



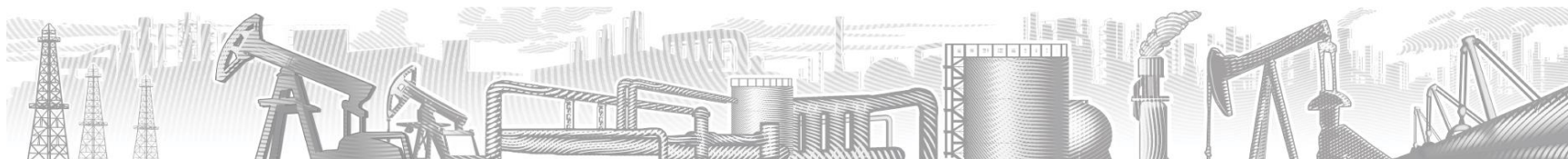
**УФИМСКИЙ ГОСУДАРСТВЕННЫЙ
НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**
ИНСТИТУТ НЕФТИ И ГАЗА (филиал в г. Октябрьском)



Кафедра «Информационные технологии, математика и естественные науки»

Курс ДОП
ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И НЕЙРОННЫЕ СЕТИ

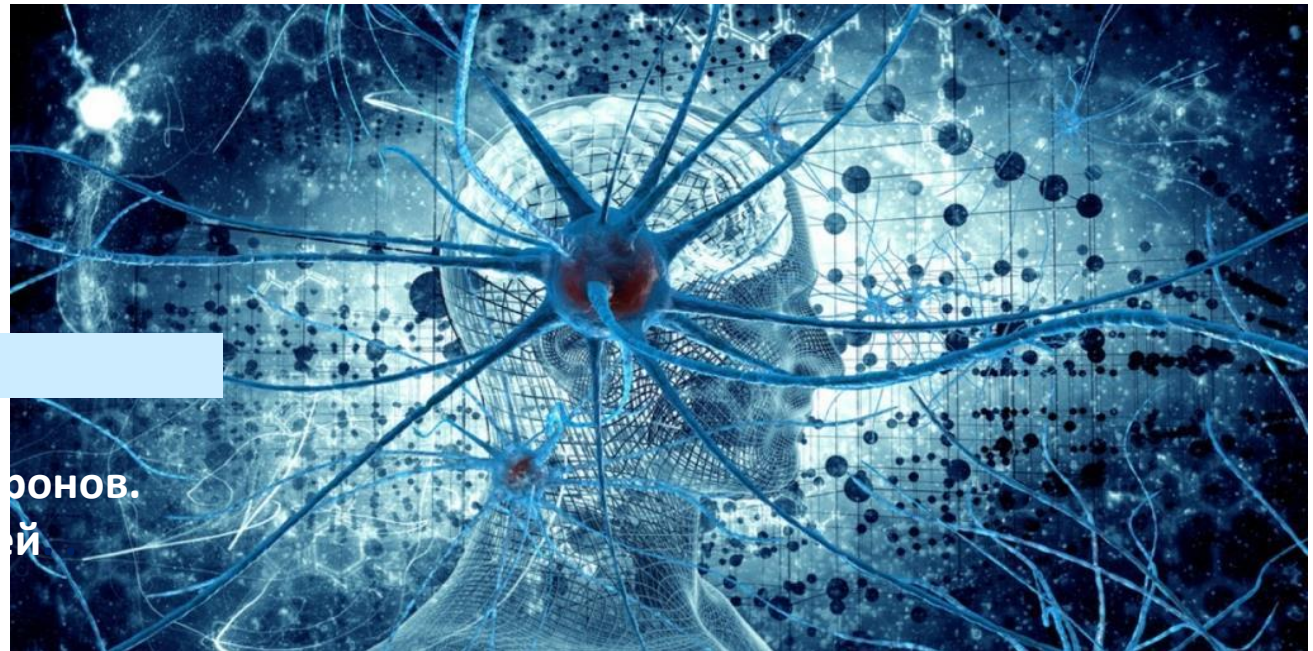
Разработал д.т.н. Тынчеров Камиль Талятович



РАЗДЕЛ 3. НЕЙРОПРОЦЕССОРЫ.

Лекция 7. Нейронные сети

Тема предыдущей лекции: Модели представления знаний (логическая и продукционная модели; модули, управляемые образцами; семантические модели; фреймы; объектно-ориентированный подход; практика использования моделей представления знаний в экспертных системах).



Учебные вопросы лекции

1. Нейрон. Модели нейронов.
2. Типы нейронных сетей

Нейронные сети – это современный тренд, применяемый в науке и технике. С их помощью улучшаются программы и создаются целые системы, способные автоматизировать, ускорять и помогать в работе человеку. Основная урбанистическая цель – научить систему самостоятельно принимать решения в сложных ситуациях так, как это делает человек.

Многое из того, что человечество желает достичь искусственно, уже сделано природой. Человеческий мозг наделен великолепной нейронной сетью, изучение которой ведется и по сей день. Генетически, биологические нейронные сети устроены довольно сложно и человеку без соответствующей технической подготовки будет сложно понять процесс работы естественной нейросети.

В упрощённой трактовке можно сказать так: **биологическая нейронная сеть** это та часть нервной системы, что находится в мозге человека. Совокупность нейронов и сети позволяет нам думать, принимать решения и воспринимать окружающий мир.



Рис.1. Биологическая нейронная сеть

1. Нейрон. Модели нейронов

Биологический нейрон – клетка, основные составляющие которой: ядро, отростки, тела и иные компоненты для тесной связи с тысячами нейронов. По этой связи передаются электрохимические импульсы, приводящие нейронную сеть в состояние спокойствия или возбуждения. Например, перед сдачей экзаменов или другим важным событием, порождается импульс и распространяется по всей нейронной сети в головном мозге, проводя сеть в состояние возбуждения. Затем, по нервной системе передается это состояние другим органам, что приводит к учащению сердцебиения, частому морганию ресниц и прочим проявлениям волнения.

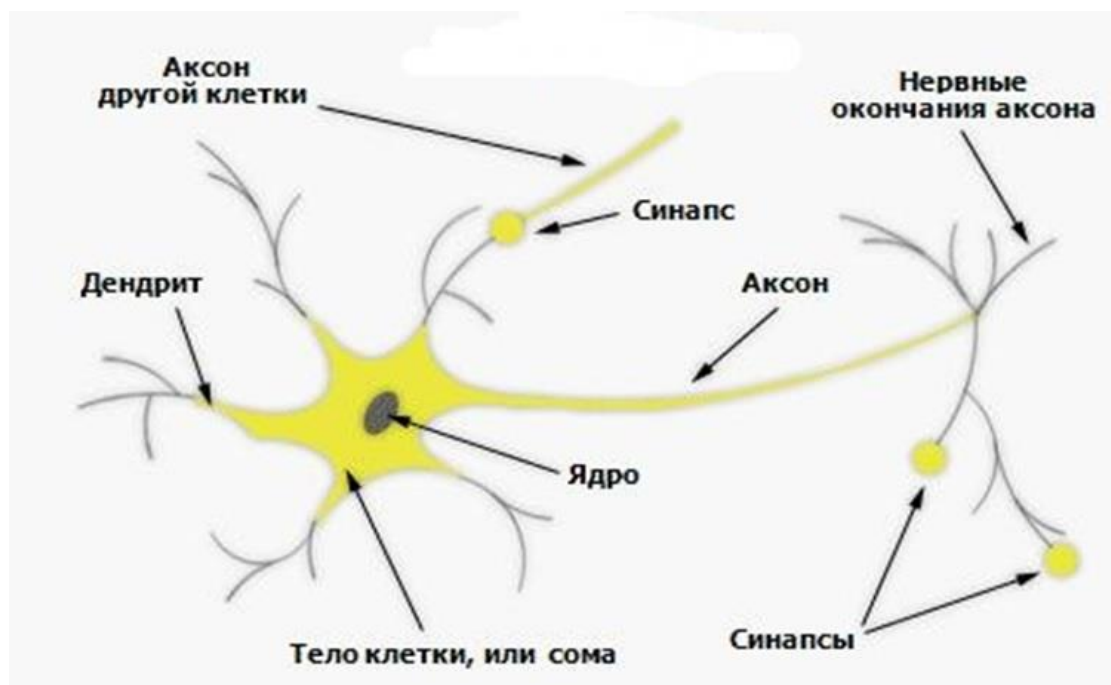


Рис.2. Состав нейрона

1. Нейрон. Модели нейронов

По упрощенной теории естественной нейросети рассмотрим составляющие нейрона.

Состоит он из **тела клетки** и **ядра**.

К телу относится множество ответвлений, называемых **дендритами**.

Длинные дендриты называются **аксонами**. С их помощью проходит связь между нейронами с помощью **синапсов** (места контакта двух нейронов, по которым проходит импульс). В этом случае можно уловить закономерность. На дендрит приходит сигнал (значит это **вход**), по аксону (**выход**) передается на другую нейронную клетку.

В точке синаптической передачи импульса, его частота и амплитуда может изменяться (переменные составляющие сложного уравнения, влияющие на состояние сигнала).

Примерно так работает естественная нейросеть в голове каждого человека.

В таком случае, почему бы не создать искусственную нейросеть наподобие естественной, откинув биологическую составляющую. Дендрит будет выступать в роли входа, аксон – выход, тело нейрона в виде сумматора, а кодом, состоящим из 0 и 1 можно регулировать частоту, импульс сигнала, перед подачей на сумматор в разделе «Веса». Это основная составляющая нейрона.

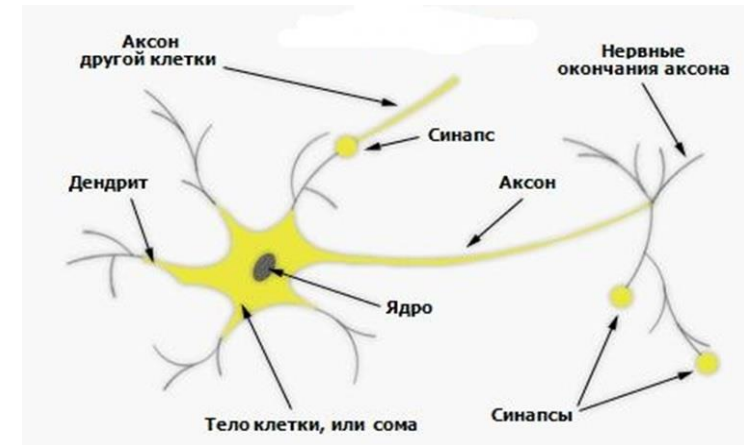


Рис.2. Состав нейрона

Модель нейрона

В целом, графическую модель нейрона можно записать так.

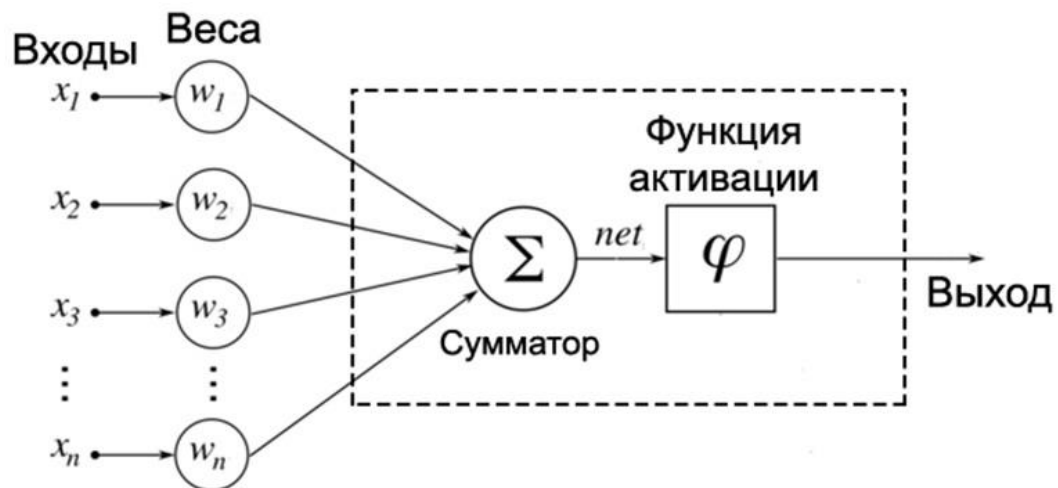


Рис.3. Модель нейрона

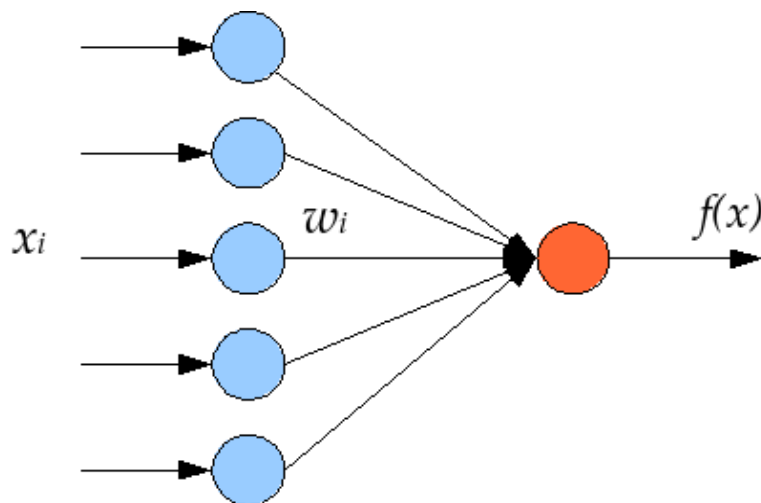
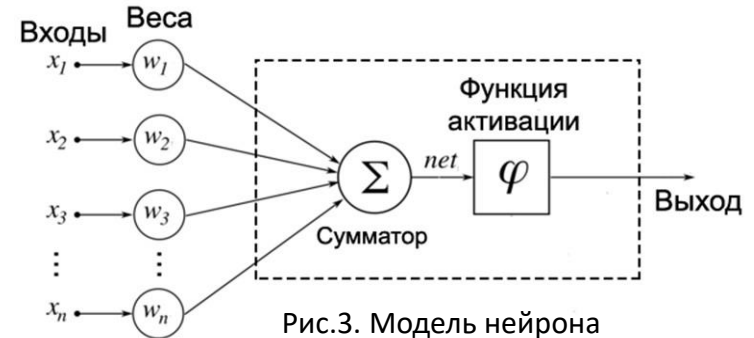


Рис.4. Схема однослойного перцептрона

1. Нейрон. Модели нейронов

Ещё раз о том, что за что отвечает

Давайте снова обратимся к графической модели на рис. 3.



Входы, веса и сумматор

У каждого нейрона, в том числе и у искусственного, должны быть какие-то входы, через которые он принимает сигнал. Мы уже вводили понятие весов, на которые умножаются сигналы, проходящие по связи. У нас веса изображены кружками и обозначены w_i . Поступившие на входы сигналы умножаются на свои веса. Сигнал первого входа x_1 умножается на соответствующий этому входу вес w_1 . В итоге получаем $x_1 w_1$. И так до n -ого входа. В итоге на последнем входе получаем $x_n w_n$.

Теперь все произведения передаются в сумматор Σ . Уже исходя из его названия можно понять, что он делает. Он просто суммирует все входные сигналы, умноженные на соответствующие веса:

$$x_1 w_1 + x_2 w_2 + \dots + x_n w_n = \sum_{i=1}^n x_i w_i$$

Результатом работы сумматора является число, называемое взвешенной суммой.

$$net = \sum_{i=1}^n x_i w_i$$

Роль сумматора очевидна – он агрегирует все входные сигналы (которых может быть много) в какое-то одно число – взвешенную сумму, которая характеризует поступивший на нейрон сигнал в целом. Еще взвешенную сумму можно представить как степень общего возбуждения нейрона.

1. Нейрон. Модели нейронов

Пример

Для понимания роли последнего компонента искусственного нейрона – **функции активации** – приведем следующий пример.

Давайте рассмотрим один искусственный нейрон. Его задача – решить, **ЕХАТЬ ЛИ ОТДЫХАТЬ НА МОРЕ**. Для этого на его входы мы подаем различные данные. Пусть у нашего нейрона будет 4 входа:

1. **Стоимость поездки**
2. **Какая на море погода**
3. **Текущая обстановка с работой**
4. **Будет ли на пляже закусочная**



Все эти параметры будем характеризовать 0 или 1. Соответственно, если погода на море хорошая, то на этот вход подаем 1. И так со всеми остальными параметрами.

Если у нейрона есть четыре входа, то должно быть и четыре весовых коэффициента.

1. Нейрон. Модели нейронов

Пример

В нашем примере весовые коэффициенты можно представить как показатели важности каждого входа, влияющие на общее решение нейрона. Веса входов распределим следующим образом:

- | | | |
|----|------------------------------|---|
| 1. | Стоимость поездки | 5 |
| 2. | Какая на море погода | 4 |
| 3. | Текущая обстановка с работой | 1 |
| 4. | Будет ли на пляже закусочная | 1 |

Нетрудно заметить, что очень большую роль играют факторы стоимости и погоды на море (первые два входа). Они же и будут играть решающую роль при принятии нейроном решения.

Пусть на входы нашего нейрона мы подаем следующие сигналы:

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

Умножаем веса входов на сигналы соответствующих входов:

- | | |
|----|---|
| 1. | 5 |
| 2. | 0 |
| 3. | 0 |
| 4. | 1 |

Взвешенная сумма для такого набора входных сигналов равна 6: $net = \sum_{i=1}^4 x_i w_i = 5 + 0 + 0 + 1 = 6$

1. Нейрон. Модели нейронов

Пример

Все классно, но что делать дальше? Как нейрон должен решить, ехать на море или нет? Очевидно, нам нужно как-то преобразовать нашу взвешенную сумму и получить ответ.

Вот на сцену выходит **функция активации**.

Просто так подавать взвешенную сумму на выход достаточно бессмысленно. Нейрон должен как-то обработать ее и сформировать адекватный выходной сигнал. Именно для этих целей и используют функцию активации. Она преобразует взвешенную сумму в какое-то число, которое и является выходом нейрона (выход нейрона обозначим переменной **out**).

Для разных типов искусственных нейронов используют самые разные функции активации. В общем случае их обозначают символом **$\varphi(\text{net})$** . Указание взвешенного сигнала в скобках означает, что функция активации принимает взвешенную сумму как параметр. Значение этой функции и является выходом нейрона (**out**):

$$\text{out} = \varphi(\text{net})$$

1. Нейрон. Модели нейронов

Пример

Самый простой вид функции активации. Выход нейрона может быть равен только 0 или 1. Если взвешенная сумма больше определенного порога b , то выход нейрона равен 1. Если ниже, то 0.

Как ее можно использовать? Предположим, что мы поедем на море только тогда, когда взвешенная сумма больше или равна 5. Значит наш порог равен 5:

$b=5$

В нашем примере взвешенная сумма равнялась 6, а значит выходной сигнал нашего нейрона равен 1. Итак, мы едем на море.

Однако если бы погода на море была бы плохой, а также поездка была бы очень дорогой, но имелась бы закусочная и обстановка с работой нормальная (входы: 0011), то взвешенная сумма равнялась бы 2, а значит выход нейрона равнялся бы 0. Итак, мы никуда не едем.

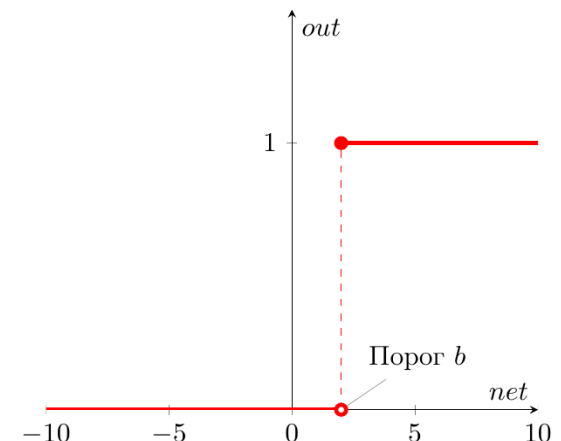
В общем, нейрон смотрит на взвешенную сумму и если она получается больше его порога, то нейрон выдает выходной сигнал, равный 1.

Графически эту функцию активации можно изобразить следующим образом.

На горизонтальной оси расположены величины взвешенной суммы. На вертикальной оси — значения выходного сигнала. Как легко видеть, возможны только два значения выходного сигнала: 0 или 1. Причем 0 будет выдаваться всегда от минус бесконечности и вплоть до некоторого значения взвешенной суммы, называемого порогом. Если взвешенная сумма равна порогу или больше него, то функция выдает 1. Все предельно просто. Теперь запишем эту функцию активации математически. Почти наверняка вы сталкивались с таким понятием, как составная функция. Это когда мы под одной функцией объединяем несколько правил, по которым рассчитывается ее значение. В виде составной функции функция единичного скачка будет выглядеть следующим образом:

$$out(net) = \begin{cases} 0, & net < b \\ 1, & net \geq b \end{cases}$$

Функция единичного скачка



1. Нейрон. Модели нейронов

Выводы:

1. Теперь вы получили полное представление о внутренней структуре искусственного нейрона.
2. У нейрона есть входы. На них подаются сигналы в виде чисел. Каждый вход имеет свой вес (тоже число). Сигналы на входе умножаются на соответствующие веса. Получаем набор «взвешенных» входных сигналов.
3. Далее этот набор попадает в сумматор, который просто складывает все входные сигналы, помноженные на веса. Получившееся число называют взвешенной суммой.
4. Затем взвешенная сумма преобразуется функцией активации и мы получаем выход нейрона.

2. Типы нейронных сетей.

Определение искусственной нейронной сети

Изучив основные аспекты, можно дать определение **искусственной нейронной сети** – это математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма.

Нарисовать и представить алгоритм работы можно с помощью абстрактной схемы. Есть несколько методов представления схем нейронных сетей, но для наглядности проще будет использовать кружки со стрелками.

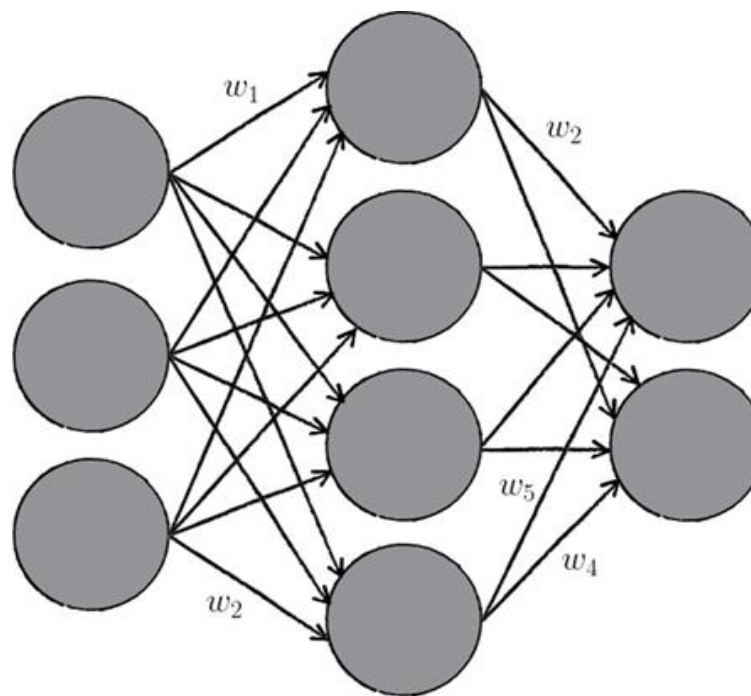


Рис.5. Модель естественной нейросети

2. Типы нейронных сетей.

За период развития, нейронные сети поделились на множество типов, которые переплетаются между собой в различных задачах. На данный момент сложно классифицировать какую-либо сеть только по одному признаку. Это можно сделать по принципу применения, типу входной информации, характеру обучения, характеру связей, сфере применения.

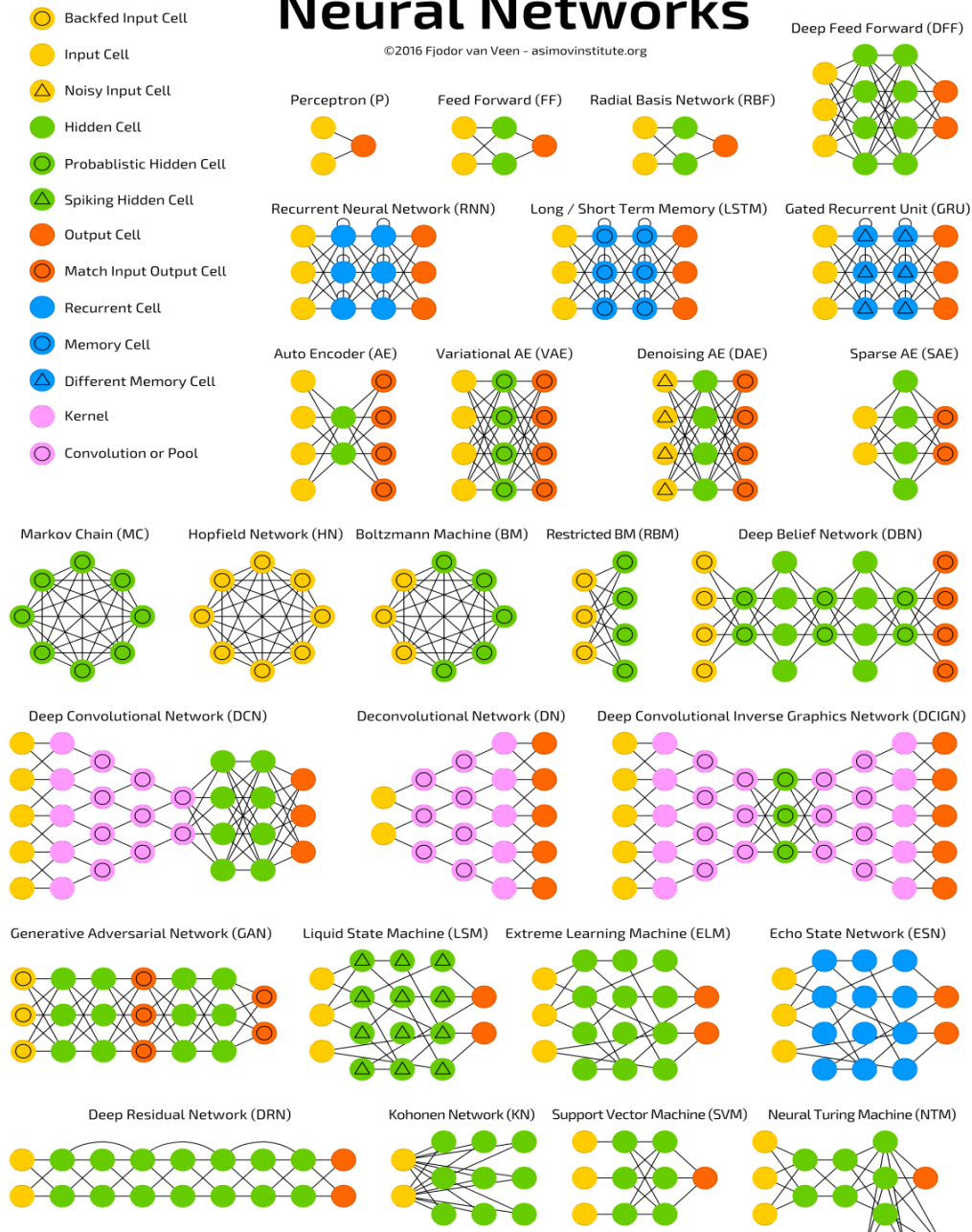
Таблица 1. Основные типы нейронных сетей

Тип нейронной сети	Назначение	Обучение с учителем или без или смешанное	Сфера применения
Перцептрон Розенблатта	Распознавание образов, принятие решений, прогнозирование, аппроксимация, анализ данных	Обучение с учителем	Практически любая сфера применения, кроме оптимизации информации
Хопфилда	Сжатие данных и ассоциативная память	Обучение без учителя	Строение компьютерных систем
Кохонена	Кластеризация, сжатие данных, анализ данных, оптимизация	Обучение без учителя	Финансы, базы данных
Радиально-базисных функций (RBF-сеть)	Принятие решений и управление, аппроксимация, прогнозирование	Смешанное обучение	Управленческие структуры, нейроуправление
Свёрточная CNN	Распознавание образов	Обучение с учителем	Обработка графических данных
Импульсная SNN	Принятие решение, распознавание образов, анализ данных	Обучение с учителем и без учителя	Протезирование, робототехника, телекоммуникации, компьютерное зрение

Есть и другие типы нейронных сетей, например **Цепи Маркова** (Markov chains, MC или discrete time Markov Chains, DTMC) — это предшественники машин Больцмана (BM) и сетей Хопфилда (HN), **Машина Больцмана** (Boltzmann machine, BM) очень похожа на сеть Хопфилда, но в ней некоторые нейроны помечены как входные, а некоторые — как скрытые, **Автокодировщик** (autoencoder, AE) и многие др. Разнообразие ИНС можно посмотреть на следующем слайде.

Neural Networks

©2016 Fjodor van Veen - asimovinstitute.org



Картинку скачал отсюда:
<https://tproger.ru/translations/neural-network-zoo-1/>

2. Типы нейронных сетей.

Перцептрон Розенблатта

Перцептрон — математическая или компьютерная модель восприятия информации мозгом (кибернетическая модель мозга), предложенная Фрэнком Розенблаттом в 1957 году и впервые реализованная в виде электронной машины «Марк-1» в 1960 году. Перцептрон стал одной из первых моделей нейросетей, а «Марк-1» — первым в мире нейрокомпьютером.

Перцептрон состоит из трёх типов элементов, а именно: поступающие от датчиков сигналы передаются ассоциативным элементам, а затем реагирующим элементам. Таким образом, перцептроны позволяют создать набор «ассоциаций» между входными стимулами и необходимой реакцией на выходе. В биологическом плане это соответствует преобразованию, например, зрительной информации в физиологический ответ от двигательных нейронов. Согласно современной терминологии, перцептроны могут быть классифицированы как искусственные нейронные сети:

- с одним скрытым слоем;
- с пороговой передаточной функцией;
- с прямым распространением сигнала.

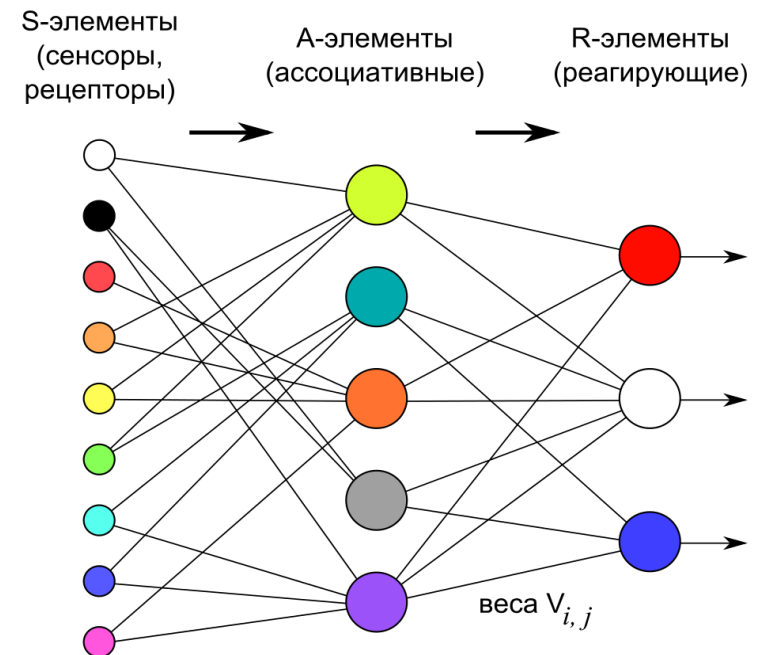


Рис.6. Логическая схема перцептрона с тремя выходами

Перцептрон Розенблатта

Элементарный перцептрон состоит из элементов трёх типов: *S*-элементов, *A*-элементов и одного *R*-элемента. ***S-элементы*** — это слой сенсоров или рецепторов. В физическом воплощении они соответствуют, например, светочувствительным клеткам сетчатки глаза или фоторезисторам матрицы камеры. Каждый рецептор может находиться в одном из двух состояний — покоя или возбуждения, и только в последнем случае он передаёт единичный сигнал в следующий слой, ассоциативным элементам. *A*-элемент активизируется, как только количество сигналов от *S*-элементов на его входе превысило некоторую величину ϑ (*порог*). Таким образом, если набор соответствующих *S*-элементов располагается на сенсорном поле в форме буквы «Д», *A*-элемент активизируется, если достаточное количество рецепторов сообщило о появлении «белого пятна света» в их окрестности, то есть *A*-элемент будет как бы ассоциирован с наличием/отсутствием буквы «Д» в некоторой области.

Сигналы от возбуждшихся *A*-элементов, в свою очередь, передаются в **сумматор *R***, причём сигнал от *i*-го ассоциативного элемента передаётся с коэффициентом w_i . Этот коэффициент называется **весом *A—R* связи**. Так же как и *A*-элементы, *R*-элемент подсчитывает сумму значений входных сигналов, помноженных на веса (линейную форму). *R*-элемент, а вместе с ним и элементарный перцептрон, выдаёт «1», если линейная форма превышает порог ϑ , иначе на выходе будет «-1».

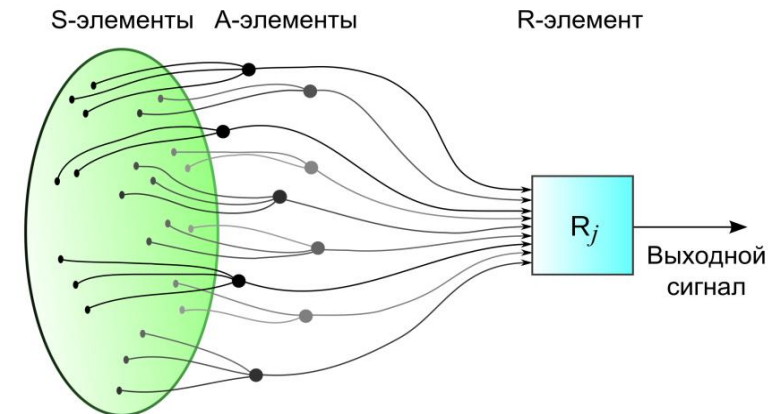


Рис.7. Поступление сигналов с сенсорного поля в решающие блоки элементарного перцептрона в его физическом воплощении.

2. Типы нейронных сетей.

Перцептрон Розенблатта

Математически, функцию, реализуемую R-элементом, можно записать так:

$$f(x) = \text{sign}(\sum_{i=1}^n w_i x_i - \vartheta)$$

Обучение элементарного перцептрона состоит в изменении весовых коэффициентов w_i связей A—R. Веса связей S—A (которые могут принимать значения $\{-1; 0; +1\}$) и значения порогов A-элементов выбираются случайным образом в самом начале и затем не изменяются.

После обучения перцептрон готов работать в режиме распознавания или обобщения [11]. В этом режиме перцептрону предъявляются ранее неизвестные ему объекты, и перцептрон должен установить, к какому классу они принадлежат. Работа перцептрона состоит в следующем: при предъявлении объекта возбуждившиеся A-элементы передают сигнал R-элементу, равный сумме соответствующих коэффициентов w_i . Если эта сумма положительна, то принимается решение, что данный объект принадлежит к первому классу, а если она отрицательна — то ко второму.

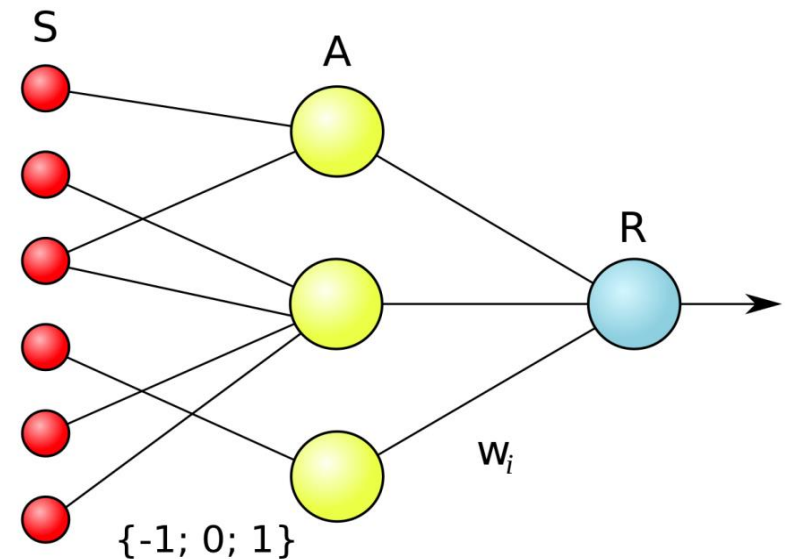


Рис.8. Логическая схема элементарного перцептрона. Веса S—A связей могут иметь значения -1 , $+1$ или 0 (то есть отсутствие связи). Веса A—R связей W могут быть

2. Типы нейронных сетей.

Перцептрон Розенблатта

Определим составные элементы перцептрона, которые являются частными случаями искусственного нейрона с **пороговой передаточной функцией**.

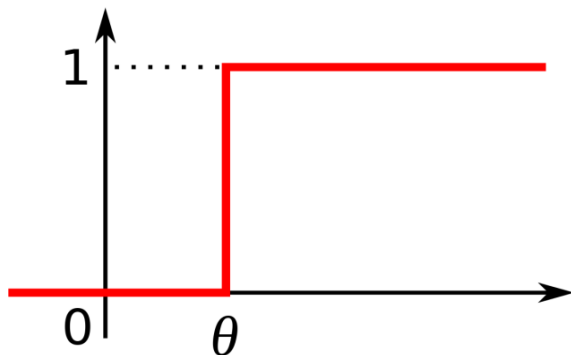


Рис.9. Пороговая функция, реализуемая простыми S- и A-элементами.

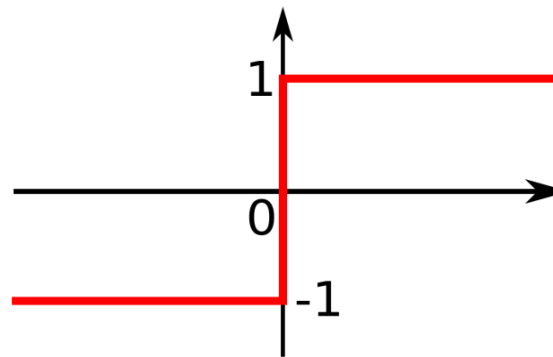


Рис.10 Пороговая функция, реализуемая простым R-элементом.

- Простым **S-элементом** (сенсорным) является чувствительный элемент, который от воздействия какого-либо из видов энергии (например, света, звука, давления, тепла и т. п.) вырабатывает сигнал. Если входной сигнал превышает некоторый порог ϑ , на выходе элемента получаем +1, в противном случае — 0.
- Простым **A-элементом** (ассоциативным) называется логический решающий элемент, который даёт выходной сигнал +1, когда алгебраическая сумма его входных сигналов превышает некоторую пороговую величину ϑ (говорят, что элемент активный), в противном случае выход равен нулю [13].
- Простым **R-элементом** (реагирующим, то есть действующим) называется элемент, который выдаёт сигнал +1, если сумма его входных сигналов является строго положительной, и сигнал -1, если сумма его входных сигналов является строго отрицательной. Если сумма входных сигналов равна нулю, выход считается либо равным нулю, либо неопределённым [13].

Если на выходе любого элемента мы получаем 1, то говорят, что элемент **активен** или **возбуждён**.

2. Типы нейронных сетей.

Перцептрон Розенблатта

Процесс обучения является процедурой настройки весов и порогов с целью уменьшения разности между желаемыми (целевыми) и получаемыми векторами на выходе. Розенблатт пытался классифицировать различные алгоритмы обучения перцептрона, называя их системами подкрепления.

Система подкрепления — это любой набор правил, на основании которых можно изменять с течением времени матрицу взаимодействия (или состояние памяти) перцептрона. Описывая эти системы подкрепления и уточняя возможные их виды, Розенблатт основывался на идеях Д. Хебба об обучении, предложенных им в 1949 году, которые можно перефразировать в следующее **правило**, состоящее из двух частей:

- Если два нейрона по обе стороны синапса (соединения) активизируются одновременно (то есть синхронно), то прочность этого соединения возрастает.
- Если два нейрона по обе стороны синапса активизируются асинхронно, то такой синапс ослабляется или вообще отмирает.

Перцептрон Розенблатта

Классический метод обучения перцептрона *с учителем* — это *метод коррекции ошибки* [8]. Он представляет собой такой вид обучения с учителем, при котором вес связи не изменяется до тех пор, пока текущая реакция перцептрона остаётся правильной. При появлении неправильной реакции вес изменяется на единицу, а знак (+/-) определяется противоположным от знака ошибки.

Допустим, мы хотим обучить перцептрон разделять два класса объектов так, чтобы при предъявлении объектов первого класса выход перцептрона был положителен (+1), а при предъявлении объектов второго класса — отрицательным (−1). Для этого выполним следующий *алгоритм обучения*: [5]

1. Случайным образом выбираем пороги для А-элементов и устанавливаем связи S—А (далее они изменяться не будут).
2. Начальные коэффициенты w_i полагаем равными нулю.
3. Предъявляем обучающую выборку: объекты (например, круги либо квадраты) с указанием класса, к которым они принадлежат.
4. Показываем перцептрону объект первого класса. При этом некоторые А-элементы возбуждаются. Коэффициенты w_i , соответствующие этим возбуждённым элементам, увеличиваем на 1.
5. Предъявляем объект второго класса и коэффициенты w_i тех А-элементов, которые возбуждаются при этом показе, уменьшаем на 1.
6. Обе части шага 3 выполним для всей обучающей выборки. В результате обучения сформируются значения весов связей w_i .

Перцептрон Розенблатта

Кроме классического метода обучения перцептрона Розенблатт также ввёл понятие об **обучении без учителя**, предложив следующий способ обучения:

Альфа-система подкрепления — это система подкрепления, при которой веса всех активных связей c_{ij} , которые ведут к элементу u_j , изменяются на одинаковую величину r , а веса неактивных связей за это время не изменяются[23].

Затем, с разработкой понятия многослойного перцептрона, альфа-система была модифицирована и её стали называть **дельта-правило**. Модификация была проведена с целью сделать функцию обучения дифференцируемой (например, сигмоидной), что в свою очередь нужно для применения метода градиентного спуска, благодаря которому возможно обучение более одного слоя.

Метод обратного распространения ошибки

Для обучения многослойных сетей рядом учёных, в том числе Д. Румельхартом, был предложен градиентный алгоритм обучения с учителем, проводящий сигнал ошибки, вычисленный выходами перцептрона, к его входам, слой за слоем. Сейчас это самый популярный метод обучения многослойных перцептронов. Его преимущество в том, что он может обучить все слои нейронной сети, и его легко просчитать локально. Однако этот метод является очень долгим, к тому же, для его применения нужно, чтобы передаточная функция нейронов была дифференцируемой. При этом в перцептронах пришлось отказаться от бинарного сигнала, и пользоваться на входе непрерывными значениями[24]

Нейронная сеть Хопфилда

Нейронная сеть Хопфилда — полносвязная нейронная сеть с симметричной матрицей связей. В процессе работы динамика таких сетей сходится (конвергирует) к одному из положений равновесия. Эти положения равновесия определяются заранее в процессе обучения, они являются **локальными минимумами функционала**, называемого энергией сети (в простейшем случае — локальными минимумами отрицательно определённой квадратичной формы на n -мерном кубе).

Такая сеть может быть использована как автоассоциативная память, как фильтр, а также для решения некоторых задач оптимизации. В отличие от многих нейронных сетей, работающих до получения ответа через определённое количество тактов, сети Хопфилда работают до достижения равновесия, когда следующее состояние сети в точности равно предыдущему: начальное состояние является входным образом, а при равновесии получают выходной образ. Ее вариацией является Нейронная сеть Хэмминга.



Джон Джозеф Хопфилд, 1933,
Калифорнийский
технологический институт

Нейронная сеть Хопфилда

Нейронная сеть Хопфилда устроена так, что её отклик на запомненные m эталонных «образов» составляют сами эти образы, а если образ немного исказить и подать на вход, он будет восстановлен и в виде отклика будет получен оригинальный образ. Таким образом, сеть Хопфилда осуществляет коррекцию ошибок и помех. Она однослойная и состоит из N искусственных нейронов. Каждый нейрон системы может принимать на входе и на выходе одно из двух состояний (что аналогично выходу нейрона с пороговой функцией активации):

$$y_i = \begin{cases} 1 \\ -1 \end{cases}$$

Из-за их биполярной природы нейронные сети Хопфилда иногда называют **спинами**.

Каждый нейрон связан со всеми остальными нейронами.

Взаимодействие нейронов сети описывается выражением:

$$E = \frac{1}{2} \sum_{i,j=1}^N w_{ij} x_i x_j$$

где w_{ij} — элемент матрицы взаимодействий $\mathbf{W}$, которая состоит из весовых коэффициентов связей между нейронами. В процессе обучения формируется выходная матрица $\mathbf{W}$, которая m эталонных «образов» — N -мерных бинарных векторов: $\mathbf{s}_m = (s_{m1}, s_{m2}, \dots, s_{mN})$, эти образы во время эксплуатации сети будут выражать отклик системы на входные сигналы, или иначе — окончательные значения выходов y_i после серии итераций.

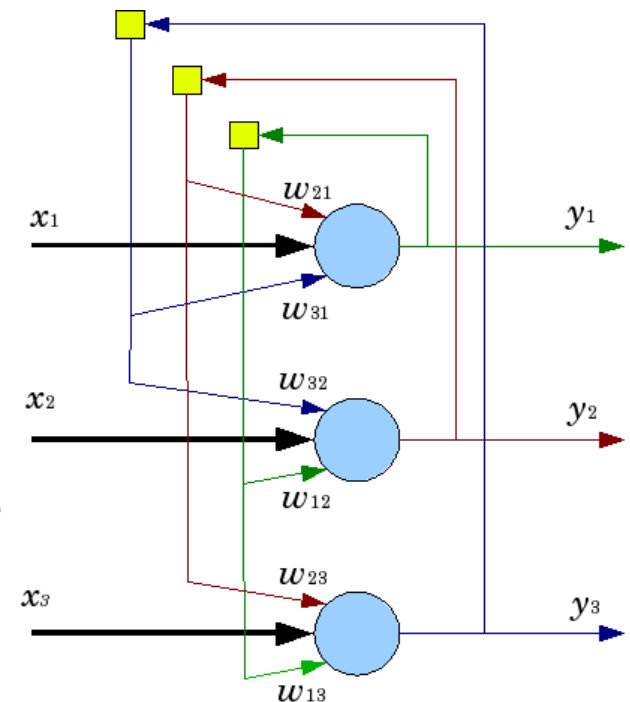


Рис.11. Схема сети Хопфилда с тремя нейронами

Нейронная сеть Хопфилда

В сети Хопфилда матрица связей является симметричной $w_{ij}=w_{ji}$, а диагональные элементы матрицы полагаются равными нулю $w_{ii}=0$, что исключает эффект воздействия нейрона на самого себя и является необходимым для сети Хопфилда, но не достаточным условием устойчивости в процессе работы сети. Достаточным является асинхронный режим работы сети. Подобные свойства определяют тесную связь с реальными физическими веществами, называемыми спиновыми стёклами.

Матрица взаимодействий хранится на самих нейронах в виде весов при связях нейронов с другими нейронами.

Так например, если входной сигнал определяется 10 параметрами, то нейронная сеть Хопфилда формируется из одного уровня с 10 нейронами. Каждый нейрон связывается со всеми остальными 9-ю нейронами, таким образом в сети образуется 90 (10 x 9) связей. Для каждой связи определяется весовой коэффициент w_{ij} . Все веса связей и образуют матрицу взаимодействий, которая заполняется в процессе обучения.

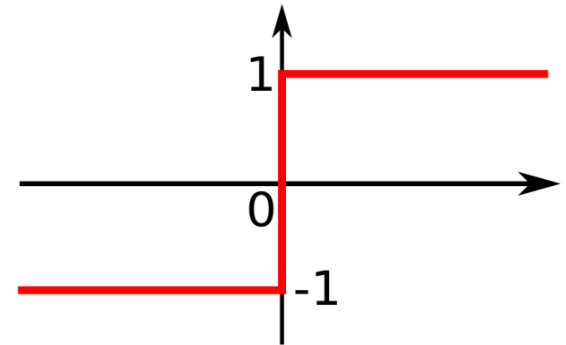


Рис.12. Пороговая функция F, реализуемая нейроном сети Хопфилда.

Нейронная сеть Хопфилда

Обучение сети заключается в том, что находятся веса матрицы взаимодействий так, чтобы запомнить m векторов (эталонных образов, составляющих "память" системы).

Вычисление коэффициентов основано на следующем правиле: для всех запомненных образов X_i матрица связи должна удовлетворять уравнению

$$X_i = WX_i$$

поскольку именно при этом условии состояния сети X_i будут устойчивы — попав в такое состояние, сеть в нём и останется.

Запоминаемые векторы должны иметь бинарный вид. Расчёт весовых коэффициентов проводится по следующей формуле:

$$w_{ij} = \frac{1}{N} \sum_{d=1...m} X_{id}X_{jd},$$

где N — размерность векторов, m — число запоминаемых выходных векторов, d — номер запоминаемого выходного вектора, X_{ij} — i -я компонента запоминаемого выходного j -го вектора. Это выражение может стать более ясным, если заметить, что весовая W может быть найдена вычислением внешнего произведения каждого запоминаемого вектора с самим собой и суммированием матриц, полученных таким образом. Это может быть записано в виде

$$W = \frac{1}{N} \sum_i X_i X_i^T,$$

где X_i — i -й запоминаемый вектор-столбец. Расчёт этих весовых коэффициентов и называется обучением сети, которое проводится только за одну эпоху.

2. Типы нейронных сетей.

Нейронная сеть Кохонена

Нейронные сети Кохонена — класс нейронных сетей, основным элементом которых является слой Кохонена, состоящий из адаптивных линейных сумматоров («линейных формальных нейронов»). Как правило, выходные сигналы слоя Кохонена обрабатываются по правилу «Победитель получает всё»: наибольший сигнал превращается в единичный, остальные обращаются в ноль.

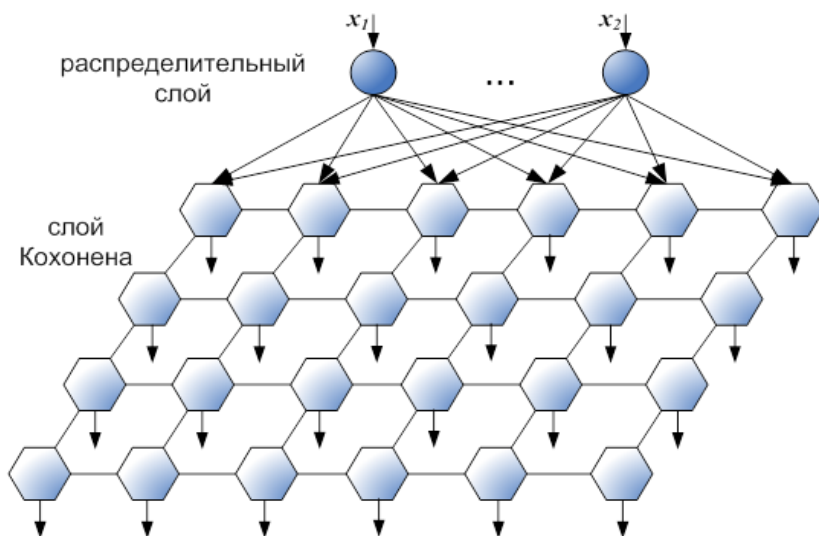


Рис.13. «Нейроны Кохонена как набор электрических лампочек, когда для любого входного вектора загорается одна из них»



Теуво Калеви Кохонен, 1934 г.р.,
Технологический университет,
Хельсинки

2. Типы нейронных сетей.

Нейронная сеть Кохонена

Базовая версия

Слой Кохонена состоит из некоторого количества n параллельно действующих линейных элементов. Все они имеют одинаковое число входов m и получают на свои входы один и тот же вектор входных сигналов $x = x_1, \dots, x_m$. На выходе j -го линейного элемента получаем сигнал

$$y_j = w_{j0} + \sum_{i=1}^m w_{ji}x_i,$$

где: w_{ji} — весовой i -го входа j -го нейрона;

i — номер входа;

j — номер нейрона;

w_{j0} — пороговый коэффициент.

После прохождения слоя линейных элементов сигналы посылаются на обработку по правилу «победитель забирает всё»: среди выходных сигналов выполняется поиск максимального y_j ; "его номер" $j_{max} = \arg \max_j \{y_j\}$. Окончательно, на выходе сигнал с номером j_{max} равен единице, остальные — нулю. Если максимум одновременно достигается для нескольких j_{max} , то:

- либо принимают все соответствующие сигналы равными единице;
- либо равным единице принимают только первый сигнал в списке (по соглашению).

2. Типы нейронных сетей.

Сеть радиально-базисных функций (RBF)

Сеть радиально-базисных функций — искусственная нейронная сеть, которая использует радиальные базисные функции как функции активации. Выходом сети является линейная комбинация радиальных базисных функций входов и параметров нейрона. Сети радиальных базисных функций имеют множество применений, в том числе функции приближения, прогнозирования временных рядов, классификации и системы управления. Впервые сформулированы в 1988 Брумхедом и Лоу.

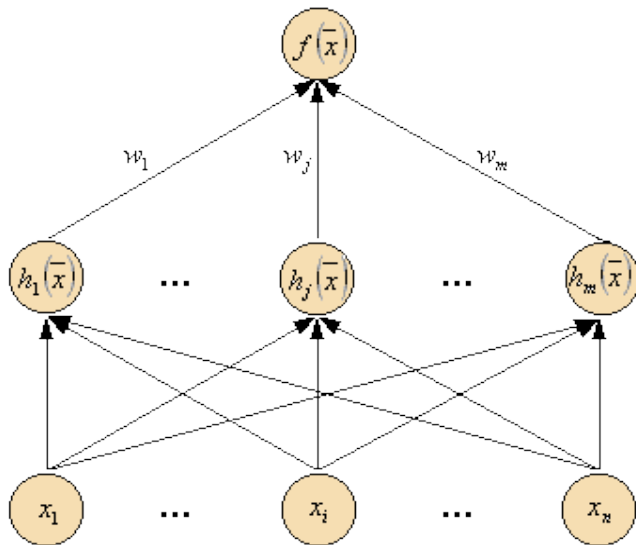


Рис.14. Радиально-симметричные функции (например, Гаусса $h(x) = \exp(-\frac{(x-c)^2}{r^2})$) в однослойной сети

Радиально-симметричные функции — простейший класс функций. В принципе, они могут быть использованы в разных моделях (линейных и нелинейных) и в разных сетях (многослойных и однослойных). Традиционно термин RBF сети ассоциируется с радиально-симметричными функциями в однослойных сетях, имеющих структуру, представленную на рисунке. То есть, каждый из n компонентов входного вектора ($\bar{x}$) подается на вход m базисных функций и их выходы линейно суммируются с весами: $\{w_j\}_{j=1}^m$. Таким образом, выход RBF сети является линейной комбинацией некоторого набора базисных функций:

$$f(\bar{x}) = \sum_{j=1}^m w_j h_j(\bar{x})$$

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Свёрточная нейронная сеть (CNN) — специальная архитектура искусственных нейронных сетей, предложенная Яном Лекунем в 1988 году и нацеленная на эффективное распознавание образов, входит в состав технологий глубокого обучения. Использует некоторые особенности зрительной коры, в которой были открыты так называемые простые клетки, реагирующие на прямые линии под разными углами, и сложные клетки, реакция которых связана с активацией определённого набора простых клеток. Таким образом, идея свёрточных нейронных сетей заключается в чередовании слоёв.

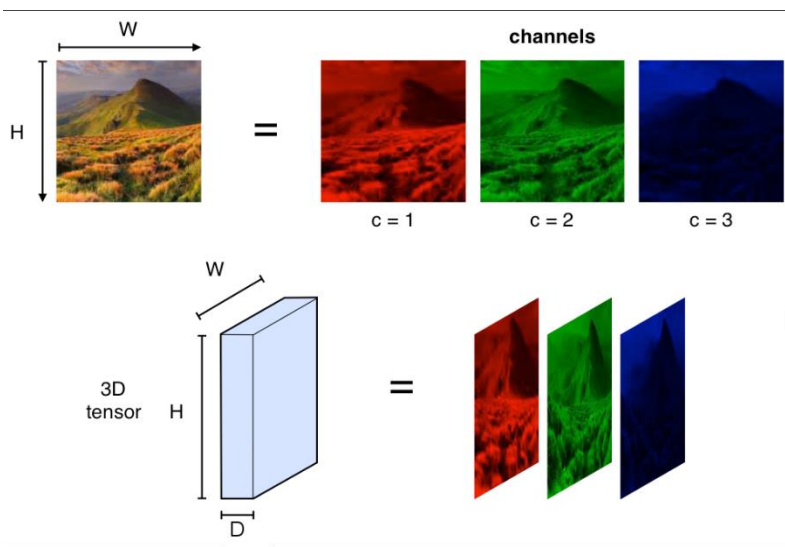
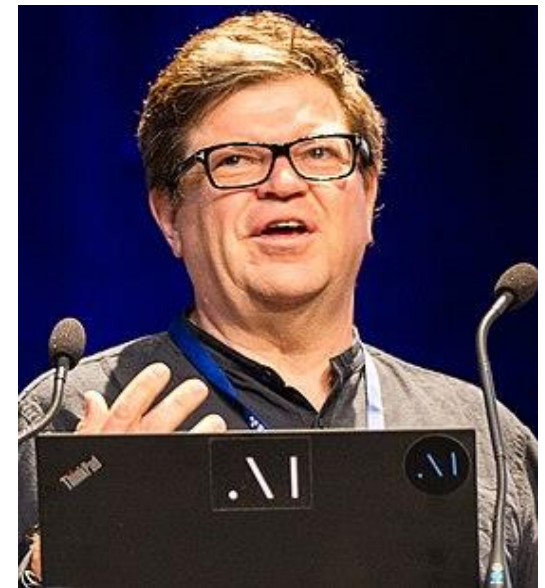


Рис.15. Свёрточная нейронная сеть



Ян Лекун, 1960, Франция

В отличие от сетей прямого распространения, которые работают с данными в виде векторов, свёрточные сети работают с изображениями в виде тензоров. **Тензоры** — это 3D массивы чисел, или, проще говоря, массивы матриц чисел.

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Изображения в компьютере представляются в виде пикселей, а каждый пиксель – это значения интенсивности соответствующих каналов. При этом интенсивность каждого из каналов описывается целым числом от 0 до 255. Чаще всего используются цветные изображения, которые состоят из RGB пикселей – пикселей, содержащих яркости по трём каналам: красному, зелёному и синему. Различные комбинации этих цветов позволяют создать любой из цветов всего спектра. Именно поэтому вполне логично использовать именно тензоры для представления изображений: каждая матрица тензора отвечает за интенсивность своего канала, а совокупность всех матриц описывает всё изображение.

Главной особенностью свёрточных сетей является то, что они работают именно с изображениями, а потому можно выделить особенности, свойственные именно им. Многослойные персептроны работают с векторами, а потому для них нет никакой разницы, находятся ли какие-то точки рядом или на противоположных концах, так как все точки равнозначны и считаются совершенно одинаковым образом. Изображения же обладают локальной связностью. Например, если речь идёт об изображениях человеческих лиц, то вполне логично ожидать, что точки основных частей лица будут рядом, а не разрозненно располагаться на изображении. Поэтому требовалось найти более эффективные алгоритмы для работы с изображениями и ими оказались свёрточные сети.

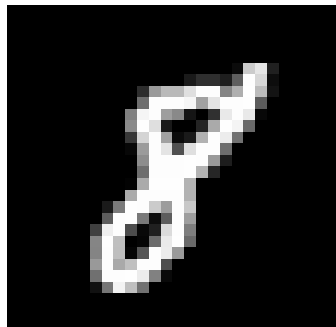


Рис.16. Изображение цифры из датасета MNIST

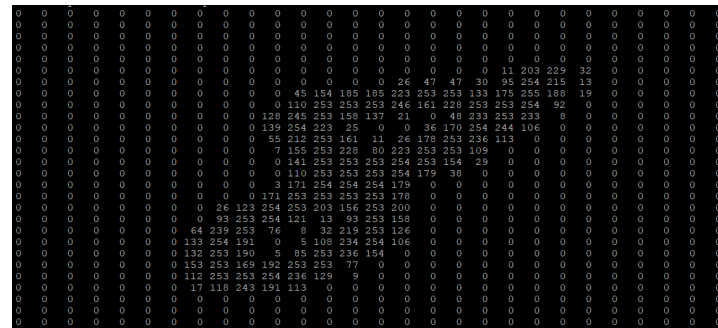


Рис.17. Эта же цифра в

компьютере

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Свёрточные нейронные сети состоят из базовых блоков, благодаря чему их можно собирать как конструктор, добавляя слой за слоем и получая всё более мощные архитектуры. Основными блоками свёрточных нейронных сетей являются **свёрточные слои**, **слои подвыборки** (пулинга), **слои активации** и **полносвязные слои**.

Так, например, LeNet5 – одна из первых свёрточных сетей, которая победила в ImageNet, состояла из 7 слоёв: слой свёртки, слой пулинга, ещё один слой свёртки ещё один слой пулинга и трёхслойная полносвязная нейронная сеть. Рассмотрим более подробно перечисленные блоки.

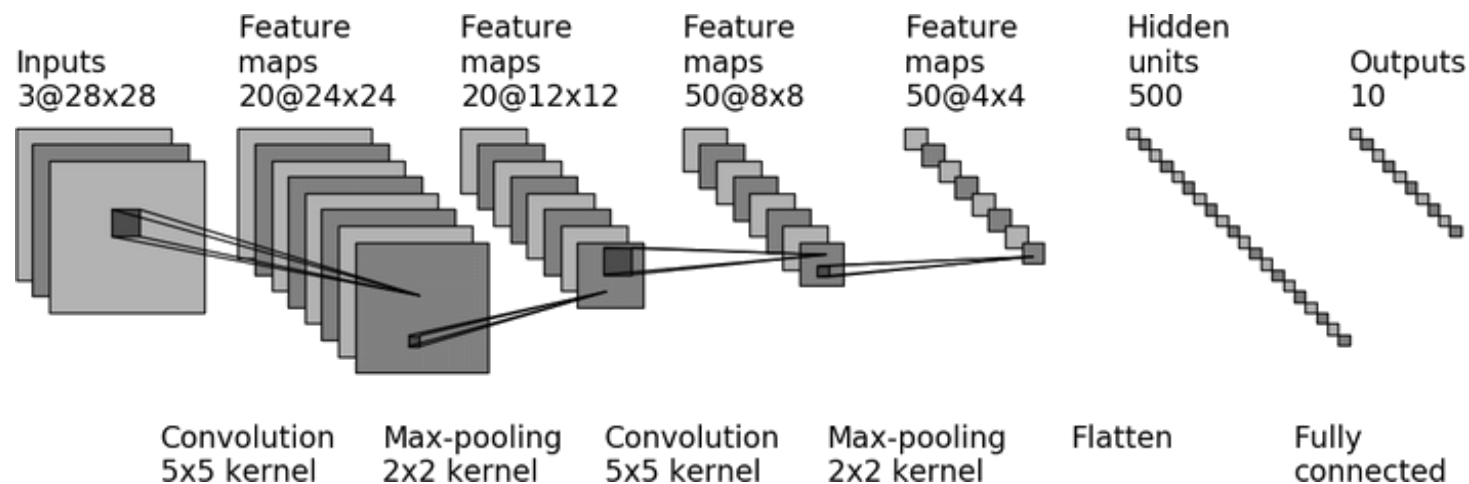


Рис.18. Свёрточная сеть LeNet5

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Слой свёртки, как можно догадаться по названию типа нейронной сети, является самым главным слоем сети. Его основное назначение – выделить признаки на входном изображении и сформировать карту признаков. Карта признаков – это всего лишь очередной тензор (массив матриц), в котором каждый канал отвечает за какой-нибудь выделенный признак.

Для того, чтобы слой мог выделять признаки, в нём имеются так называемые фильтры (или ядра). Ядра — это всего лишь набор тензоров. Эти тензоры имеют один и тот же размер, а их количество определяет глубину выходного 3D массива. При этом глубина самих фильтров совпадает с количеством каналов входного изображения. Так, если на вход свёрточному слою подаётся RGB изображение и требуется карта признаков, состоящая из 32 каналов, то свёрточный слой будет содержать в себе 32 фильтра глубиной 3.

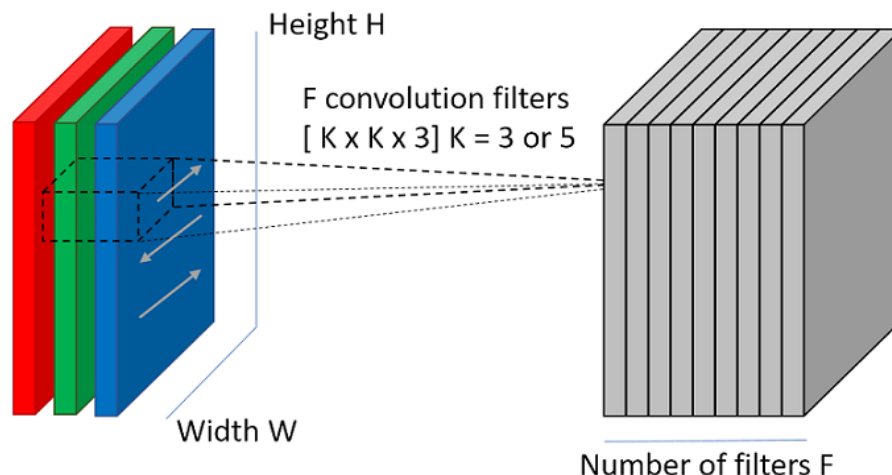


Рис.19. Параметры свёртки в свёрточной нейронной сети

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Для того, чтобы сформировать карту признаков из входного изображения, производится операция свёртки входного тензора с каждым из фильтров. **Свёртка** – это операция вычисления нового значения выбранного пикселя, учитывающая значения окружающих его пикселей. Алгоритм получения результата свёртки можно описать так:

Фильтр накладывается на левую верхнюю часть изображения и производится покомпонентное умножение значений фильтра и значений изображения, после чего фильтр перемещается дальше по изображению до тех пор, пока аналогичным образом не будут обработаны все его участки.

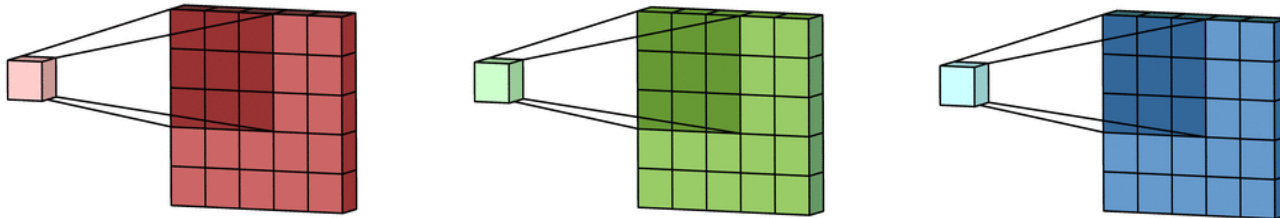
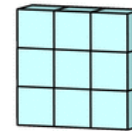
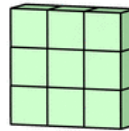
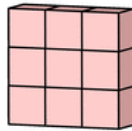


Рис.20. Покомпонентное умножение значений фильтра и значений изображения в сверточной нейронной сети

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Затем числа полученных матриц суммируются в единую матрицу — результат применения фильтра.



2. Типы нейронных сетей.

Свёрточная нейронная сеть

После этого к каждому значению матрицы добавляется одинаковое число – значение смещения данного фильтра. Полученная матрица составляет один канал выходной карты признаков.

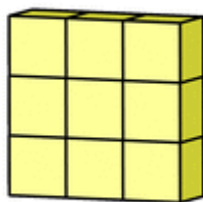


Рис.22. Получение изображения в свёрточной нейронной сети

После того, как будут получены каналы для каждого из фильтров, матрицы объединяются в единый тензор, благодаря чему на выходе снова получается изображение, с другим числом каналов и, возможно, другим размером.

Свёрточная нейронная сеть

Параметры свёрточного слоя

- **Число признаков (filters count, fc)** – это количество фильтров, которые есть в слое.
- **Размер фильтров (filter size, fs)** – это высота и ширина тензора фильтров. Обычно является нечётным числом, наиболее часто используются фильтры размером 3 или 5.
- **Шаг свёртки (stride, S)** – это количество пикселей, на которое перемещается матрица фильтра по входному изображению. Когда шаг равен 1, фильтры перемещаются по одному пикселю за раз. Когда шаг равен 2, тогда фильтры перескакивают на 2 пикселя за раз. Чем больше шаг, тем меньшего размера карты признаков получаются на выходе.
- **Дополнения нулями (padding, P)** – количество пикселей, которые добавляются с каждого края изображения. Это позволяет избежать уменьшения изображения на размер фильтра, поскольку фильтр может накладываться лишь в тех местах, в которых под каждым значением фильтра будет значение входного изображения.

Таким образом, входными параметрами свёрточного слоя являются:

- тензор размером $W_1 \times H_1 \times D_1$;
- 4 гиперпараметра: fc , fs , S , P ;

А выходным параметром слоя является тензор размером

$$W_2 \times H_2 \times D_2,$$

где $W_2 = (W_1 - fs + 2P) / S + 1$,

$H_2 = (H_1 - fs + 2P) / S + 1$,

$D_2 = fc$.

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Следующий блок сверточного кода - **слой подвыборки (пулинга)**

Данный слой позволяет уменьшить пространство признаков, сохраняя наиболее важную информацию. Существует несколько разных версий слоя пулинга, среди которых максимальный пулинг, средний пулинг и пулинг суммы. Наиболее часто используется именно слой макспулинга.

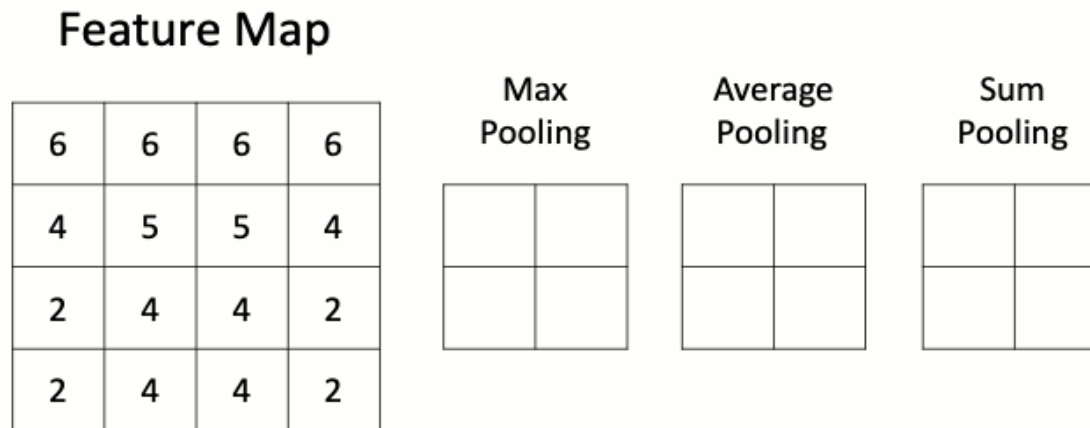


Рис. 23. Слой пулинга

Слою подвыборки требуется всего один гиперпараметр — шаг пулинга, то есть число раз, в которое нужно сократить пространственные размерности. Наиболее часто используется слой макспулинга с уменьшением размера входного тензора в два раза. Некоторые библиотеки позволяют задавать отдельные параметры уменьшения по высоте и ширине, однако чаще всего эти параметры совпадают.

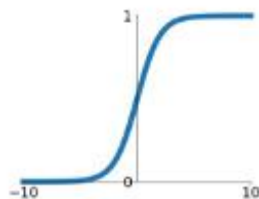
2. Типы нейронных сетей.

Свёрточная нейронная сеть

Слой активации представляет из себя некоторую функцию, которая применяется к каждому числу входного изображения. Наиболее часто используются такие функции активации, как ReLU, Sigmoid, Tanh, LeakyReLU. Обычно активационный слой ставится сразу после слоя свёртки, из-за чего некоторые библиотеки даже встраивают ReLU функцию прямо в свёрточный слой. Подробнее про функции активации можно почитать здесь: [функции активации](#).

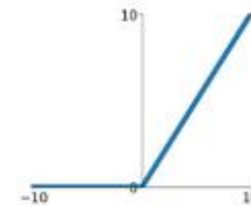
Sigmoid

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



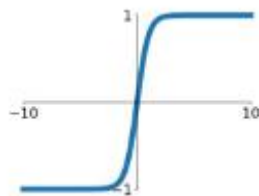
ReLU

$$\max(0, x)$$



tanh

$$\tanh(x)$$



Leaky ReLU

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

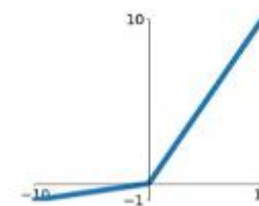


Рис.24. Функции активации

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Полносвязный слой содержит матрицу весовых коэффициентов и вектор смещений и ничем не отличается от такого же слоя в обыкновенной полносвязной сети. Единственным гиперпараметром слоя является количество выходных значений. При этом результатом применения слоя является вектор или тензор, у которого матрицы в каждом канале имеют размер 1×1 .

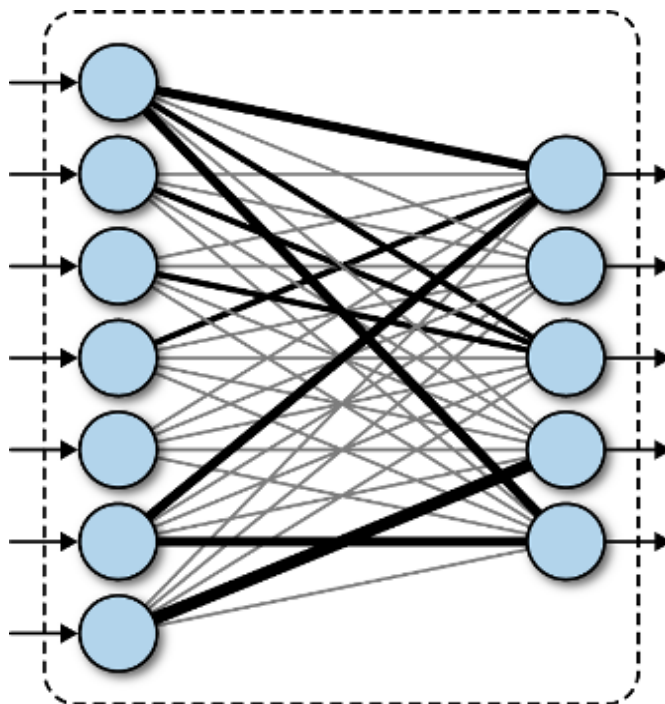


Рис. 25. Полносвязный слой свёрточной нейронной сети

2. Типы нейронных сетей.

Свёрточная нейронная сеть

Обучение свёрточной сети. Как и полносвязная нейронная сеть, свёрточная сеть обучается с помощью алгоритма обратного распространения ошибки. Сначала выполняется прямое распространение от первого слоя к последнему, после чего вычисляется ошибка на выходном слое и распространяется обратно. При этом на каждом слое вычисляются градиенты обучаемых параметров, которые в конце обратного распространения используются для обновления весов с помощью градиентного спуска.

Недостатком сверточных сетей является необходимость выполнения огромного количества математических операций и вычислений. Это приводит к высокому потреблению и к сложности интегральной реализации.

Потенциальной заменой для сверточных нейронных сетей уже в ближайшем будущем могут стать импульсные нейронные сети (ИНС или Spiking neural network, SNN).

2. Типы нейронных сетей.

Импульсная нейронная сеть

Импульсные нейронные сети. ети представляются в виде импульсов (спайков) одинаковой амплитуды и длительности. При этом информация может кодироваться с помощью частоты, фазы, порядка следования и позицией импульсов. Каждый нейрон характеризуется некоторым потенциалом (по аналогии с биологическими нейронами). Если нейрон принимает импульсы – потенциал увеличивается. При превышении некоторого порога происходит возбуждение, и нейрон сам генерирует импульс (пачку импульсов). После генерации спайка нейрон возвращается в невозбужденное состояние.

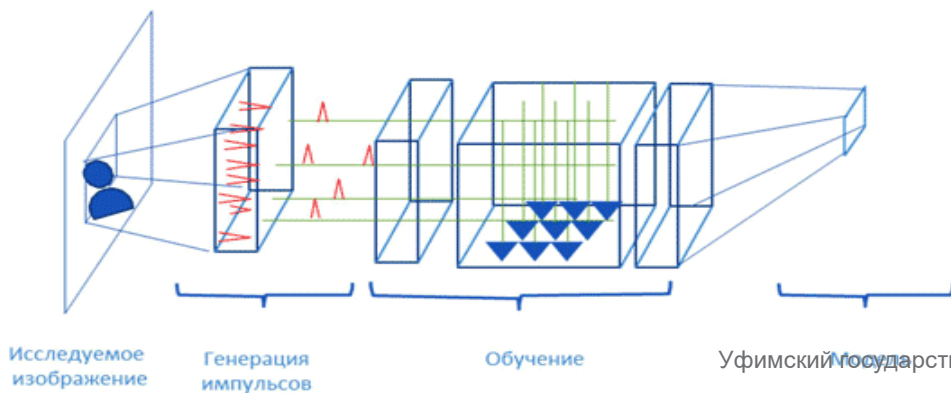
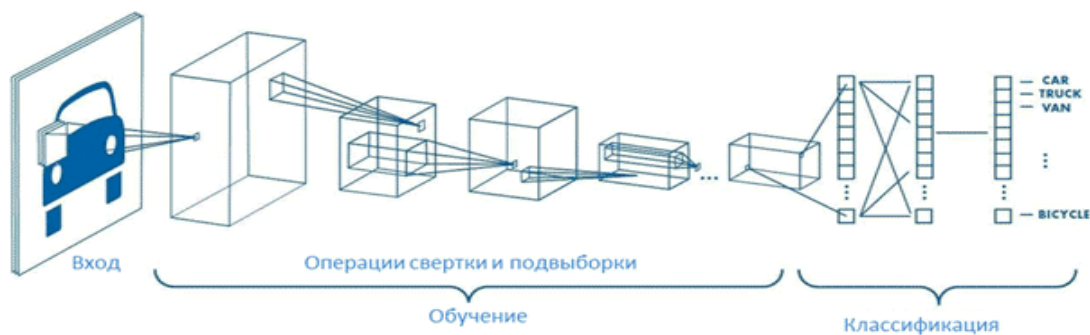
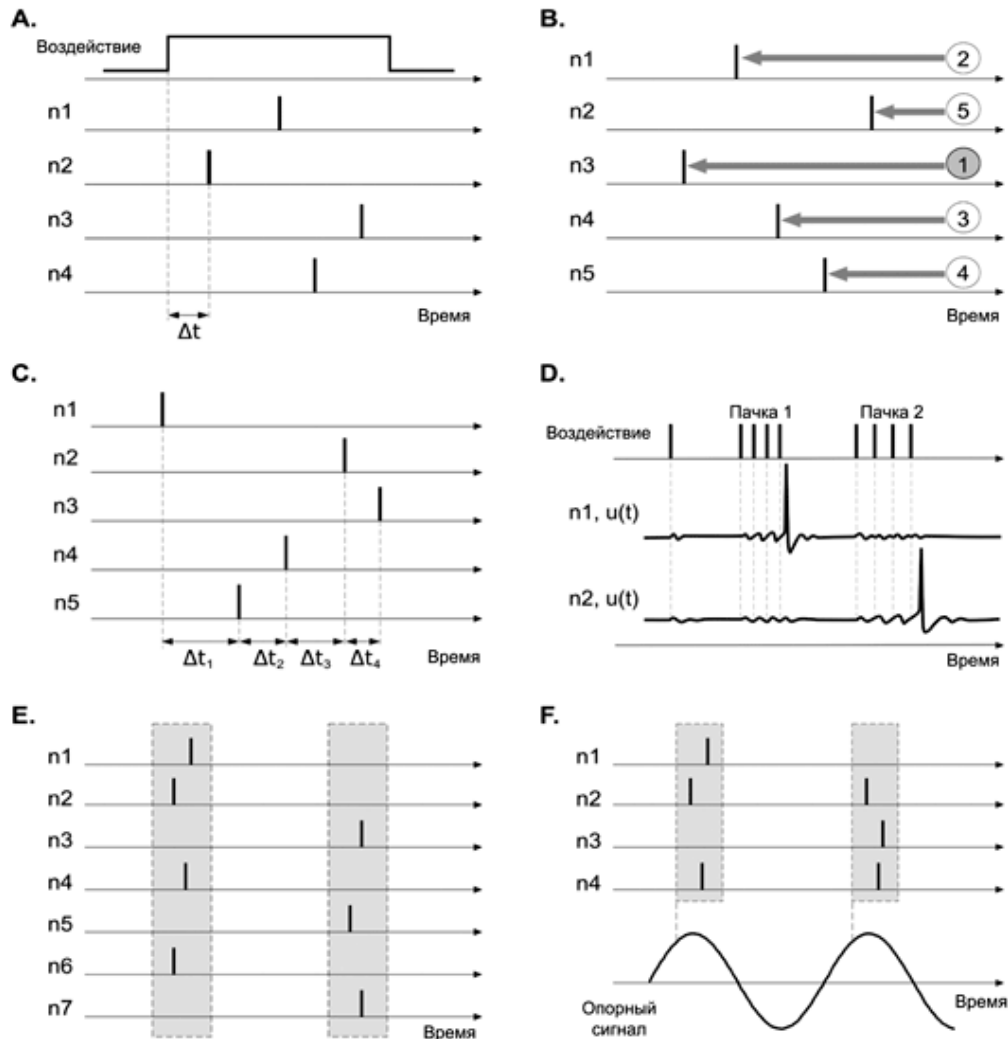


Рис. 26. Сравнение импульсной нейронной сети (SNN) и сверточной нейронной сети (CNN)

2. Типы нейронных сетей.

Импульсная нейронная сеть

Сейчас существует множество различных **способов кодирования данных**. Рассмотрим лишь некоторые из них



- Кодирование **задержкой до первого импульса** (рис. А). В данном случае информация кодируется временем задержки от начала воздействия до появления первого импульса.
- Кодирование **порядком следования** (рис. В). В данном случае информацию несет непосредственно порядок следования импульсов от группы нейронов.
- Кодирование **относительной задержкой импульсов** (рис. С). Этот способ похож на первый вариант, но в данном случае информация кодируется относительной задержкой импульсов внутри группы нейронов.
- **Резонансное кодирование** (рис. D). В данном случае, каждый нейрон реагирует только на пачку стимулирующих импульсов определенной частоты.
- **Синхронное кодирование** (рис. Е). В данном случае речь снова идет о групповом кодировании, в котором каждый нейрон кодирует один бит информации. В идеале все нейроны должны генерировать (или не генерировать) спайки одновременно, но допускается некоторая рассинхронизация в рамках временного окна, как показано на рис. Е.
- **Фазовое кодирование** (рис. F). Вариант синхронного кодирования, при котором информация кодируется фазовой задержкой относительно некоторого опорного сигнала.

2. Типы нейронных сетей.

Импульсная нейронная сеть

Таким образом, стоит отметить, что ИНС-сети работают с импульсными данными, в отличие от СНС-сетей, которые работают с матрицами чисел. С одной стороны, такое представление информации оказывается ближе к биологическому «оригиналу», а с другой стороны, при правильной реализации сеть демонстрирует лучшие результаты как по эффективности, так и по уровню потребления. Одним из лидеров по созданию SNN-технологий является компания BrainChip. В настоящий момент решения BrainChip реализуют нейронные сети прямого распространения (feed-forward), в которых данные передаются в одном направлении без сигналов обратной связи.

Сейчас компания работает в двух направлениях:

- Реализация ИНС-сетей на программном уровне для дальнейшего использования на обычных компьютерах в виде программы BrainChip Studio;
- Реализация ИНС-сетей в виде микросхемы нейропроцессора AkidaNSOC. Запуск микросхемы в производство планируется осуществить в следующем году.

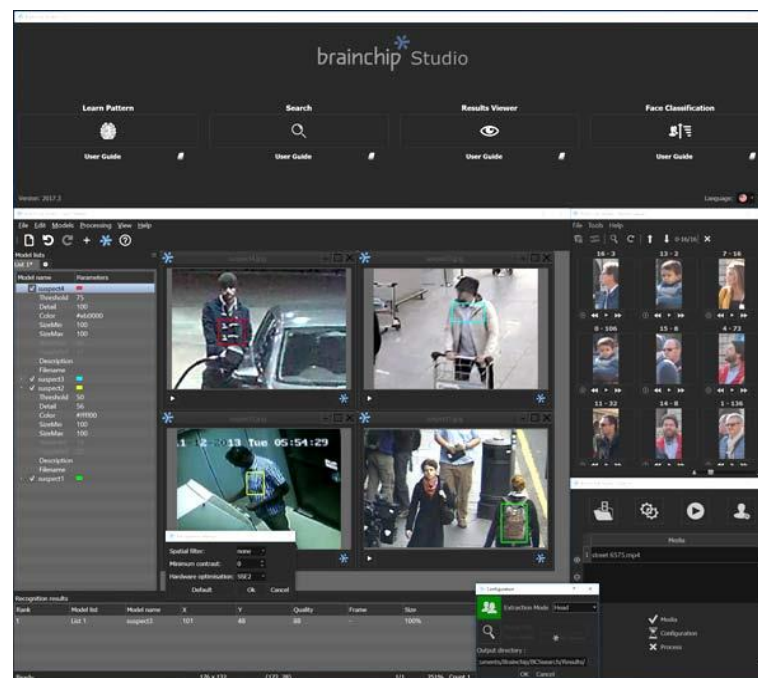


Рис. 28. Внешний вид программы BrainChip Studio. Данная программа позволяет выполнять поиск людей или объектов по фотографиям, что делает ее пригодной для широкого спектра приложений.

2. Типы нейронных сетей.

Импульсная нейронная сеть

BrainChip Studio строится на фирменном программном движке Akida Execution Engine, который реализует ИНС-сеть, содержит преобразователи данных и модель обучения (рис.). Кроме того, программа включает вспомогательные инструменты, например, программу, помогающую обнаруживать лица на фотографиях, инструменты визуализации и т.д. Благодаря ИНС-сети BrainChip Studio позволяет распознавать объекты в условиях высокого уровня шума и с исходными изображениями со сверхнизким разрешением от 24x24 пикселей.

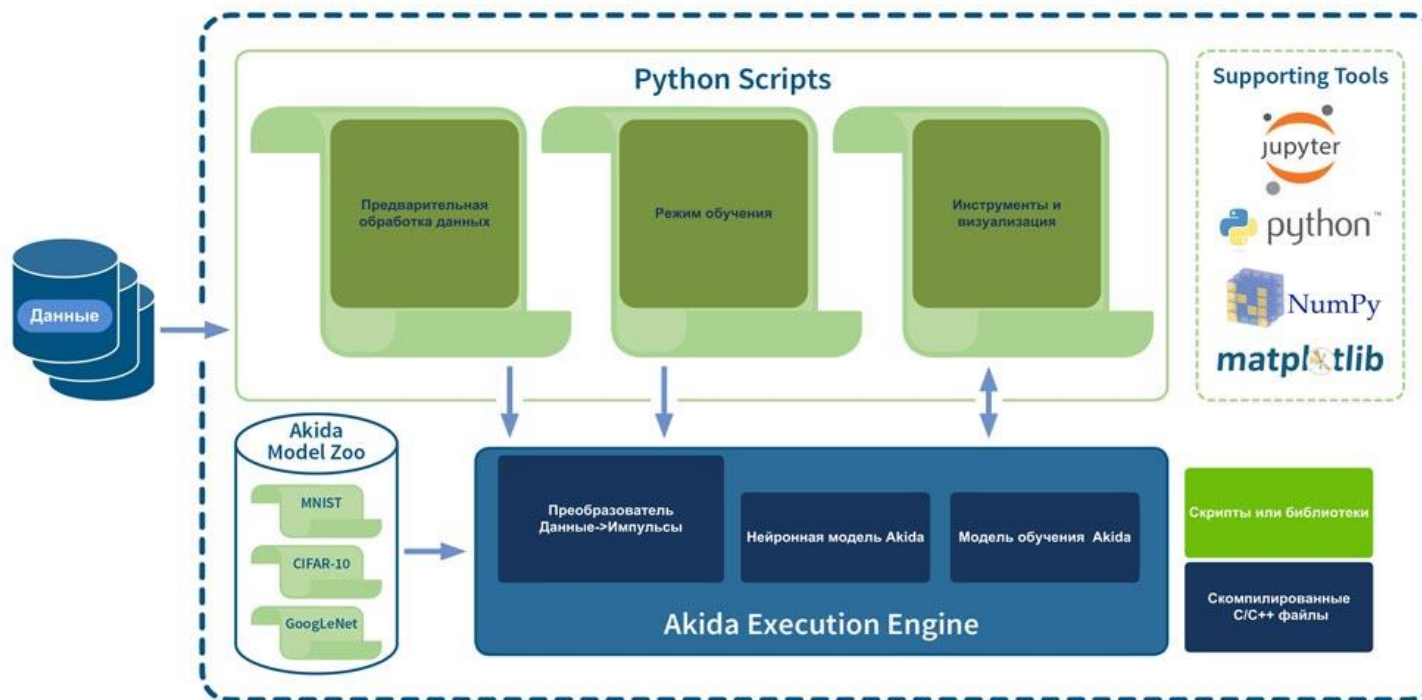


Рис. 29. Составляющие BrainChip Studio

2. Типы нейронных сетей.

Импульсная нейронная сеть

Алгоритм работы BrainChip Studio изображен на рис. В данном случае источником видео данных является камера. Программа с помощью вспомогательных инструментов выполняет поиск лиц на фотографии, после чего выполняется преобразование данных в импульсный формат, понятный ИНС-сети. Далее выполняется обработка каждого изображения в ИНС-сети. Изображение разыскиваемого человека может быть обработано параллельно. В конце программа производит сравнение результатов и обнаруживает совпадения.

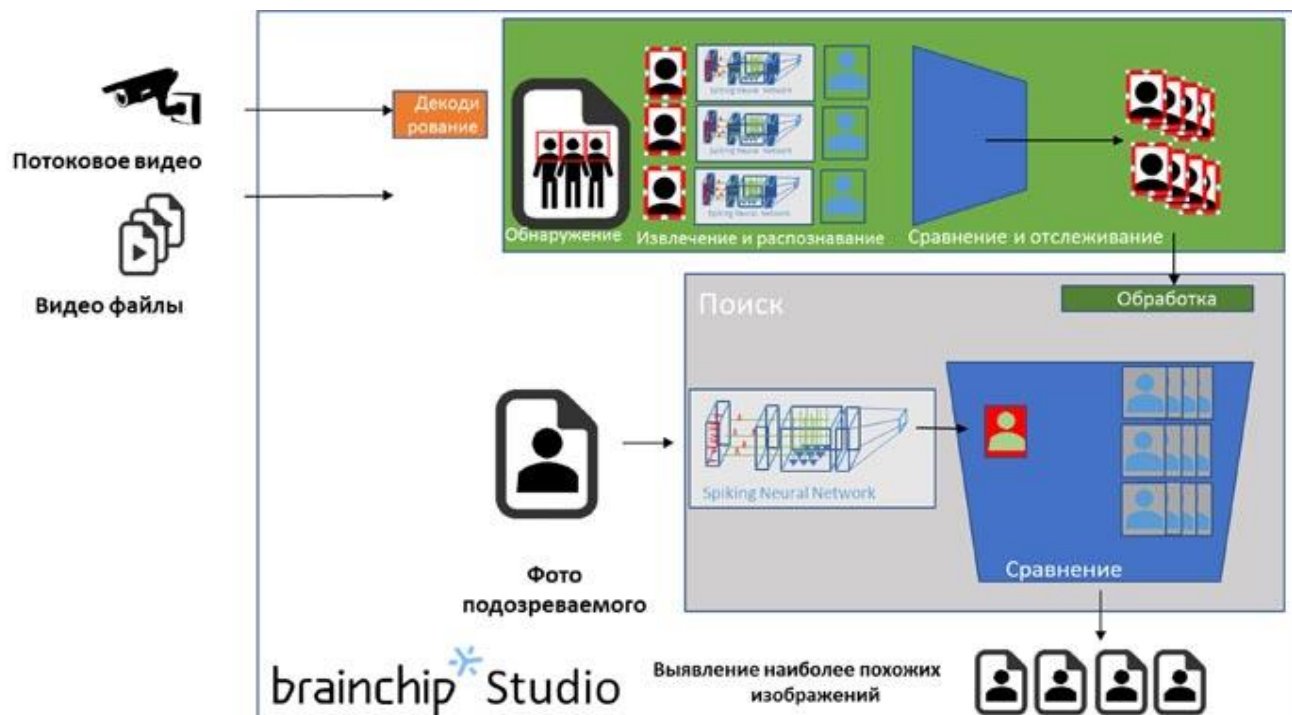


Рис. 30. Принцип работы BrainChip Studio

2. Типы нейронных сетей.

Импульсная нейронная сеть

Как видно из описания, BrainChip Studio предполагает обучение «на лету». Это весьма важно в данном конкретном случае, ведь, например, в полиции не будет большого количества времени и данных на выполнения долгого обучения нейронной сети для распознавания лица злоумышленника, появившегося в толпе. Этот чип предполагает эмуляцию ИНС-сети силами обычного процессора. Естественно, это негативно сказывается на быстродействии. Если требуется увеличить эффективность работы, то рекомендуется воспользоваться ускорителем BrainChip Accelerator, который повышает производительность BrainChip Studio в 6 раз (рис.).



Рис. 31. Ускоритель нейронных вычислений BrainChip Accelerator

2. Типы нейронных сетей.

Импульсная нейронная сеть

Следующим шагом, анонсированным компанией BrainChip, станет выпуск микросхемы нейропроцессора Akida NSOC. Ориентировочно процессор появится в следующем году. Конкретных данных по стоимости, корпусному исполнению и используемому технологическом процессу пока нет. Тем не менее, компания предоставила некоторые данные по архитектуре Akida NSOC. Она будет объединять ИНС-сеть, конвертер данных, встроенный управляющий процессор, систему питания, интерфейс для подключения внешних датчиков, интерфейс памяти, коммуникационные интерфейсы, интерфейс для каскадирования (рис.). Рассмотрим кратко каждый из блоков.

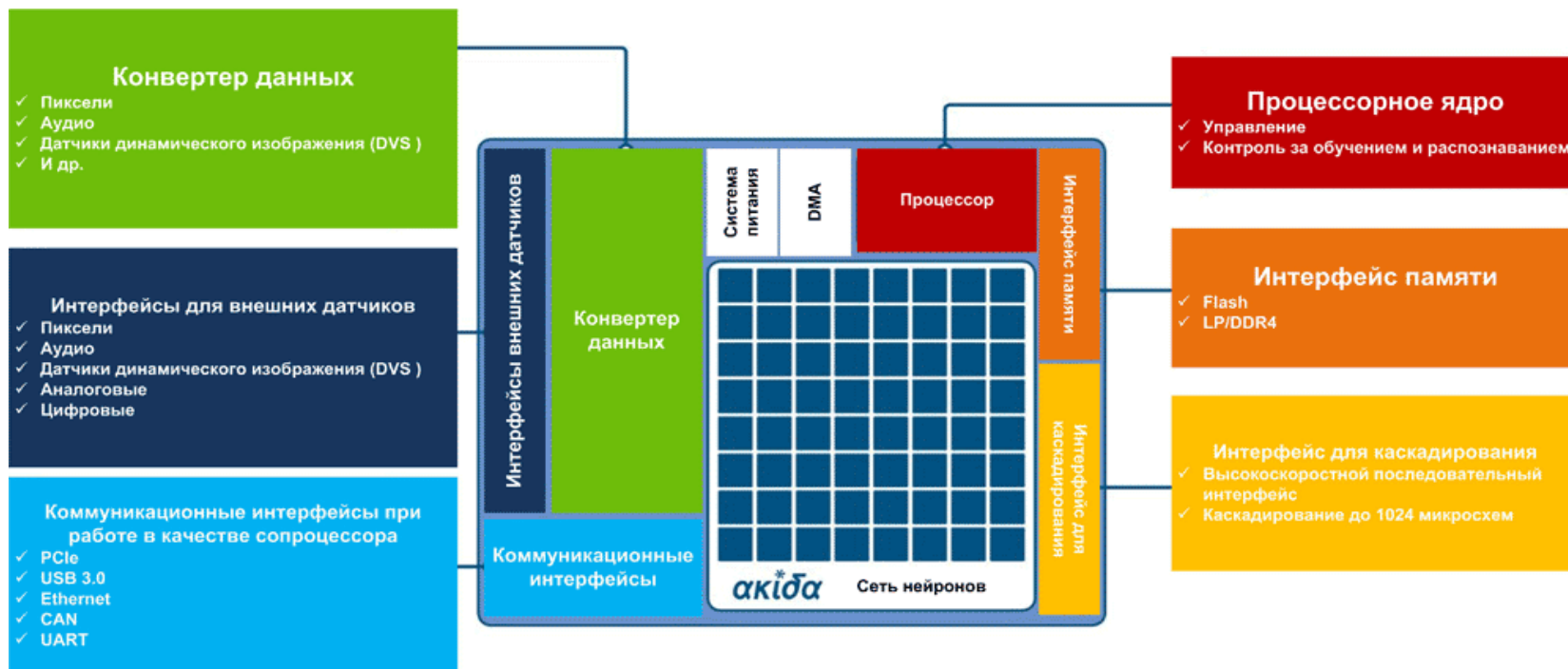


Рис. 32 Структура нейронного процессора Akida NSOC

2. Типы нейронных сетей.

Импульсная нейронная сеть

По заверениям производителя ИНС-сеть нейропроцессора Akida NSOC должна содержать 1,2 миллиона нейронов и 10 миллиардов синапсов (рис.). Это позволит превзойти производительность существующих нейропроцессоров от IBM и Intel в 100 раз.

И вот последняя информация с сайта компании <https://brainchipinc.com/akida-neuromorphic-system-on-chip/> 2021 года.

Akida NSoC представила собой революционное новое поколение вычислительных устройств нейронной обработки для устройств и систем Edge AI. Каждый Akida NSoC имеет эффективно 1,2 миллиона нейронов и 10 миллиардов синапсов, что на порядок выше эффективности, чем другие нейронные устройства обработки на рынке. Сравнения с ведущими ускорительными устройствами DNN показывают на порядок лучшие изображения/второй/ватт работающий стандартный отраслевой стандарт с MobileNet, MobileNet-SSD и Key Word Spotting, сохраняя при этом отличную точность. Предназначен для использования в качестве самостоятельного встроенного ускорителя или в качестве со-процессора. Он включает в себя интерфейсы для датчиков ADAS, аудиосенсоров и других датчиков IoT. Он также имеет высокоскоростные интерфейсы данных, такие как PCI-Express, USB, I2C и I3S. На чипе MPU используется для управления конфигурацией Akida Neuron Fabric, а также вне чиповой связи метаданных. Akida NSoC представляет собой масштабируемое решение, используя встроенный серийный чип для чипа подключения, чтобы позволить до 64 устройств, которые будут массивироваться для одного решения.

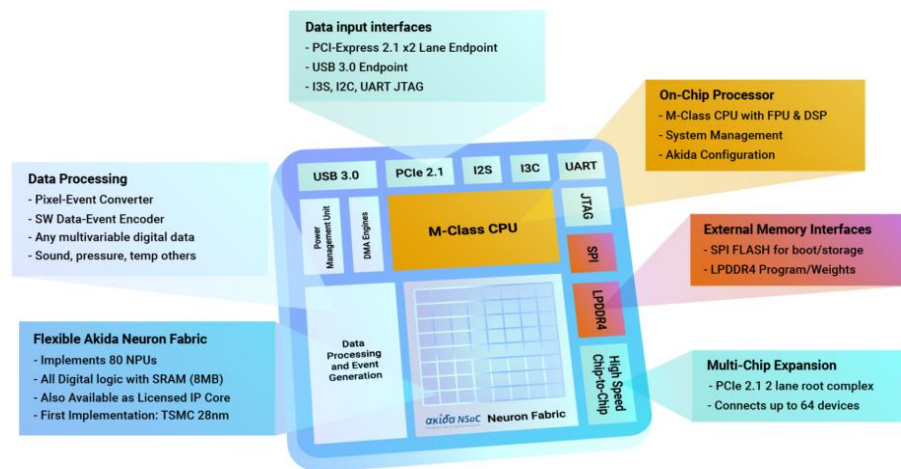


Рис. 32. ИНС-сеть нейропроцессора Akida NSOC

Заключение

Еще раз отметим, что нейронные сети используются для решения различных неформализованных задач, таких, например, как распознавание изображений, фильтрация звука и, даже, поиск брака на производстве. В настоящий момент наблюдается рост интереса к нейронным технологиям, что приводит к их дальнейшему совершенствованию. В целом это напрямую связано с развитием искусственного интеллекта. Речь идет не о компьютерных играх и роботах-пылесосах, а о по-настоящему серьезных проектах, которые используют различные технологии для решения неформализованных задач. Спектр таких задач оказывается огромен – от распознавания изображений и музыки в интернете, до фильтрации и обработки звуков и изображений в промышленности.

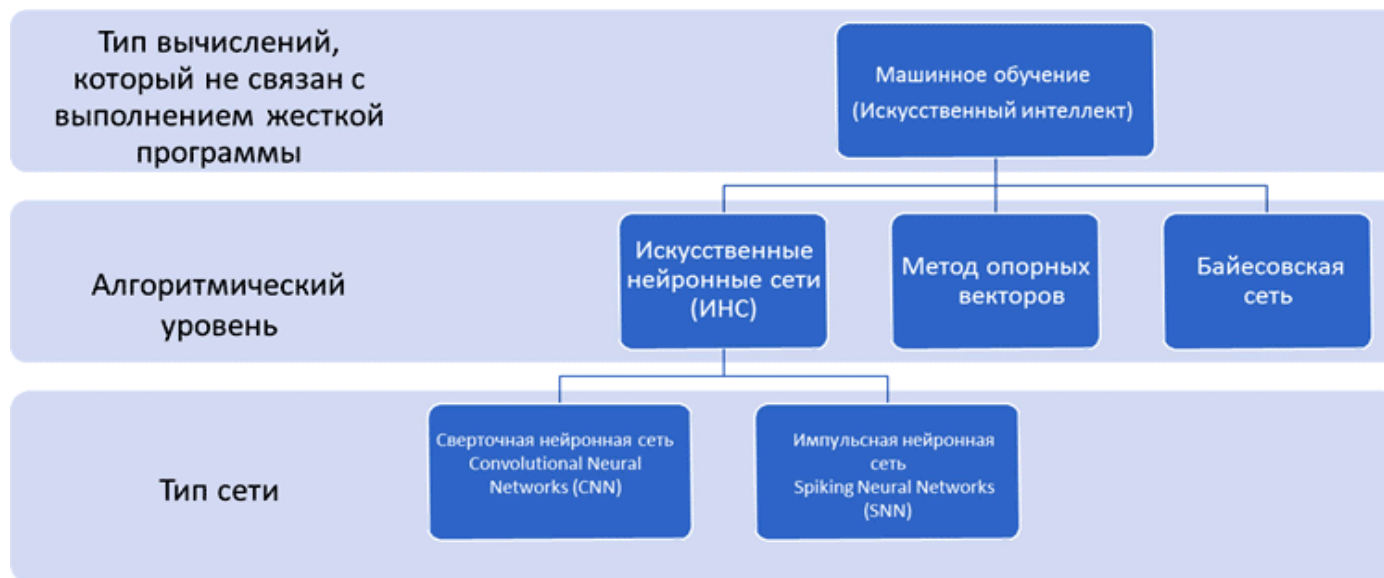


Рис. 33. Положение нейронных сетей в машинном обучении

Заключение

Несмотря на совершенствование интегральных технологий, традиционные процессоры по-прежнему не способны эффективно решать нелинейные и неформализованные задачи. В то же время, мозг животных и человека с такими задачами справляется отлично. Такую ситуацию нельзя объяснить отсталостью технологий, так как уже в 80-е годы размеры интегральных транзисторов были сопоставимы с размерами нейронов мозга. Объяснение здесь кроется в качественных архитектурных различиях между традиционным процессором и мозгом живых существ.

Естественно, инженеры уже давно горят желанием повторить структуру мозга в виде интегрального решения или в виде какого-либо программного кода. К сожалению, все не так просто. Во-первых, механизмы передачи информации в мозге до сих пор не до конца изучены. Во-вторых, нет готового решения по созданию идеальной искусственной нейронной сети.

В результате, сейчас на рынке присутствует два подхода к построению нейронных сетей – в виде специализированного ПО, которое выполняет эмуляцию параллельных вычислений с помощью обычных процессоров, и в виде нейропроцессоров, которые реализуют нейронные сети в интегральном исполнении.

2. Типы нейронных сетей.

Заключение

Архитектура традиционных процессоров сильно отличается от архитектуры нейронных сетей (рис.). Обычные процессоры строятся на базе одного процессорного ядра. Такой подход обеспечивает великолепную производительность при выполнении линейного последовательного кода. Решение нескольких параллельных задач оказывается менее эффективным. При этом, чем больше задач, тем хуже справляется процессор. Современные процессоры могут иметь несколько ядер, но это все равно не спасает ситуацию при решении большого количества параллельных задач.

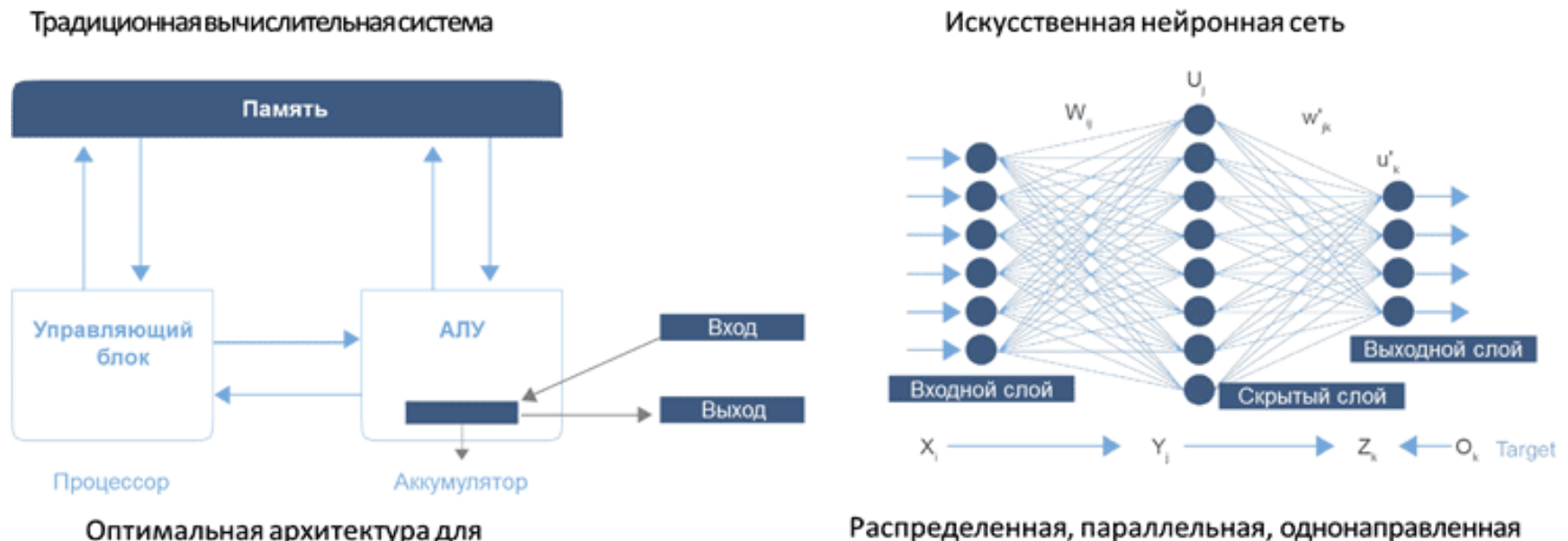


Рис. 34. Сравнение традиционной вычислительной системы и нейронной вычислительной сети

1. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel: Backpropagation Applied to Handwritten Zip Code Recognition, *Neural Computation*, 1(4):541-551, Winter 1989.
2. Convolutional Neural Networks (LeNet) - DeepLearning 0.1 documentation. DeepLearning 0.1. LISA Lab. Дата обращения: 31 августа 2013.
3. Matusugu, Masakazu; Katsuhiko Mori; Yusuke Mitari; Yuji Kaneda. Subject independent facial expression recognition with robust face detection using a convolutional neural network (англ.) // *Neural Networks : journal*. — 2003. — Vol. 16, no. 5. — P. 555—559. — doi:10.1016/S0893-6080(03)00115-1.
4. (2000) "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit". 405: 947–951.
5. (2011) "Deep sparse rectifier neural networks" in AISTATS..
6. (2010) "Rectified linear units improve restricted Boltzmann machines" in ICML.. Проверено 2016-12-22. Архивная копия от 24 марта 2014 на Wayback Machine
7. Romanuke, Vadim. Appropriate number and allocation of ReLUs in convolutional neural networks (англ.) // *Research Bulletin of NTUU "Kyiv Polytechnic Institute" : journal*. — 2017. — Vol. 1. — P. 69—78. — doi:10.20535/1810-0546.2017.1.88156.
8. Graham, Benjamin (2014-12-18), Fractional Max-Pooling, arXiv:1412.6071 [cs.CV]
9. Springenberg, Jost Tobias; Dosovitskiy, Alexey; Brox, Thomas & Riedmiller, Martin (2014-12-21), Striving for Simplicity: The All Convolutional Net, arXiv:1412.6806 [cs.LG]
10. Jain, V. and Seung, S. H. (2008). Natural image denoising with convolutional networks. In NIPS'2008.
11. Lee, H., Grosse, R., Ranganath, R., and Ng, A. Y. (2009a). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In ICML'2009.
12. Zeiler, M., Krishnan, D., Taylor, G., and Fergus, R. (2010). Deconvolutional networks. In CVPR'2010.
13. Документация с официального сайта brainchipinc.com
14. Daphne Cornelisse. An intuitive guide to Convolutional Neural Networks. <https://mediufreecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050>
15. Ponulak, Filip (2010). «Introduction to spiking neural networks: Information processing, learning and». *Acta neurobiologiae experimentalis* 71: 409-433.
16. <https://ru.wikipedia.org/wiki/>
17. <https://tproger.ru/translations/neural-network-zoo-1/>

На следующих лекциях будут изучены особенности применения интеллектуальных систем в нефтегазовой промышленности

Спасибо за внимание!