Fundamentals of Information Theory

Data Compression

Yayu Gao
School of Electronic Information and Communications
Huazhong University of Science and Technology
Email: yayugao@hust.edu.cn

Outline

- Three key questions about data compression
- What is source coding?
- Get to know some codes
- What do we want from a source code?
- Kraft inequality——constraints on prefix codes
- How to find the optimal code?
- Shannon’s first theorem——Zero-error source coding theorem
- From Theory to Applications: source coding algorithms

本节学习目标

1. 写出Kraft inequality的表达式
2. 写出最优码优化问题的建立
3. 求解最优码优化问题
4. 求解最优码长的上下界
5. 写出无失真信源编码定理
6. 说出香农第一定理的意义

重难点：

- Kraft inequality
- 最优码优化问题
- 香农第一定理

Review:上节学习目标

1. 理解效率与可靠性之间的折衷关系
2. 说出信源编码器与信源译码器各自的目标
3. 写出信源编码效率的评价指标
4. 说出信源编码优化问题
5. 说出什么是non-singular code
6. 说出什么是Uniquely decodable code
7. 说出什么是prefix code
8. 说出以上三种code的优缺点
9. 说出对信源编码的三个要求

重难点：

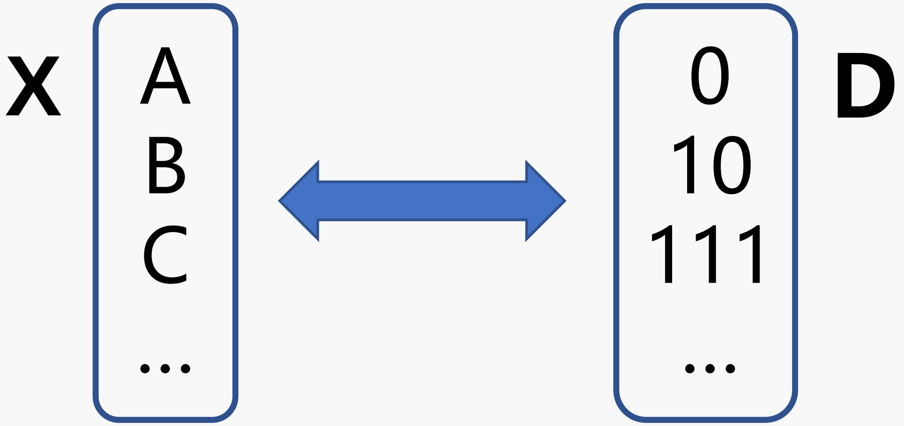
- 信源编码优化问题
- 认识几种编码类型

Review: Source code

- A source code ***C*** for a random variable X is a mapping between the space of X to the space of code D.

$$\mathbf{C} : \mathbf{X} \rightarrow \mathbf{D} : \mathbf{C}(\mathbf{x}),$$

where $\mathbf{D}$ is the set of finite length strings of symbols from a D-ary alphabet[1].



- Let $C(x)$ denote the codeword corresponding to x.
- Let $l(x)$ denote the length of $C(x)$.

Review: Expected length of a source code

- Definition: The expected length $L(C)$ of a source code $C(x)$ for a random variable X with p.m.f. $p(x)$ is given by

$$\boxed{L(C) = \sum_{x \in \mathcal{X}} p(x) l(x)}$$

where $l(x)$ is the length of the codeword associated with x.

- Shorter average code length $\Rightarrow$ Higher efficiency $\Rightarrow$ Better compression

Review: What do we want from a source code?

- **Efficiency**
  - Find codes with the minimum average code length.
- **Reversibility**
  - The code must be uniquely decodable
- **Instantaneous code**
  - Detect where the code for one input symbol ends and the next begins.
- **Easy implementation** of the code
  - From algorithm design's point of view

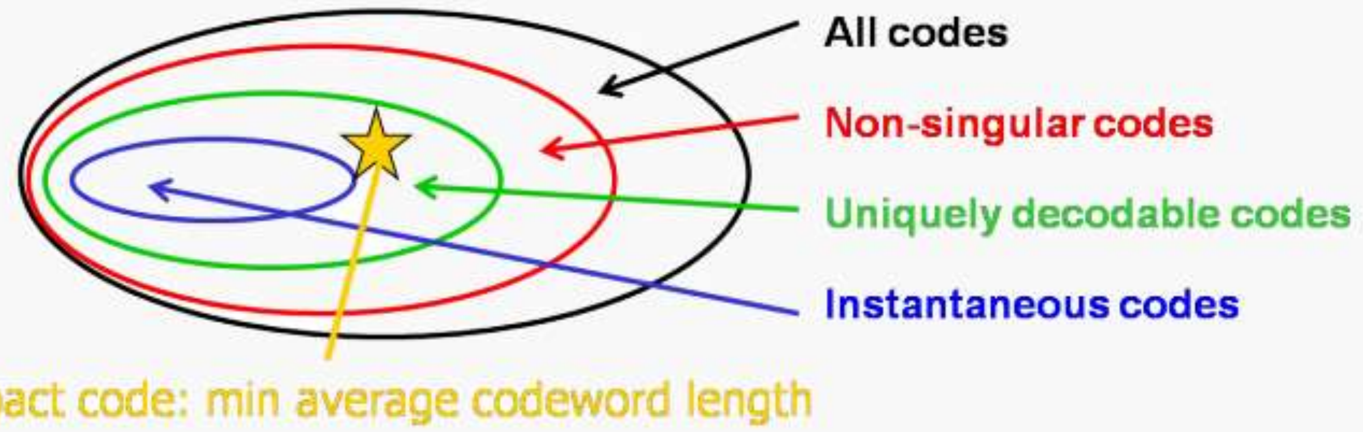
Compression

Zero-error

Engineering

Review: What do we want from a source code?

- In general, the optimal zero-error source coding problem is equivalent to find the optimal (shortest average length) uniquely decodable codes.
- Such a targeted code is called a compact code.
  - The uniquely decodable code with the smallest average code length for an information source S.
  - **How short can it be?**
    - **Shannon's first theorem**

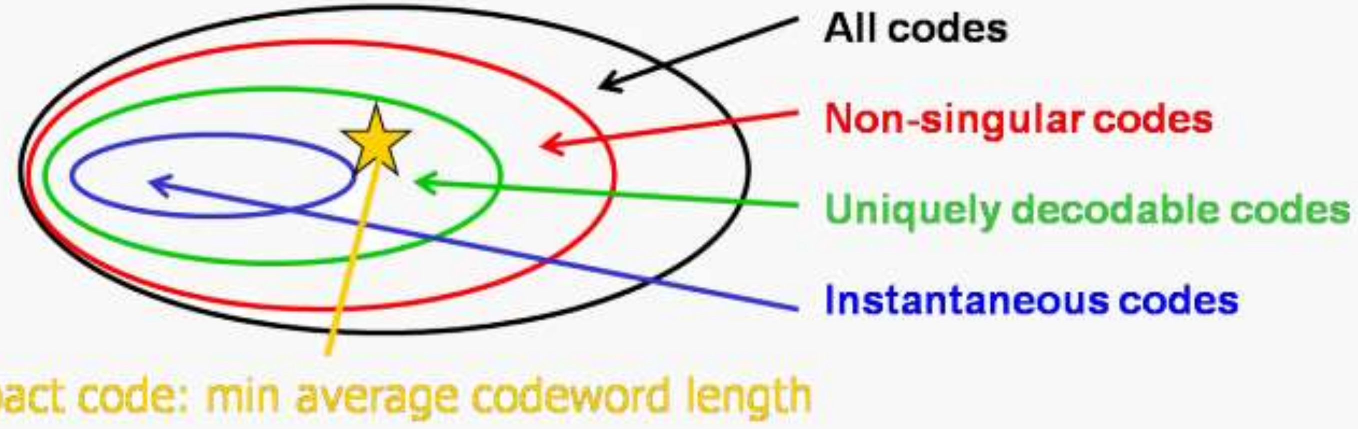


05

Kraft Inequality

——constraints on prefix codes

Kraft inequality: preview



- Kraft inequality was proposed by L. G. Kraft in 1949.
- It provides a **constraint** requirement on the **codeword lengths of any instantaneous code**.
- To construct an instantaneous code, what are the possible codeword lengths?

Kraft inequality

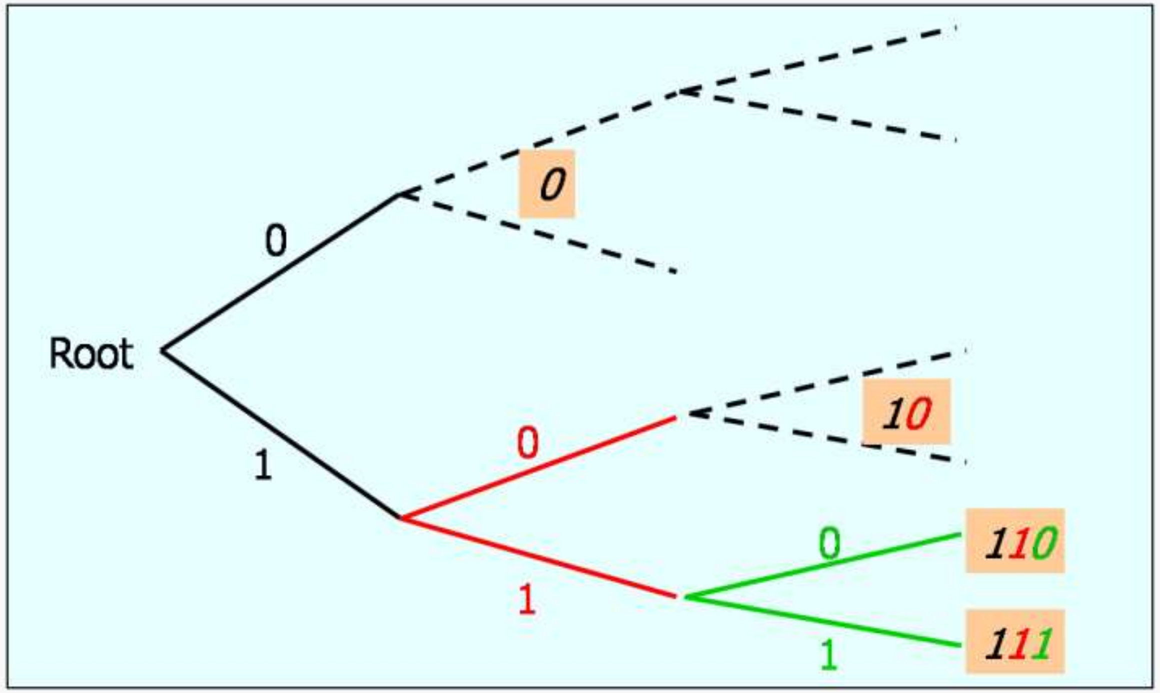
- For any instantaneous code over an alphabet of size D, the codeword lengths $\{l_1, l_2, \ldots, l_m\}$ must satisfy the inequality:

$$\boxed{\sum_{i=1}^{m} D^{-l_i} \le 1,}$$

  where m is the number of codewords.
- Converse: for codeword lengths satisfying the above inequality, there exists an instantaneous code.

Kraft inequality: code tree

- We can always construct the code tree of a prefix code.



- Each codeword of an instantaneous code must be a leaf node of the tree.
- No codeword is an ancestor of any other codeword on the tree.
- Each codeword eliminates its descendants as possible codewords.

Kraft inequality: proof

Kraft inequality: a short history

- Applicable for prefix codes: first proposed by L. G. Kraft in 1949.

- Applicable for uniquely decodable codes: proved by B. McMillan in 1956.

- Applicable for uniquely decodable codes: a simplified proof by J. Karush in 1961.

Kraft inequality: assignment #1

- Q1: $r.v.X$

$$\begin{aligned}\Pr(X=a)&=0.5,\\ \Pr(X=b)&=0.25,\\ \Pr(X=c)&=0.125,\\ \Pr(X=d)&=0.125.\end{aligned}$$

$$\begin{aligned}C(a)&=00,\\ C(b)&=10,\\ C(c)&=01,\\ C(d)&=11.\end{aligned}$$

微助教

- Is this code good enough?
- Could you design a binary instantaneous code for the information source with
  - code length 1, 2, 3 and 3, respectively?
  - code length 1, 2, 2 and 3, respectively?

06

How to find the optimal code?

What do we want from a source code?

- **Efficiency**
  - Find codes with the minimum average code length.
- **Reversibility**
  - The code must be uniquely decodable
- **Instantaneous code**
  - Detect where the code for one input symbol ends and the next begins.

$$\boxed{\sum_{i=1}^{m} D^{-l_i} \leq 1,}$$

Compression

Zero-error

Engineering

Optimal codes: formulate the problem

- Objective: find the **instantaneous code** with the **minimum expected length**

Objective function

$$\min_{l_1, l_2, \ldots, l_m} L = \sum_{i=1}^{m} p_i l_i$$

Constraint

$$\text{subject to } \sum_{i=1}^{m} D^{-l_i} \leq 1,$$

over integers $\{l_1, l_2, \ldots, l_m\}$.

- How to solve it? → Method of Lagrange multipliers

Optimal codes: solve the problem

- For an optimization problem with inequality constraints:

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & g(x) \leq 0 \end{aligned} \longrightarrow \begin{aligned} \min_{l_1, l_2, \ldots, l_m} \quad & L = \sum_{i=1}^{m} p_i l_i \\ \text{subject to} \quad & \sum_{i=1}^{m} D^{-l_i} - 1 \leq 0, \end{aligned}$$

- Construct a new function L with the Lagrange multiplier λ:

$$L(\lambda, x) = f(x) + \lambda g(x) \longrightarrow L(\lambda, l_i) = \sum_{i=1}^{m} p_i l_i + \lambda(\sum_{i=1}^{m} D^{-l_i} - 1)$$

- The optimal solution must satisfy KKT conditions:

$$\begin{cases} \frac{\partial L(\lambda, x)}{\partial x} = 0 \\ \lambda g(x) = 0 \end{cases} \longrightarrow \begin{cases} \frac{\partial L(\lambda, l_i)}{\partial l_i} = 0 \\ \lambda(\sum_{i=1}^{m} D^{-l_i} - 1) = 0 \end{cases}$$

Optimal codes: solution over real code lengths

- By solving the constrained minimization with the method of Lagrange multipliers, the optimal code lengths are given by

$$l_i^* = -\log_D(p_i)$$

- The minimum average code length is:

$$\boxed{L^* = \sum_{i=1}^{m} p_i l_i^* = -\sum_{i=1}^{m} p_i \log_D(p_i) = H_D(x).}$$

- However, it is the solution over real code lengths.
- In practice, the code lengths must be integers.

Optimal codes: lower bound

- Theorem: Expected code length L of any instantaneous D-ary code for a $r.v.\ X$.

$$\boxed{L \geq H_D(X),}$$

the equality holds if and only if $p(x_i) = D^{-l(x_i)}$.

- For uniquely decodable D-ary symbol code, define $H_D(X) = -\sum_x p(x) \log_D p(x)$.

$$L(C,X) = \sum_{i=1}^{m} p(x_i) l(x_i) = \sum_x p(x) \log_D D^{l(x)} \qquad \left(l(x) = \log_D D^{l(x)}\right)$$

$$= H_D(X) + \sum_x p(x) \log_D \frac{p(x)}{D^{-l(x)}} \qquad \left(\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq (\sum_{i=1}^{n} a_i) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}\right)$$

$$\geq H_D(X) + \sum_x p(x) \cdot \log_D \frac{\sum_x p(x)}{\sum_x D^{-l(x)}} = H_D(X) + 1 \cdot \log_D \frac{1}{\sum_x D^{-l(x)}} \qquad \left(\sum_x D^{-l(x)} \leq 1\right)$$

$$\geq H_D(X)$$

Optimal codes: is there an upper bound?

- The optimal length $l(x) = \log_D \frac{1}{p(x)}$ may not to be integer.
- Then we round it up as $l(x) = \lceil \log_D \frac{1}{p(x)} \rceil$.
- These codeword lengths satisfy the Kraft inequality.

$$\sum_x D^{-\lceil \log_D \frac{1}{p(x)} \rceil} \leq \sum_x D^{-\log_D \frac{1}{p(x)}} = \sum_x p(x) = 1$$

- So there exists a (uniquely decodable) prefix code with these codeword lengths, we have

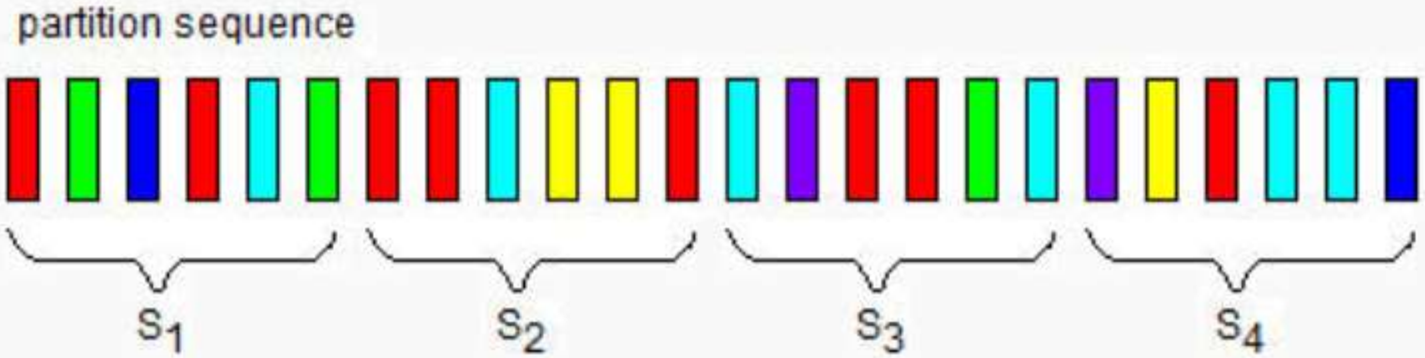
$$\begin{aligned}
\log_D\left(\frac{1}{p(x)}\right) \leq \quad & l(x) < & \log_D\left(\frac{1}{p(x)}\right)+1 \\
\sum_x p(x)\log_D\left(\frac{1}{p(x)}\right) \leq \quad & \sum_x p(x) l(x) < & \sum_x p(x)\left\{\log_D\left(\frac{1}{p(x)}\right)+1\right\} \\
H_D(X) \leq \quad & L(C,X) < & H_D(X)+1
\end{aligned}$$

Optimal codes: is there an upper bound?

- Expected code length of an optimal D-ary code for X

$$\boxed{H_D(X) \leq L^* < H_D(X) + 1,}$$

- There is an overhead that is at most 1 bit. Why?
  - The optimal code length $\log_D \frac{1}{p_i}$ may not be integer.
- What do you think? Is this overhead small enough for you?
- Can we reduce the overhead per symbol?



Can we reduce the overhead per symbol?

- Let us send a sequence of n symbols from X, which is $\{x_1, x_2, ..., x_n\}$.
- $l(x_1, x_2, ..., x_n)$: the codeword length of $\{x_1, x_2, ..., x_n\}$.
- L_n: the expected codeword length per input symbol.

$$\begin{aligned} L_n &= \frac{1}{n}\sum p(x_1, x_2, ..., x_n) l(x_1, x_2, ..., x_n) \\ &= \frac{1}{n} El(X_1, X_2, ..., X_n) \end{aligned}$$

- By applying the bounds derived above:

$$H(X_1, X_2, ..., X_n) \leq El(X_1, X_2, ..., X_n) < H(X_1, X_2, ..., X_n) + 1$$
$$\frac{H(X_1, X_2, ..., X_n)}{n} \leq L_n < \frac{H(X_1, X_2, ..., X_n)}{n} + \frac{1}{n}$$

- If $X_1, X_2, ..., X_n$ are i.i.d, then?
- If $X_1, X_2, ..., X_n$ are stationary, then?

07

Shannon's first theorem

Zero-error source coding theorem

Shannon's first theorem

- **Theorem:** the minimum expected codeword length per symbol satisfies

$$\boxed{\frac{H(X_1,X_2,\ldots,X_n)}{n} \le L_n^* < \frac{H(X_1,X_2,\ldots,X_n)}{n} + \frac{1}{n}.}$$

- Moreover, if $X_1, X_2, \ldots, X_n$ is a stationary stochastic process,

$$\boxed{L_n^* \to H(\mathcal{X}),}$$

Entropy Rate

- What is the significance of entropy rate?
  - Shortest average description length per symbol of a process.
  - Ultimate data compression rate

Shannon's first theorem: another presentation

- Source coding theorem
  - For a binary information source S and arbitrary ε, there exists a binary instantaneous code for which the average code length L per coding symbol satisfies

$$\boxed{H(S) \leq L_n^* < H(S) + \varepsilon.}$$

- **Source coding limit**: the average code length per symbol of an instantaneous code for an information source can be made as close to the entropy as desired, but never be smaller.
- If the average code length per symbol is smaller than the entropy, you cannot find an instantaneous code.
  - Errors will occur when decoding.

What if the code is designed for the wrong distribution?

- In practice, the true distribution of the source p(x) may be unknown.
- We may have a best estimation of the true distribution, a wrong distribution q(x).
- Then we may design the code length as $l(x) = \left\lceil \log \frac{1}{q(x)} \right\rceil$
- In this case, we will not achieve expected length L=H(p).
- Instead, the expected length would be

$$
\begin{aligned}
El(X) &= \sum_x p(x) \left\lceil \log \frac{1}{q(x)} \right\rceil < \sum_x p(x) \left(\log \frac{1}{q(x)} + 1 \right) \\
&= \sum_x p(x) \log \frac{p(x)}{q(x)} \frac{1}{p(x)} + 1 \\
&= \sum_x p(x) \log \frac{p(x)}{q(x)} + \sum_x p(x) \log \frac{1}{p(x)} + 1 = D(p||q) + H(p) + 1.
\end{aligned}
$$

What if the code is designed for the **wrong** distribution?

Theorem 5.4.3 (*Wrong code*) *The expected length under $p(x)$ of the code assignment* $l(x) = \left\lceil \log \frac{1}{q(x)} \right\rceil$ *satisfies*

$$H(p) + D(p||q) \le E_p l(X) < H(p) + D(p||q) + 1.$$

微助教

- Insights:
  - The increase in expected description length due to the incorrect distribution is the **relative entropy.**
  - Believing that the distribution is q(x) when the true distribution is p(x) incurs **a penalty of D(p||q) in the average description length.**
  - Relative entropy**: the increase in descriptive complexity due to incorrect information**

Revisiting: What do we want from source codes?

- **Efficiency**
  - Find codes with the minimum average code length.
- **Reversibility**
  - The code must be uniquely decodable
- **Instantaneous code**
  - Detect where the code for one input symbol ends and the next begins.
- **What if we expand the allowed codes to uniquely decodable codes?**

Compression

Zero-error

Engineering

Can we do better if we loose the constraint?

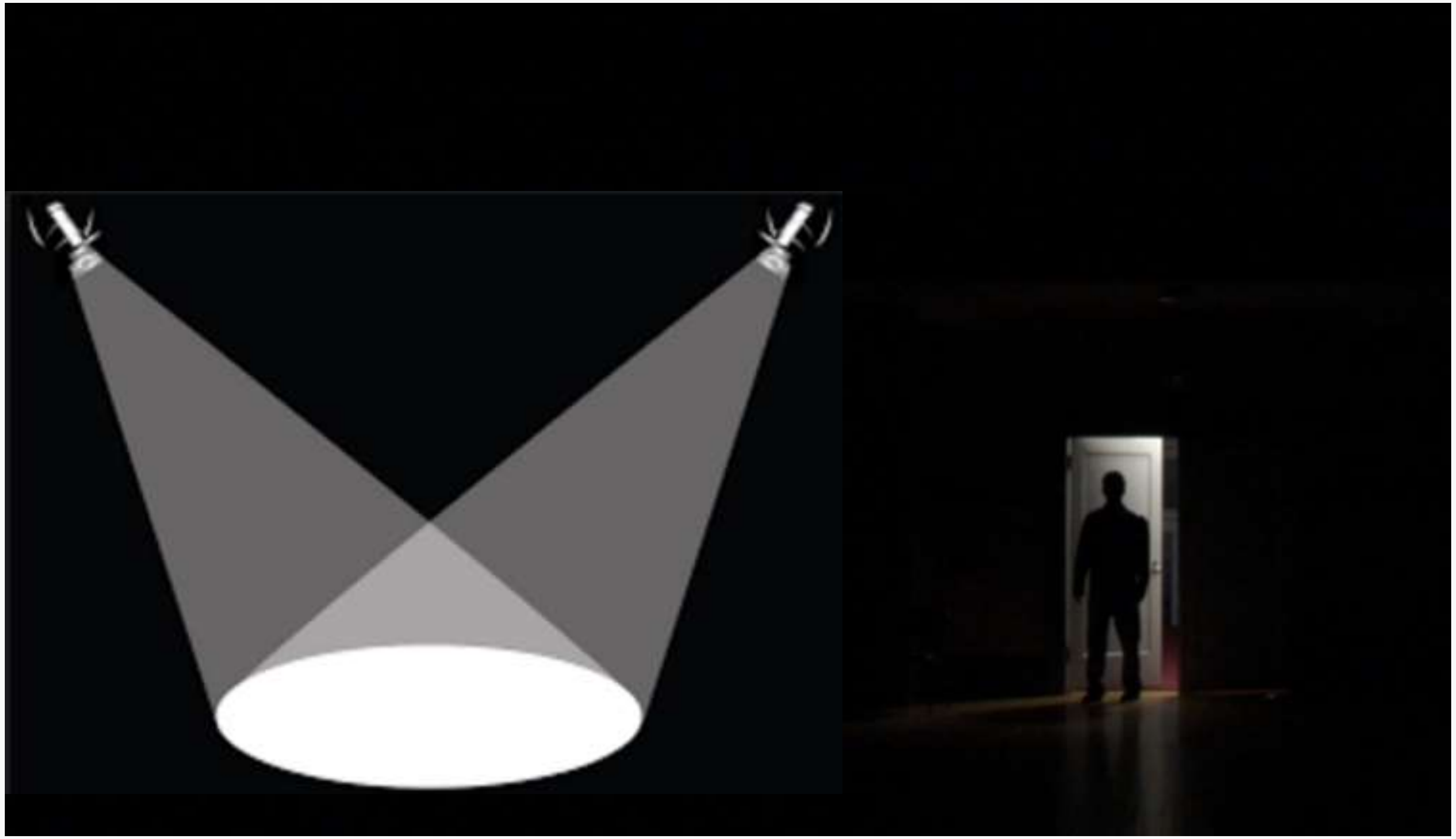
- What if we expand the allowed codes to uniquely decodable codes?
- Recall: Kraft inequality and the converse still hold for all uniquely decodable codes.
- **Surprising fact**: **Uniquely decodable codes does not offer any further choices for the codeword lengths than prefix codes.**
- The theorem can be extended to show the existence of uniquely decodable code for any information source.
  - Uniquely decodable codes are the basic requirements of the zero-error coding.
  - This theorem is also called zero-error source coding theorem

Revisiting: can we compress the data unlimitedly?

- Data compression has a limit? **Yes!**
- What is the limit? **Entropy of the source**

Source coding theorem: reflection

- Zero-error source coding theorem
    - Provide the theoretical limit to achieve the ideal coding
    - Prove the existence of the ideal source code.



本节学习目标

1. 写出Kraft inequality的表达式
2. 写出最优码优化问题的建立
3. 求解最优码优化问题
4. 求解最优码长的上下界
5. 写出无失真信源编码定理
6. 说出香农第一定理的意义
7. 理解相对熵在编码层面的意义

重难点：

- Kraft inequality
- 最优码优化问题
- 香农第一定理

My Homepage

Thank you!

Yayu Gao
School of Electronic Information and Communications
Huazhong University of Science and Technology
Email: yayugao@hust.edu.cn