М. И. ШЛЕЗИНГЕР

ВЗАИМОСВЯЗЬ ОБУЧЕНИЯ И САМООБУЧЕНИЯ В РАСПОЗНАВАНИИ ОБРАЗОВ

УДК 51:681.14:155

ВВЕДЕНИЕ

Для решения задачи распознавания образов в ее статистической постановке [1—3] необходимо, чтобы на некотором пространстве **V** изображений были заданы распределения $p(v/k)$ изображений при условии их принадлежности k-му образу; кроме того, должны быть известны априорные вероятности образов p_k ($k = 1, 2, \ldots, s$; s — количество образов). На основании этих данных может быть построена решающая функция $h(v)$, указывающая, что изображение v следует отнести к образу h. При этом может быть найдена такая решающая функция, при которой обеспечивается минимальная вероятность ошибочного распознавания.

Обучение или самообучение распознающих систем имеет место при условии, что распределения $p(v/k)$ известны не полностью, а лишь с точностью до неизвестного параметра a_k. В таком случае задача обучения или самообучения заключается в определении этих параметров. Определение неизвестных параметров производится на основании так называемой обучающей последовательности, т. е. некоторого множества изображений, предъявленных распознающей системе. При обучении показ каждого изображения сопровождается указанием, к какому образу оно относится. При самообучении отсутствуют какие-либо указания о принадлежности предъявленных изображений тому или иному образу. Именно в этом заключается разница между обучением и самообучением в распознавании образов.

ПОСТАНОВКА ЗАДАЧИ

Введем следующие обозначения. Совокупность неизвестных параметров обозначим через A:

$$A = \{a_1, a_2, \ldots, a_s; p_1, p_2, \ldots, p_s\}. \qquad (1)$$

В число неизвестных параметров включаем и априорные вероятности образов.

Распределение изображений, принадлежащих k-му образу, обозначим через $p(v/a_k)$. Распределение изображений, принадлежащих всей совокупности образов, обозначим через $P(v)$. Очевидно равенство

$$P(v) = \sum_{k=1}^{s} p_k \cdot p(v/a_k). \qquad (2)$$

Из записи (2) видно, что распределение изображений, принадлежащих всей совокупности образов, вообще говоря, зависит от совокупности неизвестных параметров A. Поэтому это распределение обозначим через $P(v/A)$:

$$P(v/A) = \sum_{k=1}^{s} p_k \cdot p(v/a_k). \qquad (3)$$

Через V_k будем обозначать выборку изображений, принадлежащих k-му образу, а через V — выборку изображений, принадлежащих всей совокупности образов. В таком случае ясно, что обучающая последовательность, предъявленная распознающей системе в режиме обучения, есть совокупность выборок V_k ($k = 1, 2, \ldots, s$), в то время как в режиме самообучения распознающей системе предъявляется выборка V. На основании этого исходного материала должны быть определены неизвестные параметры. Поскольку исходные данные имеют случайный характер, то не может ставиться задача о точном определении неизвестных параметров, а лишь о нахождении их оценок.

Оценки наибольшего правдоподобия являются в некотором отношении оптимальными. Как показано в [4], оценка наибольшего правдоподобия является эффективной, если для данного параметра вообще существует эффективная оценка. Кроме того, при увеличении объема выборки оценка наибольшего правдоподобия всегда стремится к эффективной. В силу этого можем дать следующие формулировки задач обучения и самообучения.

Задача обучения. Распределение изображений, принадлежащих k-му образу, известно с точностью до неизвестного параметра a_k, т. е. задана функция двух переменных $p(v/a_k)$. **Дана** выборка V_k изображений $v_1, v_2, \ldots, v_m$, принадлежащих k-му образу. **Требуется вычислить оценку наибольшего правдоподобия для неизвестного параметра** a_k. **Известно [4], что этой оценкой яв-**

ляется то значение a_k, которое обеспечивает максимум выражения

$$l(a_k)=\sum_{i=1}^{m}\log p(v_i/a_k). \qquad (4)$$

Оценки для априорных вероятностей образов могут быть получены лишь в том случае, если выборка изображений разных образов производилась случайно в соответствии с априорными вероятностями образов. В этом случае оценки для априорных вероятностей определяются величинами, пропорциональными объемам выборок V_k.

Задача самообучения. Дана выборка V изображений $v_1, v_2, \ldots, v_m$ из общей совокупности образов. Требуется вычислить оценку наибольшего правдоподобия для неизвестных параметров $a_1, a_2, a_3, \ldots, a_s$ и априорных вероятностей p_1, $p_2, p_3, \ldots, p_s$, т. е. найти максимум выражения

$$L(A)=\sum_{i=1}^{m}\log P(v_i/A).$$

Подставляя в это выражение вместо $P(v/A)$ правую часть (3), получим

$$L(A)=\sum_{i=1}^{m}\log\sum_{k=1}^{s}p_k\cdot p(v_i/a_k). \qquad (5)$$

АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ САМООБУЧЕНИЯ

Нетрудно понять, что задача нахождения максимума выражения (4) несущественно отличается от задачи нахождения максимума выражения

$$\sum_{i=1}^{m}\alpha_{ik}\log p(v_i/a_k). \qquad (6)$$

К отысканию максимума выражения (6) задача обучения сводится в том случае, когда предъявление каждого изображения в обучающей последовательности сопровождается не точным указанием, какому образу принадлежит изображение, а дается лишь вероятность α_{ik} того, что i-ое изображение принадлежит k-му образу. Оценки для априорных вероятностей в этом случае определяются величинами, пропорциональными $\sum_{i=1}^{m}\alpha_{ik}$.

Допустим, что нам известен алгоритм обучения, т. е. известен алгоритм нахождения значения параметра a_k, при котором обеспечивается максимум выражения (6). В таком случае можем сформулировать и алгоритм самообучения, т. е. алгоритм нахождения значений всех неизвестных параметров и априорных вероятностей, обеспечивающих максимум выражения (5).

Исходными данными для алгоритма являются предъявленные изображения $v_1, v_2, v_3, \ldots$ $\ldots, v_m$, принадлежащие всей совокупности образов.

Алгоритм представляет собой итерационную процедуру, которая по параметрам

$$A^{(t)}=\{a_1^{(t)}, a_2^{(t)}, \ldots, a_s^{(t)}; p_1^{(t)}, p_2^{(t)}, \ldots, p_s^{(t)}\}$$

определяет новые значения параметров

$$A^{(t+1)}=\{a_1^{(t+1)}, a_2^{(t+1)}, \ldots$$
$$\ldots, a_s^{(t+1)}; p_1^{(t+1)}, p_2^{(t+1)}, \ldots, p_s^{(t+1)}\},$$

которые более правдоподобны, чем $A^{(t)}$.

Одна итерация алгоритма, т. е. получение параметров $A^{(t+1)}$ по полученным ранее параметрам $A^{(t)}$, состоит из двух этапов.

Этап 1. Вычисление величин $\alpha_{ik}(A^{(t)})$ ($i=1, 2, \ldots, m$; $k=1, 2, \ldots, s$) по формуле

$$\alpha_{ik}(A^{(t)})=\frac{p_k^{(t)}\cdot p(v_i/a_k^{(t)})}{\sum_{k=1}^{s}p_k^{(t)}\cdot p(v_i/a_k^{(t)})}. \qquad (7)$$

Как видно из формулы (7), величина $\alpha_{ik}(A^{(t)})$ — вероятность того, что изображение v_i принадлежит k-му образу при условии, что совокупность образов действительно характеризуется параметрами $A^{(t)}$. Именно в вычислении этих вероятностей заключается наиболее трудоемкая часть работы распознающей системы. Поэтому можно утверждать, что первый этап — это алгоритм распознавания, результатом действия которого являются апостериорные вероятности α_{ik}, являющиеся исходными данными для второго этапа.

Этап 2 заключается в нахождении величин $a_k^{(t+1)}$ ($k=1, 2, 3, \ldots, s$), обеспечивающих максимум выражения

$$l(a_k)=\sum_{i=1}^{m}\alpha_{ik}\log p(v_i/a_k),$$

и величин p_k^{t+1} ($k=1, 2, \ldots, s$), которые пропорциональны величинам $\sum_{i=1}^{m}\alpha_{ik}$.

Легко заметить, что второй этап решает задачу обучения в приведенной нами постановке.

Результатом работы этого этапа являются величины $a_k^{(t+1)}$, p_k^{t+1} $(k = 1, 2, \ldots, s)$, которые будут использованы в первом этапе следующей итерации. Предельное значение $A^{(t)}$ является оценкой наибольшего правдоподобия для неизвестных параметров A.

Таким образом, очевидно, что алгоритм самообучения представляет собой многократное повторение двух процедур, первая из которых представляет алгоритм распознавания, а вторая — алгоритм обучения.

ОБОСНОВАНИЕ АЛГОРИТМА

Лемма 1. Пусть α_i $(i = 1, 2, \ldots, s)$ — положительные константы, x_i — переменные такие, что $\sum_{i=1}^{s} x_i = c$. Максимальное значение $f = \sum_{i=1}^{s} \alpha_i \log x_i$ достигается тогда, когда величины x_i пропорциональны величинам α_i.

Лемма легко может быть доказана для $s = 2$, а затем методом математической индукции обобщена для любого s.

Теорема 1. Пусть $A^{(t)}$, $A^{(t+1)}$ — значения неизвестных параметров, полученных соответственно после t-ой и $(t+1)$-ой итераций алгоритма самообучения. В таком случае, если $A^{(t)} \neq A^{(t+1)}$, то $L(A^{(t)}) < L(A^{(t+1)})$.

Доказательство. На основании того, что $\sum_{k=1}^{s} \alpha_{ik} = 1$ для всех i (см. формулу (7)), выражение для $L(A^{(t)})$ можно записать следующим образом:

$$L(A^{(t)}) = \sum_{i=1}^{m} \log \sum_{k=1}^{s} p_k^{(t)} \cdot p(v_i/a_k^{(t)}) =$$

$$= \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p_k^{(t)} +$$

$$+ \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k^{(t)}) -$$

$$- \sum_{i=1}^{m} \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t)} \cdot p(v_i/a_k^{(t)})}{\sum_{k=1}^{s} p_k^{(t)} \cdot p(v_i/a_k^{(t)})}. \quad (8)$$

Аналогично можно записать выражение для $L(A^{(t+1)})$:

$$L(A^{(t+1)}) = \sum_{i=1}^{m} \log \sum_{k=1}^{s} p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)}) =$$

$$= \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p_k^{(t+1)} +$$

$$+ \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k^{(t+1)}) -$$

$$- \sum_{i=1}^{m} \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}{\sum_{k=1}^{s} p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}.$$

Докажем три неравенства.

$$\sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p_k^{(t)} \leqslant \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p_k^{(t+1)}; \quad (9)$$

$$\sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k^{(t)}) \leqslant$$

$$\leqslant \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k^{(t+1)}); \quad (10)$$

$$\sum_{i=1}^{m} \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t)} \cdot p(v_i/a_k^{(t)})}{\sum_{k=1}^{s} p_k^{(t)} \cdot p(v_i/a_k^{(t)})} \geqslant$$

$$\geqslant \sum_{i=1}^{m} \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}{\sum_{k=1}^{s} p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}, \quad (11)$$

причем, по крайней мере, одно из первых двух неравенств выполняется строго.

Докажем неравенство (9).

По определению (этап 2 алгоритма) величина $p_k^{(t+1)}$ пропорциональна величине $\sum_{i=1}^{m} \alpha_{ik}(A^{(t)})$. К тому же очевидным является равенство $\sum_{k=1}^{s} p_k^{(t+1)} = \sum_{k=1}^{s} p_k^{(t)}$, так как обе эти суммы равны единице. Следовательно, соблюдаются условия леммы, и выражение

$$\sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p_k \text{ при } p_k = p_k^{(t+1)}$$

достигает абсолютного максимума. Отсюда вытекает справедливость неравенства (9). При этом равенство соблюдается лишь в том случае, если для любого k соблюдается равенство $p_k^{(t)} = p_k^{(t+1)}$, так как в силу леммы рассматриваемое выражение имеет единственный максимум.

Докажем неравенство (10).

По определению (этап 2 алгоритма) $a_k^{(t+1)}$ обеспечивает максимум выражения

$$\sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k),$$

и поэтому справедливо неравенство

$$\sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k^{(t)}) \leqslant$$

$$\leqslant \sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log p(v_i/a_k^{(t+1)}). \qquad (12)$$

Суммируя выражение (12) по индексу k, получаем неравенство (10). При этом равенство соблюдается лишь в том случае, если для любого k соблюдается равенство $a_k^{(t)} = a_k^{(t+1)}$.

По условию теоремы $A^{(t)} \neq A^{(t+1)}$. Это значит, что по крайней мере один неизвестный параметр или одна априорная вероятность изменились в результате $(t+1)$-ой итерации, а значит по крайней мере одно из неравенств (9) и (10) выполняется строго.

Докажем неравенство (11).

Рассмотрим выражение

$$\sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k \cdot p(v_i/a_k)}{\sum_{k=1}^{s} p_k \cdot p(v_i/a_k)}. \qquad (13)$$

По определению (см. (7)) при $p_k = p_k^{(t)}$, $a_k = a_k^{(t)}$ выражение под знаком логарифма равно коэффициенту $\alpha_{ik}(A^{(t)})$. Следовательно, согласно лемме 1 при $p_k = p_k^{(t)}$, $a_k = a_k^{(t)}$ выражение (13) достигает своего максимального значения, откуда непосредственно следует неравенство

$$\sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t)} \cdot p(v_i/a_k^{(t)})}{\sum_{k=1}^{s} p_k^{(t)} \cdot p(v_i/a_k^{(t)})} \geqslant$$

$$\geqslant \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}{\sum_{k=1}^{s} p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}.$$

Просуммировав последнее неравенство по индексу i, получаем неравенство (11).

Из неравенств (9) — (11) непосредственно следует неравенство $L(A^{(t)}) < L(A^{(t+1)})$.

Теорема доказана.

Следствие. Величина $L(A^{(t+1)}) - L(A^{(t)})$ с увеличением t стремится к нулю.

Доказательство. Из теоремы 1 следует, что последовательность $L(A^{(0)}), L(A^{(1)}), \ldots, L(A^{(t)}), L(A^{(t+1)}), \ldots$ является монотонно возрастающей. Эта же последовательность ограничена сверху значением действительного максимума правдоподобия. Как известно, такая последовательность имеет предел. Равенство

$$\lim_{t\to\infty} [L(A^{(t+1)}) - L(A^{(t)})] = 0 \qquad (14)$$

является необходимым условием существования предела для величины $L(A^{(t)})$. Поэтому равенство (14) справедливо, что и требовалось доказать.

Теорема 1 и следствие из нее отражают важное свойство алгоритма. Тем не менее из них не вытекает непосредственно вывод о сходимости алгоритма, т. е. вывод о существовании предела для величин, непосредственно вычисляемых алгоритмом.

Для доказательства сходимости алгоритма полезной окажется следующая теорема.

Теорема 2. Величина $\alpha_{ik}(A^{(t+1)}) - \alpha_{ik}(A^{(t)})$ $(i = 1, 2, \ldots, m;\ k = 1, 2, \ldots, s)$ с увеличением t стремится к нулю.

Доказательство. Введем обозначение

$$\Delta^{(t)} = L(A^{(t+1)}) - L(A^{(t)}).$$

На основании выражений (7) и (8) можно записать

$$\Delta^{(t)} = \sum_{k=1}^{s}\sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}{p_k^{(t)} \cdot p(v_i/a_k^{(t)})} -$$

$$- \sum_{i=1}^{m}\sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{\alpha_{ik}(A^{(t+1)})}{\alpha_{ik}(A^{(t)})}.$$

В ходе доказательства теоремы 1 выяснилась справедливость неравенства

$$\sum_{k=1}^{s}\sum_{i=1}^{m} \alpha_{ik}(A^{(t)}) \log \frac{p_k^{(t+1)} \cdot p(v_i/a_k^{(t+1)})}{p_k^{(t)} \cdot p(v_i/a_k^{(t)})} > 0.$$

Поэтому справедливо неравенство

$$\sum_{i=1}^{m}\sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{\alpha_{ik}(A^{(t)})}{\alpha_{ik}(A^{(t+1)})} < \Delta^{(t)}.$$

Поскольку в последнем выражении в силу леммы все слагаемые, находящиеся под знаком суммирования по i, неотрицательны, то для всех i имеет место неравенство

$$0 \leqslant \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{\alpha_{ik}(A^{(t)})}{\alpha_{ik}(A^{(t+1)})} < \Delta^{(t)}.$$

На основании последнего выражения, а также в силу того, что $\Delta^{(t)}$ стремится к нулю, можно записать выражение

$$\lim_{t\to\infty} \sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \frac{\alpha_{ik}(A^{(t)})}{\alpha_{ik}(A^{(t+1)})} = 0 \quad (i = 1, 2, \ldots, m). \tag{15}$$

Выражение (15) справедливо лишь в том случае, когда

$$\lim_{t\to\infty} (\alpha_{ik}(A^{(t+1)}) - \alpha_{ik}(A^{(t)})) = 0 \quad (i = 1, 2, \ldots, m; \; k = 1, 2, \ldots, s),$$

так как функция $\sum_{k=1}^{s} \alpha_{ik}(A^{(t)}) \log \alpha_{ik}$ при $\sum_{k=1}^{s} \alpha_{ik} = 1$ имеет единственный максимум, и он достигается при

$$\alpha_{ik} = \alpha_{ik}(A^{(t)}).$$

Теорема доказана.

Для дальнейших рассуждений введем некоторые новые понятия и обозначения.

Совокупность $m \cdot s$ чисел α_{ik} $(i = 1, 2, \ldots, m;$ $k = 1, 2, \ldots, s)$ будем обозначать через $\vec{\alpha}$. Множество всех возможных значений $\vec{\alpha}$ обозначим Ω. Некоторые подмножества множества Ω будем обозначать $\{\vec{\alpha}\}$. Очевидно, одну итерацию алгоритма можно рассматривать как реализацию некоторой функции $F(\vec{\alpha})$, отображающей множество Ω в самого себя.

Будем говорить, что вектор $\vec{\alpha}^{(t)}$ стремится к множеству $\{\vec{\alpha}\}$, если

$$\lim_{t\to\infty} \min_{\vec{\alpha}\in\{\vec{\alpha}\}} (\vec{\alpha}^{(t)} - \vec{\alpha}) = 0.$$

Очевидно, что если множество $\{\vec{\alpha}\}$ содержит единственную точку, то вектор $\vec{\alpha}^{(t)}$ имеет предел в обычном смысле.

Лемма 2. Если вектор $\vec{\alpha}^{(t)}$ стремится к множеству $\{\vec{\alpha}\}$, состоящему из конечного количества точек, а величина $|\vec{\alpha}^{(t+1)} - \vec{\alpha}^{(t)}|$ стремится к нулю, то вектор $\vec{\alpha}^{(t)}$ стремится к одной из точек множества $\{\vec{\alpha}\}$.

Доказательство. Допустим, что вектор $\vec{\alpha}^{(t)}$ не стремится ни к какому элементу множества $\{\vec{\alpha}\}$. В этом случае при сколь угодно большом T и сколь угодно малом ε возможны два такие вектора $\vec{\alpha}^{(t)}$ и $\vec{\alpha}^{(t+1)}$ $(t > T)$, что расстояние между ними будет больше, чем $\delta - 2\varepsilon$, где δ — расстояние между двумя ближайшими элементами множества $\{\vec{\alpha}\}$. Однако это противоречит условию, что расстояние между $\vec{\alpha}^{(t)}$ и $\vec{\alpha}^{(t+1)}$ стремится к нулю.

Лемма доказана.

Будем говорить, что множество $\{\vec{\alpha}\}^{(t)}$ стремится к множеству $\{\vec{\alpha}\}$, если любой вектор $\vec{\alpha}^{(t)} \in \{\vec{\alpha}\}^{(t)}$ стремится к $\{\vec{\alpha}\}$.

Корни уравнения

$$|\vec{\alpha} - F(\vec{\alpha})| = 0, \tag{16}$$

где $F(\vec{\alpha})$ — функция, реализуемая одной итерацией алгоритма, будем называть неподвижными точками алгоритма. Как будет показано ниже, корни уравнения (16) являются корнями системы уравнений правдоподобия, представляющие классические необходимые условия максимума функции правдоподобия.

Рассмотрим уравнение, отличающееся правой частью от уравнения (16):

$$|\vec{\alpha} - F(\vec{\alpha})| = \varepsilon^{(t)}. \tag{17}$$

Уравнению (17) соответствует множество $\{\vec{\alpha}\}^{(t)}$ корней. Мы допустим, что уравнение (16) имеет решение, которое устойчиво в малом относительно правой части, т. е. множество $\{\vec{\alpha}\}^{(t)}$ корней уравнения (17) стремится к множеству $\{\alpha\}$ корней уравнения (16), если $\varepsilon^{(t)}$ стремится к нулю.

Очевидно, что каждый из векторов $\vec{\alpha}^{(1)}$, $\vec{\alpha}^{(2)}, \ldots, \vec{\alpha}^{(t)}, \ldots$ является решением соответственно уравнений $|\vec{\alpha} - F(\vec{\alpha})| = \varepsilon^{(1)}, \varepsilon^{(2)}, \ldots, \varepsilon^{(t)}, \ldots$, причем $\varepsilon^{(t)}$ стремится к нулю. В силу этого можно утверждать, что вектор $\vec{\alpha}^{(t)}$ стремится к множеству корней уравнения (16).

Если допустить, что уравнение (16) имеет конечное число корней, то в силу теоремы 2 и леммы 2 справедливо утверждение, что последовательность векторов $\vec{\alpha}^{(1)}, \vec{\alpha}^{(2)}, \ldots, \vec{\alpha}^{(t)}, \ldots$ стремится к одному из корней уравнения (16).

Следующая теорема определяет свойства корней уравнения (16).

Теорема 3. Если некоторый вектор $\vec{\alpha}$ является неподвижной точкой алгоритма, то величины p_k^* и a_k^*, определяемые на втором этапе одной итерации алгоритма, являются корнями системы уравнений

$$\frac{\partial L(A)}{\partial a_k} = 0 \quad (k = 1, 2, \ldots, s), \tag{18}$$

$$\frac{\partial L(A)}{\partial p_k} - \lambda = 0 \quad (k = 1, 2, \ldots, s), \tag{19}$$

$$\sum_{k=1}^{s} p_k = 1. \tag{20}$$

Доказательство. Введем обозначения:

$$L^{(1)} = \sum_{k=1}^{s} \sum_{i=1}^{m} \alpha_{ik} \log p_k;$$

$$L_k^{(2)} = \sum_{i=1}^{m} \alpha_{ik} \log p(v_i / a_k);$$

$$L_i^{(3)} = \sum_{k=1}^{s} \alpha_{ik} \log \frac{p_k \cdot p(v_i / a_k)}{\sum_{k=1}^{s} p_k \cdot p(v_i / a_k)}.$$

Записанное ранее выражение (8) можно представить в виде

$$L(A) = L^{(1)} + \sum_{k=1}^{s} L_k^{(2)} - \sum_{i=1}^{m} L_i^{(3)}.$$

Поскольку величины $L^{(1)}$ и $L_k^{(2)}$ не зависят соответственно от a_k и p_k, то можно записать:

$$\frac{\partial L(A)}{\partial a_k} = \sum_{k=1}^{s} \frac{\partial L_k^{(2)}}{\partial a_k} - \sum_{i=1}^{m} \frac{\partial L_i^{(3)}}{\partial a_k}, \tag{21}$$

$$\frac{\partial L(A)}{\partial p_k} = \frac{\partial L^{(1)}}{\partial p_k} - \sum_{i=1}^{m} \frac{\partial L_i^{(3)}}{\partial p_k}. \tag{22}$$

Условие теоремы эквивалентно следующим трем условиям:

а) величина a_k^* обеспечивает максимум выражения

$$\sum_{i=1}^{m} \alpha_{ik} \log p(v_i / a_k);$$

б) величина p_k^* пропорциональна величинам $\sum_{i=1}^{m} \alpha_{ik}$, при этом $\sum_{k=1}^{s} p_k^* = 1$;

в) величина α_{ik} равна

$$\alpha_{ik} = \frac{p_k^* \cdot p(v_i / a_k^*)}{\sum_{k=1}^{s} p_k^* \cdot p(v_i / a_k^*)}.$$

В силу условия в) и леммы 1 величина $L_i^{(3)}$ при $a_k = a_k^*$, $p_k = p_k^*$ достигает абсолютного максимума. Поэтому справедливы равенства

$$\frac{\partial L_i^{(3)}}{\partial a_k} = 0, \tag{23}$$

$$\frac{\partial L_i^{(3)}}{\partial p_k} = 0. \tag{24}$$

В силу условия а) величина a_k^* обеспечивает максимум выражения $L_k^{(2)}$. При условии дифференцируемости функции $L_k^{(2)}$ производная $\frac{\partial L_k^{(2)}}{\partial a_k}$ при $a_k = a_k^*$ равна нулю:

$$\frac{\partial L_k^{(2)}}{\partial a_k} = 0. \tag{25}$$

Нетрудно убедиться, что производная $\frac{\partial L^{(1)}}{\partial p_k}$ при $p_k = p_k^*$ (а p_k^* удовлетворяет условию (1)) не зависит от k, т. е. равна некоторой постоянной величине λ:

$$\frac{\partial L^{(1)}}{\partial p_k} = \lambda. \tag{26}$$

На основании уравнений (23) и (25) убеждаемся в равенстве нулю выражения (21), а следовательно, в справедливости уравнения (18).

Аналогичным образом уравнение (19) является следствием равенств (24) и (26).

Уравнение (20) является прямым следствием условия б).

Теорема доказана.

Вследствие доказанной теоремы можно утверждать, что описанный алгоритм сходится к тому значению $\vec{\alpha}^*$, которое является корнем системы уравнений (18) — (20). Вообще говоря, не все корни системы уравнений соответствуют максимумам функции $L(A)$. Некоторые корни соответствуют минимумам или так называемым ложным экстремумам функции $L(A)$. Однако в силу того, что в процессе работы алгоритма оценки параметров могут только улучшаться, алгоритм сходится лишь к той точке, которая соответствует максимуму функции $L(A)$. В этом заключается основное преимущество предлагаемого алгоритма нахождения оценок по сравнению с методом, основанным на непосредственном решении системы уравнений.

В заключение следует отметить, что описанный алгоритм применим и в том случае, когда для каждого образа следует определить не один, а много параметров. Все утверждения статьи, кроме теоремы 3, справедливы вне зависимости от того, что обозначено через a_k: один неизвестный параметр или несколько параметров. Что касается теоремы 3, то она легко может быть сформулирована и доказана для случая многих неизвестных параметров одного образа. Для этого потребовалось бы рассмотрение частной производной не по параметру a_k, а по всем параметрам, входящим в a_k Экспериментальная проверка алгоритма проводилась именно для того случая, когда число неизвестных параметров было очень велико. Описание экспериментов должно быть содержанием отдельной статьи. Здесь лишь мы укажем, что результаты экспериментов оказались положительными.

О ДРУГИХ РЕШЕНИЯХ ЗАДАЧИ САМООБУЧЕНИЯ

Первой из известных нам работ по самообучению является работа [6]. В этой статье, так же как и в [7], не упоминаются слова «самообучение» и «распознавание». Тем не менее задачи, решаемые в этих работах, весьма близки к нашей задаче. Выражая содержание этих работ в терминах нашей статьи, можно утверждать, что рассматривается задача нахождения априорных вероятностей образов по выборке изображений, принадлежащих всей совокупности образов Условные распределения изображений, принадлежащих тому или иному образу, считаются полностью известными. Задача определения априорных вероятностей решена для некоторых частных случаев одномерных распределений. В [6] упоминается о возможности нахождения оценок для априорных вероятностей по методу наибольшего правдоподобия, однако не предложено никакого метода нахождения таких оценок.

Работа [8] содержит четкую постановку задачи самообучения. В этой работе указывается, что в ряде случаев желательно нахождение оценок наибольшего правдоподобия. Для тех случаев, когда получение таких оценок связано с большим объемом вычислений, предлагается использовать метод минимума χ^2. Оценки по этому критерию находятся по методу наискорейшего спуска.

Работы [9] и [10] весьма близки по содержанию и решают задачу самообучения в случае, когда изображения образа распределены по нормальному закону.

В работах [11—13] содержатся различные решения задач самообучения, однако формулировка их существенно отличается от задачи, решаемой в данной статье.

Понятие «самообучение» часто связывается с системами типа персептрон Ф. Розенблатта, так как именно в его работах [14—15] впервые упомянут этот термин, однако в них не содержится четкой постановки задачи и, как показано в [16], задача самообучения не решена.

Результаты этой работы были доложены на заседаниях семинаров «Распознавание образов» и «Теория автоматического управления» научного совета АН УССР по кибернетике. Автор благодарен участникам этих семинаров, чья дружеская критика способствовала более строгому обоснованию основных утверждений.

ЛИТЕРАТУРА

1. Т. Андерсон, Введение в многомерный статистический анализ, ГИФМЛ, М., 1963, стр. 175—183.
2. C. K. Chow, An Optimum Character Recognition System Using Decision Function, IRE Trans. on. El. Computer, EC-6, 1957, p. 247.
3. В. А. Ковалевский, Задача распознавания образов с точки зрения математической статистики, в сб. «Читающие автоматы», изд-во «Наукова думка», К., 1965, стр. 8—37.
4. И. Г. Крамер, Математические методы статистики, ИЛ, 1949.
5. А. С. Барашко и др., Корреляционный читающий автомат со сдвиговым регистром ЧАРС, сб. «Читающие автоматы», изд-во «Наукова думка», К., 1965, стр. 184—207.
6. Г. Роббинс, Асимптотически субминимаксные решения в составных задачах теории статистических решений, журн. «Математика», 1964, 8:2, стр. 141 — 159.

7. Г. Роббинс, Эмпирический байесовский подход к статистике, там же.
8. А. В. Миленький, Определение статистических характеристик распознаваемых образов в режиме самообучения, журн. «Кибернетика», № 3, К., 1967.
9. D. B. Cooper, P. W. Cooper, Nonsupervised Adaptive Signal Detection and Pattern Recognition. Inf. and Control, v. 7, Nu 3, 1964.
10. О. Г. Журавлев, И. Ш. Торговицкий, Оптимальный метод объективной классификации в задачах распознавания образов, журн. «Автоматика и телемеханика», т. XXVI, М., 1965, стр. 2062—2063.
11. М. И. Шлезингер, О самопроизвольном различении образов, сб. «Читающие автоматы», изд-во «Наукова думка», К., 1965, стр. 38—45.
12. А. А. Дорофеюк, Алгоритмы обучения машины распознаванию образов без учителя, основанные на методе потенциальных функций, журн. «Автоматика и телемеханика», № 10, М., 1966, стр. 78—87.
13. Э. М. Браверман, Метод потенциальных функций в задаче обучения машины распознаванию образов без учителя, журн. «Автоматика и телемеханика», № 10, М., 1966, стр. 100—121.
14. F. Rosenblatt, Two Theorems of Statistical Separability in the Perceptron, Symp. Mechaniz. Thought Proc., Nat. Phys. Lab., Teddington, England, 1958.
15. F. Rosenblatt, Perceptron Simulation Experiments. Proc. IRE, v. 48, Nu 3, 1960.
16. В. М. Глушков, К вопросу о самообучении в персептроне, «Журнал вычислительной математики и математической физики», т. 2, № 6, М., 1962.

Поступила в редакцию
20. IV 1967