Нейронные модели информационных процессов и систем

Нейронные сети — это раздел искусственного интеллекта, в котором для обработки сигналов используются явления, аналогичные происходящим в нейронах живых существ. Важнейшая особенность нейронной сети, свидетельствующая о ее широких возможностях и огромном потенциале, состоит в параллельной обработке информации всеми звеньями. При громадном количестве межнейронных связей это позволяет значительно ускорить процесс обработки информации. Во многих случаях становится возможным преобразование сигналов в реальном времени. Кроме того, при большом числе межнейронных соединений сеть приобретает устойчивость к ошибкам, возникающим на некоторых линиях. Функции поврежденных связей берут на себя исправные линии, в результате чего деятельность нейронной сети не претерпевает существенных возмущений.

Другое не менее важное свойство — способность к обучению и обобщению накопленных знаний в системе межнейронных связей. Натренированная на ограниченном множестве данных нейронная сеть способна обобщать полученную информацию и показывать хорошие результаты на данных, не использовавшихся ранее в процессе обучения (рис. 5.1).

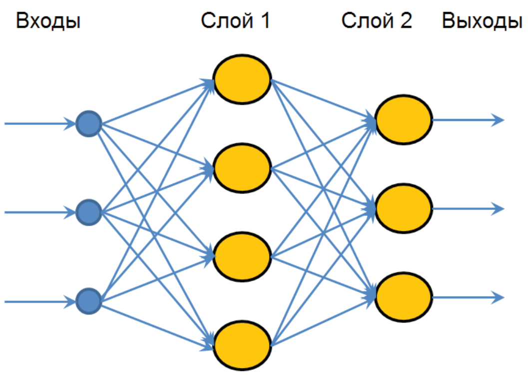

Рис. 5.1. Обобщенная структура нейронной сети

Поэтому наличие перечисленных свойств и повсеместное применение компьютерной техники вызвали в последние годы взрывной рост интереса к нейронным сетям и существенный прогресс в их исследовании. Создана база для выработки новых технологических решений, касающихся восприятия, искусственного распознания и обобщения видеоинформации, управления сложными системами, обработки ре-чевых сигналов и т. п.

Любая нейронная сеть используется в качестве самостоятельной системы представления знаний, которая в практических приложениях выступает, как правило, в качестве одного из компонентов системы управления либо модуля принятия решений, передающих результирующий сигнал на другие элементы, не связанные непосредственно с искусственной сетью. При этом выполняемые этой сетью (как структурным элементом) функции можно распределить на несколько основных групп: аппроксимация и интерполяция; распознавания и классификации образов; сжатия данных; прогнозирования; идентификации; управления; ассоциации.

Благодаря своим адаптационным свойствам в каждом из названных приложений нейронная сеть играет роль универсального аппроксиматора функции от нескольких переменных

$$y = f(x),$$

где x — это входной вектор, а y — реализация векторной функции нескольких переменных. Постановки значительного количества задач моделирования, идентификации и обработки сигналов могут быть сведены к подобному аппроксимационному представлению.

Большое число возможных приложений нейронных сетей в различ-ных областях науки и техники делает их актуальным объектом для ис-следования. Поэтому в рамках настоящей главы проводится краткий обзор литературы, различных нейросетевых архитектур и формиро-вание методического материала по использованию нейронных сетей, который может быть полезен для студентов специальностей, связан-ных с информационными технологиями.

5.1. Основные математические модели нейронов

Из сказанного выше следует, что каждый нейрон можно считать своеобразным процессором: он суммирует с соответствующими весами сигналы, приходящие от других нейронов, выполняет нелинейную (например, пороговую) решающую функцию и передает результирующее значение связанным с ним нейронам. В соответствии с действующим правилом «все или ничего» в простейших моделях нейронов выходной сигнал принимает двоичные значения 0 или 1. Значение 1 соответствует превышению порога возбуждения нейрона, а значение 0 — возбуждению ниже порогового уровня.

5.1.1. Модель МакКаллока-Питса

В соответствии с принципами функционирования биологических нейронов созданы различные математические модели, в которых с большей или меньшей степенью реализуются свойства природной нервной клетки. Обобщенная схема, составляющая основу большинства таких моделей, восходит к представленной на рис. 5.2 модели МакКаллока-Питса, которая была одной из первых моделей нейрона. В ней нейрон считается бинарным элементом, содержащим сумматор взвешенных входных сигналов и нелинейный блок обработки выходного сигнала сумматора. Структурная схема этой модели представлена на рис. 5.2.

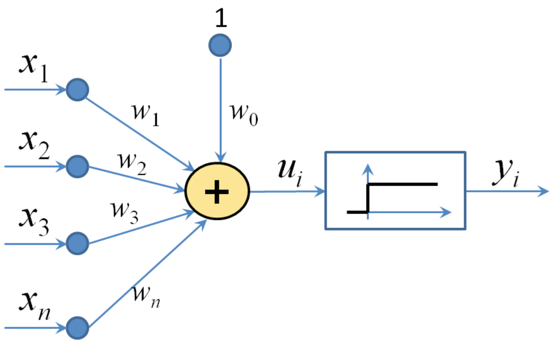

Рис. 5.2. Модель нейрона МакКаллока-Питса

Входные сигналы x_j, $j = 1, ..., N$ поступают на весовые умножители соответствующих весов w_{ij} (сигнал в направлении от входа суммируется с учетом i к узлу j) в сумматоре, после чего результат сравнивается с пороговым значением w_{i0}. Выходной сигнал нейрона y_i определяется при этом зависимостью

$$y_i = f\left(\sum_{j=1}^{N} w_{ij} x_j(t) + w_{i0}\right). \tag{5.1}$$

Аргументом функции выступает суммарный сигнал $u_i = \sum_{j=1}^{N} w_{ij} x_j(t) + w_{i0}$.

Функция $\varphi(u_i)$ называется *функцией активации*. В модели МакКаллока-Питса это *пороговая* функция вида

$$\varphi(u_i) = \begin{cases} 1, & u_i > 0; \\ 0, & u_i \le 0. \end{cases} \tag{5.2}$$

Коэффициенты w_{ij} представляют веса синаптических связей. Положительное значение w_{ij} соответствует возбуждающим синапсам, отрицательное значение w_{ij} — тормозящим синапсам, тогда как $w_{ij} = 0$ свидетельствует об отсутствии связи между i-м и j-м нейронами. Модель МакКаллока-Питса — это дискретная модель, в которой состояние нейрона в момент $(t + 1)$ рассчитывается по значениям его входных сигналов в предыдущий момент t. Построение дискретной модели обосновывается проявлением рефракции у биологических нейронов, приводящей к тому, что нейрон может изменять свое состояние с конечной частотой, причем длительность периодов бездействия зависит от частоты его срабатывания.

Через несколько лет Д. Хебб в процессе исследования ассоциативной памяти предложил теорию обучения (подбора весов w_{ij}) нейронов. При этом он использовал наблюдение, что веса межнейронных соединений при активации нейронов могут возрастать. В *нелинейной* модели Хебба приращение веса Δw_{ij} в процессе обучения пропорционально произведению выходных сигналов y_i и y_j нейронов, связанных весом w_{ij}:

$$w_{ij}(k+1) = w_{ij}(k) + \eta \cdot y_i(k) y_j(k), \tag{5.3}$$

где k означает номер цикла, η — коэффициент обучения. Согласно уравнению (5.3) в процессе обучения учитываются *корреляционные связи* между выходами нейронов.

Свойства нелинейной функции, особенно ее непрерывность, оказывают определяющее влияние на выбор способа обучения нейрона (подбор весовых коэффициентов). Другим важным фактором становится выбор стратегии обучения. Можно выделить два подхода: *обучение с учителем* и *обучение без учителя*.

При обучении с учителем предполагается, что помимо входных сигналов, составляющих вектор **X**, известны также и ожидаемые выходные сигналы нейрона d_i. Ключевым элементом процесса обучения с учителем является знание ожидаемых значений d_i выходного сигнала нейрона.

Если такой подход невозможен, остается выбрать стратегию обучения без учителя. Подбор весовых коэффициентов в этом случае проводится на основании либо конкуренции нейронов между собой (стратегии «По-бедитель получает все» или «Победитель получает больше»), либо с уче-том корреляции обучающих и выходных сигналов (обучение по Хеббу).Можно выделить три основных типа функций активации, графики зависимостей которых показаны на рис. 5.3.

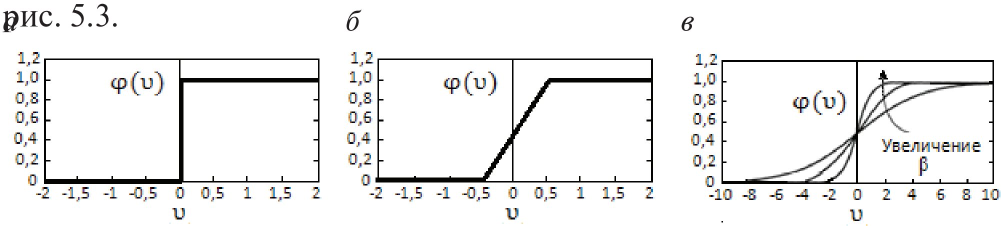

Рис. 5.3. Активационные функции: функция единичного скачка — *а*;
кусочно-линейная функция — *б*
и сигмоидальная функция для различных значений параметра β — *в*

1. *Функция единичного скачка* $\varphi(u_i)$ или пороговая функция, соответствующая на рис. 5.3, *а* формуле (5.2), где u_i — это индуцированное локальное поле нейрона.

2. *Кусочно-линейная функция*, показанная на рис. 5.3, *б*, описывается следующим выражением:

$$\varphi(u_i) = \begin{cases} 0, \quad u_i \le -1/2; \\ |u_i|, -1/2 < u_i < 1/2; \\ 1, \quad u_i \ge 1/2, \end{cases} \tag{5.4}$$

где коэффициент усиления в линейной области оператора предполагается равным единице. Эту функцию активации можно рассматривать как аппроксимацию нелинейного усилителя. Используются также два варианта особой формы кусочно-линейной функции:

- если линейная область оператора не достигает порога насыщения, он превращается в линейный сумматор;
- если коэффициент усиления в линейной области принять бесконечно большим, то кусочно-линейная функция вырождается в пороговую функцию.

3. *Сигмоидальная функция*, график которой напоминает букву *S*, является, пожалуй, самой распространенной функцией, используемой для создания искусственных нейронных сетей. Это быстро возрастающая функция, которая поддерживает баланс между линейным и нелинейным поведением. Примером сигмоидальной функции может служить логистическая функция, задаваемая следующим выражением:

$$\varphi(u_i) = \frac{1}{1 + \exp(-\beta u_i)}, \tag{5.5}$$

где β — параметр наклона сигмоидальной функции. Изменяя этот параметр, можно построить функции с различной крутизной (см. рис. 5.3, *в*). Первый график соответствует величине параметра, равной $\beta = 1/4$. В пределе, когда параметр наклона достигает бесконечности, сигмоидальная функция вырождается в пороговую. Если пороговая функция может принимать только значения 0 и 1, то сигмоидальная функция принимает бесконечное множество значений в диапазоне от 0 до 1. При этом следует заметить, что сигмоидальная функция является дифференцируемой, в то время как пороговая такой возможностью не обладает.

Область значений функций активации, определенных формулами (5.2), (5.4) и (5.5), представляет собой отрезок от 0 до +1. Однако иногда требуется функция активации, имеющая область значений от –1 до +1. В этом случае функция активации должна быть симметричной относительно начала координат, т. е. нечетной функцией индуцированного локального поля. В частности, пороговую функцию в данном случае можно определить следующим образом:

$$\varphi(u_i) = \begin{cases} -1, & u_i \le -0; \\ 0, & u_i = 0; \\ 1, & u_i \ge 0. \end{cases} \tag{5.6}$$

Эта функция обычно называется *сигнум* или *знак*.

Для сглаживания функции знака (5.6) часто используется функция гиперболического тангенса

$$\varphi(u_i) = \tanh(\beta u_i). \tag{5.7}$$

Важным свойством функции (5.7) является ее дифференцируемость.

5.1.2. Персептрон

Простой персептрон — это обычная модель МакКаллока-Питса с соответствующей стратегией обучения. Структурная схема и обозначения элементов j-го персептрона представлены на рис. 5.4. Весовые коэффициенты входов сумматора, на которые поступают входные сигналы x_i, обозначаются w_{ij}, а пороговое значение, поступающее с так называемого поляризатора, обозначается w_{i0}. Нелинейная функция активации персептрона представляет собой дискретную функцию ступенчатого типа (5.2), вследствие чего выходной сигнал нейрона может принимать только два значения 0 или 1:

$$u_i = \sum_{j=0}^{N} w_{ij} x_j(t), \tag{5.8}$$

где u_i обозначен выходной сигнал сумматора.

В приведенной формуле подразумевается, что имеющий длину N вектор $\mathbf{x} = [x_0, x_1, x_2, \dots, x_N]$ дополнен нулевым членом $x_0 = 1$, формирующим сигнал поляризации. Обучение персептрона требует наличия учителя и состоит в таком подборе весов w_{ij}, чтобы выходной сигнал y_i был наиболее близок к заданному значению d_i. Это обучение гетероассоциативного типа, при котором каждой обучающей выборке, представляемой вектором $\mathbf{x}$, априори поставлено в соответствие ожидаемое значение d_i, на выходе i-го нейрона.

Наиболее популярный метод обучения персептрона состоит в применении *правила персептрона*, в соответствии с которым подбор весов осуществляется по следующему алгоритму.

1. При первоначально выбранных (как правило, случайным образом) значениях весов w_{ij} на вход нейрона подается обучающий вектор $\mathbf{x}$ и рассчитывается значение выходного сигнала y_i. По результатам сравнения фактически полученного значения y_i с заданным значением d_i уточняются значения весов.

2. Если значение y_i совпадает с ожидаемым значением d_i, то весовые коэффициенты w_{ij} не изменяются.

3. Если $y_i = 0$, а соответствующее заданное значение $d_i = 1$, то значения весов уточняются в соответствии с формулой $w_{ij}(t+1) = w_{ij}(t) + x_j$, где t обозначает номер предыдущего цикла, а $(t+1)$ — номер текущего цикла.

4. Если $y_i = 1$, а соответствующее заданное значение $d_i = 0$, то значения весов уточняются в соответствии с формулой $w_{ij}(t+1) = w_{ij}(t) - x_j$.

По завершении уточнения весовых коэффициентов предъявляются персептрону очередной обучающий вектор **x** и связанное с ним ожидаемое значение d_i, и значения весов уточняются заново. Этот процесс многократно повторяется на всех обучающих выборках, пока не будут минимизированы различия между всеми значениями y_i и соответствующими им ожидаемыми значениями d_i [38].

5.1.3. Сигмоидальный нейрон

Нейрон сигмоидального типа имеет структуру, подобную модели МакКаллока-Питса (см. рис. 5.2), с той разницей, что функция активации является непрерывной и может быть выражена в виде сигмоидальной униполярной и/или биполярной функций. Униполярная функция, как правило, представляется формулой (5.5), тогда как биполярная функция задается в виде функции гиперболического тангенса (см. выражение (5.7)). В этих формулах параметр β подбирается пользователем. Его значение влияет на функцию активации.

Сигмоидальный нейрон, как правило, обучается с учителем по принципу минимизации целевой функции, которая для единичного обучающего набора <**x**, **d**> некоторого i-го нейрона определяется по формуле:

$$E = \frac{1}{2}(y_i - d_i)^2, \qquad (5.9)$$

где $y_i = \varphi(u_i) = \varphi\left(\sum_{j=0}^{N} w_{ij} x_j\right)$.

Применение непрерывной функции активации позволяет использовать при обучении градиентные методы. Проще всего реализовать метод *наискорейшего спуска*, в соответствии с которым уточнение вектора весов $\mathbf{w} = [w_{i0}, w_{i1}, w_{i2}, \ldots, w_{iN}]^{\mathrm{T}}$ проводится в направлении отрицательного градиента целевой функции:

$$\nabla E = \frac{dE}{dw_{ij}} = e_i x_j \frac{df(u_i)}{du_i}, \tag{5.10}$$

где $e_i = (y_i - d_i)$ означает разницу между фактическим и ожидаемым значением выходного сигнала нейрона [38].

5.1.4. Нейрон типа «адалайн»

Модель нейрона типа «адалайн» (англ.: *Adaptive Linear Neuron* — адаптивный линейный нейрон) была предложена Б. Уидроу [36]. Ее структурная схема, демонстрирующая адаптивный способ подбора весовых коэффициентов, изображена на рис. 5.4. По методу весового суммирования сигналов нейрон типа «адалайн» аналогичен представленным ранее моделям нейронов. Функция активации имеет тип «знак» (см. формулу (5.6)).

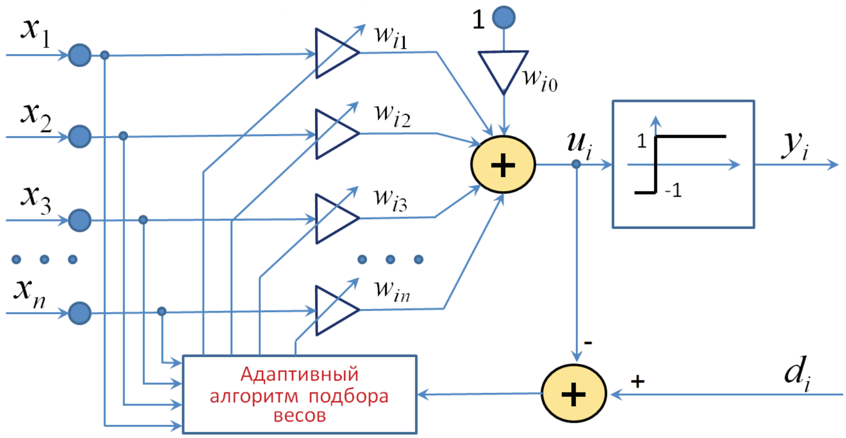

Рис. 5.4. Структурная схема нейрона типа «адалайн»

Адаптивный подбор весовых коэффициентов осуществляется в процессе минимизации квадратичной ошибки, определяемой как

$$E(\mathbf{w}) = \frac{1}{2} e_i^2 = \frac{1}{2}\left[d_i - \sum_{j=0}^{N} w_{ij} x_j\right]^2. \tag{5.11}$$

Следует обратить внимание на то, что, несмотря на нелинейный характер модели, в целевой функции присутствуют только линейные члены, представляющие собой сумму взвешенных входных сигналов. В связи с выполнением условия непрерывности целевой функции стало возможным применение алгоритма градиентного обучения. Как и в ситуации с сигмоидальным нейроном, в алгоритме Уидроу для минимизации целевой функции применяется метод наискорейшего спуска.

5.1.5. Нейроны типа WTA

Нейроны типа WTA (англ.: *Winner Takes All* — победитель получает все) имеют входной модуль в виде стандартного сумматора, рассчитывающего сумму входных сигналов с соответствующими весами w_{ij}. Выходной сигнал i-го сумматора определяется согласно формуле $u_i = \sum_{j=0}^{N} w_{ij} x_j$.

Группа конкурирующих между собой нейронов (рис. 5.5) получает одни и те же входные сигналы x_j. В зависимости от фактических значений весовых коэффициентов суммарные сигналы u_i отдельных нейронов могут различаться. По результатам сравнения этих сигналов победителем признается нейрон, значение u_i, у которого оказалось наибольшим. Нейрон-победитель переходит на своем выходе в состояние 1, а остальные (проигравшие) нейроны переходят в состояние 0.

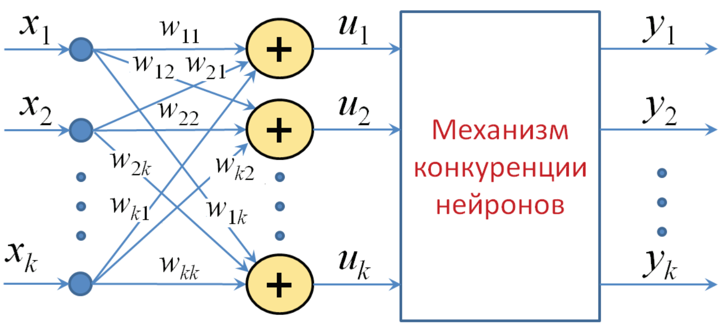

Рис. 5.5. Схема соединения нейронов типа WTA

Для обучения нейронов типа WTA не требуется учитель, оно протекает с использованием нормализованных входных векторов **x**. На начальном этапе случайным образом выбираются весовые коэффициенты каждого нейрона, нормализуемые относительно 1. После подачи первого вектора **x** определяется победитель этапа. Победивший в этом соревновании нейрон переходит в состояние 1, что позволяет ему провести уточнение весов его входных линий w_{ij}. Проигравшие нейроны формируют на своих выходах состояние 0, что блокирует процесс уточ-нения их весовых коэффициентов.

5.1.6. Стохастическая модель нейрона

В отличие от всех детерминированных моделей, определенных ранее, в стохастической модели выходное состояние нейрона зависит не только от взвешенной суммы входных сигналов, но и от некоторой случайной переменной, значения которой выбираются при каждой реализации из интервала (0,1).

В стохастической модели нейрона выходной сигнал y_i принимает значения ± 1 с вероятностью

$$\Pr(y_i = \pm 1) = 1 / \left[1 + \exp(\mp 2\beta u_i)\right], \qquad (5.12)$$

где u_i обозначена взвешенная сумма входных сигналов i-го нейрона, а β — положительная константа, чаще всего равная 1. Процесс обучения нейрона в стохастической модели состоит из следующих этапов [38].

1. Расчет взвешенной суммы $u_i = \sum_{j=0}^{N} w_{ij} x_j$ для каждого i-го нейрона сети.

2. Расчет вероятности того, что y_i принимает значение ± 1 в соответствии с формулой (5.12).

3. Генерация значения случайной переменной $R \in (0,1)$ и формирование выходного сигнала $y_i = \pm 1$, если $R < \Pr(y_i = \pm 1)$ или $y_i = \mp 1$ в противном случае.

4. Определенный таким образом процесс осуществляется на случайно выбранной группе нейронов, вследствие чего их состояние модифицируется в соответствии с предложенным правилом.

5. После фиксации состояния отобранных нейронов их весовые коэффициенты модифицируются по применяемому правилу уточнения весов. Например, при обучении с учителем по правилу Уидроу-Хоффа адаптация весов проводится по формуле

$$\Delta w_{ij} = \eta(y_i - d_i). \qquad (5.13)$$

5.1.7. Сравнение моделей абстрактных и биологических нейронов

Биологический нейрон — сложная система, математическая модель которого до сих пор полностью не построена. Введено множество моделей, различающихся вычислительной сложностью и сходством с реальным нейроном, но все они имеют ряд ограничений по сравнению с биологическими нейронами.

Вычисления выхода абстрактных нейронов предполагаются мгновенными, не вносящими задержки. Непосредственно моделировать динамические системы, имеющие «внутреннее состояние», с помощью таких нейронов нельзя. В модели отсутствуют нервные импульсы, нет модуляции уровня сигнала плотностью импульсов, как в нервной системе, не появляются эффекты синхронизации, когда скопления нейронов обрабатывают информацию синхронно, под управлением периодических волн возбуждения и торможения.

В абстрактных нейронах нет четких алгоритмов для выбора функции активации, отсутствуют механизмы, регулирующие работу сети в целом, например, гормональная регуляция активности в биологических нервных сетях.

Большое внимание в формальных моделях уделено понятиям «порог» и «весовые коэффициенты». В реальном нейроне нет числового порога, он динамически меняется в зависимости от активности самого нейрона и общего состояния сети. Весовые коэффициенты биологических синапсов тоже не постоянны и обладают пластичностью и стабильностью, весовые коэффициенты настраиваются в зависимости от сигналов, проходящих через синапс. Существует большое разнообразие биологических синапсов. Они встречаются в различных частях клетки и выполняют различные функции. Тормозные и возбуждающие синапсы реализуются в формальной модели в виде весовых коэффициентов противоположного знака, но разнообразие синапсов этим не ограничивается. Дендро-дендритные, аксо-аксональные синапсы не реализуются в модели формального нейрона.

В абстрактных моделях не прослеживается различие между градуальными потенциалами и нервными импульсами. Любой сигнал представляется в виде одного числа.

Итак, модель формального нейрона не является биоподобной и скорее похожа на математическую абстракцию, чем на живой нейрон. Тем удивительнее оказывается многообразие задач, решаемых с помощью таких нейронов, и универсальность получаемых алгоритмов.

5.2. Модели нейронных сетей

Существенную часть в теории нейронных сетей занимают биофизические проблемы. Для построения адекватной математической модели сети необходимо детально изучить работу биологических нервных клеток и их взаимодействующих систем с точки зрения химии, физики, теории информации и синергетики. Должны быть известны ответы на следующие основные вопросы.

1. Как работает нервная клетка — биологический нейрон? Необходимо иметь математическую модель, адекватно описывающую информационные процессы в нейроне. Какие свойства нейрона важны при моделировании, а какие — нет?

2. Как передается информация через соединения между нейронами — синапсы? Как меняется проводимость синапса в зависимости от проходящих по нему сигналов?

3. По каким законам нейроны связаны друг с другом в сеть? Откуда нервная клетка знает, с какими соседями должно быть установлено соединение?

4. Как биологические нейронные сети обучаются решать задачи? Как выбираются параметры сети, чтобы давать правильные выходные сигналы? Какой выходной сигнал считается «правильным», а какой — ошибочным?

При любом объединении объектов, имеющих различные свойства, возникают системные эффекты. Тем более они появляются при соединении нейронов как формальных, так и биологических. Отметим важнейшие из свойств биологических нейросетей.

1. Параллельность обработки информации. Каждый нейрон формирует свой выход только на основе своих входов и собственного внутреннего состояния под воздействием общих механизмов регуляции нервной системы.

2. Способность к полной обработке информации. Все известные человеку задачи решаются нейронными сетями. К этой группе свойств относятся ассоциативность (сеть может восстанавливать полный образ по его части), способность к классификации, обоб-

щению, абстрагированию и множество других. Они до конца не систематизированы.

3. Самоорганизация. В процессе работы биологические нейросети самостоятельно, под воздействием внешней среды, обучаются решению разнообразных задач. Неизвестно никаких принципиальных ограничений на сложность задач, решаемых биологическими нейронными сетями. Нервная система сама формирует алгоритмы своей деятельности, уточняя и усложняя их в течение жизни. Человек пока не сумел создать систем, обладающих самоорганизацией и самоусложнением. Это свойство нейросетей рождает множество вопросов. Ведь каждая замкнутая система в процессе развития упрощается, деградирует. Следовательно, подвод энергии к нейронной сети имеет принципиальное значение. Почему же среди всех диссипативных (рассеивающих энергию) нелинейных динамических систем только у живых существ и, в частности, биологических нейросетей проявляется способность к усложнению? Какое принципиальное условие упущено человеком в попытках создать самоусложняющиеся системы?

4. Биологические нейросети являются аналоговыми системами. Информация поступает в сеть по большому количеству каналов и кодируется по пространственному принципу: вид информации определяется номером нервного волокна, по которому она передается. Амплитуда входного воздействия кодируется плотностью нервных импульсов, передаваемых по волокну.

5. Надежность. Биологические нейросети обладают фантастической надежностью: выход из строя даже 10 % нейронов в нервной системе не прерывает ее работы. И это по сравнению с последовательными ЭВМ, основанными на принципах фон Неймана, где сбой одной ячейки памяти или одного узла в аппаратуре приводит к краху системы.

Современные искусственные нейросети по сложности и интеллекту приближаются к нервной системе таракана, но уже сейчас демонстрируют ценные свойства:

1. Обучаемость. Выбрав одну из моделей нейросетей, создав сеть и выполнив алгоритм обучения, можно обучить такую сеть решению задачи, которая ей по силам. Нет никаких гарантий, что это удастся сделать при выбранных сети, алгоритме и задаче, но если все сделано правильно, то обучение бывает успешным.

2. Способность к обобщению. После обучения сеть становится нечувствительной к малым изменениям входных сигналов (шуму

или вариациям входных образов) и дает правильный результат на выходе.

3. Способность к абстрагированию. Если предъявить сети несколько искаженных вариантов входного образа, то сеть сама может создать на выходе идеальный образ, с которым она никогда не встречалась.

5.2.1. Виды нейронных сетей

В настоящее время кроме многослойного персептрона существует множество способов задания нейроподобных структур. Все виды нейронных сетей можно условно разделить на сети прямого распространения и сети с обратными связями. Как следует из названия, в сетях первого типа сигналы от нейрона к нейрону распространяются в четко заданном направлении — от входов сети к ее выходам. В сетях второго типа выходные значения любого нейрона сети могут передаваться к его же входам. Это позволяет нейронной сети моделировать более сложные процессы, например временные, но делает выходы подобной сети нестабильными, зависящими от состояния сети на предыдущем цикле. На рис. 5.6 представлены наиболее распространенные типы нейронных сетей. Разнообразие нейронных сетей увеличивается еще больше благодаря огромному количеству алгоритмов и методик обучения, а также наличию нескольких видов пороговых функций.

В учебном пособии будет рассмотрен самый распространенный класс сетей — сети прямого распространения. Это — персептрон и многослойный персептрон.

5.2.2. Многослойный персептрон

Формальные нейроны могут объединяться в сети различным образом. Самым распространенным видом сети стал *многослойный персептрон*, изображенный на рис. 5.7. Сеть состоит из произвольного количества слоев нейронов. Нейроны каждого слоя соединяются с нейронами предыдущего и последующего слоев по принципу «каж-дый с каждым». Первый слой (слева) называется *сенсорным* или *вход-ным*, внутренние слои называются *скрытыми* или *ассоциативными*, последний (самый правый на рисунке, состоит из одного нейрона) —

выходным или *результативным*. Количество нейронов в слоях может быть произвольным. Обычно во всех скрытых слоях одинаковое количество нейронов.

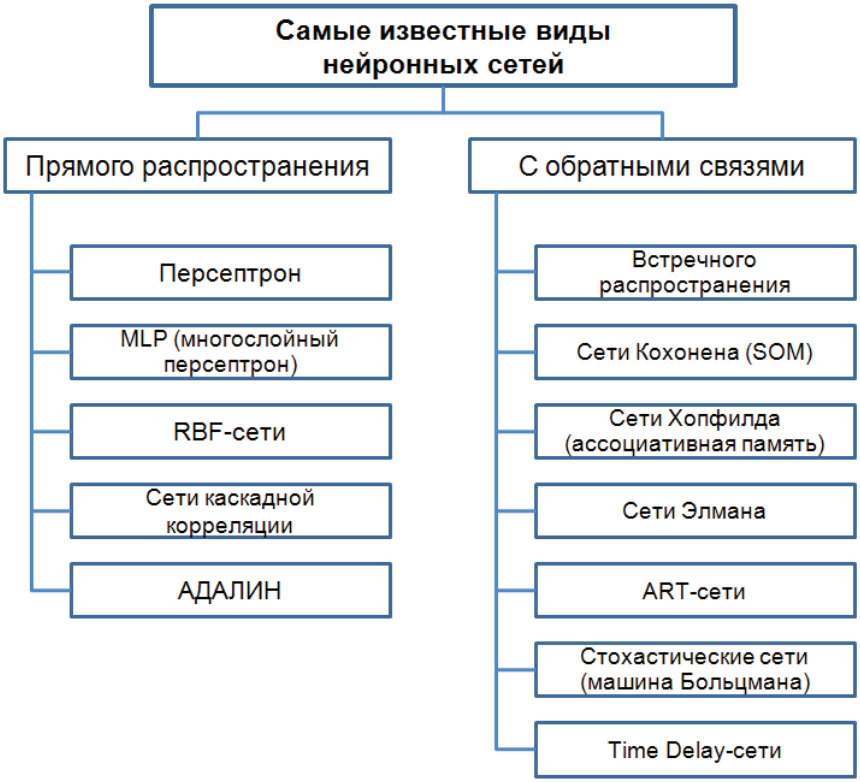

Рис. 5.6. Основные виды нейронных сетей

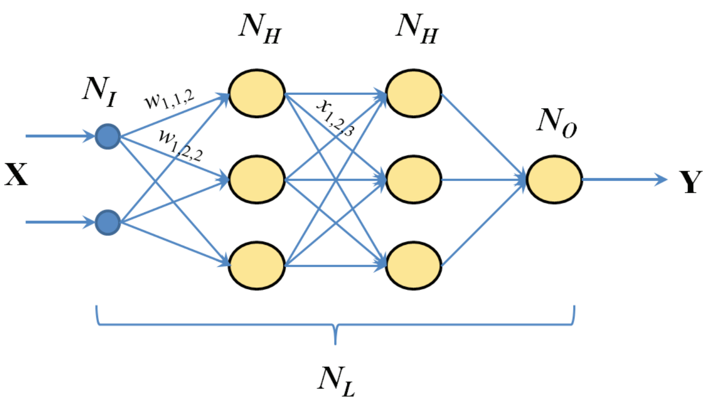

Рис. 5.7. Многослойный персептрон

Обозначим количество слоев и нейронов в слое. Входной слой имеет N_I нейронов, в каждом скрытом слое N_H нейронов и N_O выходных нейронов. Пусть, как и ранее, $\mathbf{x}$ — вектор входных сигналов сети, $\mathbf{y}$ — вектор выходных сигналов.

Существует путаница с подсчетом количества слоев в сети. Входной слой не выполняет никаких вычислений, а лишь распределяет входные сигналы, поэтому иногда его считают, иногда — нет. Обозначим через N_L полное количество слоев в сети, считая входной.

Работа многослойного персептрона (МСП) описывается формулами:

$$NET_{jl} = \sum_i w_{ijl} x_{ijl}; \tag{5.14}$$

$$OUT_{jl} = F(NET_{jl} - \theta_{jl}); \tag{5.15}$$

$$x_{ij(l+1)} = OUT_{il}, \tag{5.16}$$

где индексом i всегда будем обозначать номер входа, j — номер нейрона в слое, l — номер слоя, x_{ijl} — i-й входной сигнал j-го нейрона в слое l; w_{ijl} — весовой коэффициент i-го входа нейрона номер j в слое l; NET_{jl} — сигнал сети j-го нейрона в слое l; OUT_{jl} — выходной сигнал нейрона; θ_{jl} — пороговый уровень нейрона j в слое l. Введем также обозначения: $\mathbf{w}_{il}$ — вектор-столбец весов для всех входов нейрона j в слое l; W_l — матрица весов всех нейронов в слое l. В столбцах матрицы расположены векторы $\mathbf{w}_{il}$. Аналогично $\mathbf{x}_{jl}$ — входной вектор-столбец слоя l.

Каждый слой рассчитывает нелинейное преобразование от линейной комбинации сигналов предыдущего слоя. Отсюда видно, что линейная функция активации может применяться только для тех моделей сетей, где не требуется последовательное соединение слоев нейронов друг за другом. Для многослойных сетей функция активации должна быть нелинейной, иначе можно построить эквивалентную однослойную сеть, и многослойность оказывается ненужной. Если применена линейная функция активации, то каждый слой будет давать на выходе линейную комбинацию входов. Следующий слой даст линейную комбинацию выходов предыдущего, а это эквивалентно одной линейной комбинации с другими коэффициентами и может быть реализовано в виде одного слоя нейронов.

В многослойном персептроне нет обратных связей. Такие модели называются *сетями прямого распространения*. Они не обладают внутренним состоянием и не позволяют без дополнительных приемов моделировать развитие динамических систем.

Многослойная сеть может формировать на выходе произвольную многомерную функцию при соответствующем выборе количества слоев, диапазона изменения сигналов и параметров нейронов. Как и ряды, многослойные сети оказываются универсальным инструментом аппроксимации функций, однако наблюдается отличие работы нейронной сети

$$f(x) = F\left(\begin{array}{c} F\left(\underbrace{\sum_{i_2} w_{i_2 j_2 2} \underbrace{F\left(\sum_{i_1} w_{i_1 j_1 1} - \theta_{j_1 1}\right)}_{\text{слой } 1}}_{\text{слой } 2} - \theta_{j_2 2} \right) \\ \ldots\ldots\ldots \\ \underbrace{\sum_{iN} w_{i_N j_N N}}_{\text{слой } N} - \theta_{j_N N} \end{array} \right) \tag{5.17}$$

от разложения функции в ряд

$$f(x) = \sum_i g_i(x). \tag{5.18}$$

За счет поочередного расчета линейных комбинаций и нелинейных преобразований достигается аппроксимация произвольной многомерной функции при соответствующем выборе параметров сети.

5.2.3. Решение информационных задач с помощью МСП

Чтобы построить МСП, необходимо выбрать его параметры. Чаще всего выбор значений весов и порогов требует *обучения*, т. е. пошаговых изменений весовых коэффициентов и пороговых уровней.

Общий алгоритм решения состоит из следующих шагов.

1. Определить, какой смысл вкладывается в компоненты входного вектора **x**. Входной вектор должен содержать формализованное условие задачи, т. е. всю информацию, необходимую для получения ответа.

2. Выбрать выходной вектор **y** таким образом, чтобы его компоненты содержали полный ответ поставленной задачи.

3. Выбрать вид нелинейности в нейронах (функцию активации $\varphi(u)$). При этом желательно учесть специфику задачи, т. к. удачный выбор $\varphi(u)$ сократит время обучения.

4. Выбрать число слоев и нейронов в слое.

5. Задать диапазон изменения входов, выходов, весов и пороговых уровней, учитывая множество значений выбранной функции активации.

6. Присвоить начальные значения весовым коэффициентам и пороговым уровням и дополнительным параметрам (например, крутизне β функции активации φ (β*u*), если она будет настраиваться при обучении). Начальные значения не должны быть большими, чтобы нейроны не оказались в насыщении (на горизонтальном участке функции активации), иначе обучение будет очень медленным. Начальные значения не должны быть и слишком малыми, чтобы выходы большей части нейронов не были равны нулю, иначе обучение также замедлится.

7. Провести обучение, т. е. подобрать параметры сети так, чтобы задача решалась наилучшим образом. По окончании обучения сеть готова решить задачи того типа, которым она обучена.

8. Подать на вход сети условия задачи в виде вектора **x**. Рассчитать выходной вектор **y**, который и даст формализованное решение задачи.

Многослойный персептрон может рассчитывать выходной вектор **y** для любого входного вектора **x**, т. е. давать значение некоторой векторной функции $\mathbf{y} = \mathbf{f}(\mathbf{x})$. Следовательно, условием любой задачи, которая может быть поставлена персептрону, должно являться множество векторов $\{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^s\}$ с N_{in} компонентами каждый: $\mathbf{x}^s = [x_1^s, x_2^s, x_3^s, ..., x_{N_{in}}^s]^{\mathrm{T}}$. Решением задачи будет множество векторов $\{\mathbf{y}^1, \mathbf{y}^2, ..., \mathbf{y}^R\}$, каждый вектор $\mathbf{y}^r$ с N_{out} компонентами $\mathbf{y}^r = \mathbf{f}(\mathbf{x}^s)$, где $s = 1, 2, ..., S$ — номер предъявленного образа, а $r = 1, 2, ..., R$ — номер полученного после обработки образа.

Все, что способен сделать персептрон, — это сформировать отображение $X \to Y$ для $\forall x \in X$. Данное отображение полностью «извлечь» из персептрона невозможно, можно только получить отображение произвольного количества точек $\begin{pmatrix} x^1 & \to & y^1 \\ x^2 & \to & y^5 \\ x^3 & \to & y^R \\ & \cdots & \\ x^S & \to & y^r \end{pmatrix}$, где множество векторов $\{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^s\}$ — *формализованное условие задачи*, а множество

$\{\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^R\}$ — *формализованное решение*. Задача формализации, т. е. выбор смысла, которым наделяются компоненты входного и выходного векторов, пока решается только человеком на основе практического опыта. Жестких рецептов формализации для нейронных сетей пока не создано.