Shannon's Noisy-Channel Coding Theorem

Lucas Slot Sebastian Zur

February 2015

Abstract

In information theory, Shannon's Noisy-Channel Coding Theorem states that it is possible to communicate over a noisy channel with arbitrarily small chance of error when the rate of communication is kept below a maximum which is constant for a channel. In this report we will first provide some basic concepts surrounding communication over noisy channels and then give a rigorous proof of the theorem. In conclusion we will provide some thoughts on practical applications of the theorem.

Contents

1 Introduction

The topic of this report is communication over a noisy channel. Informaly, this comes down to trying to send some form of information (for instance a stream of bits) over some channel (for instance an optic-fiber cable) that is *noisy*. What we mean by this is that even if we know the input, the output of our channel is not certain. For example our optic-fiber cable might have an impurity that results in sometimes outputting a 1 when a 0 is inputted. Rather than giving a fixed output for each input, our channel will instead have a probability to give each possible output that is dependent on the input. When we input a 0 into our optic-fiber we could for instance have an 80% chance to get a 0 zero as output, and a 20% chance to instead get a 1.

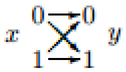
When communicating over these noisy channels it is inevitable that we will make mistakes. When trying to send a message to a friend over my cable some of the bits might get flipped, and he will end up getting the wrong message. It is not hard to come up with a crude way of limiting the chance of this happening. We could, for instance, instead of sending the bit we want to send once, send it 10 times. Our friend on the other side of the line would then simply count the amount of zeroes and ones in the output, and pick the most prevalent one. Though this simple scheme would severely limit the chance of a mistake, it would also cut the effective rate at which I can send information tenfold!

Interestingly enough, using the right scheme, it is possible to reduce the chance of error to arbitrarily low amounts, while retaining a decent rate of information transfer, specifically this rate is dependent only on the channel, and *not* on the error-bound we wish to achieve. Proving the existence of such a scheme will be the main objective of this report. Before we do that, however, we will first formalize the concept of communication over noisy channels.

2 Discrete Memoryless Channels and Coding

Definition 2.1 (Discrete Memoryless Channel)**.** A discrete memoryless channel consist of a random variable X, the *input*, over an alphabet $\mathcal{A}_X$, a random variable Y, the *output* over an alphabet $\mathcal{A}_Y$ and a conditional probability distribution $P_{Y|X}$ such that the chance of receiving output $y \in \mathcal{A}_Y$ given input $x \in \mathcal{A}_X$ is equal to $P_{Y|X}(y|x)$. Within this report we will always assume that $\mathcal{A}_X$ and $\mathcal{A}_Y$ are both finite.

Example 2.1. The following is an example of a discrete memoryless channel from MacKay[1], where $f \in [0, 1]$



$$\begin{aligned} P(y=0 \mid x=0) &= 1-f; & P(y=0 \mid x=1) &= f; \\ P(y=1 \mid x=0) &= f; & P(y=1 \mid x=1) &= 1-f. \end{aligned}$$

When $f = 0 \vee f = 1$, we know exactly what the output of the channel will be given a certain input, and the other way around. If $f \in (0, 1)$ we cannot be certain.

Definition 2.2 (Block Code)**.** Given a channel with input alphabet $\mathcal{A}_X$ a (N, K) block code $\mathcal{C}$ consist of $S := 2^K$ codewords $x^{(i)} = (x_1^{(i)}, x_2^{(i)}, \dots, x_N^{(i)}) \in \mathcal{A}_X^N$ and an *encoder* which maps each message in $\mathcal{S} := \{1, 2, ..., 2^K\}$ to a codeword. So when we want to send a message $s \in \mathcal{S}$, we send the corresponding codeword $x^{(s)}$ into the channel (by consecutively sending the $x_j^{(s)}$). The output of the channel $y \in \mathcal{A}_y^N$ will then be mapped back to $\hat{s} \in \{0, 1, 2, ..., 2^K\}$, where the extra symbol 0 is used to indicate a failure. This mapping is done by a *decoder*.

Definition 2.3. The rate R of an (N, K) block code is defined as

$$R := \frac{K}{N}.$$

Definition 2.4 (Block Error)**.** Given a channel, a message set $\mathcal{S}$, an encoder, a decoder and a probability distribution P_S on the message set, the probability of block error p_B is:

$$p_B = \sum_{s \in \mathcal{S}} P_S(s) P(\hat{s} \neq s | s).$$

If our channel is not noisy there exist an encoder and a decoder such that $p_B = 0$.

Definition 2.5 (Optimal Decoder)**.** An optimal decoder for a channel and code is the decoder that minimizes p_B, by maximizing $P(s|y)$, where:

$$P(s|y) = \frac{P(y|s)P_S(s)}{\sum_{s'} P_S(s')P(y|s')}.$$

Definition 2.6 (Bit Error)**.** When our message s is represented by K bits (since $s \in 1, 2, \dots, 2^K$), the probabilitiy bit error p_b is the average probability, that a bit in $\hat{s}$ is not equal to the corresponding bit in s.

3 Typicality

Codewords for messages will often consist of more than one symbol. Because of this we are interested not only in the behaviour of a single symbol when passed through a channel, but also in the behaviour of sequences of these symbols. Consider a discrete memoryless channel with input and output alphabets $\mathcal{A}_x$ and $\mathcal{A}_y$ respectively. For any $N \in \mathbb{N}$ we can simply extend the original definition of the channel to encorporate input $x = (x_1, x_2, \dots x_N) \in \mathcal{A}_x^N$ and output $y = (y_1, y_2, \dots y_N) \in \mathcal{A}_y^N$ by defining a new conditional probability as follows:

$$P(y|x) := P_{Y^N|X^N}(y|x) = \prod_{i=1}^{N} P_{Y|X}(y_i|x_i).$$

which is justified by the fact that the channel is memoryless, that is, each of the single-symbol communications are independent of one another. We will use this extension extensively in our proof of Shannon's theorem. Firstly, however, we will define some properties of sequences $x \in \mathcal{X}^N$, with respect to random variables over the alphabet $\mathcal{X}$.

3.1 Typicality

We will define a property called *typicality.* Intuitively, a sequence x will be typical with respect to a probability distribution P_X if P_X is relatively likely to produce x, that is, if we are not very *surprised* to see x as the result of drawing from a random variable $X \sim P_X$ a few times independently. Conversely, if x is not very likely to be the result of drawing from X, in other words if we are surprised to see x, it will not be typical.

Definition 3.1 (Observed Surprisal Value)**.** Let X be a random variable over an alphabet $\mathcal{X}$ with distribution P_X. Let $x = (x_1, x_2, \dots, x_N) \in \mathcal{X}^N$ be a sequence of length N. We will say

$$P(x) := P_{X^N}(x) = \prod_{i=1}^{N} P_X(x_i),$$

which is the chance to retrieve x from X after drawing independently N times. We now define the *observed surprisal value* $H_X(x)$ of x with respect to P_X as

$$H_X(x) := \frac{1}{N} \log \frac{1}{P(x)} = \frac{1}{N} \sum_{i=1}^{N} \log \frac{1}{P_X(x_i)}.$$

Definition 3.2 (Typicality)**.** Let X be a random variable over an alphabet $\mathcal{X}$ with distribution P_X. Let $x = (x_1, x_2, \dots, x_N) \in \mathcal{X}^N$ be a sequence of length N. We say x is typical with respect to P_X to tolerance β if

$$|H_X(x) - H(X)| < \beta.$$

Definition 3.3 (Typical Set)**.** Let X be a random variable over an alphabets $\mathcal{X}$ with distribution P_X . We define the *typical set* $T_{X,N,\beta}$ with respect to P_X of sequence pairs of length N as

$$T_{X,N,\beta} := \{x \in \mathcal{X}^N : x \text{ is typical with respect to } P_X \text{ to tol. } \beta\}$$

Example 3.1. Suppose X is the result of biased coin flip, with $P_X(1) = 0.6$ and $P_X(0) = 0.4$. The sequence

$$x := 1110011100$$

is typical with respect to P_X to any tolerance $\beta \geq 0$, as

$$\begin{aligned} |H_X(x) - H(X)| &= |\frac{1}{10} \log \frac{1}{P(x)} - h(0.6)| \\ &= |\frac{1}{10}(6 \log 0.6 + 4 \log 0.4) - (0.6 \log 0.6 + 0.4 \log(0.4))| \\ &= 0. \end{aligned}$$

Example 3.2. Suppose X is the result of a fair coin flip, so $P_X(1) = P_X(0) = 0.5$. Any sequence $x \in \{0,1\}^N$ is typical with respect to P_X to any tolerance $\beta \geq 0$. *Every sequence has the same chance to be the result of N coin flips!*

3.2 Joint Typicality

We will now extend the concept of typicality to joint distributions as this is done in MacKay[1].

Definition 3.4 (Joint Typicality)**.** Let X, Y be random variables over alphabets $\mathcal{X}$ and $\mathcal{Y}$ with distributions P_X and P_Y respectively. Let $x = (x_1, x_2, \dots, x_N) \in \mathcal{X}^N$ and $y = (x_1, x_2, \dots, x_N) \in \mathcal{Y}^N$ be sequences of length N such that x is typical to tolerance β with respect to P_X, and y is typical to tolerance β with respect to P_Y. We define the *joint oberserved surprisal value* as

$$H_{XY}((x,y)) := \frac{1}{N} \log \frac{1}{P(x,y)} = \frac{1}{N} \sum_{i=1}^{N} \log \frac{1}{P_{XY}(x_i, y_i)}.$$

The pair (x, y) is called *jointly typical* with respect to the joint distribution P_{XY} to tolerance β if

$$|H_{XY}((x,y)) - H(XY)| < \beta.$$

Once again, intuitively what this means is that the pair (x, y) is typical if it does not surprise us as the result of drawing from (X, Y) independently a few times.

Definition 3.5 (Jointly Typical Set)**.** Let X, Y be random variables over alphabets $\mathcal{X}$ and $\mathcal{Y}$ with distributions P_X and P_Y respectively. We define the *jointly typical set* $J_{N,\beta}$ with respect to P_{XY} of sequence pairs of length N as

$$J_{N,\beta} := \{(x,y) \in \mathcal{X}^N \times \mathcal{Y}^N : (x,y) \text{ jointly typical with respect to } P_{XY} \text{ to tol. } \beta\}$$

3.3 Joint Typicality Theorem

Observation. For any two random variables X, Y over $\mathcal{X}, \mathcal{Y}$, for any $N \in \mathbb{N}$ and $\beta > 0$ we have

$$\mathcal{X}^N \times \mathcal{Y}^N \supseteq T_{X,N,\beta} \times T_{Y,N,\beta} \supseteq J_{N,\beta}.$$

We formalise this observation in the following theorem, stated much like in MacKay[1]

Theorem 3.1 (Joint Typicality Theorem)**.** *Let $X \sim P_X$ and $Y \sim P_Y$ be random variables over $\mathcal{X}$ and $\mathcal{Y}$ respectively and let P_{XY} be their joint distribution. The following statements hold:*

1. *If $(x, y) \in \mathcal{X}^N \times \mathcal{Y}^N$ is drawn i.i.d from P_{XY}, so with probability distribution*

   $$P((x,y)) = \prod_{i=1}^{N} P_{XY}(x_i, y_i),$$

   the probability that (x, y) is jointly typical to tolerance β tends to 1 as N tends to ∞.

2. *The number of jointly typical sequence pairs* $|J_{N,\beta}| \leq 2^{N(H(X,Y)+\beta)}$

3. *For any two sequences* $x \in \mathcal{X}^N$ *drawn i.i.d from* P_X *and* $y \in \mathcal{Y}^N$ *drawn i.i.d from* $P_{Y|X}$ *we have*

$$P((x,y) \in J_{N,\beta}) \leq 2^{-N(I(X;Y)-3\beta)}$$

Proof. We prove each statement seperately, using Christian Schaffner's lecture notes[3] for part 2, and MacKay[1] for part 3.

1. By the law of large numbers the observed surprisal values $H_X(x), H_Y(y)$ and $H_{XY}((x,y))$ will tend to $H(X), H(Y)$ and $H(XY)$ respectively as N tends to ∞. This means for large N, $|H_X(x) - H(X)|, |H_Y(y) - H(Y)|$ and $|H_{XY}((x,y)) - H(XY)|$ will be very small, and so (x,y) will be jointly typical.

2. We have

$$\begin{aligned} 1 &= \sum_{(x,y) \in \mathcal{X}^N \times \mathcal{Y}^N} P((x,y)) \\ &\geq \sum_{(x,y) \in J_{N,\beta}} P((x,y)) \\ &\geq |J_{N,\beta}| 2^{-N(H(XY)+\beta)} \end{aligned}$$

   which implies

$$|J_{N,\beta}| \leq 2^{N(H(XY)+\beta)}$$

3. We have

$$\begin{aligned} P((x,y) \in J_{N,\beta}) &= \sum_{(x,y) \in J_{N,\beta}} P(x)P(y) \\ &\leq |J_{N,\beta}| 2^{-N(H(X)-\beta)} 2^{-N(H(Y)-\beta)} \\ &\leq 2^{N(H(XY)+\beta)-N(H(X)+H(Y)-2\beta)} && \text{(using part 2)} \\ &= 2^{-N(I(X;Y)-3\beta)} \end{aligned}$$

□

4 Shannon's Noisy-Channel Coding Theorem

Below is the main theorem of this report, following the formulation of MacKay [1]

Theorem 4.1 (Shannon's Noisy-Channel Coding Theorem)**.** *For any discrete memoryless channel with input X and output Y the following statements hold:*

1. *The channel capacity*
$$C := \max_{P_X} I(X;Y)$$
satisfies the following property. For any $\epsilon > 0$ and rate $R < C$, for sufficiently large N, there is a code of length N and rate $\geq R$ and a decoding algorithm, such that the maximimal probability of block error is $< \epsilon$.

2. *If we accept bit error with probability p_b, it is possible to achieve rates up to $R(p_b)$, where*
$$R(p_b) := \frac{C}{1 - h(p_b)}.$$

3. *Rates greater than $R(p_b)$ are not achievable without having a higher probability of bit error than p_b.*

4.1 Proof of the First Part

Proof. We will give a restructured version of the proof in MacKay [1]. We start off by proving the following lemma

Lemma 4.2. *Given a channel with input $X \sim P_X$ over $\mathcal{X}$ and output Y over $\mathcal{Y}$ defined by $P_{Y|X}$. For any $\epsilon > 0$ and rate $R < I(X;Y)$, for sufficiently large N, there is a code of length N and rate $\geq R$ and a decoding algorithm, such that the maximimal probability of block error is $< \epsilon$.*

Proof. Suppose we have the message set $\mathcal{S} = \{1, 2, \ldots, 2^{NR'}\}$. We will assign a codeword $x^{(s)} = (x_1^{(s)}, x_2^{(s)}, \ldots, x_N^{(s)}) \in \mathcal{X}^N$ of length N to each message s at random according to

$$P(x^{(s)} = (x_1, x_2, \ldots x_N)) = \prod_{i=1}^{N} P_X(x_i).$$

Note that this code has a rate of $\log(2^{NR'})/N = R'$. Using this code to encode our keywords the output of the channel will be $y = (y_1, y_2, \ldots, y_N)$, where

$$P(y|x^{(s)}) = \prod_{i=1}^{N} P_{Y|X}(y_i|x_i^{(s)}).$$

We will decode using *typical-set decoding* meaning we will decode y as $\hat{s}$ if $(x^{(\hat{s})}, y)$ are jointly typical and there is no other message s' such that $(x^{(s')}, y)$ are jointly typical. If y is not jointly typical with any codeword, or if y is jointly typical with multiple different codewords, we will decode as an error. Now that we have established a method of encoding and decoding, we will define three types of errors we will analyse for our method. For simplicity we will assume that we select messages to send uniformly, this will not affect the generality of the proof.

Definition 4.1. For a code $\mathcal{C}$, generated using the method defined above we define

1. The probability of block error:

$$p_B(\mathcal{C}) \equiv \frac{1}{2^{NR'}} \sum_{s \in \mathcal{S}} P(\hat{s} \neq s | \mathcal{C})$$

2. The maximal probability of block error:

$$p_{BM}(\mathcal{C}) \equiv \max_{s \in \mathcal{S}} P(\hat{s} \neq s | s, \mathcal{C})$$

and additionally the average[1] probability of block error

$$\bar{p_B} \equiv \sum_{\mathcal{C}} p_B(\mathcal{C}) P(\mathcal{C})$$

.

It is now time to find an upper bound for the average probability of block error. Because we constructed a codeword in the same (symmetrical) way for each message, we may assume without loss of generality that we always send message 1. Suppose now we have sent message 1 through the channel and we are trying to decode. There are two ways to make an error. The first one is for $x^{(1)}$ and y not to be jointly typical. We know however, by the first part of the joint typicality theorem, that for any $\delta > 0$ there exists an N_δ such that

$$\forall N > N_\delta : P((x^{(1)}, y) \notin J_{N,\beta}) < \delta.$$

The second way of making a mistake is for a codeword $x^{(k)}, k \neq 1$ to be jointly typical with y. We know by the third part of the joint typicality theorem that the chance for this to happen for any single codeword is $\leq 2^{-N(I(X;Y)-3\beta)}$. Combined with the fact that there are $2^{NR'} - 1$ competitors we find that there exists an N_δ such that for any $N > N_\delta$

$$\begin{aligned} \bar{p_B} &\leq P((x^{(1)}, y) \notin J_{N,\beta}) + \sum_{s'=2}^{2^{NR'}} P((x^{(s')}, y) \in J_{N,\beta}) \\ &\leq \delta + 2^{-N(I(X;Y)-R'-3\beta)} \end{aligned}$$

Where the first inequality holds because of the *union bound*. Now if $R' < I(X;Y) - 3\beta$, the expression $I(X;Y) - R' - 3\beta$ will be positive. This being the case, there exists an M_δ for each $\delta > 0$ such that for $M > M_\delta$

$$2^{-M(I(X;Y)-R'-3\beta)} < \delta$$

[1] Though perhaps *expected* probability of block error would be more correct

And so for each $\delta > 0$, there exists an $L_\delta = \max(N_\delta, M_\delta)$ such that for any $N > L_\delta$ we have

$$\bar{p_B} < 2\delta.$$

Since the *average* probability of block error $< 2\delta$ there must exist at least one code $\mathcal{C}'$ such that $p_B(\mathcal{C}') < 2\delta$.[2] For this code, we will call the $2^{NR'}/2$ messages least likely to produce an error $\mathcal{S}_1$. Now suppose there exists an $s_1 \in \mathcal{S}_1$ such that the probability of error when sending s_1 using $\mathcal{C}'$ is $\geq 4\delta$. This implies that for each messages s_2 in $\mathcal{S} - \mathcal{S}_1$ the probability of error when sending $s_2 \geq 4\delta$. But this would imply that $\bar{p_B} \geq 2\delta$, which is a contradiction. We conclude that for messages in $\mathcal{S}_1$ the probability of error is $< 4\delta$. We now use the messages in $\mathcal{S}_1$ and the codewords assigned to them in $\mathcal{C}'$ to produce a new code $\mathcal{C}^*$. This code has a rate of $R' - \frac{1}{N}$ (which is arbitrarily close to R' for large N) and $P_{BM}(C^*) < 4\delta$. Choosing $R' = \frac{R+I(X;Y)}{2}$ [3], $\delta = \frac{\epsilon}{4}, \beta < \frac{(I(X;Y)-R')}{3}$ and N large enough to satisfy the bounds we made earlier we have constructed a code of rate $\geq R$ with maximal probability of error $< \epsilon$ and thus proven the lemma. □

From this lemma we can now easily proof the first part of the theorem, by simply choosing P_X such that $I(X;Y) = C$. □

4.2 Proof of the Second and Third Part

For the second and thid part of the theorem we use a proof from Cramer and Fehr[2] First, we will prove the following lemma:

Lemma 4.3. *Let $Y^N = (Y_1, Y_2, ..., Y_N)$ be the output of a channel using an (N, K) block code with probability of bit error $p_b \in (0, 1)$. Let $S \sim P_S$ be the random variable that denotes which message was sent. We have*

$$h(p_b) \geq \frac{H(S|Y^N)}{K}$$

Proof. We find that:

$$\begin{aligned}
h(p_b) &= h\Big(\frac{\sum_{i=1}^{K} P(\text{bit error on bit number i})}{K}\Big) && \text{(definition of } p_b\text{)} \\
&\geq \frac{\sum_{i=1}^{K} h(P(\text{bit error on bit number i}))}{K} && \text{(Jensen's Inequality (h is concave))} \\
&\geq \frac{\sum_{i=1}^{K} H(\text{bit number i in message}|Y^N)}{K} && \text{(Fano's Inequality)} \\
&\geq \frac{H(S|Y^N)}{K}. && \text{(Chain Entropy Bound)}
\end{aligned}$$

[2] This is true because

$$2\delta > \sum_{\mathcal{C}} p_B(\mathcal{C}) P(\mathcal{C}) \geq \sum_{\mathcal{C}} \min_{\mathcal{C}^*}(p_B(\mathcal{C}^*) P(\mathcal{C}) = \min_{\mathcal{C}^*} p_B(\mathcal{C}^*).$$

[3] which is smaller than $I(X;Y) - 3\beta$!

□

With this lemma we are now ready to prove the second and third part of the Noisy-Channel Coding Theorem.

Proof. Let Y^N an S be as in the lemma. Additionaly let $X^N = (X_1, X_2, \dots X_N)$ be the the channel input. We have

$$\begin{aligned} H(X^N Y^N S) &= H(X^{N-1} Y^{N-1} S) + H(X_N | X^{N-1} Y^{N-1} S) + H(Y_N | X^{N-1} Y^{N-1} S) \\ &= H(X^{N-1} Y^{N-1} S) + H(Y_N | X_N), \end{aligned}$$

where the first equality follows from the chain rule. The second equality follows from the fact that X_N is known, if the other symbols and the original message are known. We do assume here that every message has a unique codeword, that every codeword is used only once, and that the Block Code is known. Also, since channel uses are independent, Y_N only depends on X_N. Repeating this procedure, we will eventually get:

$$H(X^N Y^N S) = H(S) + \sum_{i=1}^{N} H(Y_i | X_i).$$

This is not the only equality we can derive from the chain rule, as we also have

$$H(X^N Y^N S) = H(Y^N S) + H(X^N | Y^N S) = H(Y^N S).$$

Once again, if we know the original message, we will know the set of codewords, so now we can write the mutual information between the original message and the channel output as:

$$\begin{aligned} I(S; Y^N) &= H(S) + H(Y^N) - H(Y^N S) \\ &= H(Y^N) - \sum_{i=1}^{N} H(Y_i | X_i) \\ &\leq \sum_{i=1}^{N} H(Y_i) - H(Y_i | X_i) \\ &= \sum_{i=1}^{N} I(Y_i; X_i) \\ &\leq NC = \frac{NK}{R}. \end{aligned}$$

Using our lemma and assuming the bits in our message are uniformly distributed, we can write:

$$h(p_b) \geq \frac{H(S | Y^N)}{K} = \frac{H(S) - I(S; Y^N)}{K} \geq \frac{K - \frac{NK}{R}}{K} = 1 - \frac{C}{R}.$$

Rewriting this will give us:

$$R \leq \frac{C}{1 - h(p_b)}.$$

Now because all the bounds we used can be sharp we have shown that rates up to

$$R(p_b) = \frac{C}{1 - h(p_b)}.$$

are achievable and that rates above $R(p_b)$ are not achievable. □

5 Concluding Remarks

Shannon's Theorem is a strong theoretical result. The first part especially proves the existence of a coding method that seems counter-intuitive: it keeps on reducing the chance of error without reducing the transmission rate below a preset value. The second and third part provide strict bounds on communications where certain chances of error are acceptable.

It is not easy to apply the (first part of the) theorem in practice in a direct way. Four main problems hinder our ability to do this. Firstly, while possible, it can be computationally intensive to calculate the maximal rate C (and it's corresponding distribution P_X). Secondly, once we have found C and P_X, it is not easy to find the promised code $\mathcal{C}$. Neither the theorem nor it's proof provide any clue on how to find it, and though brute-forcing over finite alphabets is of course possible, it can once again be very demanding, especially because the codeword length N might be very large. This brings us to the third concern: large codeword lengths. The theorem provides no bounds whatsoever on the amount of channel uses we need for our code to start working (that is, achieve it's promised rate). N might simply be too large to be feasable in practical applications. Finally, discrete memoryless channels, the only type of channels the theorem tells us anything about, are rare in the real world. Most physical channels have at least some sort of memory (though this may be negligable in some cases).

Nonetheless these results are not useless in practice. For us personally, the idea of being able to demand stricter error-bounds without giving up transmission rate is an interesting concept on its own. The given bounds might also be seen as something of a goal: go try and achieve in practice what was proven possible in theory. Lastly, they give a sense of scale. A researcher working on encoding schemes that reach a rate of say 90% of C knows that he will not be able to improve his design much further.

References

[1] David J.C. MacKay - *Information Theory, Inference, and Learning Algorithms* - Cambridge University Press 2003 - Accessed via `http://www.inference.phy.cam.ac.uk/itprnn/book.pdf`

[2] Ronald Cramer and Serge Fehr - *The Mathematical Theory of Information, and Applications (Version 2.0)* - lecture notes - Accessed via `https://projects.cwi.nl/crypto/docs/InfTheory2.pdf`

[3] Christian Schaffner's lecture notes (blackboard photos) - `http://homepages.cwi.nl/~schaffne/courses/inftheory/2014/blackboard`