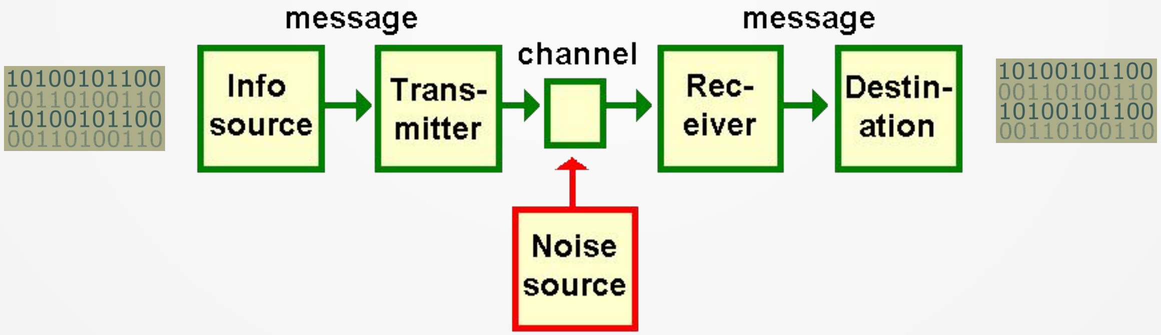
Lecture 1

Source Coding and Compression



Dr.-Ing. Khaled Shawky Hassan

Room: C3-222, ext: 1204,

Email: khaledshawkyhassan@guc.edu.eg

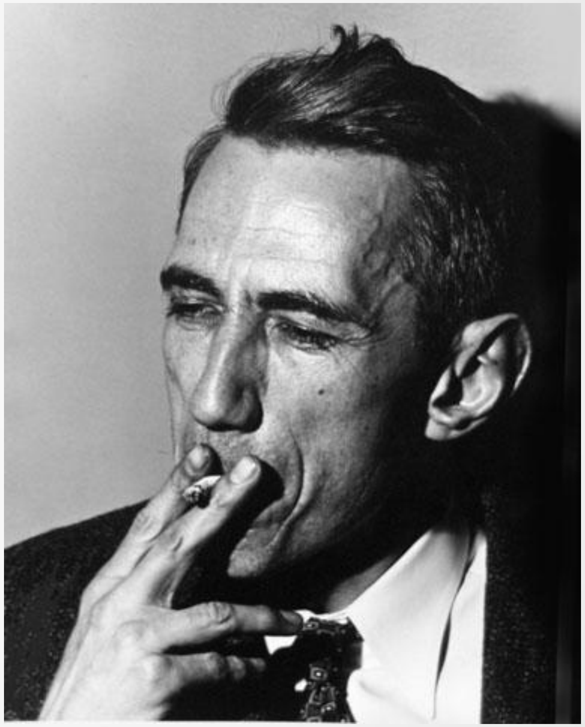
Information Theory

*Definition 1.1: The fundamental problem of communication is that of **reproducing** at one point either **exactly** or **approximately** a message selected at another point.*

(Claude Shannon, 1948)

Why Shannon used the word reproducing and not receiving?

What is meant by exactly and approximately



SmOkInG CaN KiLL YoU!

Information Theory

Definition 1.1: *The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.*

(Claude Shannon, 1948)

How can we achieve perfect communication over an imperfect, noisy communication channel?

Example:

an analogue telephone line, over which two modems communicate digital Information.

modem → line+noise → modem

"THEN, OUR PROBLEM IS THE NOISE!"

Shannon's Information Theory

The theory provides answers to two fundamental questions (among others):

- What is the irreducible complexity below which a signal cannot be compressed? (Shannon Theorem 1)

- How Can we correct errors or know their locations (Shannon Theorem 2)

- What is the ultimate transmission rate for reliable communication over a noisy channel? (Shannon Theorem 3)

- How can we protect them against hacking ? (Shannon Security Theorem)

Information Theory ?

- The information theory (IT) frame work:
  - Communication System
  - Line Coding
  - Channel Coding (Shannon's 2nd theorem)
  - Cryptology (Shannon's security theory)
  - Source Coding (Shannon's 1st theorem)

Information Theory ?

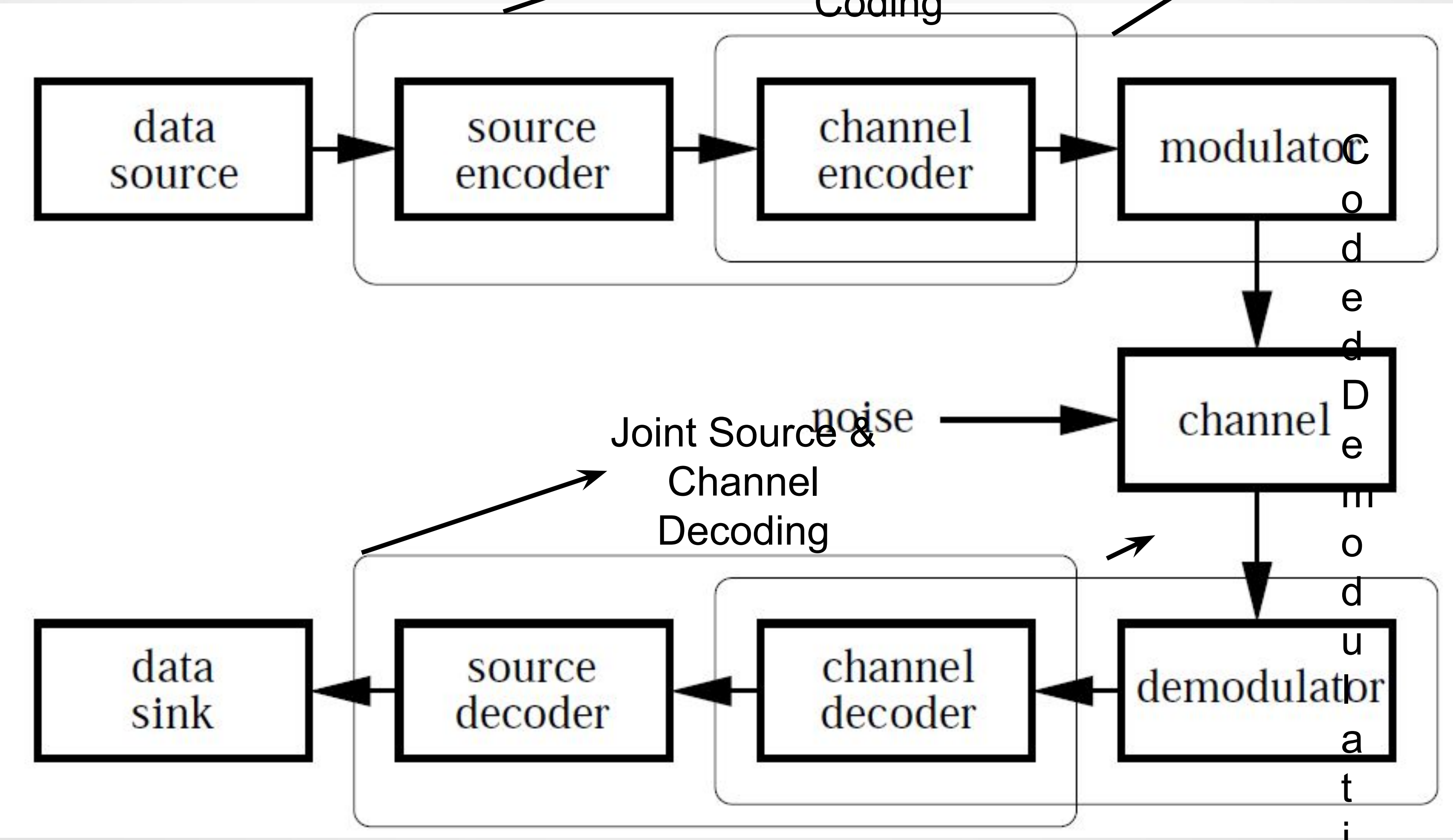
- The information theory (IT) frame work:
  - Communication System
  - Line Coding
  - Channel Coding (Shannon's 2nd theorem)
  - Cryptology
  - Source Coding (Shannon's 1st theorem)

Information Theory ?

- Communication System

Joint Source & Channel Coding

Coded Modulation

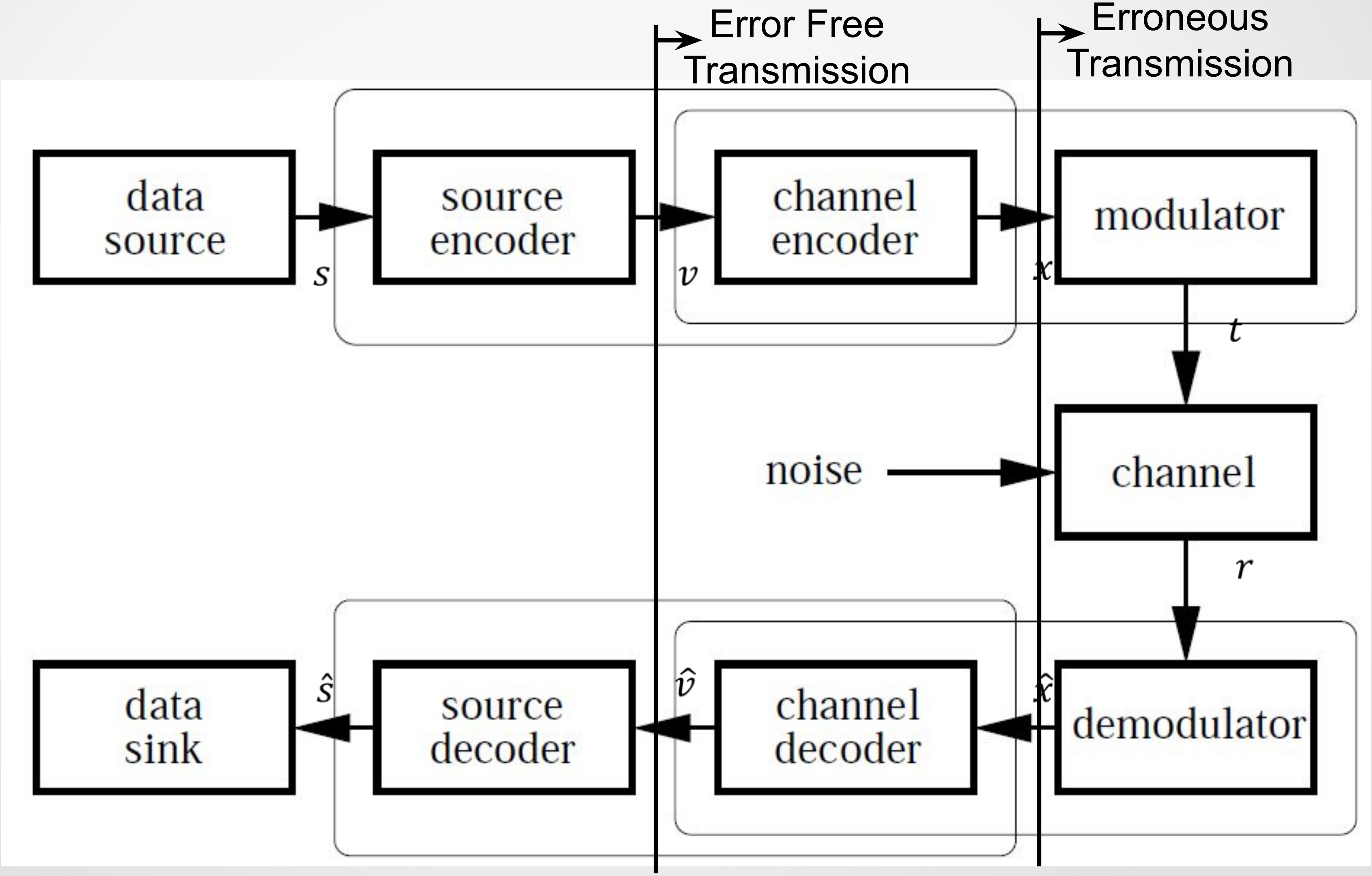


Joint Source & Channel Decoding

Coded Demodulatio

Information Theory ?

- Communication System

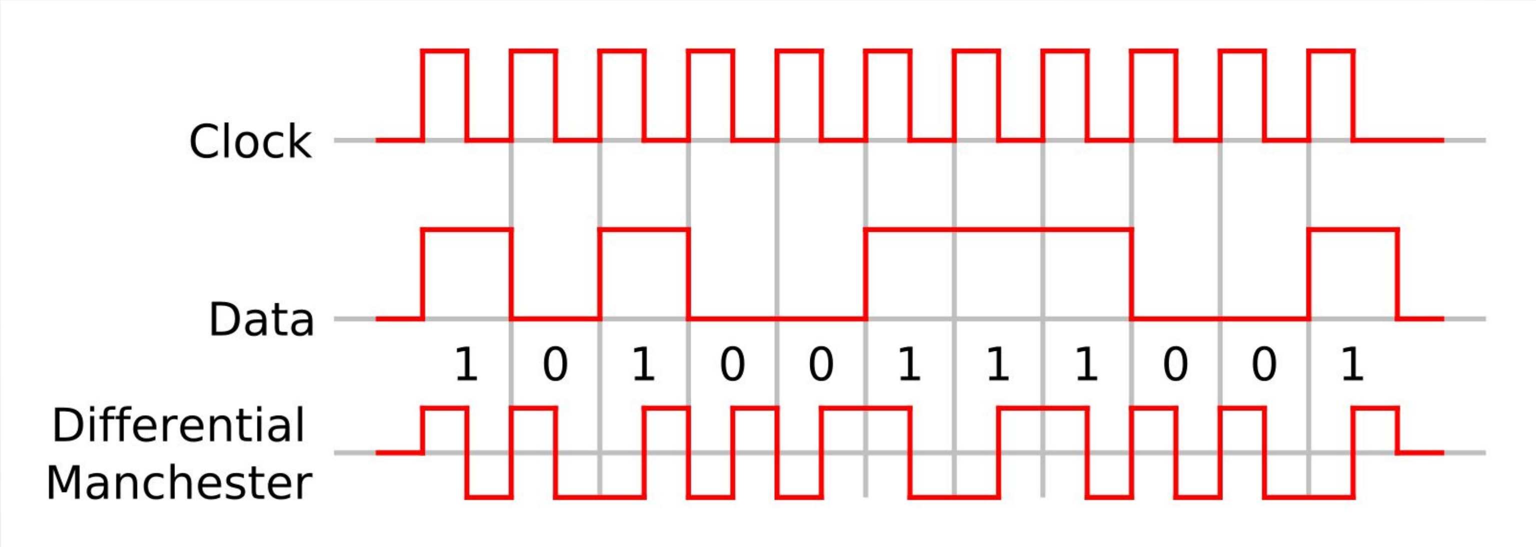


Information Theory ?

- The information theory (IT) frame work:
  - Communication System
  - Line Coding
  - Channel Coding (Shannon's 2nd theorem)
  - Cryptology
  - Source Coding (Shannon's 1st theorem)

Line Coding

- Also known as digital baseband modulation (1,0,1,1,0, …)
- encoding digital information to make it resistant to certain forms of signal loss during transmission



- Example: Differential Manchester !!! Not returning to zero! Why?
- a binary '1' is referred to as a "phase inversion"; a binary '0' is a "constant phase"

Information Theory ?

- The information theory (IT) frame work:
  - Communication System
  - Line Coding
  - Channel Coding (Shannon's 2nd theorem)
  - Cryptology
  - Source Coding (Shannon's 1st theorem)

The Three Channel Properties

I. Channels can only transport physical signals, e.g., electrical signals. Therefore, digital signals must be converted to appropriate formats (remember the line coding or RF)

II. EvEn iF the signal is adapted to the channel, it does not pass it undisturbed !! The channel introduce errors

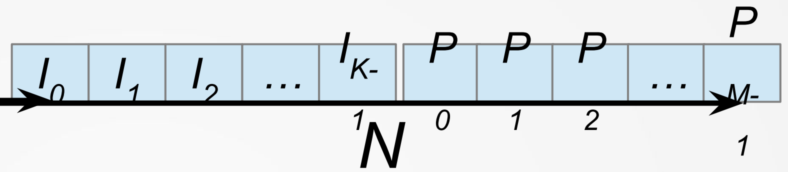
III. There is always an upper bound to the number of correct bits that you can send over the channel (Shannon Channel Capacity: *C*)

Channel Coding

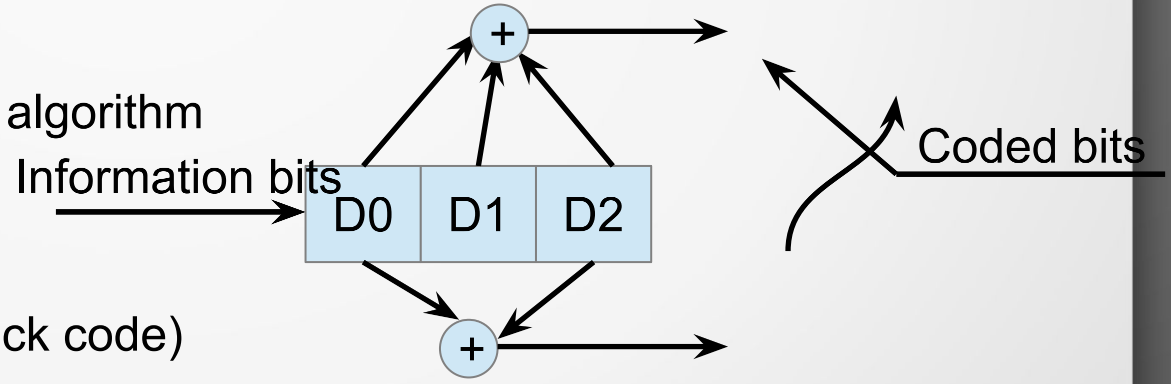
- A.K.A: Forward error correction (FEC)
- For controlling errors in data transmission over unreliable (generally → noisy) communication channels
- Tx Encode the message with redundancy
- The first error-correcting code in 1950 by Richard Hamming: Hamming (7,4) code
- The simplest code is the repetition code!! transmit each bit several times until and try to find the maximum likelihood
  - Example:
    - 0 0 0 → correct 0
    - 0 0 1 → maybe 0
    - 1 1 1 → correct 1
    - 1 0 1 → maybe 1
    - 1 0 0 → maybe 0

Types of Channel Coding

- Block codes:
  - A codeword with a length *N* consists of *K* information symbols (I_j) and M parity bits symbols
  - Code rate *R* = *K/N*



- Convolution Coding:
  - information bits are transformed to n bits (not injection; rather tapped delay model!!)
  - Can be decoded using Viterbi algorithm



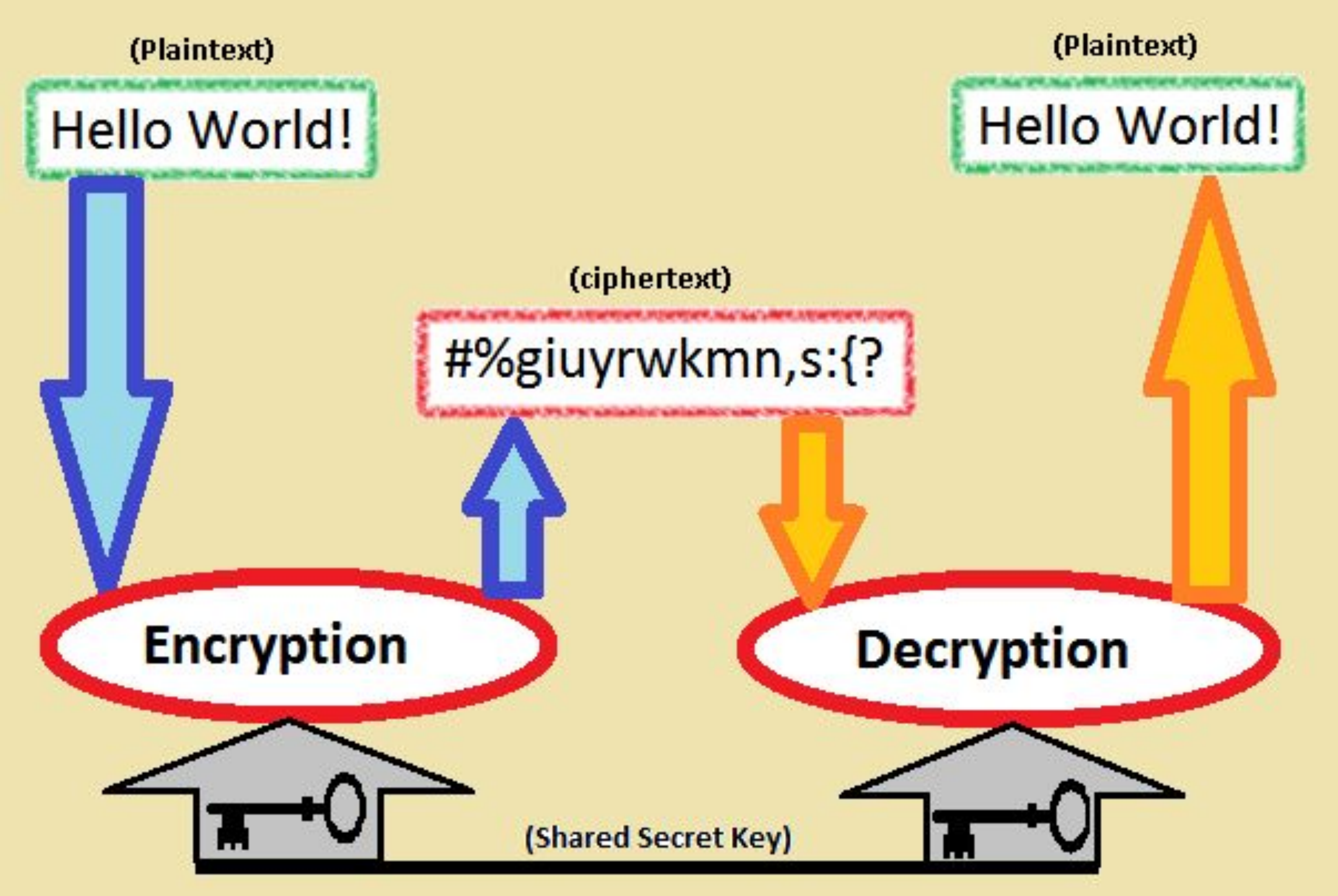
- Iterative decoding:
  - LDPC (Low-density parity-check code)
    - Exploit the low “1” counts into a graph (Tanner graph), then decode iteratively (needs very long codes)
  - Turbo Codes:
    - Combine two convolution codes, inner code and outer code

Information Theory ?

- The information theory (IT) frame work:
  - Communication System
  - Line Coding
  - Channel Coding (Shannon's 2nd theorem)
  - Cryptology
  - Source Coding (Shannon's 1st theorem)

Cryptology or the “Hidden/Secret Study”

- Cryptography mechanism



Cryptology or the “Hidden/Secret Study”

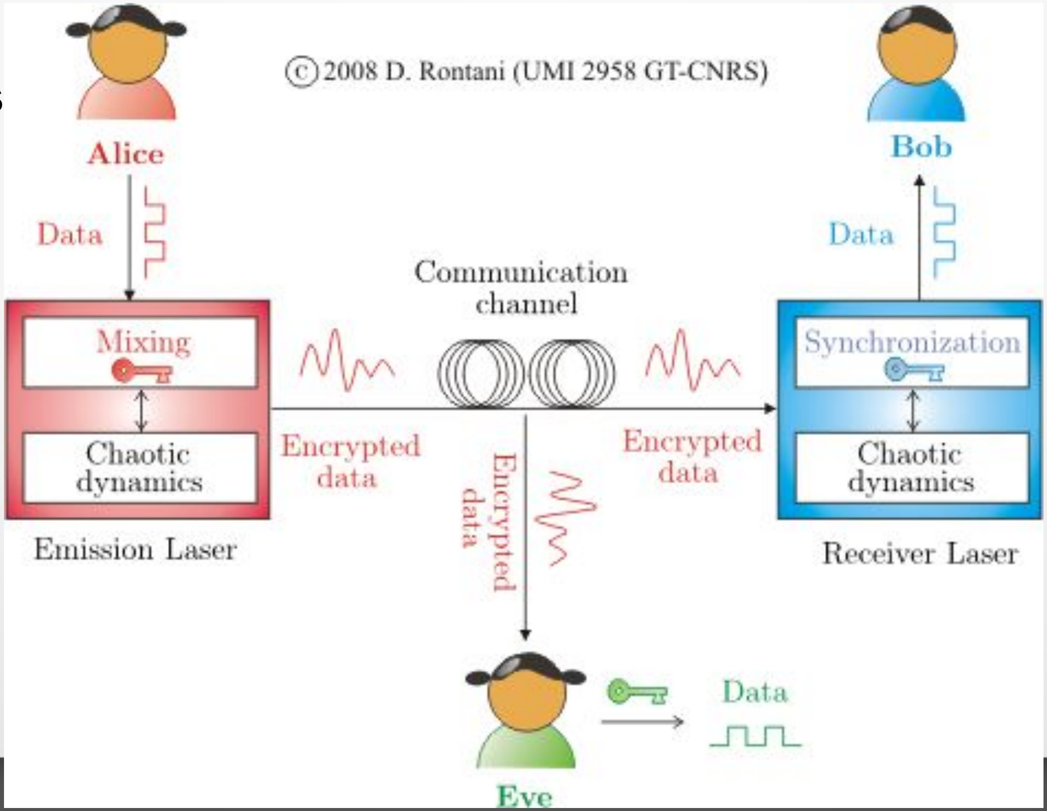
Description of security in communication scheme

Alice encrypts her data using certain security parameters.

Then, the encrypted data is transmitted to Bob, who decrypts it using synchronization.

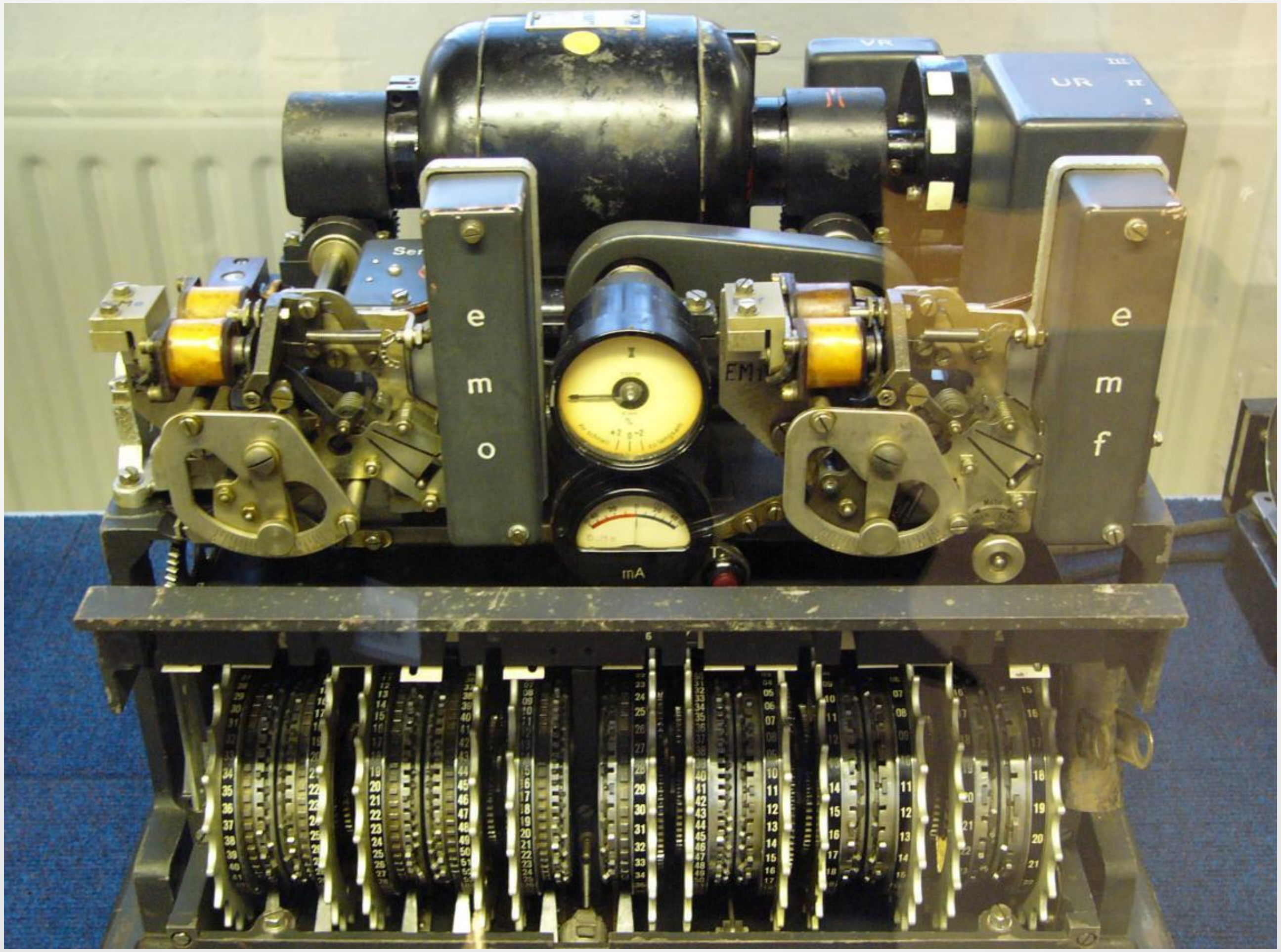
An eavesdropper, Eve, can hack the communication channel and decrypt the encrypted

data, if she manages



Cryptology World War II

- The German Lorenz cipher SZ42 (SZ for Schlüsselzusatz), one of the first security “machines” in the world … see, no software ;-)

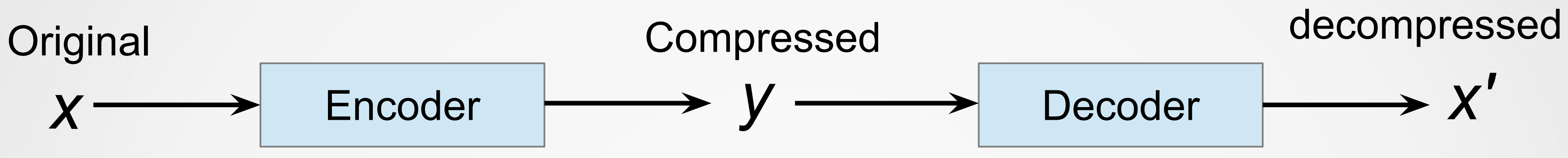


Information Theory ?

- The information theory (IT) frame work:
  - Communication System
  - Line Coding
  - Channel Coding (Shannon's 2nd theorem)
  - Cryptology
  - Source Coding (Shannon's 1st theorem)

The Source Coding Theorem

a.k.a.: The Noiseless Coding Theorem, Bit-rate (Data) Reduction, and Data Compression


Original

x

Encoder

Compressed

y

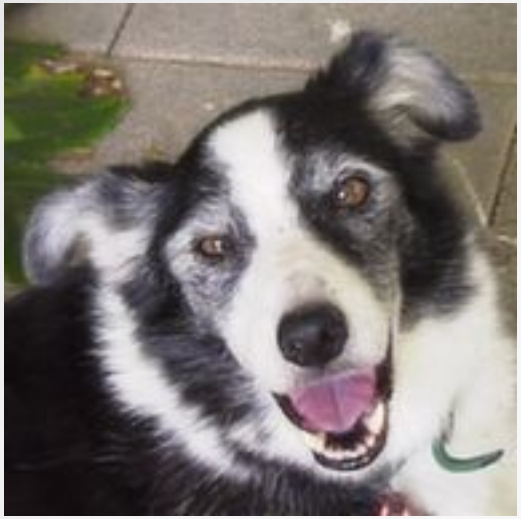
Decoder

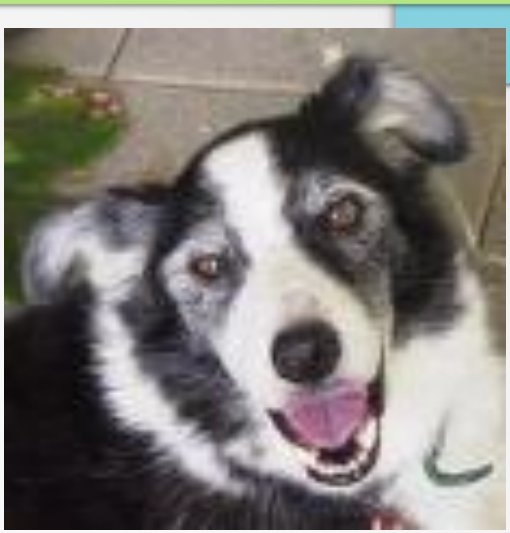
decompressed

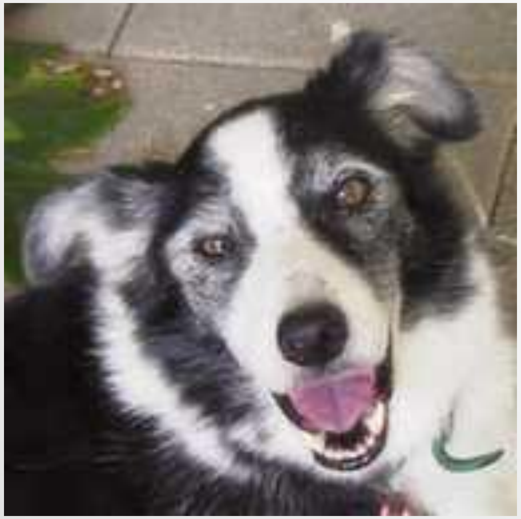
x'

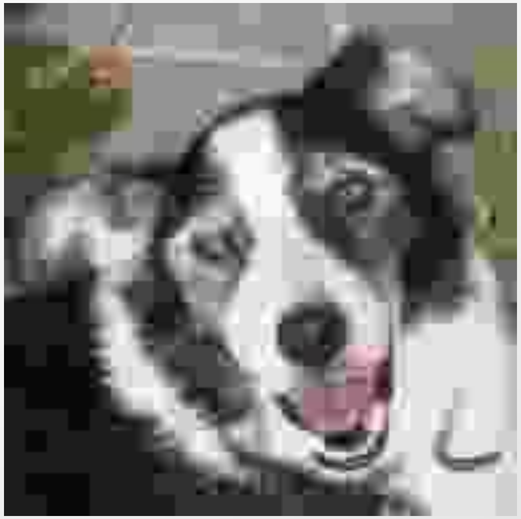
- **Lossless compression: *x* = *x’***
  - (a.k.a. *entropy coding, reversible coding*)
    - *Applications: text compression*
      - *“Do not send money” → “Do now send money”*
- **Lossy compression: *x* ≠ *x’***
  - (a.k.a. *irreversible coding*)
    - *Applications: image compression*
      - *Keywords: distortion vs. perception*

The Source Coding Theorem


Original (108.5 KB)


Low comp. (84% less info) 9.37 KB


Medium comp. (92% less info), 4.82 KB


High comp. (98% less info) 1.14 KB)

Compression Measurement Criteria

Compression ratio: |*x*|/|*y*|

|*x*| represents the number of bits in *y*

E.g.: |*x*| = 65,536, |*y*| = 16384, compression = 4:1

Alt., data has been reduced by (|*x*|-|*y*|)/|*y*| = 75%

Other measures of source coding performance

Bits per sample

E.g. **ASCII**: 8 bits/char, **RGB**: 24/48/72 bits/pixel

Distortion (Only on lossy methods)

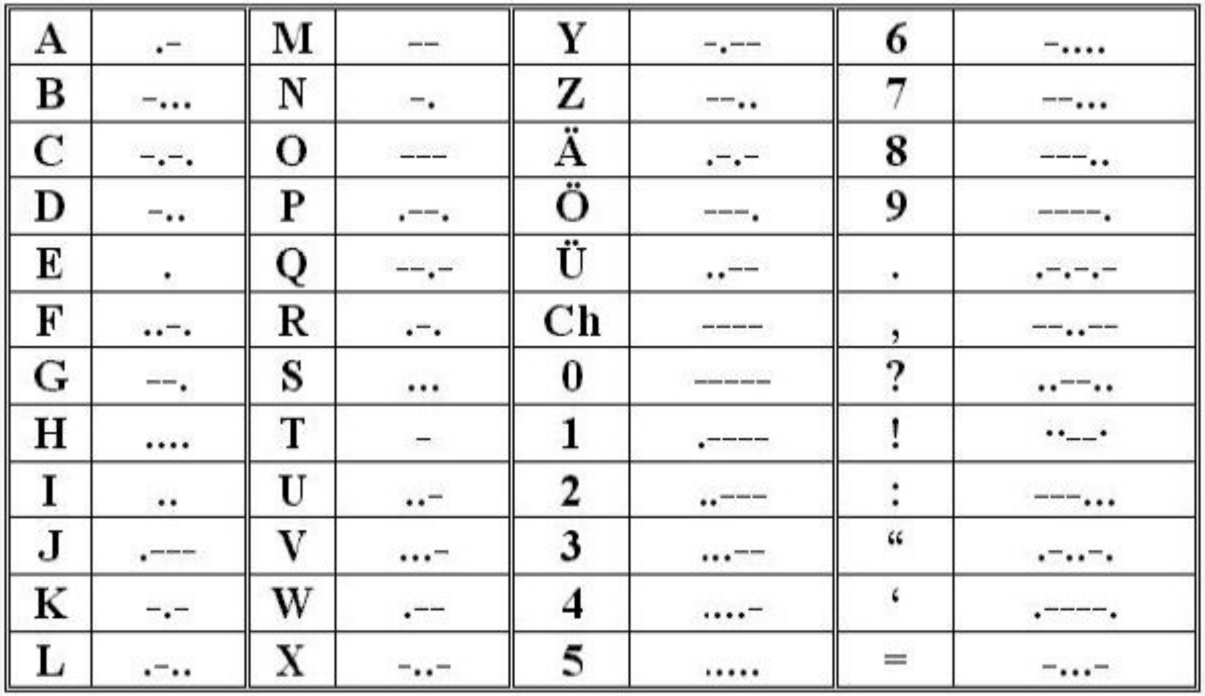
Depends on the human-perceived and/or mathematical difference between *x* and *x*’

The Source Coding Theorem

Now we will learn 3 concepts:

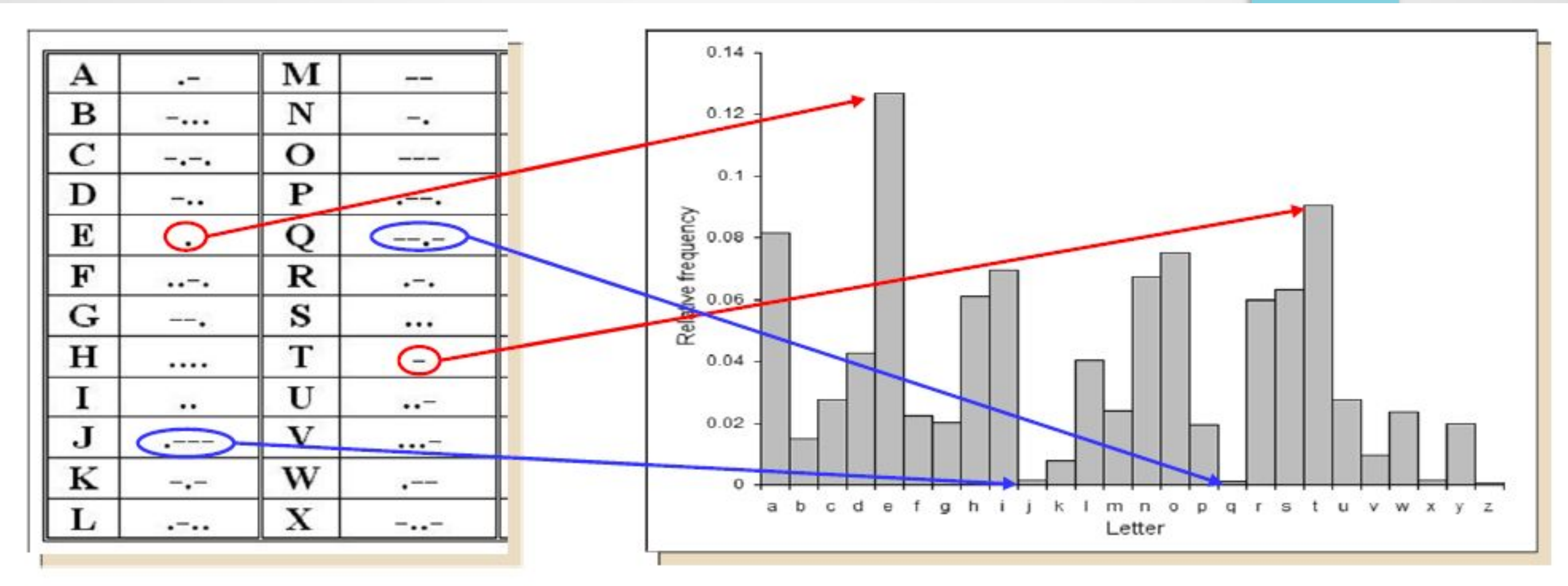
- Self-information
- Entropy
- Kraft's inequality

Real-World Coding: Morse(1844)



A	.-	M	--	Y	-.--	6	-....
B	-...	N	-.	Z	--..	7	--...
C	-.-.	O	---	Ä	.-.-	8	---..
D	-..	P	.--.	Ö	---.	9	----.
E	.	Q	--.-	Ü	..--	.	.-.-.-
F	..-.	R	.-.	Ch	----	,	--..--
G	--.	S	...	0	-----	?	..--..
H	....	T	-	1	.----	!	..__.
I	..	U	..-	2	..---	:	---...
J	.---	V	...-	3	...--	“	.-..-.
K	-.-	W	.--	4	....-	‘	.----.
L	.-..	X	-..-	5	.....	=	-...-

Real-World Coding: Morse(1844)



A	.-	M	--
B	-...	N	-.
C	-.-.	O	---
D	-..	P	.--.
E	.	Q	--.-
F	..-.	R	.-.
G	--.	S	...
H	....	T	-
I	..	U	..-
J	.---	V	...-
K	-.-	W	.--
L	.-..	X	-..-

- Generally, high/low frequency => short/long codes!!
- - Not 100% consistent!
- Example: E vs. T and I vs. M

The Source Coding Theorem - I

Assume:

- A source with finite number of symbols $S = \{s_1, s_2, \ldots, s_N\}$
- Symbol s_n has a probability $P(s_n) = p_n$
- Lets assume that for every symbol s_n, there is a l_n different bits (bits/symbol)
- Before we do any further progress, lets define the term: self-information that is:

Theorem 1.1: (Self Information)

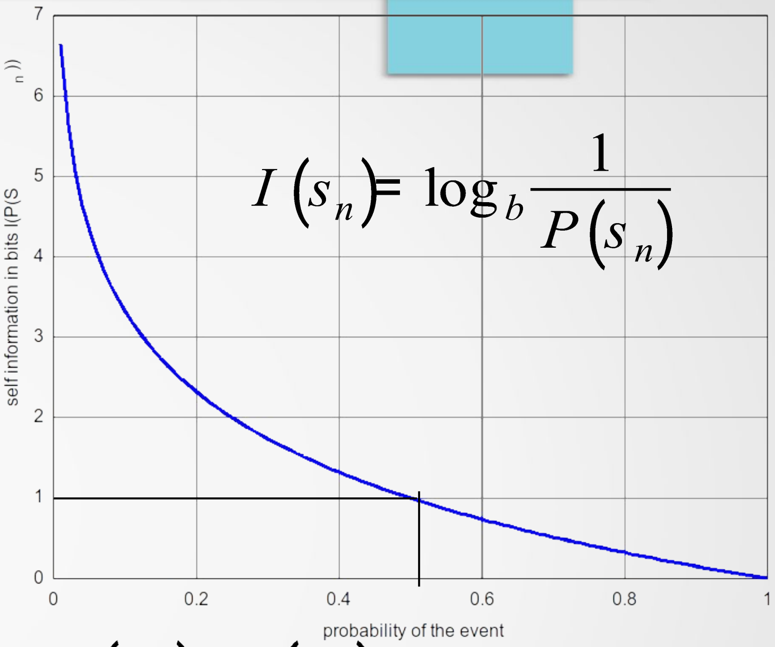
The self information is measured by:

$$I(s_n) = \log_b \frac{1}{P(s_n)} = -\log_b P(s_n)$$

Self-Information - I

- Properties of self-information:
  - barking of a dog during breaking a house carry or does not carry too much info if
    - If he only bark in dangerous cases (info)
    - If he is barking all the time (no-info)
  - Independent events have accumulated self-information, i.e.,

$$I\left(s_n s_j\right) = \log_b \frac{1}{P\left(s_n s_j\right)} = \log_b \frac{1}{P\left(s_n\right)P\left(s_j\right)} = I\left(s_n\right) + I\left(s_j\right)$$



$$I\left(s_n\right) = \log_b \frac{1}{P\left(s_n\right)}$$

  - to calculate the information in bits we need to take the logarithm with the base (*b* in the log base) = 2 and this mean:

$$\log_2(2) = 1\, bit$$

  - for b = *e (2.7183)*, information is measured in “nat” and for b = 10 is in dit (decimal digits)

Self-Information Examples

GUC

- Example 1: the out come of flipping a coin if:
  - The coin is fair, i.e., $P(H) = P(T) =$
    - then: $I(H) = I(T) = 1$
  - The coin tossing is not fair; some one cheating towards “T” ;-) :
  - $P(H) =$ and $P(T) =$
    - Then: $I(H) = 3bits, I(T) = 0.$ (No Surprise)

This is the entropy (H)

- If we have a set of independent events, s_n , then the average self-information associated with the experiment is

$$H = \sum P(s_n)I(s_n) = -\sum P(s_n)\log_2 P(s_n) = E_P($$

Entropy as Information Content

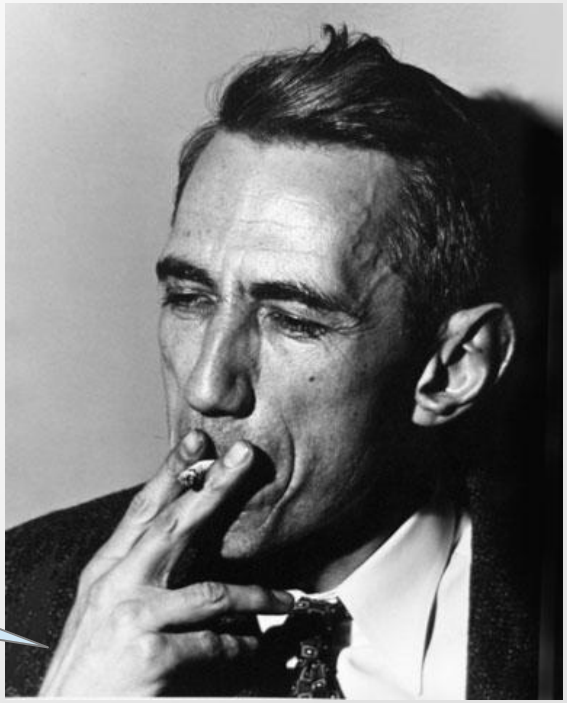
Entropy is a probabilistic model such that:

Independent fair coin flips have an entropy of 1 bit per flip. A source that always generates a long string of B's has an entropy of 0, since the next character will always be a 'B'.

Shannon showed that:

Definition 1.2: *If the experiments is a source that puts out symbols s_n from a set A, then the entropy is a measure of the average number of binary symbols (bits) needed for encoding the source*

Avg. message size in bits



The Source Coding Theorem - II

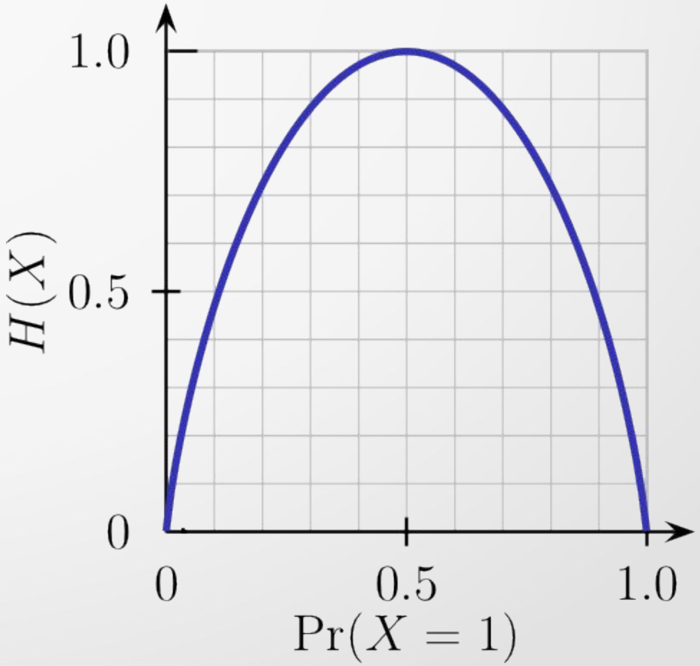
Theorem 1.2: (Entropy)

The minimum average length of a codeword is:

$$H\left(s_n\right)=P\left(s_n\right)\log_b \frac{1}{P\left(s_n\right)} = -P\left(s_n\right)\log_b P\left(s_n\right)$$

Entropy is the minimum expected average length.

If the average length decreased than *H*, then the code will not be decoded



The Source Coding Theorem - III

Code Word and Code Length:

Definition 1.2: A (binary) source code *C* for a random variable *S* is a mapping from *S* to a (finite) binary string (string of bits). Let $C(s_n)$ be the codeword corresponding to s_n and let $l(s_n)$ denote the length of $C(s_n)$.

To proceed, let us focus on codes that are "instantaneously" decoded, e.g., *Prefix Code*

Definition 1.3: A code is called a prefix code or an instantaneous code if no codeword is a prefix of any other codeword.

- Example:
  - Non-prefix: $\{s_1=0, s_2=01, s_3=011, s_4=0111\}$
  - Prefix: $\{s_1=0, s_2=10, s_3=110, s_4=111\}$

The Source Coding Theorem – IV

GUC

Theorem 1.3: (Shannon, 1992)

Any prefix code satisfies

$$\bar{l}(s_m) \geqslant H(a$$

Can you say why?

The Source Coding Theorem - V

Theorem 1.4: (Kraft-McMillan Inequality)

Any prefix (prefix-free) code satisfies

$$K(C) = \sum 2^{-l(s)}$$

Conversely, given a set of codeword lengths satisfying the above Inequality, one can have instantaneous code with these word lengths!! ***(see only can!)***

The Source Coding Theorem – IV

Theorem 1.4: (Kraft-McMillan Inequality)

Any prefix (prefix-free) code satisfies

$$K(C) = \sum 2^{-l(s)}$$

Proof:

Try to prove that $[K(C)]^n = \left[\sum 2^{-l(s)}\right]^n$ **does not grow up! i.e., less than 1 !**

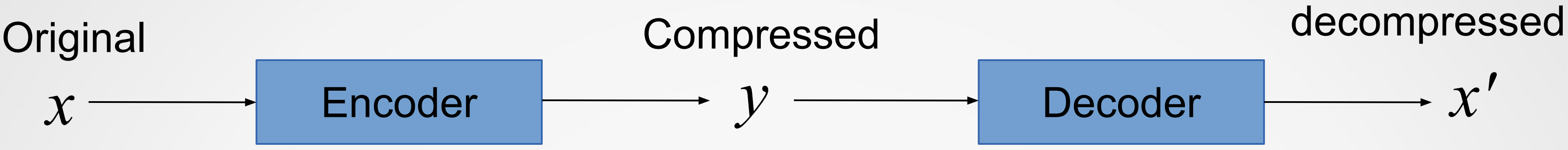
$$\therefore [K(C)]^n = \left[\sum 2^{-l(s)}\right]^n = \sum \sum \cdots \sum 2^{-(l(s_1)+l(s_2)+\cdots+l(s_n))}$$

Do It Your Self!!

$l(s_1) + l(s_2) + \cdots + l(s_n)$ **is simply the length of *n* codewords; can this < *n* ?**

Can you follow ? (self study; K. Sayood page: 32, chapter 2)

Notation for Sequences & Codes



Original x

Encoder

Compressed y

Decoder

decompressed x'

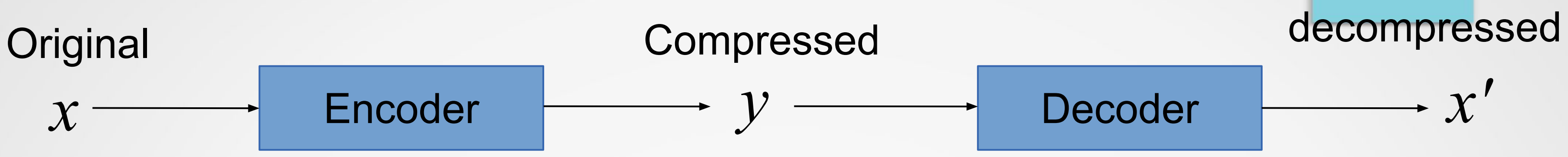
- Assume a sequence of symbols X = {X_1;X_2; … ;X_n} from a finite source alphabet A_X

- We almost always use A = {0, 1} (e.g. computer files, digital, communication) but the theory can be generalized to any finite set.

- Encoder: outputs a new sequence $Y = \{Y_1; Y_2; \ldots ; Y_n\}$, (using a possibly different code alphabet A_Y).

- A symbol code, C, is a mapping of $A_X \rightarrow A_Y$; We use $c(x)$ to denote the codeword to which C maps x.

Lossless Data Compression



- Let's focus on the lossless data compression problem for now, and not worry about noisy channel coding for now. In practice these two problems (noise and compression) are handled separately.
- An efficient code for the source data is able to (remove source redundancy)
- Then (if necessary) we design a channel code to help us transmit the source code over the channel (adding redundancy) almost error free.

Assumptions (for now):

  - the channel is perfectly noiseless, i.e., the receiver sees exactly the encoder's output
  - we always require the output of the decoder to exactly match the original sequence *X*.
  - *X* is generated according to a fixed probabilistic model, *p(X)*, i.e., which is known!!

- We will measure the quality of our compression scheme by examining the average length of the encoded string *Y*, i.e., over *p(X)*.

For Decoding, What Do We Need?

- We need to set some rules like:
  - How does the channel terminate the transmission? (e.g. it could explicitly mark the end, it could send only 0s after the end, it could send random garbadge after the end,...)
  - How soon do we require a decoded symbol to be known? (e.g. “instantaneously” { as soon as the codeword for the symbol is received, within a fixed delay of when its codeword is received, not until the entire message has been received,...)

- Thus, we need to study the code classes!