

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ "МОСКОВСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ имени М.В. ЛОМОНОСОВА"

ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА НАНОФОТОНИКИ

## КУРСОВАЯ РАБОТА

# ОПТИЧЕСКИЕ МЕТОДЫ СЖАТИЯ ИЗОБРАЖЕНИЙ В ПРОСТРАНСТВЕННО НЕКОГЕРЕНТНОМ СВЕТЕ ДЛЯ НЕЙРОННЫХ СЕТЕЙ

Выполнил:  
студент 437 группы  
Белашов Е. Ю.

Научный руководитель:  
к. ф.-м. н. А. В. Четвертухин.

Москва, 2023

## Оглавление

1. Введение .....	1
2. Обзор литературы .....	1
2.1. Нейронные сети .....	1
Идея нейронной сети . . . . .	1
Современные тенденции в области нейронных сетей . . . . .	4
2.2. Оптические вычислители. ....	6
3. Оригинальная часть. ....	7
4. Заключение .....	7
<b>Список литературы</b> .....	<b>8</b>

## 1. Введение

Текст введения, напишу в самом конце

## 2. Обзор литературы

### 2.1. Нейронные сети

Задача развитие искусственного интеллекта является приоритетной для Российской Федерации [1]. Нейронные сети предназначены для задач, которые не могут быть формализованы явным образом. Примером таких задач являются задачи распознавания и классификации объектов на изображениях. Такие проблемы также называют когнитивными, потому что с ними хорошо справляется человеческий мозг, но плохо справляются классические компьютерные алгоритмы.

#### Идея нейронной сети

Первая нейронная сеть была предложена в [2] и имела архитектуру перцептрона. Эта архитектура вдохновлена работой реальных нейронов в биологических нейронных сетях [3]. Каждый нейрон слоя в такой сети принимает множество сигналов с предыдущего слоя, умножает их на веса, суммирует, прибавляет смещение и применяет к полученному значению нелинейную функцию активации, после чего сигнал передаётся на следующий слой.

$$I_{l,i} = F \left( b_i + \sum_j w_{i,j} I_{l-1,j} \right) \quad (1)$$

Где  $F$  - нелинейная функция активации,  $w_{i,j}$  - вес связи  $i$ -го и  $j$ -го нейрона,  $b_i$  - смещение сигнала  $i$ -го нейрона,  $I_{l,i}$  - сигнал  $i$ -го нейрона  $l$ -го слоя,  $I_{l-1,j}$  - сигнал  $j$ -го нейрона  $l - 1$ -го слоя. Переход от слоя к слою в векторном представлении:

$$\vec{I} = F \left( \vec{J} \hat{w} + \vec{b} \right) \quad (2)$$

Где  $\vec{I}$ ,  $\vec{J}$  - вектор сигналов текущего и предыдущего слоя соответственно,  $\hat{w}$  - матрица весов перехода,  $\vec{b}$  - вектор смещения. По своей сути, такой переход между слоями, до применения функции

активации, реализует максимальную линейную связь двух  $\vec{I}$  и  $\vec{J}$ . Нейронная сеть типа перцептрон формируется путём совмещения множества таких слоёв разного размера. Если в такой структуре отсутствовали бы функции активации между слоями, то работа всей нейронной сети могла бы быть представлена, как совокупность линейных переходов, т.е. линейный переход. Единственный слой и так реализует максимальную линейную связь между двумя векторами и всё структуру можно было бы заменить на один слой. Поэтому, без нелинейной функции активации, нету смысла увеличивать количество слоёв. В свою очередь, увеличение количества слоёв ведёт к увеличению весов нейронной сети и к её общему усложнению.

Для обучения нейронной сети нужно иметь большой обучающий набор входных векторов и функцию ошибки, которая по выходу нейронной сети вычисляет оценку правильности выходного вектора. Например, если для каждого входного вектора из обучающего набора, известен желаемый выходной вектор, то функция ошибки является мерой разницы между желаемыми выходными векторами и векторами, полученным в результате обработки нейронной сетью входных векторов из обучающего набора. Введём обозначения:  $\vec{x}_i$  -  $i$ -ый вектор обучающего набора,  $\vec{y}_i$  - вектор, полученный в результате обработки нейронной сетью  $\vec{x}_i$ ,  $M(\dots)$  - функция нейронной сети -  $\vec{y}_i = M(\vec{x}_i)$ ,  $\hat{w}$  - ли-неаризованный набор параметров всех слоёв нейронной сети. Тогда функция ошибки будет иметь вид:

$$L(\vec{y}_1, \vec{y}_2, \dots) = L(M(\vec{x}_1), M(\vec{x}_2), \dots) = L(\vec{x}_1, \vec{x}_2, \dots, \hat{w}) = L(\hat{w}) \quad (3)$$

Обучающий набор постоянен, поэтому функция ошибки является функцией весов нейронной сети. Обучение заключается в минимизации функции ошибки путём варьирования параметров. Основными способами поиска глобального минимума функции ошибки являются различные модификации алгоритма градиентного спуска. Суть градиентного спуска заключается в вычислении частных производных функции ошибки по весам. Вектор этих производных является направлением, вдоль которого функция ошибки растёт наиболее сильно. Поэтому, веса изменяются в обратном направлении. На рисунке 1 из [4] отображена визуализация поверхности функции ошибки в зависимости от двух параметров и нарисована траектория изменения двух весов нейронной сети.

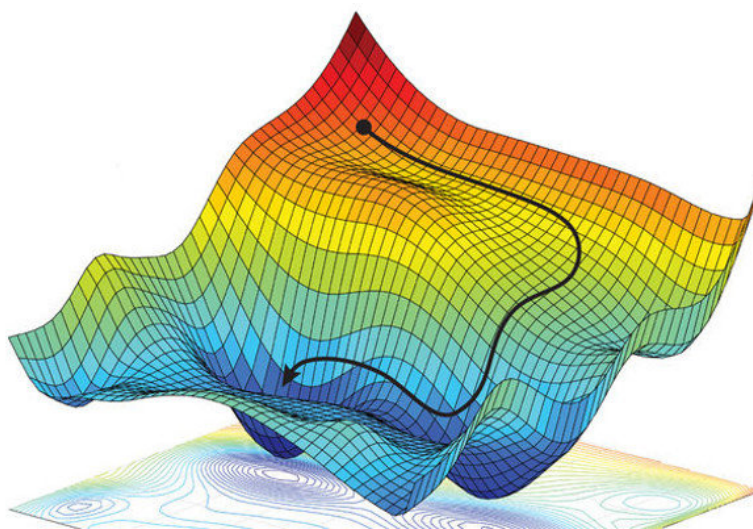


Рис. 1: Визуализация функции ошибки в зависимости от двум весов нейронной сети и траектория изменения этих весов.

Сегодня, помимо перцептрона существует большое множество архитектур нейронных сетей. На

рисунке 2 из [5] отображено их многообразие. Конкретные архитектуры лучше подходят для конкретных задач, в таблице 1 отображены характеристики и особенности некоторых архитектур.

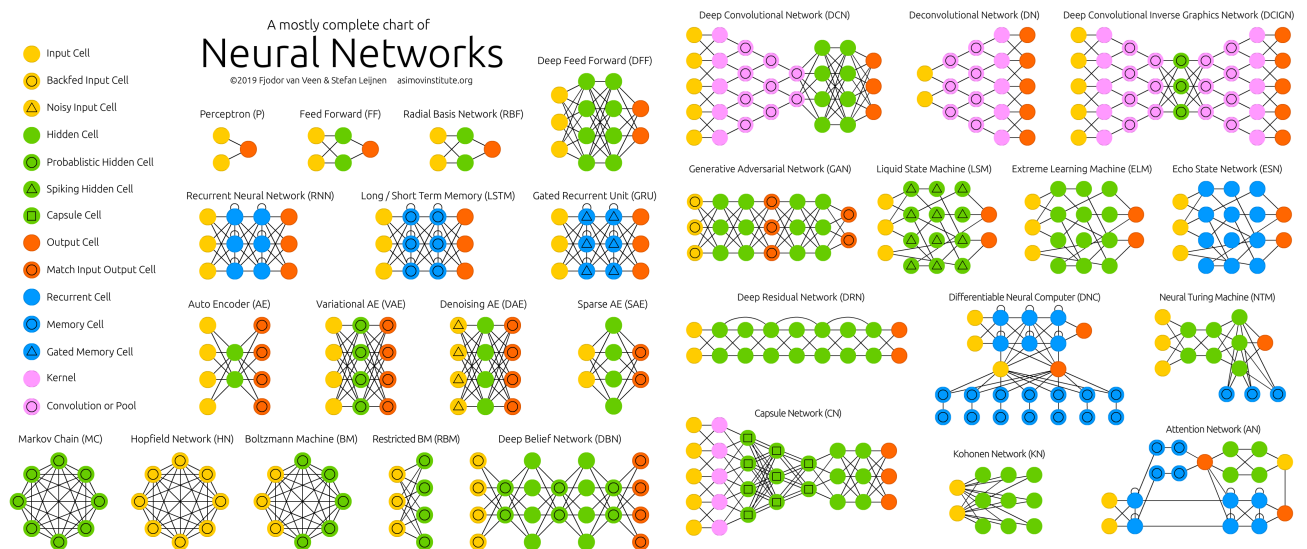


Рис. 2: Архитектуры нейронных сетей.

Стоит понимать, что нейронная сеть это лишь многомерная параметризованная и очень сложная функция функция  $\vec{y} = M(\vec{x}, \hat{w})$ . При обучении подбираются параметры этой функции в зависимости от конкретной задачи. Чем сложнее эта функция и чем больше у неё параметров, тем выше вероятность того, что существует их комбинация, минимизирующая функцию ошибки достаточно, чтобы можно было считать, что нейронная сеть выполняет поставленную задачу.

Архитектура	Идеально подходит для	Особенности
Перцептроны (FFNN)	Простые задачи классификации и регрессии	Простая структура, информация передается напрямую от входа к выходу
Рекуррентные сети (RNN)	Обработка последовательностей, текст, речь	Могут обрабатывать данные последовательно, сохраняя информацию о предыдущих входах
LSTM	Сложные последовательности, длинные тексты	Используются ворота для контроля потока информации, что помогает сохранять важные данные на длинных интервалах
GRU	Подобные LSTM задачи, но требуют меньше ресурсов	Упрощенная версия LSTM с меньшим количеством ворот, балансируют производительность и выразительность
Сверточные сети (CNN)	Распознавание изображений и видео	Используют свертки для эффективной работы с изображениями, улавливают пространственные иерархии
Генеративно-состязательные сети (GAN)	Генерация реалистичных изображений	Состоят из двух сетей (генератора и дискриминатора), которые соревнуются, улучшая друг друга
Автоэнкодеры (AE, VAE)	Сжатие данных, генерация данных	Обучаются сжимать входные данные в более мелкую форму и восстанавливать их
Капсульные сети (CapsNet)	Задачи, где важны пространственные отношения	Сети, использующие группы нейронов для сохранения информации о позиции и ориентации объектов
Трансформеры и сети внимания	Обработка естественного языка	Используют механизмы внимания для динамической фокусировки на разных частях данных

Таблица 1: Обзор архитектур нейронных сетей и их применений.

### Современные тенденции в области нейронных сетей

Сегодня, нейронные сети и глубокое обучение являются актуальными темами исследований [6]. На графике 3 из [6] показано, что, в период с 2000г по 2019г, количество публикаций, связанных с искусственным интеллектом, выросло примерно в 12 раз. На графике 4 из статьи [7] показано, как растёт вычислительная мощность современных вычислителей со временем. Из этой зависимости видно, что с 2015г по 2020г количество операций с 32-ух битным вещественным числом в секунду выросло примерно с  $2 \times 10^{12}$  до  $8 \times 10^{12}$ , т.е. в 4 раза. На другом графике 5 из [8] изображена зависимость параметров модели от года её публикации. В аналогичный период с 2015г по 2020г, количество параметров выросло примерно с  $5 \times 10^6$  до  $1 \times 10^9$ , т.е. примерно в  $10^3$  раз. Это сравнение показывает, что количество параметров нейронных сетей, которое связано с количеством операций необходимых для их исполнения, растёт намного быстрее, чем вычислительная мощность современных вычислителей. Таким образом, существует большой недостаток вычислительных мощностей, что также подтверждает компания OpenAI на иллюстрации 6. Для обучения больших моделей требуется создавать кластеры с большим количеством вычислителей. Так, компания OpenAI в 2023г сообщила, что им понадобится 30000 видеокарт Nvidia для обучения и исполнения их моделей ГПТ(Генеративный Предварительно-обученный Трансформер).

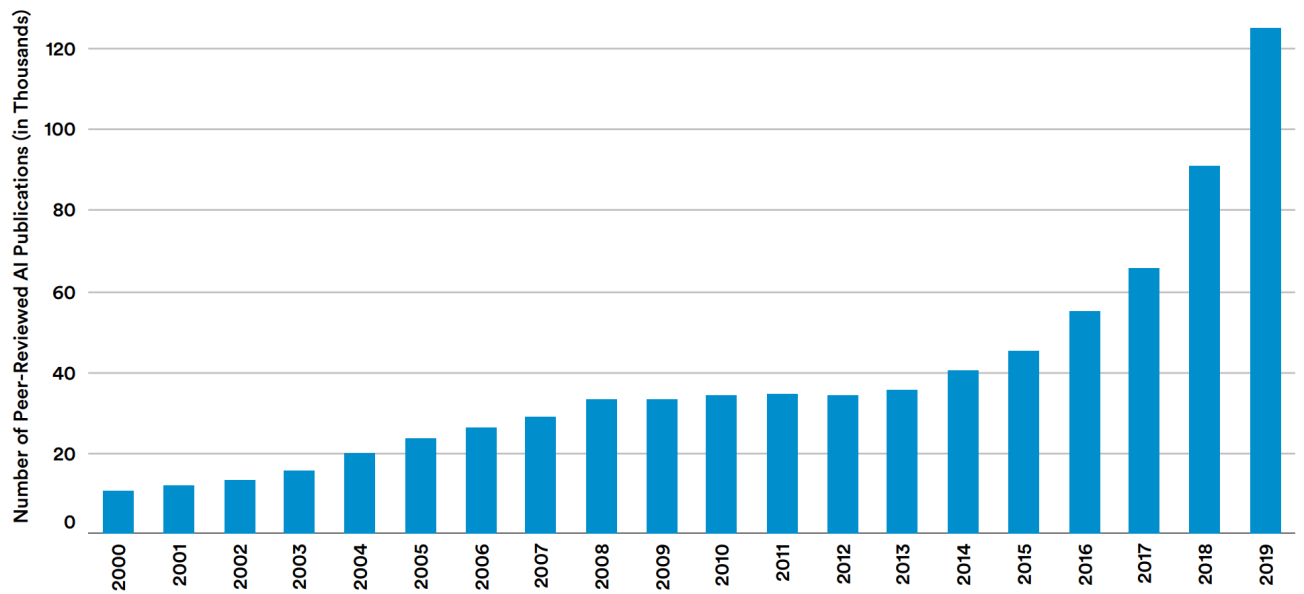


Рис. 3: Количество рецензированных публикаций по ИИ с 2000г по 2019г.

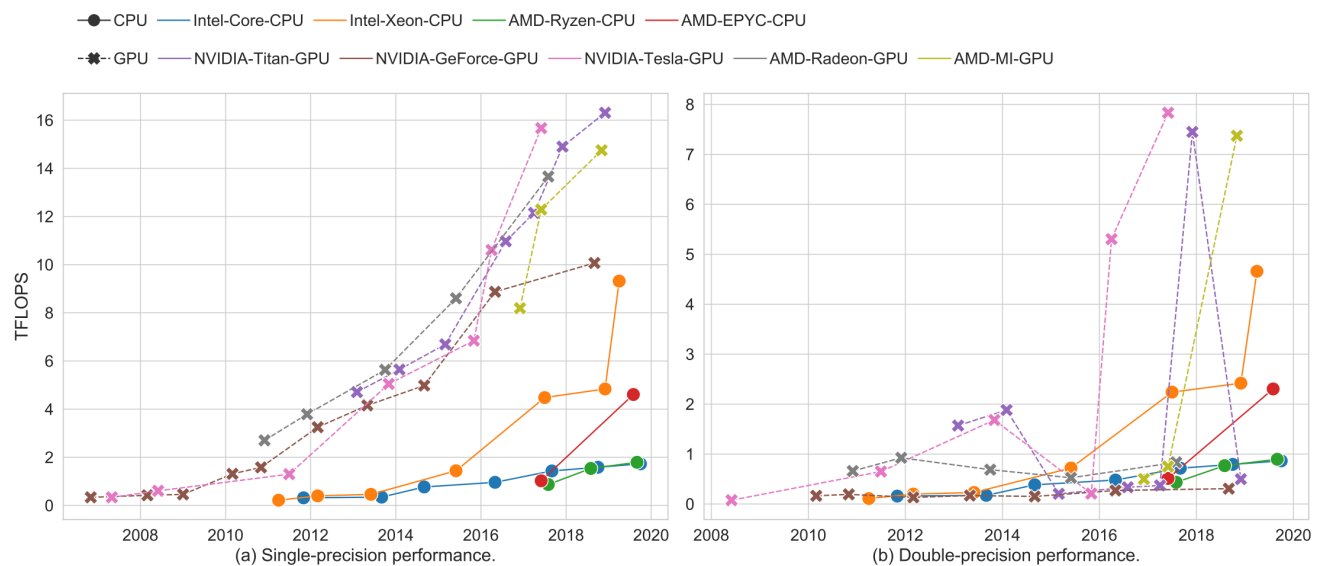


Рис. 4: Скорость работы вычислительных единиц, в операциях с 32/64 битным вещественным числом в секунду в зависимости от года выпуска.

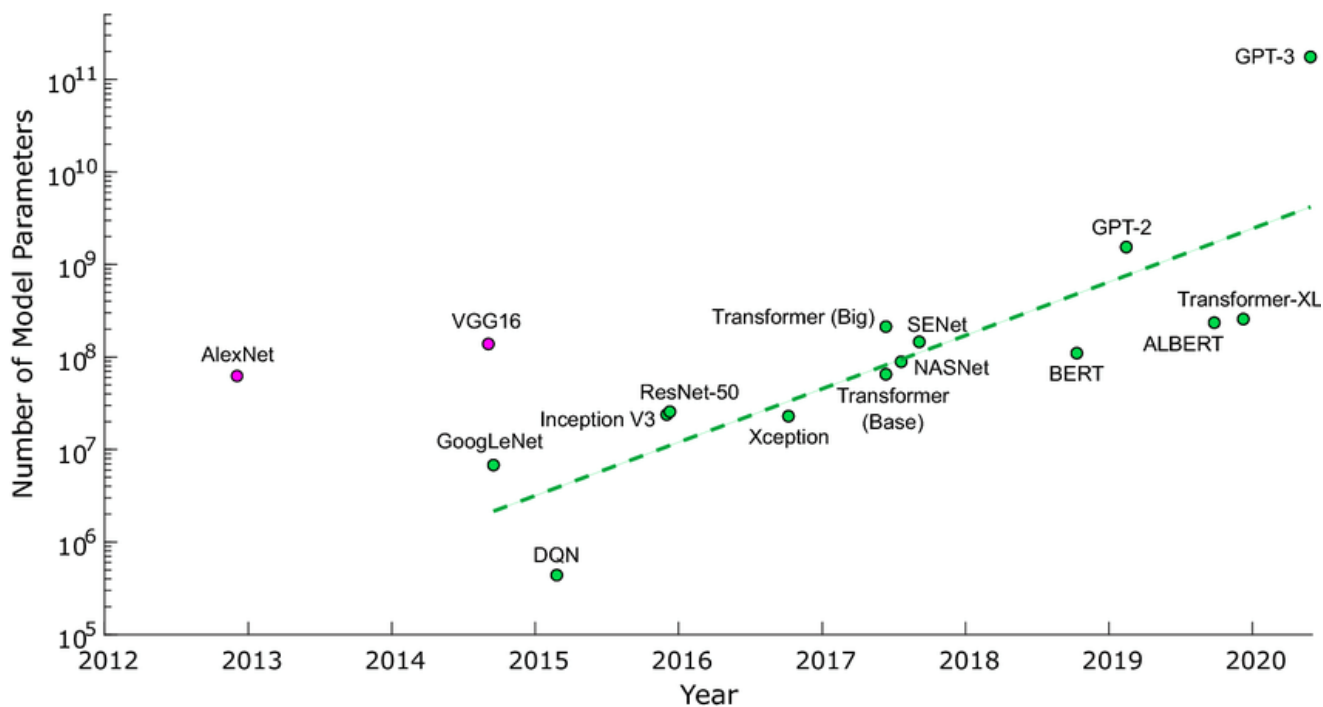


Рис. 5: Количество параметров модели в зависимости от года её публикации.

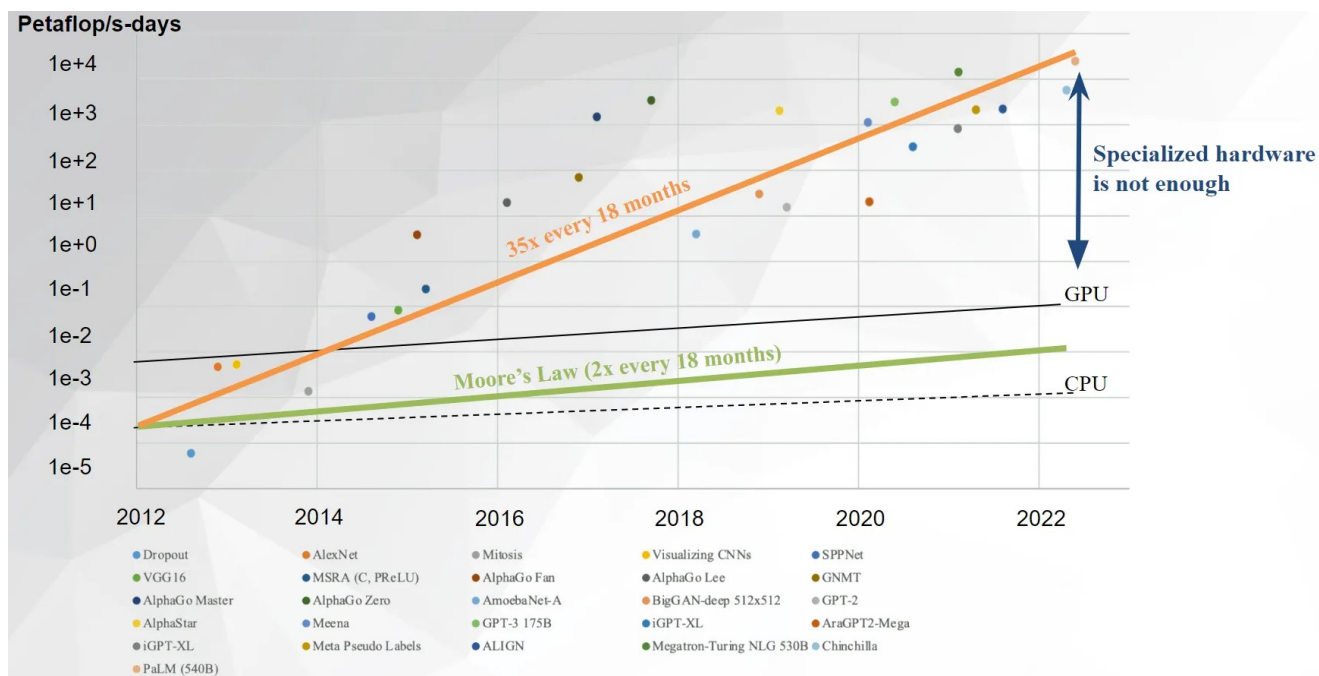


Рис. 6: Количество операций с плавающей точкой (вещественным числом) в секунду, умножить на день. \*Petaflops – days - величина похожая на  $KWt \times \text{Ч}$ , т.е. 1 Petaflops – days - вычисления со скоростью 1 операция с плавающей точкой в секунду в течение одного дня

## **2.2. Оптические вычислители**

Привет привет привет привет

## **3. Оригинальная часть**

## **4. Заключение**



## Список литературы

- [1] Указ Президента Российской Федерации от 10.10.2019 № 490 "О развитии искусственного интеллекта в Российской Федерации" [publication.pravo.gov.ru](https://publication.pravo.gov.ru) 2019-2030.
- [2] Frank Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain.// Psychological review – 1958. – Vol. 65, №6. – p. 386.
- [3] Hans-Dieter Block, The perceptron: A model for brain functioning. i// Reviews of Modern Physics – 1962. – Vol. 34, №1. – p. 123.
- [4] Alexander Amini, Ava Soleimany, Sertac Karaman, Daniela Rus, Spatial uncertainty sampling for end-to-end control// arXiv preprint [arXiv:1805.04829](https://arxiv.org/abs/1805.04829) – 2018. – , .
- [5] Stefan Leijnen, Fjodor van Veen The neural network zoo In: *Proceedings* V. 47 page 9. MDPI 2020.
- [6] Daniel Zhang, Saurabh Mishra, Erik Brynjolfsson, John Etchemendy, Deep Ganguli, Barbara Grosz, Terah Lyons, James Manyika, Juan Carlos Niebles, Michael Sellitto, others, The ai index 2021 annual report// arXiv preprint [arXiv:2103.06312](https://arxiv.org/abs/2103.06312) – 2021. – , .
- [7] Yifan Sun, Nicolas Bohm Agostini, Shi Dong, David Kaeli, Summarizing cpu and gpu design trends with product data// arXiv preprint [arXiv:1911.11313](https://arxiv.org/abs/1911.11313) – 2019. – , .
- [8] Liane Bernstein, Alexander Sludds, Ryan Hamerly, Vivienne Sze, Joel Emer, Dirk Englund, Freely scalable and reconfigurable optical hardware for deep learning// Scientific reports – 2021. – Vol. 11, №1. – p. 3144.