Saint Mary's College of California

Department of Mathematics

Senior Thesis

Shannon's Coding Theorems

Author:
Camille Santos

Faculty Advisors:
Professor Michael Nathanson
Professor Andrew Conner
Professor Kathy Porter

May 16, 2016

1 Introduction

A common problem in communications is how to send information reliably over a noisy communication channel. With his groundbreaking paper titled *A Mathematical Theory of Communication*, published in 1948, Claude Elwood Shannon asked this question and provided all the answers as well. Shannon realized that at the heart of all forms of communication, e.g. radio, television, etc., the one thing they all have in common is *information*. Rather than amplifying the information, as was being done in telephone lines at that time, information could be converted into sequences of 1s and 0s, and then sent through a communication channel with minimal error. Furthermore, Shannon established fundamental limits on what is possible or could be acheived by a communication system. Thus, this paper led to the creation of a new school of thought called Information Theory.

2 Communication Systems

In general, a *communication system* is a collection of processes that sends information from one place to another. Similarly, a storage system is a system that is used for storage and later retrieval of information. Thus, in a sense, a storage system may also be thought of as a communication system that sends information from one place (now, or the present) to another (then, or the future) [3].

In a communication system, information always begins at a **source** (e.g. a book, music, or video) and is then sent and processed by an **encoder** to a format that is suitable for transmission through a physical communications medium, called a **channel**. Sometimes a disturbance, called **noise**, may be

present in the channel and introduces error to the information that is being transmitted. Finally, the sent information is then processed by a **decoder** to restore its original format, and then sent to the final user, called the **sink**. Such a communications system is modeled below in *Figure 2.1.*

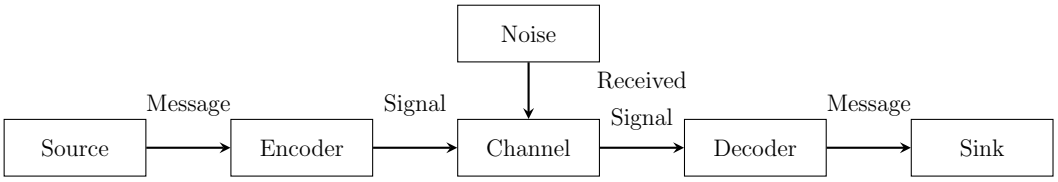


Figure 2.1: A basic communications system

2.1 The Information Source

An **information source** generates a finite sequence of symbols or letters called a **message**, denoted by $s_1 s_2 \cdots s_m$, where m is the total number of symbols in the message. The set of all possible symbols is called the **source alphabet** and is denoted by $S = \{s_1, s_2, \ldots, s_n\}$. Messages of length m will be denoted by t, and since the source alphabet contains n symbols, there are n^m possible messages. Thus, the set $T = \{t_1, t_2, \ldots, t_{n^m}\}$ is the set of all possible messages of length m. We assume that each symbol s_i of the source alphabet S has a certain probability p_i of occurring, or being sent, so $P(s_i) = p_i$ where $0 \leq p_i \leq 1$ for $i = 1, 2, \ldots, n$ and $\sum_{i=1}^{n} p_i = 1$.

We also assume that these probabilities remain constant over time, and hence we say the source is *stationary.* In addition, we assume that each symbol is sent independently of all previous symbols, so we also say the source is *memoryless.*

Example 2.2. Let the source be a fair coin. There are two possible outcomes of flipping the coin, heads (H) or tails (T). Thus we have $S = \{H, T\}$ with $P(H) = P(T) = \frac{1}{2}$.

Example 2.3. Let the source be a fair die. There are six possible outcomes of rolling the die, so $S = \{1, 2, 3, 4, 5, 6\}$ with $P(s_i) = \frac{1}{6}$ for $i = 1, 2, \ldots, 6$.

2.2 The Encoder

The output of a source is first processed by an **encoder**, which converts the message from one format to another, typically a binary stream, for more efficient transmission or storage of the information. The output of the encoder is referred to as a *signal*. There are two basic processes that an encoder can execute: source coding and channel coding.

The goal of **source coding** is to eliminate redundancy; that is, the parts of the information that can be removed while still guaranteeing exact reconstruction of the signal back to the original message that was sent by the source. This method of encoding is largely utilized in data compression. For example, consider the compression of audio data into MP3 files. In the original data, there exist audio waves or signals that are undetectable by the human ear. Removing these signals reduces the file size, and playback of the edited data still sounds the same to the human ear as the original data.

On the other hand, the goal of **channel coding** is to add or introduce extra redundancy in order to account for possible disturbances or *noise* that may affect the information during transmission or storage. The added

redundancy increases the length of the signal, but allows for greater detection and correction of errors during the decoding process. Common examples of channel coding include the use of parity check bits and repetition codes. A parity check bit is simply an extra bit that is added to a binary sequence such that there is an even number of 1's. If a sequence with an odd number of 1's is received, then the decoder can detect that an error has occurred during transmission through the channel. A repetition code simply repeats the message a certain number of times in hopes that the noise in the channel would corrupt only a small fraction of the signal. A common, everyday example is the case of human conversation wherein the listener cannot understand what is being said, so the speaker simply repeats what was said.

2.3 The Channel

After being processed by the encoder, the information, now in the form of a signal, then travels through a physical communications medium, which we call a **channel**. Some examples of a channel include fiber optic cables, coaxial cables, or in the case of storage, CD-ROMs or DVDs. Sometimes a disturbance, called **noise** is present in the channel and introduces error to the signal. A channel is said to be *noisy* if the channel output is not the same as the channel input. Common examples of noise include thermal noise on a telephone line or scratches on a CD-ROM that cause a song to skip during playback.

2.4 The Decoder and Sink

The output of the channel is then received by a **decoder**, which attempts to convert the received signal back to the original message. Then, finally, the output of the decoder is sent to the final user or destination, which is referred to as the **information sink**.

3 Information Theory

3.1 Information

Information theory is not concerned with the *meaning* of the information that is being transmitted, it is only concerned with the *amount* of information that is being transmitted, or that can be gained. It is typically the user that assigns meanings, whereas the computer merely interprets information according to how it is built and how it is programmed [1]. Ignoring the meaning of messages allows us to analyze and understand the system that processes the information. But how exactly do we quantify information? We will begin with an example.

Example 3.1. Going back to Example 2.2, let the source S be a fair coin with $S = \{H, T\}$ and $P(H) = P(T) = \frac{1}{2}$. Thus, the outcome is completely unpredictable, and there is a certain amount of information we gain about the coin by observing a given outcome, regardless of whether it is heads or tails. On the other hand, suppose we flip a weighted coin with $P(H) = \frac{9}{10}$ and $P(T) = \frac{1}{10}$. Before we even flip the coin, we are less uncertain about the outcome as we can predict that the outcome will be heads; hence if it does in fact land on heads, then there is not as much "surprise" or

information gained since we can already expect the outcome to be heads. On the other hand, if the outcome is actually tails, then there is more surprise and we gain a greater amount of information since the probability of occurrence for tails is so low.

To generalize this, consider a source with n symbols where $S = \{s_1, s_2, \ldots, s_n\}$. Suppose that each symbol s_i has a probability p_i of being sent, so $P(s_i) = p_i$ for $i = 1, 2, \ldots, n$. Let $I(p)$ be a function that quantifies the amount of information gained by receiving a symbol of probability p_i. We wish to define $I(p)$ such that the following properties are satisfied [1]:

(1) $I(p) \geq 0$;

(2) $I(p) = 0$ if $p = 1$;

(3) $I(p_1 \cdot p_2) = I(p_1) + I(p_2)$;

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

First of all, the output of $I(p)$ must be a real, nonnegative value since we cannot have a negative information gain, or in other words, we cannot lose information; receiving a symbol can only result in a gain of information, or no information at all. Second, we gain no information by receiving a symbol that is inevitable or guaranteed to be sent. This is because, similar to the weighted coin in Example 3.1, we already know the outcome prior to receiving a symbol or observing an event, so we do not gain any new information. On the other hand, if we receive a symbol that has a low probability of being sent, then we gain a lot more information. Condition (3) requires that the amount of information gained from

receiving two consecutive symbols with probabilities p_1 and p_2 independently is equal to the sum of the individual amounts of information for each p_i. Again going back to Example 3.1, the amount of information gained from flipping a single coin twice is equal to the amount of information gained from flipping two coins once.

Theorem 3.2. The only function to satisfy these properties is of the form

$$I(p) = -\log_b p = \log_b \frac{1}{p}$$

for some logarithm base b.

Proof. Consider the case where s_1 and s_2 are distinct symbols but have the same probability p_i, so $p_1 = p_2 = p$. By additivity in condition (3), we have

$$I(p_1 \cdot p_2) = I(p^2) = I(p \cdot p) = I(p) + I(p) = 2 \cdot I(p).$$

Now suppose $p_1 = p$ and $p_2 = p^2$. Then

$$I(p_1 \cdot p_2) = I(p^3) = I(p \cdot p^2) = I(p) + 2 \cdot I(p) = 3 \cdot I(p)$$

and by induction we find that

$$I(p^n) = n \cdot I(p)$$

for some positive integer n. We can also write $I(p)$ as

$$I(p) = I((p^{1/m})^m) = m \cdot I(p^{1/m})$$

for some positive integer m, and dividing both sides by m, we have

$$I(p^{1/m}) = \frac{1}{m} \cdot I(p)$$

So in general we have

$$I(p^{n/m}) = \frac{n}{m} \cdot I(p)$$

We can see that $I(p)$ follows the same formula as the logarithm function for rational numbers. By continuity in condition (4), we can extend this function over both rational and irrational numbers such that $0 \leq p \leq 1$, and hence

$$I(p) = k \cdot \log_b p$$

for some constant k and some logarithm base b. Since $\log_b p$ yields a negative value for any $0 < p < 1$ and condition (1) requires that $I(p)$ is nonnegative, we can choose $k = -1$ and finally we get

$$I(p) = -\log_b p = \log_b \frac{1}{p}$$

for some logarithm base b. □

The output of $I(p)$ is the amount of information gained from receiving a symbol with probability p. The log base b specifies the units of information we are using. Since $x = b^{\log_b x}$ for all $x > 0$, we can easily change the base using the formula

$$\log_a x = \frac{\log_b x}{\log_b a} = \log_a b \cdot \log_b x$$

and hence a change in base b is simply a change in units of information. Choosing base 2, the resulting units are called *bits*. For base e, the units are called *nats*, for natural log. If we use base 10, the unit is called a *Hartley*, named after the mathematician Ralph Hartley who first tried to define a measure for information.

The log base 2 is the most commonly used base in information theory, and will be used throughout the rest of this paper. Hence, rather than

writing $\log_2 p$, we will simply write $\log p$. It is important to note that in this case, the term *bit* is used as a *unit of information*, rather than the popular use as a *unit of storage*, where it is represented as either a 0 or 1. Thus in the context of information theory, a bit is also sometimes called a *Shannon* since Claude Shannon is the first person to use the term in print.

Example 3.3. Going back to Example 3.1 we can see that in the case of the fair coin,

$$I(\mathrm{H}) = I(\mathrm{T}) = I(\tfrac{1}{2}) = \log(1/\tfrac{1}{2}) = \log 2 = 1 \text{ bit}$$

so one flip of a fair coin conveys 1 bit of information. On the other hand, in the case of the weighted coin, we have

$$I(\mathrm{H}) = I(\tfrac{9}{10}) = \log(1/\tfrac{9}{10})) = \log \tfrac{10}{9} \approx 0.152 \text{ bits}$$

and

$$I(\mathrm{T}) = I(\tfrac{1}{10}) = \log(1/\tfrac{1}{10})) = \log 10 \approx 3.32 \text{ bits}$$

so we can see that if the outcome of the coin flip is heads, a significantly smaller amount of information is conveyed than if the outcome is tails, since $P(\mathrm{H})$ is closer to 1 and hence is more predictable.

3.2 Entropy

So far we have only been looking at the amount of information gained by receiving a single symbol. Shannon took into consideration the probabilities of all the symbols $s_1, s_2, \ldots, s_n$ of the source alphabet S and extended our previous definition to quantify the amount of information generated by S on average, per symbol.

Definition 3.4. [6] Let $S = \{s_1, s_2, \ldots, s_n\}$ be a source alphabet with probability distribution $P(s_i) = p_i$ for $i = 1, 2, \ldots, n$. The average amount of information per symbol, called the **entropy of the source**, is given by

$$H(S) = -\sum_{i=1}^{n} p_i \log p_i = \sum_{i=1}^{n} p_i \log \frac{1}{p_i}$$

The term entropy can be used synonymously with uncertainty, surprise, and information. First, there is the amount of uncertainty we have about a symbol prior to receiving it. Next, there is the amount of surprise we have after we receive the symbol. Finally, there is the amount of information we gain about the system by receiving the symbol.

Example 3.5. Again going back to Example 3.1, in the case of the fair coin we have,

$$H(S) = \frac{1}{2} \log 2 + \frac{1}{2} \log 2 = 1 \text{ bit per symbol}$$

and in the case of the weighted coin we have,

$$H(S) = \frac{9}{10} \log \frac{10}{9} + \frac{1}{10} \log 10 \approx 0.469 \text{ bits per symbol}$$

The graph of the binary entropy function for two symbols, or two probabilities, is shown below in Figure 3.6. Note that the entropy is maximized when the two probabilities are equal. The same result holds for the entropy of a source of n symbols with equal probability $p_i = \frac{1}{n}$ for $i = 1, 2, \ldots, n$, in which case the entropy function simplifies to $H(S) = \log n$.

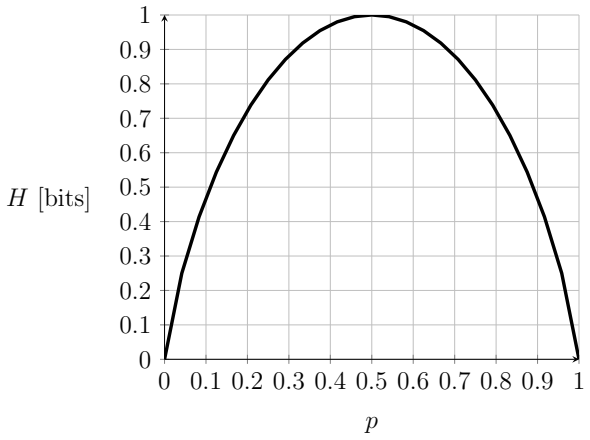


Figure 3.6: The Binary Entropy Function

3.3 The Discrete Memoryless Channel

Recall that after being generated by a source, information is then sent and transmitted through a physical communication medium, called a channel. A discrete memoryless channel (DMC) is often used to model channel communications.

Definition 3.7. [3] A **discrete memoryless channel**, denoted by $(A, P_{B|A}(b_j|a_i), B)$, consists of

1. an input alphabet $A = \{a_1, a_2, \ldots, a_q\}$, where q denotes the number of input symbols;

2. an output alphabet $B = \{b_1, b_2, \ldots, b_r\}$, where r denotes the number of output symbols; and

3. a conditional probability distribution $P_{B|A}(b_j|a_i)$, where $1 \leq i \leq q$ and $1 \leq j \leq r$, which specifies the probability of receiving the symbol b_j at the output given that the symbol a_i is sent.

Such a channel is referred to as *discrete* because it is comprised of two finite sets of letters or symbols, and *memoryless* because the noise present in the channel affects only the current input and hence is independent of all previous inputs. The input alphabet A represents the symbols a_i that are sent into the channel, and the output alphabet B represents the symbols b_i that are received from the channel. Sometimes the noise present in the channel corrupts the information in such a way as to introduce "new" symbols. Thus, the cardinalities of the sets A and B of input and output symbols, respectively, are not always equal. The conditional probabilities $P_{B|A}(b_j|a_i)$ can be represented using a matrix representation:

$$P = \begin{bmatrix} P(b_1|a_1) & P(b_2|a_1) & \dots & P(b_r|a_1) \\ P(b_1|a_2) & P(b_2|a_2) & \dots & P(b_r|a_2) \\ \vdots & \vdots & \ddots & \vdots \\ P(b_1|a_q) & P(b_2|a_q) & \dots & P(b_r|a_q) \end{bmatrix}$$

where P is called the *channel transition matrix*. Each row of P contains the probabilities of all possible outputs from the same input to the channel, whereas each column of P contains the probabilities of all possible inputs for a particular output from the channel. If the symbol a_i is transmitted, the probability that we receive an output symbol b_j must be 1, that is,

$$\sum_{j=1}^{r} P(b_j|a_i) = 1 \text{ for } i = 1, 2, \dots, q.$$

In other words, all the probabilities in each row of P must sum to 1.

A common example of such a channel is the *binary symmetric channel* (BSC), shown below in Figure 3.8. The input alphabet is $A = \{0, 1\}$ and, similarly, its output alphabet is $B = \{0, 1\}$. The crossover probability p is

the probability that a 0 becomes a 1, and vice versa, after being transmitted through the channel.

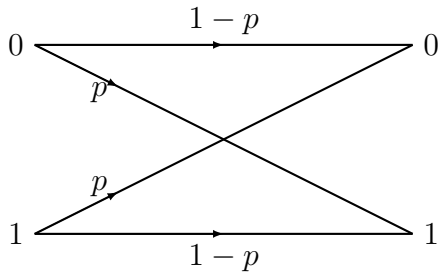


Figure 3.8: The Binary Symmetric Channel

Example 3.9. Consider the binary symmetric channel modeled above. Suppose the channel is noisy and that a bit is inverted 10% of the time. So $p = 0.10$ and $1 - p = 0.90$, and hence the channel matrix is given by

$$P = \begin{bmatrix} P(b_1|a_1) & P(b_2|a_1) \\ P(b_1|a_2) & P(b_2|a_2) \end{bmatrix} \begin{bmatrix} P(0|0) & P(1|0) \\ P(0|1) & P(1|1) \end{bmatrix} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix} = \begin{bmatrix} 0.90 & 0.10 \\ 0.10 & 0.90 \end{bmatrix}$$

3.4 Channel Probabilities

Assuming that a channel $(A, P_{B|A}(b_j|a_i), B)$ is stationary, wherein the conditional probabilities $P_{B|A}(b_j|a_i)$ do not change over time, these probabilities allow us to observe the behavior of a channel in the presence of noise. Before doing so, we must first observe the characteristics of the inputs and outputs of a channel. The probability that the i^{th} input symbol a_i and the j^{th} output symbol b_j occur simultaneously, that is, a_i is sent and b_j is received, is called the joint probability.

Definition 3.10. The **joint probability** of a_i and b_j is defined as

$$P_{A,B}(a_i, b_j) = P_{B|A}(b_j|a_i) \cdot P_A(a_i)$$

for $1 \leq i \leq q$ and $1 \leq j \leq r$.

Since each input symbol a_i is sent with probability $P_A(a_i)$, then the probability that the j^{th} output symbol b_j will be received, denoted $P_B(b_j)$, is given by

$$\begin{aligned} P_B(b_j) &= P_{B|A}(b_j|a_1) \cdot P_A(a_1) + P_{B|A}(b_j|a_2) \cdot P_A(a_2) + \ldots + P_{B|A}(b_j|a_q) \cdot P_A(a_q) \\ &= \sum_{i=1}^{q} P_{B|A}(b_j|a_i) \cdot P_A(a_i) \end{aligned}$$

which can be rewritten in terms of the joint probability of A and B as

$$P_B(b_j) = \sum_{i=1}^{q} P_{A,B}(a_i, b_j), \quad 1 \leq j \leq r.$$

We can now rewrite the joint probability of A and B as

$$P_{A,B}(a_i, b_j) = P_{A|B}(a_i|b_j) \cdot P_B(b_j)$$

and setting this equal to the original definition of joint probability results in

$$P_{A|B}(a_i|b_j) = \frac{P_{B|A}(b_j|a_i) \cdot P_A(a_i)}{P_B(b_j)}$$

which is known as ***Bayes' Theorem* on conditional probabilities**. The denominator $P_B(b_j)$ can be rewritten to give us the equation

$$P_{A|B}(a_i|b_j) = \frac{P_{B|A}(b_j|a_i) \cdot P_A(a_i)}{\sum_{i'=1}^{q} P_{B|A}(b_j|a_{i'}) \cdot P_A(a_{i'})}$$

where summing over all the input symbols a_i gives

$$\begin{aligned}\sum_{i=1}^{q} P_{A|B}(a_i|b_j) &= \frac{\sum_{i=1}^{q} P_{B|A}(b_j|a_i) \cdot P_A(a_i)}{\sum_{i'=1}^{q} P_{B|A}(b_j|a_{i'}) \cdot P_A(a_{i'})} \\ &= 1\end{aligned}$$

which means that given some output symbol b_j was received, some input symbol a_i was definitely sent into the channel.

3.5 Mutual Information

The difference in uncertainty before and after receiving b_j represents the amount of information that is gained by receiving b_j. We call this the **mutual information** where

$$\begin{aligned}I(a_i; b_j) &= \log \frac{1}{P_A(a_i)} - \log \frac{1}{P_{A|B}(a_i|b_j)} \\ &= \log \frac{P_{A|B}(a_i|b_j)}{P_A(a_i)}\end{aligned}$$

Since this is the mutual information of a single pair of input and output symbols (a_i, b_j), we average the mutual information over both alphabets A and B in order to determine the behavior of the whole channel.

Definition 3.11. The **system mutual information** is defined as

$$\begin{aligned}I(A; B) &= \sum_{i=1}^{q} \sum_{j=1}^{r} P_{A,B}(a_i, b_j) \cdot I(a_i; b_j) \\ &= \sum_{i=1}^{q} \sum_{j=1}^{r} P_{A,B}(a_i, b_j) \log \frac{P_{A,B}(a_i, b_j)}{P_A(a_i) \cdot P_B(b_j)}\end{aligned}$$

3.6 System Entropies

The entropy of the channel input alphabet A is given by $H(A) = \sum_{i=1}^{q} P_A(a_i) \log \frac{1}{P_A(a_i)}$, and similarly, the entropy of the channel output alphabet B is given by $H(B) = \sum_{j=1}^{r} P_B(b_j) \log \frac{1}{P_B(b_j)}$. This leads us to the following definition.

Definition 3.12. The **joint entropy** of A and B is defined as

$$H(A,B) = \sum_{i=1}^{q} \sum_{j=1}^{r} P_{A,B}(a_i, b_j) \log \frac{1}{P_{A,B}(a_i, b_j)}$$

which measures the total amount of uncertainty we have about the channel input and its corresponding output.

Note that if A and B are statistically independent, then $H(A,B) = H(A) + H(B)$. However, in most cases, the channel output b_j depends at least partly on the channel input a_i. In this case, we have $H(A,B) = H(A) + H(B|A)$, where

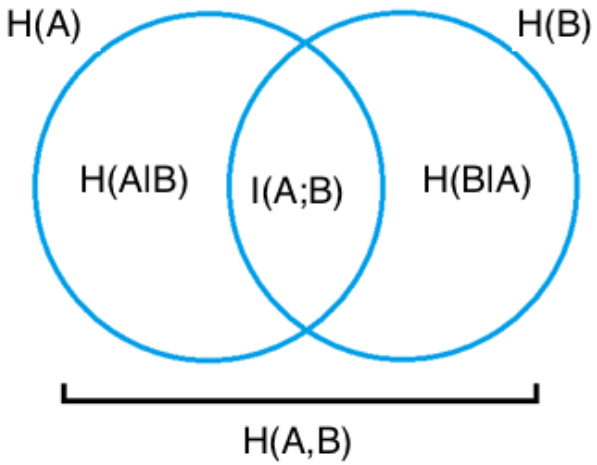
$$H(B|A) = \sum_{i=1}^{q} \sum_{j=1}^{r} P_{A,B}(a_i, b_j) \log \frac{1}{P_{B|A}(b_j|a_i)}$$

is called the conditional entropy of B given A. $H(B|A)$ is also referred to as the **equivocation**, or the **noise entropy** of the channel. It represents how much should be added to the input entropy in order to achieve or obtain the joint entropy. Alternatively, the joint entropy could also be written as $H(A,B) = H(B) + H(A|B)$.

Relating the system mutual information to the entropies of the system, we have the following equations that are equivalent:

$$I(A;B) = H(A) + H(B) - H(A,B)$$
$$I(A;B) = H(A) - H(A|B)$$
$$I(A;B) = H(B) - H(B|A)$$

where $H(A)$ is the uncertainty of the input before receiving B, $H(B)$ is the uncertainty of the output, and $H(A|B) = H(B|A)$ is the uncertainty of the input after receiving B. The relationships between the mutual information and system entropies can be seen in the figure below.



4 Shannon's Theorems

Definition 4.1. The **channel capacity C** is defined as

$$C = \max_{\{P_A(a)\}} I(A;B)$$

that is, the maximum mutual information over all possible input probabilities, $P_A(a)$.

The channel capacity quantifies how much information can be sent through the channel per use.

Example 4.2. Consider the binary symmetric channel described in Example 3.9. The channel capacity C is given by

$$\begin{aligned} C &= H(A) + H(B) - H(A,B) \\ &= 1 + 1 - (1 - p\log p - (1-p)\log(1-p)) \\ &= 1 + p\log p + (1-p)\log(1-p) \\ &= 1 + (.10)\log .10 + (.90)\log .90 \\ &\approx 0.531 \text{ bits} \end{aligned}$$

4.1 Source Coding

Theorem 4.3 (**Noiseless Channel Coding Theorem** [4])**.** Let a source have entropy H (bits per symbol) and a channel have capacity C (bits per second). Then it is possible to encode the output of the source in such a way as to transmit at the average rate $\frac{C}{H} - \epsilon$ symbols per second over the channel, where ϵ is arbitrarily small. It is not possible to transmit at an average rate greater than $\frac{C}{H}$.

Proof. Consider a source that produces a set of sequences of N symbols. If N is large, we can separate the symbols into two groups. The first group contains less than $2^{(H+\eta)N}$ symbols, while the second group contains less than 2^{RN} symbols (where R is the logarithm of the total number of possible symbols) with total probability less than μ. As N increases, η and μ approach zero. The total number of signals of length m in the channel is greater than $2^{(C-\theta)m}$ with small θ when m is large. We can choose

$$m = (\frac{H}{C} + \lambda)N$$

so that we have a large number of sequences of channel symbols for the group with high probability when N and m are large enough and λ is arbitrarily small. The group with high probability can be encoded using an arbitrary one-to-one map into this set. The remaining sequences can be represented by larger sequences, starting and ending with one of the sequences that has not been used for the group with high probability. A delay is allowed to give enough different sequences for the messages with low probability. This requires

$$m_1 = (\frac{R}{C} + \phi)N$$

where ϕ is small. The average rate of transmission in symbols per second will be greater than

$$[(1-\delta)\frac{m}{N} + \delta\frac{m_1}{N}]^{-1} = [(1-\delta)(\frac{H}{C} + \lambda + \delta(\frac{R}{C} + \phi)]^{-1}$$

As N increases, δ, λ, and ϕ approach zero and the rate approaches $\frac{C}{H}$. □

4.2 Channel Coding

Theorem 4.4 (**Noisy Channel Coding Theorem** [4])**.** Let a discrete channel have the capacity C and a discrete source the entropy per second H. If $H \leq C$ there exists a coding system such that the output of the source can be transmitted over the channel with an arbitrarily small frequency of errors (or an arbitrarily small equivocation). If $H > C$ it is possible to encode the source so that the equivocation is less than $H - C + \epsilon$ where ϵ is arbitrarily small. There is no method of encoding which gives an equivocation less than $H - C$.

Proof. Consider a source with entropy $H(A)$. We assume that the probability distribution achieves the maximum capacity C. Hence, $C = H(A) - H(A|B)$.

The total number of messages of length m that are highly likely to be sent is $T_A = 2^{m \cdot H(A)}$. Similarly, the remaining number of messages of length m are least likely to be sent and the total number of them is $T_B = 2^{m \cdot H(B)}$. Each output with high probability could be sent by $2^{m \cdot H(A|B)}$ inputs, and similarly, each input with high probability could result in $2^{m \cdot H(A|B)}$ outputs. All other cases have a low probability.

Now consider another source that is generating information at rate R with $R < C$. Then this source will have $2^{m \cdot R}$ messages with high probability. Suppose the output b_1 is received. There are $2^{m \cdot R}$ messages sent at random in $2^{m \cdot H(A)}$ points. The probability that a specific point is a message is $2^{m(R-H(A))}$, whereas probability that none of the points is a message is $P = [1 - 2^{m(R-H(A))}]^{2^{m \cdot H(A|B)}}$. We now have $R < H(A) - H(A|B)$, so $R - H(A) = -H(A|B) - \eta$ with $\eta > 0$. Now, as $m \to \infty$,

$$P = [1 - 2^{-m \cdot H(A|B) - m \cdot \eta}]^{2^{m \cdot H(A|B)}} \text{ approaches } 1 - 2^{-m\eta}.$$

Thus, the error probability approaches zero, proving the first part of the theorem.

To prove the second part, note that if $H(A) > C$, the remainder of the information that is generated will be neglected. Thus, the part that is neglected yields an equivocation $H(A) - C$, and the part that is sent only needs to add some positive ϵ.

Finally, to prove the last part of the theorem, suppose we can encode a source with entropy $H(A) = C + \alpha$ such that we get an equivocation $H(A|B) = \alpha - \epsilon$ with positive ϵ. Then we have $H(A) - H(A|B) = C + \epsilon$, which contradicts the definition of C as the maximum of $H(A) - H(A|B)$. $\square$

5 Conclusion

Information theory and coding theory are closely related in the sense that information theory establishes what is possible and can be done with information, whereas coding theory establishes ways to achieve these results. Information theory also has applications in investing, gambling, and game theory. Since it relies only on probabilities when given a partial amount of information, information theory can be applied to determine an optimal betting strategy or to maximize one's capital depending on the given odds of a situation. Overall, information theory has a profound impact on the way in which we communicate today, as it is the basic theory behind digital communication and storage. Without it, we may not have many of the technologies we use today such as fax machines, MP3 players, and especially digital wireless telephones.

References

[1] Hamming, R. W. (1980). *Coding and information theory* (2nd ed.). Englewood Cliffs, NJ: Prentice-Hall.

[2] Jones, G. A., & Jones, J. M. (2000). *Information and coding theory*. London, UK: Springer.

[3] Moser, S. M., & Chen, P. (2012). *A student's guide to coding and information theory*. Cambridge, UK: Cambridge University Press.

[4] Shannon, C. E., & Weaver, W. (1949). *The mathematical theory of communication*. Urbana, IL: University of Illinois Press.

[5] Togneri, R., & DeSilva, C. J. (2002). *Fundamentals of information theory and coding design*. Boca Raton, FL: Chapman & Hall/CRC.

[6] Van der Lubbe, J. C. (1997). *Information theory*. Cambridge, UK: Cambridge University Press.