Р. А. ОГАНЯН

# ПОСТАНОВКА И АЛГОРИФМ РЕШЕНИЯ ЗАДАЧИ РАСПОЗНАВАНИЯ ОБРАЗОВ

## I. Введение

Рассмотрим произвольную последовательность букв в алфавите $A = \{a_i\}_1^n$

$$a_{i_1}, a_{i_2}, \ldots, a_{i_l} \tag{1}$$

и предположим, что вероятность совпадения $a_{i_k}$ с $a_i$ не зависит от $k$ и равна $q_i$.

Допустим, что мы располагаем некоторым оператором $C$, который каждому элементу последовательности (1) сопоставляет некоторый символ.

Пусть $B_i$ — множество всех символов, в которые может перейти буква $a_i \in A$ при отображении ее оператором $C$, а $B = \sum_{i=1}^{n} B_i$.

Букву $a_i$ назовем $i$-тым образом, а элементы множества $B_i$ назовем изображениями $i$-того образа.

Вообще говоря, множества $B_i$ и $B_j$ могут пересекаться.

Для простоты мы предположим, что множество $B$ конечно; перенумеровав его элементы, представим его в виде: $B = \{b_l\}_1^m$.

Последовательность, полученную в результате отображения последовательности (1) оператором $C$, обозначим через

$$b_{j_1}, b_{j_2}, \ldots, b_{j_l}. \tag{2}$$

Теперь, пока не в строгом виде, сформулируем задачу, в которой требуется принятие решения в условиях, вообще говоря, неполной информации.

*Задача*. По каждому символу последовательности (2) распознать породившую его букву.

Далее мы уточним постановку указанной задачи и приведем алгорифм для ее решения.

## 2. Вероятностные автоматы

Рассмотрим вероятностный автомат без памяти $C$, для описания которого введем следующее более подробное обозначение:

$$C = \{A, B, \widetilde{C}\},$$

где $A = \{a_i\}_1^n$ — входной алфавит автомата $C$, $B = \{b_l\}_1^m$ — выходной алфавит автомата $C$, а $\widetilde{C} = \| c_{lj} \|$ — матрица переходов, где $c_{lj}$ — вероятность того, что $a_j$ перейдет в $b_l$, т. е. $c_{lj} = p(b_l / a_j)$. Ясно, что

$$\sum_{l=1}^{m} c_{lj} = 1 \quad (j = 1, \dots, n).$$

Обозначим через $Q$ датчик случайных букв из алфавита $A$, его можно интерпретировать как вероятностный автомат без памяти:

$$Q = \{d, A, \vec{Q}\},$$

где $d$ — единственная буква входного алфавита, $A = \{a_i\}_1^n$ — выходной алфавит, а $\vec{Q} = (q_1, \dots, q_n)$ — распределение вероятностей на выходном алфавите $A$, т. е. $q_i = P(a_i / d)$. Очевидно,

$$\sum_{i=1}^{n} q_i = 1.$$

Испытанием автомата назовем подачу какой-нибудь буквы из его входного алфавита на его вход.

Теперь мы можем несколько уточнить задачу распознавания. Допустим, что в результате $l$ испытаний датчика $Q$ на его выходе появилась некоторая последовательность образов

$$a_{i_1}, a_{i_2}, \dots, a_{i_l}, \tag{3}$$

которая после поступления на вход автомата преобразовалась в последовательность изображений указанных образов:

$$b_{j_1}, b_{j_2}, \dots, b_{j_l}. \tag{4}$$

*Задача.* Исходя из последовательности (4), восстановить последовательность (3).

В условиях неполной информации эта задача не имеет единственного решения, поэтому для ее уточнения нам предстоит еще описать множество допустимых решений и ввести понятие оптимального решения в смысле некоторого критерия, определенного над множеством всех допустимых решений.

Решение задачи, т. е. алгорифм распознаваний, естественно искать в виде вероятностного автомата следующего типа:

$$S = \{B, A, \widetilde{S}\},$$

где $B = \{b_j\}_1^m$ — входной алфавит, $A = \{a_i\}_1^n$ — выходной алфавит, $\overline{S} = \|s_{ij}\|$, где $s_{ij} = P(a_i / b_j)$; причем алфавиты $B$ и $A$ зафиксированы, а $\overline{S}$ — произвольная стохастическая матрица порядка $n \cdot m$, т. е.

$$\sum_{i=1}^{n} s_{ij} = 1 \quad (j = 1, \cdots, m). \tag{5}$$

Множество всех автоматов типа $S$ назовем множеством допустимых решений задачи распознавания и обозначим через $H$.

### 3. Постановка задачи распознавания

Рассмотрим упорядоченную цепочку из трех автоматов:

$$\to Q \to C \to S \to, \tag{6}$$

где выход каждого предыдущего автомата является входом последующего.

Каждое возбуждение датчика $Q$, включенного в цепочку (6), назовем испытанием цепочки (6).

Введем матрицу потерь

$$\overline{A} = \|a_{ij}\|,$$

где $a_{ij}$ — цена потерь в случае, если при испытании цепочки (6) датчик $Q$ выдаст букву $a_j$, а автомат $S$ распознает его как букву $a_i$.

Среднюю потерю при одном испытании цепочки (6), т. е. математическое ожидание потерь, обозначим через $M[Q, C, S, \overline{A}]$.

Эффективность алгоритма распознавания $S$ из $H$ при фиксированных $Q$, $C$, и $\overline{A}$ естественно измерять следующим функционалом:

$$L(S) = \frac{M[Q, C, S, \overline{A}] - \min\limits_{S \in H} M[Q, C, S, \overline{A}]}{\max\limits_{S \in H} M[Q, C, S, \overline{A}] - \min\limits_{S \in H} M(Q, C, S, \overline{A}]}.$$

Очевидно

$$O \leq L(S) \leq 1.$$

Чем меньше средняя потеря в процессе распознавания автоматом $S$, т. е. чем ближе к нулю величина $L(S)$, тем лучше алгорифм распознавания $S$.

*Задача распознавания.* Для любых фиксированных $Q$, $C$ и $\overline{A}$ найти такой автомат $S$ из $H$ (т. е. алгорифм распознавания), который минимизирует функционал $M[Q, C, S, \overline{A}]$.

Автомат $S$ из $H$, минимизирующий функционал $M[Q, C, S, \overline{A}]$, назовем оптимальным.

## 4. Алгорифм нахождения оптимальных автоматов

Введем матрицу $\breve{B} = \| b_{ik} \|$ $(i = 1, \ldots, n;\ k = 1, \ldots, m)$, где

$$b_{ik} = \sum_{j=1}^{n} q_j a_{ij} c_{kj}. \tag{7}$$

Заметим, что $b_{ik}$—математическое ожидание потерь в случае, если всегда $b_k$ распознавать как $a_i$.

Пусть

$$r_k = \min_{1 < i < n} \{ b_{ik} \}. \tag{8}$$

Обозначим через $\varphi(k)$ однозначную функцию, которая отображает множество $\{k\}_{k=1}^{m}$ в множество $\{i\}_{i=1}^{n}$ так, что

$$b_{\varphi(k),\, k} = r_k \ (k = 1, \ldots, m). \tag{9}$$

Определим матрицу $\breve{S}^* = \| s^*_{ik} \|$ порядка $n \cdot m$ по формуле

$$s^*_{ik} = \begin{cases} 1, \text{ если } \varphi(k) = i, \\ 0, \text{ если } \varphi(k) \neq i, \end{cases} \tag{10}$$

т. е.

$$s^*_{\varphi(k),\, k} = 1 \ (k = 1 \ldots, m),$$

а остальные $s^*_{ik}$ равны нулю.

Рассмотрим автомат

$$S^* = \{ B, A, \breve{S}^* \};$$

очевидно $S^* \in H$, причем $S^*$—детерминированный.

*Теорема 1.* Множество всех оптимальных автоматов не пусто. Оно содержит, по крайней мере, один детерминированный автомат. Автомат $S^*$ — оптимальный.

Доказательство. Обозначим через $E$ суперпозицию автоматов $C$ и $S$, тогда автомат $E$ можно описать следующим образом:

$$E = \{ A, A, \breve{E} \},$$

где $A = \{ a_i \}_1^n$ — входной и выходной алфавит, $\breve{E} = \| l_{ij} \|$ — матрица переходов, т. е. $l_{ij} = P(a_i \mid a_j)$.

Очевидно

$$M\,[Q, C, S, \breve{A}) = \sum_{i=1}^{n} \sum_{j=1}^{n} q_j a_{ij} l_{ij}.$$

Легко показать, что

$$\breve{E} = \breve{S} \cdot \breve{C},$$

т. е.

$$l_{ij} = \sum_{k=1}^{m} s_{ik} c_{kj},$$

поэтому

$$M[Q, C, S, \widetilde{A}] = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} q_i a_{ij} s_{jk} c_{kj}$$

или, ввиду (7),

$$M[Q, C, S, \widetilde{A}] = \sum_{i=1}^{n} \sum_{k=1}^{m} b_{ik} s_{ik}. \tag{11}$$

Теперь задача нахождения оптимального автомата сводится к минимизации функционала

$$\sum_{i=1}^{n} \sum_{k=1}^{m} b_{ik} s_{ik}$$

при условиях

$$\sum_{i=1}^{n} s_{ik} = 1 \ (k = 1, \dots, m), \qquad s_{ik} \geqslant 0.$$

Итак, мы пришли к специальной задаче линейного программирования, которая легко решается. Действительно, для любого $S$ из $H$, ввиду (8), (11) и (5), получим

$$M[Q, C, S, \widetilde{A}] > \sum_{i=1}^{n} \sum_{k=1}^{m} r_k s_{ik} = \sum_{k=1}^{m} r_k \sum_{i=1}^{n} s_{ik} = \sum_{k=1}^{m} r_k. \tag{12}$$

С другой стороны, ввиду (11), (10) и (9)

$$M[Q, C, S^*, \widetilde{A}] = \sum_{i=1}^{n} \sum_{k=1}^{m} b_{ik} s^*_{ik} = \sum_{k=1}^{m} b_{\varepsilon(k), k} s^*_{\varepsilon(k), k} = \sum_{k=1}^{m} r_k. \tag{13}$$

Теорема доказана.

*Теорема 2.* Пусть $\widetilde{r}_k = \max\limits_{1 \leq i \leq n} \{ b_{ik} \}$, тогда

$$\min_{S \in H} M[Q, C, S, \widetilde{A}] = \sum_{k=1}^{m} r_k. \tag{14}$$

$$\max_{S \in H} M[Q, C, S, \widetilde{A}] = \sum_{k=1}^{m} \widetilde{r}_k. \tag{15}$$

Доказательство. Формула (14) следует из (12) и (13), а (15) следует из аналогичных рассуждений. Теорема доказана.

Далее, рассмотрим упорядоченную цепочку из четырех автоматов

$$\to Q \to C \to S \to K \to, \tag{16}$$

где $Q$, $C$ фиксированы, $S \in H$, а $K$—произвольный автомат следующего типа

$$K = \{A, A, \widetilde{K}\};$$

здесь $A = \{a_i\}_1^n$ — входной и выходной алфавит, $\widetilde{K} = \| k_{ij} \|$ — матрица переходов, т. е. $k_{ij} = P(a_i / a_j)$.

Теперь возникает следующий вопрос: можно ли при испытаниях цепочки (16) корректировать показания распознающего автомата $S$ некоторым автоматом $K$ так, чтобы уменьшить среднюю потерю при одном испытании цепочки (16). Ответ на этот вопрос дается следующей теоремой.

*Теорема 3.* Если в цепочке (16) автомат $S$ оптимальный, то $K$ не уменьшит среднюю потерю.

Доказательство. Суперпозицию автоматов $S$ и $K$ обозначим через $T$. Очевидно $T \in H$, поэтому для любых автоматов $S$ и $K$

$$M[Q, C, T, \widetilde{A}] \geqslant M[Q, C, S^*, \widetilde{A}].$$

Теорема доказана.

### 5. Другая постановка задачи распознавания

Теорема 1 позволяет сузить множество допустимых решений задачи распознавания.

Обозначим через $F = \{f\}$ множество всех однозначных отображений алфавита $B = \{b_k\}_1^m$ на алфавит $A = \{a_i\}_1^n$.

Каждому $f$ из $F$ сопоставим детерминированный автомат $Sf$ из $H$, для которого матрица переходов $\widetilde{S}_f$ определяется следующим образом:

$$s_{ik}^f = \begin{cases} 1, \text{ если } f(b_k) = a_i, \\ 0, \text{ если } f(b_k) \neq a_i. \end{cases}$$

Теперь, полагая фиксированными автоматы $Q$, $C$ и матрицу потерь $\widetilde{A}$, на множестве $F$ определим функционал $G[Q, C, f, \widetilde{A}]$ по формуле

$$G[Q, C, f, \widetilde{A}] = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} q_j a_{ij} s_{ik}^f c_{kj}. \tag{17}$$

*Задача распознавания.* Для любых фиксированных $Q$, $C$ и $\widetilde{A}$ найти такую функцию $f$ из $F$ (т. е. алгорифм распознавания), которая минимизирует функционал (17).

*Алгорифм распознавания.* Введем функцию расстояния

$$\rho(a_i, b_k) = \sum_{j=1}^{n} q_j a_{ij} c_{kj} \quad (i = 1, \ldots, n; \quad k = 1, \ldots, m).$$

Функцию $f^*$ из $F$ определим следующим образом:

$$f^*(b_k) = a_i,$$

где $i$ (при фиксированном $b_k$)—номер, при котором расстояние $\rho(a_i, b_k$ $(i = 1, \ldots, n)$ достигает минимума.

Рассуждая так же, как и при доказательстве теоремы 1, легко показать, что $f^*$ — оптимальный алгорифм, т. е. минимизирует функционал (17).

## 6. Случай двоичной сетчатки

Обозначим через $L$ множество всех $l$-мерных булевых векторов. Предположим, что между $B=\{b_k\}_1^m$ и $L$ существует взаимно-однозначное соответствие, и вектор из $L$, соответствующий букве $b_k$, обозначим через

$$(\beta_{1k},\ldots,\beta_{ik},\ldots,\beta_{lk}).$$

Рассмотрим матрицу $\|p_{ij}\|$ $(i=1,\ldots,l;\ \ j=1,\ldots,n)$, где $p_{ij}$ — вероятность того, что при подаче на вход автомата $C$ буквы $a_j$ $i$-тая компонента вектора из $L$, соответствующая выходной букве, будет равняться единице.

Предположим, что вероятность того, что при испытании автомата $C$ $i$-тая $(i=1,\ldots,l)$ компонента выходного булева вектора примет значение ноль или единица, не зависит от того, какие значения примут остальные компоненты указанного вектора.

При сделанных предположениях легко показать, что

$$c_{kj}=\prod_{i=1}^{l}[\beta_{ik}p_{ij}+\overline{\beta}_{ik}(1-p_{ij})],$$

где $\overline{\beta}_{ik}$ — отрицание булевой переменной $\beta_{ik}$.

Эта формула позволяет экономить время и память. Действительно, вместо вычисления и запоминания матрицы $\|c_{kj}\|$, т. е. $m\cdot n$ чисел, достаточно вычислить и запомнить матрицу $\|p_{ij}\|$, т. е. $n\cdot\log m$ чисел.

Ռ. Ա. ՕՀԱՆՅԱՆ

### ՊԱՏԿԵՐՆԵՐԻ ՃԱՆԱՉՄԱՆ ԽՆԴՐԻ ԴՐՎԱԾՔԸ ԵՎ ՆՐԱ ԼՈՒԾՄԱՆ ԱԼԳՈՐԻԹՄԸ

Ամփոփում

*Տվյալ հոդվածում բերված է ավտոմատների տեսության և խաղերի տեսության կիրառումը պատկերների ճանաչման խնդրի դրվածքի հարցում, այնուհետև կառուցված է նշված խնդրի լուծման օպտիմալ ալգորիթմը, որը շատ պարզեցված է մի մասնավոր, բայց կարևոր դեպքում։*