Information Theory. 3rd Chapter Slides

Boris Kudryashov

ITMO University

December 18, 2016

Agenda

1. Universal coding task
2. Useful combinatorial formulas
3. Two pass encoding
4. Enumerative coding
5. Asymptotic bounds of redundancy
6. Adaptive coding
7. Algorithm comparison

Universal coding task

- *Encoding redundancy* for a model class Ω is

$$r_n(\Omega) = \sup_{\omega \in \Omega} \left[\bar{R}_n(\omega) - H_\omega \right]. \qquad (1)$$

- Coding is called *Universal* if for algorithm holds

$$\lim_{n \to \infty} r_n(\Omega) = 0,$$

- Consider sequences $\boldsymbol{x} = (x_1, \ldots, x_n)$, where x_i has one of M_i values, $i = 1, \ldots, n$. Number of different $\boldsymbol{x}$ is

$$|\{\boldsymbol{x} = (x_1, \ldots, x_n) : x_i \in \{0, \ldots, M_i - 1\}, i = 1, \ldots n\}| = \tag{2}$$

-

$$A_M^n = M(M-1) \times \ldots \times (M-n+1) = \frac{M!}{(M-n)!}. \tag{3}$$

- Number of combinations

$$C_M^n = \binom{M}{n} = \frac{A_M^n}{P_n} = \frac{M(M-1)\times \ldots \times (M-n+1)}{n!} = \frac{M!}{n!(M-n)!}. \quad (4)$$

- Number of combinations

$$\binom{n}{k} = \begin{cases} \frac{n!}{k!(n-k)!}, & \text{если } n \geq k \geq 0 \\ 1, & \text{если } n \geq 0 \text{ и } k = 0 \text{ или } k = n \\ 0, & \text{если } k < 0 \text{ или } k > n \end{cases} \quad (5)$$

Useful combinatorial formulas

- binomial coefficient

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k}.$$

- Number os binary sequences of length n, which contain τ_1 ones and $\tau_0 = n - \tau_1$ zeros.

$$N(\tau_0, \tau_1) = \binom{n}{\tau_0} = \frac{n}{\tau_0! \tau_1!}. \quad (6)$$

- Composition of sequence $\boldsymbol{x}$ is vector $\boldsymbol{\tau}(\boldsymbol{x}) = (\tau_0(\boldsymbol{x}), ..., \tau_{M-1}(\boldsymbol{x}))$, where $\tau_i(\boldsymbol{x})$ denotes number of elements $x_t = i$ in sequence $\boldsymbol{x} = (x_1, ..., x_n)$.

- For $M = 3$

$$N(\boldsymbol{\tau}) = \binom{n}{\tau_0}\binom{n-\tau_0}{\tau_1} = \frac{n!}{\tau_0!(n-\tau_0)!}\frac{(n-\tau_0)!}{\tau_1!(n-\tau_0-\tau_1)!} = \frac{}{\tau_0!}$$

- For arbitrary M

$$N(\boldsymbol{\tau}) = \frac{n!}{\tau_0!\ldots\tau_{M-1}!}. \qquad (7)$$

- Newton formula generalization

$$(a_0 + \ldots + a_{M-1})^n = \sum_{\boldsymbol{\tau}:\ \tau_0+\ldots+\tau_{M-1}=n} N(\boldsymbol{\tau}) \prod_{i=0}^{M-1} a_i^{\tau_i}.$$

- Consider the following lemma:

Lemma

$n \in$ mathbbN can be written as sum of M non-negative integer terms in $\binom{n+M-1}{M-1}$ ways.

- Number of different compositions of sequence of length n over M-size alphabet is

$$N_\tau(n, M) = \binom{n+M-1}{M-1} \tag{8}$$

- Stirling formula

$$\sqrt{2\pi n} n^n e^{-n} \exp\left\{\frac{1}{12n+1}\right\} < n! < \sqrt{2\pi n} n^n e^{-n} \exp\left\{\frac{1}{12n}\right\}. \tag{9}$$

Useful combinatorial formulas

- Consider

$$N(\boldsymbol{\tau}) < (2\pi n)^{-\frac{M-1}{2}} 2^{n \log n - \sum_i \tau_i \log \tau_i} \left(\prod_i \frac{n}{\tau_i} \right)^{1/2} \times$$

$$\times \exp \left\{ \frac{1}{12n} - \sum_i \frac{1}{12\tau_i + 1} \right\}. \qquad (10)$$

- Logarithm of number of sequences with specified composition

$$\log N(\boldsymbol{\tau}) < nH(\hat{\boldsymbol{p}}) - \frac{M-1}{2} \log(2\pi n) - \frac{1}{2} \sum_i \log(\hat{p}_i), \qquad (11)$$

- More compact estimation

$$\log N(\boldsymbol{\tau}) < nH(\hat{\boldsymbol{p}}) - \frac{M-1}{2}\log(2\pi n) + \frac{1}{2}\log\frac{n}{n-M+1}. \tag{12}$$

- Recurrent formula holds

$$\binom{n+1}{w} = \binom{n}{w} + \binom{n}{w-1}. \tag{13}$$

-

$$\binom{n+1}{w} = \binom{n}{w} + \binom{n-1}{w-1} + \ldots + \binom{n-w+1}{1}. \tag{14}$$

Two pass encoding

- $$\texttt{IF\_WE\_CANNOT\_DO\_AS\_WE\_WOULD\_WE\_SHOULD\_DO\_AS\_WE\_CAN} \quad (15)$$
- $$l_2 = 6 + 6 + 12 \times 2 + 5 \times 3 + ... + 6 = 178.$$
  - 00010000010100110111101101111.

- $$l_1 = 29 + 8 \times 15 = 149 \text{ bit.}$$
- $$l = l_1 + l_2 = 149 + 178 = 327 \text{ bit.} \qquad (16)$$

Table: Huffman code for text (15)

Character	Number of iterations	Codeword length	Codeword
I	1	6	010000
F	1	6	010001
_	12	2	00
W	5	3	100
E	4	4	0101
C	2	5	01001
A	4	4	1010
N	3	4	1011
O	5	3	110
T	1	6	011110
D	4	4	0110
S	3	4	1110
U	2	4	1111
L	2	5	01110
H	1	6	011111

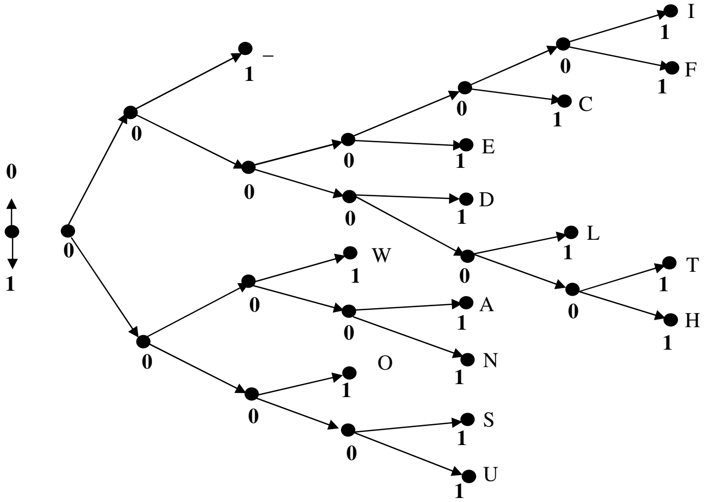

Figure: Huffman codetree for (15)

Two pass encoding

Table: Regular Huffman code

Character	Codeword length	Codeword
_	2	00
O	3	010
W	3	011
A	4	1000
D	4	1001
E	4	1010
N	4	1011
S	4	1100
U	4	1101
C	5	11100
L	5	11101
F	6	111100
H	6	111101
I	6	111110
T	6	111111

Table: Number of bits for regular code tree transmitting

Level	Number of nodes	Number of leaves n_i	Range of values n_i	Expenses in bits
0	1	0	$0 \dots 1$	1
1	2	0	$0 \dots 2$	2
2	4	1	$0 \dots 4$	3
3	6	2	$0 \dots 6$	3
4	8	6	$0 \dots 8$	4
5	4	2	$0 \dots 4$	3
6	4	4	$0 \dots 4$	3
Всего				19

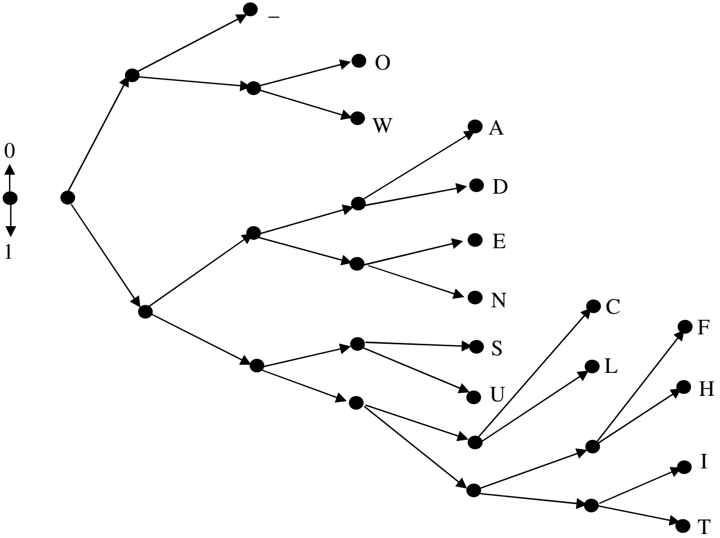

Figure: Codetree for regular Huffman code

- Enough for transmitting information about letters that are associated with regular codetree nodes:

$$\left\lceil \log \binom{256}{1} \right\rceil + \left\lceil \log \binom{255}{2} \right\rceil + \left\lceil \log \binom{253}{6} \right\rceil +$$

$$+ \left\lceil \log \binom{247}{2} \right\rceil + \left\lceil \log \binom{245}{4} \right\rceil = 105 \text{ бит}$$

- More precise

$$I = 178 + 19 + 105 = 302 \text{ бит.} \qquad (17)$$

Theorem

For two pass coding with Huffman code of Discrete Memoryless Source with alphabet size M and entropy H, average code rate satisfies

$$\bar{R} \leq H + 1 + \frac{1}{n}\left(M \log M + 3M - 1\right). \qquad (18)$$

Proof.

- $l_1(\boldsymbol{x}) \leq 2M - 1 + M \lceil \log M \rceil \leq M \log M + 3M - 1.$
-

$$\begin{aligned}
l_2(\boldsymbol{x}) &\overset{\text{(a)}}{=} \sum_{i=1}^{n} l(x_i) = \\
&\overset{\text{(b)}}{=} \sum_{x \in X} \tau_n(x) l(x) = \\
&\overset{\text{(c)}}{=} n \sum_{x \in X} \frac{\tau_n(x)}{n} l(x) = \\
&\overset{\text{(d)}}{=} n \sum_{x \in X} \hat{p}_n(x) l(x) = \\
&\overset{\text{(e)}}{=} n \mathbf{M}_{\hat{\boldsymbol{p}}_n} \left[l(x) \right] \leq \\
&\overset{\text{(f)}}{\leq} n(H(\hat{\boldsymbol{p}}_n) + 1).
\end{aligned} \tag{19}$$

Proof.

- $$\bar{R}(\boldsymbol{x}) = \frac{l(\boldsymbol{x})}{n} = \frac{l_1(\boldsymbol{x}) + l_2(\boldsymbol{x})}{n} \le \tag{20}$$
  $$\le H(\hat{\boldsymbol{p}}_n) + 1 + \frac{1}{n}\left(M \log M + 3M - 1\right). \tag{21}$$
- $$\mathbf{M}\left[H(\hat{\boldsymbol{p}}_n)\right] \overset{(a)}{\le} H\left(\mathbf{M}\left[\hat{\boldsymbol{p}}_n\right]\right) \overset{(b)}{=} H(\boldsymbol{p}) = H. \tag{22}$$
- $$\mathbf{M}\left[\hat{\boldsymbol{p}}_n\right] = \boldsymbol{p}, \tag{23}$$
- $$\mathbf{M}\left[\frac{\tau_n(a)}{n}\right] = p(a),\ a \in X.$$

Proof.

- $$\chi_a(x) = \begin{cases} 1, & \text{при } x = a, \\ 0, & \text{при } x \neq a. \end{cases}$$

- $$\mathbf{M}\left[\chi_a(x)\right] = 1 \times p(a) + 0 \times (1 - p(a)) = p(a).$$

- $$\begin{aligned} \mathbf{M}\left[\frac{\tau_n(a)}{n}\right] &= \frac{1}{n}\mathbf{M}\left[\sum_{i=1}^{n} \chi_a(x_i)\right] = \\ &= \frac{1}{n}\sum_{i=1}^{n} \mathbf{M}\left[\chi_a(x_i)\right] = \\ &= p(a),\ \ a \in X. \end{aligned}$$

- Note, that coding redundancy satisfies

$$r = \bar{R} - H \leq 1 + \frac{K}{n}, \tag{24}$$

- When using arithmetic coding, the redundancy can be achieved:

$$r(n) = \frac{M-1}{n} \log n + \frac{K}{n}, \tag{25}$$

where M alphabet size, K is a constant.

Two pass encoding

Table: Universal coding algorithm comparison

Algorithm	Number of traverses	Asymptotic redundancy	codeword length for text (15)
2-traverse coding, Huffman code	2	$1 + K_1/n$	302
Enumerative coding	2	$\frac{M \log n + K_3}{2n}$	283
Adaptive coding (A)	1	$\frac{M \log n + K_4}{2n}$	291
Adaptive coding (D)	1	$\frac{M \log n + K_5}{2n}$	283