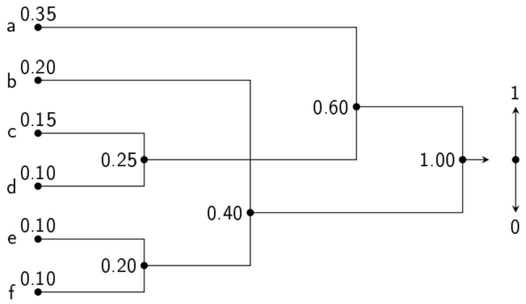
Код Хаффмана и его построение с обходом дерева

5 января 2023 г. 1:21

Код Хаффмана - наилучший неравномерный побуквенный код



a	11
b	01
c	101
d	100
e	001
f	000

$$H = -\sum_x p(x)\log p(x) = 2.4016$$
$$\bar{l} = \sum_x l(x)p(x) = 2.4500$$

Главная идея построения дерева - 2 наименьших по вероятности символа "объединяются вершиной", вероятности суммируются, это происходит до тех пор, пока не получим корень дерева (единственную вершину, где вероятность = 1). Далее, проходя по этому дереву от корня к листьям, можно найти код символа (если 0й потомок - 0, 1й потомок - 1)

В каком случае код Хаффмана не избыточный? Т.е. когда он достигает энтропии?
Избыточность $R = \overline{l} - H = 0$
$\sum_i p_i l_i = -\sum_i p_i log\left(p_i\right)$
Ответ: в случае, если вероятности - степени двойки
Например, если вероятности 1/2, 1/4, 1/4, то избыточность = 0

Есть двоичный источник, т.е. алфавит {0,1}
Вероятность нуля 0,9
Вероятность единицы 0,1
(Максимальная энтропия была бы при вероятностях 0,5 и была бы равна 1)
Для этого случая будет код Хаффмана {0,1}
То есть по сути принимаем один бит и кодируем его так же одним битом
Это тот самый редкий случай, когда избыточность кода Хаффмана большая
Еще хуже будет, если вероятности = 0 и 1, тогда энтропия будет 0, и кодер будет вынужден тратить по 1 биту постоянно
Если сталкиваемся с подобной ситуацией, можем применить расширение алфавита
Например, от алфавита {0,1} переходим к алфавиту {1, 01, 001, 000}, считаем вероятности: {0,1; 0,9*0,1; 0,9*0,9*0,1; 0,9*0,9*0,9} и строим код Хаффмана для него