

АЛГОРИТМ ПОИСКА ОБЪЕКТОВ, ЗАДАНЫХ ПОЛНЫМ НАБОРОМ ПРИЗНАКОВ

В. А. Ловицкий

Одной из типовых задач, встречающихся при обработке больших объектов информации, представленной на естественном языке (машинный перевод, поиск научно-технической информации и т. п.), является нахождение слов в словаре.

При решении подобных задач применяются различные искусственные приемы и способы расположения информации в запоминающих устройствах машин [1, 2], повышающие эффективность ее поиска, а также различные методы представления самих процессов поиска и обработки материала. С целью получения наиболее эффективного механизма поиска данных в запоминающих устройствах вычислительных машин необходимо обратиться к изучению памяти человека.

Память человека можно рассматривать как кибернетическую систему, обрабатывающую информацию по определенным законам, строгое описание которых средствами современной науки в настоящее время вызывает значительные затруднения. В связи с этим особый интерес представляет метод эвристического моделирования [3] как способа исследования сложных форм психической деятельности на уровне информационных процессов.

Цель наших исследований заключается в выявлении законов переработки информации вербальной системой памяти и в применении этих законов для решения различных типов информационно-логических задач.

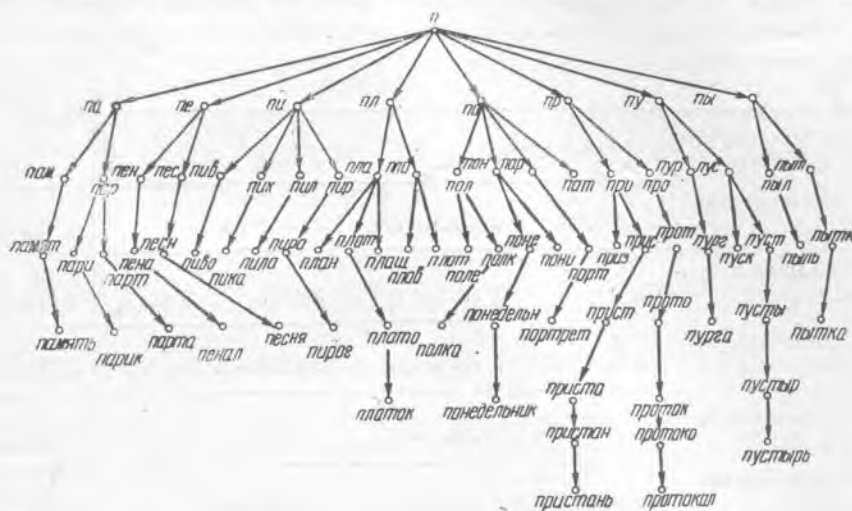
В данной работе исследуется алгоритм поиска информации человеком. Приводится пример, показывающий, что использование человеком особенностей поиска информации биологической системой позволяет существенно сократить время поиска за счет уменьшения числа сравнений, необходимых для нахождения искомого объекта. В примере алгоритм поиска описан на языке ассоциативного программирования [4], основанного на использовании АЛГОЛа и его расширения — АЛГЭМа. Основным отличием языка ассоциативного программирования (АП) от языков списковой обработки данных ЛИСП, ИПЛ-5, ФЛПЛ, АЛП и других является непосредственная связь языка АП с АЛГОЛом.

Для изучения алгоритма поиска информации человеком был проведен элементарный психологический эксперимент, заключающийся в том, что испытуемый в ответ на последовательное предъявление экспериментатором буквосочетаний, составленных из одной, двух, трех и т. д. букв должен был отвечать словом, начало которого представляет собой показанное буквосочетание. Буквосочетания получают из слова путем последовательного отделения сначала первой буквы, затем первых двух, первых трех и т. д. Результаты эксперимента сведены в табл. 1 и представлены в виде дерева на рисунке. Всего было проведено восемь опытов по предъявлению буквосочетаний, состоящих из одной, двух, трех и четы-

рых букв. По таблице строится дерево таким образом, что корнем его служит первая буква всех воспроизведенных слов; от нее ответвляются вершины, представляющие собой две первые буквы воспроизведенных слов, и т. д.

Анализ результатов, полученных в данном эксперименте, позволяет заключить следующее:

1) при поиске слова в памяти человека выделяется определенная древовидная структура, корнем которой служит начальная буква слова:



2) поиск слова, начинающегося на определенное буквосочетание, осуществляется по принципу воспроизведения недостающих букв, имеющих максимальный вес (под весом буквы понимается частота ее встречаемости на определенном месте и после определенного сочетания букв во входных словах). Воспроизведенные недостающие буквы дополняют предъявленное буквосочетание до слова, представляющего собой модель предъявляемого слова;

3) поиск слова — это активный процесс, который заключается в том, что при предъявлении первой буквы слова у испытуемого возникает модель предъявляемого слова, начинающегося с указанной буквы. По мере дальнейшего предъявления двух первых букв слова, трех первых букв и т. д. происходит коррекция модели до полного совпадения с предъявляемым словом. Возможны случаи (см. опыт I, табл. I), когда предъявленное слово не совпадает полностью с моделью, но обязательно входит в модель предъявленного слова.

Вернемся к задаче поиска слов в словаре. Покажем, что скорость поиска слов в древовидном словаре можно существенно увеличить, если учитывать вес букв. Опишем структуру древовидного словаря.

Все данные, хранящиеся в памяти машины, представлены в виде списков. Список — это группа последовательно расположенных данных. Последовательный способ построения списков выбран исключительно для простоты организации данных. Некоторые члены списка могут указывать на ответвления к другим спискам, называемых под списками. Любой список со всеми отходящими от него под списками называется списковой структурой. Списковая структура имеет древовидный характер. В вершине такой структуры располагается список 1-го слоя.

осуществляющий деление всех объектов по одному какому-либо признаку. От членов этого списка отходят подспики 2-го слоя, соответствующие делению объектов по другому признаку, и т. д. В нашем случае объектами являются слова естественного языка. Каждое слово характеризуется определенным количеством признаков: начальная буква слова, вторая буква и т. д., т. е. число признаков соответствует количеству букв в слове. Каждый признак способен принимать определенное число значений, например, первой буквой слова в русском языке может быть любая буква алфавита, кроме Ъ, Ь, Ъ. Различные значения одного и того же признака несовместимы, в то время как различные признаки являются совместными.

Таблица 1

1	Предъявляемые буквосочетания	1 П	2 ПА	3 ПАР	4 ПАРИ
	Воспроизведенные слова	1 ПИКА	2 ПАМЯТЬ	3 ПАРТА	4 ПАРИК
2	Предъявляемые буквосочетания	1 П	2 ПЕ	3 ПЕН	4 ПЕНА
	Воспроизведенные слова	1 ПИРОГ	2 ПЕСНЯ	3 ПЕНАЛ	4 ПЕНА
3	Предъявляемые буквосочетания	1 П	2 ПИ	3 ПИЛ	4 ПИЛА
	Воспроизведенные слова	1 ПИЛА	2 ПИЛА	3 ПИЛА	4 ПИЛА
4	Предъявляемые буквосочетания	1 П	2 ПЛ	3 ПЛА	4 ПЛАЩ
	Воспроизведенные слова	1 ПИР	2 ПЛОВ	3 ПЛАТОК	4 ПЛАЩ
5	Предъявляемые буквосочетания	1 П	2 ПО	3 ПОН	4 ПОНИ
	Воспроизведенные слова	1 ПЛОТ	2 ПОРТРЕТ	3 ПОНЕДЕЛЬНИК	4 ПОНИ
6	Предъявляемые буквосочетания	1 П	2 ПУ	3 ПУС	4 ПУСК
	Воспроизведенные слова	1 ПОТ	2 ПУРГА	3 ПУСТЫРЬ	4 ПУСК
7	Предъявляемые буквосочетания	1 П	2 ПР	3 ПРИ	4 ПРИЗ
	Воспроизведенные слова	1 ПРЕСС	2 ПРЕСС	3 ПРИСТАНЬ	4 ПРИЗ
8	Предъявляемые буквосочетания	1 П	2 ПЫ	3 ПЫЛ	4 ПЫЛЬ
	Воспроизведенные слова	1 ПОЛЕ	2 ПЫТКА	3 ПЫЛ	4 ПЫЛЬ

Поиск слова, заданного полным набором признаков, начинается с просмотра списка первого слоя. В этом списке отыскивается буква,

имеющая значение, соответствующее значению первого признака (начальной буквы слова) в заданном наборе. По найденному значению признака вычисляется адрес подсписка, отходящего от найденной буквы. В этом подсписке находится буква, имеющая значение, соответствующее значению второго признака, и т. д.

Если число значений каждого признака равно k , то в любом подсписке i -го слоя для нахождения искомого значения признака необходимо выполнить $\mu_i k$ сравнений. Так как в подсписке может быть только одна буква с искомым значением признака, то просмотр каждого подсписка нужно вести до встречи с буквой, имеющей заданное значение проверяемого признака, и поэтому $\mu_i \leq 1$.

Общее количество сравнений S , которое необходимо выполнить для нахождения слова, заданного последовательностью из n признаков, равно

$$S = \sum_{i=1}^n \mu_i k.$$

Из описанного выше психологического эксперимента (пункт 2) видно, что для сведения μ_i , а следовательно, и S , к минимуму, необходимо располагать буквы в подсписках древовидной структуры в порядке убывания их веса.

Формат списка выбирается с таким расчетом, чтобы весь список помещался в полную ячейку машины «Урал-2», имеющую 40 двоичных разрядов. Каждый список состоит из членов списка (ЧС) и соответствующего каждому ЧС признака конечной буквы (КБ) слова. В свою очередь, ЧС состоит из кода буквы последовательного списка (БПС) и веса БПС (ВБПС). Величина ВБПС определяется частотой встречаемости этой буквы в словах входной информации.

Ниже показан формат последовательного списка.

ЧС1			ЧС2			ЧС3			ЧС4		
КБ1	БПС1	ВБПС1	БПС2	ВБПС2	БПС3	ВБПС3	БПС4	ВБПС4	КБ2	КБ3	КБ4

Рассмотрев структуру древовидного словаря, перейдем к описанию типовых процедур, используемых при составлении алгоритма поиска слов. Сам алгоритм представлен в виде описания типовой процедуры. Описания процедур приводятся не изолированно, а в виде перечня описаний процедур, входящих в состав описаний некоторого блока программы.

Идентификаторы НБ, ВНБ, ЗПБПС, ЗП, ПСДС, АНС, БВС, БВЫСЛ, СЧС, АПС, АРПС и БСМВ обозначают соответственно начальную букву, вес начальной буквы, значения признаков букв подсписка, значение признака, подсписк древовидного словаря, адрес начального списка, букву входного слова, букву выходного слова, счетчик числа сравнений, адрес последовательного списка, адрес рабочего последовательного списка и букву с максимальным весом, а идентификаторы n, k, v — рабочие переменные.

Начало целые f, r, t ; ввод (f); ввод (r); ввод (t);

начало список последовательный список;

составной ЧС; целые БПС вид 1(6), ВБПС вид 1(3) уровень: логический КБ уровень;

список ВХОДНОЕ СЛОВО; целые БВС вид 1(6) уровень;

список НАЧАЛЬНЫЕ БУКВЫ; составной ЧС; целые НБ вид 1(6),

ЕНБ вид 1 (3) уровень; логический КБ уровень;
 целые массивы ЗП [1:f], АНС [1:f], БВС [1:f], БПС [1:f]
 БВЫХСЛ [1:f], ЗПБПС [1:f],
 целые АПС вид 1(20), АРПС вид 1(20), АРПС вид 1(20),
 ПСДС вид 1(20), n вид 1(20), k вид 1(20), v вид 1(20),
 БСМВ вид 1(20), СЧС вид 1(20);
 — ПОСЛЕДОВАТЕЛЬНЫЙ СПИСОК $\neg$ = АПС = АРПС1 = АРПС2 =
 ПСДС; процедура ДЛИНВХОДСЛОВ (l , b);
 примечание производится определение длины входного слова (ДЛИН-
 ВХОДСЛОВ). Число букв входного слова (ВС) присваивается
 величине n ;
 значения l , b ; целые l , b ;
 ВХОДНОЕ СЛОВО $\neg$ = АПС;
 начало целые i , m ;
 $m := l + b$;
 для $i := 1$ шаг 1 до f цикл
 начало если $(l:m) \sqsubset$ АПС $\neg$ = 0 то
 $n := i - 1$ перейти к M конец;
 $l := m + 1$; $m := l + b$ конец
 M : конец
 процедура РАЗДЕЛЕНИЕ (p , l , b , c , РЯ);
 примечание производится разделение последовательного списка
 на БПС или на ВБПС. Значение БПС или ВБПС присваивается
 величинам массива РЯ;
 значения p , l , b , c , целые p , l , b , c ; массив РЯ;
 начало целые i , m ;
 $m := l + b$;
 для $i := 1$ шаг 1 до p цикл
 начало РЯ [i] := $(l:m) \sqsubset$ АПС $\neg$; $l := m + c$;
 $m := l + b$ конец
 конец;
 процедура УЗНАВАНИЕ (p , РЯ);
 примечание производится поиск буквы входного слова (БВГ) в списке
 буквы которого расположены в ячейках массива РЯ;
 значение p ; целое p ; массив РЯ;
 начало целое i ;
 для $i := 1$ шаг 1 до p цикл
 если РЯ [i] = БВС [v] то
 $k := i$; перейти к C конец;
 перейти к B ;
 конец;
 процедура ВЫЧАД (h , A);
 примечание производится вычисление адреса (ВЫЧАД) подписка в $h + 1$
 слое поискового дерева по h значениям признаков. Значения при-
 знаков расположены в ячейках массива ЗП;
 значения h , целые h , A ;
 начало целые i , a ;
 $a := 0$;
 для $i := 1$ шаг 1 до h цикл
 $a := a \times r + \text{ЗП}[i]$;
 $A := \text{АНС}[v] + 2 \times a$
 конец;
 процедура МАКСВЕГ (p , МВ);
 примечание производится выделение в последовательном списке с ад-
 ресом АПГ буквы с максимальным весом (МАКСВЕС). Список пред-

варительно разделен на буквы, которые хранятся в ячейках массива БПС, и на их веса, которые размещены в ячейках массива ВБПС. Значение буквы с максимальным весом присваивается величине БСМВ, а место расположения этой буквы в последовательном списке-величине k ;

значение p ; целые p , МВ;

начало целое i ;

МВ: = 0;

для $i := 1$ шаг 1 до p цикл

если ВБПС [i] $\geq$ МВ то

начало МВ: = ВБПС [i]; $k := i$;

БСМВ: = БПС [i] конец

конец;

процедура РАСБУКВПОУБВЕС (p , d , b , c , q , РАЗДЕЛЕНИЕ, МАКСВЕС);

примечание производится распределение букв по убывающим весам (РАСБУКВПОУБВЕС). Буквы заданы в виде последовательного списка с адресом, равным АПС. Новой последовательности присваивается адрес АРПС1;

значения p , d , b , c , q ; целые p , d , b , c , q ; процедуры РАЗДЕЛЕНИЕ, МАКСВЕС;

начало целые i , l , m , j , $l1$, $l2$, $b1$, $b2$, $c1$, $c2$;

РАЗДЕЛЕНИЕ (p , $l1$, $b1$, $c1$, БПС);

РАЗДЕЛЕНИЕ (p , $l2$, $b2$, $c2$, ВБПС);

для $i := 1$ шаг 1 до p цикл

начало МАКСВЕС (p , МВ); $l := (i - 1) \times d + q$;

$m := l + b$; ($l : m$) $\lfloor$ АРПС1 $\rfloor :=$ БСМВ; $l := m + 1$;

$m := l + c$; ($l : m$) $\lfloor$ АРПС1 $\rfloor := k$;

для $j := 1$ шаг 1 до p цикл

если $j = k$ то

начало ВБПС [j] := 0 перейти к F конец;

F : конец

конец;

процедура ПОИСК 1 (ДЛИНВХОДСЛОВ, РАЗДЕЛЕНИЕ, УЗНАВАНИЕ, ВЫЧАД);

примечание производится поиск входного слова в словаре. Буквы в подписках словаря расположены без учета их веса. Количество сравнений, которое было выполнено для нахождения входного слова в словаре, присваивается величине СЧС;

процедуры ДЛИНВХОДСЛОВ, РАЗДЕЛЕНИЕ, УЗНАВАНИЕ, ВЫЧАД;

начало $\lfloor$ АПС $\rfloor :=$ ВХОДНОЕ СЛОВО; ДЛИНВХОДСЛОВ (1,5);

РАЗДЕЛЕНИЕ (n , 1, 5, 1, БВС); $\lfloor$ АПР $\rfloor :=$ НАЧАЛЬНЫЕ БУКВЫ; СЧС: = 0;

для $v := 1$ шаг 1 до n цикл

начало РАЗДЕЛЕНИЕ (r , 1, 5, 4, БПС); УЗНАВАНИЕ (r , БПС);

С: СЧС: = СЧС + k ; ЗП [v] := $k - 1$;

ВЫЧАД (v , АПС) конец; вывод (СЧС);

V : конец;

процедура САМООБУЧЕНИЕ (РАСБУКВПОУБВЕС);

примечание производится распределение букв в j -м ($j = 1, 2, \dots, r^{i-1}$) подписке i -го ($i = 1, 2, \dots, t$) слоя древовидного словаря в порядке убывания их веса;

процедура РАСБУКВПОУБВЕС;

начало целые i , j , z ;

$z := 0$;

для $i := 1$ шаг 1 до t цикл
 начало $z := z \times r + \text{entier}\left(\frac{1}{i}\right)$;
 для $j := 1$ шаг 1 до z цикл
 если $\lfloor \text{ПСДС}[i, j] \rfloor \neq 0$ то
 начало $\lfloor \text{АПС} \rfloor := \lfloor \text{ПСДС}[i, j] \rfloor$; РАСБУКВПОУБВЕС
 $(r, d, b, c, q, \text{РАЗДЕЛЕНИЕ, МАКСВЕС}), \lfloor \text{ПСДС}[i, j] \rfloor :=$
 $= \lfloor \text{АРПС1} \rfloor$
 конец
 конец
 конец;
 процедура ПОИСК2 (ДЛИВХОДСЛОВ, РАЗДЕЛЕНИЕ, УЗНАВАНИЕ,
 ВЫЧАД);
 примечание производится поиск входного слова в словаре после его
 «самообучения». Количество сравнений, которое было выполнено для
 нахождения входного слова в словаре, присваивается величине СЧС;
 процедуры ДЛИВХОДСЛОВ, РАЗДЕЛЕНИЕ, УЗНАВАНИЕ,
 ВЫЧАД;
 начало $\lfloor \text{АПС} \rfloor := \text{ВХОДНОЕ СЛОВО}$; ДЛИВХОДСЛОВ (1,5);
 РАЗДЕЛЕНИЕ ($n, 1, 5, 1, \text{БВС}$); $\lfloor \text{АПС} \rfloor := \text{НАЧАЛЬНЫЕ}$
 БУКВЫ; СЧС := 0;
 для $v := 1$ шаг 1 до n цикл
 начало РАЗДЕЛЕНИЕ ($r, 1, 5, 4, \text{БПС}$); РАЗДЕЛЕНИЕ ($r, 7, 2,$
 $7, \text{ЗПБПС}$);
 УЗНАВАНИЕ ($r, \text{БПС}$);
 $\text{СЧС} := \text{СЧС} + k$; $\text{ЗП}[v] := \text{ЗПБПС}[v] - 1$;
 ВЫЧАД ($v, \text{АПС}$) конец; вывод (СЧС)
 В : конец
 конец
 конец

Приведенные алгоритмы ПОИСК1 и ПОИСК2 были реализованы на ЭВМ «УРАЛ-2» в вычислительном центре Харьковского института радиозлектроники.

Древовидный словарь был построен путем распределения слов 40 предложений в памяти ЭВМ. Входные предложения составлены из 72 различных слов. Согласно описанным алгоритмам осуществлялся ПОИСК1 слов восьми предложений, состоящих из 31 (17) четырехбуквенного слова, семи (четыре) трехбуквенных и одного однобуквенного слова. В скобках указано число различных слов. При поиске слов без «самообучения» словаря было выполнено 306 сравнений, т. е. $S_1 = 306$. После «самообучения» словаря ПОИСК2 дал $S_2 = 263$.

ЛИТЕРАТУРА

1. Л. С. Лозинский. Внутренняя сортировка информации при ограниченной памяти. Ж. «Кибернетика», 1965, № 3.
2. А. Н. Лук, С. А. Талаев. Системы памяти с адресацией по содержанию. Сб. «Некоторые проблемы биокрибернетики и применение электроники в биологии и медицине», вып. 2, К., 1967.
3. Н. М. Амосов и др. Эвристические методы в моделировании психики. Сб. «Некоторые проблемы биокрибернетики и применение электроники в биологии и медицине», вып. 3, К., 1967.
4. А. И. Китов. Программирование информационно-логических задач. «Сов. радио», М., 1967.