1994 Shannon Lecture

Typical Sequences and All That: Entropy, Pattern Matching, and Data Compression

Aaron D. Wyner

AT&T Bell Laboratories
Murray Hill, New Jersey 07974
USA
(adw@research.att.com)

I. Introduction

This will be a talk about how pattern matching relates to certain problems in information theory. Here is a typical pattern matching problem.

> A monkey sits at a typewriter and every second types a single Latin letter. Assume that all 26 letters are equally likely, and successive letters are independent. How long on the average will it take the monkey to type "CLAUDESHANNON"?

The answer is that the average waiting time (in seconds) is

$$26^{13} = 2^{13\log_2 26} = 2^{\ell H},$$

where

$\ell = 13 =$ number of letters in "CLAUDESHANNON"

$H = \log 26 =$ entropy of the monkey's data sequence.

(All logarithms are taken to the base two.) We will show that entropy and pattern matching are closely connected by looking at three problems:

A. Observe the output of a data source, $X_1, X_2, X_3, \ldots$, and estimate the entropy of the source.

B. Encode a data source $\{X_k\}$ into binary symbols using about H bits/source symbol (optimal lossless data-compression).

C. Observe N_0 symbols from an unknown data source $\{X_k\}$, and decide whether or not the source statistics are the same as those of a given known source (classification).

In the next section we will give some preliminary definitions and facts, and then discuss these problems in the following three sections.

II. Preliminaries

We need a bit of notation. An *information* or *data source* is a random sequence $\{X_k\}$, $-\infty < k < \infty$. We assume that the sequence is stationary and ergodic, and X_k takes values in the finite set $\mathcal{A}$, with cardinality $|\mathcal{A}| = A$. The probability law defining the data source is given by

$$P_K\left(\mathbf{x}_1^K\right) \triangleq \Pr\left\{\mathbf{X}_1^K = \mathbf{x}_1^K\right\}, \qquad K = 1, 2, \ldots, \quad (2.1)$$

where we use the notation $\mathbf{x}_i^j = (x_i, x_{i+1}, \ldots, x_j)$, $i < j$. When sub- and super-scripts are obvious from the context, they will be omitted.

The *entropy* of the data source is

$$\begin{aligned} H &\triangleq \lim_{K\to\infty} \sum_{\mathbf{x}} \frac{1}{K} P_K\left(\mathbf{x}_1^K\right) \log \frac{1}{P_K\left(\mathbf{x}_1^K\right)} \\ &= \lim_{K\to\infty} \frac{1}{K} E \log \frac{1}{P_K\left(\mathbf{X}_1^K\right)}. \end{aligned} \quad (2.2)$$

The indicated limit always exists. It is easy to show that $H \le \log A$, with equality iff the $\{X_k\}$ are i.i.d. and equally probable.

The theorem (due to Shannon and McMillan) that lies at the heart of much of information theory is the "Asymptotic Equipartition Property" or "AEP". We state it as follows. For $\epsilon > 0$, and $\ell = 1, 2, \ldots$, let the set

$$T(\ell, \epsilon) = \left\{\mathbf{x} \in \mathcal{A}^\ell : \left|\frac{1}{\ell} \log \frac{1}{P_\ell(\mathbf{x})} - H\right| \le \epsilon\right\}. \quad (2.3)$$

Thus if $\mathbf{x} \in T(\ell,\epsilon)$, $P_\ell(\mathbf{x}) = 2^{-\ell(H\pm\epsilon)}$. The AEP is a theorem which states that with $\epsilon > 0$ held fixed, as ℓ becomes large, the probability of $T(\ell,\epsilon)$ approaches 1. That is,

Theorem 2.1 (AEP) *For fixed $\epsilon > 0$,*

$$\lim_{\ell\to\infty} \Pr\left(T(\ell,\epsilon)\right) = 1.$$

Since with probability close to 1, the random vector $\mathbf{X}_1^\ell \in T(\ell,\epsilon)$ (ℓ large), the set $T(\ell,\epsilon)$ is called the "typical set". A proof of the AEP can be found in many textbooks and elsewhere. See for example [3]. An important property of $T(\ell,\epsilon)$ is that it is not too large:

Proposition 2.2. $|T(\ell,\epsilon)| \le 2^{\ell(H+\epsilon)}$.

Proof:

$$1 \ge \Pr\left(T(\ell,\epsilon)\right) = \sum_{\mathbf{x}\in T} P_\ell(\mathbf{x}) \ge 2^{-\ell(H+\epsilon)}|T(\ell,\epsilon)|.$$

The second inequality follows from the definition of $T(\ell,\epsilon)$ (2.3).

Let us remark that there is a stronger version of the AEP called the Shannon-McMillan-Breiman Theorem. (See for example [1].) We state this as

Theorem 2.1′. *For a stationary, ergodic source, with probability 1, as $\ell \to \infty$,*

$$\frac{1}{\ell}\log\frac{1}{P_\ell(\mathbf{X}_1^\ell)} \to H.$$

We next turn to pattern matching and give some definitions and theorems that we will need later.

Definition 2.3. For $\mathbf{x} = \mathbf{x}_{-\infty}^{\infty}$, and $\ell = 1, 2, \ldots$, define $N_\ell(\mathbf{x})$ as the *smallest* integer $N \ge 1$ such that $\mathbf{x}_1^\ell = \mathbf{x}_{-N+1}^{-N+\ell}$.

$N_\ell(\mathbf{x})$ is a (backward) "recurrence time" for $\mathbf{x}_1^\ell$. As an example suppose that $\{x_k\}$ is as follows

$k:$	-5	-4	-3	-2	-1	0	1	2	3	4	5
$x_k:$	a	b	b	c	a	b	b	b	c	a	c

Let $\ell = 4$, and think of $\mathbf{x}_1^\ell = \mathbf{x}_1^4$ as a template. Slide the template to the left until we see a perfect match. In the example, $\mathbf{x}_1^\ell = (b\ b\ c\ a)$, and we get the first perfect match 5 places to left (since $\mathbf{x}_1^4 = \mathbf{x}_{-4}^{-1}$). Thus $N_4(\mathbf{x}) = 5$. Note also that $N_1 = N_2 = 1$, $N_3 = 5$ and $N_5 > 6$.

We now state a theorem about $N_\ell(\mathbf{x})$, which is a special case of a theorem of Kac [4]. A proof is given in the Appendix.

Theorem 2.4 (Kac) *For all $\mathbf{z} \in \mathcal{A}^\ell$,*

$$E\left(N_\ell(\mathbf{X}) \mid \mathbf{X}_1^\ell = \mathbf{z}\right) = 1/P_\ell(\mathbf{z}). \tag{2.4}$$

A plausibility argument for Theorem 2.4 goes as follows. Fix $\mathbf{z}$. Define the random variables

$$W_i = \begin{cases} 1, & \text{if } \mathbf{X}_{-i}^{-i+\ell-1} = \mathbf{z} \\ 0, & \text{otherwise.} \end{cases}$$

Of course, $EW_i = P_\ell(\mathbf{z})$. Then it is reasonable to write

$$\left\{\begin{array}{c}\text{Average (backward) recurrence}\\ \text{time for } \mathbf{z}\end{array}\right\}$$

$$= \left\{\begin{array}{c}\text{Average time between}\\ \text{occurrences of } \mathbf{z}\end{array}\right\}$$

$$= \lim_{K\to\infty} \frac{K}{\left\{\begin{array}{c}\text{no. of occurrences}\\ \text{of } \mathbf{z} \text{ in } X_{-K+1}^{\ell-1}\end{array}\right\}} = \lim_{K\to\infty} \frac{K}{\sum_{i=0}^{K-1} W_i}$$

$$\stackrel{\text{(a)}}{\longrightarrow} \frac{1}{EW_i} = \frac{1}{P_\ell(\mathbf{z})}.$$

Step (a) follows from the ergodic theorem. The plausibility argument is completed by observing that reasonably $\{\text{Average recurrence time for } \mathbf{z}\} = E\left(N_\ell \mid \mathbf{X}_1^\ell = \mathbf{z}\right)$.

Using Theorem 2.4 and the AEP (which tells us that the right-member of (2.4) is about $2^{\ell H}$) we can obtain

Theorem 2.5 (Wyner and Ziv) *As $\ell \to \infty$*

$$\frac{1}{\ell}\log(N_\ell(\mathbf{X}) \to H \qquad (\textit{in probability})$$

Actually the convergence is with probability 1, and a proof of this stronger form is contained in the Appendix.

We next define another quantity which is closely related to $N(\mathbf{x})$.

Definition 2.6. For $\mathbf{x} = \mathbf{x}_{-\infty}^{\infty}$, and $n = 1, 2, \ldots$, define $L_n(\mathbf{x})$ as the *largest* integer $L \ge 1$ such that a copy of $\mathbf{x}_1^L$ begins in $[-n+1, 0]$. (Think of $\mathbf{X}_{-n+1}^0$ as a "window".)

As an example, suppose that $\{x_k\}$ is as before:

$k:$	-5	-4	-3	-2	-1	0	1	2	3	4	5
$x_k:$	a	b	b	c	a	b	b	b	c	a	c

Let $n = 5$. Then $(b\ b\ c\ a) = \mathbf{x}_1^4$ is the *longest* string starting at position 1, a copy of which begins in the window $\mathbf{x}_{-4}^0 = (b\ b\ c\ a\ b)$. (Note that a copy of $\mathbf{x}_1^5 = (b\ b\ c\ a\ c)$ does *not* begin in $\mathbf{x}_{-4}^0$.) Thus $L_5(\mathbf{x}) = 4$.

N_ℓ and L_n are in a sense dual quantities since the events

$$\{N_\ell(\mathbf{X}) > n\} = \left\{ \begin{array}{c} \text{a copy of } \mathbf{X}_1^\ell \textit{ does not} \\ \text{begin in } [-n+1, 0] \end{array} \right\}$$
$$= \{L_n(\mathbf{X}) < \ell\}. \quad (2.5)$$

Thus Theorem 2.5 implies that $L_n(\mathbf{X}) \to \frac{\log n}{H}$ (in probability). Collecting the above results, we have

Theorem 2.7. *(a) As $\ell \to \infty$,*

$$\frac{1}{\ell} \log N_\ell(\mathbf{X}) \to H \qquad (\textit{in probability}),$$

(b) As $n \to \infty$,

$$L_n(\mathbf{X}) \to \frac{\log n}{H} \qquad (\textit{in probability}).$$

III. Entropy Estimation (Problem A)

We first show how the pattern matching ideas in Section II can be used to obtain an efficient "sliding window" entropy estimation technique (Problem A in Section I).

Observe $\{X_1, X_2, \ldots\}$. Initially, let $\mathbf{X}_1^n$ define a "window", and let $L^{(1)}$ be the largest integer L such that

$$X_{n+1}^{n+L} = X_m^{m+L-1}, \qquad \text{for some } m \in [1, n].$$

Thus $L^{(1)}$ is the length of the longest string starting at X_{n+1} a copy of which begins in the window $\mathbf{X}_1^n$. Of course, $L^{(1)}$ has the same statistics as L_n (Def. 2.6).

Next shift the window 1 position, so that the new window is $\mathbf{X}_2^{n+1}$, and define $L^{(2)}$ in the same way. Repeat this process to get $L^{(k)}$, $k = 1, 2, 3, \ldots$.

Now if the source has finite memory, it can be shown [9, 10] that, as $n \to \infty$,

$$EL_n \sim \frac{\log n}{H} \quad (3.1)$$

(Note that Eq. (3.1) is close to, but not the same as Theorem 2.7(b).). Thus, it follows from the ergodic theorem that

$$\frac{1}{K} \sum_{k=1}^{K} L^{(k)} \to E\, L_n \sim \frac{\log n}{H}, \quad (3.2)$$

and a good estimate for the entropy is

$$\widehat{H} \triangleq \frac{K \log n}{\sum_{k=1}^{K} L^{(k)}} \quad (3.3)$$

for some large K. Even if the source does not satisfy (3.1), Theorem 2.7(b) can be used to obtain an estimate of $\widehat{H}$ from the $\left\{L^{(k)}\right\}$.

The technique was used very effectively in [2], where the entropy of the information bearing and non-information bearing parts ("exons" and "introns", respectively) were estimated and compared.

IV. Data-Compression (Problem B)

The AEP immediately suggests a data-compression scheme. Theorem 2.1 and Proposition 2.2 together imply that, when ℓ is large, the set $T(\ell, \epsilon)$ has no more than $2^{\ell(H+\epsilon)}$ members and has probability nearly 1. Thus, assuming that the source statistics are known, the system designer can index the members of $T(\ell, \epsilon)$, using no more than $\ell(H + \epsilon)$ bits.

The scheme is as follows. If $\mathbf{X}_1^\ell \in T(\ell, \epsilon)$, then encode $\mathbf{X}_1^\ell$ as its index in $T(\ell, \epsilon)$. This requires* $\le \ell(H + \epsilon)$ bits. If $\mathbf{X}_1^\ell$ does not belong to $T(\ell, \epsilon)$, then encode $\mathbf{X}_1^\ell$ uncompressed. This requires $\le \ell \log A$ bits. Including a 1-bit flag to distinguish the two modes, we have described a ("fixed-to-variable-length") lossless code with rate

$$E\frac{1}{\ell} \left\{\text{no. of bits to encode } X_1^\ell\right\}$$
$$\le P\left(T(\ell, \epsilon)\right) \frac{\ell(H+\epsilon)}{\ell} + P\left(T^c(\ell, \epsilon)\right) \ell \frac{\log A}{\ell}$$
$$+ \frac{1}{\ell} \to H + \epsilon, \qquad \text{as } \ell \to \infty.$$

Thus the source is encoded into binary symbols using about H bits/source symbol, and this rate is known to be optimal. But what can be done if the source statistics are unknown to the system designer?

The Lempel-Ziv data-compression algorithms provide a universal compression technique for coding a data source into binary using about H bits/source symbol without knowledge of the source statistics. Their technique is intimately connected to pattern matching. We'll describe the "sliding-window Lempel-Ziv algorithm" (also called "LZ '77").

Here is how the algorithm works. Let n be an integer parameter. Assume that the n-string $\mathbf{X}_{-n+1}^0$ is available to both the encoder and decoder — say by encoding $\mathbf{X}_{-n+1}^0$ with no compression. We will encode $X_1, X_2, \ldots$, so that the cost of encoding $\mathbf{X}_{-n+1}^0$ is "overhead" which can be amortized over an essentially infinite time, and this cost doesn't contribute to the rate. Think of $\mathbf{X}_{-n+1}^0$ as our first "window".

*We ignore integer constraints.

We now begin the encoding process. Let L_n be as in Section II, the largest integer $L \geq 1$, such that

$$\mathbf{X}_1^L = \mathbf{X}_m^{m+L-1}, \qquad m \in [-n+1, 0].$$

The quantity m is called the "offset" corresponding to the "phrase" $X_1^{L_n}$. This first phrase is encoded by

(a) a binary representation of L_n. This requires about $\log L_n + O(\log\log n)$ (for large n, see [8]).

(b) a binary representation of the offset m. This requires $\log n$ bits.

If $L_n = 0$ (i.e. $X_1 \neq X_{-m}$, $m \in [-n+1, 0]$) we let the first phrase be X_1, and encode it uncompressed. Also if a phrase is so short that number of bits to encode it ((a)+(b) above) exceeds $L_n \lceil \log A \rceil$, we encode the phrase uncompressed. We also need a flag bit to distinguish these two modes. Note that from Theorem 2.7(b), $L_n \sim \frac{\log n}{H}$ with high probability, so that the latter mode is very unlikely.

With the encoding done, the window is now shifted L_n positions to become $X_{-n+1+L_n}^{L_n}$, and the encoding procedure is repeated to form and encode a second phrase beginning with X_{L_n+1} using this new window. The process is continued indefinitely.

Now let's look at the decoding procedure. The decoder knows the first window, $\mathbf{X}_{-n+1}^0$, the offset m, and the length of the first phrase L_n. It can reconstruct this first phrase by starting at X_m (in the window) and moving ahead L_n positions. For example, if $n = 5$ and

$$(X_{-4}, X_{-3}, \cdots) = (a\ b\ c\ d\ e \,\vdots\, d\ e\ d\ a \cdots),$$

then $L_n = 3$ and $m = -1$. (This is because $\mathbf{X}_1^3 = \mathbf{X}_{-1}^1$). With knowledge of the window, $\mathbf{X}_{-4}^0 = (a\ b\ c\ d\ e)$, the decoder copies "$d$" and "$e$" to positions 1 and 2, respectively, and then copies the "d" in position 1 to position 3: Thus the decoder can recover the first phrase $\mathbf{X}_1^3$. Successive phrases are decoded in the same way.

We can now give an estimate of the rate of this algorithm. Since, with high probability, the phrase length will be long enough to use the first encoding mode,

$$\begin{aligned} \text{code rate} &\approx \frac{\text{no. of bits to encode phrase}}{\text{length of a phrase}} \\ &\approx \frac{\log n + \log L_n + O(\log\log L_n)}{L_n}. \end{aligned}$$

Since, from Theorem 2.7(b), $L_n \sim \frac{\log n}{H}$, with high probability when n is large, the code rate is about H.

The above analysis is not at all precise. For a careful discussion of the Sliding-Window Lempel-Ziv algorithm, the reader is referred to [8]. In particular the following is proved there.

Theorem 4.1. *When the sliding-window Lempel-Ziv algorithm is applied to a stationary ergodic source, for all $\epsilon > 0$, there exists a window size n (sufficiently large) such that*

$$\limsup_{K\to\infty} \frac{1}{K} E \left\{ \begin{array}{c} \text{no. of bits to} \\ \text{encode } \mathbf{X}_1^K \end{array} \right\} \leq H + \epsilon. \tag{4.1}$$

An interesting question is how large does the window size n have to be so that (4.1) will hold for a given $\epsilon > 0$? The answer of course is that it depends on the source statistics. Thus, although the specification of the algorithm does not depend of the source statistics explicitly, the choice of the proper window size does. In practice, a window size is chosen, and the algorithm is used on a variety of sources — for some sources the compression rate is close to the entropy, for others it may not be. It seems obvious that any so-called "universal algorithm" with a given memory size cannot perform well for all sources.

Concerning the main thrust of this talk, we observe that the window in the Lempel-Ziv algorithm plays the role of the typical set $T(\ell, \epsilon)$ in the classical compression scheme.

Some historical comments: The sliding-window LZ algorithm (LZ '77) was published in 1977 by A. Lempel and J. Ziv [11]. They published another less powerful but easier to implement version in 1978 [12]. In 1977 they established the optimality of LZ '77 in a combinatorial, non-probabilistic sense. True optimality was established for LZ '78 in [12]. (Also see [1, Section 12.10].) Finally, optimality of LZ '77 (Theorem 4.1) was established by Wyner and Ziv in [8]. Sliding-window LZ is the basis for the UNIX "gzip", and for the "Stacker" and "Doublespace" programs for personal computers.

V. Classification (Problem C)

Here is the sort of problem that we will address in this section.

> We are allowed to observe N_0 characters from the corpus of work of a newly discovered 16^{th} century author. We want to determine if this unknown author is Shakespeare. And we want to do it with minimum N_0.

A mathematical version of the problem is depicted in Figure 1. This classifier observes N_0 symbols from a stationary data source $\mathbf{X}_1^{N_0}$ ("newly discovered author") with probability law $P(\cdot)$ and alphabet $\mathcal{A}$. It also knows a second distribution on ℓ-vectors, $Q_\ell(\mathbf{z})$, $\mathbf{z} \in \mathcal{A}^\ell$ ("Shakespeare"). Its task is to decide whether or not $P_\ell(\mathbf{z})$ ($\mathbf{z} \in \mathcal{A}^\ell$), the ℓ-th order marginal distribution corresponding to $P(\cdot)$, is the same as Q_ℓ. Specifically, the classifier must produce a function $f_c\left(\mathbf{X}_1^{N_0}, Q_\ell\right)$ which, with high probability, equals 0 when $P_\ell \equiv Q_\ell$, and 1 when the Kullback-Liebler divergence $D_\ell(Q_\ell; P_\ell) \geq \Delta$, where

$$D_\ell(Q_\ell; P_\ell) \triangleq \frac{1}{\ell} \sum_{\mathbf{z} \in \mathcal{A}^\ell} Q_\ell(\mathbf{z}) \log \frac{Q_\ell(\mathbf{z})}{P_\ell(\mathbf{z})}, \tag{5.1}$$

and Δ is a fixed parameter. Recall that $D_\ell(Q_\ell, P_\ell) \geq 0$, with equality iff $Q_\ell \equiv P_\ell$, and is a measure of "differentness". If $0 < D_\ell(Q_\ell, P_\ell) < \Delta$, then nothing is expected of the classifier. The problem is to design a classifier as above with minimum possible N_0.

In [6], it is shown that for a finite-memory (Markov) source, when ℓ is large, the minimum N_0 is about $2^{\ell H + o(\ell)}$, where H is the source entropy. The intuition for this is the following. The classifier knows $Q(\mathbf{z})$, $\mathbf{z} \in \mathcal{A}^\ell$, and therefore it knows the corresponding typical set $T(\ell, \epsilon)$. It turns out that for $N_0 \geq 2^{\ell(H+\epsilon)}$, the sequence $\mathbf{X}_1^{N_0}$ will (with ℓ large and with high probability) contain the typical set corresponding the $P_\ell(\cdot)$ as subsequences. If these sets agree substantially, then the classifier declares that $Q_\ell \equiv P_\ell$. Otherwise, it declares $D_\ell(Q_\ell; P_\ell) > \Delta$.

More precisely, for $\mathbf{z} \in \mathcal{A}^\ell$ and $\mathbf{x} \in \mathcal{A}^{N_0}$, let $\widehat{N}(\mathbf{z}, \mathbf{x})$ be the smallest integer $N \in [1, N - \ell + 1]$ such that a copy of $\mathbf{z}$ is a substring of $\mathbf{x}$, i.e. $\mathbf{z} = \mathbf{x}_N^{N+\ell-1}$. If $\mathbf{z}$ is not a substring of $\mathbf{x}$, then take $\widehat{N}(\mathbf{z}, \mathbf{x}) = N_0 + 1$. Now the following can be shown to hold: For fixed $\mathbf{z}$, when ℓ is large and $N_0 = 2^{(H+\epsilon)\ell}$,

$$\text{(a)} \quad \Pr\left\{\widehat{N}\left(\mathbf{z}, X_1^{N_0}\right) \leq N_0\right\} \approx 1$$

$$\text{(b)} \quad \Pr\left\{\frac{1}{\ell}\left|\log \widehat{N}\left(\mathbf{z}, \mathbf{X}_1^{N_0}\right) - \log \frac{1}{P_\ell(\mathbf{z})}\right| < \epsilon\right\} \approx 1. \tag{5.2}$$

Based on (5.2), the classifier might work as follows. For each $\mathbf{z} \in \mathcal{A}^\ell$ it computes $\widehat{N}\left(\mathbf{z}, \mathbf{X}_1^{N_0}\right)$, and lets

$$\widehat{P}_\ell(\mathbf{z}) = 1/\widehat{N}\left(\mathbf{z}, \mathbf{X}_1^{N_0}\right) \tag{5.3}$$

be an estimate of $P_\ell(\mathbf{z})$. It then plugs this estimate into the equation for D_ℓ, to obtain

$$\widehat{D}_\ell(Q_\ell, P_\ell) \triangleq \sum_{\mathbf{z} \in \mathcal{A}^\ell} Q_\ell(\mathbf{z}) \log \frac{Q_\ell(\mathbf{z})}{\widehat{P}_\ell(\mathbf{z})}. \tag{5.4}$$

Finally, it then sets $f_c\left(\mathbf{X}_1^N, Q\right) = 1$ or 0 according as $\widehat{D}$ exceeds a threshold.

It turns out that this technique works, but with the following modification. Break the sequence $\mathbf{X}_1^{N_0}$ into K subsequences, where K is a constant that doesn't grow with ℓ. Then if $N_0 \approx 2^{H\ell}$, the length of each of the K substrings, N_0/K, has the same exponent as N_0. Then replace (5.3) by

$$\widehat{P}_\ell(\mathbf{z}) = \frac{1}{\max\limits_{1 \leq k \leq K} \widehat{N}\left(\mathbf{z}, \mathbf{X}_{(k-1)N_0/K+1}^{kN_0/K}\right)}, \tag{5.3$'$}$$

and use (5.4) to compute the estimate $\widehat{D}$. Complete details are given in [6].

Thus we see again how the typical set $T(\ell, \epsilon)$ is roughly the same as the collection of substrings of length ℓ of $(X_1, X_2, \ldots, X_{N_0})$ where $N_0 \approx 2^{\ell(H+\epsilon)}$.

Acknowledgement

I would like to express my thanks to the Information Theory Society for selecting me to present this Shannon Lecture. Receiving an award from the group that has been my intellectual home for over 30 years is especially gratifying. I owe a great deal to countless other workers in information theory, but I would like to make special mention of several to whom I am especially indebted.

David Slepian hired me into Bell Labs, and put me on my feet as a researcher. I also would like to publicly thank the following friends and colleagues, from whose help I benefited enormously: Jim Mazo, Larry Ozarow, Larry Shepp, Hans Witsenhausen, and Jack Wolf. Finally, to my long-time friend and collaborator Jacob Ziv, I extend an especially warm and grateful thank you. Without the help of these people, and many others in the information theory community and at Bell Labs, there is no doubt that I would not be presenting this lecture.

References

[1] Cover, T. and J. Thomas, *Elements of Information Theory*, Wiley, New York, 1991.

[2] Farach, M., M. Noordeweir, S. Savari, L. Shepp, A.J. Wyner, J. Ziv, "On the Entropy of DNA: Algorithms and Measurements based on Memory and Rapid Convergence", *Proceedings of the 1995 Symposium on Discrete Algorithms.*

[3] Gallager, R.G., *Information Theory and Reliable Communication*, Wiley, New York, 1968 (Theorem 3.5.3).

[4] Kac, M., "On the notion of Recurrence in Discrete Stochastic Processes", *Bull. of the Amer. Math. Soc.*, Vol. 53, 1947, pp. 1002-10010.

[5] Orenstein, D.S. and B. Weiss, "Entropy and Data Compression Schemes", *IEEE Transactions on Information Theory*, Vol. 39, Jan. 1993, pp. 78-83.

[6] Wyner, Aaron D. and Jacob Ziv, "Classification with Finite Memory", to appear in the IEEE Transactions on Information Theory.

[7] Wyner, Aaron D. and Jacob Ziv, "Some Asymptotic Properties of the Entropy of a Stationary Ergodic Data Source with Applications to Data Compression", *IEEE Transactions on Information Theory*, Vol. 35, Nov. 1989, pp 1250-1258.

[8] Wyner, Aaron D. and Jacob Ziv, "The Sliding-Window Lempel-Ziv Algorithm is Asymptotically Optimal", *Proceedings of the IEEE*, Vol. 82, June 1994, pp. 872-877.

[9] Wyner, Abraham J., "The Redundancy and Distribution of the Phrase Lengths of the Fixed-Database Lempel-Ziv Algorithm", submitted to the *IEEE Transactions on Information Theory.*

[10] Wyner, Abraham, J. "String Matching Theorems and Applications to Data Compression and Statistics", Ph.D. Thesis, Statistics Dept., Stanford University, June, 1993.

[11] Ziv, J. and A. Lempel, "A Universal Algorithm for Sequential Data Compression", *IEEE Transactions on Information Theory*, Vol. 23, May 1977, pp. 337-343.

[12] Ziv, J. and A. Lempel, "Compression of Individual Sequences by Variable Rate Coding", *IEEE Transactions on Information Theory*, Vol. 24, Sept. 1978, pp. 530-536.

Appendix

In this appendix we will give precise proofs of Theorem 2.4 and Theorem 2.5. We begin with

Proof of Theorem 2.4: For a given $\mathbf{z} \in \mathcal{A}^\ell$, define the binary random sequence $\{Y_i\}_{-\infty}^{\infty}$ by

$$Y_i \triangleq \begin{cases} 1, & \text{if } \mathbf{X}_{i+1}^{i+\ell} = \mathbf{z} \\ 0, & \text{otherwise.} \end{cases} \tag{A.1}$$

Then

$$\begin{aligned} &\Pr\left\{N(\mathbf{X}) = k \mid \mathbf{X}_i^\ell = z\right\} \\ &\quad = \Pr\left\{Y_{-k} = 1,\ Y_{-j} = 0 \ \text{ for } \ 1 \le j < k \mid Y_0 = 1\right\} \\ &\quad \triangleq Q(k). \end{aligned} \tag{A.2}$$

Write

$$\begin{aligned} 1 &\overset{(a)}{=} \sum_{k=1}^{\infty}\sum_{i=0}^{\infty} \Pr\left\{Y_{-k} = 1, Y_j = 0 \text{ for } -k < j < i, Y_i = 1\right\} \\ &= \sum_{k=1}^{\infty}\sum_{i=0}^{\infty} \Pr\left\{Y_i = 1\right\} \\ &\qquad \Pr\left\{Y_{-k} = 1, Y_j = 0 \text{ for } -k < j < i \mid Y_i = 1\right\} \\ &\overset{(b)}{=} \Pr\left\{Y_0 = 1\right\} \sum_{k=1}^{\infty}\sum_{i=0}^{\infty} Q(i+j) \\ &\overset{(c)}{=} \Pr\left\{Y_0 = 1\right\} \sum_{k=1}^{\infty} kQ(k) \\ &\overset{(d)}{=} \Pr\left\{\mathbf{X}_1^\ell = \mathbf{z}\right\} E\left(N(\mathbf{X}) \mid \mathbf{X}_1^\ell = \mathbf{z}\right). \end{aligned} \tag{A.3}$$

Step (a) follows from the ergodicity of $\{X_k\}$, which implies that with probability 1, $Y_n = 1$ for at least one $n < 0$ and one $n \ge 0$. Step (b) follows from the stationarity of $\{X_k\}$. Step (c) follows from the fact that $Q(k)$ appears in the left member of (c) exactly k times — for $(i,j) = (0,k), (1,k-1), \ldots, (k-1,1)$. Step (d) follows from (A.1) and (A.2). Eq. (A.3) is Theorem 2.5.

Before proving Theorem 2.7, we will give several lemmas. Let $\{\mathcal{E}_\ell\}_{\ell=1}^{\infty}$ be a sequence of events in a probability space. Define the events

$$[\mathcal{E}_\ell \text{ i.o.}] \triangleq \bigcap_{k=1}^{\infty} \bigcup_{n \ge k} \mathcal{E}_n, \tag{A.4a}$$

and

$$[\mathcal{E}_\ell \text{ a.a.}] \triangleq \bigcup_{k=1}^{\infty} \bigcap_{n \ge k} \mathcal{E}_n. \tag{A.4b}$$

$[\mathcal{E}_\ell \text{ i.o.}]$ is the event that $\mathcal{E}_\ell$ occurs infinitely often, and $[\mathcal{E}_\ell \text{ a.a.}]$ is the event that all but a finite number of the $\{\mathcal{E}_\ell\}$ occur. ("a.a." stands for "almost always".) The following is easy to prove.

Lemma A.1. *Let $\{C_\ell\}$ and $\{\mathcal{E}_\ell\}$ be sequences of events. If $P[\mathcal{E}_\ell \text{ a.a.}] = 1$ then $P[C_\ell \text{ i.o.}] \le P[C_\ell \mathcal{E}_\ell \text{ i.o.}]$.*

Next we observe that the strong form of the AEP (Theorem 2.1′) states that with probability 1,

$$\frac{1}{\ell} \log P_\ell\left(\mathbf{X}_1^\ell\right) \to H, \quad \text{as } \ell \to \infty. \tag{A.5}$$

Further a conditional form of the AEP states that with probability 1, as $\ell \to \infty$,

$$\frac{-1}{\ell} \log P_\ell \left(\mathbf{X}_1^\ell \mid \mathbf{X}_{-\infty}^0 \right) \to H. \tag{A.6}$$

(A.6) follows from the ergodic theorem on writing

$$\frac{-1}{\ell} \log P_\ell \left(\mathbf{X}_1^\ell \mid \mathbf{X}_{-\infty}^0 \right) = \frac{-1}{\ell} \sum_{i=1}^{\ell} \log P \left(X_i \mid \mathbf{X}_{-\infty}^0 \right)$$

$$\stackrel{\ell \to \infty}{\longrightarrow} E - \log P \left(X_i \mid \mathbf{X}_{-\infty}^0 \right) = H. \quad \text{(a.s.)} \tag{A.7}$$

For $\epsilon > 0$, and $\ell = 1, 2, \ldots$, let

$$B_\ell = \left\{ \mathbf{x}_1^\ell : \left| \frac{1}{\ell} \log \frac{1}{P(\mathbf{X}_1^\ell)} - H \right| \le \epsilon/2 \right\} \tag{A.8}$$

be the typical set defined in (2.3). ¿From Proposition 2.2

$$|B_\ell| \le 2^{\ell(H+\epsilon/2)}. \tag{A.9}$$

Also define a conditional version of B_ℓ, for $\epsilon > 0$, $\ell = 1, 2, \ldots$,

$$B'_\ell \triangleq \left\{ \mathbf{x}_{-\infty}^\ell : \left| \frac{1}{\ell} \log \frac{1}{P_\ell(\mathbf{X}_1^\ell \mid \mathbf{X}_{-\infty}^0)} - H \right| \le \epsilon/2 \right\}. \tag{A.10}$$

Note that (A.6) and (A.7) imply that

$$P[B_\ell \text{ a.a.}] = P[B'_\ell \text{ a.a.}] = 1. \tag{A.11}$$

We are now ready to begin the proof of Theorem 2.5. Define the events, for $\epsilon > 0$, $\ell = 1, 2, \ldots$,

$$A_\ell \triangleq \left\{ \frac{1}{\ell} \log N_\ell(\mathbf{X}) \ge H + \epsilon \right\}, \tag{A.12a}$$

$$A'_\ell \triangleq \left\{ \frac{1}{\ell} \log N_\ell(\mathbf{X}) \le H - \epsilon \right\}. \tag{A.12b}$$

Theorem 2.5 follows from the following lemmas.

Lemma A.2. $P[A_\ell \ i.o.] = 0$.

Lemma A.3. $P[A'_\ell \ i.o.] = 0$.

These lemmas imply that with probability 1,

$$\frac{1}{\ell} \log N_\ell(\mathbf{X}) \to H, \quad \text{as} \ \ \ell \to \infty, \tag{A.13}$$

which is the stronger form of Theorem 2.5.

Proof of Lemma A.2: Write

$$\begin{aligned} P(A_\ell B_\ell) &= \sum_{\mathbf{z} \in B_\ell} P_\ell(\mathbf{z}) \Pr \left\{ A_\ell \mid \mathbf{X}_1^\ell = \mathbf{z} \right\} \\ &= \sum_{\mathbf{z} \in B_\ell} P_\ell(\mathbf{z}) \Pr \left\{ N_\ell(\mathbf{X}) \ge 2^{\ell(H+\epsilon)} \mid \mathbf{X}_1^\ell = \mathbf{z} \right\} \\ &\stackrel{\text{(a)}}{\le} \sum_{\mathbf{z} \in B_\ell} P_\ell(\mathbf{z}) E \left(N_\ell(\mathbf{X}) \mid \mathbf{X}_1^\ell = \mathbf{z} \right) 2^{-\ell(H+\epsilon)} \\ &\stackrel{\text{(b)}}{=} \sum_{\mathbf{z} \in B_\ell} 2^{-\ell(H+\epsilon)} = 2^{-\ell(H+\epsilon)} |B_\ell| \stackrel{\text{(c)}}{\le} 2^{-\ell\epsilon/2}. \end{aligned} \tag{A.14}$$

Step (a) follows from the Markov inequality[†] step (b) from Theorem 2.4, and step (c) from (A.9). ¿From (A.12) $\sum_\ell P(A_\ell B_\ell) < \infty$, so that the Borel-Cantelli Lemma implies $P[A_\ell B_\ell \text{ i.o.}] = 0$. Thus (A.11) and Lemma A.1 (with $C_\ell = A_\ell$, $\mathcal{E}_\ell = B_\ell$) imply Lemma A.2.

[†] $\Pr\{|U| \ge a\} \le E|U|/a$, for $a \ge 0$.

Proof of Lemma A.3: First observe that $N_\ell(\mathbf{x})$is a function of $\mathbf{x}_{-\infty}^\ell$, so that from now we write $N_\ell = N_\ell(\mathbf{x}_{-\infty}^\ell)$. We condition on $\mathbf{X}_{-\infty}^0 = \mathbf{x}_{-\infty}^0$. Define the section of A'_ℓ.

$$A'_\ell(\mathbf{x}_{-\infty}^0) = \left\{ \mathbf{x}_1^\ell : \mathbf{x}_{-\infty}^\ell \in A'_\ell \right\}, \tag{A.15a}$$

and

$$B'_\ell(\mathbf{x}_{-\infty}^0) = \left\{ \mathbf{x}_1^\ell : \mathbf{x}_{-\infty}^\ell \in B'_\ell \right\}. \tag{A.15b}$$

Note that for a given $\mathbf{x}_{-\infty}^0$, $\mathbf{x}_1^\ell$ is determined by $N_\ell(x_{-\infty}^\ell)$; i.e. if $N_\ell(\mathbf{x}_{-\infty}^\ell) = N$, $\mathbf{x}_1^\ell = \mathbf{x}_{-N+1}^{-N+\ell}$. Thus there are no more than N, $\mathbf{x}_1^\ell$'s such that $N(\mathbf{x}_{-\infty}^\ell) \le N$. In particular,

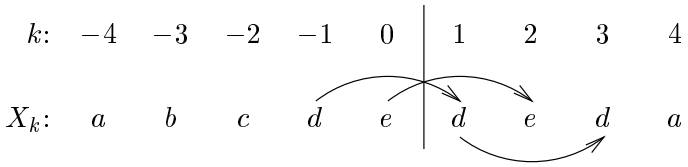
$$|A'_\ell(\mathbf{x}_{-\infty}^0)| \le 2^{\ell(H-\epsilon)}. \tag{A.16}$$

Now for a given $\mathbf{x}_{-\infty}^0$,

$$\begin{aligned} &\Pr \left\{ A'_\ell B'_\ell \mid \mathbf{X}_{-\infty}^0 = x_{-\infty}^0 \right\} \\ &\quad = \sum_{\mathbf{x}_1^\ell \in A'_\ell(\mathbf{x}_{-\infty}^0) B'_\ell(\mathbf{x}_{-\infty}^0)} P_\ell \left(\mathbf{x}_1^\ell \mid \mathbf{x}_{-\infty}^0 \right) \\ &\quad \stackrel{\text{(a)}}{\le} 2^{-\ell(H-\epsilon/2)} \left| A'_\ell(\mathbf{x}_{-\infty}^0) \right| \stackrel{\text{(b)}}{\le} 2^{-\ell\epsilon/2}, \end{aligned} \tag{A.17}$$

where step (a) follows from $\mathbf{x}_1^\ell \in B_\ell(\mathbf{x}_{-\infty}^0)$, and step (b) from (A.16). Ineq. (A.17) implies that $P(A'_\ell B'_\ell) \le 2^{-\ell\epsilon/2}$, so that from Borel-Cantelli, $P[A'_\ell B'_\ell \text{ i.o.}] = 0$, and from (A.11) and Lemma A.1, $P[A'_\ell \text{ i.o.}] = 0$. This is Lemma A.3.

Historical Note: Lemma A.2 was established in [7]. Lemma A.3 was found first by Orenstein and Weiss [5]. The proof given here of Lemma A.3 is new.


k: −4 −3 −2 −1 0 1 2 3 4
X_k: a b c d e d e d a

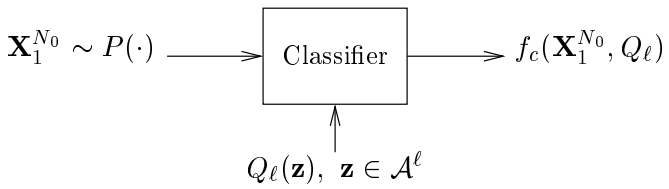

$\mathbf{X}_1^{N_0} \sim P(\cdot)$
Classifier
$f_c(\mathbf{X}_1^{N_0}, Q_\ell)$
$Q_\ell(\mathbf{z}),\ \mathbf{z} \in \mathcal{A}^\ell$

Figure 1: