Applications of Information Theory

- Data compression
- Error correcting codes
- Cryptology
- Linguistics
- Statistics
- Computer science: distributed storage systems, caching, . . .
- Network information theory
- Bioinformatics: computational genomics, information flow in neural systems,. . .
- Machine learning

Machine Learning

- a large variety of machine learning and data-mining problems are about inferring global properties on a collection of agents by observing local noisy interactions of these agents.
- Examples
  - community detection in social networks,
  - image segmentation
  - data classification/clustering and information retrieval
  - protein-to protein interactions

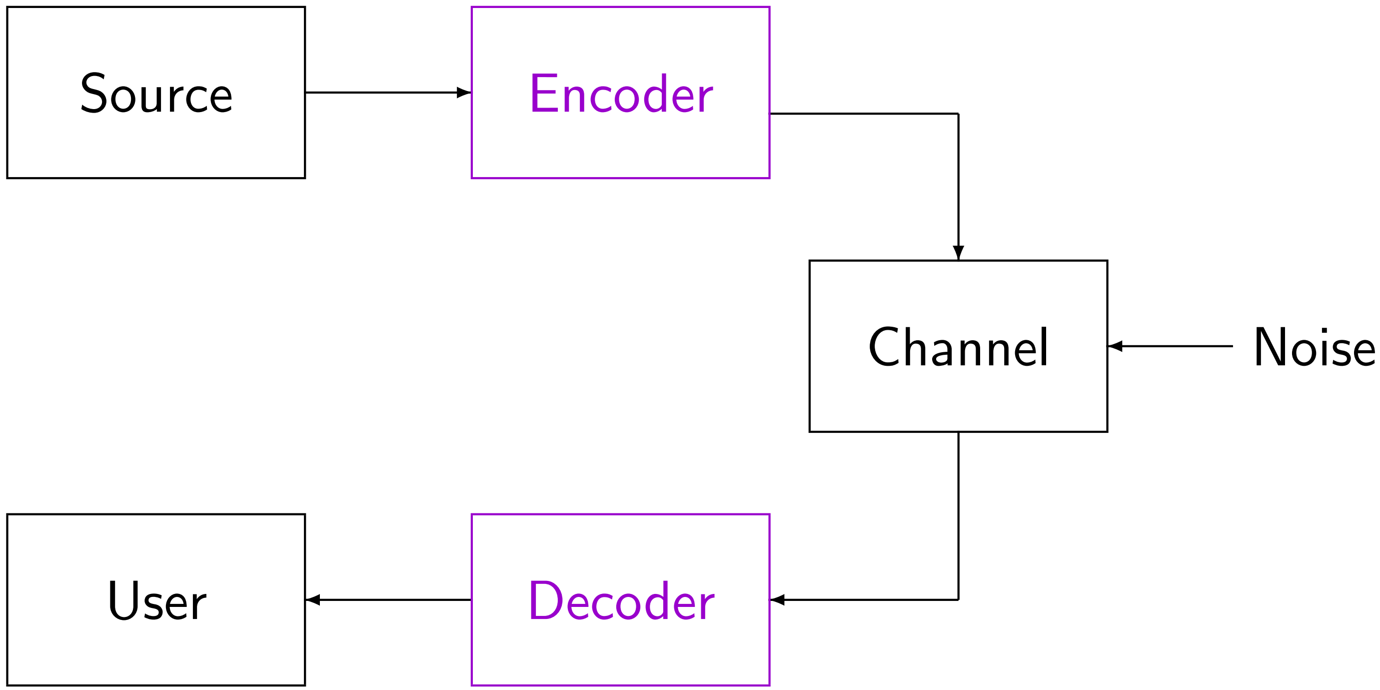
Information Theory?

- $\mathcal{L}_1$ and $\mathcal{L}_2$ two very large lists.
- Problem: find $(x_1, x_2) \in \mathcal{L}_1 \times \mathcal{L}_2$ such that $d(x_1, x_2)$ is small with time complexity $\ll |\mathcal{L}_1| \cdot |\mathcal{L}_2|$.
- Problem: given y, find whether there exists $x_1 \in \mathcal{L}_1$ such that $d(y, x_1)$ is small with time complexity $\ll |\mathcal{L}_1|$.

Content

- Upper bound on the compression rate of a source and on the information that passes through a noisy channel,
- algorithms that allow to attain these upper bounds.
- Other applications: distributed data storage.

Communication System

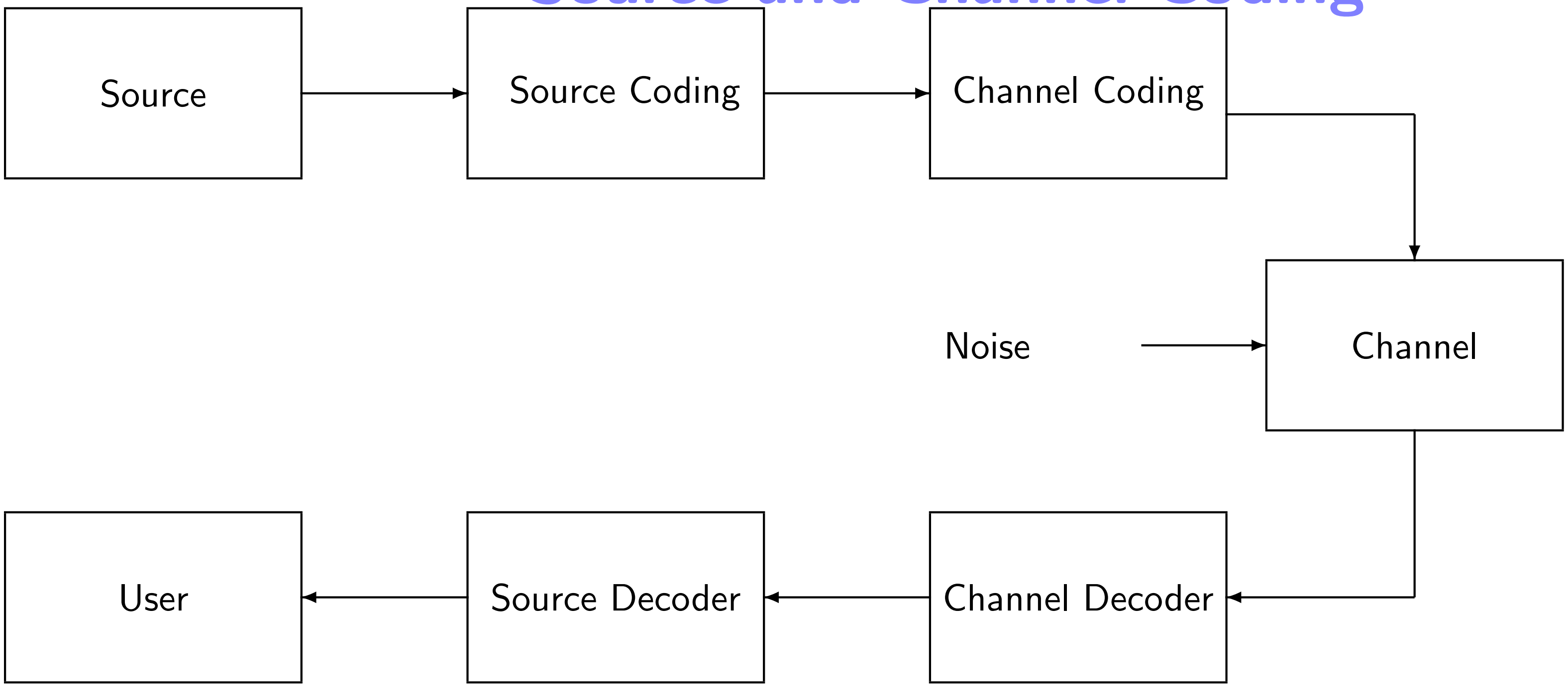


Source : voice, music, images, text, . . .

Channel : wireless communications, wire, optical fiber, flash drive,

Noise : electromagnetic perturbations, rain, inter-cell interferences, . . .

Source and Channel Coding



Efficiency: Transmit a maximum amount of information to another user by using a minimum amount of resources.

Reliability: The user should be able to recover the correct information (as much as possible)

Source/Channel Coding

Problem:

- Source Coding: compress efficiently a given source at a maximal compression rate. Ex:

$$\mathbf{x} = x_1 \dots x_n, \ \mathbf{Prob}(x_i = 1) = p.$$

- Channel Coding : transmit efficiently a maximum amount of information through a noisy channel. Ex:

$$\mathbf{x} = x_1 \dots x_n \overset{\text{channel}}{\rightsquigarrow} \mathbf{y} = y_1 \dots y_n, \ \mathbf{Prob}(y_i \neq x_i) = p.$$

A same quantity is used in both cases : entropy.

Source/Channel Coding

Problem:

- Source coding: compress efficiently a given source at a maximal compression rate. Ex:

  $$\mathbf{x} = x_1 \ldots x_n, \ \mathbf{Prob}(x_i = 1) = p.$$

  $\Rightarrow$ compress into a sequence of size $\approx nh(p)$ bits.

- Channel Coding : transmit efficiently a maximum amount of information through a noisy channel. Ex:

  $$\mathbf{x} = x_1 \ldots x_n \overset{\text{channel}}{\rightsquigarrow} \mathbf{y} = y_1 \ldots y_n, \ \mathbf{Prob}(y_i \neq x_i) = p.$$

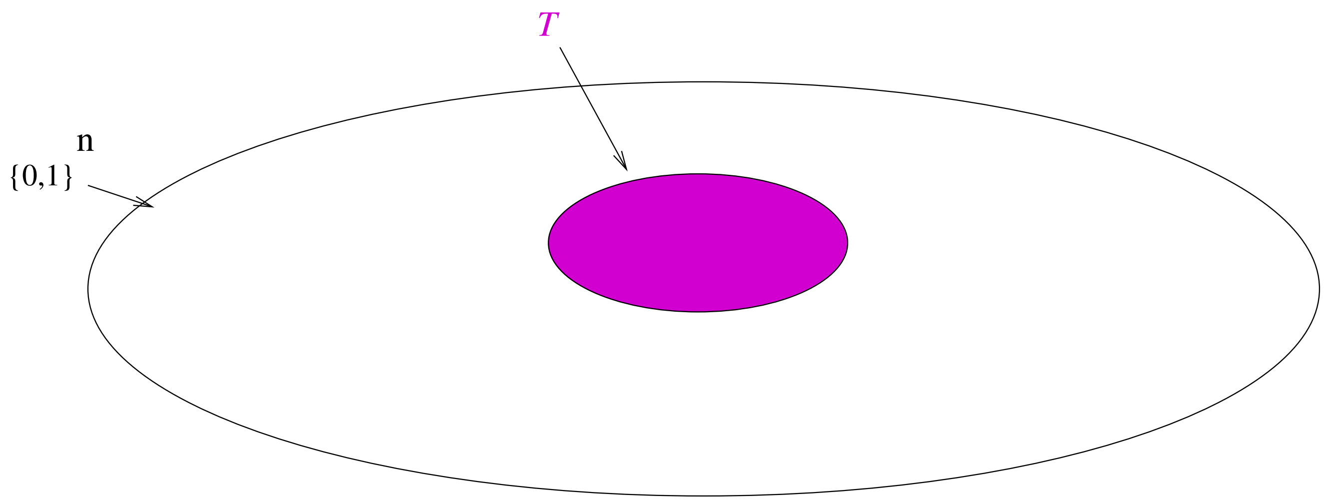
  $\Rightarrow$ transmit $\approx n(1 - h(p))$ bits of information.

A same quantity is used in both cases : entropy.

$$h(p) \overset{\text{def}}{=} -p \log_2 p - (1-p) \log_2 (1-p)$$

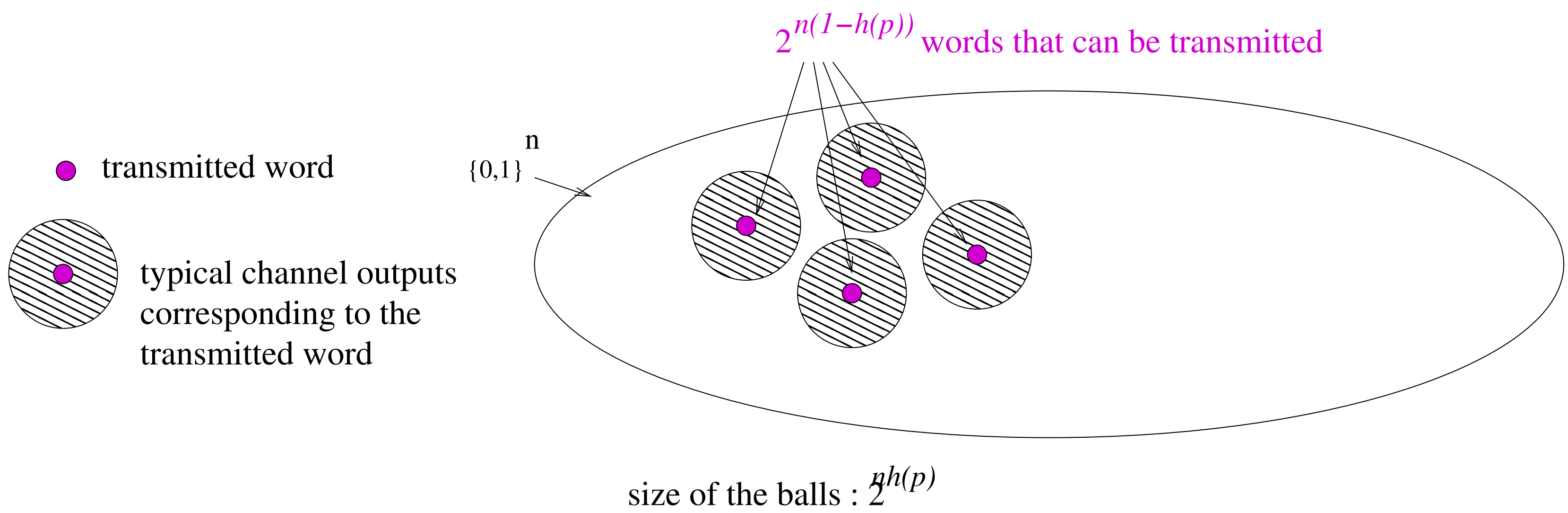
Entropy and typical sequences

A common principle : focus on typical outputs



$$
\begin{aligned}
T &= \{\mathbf{x}; |\mathbf{x}| \approx pn\} \\
\mathbf{Prob}(\mathbf{x} \in T) &\approx 1 \\
|T| &\approx 2^{nh(p)} \\
\log_2 |T| &\approx \text{Entropy}
\end{aligned}
$$

- Source Coding : number with $nh(p)$ bits the elements of T and do nothing for the others.
- Channel Coding :



$$\log(\text{number of words that can be transmitted}) = \text{number of bits of information}$$

Entropy

Formula can be explained by two facts

(i) $\log$ transforms a product into a sum,

(ii) concentration of a sum of i.i.d. r.v. around their expectation.

$$\begin{aligned} \log \mathbf{Prob}(\mathbf{x}) &\overset{(i)}{=} \log \mathbf{Prob}(x_1) + \cdots + \log \mathbf{Prob}(x_n) \\ &\overset{(ii)}{\approx} n\left(p \log p + (1-p)\log(1-p)\right) = -nh(p)(a.s.) \\ \Rightarrow \mathbf{Prob}(\mathbf{x}) &\approx 2^{-nh(p)}(a.s.) \end{aligned}$$

More generally for a r.v. X taking its values in $\mathcal{A}$:

$$\text{Entropy}(X) \overset{\text{def}}{=} -\sum_{a \in \mathcal{A}} \mathbf{Prob}(X=a) \log \mathbf{Prob}(X=a).$$

Repetition Code

To fight against noise, redundancy is added. For instance with the repetition code of length 3

$$\begin{aligned} 0 &\mapsto 000 \\ 1 &\mapsto 111 \end{aligned}$$

or more generally with a repetition code of length $2m+1$.

$$\begin{aligned} 0 &\mapsto \overbrace{0\ldots0}^{2m+1} \\ 1 &\mapsto \overbrace{1\ldots1}^{2m+1} \end{aligned}$$

Repetition Code

For an error probability of the channel $p = 0.01$, there are 0 or 1 corrupted bits with probability

$$(1-p)^3 + 3p(1-p)^2 \approx 0.9997$$

and 2 or 3 errors with probability

$$3p^2(1-p) + p^3 \approx 3 \times 10^{-4}$$

Information is badly recovered with probability $\approx 3 \times 10^{-4}$. With a repetition code of length 5, this probability drops to 10^{-5}.

$$10p^3(1-p)^2 + 5p^4(1-p) + p^5 \approx 10^{-5}$$

The rate of this code is 0.2.

Rate of a code

The repetition code of length 3 has rate $1/3 = 0.33$ and corrects one error.

The repetition code of length 5 has rate $1/5 = 0.2$ and corrects two errors.

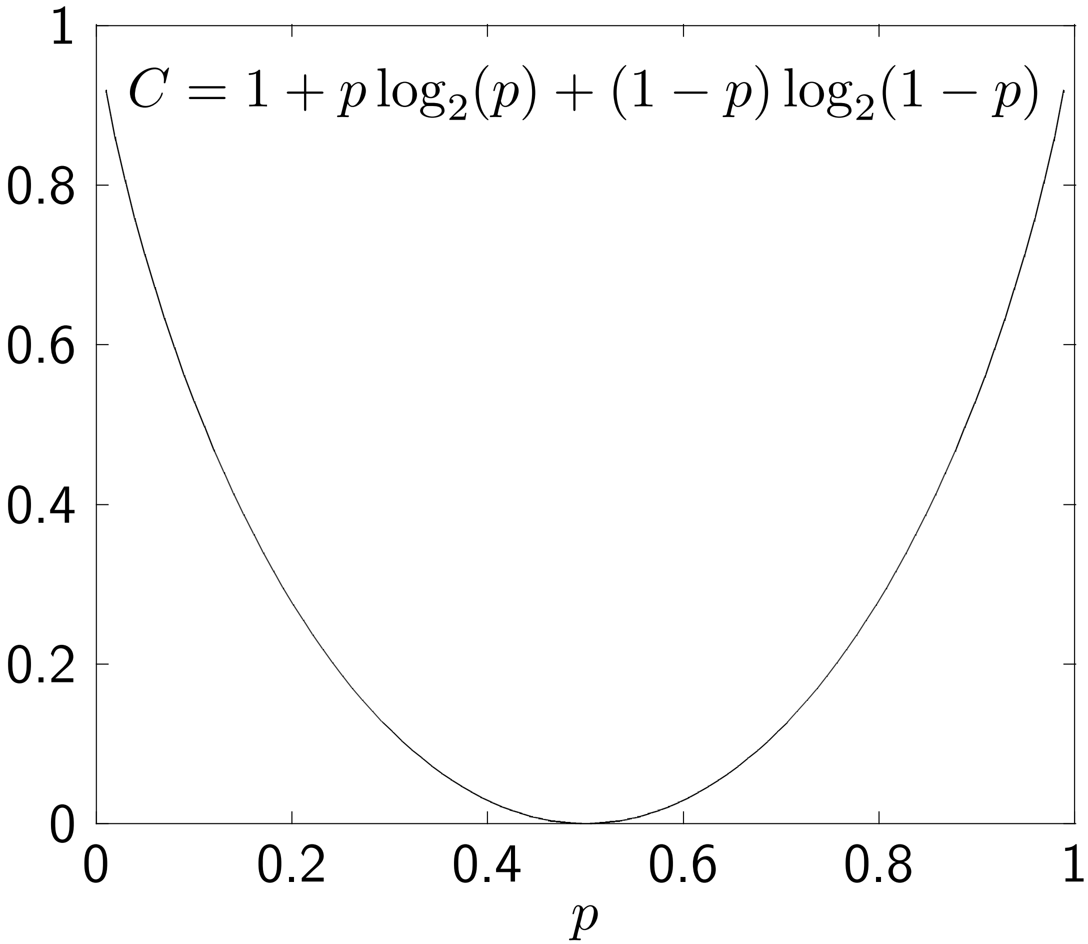
By lowering the rate, one can also lower the error probability after decoding.

Does the best transmission rate have to go to 0 ?

No ! Shannon's second theorem.

Notion of channel capacity.

Capacity of a binary symmetric channel



The capacity is the **maximum rate** at which reliable transmission is still possible.

For ex. $C(0.01) = 0.919$. It is therefore possible to improve significantly upon the repetition code.

Fundamental Results of this Course

Shannon's 1st theorem (Source Coding)

1. Every "reasonable" source can be encoded by using a number of bits per source symbol which is arbitrarily close to the source entropy.
2. It is impossible to do better...

Shannon's 2nd theorem (Channel Coding)

1. Information can be transmitted reliably by using an error correcting code with a rate smaller than the channel capacity.
2. It is impossible to do better.

TD

Today : exercise session.

Then programming algorithms in Java (4TD's on source coding, 3 TD's on channel coding, 1TD on distributed data storage).