С. А. Абрамов, Г. Г. Гнездилова

АЛГОРИТМ УПРАВЛЕНИЯ ВОПРОСНИКОМ В АВТОМАТИЗИРОВАННОЙ ОБУЧАЮЩЕЙ СИСТЕМЕ

1. Вопросником обучающей системы назовем набор вопросов, который задается системой обучаемому. При использовании известного педагогического приема — задания вопросов вразбивку — с каждым выбранным системой вопросом может быть связана следующая цепочка ее действий: вопрос обучаемому; ожидание, прием и проверка его ответа; сообщение обучаемому, правилен или нет его ответ; объявление правильного ответа, если обучаемым был дан неправильный ответ. Нетрудно привести примеры тем, для которых такой подход к обучению выглядит разумно: таблица умножения, исторические даты, иностранные слова (начальные формы слов или, скажем, грамматические формы неправильных глаголов), значения иероглифов, названия созвездий (обучаемому демонстрируются на экране соответствующие картинки), определение на слух музыкальных интервалов — терций, кварт и др.

В процессе проработки такого рода тем естественно многократно повторять вопросы так, чтобы тот из них, который оказывается для обучаемого более трудным, задавался чаще более легкого, а вопрос, про который можно считать, что ответ на него уже усвоен достаточно хорошо, вообще бы исключался из рассмотрения. Предлагается простая стратегия выбора вопросов, которая удовлетворяет сформулированным условиям. В определенном смысле принцип этой стратегии вписывается в теорию адаптивных обучающих систем [1] и в статистическую теорию обучения и контроля знаний [2]. Эта стратегия позволяет проводить занятия и с таким обучаемым, который вообще не знает ответов на вопросы из используемого вопросника: он сможет постепенно запоминать правильные ответы.

Прообразом стратегии служит известный способ заучивания ответов на ряд вопросов с помощью колоды карточек. На каждой из карточек на лицевой стороне написан вопрос, а на обороте — ответ. Из колоды выбирается наугад карточка, и делается попытка ответить на содержащийся в ней вопрос. Если последнее не удается, ответ прочитывается на обороте. Затем карточка возвращается в колоду и все повторяется. На примере колоды карточек предлагаемая стратегия (которую мы будем далее называть алгоритмом управления вопросником) может быть проинтерпретирована следующим образом: выбранная карточка, содержащая вопрос с известным обучаемому ответом, уже не возвращается в колоду; если же правильный ответ обучаемый дать не смог, то, кроме того что в колоду возвращается эта карточка, туда добавляется еще один ее экземпляр.

Сразу подчеркнем, что предлагаемый алгоритм управления вопросником не рассматривает связи между отдельными вопросами (в прежней интерпретации этому соответствует то обстоятельство, что карточки выбираются из колоды наугад).

Пусть вопросы, содержащиеся в вопроснике, пронумерованы числами $1,\ldots,n$. Задание вопросов вразбивку обеспечивается использованием датчика случайных чисел. Пусть имеется датчик, позволяющий получать значения случайной величины, равномерно распределенной на интервале (0, 1). Известна процедура (см., например, [3]), дающая возможность получать исходя из значений ξ значения дискретной случайной величины η с распределением

$$\begin{pmatrix} 1 & 2 & \ldots & n \\ p_1 & p_2 & \ldots & p_n \end{pmatrix},$$

где $0\leqslant p_i\leqslant 1$ $(i=1, 2,\ldots,n)$ и $p_1+\ldots+p_n=1$:

$$\eta=\begin{cases} 1, & \text{если } 0<\xi\leqslant p_1, \\ 2, & \text{если } p_1<\xi\leqslant p_1+p_2, \\ \ldots\ldots & \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ n, & \text{если } p_1+\ldots+p_{n-1}<\xi<p_1+\ldots+p_n. \end{cases}$$

Номер вопроса выбирается случайно, сначала все номера равновероятны и распределение есть

$$\begin{pmatrix} 1 & 2 & \ldots & n \\ 1/n & 1/n & \ldots & 1/n \end{pmatrix}.$$

Пусть самым первым был предложен вопрос, имеющий номер i, $1\leqslant i\leqslant n$. Если на него был получен правильный ответ, то этот вопрос изымается из вопросника и производится переход к распределению

$$\begin{pmatrix} 1 & \ldots & i-1 & i+1 & \ldots & n \\ 1/(n-1) & \ldots & 1/(n-1) & 1/(n-1) & \ldots & 1/(n-1) \end{pmatrix},$$

если же ответ был неправильным, то объявляется правильный ответ и производится переход к распределению

$$\begin{pmatrix} 1 & \ldots & i-1 & i & i+1 & \ldots & n \\ 1/(n+1) & \ldots & 1/(n+1) & 2/(n+1) & 1/(n+1) & \ldots & 1/(n+1) \end{pmatrix},$$

т. е. как бы считается, что вопросник состоит теперь из $n+1$-го вопроса, два из которых совпадают (i-й вопрос входит с кратностью 2). В результате этого трудный вопрос будет предлагаться в дальнейшем чаще, чем более легкие. После каждого полученного ответа распределение меняется так, что если был предложен вопрос, имеющий кратность $k>0$, и на него был получен правильный ответ, то k уменьшается на 1, если же был получен неправильный ответ, то k увеличивается на 1.

Так как в каждом возникающем распределении все вероятности задаются как отношения целых чисел, второе из которых есть суммарная кратность вопросов, то в программной реализации удобнее работать не с вероятностями, а непосредственно с кратностями вопросов

$k_1, \ldots, k_n$ ($k_i \geqslant 0$, $i=1, \ldots, n$). Вместе с $k_1, \ldots, k_n$ рассмотрим и суммарную кратность вопросов $m=k_1+\ldots+k_n$. Пусть $m>0$. Тогда, взяв значение ξ_0 случайной величины ξ, равномерно распределенной на интервале (0, 1), можно вычислить $m\xi_0$ и в качестве очередного предложить вопрос с номером t, где

$$t=\min\left\{l \mid m\xi_0 < \sum_{i=1}^{l} k_i\right\}.$$

(Следует обратить внимание на то, что эта формула исключает возможность выбора вопроса, имеющего кратность 0.) После получения ответа на предложенный вопрос следует изменить k и m. При $m=0$ сеанс обучения оканчивается, и знания (или, соответственно, способность к усвоению, «схватливость») могут быть оценены по числу заданных вопросов.

Приведем точное описание этого алгоритма в виде подпрограммы на языке бейсик [4]. Подпрограмма ВОПРОС (строки 150—210) вычисляет номер очередного вопроса, изменив предварительно кратность вопроса, заданного последним. Взаимодействие с подпрограммой осуществляется через переменные ANS и QNUM. В результате обращения к подпрограмме переменная QNUM получает значение, равное номеру очередного вопроса (значение QNUM=0 показывает, что кратность всех вопросов равна нулю). Перед обращением к подпрограмме переменной ANS должно быть присвоено значение —1, если на последний вопрос был получен правильный ответ, и значение 1 — в противном случае. Если же ни один из вопросов еще не был задан (обращение к подпрограмме выполняется впервые), переменной ANS должно быть присвоено значение 0.

```
100 'ПОДПРОГРАММА ИНИЦИАЛИЗАЦИЯ
110 IF L< >0 THEN ERASE K
120 DIM K(N) : M=N: FOR I=1 TO N: K(I)=1: NEXT
130 RANDOMIZE (VAL (RIGHT$ (TIME$, 2)))
140 RETURN
150 'ПОДПРОГРАММА ВОПРОС
160 IF ANS=0 THEN 190
170 K(QNUM)=K(QNUM)+ANS: M=M+ANS
180 IF M=0 THEN QNUM=0: RETURN
190 KSI=RND(1)*M: S=0
200 FOR I=1 TO N: S=S+K(I)
210 IF KSI<S THEN QNUM=I: RETURN
220 NEXT
```

Подпрограмма ИНИЦИАЛИЗАЦИЯ (строки 100—140) настраивает подпрограмму ВОПРОС на работу с заданным числом вопросов: очередное обращение к этой подпрограмме истолковывается как переход к новому сеансу обучения с вопросником, число вопросов в котором равно заданному значению. Взаимодействие с подпрограммой выполняется через переменные N и L. Перед обращением к подпрограмме переменной N должно быть присвоено значение, равное числу вопросов в вопроснике. Переменной L должно быть присвоено значение 0 при самом первом обращении к подпрограмме и любое другое значение при последующих обращениях.

Переменные M и K, используемые в обеих подпрограммах, не должны изменяться в программе.

Заметим по поводу рассмотренного алгоритма следующее. Пусть обучаемый вначале неправильно ответил на некоторый вопрос, но впоследствии ему все-таки удалось свести кратность вопроса к нулю. Тогда очевидно, что по крайней мере два последних ответа на этот вопрос были правильными.

Если несколько усложнить алгоритм, то можно будет устанавливать первоначальную кратность всех вопросов равной любому натуральному числу s. Алгоритм при этом можно устроить так, что для того, чтобы в результате правильных ответов вопрос получил кратность 0, необходимо по крайней мере s последовательных правильных ответов на этот вопрос.

На примере колоды карточек это обобщение алгоритма можно проинтерпретировать следующим образом. Пусть ответ на вопрос вспомнить не удалось, тогда если число v экземпляров карточек с данным вопросом больше или равно s, то в колоду добавляется еще один экземпляр этой карточки; если же $v<s$, то добавляется сразу $s-v$ экземпляров.

В программной реализации все рассмотренные переменные сохраняют прежний смысл. Дополнительно вводится переменная SINIT, которая должна получать соответствующее значение s для первого обращения к подпрограмме ИНИЦИАЛИЗАЦИЯ.

Строка с номером 120 изменяется следующим образом:

```
120 DIM K(N): M=N*SINIT: FOR I=1 TO N: K(I)=SINIT:
NEXT
```

В подпрограмму ВОПРОС добавляется еще одна строка:

```
165 IF ANS=1 AND K(QNUM)<SINIT THEN ANS=SINIT—
K(QNUM)
```

2. Проведем математическое исследование описанного алгоритма. Пусть в некоторый момент только один вопрос (будем считать, что это вопрос с номером 1) имеет кратность 1, а все остальные — кратность большую, чем 1 (вопросы с кратностью 0 не принимаются во внимание). Пусть, начиная с этого момента, обучаемый стабильно дает неправильные ответы на предлагаемые в соответствии с данным алгоритмом вопросы. Тогда, как нетрудно показать, среднее значение, т. е. математическое ожидание, времени, проходящего до появления вопроса с номером 1, равно бесконечности (считаем, что время измеряется количеством заданных вопросов). В самом деле, пусть m — суммарная кратность, а n — общее количество вопросов, $m>n\geqslant 2$. Вероятность получить вопрос с номером 1 после i вопросов с другими номерами при $i\geqslant 0$ есть

$$\frac{m-1}{m}\,\frac{m}{m+1}\cdots\frac{m+i-2}{m+i-1}\,\frac{1}{m+i}=\frac{m-1}{(m+i-1)(m+i)},$$

и упомянутое среднее представляется расходящимся рядом

$$\sum_{i=0}^{\infty}\frac{m-1}{(m+i-1)(m+i)}(i+1)=(m-1)\sum_{i=0}^{\infty}\frac{i+1}{(m+i-1)(m+i)}.$$

В этом вычислении предполагалось, что неправильный ответ на заданный вопрос увеличивает суммарную кратность вопросов на 1. При использовании второго (более общего) варианта алгоритма возможны скачки, приводящие к бо́льшим увеличениям. Однако такого рода скачков при стабильных неправильных ответах может быть лишь конечное число, и их можно не учитывать в исследованиях сходимости рядов.

Пусть теперь u вопросов (будем считать, что это вопросы с номерами $1, \ldots, u$) имеют кратность 1, а остальные $n-u$ вопросов — кратность бо́льшую, чем 1, и пусть $u \geqslant 2$. Оказывается, что при стабильных неправильных ответах обучаемого на задаваемые вопросы среднее значение времени, проходящего до получения всех, кроме какого-то одного, вопросов с номерами $1, \ldots, u$, есть конечная величина. Докажем конечность среднего времени, проходящего до получения одного вопроса с номером от 1 до u, далее можно применить индукцию. Вероятность получить такого рода вопрос после i вопросов с номерами из диапазона от $u+1$ до n есть

$$\frac{m-u}{m} \frac{m-u+1}{m+1} \ldots \frac{m-u+i-1}{m+i-1} \frac{u}{m+i}.$$

Эта вероятность при $i>u$ равна

$$\frac{u(m-u)(m-u+1)\ldots(m-1)}{(m-u+i)(m-u+i+1)\ldots(m+i-1)(m+i)},$$

поэтому искомое среднее есть сумма некоторого конечного числа слагаемых и ряда

$$\sum_{i=u+1}^{\infty} \frac{u(m-u)(m-u+1)\ldots(m-1)}{(m-u+i)(m-u+i+1)\ldots(m+i-1)\ldots(m+i)}(i+1)=$$

$$=u(m-u)(m-u+1)\ldots(m-1) \sum_{i=u+1}^{\infty} \frac{i+1}{(m-u+i)\ldots(m+i)};$$

последний ряд сходится, так как $u \geqslant 2$.

Аналогичным образом можно установить, что если в некоторый момент каждый из вопросов имеет кратность большую единицы, то среднее время ожидания всех вопросов конечно.

Все это показывает, что предложенный алгоритм является алгоритмом задания вопросов вразбивку (а не алгоритмом последовательной проработки вопросов) со следующей характерной чертой: если обучаемому удалось дать ряд правильных ответов на один из вопросов, понизив кратность этого вопроса до единицы, и если добиться этого ему удалось на фоне плохих ответов на все остальные вопросы с ненулевой кратностью, то дальнейшее рассмотрение этого вопроса будет как бы отложено до тех пор, пока обучаемый не усвоит еще по крайней мере один вопрос, сведя и его кратность к единице. Алгоритм препятствует слишком быстрому изъятию из рассмотрения отдельных вопросов. В особенности это касается второго варианта алгоритма при $s \geqslant 2$.

СПИСОК ЛИТЕРАТУРЫ

1. Савельев А. Я., Новиков В. А., Лобанов Ю. И. Подготовка информации для автоматизированных обучающих систем. М., 1986.
2. Свиридов А. П. Основы статистической теории обучения и контроля знаний. М., 1981.
3. Кнут Д. Искусство программирования для ЭВМ. Т. 2. М., 1977.
4. Пул Л. Работа на персональном компьютере. М., 1986.

Поступила в редакцию
12.06.87