

### Тема 3. Нейронные сети

Так как основная задача искусственного интеллекта – моделирование рассуждений, а природное «устройство способное думать» – это мозг, то очевидной задачей является создание «искусственного мозга» «по образу и подобию» человеческого. Исследованием этого вопроса стали заниматься в рамках направления искусственного интеллекта нейрокибернетика.

Устройства мозга изучают такие науки как психология, нейрофизиология, нейробиология, обладающие достаточным объемом знаний. Основной идеей нейрокибернетики стало воссоздание «в железе» клетки мозга – нейрона.

Нейроны - специализированные клетки, способные принимать, обрабатывать, кодировать, передавать и хранить информацию, организовывать реакции на раздражения, устанавливать контакты с другими нейронами, клетками органов. Уникальными особенностями нейрона являются способность генерировать электрические разряды и передавать информацию с помощью специализированных окончаний — синапсов.

Число нейронов мозга человека приближается к  $10^{11}$ . На одном нейроне может быть до 10 000 синапсов. Если только эти элементы считать ячейками хранения информации, то можно прийти к выводу, что нервная система может хранить  $10^{19}$  ед. информации, т. е. способна вместить практически все знания, накопленные человечеством.

На рис. 8. приведена схема строения "типичного" нейрона.

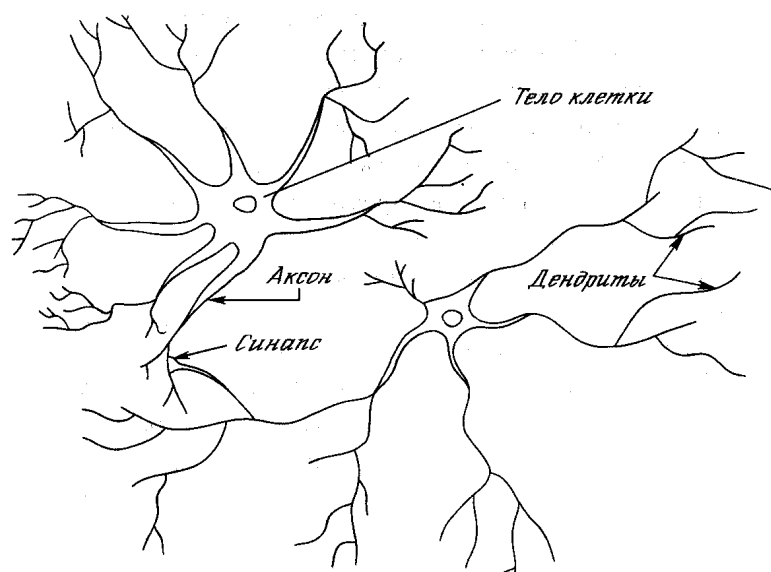


Рис. 8. Общая схема строения биологического нейрона.

Тело клетки содержит множество ветвящихся отростков двух типов. Отростки первого типа, называемые дендритами за их сходство с кроной

раскидистого дерева, служат в качестве входных каналов для нервных импульсов от других нейронов. Эти импульсы поступают в сому или тело клетки размером от 3 до 100 микрон, вызывая ее специфическое возбуждение, которое затем распространяется по выводному отростку второго типа - аксону. Длина аксонов обычно заметно превосходит размеры дендритов, в отдельных случаях достигая десятков сантиметров и даже метров. Нейрон может находиться в 2 состояниях: возбужденном или невозбужденном.

1943 год стал годом рождения теории искусственных нейронных сетей. Дж. Маккалок и У. Питт предложили модель формального нейрона (рис. 9) и описали основные принципы построения нейронных сетей.

Синапсы нейрона – места контактов нервных волокон – согласно этой модели, передают сигналы, определяя силу воздействия сигнала с этого входа на выходной сигнал, поступающий на аксон. Для этого каждому входу ставится в соответствие весовой коэффициент  $w_i$ . Дендриты получают входной сигнал, представленный вектором  $x_i$ . Затем нейрон обрабатывает поступивший сигнал, производя взвешенное суммирование и нелинейное преобразование, используя для этого активационную функцию (функции активации могут быть разными, наиболее распространенные линейная, пороговая и сигмоида), аргументом которой будет результат суммирования минус пороговое значение. Это значение определяет уровень сигнала, на который нейрон будет реагировать.

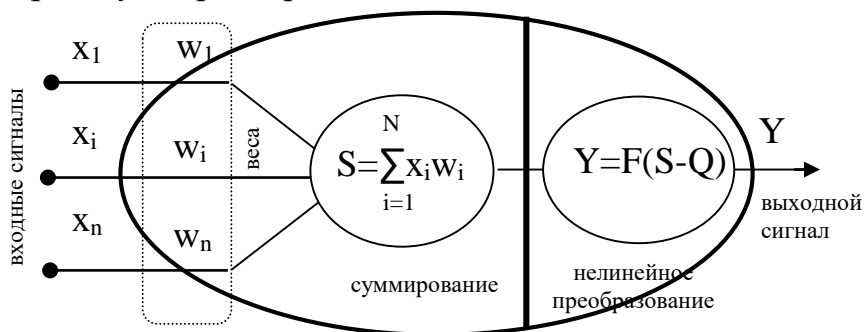


Рис. 9. Модель формального нейрона

Главным свойством биологического нейрона является его способность к обучению, его универсальность, способность решать различные задачи. Описанная выше модель неспособна к этому. Обучаемой она стала лишь в 1949 году благодаря Д.Хеббу (D. Hebb), который, опираясь на физиологические и психологические исследования, выдвинул гипотезу об обучаемости биологических нейронов. Его метод обучения стал отправной точкой для алгоритмов обучения нейронных сетей без учителя.

Уже через год после этого в 1957 году в мировом научном мире произошло второе по значимости в истории нейронных сетей событие:

американский физиолог Ф. Розенблатт разработал модель зрительного восприятия и распознавания — перцептрон (perceptron), а затем и построил первый нейрокомпьютер Марк-1.

Нейронные сети были просты, понятны и многообещающи. Сложные процессы мышления, казалось, были готовы раскрыться перед человеком, для их описания требовались лишь элементарные математические операции: сложение, умножение и линейная функция. На перцептрон, а затем и многослойный перцептрон возлагались большие надежды, поэтому бурный энтузиазм, с которым он был принят, достаточно быстро сменился жесткой критикой.

В 1969 году вышла в свет книга «Перцептроны» М. Минского и С. Паперта, которая ознаменовала окончание первого этапа в истории нейронных сетей. В этой книге был проведен анализ возможностей однослойных нейронных сетей и их ограничений.

Ограничения, обнаруженные Минским, были характерны однослойным нейронным сетям, но не многослойным нейронным сетям, которым была присуща другая проблема – их обучение.

Предложенный в 1986 году Д. Румельхардом алгоритм обратного распространения ошибки стал одним из ведущих факторов, породивших современный нейросетевой бум, т.к. являлся эффективным способом обучения нейронной сети достаточно произвольной структуры.

Нейронные сети могут реализовываться как программно, так и аппаратно. Постепенно направление нейрокибернетика преобразовалось в нейрокомпьютинг, шестое поколение компьютеров, базирующееся на нейронных сетях, имеющее свои достоинства и недостатки. Сравнительный анализ машины фон Неймана, идеи которой воплощены в первых 4 поколениях, и биологической нейронной системы, являющейся прообразом нейрокомпьютера, представлен в таблице 1.

*Таблица 2. Сравнение машины фон Неймана с биологической нейронной системой.*

	<b>Машина фон Неймана</b>	<b>Биологическая нейронная система</b>
Процессор	Сложный	Простой
	Высокоскоростной	Низкоскоростной
	Один или несколько	Большое количество
Память	Отделена от процессора	Интегрирована в процессор
	Локализована	Распределенная

	Адресация не по содержанию	Адресация по содержанию
Вычисления	Централизованные	Распределенные
	Последовательные	Параллельные
	Хранимые программы	Самообучение
Надежность	Высокая уязвимость	Живучесть
Специализация	Численные и символьные операции	Проблемы восприятия
Среда функционирования	Строго определенная	Плохо определенная
	Строго ограниченная	Без ограничений

## Классификация искусственных нейронных сетей

**По топологии (рис. 10, рис. 11):**

- полносвязные (каждый нейрон связан со всеми остальными нейронами, в том числе и сам с собой);
- многослойные (нейроны располагаются слоями и каждый нейрон последующего слоя связан со всеми нейронами текущего слоя).

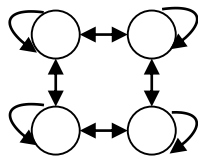


Рис. 10. Полносвязная нейронная сеть

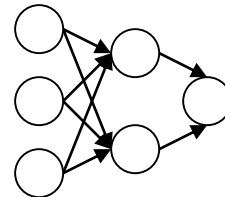


Рис. 11. Слоистая нейронная сеть

**По организации обучения:**

- с учителем (нейронную сеть обучают, подавая на вход значения обучающей выборки и предоставляя требуемые выходные значения);
- без учителя (на входы нейронной сети подают множество объектов, и нейронная сеть сама делит их на кластеры или классы).

**По типам структур (рис. 12):**

- нейроны с одним типом функции активации (все нейроны сети имеют одну функцию активации  $f(x)$ , например, линейную);
- нейроны с несколькими типами функций активации (нейроны сети имеют различные функции активации).



Рис.12 Активационные функции

### **По типу связей:**

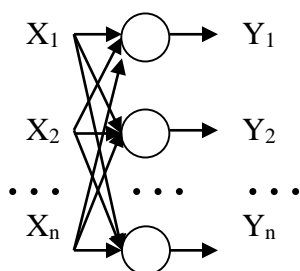
- прямого распространения (без обратных связей между нейронами, к таким сетям относятся однослойный и многослойный персептрон, сеть радиальных базисных функций);
- рекуррентные (с обратной связью, от выходов нейронов к входам, к таким сетям относятся соревновательные сети и сеть Хопфилда).

### **По типу сигнала:**

- бинарные (на входы подаются только нули и единицы);
- аналоговые (на входы нейронов подаются значения непрерывных функций).

## **Однослойные искусственные нейронные сети**

Один нейрон способен выполнять простейшие процедуры распознавания, но только соединение нескольких нейронов способно решить практически полезную задачу. Простейшая сеть (рис. 13) состоит из группы нейронов, образующих слой).



*Рис. 13. Однослойная нейронная сеть*

### **Обучение по дельта-правилу**

Дельта-правило является обобщением алгоритма обучения персептрона. Дельта-правило работает только с непрерывными, дифференцируемыми функциями в режиме обучения с учителем (supervised learning). Ошибка, вычисляемая в процессе обучения сети, - это функция, характеризующая качество обучения данной сети, поэтому процесс обучения нейронной сети можно представить, как процесс минимизации функции ошибки. Направление изменения значения функции можно установить, вычислив производную для функции одного переменного или градиент для функции многих переменных. При минимизации значения функции многих переменных меньшее значение необходимо искать в направлении антиградиента.

В данном алгоритме обучения начальные веса могут быть любыми.

Процесс обучения можно считать завершенным, если достигнута некая заранее установленная минимальная ошибка или алгоритм проработал условленное количество раз.

Алгоритм обучения по дельта-правилу:

**1 шаг:** инициализация матрицы весов (и порогов, в случае использования пороговой функции активации) случайным образом.

**2 шаг:** предъявление нейронной сети образа (на вход подаются значения из обучающей выборки – вектор  $X$ ), берется соответствующий выход (вектор  $D$ ).

**3 шаг:** вычисление выходных значений нейронной сети (вектор  $Y$ ).

**4 шаг:** вычисление для каждого нейрона величины расхождения реального результата с желаемым.

$$\varepsilon_i = (d_i - y_i),$$

где  $d_i$  – желаемое выходное значение на  $i$ -нейроне,  $y_i$  – реальное значение на  $i$ -нейроне.

**5 шаг:** изменение весов (и порогов при использовании пороговой функции) по формулам:

$$w_{ij}(t+1) = w_{ij}(t) - \eta \cdot \varepsilon_i \cdot x_j,$$

$$\theta_i(t+1) = \theta_i(t) - \eta \cdot \varepsilon_i,$$

где  $t$ -номер текущей итерации цикла обучения,  $w_{ij}$  – вес связи  $j$ -входа с  $i$ -нейроном,  $\eta$  – коэффициент обучения, задается от 0 до 1,  $x_j$  – входное значение,  $\theta_i$  – пороговое значение  $i$ -нейрона.

**6 шаг:** проверка условия продолжения обучения (вычисление значения ошибки и/или проверка заданного количества итераций). Если обучение не завершено, то 2 шаг, иначе заканчиваем обучение.

### Многослойные нейронные сети

Однослойные нейронные сети имеют свои ограничения, на которые указал Минский, многослойные сети свободны от них. Они обладают большими вычислительными возможностями. Хотя созданы сети всех конфигураций, какие только можно себе представить, послойная организация нейронов копирует слоистые структуры определенных отделов мозга. Многослойные сети образуются каскадами слоев. Выход одного слоя является входом для последующего слоя, такая организация сети образует многослойную нейронную сеть с прямым распространением (рис. 14).

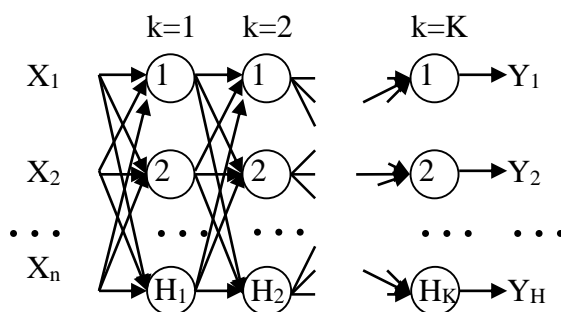


Рис. 14. Многослойная нейронная сеть прямого распространения

Многослойная сеть может содержать произвольное количество слоев (K), каждый слой состоит из нескольких нейронов, число которых также может быть произвольно ( $N_k$  – количество нейронов в слое), количество входов  $n$ , количество выходов  $N=N_k$  - числу нейронов в выходном (последнем) слое.

Слои между первым и последним называются промежуточными или скрытыми. Веса в такой сети имеют три индекса  $i$ - номер нейрона следующего слоя, для которого связь входная,  $j$  – номер входа или нейрона текущего слоя, для которого связь выходная,  $k$  – номер текущего слоя в нейронной сети (для входов, вектора  $X$ ,  $k=0$ ).

### **Обучение методом обратного распространения ошибки**

Обучение алгоритмом обратного распространения ошибки предполагает два прохода по всем слоям сети: прямого и обратного.

При прямом проходе входной вектор подается на входной слой нейронной сети, после чего распространяется по сети от слоя к слою. В результате генерируется набор выходных сигналов, который и является фактической реакцией сети на данный входной образ. Во время прямого прохода все синаптические веса сети фиксированы.

Во время обратного прохода все синаптические веса настраиваются в соответствии с правилом коррекции ошибок, а именно: фактический выход сети вычитается из желаемого, в результате чего формируется сигнал ошибки. Этот сигнал впоследствии распространяется по сети в направлении, обратном направлению синаптических связей. Отсюда и название – алгоритм обратного распространения ошибки. Синаптические веса настраиваются с целью максимального приближения выходного сигнала сети к желаемому.

Алгоритм обучения по дельта-правилу:

**1 шаг:** инициализация матриц весов случайным образом (в циклах).

**2 шаг:** предъявление нейронной сети образа (на вход подаются значения из обучающей выборки – вектор  $X$ ) и берется соответствующий выход (вектор  $D$ ).

**3 шаг (прямой проход):** вычисление в циклах выходов всех слоев и получение выходных значений нейронной сети (вектор  $Y$ ).

$$y_i^k = f\left(\sum_{j=0}^{H_{k-1}} w_{ij}^k \cdot y_j^{k-1}\right),$$

$$y_j^0 = x_j,$$

$$y_0^{k-1} = 1,$$

$$x_0 = 1,$$

где  $y_i^k$  - выход  $i$ -нейрона  $k$ -слоя,  $f$  - функция активации,  $w_{ij}^k$  - синаптическая связь между  $j$ -нейроном слоя  $k-1$  и  $i$ -нейроном слоя  $k$ ,  $x_j$  - входное значение.

**4 шаг (обратный проход):** изменение весов в циклах по формулам:

$$w_{ij}^k(t+1) = w_{ij}^k(t) + \eta \cdot \delta_i^k \cdot y_j^{k-1},$$

$$\delta_i^k = (d_i - y_i) \cdot y_i \cdot (1 - y_i) -$$

для последнего (выходного) слоя,

$$\delta_i^k = y_i \cdot (1 - y_i) \cdot \sum_{l=1}^{H_{k+1}} \delta_l^{k+1} \cdot w_l^{k+1} -$$

для промежуточных слоев, где  $t$ -номер текущей итерации цикла обучения (номер эпохи),  $\eta$ - коэффициент обучения задается от 0 до 1,  $y_i^k$  - выход  $i$ -нейрона  $k$ -слоя,

$w_{ij}^k$  - синаптическая связь между  $j$ -нейроном слоя  $k-1$  и  $i$ -нейроном слоя  $k$ ,  $d_i$  - желаемое выходное значение на  $i$ -нейроне,  $y_i$  - реальное значение на  $i$ -нейроне выходного слоя.

**5 шаг:** проверка условия продолжения обучения (вычисление значения ошибки и/или проверка заданного количества итераций). Если обучение не завершено, то 2 шаг, иначе заканчиваем обучение. Среднеквадратичная ошибка вычисляется следующим образом:

$$\varepsilon = \frac{1}{Q} \cdot \sum_{q=1}^Q \sum_{i=1}^H (d_i - y_i)^2,$$

где  $Q$  - общее число примеров,  $H$ - количество нейронов в выходном слое,  $d_i$  - желаемое выходное значение на  $i$ -нейроне,  $y_i$  - реальное значение на  $i$ -нейроне выходного слоя.

### Задачи, решаемые нейронными сетями

1. **Классификация образов.** Задача состоит в определении принадлежности входного образа (например, языкового сигнала или рукописного символа), представленного вектором признаков к одному или нескольким предварительно определенным классам. К известным приложениям относятся распознавание букв, распознавание языка, классификация сигнала электрокардиограммы, классификация клеток крови.
2. **Кластеризация/категоризация.** При решении задачи кластеризации обучающее множество не имеет меток классов. Алгоритм кластеризации основан на подобии образов и помещает похожие образы в один кластер. Известны случаи применения кластеризации для добычи знаний, сжатия данных и исследования свойств данных.



3. **Аппроксимация функций.** Предположим, что есть обучающая выборка  $((x_1, y_1), (x_2, y_2), \dots, (x_n, y_n))$  (пары данных вход-выход), которая генерируется неизвестной функцией  $F$ , искаженной шумом. Задача аппроксимации состоит в нахождении неизвестной функции  $F$ . Аппроксимация функций необходима при решении многочисленных инженерных и научных задач моделирования.
4. **Предвидение/прогноз.** Пусть заданы  $n$  дискретных отсчетов  $\{y(t_1), y(t_2), \dots, y(t_n)\}$  в последовательные моменты времени  $t_1, t_2, \dots, t_n$ . Задача состоит в предвидении значения  $y(t_{n+1})$  в следующий момент времени  $t_{n+1}$ . Предвидение/прогноз имеют большое значение для принятия решений в бизнесе, науке и технике (предвидение цен на фондовой бирже, прогноз погоды).
5. **Оптимизация.** Многочисленные проблемы в математике, статистике, технике, науке, медицине и экономике могут рассматриваться как проблемы оптимизации. Задачей алгоритма оптимизации является нахождение такого решения, которое удовлетворяет системе ограничений и максимизирует или минимизирует целевую функцию.
6. **Память, адресуемая по содержанию.** В традиционных компьютерах обращение к памяти доступно только с помощью адреса, не зависящего от содержания памяти. Более того, если допущена ошибка в вычислении адреса, то может быть найдена совсем другая информация. Ассоциативная память или память, адресуемая по смыслу, доступна по указанию заданного содержания. Содержимое памяти может быть вызвано даже по частичному входу или поврежденному содержанию. Ассоциативная память может быть использована в мультимедийных информационных базах данных.
7. **Управление.** Рассмотрим динамическую систему, заданную совокупностью  $\{u(t), y(t)\}$ , где  $u(t)$  - входное управляющее воздействие, а  $y(t)$  - выход системы в момент времени  $t$ . В системах управления с эталонной моделью целью управления является расчет такого входного воздействия  $u(t)$ , при котором система действует по желательной траектории, заданной эталонной моделью. Примером является оптимальное управление двигателем.