Entropy, Coding and Data Compression

S Natarajan

S Natarajan is a Professor at the Bangalore Centre of the Indian Statistical Institute.

Introduction

Rapid progress in all fields of scientific and technological activity has been the hallmark of the twentieth century. Though much of these was the result of the combined efforts of several scientists, some of the path-breaking developments were due to the original work of a few gifted ones. Claude Elwood Shannon (1916-2001) was one such in the field of communication science and technology. During the nineteen forties, he was working as a communications engineer in Bell Telephone Laboratories. The major problem at that time was reliable communication over radio, telephones, telegraphy, etc. The approach to these problems was that of A N Kolmogorov and N Wiener who advocated a model of a stochastic process of signals corrupted by disturbance, called noise, due to the medium used for transmission. The problem then is to recover the original signal from the corrupted one. Shannon made a radical departure from their approach by introducing the idea of encoding the signal before passing through the medium. In a single (two part) paper [1] entitled 'A Mathematical Theory of Communication' he not only provided a new set-up for the problem, but also proved a number of basic theorems and gave a few practical methods to achieve what the theorems promised. He introduced the concept of entropy of a probability distribution (which already was in use in thermodynamics as a measure of disorder) and showed how this determines the minimum possible rate of reliable communication. He also named the new subject 'information theory' since entropy is a measure of the information content of a probability distribution. The entropy idea was later taken to ergodic theory in 1958 by A N Kolmogorov to solve the outstanding prob-

Entropy is a measure of the information content of a probability distribution.

The entropy idea was taken to ergodic theory by Kolmogorov to solve the outstanding problem of isomorphism of Bernoulli shifts, thus giving birth to a very rich area of research known as entropy theory of measure-preserving transformations.

lem of isomorphism of Bernoulli shifts, thus giving birth to a very rich area of research known as entropy theory of measure preserving transformations.

As laid down by Shannon, the fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. Thus there is a *source* producing messages in discrete or continuous time for transmission to the *destination* through a *channel* which acts as the medium. We will restrict ourselves to the discrete case: the source emits one symbol per unit of time. Each message is a finite string of symbols from a given collection of symbols called the *source alphabet.* The channel, as in telegraphy, may be capable of receiving and sending messages using a different alphabet (like dots, dashes and spaces as in Morse code) and so the selected message has to be *encoded* before transmission through the channel. The output of the channel then should be *decoded* by the receiver and presented to the destination. The channel is said to be *noiseless* if its output is exactly the same as the input – that is, there is no error due to the channel; otherwise it is called *noisy.* Shannon considered both noisy and noiseless channels and proved theorems on the limitations to reliable communication. Very soon, practical methods for communication through noisy channels were investigated, leading to the theory of error correcting codes for which the theory of finite fields is a fundamental tool. A series of three articles by Priti Shankar [2] gives an elaborate account of these with some interesting applications. However only after about thirty years of Shannon's work, similar methods were devised for noiseless coding; these are now better known as data compression techniques for the following reason. The source output may be regarded as 'data' which may be encoded or 'compressed' and stored in a channel like a computer for retrieval at a later date. In this article we confine ourselves to a discussion of noiseless coding only.

We give a quick account of Shannon's theorems and a description of one of the data compression procedures.

Models for the Source and the Channel

Let $S = \{a_1, a_2, \quad , a_M\}$ be the source alphabet. Fundamental to Shannon's model is the assumption that the output of the source is considered as a stochastic process $X_1, X_2, \quad$ with values in S. The simplest model for this prescribes a probability distribution $P = (p_1, p_2, \quad , p_M)$ on S and the messages are built up by choosing the letters from S independently and using the given probability distribution. This is known as a *discrete memoryless source.* (Shannon himself considered $\{X_n\}$ as a Markov chain. All the results we shall quote are true more generally for 'stationary ergodic processes' but we confine ourselves to the memoryless case for the sake of simplicity.) The channel accepts and transmits symbols from a possibly different finite set called the *channel alphabet.* We will assume that this is the binary alphabet $B = \{0, 1\}$. If the source is producing messages at a constant rate, we should try to maximise the number of messages which can be sent in a given time. For this we may use the following method introduced by Shannon.

Variable Length Coding

Given the source alphabet $S = \{a_1, a_2, \quad , a_M\}$, a *code* is a map C from S to B^*, the set of all strings of zeros and ones of finite length. The range of C is called the *codebook* each of whose elements is a *codeword.* The length function $\ell(.)$ on S is defined as the length of the corresponding codeword. Given a probability distribution P on S, the average codeword length is the expected value of the length function. We can send more messages in a given time if the average codeword length is minimised.

As a practical example, consider the Morse code which

Given a probability distribution P on S, the average codeword length is the expected value of the length function. We can send more messages in a given time if the average codeword length is minimised.

A basic result about prefix codes is that for any probability distribution P on S the average codeword length is greater than the entropy of P.

has been in use for sending messages in the English language telegraphically. In that code, combinations of dots, dashes and spaces are used as codewords; frequently occurring letter e is given a short codeword '. whereas an infrequently occurring letter like q or z is given a longer codeword. This has the effect of minimising the average codeword length.

When the source is producing a sequence of symbols, these are coded using the code C by concatenation, i.e., by consecutively writing the corresponding codewords. When these are received at the destination, it should be possible to separately identify the codewords, so that using the codebook, one can reconstruct the original message. If u and v are two codewords and uv their concatenation, we should be able to split this only as u and v and not in any other way as codewords.

A codeword u is a *prefix* of a codeword w if $w = uv$ for some v. A codebook is said to be prefix-free if no codeword is a prefix of some other codeword. A code is said to be a *prefix code* if it is one-to-one and its range is prefix-free. Then it is possible to decode the original message from the coded message without error. For example, when $M = 4$, the code $C = \{0, 10, 110, 111\}$ is a prefix code.

A basic result about prefix codes is that for any probability distribution P on S, the average codeword length is greater than the *entropy* of P or the *source entropy* defined as follows:

Definition. The entropy of the probability distribution $P = (p_1, p_2, \quad , p_M)$ is defined as

$$H(P) = -\sum p_i \log p_i,$$

where 0 log 0 is taken as 0, and here and in what follows, logarithms are taken to base 2.

Shannon interpreted entropy as the amount of informa-

tion gained by one single performance of the random experiment (S, P), i.e., the experiment of choosing a letter from S at random using the probability distribution P. If one of the probabilities p_i is large compared to others then there is a high probability of i being chosen by the source and so if i is actually chosen, then the surprise element in the observed outcome is small; so also the information conveyed by this outcome. One way to associate a measure to this information is $1/p_i$. But then, since we are assuming independence of successive choices, the information conveyed by the event that the first two outcomes are i and j would be $1/p_ip_j$. A natural property we would like to have of any measure of information is that for independent choices, the amount of information should add up. A simple way to ensure this is to take logarithms; thus we define the information conveyed by the outcome i as $\log\,(1/p_i)$. Since the experiment can lead to any one of the possible M outcomes, the information content or the average amount of information gained by performing the experiment once is $\sum p_i \log\,(1/p_i) = H(P)$. If $p_i = 1$ for some i, then $H(P) = 0$. When all the p_i are equal and equal to $1/M$, then $H(P) = \log M$, which is also the maximum value that the entropy function can take.

Shannon intrepreted entropy as the amount of information gained by one single performance of the random experiment of choosing a letter from the source alphabet using a probability distribution on it.

It is possible to put down a set of four reasonable conditions which any measure of information should satisfy and then show that $H(P)$ is indeed the unique function of the probabilities satisfying these conditions (see [4]).

The important connection between prefix codes and source entropy was proved by Shannon:

Theorem 1. For any prefix code C, and any probability distribution P on S, the average codeword length is at least $H(P)$. There exists a prefix code with average codeword length smaller than $H(P) + 1$.

Instead of coding individual outputs of the source, we can consider coding blocks of source outputs using vari-

Box 1. Hide and Seek

Consider the following guessing game which we call 'hide and seek'. Your friend chooses an element from a finite set, say $S = \{a_1, a_2, \ldots, a_M\}$. You have to discover her choice by putting questions to her to which she will answer only by saying 'yes' or 'no'.

Here is a naive sequence of questions. Is her choice a_1? If the answer is yes, you stop. Otherwise ask: is her choice a_2? If yes, stop; otherwise proceed with a_3. And so on. Writing 1 for an 'yes' and 0 for a 'no', we can code the possible outcomes of the above procedure in the following way: $1, 01, 001, 0001, \ldots$, where the length of the codeword for a particular outcome equals the number of questions asked to catch her choice. (Note that the last question need not be asked.) Lo and behold! This is a prefix code for the set S.

You can of course think of other procedures. For instance, fixing a subset A_1 of S, first ask: is her choice in A_1? If yes, repeat the question with a subset A_2 of A_1; otherwise use a subset of the complement of A_1 and proceed. Codewords can be constructed in the same way as above. A codeword will have 1 in the first place if her choice is in A_1 and 0 otherwise. The second digit in the codeword depends on her answer to the second question. And so on. We again end up with a prefix code. A moment's reflection will convince you that we can also go backwards from a prefix code to a sequence of yes or no questions.

Thus there are different sequences of questions to locate the element of S which your friend chose. Which of these would you consider the 'best'? Naturally, the one which requires the minimum number of questions on the average. Why 'on the average'? Since you do not know how your friend makes her choice, you would assume that she does so using a probability distribution, say $P = \{p_1, p_2, \ldots, p_M\}$ on S. The aim then is to minimise the average number of questions asked. One should therefore start with elements or subsets of S which have high probability. This is indeed the idea behind the proof of Theorem 1 as well as the Shannon–Fano code. (See [4].)

For a leisurely and masterly account of various ideas in information theory as well as in games of chance, the reader is well-advised to browse through the book by Alfred Renyi[3].

able length codes. If n is the length of the blocks we are using, then it can be derived from the above theorem that the average codeword length per source symbol, also called the rate of the code or the rate of transmission of information, is greater than $H(P)$ and can be made smaller than $H(P) + 1/n$ and so as close to the entropy as desired by increasing n. Thus entropy is the best possible rate of transmission of information using variable length codes.

Entropy is the best possible rate of transmission of information using variable length codes.

Block Coding

Sometimes the above procedure can be inconvenient in practice: for instance, if the encoding has to be done at a fixed rate (in time units), it would be preferable to use codewords all of which have the same length, say k. Suppose we are going to code source messages of length n. There are 2^k binary strings of length k and M^n source output strings of length n. Thus if $M^n \leq 2^k$, then we can assign uniquely one codeword to each message block of length n. The ratio $R = k/n$ is the number of binary digits (often called bits) used per source symbol and is called the rate of transmission of information or the rate of the code. Thus if $R \geq \log M$, then it is possible to devise a procedure which is error-free. However this is generally too high and does not make use of the probabilistic nature of the source. Shannon's observation was that if we are willing to tolerate a small probability of making mistakes, then we can bring down the rate considerably. The important point to note is that we cannot however bring it down to an arbitrarily small quantity; there is a lower bound to the rate of transmission, even allowing for a small probability of error. This lower bound again turns out to be the source entropy. This follows from an important result on block coding proved by Shannon known as the asymptotic equipartition property (AEP). Since P is fixed, we shall write H for $H(P)$.

Shannon's observation was that if we are willing to tolerate a small probability of making mistakes, then we can bring down the rate of the code considerably.

Theorem 2. Given an $\epsilon > 0$ and a $\delta > 0$, there is an N such that for any $n > N$, the set D_n of all source output strings of length n is the disjoint union of two sets D_{n1} and D_{n2}, which have the following properties:

(i) If $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in D_{n1}$, then $2^{-n(H+\delta)} < \Pr(\mathbf{x}) < 2^{-n(H-\delta)}$ and

(ii) $\Pr(D_{n2}) < \epsilon$,

where $\Pr(A)$ denotes the probability of the set A.

This can be proved using the weak law of large numbers applied to the sequence $\{-\log P(X_n)\}$ of random variables where of course, $\{X_n\}$ is the source output sequence. From this we can get the following procedure for encoding the source output. Fixing an $\epsilon > 0$ and a $\delta > 0$, let n be an integer as above. List all the source output strings of length n in decreasing order of probability. Go down the list adding the probabilities till a value greater than $1 - \epsilon$ is reached. Denote by $N_n(\delta)$ the cardinality of the set of source outputs thus obtained. Clearly, $N_n(\delta) < 2^{n(H+\delta)} < N^n$. These source outputs can be mapped one-to-one into the set of zero-one sequences of length $[n(H + \delta)] + 1$, the smallest integer greater than $n(H + \delta)$. Map the remaining source outputs (which together have a probability of not more than ϵ) into one single sequence of the same length. An error in decoding occurs only when the latter sequence is received and so the probability of error in decoding is less than ϵ. The rate of this code is close to $H + \delta$ for large n and so can be made arbitrarily close to H It can be shown that if we try to use a scheme with rate less than H, then the probability of error goes to one as $n \to \infty$.

For filling up the details in the above discussion, the reader may refer [4] or [5].

Thus once again entropy turns out to be the minimum rate at which coding has to be done in order to achieve reliable transmission of messages. More surprising re-

sults await you in the next section!

Universal Source Coding

All the above constructions assume the knowledge of source statistics; i.e., the probabilities with which the source is producing messages. In practice, however these may be unknown. Thus there arose investigations on finding good coding procedures without using the source statistics. These came to be known as universal codes. The idea is to use the output of the source itself in order to reduce codeword length. Roughly speaking, if a phrase is likely to occur again and again we could assign a short symbol for it. In other words, the first occurrence has to be transmitted in full, but subsequent occurrences can be transmitted by just referring to the previous occurrence in some way. This idea is behind the simple Lempel–Ziv (LZ) algorithm which was discovered in 1977. We will describe this in some detail. There have been several modifications to this for practical applications: one area where these are applied in practice is in data compression for computer storage of information.

Roughly speaking, if a phrase is likely to occur again and again we could assign a short symbol for it. In other words, the first occurrence has to be transmitted in full, but subsequent occurrences can be transmitted by just referring to the previous occurrence in some way.

OK, you may say, but how do we judge their optimality properties? In fact these universal procedures turn out to be optimal, whatever be the source probabilities (more generally even if $\{X_n\}$ is any stationary ergodic process); the entropy of the source under consideration being once again the minimal rate of transmission. Let us first describe the algorithm and then state the optimality theorem.

The simple LZ algorithm parses an n-length output of the source $x = x_1x_2 \quad ,x_n$ as a concatenation of words of variable length, say $x = w_1w_2 \quad$ The word formation can be described by saying that the next word is the shortest NEW word. The first word w_1 is of course the first symbol x_1. Suppose we have fixed the first j words, say, $w_1w_2 \quad .w_j = x_1x_2 \quad .x_{n_j}$. If $x_{n_j+1} \notin$

$\{w_1, w_2, \ldots, w_j\}$ then $w_{j+1} = x_{n_j+1}$. Otherwise, $w_{j+1} = x_{n_j+1}x_{n_j+2} \ldots x_{m+1}$ where m is the smallest integer larger than n_j such that $x_{n_j+1}x_{n_j+2} \ldots x_m \in \{w_1, w_2, \ldots w_j\}$ and $x_{n_j+1}x_{n_j+2} \ldots x_{m+1} \notin \{w_1, w_2, \ldots w_j\}$.

As an illustrative example, if the source puts out 1100101 0001000100... the successive words are 1, 10, 0, 101, 00, 01, 000, 100, ... Note that this is a sequential operation; the later words do not have any effect on the earlier ones. Thus the initial segment of length n is expressed as $w_1w_2 \ldots w_kv$, where the final block v is either empty or is equal to w_j for some $j \leq k$. This parsing defines a prefix code since each new word is new only because of its last symbol and hence it is specified by giving a pointer to where the part before its final symbol occurred earlier, together with a description of its final symbol.

More explicitly the code is constructed as follows. Let, as usual, $[x]$ denote the integral part of x. To define the code, we need two one-to-one functions, say f and g. The function f maps the set $\{1, 2, \ldots, n\}$ into the set of zero-one sequences of length $[\log n] + 1$ and will be used to locate part of the parsed word at its earlier occurrence. The function g will map S into the set of zero-one sequences of length $[\log N] + 1$ and will be used to identify the last source symbol in the parsed word. The parsed word $w_1w_2 \ldots w_kv$ is thus mapped into the concatenation $b_1b_2 \ldots b_kb_{k+1}$ of binary words $b_1, b_2, \ldots, b_k, b_{k+1}$ defined as follows:

(a) If $j \leq k$ and w_j has length 1, then $b_j = 0g(w_j)$.

(b) If $j \leq k$ and $i < j$ is the least integer for which $w_j = w_ia, a \in S$, then $b_j = 1f(i)0g(a)$.

(c) If v is empty, then b_{k+1} is empty. Otherwise, $b_{k+1} = 1f(i)$ where i is the least integer such that $v = w_i$.

The theorem giving the optimality of the procedure can be stated as follows:

Theorem 3. If P is any probability distribution on S, then

$$\lim_{n\to\infty} \frac{E(\ell_n(.))}{n} = H(P),$$

where $E(\ell_n(.))$ denotes the average codeword length of the above scheme.

In fact, conclusions stronger than the above are known but some more sophisticated mathematics will be needed for stating them. The reader is referred to the book [6] by Paul Shields for an excellent exposition of these and also many other results in this area where entropy sits like the *king* in the centre.

Suggested Reading

[1] C E Shannon, A Mathematical Theory of Communication, *Bell Systems Technical Journal*, Vol.27, pp. 379-423, 623-656, 1948.

[2] Priti Shankar, Error Correcting Codes, *Resonance*, Vol. 1, 10, 1996 and Vol.2, Nos 1 and 3, 1997.

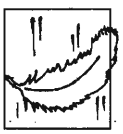
[3] Alfred Renyi, *A Diary on Information Theory*, John Wiley and Sons, New York, 1987.

[4] R Ash, *Information Theory*, Interscience Publishers, New York, 1965.

[5] D S Jones, *Elementary Information Theory*, Oxford University Press, Oxford, 1979.

[6] Paul C Shields, *The Ergodic Theory of Discrete Sample Paths*, American Mathematical Society, 1996.

Address for Correspondence
S Natarajan
Indian Statistical Institute
Bangalore 560059, India.
e-mail: sn@isibang.ac.in



It is an ironic fact that physicists spent many an hour gazing through window glass speculating if an amorphous material could have a bandgap. Window glass is indeed amorphous and yet, precisely because it is transparent to visible light, it must have a bandgap beyond the visible; depending on the glass, the absorption edge lies at about 3000 Å or 4.0 eV.

J Mort
Senior Research Fellow at Xerox Corp., Webster, NY
The Anatomy of Xerography, Its Invention and Evolution
McFarland & Company Inc., Jefferson
North Carolina, USA, 1989, p.77