

В матрице $\Phi_s^{(s-1)}$ рассмотрим блок $[b_{(s)}^{(s-1)}, H_{\rho_s} b_{(s)}^{(s-1)}, \dots, H_{\rho_s}^{n_{\rho_s}-1} b_{(s)}^{(s-1)}]$. Разделим каждый столбец этого блока на отличное от нуля число $\lambda_s^{(s-1)} (\sigma - n_{\rho_s} + 1)$, где $\sigma = \sum_{i=1}^s n_{\rho_i}$. Из i -ого ($i < \sigma$) столбца вычтем j -ый ($j = i + 1, \sigma$) столбец, умноженный на $C_{\sigma - n_{\rho_s} + 1}^{j-i} / \lambda_s^{(s-1) (j-i)}$. В результате, вместо блока $[b_{(s)}^{(s-1)}, H_{\rho_s} b_{(s)}^{(s-1)}, H_{\rho_s}^{n_{\rho_s}-1} b_{(s)}^{(s-1)}]$ получим блок $[b_{(s)}^{(0)} \dots H_{\rho_s}^{n_{\rho_s}-1} b_{(s)}^{(0)}]$. Вычитая соответствующие комбинации столбцов последнего блока из предыдущих столбцов матрицы $\Phi_s^{(s-1)}$, в первых $(s-1)$ блоках получим нули (очевидно, что остальные матрицы $\Phi_1^{(s-1)}, \dots, \Phi_{s-1}^{(s-1)}$ не изменяются). С матрицами $\Phi_1^{(s-1)}, \dots, \Phi_{s-1}^{(s-1)}$ поступаем аналогичным образом. Итак, существует невырожденное преобразование столбцов, приводящее матрицу $\Phi^{(0)} \begin{pmatrix} 1 & \dots & n \\ 1 & \dots & n \end{pmatrix}$ к виду

$$\begin{bmatrix} b_{(1)}^{(0)} H_{\rho_1} b_{(1)}^{(0)} \dots H_{\rho_1}^{n_{\rho_1}-1} b_{(1)}^{(0)} & 0 & \dots & 0 \\ & b_{(2)}^{(0)} \dots H_{\rho_2}^{n_{\rho_2}-1} b_{(2)}^{(0)} & \dots & 0 \\ & 0 & \dots & 0 \\ & & 0 & \dots & 0 \\ & & & \dots & b_{(s)}^{(0)} \dots H_{\rho_s}^{n_{\rho_s}-1} b_{(s)}^{(0)} \end{bmatrix}$$

Вид последней матрицы (поскольку доказательство приводилось для произвольного столбца b матрицы TB) и доказывает, что соотношение (4) выполняется.

Теперь, используя (4), нетрудно видеть, что проверка критерия (2) сводится к проверке равенств

$$\text{rang} \{B_i, H_{\rho_i} B_i, \dots, H_{\rho_i}^{n_{\rho_i}-1} B_i\} = \rho_i, \quad i = \overline{1, s}. \quad (6)$$

Если же окажется, что и в соотношениях (6) матрицы все же имеют большие размеры, то для их проверки в дальнейшем следует использовать алгоритм, изложенный в работе [2].

ЛИТЕРАТУРА

1. Калман Р.— Труды I конгресса ИФАК.— М., 1961, т. 2, с. 521.
2. Размыслович Г. П.— Дифференц. уравнения, 1977, т. 13, № 6, с. 1141.
3. Гантмахер Ф. Р. Теория матриц.— М., 1967.

Зак. 81 Бубневич 3-3 нал. Громак

Поступила в редакцию
24.09.81.

Кафедра высшей математики ФПМ

УДК 519.76

И. А. КОРОЛЬ

СИНТЕЗ В ОПРЕДЕЛЕННЫХ СИСТЕМАХ МАШИННОГО ПЕРЕВОДА КАК ЗАДАЧА ИНФОРМАЦИОННОГО ПОИСКА

При решении многих прикладных задач для описания языков (в частности, языки описания данных, алгоритмические и естественные языки) удобно использовать определенные формализмы, так называемые *метаязыки* [1]. Так, для описания синтаксиса языков используют *метасинтаксические языки*.

Наиболее распространенным метасинтаксическим языком являются *нормальные формы Бекуса* (или *металингвистические формулы*), основ-

ное назначение которых представить в сжатом и компактном виде строго формальные и однозначные правила написания основных конструкций описываемого языка.

При разработке систем машинного перевода (МП) на отдельных этапах возникают задачи компактного описания лингвистических данных и алгоритмов анализа и синтеза текстов естественного языка с целью их ввода в память ЭВМ и рациональной автоматической обработки.

Рассмотрим одну из задач для случая разработки системы МП научно-технических текстов в рамках инженерно-лингвистического подхода [2]. На этапе анализа в таких системах [3] каждое входное предложение разбивается на сегменты — семантически связанные и грамматически оформленные совокупности словоформ, устанавливаются формальные, а затем функциональные типы сегментов (сегмент-сказуемое (1), сегмент-подлежащее (2), именной предложный сегмент (3), сегмент союз (4), сегмент-прямое дополнение (5), так называемый «остаточный» сегмент (6)), устанавливается межсегментная зависимость и на ее основе снимается конверсионная и грамматическая омонимия. Поскольку в указанных системах семантика используется фактически только на уровне оборотов, на этапе анализа смысл каждой получаемой информационной структуры устанавливается неоднозначно, поэтому процедура автоматического морфологического и синтаксического синтеза (АСС) [4] основана на сложном взаимодействии свойств информационной структуры, получаемой на этапе анализа, и широких свойств элементов определенного информационного множества $T = \{t_j\}$, ($j = \overline{1, m}$), в качестве которого выступает множество словарных форм (СФ) русского языка. Это взаимодействие фиксируется в виде конечного множества P специальных лингвистических алгоритмов. Задача синтеза основана на сложном взаимодействии свойств заданных структур, поэтому для получения оптимального решения ее надо рассматривать как задачу информационного поиска. Причем в качестве запросов выступают информационные структуры, получаемые на этапе анализа, а в качестве массива поиска — множество специальных лингвистических алгоритмов.

Каждый алгоритм $P_i \subset P$ содержит условие поиска W_i и перечень определенных действий с конечной целью — выбором соответствующих эквивалентов для каждой СФ или совокупности СФ входного языка. Условие поиска W_i можно определить в виде

$$W_i = F_i(f_i^{(1)}(X_i^{(1)}), f_i^{(2)}(X_i^{(2)}), \dots, f_i^{(p_i)}(X_i^{(p_i)})) (i = \overline{1, n}), \quad (1)$$

где функции $f_i^{(1)}, f_i^{(2)}, \dots, f_i^{(p_i)}$ задают условия поиска отдельных объектов из T ; F_i определяет ту часть условия W_i , которая учитывает необходимые отношения между этими объектами, а $X_i^{(j)} = \{x_{i_1}^{(j)}, x_{i_2}^{(j)}, \dots, x_{i_{k_j}}^{(j)}\}$ —

некоторое подмножество множества $X = \{x_i\}$, ($i = \overline{1, n}$), попарно различных поисковых признаков, которыми являются грамматические коды словоформ. Формальное представление условия поиска позволяет разработать четкий метасинтаксический язык описания данных и построить оптимальный машинный алгоритм поиска.

Функция F_i является функцией многих переменных, и поэтому правило P_i задает для каждого определенного поискового признака $X_i^{(k)}$ ($1 \leq k \leq p_i$), обозначаемого в дальнейшем через X_3 , конечную последовательность поисковых признаков

$$X_i^{(1)}, X_i^{(2)}, \dots, X_i^{(k-2)}, X_i^{(k-1)}, \quad (2)$$

предшествующих X_0 и конечную последовательность поисковых признаков

$$X_i^{(k+1)}, X_i^{(k+2)}, \dots, X_i^{(p_i-1)}, X_i^{(p_i)}, \quad (3)$$

следующих за X_0 . Одновременно для каждого поискового признака указывается набор определенных кодов русских эквивалентов.

В дальнейшем (2) будем называть *левой ситуацией* алгоритма P_i ,

(3) — *правой ситуацией*. Множество значений аргумента $X_i^{(j)}$ ($1 \leq j \leq \leq k-1$) — *левой i -ой цепочкой*, множество значений аргумента $X_i^{(j)}$ ($k+1 \leq j \leq \leq p_i$) — *правой i -ой цепочкой*, множество значений аргумента X_0 — *главной цепочкой*, а каждый элемент $x_0 \in X_0$ — *главным поисковым признаком*. Такой алгоритм $P_i \subseteq P$, что все $X_i^{(j)}$ ($1 \leq j \leq p_i$) однозначны, назовем *элементарным* алгоритмом. Если множество значений хотя бы одного $X_i^{(j)}$ ($1 \leq j \leq p_i$) отлично от единицы, то соответствующий алгоритм будем называть *составным*.

Приведем пример одного из алгоритмов. Если в сегменте (2) для опорного S с кодами 7501, 10001, 10002, ..., 7517, 10037 код первой словоформы в соответствующем лексическом гнезде равен 9001, 8010, 3082, ..., 8018, 9017, то

для S всегда бери из лексического гнезда словоформу с кодами 3026, 9013, 3108, ..., 9011, 3082;

для A (или 12002) — 2808, 2889, 9158, 9122, 9032;

для H (или 12006) — 2233, 1059, 1163, ..., 1527, 2311;

для G (или 12014) — 2203, 2493;

для T — 2233, 1059, 1163, ..., 1735, 1683;

для D — 4501, 4502, 4505, 4506.

Для приведенного алгоритма условие поиска (1) запишется следующим образом:

$$W = F(f(X_1), f(X_2), \dots, f(X_N)), \quad (1')$$

где $N = 14$. Рассмотрим некоторые свойства функций (1').

$f(X_1)$ — функция *сегментации*: аргумент X_1 принимает значения из множества $\{1, 2, \dots, 6\}$ номеров сегмента.

$f(X_2)$ — *опорная* функция: аргумент X_2 принимает значение из множества $\{S, Q, N\}$ опорных символов.

$f(X_3)$ — функция *1-ой словоформы*: X_3 принимает значения из множества $\{0, 1\}$, где 0 означает код не первой словоформы, 1 — код первой словоформы.

$f(X_n)$ ($n = 4, N$) — *кодовые* функции: X_n принимает значения из множества $\{S, A, 12002, F, 12006, W, 12014, \dots, T, L, \dots, D\}$ поисковых кодов.

Для приведенного выше алгоритма эти функции примут вид:

$$f(X_1) = 2 \quad (4), \quad f(X_2) = 7501 \vee 10001 \vee 1002 \vee \dots \vee 7517 \vee 10037, \quad (5)$$

$$f(X_3) = 9001 \vee 8010 \vee 3082 \vee \dots \vee 8018 \vee 9017, \quad (6)$$

$$f(X_4) = S, \quad f(X_5) = A \vee 12002, \dots, f(X_{14}) = D. \quad (7)$$

Тогда функция F выглядит так:

$$F: f(X_1) \wedge f(X_2) \wedge \dots \wedge f(X_N). \quad (8)$$

При использовании функций (4)–(7) алгоритм, приведенный ранее, запишется следующим образом:

$$F(f(X_1), \dots, f(X_N)) = ((3026 \vee 9013 \vee \dots \vee 3082) \wedge (2808 \vee 2889 \vee \dots \vee 9032) \wedge \dots \wedge (4501 \vee 4502 \vee 4505 \vee 4506)). \quad (9)$$

В (9) функция сегментации $f(X_1)$ представляет собой левую ситуацию алгоритма P_i , $f(X_3), \dots, f(X_N)$ — правую ситуацию, множество значений аргумента X_1 — левую цепочку, X_2 — главную цепочку, $X_3, \dots, X_N$ — правую цепочку. Так как множество значений X_n ($n = 1, N$) в (9) отлично от единицы, то данный алгоритм будет составным.

Для удобства записи лингвистических алгоритмов и рациональной автоматической обработки их при формировании поискового массива в памяти ЭВМ был разработан метасинтаксический язык $L(G)$, с помощью которого каждый лингвистический алгоритм довольно просто записывается на стандартных бланках записи программ для ЕС ЭВМ в виде от-

дельных фраз. Для описания синтаксиса языка $L(G)$ используется нормальная форма Бекуса [5].

ЛИТЕРАТУРА

1. Братчиков И. Л. Синтаксис языков программирования.— М., 1975.
2. Пиотровский Р. Г. Текст, машина, человек.— Л., 1975.
3. Гончаренко В. В., Добрускина Э. М., Котельникова Н. М., Кричевич В. С., Совпель И. В. Автоматический перевод.— Кишинев, 1978.
4. Король И. А., Совпель И. В. Синтезатор естественного языка в системах машинного перевода.— Тез. докл. на 5-й Республиканской конференции математиков Белоруссии. Гродно, 1980, ч. 1, с. 83.
5. Алгоритмический язык АЛГОЛ-60: Пересмотренное сообщение.— М., 1965.

Поступила в редакцию
15.10.81.

Кафедра МО АСУ

УДК 62-50

А. В. БОРЗЕНКОВ, Р. ГАБАСОВ

АЛГОРИТМ РЕШЕНИЯ ОДНОЙ СПЕЦИАЛЬНОЙ ЗАДАЧИ НА МИНИМАКС

1. Постановка задачи. Пусть X — множество планов x , определенных соотношениями

$$Ax = b, d_* \leq x \leq d^*, \quad (1)$$

где $A = A(I, J)$, $b = b(I)$, $d^* = d^*(J)$, $d_* = d_*(J)$, $I = \{1, 2, \dots, m\}$, $J = \{1, 2, \dots, n\}$. Рассмотрим функции $V_k(x) = c_k x + f_k$, $k \in K = \{1, 2, \dots, \eta\}$, $c_k = c_k(J) \in R^n$, $f_k \in R$,

$$t_k(x) = \begin{cases} t_k, & \text{если } V_k(x) > V_k^*, \\ 0, & \text{если } V_k(x) \leq V_k^*, t_k > 0. \end{cases} \quad (2)$$

Качество плана x оценим числом

$$c(x) = t_{l(x)} = \max_{k \in K} t_k(x), \quad k \in K. \quad (3)$$

Оптимальным назовем план, на котором функция $c(x)$ достигает минимального значения

$$c(x_0) = \min_{x \in X} c(x). \quad (4)$$

2. Критерий оптимальности. На плане x определим множество $K(x) = \{j \in K : t_j \leq t_{l(x)}\}$. Из определения индекса $l(x)$ следует, что $t_j(x) = 0$ для всех $j \in K \setminus K(x) = K_-(x)$. Построим матрицу $Q = Q(I \cup K_-(x), J) = \begin{cases} q_{ij} = \begin{cases} a_{ij}, & i \in I \\ c_{ij}, & i \in K_-(x), j \in J \end{cases} \end{cases}$. Пусть $z = z(J)$ — произвольный n -вектор, удовлетворяющий соотношениям

$$Q(I, J)z(J) = b(I), Q(K_-(x), J)z(J) \leq V^*(K_-(x)) - F(K_-(x)), \quad (5)$$

где $V^* = V^*(K)$, $F = F(K)$ — η -векторы. Под опорой будем понимать пару $\{z, Q_{\text{оп}}\} = P$, где $Q_{\text{оп}} = Q(I_{\text{оп}}^+(x), J_{\text{оп}})$, $I_{\text{оп}}^+(x) = I_{\text{оп}} \cup K_{-\text{оп}}(x)$, — неособая матрица, составленная из элементов q_{ij} , $i \in I_{\text{оп}}^+(x) = I \cup K_-(x)$, $j \in J$, матрицы $Q(I^+(x), J)$. Пару $\{x, P\}$ назовем опорным планом. Опорный план считается невырожденным, если: $Q(K_{-\text{н}}(x), J)z(J) < V^*(K_{-\text{н}}(x)) - F(K_{-\text{н}}(x))$, $d_*(J_{\text{оп}}) < z(J_{\text{оп}}) < d^*(J_{\text{оп}})$, где $K_{-\text{н}}(x) = K_-(x) \setminus K_{-\text{оп}}(x)$. Вычислим вектор потенциалов: $\bar{u}' = u'(I_{\text{оп}}^+(x)) = c_{e(x)}(J_{\text{оп}}) \cdot Q_{\text{оп}}^{-1}$ и вектор оценок: $\bar{\Delta}' = \bar{\Delta}'(J) = u'Q(I_{\text{оп}}^+(x), J) - c_{e(x)}$. Введем обозначения: $q_i = Q(i, J)$, $i \in I^+(x)$; $K_-(x) = \{i \in K_-(x), q_i z = V_i^* - f_i\}$; $K_{-}(x) = K_-(x) \setminus K_{-}^*(x)$; $K_{-\text{оп}}(x) = K_{-}^*(x) \cap K_{-\text{оп}}(x)$; $K_{-\text{оп}}^*(x) = K_{-}(x) \cap K_{-\text{оп}}(x)$; $J_* = \{j \in J, z_j = d_{*j}\}$; $J^* = \{j \in J, z_j = d_{*j}^*\}$; $J_{-} = J \setminus (J^* \cup J_*)$; $J_{*n} = J_n \cap J_*$; $J_{-n} = J_n \cap J_{-}$; $J_{-n}^* = J_{-n} \setminus (J_{-n}^* \cup J_{*n})$; $J_n = J \setminus J_{\text{оп}}$.

Теорема 1 (Критерий оптимальности). В задаче (1)—(4) соотношения