Shannon's Theory (contd.)

Debdeep Mukhopadhyay

Assistant Professor
Department of Computer Science and Engineering
Indian Institute of Technology Kharagpur
INDIA -721302

Theorem

- Let (P,C,K,D,E) be an encryption algorithm. Then
  - **H(K|C)=H(K)+H(P)-H(C)**
- **Proof:** H(P,K)=H(C,K) [why?]

  or, H(P)+H(K) = H(K|C)+H(C)

  or, **H(K|C)=H(K)+H(P)-H(C)**

Equivocation (ambiguity) of key given the ciphertext

Perfect vs Ideal Ciphers

- **H(P)=H(C), then we have H(K|C)=H(K)**
  - **That is the uncertainty of the key given the cryptogram is the same as that of the key without the cryptogram.**
- ***Such kinds of ciphers are called "ideal ciphers"***
  - **For perfect ciphers, we had H(P)=H(P|C) or, equivalently H(C)=H(C|P)**

Perfect vs Ideal Ciphers

- For perfect ciphers, the key size is infinite if the message size is infinite.
  - however if a shorter key size is used then the cipher can be attacked by someone with infinite computational power.
- Thus, H(K|C) gives us this idea of security (or, insecurity)…

Unicity and Brute Force Attack

- Q: How to protect data against a brute force attacker with infinite computation power?
  - Shannon defined "**unicity distance**" (we shall call it unicity), as the least amount of plaintext which can be deciphered uniquely from the corresponding ciphertext: given unbounded resources by the attacker.
  - Often measured in units of bytes, letters, symbols.

An Important Point

- A common misconception: "any cipher can be attacked by exhaustively trying all possible keys":
- Thus DES which has a 56 bit key can also be broken by brute force.
  - But if the cipher is used within its unicity then even DES is theoretically secured, like the One Time Pad (OTP).

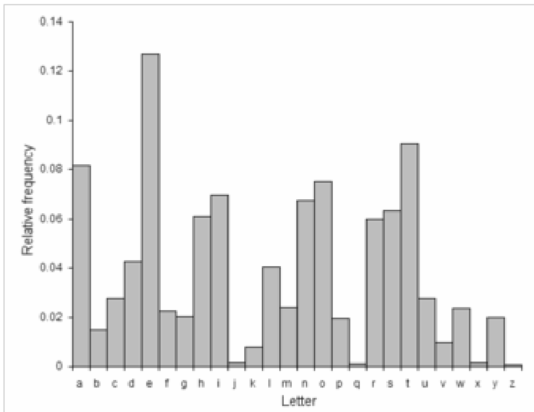
Spurious Keys

- Thus, H(K|C) is the amount of uncertainty that remains of the key after the cipher text is revealed.
  - We know, it is called the key equivocation
- Attacker to guess the key from the ciphertext shall guess the key and decrypt the cipher.
- He checks whether the plaintext obtained is "meaningful" English. If not, he rules out the key.
- But due to the redundancy of language more than one key will pass this test.
- Those keys, apart from the correct key, are called spurious.

Entropy of Plain Text

- H_L: measure of the amount of information per letter of "meaningful" strings of plaintext.
- A random string of plaintext formed using English letter has an entropy of $\log_2|26|\approx 4.76$
- But English letters have a probability distribution.

Frequency of English letters



A first order entropy of the English text is H(P)≈4.19

Higher Order Approximations

- A large number of digrams are tabulated and $H(P^2)$ is computed.
- The value is divided by 2 to obtain a second order approximation, $H(P^2)/2 \approx 3.90$
- One could continue obtain trigrams, etc and compute higher order approximations for the entropy.

In general…

- Successive letters have correlation, which reduces the entropy.
- Define P_n to be the random variable that has a probability distribution of n-grams of plaintext
- Define H_L as the **entropy of a natural language** L:

$$H_L = \lim_{n\to\infty} \frac{H(P_n)}{n}$$

Redundancy

Fraction of "excess letters"

Entropy of the language

$$R_L = 1 - \frac{H_L}{\log_2 |P|}$$

Entropy of the random language

For English Language, $1 \leq H_L \leq 1.5$. Considering $H_L = 1.25$, and $|P| = 26$, $R_L \approx 0.75$.

English Language is 75% redundant.

A lower Bound of equivocation of key

- P^n: r.v representing n-gram plaintext
- C^n: r.v representing n-gram ciphertext
- $H(K|C^n)=H(K)+H(P^n)-H(C^n)$
  - $H(P^n)\approx nH_L$ (assuming large n)

    $=n(1-R_L)\log_2|P|$
  - $H(C^n)\leq n\log_2|C|$
- If $|P|=|C|$,
  - $\mathbf{H(K|C^n)\geq H(K)-nR_L\log_2|P|}$

Possible Keys

- Define, K(y)={possible keys given that y is the ciphertext}
  - that is K(y) is the set of those keys for which y is the ciphertext for meaningful plaintexts
- When y is the ciphertext, number of keys is $|K(y)|$
- Out of them, only one is correct. Rest are spurious.
- So, number of spurious keys=$|K(y)|-1$

Expected number of spurious keys

- Expected number of spurious keys=average number of spurious keys over all possible ciphertexts is denoted by s_n.

$$\boxed{\begin{aligned} s_n &= \sum_{y \in C^n} p(y)(|K(y)|-1) \\ &= (\sum_{y \in C^n} p(y)|K(y)|)-1 \end{aligned}}$$

Computing the upper bound of equivocation of key

$$\begin{aligned} H(K|C^n) &= \sum_{y \in C^n} p(y)H(K|y) \\ &\leq \sum_{y \in C^n} p(y)H(K(y)) \\ &\leq \sum_{y \in C^n} p(y)\log_2(|K(y)|) \\ &\leq \log_2(\sum_{y \in C^n} p(y)|K(y)|) = \log_2(s_n+1) \end{aligned}$$

Lower Bound of spurious keys

- Combining the previous results:

$$H(K) - nR_L \log_2 |P| \le \log_2(s_n + 1)$$
$$\therefore \log_2(s_n + 1) \ge H(K) - nR_L \log_2 |P|$$

- If the keys are chosen equi-probably:
  $H(K)=\log_2|K|$. Hence, we have:

$$s_n \ge \frac{|K|}{|P|^{nR_L}} - 1$$

Unicity Distance

- Thus increasing n, reduces the number of spurious keys.
- **Unicity Distance** is the number of ciphertexts, n_0 for which the number of spurious keys is reduced to zero.

$$n \ge n_0 = \frac{\log_2 |K|}{R_L \log_2 |P|}$$

This calculation may not be accurate for small values of n

Unicity Distance for Substitution Ciphers

- $|P|=26$
- $|K|=26! \approx 4 \times 10^{26}$, $R_L=0.75$
- $n_0=25$ (approx)
- Given a ciphertext string of length 25, it is possible to predict the correct key uniquely
  - Thus key size alone does not guarantee security, if brute force is possible to an attacker with infinite computational power.

Idea of Product Ciphers

- Another innovation introduced by Shannon in 1949 was the idea of forming “product”
- The idea is of fundamental importance and is used even for the present day standard, Advanced Encryption Standard.

Endomorphic Ciphers

- If P=C, then we have an endomorphic cipher.
- Thus the shift cipher on English alphabets is an endomorphic cipher.

What we have learnt from history?

- **Observation:** If we have an endomorphic cipher C_1=(P,P,K1,e1,d1) and a cipher C_2 (P,P,K2,e2,d2).
- We define the product cipher as $C_1 x C_2$ by the process of first applying C_1 and then C_2
- Thus $C_1 x C_2$=(P,P,K1xK2,e,d)
- Any key is of the form: (k1,k2)

and $e=e_2(e_1(x,k1),k2)$. Likewise d is defined.

Note that the product rule is always associative

Question:

- Thus if we compute product of ciphers, does the cipher become stronger?
  - The key space become larger
  - 2nd Thought: Does it really become larger.
- Let us consider the product of a
  1. multiplicative cipher (M): y=ax, where a is co-prime to 26 //Plain Texts are characters
  2. shift cipher (S) : y=x + k

Is MxS=SxM?

- MxS: y=ax+k : key=(a,k). This is an affine cipher, as total size of key space is 312.
- SxM: y=a(x+k)=ax+ak
  - Now, since gcd(a,26)=1, this is also an affine cipher.
  - key = (a,ak)
  - As gcd(a,26)=1, a^{-1} exists. There is a one-one relation between ak and k. Thus the total size of the key space in SxM is still 312. Thus this is also the affine cipher
- Thus S and M are commutative.

Idempotent Cipher

- M is a permutation cipher.
- S is a substitution cipher.
- Composed cipher has a larger key but no extra security.
- If we had computed MxM or SxS, would that have lead to the increase of key space? No.
  - This is because SxS=S and MxM=M
  - These are called idempotent ciphers

Inference

- Thus there is no point of obtaining products of idempotent functions.
- Rather we would get "product ciphers" from non-idempotent ciphers
  - That is by iterating them (rounds)
- How to make non-idempotent functions?
  - Compose two small different cryptosystems which do not commute

Why?

- If there are two cryptosystems which are idempotent and also commute then their product is also idempotent.
- $(S_1 x S_2) \times (S_1 x S_2) = S_1 x (S_2 \times S_1) x S_2$

$= S_1 x (S_1 x S_2) x S_2$

$= (S_1 x S_1) \times (S_2 x S_2)$

$= S_1 x S_2$

Thus, MxS is also idempotent. Why?

Thus, composing MxS does not help.

Concept of Rounds

- Consider : S=f(x) and P=x+k
- What is SxP? f(x)+k
- What is (SxP)x(SxP)? f(f(x)+k)+k
  - For this multiplication to increase the key length, thus SxP should not be idempotent.
  - that is $f(f(x)+k)+k \neq f^2(x)+k'$
  - This happens if f is non-linear wrt. +
  - **Hence we compose linear and non-linear functions to increase the security of a cipher**

Assignment

- Show that the unicity distance of the Hill Cipher (with an m x m encryption matrix) is less than m/R_L.

Further Reading

- C. E. Shannon, *Communication Theory of Secrecy Systems.* Bell Systems Technical Journal, 28(1949), 656-715
- Douglas Stinson, *Cryptography Theory and Practice, 2nd Edition*, Chapman & Hall/CRC

Next Day's Topic

- Symmetric Key Ciphers:
  - Block Ciphers
  - Stream Ciphers