов. Перед тем, как обучать алгоритм на данных, их необходимо подготовить к обработке: очистить и собрать словарь, который нейросеть потом будет использовать для перевода текста на "машинный" язык. В результате предобработки каждый текст представляется в виде последовательности номеров слов в словаре.
- Векторные представления слов(эмбеддинги). Для каждого слова в словаре слой извлекает набор чисел фиксированного размера, например 256 чисел — этот набор и называется векторным представлением, или эмбеддингом. У разных слов — разные векторные представления. При этом часто получается, что у слов, близких по смыслу, векторные представления похожи. Каждое число в эмбеддинге соответствует какому-то «смыслу».
- Использование рекуррентного слоя или архитектуры трансформер

Рекуррентные нейронные сети

Основная идея рекуррентного слоя — повторение одной и той же процедуры обработки для каждого слова.

Внутри рекуррентный слой устроен как полносвязанный слой: в качестве входных нейронов выступают эмбеддинг слова и предыдущее скрытое состояние, в качестве выходных нейронов — новое скрытое состояние. Самое первое скрытое состояние обычно делают состоящим из нулей. Рекуррентных слоев можно использовать несколько, один за другим. В результате получится рекуррентная нейронная сеть (Recurrent Neural Network, RNN).

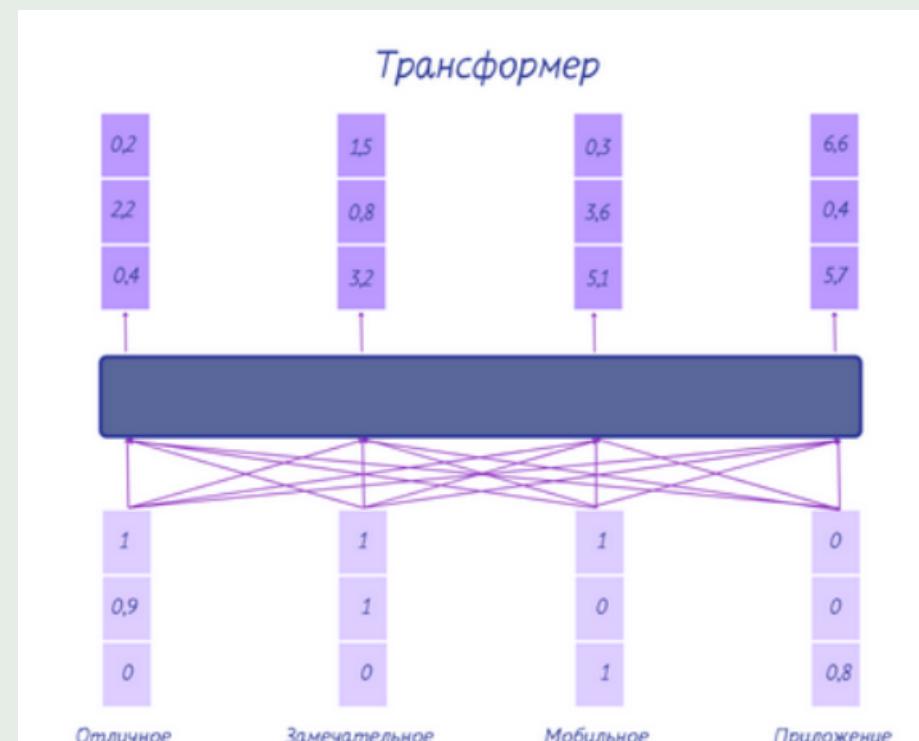


Трансформеры

Недостатки рекуррентных нейросетей, отсутствие связи между далекими словами и забывание слов, были исправлены в другой архитектуре, называемой Трансформером.

В Трансформере новое скрытое состояние для слова вычисляется на основе векторных представлений всех остальных слов.

В итоге каждое слово связано с каждым и проблем с учетом «длинных» зависимостей нет. Обычно Трансформер состоит из четырех или шести слоев, каждый из которых включает попарные связи.

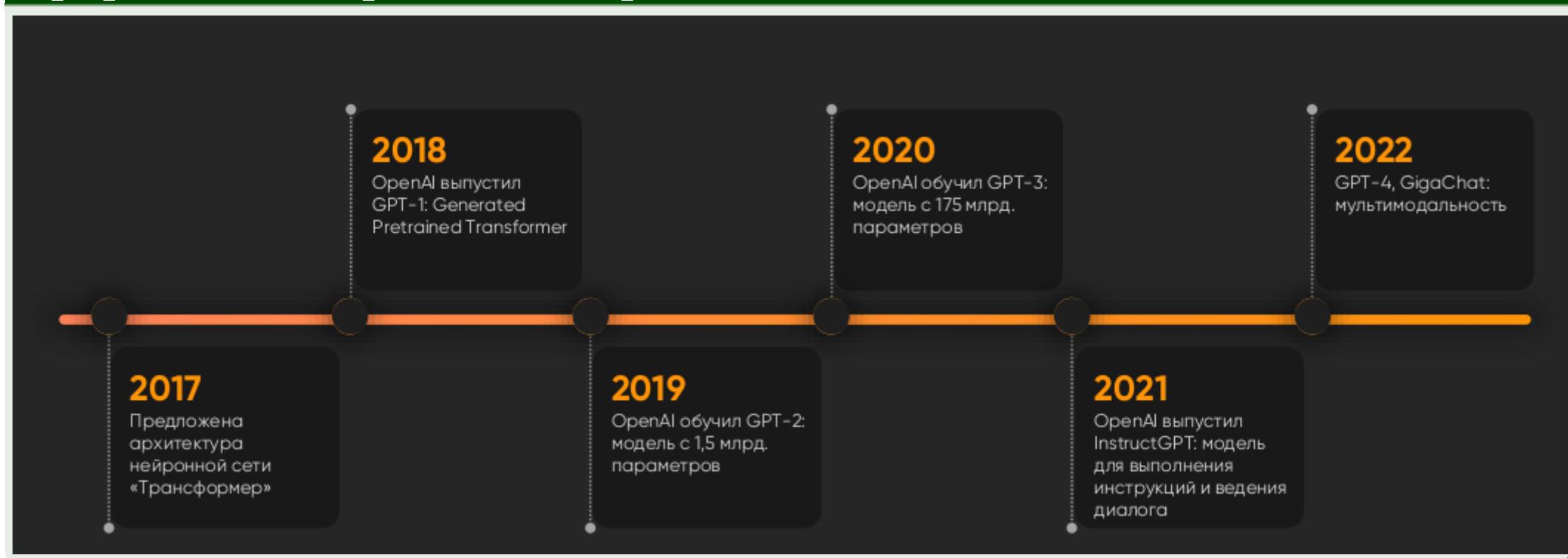


Задачи анализа текстов

- Задача классификации текстов. Например, обнаружение спама, оскорблений и т.п.
- Задача регрессии. Например, по тексту объявления о вакансии предсказывать заработную плату.
- Задача тегирования - нужно выполнить отдельное предсказание для каждого слова в тексте. К этой группе относятся задача определения частей речи для каждого слова и задача определения синтаксических ролей в предложении — такие алгоритмы помогают, в частности, выполнять автокоррекцию текста.
- Задача выделения именованных сущностей - найти в тексте города, компании, персоналии и другие объекты реального мира.
- Задача генерации текста.

Сказать, что нейросети достигли полного понимания текстов, написанных людьми, и способны генерировать абсолютно качественные тексты самостоятельно, пока нельзя, однако существенный прорыв в области NLP уже произошел.

Прорывы и открытия в мире языковых моделей



Общие сведения о чат боте

ChatGPT — нейросеть для генерации текстового контента, и обучали ее на огромных массивах текста из Интернета. Хорошо понимает разные языки. Может просто общаться, помогать решать задачи, писать программы на языке программирования, генерировать изображения. Однако сеть неидеальна и правильность своих ответов она не гарантирует, но она умеет обучаться и корректировать свое поведение. Текущая доступная рабочая версия 3.5.

В России прямой доступ к чат боту заблокирован, но имеется возможность доступа через телеграмм каналы.

На текущий момент компания OpenAI анонсировала новую версию GPT-4. На их официальном сайте появилось следующее сообщение:

GPT-4 — это большая мультимодальная модель (принимающая изображения и текст на входе и выводящая текст на выходе), которая, хотя и хуже, чем люди во многих реальных сценариях, демонстрирует производительность на уровне человека в различных профессиональных и академических тестах.

Примеры использования ChartGPT

Программируем...

Today

/start 17:19✓

Напишите свой вопрос, задачу или код. 17:19

Напиши программу: вводится число N, программа вычисляет и выводит на экран сумму:
1-2+3-4+5-...+(-)N 17:29✓

Юрий Артамонов
Напиши программу: вводится число N, программа вычисляет и ...
num = int(input("Введите число N: "))
sum = 0
sign = 1
for i in range(1, num+1):
 sum += sign * i
 sign *= -1
print("Сумма ряда равна:", sum)

17:29

Message

Решаем...

Реши математическую задачу:
Монету бросают трижды. Сколько разных последовательностей орлов и решек можно при этом получить?

17:35 ✓

Юрий Артамонов

Реши математическую задачу: Монету бросают трижды. Сколь...

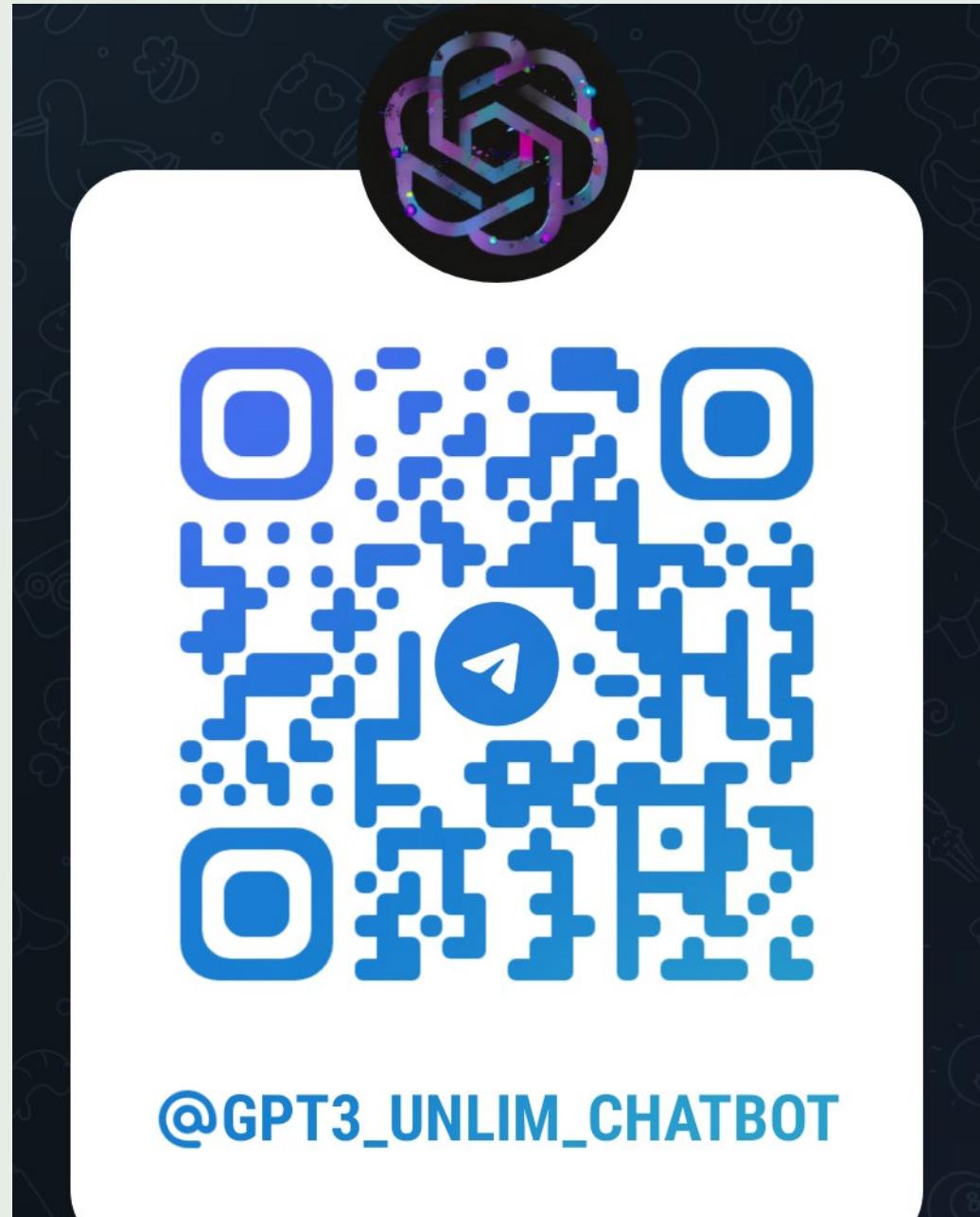
Каждый бросок может закончиться орлом или решкой, то есть у каждого броска два возможных исхода. Так как монету бросают трижды, то всего возможных исходов будет $2 \times 2 \times 2 = 8$, то есть восемь различных последовательностей орлов и решек.

Ответ: 8.

17:35

Попробуйте сами!

ChatGPT4 - телеграмм-бот



Попробуйте сами!

Deepseek



<https://chat.deepseek.com/>

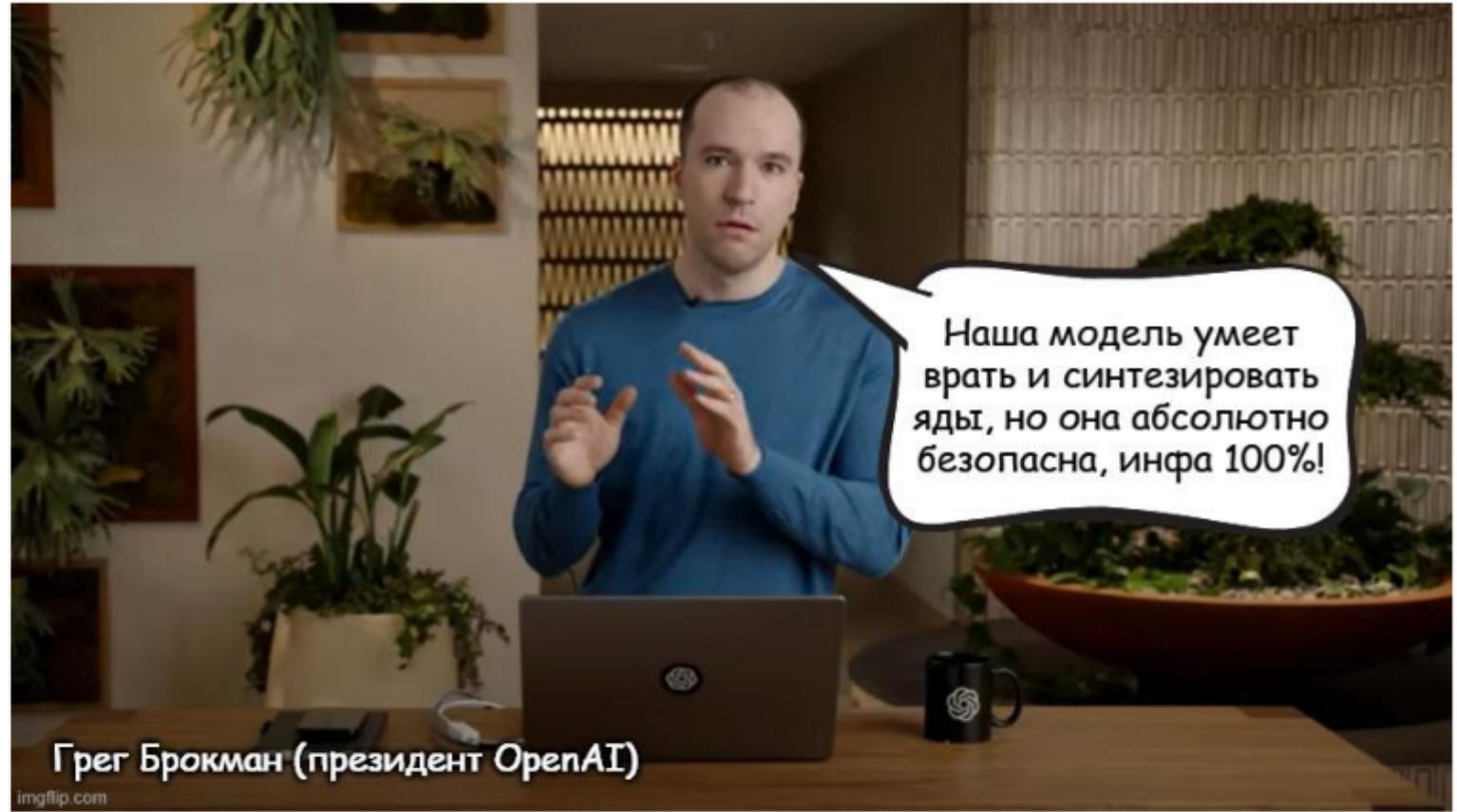
Отличия ChatGPT-4

А еще известно:

- GPT-4 предусматривает несколько способов взаимодействия с человеком - не только с помощью ввода текста, но также при голосовом общении, отправкой изображения, звука или видео.
- ИИ сможет как воспринимать информацию в виде файлов и видео, так и генерировать в таком же формате. То есть он сможет делать музыку, видео по текстовому сообщению.
- Ответы ChatGPT при использовании модели GPT-4 будут ещё более человечными, то есть сильнее приближенными к речи, которую обычно используют люди, общаясь друг с другом.
- GPT-4 будет поддерживать "буквально все языки".
- GPT-4 будет в 537 раз способнее своего предшественника.

GPT 4

1. Может видеть и понимать изображения
2. Умеет писать код для «типовых» задач
3. Обладает большей памятью 32768 токенов (vs 4096 Chat GPT)
4. «Управляемость» — способность менять свое поведение по требованию



Грег Брокман (президент OpenAI)

<https://openai.com/product/gpt-4>

<https://arxiv.org/abs/2303.12712>

Все пытаются построить свой GPT

Мы живем во время больших изменений

«Второй пилот» для работы, встраивание ИИ в продукты MS Office и создание Business Chat

Text2video-модель, создающая короткие видеоролики по текстовому описанию

Облачный сервис для создания графического контента

Обновленный инструмент для написания кода по текстовым и голосовым запросам

Языковая модель с ChatGPT-подобной функциональностью

Новая генеративная модель, создающая изображения высокого качества

Модель по выделению объектов на изображении

16 марта 365 Copilot Microsoft	14 марта GPT-4 OpenAI	Большая языковая модель, которая может работать с изображениями
19 марта ModelScope Alibaba	17 марта MidJourney v5 MidJourney	Генерирует более реалистичные изображения, больше стилей, лучше детализация
21 марта Picasso NVIDIA	20 марта Gen-2 RunWay	Новая модель генерации видео по текстовому описанию
22 марта Copilot X GitHub	21 марта Инфраструктура AWS и NVIDIA	AWS и NVIDIA планируют создать AI инфраструктуру в 20 exaFLOPS
29 марта Dolly Databricks	23 марта ChatGPT plugins OpenAI	OpenAI начала внедрение первой версии поддержки плагинов в нейросеть ChatGPT
4 апреля Kandinsky 2.1 Sber AI	30 марта BloombergGPT Bloomberg	Крупнейшая в мире модель, построенная на финансовых данных
10 апреля SAM Meta	5 апреля Суперкомпьютер Google	Экологичный суперкомпьютер, созданный на базе чипов TPU
	11 апреля Tongyi Qianwen Cloud	Языковая модель для создания аналога ChatGPT

Сравнение по числу параметров, длина контекста, доступность в Open-Source

	Vicuna Berkeley	Falcon TTI Dubai	LLaMA Meta.AI	PaLM 2 Google AI	GPT - 4 OpenAI	Claude 2 Anthropic AI
Контекст	2 048	4 096	2 048	4 096	32 768	75 000
Параметров	До 33B	До 40B	До 65B	До 540B	До 1000B	-
Модальности	Текст	Текст	Текст	Текст, Изображения	Текст, Изображения	Да
Плагины	Нет	Нет	Нет	Нет	Да	Да

 Полностью доступно в Open – Source  Использование доступно по API

По большому числу бенчмарков ИИ уже **превосходит человека**

ИИ > Человек

лучше любого эксперта
определяет

дипфейки

99% > 66%

лучше среднего взрослого

классифицирует

изображения

98% > 95%

ИИ ≈ Человек

как взрослый человек
решает общие тесты на

**понимание
языка**

90% ≈ 90%

на основе утверждений
на уровне взрослого

делает выводы

92% ≈ 92%

ИИ < Человек

AI не способен
к сознательному проявлению

воли

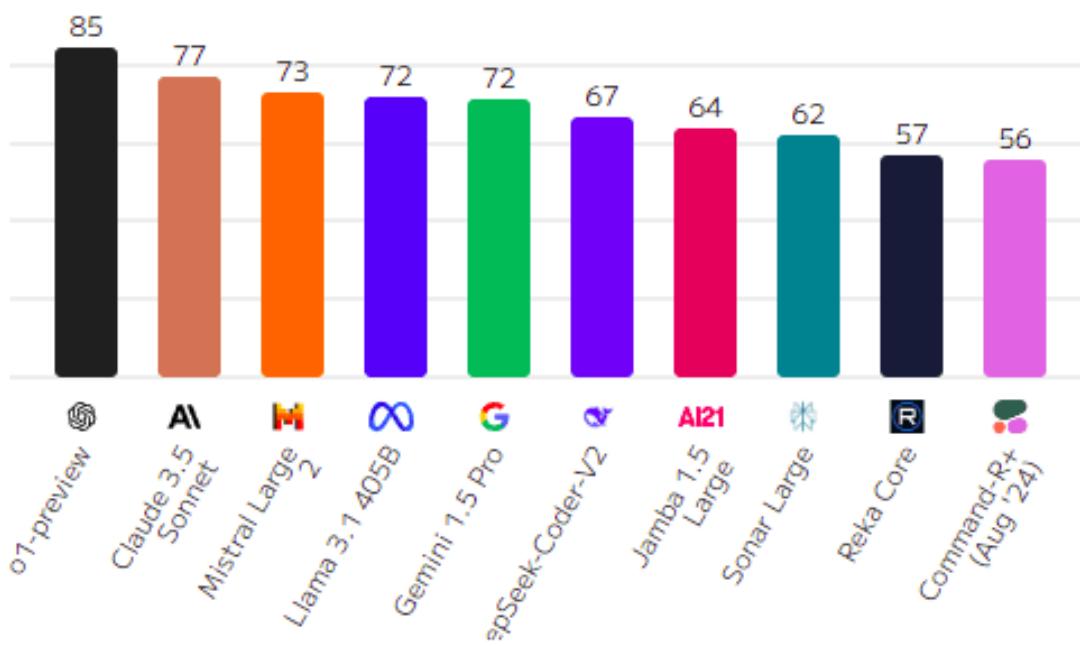
0 < 100%

на уровне ребенка
**объясняет свои
ответы**

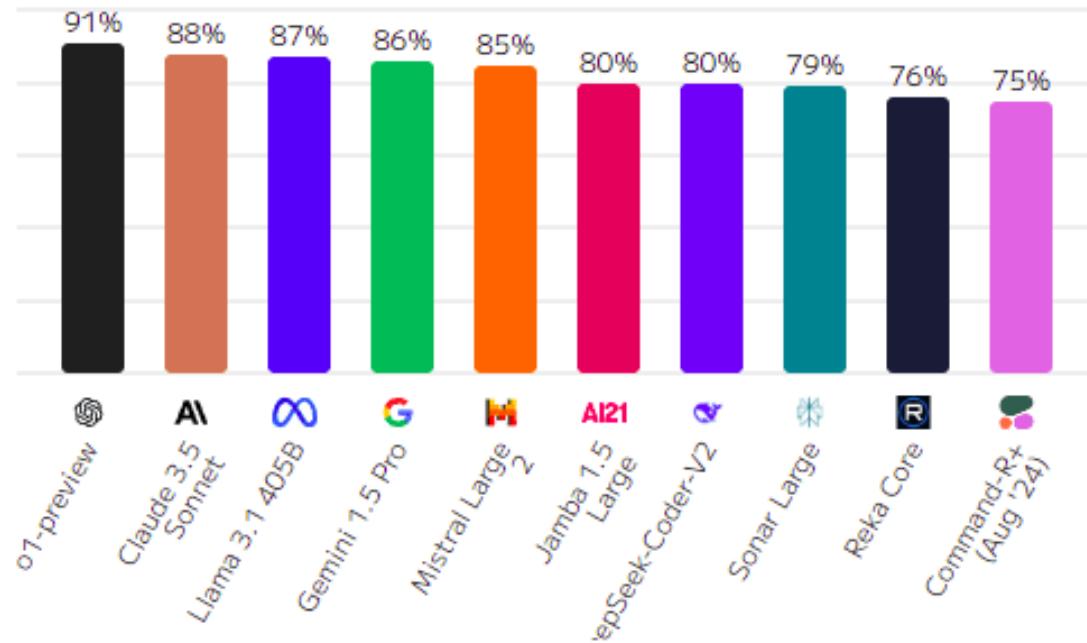
72% < 85%

Индекс качества

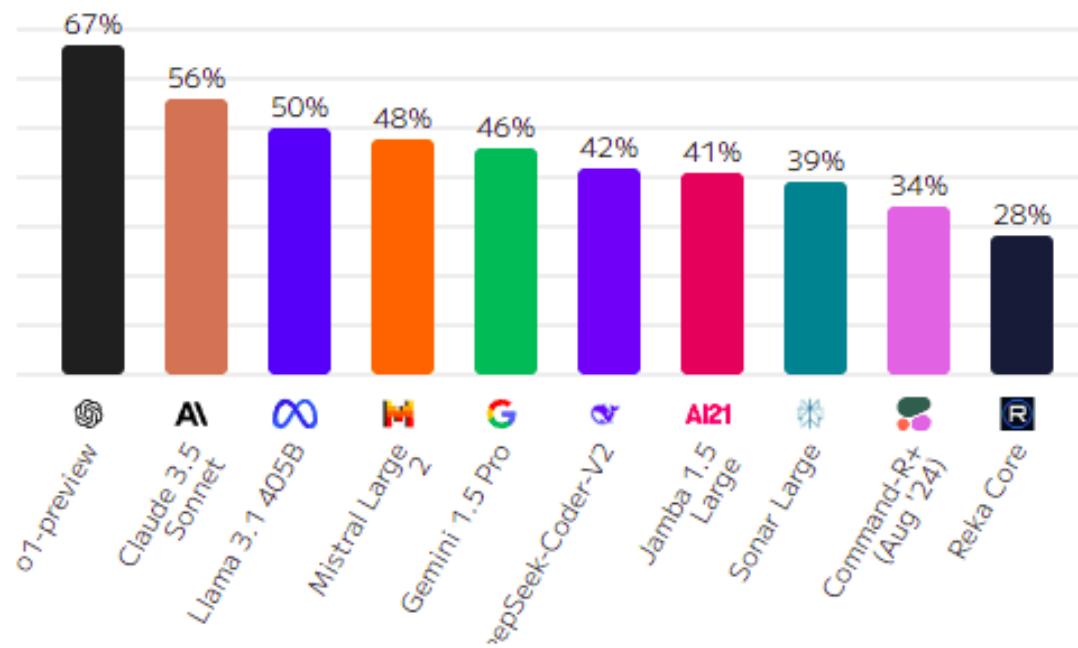
Artificial Analysis Quality Index

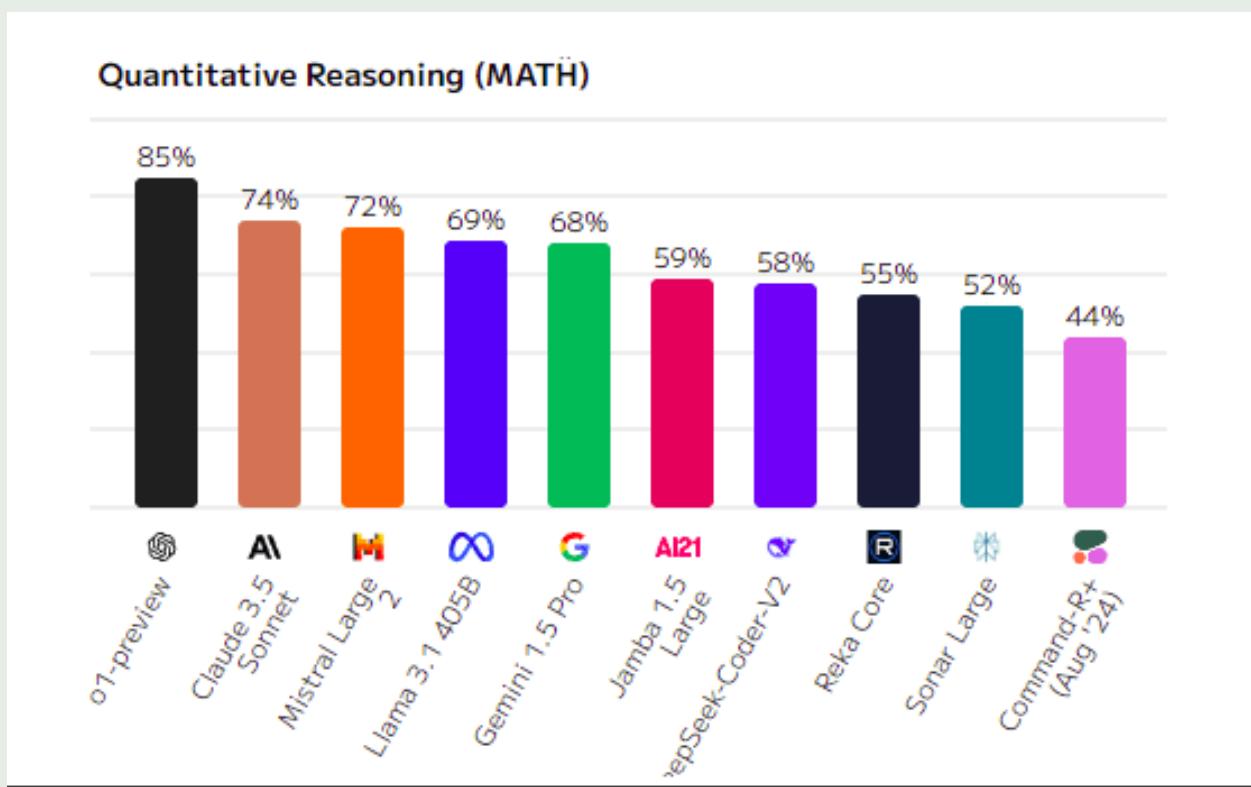


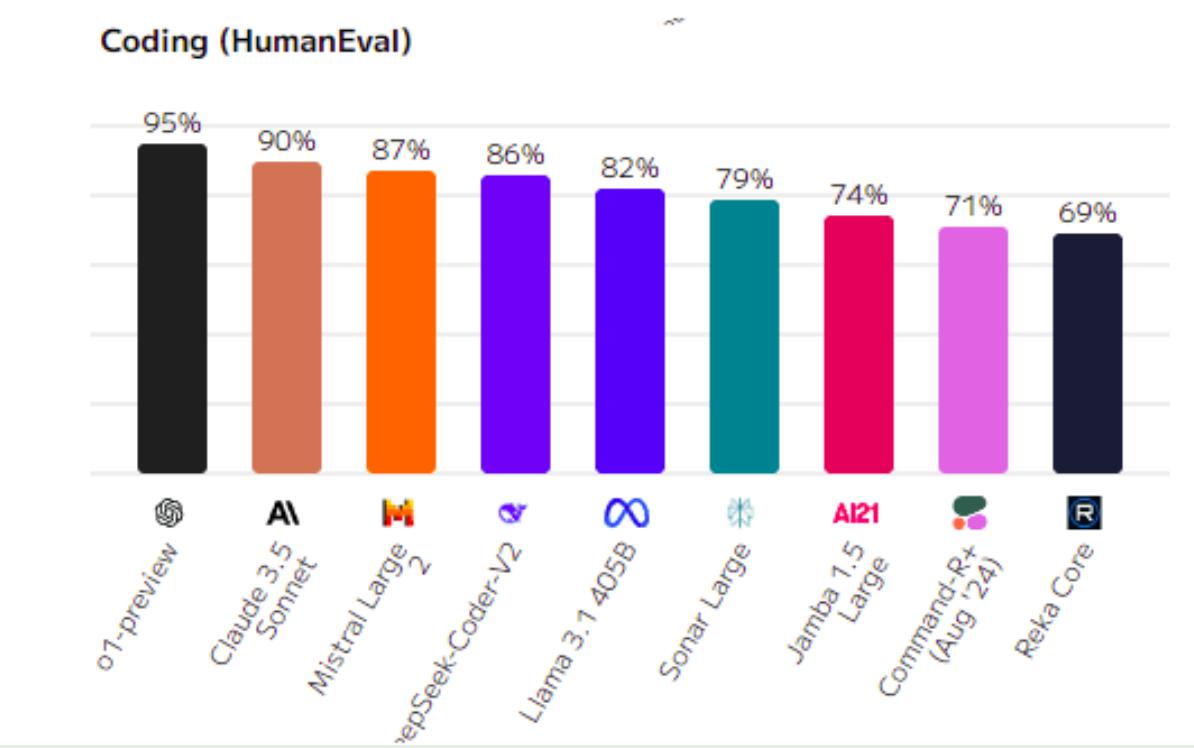
Reasoning & Knowledge (MMLU)



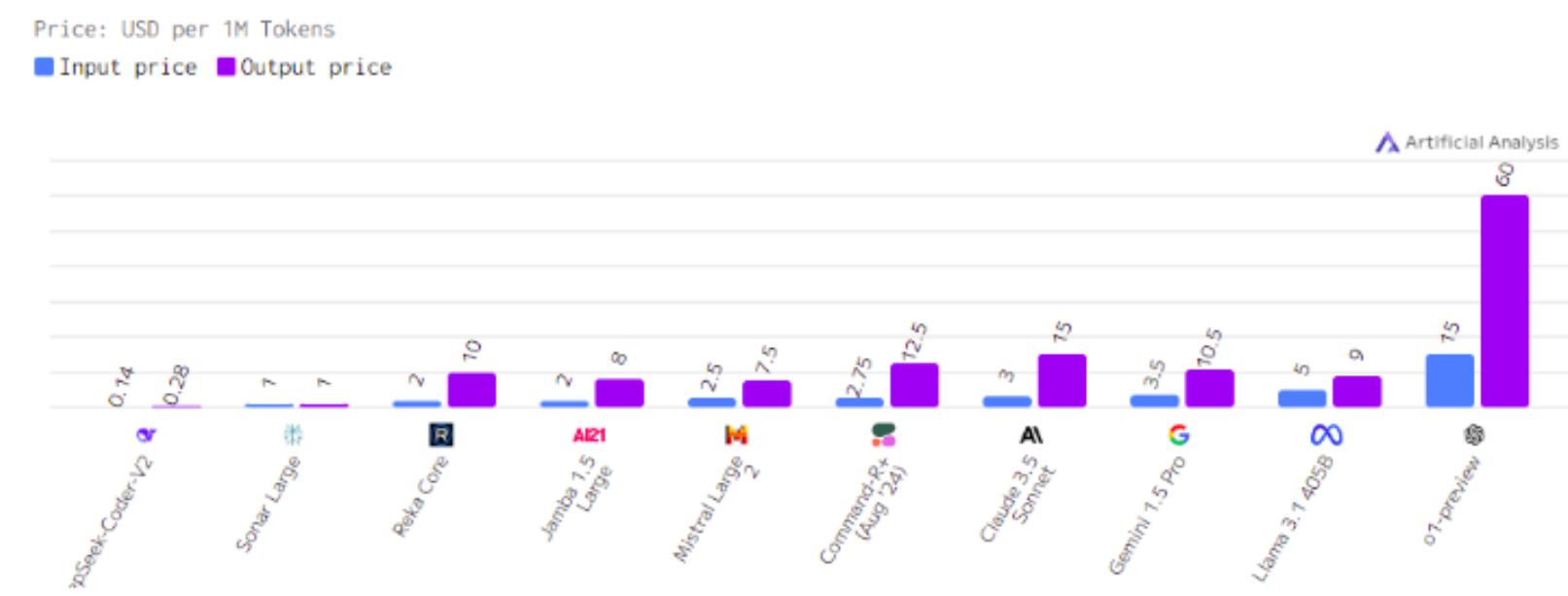
Scientific Reasoning & Knowledge (GPQA)







Стоимость обучения



Настоящее - будущее

Новые модальности

ИИ сегодня

- Транзакции (скоринг)
- Текст (NLP)
- Программный код (PLP)
- Речь (распознавание и синтез)
- Изображения (CV)
- Музыка (распознавание и генерация)

Базовые модальности
гигабайты и терабайты данных

В ближайшие 3-4 года

- Видео и 3D объекты
- Жесты и движения
- Большие графы знаний и связей
- Геном человека
- Структура белков

Бесшовное переключение между
модальностями в real-time
терабайты данных в real-time

ИИ в 2035 году

- Нейроимпульсы
- Запахи
- Различные виды излучения
- Сейсмические данные
- Космическая активность
- ...

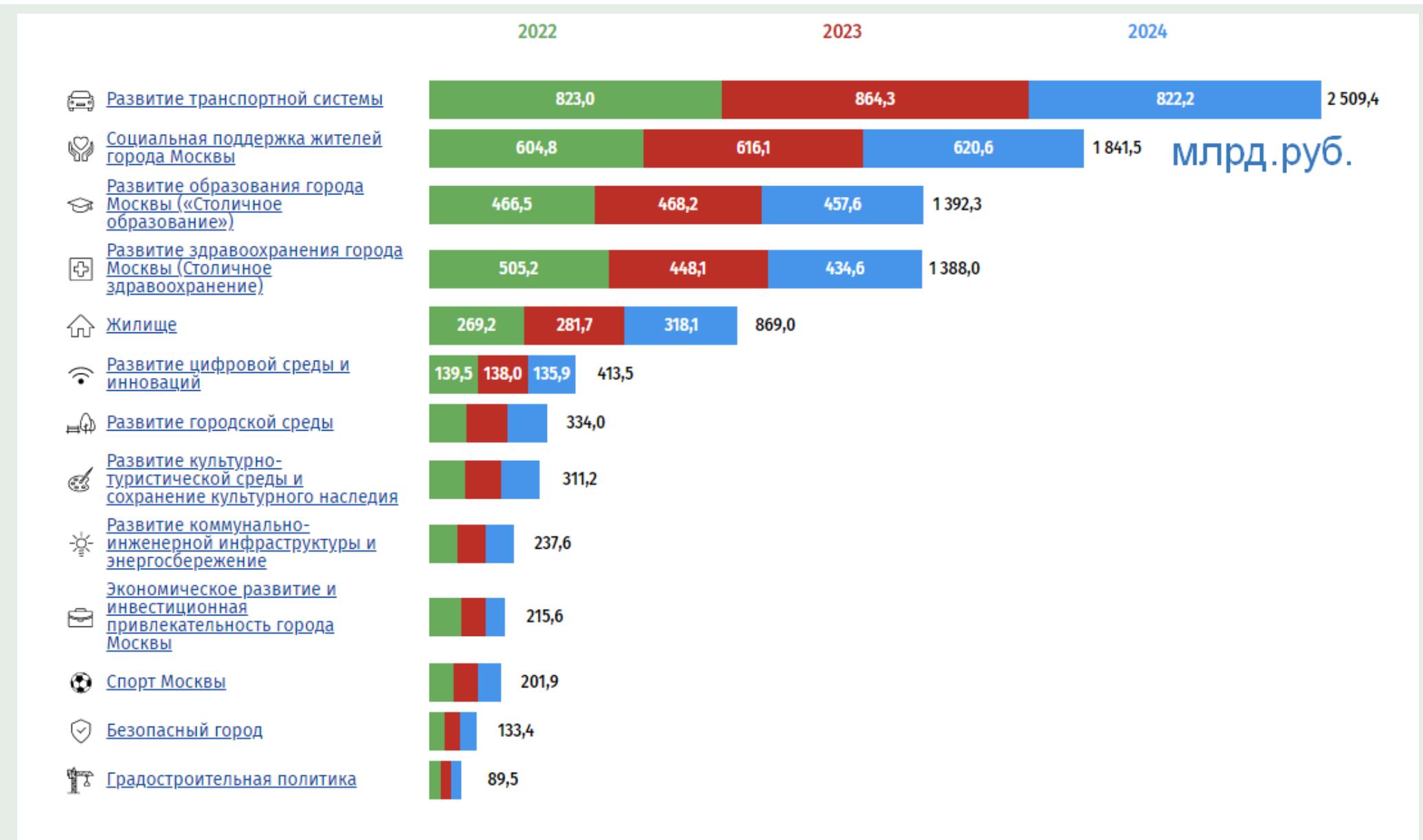
Абсолютное знание мира
экзабайты данных в real-time

- Большие данные.
- Нейротехнологии и искусственный интеллект.
- Интернет вещей.
- Системы распределенного реестра «Блокчейн».
- Технологии виртуальной и дополненной реальности.
- Квантовые компьютеры.

Основные инициативы национальной программы «Цифровая экономика России»

- Регулирование цифровой среды.
- Информационная инфраструктура.
- Кадры для цифровой экономики.
- Информационная безопасность.
- Цифровые технологии.
- Цифровое государственное управление.
- Искусственный интеллект.

Государственные программы города Москвы



Источник: <https://budget.mos.ru/budget/gp>

Направления цифровизации в городе Москве

Официальный портал Мэра и Правительства Москвы (mos.ru)

mos.ru Официальный сайт Мэра Москвы

Новости Афиша Услуги Мэр Власть Карта Мой район Инструкции Обратная связь

Каталог услуг для жителей / для бизнеса

Найти услугу

Популярное 10 >

Образование 66 >

Здоровье 40 >

Жилье, недвижимость, земля 81 >

Транспорт 38 >

Семья, дети 111 >

Социальная поддержка 171 >

Культура, спорт, отдых 44 >

Документы 111 >

Мировые судьи 1 >

Работа 19 >

Штрафы, налоги, оплата услуг 24 >

Инвалиды 20 >

Гражданская активность 2 >

Электронная медицинская карта

Электронный дневник школьника

Посмотреть информацию о посещении и питании в школе, детском саду, колледже

Запись в кружки, спортивные секции, дома творчества

Запись на прием к врачу, отмена и перенос записи

Получить и оплатить единый платежный документ (ЕПД)

Запись на прием в центры госуслуг и ведомства

Мои платежи

Прием показаний приборов учета воды и тепла

Прием показаний и оплата электроэнергии

Направления цифровизации в городе Москве

Портал «Наш город», проект «Активный гражданин» (gorod.mos.ru, ag.mos.ru)

The screenshot shows the homepage of the 'Nash Gorod' (Our City) website. At the top, there are two banners: one for 'MY PARK' in TINAO with 10 topics and another for 'MY HOME' in TINAO with 5 new topics. The main header 'МОСКАУСКИЙ ГРАЖДАНЕ' is displayed, along with a search bar and login links. The main content area features a large banner for a competition titled 'Большая викторина в «Активном гражданине»'. It includes a call-to-action button 'УЧАСТВОВАТЬ', a description of the competition, and illustrations of two people using mobile devices. Below this are three large numerical statistics: 5 912 358 active citizens, 5 380 proposals for discussion, and 187 271 856 responses received. A sidebar on the right lists news items, such as 'Подарки самим активным в честь 8-летия проекта' and 'Активные граждане выбрали названия новых улиц и станций метро'.

Проект «Электронный дом» (ed.mos.ru)

Собрания собственников Вопросы и ответы База знаний О проекте Карта домов Обратная связь [Войти](#)

Электронный дом

Эффективный инструмент для управления многоквартирным домом онлайн

Платформа для взаимодействия жителей и управляющих организаций

Скачайте приложение

Загрузите в App Store Доступно в Google Play

СКАЧАТЬ ИНСТРУКЦИЮ



Проект «Московская электронная школа» (school.mos.ru)

МЭШ

Цифровое образование в школах Москвы

Московская Электронная Школа помогает учиться лучше, быть в курсе всех школьных событий и готовиться к урокам с новыми технологиями. В любое время и с любых устройств.



Вход

Войти через mos.ru

для жителей регионов

госуслуги

Как получить доступ?

Уважаемые пользователи!

Добро пожаловать на новую страницу МЭШ. Теперь для всех пользователей доступна возможность входа в сервисы МЭШ под единой учетной записью.

Подробнее



Информационная поддержка

Для родителей и учеников

Для сотрудников

Написать нам

© Департамент Информационных Технологий города Москвы
Департамент образования и науки города Москвы

Направления цифровизации в городе Москве

Единая медицинская информационно-аналитическая система города Москвы (emias.info)

The screenshot displays the main interface of the Emias.info website. At the top, there is a navigation bar with links for 'Запись к врачу' (Appointment with doctor), 'Учреждения' (Institutions), 'Справки' (Certificates), 'Медкарта' (Medical card), and 'Профилактика' (Prevention). A 'Войти' (Log in) button is also present. Below the navigation, a large blue banner promotes 'Экспресс-тестирование на COVID-19' (Express testing for COVID-19) with text explaining the process and a 'Оформить заявку' (Apply for application) button. To the right of the banner are two stylized virus icons. The main content area features a section titled 'Запись к врачу в городские поликлиники Москвы' (Appointment with doctors in Moscow city clinics) with fields for 'Номер полиса' (Policy number) and 'Дата рождения' (Date of birth), both with placeholder text. A 'Записаться' (Book) button is located next to the date field. To the right of this form is a link for 'Экспресс-тест на COVID-19'. Below the main form, there are three colored buttons: pink for 'Запись к врачу' (Appointment with doctor), yellow for 'Учреждения' (Institutions), and light purple for 'Справки' (Certificates). At the bottom of the page, there are footer links for 'Что такое полис ОМС?' (What is an OMC policy?), 'Условия использования' (Usage conditions), and 'Как работает запись?' (How does booking work?).

Перечень некоторых других городских цифровых сервисов

- Городская система видеонаблюдения «Единый центр хранения и обработки данных» (echd.mos.ru)
- Автоматизированная система учета потребления ресурсов (asupr.mos.ru)
- Единый диспетчерский центр города Москвы (edc.mos.ru)
- Портал открытых данных города Москвы (<https://data.mos.ru/>)

Направления цифровизации в городе Москве

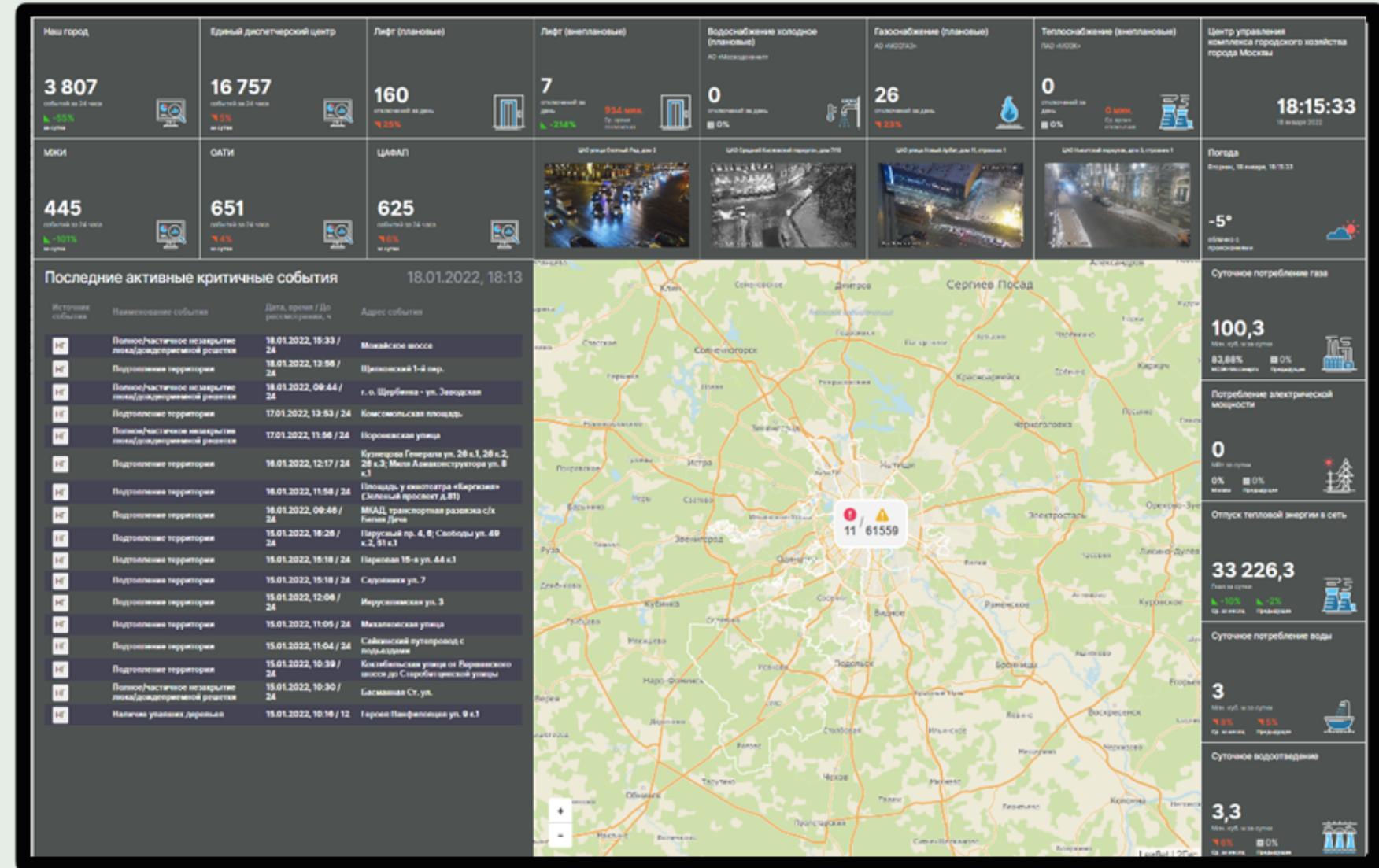
Ситуационный зал центра управления комплекса городского хозяйства



Источник: материалы доклада руководителя ГБУ МАЦ

Направления цифровизации в городе Москве

Главная информационная панель центра управления комплекса городского хозяйства



Нужно бежать со всех ног,
чтобы только оставаться
на месте, а чтобы куда-то
попасть, надо бежать как
минимум вдвое быстрее!

«Алиса в стране чудес»,
Льюис Кэрролл

Источник: (не)случайные генерации
<https://t.me/KandinskyTV>

