Лабораторная работа №4

«Оптимальное кодирование. Алгоритм Хаффмана»

Вариант 1

Цель работы: освоить алгоритм Хаффмана. Научиться сжимать сообщения с помощью алгоритма Хаффмана.

1

Условие: Составить коды для каждого символа алфавита с помощью алгоритма Хаффмана

a	A	B	C	D	E	F
p	0,25	0,05	0,1	0,25	0,05	0,3

Решение:

Сперва упорядочим входную последовательность символов по невозрастанию их вероятностей:

a	F	A	D	C	B	E
p	0,3	0,25	0,25	0,1	0,05	0,05

Далее найдём коды для каждого символа используя следующий алгоритм (Хаффмана):

1. Берём два последних символа из упорядоченной последовательности и объединяем их в один «узел», суммируя вероятности.
2. Сортируем последовательность по невозрастанию вероятностей, учитывая узел созданный на прошлом шаге, но не учитывая его составляющие.
3. Повторяем п. 1, пока в последовательности не останется всего один узел, который будет корнем построенного дерева.
4. Для того, чтобы закодировать символ, мы должны начать из корня, а далее прослеживать «вверх» по дереву все повороты ветвей. И если мы делаем левый поворот, то запоминаем 0-й бит, и, аналогично, 1-й бит для правого. Таким образом, доходя до нужного символа, получаем его код.

<table>
<tr><td rowspan="3">F
0,3
0</td><td rowspan="3">A
0,25
1</td><td rowspan="3">D
0,25
0</td><td rowspan="2">C
0,1
0</td><td>B
0,05
0</td><td>E
0,05
1</td></tr>
<tr><td colspan="2">BE
0,1
1</td></tr>
<tr><td colspan="3">CBE
0,2
1</td></tr>
<tr><td colspan="2">FA
0,55
0</td><td colspan="4">DCBE
0,45
1</td></tr>
<tr><td colspan="6">FADCBE
1</td></tr>
</table>

Получаем следующие коды для символов:

- F – 00
- A – 01
- D – 10
- C – 110
- B – 1110
- E – 1111

2

Условие: Закодировать сообщение (ADADBDCBBDCABFDAFCEB), используя коды для символов. Вычислить среднюю длину символа. Вычислить энтропию алфавита. Сравнить длину и энтропию. Сделать выводы.

Решение:

Закодируем сообщение, используя найденные выше коды:

ADADBDCBBDCABFDAFCEB =

0110011011101011011101110101100111100010010011011111110

Вычислим среднюю длину символа, то есть среднюю длину его оптимального неравномерного кода:

$$L_{ср} = \sum L_i \times p_i = 2 \times 0{,}3 + 2 \times 0{,}25 + 2 \times 0{,}25 + 3 \times 0{,}1 + 4 \times 0{,}05 + 4 \times 0{,}05 = \\ = 2{,}3 \text{ бит}$$

Подсчитаем энтропию алфавита:

$$H = -\sum p_i \times \log_2(p_i) = 0,3 \times \log_2(0,3) + 0,25 \times \log_2(0,25) + 0,25 \times \log_2(0,25) + \\ + 0,1 \times \log_2(0,1) + 0,05 \times \log_2(0,05) + 0,05 \times \log_2(0,05) = 2,285 \ \text{бит/сим}$$

Эффективность оптимального кодирования тем выше, чем больше средняя длина кода (символа) стремится к энтропии алфавита. Высчитаем коэффициент эффективности:

$$K_э = \frac{H}{L_{ср}} = \frac{2,285}{2,3} = 0,993$$

Поскольку коэффициент эффективности недалёк от единицы, данное кодирование действительно оптимально и эффективно.

3

Условие: Составить список биграмм для данного алфавита. Вычислить вероятность каждой биграммы. Составить коды для каждой биграммы с помощью алгоритма Хаффмана.

Решение:

Составим список биграмм:

AA, AB, AC, AD, AE, AF, BA, BB, BC, BD, BE, BF, CA, CB, CC, CD, CE, CF, DA, DB, DC, DD, DE, DF, EA, EB, EC, ED, EE, EF, FA, FB, FC, FD, FE, FF

Вычислим вероятность каждой биграммы:

$$P(AA) = P(A) \times P(A) = 0,25 \times 0,25 = 0,0625$$
$$P(AB) = P(A) \times P(B) = 0,25 \times 0,05 = 0,0125$$
$$P(AC) = P(A) \times P(C) = 0,25 \times 0,1 = 0,025$$
$$P(AD) = P(A) \times P(D) = 0,25 \times 0,25 = 0,0625$$
$$P(AE) = P(A) \times P(E) = 0,25 \times 0,05 = 0,0125$$
$$P(AF) = P(A) \times P(F) = 0,25 \times 0,3 = 0,075$$
$$P(BA) = P(B) \times P(A) = 0,05 \times 0,25 = 0,0125$$
$$P(BB) = P(B) \times P(B) = 0,05 \times 0,05 = 0,0025$$
$$P(BC) = P(B) \times P(C) = 0,05 \times 0,1 = 0,005$$
$$P(BD) = P(B) \times P(D) = 0,05 \times 0,25 = 0,0125$$
$$P(BE) = P(B) \times P(E) = 0,05 \times 0,05 = 0,0025$$
$$P(BF) = P(B) \times P(F) = 0,05 \times 0,3 = 0,015$$
$$P(CA) = P(C) \times P(A) = 0,1 \times 0,25 = 0,025$$
$$P(CB) = P(C) \times P(B) = 0,1 \times 0,05 = 0,005$$

$P(CC) = P(C) \times P(C) = 0,1 \times 0,1 = 0,01$
$P(CD) = P(C) \times P(D) = 0,1 \times 0,25 = 0,025$
$P(CE) = P(C) \times P(E) = 0,1 \times 0,05 = 0,005$
$P(CF) = P(C) \times P(F) = 0,1 \times 0,3 = 0,03$
$P(DA) = P(D) \times P(A) = 0,25 \times 0,25 = 0,0625$
$P(DB) = P(D) \times P(B) = 0,25 \times 0,05 = 0,0125$
$P(DC) = P(D) \times P(C) = 0,25 \times 0,1 = 0,025$
$P(DD) = P(D) \times P(D) = 0,25 \times 0,25 = 0,0625$
$P(DE) = P(D) \times P(E) = 0,25 \times 0,05 = 0,0125$
$P(DF) = P(D) \times P(F) = 0,25 \times 0,3 = 0,075$
$P(EA) = P(E) \times P(A) = 0,05 \times 0,25 = 0,0125$
$P(EB) = P(E) \times P(B) = 0,05 \times 0,05 = 0,0025$
$P(EC) = P(E) \times P(C) = 0,05 \times 0,1 = 0,005$
$P(ED) = P(E) \times P(D) = 0,05 \times 0,25 = 0,0125$
$P(EE) = P(E) \times P(E) = 0,05 \times 0,05 = 0,0025$
$P(EF) = P(E) \times P(F) = 0,05 \times 0,3 = 0,015$
$P(FA) = P(F) \times P(A) = 0,3 \times 0,25 = 0,075$
$P(FB) = P(F) \times P(B) = 0,3 \times 0,05 = 0,015$
$P(FC) = P(F) \times P(C) = 0,3 \times 0,1 = 0,03$
$P(FD) = P(F) \times P(D) = 0,3 \times 0,25 = 0,075$
$P(FE) = P(F) \times P(E) = 0,3 \times 0,05 = 0,015$
$P(FF) = P(F) \times P(F) = 0,3 \times 0,3 = 0,09$

Составим коды для каждой биграммы с помощью алгоритма Хаффмана (аналогично заданию №2):

- FF – 111
- AF – 0001
- DF – 0010
- FA – 0011
- FD – 0100
- AA – 0101
- AD – 0110
- DA – 0111
- DD – 1000
- CF – 10100
- FC – 10101
- AC – 000000
- CA – 000001
- CD – 11010
- DC – 11011
- BF – 100110
- EF – 100111
- FB – 100100
- FE – 100101

- AB – 101110
- AE – 101111
- BA – 101100
- BD – 101101
- DB – 110010
- DE – 110011
- EA – 110000
- ED – 110001
- CC – 0000110
- BC – 00001010
- CB – 00001011
- CE – 00001000
- EC – 00001001
- BB – 000011110
- BE – 000011111
- EB – 000011100
- EE – 000011101

4

Условие: Закодировать сообщение, используя коды для биграмм. Вычислить среднюю длину биграммы. Разделить результат на 2. Сравнить полученное число со средней длиной для посимвольного кодирования. Сделать выводы о целесообразности кодировать сообщения поблочно.

Решение:

Закодируем сообщение используя найденные выше коды биграмм:

ADADBDCBBDCABFDAFCEB =

0110011010110100001011101101000001100110011110101000011100

Вычислим среднюю длину биграммы, то есть среднюю длину её оптимального неравномерного кода:

$$L_{ср} = \Sigma L_i \times p_i = 3 \times 0{,}09 + 4 \times 0{,}075 + 4 \times 0{,}075 + 4 \times 0{,}075 + 4 \times 0{,}075 + 4 \times 0{,}0625 + 4 \times 0{,}0625 + 4 \times 0{,}0625 + 4 \times 0{,}0625 + 5 \times 0{,}03 + 5 \times 0{,}03 + 6 \times 0{,}025 + 6 \times 0{,}025 + 5 \times 0{,}025 + 5 \times 0{,}025 + 6 \times 0{,}015 + 6 \times 0{,}015 + 6 \times 0{,}015 + 6 \times 0{,}015 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 6 \times 0{,}0125 + 7 \times 0{,}01 + 8 \times 0{,}005 + 8 \times 0{,}005 + 8 \times 0{,}005 + 8 \times 0{,}005 + 9 \times 0{,}0025 + 9 \times 0{,}0025 + 9 \times 0{,}0025 + 9 \times 0{,}0025 = 4{,}5995 \text{ бит}$$

Разделим полученное число на 2:

$$\frac{4{,}5975}{2} = 2{,}29975 \text{ бит}$$

Поскольку число получилось меньше, чем при кодировании по символьно, можно сделать вывод, что кодировать поблочно выгоднее, чем посимвольно.

5

Условие: Создать подпрограмму для составления кодов для символов по алгоритму Хаффмана.

6

Условие: Создать подпрограмму для кодирования сообщения. Подпрограмме передаётся сообщение, состоящее из символов алфавита и коды для кодирования сообщения, полученные подпрограммой из предыдущего пункта.

7

Условие: Проверить работоспособность подпрограмм, данные из п. 1 и 2 использовать как тестовые.

8

Условие: Модернизировать подпрограммы из п. 5 и 6 для случая поблочного кодирования. Создать программу, на вход которой подаются символы алфавита и их вероятности. Далее пользователь вводит размер блока (от 1 символа). В случае с блоком размера 1, имеет место посимвольное кодирование. Иначе составляются различные возможные комбинации блоков и вычисляются их вероятности. Вычисляется энтропия и средняя длина 1 символа (в поблочном случае вычисляется энтропия и длина блока, и делится на размер блока), результат выводится на экран. Далее пользователь вводит сообщение, программа кодирует его и выводит результат. Выполнить программу для блоков различного размера. Установить размер блока, на котором средняя длина символа минимальна. Сделать выводы.

Решение:

Длина блока	2	3	4	5	6
Ср. длина символа	2,3	2,29	2,28	2,27	2,25

Вывод: чем больше длина блока, тем эффективнее кодирование.