Information Theory
3CC: Codes and Ciphers (530.334) Tim Penttila
September 8, 2005

Lecture : 31st August, 2005

1 Introduction

We have been using terms, such as redundancy and information, in the preceding lectures on codes and ciphers in an intuitive fashion. Here we will face the questions

What is redundancy?

What is information?

Redundancy has been our friend in coding theory, wrapping our precious message in bubble-wrap in order that it be received intact; but our enemy in cryptology, betraying our secrets to the spying Oscar. The following email joke, circulating about a year ago, illustrates redundancy beautifully.

Aocdndrig to rscheearch at an Elingsh uinervtisy, it deosn’t mttaer in waht oredr the ltteers in a wrod are, the only iprmoent tihng is taht the frist and lsat ltteer is at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit a porbelm. Tihs is bcuseae we do not raed ervey lteter by it slef but the wrod as a wlohe.

Redundancy makes crossword puzzles possible. Redundancy allows SMS text messages to be shorter. The first name for this abbreviated style was “telegraphese”.

What are our motivations here? Two questions, one from cryptology and the other from coding theory are to the point:

How can we prove the one-time pad mathematically secure?

Is there an upper bound to the rate of a code that we can achieve, even with any amount of error-correcting capacity desired, by varying the length of the code?


## 2 History

Interest in the concept of information grew directly from the invention of the telegraph and the telephone. In 1844 Samuel Morse built a telegraph line from Washington, D.C., to Baltimore, Maryland. He encountered many electrical problems when he sent signals through buried transmission lines, but inexplicably fewer when the lines were suspended from poles. This attracted the attention of the Scottish physicist Lord Kelvin (real name William Thomson). (A digression here about the Marquis de L’Hôpital (real name Guillaume François Antoine) not discovering the rule on limits that bears his name, but buying it from Johann Bernoulli, and signing him to a non-disclosure agreement for 20 years occurred.) Similarly, the invention of the telephone by Alexander Graham Bell in 1875 attracted attention to the problems associated with transmitting signals over wires from the French mathematician Henri Poincare and the English physicist and engineer Oliver Heaviside (the man who is reputed to have said: “Some sequences are absolutely convergent. These are absolutely useless. Some sequences are conditionally convergent. These are sometimes useful. This series is divergent, therefore we may be able to do something with it.”) Much of their work used Fourier analysis.

The formal study of information theory was initiated with the publication in 1924 of “Certain Factors Affecting Telegraphy Speed” by Harry Nyquist, a researcher at Bell Laboratories. Nyquist realized that communication channels had maximum transmission rates, and he derived a formula for calculating these rates in finite bandwidth noiseless channels

\[ W = k \log(m) \]

(where \( W \) is the speed of transmission (i.e., the number of characters transmitted in a given length of time), \( m \) is the number of current values that can be transmitted, and \( k \) is a constant).

Another pioneer was Nyquist’s colleague at Bell Labs, R.V.L. Hartley, whose paper “Transmission of Information” (1928) established the first mathematical foundations for information theory. He stressed that the capacity of a system to transmit information depended solely on distinguishing at the receiving end between the various selections made at the sending end - not on the meanings of these sequences. Information was thus demarcated from meaning. He used the definition of information as the number of possible messages to derive a logarithmic law for information transmission

\[ H = K \log(q^n) \]

(where \( H \) is the amount of information, \( n \) is the number of symbols in the message, \( q \) is the size of the alphabet and \( K \) is a constant).

Thus Hartley had established key concepts of a mathematical theory of communication: the difference between information and meaning, information as physical quantity, the logarithmic rule of information transmission. He also formulated the concept of noise as an impediment to information transmission.
Now we come to the founder of modern information theory, Claude Shannon. He was an extraordinarily promising MIT Ph. D. in mathematics who joined Bell Labs in 1941. He’d worked on communication systems and was appointed to some of the committees studying cryptanalytic techniques. He developed his parallel theories of mathematical communication and of cryptology from 1941 on, but published them as separate papers in 1948 and 1949. “Communication Theory of Secrecy Systems” (1949) treated cryptology in information-theoretic terms. Introducing such concepts as mathematical redundancy and binary coding into communication, it furnished tools to both information theory and cryptanalysis. “The Mathematical Theory of Communication” (1948) (available for free online at acm.org, follow the link from scholar.google.com) linked these new features with older concepts of information transmission to form a general theory applicable to any system in which information can be quantified and transmitted.

A key step in Shannon’s work was his realization that, in order to have a theory, communication signals must be treated in isolation from the meaning of the message they transmit. “All bits are equal.” This view is in sharp contrast from the popular conception of information, in which meaning has an essential role. (If I change a bit referring to a pixel in the top right hand corner of a newspaper photograph, turning the colour from dark grey to a lighter grey, it matters not a jot to anyone. But if I change the leading bit of your account balance at your bank from a one to a zero, then it matters a lot to you!) Shannon also realized that the amount of knowledge conveyed by the signal is not directly related to the size of the message. A famous illustration of this distinction is the correspondence between French novelist Victor Hugo and his publisher following publication of *LesMiserables* in 1862. Hugo sent his publisher a card with just the symbol “?”. In return he received a card with just the symbol “!” But much meaning was conveyed! Similarly, a long message in perfect French would convey little useful knowledge to someone who could understand only English.

The practical stimuli for Shannon’s work at Bell Labs were the problems faced in creating a reliable telephone system, such as how to transmit the maximum number of telephone conversations over existing cables. Shannon’s work defined communication channels and showed how to assign a capacity to them.

**Counting bits : the scientific measure of information**

In contrast to the vague verbal definition of information (as the communication of relationships), the technical definition, though skeletal, is a model of specificity and succinctness. Shannon produced a way to measure the amount of information without defining the word information itself, producing, in effect, an operational definition like that of temperature, except that his measuring device - a simple recipe - is not a physical apparatus, like a thermometer, but a conceptual tool. (Temperature began in science around 1600 as the quantity measured by a thermometer, and was only understood as a measure of the average speed of particles in the middle of the nineteenth century.)
Shannon’s recipe is simple: *To find the information content of any message, translate the message into binary and count the number of bits.*

Shannon also introduced the term “bit” in the 1948 paper, credited the suggestion to the statistician J. W. Tukey, also at Bell Labs. (Perhaps childishly, I sometimes wonder if the suggestion was motivated by the corresponding contraction for ternary digit.) *Here a bit is the unit of information for the first time ever.*

Information theory is a difficult subject. It needs to be approached with care and slowly. One of its central concepts is **entropy**, in analogy with the physical concept from thermodynamics and statistical mechanics, and a statistical idea itself. Entropy is notorious in physics as one of the hardest concepts to teach. It is seen as a measure of disorder in physics; its negative as a measure of order. Similarly, in information theory, it is a measure of average information. (It should not surprise that information is correlated with order.)

So we will need a number of easy paths into information theory - shallow slopes we will struggle up to point at the mountain peak above, only to descend and try another path to the summit. The first of these we shall attempt lies in the area of **data compression**. (This is also called **source encoding**.) There are four basic situations in communication theory, accordingly as the signals are digital or analog, and accordingly as the channel is noiseless or has noise. In this unit, we will only be concerned with the digital (discrete) - we leave the analog (continuous) to other times and places. So concepts such as bandwidth and signal-to-noise ratio are not for us here. We are interested in noiseless digital source encoding, at least at first.

For an immediate illustration of source encoding, let’s return to Sam Morse and the telegraph and his Morse code. Morse confronted a problem that is at the root of modern information theory: what is the most efficient way to choose symbols, composed of dots, dashes and spaces, for the letters of the alphabet? The ingenious way that he tacked the problem compared with Shannon’s approach illustrate in microcosm the difference between engineering and science, contrasting Yankee resourcefulness and the power of mathematics.

The principle is obvious: the most efficient code assigns short strings to the common letters and long strings to the rare ones. But what is common and what is rare? With access to frequency analysis tables restricted, Morse walked into a newspaper office and counted the number of letters in each compartment of the compositor’s type box. Presumably decades of experience had reduced its contents to an efficient compromise between supply and demand. In Morse code, E is represented by a single dot and T by a single dash; in contrast, X, Y and Z drew four symbols each. It has since been shown that Morse’s approach achieved 85% of the possible maximum efficiency of any such scheme, which is an extremely good outcome.

So the fundamental problem in data compression is the construction of codes that encode a source with the smallest possible average codeword length.

That ends this lecture, where the signal-to-noise ratio was low, with not enough formulae and too many anecdotes!
3 Source encoding: discrete, noiseless communication

The theory of probability became a serious science in the seventeenth century, but its roots are buried in the ancient games of dice and cards. Long before mathematicians refined their tools sufficiently to tackle the subject, gamblers knew from experience how to use partial information as a basis for estimating the odds on their bets. In this way they intuitively assigned quantitative values to the information they had, or lacked, about whatever they were betting on. (See the bonus question on assignment 3.)

Indeed the origins of probability lie in gambling. A gambler’s dispute in 1654 led to the creation of a mathematical theory of probability by two famous French mathematicians, Blaise Pascal and Pierre de Fermat. Antoine Gombaud, Chevalier de Mere, a French nobleman with an interest in gaming and gambling questions, called Pascal’s attention to an apparent contradiction concerning a popular dice game. The game consisted in throwing a pair of dice 24 times; the problem was to decide whether or not to bet even money on the occurrence of at least one “double six” during the 24 throws. A seemingly well-established gambling rule led de Mere to believe that betting on a double six in 24 throws would be profitable, but his own calculations indicated just the opposite. A game of dice interrupted by a fight in a tavern led to the problem of points: how to divide the stakes if a game of dice is incomplete?

These problems and others posed by de Mere led to an exchange of letters between Pascal and Fermat in which the fundamental principles of probability theory were formulated for the first time. Although a few special problems on games of chance had been solved by some Italian mathematicians (Cardan, Pacioli and Tartaglia) in the 15th and 16th centuries, no general theory was developed before this famous correspondence.

The Dutch scientist Christian Huygens, a teacher of Leibniz, learned of this correspondence and shortly thereafter (in 1657) published the first book on probability; entitled De Ratiociniis in Ludo Aleae, it was a treatise on problems associated with gambling. Because of the inherent appeal of games of chance, probability theory soon became popular, and the subject developed rapidly during the 18th century. The major contributors during this period were Jakob Bernoulli (1654-1705) and Abraham de Moivre (1667-1754).

In 1812 Pierre de Laplace (1749-1827) introduced a host of new ideas and mathematical techniques in his book, Theorie Analytique des Probabilites. Before Laplace, probability theory was solely concerned with developing a mathematical analysis of games of chance. Laplace applied probabilistic ideas to many scientific and practical problems. The theory of errors, actuarial mathematics, and statistical mechanics are examples of some of the important applications of probability theory developed in the 19th century.

A (memoryless) information source consists of a probability distribution $P$ on
an alphabet, called the message alphabet $M$.

Examples

Source $S_1$ (Uniform distribution)

$M = \{a, b, c, d\}$.

$P(a) = P(b) = P(c) = P(d) = 0.25$

Source $S_2$.

$M = \{a, b, c, d\}$.

$P(a) = 0.5, P(b) = 0.25, P(c) = 0.125$ and $P(d) = 0.125$.

An encoding scheme consists of a signal alphabet $S$, a code $C$ over $S$ and a function $f: M \rightarrow C$.

Examples

Encoding scheme A.

$f: a \rightarrow 00, b \rightarrow 01, c \rightarrow 10, d \rightarrow 11$

Encoding scheme B.

$f: a \rightarrow 0, b \rightarrow 10, c \rightarrow 110, d \rightarrow 111$

The average codeword length is

$$\sum_{x \in M} P(S = x) \text{length}(f(x))$$

Scheme A applied to $S_1$ gives an average codeword length of 2. Scheme B applied to $S_1$ gives an average codeword length of 2.25. So, for $S_1$, Scheme A is better than Scheme B - it compresses the data more and so leads to faster rates of transmission.

Scheme A applied to $S_2$ gives an average codeword length of 2. Scheme B applied to $S_2$ gives an average codeword length of 1.75. So, for $S_2$, Scheme B is better than Scheme A - it compresses the data more and so leads to faster rates of transmission.

Thus the choice of coding scheme depends on the statistical characteristics of the source. Later we will see that, amongst uniquely decodable schemes, Scheme A is optimal for $S_1$ and Scheme B is optimal for $S_2$. But we will need more theory for that.
4 Figuring the odds : how probability measures information

Let’s play a game. There are two player, each with a source. In each round, the players source emits a symbol. The winner is the player who first has the set of all symbols of the opponent’s source.

Specifically, Player 1 has the source $M = \{a, b\}$ with $P(a) = 0.99, P(b) = 0.01$ and Player 2 has the source $M = \{c, d\}$ with $P(c) = 0.99, P(d) = 0.01$. If in the first round Player 1’s source emits $b$ and Player 2’s source emits $c$, who is more likely to win? Since in the next round, the probability of Player 1’s source emitting $a$ is 0.99 and the Probability of Player 2’s source emitting $c$ is 0.99, and these events are independent, the probability of Player 2 winning in the next round is $0.99 \times 0.99 = 0.9801$. Thus Player 2 is far more likely to win than Player 1. What does this mean in terms of information? In some sense, Player 2 has received more information about Source 1 than Player 1 has about Source 2. Indeed, if Player 1 were a professional gambler with a rigged source that $M = \{a, b\}$ with $P(a) = 1, P(b) = 0$ then Player 2 could never learn anything about $b$: the source always emits $a$ and $b$ can never be discovered. So the less likely a source symbol is to occur, the more information we obtain from an occurrence of that symbol, and conversely. If $a$ is always emitted, then it conveys NO information at all.

Define $I(p)$ to be the information obtained from a source symbol with probability of occurrence $p$. (It only depends on the probability, not on the symbol.) What can we say about $I(p)$?

(I1)........................ $I(p) \geq 0$

Information can not be negative in amount; although it can be zero.

If we vary the original game slightly, changing $P(a) = 0.99$ to $P(a) = 0.98999999$, what would we expect to happen? A slight change in the probability should lead to a slight change in the amount of information conveyed. There should be no sudden jumps. Mathematically speaking,

(I2)... $I(p)$ is a continuous function of $p$.

Suppose now that our gambler likes the daily double. So he plays two games at once, with different sources, and bets with a spectator that he will win both. How does the information he receives from the two independent games accrete? Whatever the units of information are, if he has $m$ units of information about his opponent’s source in the first game and $n$ units
of information about his opponents source in the second game, then he must have \( a + b \) units of information in total. Information adds. If you get 9 units of information from the newspaper and 2 from a (difficult) book, then you have 11 units of information. On the other hand, probabilities multiply (for independent events).

\[
P(X = a \& Y = b) = P(X = a)P(Y = b)
\]

for independent random variables \( X \) and \( Y \). Thus we are saying that

\[(I3)...................I(p_1p_2) = I(p_1) + I(p_2),\]

for all probabilities \( p_1 \) and \( p_2 \).

5th September, 2005 (New lecture)

In summary, whatever information is, it only depends on the (non-zero) probability of occurrence of an event. Thus it gives rise to a function \( I \) with domain \((0, 1]\) satisfying properties \((I1),(I2)\) and \((I3)\). We can add the extra condition that \( I \) is not the constant function with value 0: Information exists.

Let’s delineate further the fact that information adds. Suppose you are simultaneously, by e-correspondence, playing two games - one of dice and the other involving coin-tossing, and that the games are independent. In game 1, so far the coin has come up \( H, T, H, H \). In game 2, so far the octahedral die (with faces numbered 0 to 7) has rolled 1, 7. To record the outcomes in binary with \( H \) as 0 and \( T \) as 1, the record of the first game is 0100; while with the 3-bit binary expansion of the number on a face as a record, the second game reads 001111. The 4 bits of information in game 1 so far can be combined with the 6 bits of information from game 2 to give 4 + 6 = 10 bits in memory. Information adds.

What function \( f \) has the property \( f(xy) = f(x) + f(y) \)?

The characteristic property of logarithms is that they convert products into sums.

Does the base matter?

No.

Does \( \log_q \) have all the properties we want?

No, because logarithms of numbers less than 1 (such as probabilities) are negative.

What about \(-\log_q\)?
We could try $I(p) = -\log_q(p)$ - it’s non-negative, a continuous function of $p$ and converts products into sums.

Before we leap off into the unknown, was there any choice? Apart from the choice of base of the logarithms, is there any freedom in assigning an information function?

Theorem The only functions (other than the constant function with value 0) with domain $(0,1]$ satisfying (I1), (I2) and (I3) are the functions $-\log_q$, for real number $q > 1$.

Proof: Suppose the function $I$ satisfies (I1), (I2) and (I3). Then $I(x^2) = I(x.x) = I(x) + I(x) = 2I(x)$.

Generalising, by mathematical induction $I(x^n) = nI(x)$, for all positive integers $n$. Replacing $x$ by $x^{1/n}$, we have $I(x^{1/n}) = I(x)/n$. Hence $I(x^{n/m}) = (n/m)I(x)$. Thus $I(x^r) = rI(x)$ for all positive rational numbers $r$. Any positive real number $t$ is the limit of a sequence $r_n$ of positive rational numbers: $t = \lim(r_n)$. Since $I$ is continuous,

$$I(x^t) = I(x^{\lim(r_n)}) = \lim(I(x^{r_n})) = \lim(r_nI(x)) = I(x)\lim(r_n) = I(x)t$$

Suppose $I$ is not the constant function with value 0. Then there exists $x_0$ with $I(x_0) > 0$. Let $t = 1/I(x_0)$ and $q = x_0^{-t}$. Then $I(q^{-1}) = I(x_0^t) = tI(x_0) = 1$. Now

$I(x) = I((q^{-1})^{-\log_q(x)}) = -\log_q(x)I(q^{-1}) = -\log_q(x)$, for all $x$ with $0 < x < 1$.

Finally, continuity of $I$ forces $I(1) = 0$. Thus $I = -\log_q$. Moreover, that $I(x)$ is non-negative for $x \in (0,1]$ forces $q > 1$.

Remark: The above theorem is a result of topological group theory, equivalent to the statement that the only continuous automorphisms of the topological group $(0,1]$ of real numbers under multiplication are the power maps $x \to x^q$, for $q \in (0, \infty)$.

Usually we choose base 2 for the logarithm and measure information in bits. It seems as though the bit is the smallest quantity of information - the answer to a yes/no question - so this motivates the choice of unit. (However, with source encoding with a signal alphabet of size $r$, often base $r$ is used. If natural logs are used, the units are called nats. If base 10 logs are used, the units are called Hartleys.)

Examples 1. $I(1/2) = 1$ bit.

2. $I(1/4) = 2$ bits.
We must emphasise that we are measuring the *a posteriori* information; that gained only after the event of a source emission. The higher the *a priori* information (that before the event), the less learned from the event. If we know *a priori* that the probability of the emission of *a* is 1, then we learn nothing from the emission of *a*. It is possible, even useful, to think of *a posteriori* information as *a priori* uncertainty.
5 Entropy

The average information obtained from a single sample of a source $S$ with $q$ message symbols of probabilities $p_1, \ldots, p_q$ is

$$H(S) = \sum_{i=1}^{q} p_i I(p_i) = - \sum_{i=1}^{q} p_i \log_2(p_i)$$

$H(S)$ is called the entropy of the source [11]. (Since $\log_2(0)$ is undefined, we interpret $p \log_2(p)$ as 0 if $p = 0$.)

Examples 1. For source 1 above, $H(S_1) = 2$ bits.

2. For source 2 above, $H(S_2) = 1.75$ bits.

Again, entropy is a measure of average a posteriori information, or of average a priori uncertainty.

Is it a coincidence that the entropy for these sources was the average coding length for encoding scheme A applied to source 1 and encoding scheme B applied to source 2? What does entropy have to do with average codeword lengths?

7th September, 2005 (New lecture)

Lemma Let $\{p_1, \ldots, p_q\}$ be a probability distribution. Let $\{r_1, \ldots, r_q\}$ have the property that $0 \leq r_i \leq 1$ for all $i$, and

$$\sum_{i=1}^{q} r_i \leq 1$$

(Note the inequality here.) Then

$$- \sum_{i=1}^{q} p_i \log_2(p_i) \leq - \sum_{i=1}^{q} p_i \log_2(r_i)$$

with equality holding if and only if $p_i = r_i$ for all $i$.

Theorem For a source $S$ on a message alphabet of size $q$, the entropy $H(S)$ satisfies

$$0 \leq H(S) \leq \log_2(q)$$

Furthermore, $H(S) = \log_2(q)$ if and only if all of the source symbols are equally likely to occur and $H(S) = 0$ if and only if one of the source symbols has probability 1 of occurring.

Proof: Apply the Lemma with $R = \{1/q, 1/q, \ldots, 1/q\}$.

This theorem confirms that, on the average, the most information is obtained from sources for which each source symbol is equally likely to occur.
6 Uniquely decipherable codes

A code $C$ over an alphabet $A$ is uniquely decipherable if, for every string $x_1x_2\ldots x_n$ over $A$, there is at most one sequence of codewords $c_1c_2\ldots c_m$ with

$$c_1c_2\ldots c_m = x_1x_2\ldots x_n$$

In other words, no two sequences of codewords represent the same string.

Examples

A. The code of encoding scheme A above is uniquely decipherable. Indeed, all block codes are uniquely decipherable.

B. The code of encoding scheme B above is uniquely decipherable. Whenever you hit a zero, that’s the end of a message symbol. If you’ve gone three places without a zero that’s the message symbol $d$.

C. $f: a \rightarrow 0, b \rightarrow 00, c \rightarrow 000, d \rightarrow 1$ is not uniquely decipherable, because 000 could be both $ab$ and $c$.

Duplicate encodings lead to ambiguity in messages, so unique decipherability is a desirable feature on an encoding scheme.

If a code is uniquely decipherable, then it cannot have very many short codewords. In fact,

Theorem (McMillan 1956 [8]) Let $S = \{0, 1\}$ and $f: M \rightarrow C$ be a uniquely decipherable coding scheme, where $C$ is a binary code. Then

$$\sum_{c \in C} \frac{1}{2^{\text{length}(c)}} \leq 1$$

Proof: An exercise in nasty algebra - omitted.

Theorem (Shannon’s Noiseless Coding Theorem 1948 [11]) The entropy of a source $S$ is a lower bound for the average code length for any uniquely decipherable binary coding scheme for $S$.

Proof: Let $M = \{a_1, \ldots a_q\}$, $p_i = P(a_i)$ and $C$ be the code used in the uniquely decipherable binary coding scheme, with encoding function $f: M \rightarrow C$. Apply the lemma with $R = \{\frac{1}{2^{\text{length}(c)}} : c \in C\}$. By McMillan’s Theorem, the inequality of the lemma is satisfied, so the lemma gives

$$H(S) = -\sum_{i=1}^q p_i \log_2(p_i) \leq \sum_{i=1}^q p_i \log_2(2^{\text{length}(f(a_i))}) = \sum_{i=1}^q p_i \text{length}(f(a_i))$$

which is the average code length for the encoding scheme.
So it was no accident that the entropy for these sources was the average coding length for encoding scheme A applied to source 1 and encoding scheme B applied to source 2 - rather that is equivalent to the statement that these are the best possible uniquely decoding binary coding schemes for these sources. They show that, for these sources, the bound is sharp.

By encoding blocks of source symbols rather than source symbols, we can reduce the average coding length per source symbol to as close to the entropy of the source as we desire. (A semi-formal statement of this is:

\[ N \] independent identically distributed sources each with entropy \( H \) can be compressed into \( NH \) bits with negligible loss of information; moreover, if they are compressed into fewer than \( NH \) bits there is a dramatic falloff of information.) So it is impossible to improve this bound. Thus the entropy can be used to determine the maximum possible (lossless) compression.

It’s a good idea to pause here and think. Entropy is our attempt at measuring the average information of a source. Lossless data compression was another measure: how far you can compress data on average without loss of information is a good measure of how much information on average is in the data. Shannon’s Noiseless Coding Theorem says that these two attempts at measuring average information content agree.

Shannon’s theorem is not constructive. The next section looks at algorithms for achieving lossless data compression.

7 Instantaneous codes

Example The code \( C = \{c_1 = 0, c_2 = 01\} \) is uniquely decipherable. However, if we transmit 0001 then we can’t decode on the fly - the first zero could be \( c_1 \) or the first symbol of \( c_2 \), and, until we get the second symbol we’re not sure (after which we know it was \( c_1 \)). Then the second symbol could be \( c_1 \) or the first symbol of \( c_2 \), and, until we get the third symbol we’re not sure (after which we know it was \( c_1 \)). Then the third symbol could be \( c_1 \) or the first symbol of \( c_2 \), and, until we get the fourth symbol we’re not sure (after which we know it was \( c_2 \)).

On the other hand, for the code \( D = \{d_1 = 0, d_2 = 10\} \), individual codewords can be interpreted as soon as they are received. For instance, consider the string 00100. As soon as the first 0 is received, we know immediately that it must be \( d_1 \), and similarly for the second 0. When the 1 is received, we know that a 0 is coming next, and that the string 10 must be \( d_2 \). Thus the message can be interpreted codeword by codeword.

A code is instantaneous if, whenever any sequence of codewords is transmitted, each codeword can be interpreted as soon as it is received.

A code has the prefix property if no codeword is a prefix of any other codeword.
Example $C$ does not have the prefix property, as $c_1$ is a prefix of $c_2$. $D$ has the prefix property.

Theorem A code is instantaneous if and only if it has the prefix property.

Proof: If $C$ has the prefix property, then the moment a codeword is received it can be decoded without ambiguity, as it is not the prefix of any other codeword. Conversely, if $C$ is instantaneous and a codeword $c$ is a prefix of a codeword $d$, then the first $c$ in the message $cc$ could not be instantaneously decoded, as the message could be $d$. This contradiction completes the proof.

Theorem (Kraft 1949) If a uniquely decipherable code exists with codeword lengths $l_1, l_2, \ldots, l_n$, then an instantaneous code must also exist with these same lengths.

This says that instantaneous codes cost no more than uniquely deciph erable codes - so we may as well shoot for them.

In 1952, D.A. Huffman [5] published a method for constructing highly efficient instantaneous encoding schemes. This method is now known as Huffman encoding. Huffman encoding is based on the use of binary trees. First we convert the source into a binary tree with leaves the letters of the alphabet in a particular way.

The first step is to represent each character by a trivial binary tree, with the character as the only node, and to place them in increasing order of the probabilities.

At each subsequent step, the two leftmost trees are joined as the subtrees of a new root that is assigned the sum of their frequencies.

This is repeated until all characters are in one tree.

Finally, discard all the probabilities and label each edge that slopes up from left to right with a 0 and each edge that slopes down from left to right with a 1. To determine the codeword associated to each source symbol, start at the root and write down the sequence of bits encountered en route to the source symbol.

One code bit represents each level. Thus more frequent characters are near the root and are coded with few bits, and rare characters are far from the root and are coded with many bits.
Example $M = \{a, b, c, d, 1, 2\}$ with $P(a) = 0.35, P(b) = 0.10, P(c) = 0.19, P(d) = 0.25, P(1) = 0.06, P(2) = 0.05$. See Roman, pp. 58-61. The average codeword length of the resulting Huffman encoding is 2.32. The entropy of this source $S$ is $H(S) = 2.277$ (approx.). Was it an accident that the Huffman encoding got so close to the lower bound given by the entropy? No.

All Huffman encoding schemes have the prefix property, since each symbol is a leaf and so cannot be a prefix of another symbol. Thus they are instantaneous.

Theorem [5] Huffman encoding schemes have the smallest average codeword length over all uniquely decipherable encoding schemes.

Huffman encoding schemes are state-of-the-art for data compression where the statistical characteristics of the source are known. Dynamic encoding schemes, such as that of Lempel and Ziv are used when such characteristics are not available initially. The adaptation by Welch of Lempel-Ziv led to use in .gif and .zip formats.

8 Redundancy

Now we return to the concept of redundancy. For a given alphabet (with $q$ symbols), the maximum entropy is obtained from a uniform probability distribution and equals $\log_2(q)$. The ratio of the entropy of a source (on the same alphabet) to $\log_2(q)$ is thus a number in the range from 0 to 1. If the ratio is near 1 then the source is close to random and the information per letter is low, thus many letters are needed to pass a certain amount of information. If the ratio is near 0 then the information content per letter is high and the same amount of information is passed with fewer letters. If we compare messages of the same length determined by two sources then in the first case the information is spread out over the message and the redundancy is low, while in the second case a small portion of the message has the information and the total message contains much redundancy. Thus, it makes sense to define the redundancy of a source $S$ by [11]

$$R(S) = 1 - \frac{H(S)}{\log_2(q)}$$

One of Shannon’s estimates for entropy in English (from 1951) was 2.62 bits per letter. This would put the redundancy of English at about 45%. However, other estimates (e.g., those referred to in [14], pp.61-2) have the entropy of English at about 1.25 bits per letter, which would put the redundancy of English at about 75%. Of course, a source emitting English words is not a memoryless source, e.g., ‘q’ is usually followed by ‘u’ (which increases the redundancy). So more complex probability distributions are required. This involves distributions on bigrams, trigrams, . . . , $n$-grams of
letters. In this setting, the \( n \)th approximation of the rate of the language in \( H(M_n)/n \), where \( M_n \) is the distribution on \( n \)-grams; the rate is the entropy of the message source per symbol (measured in bits per letter). The term is related to the term rate in coding theory.

9 Further properties of entropy

Given sources \( S \) and \( T \), it is possible to measure the degree of uncertainty or information associated with them. It is called the joint entropy \( H(S,T) \). If \( S \) and \( T \) have alphabets of size \( q \) and \( r \) respectively, the joint entropy is computed as in the equation below:

\[
H(S,T) = -\sum_{x=1}^{q} \sum_{y=1}^{r} p_{ij}(x,y) \log_2 p_{ij}(x,y)
\]

where \( p_{ij} \) represents the probability of both symbol \( i \) being emitted by source \( S \) and symbol \( j \) being emitted by source \( T \).

The joint entropy varies from 0 to \( \log_2(q) + \log_2(r) \). The relation between the individual entropies and their joint entropy is given by the next result.

Theorem \( H(S,T) \leq H(S) + H(T) \) with equality holds only when the two sources are independent.

The conditional entropy is

\[
H(S|T) = -\sum_{t} \sum_{s} P(T = t)P(S = s|T = t) \log_2 P(S = s|T = t)
\]

(Shannon called this equivocation.) The conditional entropy measures the average amount of information about \( S \) that is revealed by \( T \). This is why it is of importance in cryptology.

Theorem \( H(S,T) = H(T) + H(S|T) \).

Theorem \( H(S|T) \leq H(S) \), with equality if and only if \( S \) and \( T \) are independent.
10 Application of information theory to cryptography and coding theory

Both of our communication models are improved by preceding transmission with source encoding (and finishing with source decoding). For cryptography, we have taken out the redundancy. Thus Oscar has no information to gain through frequency analysis. So provided we have a lot of entropy in the key (a lot of possible keys), this is an improvement. For coding theory, the rate of transmission will be better if we compress first. This removes redundancy that serves no engineering purpose, before we add redundancy serving an engineering purpose: error-correction.

However, there are theoretical applications as well. The first of these is the perfect secrecy of the one-time pad. The second is Shannon’s noisy coding theorem. The third is an explanation of our definition of the rate of a code.

11 Perfect secrecy

A cryptosystem has perfect secrecy if

\[ P(\text{plaintext} = x|\text{ciphertext} = y) = P(\text{plaintext} = x) \]

for all plaintext \( x \) and all ciphertext \( y \). This means that knowledge of the ciphertext gives no information about the plaintext.

For perfect secrecy to occur, the number of keys must be at least as great as the number of possible ciphertexts. Every piece \( y \) of ciphertext has positive probability, as otherwise we could delete it from the list of possible ciphertexts. But, by Bayes’ Theorem,

\[ P(\text{plaintext} = x|\text{ciphertext} = y)P(\text{ciphertext} = y) = P(\text{ciphertext} = y|\text{plaintext} = x)P(\text{plaintext} = x) \]

Since we have perfect secrecy, it follows that

\[ P(\text{plaintext} = x)P(\text{ciphertext} = y) = P(\text{ciphertext} = y|\text{plaintext} = x)P(\text{plaintext} = x) \]

giving

\[ 0 < P(\text{ciphertext} = y) = P(\text{ciphertext} = y|\text{plaintext} = x) \]

Hence for any ciphertext \( y \) at least one key must encrypt \( x \) as \( y \). Thus, the number of keys must be at least as great as the number of possible
ciphertexts. Equality occurs if and only if, for every plaintext $x$ and every ciphertext $y$ there must be exactly one key that encrypts $x$ as $y$ and every key is used with equal probability, namely, the reciprocal of the number of keys. Hence

Theorem (Shannon (1949) [12]) The one-time pad has perfect secrecy.

Proof: The number of keys in the one-time pad is the number of possible messages, and every key is used with equal probability.

This can be generalised: if the entropy $H(K)$ of the key $K$ is much larger than the redundancy of $n$-grams from the message source $M$ and a message of length $n$ is sent, then there is a high probability of obtaining a meaningful decipherment, and, therefore, a low likelihood of determining the correct message. Since English has a high redundancy, most decipherments lead to meaningless messages. This is what allows cryptanalysis - the bad hypotheses on the key are rejected as they lead to meaningless messages. This is an explanation of why source coding before enciphering can improve security by removing redundancy.

The redundancy of the English language is high enough that often the amount of information conveyed by every ciphertext component is greater than the rate at which equivocation (i.e., the uncertainty about the plaintext that the cryptanalyst must resolve to crack the cipher) is introduced by the key. The amount of uncertainty we introduce into cryptanalysis of a message cannot be greater than the uncertainty in the key.

12 Capacity of a channel

A channel is memoryless if the outcome of any one transmission is independent of the outcome of the previous transmissions. A discrete memoryless channel is composed of the following three components:

1. The input to the channel, a source $S$ on the message alphabet $M = \{m_1, \ldots, m_r\}$ with probabilities $p_1, \ldots, p_q$.

2. The output to the channel, a source $T$ on the signal alphabet $A = \{a_1, \ldots, a_q\}$.

3. A transition matrix $Q = [P(T = a_j | S = m_i)]$.

A drawing goes here.

If $M = A = \{0, 1\}$, the channel is binary.
A binary channel is symmetric if $Q$ is symmetric, i.e., $Q = Q^T$. In that case,

$$Q = \begin{pmatrix} 1 - p & p \\ p & 1 - p \end{pmatrix}$$

$p$ is called the crossover probability. It is the probability of a bit error. We have

$$p = P(1\text{received}|0\text{sent}) = P(0\text{received}|1\text{sent}).$$

Consider a binary symmetric channel with crossover probability $p$. Suppose that the probability distribution on the message source $S$ is $P(S = 0) = p_1$ and $P(S = 1) = 1 - p_1$. Then if $T$ is the output of the channel, it has probability distribution given by:

$$P(T = 0) = P(T = 0|S = 0)P(S = 0) + P(T = 0|S = 1)P(S = 1) = (1 - p)p_1 + p(1 - p_1) = q_1, \text{ say and}$$

$$P(T = 1) = 1 - q_1.$$ 

Now the entropy of the channel output is $H(T) = -q_1\log_2(q_1) - (1 - q_1)\log_2(1 - q_1)$.

Moreover, the conditional entropy

$$H(T|S) = p_1H(T|S = 0) + (1 - p_1)H(T|S = 1) = p_1[-(1 - p)\log_2(1 - p) - p\log_2(p)] + (1 - p_1)[-p\log_2(p) - (1 - p)\log_2(1 - p)] = -p\log_2(p) - (1 - p)\log_2(1 - p)$$

is independent of the source probability distribution.

The mutual information between the input $S$ and output $T$ of a channel is

$$I(S; T) = H(S) - H(S|T) = H(T) - H(T|S).$$

Intuitively, this is the difference between the uncertainty that we have at the channel output when nothing is known, minus that which we have (on average) when we know the emitted symbol. It therefore is some measure of the quality of the channel. If $I(S; T) = 0$, then $H(T) = H(T|S)$, and so it doesn’t matter what symbol was transmitted - the channel is useless.

If $I(S; T) = H(S)$, then $H(S|T) = 0$, so $H(S, T) = H(T)$ which implies $H(T|S) = 0$ and the channel is error-free.

For the binary symmetric channel with crossover probability $p$,
\[ I(S;T) = H(T) - H(T|S) = \]
\[ -q_1 \log_2(q_1) - (1 - q_1) \log_2(1 - q_1) + p \log_2(p) + (1 - p) \log_2(1 - p). \]

The capacity of a channel is

\[ C = \max I(S;T), \]

where the maximum is taken over all source probability distributions. As \( C \) is a maximum over the input source probability distributions, it is a function of the parameters appearing in the channel transition matrix. For the binary symmetric channel, the capacity \( C(p) \) is a function of the crossover probability \( p \).

In the case of the binary symmetric channel, since \( H(T|S) \) does not depend on the input source distribution,

\( C(p) \) is maximised if and only if \( H(T) \) is maximised.

But entropy is maximised for the uniform distribution only, so \( H(T) \) is maximal when \( q_1 = \frac{1}{2} \), i.e, when

\[ p + p_1 - 2pp_1 = p_1(1 - p) + (1 - p_1)p = \frac{1}{2}, \]

which implies that

\[ p_1(1 - 2p) = \frac{1}{2} - p = \frac{1 - 2p}{2} \]

which implies that \( p_1 = \frac{1}{2} \). Then \( H(T) = 1 \). Thus

\[ C(p) = 1 + p \log_2(p) + (1 - p) \log_2(1 - p) \]

A graph of \( C(p) \) versus \( p \) goes here.

Introducing a new random variable, noise \( N \), with events a bit error occurs and a bit error doesn’t occur, so that

\[ P(N = \text{bit error}) = p, \text{ and } P(N = \text{no bit error}) = 1 - p, \]

we have

\[ H(N) = -p \log_2(p) - (1 - p) \log_2(1 - p). \]

Thus the capacity of the channel is

\[ C(p) = 1 - H(N). \]
In other words, noise decreases the capacity of a channel. Average information carried by the channel is reduced by the average information in noise. Of course, $0 \leq H(N) \leq 1$, with

$$H(N) = 0 \text{ if and only if } p = 0 \text{ (or } p = 1 \text{) if and only if the channel is perfectly noiseless (after rewiring if } p = 1 \text{) if and only if the capacity of the channel is 100\%,}$$

and

$$H(N) = 1 \text{ if and only if } p = \frac{1}{2} \text{ if and only if the channel is useless if and only if the capacity of the channel is 0\%.}$$

The capacity $C$ of a channel is an upper bound for the rate of error-free transmission. A decoding error is an instance where nearest neighbour/majority logic/maximum likelihood decoding results in an incorrect interpretation of the message.

**Theorem** (Shannon’s Noisy coding theorem (1948)[11])

If $R < C(p)$ then for any $\epsilon > 0$, there exists, for $n$ sufficiently large, an $(n, K)$-code $C$ with transmission rate at least $R$, and for which the probability of a decoding error $< \epsilon$.

While the rate of error-free transmission cannot exceed the channel capacity, the rate of nearly error-free transmission can approach as close as desired to the capacity.

So the capacity is the best possible upper bound on the rate of error-free transmission on the channel, and we can get arbitrarily close to this bound. Amazingly, the proof (an abstract existence proof) says that choosing the code at random (within appropriate guidelines) will achieve this optimal outcome!
13 The transmission rate of a code

There is a problem in attempting to describe the transmission rate of a code. The problem centres around the fact that there are two sets of symbols involved in transmitting information: the source symbols and the code symbols. For example, we could be talking about the source being English language symbols and the code being the ASCII code (three digit numbers from 033 to 122) or we could use a binary encoding scheme scheme using 7 binary digits (that will encode 128 different source symbols). So it makes sense to have both the source and code expressed in the same units. One way to do that is assume the units of the code and translate the source into those units.

Theorem The average rate of transmission of a $q$-ary $(n, K)$-code in which each codeword is equally likely to be transmitted is $\log_q(K)/n$.

Proof: The $K$ source symbols from the source $S$ can be compressed into $\log_q(K)$ bits : $H(S) = \log_q(K)$. Thus the fastest possible rate uses $\log_q(K)$ bits per source symbol. On the other hand, they are being transmitted using $n$ base $q$ digits : that is, $n\log_2(q)$ bits per source symbol. The ratio is the relative rate : the proportion of the fastest possible rate. It is

$$\log_q(K)/n\log_2(q) = \log_q(K)/n$$

Thus our definition of the rate of a code was a reasonable one.

14 Entropy and physics

Remarkably, Shannon’s formula for entropy turns up in textbooks on statistical mechanics published before Shannon’s papers (in the 1930’s). However, there it described not an informational-theoretic concept, but a physical concept. In the description of a number of electron spin states with possible values after measurement in the $z$ direction spin up/spin down, the statistical mechanical entropy of the state is identical with the information-theoretic entropy. So it would be a reasonable thing to do to talk a little about physical entropy.

The term entropy was originally introduced by the German physicist Rudolf Clausius (1822 - 1888) in his work on thermodynamics in the nineteenth century. Clausius invented the word so it would be as close as possible to the word energy. (In a period of 20 years in the middle of the nineteenth century, energy was invented, defined and established as
The laws of thermodynamics are

The first law Energy is conserved.

The second law In all energy exchanges, if no energy enters or leaves the system, the potential energy of the state will always be less than (or equal to) that of the initial state.

The third law The limiting value of the entropy of a system can be taken as zero as the absolute value of temperature approaches zero. It is impossible to cool a body to absolute zero by any finite process. Although one can approach absolute zero as closely as one desires, one cannot actually reach this limit.

The joke version of these laws (the Universe, the ultimate casino):

The first law You can’t win, you can only break even.

The second law You can only break even at absolute zero.

The third law You can’t reach absolute zero.

Entropy is defined as the amount of energy unavailable to do work (or as heat divided by temperature). The second law says that entropy always increases. (The limit over time being the heat death of the Universe - no energy available to do work.)

The Austrian physicist Ludwig Boltzmann (1844-1906) pondered the concept of entropy after heat had been unmasked as the total energy of motion of countless molecules, and temperature as the average energy of individual molecules. Boltzmann realized that a change in entropy was a change in arrangement of the molecules. High entropy systems looked random, so he decided to relate entropy to probability. In the process, he invented statistical mechanics, which was further refined and developed by the American J. Willard Gibbs (1839-1903). Boltzmann’s ideas can be recast in a way that avoids probability. If $W$ is the number of ways in which a system can be arranged without affecting the measured properties of the system, and $S$ is the physical entropy, then

$$S = k \log(W)$$

where $k$ is a constant with the appropriate units, known as Boltzmann's constant ($1.38 \times 10^{-23}$ Joules per degree Kelvin). This equation is inscribed on Boltzmann’s tombstone in the Central Cemetery in Vienna. By deriving the Clausius formula for entropy from his interpretation, Boltzmann was able to show that his interpretation was indeed correct.
The significance of Boltzmann’s interpretation of entropy transcends the fact that it furnished, for the first time, a mechanical model of this useful quantity. Since the number $W$ is defined in terms of the measured properties of the system, it has a subjective component - it depends on the information you have available. Boltzmann understood this connection and made it more specific. He pointed out that since the value of entropy rises from zero, when we know all about a system, to its maximum value when we know least, it measures our ignorance about the details of the motions of the molecules of a system. (Physical) entropy is not about speeds or positions of particles, the way temperature and pressure and volume are, but about our lack of information. Boltzmann’s measure of information was simply $W$, the number of ways of rearranging a system. When that number is large, our ignorance is large; when it is small, our ignorance is correspondingly small. By identifying entropy with missing information, Boltzmann hurled the concept of information into the realm of physics.

The relationship between physical and information-theoretic entropy is sharpened by consideration of the thought experiment of the Scottish physicist James Clerk Maxwell (1831-1879) in 1871. Maxwell postulated two gas-filled vessels at equal temperatures, connected by a valve and a daemon that is able to rapidly open and close the valve so as to allow only fast-moving molecules to pass in one direction and only slow-moving molecules to pass in the other direction. This violates the second law of thermodynamics. Information theory allows an exorcism of Maxwell’s daemon - either by looking at the sum of the two types of entropy or (as done by the Hungarian physicist Leo Szilard in 1929) by looking at the cost in energy of obtaining the information in order to select the molecules. Erwin Schrödinger (1887-1961) in his book “What is life?” (1944) had suggested that negative entropy was the hallmark of life, but considered that the connection with information was merely an analogy.

An interesting footnote is that the study of black holes and Hawking radiation (1974) has led to a concept of entropy not yet reducible to statistical mechanics, resurrecting the status of thermodynamics to primacy over statistical mechanics, rather than the secondary status it held in the period from Boltzmann’s work until 1974.

Section 2 leans on [3] and [2] heavily, with supplementary borrowings from [6]. The potted history of probability is from [1], and the rest of Section 3 from [3]. The opening of Section 4 (and the title) come from [2], there is influence from [10] and [9], but the style and many of the explanations are my own. Section 5 follows [10] closely, as do Sections 6 and 7. Section 8 opens with Bill Cherowitzo’s words from his website at University of Colorado at Denver. Section 9 is from [14], as is Section 11. Section 12 mostly follows [9]. Section 13 opens with Richard Kuntz’s words from his website at Monmouth University. Section 14 follows [4], with supplementary material from [2], and from general web-surfing.
I recommend von Baeyer’s book highly. It is truly excellent. I find Stinson’s book a total disappointment. Roman’s book is easy, breezy and a delight for a formal book (at a low level). Poli and Huguet is typically French and stuffed full of information, but probably too hard for many 3CC students. The Encyclopaedia Brittanica articles are a model of excellent writing and also relatively short and to the point. I love Kay’s book; it is an extremely difficult book to read - very post-modern - but the challenge is well worth it. (Of course, it’s really off-topic here, but the discussion of the cold-war social history context, such as the role of the Rockefeller foundation, in the rise of cybernetics and information theory is invaluable, and the discussion of similarities between the work of Shannon and of Wiener, as well as “precursors” such as Szilard and Schrödinger (and even Hartley and Nyquist) are also irreplaceable.)

References


