ITMO

# Neuron-Level Architecture Search for Efficient Model Design

Лунев Артем Евгеньевич

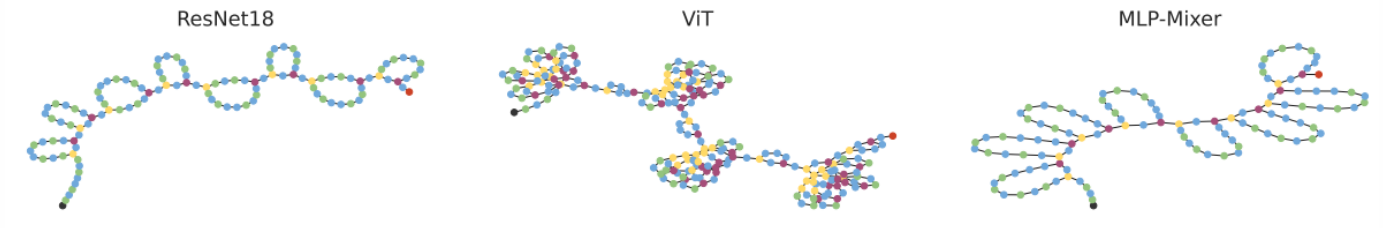
Научный руководитель: к.т.н. Никитин Н.О.

# Гипотеза

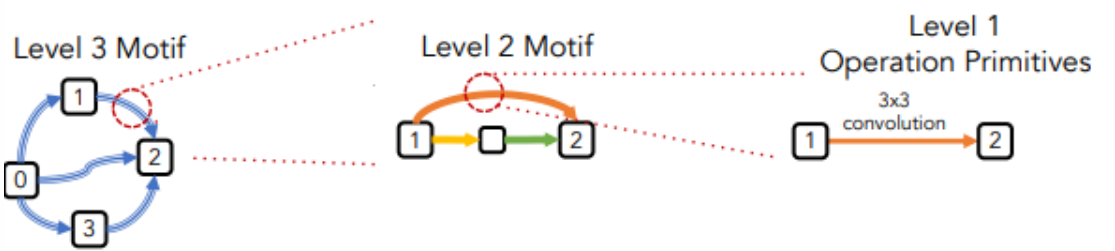
Моделирование и оптимизация архитектур нейронных сетей на уровне отдельных нейронов и их связей позволяет создавать более вычислительно эффективные модели за счёт устранения структурной избыточности и повышения адаптивности архитектуры.

# Обзор схожих решений

- AutoML-Zero: поиск алгоритмов машинного обучения из примитивных операций
- Einspace: пространство поиска на основе параметризованных контекстно-свободных грамматик, включающая множество архитектур




- Иерархические пространства поиска




1. AutoML-Zero: Evolving Machine Learning Algorithms From Scratch
2. einspace: Searching for Neural Architectures from Fundamental Operations

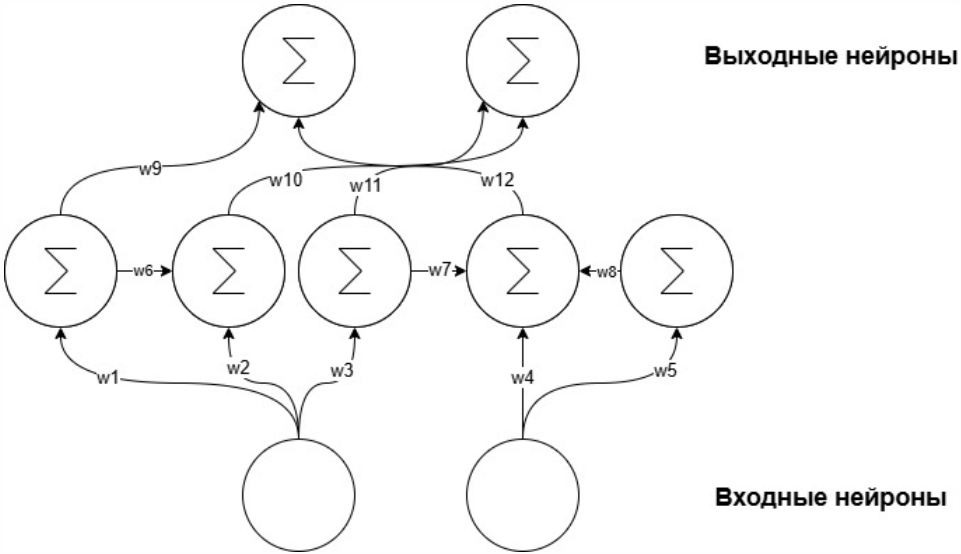
# Общий подход к NAS с использованием эволюционных алгоритмов

**Algorithm 2** General Evolutionary NAS Algorithm

**Input:** Search space $\mathcal{A}$, number of iterations $T$.
Randomly sample and train a population of architectures from the search space $\mathcal{A}$.
**for** $t = 1, \ldots, T$ **do**
  Sample (based on accuracy) a set of parent architectures from the population.
  Mutate the parent architectures to generate children architectures, and train them.
  Add the children to the population, and kill off the architectures that are the oldest (or have the lowest accuracy) among the current population.
**end for**
**Output:** Architecture from the population with the highest validation accuracy.

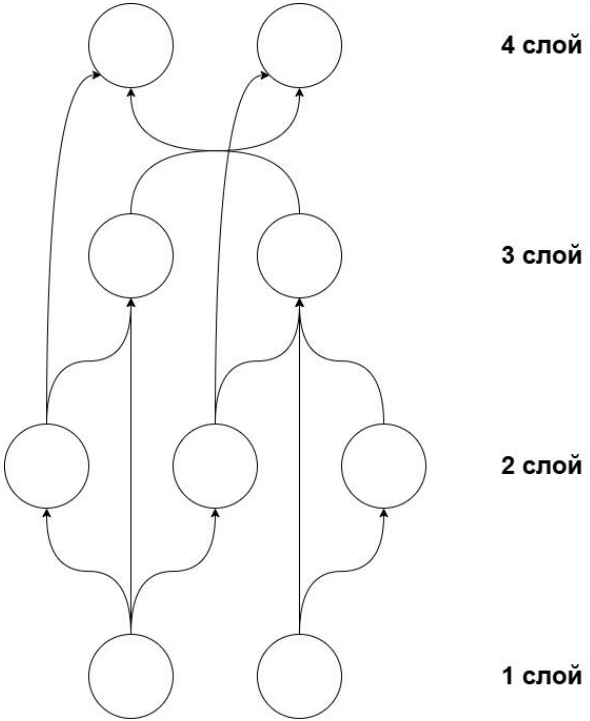
* C White, M Safari. Neural Architecture Search: Insights from 1000 Papers

# Представление архитектур (часть 1)




- Архитектура представляет собой DAG
- Веса нейронной сети связаны с ребрами
- Вершины представляют собой нейроны, выполняющие взвешенное суммирование сигналов от входящих нейронов
- Входные и выходные нейроны изначально фиксированы
- Функция активации для всех нейронов ReLU. За исключением выходных нейронов

# Представление архитектур (часть 2)

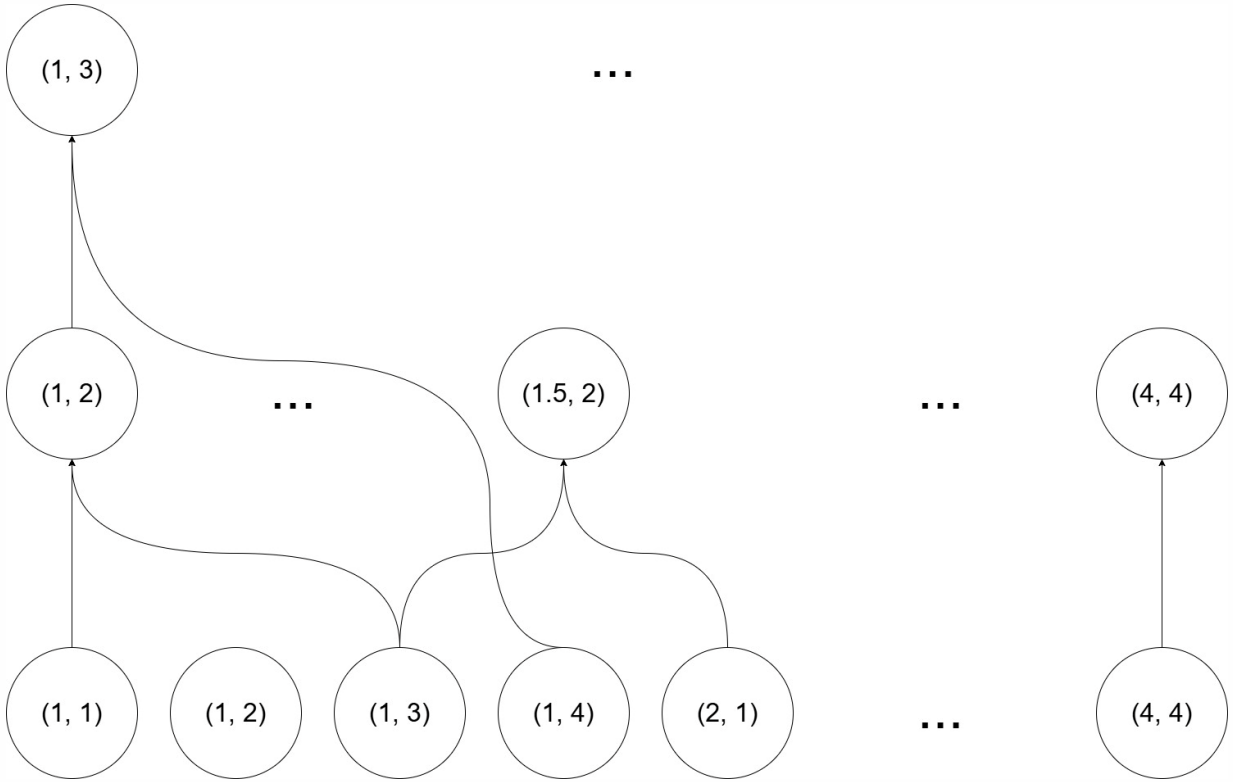



- Топологическая сортировка выстраивает нейроны по слоям
- Для каждой пары слоев, между которыми существуют связи, создается матрица весов

Пример:

- Разряженные матрицы весов $W_{12}$ (2x3), $W_{13}$ (2x2), $W_{23}$ (3x2), $W_{24}$ (3x2), $W_{34}$ (2x2)
- выход 3 слоя будет вычисляться как ReLU(LayerNorm($out_1 * W_{13} + out_2 * W_{23}$))

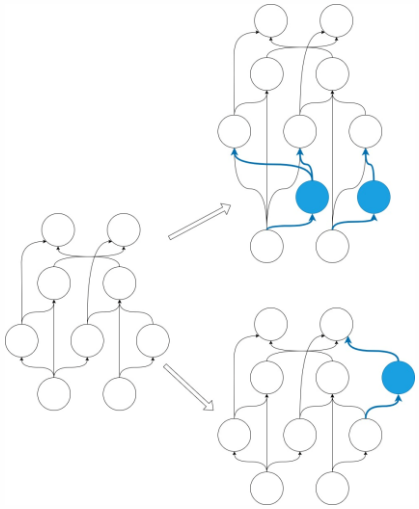
# Определение пространственных отношений нейронов



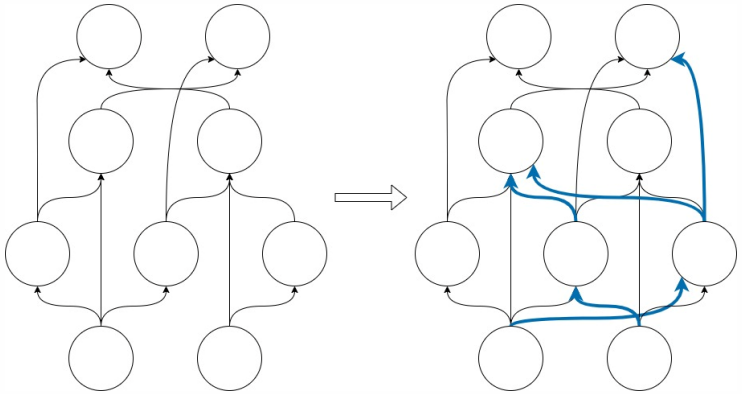
# Мутации (часть 1)

Во всех мутациях вероятность появления новых ребер задается в соответствие с пространственным отношением нейронов и вычисляется по формуле:

$$exp(-dist/(4 + layer * 1.5))$$

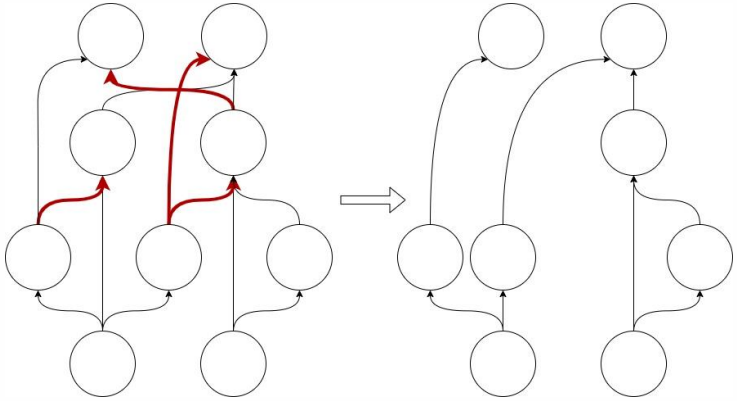


Добавление новых вершин и инцидентных им ребер

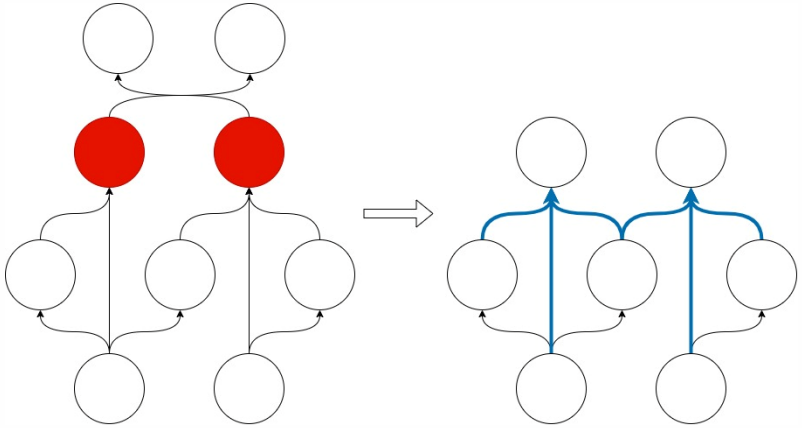


Добавление новых ребер между 3 парами слоев

# Мутации (часть 2)



Удаление существующих ребер между 3 парами слоев

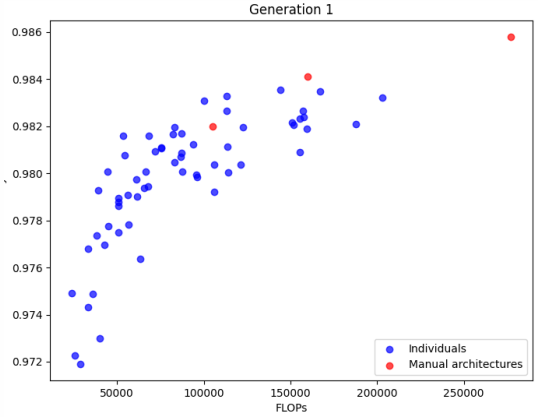


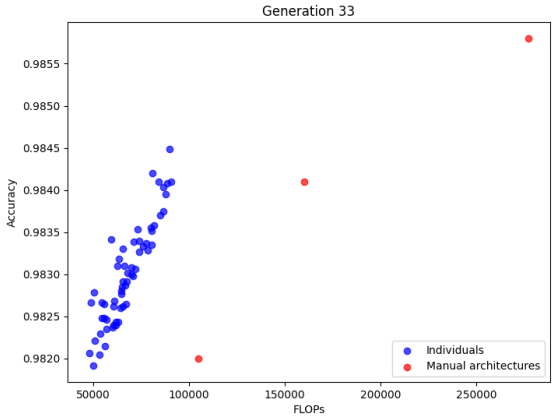
Удаление слоя и прилегающих к нему ребер

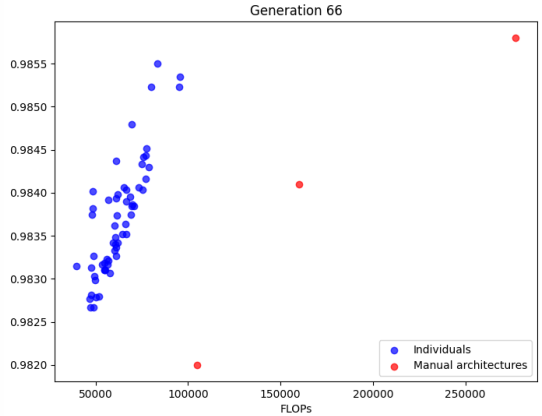
# Эволюционный алгоритм

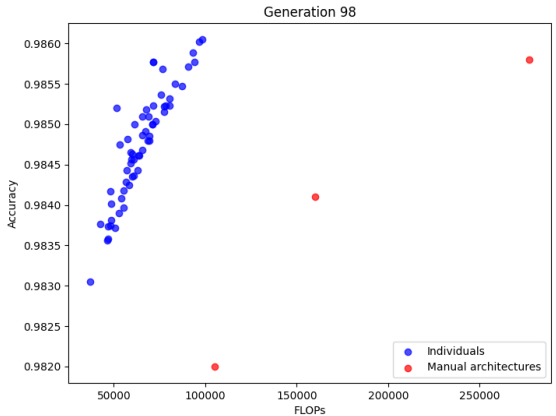
**Input:** number of iterations T, number of mutations per itration M
Initialize population $P_0$
Assign rank based on Pareto dominance to population individuals
**for** t = 1, ..., T **do**
    **for** m = 1, ..., M **do**
        Select parent from population $P_t$ with tournament selection based on rank and crowding distance
        Mutate parent to generate child, and train it
        Add child to population
    Remove M oldest individuals from population
    Assign new Pareto ranks to the population
**Output:** Architecture from the population with the highest validation accuracy or lowest FLOPs

# Результаты обучения на MNIST

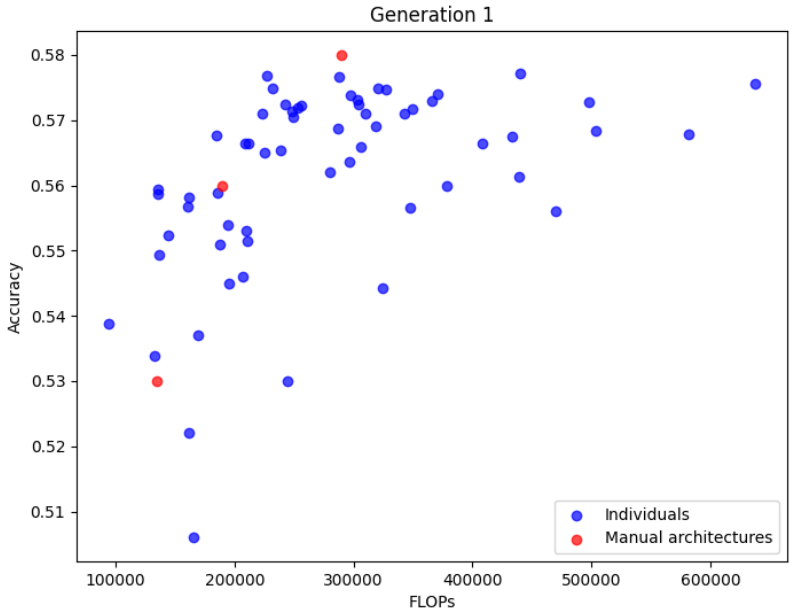


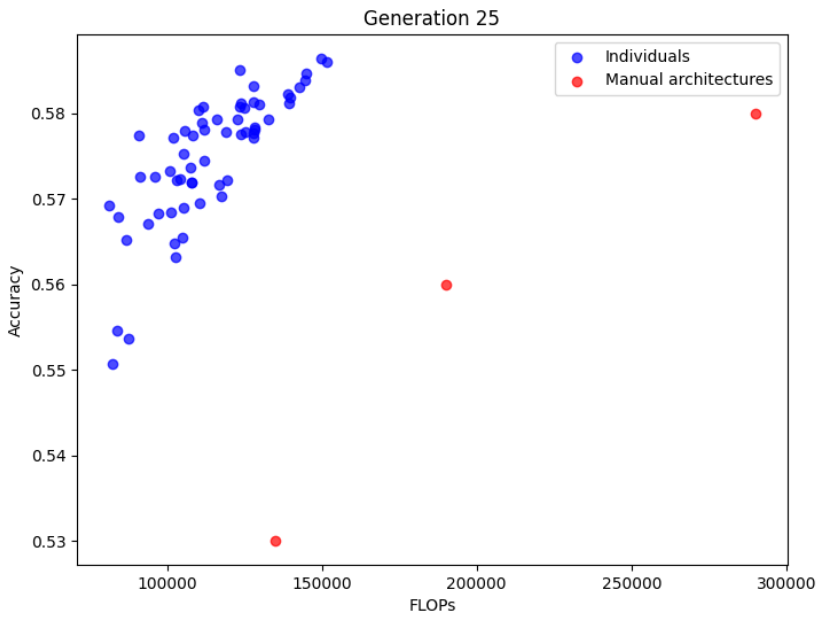



# Результаты обучения на CIFAR-10

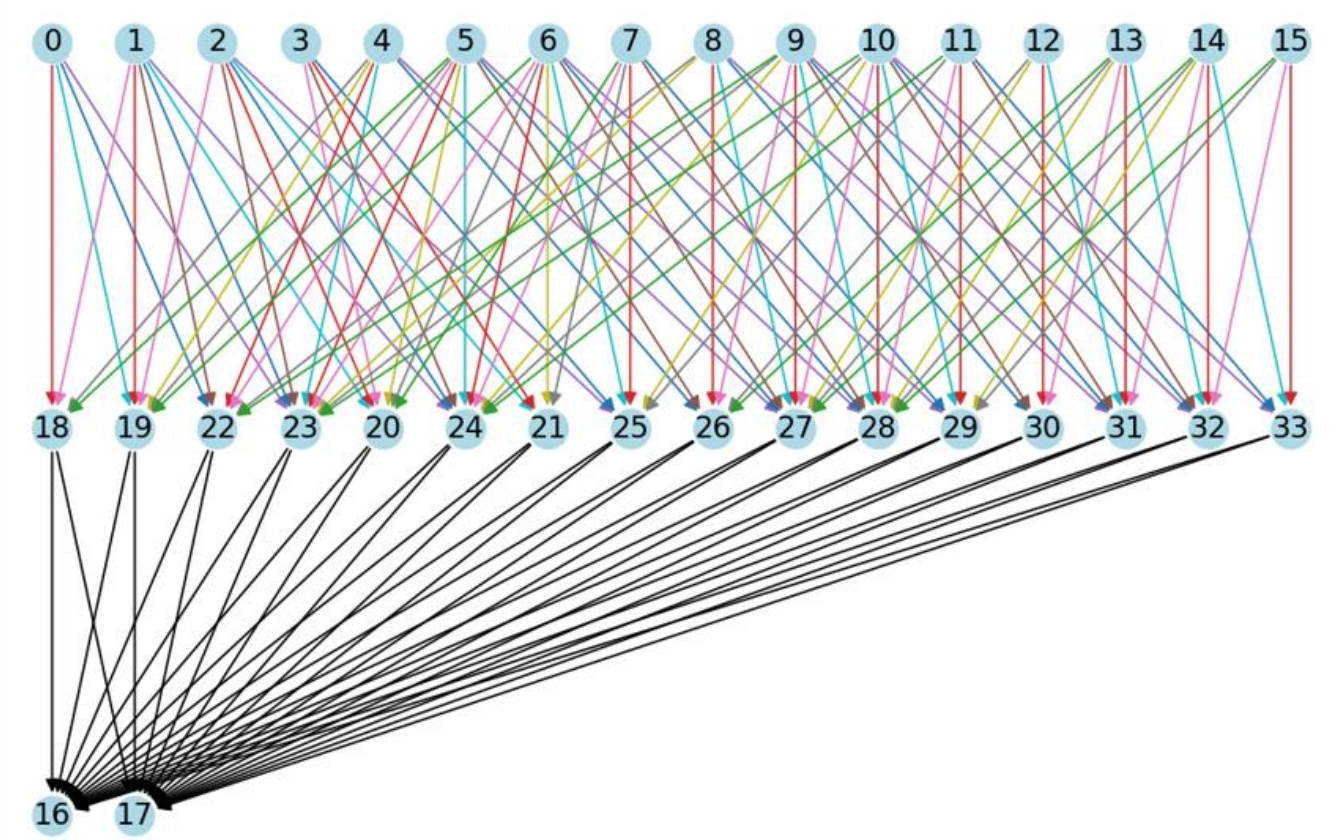


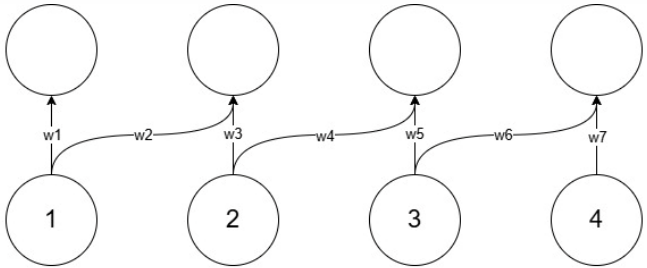
# THANK YOU
# FOR YOUR TIME!

ITMO more than a UNIVERSITY

# Дополнительный слайд. Мутация по дублированию весов



# Дополнительный слайд. Подход к умножению разряженных матриц




| | |
|---|---|
| 1 | 0 |
| 1 | 2 |
| 2 | 3 |
| 3 | 4 |

*

| | |
|---|---|
| w1 | 0 |
| w2 | w3 |
| w4 | w5 |
| w6 | w7 |

=

+

| | |
|---|---|
| | |
| | |
| | |
| | |