Information data coding

Objectives

- Transcription of information to facilitate coding

  code $\Rightarrow$ signal *(Transcoding)*

- Information compression

  reducing information size

- Protection against transmission errors

  against loss and decision errors

- Keeping transmitted information secret

  encryption

Source of the messages

Alphabet

Definition of a code

application of S in $\mathcal{A} = \{ a_1, a_2, \ldots\ldots, a_q \}$

message $m_i \in S$ $\Rightarrow$ code-word $M_i \in \mathcal{M}$ finite sequences of $\mathcal{A}$

Information coding

- *Example of a simple source **S** that delivers 4 messages:*

S:
- m_1, probability: $Pr(m_1) = 0{,}5$
- m_2, probability : $Pr(m_2) = 0{,}25$
- m_3, probability : $Pr(m_3) = 0{,}125$
- m_4, probability : $Pr(m_4) = 0{,}125$

- ***Variable length** and **fixed length** Codes:*

Codes \ Messages	m_1	m_2	m_3	m_4
Fixed length code	**0 0**	**0 1**	**1 0**	**1 1**
Variable length code	~~1~~ 0	**1 0**	**1 1 0**	**1 1 1**

The ***variable length coding*** allows you to represent the messages more ***efficiently*** because it is based on the ***statistical properties*** of the source in opposition to the ***fixed length coding***

➢ Mean length of the ***fixed length code***:

$L_1 = 2$ bits

➢ Mean length of the ***variable length code***:

$L_2 = 1\times0{,}5 + 2\times0{,}25 + 3\times0{,}125 + 3\times0{,}125 = 1{,}75$

The most probable messages e.g. m_1 are encoded with a low number of bits

Concept of separator characters

- *Difference between an* ***irreducible*** *variable length code and a* ***reducible*** *variable length code:*

Messages / Codes	m_1	m_2	m_3	m_4
Reducible code	**1**	**1 0**	**1 0 0**	**1 0 0 0**
Irreducible code	~~1~~ 0	**1 0**	**1 1 0**	**1 1 1**

➢ The ***reducible*** codes can not be decoded without ambiguity because the code-words of the symbols s_i are the prefixes of the code-words of the symbols s_j once $i < j$: it is necessary to add an ***explicit separator***

- ***Example from the « code » of the English language*:**

The word « ***seven*** » is the prefix of the word « ***seventeen*** », thus to know if we must read « ***seventeen*** » or « ***seven teen*** », we must add an ***explicit separator*** « ***space*** ».

➢ The ***irreducible*** code can be decoded ***without ambiguity*** because the code-words of the symbols s_i cannot be the prefixes of the code-words s_j. In that case, the ***separators*** are ***implicit*** (e.g: Huffman coding)

Information data coding

- Alphabet $A = \{ a_1, a_2, \ldots\ldots, a_q \}$
- Finite set of messages $S = \{ m_1 , m_2 , \ldots., m_i ,\ldots\ldots\ldots\ldots , m_N \}$

Coding $\downarrow$

$C = \{ M_1 , M_2 , \ldots., M_i ,\ldots\ldots\ldots\ldots , M_N \}$

- Length of code-words: $n_i = n (M_i)$
- Average length of code-words: $E (n) = \sum_{i=1;N} p_i n_i$
- Entropy of the source H: $H(p_1, \ldots , p_N) \leq \log_2 N$
- Average quantity of information per character = $H / E(n)$

  or $H / E(n) \leq \log_2 q \Rightarrow E(n) \geq H / \log_2 q$
- Flow of a source of information coded with an average D characters per second: $R = D H/E(n)$

$\Rightarrow$ $\mathbf{R \leq D \log_2 q}$ *R in bits/second*

Coding and decoding information

- ***Efficiency*** η of a code: $\eta = n_{min} / E(n)$ => $\eta = H / (E(n) \log_2 q)$
- ***Redundancy*** ρ of a code : $\rho = 1 - \eta$
- *Simple examples: codes C_1 and C_2*
- **Constraints**: separation of code-words & unambiguous reading of code-words => *regular and inverting codes*
- ***Regular code***: if $m_i \neq m_j$ ==> $M_i \neq M_j$ *(injective application)*
- ***Inverting codes*** : 2 sequences of distinct messages ==> 2 sequences of distinct codes

if $(m_{\alpha 1},\ldots, m_{\alpha i}) \neq (m_{\beta 1},\ldots, m_{\beta j})$ => $(M_{\alpha 1},\ldots, M_{\alpha i}) \neq (M_{\beta 1},\ldots, M_{\beta j})$

examples: fixed length codes; codes with separator

- ***Irreducible code***: inverting code that can be decoded without any device M_i is not a prefix of M_j $\forall$ i , j

Code examples

Regular codes / Inverting codes / Irreducible codes

Messages Proba.	m_1 0.5	m_2 0.25	m_3 0.125	m_4 0.125
C_1	1	1	0	00
C_2	0	1	11	01
C_3	1	01	001	000
C_4	1	10	100	1000

- C_1 is a non regular code
- C_2 is a non-inverting code
- C_3 is an inverting and irreducible code
- C_4 is only an inverting code

Information data coding

Information data coding ≡ coded representation of information

$$\text{Message} \xrightarrow{\textit{Injective correspondence}} \{b_n\}$$

Multiples roles of coding

- Preparing the transformation message => transmitted signal
- Adapting the source bit rate - channel capacity (compression)
- Protective encoding against transmission errors (error detection / correction)
- Encrypting (secretive communications)
- Tattooing (ownership markers)
- Transcoding (alphabet changes, transmission constraints)

Information data coding

➢ *Definitions*

- Message sources S: production of a sequence of messages, each of them being selected in a set $\mathcal{M}$ of messages
  ($\mathcal{M}$: codebook of possible messages M = { m_1 , m_2 , ….}, the m_i are also called "words")

- Message: finite sequence of symbols
  (characters taken from $\mathcal{A}$: alphabet)

- Alphabet: finite set of symbols $\mathcal{A}$ = { a_1, a_2, ……, a_k }

Entropy of a source (SHANNON 1948)

- Definition of ***uncertainty*** and of ***entropy***
  - **Uncertainty** I of an event E:

    $I(E) = -\log_2 \Pr\{E\}$ *Units*: bit (BInary uniT if $\log_2$)

    nat (NAtural uniT if Log_e): 1 nat=1.443 bits

    if source simple s_n $=> I(s_n) = \sum_{i=1;n} I(m_{\alpha i})$
  - **Entropy** H of a discrete random variable X:

    $$H(X) = E_X[\, I(X)\,] = \sum_{i=1;n} p_i\, I(X_i) = -\sum_{i=1;n} p_i \log_2(p_i)$$
  - Properties of entropy
    - $H \geq 0$; H is continuous, symmetrical; $H(p_1, \dots, p_N) \leq \log_2 n$
    - if $(p_1, \dots, p_n)$ and $(q_1, \dots q_n)$ are 2 distributions of probabilities

      $==> \sum_{i=1;n} p_i \log_2(q_i / p_i) \leq 0$ car $\mathrm{Log}\, x < x - 1$

Optimal statistical coding

- ***Definitions***:

  - S: discrete and simple source of messages m_i with probability law $p = (p_1, \ldots\ldots\ldots\ldots, p_N)$ (homogeneous source)

  - Coding of an alphabet $A = \{ a_1, a_2, \ldots\ldots, a_q \}$

  - Entropy of the source H(S) and average length of code-words E(n)

- ***MacMillan's theorem***:

  - There exists at least one irreducible inverting code that matches:

$$H / \log 2\ q \leq E(n) < (H / \log 2\ q) + 1$$

$\Rightarrow$ Equality if p_i of the form: $p_i = q^{-ni}$ (if $q = 2$ => $n_i = -\log 2\ p_i$)

- ***Shannon's theorem*** (1st theorem on noiseless coding)

$$H / \log 2\ q \leq E(n) < (H / \log 2\ q) + \varepsilon$$

Optimal statistical coding

- ***Fano - Shannon coding***

- ***Arithmetic coding*** ***(block encoding, interval type encoding)***
  possibilities of on line adaptation

- ***Huffman coding***
  3 basic principles:
  - if $p_i < p_j$ => $n_i \geq n_j$
  - the 2 unlikeliest codes have the same length
  - the 2 unlikeliest codes (of max length) have the same prefix of length n_{max}-1