ITMO more than a UNIVERSITY

Современная теория информации

Лекция 3. Энтропия на сообщение дискретного источника. Префиксные коды.

Беляев Евгений Александрович
eabelyaev@itmo.ru

Содержание лекции

ИТМО

1. Энтропия на сообщение дискретного источника.
2. Префиксные коды.
3. Неравенство Крафта.
4. Прямая и обратная теоремы побуквенного кодирования.
5. Код Хаффмана.

Рассмотрим $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ из $X_1X_2 \ldots X_n = X^n$.
Энтропия $H(X_1X_2 \ldots X_n) = H(X^n)$ называется n-**мерной энтропией** процесса.

Энтропия на символ для последовательности длины n определяется как:

$$H_n(X) = \frac{H(X^n)}{n}.$$

Другой способ:

$$H(X_n|X_1, \ldots, X_{n-1}) = H(X|X^{n-1}).$$

Энтропия на сообщение:

$$\lim_{n\to\infty} H_n(X) \text{ и } \lim_{n\to\infty} H(X|X^n).$$

Теорема. Для дискретного стационарного процесса (источника):

A. $H(X|X^n)$ не возрастает с увеличением n;

B. $H_n(X)$ не возрастает с увеличением n;

C. $H_n(X) \geq H(X|X^{n-1})$;

D. $\lim\limits_{n\to\infty} H_n(X) = \lim\limits_{n\to\infty} H(X|X^n)$.

Доказательство.

A. Следует из невозрастания энтропии с увеличением числа условий.

C. $H(X^n) = H(X) + H(X|X^1) + ... + H(X|X^{n-1}) \geq$

$$\geq \quad nH(X|X^{n-1}) \geq nH(X|X^n)$$

B. $H(X^{n+1}) \overset{(a)}{=} H(X_1...X_nX_{n+1})$

$$\begin{aligned} &\overset{(b)}{=} \quad H(X_1...X_n) + H(X_{n+1}|X_1, ..., X_n) \\ &\overset{(c)}{\leq} \quad nH_n(X) + H_n(X) \\ &\overset{(d)}{=} \quad (n+1)H_n(X). \end{aligned}$$

$$H(X^{n+1}) \leq (n+1)H_n(X).$$

$$\Longrightarrow \frac{H(X^{n+1})}{n+1} = H_{n+1}(X) \leq H_n(X)$$

Доказательство.

Д1. $H_n(X)$ и $H(X|X^n)$ ограничены снизу (≥ 0) и не возрастают, т.е., существуют пределы $\lim\limits_{n\to\infty} H_n(X)$ и $\lim\limits_{n\to\infty} H(X|X^n)$.
Из С следует, что: $\lim\limits_{n\to\infty} H_n(X) \geq \lim\limits_{n\to\infty} H(X|X^n)$.

Д2. Для $m < n$:

$$
\begin{aligned}
&H(X^n) = H(X_1 \dots X_n) = \\
&\overset{(a)}{=} H(X_1 \dots X_m) + H(X_{m+1}|X_1, \dots, X_m) + \dots + H(X_n|X_1, \dots, X_{n-1}) \\
&\overset{(b)}{\leq} {}^{1} mH_m(X) + (n-m)H(X|X^m).
\end{aligned}
$$

После деления на n: $\lim\limits_{n\to\infty} H_n(X) \leq H(X|X^m)$, для любого m.
Устремляем $m \to \infty$: $\lim\limits_{n\to\infty} H_n(X) \leq \lim\limits_{m\to\infty} H(X|X^m)$.

[1] В правой части учитываем m предыдущих символов, вместо n

Обозначим $H_\infty(X) = \lim\limits_{n\to\infty} H_n(X)$, $H(X|X^\infty) = \lim\limits_{n\to\infty} H(X|X^n)$, тогда

$$H_\infty(X) = H(X|X^\infty)$$

Два способа кодирования:

- Расширение алфавита: буквы это последовательности исходных букв длины n.
- Учёт зависимости текущей буквы от n предшествующих букв.

- $H(X_1...X_n) = H(X_1) + ... + H(X_n)$.
- $H(X^n) = nH(X)$.
- $H_n(X) = H(X)$,
- $H_\infty(X) = H(X)$.
- $H(X|X^n) = H(X_{n+1}|X_1, ..., X_n) = H(X)$,
- $H(X|X^\infty) = H(X)$.

Отсюда не следует, что для такого источника нужно кодировать каждую букву независимо от других.

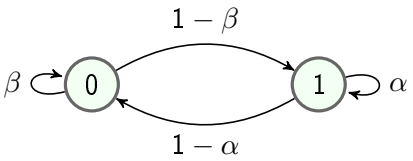
- $H(X|X^n) = H(X_{n+1}|X_1, ..., X_n) = H(X_{n+1}|X_{n-s+1}, ..., X_n) = H(X|X^s)$.
- $H(X|X^\infty) = H(X|X^s)$.
- $H(X^n) = H(X_1 \dots X_s X_{s+1} \dots X_n)$

$$= H(X_1 \dots X_s) + H(X_{s+1} \dots X_n|X_1, \dots, X_s).$$

- $H(X_{s+1} \dots X_n|X_1, \dots, X_s) = H(X_{s+1}|X_1, \dots, X_s) +$

$$\begin{aligned} &+ \quad H(X_{s+2}|X_2, \dots, X_{s+1}) + \dots \\ &+ \quad H(X_n|X_{n-s}, \dots, X_{n-1}) \\ &= \quad (n-s)H(X|X^s). \end{aligned}$$

- $\dfrac{H(X^n)}{n} = \dfrac{sH_s(X)}{n} + \dfrac{(n-s)H(X|X^s)}{n}$



$$\Pi = \begin{bmatrix} \beta & 1-\beta \\ 1-\alpha & \alpha \end{bmatrix}, \begin{cases} p_0 = \frac{(1-\alpha)}{2-\beta-\alpha}, \\ p_1 = \frac{(1-\beta)}{2-\beta-\alpha}. \end{cases}$$

- $H(X) = -p_0 \log p_0 - p_1 \log p_1 = \eta(p_0) = \eta(p_1)$.
- $H(X|X) =$

$$= -\sum_{x_1 \in X} \left(p(x_1) \sum_{x_2 \in X} p(x_2|x_1) \log p(x_2|x_1) \right) = -\sum_{x_1=0}^{1} \left(p(x_1) \sum_{x_2=0}^{1} p(x_2|x_1) \log p(x_2|x_1) \right) =$$
$$-p_0 p(0|0) \log p(0|0) - p_0 p(1|0) \log p(1|0) - p_1 p(0|1) \log p(0|1) - p_1 p(1|1) \log p(1|1) =$$
$$-p_0 \beta \log \beta - p_0 (1-\beta) \log(1-\beta) - p_1 \alpha \log \alpha - p_1 (1-\alpha) \log(1-\alpha) = p_0 \eta(\beta) + p_1 \eta(\alpha).$$

- $H_2(X) = \frac{H(X)+H(X|X)}{2}$, $H_3(X) = \frac{H(X)+2H(X|X)}{3}$, $H_n(X) = \frac{H(X)+(n-1)H(X|X)}{n}$.

- Информационная производительность дискретного источника без памяти определяется его энтропией $H(X)$.
- $H(X) \leq \log |X|$, равенство когда все символы равновероятны.
- Энтропия на символ дискретного стационарного источника определяется как $\lim\limits_{n\to\infty} H_n(X) = \lim\limits_{n\to\infty} H(X|X^n)$
- Наилучшее сжатие может быть достигнуто либо кодированием длинных блоков символов, либо с учётом длинной предыстории для каждого символа.

Рассмотрим дискретный источник без памяти.

- $X = \{1, ..., M\}$, $\{p_1, ..., p_M\}$. $C = \{\boldsymbol{c}_1, ..., \boldsymbol{c}_M\}$, кодовые слова длины $l_1, \ldots, l_M$.
- Средняя длина кодового слова:

$$\bar{l} = \mathrm{E}[l_i] = \sum_{i=1}^{M} p_i l_i$$

$H(X)$ – нижняя граница для $\bar{l}$.

Буква	Кодовое слово
e	$\cdot$
a	$\cdot-$
j	$\cdot---$
q	$--\cdot-$

Декодировать: $\cdot - - - \cdot -$

Буква	Кодовое слово
e	$\cdot$
a	$\cdot -$
j	$\cdot - - -$
q	$- - \cdot -$

Декодировать: $\cdot - - - \cdot - \to$ *aq* или *ja* ?

Для однозначного декодирования кода необходимы разделители (“паузы”) между кодовыми словами.

Пример: $X = \{a, b, c, d\}$

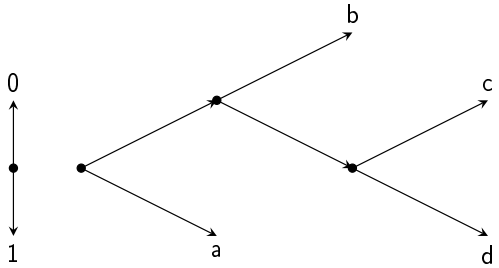


Рис.: Пример двоичного кодового дерева

Буква	Кодовое слово
a	1
b	00
c	010
d	011

Декодировать: 0101001010 ...

Свойства:

- Код называется ***префиксным***, если ни одно кодовое слово не является началом другого кодового слова.
- Префиксный код является однозначно декодируемым.
- Если только листья двоичного дерева соответствуют кодовым словам, то код является префиксным.
- Однозначно декодируемый код не обязательно является префиксным.
- Древовидный код является префиксным.

$X = \{0, 1, 2, 3\}$

Какой код является

- префиксным?
- однозначно декодируемым?

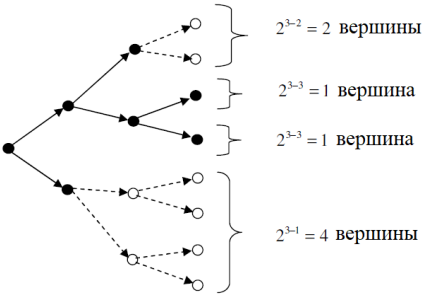
1. $C_1 = \{00, 01, 10, 11\}$;
2. $C_2 = \{1, 01, 001, 000\}$;
3. $C_3 = \{1, 10, 100, 000\}$;
4. $C_4 = \{0, 1, 10, 01\}$;
5. $C_5 = \{0, 1, 12, 31\}$;

Теорема. Необходимым и достаточным условием существования префиксного кода объёмом M с длинами кодовых слов $l_1, ..., l_M$ является выполнение неравенства:

$$\sum_{i=1}^{M} 2^{-l_i} \leq 1.$$

Неравенство верно для любого префиксного кода.

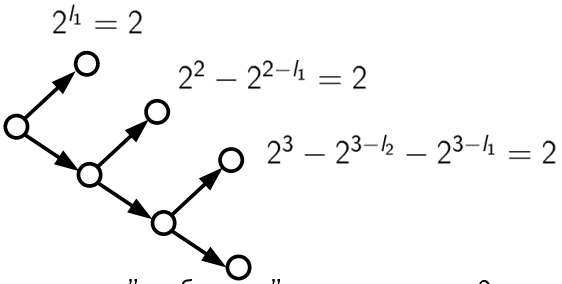
Выберем L, такое что $L \geq \max\limits_i l_i$. Концевая вершина исходного дерева, расположенная на глубине l_i, имеет 2^{L-l_i} потомков на глубине L. Причем, множества концевых потомков не пересекается.



$$\sum_{i=1}^{M} 2^{L-l_i} \leq 2^L.$$

Если неравенство Крафта выполняется, то существует код с заданным набором длин кодовых слов.

Пример: $l_1 = 1,\ l_2 = 2,\ l_3 = l_4 = 3$.



На каждом шаге количество "свободных" вершин равно 2, т.е., можно поместить следующее кодовое слово.

- Сортируем $\{l_i\}$ по убыванию.
- Из всего числа вершин на ярусе l_1 выберем любую и закрепим её за первым кодовым словом. Продолжим оставшиеся вершины до яруса l_2. На этом ярусе будет свободно $2^{l_2} - 2^{l_2 - l_1}$ вершин. Умножив $\sum_{i=1}^{M} 2^{-l_i} \leq 1$ для $M = 2$ на 2^{l_2}, получим $2^{l_2} - 2^{l_2 - l_1} \geq 1$, то есть как минимум одна свободная вершина есть.
- $2^{l_3} - 2^{l_3 - l_2} - 2^{l_3 - l_1} \geq 1$.
- ...
- $2^{l_M} - 2^{l_M - l_{M-1}} - 2^{l_M - l_{M-2}} - \ldots - 2^{l_M - l_1} \geq 1$

Теорема. Для любого однозначно декодируемого двоичного кода объёмом M с длинами кодовых слов $l_1, \ldots, l_M$ справедливо неравенство:

$$\sum_{i=1}^{M} 2^{-l_i} \leq 1.$$

Теорема. Для ансамбля $X = \{x, p(x)\}$ с энтропией $H(X) = H$ существует побуквенный неравномерный префиксный код со средней длиной кодовых слов $\bar{l} < H + 1$.

Доказательство.

1. Пусть $l_i = \lceil -\log p_i \rceil$. Тогда $\sum\limits_{i=1}^{M} 2^{-l_i} = \sum\limits_{i=1}^{M} 2^{-\lceil -\log p_i \rceil} \leq \sum\limits_{i=1}^{M} 2^{\log p_i} = 1 \Rightarrow$ такой префиксный код существует.
2. $\bar{l} = \sum\limits_{m=1}^{M} p_m l_m < \sum\limits_{m=1}^{M} p_m(-\log p_m + 1) < H + 1.$

Теорема. Для любого однозначно декодируемого кода для дискретного источника $\{X, p(x)\}$ с энтропией H, $\bar{l} \geq H$.

Доказательство.

$$\begin{aligned} H - \bar{l} &= -\sum_{x \in X} p(x) \log p(x) - \sum_{x \in X} p(x) l(x) \\ &= \sum_{x \in X} p(x) \log \frac{2^{-l(x)}}{p(x)} \leq \log e \sum_{x \in X} p(x) \left(\frac{2^{-l(x)}}{p(x)} - 1 \right) \\ &\leq \log e \left(1 - \sum_{x \in X} p(x) \right) = 0 \end{aligned}$$

1. Вероятности символов сортируются по убыванию вероятностей и затем разбиваются на два множества с почти равной вероятностью.
2. Каждое подмножество кодируется 0 или 1 в двоичном дереве.
3. Каждое из полученных множеств разбивается еще на два аналогичным образом.
4. Этот процесс продолжается до тех пор, пока каждое подмножество не будет содержать только один символ[2].

x	$p(x)$	$c(x)$
a	0.35	00
b	0.20	01
c	0.15	100
d	0.1	101
e	0.1	110
f	0.1	111

[2] R.M. Fano, Technical Report No. 65, The Research Laboratory of Electronics, M.I.T., 1949.

Свойства оптимального кода:

1. Если $p_i < p_j$, то $l_i \geq l_j$.
2. Не менее двух кодовых слов имеют одинаковую длину $l_M = \max\limits_m l_m$. Если у нас имеется только одно кодовое слово максимальной длины, то код не оптимален, так как мы можем убрать последний символ такого кодового слова.
3. Среди кодовых слов длиной $l_M = \max\limits_m l_m$ найдутся два слова, различающиеся только в одном последнем символе.

Этими свойствами обладает код Хаффмана[3].

[3] D.Huffman, A Method for the Construction of Minimum-Redundancy Codes, *Proceedings of the IRE*, 1952

Свойства оптимального кода:

4. Пусть $p_1 \geq p_2 \geq ... \geq p_M$.
   - Для ансамбля $X = \{1, ..., M\}$ и кода C, удовлетворяющего свойствам 1–3, введем ансамбль $X' = \{1, ..., M-1\}$, сообщениям которого приписаны вероятности $\{p'_1, ..., p'_{M-1}\}$ так, что

$$p'_1 = p_1,$$
$$p'_2 = p_2,$$
$$p'_{M-1} = p_{M-1} + p_M.$$

   - Из кода C построим код C' для ансамбля X', приписав сообщениям $x'_1, ..., x'_{M-2}$ те же кодовые слова, что и в коде C, т.е. $c'_i = c_i$, а сообщению x'_{M-1} слово c'_{M-1}, как общую часть слов C_{M-1} и C_M.
   - Тогда, если C' оптимален для X', то код C оптимален для X.

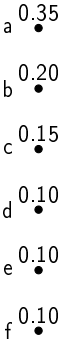
Из свойства 3 следует, что:

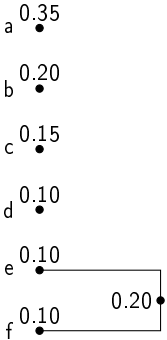
$$l_m = \begin{cases} l'_m \text{ для } m \leq M-2, \\ l'_{M-1} + 1 \text{ для } m = M-1, M. \end{cases}$$

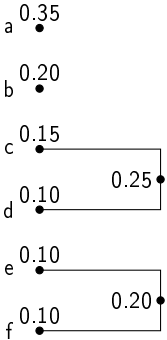
Тогда средняя длина кодового слова:

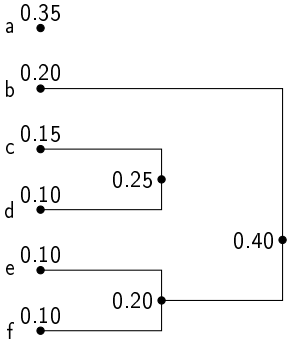
$$\begin{aligned} \overline{l} &= \sum_{m=1}^{M} p_m l_m = \sum_{m=1}^{M-2} p_m l_m + p_{M-1} l_{M-1} + p_M l_M = \\ &= \sum_{m=1}^{M-2} p_m l_m + (p_{M-1} + p_M)(l'_{M-1} + 1) = \\ &= \sum_{m=1}^{M-2} p'_m l'_m + p'_{M-1} l'_{M-1} + p_{M-1} + p_M = \\ &= \sum_{m=1}^{M-1} p'_m l'_m + p_{M-1} + p_M = \overline{l}' + p_{M-1} + p_M. \end{aligned}$$

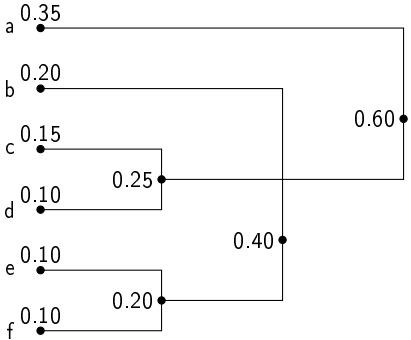
где $\overline{l}' = \sum_{m=1}^{M-1} p'_m l'_m$ – среднее длина кодового слово кода C'.

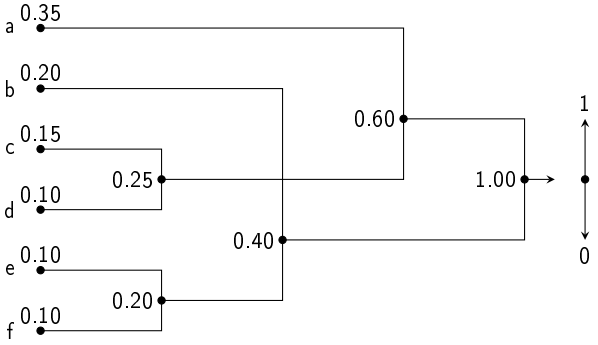

a 0.35
b 0.20
c 0.15
d 0.10
e 0.10
f 0.10


a 0.35
b 0.20
c 0.15
d 0.10
e 0.10
f 0.10
0.20

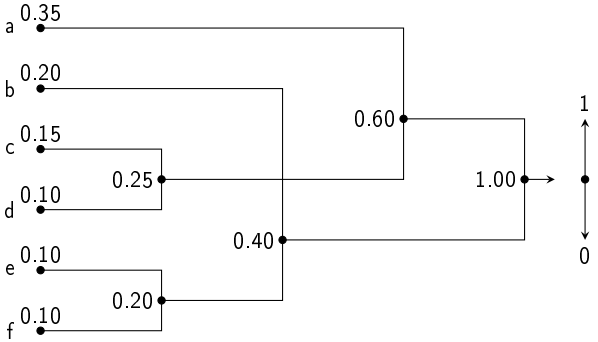

a 0.35
b 0.20
c 0.15
0.25
d 0.10
e 0.10
0.20
f 0.10


a
0.35
b
0.20
c
0.15
0.25
d
0.10
0.40
e
0.10
0.20
f
0.10


a
0.35
b
0.20
c
0.15
d
0.10
e
0.10
f
0.10
0.25
0.20
0.40
0.60

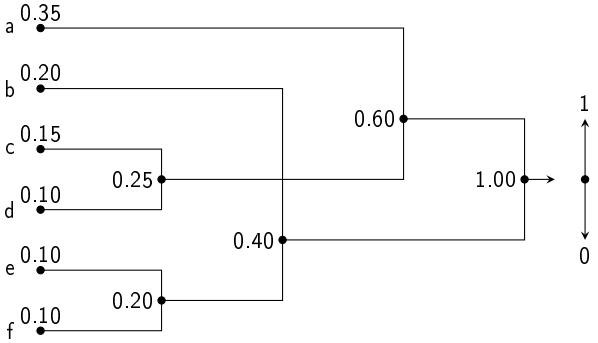


a	11
b	01
c	101
d	100
e	001
f	000



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

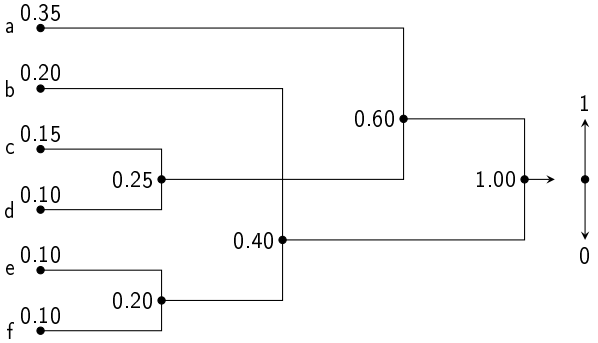
$$H = -\sum_{x} p(x) \log p(x) = 2.4016$$



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

$$H = -\sum_x p(x) \log p(x) = 2.4016$$

$$\bar{l} = \sum_x l(x) p(x) = 2.4500$$



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

$$H = -\sum_x p(x) \log p(x) = 2.4016$$

$$\bar{l} = \sum_x l(x) p(x) = 2.4500$$

- P – матрица, в которой хранятся вероятности появления сообщений.
- L – матрица длин кодовых слов.
- C – матрица кодовых слов размером $M \times M$.
- T – матрица потомков узлов $M \times M$.

$$P = \begin{pmatrix} 0.5 \\ 0.25 \\ 0.125 \\ 0.125 \end{pmatrix}, C = \begin{pmatrix} - & - & - & - \\ - & - & - & - \\ - & - & - & - \\ - & - & - & - \end{pmatrix}, T = \begin{pmatrix} 0 & - & - & - \\ 1 & - & - & - \\ 2 & - & - & - \\ 3 & - & - & - \end{pmatrix}, L = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

1. Два сообщения x_2 и x_3 с минимальными вероятностями объединяются. Вероятность нового сообщения записывается во вторую строку P.
2. $C[2, L[2]] \leftarrow 0,\ C[3, L[3]] \leftarrow 1.$
3. Во вторую строку матрицы T записываются номера потомков объединенного узла, то есть все номера узлов, которые находились в строках 2 и 3.
4. $L[2] \leftarrow L[2] + 1,\ L[3] \leftarrow L[3] + 1.$

$$P = \begin{pmatrix} 0.5 \\ 0.25 \\ 0.25 \\ - \end{pmatrix}, C = \begin{pmatrix} - & - & - & - \\ - & - & - & - \\ 0 & - & - & - \\ 1 & - & - & - \end{pmatrix}, T = \begin{pmatrix} 0 & - & - & - \\ 1 & - & - & - \\ 2 & 3 & - & - \\ - & - & - & - \end{pmatrix}, L = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}.$$

- Объединяются x_1 и x_{23}.
- $C[1, L[1]] \leftarrow 0,\ C[2, L[2]] \leftarrow 1,\ C[3, L[3]] \leftarrow 1.$
- В $T[1][..]$ записываются номера потомков, которые находились в первой и второй строках.
- $L[T[1][..]] \leftarrow L[T[1][..]] + 1.$

$$P = \begin{pmatrix} 0.5 \\ 0.5 \\ - \\ - \end{pmatrix}, C = \begin{pmatrix} - & - & - & - \\ 0 & - & - & - \\ 0 & 1 & - & - \\ 1 & 1 & - & - \end{pmatrix}, T = \begin{pmatrix} 0 & - & - & - \\ 1 & 2 & 3 & - \\ - & - & - & - \\ - & - & - & - \end{pmatrix}, L = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 2 \end{pmatrix}.$$

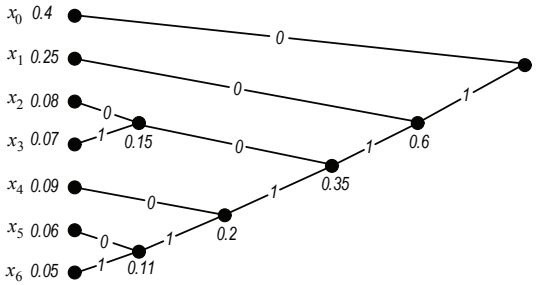
- Объединяются x_0 и x_{123}.
- $C[0, L[0]] \leftarrow 0$,
  $C[1, L[1]] \leftarrow 1,\ C[2, L[2]] \leftarrow 1,\ C[3, L[3]] \leftarrow 1$.
- В $T[0][..]$ записываются номера потомков, которые находились в нулевой и первой.
- $L[T[0][..]] \leftarrow L[T[0][..]] + 1$.

$$P = \begin{pmatrix} 1.0 \\ - \\ - \\ - \end{pmatrix}, C = \begin{pmatrix} 0 & - & - & - \\ 0 & 1 & - & - \\ 0 & 1 & 1 & - \\ 1 & 1 & 1 & - \end{pmatrix}, T = \begin{pmatrix} 0 & 1 & 2 & 3 \\ - & - & - & - \\ - & - & - & - \\ - & - & - & - \end{pmatrix}, L = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 3 \end{pmatrix}.$$

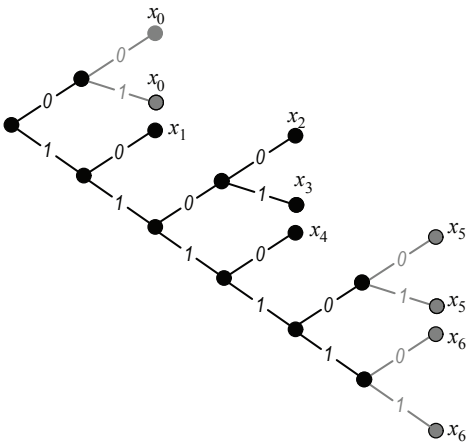
- Так как длина кодовых слов не кратна 8 битам, кодер должен использовать промежуточный буфер в 2-4 байта.
- На практике используется заранее подготовленная таблицу декодирования.

Таблица: Пример однобитной таблицы декодирования для $X = \{x_0, x_1, x_2, x_3\}$ и $C = \{0, 10, 110, 111\}$

Адрес, a	b_j	Адрес перехода, $A[a]$	$x[a]$
0	0	–	x_0
1	1	2	–
2	0	–	x_1
3	1	4	–
4	0	–	x_2
5	1	–	x_3



a	b_j	$A[a]$	$x[a]$
0	0	–	x_0
1	1	2	–
2	0	–	x_1
3	1	4	–
4	0	6	–
5	1	8	–
6	0	–	x_2
7	1	–	x_3
8	0	–	x_4
9	1	10	–
10	0	–	x_5
11	1	–	x_6



a	$b_j b_{j+1}$	$l[a]$	$A[a]$	$x[a]$
0	00	1	–	x_0
1	01	1	–	x_0
2	10	2	–	x_1
3	11	–	4	–
4	00	4	–	x_2
5	01	4	–	x_3
6	10	4	–	x_4
7	11	–	8	–
8	00	5	–	x_5
9	01	5	–	x_5
10	10	5	–	x_6
11	11	5	–	x_6

ITMO more than a UNIVERSITY

Спасибо за внимание!