МАТЕМАТИКА 2012. Вып. 2

УДК 510.5+512.6

© *А. Ю. Сапаров, А. П. Бельтюков*

ПРИМЕНЕНИЕ РЕГУЛЯРНЫХ ВЫРАЖЕНИЙ В РАСПОЗНАВАНИИ МАТЕМАТИЧЕСКИХ ТЕКСТОВ

Работа посвящена использованию регулярных выражений при распознавании рукописных математических текстов. Основная проблема в распознавании рукописных математических формул состоит в том, что эти тексты, как правило, состоят из большого числа маленьких фрагментов, расположенных в соответствии с некоторыми строгими правилами. Несмотря на то, что формальное определение синтаксиса математических текстов может вовлекать бесконтекстные грамматики и даже более сложные конструкции, на практике часто для успешного распознавания достаточно определения математического языка на базе регулярных выражений. Поскольку некоторые конструкции в математических текстах могут встречаться чаще других, мы вводим понятие взвешенного регулярного выражения. Веса в нём определяют предпочтение одних конструкций перед другими. В работе вводится математический аппарат для использования таких выражений при распознавании. В частности, доказываются теоремы о пересечении взвешенных множеств, задаваемых такими регулярными выражениями. Даются некоторые оценки сложности работы алгоритмов использующих такие регулярные выражения для распознавания.

Ключевые слова: регулярные множества, регулярные выражения, регулярные операции.

Введение

В ходе решения некоторых задач в различных областях в результате могут быть получены некоторые неопределенности, то есть решение не удается найти однозначно. Но такой результат не всегда допустим. Например, при оптическом распознавании текстов некоторые символы не удается правильно распознать из-за того, что они плохо пропечатались или написаны плохим почерком (если текст рукописный). В таких случаях рассматриваются не отдельные символы, а сразу некоторые выражения, например, отдельные слова. В качестве верно распознанного слова берется то, которое содержится в специальном словаре.

Данный метод может быть использован, если есть возможность выделения конечного словаря, но это не всегда удается. Например, при распознавании математических формул множество всевозможных выражений содержит очень большое число элементов, поэтому невозможно всех их перечислить. Но типов математических формул конечное число, поэтому есть возможность сгруппировать их по некоторым признакам. Для задания такого множества можно воспользоваться записью, называемой регулярным выражением. Эта технология позволяет с помощью выражений конечной длины задавать множества, состоящие из бесконечного числа элементов, что дает возможность использования бесконечных множеств на практике.

§ 1. Регулярные множества и регулярные выражения

Рассмотрим класс множеств слов над конечным алфавитом, которые можно описать с помощью формул приведенного ниже вида. Данные множества описаны в работе [1] и называются регулярными.

Определение 1. Пусть V_1 и V_2 — множества цепочек. Определим три операции на этих множествах.

1. *Объединение*: $V_1 \bigcup V_2 = \{\alpha | \alpha \in V_1 \text{ или } \alpha \in V_2\}$.
2. *Конкатенация*: $V_1 V_2 = \{\alpha\beta | \alpha \in V_1, \beta \in V_2\}$.
3. *Итерация*: $V^* = V^0 \bigcup V^1 \bigcup V^2 \bigcup \ldots = \bigcup\limits_{n=0}^{\infty} V^n$, где V^0 состоит из пустого слова, $V^1 = V$, $V^n = VV \ldots V$ (n раз).

Определение 2. Класс *регулярных множеств* над конечным алфавитом V определяется следующим образом:

1. $\emptyset$ — регулярное множество;
2. $\{\epsilon\}$(ϵ — пустое слово) — регулярное множество;
3. $\{a\}$ — регулярное множество для любого a из V (a считается также однобуквенным словом);
4. Если S и T — регулярные множества, то регулярны следующие множества:
— объединение $S \bigcup T$;
— конкатенация ST;
— итерации S^* и T^*.

Определение 3. Класс *регулярных выражений* над конечным алфавитом V определяется следующим образом:

1. $\emptyset$ и ϵ — регулярные выражения;
2. a — регулярное выражение для любого a из V (a считается также однобуквенным словом);
3. Если R и S — регулярные выражения, регулярными выражениями являются следующие выражения:
— их объединение $(R|S)$;
— их произведение (RS);
— их итерации $(R)^*$ и $(S)^*$.
4. Если выражение является регулярным, то оно построено конечным числом применения правил 1–3.

Для уменьшения числа скобок, как и в любой алгебре, используются приоритеты операций: итерация самая приоритетная; менее приоритетно произведение; самый низкий приоритет у объединения. Операцию объединения регулярных выражений иногда называют суммой.

§ 2. Свойства регулярных выражений

Определение 4. Два регулярных выражения $R1$ и $R2$ называются *эквивалентными* (обозначается $R1 = R2$) тогда и только тогда, когда равны их соответствующие регулярные множества.

Для любых регулярных выражений R, S и T справедливо:

$$\begin{aligned} &R|S = S|R, && R\epsilon = \epsilon R = R,\\ &R|R = R, && R(ST) = (RS)T,\\ &R|(S|T) = (R|S)|T, && \emptyset R = R\emptyset = \emptyset,\\ &\emptyset|R = R, && R(S|T) = RS|RT. \end{aligned}$$

§ 3. Описание простой задачи распознавания

Простейшая из рассматриваемых здесь задач распознавания формулируется следующим образом. Пусть дано конечное непустое множество $S = \{s_1, s_2, \ldots, s_m\}$ и регулярные множества $P_1, P_2, \ldots, P_n$, заданные регулярными выражениями. Для каждого элемента множества S задан числовой вес q_j, $j = 1, 2, \ldots, m$. Для каждого P_i, где $i = 1, 2, \ldots, n$, также задан соответствующий числовой вес w_i. Считаем, что веса принимают рациональные значения, а множество S задано регулярным выражением S^r, содержащим только операции объединения и произведения:

$S^r = (a_{1,1}|a_{1,2}|\ldots|a_{1,t_1})(a_{2,1}|a_{2,2}|\ldots|a_{2,t_2})\ldots(a_{l,1}|a_{l,2}|\ldots|a_{l,t_l})$.

Пусть $P = \bigcup\limits_{i=1}^{n} P_i$, $\quad S_j = \{s_{j_1}, s_{j_2}, \ldots, s_{j_r}\} = P \bigcap S$, $\quad M = \max\limits_{k=1,2,\ldots,r} \Big\{ q_{j_k} \cdot \sum\limits_{i: s_{j_k} \in P_i} w_i \Big\}$.

Требуется найти $s_{j_k} \in P \bigcap S$, для которого выполнено условие:
$q_{j_k} \cdot \sum\limits_{i:s_{j_k} \in P_i} w_i = M.$

Задачи такого рода реально возникают при распознавании математических выражений.

§ 4. Описание алгоритма

Опишем кратко идею алгоритма решения поставленной задачи.

Пусть $A^i = a_{i,1}|a_{i,2}|\dots|a_{i,t_i}$. Тогда $S^r = A^1 A^2 \dots A^l$.

Пусть $P_1^r, P_2^r, \dots, P_n^r$ — регулярные выражения соответственно для $P_1, P_2, \dots, P_n$. Регулярные выражения имеют конечную длину, так как по определению они могут быть построены только применением конечного числа регулярных операций, поэтому для каждого P_j^r можем построить конечное множество $\{p_{j,1}, p_{j,2}, \dots, p_{j,k_j}\}$, где каждый элемент $p_{j,i}$ — отдельный символ, содержащийся в данном регулярном выражении.

Найдем пересечения множеств:

$A^{i,j} = \{a_{i,1}, a_{i,2}, \dots |a_{i,t_i}\} \bigcap \{p_{j,1}, p_{j,2}, \dots, p_{j,k_j}\}$ $i = 1, 2, \dots l, j = 1, 2, \dots n.$

Для каждого j строим новое регулярное выражение $S_j^r = A^{1,j} A^{2,j} \dots A^{l,j}$, которое задает новое регулярное множество S', где $S' \subset S$. Таким образом, из множества S были исключены те элементы, которые не входят в P_j, так как составлены из символов, не содержащихся в соответствующем регулярном выражении P_j^r.

Из проведенных построений следует:

(1) Если $\exists i\ A^{i,j} = \emptyset$, тогда $S \bigcap P_j = \emptyset$.

(2) Если $\forall i\ A^{i,j} \neq \emptyset$, тогда $S \bigcap P_j = S' \bigcap P_j$.

(3) $P \bigcap S = \bigcup\limits_{j=1}^{n} (S' \bigcap P_j)$.

В результате применения данного алгоритма получается множество S', содержащее только те элементы, которые нас интересуют в данной задаче. Кроме того, были исключены те регулярные множества P_j, которые также по условию задачи нас не интересуют.

В случае, если $\forall j((S' \bigcap P_j) = \emptyset)$ $(P \bigcap S = \emptyset)$, задача в данном состоянии решения не имеет. Для решения задачи при $P \bigcap S \neq \emptyset$ необходимо вычислить значение

$$M = \max_{k=1,2,\dots,r} \Big\{ q_{j_k} \cdot \sum_{i:s_{j_k} \in P_i} w_i \Big\}.$$

Для этого вычислим значения $M_k = q_{j_k} \cdot \sum\limits_{i:s_{j_k} \in P_i} w_i$, $k = 1, 2, \dots, r$. Эти равенства можно записать в следующем виде: $M_k = q_{j_k} \cdot \sum\limits_{i:S' \bigcap P_i \neq \emptyset} w_i$, в котором есть возможность непосредственного вычисления значения. При этом следует решить задачу построения пересечения регулярных множеств, результатом которой будет нахождение множества S_j. Решением же исходной задачи является $s = s_{j_k} \in S_j$, где $M_k = M$.

§ 5. Пересечение регулярных множеств

Алгоритм нахождения пересечения двух регулярных множеств непосредственно базируется на теоремах, доказанных в настоящем разделе.

Определение 5. *Делением множества слов на множество слов слева* называется:

$$B \backslash A = \{w | \exists v \in B\ , vw \in A\}.$$

Теорема 1 (о делении слева на символ множеств, заданных регулярными выражениями).

$$\begin{array}{lll} a\backslash(e_1|e_2) = (a\backslash e_1|a\backslash e_2) & a\backslash e^* = (a\backslash e)e^* & a\backslash\epsilon = \emptyset \\ a\backslash(e_1 e_2) = \begin{cases} ((a\backslash e_1)e_2|a\backslash e_2),\ \epsilon \in e_1 \\ ((a\backslash e_1)e_2),\ \epsilon \notin e_1 \end{cases} & a\backslash a = \epsilon & a\backslash b = \emptyset,\ a \neq b. \end{array} \qquad (5.1)$$

Д о к а з а т е л ь с т в о.

1) Пусть $e = (e_1|e_2)$. Докажем, что $a\backslash(e_1|e_2) = (a\backslash e_1|a\backslash e_2)$.

а) Пусть $x \in a\backslash(e_1|e_2)$; докажем, что $x \in (a\backslash e_1|a\backslash e_2)$. Из $x \in a\backslash(e_1|e_2)$ следует, что $ax \in (e_1|e_2)$. Откуда получаем, что $ax \in e_1$ либо $ax \in e_2$. Из $ax \in e_1$ следует $x \in a\backslash e_1$, а из $ax \in e_2$ следует $x \in a\backslash e_2$. Объединяя результаты, получаем $x \in (a\backslash e_1|a\backslash e_2)$. Следовательно, $a\backslash(e_1|e_2) \subseteq (a\backslash e_1|a\backslash e_2)$.

б) Пусть $x \in (a\backslash e_1|a\backslash e_2)$. Докажем, что $x \in a\backslash(e_1|e_2)$. Из $x \in (a\backslash e_1|a\backslash e_2)$ следует, что $x \in a\backslash e_1$ либо $x \in a\backslash e_2$. Из $x \in a\backslash e_1$ следует $ax \in e_1$, а из $x \in a\backslash e_2$ следует $ax \in e_2$. Объединяя результаты, получаем $ax \in (e_1|e_2)$, или $x \in a\backslash(e_1|e_2)$. Следовательно, $(a\backslash e_1|a\backslash e_2) \subseteq a\backslash(e_1|e_2)$.

Из $a\backslash(e_1|e_2) \subseteq (a\backslash e_1|a\backslash e_2)$ и $(a\backslash e_1|a\backslash e_2) \subseteq a\backslash(e_1|e_2)$ следует, что $a\backslash(e_1|e_2) = (a\backslash e_1|a\backslash e_2)$.

2) Пусть $e = a\backslash(e_1e_2)$. Докажем, что $a\backslash(e_1e_2) = \begin{cases} ((a\backslash e_1)e_2|a\backslash e_2), & \epsilon \in e_1, \\ ((a\backslash e_1)e_2), & \epsilon \notin e_1. \end{cases}$

I. Рассмотрим случай $\epsilon \in e_1$. Докажем, что $a\backslash(e_1e_2) = ((a\backslash e_1)e_2|a\backslash e_2)$.

а) Пусть $x \in a\backslash(e_1e_2)$. Докажем, что $x \in ((a\backslash e_1)e_2|a\backslash e_2)$. Из $x \in a\backslash(e_1e_2)$ следует, что $ax \in e_1e_2$. Очевидно, что $a\backslash e_1 = \emptyset$ либо $a\backslash e_1 \neq \emptyset$. Если $a\backslash e_1 = \emptyset$, то $ax \in e_2$, так как $\epsilon \in e_1$, или $x \in a\backslash e_2$. Если $a\backslash e_1 \neq \emptyset$, то $x \in (a\backslash e_1)e_2$ либо $x \in a\backslash e_2$. Объединяя все полученные результаты, получаем $x \in ((a\backslash e_1)e_2|a\backslash e_2)$. Следовательно, $a\backslash(e_1e_2) \subseteq ((a\backslash e_1)e_2|a\backslash e_2)$.

б) Пусть $x \in ((a\backslash e_1)e_2|a\backslash e_2)$. Докажем, что $x \in a\backslash(e_1e_2)$. Из $x \in ((a\backslash e_1)e_2|a\backslash e_2)$ следует, что $x \in (a\backslash e_1)e_2$ либо $x \in a\backslash e_2$. Из $x \in (a\backslash e_1)e_2$ получаем $ax \in a(a\backslash e_1)e_2$, или $ax \in e_1e_2$. Из $x \in a\backslash e_2$ получаем $ax \in e_2$, но так как $\epsilon \in e_1$, то можем написать $ax \in e_1e_2$. В результате как при $x \in (a\backslash e_1)e_2$, так и при $x \in a\backslash e_2$ получаем $ax \in e_1e_2$, то есть $x \in a\backslash(e_1e_2)$. Следовательно, $((a\backslash e_1)e_2|a\backslash e_2) \subseteq a\backslash(e_1e_2)$. Из $a\backslash(e_1e_2) \subseteq ((a\backslash e_1)e_2|a\backslash e_2)$ и $((a\backslash e_1)e_2|a\backslash e_2) \subseteq a\backslash(e_1e_2)$ получаем $a\backslash(e_1e_2) = ((a\backslash e_1)e_2|a\backslash e_2)$.

II. Рассмотрим случай $\epsilon \notin e_1$. Докажем, что $a\backslash(e_1e_2) = (a\backslash e_1)e_2$.

а) Пусть $x \in a\backslash(e_1e_2)$. Докажем, что $x \in (a\backslash e_1)e_2$. Из $x \in a\backslash(e_1e_2)$ следует, что $ax \in e_1e_2$. Очевидно, что $a\backslash e_1 \neq \emptyset$, иначе $a\backslash(e_1e_2)$ было бы пустым, так как $\epsilon \notin e_1$. Следовательно, как и в I (при $a\backslash e_1 \neq \emptyset$), получаем $x \in (a\backslash e_1)e_2$. В итоге получено, что $a\backslash(e_1e_2) \subseteq (a\backslash e_1)e_2$.

б) Пусть $x \in (a\backslash e_1)e_2$. Докажем, что $x \in a\backslash(e_1e_2)$. Из $x \in (a\backslash e_1)e_2$ получаем $ax \in a(a\backslash e_1)e_2$, то есть $ax \in e_1e_2$ или $x \in \backslash(e_1e_2)$. Следовательно, $(a\backslash e_1)e_2 \subseteq a\backslash(e_1e_2)$.

Из $a\backslash(e_1e_2) \subseteq (a\backslash e_1)e_2$ и $(a\backslash e_1)e_2 \subseteq a\backslash(e_1e_2)$ получаем $a\backslash(e_1e_2) = (a\backslash e_1)e_2$.

Из I и II следует, что $a\backslash(e_1e_2) = \begin{cases} ((a\backslash e_1)e_2|a\backslash e_2), \epsilon \in e_1 \\ ((a\backslash e_1)e_2), \epsilon \notin e_1 \end{cases}$

3) Пусть $e = \epsilon$. Из определения следует $a\backslash\epsilon = \{w|\exists v \in a\ , vw \in \epsilon\}$. Но vw не может быть пустой цепочкой, так как $v \neq \epsilon$. Следовательно $a\backslash\epsilon = \emptyset$.

4) Пусть e является итерацией. Докажем, что $a\backslash e^* = (a\backslash e)e^*$. $a\backslash e^* = a\backslash((e)e^*|\epsilon) = (a\backslash((e)e^*)|(a\backslash\epsilon)) = (a\backslash((e)e^*)|\emptyset) = a\backslash((e)e^*) = (a\backslash e)e^*$. Что и требовалось доказать.

5) Пусть $e = a$, то есть $e = \{a\}$. Докажем, что $a\backslash a = \epsilon$. Из определения следует $a\backslash a = \{w|\exists v \in a\ , vw \in a\} = \epsilon$, так как $v = a$ и $vw = a$.

6) Пусть $e = b$, $b \neq a$, то есть $e = \{b\}$. Докажем, что $a\backslash a = \emptyset$. Из определения следует $a\backslash b = \{w|\exists v \in a\ , vw \in b\} = \emptyset$, так как $v = a$ и $vw = b$.

Теорема доказана. □

Теорема 2 (о пересечении двух множеств, заданных регулярными выражениями).

$$\Big(\sum_i b_i \cdot C\Big) \bigcap A = \sum_i \Big(b_i(C \bigcap (b_i \backslash A))\Big). \tag{5.2}$$

Доказательство.

I. Рассмотрим пересечение двух регулярных множеств: $(a|b)C \bigcap A$. Пусть $x \in (a|b)C \bigcap A$. Докажем, что $x \in a(C \bigcap (a\backslash A))|b(C \bigcap (b\backslash A))$.

Из $x \in (a|b)C \bigcap A$ следует, что $x \in (a|b)C$ и $x \in A$. Из $x \in (a|b)C$ получаем $x \in aC|bC$, из которого следует $x \in aC$ либо $x \in bC$.

Пусть $x \notin a(C \bigcap (a\backslash A))|b(C \bigcap (b\backslash A))$. Тогда $x \notin a(C \bigcap (a\backslash A))$ и $x \notin b(C \bigcap (b\backslash A))$.

Рассмотрим два случая: 1) $x \in aC$ и 2) $x \in bC$.

1) Так как $x \in aC$, то $x = ax_1$, где $x_1 \in C$. Из $x \in A$ получаем $ax_1 \in A$. По определению $a\backslash A = \{y | ay \in A\}$, откуда следует, что $x_1 \in a\backslash A$.

Из $x \notin a(C \bigcap (a\backslash A))$ получаем $ax_1 \notin a(C \bigcap (a\backslash A)) \implies x_1 \notin C \bigcap (a\backslash A) \implies x_1 \notin C$ или $x_1 \notin a\backslash A$, где в обоих случаях получается противоречие, значит $x \notin aC$.

2) Рассмотрим случай $x \in bC$. Аналогично получаем $x = bx_1$, где $x_1 \in C$ и $x_1 \in b\backslash A$.

Из $x \notin b(C \bigcap (b\backslash A))$ получаем $x_1 \notin C$ или $x_1 \notin b\backslash A$. Таким образом, и при $x \in bC$ получается противоречие. Следовательно, $x \in a(C \bigcap (a\backslash A))|b(C \bigcap (b\backslash A))$. По свойству суммы получаем:

$\forall x \left(x \in \left(\sum_i b_i \cdot C\right) \bigcap A \Rightarrow x \in \sum_i \left(b_i(C \bigcap (b_i \backslash A))\right)\right)$, что означает

$\left(\sum_i b_i \cdot C\right) \bigcap A \subseteq \sum_i \left(b_i(C \bigcap (b_i \backslash A))\right)$.

II. Рассмотрим сумму: $a(C \bigcap (a\backslash A))|b(C \bigcap (b\backslash A))$. Пусть $x \in a(C \bigcap (a\backslash A))|b(C \bigcap (b\backslash A))$. Докажем, что $x \in (a|b)C \bigcap A$.

Из $x \in a(C \bigcap (a\backslash A))|b(C \bigcap (b\backslash A))$ получаем $x \in a(C \bigcap (a\backslash A))$ либо $x \in b(C \bigcap (b\backslash A))$.

Рассмотрим 2 случая:

1) Пусть $x \in a(C \bigcap (a\backslash A))$. Тогда x можно представить в следующем виде: $x = ax_1$. Имеем $ax_1 \in a(C \bigcap (a\backslash A)) \implies x_1 \in C \bigcap (a\backslash A) \implies x_1 \in C$ и $x_1 \in a\backslash A$.

Из $x_1 \in C$ получаем $ax_1 \in aC$, $x \in aC$; далее, применяя свойство суммы, получаем: $x \in aC|bC$ и $x \in (a|b)C$. Из $x_1 \in a\backslash A$ получаем $ax_1 \in A$, или $x \in A$. Так как $x \in (a|b)C$ и $x \in A$, то $x \in (a|b)C \bigcap A$.

2) Пусть $x \in b(C \bigcap (b\backslash A))$. Тогда x можно представить в следующем виде: $x = bx_1$. Имеем $bx_1 \in b(C \bigcap (b\backslash A)) \implies x_1 \in C \bigcap (b\backslash A) \implies x_1 \in C$ и $x_1 \in b\backslash A$.

Из $x_1 \in C$ получаем $bx_1 \in bC$, $x \in bC$, далее, применяя свойство суммы, получаем: $x \in aC|bC$ и $x \in (a|b)C$. Из $x_1 \in b\backslash A$ получаем $bx_1 \in A$, или $x \in A$. Так как $x \in (a|b)C$ и $x \in A$, то $x \in (a|b)C \bigcap A$. Так как в обоих случаях получен результат $x \in (a|b)C \bigcap A$, то по свойству суммы получаем $\forall x \left(x \in \sum_i \left(b_i(C \bigcap (b_i \backslash A))\right) \Rightarrow x \in \left(\sum_i b_i \cdot C\right) \bigcap A\right)$, что означает $\sum_i \left(b_i(C \bigcap (b_i \backslash A))\right) \subseteq \left(\sum_i b_i \cdot C\right) \bigcap A$. Из вложенности двух множеств в друг друга, которые доказаны в I и II, получаем: $\left(\sum_i b_i \cdot C\right) \bigcap A = \sum_i \left(b_i(C \bigcap (b_i \backslash A))\right)$, что и требовалось доказать.

§ 6. Оценка сложности вычислений

Оценим сложность вычислений предложенного алгоритма. Для этого рассмотрим пересечение регулярных множеств:

$$(\sum_{i_1=1}^{n_1} b_{i_1}^1)(\sum_{i_2=1}^{n_2} b_{i_2}^2) \ldots (\sum_{i_m=1}^{n_m} b_{i_m}^m) \bigcap A \sim \sum_{i_1=1}^{n_1} b_{i_1}^1 (\sum_{i_2=1}^{n_2} b_{i_2}^2 (\ldots (\sum_{i_m=1}^{n_m} b_{i_m}^m (\ldots))) \sim \prod_{i=1}^{m} n_i.$$

Возникает следующая задача:

$$\begin{cases} f(m, n_1, \ldots, n_m) \to \max, \\ \text{где } f(m, n_1, \ldots, n_m) = \prod\limits_{i=1}^{m} n_i, \quad \sum\limits_{i=1}^{m} n_i = C, \quad n_i, m \in \mathbb{N}^+. \end{cases} \tag{6.1}$$

Для решения данной задачи рассмотрим другую задачу: $\left(\frac{c}{m}\right)^m \to \max$,

решением которой является $m = \frac{C}{e}$.

Очевидно, что решением нашей задачи будет либо $m = \frac{C}{2}, n_i = 2$, либо $m = \frac{C}{3}, n_i = 3$, так как 2 и 3 являются ближайшими натуральными числами к e. Но это верно только при $\frac{C}{2} \in \mathbb{Z}$ и $\frac{C}{3} \in \mathbb{Z}$ соответственно.

Рассмотрим случаи, когда $\frac{C}{2} \in \mathbb{R}$ и $\frac{C}{3} \in \mathbb{R}$. В первом случае получаем:

$$n_i = 2, \quad i = 1, \ldots, m-1; \quad m = \begin{cases} \frac{C}{2}, \frac{C}{2} \in \mathbb{Z} \\ \left[\frac{C}{2}\right] + 1, \frac{C}{2} \notin \mathbb{Z} \end{cases}; \quad n_m = \begin{cases} 2, \frac{C}{2} \in \mathbb{Z} \\ 1, \frac{C}{2} \notin \mathbb{Z} \end{cases}.$$

Во втором случае:

$$n_i = 3, \quad i = 1, \ldots, m-1; \quad m = \begin{cases} \frac{C}{3}, \frac{C}{3} \in \mathbb{Z} \\ \left[\frac{C}{3}\right] + 1, \frac{C}{3} \notin \mathbb{Z} \end{cases}; \quad n_m = \begin{cases} 3, \frac{C}{3} \in \mathbb{Z} \\ C - 3\left[\frac{C}{3}\right], \frac{C}{3} \notin \mathbb{Z} \end{cases}.$$

Вычислим значения для $\prod\limits_{i=1}^{m} n_i$ в каждом из случаев.

(1) $\prod\limits_{i=1}^{m} n_i = 2^{\left[\frac{C}{2}\right]}$.

(2) $\prod\limits_{i=1}^{m} n_i = 3^{\left[\frac{C}{3}\right]} \cdot r(n_m)$, где $r(n_m) = \begin{cases} 1, \frac{C}{3} \in \mathbb{Z} \\ C - 3\left[\frac{C}{3}\right], \frac{C}{3} \notin \mathbb{Z} \end{cases}$.

Так как $2^{\frac{C}{2}} < 3^{\frac{C}{3}}$, то очевидно, что $\forall i(n_i = 3)$ — сложность вычислений по времени является максимальной. Пусть $A = C$, тогда

$$||C \bigcap A|| = ||(\sum_{i_1=1}^{n_1} b_{i_1}^1)(\sum_{i_2=1}^{n_2} b_{i_2}^2)\ldots(\sum_{i_m=1}^{n_m} b_{i_m}^m) \bigcap (\sum_{i_1=1}^{n_1} b_{i_1}^1)(\sum_{i_2=1}^{n_2} b_{i_2}^2)\ldots(\sum_{i_m=1}^{n_m} b_{i_m}^m)||$$

$$= ||\sum_{i_1=1}^{n_1} b_{i_1}^1 (\sum_{i_2=1}^{n_2} b_{i_2}^2 (\ldots (\sum_{i_m=1}^{n_m} b_{i_m}^m)))|| \sim \prod_{i=1}^{m} n_i.$$

При $A = C$ и $\forall i(n_i = 3)$ — сложность по использованию памяти является максимальной.

§ 7. Взвешенные регулярные множества и взвешенные регулярные выражения

Определение 6. *Множество взвешенных слов* [2] из V — это функция $\varphi : V \longrightarrow \mathbb{R}^+$. В этом случае $\varphi(v)$ называется *весом слова* v. Если $\varphi(v) = 0$, то v не принадлежит данному множеству. Для краткости каждый элемент будем обозначать как произведение веса α на соответствующую цепочку v: $\varphi \cdot v$. Если выполнено условие $\sum\limits_{v \in V} \varphi(v) = 0$, то множество является пустым.

Определение 7. Множество взвешенных слов из V называется *нормированным*, если выполнено условие: $\sum\limits_{v \in V} \varphi(v) = 1$.

Определение 8. Пусть V_1 и V_2 — множества взвешенных цепочек. Определим три операции на этих множествах.

1. *Взвешенное объединение*:

   $\epsilon \cdot V_1 \bigcup \delta \cdot V_2 = \{(\epsilon \cdot a + \delta \cdot b) \cdot v | a \cdot v \in V_1$ или $b \cdot v \in V_2\}$, где $\epsilon + \delta = 1$.

   Пример: $V_1 = \{\frac{1}{2}a, \frac{1}{2}b\}, V_2 = \{\frac{2}{3}a, \frac{1}{3}c\}, \epsilon = \frac{1}{2}, \delta = \frac{1}{2}$.

   $\epsilon \cdot V_1 \bigcup \delta \cdot V_2 = \frac{1}{2} \cdot \{\frac{1}{2}a, \frac{1}{2}b\} \bigcup \frac{1}{2} \cdot \{\frac{2}{3}a, \frac{1}{3}c\} = \{(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{2}{3}) \cdot a, (\frac{1}{2} \cdot \frac{1}{2}) \cdot b, (\frac{1}{2} \cdot \frac{1}{3}) \cdot c\} = \{\frac{7}{12}a, \frac{1}{4}b, \frac{1}{6}c\}$.

2. *Взвешенная конкатенация*:

$V_1V_2 = \{\gamma \cdot vw | \alpha \cdot v \in V_1, \beta \cdot w \in V_2, \gamma = \sum\limits_{\{i,j|vw=v_iw_j, \alpha_i \cdot v_i \in V_1, \beta_j \cdot w_j \in V_2\}} (\alpha_i \cdot \beta_j)\}$.

Пример: $V_1 = \{\frac{1}{2}\epsilon, \frac{1}{2}a\}, V_2 = \{\frac{2}{3}\epsilon, \frac{1}{3}a\}$.

$V_1V_2 = \{\frac{1}{2}\epsilon, \frac{1}{2}a\}\{\frac{2}{3}\epsilon, \frac{1}{3}a\} = \{(\frac{1}{2} \cdot \frac{2}{3}) \cdot \epsilon, (\frac{1}{2} \cdot \frac{1}{3} + \frac{1}{2} \cdot \frac{2}{3}) \cdot a, (\frac{1}{2} \cdot \frac{1}{3}) \cdot aa\} = \{\frac{1}{3}\epsilon, \frac{1}{2}a, \frac{1}{6}aa\}$.

3. *Взвешенная итерация*:

$V_\alpha^* = \sum\limits_{i=0}^{\infty} \alpha_i V^i$, где $\sum\limits_{i=0}^{\infty} \alpha_i = 1$, $\forall i(\alpha_i \geqslant 0)$, V^0 состоит из пустого слова, $V^1 = V$, $V^n = VV \ldots V$ (n раз).

Пример: $V = \{a\}, \quad \alpha = \{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots\}. \quad V_\alpha^* = \{\frac{1}{2}\epsilon, \frac{1}{4}a, \frac{1}{8}aa, \ldots\}$.

Определение 9. *Класс взвешенных регулярных множеств* над конечным алфавитом V определяется следующим образом:
1. $\emptyset$ — взвешенное регулярное множество;
2. $\{\epsilon\}$ — взвешенное регулярное множество;
3. $\{a\}$ — взвешенное регулярное множество для любого a из V (a считается также однобуквенным словом);
4. Если S и T — взвешенные регулярные множества, то взвешенными регулярными множествами являются следующие множества:
— взвешенное объединение $\epsilon \cdot S \bigcup \delta \cdot T$, где $\epsilon + \delta = 1$;
— взвешенная конкатенация (ST);
— взвешенные итерации S_α^* и T_α^*, где $\sum\limits_{i=0}^{\infty} \alpha_i = 1$ и $\forall i(\alpha_i \geqslant 0)$.

Определение 10. *Класс взвешенных регулярных выражений* над конечным алфавитом V определяется следующим образом:
1. $\emptyset$ и ϵ — взвешенные регулярные выражения;
2. a — взвешенное регулярное выражение для любого a из V (a считается также однобуквенным словом);
3. Если R и S — взвешенные регулярные выражения, взвешенными регулярными выражениями являются следующие выражения:
— их взвешенное объединение $(\epsilon \cdot R)|(\delta \cdot S)$, где $\epsilon + \delta = 1$;
— их взвешенное произведение $(R)(S)$;
— их взвешенные итерации $(R)_\alpha^*$ и $(S)_\alpha^*$, где $\sum\limits_{i=0}^{\infty} \alpha_i = 1$ и $\forall i(\alpha_i \geqslant 0)$.
4. Если выражение является регулярным, то оно построено конечным числом применения правил 1–3.

Определение 11. Два взвешенных регулярных выражения $R1$ и $R2$ называются *эквивалентными*, (обозначается $R1 = R2$) тогда и только тогда, когда равны их соответствующие взвешенные регулярные множества.

Для любых взвешенных регулярных выражений R, S и T справедливо:

$R\|S = S\|R$	$R(ST) = (RS)T$
$R\|R = R$	$\emptyset R = R\emptyset = \emptyset$
$R\|(S\|T) = (R\|S)\|T$	$R(S\|T) = RS\|RT$
$\emptyset\|R = R$	$R\alpha S = \alpha RS$
$R\epsilon = \epsilon R = R$	$\alpha R\beta S = \alpha\beta RS$.

§ 8. Пересечение взвешенных регулярных множеств

Определение 12. *Делением взвешенного множества слов на взвешенное множество слов слева* называется:

$$B\backslash A = \{\gamma \cdot w | \exists \alpha \cdot v \in B\ , \beta \cdot u \in A, u = vw\} = \{\gamma_1 \cdot w_1, \gamma_2 \cdot w_2, \ldots, \gamma_m \cdot w_m, \ldots\}, \tag{8.1}$$

где выполнено условие: $\forall i, j\Big(\gamma_i/\gamma_j = \big(\sum\limits_{\{k,l|v_k w_i = u_l\}} \alpha_k \cdot \beta_l\big)/\big(\sum\limits_{\{k,l|v_k w_j = u_l\}} \alpha_k \cdot \beta_l\big)\Big)$.

Значения весов можно вычислить по следующей формуле: $\Gamma = \{\gamma_i | i = 1, 2, \ldots, m, \ldots\} = \{\sum\limits_{\{k,l|v_k w_i = u_l\}} \alpha_k \cdot \beta_l | i = 1, 2, \ldots, m, \ldots\}$.

При этом, если необходимо в результате получить нормированное множество, то должно быть выполнено дополнительное условие: $\sum\limits_i \gamma_i = 1$. Значения весов в этом случае можно вычислить по следующей формуле: $\gamma_i = x_i/(\sum\limits_j x_j)$, где

$X = \{x_i | i = 1, 2, \ldots, m, \ldots\} = \{\sum\limits_{\{k,l|v_k w_i = u_l\}} \alpha_k \cdot \beta_l | i = 1, 2, \ldots, m, \ldots\}$. Очевидно, что если $B\backslash A = \emptyset$, то результат является ненормируемым. Если $\sum\limits_j x_j = \infty$, то множество также является ненормируемым.

Пример: $B = \{\frac{1}{2}a, \frac{1}{2}b\}$, $A = \{\frac{1}{4}af, \frac{1}{4}bg, \frac{1}{4}ch, \frac{1}{4}ag\}$. $B\backslash A = \{\frac{1}{2}a, \frac{1}{2}b\}\backslash\{\frac{1}{4}af, \frac{1}{4}bg, \frac{1}{4}ch, \frac{1}{4}ag\} = \{\gamma_1 f, \gamma_2 g\}$. $X = \{x_1, x_2\}$. $x_1 = \frac{1}{2} \cdot \frac{1}{4} = \frac{1}{8}$, $x_2 = \frac{1}{2} \cdot \frac{1}{4} + \frac{1}{2} \cdot \frac{1}{4} = \frac{1}{4}$. $\gamma_1 = \frac{x_1}{x_1 + x_2} = \frac{\frac{1}{8}}{\frac{1}{8} + \frac{1}{4}} = \frac{1}{3}$. $\gamma_2 = \frac{x_2}{x_1 + x_2} = \frac{\frac{1}{4}}{\frac{1}{8} + \frac{1}{4}} = \frac{2}{3}$. $B\backslash A = \{\frac{1}{3}f, \frac{2}{3}g\}$.

Теорема 3 (о делении слева множеств, заданных взвешенными регулярными выражениями).

$$\begin{array}{lll} a\backslash(e_1|e_2) = (a\backslash e_1|a\backslash e_2) & a\backslash e^* = (a\backslash e)e^* & a\backslash\epsilon = \emptyset \\ a\backslash(e_1 e_2) = \begin{cases} ((a\backslash e_1)e_2|a\backslash e_2),\ \epsilon \in e_1 \\ ((a\backslash e_1)e_2),\ \epsilon \notin e_1 \end{cases} & a\backslash a = \epsilon & a\backslash b = \emptyset,\ a \neq b. \end{array} \tag{8.2}$$

Доказательство аналогично доказательству теоремы 1.

Если R, S и T взвешенные регулярные выражения, то справедливо:

$$\begin{array}{ll} (R|S)\backslash T = (R\backslash T)|(S\backslash T) & (\alpha R)\backslash S = \alpha(R\backslash S) \\ (RS)\backslash T = S\backslash(R\backslash T) & R\backslash(\alpha S) = \alpha(R\backslash S). \end{array}$$

Теорема 4 (о пересечении двух множеств, заданных взвешенными регулярными выражениями).

$$\left(\sum_i b_i \cdot C\right) \bigcap A = \sum_i \left(b_i (C \bigcap (b_i\backslash A))\right). \tag{8.3}$$

Доказательство аналогично доказательству теоремы 2.

§ 9. Сложность вычислений со взвешенными регулярными выражениями

Оценим сложность алгоритма нахождения пересечения двух взвешенных множеств. Аналогично с пересечением регулярных множеств получаем:

$$(\sum_{i_1=1}^{n_1} b_{i_1}^1)(\sum_{i_2=1}^{n_2} b_{i_2}^2)\ldots(\sum_{i_m=1}^{n_m} b_{i_m}^m) \bigcap A \sim \sum_{i_1=1}^{n_1} b_{i_1}^1 (\sum_{i_2=1}^{n_2} b_{i_2}^2 (\ldots (\sum_{i_m=1}^{n_m} b_{i_m}^m (\ldots))) \sim \prod_{i=1}^{m} n_i < 3^{\frac{C}{3}},$$

где $C = \sum\limits_{i=1}^{m} n_i$.

Оценим используемую память. Так как в худшем случае элемент пересечения имеет вес $\frac{1}{3^{\frac{C}{3}}}$, то минимальным требованием является возможность хранения $3^{\frac{C}{3}}$ различных весов. Вычислим необходимую точность, для этого решим уравнение $2^k = 3^{\frac{C}{3}}$ относительно k. Решением является $k = \frac{C}{3}\log_2 3 \approx 0,53C$. Следовательно, минимально допустимая точность $\frac{1}{2^{0,53C}}$.

Пусть $A = C$, тогда

$||C \bigcap A|| = ||(\sum\limits_{i_1=1}^{n_1} b_{i_1}^1)(\sum\limits_{i_2=1}^{n_2} b_{i_2}^2)\dots(\sum\limits_{i_m=1}^{n_m} b_{i_m}^m) \bigcap (\sum\limits_{i_1=1}^{n_1} b_{i_1}^1)(\sum\limits_{i_2=1}^{n_2} b_{i_2}^2)\dots(\sum\limits_{i_m=1}^{n_m} b_{i_m}^m)||$

$= \sum\limits_{i_1=1}^{n_1}(0,53\frac{C}{8}+1)(\sum\limits_{i_2=1}^{n_2}(0,53\frac{C}{8}+1)(\dots(\sum\limits_{i_m=1}^{n_m}(0,53\frac{C}{8}+1)))) = \prod\limits_{i=1}^{m} n_i \cdot (0,53\frac{C}{8}+1)^m < 3^{\frac{C}{3}} \cdot (0,53\frac{C}{8})^{\frac{C}{3}}$

$= (3(\cdot 0,53\frac{C}{8}+1))^{\frac{C}{3}}$.

При $A = C$ и $\forall i(n_i = 3)$ сложность по использованию памяти является максимальной.

При небольших значениях C сложность вычислений имеет разумные границы. Например, при $C = 30$ в худшем случае сложность равна 3^{10}. Для современных компьютеров вполне приемлемая величина, чего нельзя сказать уже при $C = 99$, когда сложность составляет 3^{33}.

По использованию памяти аналогично, при $C = 30$ и $C = 99$ получаем величины примерно равные соответственно 9^{10} и 9^{33}.

Рассмотрим более благоприятные случаи. Так как сложность составляет $\prod\limits_{i=1}^{m} n_i$, то сложность приемлема, когда m близко по значению к C, то есть большинство $n_i = 1$ и когда m близко к 1. Так если $m = C$, то сложность равна 1, при $m = 1$ сложность составляет C. Очевидно, что $n_i = 1$ не увеличивают сложность, поэтому сложность растет только при увеличении числа сумм с двумя или более элементами, соответственно, сложность имеет разумные границы вне зависимости от величины C, когда таких сумм в выражении не много (не более 10).

§ 10. Приближенные вычисления

Так как на практике невозможно производить вычисления с неограниченной точностью, то рассмотрим способ приближенного вычисления весов.

Пусть точность равна $\frac{1}{2^n}$, тогда из условия нормируемости взвешенного множества получаем множество всевозможных значений весов $P = \{\frac{k}{2^n} | k = 0, 1, 2, \dots, 2^n\}$. Так при $n = 1$ получаем $P = \{0, \frac{1}{2}, 1\}$, при $n = 2$ получаем $P = \{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\}$ и так далее. Так как данное множество имеет конечное число элементов, то при вычислении весов производится округление.

Выполнение свойства нормируемости при приближенных вычислениях уже не требуется. Так множество $\{\frac{1}{2}\cdot a, \frac{1}{2}\cdot b, \frac{1}{2}\cdot c\}$ так же является взвешенным регулярным множеством, в котором сумма весов равна $1,5$. Данное множество получается округлением весов в множестве $\{\frac{1}{3}\cdot a, \frac{1}{3}\cdot b, \frac{1}{3}\cdot c\}$ с точностью $\frac{1}{2}$.

Так как вес элемента в большинстве случаев меньше 1, то при перемножении весов округленный результат может быть равен 0, то есть в этом случае элемент множества теряет значимость.

Пример: $\frac{1}{2}\{\frac{1}{4}\cdot a, \frac{3}{4}\cdot b\} \bigcup \frac{1}{2}\{\frac{1}{2}\cdot b, \frac{1}{2}\cdot c\} = \{\frac{1}{8}\cdot a, \frac{5}{8}\cdot b, \frac{1}{4}\cdot c\}$. При округлении весов с точностью $\frac{1}{4}$ получаем множество $\{0\cdot a, \frac{3}{4}\cdot b, \frac{1}{4}\cdot c\}$, то есть $\frac{1}{2}\cdot\frac{1}{4} = 0$. Элемент a имеет вес 0 и должен быть исключен из множества. В результате остаются только элементы с положительным весом: $\{\frac{3}{4}\cdot b, \frac{1}{4}\cdot c\}$.

Потеря значимости в приближенных вычислениях в большинстве случаях нежелательна, поэтому точность вычислений необходимо выбирать так, чтобы она не повлияла на конечный результат.

§ 11. Сложность on-line вычислений

Рассмотрим такие вычисления, входные данные в которых меняются со временем. Пусть уже найдено пересечение двух регулярных выражений. Требуется найти пересечение после внесенных изменений в одно из регулярных выражений. Так как правая часть пересечения

представлена произведением сумм, то рассмотрим изменение количества слагаемых в данном произведении. В задаче распознавания данное изменение эквивалентно добавлению новых символов в текст или их удалению.

Рассмотрим два случая.

1) Пусть уже найдено $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})(\sum\limits_{i_2=1}^{n_2} b_{2,i_2})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})\bigcap A$.

Требуется найти $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})(\sum\limits_{i_2=1}^{n_2} b_{2,i_2})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})(\sum\limits_{i_{m+1}=1}^{n_{m+1}} b_{m+1,i_{m+1}})\bigcap A$.

По теореме получаем $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})(\sum\limits_{i_2=1}^{n_2} b_{2,i_2})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})\bigcap A$

$= \sum\limits_{i_1=1}^{n_1} b_{1,i_1}(\sum\limits_{i_2=1}^{n_2} b_{2,i_2}(\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m}(\epsilon\bigcap(b_{m,i_m}\backslash(\dots(b_{2,i_2}\backslash(b_{1,i_1}\backslash A)))))))).$

Подставим вместо ϵ $(\sum\limits_{i_{m+1}=1}^{n_{m+1}} b_{m+1,i_{m+1}})\cdot\epsilon$ и каждую $(b_{m,i_m}\backslash(\dots(b_{2,i_2}\backslash(b_{1,i_1}\backslash A))))$ слева разделим на все $b_{m+1,i_{m+1}}$ по отдельности. В итоге получим

$\sum\limits_{i_1=1}^{n_1} b_{1,i_1}(\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m}(\sum\limits_{i_{m+1}=1}^{n_{m+1}} b_{m+1,i_{m+1}}(\epsilon\bigcap(b_{m+1,i_{m+1}}\backslash(b_{m,i_m}\backslash(b_{1,i_1}\backslash A)))))))$, что по теореме равно $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})(\sum\limits_{i_2=1}^{n_2} b_{2,i_2})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})(\sum\limits_{i_{m+1}=1}^{n_{m+1}} b_{m+1,i_{m+1}})\bigcap A$.

Очевидно, что сложность в данном случае равна той, которая получена в § 6.

2) Пусть уже найдено $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})\dots(\sum\limits_{i_{k-1}=1}^{n_{k-1}} b_{k-1,i_{k-1}})(\sum\limits_{i_{k+1}=1}^{n_{k+1}} b_{k+1,i_{k+1}})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})\bigcap A$.

Требуется найти $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})\dots(\sum\limits_{i_{k-1}=1}^{n_{k-1}} b_{k-1,i_{k-1}})(\sum\limits_{i_k=1}^{n_k} b_{k,i_k})(\sum\limits_{i_{k+1}=1}^{n_{k+1}} b_{k+1,i_{k+1}})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})\bigcap A$.

Рассмотрим $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})\dots(\sum\limits_{i_{k-1}=1}^{n_{k-1}} b_{k-1,i_{k-1}})\bigcap A$. Сложность нахождения данного пересечения обозначим через: $O(n_1, ..., n_{k-1})$.

Аналогично с 1) находим $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})\dots(\sum\limits_{i_{k-1}=1}^{n_{k-1}} b_{k-1,i_{k-1}})(\sum\limits_{i_{k+1}=1}^{n_{k+1}} b_{k+1,i_{k+1}})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})\bigcap A$ и $(\sum\limits_{i_1=1}^{n_1} b_{1,i_1})\dots(\sum\limits_{i_{k-1}=1}^{n_{k-1}} b_{k-1,i_{k-1}})(\sum\limits_{i_k=1}^{n_k} b_{k,i_k})(\sum\limits_{i_{k+1}=1}^{n_{k+1}} b_{k+1,i_{k+1}})\dots(\sum\limits_{i_m=1}^{n_m} b_{m,i_m})\bigcap A$, сложности нахождения которых равны соответственно: $O(n_1, ..., n_{k-1}, n_{k+1}, ..., n_m)$ и $O(n_1, ..., n_{k-1}, n_k, n_{k+1}, ..., n_m)$.

В итоге сложность текущего on-line вычисления равна
$O(n_1, \dots, n_{k-1}, n_{k+1}, \dots, n_m) + O(n_1, \dots, n_{k-1}, n_k, n_{k+1}, \dots, n_m) - O(n_1, \dots, n_{k-1})$.

Заключение

Сформулирована и доказана теорема о пересечении двух регулярных множеств, заданных регулярными выражениями. Основное достоинство применения данной теоремы состоит в том, что она освобождает от необходимости рассматривать сразу все выражение целиком и при рассмотрении отдельных фрагментов позволяет последовательно исключать лишние элементы множества. Тем самым происходит значительное уменьшение сложности процедуры нахождения пересечения двух регулярных множеств.

Но, несмотря на это, сложность алгоритма в худших случаях экспоненциальная величина. Выделены и благоприятные случаи, когда сложность вычислений приемлема для решения тех или иных задач. В некоторых случаях она может увеличиваться линейно в зависимости от длины входных данных.

Для решения задач распознавания текстов, что является основной целью работы, сложность алгоритма имеет вполне разумные границы. Во-первых, потому что рассматривается не весь текст сразу, а только отдельные слова или выражения, которые имеют небольшую длину. Во-вторых, шаблонов, пересечения с которым необходимо найти, ограниченное количество.

МАТЕМАТИКА 2012. Вып. 2

В-третьих, уже после применения нескольких шагов алгоритма, исключается большинство лишних элементов, то есть число возможных элементов быстро уменьшается.

Стоит отметить, что основная цель состоит не в распознавании обычных текстов, а в распознавании математических формул. Применять регулярные выражения для представления математических формул довольно сложно, так как они имеют нелинейную структуру. Для решения данной задачи в дальнейшем потребуется ввести понятия регулярных и взвешенных регулярных деревьев, а также сформулировать теорему о пересечении регулярных множеств заданных регулярными деревьями.

СПИСОК ЛИТЕРАТУРЫ

1. Карпов Ю.Г. Теория автоматов. СПб.: Питер, 2003. 208 с.
2. Чашкин А.В. Лекции по дискретной математике. М.: МГУ, 2007. 261 с.

Поступила в редакцию 30.03.2012

Сапаров Алексей Юрьевич, аспирант, кафедра теоретических основ информатики, Удмуртский государственный университет, 426034, Россия, г. Ижевск, ул. Университетская, 1.
E-mail: say.saplh@gmail.com

Бельтюков Анатолий Петрович, профессор, д.ф.-м.н., заведующий кафедрой теоретических основ информатики, Удмуртский государственный университет, 426034, Россия, г. Ижевск, ул. Университетская, 1. E-mail: belt@uni.udm.ru

A. Yu. Saparov, A. P. Beltyukov
Regular expressions in the mathematical text recognition problem

Keywords: regular sets, regular expressions, regular operations.

Mathematical Subject Classifications: 03D05, 68Q17

The work is devoted to use of regular expressions at recognition of hand-written mathematic texts. The main problem in handwritten mathematical formula recognition is that these texts mainly consist of a large number of small fragments, arranged in accordance with some strict rules. Despite the fact that formal definition of syntax of mathematic texts can involve context-free grammars and even more complicated constructions, it frequently suffices definition of mathematical language on the base of regular expressions for successful recognition. Since some constructions can occur in mathematic texts frequently than other, we introduce the concept of the weighed regular expression. The weights determine preference of some constructions before other ones. In the work, mathematical tools for use of such expressions at recognition is introduced. Theorems about intersection of weighed sets defined by such regular expressions are proved. Some estimations are given to complexity of algorithms work using such regular expressions for recognition.

REFERENCES

1. Karpov Y.G. *Teoriya avtomatov* (Automata theory), St. Petersburg: Piter, 2003, 208 p.
2. Chashkin A.V. *Lektsii po diskretnoi matematike* (Lectures on discrete mathematics), Moscow: Moscow State University, 2007, 261 p.

Received 30.03.2012

Saparov Aleksei Yur'evich, post-graduate student, Department of Theoretical Foundations of Computer Science, Udmurt State University, ul. Universitetskaya, 1, Izhevsk, 426034, Russia.
E-mail: say.saplh@gmail.com

Beltyukov Anatolii Petrovich, Professor, Doctor of Ph.-Math. Sci., Head of the Department of Theoretical Foundations of Computer Science, Udmurt State University, ul. Universitetskaya, 1, Izhevsk, 426034, Russia.
E-mail: belt@uni.udm.ru