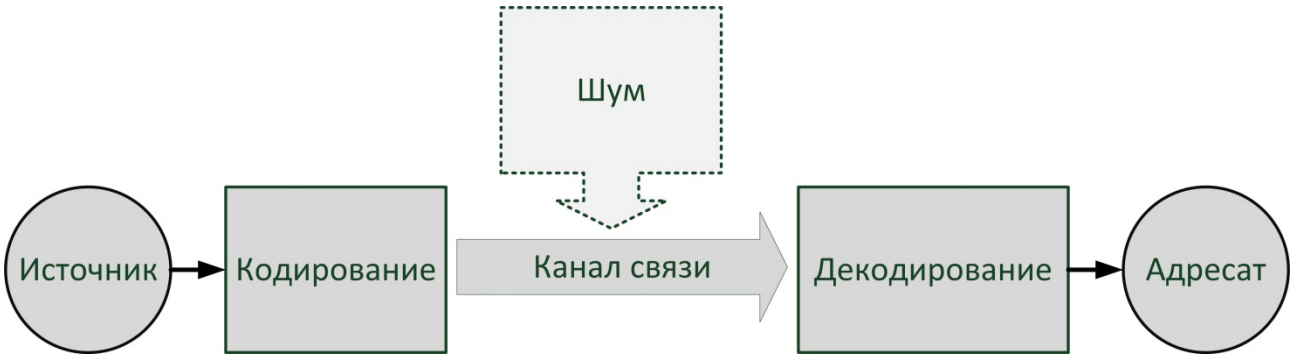
Тема 4.1. Основные понятия теории кодирования. Оптимальные коды

Аннотация: Коды. Алфавитное кодирование. Разделимые коды. Критерий однозначности декодирования. Неравенство Макмиллана. Условие существования разделимого кода с заданными длинами кодовых слов. Оптимальные коды и их свойства. Лемма о редукции. Алгоритмы Фано и Хаффмана построения оптимальных кодов.

Теория кодирования - раздел теории информации, связанный с задачами кодирования и декодирования сообщений, поступающих к потребителям и посылаемых из источников информации.

При передаче данных часто возникает необходимость кодирования пересылаемой информации. На рисунке ниже представлена стандартная модель передачи информации.



Выделяю две основные цели кодирования

1. Повышение эффективности передачи данных, за счет достижения максимальной скорости передачи данных.
2. Повышение помехоустойчивости при передаче данных.

С необходимостью кодирования данных впервые столкнулись в середине XIX века, вскоре после изобретения телеграфа. Каналы были дороги и ненадежны, что потребовало минимизации стоимости и повышения надежности передачи телеграмм. С 1845 для указанных целей применялись специальные кодовые книги, содержащие таблицы, с помощью которых можно было вручную заменять часто встречающиеся длинные последовательности слов более короткими кодами. Тогда же для проверки правильности передачи стали использовать контроль четности.

С развитием каналов связи потребовался более эффективный механизм контроля. Первое теоретическое решение проблемы передачи данных по зашумленным каналам было предложено Клодом

Шенноном. В 1948 году К. Шеннон опубликовал работу «Математическая теория передачи сообщений», где показал, что если пропускная способность канала выше энтропии источника сообщений, то сообщение можно закодировать так, что оно будет передано без излишних задержек.

Основные понятия и определения

Для передачи в канал связи сообщения преобразуются в сигналы.

Пусть $A=\{a_1,a_2,...,a_r\}$ – исходный алфавит, $B=\{b_1,b_2,...,b_m\}$ – кодирующий алфавит и $A^*=\varnothing\cup A\cup A^2\cup A^3\cup...\cup A^n\cup...$, $B^*=\varnothing\cup B\cup B^2\cup B^3\cup...\cup B^n\cup...$, тогда ***алфавитным кодированием*** $A^*\to B^*$ назовём отображением $\varphi: A\to B^*$ такое, что $a_i\to B_i$. Т.е. ***кодирование*** - это преобразование сообщений в сигнал (кодовые комбинации).

Алфавитное кодирование – кодирование, выполняющееся поразрядно.

Любая конечная последовательность символов из B называется ***словом*** в алфавите B.

Кодирование $A^*\to B^*$ называется ***взаимно однозначным (декодируемым, разделимым)***, если для любых слов $\overline{a}_1\in A^*$ и $\overline{a}_2\in A^*$, если $\overline{a}_1\neq\overline{a}_2\Rightarrow\varphi(\overline{a}_1)\neq\varphi(\overline{a}_2)$.

Код - система соответствия между элементами сообщений и кодовыми комбинациями. ***Кодер*** - устройство, осуществляющее кодирование. ***Декодер*** - устройство, осуществляющее обратную операцию, т.е. преобразование кодовой комбинации в сообщение.

Код называется ***равномерным***, если длины всех его кодовых слов одинаковы.

Любой равномерный код является взаимно однозначным.

Если слово имеет вид $\alpha_1\alpha_2$, тогда подслово α_1 называется ***префиксом***, а α_2 – ***суффиксом*** слова $\alpha_1\alpha_2$.

Код называется ***префиксным***, если никакое кодовое слово не является началом другого.

Свободным (или ***разделимым***) называется код, который декодируется однозначно. Соответственно, код, допускающий неоднозначное декодирование, называется ***несвободным (или неразделимым***).

Любое префиксное кодирование является взаимно однозначным.

Свободный код может быть непрефиксным**.**

Пример: Код V = {1, 10} – не является префиксным по определению, но в то же время он не допускает неоднозначного декодирования, т.е. он свободный.

Слово $\overline{b} \in B^*$ называется ***неприводимым***, если $\overline{b}$ декодируется неоднозначно, однако, при выбрасывании из $\overline{b}$ лобового связанного непустого куска получается слово, которое декодируется не более, чем одним способом.

Рассмотрим прямую теорему о спектре свободного кода:

Теорема (Неравенство Макмиллана)

Пусть задано кодировании $\varphi: a_i \to B_i \ \ (i = 1,2,...,r)$ *и пусть в кодирующем алфавите* $B \to q$ *букв и длина* $\text{длина}(B_i) = l_i \ (i = 1,2,...,r)$. *Тогда если* φ *взаимно однозначна, то* $\sum_{i=1}^{r} \frac{1}{q^{l_i}} \le 1.$

Доказательство:

➢ Положим $x = \sum_{i=1}^{r} \frac{1}{q^{l_i}}$.Тогда для любого натурального числа n

$$x^n = \left(\sum_{i=1}^{r} \frac{1}{q^{l_{i1}}}\right)\left(\sum_{i=1}^{r} \frac{1}{q^{l_{i2}}}\right)...\left(\sum_{i=1}^{r} \frac{1}{q^{l_{in}}}\right) = \sum_{i_1=1}^{r}\sum_{i_2=1}^{r}...\sum_{i_n=1}^{r} \frac{1}{q^{l_{i_1}+l_{i_2}+...+l_{i_n}}}.$$

Обозначая $l_{\max} = \max_{1 \le i \le r} l_i$,получим, что эта сумма равна $\sum_{k=1}^{n \cdot l_{\max}} \frac{c_k}{q^k}$.

Докажем теперь, что $c_k \le q^k \ (\forall k)$: за c_k обозначен, очевидно, число наборов $(i_1,...,i_n)(1 \le i_j \le r)$, для которых $l_{i_1} + l_{i2} + ... + l_{i_n} = k$. Но такой сумме соответствует слово $B_{i_1}, B_{i2}...B_{i_n}$ и

$$\text{длина}\left(B_{i_1}, B_{i_2}...B_{i_n}\right) = l_{i_1} + l_{i2} + ... + l_{i_n} = k.$$

В силу того, что кодирование взаимно однозначно, различным наборам, соответствуют различные сообщения, а различных сообщений длины k в алфавите из q букв не более $q^k \Rightarrow c_k \le q^k \ (\forall k)$.

Тогда получим, что

$$x^n = \sum_{k=1}^{n \cdot l_{\max}} \frac{c_k}{q^k} \le \sum_{k=1}^{n \cdot l_{\max}} 1 = n l_{\max} \Leftrightarrow x \le \sqrt[n]{n l_{\max}}, \forall n.$$

Устремляя n к бесконечности, получаем $x < 1$. Теорема

доказана. ◅

Также можно рассмотреть обратную теорему о спектре свободного кода:


Теорема

Если $|B| = q$ *и натуральные числа* $l_1, l_2, ..., l_r$ *удовлетворяет неравенству* $\sum_{i=1}^{r} \frac{1}{q^{l_i}} \le 1$, *то существует префиксный код* $B_1, B_2, ..., B_r$ *такой, что* $длина(B_i) = l_i (i = 1, 2, ..., r)$.

Доказательство:

➢ Пусть $\sum_{i=1}^{r} \frac{1}{q^{l_i}} \le 1$ и для любого k существует ровно d_k таких i, что $l_i = k$, то есть $\sum_{k=1}^{l_{\max}} \frac{d_k}{q^k} \le 1$. Тогда надо построить префиксный код, в котором ровно d_1 слов длины 1, d_2 слов длины 2, и т. д., $d_{l_{\max}}$ слов длины $l_{\max}$. Имеем $\forall m (1 \le m \le l_{\max}) \sum_{k=1}^{m} \frac{d_k}{q^k} \le 1$, или, что то же самое

$$\frac{d_1}{q} + \frac{d_2}{q^2} + ... + \frac{d_{m-1}}{q^{m-1}} + \frac{d_m}{q^m} \le 1 \Leftrightarrow d_m \le q^m - \left(d_1 q^{m-1} + d_2 q^{m-2} + ... + d_{m-1} q \right).$$

Рассмотрим это неравенство для $m = 1: d_1 \le q$. Для слов длины 1 всего предоставляется возможностей в алфавите мощности q – ровно q вариантов. После выбора d_1 слов длины 1 рассмотрим неравенство для $d_2 \le q^2 - d_1 q$.

Всего слов длины $2 - q^2$, однако все они могут начинаться лишь с тех букв, которые не были выбраны в качестве слов длины 1, следовательно, остается ровно $q^2 - d_1 q$ возможностей выбрать слова длины 2, что удовлетворяет условию

$d_2 \le q^2 - d_1 q$. Пусть уже выбраны d_1 слов длины 1, d_2 слов длины 2, и т. д., d_{m-1} слов длины $m-1$. Тогда для слов длины m разрешено возможностей не меньше, чем $q^m - d_{m-1} q - d_{m-2} q^2 - ... - d_2 q^{m-2} - d_1 q^{m-1}$,

Рассмотрим это неравенство для m=1: $d_1 \le q$. Для слов длины 1

всего предоставляется возможностей в алфавите мощности q – ровно q вариантов. После выбора d_1 слов длины 1 рассмотрим неравенство для $m = 2 : q^2 - d_1 q$.

Всего слов длины $2 - q^2$,однако все они могут начинаться лишь с тех букв, которые не были выбраны в качестве слов длины 1, следовательно, остается ровно $q^2 - d_1 q$ возможностей выбрать слова длины 2, что удовлетворяет $d_2 \le q^2 - d_1 q$.

Пусть уже выбраны d_1слов длины 1, d_2 слов длины 2, и т. д., d_{m-1} слов длины $m-1$. Тогда для слов длины m разрешено возможностей не меньше, чем $q^m - d_{m-1} q - d_{m-2} q^2 - ... - d_2 q^{m-2} - d_1 q^{m-1}$, что удовлетворяет условию. Теорема доказана. ◅

Следствие. Если существует взаимно однозначное кодирование со спектром длин слов $l_1, l_2, ..., l_r$ в алфавите B, то в B существует префиксный код с тем же спектром длин слов.

Оптимальные коды, и их свойства.

Будем рассматривать кодирование $A^* \to \{0,1\}^*$. Пусть известны некоторые частоты $p_1, p_2, ..., p_k$ появления символов кодируемого алфавита в тексте:

$$\begin{array}{cccc} p_1 & - & a_1 & \to B_1 - l_1 \\ p_2 & - & a_2 & \to B_2 - l_2 \\ \vdots & & \vdots & \vdots \\ p_k & - & a_k & \to B_k - l_k \end{array},$$

где l_j – длина j-го кодового слова, $p_1 + p_2 + ... + p_k = 1, p_j > 0$.

Ценой (стоимостью) кодирования φ называется функция $c(\varphi) = \sum_{i=1}^{k} p_i l_i$. При кодировании текста длины *N* его длина становится примерно равной

$$\sum_{i=1}^{k} (N p_i) l_i = N \sum_{i=1}^{k} p_i l_i.$$

Оптимальным кодом для заданного набора частот *P*=(p_1, …, p_k) будем называть код *B*={B_1, …, B_k} с наименьшей стоимостью кодирования

Можно доказать утверждение о том, что если существует оптимальный код, то существует оптимальный префиксный код с тем же спектром длин слов.

Теорема (редукции)

Пусть заданы 2 набора частот и 2 набора слов:

$$\varphi:\begin{matrix}p_1,p_2,..,p_k\\B_1,B_2,..,B_k\end{matrix} \quad \text{и} \quad \varphi':\begin{matrix}p_1,p_2,..,p_{k-1},p',p''\\B_1,B_2,..,B_{k-1},B_k0,B_k1\end{matrix}$$

1)Тогда если φ'-оптимальное префиксное кодирование, то и φ – оптимальное префиксное кодирование.

2)Если же φ – оптимальное префиксное кодирование и $p_1 \geq p_2 \geq ... \geq p_{k-1} \geq p_k$, то φ'- также оптимальное префиксное кодирование.

Доказательство:

➢ 1) Очевидно, из префиксности φ' следует φ. Допустим, что φ не оптимально. Тогда существует префиксный код $\varphi_1 : c(\varphi_1) < c(\varphi)$ для тех же распределений частот. Пусть

$$\varphi_1:\begin{matrix}p_1,p_2,..,p_k\\D_1,D_2,..,D_k\end{matrix} \quad \text{и} \quad \varphi_1':\begin{matrix}p_1,p_2,..,p_{k-1},p',p''\\D_1,D_2,..,D_{k-1},D_k0,D_k1\end{matrix}$$

Очевидно, кодирование φ_1' также является префиксным и

$$\begin{cases}c(\varphi_1)=c(\varphi)+p_k\\c(\varphi_1')=c(\varphi_1)+p_k\end{cases} \Rightarrow \{c(\varphi_1)<c(\varphi)\} \Rightarrow$$

$$c(\varphi_1')=c(\varphi_1)+p_k<c(\varphi)+p_k=c(\varphi_1').$$

Следовательно, φ' не является оптимальным кодированием, что противоречит условию. Остается предположить, φ оптимально.

2) Пусть φ – оптимальное префиксное кодирование и $p_1 \geq p_2 \geq ... \geq p_{k-1} \geq p_k$. Допустим, что φ' не оптимально. Тогда для частот $p_1, p_2, ..., p_{k-1}, p_k$ существует оптимальное префиксное кодирование $\varphi_1' : D_1, ..., D_{k-1}, D_k0, D_k1$ и $c(\varphi_1') < c(\varphi)$. Тогда для частот $p_1, p_2, ..., p_k$ рассмотрим кодирование $\varphi_1 : D_1, ..., D_{k-1}, D_k$ Получим

$$c(\varphi_1)=c(\varphi_1')-p_k=c(\varphi)\Rightarrow c(\varphi_1)<c(\varphi)$$

и φ не оптимально, что противоречит условию.

Теорема доказана. ➢

Алгоритмы Фано и Хаффмана

Алгорим Шеннона-Фано

Алгоритм Шеннона-Фано — один из первых алгоритмов сжатия, который впервые сформулировали американские учёные Клод Шеннон и Роберт Фано. Алгоритм использует коды переменной длины: часто встречающийся символ кодируется кодом меньшей длины, редко встречающийся — кодом большей длины. Код Фано — префиксный.

Алгоритм:

1. Символы первичного алфавита выписывают по убыванию вероятностей.
2. Символы полученного алфавита делят на две части, суммарные вероятности символов которых максимально близки друг другу.
3. «Верхней» части ставим в соответствие префиксную часть кода 0, а «нижней» - 1.
4. Полученные части рекурсивно делятся и их частям назначаются соответствующие двоичные цифры в префиксном коде.

Пример: закодировать по Фано сообщения, имеющие следующие вероятности:

символ	a	e	b	d	c	f	g
вероятность	0,4	0,2	0,1	0,1	0,1	0,05	0,05

Решение:

Проверим выполнимость необходимого условия:

0,4 + 0,2 + 0,1 + 0,1 + 0,1+ 0,05 + 0,05 = 1.

Расположим элементы в порядке убывания вероятностей. Затем будем последовательно делить, не меняя порядка, все элементы на две группы, максимально близкие по суммарной вероятности (т.е. модуль разности сумм вероятностей первой и второй группы должен быть минимальных из всех возможных разбиений на группы). Для «верхней» группы будем ставить значение 0, «нижней» - 1:

Символ	Вероятность	Шаг 1	Шаг 2	Шаг 3	Шаг 4	Полученный код
a	0,4	0				0
e	0,2	1	0	0		100
b	0,1			1		101

d	0,1		1	0	0	1100
c	0,1				1	1101
f	0,05			1	0	1110
g	0,05				1	1111

Найдем стоимость кода (средняя длина кодового слова). Он является критерием степени оптимальности кодирования. Вычислим ее в нашем случае.

$$l = \sum_{i=1}^{7} l_i \cdot p_i = 1 \cdot 0,4 + 3 \cdot 0,2 + 3 \cdot 0,1 + 4 \cdot \left(0,1 \cdot 2 + 0,05 \cdot 2\right) = 2,5.$$

Также можно строить с помощью дерева. Его построение начинается от корня. Всё множество кодируемых элементов разбивается на два подмножества с примерно одинаковыми суммарными вероятностями. Эти подмножества соответствуют двум вершинам второго уровня, которые соединяются с корнем. Далее каждое из этих подмножеств разбивается на два подмножества с примерно одинаковыми суммарными вероятностями. Им соответствуют вершины третьего уровня. Если подмножество содержит единственный элемент, то ему соответствует концевая вершина кодового дерева; такое подмножество разбиению не подлежит. Подобным образом поступаем до тех пор, пока не получим все концевые вершины. Ветви кодового дерева размечаем символами 1 и 0. Тем не менее, мы считаем, что метод «дерево» не столь нагляден как построение таблиц и не имеет существенных преимуществ перед ним. Как говорится, дело вкуса.

Существенное замечание: алгоритм Фано не гарантирует получение оптимального кода!

Алгоритм Хаффмана

Алгоритм Хаффмана — жадный алгоритм оптимального префиксного кодирования алфавита с минимальной избыточностью. Был разработан в 1952 году аспирантом Массачусетского технологического института Дэвидом Хаффманом при написании им курсовой работы. В настоящее время используется во многих программах сжатия данных. В отличие от алгоритма Шеннона-Фано, алгоритм Хаффмана всегда дает оптимальный код.

Алгоритм Хаффмана:

1. Символы алфавита образуют список узлов, каждый из которых имеет свой вес (вероятность).
2. Выбираются два узла дерева с наименьшими весами, для которых создается их родитель с весом, равным их суммарному весу (или можно сказать, что два этих узла объединяются в один).
3. Одной дуге, выходящей из родителя, ставится в соответствие бит 0, другой — бит 1 (этот символ просто дописывается к коду).
4. Шаги, начиная со второго, повторяются до тех пор, пока в списке свободных узлов не останется только один свободный узел. Он и будет считаться корнем дерева.

Пример: закодировать по Хаффману алфавит, имеющие следующие частоты в сообщении:

символ	a	e	b	d	c	f	g
вероятность	7	5	2	3	3	1	1

Решение:

Расположим символы в порядке убывания вероятностей:

символ	частота	Шаг 1	Шаг 2	Шаг 3	Шаг 4	Шаг 5
a	7	7	7	7	7	13
e	5	5	5	5	6	9
d	3	3	3	6	9	
c	3	3	3	4		
b	2	2	4			
f	1	2				
g	1					

А теперь развернем код в соответствии с шагом 3 алгоритма (мы запишем это в отдельной таблице, просто чтобы было нагляднее, а обычно это делают в исходной таблице сразу):

символ	частота	Шаг 1	Шаг 2	Шаг 3	Шаг 4	Шаг 5
a	00	00	00	00	00	0
e	10	10	10	10	01	1
d	010	010	010	01	1	
c	011	011	011	11		
b	110	110	11			
f	1110	111				
g	1111					

Можете проверить, что код префиксный. Стоимость кода ищется также, как и для кода Фано в предыдущем примере.

Вопросы для самоконтроля:

1. Две основные задачи теории кодирования.
2. Коды. Алфавитное кодирование.
3. Разделимые коды. Критерий однозначности декодирования.
4. Неравенство Макмиллана.
5. Условие существования разделимого кода с заданными длинами кодовых слов.
6. Оптимальные коды и их свойства.
7. Лемма о редукции.
8. Алгоритмы Фано и Хаффмана построения оптимальных кодов.