

On Ideal Code Length

Glen G. Langdon Jr.
Department of Computer Engineering
University of California
Santa Cruz, CA 95064
langdon@cse.ucsc.edu

Derivation of the NlogN Formula

Consider sequence \mathcal{S} , a sequence of symbols drawn from M-symbol alphabet $\mathcal{A} = \{a_1, a_2, \dots, a_m, \dots, a_M\}$.

The definition of the **self-information**, denoted $il(a_m)$, of a given symbol a_m , is the negative \log_2 of its probability when we use bits as the units of self-information: $-\log_2 p(a_m)$ or (which is the same by a mathematical identity): $\log_2 \frac{1}{p(a_m)}$.

Assumption

Let us assume we know the source probabilities $p(a_m)$. We can now calculate the self-information $il(a_m)$. Recall, if the *code lengths* for each symbol equal the *self-information* for that symbol, the compression *performance* achieves the *ideal* performance. The *entropy* value in bits per symbol is the ideal compression achievable. The duality between a *probability* and its respective *self-information* is expressed in Eq 1.

$$il(a_m) = \log_2 \frac{1}{p(a_m)} = -\log_2 p(a_m) \quad (1)$$

The assumptions for Eq 1 for the probability $p(a_m)$ can also apply to any probability. We can apply Eq 1 to a sequence as shown below.

Let $p(S_1^N)$ be the probability of an *i.i.d.* (independent and identically distributed) sequence \mathcal{S} of length N , by (a) the assumption of *independence* and (b) its associated *product rule*. Index k is the position of the symbol s_k in \mathcal{S} .

$$p(S_1^N) = \prod_{k=1}^N p(s_k) \quad (2)$$

The self-information $il(S_1^N)$ of a probability such as $p(S_1^N)$ is $-\log_2 p(S_1^N)$:

$$il(S_1^N) = -\log_2 \prod_{k=1}^N p(s_k). \quad (3)$$

The self-information value $il(S_1^N)$ calculated in Eq 3 is the *ideal length* IL (or ideal code length) for sequence \mathcal{S} . Shannon has shown that if sequence \mathcal{S} is indeed i.i.d., the sequence cannot be compressed with fewer bits than the ideal length il (in bits) without a loss of information. So, in Eq 3, $il(\mathcal{S})$ is synonymous with its *ideal code length* $IL(\mathcal{S})$.

The log of a *product* of factors is the sum of the log of each factor. In Eq 4, we calculate the self-information of a sequence using a summation (instead of a product as done in Eq 3).

$$IL(S_1^N) = il(S_1^N) = \sum_{k=1}^N -\log_2 p(s_k) \quad (4)$$

We note the each term of sum of Eq 4 is the self-information $il(p(s_k))$ for each symbol appearing in S_1^N . We can rewrite Eq 4 as Eq 5:

$$il(S_1^N) = \sum_{k=1}^N il(p(s_k)). \quad (5)$$

Note that sequence $il(S_1^N)$ of Eq 5 is also comprised of symbols a_m of alphabet \mathcal{A} . Each such symbol may appear none, one, or several times in \mathcal{S} . If we know the "count" of each symbol, we can estimate the *empirical* probabilities from a sequence.

The entropy H_0 of a zero-order (memoryless) source depends on *knowing* the source probabilities. Entropy H_0 is a *weighted sum* of the self-information $il(a_m)$, where the weight is $p(a_m)$.

$$H_0 = \sum_{m=1}^M p(a_m) \times il(a_m) \quad (6)$$

Question: *Suppose we do not know the source probabilities?*

A *zero-order* or *memoryless* source generates output sequences \mathcal{S} that are *independent and identically distributed*, abbreviated as i.i.d.

From such sequences that are sufficiently long, we can calculate the source entropy H_0 as the per-symbol *ideal length*, denoted $IL(\mathcal{S}_1^N)$. Notation \mathcal{S}_1^N , explicitly means all N symbols in the sequence, from s_1 , through symbol s_N . For convenience, \mathcal{S} also denotes the entire N -symbol sequence.

Let $Ct(a_1), \dots, Ct(a_m), \dots, Ct(a_M)$ respectively denote the number of times that each alphabetic symbol of alphabet \mathcal{A} appears in \mathcal{S} . The following two relationships of Eq 7 must hold, since each of the N symbols adds a count of 1 to the total symbol count of sequence \mathcal{S} .

In the first sum, we are counting the instances of each sequential symbol of \mathcal{S} , and thus compute a total of N symbols. For each next symbol s_k of \mathcal{S} we observe, we add the value 1 to the running total.

The second sum is computed in two steps as follows. In the first step, we

1. count the number of times each of the unique M symbols a_m occur in \mathcal{S} : there are $Ct(a_m)$ instances of each respective symbol a_m .
2. Next, we know each of the M symbols a_m of alphabet \mathcal{A} has count $Ct(a_m)$, and so we sum the M counts of type $Ct(a_m)$. Thus step 2 computes the value of number N , which is the same number as the first sum of Eq 7.

$$N = \sum_{k=1}^N s_k^0 = \sum_{m=1}^M Ct(a_m) \quad (7)$$

In the first summation of Eq 7, we assume symbol s_k has a *numerical* representation on a computer (eg., 7-bit ASCII), and any number raised to the 0 power yields value 1. Therefore the summation of the

first sum delivers the cardinality (number of symbols comprising) sequence \mathcal{S} .

Now the result of Eq 5, $il(\mathcal{S}_1^N)$ can be obtained a different way by using Eq 10. We can estimate probabilities $p(a_m)$ as:

$$p(a_m) = \frac{Ct(a_m)}{N}. \quad (8)$$

Equation 8 calculates the so-called *empirical* probabilities of sequence \mathcal{S} .

The self-information $il(a_m)$ may be expressed in positive form as $+\log_2 \frac{N}{Ct(a_m)}$, by inverting the count ratio of Eq 8 and thus eliminate the minus (-) sign in front of the probability in the definition of self-information. We now have the following expression:

$$il(a_m) = +\log_2 \frac{N}{Ct(a_m)} \quad (9)$$

The value contributed to $IL(\mathcal{S})$ by symbol a_m is the product $Ct(a_m)il(a_m)$. Eq 10 sums the corresponding self-information of \mathcal{S} , based on the self-information contributed to $IL(\mathcal{S})$ by each symbol.

$$il(\mathcal{S}) = IL(\mathcal{S}) = \sum_{m=1}^M Ct(a_m)il(a_m). \quad (10)$$

In Eq 10, we replace $il(a_m)$ with the value derived in Eq 9, $+\log_2 \frac{N}{Ct(a_m)}$:

$$IL(\mathcal{S}) = \sum_{m=1}^M Ct(a_m) \times \log_2 \frac{N}{Ct(a_m)} \quad (11)$$

In the second factor, we replace $\log_2 \frac{N}{Ct(a_m)}$ with $\log_2 N - \log_2 Ct(a_m)$:

$$IL(\mathcal{S}) = \sum_{m=1}^M Ct(a_m)(\log_2 N - \log_2 Ct(a_m)) \quad (12)$$

We distribute factor $\sum_{m=1}^M Ct(a_m)$ first over $\log_2 N$, and next over $-\log_2 Ct(a_m)$. We rewrite Eq 12 as:

$$IL(\mathcal{S}) = \sum_{m=1}^M Ct(a_m) \times (\log_2 N) - \sum_{m=1}^M Ct(a_m) \times \log_2 Ct(a_m) \quad (13)$$

The first summation in Eq 13 yields N , since the counts $Ct(a_m)$ must sum to N :

$$N \log_2 N \quad (14)$$

The result, called the IL formula for calculating the *ideal code length* for a sequence \mathcal{S} of N symbols, also called the “NlogN” formula, appears in Eq 15 below.

$$IL(\mathcal{S}) = N \log_2 N - \sum_{m=1}^M Ct(a_m) \log_2 Ct(a_m) \quad (15)$$

We can divide IL by N to convert the result of Eq 15 to an *entropy* value in bits per symbol:

$$\frac{IL(\mathcal{S}_1^N)}{N} = \sum_{i=1}^M \frac{Ct(a_m)}{N} \times il(a_m) \quad (16)$$

$$= \sum_{m=1}^M p(a_m) \times il(a_m) = H_0 \quad (17)$$

Eq 17 uses the empirical probabilities. Note that Eq 17 has the form of Eq 6.

We can also replace each of the M instances $il(a_1), \dots, il(a_m), \dots, il(a_M)$ in the product of Eq 16 as follows

$$il(a_m) = -\log_2 \frac{Ct(a_m)}{N} = \quad (18)$$

$$= \log_2 N - \log_2 Ct(a_m) \quad (19)$$

Eq 19 represents the *ideal length* contribution of a single instance of symbol a_m to the ideal length of sequence \mathcal{S} . If we weight these contributions from a_m by the result of Eq 20, the result is equivalent to H_0 .

The self-information of sequence \mathcal{S} is the sum of the self-information contributed by each symbol a_m . If we multiply the right-hand side of Eq 19 by the probability $p(a_m)$ of each respective symbol a_m , the result is an new expression for the value of H_0 .

$$H_0 = \sum_{m=1}^M p(a_m) \log_2 N - p(a_m) \log_2 Ct(a_m) \quad (20)$$

Higher-order Source Entropy

Consider a 0-order source as having a single context, whose entropy is H_0 . The sequence of symbols “seen” by context z_i of a set of contexts denoted \mathcal{Z} has the equivalent of a zero-order entropy denoted $H(z_i)$. Also, given the original sequence \mathcal{S} , each of the contexts is “in force” for a portion $p(z_i)$ of the symbols. Each symbol must belong to exactly one context z_i . The context set, $z_1, \dots, z_i, \dots, z_{|\mathcal{Z}|}$ has a probability distribution, since the probabilities sum to 1.

$$\sum_{i=1}^{|\mathcal{Z}|} p(z_i) = 1, \quad (21)$$

Notation $|\mathcal{Z}|$ of Eq 21 is the number of contexts, or *cardinality*, of context set \mathcal{Z} .

The entropy of a higher-order source is denoted below with the notation $H(\mathcal{S}|\mathcal{Z})$, where the dependency is based on the contexts, or conditioning states, or the dependent distributions of context set \mathcal{Z} , typical context probability $p(z_i)$ within the set of contexts.

$$H(\mathcal{S}|\mathcal{Z}) = \sum_{i=1}^{|\mathcal{Z}|} p(z_i) \times H(z_i) \quad (22)$$