# **Introduction**

Probability theory and probabilistic models of random processes lend themselves to major applications in many fields of scientific research, from statistics and finance to artificial intelligence and machine learning to physics ”thermo dynamics and quantum physics”,

Of particular interest to the student of communication systems are the ideas from a branch of mathematics related to probability theory: “information theory”,   
their usefulness becomes evident for example in the case of a digital signal source, emitting symbols associated with probabilities.

Probability theory provides a basis for the formal treatment of a range of topics and concepts belonging to information theory, The entry point to such a treatment is a definition of the abstract concept of “Information”, which utilizes the “probability” concept encountered in the study of probability theory, thus any further employment of the “information” concept in other abstractions or applications, which is indeed the case for the ideas of information theory discussed here, implies a strong relation between such ideas and the ones from probability theory.

With this in mind, and with the widespread usage of information theory methods and concepts in modern computer and communication systems, we can consider information theory as a branch of applied mathematics.

Motivations for such work on information theory include the economic transmission and storage of digital information, both motives have driven great figures in this field to create and develop methods aimed at reducing the resources needed to transmit or store a given amount of data or information “source coding”, another motive is increasing the integrity of information transmission over noisy channels “channel coding”, consequently, various methods of data compression and error detection have been devised .

Information theory was developed by Claude E. Shannon in the 1940’s,  
 in 1948 he published a paper “a mathematical theory of communication” considered the inception of the theory, Shannon borrowed ideas developed by Ralph Hartley “who considered information a measurable quantity” and the notion of “entropy” from thermodynamics which was developed by   
L. Boltzmann and J. W. Gibbs.

+++ add some general description and historical background for other concepts researched. \_\_marwan.

## **Information**

*“Information is the resolution of uncertainty” ~Claude E. Shannon*

The definition of information  
most relevant to a theoretical study is a measurable quantity which is related to the probability of an outcome of a random experiment,

For example: a source transmitting a sequence of symbols, each transmission is a random experiment, the outcome (S**i**) of which corresponds to a particular probability P(S**i**), there is a finite set of possible outcomes (or symbols) each with a specific probability of occurrence, thus we have a complete probability model of the transmission experiment, we define the information obtained from an experiment as:



A unit of measurement for information is the “Bit” an acronym for (binary information unit) not to be confused with the binary digit, although we also use this quantity to specify the minimum number of binary digits required to transmit a symbol.

From the definition it is evident that the more likely a symbol is to occur, the less information it gives us and subsequently the less resources it should take to be transmitted (ideally).

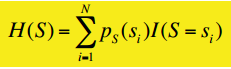
A curious consequence of the axioms of probability on our definition of information is the fact that information of independent outcomes is additive, to put it in a different way” if the sequence of symbols transmitted by a source consists of independent and identically distributed symbols, then the information obtained from the whole sequence is the sum of the information obtained from each symbol alone”, this can be easily proved using the properties of logarithms and probabilities of independent events  
(if the events A,B,C are mutually independent then P(ABC) =P(A)\* P(B)\* P(C) ).

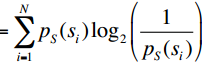
# **Entropy**

It is the expected value of information obtained from learning the outcome of a random experiment with a known probability model.

An example of such an experiment is a digital signal source emitting a single symbol at a time, the symbol belongs to a finite set of symbols all having associated probabilities.

The formal expression for obtaining this quantity is:





Where N: the number of symbols in a finite set of symbols available to the source.

The use of a probabilistic notion is clear, the notion of expectation (mean),   
here it is the weighted sum of the information from each possible symbol.

# **The significance of entropy**

The idea of entropy becomes important when dealing with efficient binary encoding of symbols.

Binary encoding is the process of assigning binary codes to each symbol in order to be transmitted over a digital communication channel.

Efficient here means exploiting the redundancy in the source alphabet (set of possible symbols with varying probabilities) to produce a varying length encoding for each symbol that depends on its associated probability, with the result of reducing the overall binary digits required to transmit a message (a sequence of symbols with specific length).

The degree of optimization of the message length has a lower limit equal to the entropy of the source, various encoding schemes have been devised in an effort to approach that boundary.

To relate entropy to encoding efficiency we can look at it this way:

Entropy, measured in bits, tells us the average amount of information required to resolve the uncertainty about an outcome of an experiment,  
 equivalently, it is average number of binary digits required to transmit a single symbol, if a fewer binary digits are transmitted for each symbol on average, there would be uncertainty about that symbol meaning (undetermined).

# **Fixed length vs. variable length encoding schemes**

The easiest way to encode a message comprised from a set of symbols is to encode each symbol (regardless of its probability) with a fixed length code, i.e. for the case of the English language there would be 27 symbols (26 letters + space) requiring a 5 bit code for each symbol,   
so to transmit a 1000 symbol message we need 5000 bits transmitted.

With the knowledge of the probability weights of the English letters (which can depend on context also), we can calculate the entropy (the lower limit) of this kind of source, in fact Shannon calculated the entropy of English letters, a recent estimate puts the value between 1 and 1.5 bits of information per symbol,   
consequently, the minimum number of binary digits to encode a 1000 letter message goes down to 1500 bits, the way we can approach this lower limit is by using a variable length encoding scheme that can exploit the fact that more probable symbols convey much less information than unlikely symbols, thus requiring less space and resources(binary digits).

An example of such scheme is the Huffman coding.

# **Huffman coding**

A development upon the fixed encoding scheme was devised by an MIT graduate in 1951, the Huffman code by David Huffman, it is a simple yet powerful algorithm that generates an optimal encoding provided a set of iid symbols, in its simplest form, the encoding is done per symbol, for more refined results (shorter message lengths) encoding pairs or triples of symbols can be chosen.

It’s important to note that the Huffman technique is not adaptive, meaning it wouldn’t be optimal if the symbol probabilities change from a message to the next.

A description of the algorithm in its simplest form (per symbol encoding) follows:

-a prerequisite is the mapping of all symbols available to the source to their probabilities, it’s assumed implicitly that this mapping is static (doesn’t change from a message to the next).

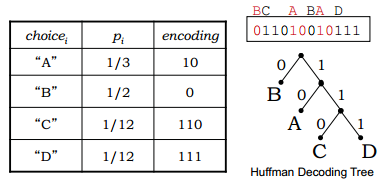
-1: choose two symbols with least probabilities.

-2: construct a tree diagram with those two symbols as leafs.

-3: merge both symbols and add their probabilities, this new symbol is the node linking the two in the diagram.

-4: repeat the previous steps until all symbols are included on the tree.

-5: assign 0’s to symbols on the leafs of the tree except one of the last two leafs,   
then assign 1’s to branches.



# **Limitations to Huffman coding**

But in reality, a complete probability model of the source may not be known, also this model might change with time and a static encoding won’t be much of a use in such case,  
another useful property which can’t be exploited by the Huffman technique , is the dependence of symbols upon previously transmitted symbols (context) ,this can in theory reduce the entropy of the source significantly, as the symbols become more predictable from their context, i.e. the case with English words and letters which are mostly used together like ‘q’ and ‘u’.

For this reason, an adaptive variable length encoding scheme is needed to exploit such redundancy in sources.

# **Lempel-Ziv-welch compression technique**

A lossless compression of symbol sequences which is adaptive to the context of the symbols, i.e. has memory, is the technique developed by ziv and Lempel, and improved by welch , this method is widely used to compress text, images(gif ,png ,tiff),generic files(zip).

A great advantage is that the source probability model neither has to be known or constant over time.

The core idea of it is that what is actually transmitted through the channel is a fixed-length code representing dictionary addresses (=array indices), both the receiving and the sending systems have a data structure (array/dictionary) that is dynamically built as the message is sent, starting from a core collection of symbols, both dictionaries are built simultaneously from symbol sequences “strings” within the message, those sequences are then referenced wholly by indices instead of referencing each constituting symbol alone, this can be of great usefulness in the case of English text for example, where many words are repeated over and over throughout the message and can be considered redundancies (don’t add information).

The algorithm for the emitter and receiver of the message  
(equivalently the compressor/decompressor )  
ensures that the dictionaries on both sides are mirrored (identical).

\_\_there’s a good video on “computerphile” on lzw compression, I don’t recall the name\_\_\_marwan