

УДК 621.391; 519.21; 519.245

© А. В. Меркушева

ПРИМЕНЕНИЕ НЕЙРОННОЙ СЕТИ ДЛЯ ТЕКУЩЕГО АНАЛИЗА НЕСТАЦИОНАРНОГО СИГНАЛА (РЕЧИ), ПРЕДСТАВЛЕННОГО ЕГО ВЕЙВЛЕТ-ОТОБРАЖЕНИЕМ. II. ИССЛЕДОВАНИЕ И ОПТИМИЗАЦИЯ НЕЙРОННОЙ СЕТИ

В статье обоснован выбор и уровень сложности нейросетевого многослойного персептрона и получены характеристики его обучения. Предложен способ снижения размерности входных векторов при обучении персептрона и распознавании состояния сигнала, основанный на методе главных компонент. Получены основные характеристики нейросетевого алгоритма.

ИССЛЕДОВАНИЕ ХАРАКТЕРИСТИК МЕТОДОВ ОБУЧЕНИЯ ПЕРСЕПТРОНА

Для алгоритма обучения персептрона задаче детектирования речевой активности важно иметь быструю сходимость и малые вероятности ошибочных решений на этапе распознавания. Поэтому целесообразно привести характеристики алгоритмов обучения персептрона, использованных для задачи распознавания типа сегментов речь/пауза. Это:

- метод сопряженных градиентов;
- простое и обобщенное правила дельта—дельта;
- метод случайного поиска;
- метод Бройдена—Флетчера;
- метод параллельных касательных (метод партан).

Обучение персептрона с настройкой его параметров (весов синаптических связей) основано на алгоритме с обратным распространением ошибки (алгоритм ОРО) [1–7]. Алгоритм использует понятия:

- ошибки $e_j(n)$ на выходе нейрона j на итерации n : $e_j(n) = d_j(n) - y_j(n)$, где d_j — желаемый отклик, y_j — выход нейрона j , являющегося выходным узлом сети;
- текущего значения суммы квадратов ошибок:

$E(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n)$, где C — множество выходных нейронов сети.

Цель обучения состоит в подстройке свободных параметров сети, обеспечивающей минимум

средней ошибки: $E_{cp} = \frac{1}{N} \sum_{n=1}^N E(n)$, где N — количество обучающих векторов.

Алгоритм ОРО реализует обучение итеративно с двумя проходами по сети — прямым и обратным, при которых соответственно вычисляются функциональные преобразования вектора входного сигнала $\mathbf{x} = (x_1, \dots, x_p)^T$ и корректировка вектора параметров \mathbf{w} . При обратном проходе на каждой итерации n определяется локальный градиент $\delta_j(n)$, для чего служит выражение (1), если нейрон находится в выходном слое, и выражение (2), если нейрон в скрытом слое:

$$\delta_j(n) = e_j(n) \varphi'(v_j(n)), \quad (1)$$

$$\delta_j(n) = \varphi'(v_j(n)) \sum_k \delta_k(n) w_{jk}(n), \quad (2)$$

где $v_j(n)$ — возбуждение j -го нейрона, $y_j(n)$ — выход j -го нейрона, $e_j(n)$ — ошибка на выходе j -го нейрона, $\varphi'(v_j(n)) = y_j(n)[1 - y_j(n)]$ — производная функции нелинейности, w_{jk} — синаптические веса. Локальный градиент ошибки $\delta_j(n)$ используется для корректировки весов в соответствии с δ -правилом по (3) или в соответствии с модифицированным δ -правилом по выражению (4):

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n), \quad (3)$$

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) y_i(n), \quad (4)$$

где α и η — параметры, определяющие скорость обучения.

Непосредственное применение метода обратного распространения при реализации алгоритма обучения предполагает задание свободных параметров α и η .

Более продуктивный подход к обучению основан на нелинейной минимизации функции ошибок, или оптимизации нелинейной функции отображения вход—выход [8–11].

Метод сопряженных градиентов

В этом методе [12] коррекция вектора весов персептрона $\mathbf{w}(n)$ осуществляется не вдоль вектора градиента ошибки $\mathbf{g}(n)$, а вдоль вектора сопряженного градиента $\mathbf{p}(n)$, который обновляется на каждом шаге итерации:

$$\begin{aligned} \mathbf{w}(n+1) &= \mathbf{w}(n) + \eta(n)\mathbf{p}(n), \\ \mathbf{p}(n+1) &= -\mathbf{g}(n+1) + \beta(n)\mathbf{p}(n), \end{aligned} \quad (5)$$

где $\mathbf{g}(n)$ — вектор градиента функции ошибки $E(\mathbf{w}_n)$. Параметр $\beta(n)$ определяется по (6) для метода Флетчера—Ривса и по (7) — для метода Полака—Рибера [9]:

$$\beta(n) = \frac{\mathbf{g}^T(n+1) \cdot \mathbf{g}(n+1)}{\mathbf{g}^T(n) \cdot \mathbf{g}(n)}, \quad (6)$$

$$\beta(n) = \frac{\mathbf{g}^T(n+1) \cdot [\mathbf{g}(n+1) - \mathbf{g}(n)]}{\mathbf{g}^T(n) \cdot \mathbf{g}(n)}. \quad (7)$$

Оценка параметра $\eta(n)$ включает линейный поиск, состоящий в определении η , для которого критерий $E_{cp}[\mathbf{w}(n) + \eta \cdot \mathbf{p}(n)]$ имеет минимальную величину при данных значениях $\mathbf{w}(n)$ и $\mathbf{p}(n)$:

$$\eta(n) = \arg \left(\min_{\eta} \{E_{cp}[\mathbf{w}(n) + \eta \cdot \mathbf{p}(n)]\} \right). \quad (8)$$

Таким образом, при непосредственном использовании алгоритма ОРО параметр скорости обучения η определяется на основе оптимизации критерия минимума функции ошибок.

Метод дельта—дельта и обобщенное правило дельта—дельта

В качестве более совершенного метода выбора $\eta(n)$ применена форма адаптации параметра скорости обучения на основе правил Джекобса: правила дельта—дельта адаптации и обобщенного правила дельта—дельта [13]. Оба правила основаны на принципе быстрого спуска по поверхности функции ошибок, в качестве которой служит суммарная квадратичная ошибка персептрона $E(n)$ на текущем шаге процесса обучения (9), которая в этом методе рассматривается в пространстве $\{\eta_i(n)\}_{i=1,2,\dots}$ параметров скорости обучения для синаптических весов $w_{ji}(n)$ нейронной сети:

$$E(n) = \frac{1}{2} \sum_j e_j^2(n) = \frac{1}{2} \sum_j [d_j(n) - y_j(n)]^2. \quad (9)$$

Определение градиента $\frac{\partial E}{\partial \eta_i(n)}$ основано на

правиле (10) дифференцирования сложной функции и общих соотношениях (11), (12):

$$\frac{\partial E}{\partial \eta_i(n)} = \frac{\partial E(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial \eta_i(n)}, \quad (10)$$

$$v_j(n) = \sum_i w_{ji}(n) y_i(n), \quad (11)$$

$$w_{ji}(n) = w_{ji}(n-1) - \eta_i(n) \frac{\partial E(n-1)}{\partial w_{ji}(n-1)}. \quad (12)$$

Соотношение (11) выражает возбуждение j -нейрона выходами y_i нейронов предшествующего слоя, а (12) — правило градиента для изменения параметров персептрона $w_{ji}(n)$. Соотношения (11) и (12) позволяют выразить $v_j(n)$ в виде:

$$v_j(n) = \sum_i y_i(n) \left[w_{ji}(n-1) - \eta_i(n) \frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \right]. \quad (13)$$

Это приводит к определению компонент (14), (15) и (16), необходимых для получения формы выражения (10), пригодной для вычислительного алгоритма:

$$\frac{\partial v_j(n)}{\partial \eta_i(n)} = -y_i(n) \frac{\partial E(n-1)}{\partial w_{ji}(n-1)}, \quad (14)$$

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \phi'_j(v_j(n)), \quad (15)$$

$$\frac{\partial E(n)}{\partial y_j(n)} = -[d_j(n) - y_j(n)] = -e_j(n). \quad (16)$$

Полученная таким образом окончательная форма выражения $\frac{\partial E}{\partial \eta_i(n)}$ имеет вид:

$$\begin{aligned} \frac{\partial E(n)}{\partial \eta_i(n)} &= \\ &= -\phi'_j(v_j(n-1)) e_j(n) y_i(n) \left[-\frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \right]. \end{aligned} \quad (17)$$

Если учесть, что для градиента функции ошибки относительно весов справедливо соотношение:

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -\phi'_j(v_j(n)) e_j(n) y_i(n), \quad (18)$$

то получается конечная форма градиента в пространстве параметров скорости обучения:

$$\frac{\partial E}{\partial \eta_i(n)} = -\frac{\partial E(n)}{\partial w_{ji}(n)} \cdot \frac{\partial E(n-1)}{\partial w_{ji}(n-1)}. \quad (19)$$

Таким образом, значение параметра скорости обучения η , обеспечивающее наискорейший спуск к минимуму поверхности ошибок в пространстве параметров $\{\eta_i(n)\}$, адаптивно обновляется по алгоритму в форме соотношения:

$$\begin{aligned} \Delta \eta_i(n+1) &= \\ &= -\gamma \frac{\partial E(n)}{\partial \eta_i(n)} = \gamma \frac{\partial E(n)}{\partial w_{ji}(n)} \frac{\partial E(n-1)}{\partial w_{ji}(n-1)}, \end{aligned} \quad (20)$$

где γ — константа, контролирующая шаг адаптации параметра скорости обучения η .

Значения $\frac{\partial E(n)}{\partial w_{ji}(n)}$ и $\frac{\partial E(n-1)}{\partial w_{ji}(n-1)}$ (компоненты градиента функции ошибок на шаге n и $(n-1)$ в итеративной процедуре обучения) служат основой следующих правил, сформулированных Джекобсом.

— Если компонента градиента $\frac{\partial E(n)}{\partial w_{ji}(n)}$ имеет одинаковый знак на двух итерациях, то $\Delta \eta_i(n+1) > 0$, т. е. адаптация увеличивает скорость обучения для веса w_{ji} .

— Если у градиента меняется знак на соседних итерациях, $\Delta \eta_i(n+1) < 0$, то процедура адаптации уменьшает параметр скорости обучения для веса w_{ji} .

Приведенная форма алгоритма на основе правила дельта—дельта проста в вычислениях, но может дать неуверенную адаптацию при очень малых и больших значениях градиента. Лучшие результаты дает более сложная модификация алгоритма — обобщенное правило дельта—дельта [13], которое требует большего объема вычислений. Адаптация параметра $\eta_i(n)$ для синаптической связи w_{ji} определяется выражениями (21)–(23):

$$\Delta \eta_{ji}(n+1) = \begin{cases} k & \text{при } S_{ji}(n-1)D_{ji}(n) > 0, \\ -\beta \eta_{ji}(n) & \text{при } S_{ji}(n-1)D_{ji}(n) < 0, \\ 0 & \text{иначе,} \end{cases} \quad (21)$$

где $D_{ji}(n)$ и $S_{ji}(n)$ определяются по (22) и (23):

$$D_{ji}(n) = \frac{\partial E(n)}{\partial w_{ji}}, \quad (22)$$

$$S_{ji}(n) = (1-\xi)D_{ji}(n-1) + \xi \cdot S_{ji}(n-1). \quad (23)$$

В выражениях (22) и (23) ξ — положительная константа; $D_{ji}(n)$ — текущее значение градиента; $S_{ji}(n)$ — определяется соотношением типа авто-регрессии, которое использует параметр веса ξ и компоненту градиента $D_{ji}(n-1)$ в качестве базы. Процедура адаптации параметра скорости обучения (21)–(23) составляет основу метода обучения по обобщенному правилу дельта—дельта. Адаптация основана на контроле совпадения или различия знаков на последовательных итерациях у соответствующих компонент $S_{ji}(n-1)$ и градиента $D_{ji}(n)$. При совпадении знаков $\Delta \eta_{ji}(n)$ составляет k , а при несовпадении уменьшается экспоненциально $\Delta \eta_{ji}(n+1) = -\beta \eta_{ji}(n)$.

Метод случайного поиска

Метод случайного поиска основан на случайном выборе вектора приращения весов по принципу Монте-Карло. Выбор вектора приращения весов Δw_{ji} реализуется поиском в пространстве весов, при этом принимаются только те изменения Δw_{ji} , которые ведут к уменьшению квадратичной ошибки на каждой итерации. Метод достаточно прост по вычислениям, но недостаточно эффективен и медленно сходится.

Метод Бройдена—Флетчера

Метод нелинейной минимизации Бройдена—Флетчера основан на методе Ньютона [8], в котором используется разложение функции ошибок, включающее вторые производные $\frac{\partial^2 E(\mathbf{w})}{\partial w_i \partial w_j}$, т. е. матрицу Гессе $H = \left[H_{ij} = \frac{\partial^2 E(\mathbf{w})}{\partial w_i \partial w_j} \right]$:

$$\begin{aligned} E(\mathbf{w}(n) + \Delta \mathbf{w}(n)) &\cong \\ &\cong E(n) + \mathbf{g}^T(n) \cdot \Delta \mathbf{w}(n) + \frac{1}{2} \Delta \mathbf{w}^T(n) \cdot H^T, \end{aligned} \quad (24)$$

где $E(n) \equiv E(\mathbf{w}(n))$; $\{\mathbf{g}(n)\}_i \equiv \frac{\partial E(n)}{\partial w_i(n)}$;

$\{H(n)\}_{ij} = \frac{\partial^2 E(\mathbf{w}(n))}{\partial w_i(n) \partial w_j(n)}$; $\mathbf{g}(n)$ — градиент ошибки.

Изменение весов $\Delta \mathbf{w}(n)$ вектора параметров персептрона $\mathbf{w}(n)$ на n -й итерации определяется условием

$$\Delta \mathbf{w}(n) = \arg \min_{\Delta \mathbf{w}} \{E(\mathbf{w}(n) + \Delta \mathbf{w}(n))\}. \quad (25)$$

Соотношение (25) с учетом (24) эквивалентно выражению (26), которое лежит в основе получе-

ния $\Delta \mathbf{w}(n)$ по методу Ньютона:

$$H(n) \cdot \Delta \mathbf{w}(n) = -\mathbf{g}(n). \quad (26)$$

При решении матричного уравнения (26) относительно $\Delta \mathbf{w}(n)$ необходимо, чтобы матрица Гессе была положительно определенной и хорошо обусловленной. Но эти условия не всегда выполняются на практике.

Модификация Бroyдена—Флетчера для метода Ньютона связана с построением положительно определенной матрицы $\tilde{H}(n)$, которая является аппроксимацией матрицы Гессе на основе конечных разностей градиента функции ошибок [14–18]. Матрицей $\tilde{H}(n)$ представляется разложение (27) градиента рядом Тэйлора, которое в более короткой форме эквивалентно (28):

$$\mathbf{g}(\mathbf{w}(n) + \Delta \mathbf{w}(n)) \cong \mathbf{g}(\mathbf{w}(n)) + \tilde{H}(n) \cdot \Delta \mathbf{w}(n), \quad (27)$$

$$\tilde{H}(n) \cdot \Delta \mathbf{w}(n) = \mathbf{y}(n), \quad (28)$$

где через $\mathbf{y}(n) \equiv \mathbf{g}(n+1) - \mathbf{g}(n)$ обозначено приращение градиента на n -й итерации.

В методе Бroyдена—Флетчера аппроксимация матрицы Гессе в алгоритме обучения вычисляется итеративно с помощью соотношения (29):

$$\begin{aligned} \tilde{H}(n+1) = & \tilde{H}(n) - \\ & - \frac{1}{\Delta \mathbf{w}^T(n) \cdot \tilde{H}(n) \cdot \Delta \mathbf{w}(n)} \tilde{H}(n) \cdot \Delta \mathbf{w}(n) \cdot \Delta \mathbf{w}^T(n) + \\ & + \frac{1}{\mathbf{y}^T(n) \cdot \Delta \mathbf{w}(n)} \mathbf{y}(n) \cdot \mathbf{y}^T(n). \end{aligned} \quad (29)$$

При этом вектор $\mathbf{p}(n)$ направления изменения параметров персептрона $\Delta \mathbf{w}(n)$ определяется соотношением (30), которое по форме аналогично выражению (26) в методе Ньютона [8]:

$$\tilde{H}(n) \cdot \mathbf{p}(n) = -\mathbf{g}(n). \quad (30)$$

Изменение весов персептрона $\Delta \mathbf{w}(n)$ вдоль этого направления с параметром скорости обучения η определяется соотношением (31):

$$\Delta \mathbf{w}(n) = \eta \cdot \mathbf{p}(n). \quad (31)$$

Таким образом, выражение (29) преобразуется к виду (32)

$$\begin{aligned} \tilde{H}(n+1) = & \tilde{H}(n) + \\ & + \frac{1}{\mathbf{g}^T(n) \cdot \mathbf{p}(n)} \mathbf{g}(n) \cdot \mathbf{g}^T(n) + \end{aligned}$$

$$+ \frac{1}{\eta(n) \mathbf{y}(n) \cdot \mathbf{p}(n)} \mathbf{y}(n) \mathbf{y}^T(n). \quad (32)$$

Выражения (30)–(32) служат основой алгоритма обучения персептрона по методу Бroyдена—Флетчера. Алгоритм включает оценку градиента $\mathbf{g}(n)$ функции ошибок $E(\mathbf{w}(n))$, определение вектора направления $\mathbf{p}(n)$ и схему обновления матрицы $\tilde{H}(n)$ [8].

Метод Бroyдена—Флетчера использует не только первую, но и вторую производную функции ошибок (в форме ее разностной аппроксимации) и более точно определяет направление при поиске минимума.

Метод параллельных касательных (метод partan)

Метод реализует движение в направлении, которое параллельно касательной к линии равного уровня поверхности ошибок, найденной градиентным методом для предшествующей точки [8, 19]. Метод использует матрицу Гессе (33) и алгоритм сопряженных направлений \mathbf{S}_n и \mathbf{S}_{n+1} по соотношению (34):

$$\begin{aligned} H[E(\mathbf{w}_n)] = & \nabla^2 \{E[\mathbf{w}(n)]\}, \\ \text{т. е. } H_{ij} = & \frac{\partial^2 E(\mathbf{w}_n)}{\partial w_i \partial w_j}, \end{aligned} \quad (33)$$

$$(\mathbf{S}_{n+1})^T \cdot [\nabla^2 \{E[\mathbf{w}(n)]\}] \mathbf{S}_n = 0. \quad (34)$$

Условие (34) — это условие ортогональности вектора направления \mathbf{S}_{n+1} на $(n+1)$ шаге относительно вектора $[\nabla^2 \{E[\mathbf{w}(n)]\}] \mathbf{S}_n$, где $\nabla^2 \{E[\mathbf{w}(n)]\}$ — матрица Гессе (33).

Корректировка вектора весов персептрона выполняется по соотношению

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \lambda_n \mathbf{S}_n. \quad (35)$$

Значение параметра λ , соответствующее минимуму $E(\mathbf{w}_{n+1})$ в направлении вектора \mathbf{S}_n , определяется выражением

$$\lambda_n = - \frac{[\nabla E(\mathbf{w}_n)]^T \cdot \mathbf{S}_n}{\mathbf{S}_n^T \cdot [\nabla^2 \{E(\mathbf{w}_n)\}] \cdot \mathbf{S}_n}, \quad (36)$$

где $\nabla E(\mathbf{w}_n)$ — вектор градиента $[\nabla E(\mathbf{w}(n))]_i = [\text{grad} \{E(\mathbf{w}_n)\}]_i \equiv \frac{\partial E(\mathbf{w}(n))}{\partial w_i(n)}$.

Таким образом, метод параллельных касательных требует вычисления гиперплоскости, касательной к функции ошибок, на каждой итерации.

Поэтому метод связан с большими вычислительными затратами, особенно в задачах большой размерности.

В двухмерном варианте метод основан на движении по направлению, параллельному касательной к линии постоянного уровня функции ошибок. При большой размерности аргумента (вектора весов персептрона) используются касательные гиперплоскости и движение в направлении, параллельном этим гиперплоскостям. Требование вычисления касательных гиперплоскостей и определения $\min\{E(\mathbf{w}_n)\}$ вдоль направления, параллельного этим гиперплоскостям, является недостатком метода. При значительной размерности вектора весов (в функции ошибок персептрона) это приводит к росту вычислительных затрат и метод постепенно теряет эффективность, снижая точность и увеличивая длительность обучения.

ВЫБОР МЕТОДА ОБУЧЕНИЯ ПЕРСЕПТРОНА ДЛЯ РЕШАЕМОЙ ЗАДАЧИ

При использовании любого алгоритма обучения персептрона важна инициализация синаптических весов и порогов нейронов сети. В качестве начальных значений весов и уровней порога нейронов для создания им равноценных условий подстройки в процессе обучения целесообразно выбирать равномерно распределенные числа. Значения параметров синаптической связи для нейрона i в сети рекомендуется обычно ограничивать интервалом $\left[-\frac{2.4}{F_i}, \frac{2.4}{F_i}\right]$, где F_i — общее количество входов i -го нейрона [20].

Обобщая представленный выше анализ, можно утверждать, что в алгоритмах обучения персептрона для задачи детектирования речевой активности может быть использован один из рассмотренных методов, хотя наиболее совершенными представляются два из них:

- обобщенный метод дельта—дельта (правило Джекобса), который включает адаптивный выбор параметра скорости обучения;

- метод Бройдена—Флетчера, который использует приближение для производных второго порядка (матрицу Гессе) и позволяет определять величину параметра скорости обучения.

Однако только на основе теоретического анализа нельзя сделать окончательный выбор метода обучения персептрона задаче распознавания речевого сигнала, поскольку невозможно учесть поведение алгоритмов в условиях многообразия характеристик сигнала и шума, величин пауз и наличия переходных сегментов. Исходя из этого, для обоснованного выбора метода обучения проведена

экспериментальная проверка алгоритмов при различных видах шумов и различных отношениях сигнал/шум на достаточно представительных данных.

Экспериментальное исследование характеристик алгоритмов обучения персептрона

В целях объективной оценки преимуществ алгоритмов обучения для детектирования речевой активности проведен компьютерный эксперимент, в котором на речевой сигнал, включающий интервалы активности и микропаузы, наложены различные виды шума:

- белый шум, имеющий равномерный спектр мощности в звуковом диапазоне частот;

- окрашенный шум с постепенным спадом спектра при росте частоты (для розового шума — 6 дБ/октава), представителем которого может служить шум пылесоса или шум льющейся воды; музыкальный фон;

- узкополосный шум, состоящий из детерминированных или случайных компонент, постоянных по своему положению и амплитуде (шум вентилятора).

Эксперимент проведен с использованием нейросетевого эмулятора Essence 1.0 [21], выполненного на языке Java и позволяющего программно моделировать многослойный персептрон с любым количеством слоев, устанавливать отдельно количество и вид активационной функции нейронов для каждого слоя, выполнять все перечисленные алгоритмы обучения и использовать различные виды ошибок.

Данными служили оцифрованные записи речевого сигнала и сопутствующего шума с частотой дискретизации 22.5 кГц. На каждом сегменте записи длительностью 20 мс выполнено вейвлет-разложение в соответствии с персептуальной моделью, вычислена средняя мощность вейвлет-коэффициентов в каждой области разложения и полученные векторы мощности вейвлет-коэффициентов нормированы.

На основе анализируемого зашумленного речевого сигнала получено 2000 векторов, которые составили два множества (обучающее и тестовое) по 1000 векторов. Использован принцип обучения с учителем, при котором сети сообщается желаемый отклик на каждый входной обучающий вектор. Программа представляет динамику изменения ошибки в процессе обучения и на тестирующей выборке. При сходимости обучения ошибка почти не изменяется, но если она недостаточно мала, то сеть усложняется и повторяется ее обучение и тестирование. Напротив, если ошибка по обучающей выборке стремится к нулю, а при тестировании сохраняет недопустимо большое значение, то размер сети уменьшается.

После ряда экспериментов с изменением размера сети для исследования различных алгоритмов обучения принята двухслойная сеть с числом нейронов десять и один (10-1) и с логистической активационной функцией нейронов. Для всех алгоритмов при равном времени обучения (2 мин) зафиксировано число циклов обучения и процент ошибочных решений по обучающей и тестовой выборкам. Эксперимент проведен для всех видов шума при отношениях среднеквадратического отклонения шума к сигналу $\zeta = 0.02, 0.1$ и 0.3 .

Результаты компьютерного эксперимента, позволяющие сопоставить различные алгоритмы обучения персептрона, содержит табл. 1, где приведено число циклов обучения и процент ошибочных решений при обучении и тестировании для нескольких уровней сложности нейронной сети, типов и уровней сопутствующего шума. Число циклов различается в зависимости от алгоритма и определяется его вычислительной сложностью. Для всех методов обучения процент ошибочных решений на обучающей выборке меньше, чем на тестовой выборке. Однако это отличие не превышает 20 %, что подтверждает адекватность выбранной модели персептрона.

При сопоставлении алгоритмов по оценке вероятности ошибочных решений наилучшие характеристики получены для метода Бройдена—Флетчера. Вероятность ошибочных решений зависит от интенсивности шума: для $\zeta = 0.02$ вероятность

ошибочных решений по тестовой выборке не превышает 3.7 %, для $\zeta = 0.1$ — 4.3 %, а для $\zeta = 0.3$ — 5.5 %. Вероятность ошибочных решений для узкополосного шума и музыкального фона выше, поскольку спектральный состав шума меньше отличается от речевого сигнала.

Различаются ошибочные решения первого и второго рода [22, 23]. Пусть ω_1 — событие, состоящее в том, что текущий сегмент или фрейм содержит речевой сигнал; ω_2 — событие, состоящее в том, что текущий фрейм содержит шум; Ω_1 — решение о том, что текущий фрейм содержит речевой сигнал; Ω_2 — решение о том, что текущий фрейм содержит шум. Тогда вероятность ошибочного обнаружения речевого сигнала во время паузы определяется соотношением

$$P(\Omega_1 / \omega_2) = \frac{P(\Omega_1, \omega_2)}{P(\omega_2)}. \quad (37)$$

Вероятность ошибочного обнаружения паузы при наличии речевого сигнала — соотношением:

$$P(\Omega_2 / \omega_1) = \frac{P(\Omega_2, \omega_1)}{P(\omega_1)}. \quad (38)$$

Компьютерный эксперимент показал, что вероятность ошибочного обнаружения паузы при наличии речевого сигнала ниже, чем вероятность ошибочного обнаружения речевого сигнала.

Табл. 1. Сравнение методов обучения персептрона

Вид шума	Относительный вклад шума	Метод обучения	Обучение			Тестирование
			Число циклов	Время, мин	Ошибка, %	Ошибка, %
1	2	3	4	5	6	7
Белый шум	0.02	Случайный поиск	500	5	10	13
		Сопряжен. градиенты	800	3.2	3.5	5.7
		Partan	600	3.3	4	4.7
		Дельта—дельта	600	3.3	4.5	5.2
		Обобщ. дельта—дельта	650	4	2.5	3.8
		Бройдена—Флетчера	700	2	2.7	3.1
		0.1	Случайный поиск	500	6.5	12
	Сопряжен. градиенты		900	4.5	5	8
	Partan		700	4	6.5	6.9
	Дельта—дельта		700	4	6.5	7
	Обобщ. дельта—дельта		800	4.4	5	5.3
	Бройдена—Флетчера		800	2.2	3.4	3.6

Табл. 1 (продолжение)

1	2	3	4	5	6	7
Белый шум	0.3	Случайный поиск	900	7	14	15
		Сопряжен. градиенты	900	5	7	8.2
		Partan	1000	4.8	9.5	9.3
		Дельта—дельта	700	4	9	5.5
		Обобщ. дельта—дельта	900	4.5	7	7.3
		Бройдена—Флетчера	1000	2.5	4.5	4.7
Окрашенный шум	0.02	Случайный поиск	600	5	11	12
		Сопряжен. градиенты	800	3.3	3.7	6.0
		Partan	600	3.5	4.1	4.9
		Дельта—дельта	600	3.5	4.6	5.5
		Обобщ. дельта—дельта	700	4.5	2.4	4
		Бройдена—Флетчера	800	2.1	2.6	3.3
	0.1	Случайный поиск	500	6.4	12.5	13
		Сопряжен. градиенты	850	4.6	4.8	4.8
		Partan	800	4.3	6	6.8
		Дельта—дельта	600	4.1	7	7
		Обобщ. дельта—дельта	850	4	4.2	4.2
		Бройдена—Флетчера	900	2.0	3.2	3.4
	0.3	Случайный поиск	600	5	14.5	15
		Сопряжен. градиенты	800	4	6.5	6.8
		Partan	650	3.5	8	8.5
		Дельта—дельта	700	3.0	7	8.3
		Обобщ. дельта—дельта	900	3.6	6.5	6
		Бройдена—Флетчера	800	2	4.8	5
Музыкальный фон	0.02	Случайный поиск	500	4	10	10.5
		Сопряжен. градиенты	750	3.1	3.8	4.0
		Partan	650	3.5	4	4.2
		Дельта—дельта	600	3.3	4.5	4.6
		Обобщ. дельта—дельта	750	4.0	2.6	2.8
		Бройдена—Флетчера	900	2.8	2.9	3.0
	0.1	Случайный поиск	400	7	12	12.5
		Сопряжен. градиенты	800	5	5	5.2
		Partan	850	4	5.8	6
		Дельта—дельта	550	4.2	7	7.1
		Обобщ. дельта—дельта	800	3.9	4.1	4.2
		Бройдена—Флетчера	1000	2.3	3.1	3.3
	0.3	Случайный поиск	600	4.5	14	15.2
		Сопряжен. градиенты	700	4	6.7	6.9
		Partan	600	3.2	8	8.2
		Дельта—дельта	650	3.2	7.3	8
		Обобщ. дельта—дельта	800	3.5	5.3	5.6
		Бройдена—Флетчера	900	1.8	6	5.5
Узкополосный шум	0.02	Случайный поиск	500	5	10.5	10
		Сопряжен. градиенты	700	3.2	4.5	4.7
		Partan	600	3.3	5	5.4
		Дельта—дельта	600	3.6	5.8	6.2
		Обобщ. дельта—дельта	600	4	2.6	2.8
		Бройдена—Флетчера	600	2.6	3.5	3.7

Табл. 1 (продолжение)

1	2	3	4	5	6	7
Узкополосный шум	0.1	Случайный поиск	600	4.5	12	11.8
		Сопряжен. градиенты	700	3.8	6.0	6.4
		Partan	550	3.5	6	6.6
		Дельта—дельта	700	3.8	7.0	7.3
		Обобщ. дельта—дельта	800	4.2	3.5	3.8
		Бройдена—Флетчера	800	4.5	4.1	4.3
		Случайный поиск	750	4	14	15
	0.3	Сопряжен. градиенты	900	5	8	9
		Partan	800	4.3	8.2	8.1
		Дельта—дельта	900	4.5	9	9.5
		Обобщ. дельта—дельта	1000	5	4.9	5.3
		Бройдена—Флетчера	1000	2.8	3.2	3.6

ВЫБОР СТРУКТУРЫ ПЕРСЕПТРОНА

Выбор структуры многослойного персептрона (МСП) важен, поскольку повышенная сложность сети приводит к увеличению объема вычислений и длительности обучения, возникновению неустойчивости и переобученности, которая ухудшает показатели точности на материале тестирования. Чрезмерное упрощение в свою очередь не позволяет получить хорошее качество детектирования свойств сигнала. Поэтому построение нейронной сети для системы детектирования состояния процесса связано с выбором минимальной конфигурации, обеспечивающей хорошее функционирование.

Для сети минимального размера маловероятно обучиться несущественным деталям в обучающих данных, поэтому она может давать лучшие результаты на материале тестирования. Способ реализации такого подхода — это выбор сети малой сложности и наращивание ее до получения приемлемых показателей алгоритма обучения и правильности работы на тестовом материале.

Другой подход состоит в начальном выборе сети достаточно сложной структуры, обеспечивающей приемлемую ошибку обучения, после чего производится упрощение структуры сети путем удаления избыточных нейронов. Дополнительным ресурсом является метод регуляризации структуры персептрона, в котором вместо простого критерия среднеквадратической ошибки функция риска учитывает также меру сложности структуры персептрона.

Выбор структуры нейронной сети осуществлен на основе совместного использования методов упрощения и наращивания и метода регуляризации.

Для метода наращивания в ходе проведения серии циклов обучения, основываясь на критерии ошибки (9), определялось место размещения в сети дополнительного нейрона. Одновременно из сети удалялся нейрон, веса которого флуктуировали после сходимости обучения, т. к. этот нейрон не является существенным для задачи обучения. Результаты моделирования получены на ЭВМ с использованием программы Essence 1.0 для речевого сигнала с белым шумом при отношении среднеквадратического отклонения шума и сигнала 0.1 и 0.3. Метод наращивания сложности МСП проанализирован для структур: 3-1; 5-1; 5-1-1; 5-2-1; 5-3-1 и двух лучших алгоритмов обучения: Бройдена—Флетчера и обобщенного правила дельта—дельта (табл. 2). Введение третьего нейрона во втором скрытом слое МСП несущественно меняет ошибку, но обнаруживает флуктуации вектора весов дополнительного нейрона в конце цикла обучения. Поэтому метод указывает на предпочтительность структуры 5-2-1.

Метод упрощения, где после каждого цикла обучения удалялся нейрон, отсутствие которого дает минимальное увеличение ошибки обучения, применен к структурам персептрона 10-3-1; 10-2-1; 8-2-1; 5-2-1 (табл. 3). Результаты анализа согласуются с выводом, полученным по методу наращивания.

Метод регуляризации [5] вместо критерия среднеквадратической ошибки использует функцию риска $R(\mathbf{w})$:

$$R(\mathbf{w}) = E(\mathbf{w}) + \lambda \cdot E_c(\mathbf{w}), \tag{39}$$

где $E(\mathbf{w})$ — среднеквадратическая ошибка,

$E_c(\mathbf{w})$ — мера сложности структуры персептрона, а параметр λ определяет относительную значимость компоненты $E_c(\mathbf{w})$ и служит параметром регуляризации (поскольку форма критерия $R(\mathbf{w})$ опирается на теорию регуляризации А.Н. Тихонова). В качестве меры сложности структуры персептрона использован квадрат нормы вектора синаптических весов МСП

$$E_c(\mathbf{w}) = \|\mathbf{w}\|^2 = \sum_{i \in C} w_i^2, \quad (40)$$

где C — множество индексов всех весов сети. Эксперимент проведен на структурах МСП 5-2-1 и 5-3-1 с изменением параметра регуляризации от 0 (в исходном состоянии сети) до 0.01 с шагом 0.001 и показал, что один из нейронов второго скрытого слоя в 5-3-1 имеет 1.5 %-й вклад в $E_c(\mathbf{w})$ и может считаться несущественным. Это подтверждает результат, полученный методом наращивания сложности персептрона.

Табл. 2. Результаты определения оптимальной структуры многослойного персептрона методом наращивания

Структура МСП	Отношение шум/сигнал	Доля ошибочных решений персептрона (%)	
		Обобщенное правило дельта—дельта	Алгоритм Бройдена—Флетчера
3-1	0.1	8.1	6.5
	0.3	9.7	7.0
5-1	0.1	6.5	5.0
	0.3	8.3	5.5
5-1-1	0.1	4.9	4.0
	0.3	6.9	4.5
5-2-1	0.1	3.7	3.2
	0.3	5.5	4.0
5-3-1	0.1	3.9	3.4
	0.3	5.8	4.2

Табл. 3. Результаты определения оптимальной структуры многослойного персептрона методом упрощения

Структура персептрона	Отношение шум/сигнал	Доля ошибочных решений персептрона (%)	
		Обобщенное правило дельта—дельта	Алгоритм Бройдена—Флетчера
10-3-1	0.1	2.1	2.3
	0.3	4.2	3.1
10-2-1	0.1	2.3	2.5
	0.3	4.5	3.6
8-2-1	0.1	2.9	2.8
	0.3	5.0	3.8
5-2-1	0.1	3.7	3.2
	0.3	5.5	4.0

ИСПОЛЬЗОВАНИЕ ГЛАВНЫХ КОМПОНЕНТ СИГНАЛА ДЛЯ ОБУЧЕНИЯ И РАСПОЗНАВАНИЯ

Дальнейшее улучшение алгоритма детектирования речевой активности достигнуто уменьшением размерности входного вектора персептрона, которое позволило уменьшить число синаптических весов, уменьшить объем обучающей выборки и ускорить процесс обучения. Уменьшение размерности входных векторов при условии сохранения содержащейся в них информации о речевом сигнале основано на методе главных компонент [24, 25].

Входной вектор МСП $\mathbf{x}_j = [x_{1j}, x_{2j}, \dots, x_{pj}]$ имеет $p = 22$ компоненты, число которых соответствует персептуальной модели. Идея метода состоит в получении p линейных комбинаций из компонент вектора — главных компонент, каждая из которых линейно независима от других. Главная компонента y выражается соотношением:

$$\mathbf{y} = X\hat{\mathbf{a}}, \tag{41}$$

где

$$X = \begin{bmatrix} x_{11} & x_{21} & \dots & x_{p1} \\ x_{12} & x_{22} & \dots & x_{p2} \\ \dots & \dots & \dots & \dots \\ x_{1N} & x_{2N} & \dots & x_{pN} \end{bmatrix},$$

$\hat{\mathbf{a}} = [\hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_p]^T$ — вектор неизвестных коэффициентов, X — матрица обучающих векторов размерности $N \times p$, N — объем обучающей выборки.

Выборочная дисперсия y определяется выражением:

$$\text{var}(\mathbf{y}) = \hat{\mathbf{a}}^T S \hat{\mathbf{a}}, \tag{42}$$

где S — выборочная ковариационная матрица обучающих векторов \mathbf{X}

$$S = \frac{1}{N-1} \times \begin{bmatrix} \sum_{i=1}^N x_{1i}^2 & \sum_{i=1}^N x_{1i}x_{2i} & \dots & \sum_{i=1}^N x_{1i}x_{pi} \\ \sum_{i=1}^N x_{2i}x_{1i} & \sum_{i=1}^N x_{2i}^2 & \dots & \sum_{i=1}^N x_{2i}x_{pi} \\ \dots & \dots & \dots & \dots \\ \sum_{i=1}^N x_{pi}x_{1i} & \sum_{i=1}^N x_{pi}x_{2i} & \dots & \sum_{i=1}^N x_{pi}^2 \end{bmatrix}. \tag{43}$$

Главные компоненты, начиная с первой y_1 , определяются по критерию максимальной дисперсии $\hat{\mathbf{a}}^T S \hat{\mathbf{a}}$ при условии, что вектор коэффициентов $\hat{\mathbf{a}}$ нормирован, т.е. $\hat{\mathbf{a}}^T \hat{\mathbf{a}} = 1$. Решение условной максимизации использует функцию $\varphi = \hat{\mathbf{a}}^T S \hat{\mathbf{a}} - \hat{\lambda}(\hat{\mathbf{a}}^T \hat{\mathbf{a}} - 1)$ ($\hat{\lambda}$ — множитель Лагранжа), градиент которой $\frac{d\varphi}{d\hat{\beta}} = 2S\hat{\mathbf{a}} - 2\hat{\lambda}\hat{\mathbf{a}}$ после

приравнивания к нулю дает уравнение $(S - \hat{\lambda}I)\hat{\mathbf{a}} = 0$ для определения компонент вектора $\hat{\mathbf{a}}$. Это матричное уравнение имеет решение, если равен нулю определитель $|S - \hat{\lambda}I| = 0$. Определитель представляет собой алгебраическое уравнение степени p относительно $\hat{\lambda}$ и позволяет найти p корней, являющихся характеристическими числами $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_p$ ковариационной матрицы S .

Наибольшее $\hat{\lambda}_1$ максимизирует дисперсию y_1 , а решение системы $(S - \hat{\lambda}I)\hat{\mathbf{a}} = 0$ дает вектор $\hat{\mathbf{a}}_1$, определяющий первую главную компоненту. Следующие по величине характеристические числа в качестве решения системы $(S - \hat{\lambda}I)\hat{\mathbf{a}} = 0$ дают векторы $\hat{\mathbf{a}}$, которые определяют остальные главные компоненты по соотношению (41). Таким образом, для p -мерных данных первая главная компонента $y_1 = X \cdot \hat{\mathbf{a}}_1$ представляет собой линейную комбинацию p компонент с коэффициентами, равными нормированному собственному вектору корреляционной матрицы S , соответствующему максимальному характеристическому числу этой матрицы. Аналогично остальные главные компоненты соответствуют следующим по величине характеристическим числам матрицы S . Главные компоненты линейно независимы, и дисперсия каждой компоненты равна соответствующему характеристическому числу матрицы S .

Общая дисперсия p компонент равна следу выборочной ковариационной матрицы $\text{tr}(S) = \text{var}(y_1) + \text{var}(y_2) + \dots + \text{var}(y_p)$. Относительная значимость j -й главной компоненты определяется соотношением

$$\frac{\text{var}(y_j)}{\sum_{j=1}^p \text{var}(y_j)} = \frac{\hat{\lambda}_j}{\text{tr}(S)} = \frac{\hat{\lambda}_j}{\sum_{j=1}^p \hat{\lambda}_j}. \tag{44}$$

Для уменьшения размерности выполнен анализ и выбраны q наиболее значимых компонент $q \leq p$, которые определяют η % от общей дисперсии всех p компонент. Для $\eta = 95 \dots 98$ % вклад оставших-

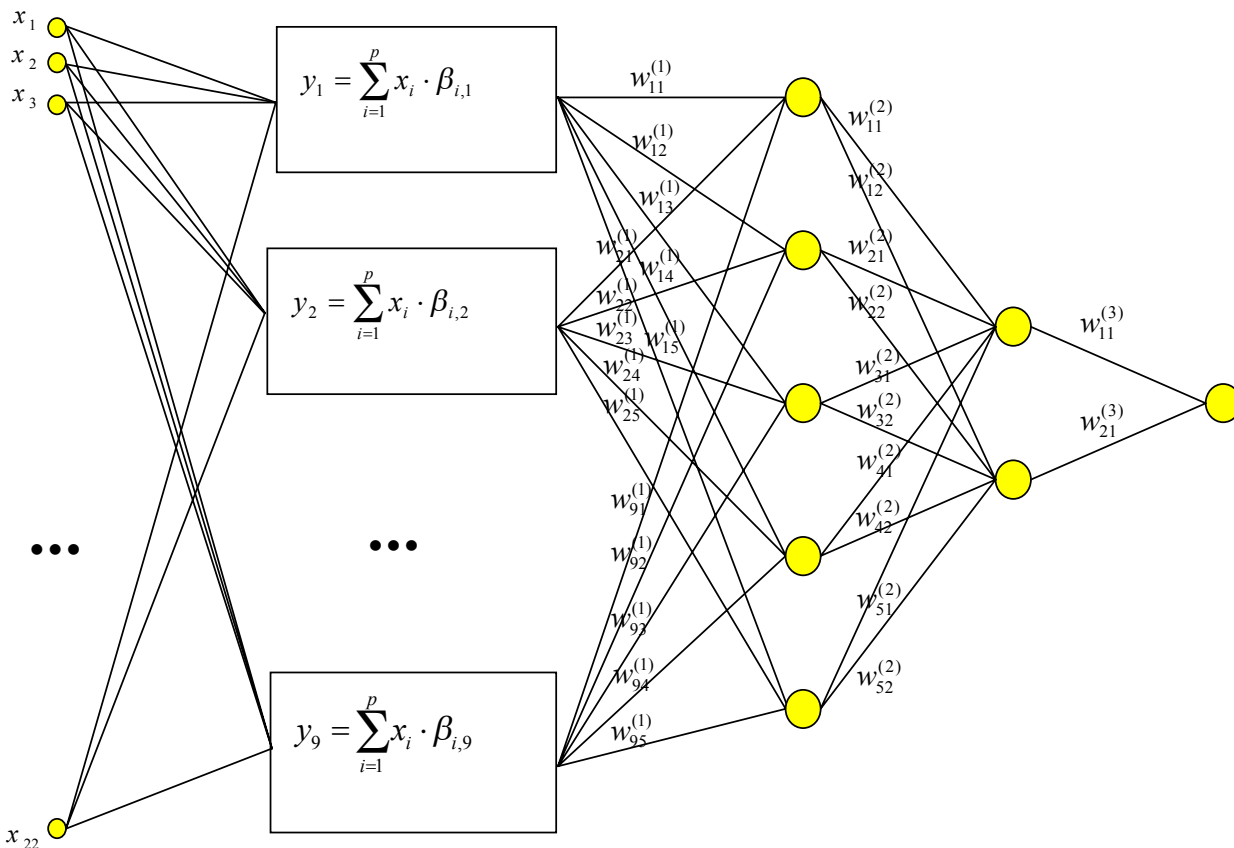
ся $p - q$ компонент может считаться несущественным.

При проведении исследования программа Essence 1.0 дополнена методом главных компонент. Исходными данными служили 100 векторов, представляющих нормированную мощность коэффициентов вейвлет-разложения речевого сигнала в соответствии с перцептуальной моделью.

Отношение сигнал/шум варьировалось от 3 до 50. Для каждого из четырех уровней шума речевой сигнал, преобразованный вейвлет-пакетом и имеющий первоначально 22 компоненты, обработан программой главных компонент, в результате чего получены 9 компонент, содержащих не менее 97.5 % энергии исследуемого сигнала. При этом допустимый уровень потери энергии составил не более $\eta = 2.5$ %. Использование главных компонент в структуре перцептрона показано на рисунке. Обучение перцептрона проведено с использо-

ванием алгоритмов Бройдена—Флетчера и обобщенного правила дельта—дельта, т. к. они дали лучшие результаты обучения на исходных обучающих векторах. При уменьшении отношения сигнал/шум (в исследованном диапазоне 3–50) ошибка обучения возрастает на 60 % для метода Бройдена—Флетчера и вдвое — для обобщенного правила дельта—дельта Джекобса. Для белого и окрашенного шумов метод обобщенного правила дельта—дельта имеет чуть меньшую ошибку обучения (на 0.1 %) при малом отношении шум/сигнал (до 0.05) и уступает при больших уровнях шума.

Для условий узкополосного шума и музыкального фона метод обобщенного правила дельта—дельта дает меньшую ошибку обучения и тестирования на всем диапазоне уровней шума от 0.5 до 0.2 %.



Использование главных компонент в структуре перцептрона 5-2-1

Для проверки этой закономерности проанализированы алгоритмы обучения по методу сопряженных градиентов и методу параллельных касательных (метод partan), которые в аналогичных условиях давали в среднем на 1 % большую ошибку обучения, чем первые два метода. Для белого и окрашенного шумов метод параллельных касательных уступает методу сопряженных градиентов — ошибка на 1 % больше во всем диапазоне отношений шум/речь. Для музыкального фона это различие несколько меньше, а для узкополосного шума метод параллельных касательных дает уже на 0.5 % меньшую ошибку обучения.

Методы Бройдена—Флетчера и обобщенное правило дельта—дельта дают близкие показатели относительной ошибки и количество циклов обучения.

Метод главных компонент дает возможность снизить количество циклов на 30–35 % (в зависимости от метода и вида шума) и уменьшить время обучения на 40–50 %. В то же время ошибка обучения увеличивается на 0.4–0.6 %. Для узкополосного шума и музыкального фона это снижение меньше, чем для белого и окрашенного шумов, во всем диапазоне уровней шума.

Применение метода главных компонент позволило

— уменьшить размерность входного вектора перцептрона с 22 до 9 компонент;

— уменьшить число оцениваемых в процессе обучения свободных параметров сети (синаптических весов) со ста двадцати двух до пятидесяти двух для нейронной сети 5-2-1;

— соответственно увеличить скорость обучения и уменьшить объем обучающей выборки.

ЗАКЛЮЧЕНИЕ

Для детектирования речевой активности предложен метод, включающий вейвлет-преобразование сигнала в соответствии с перцептуальной моделью и нейросетевой алгоритм (см. статью I этой серии). Аппроксимация перцептуальной модели реализована на основе вейвлет-пакета, разделяющего частотный диапазон сигнала на 22 субполосы.

Принятие решения о наличии речевой активности на анализируемом сегменте осуществлено на основе нейронной сети в форме многослойного перцептрона (МСП). На входы МСП поступает нормированный вектор с компонентами, равными мощности вейвлет-коэффициентов в субполосах разложения сигнала. Исследование показало, что для обучения МСП целесообразно использовать метод Бройдена—Флетчера, который по результа-

там компьютерного эксперимента имеет наилучшие характеристики по скорости обучения, величине ошибок (на обучающей и тестовой выборке) и по вероятности ошибочной классификации сегмента сигнала.

Проведены экспериментальные исследования по регуляризации структуры перцептрона, основанные на методах наращивания, упрощения и регуляризации, которые показали, что оптимальной для решения данной задачи является структура перцептрона 5-2-1.

Применение метода главных компонент позволяет уменьшить размерность входного вектора нейронной сети, уменьшить число оцениваемых в процессе обучения синаптических весов и увеличить скорость обучения при уменьшенном объеме обучающей выборки.

СПИСОК ЛИТЕРАТУРЫ

1. *Battiti R.* First and Second Order Methods for Learning: Between Steepest Descent and Newtons' Method // *Neural Computations*. 1992. N 4. P. 141–166
2. *Baum E.B.* Neural Net Algorithms that Learn in Polynoial Time from Examples and Qeries // *IEEE Transactions on Neural Networks*. 1991. N 2. P. 5–19.
3. *Baum E.B., Wilczek F.* Supervised learning of probability distributions by neural networks / Ed. D.Z. Anderson. N.Y.: American Institute of Physics, 1988. P. 52–61.
4. *Gallant A.R., White H.* // *Neural Networks*. 1990. N 5. P. 129–138.
5. *Hinton G.E., Nowban S.J.* How Learning Can Guide Evolution // *Complex systems*. 1987. N 1. P. 495–512.
6. *Hopfield J.J.* The Effectiveness of Analogue Neural Network.Hardware // *Networks*. 1990. N 1. P. 27–40.
7. *Saarinен S. et al.* Neural networks, back-propagation and automatic differentiation // *Automatic Differentiation of Algorithm: Theory, Implementation and Application* / Eds. Grievank A., Corless G.F. Philadelphia, MA, SIAM, 1991. P. 31–42.
8. *Гулл Ф., Мюппей У., Райт М.* Прикладная оптимизация. М.: Мир, 1985. 209 с.
9. *Химмельблау Р.* Прикладное нелинейное программирование. М: Мир, 1975. 98 с.
10. *Ariel M.* Nonlinear Programming: Analyses and Methods. N.Y.: Prentice Hall, 1976.
11. *Dorny C.N.* A Vector Space Approach to Models and Optimization. N.Y.: Wiley (Interscience Publishing), 1975. 289 p.

12. *Hestenes M.G.* Conjugate Direction Method in Optimization. Berlin—Heidelberg—N.Y., 1980. 48 p.
13. *Jackobs R.A.* Increased Rates of Convergence through Learning Rate Adaptation // *Neural Networks*. 1989. V. 1. P. 295–307.
14. *Broyden C.G.* A Class of Methods for Solving Nonlinear Simultaneous Equations // *Mathematics of Computation*. 1965. N 19. P. 577–593.
15. *Broyden C.G.* Quasi-Newton Methods and their Application to Function Minimization // *Mathematics of Computation*. 1967. N 21. P. 368–381.
16. *Broyden C.G.* The Convergence of Minimization Algorithms // *Journal of Institute of Mathematical Applications*. 1970. N 6. P. 76–90.
17. *Broyden C.G., Dennis J.E., Moro J.J.* On the Local and Superlinear Convergence of Quasi-Newton Methods // *Journal of Institute of Mathematical Applications*. 1973. N 12. P. 223–245.
18. *Shanno D.F.* Conditioning of quasi-Newton methods for function initialization // *Mathematics of Computation*. 1970. N 24. P. 647–657.
19. *Shah B.V., Buchler R.J., Kempthorne O.* // *Journal Society of Industrial and Applied Mathematics*. 1964. V. 12. P. 74.
20. *Haykin S.* *Neural Networks. A Comprehensive Foundation*. N.Y.: Prentice Hall, 1994. 680 p.
21. *Власов Л.В., Мальхина Г.Ф., Тархов Д.А.* Нейронный эмулятор "ESSENCE" // Датчики и системы: Сборник докладов Международной конференции. СПб.: Изд-во СПбГТУ, 2002. Т. 3. С. 149–153.
22. *Загрудинов Г. М.* Достоверность автоматизированного контроля. Казань: Изд-во Казанского университета, 1980. 280 с.
23. *Любатов Ю.В.* Теоретические основы моделирования цифровых систем. М.: МАИ, 1989. 77 с.
24. *Болч Б., Хуань К. Дж.* Многомерные статистические методы для экономики. М.: Статистика, 1979. 316 с.
25. *Кендал М. Дж., Стьюарт А.* Многомерный статистический анализ и временные ряды (Пер.с англ. под ред. А.Н. Колмогорова и Ю.В. Прохорова). М.: Изд-во Наука, 1976. 736 с.

Санкт-Петербург

Материал поступил в редакцию 10.11.2002.

**APPLICATION OF A NEURAL NETWORK
TO ON-LINE ANALYSIS OF NON-STATIONARY (SPEECH)
SIGNALS REPRESENTED BY THEIR
WAVELET TRANSFORM.
II. STUDY AND OPTIMIZATION OF THE NEURAL NETWORK**

A. V. Merkusheva

Saint-Petersburg

The neural-network perceptron complexity level is well grounded and main learning characteristics are obtained. We suggest the tools for input vectors dimensionality reducing for perceptron learning and signal state recognition based on the principal component analysis.