On the Analysis of Variable-to-Variable Length Codes

Serap A. Savari
Bell Labs, Lucent Technologies, Murray Hill, NJ 07974

Wojciech Szpankowski
Dept. of Computer Science, Purdue University, W. Lafayette, IN

Abstract

We use the "conservation of entropy" [1] to simplify the formula for the redundancy of a large class of variable-to-variable length codes on discrete, memoryless sources. This result leads to new asymptotic upper bounds on the redundancy of the "Tunstall-Huffman" code and the "Tunstall-Shannon-Fano" code.

I. Introduction

The goal of lossless data compression is to represent a string from a given source alphabet by a binary code string so that

- the source string can be unambiguously recovered from its corresponding code string and
- the length of the code string is minimized.

Toward this end, many families of lossless data compression techniques have been devised and investigated.

A common approach to evaluating the performance of a lossless data compression algorithm is to assume a probabilistic model for the generation of source output symbols and determine the expected number of code symbols used per source symbol. This approach is most interesting when the length of the source string to be encoded is large enough to capture all of the "typical" behavior of the data model. For the family of ergodic sources, Shannon [2] defined the entropy of a source and showed that the average number of binary digits per source symbol that can be achieved by any lossless data compression scheme is bounded from below by the binary entropy of the source. The *redundancy* of a source code is the amount by which the expected number of binary digits per source symbol for that code exceeds Shannon's entropy bound. One of the goals in designing source coding algorithms is to minimize redundancy.

The first lossless data compression schemes to be investigated are known as fixed-to-variable length codes; i.e., the encoder maps fixed length blocks of source symbols into variable-length binary code strings. Two important fixed-to-variable length coding schemes which we will describe in detail later in the paper are the Shannon-Fano code [2] and the Huffman code [3]. [4] summarizes much of the vast literature on fixed-to-variable length codes. For a memoryless source, [5] provides an asymptotic analysis of the Huffman code on fixed length blocks of source symbols.

Variable-to-fixed length codes are a dual approach to lossless data compression. A variable-to-fixed length encoder partitions the source output string into a concatenation of variable-length phrases. Each phrase, possibly excepting the final one, belongs to a given dictionary of source strings and the last phrase is a non-null prefix of a dictionary entry. The encoder represents each dictionary string by its corresponding fixed-length code word. Variable-to-fixed length codes are considered to be particularly effective when the data to be encoded has a lot of predictability. The intuition in this case is that by allowing variable-length dictionary entries it is possible to construct dictionaries so that all entries are roughly equiprobable.

Run-length codes were the first variable-length codes to be studied, and these codes have long been recognized to be effective for binary, memoryless sources with small entropies (see [2, p. 33] and [6]). Tunstall [7] considered the problem of designing an optimal variable-to-fixed length code for any discrete, memoryless source, and we will state his algorithm in Section III. Khodak (see [8] and [9]) independently found the same algorithm. [10] summarizes other work in this area. [11] precisely represents the dominant term in an asymptotic expansion of the redundancy of the Tunstall code.

Variable-to-variable length codes form a family of source coding algorithms that contain fixed-to-variable length codes and variable-to-fixed length codes as subfamilies. A variable-to-variable length encoder has two components, a parser and a string encoder. The parser segments the source output sequence into a concatenation of phrases from a predetermined dictionary of source strings. The string encoder takes the sequence of dictionary strings and maps each one into its corresponding binary codeword. Aside from the special cases where either the dictionary strings or the codewords have a fixed length, very little is known about variable-to-variable length codes, even in the case of a binary, memoryless source.

The codes considered in [3]-[11] use dictionaries that have the *unique parsability* property; i.e., any source sequence has a unique prefix in the dictionary. Throughout the paper, we will maintain this assumption on the dictionary of source strings. A useful tool in the analysis of variable-to-fixed length codes is the *conservation of entropy* (see, e.g., [1], [12], [13]). We will state it in Section II and demonstrate how it can be used to simplify the formula for the redundancy of a variable-to-variable length code for a discrete, memoryless source. We will use this result to derive new upper bounds on the asymptotic redundancy of the "Tunstall-Huffman" code and the "Tunstall-Shannon-Fano" code.

II. Unique Parsability and the Conservation of Entropy

A dictionary is said to be *uniquely parsable* if every source string, even those of zero probability, can be uniquely parsed into a concatenation of dictionary entries with a final string that is a non-null prefix of a dictionary entry. It is often convenient to picture the entries of a uniquely parsable dictionary as the leaves of a rooted tree in which the root node corresponds to the null string, each edge is a source alphabet symbol, and each dictionary entry corresponds to the path from the root to a leaf. The tree corresponding to a uniquely parsable dictionary is complete in the sense that every intermediate node in the tree has a full set of edges coming out of it; i.e., any single letter extension of a string that is a proper prefix of a dictionary entry is either a dictionary entry or a proper prefix of a dictionary entry. If there are K possible source output symbols and the dictionary contains M entries, unique parsability implies that $M = \alpha(K-1)+1$ for some integer α; here, α is the number of intermediate nodes in the dictionary tree, including the root.

We introduce the following notation. Let

- $\mathcal{H}$ symbolize the binary entropy of a discrete, memoryless source

- $\mathcal{D}$ denote a uniquely parsable dictionary of source output strings
- $\mathcal{L}_\mathcal{D}$ represent the average number of source letters per dictionary string when the parser uses dictionary $\mathcal{D}$
- $\rho(d),\ d \in \mathcal{D}$ represent the probability of using dictionary string d
- $\ell(d),\ d \in \mathcal{D}$ symbolize the length of the binary codeword corresponding to dictionary string d
- $\mathcal{H}_\mathcal{D}$ denote the binary entropy of the probability distribution $\{\rho(d),\ d \in \mathcal{D}\}$
- $\mathcal{R}_{\mathcal{D},\ell}$ denote the redundancy of the variable-to-variable length code.

It is possible to use renewal reward theory [14, §3.4] to show that as the length of the encoded source string increases, the number of code letters per source letter of this variable-to-variable length code approaches approaches $\left(\sum_{d\in\mathcal{D}} \rho(d)\ell(d)\right)/\mathcal{L}_\mathcal{D}$ with probability one, and hence,

$$\mathcal{R}_{\mathcal{D},\ell} \ = \ \frac{\sum_{d\in\mathcal{D}} \rho(d)\ell(d)}{\mathcal{L}_\mathcal{D}} - \mathcal{H} \ = \ \frac{\mathcal{H}_\mathcal{D} + \left(\sum_{d\in\mathcal{D}} \rho(d)\ell(d) - \mathcal{H}_\mathcal{D}\right)}{\mathcal{L}_\mathcal{D}} - \mathcal{H}. \tag{1}$$

Since $\mathcal{D}$ is a uniquely parsable dictionary, the conservation of entropy theorem [1], [12], [13] states that

$$\mathcal{H}_\mathcal{D} \ = \ \mathcal{H} \cdot \mathcal{L}_\mathcal{D}. \tag{2}$$

By combining (1) and (2), we find that

$$\mathcal{R}_{\mathcal{D},\ell} \ = \ \frac{\sum_{d\in\mathcal{D}} \rho(d)\ell(d) - \mathcal{H}_\mathcal{D}}{\mathcal{L}_\mathcal{D}}. \tag{3}$$

Equation (3) decomposes the redundancy of the variable-to-variable length code into two terms. The denominator represents the expected length of a dictionary phrase and the numerator is the redundancy of a fixed-to-variable length code over an auxiliary source with "symbol" probabilities $\{\rho(d),\ d \in \mathcal{D}\}$.

III. On the Asymptotic Performance of Two Variable-to-Variable Length Codes

In order to specify a variable-to-variable length code, we need to select a dictionary $\mathcal{D}$ of source output strings and a fixed-to-variable length code on the words of $\mathcal{D}$. It appears to be a difficult optimization problem to find the uniquely parsable dictionary with M entries that will lead to minimal redundancy. We will instead focus on the *Tunstall dictionary* of size M.

Tunstall found a very simple algorithm to construct a uniquely parsable dictionary $\mathcal{D}$ with M entries that maximizes $\mathcal{L}_\mathcal{D}$. It is known (see, e.g., [15]) that for a uniquely parsable dictionary, $\mathcal{L}_\mathcal{D}$ is the sum of the probabilities associated with each intermediate node in the tree, including the root. Therefore, an optimal uniquely parsable dictionary will correspond to a set of intermediate nodes with maximal probabilities.

The Tunstall algorithm given below finds the optimal uniquely parsable dictionary for a discrete, memoryless source with an alphabet of size K:

1. Start with each source symbol as a dictionary entry.
2. If the total number of entries is less than M, then goto step 3, else stop.
3. Take the most probable entry σ and replace it with the K strings which are single letter extensions of σ. Do not alter the other entries. Goto step 2.

For the Tunstall dictionary with M entries, let L_M denote the average length of a parsed string and p_M symbolize the probability of the most probable dictionary entry. It is established in [11], [12], and [15] that for all M,

$$\frac{\log_2 M}{L_M} - \mathcal{H} \;=\; \Theta\left(\frac{1}{\log_2 M}\right). \tag{4}$$

[11] demonstrates that the relationship between p_M and M satisfies

$$M \cdot p_M \;=\; \Theta(1). \tag{5}$$

The two fixed-to-variable length codes that we will apply to the words of the Tunstall dictionary are the Shannon-Fano code and the Huffman code. We let $R_{T-SF}(M)$ and $R_{T-H}(M)$ denote the redundancies of the respective variable-to-variable length codes of size M. For the Shannon-Fano code, the length of the codeword assigned to Tunstall dictionary entry d is

$$\ell(d) \;=\; \lceil \log_2(1/\rho(d)) \rceil \tag{6}$$

We can combine the preceding results to obtain the following asymptotic performance bound on the Tunstall-Shannon-Fano code

Theorem 1: The redundancy $R_{T-SF}(M)$ of the Tunstall-Shannon-Fano code with M entries satisfies

$$\limsup_{M\to\infty} \log_2 M \cdot R_{T-SF}(M) \;<\; \mathcal{H}.$$

Proof: Since $\lceil \log_2(1/\rho(d)) \rceil < \log_2(1/\rho(d)) + 1$ for all dictionary entries d, it follows from (3), (4) and (6) that

$$\begin{aligned} \log_2 M \cdot R_{T-SF}(M) \;&=\; \frac{\log_2 M}{L_M}\left(\sum_d \rho(d)\left\{\lceil \log_2(1/\rho(d)) \rceil - \log_2(1/\rho(d))\right\}\right) \\ &<\; \mathcal{H} + \Theta\left(\frac{1}{\log_2 M}\right). \end{aligned}$$

■

The optimal fixed-to-variable length code for *any* discrete memoryless source, including the auxiliary source consisting of words from a dictionary of source strings, is the Huffman code. The Huffman code is a constructive procedure which relies on the observation that an optimum code exists in which the two least likely code words have the same length and differ only in the last binary digit. The corresponding source letters are then merged into a single set containing them both, whose probability is the sum of the two letter probabilities. The remaining problem is to find the optimal fixed-to-variable length code of a discrete, memoryless source with a reduced alphabet which is one smaller than the original. The procedure of grouping together the two least probable source symbols is repeated until the source alphabet consists of only two symbols, and then an optimum code is generated by assigning distinct binary digits to the two symbols.

Gallager [16] established that for a discrete, memoryless source with maximum symbol probability q, the redundancy of the Huffman code for that symbol is at most

$$\log_2\left(\frac{2\log_2 e}{e}\right) + q \;\approx\; 0.086 + q. \tag{7}$$

We therefore have

Theorem 2: The Tunstall-Huffman code of size M has the property

$$\limsup_{M\to\infty} \log_2 M \cdot R_{T-H}(M) \;\leq\; \mathcal{H} \log_2 \left(\frac{2\log_2 e}{e}\right).$$

Proof: Results (3)-(5) and (7) imply that

$$\begin{aligned}\log_2 M \cdot R_{T-H}(M) &\leq \frac{\log_2 M}{L_M}\left(\log_2\left(\frac{2\log_2 e}{e}\right) + p_M\right)\\ &= \left(\mathcal{H} + \Theta\left(\frac{1}{\log_2 M}\right)\right)\cdot\left(\log_2\left(\frac{2\log_2 e}{e}\right) + \Theta\left(\frac{1}{M}\right)\right).\end{aligned}$$

■

IV. Discussion

As we indicated in the introduction, variable-to-fixed length codes are suitable for minimizing redundancy if the dictionary entries are roughly equiprobable. Is this true for the Tunstall code? For a dictionary of size M, each Tunstall dictionary entry is mapped into a binary codeword of length $\lceil \log_2 M \rceil$. By (4), we see that the redundancy of the Tunstall code can be made arbitrarily close to zero with sufficiently large dictionaries. Therefore, it is not unreasonable to claim that the Tunstall dictionary entries are "close" in some sense to being equiprobable. However, we can also use asymptotic analyses to argue that the Tunstall dictionary entries are not equiprobable and that there is something to be gained by using a Huffman code on the entries instead of fixed length codewords. It is known (see, e.g., [11]) that the ratio of the probabilities of the least likely Tunstall dictionary entry to the most likely entry tends to the minimum source symbol probability. Variable-to-fixed length codes are considered most interesting when one of the source symbols occurs much more frequently than the other source symbols, and it is in this case that the ratio of probabilities of Tunstall dictionary entries is most skewed. Furthermore, suppose that we have a binary, memoryless source and the probability of the symbols are x and $1-x$. In this case, [11] demonstrated that if $(\log_2(1/x))/(\log_2(1/(1-x)))$ is irrational, then

$$\lim_{M\to\infty} \log_2 M \cdot \left(\frac{\log_2 M}{L_M} - \mathcal{H}\right) = \mathcal{H}\log_2\left(\frac{\log_2 e}{\mathcal{H}}\right) - \frac{x[\log_2(1/x)]^2 + (1-x)[\log_2(1/(1-x))]^2}{2}.$$

Let us compare this to the result in Theorem 2. We consider how

$$\begin{aligned} f(x) &\doteq \mathcal{H}\log_2\left(\frac{\log_2 e}{\mathcal{H}}\right) - \frac{x[\log_2(1/x)]^2 + (1-x)[\log_2(1/(1-x))]^2}{2} - \mathcal{H}\log_2\left(\frac{2\log_2 e}{e}\right)\\ &= \mathcal{H}\log_2\left(\frac{e}{2\mathcal{H}}\right) - \frac{x[\log_2(1/x)]^2 + (1-x)[\log_2(1/(1-x))]^2}{2}\end{aligned}$$

varies with x.

Theorem 3: $f(x) > 0,\ 0 < x < 1$.

Proof: Since $f(x) = f(1-x)$, $0 \le x \le 1$, we need only consider $x \in (0, 0.5)$. Observe that

$$\begin{aligned}
\frac{df(x)}{dx} &= \frac{1}{2}\log_2\left(\frac{e^2 x(1-x)}{4\mathcal{H}^2}\right)\cdot\log_2\left(\frac{1-x}{x}\right) &(8)\\
\frac{d}{dx}\left(\frac{e^2 x(1-x)}{4\mathcal{H}^2}\right) &= \frac{e^2[x\log_2 x - (1-x)\log_2(1-x)]}{4\mathcal{H}^3} &(9)\\
\frac{d^2}{dx^2}\left(x\log_2 x - (1-x)\log_2(1-x)\right) &= \log_2 e\cdot\frac{1-2x}{x(1-x)} &(10)
\end{aligned}$$

Since the right-hand side of (10) is positive for $0 < x < 0.5$, we have that $x\log_2 x - (1-x)\log_2(1-x)$ is strictly convex in this region, and hence its maximum value is at an endpoint. The function $x\log_2 x - (1-x)\log_2(1-x)$ is zero at $x = 0$ and $x = 0.5$, and is therefore negative on the interval $(0, 0.5)$. Thus, the right-hand side of (9) is negative for $0 < x < 0.5$, and so the function $e^2x(1-x)/(4\mathcal{H}^2)$ is decreasing on this interval. As x approaches 0, the function $e^2x(1-x)/(4\mathcal{H}^2)$ approaches infinity, and the function is $e^2/16 < 1$ at $x = 0.5$. Therefore, there is a value of x^* for which the right-hand side of (8) is positive for $0 < x < x^*$ and negative for $x^* < x < 0.5$. Hence, the minimum value of $f(x)$ over the interval $(0, 0.5)$ will be at an endpoint. The theorem follows because $f(0) = 0$ and $f(0.5) = 0.5\log_2(e^2/8) > 0$. ■

Theorem 3 implies that the redundancy of the Tunstall-Huffman code falls more rapidly to zero than that of the Tunstall code alone. We obtain this result using Theorem 2, which is probably not a tight bound. We plan to obtain precise asymptotic expansions of $R_{T-SF}(M)$ and $R_{T-H}(M)$ using the techniques of [17].

References

[1] S. A. Savari, "Variable-to-fixed length codes and the conservation of entropy," *I.E.E.E. Trans. Inform. Theory* 45, 1612-1620, 1999.

[2] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication,* University of Illinois Press, Urbana, 1949.

[3] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. I.R.E.* 40, 1098-1101, 1952.

[4] J. Abrahams, "Code and parse trees for lossless source encoding," in Proc., Compression and Complexity of SEQUENCES '97, Positano, Italy, 1997.

[5] W. Szpankowski, "Asymptotic average redundancy of Huffman (and other) block codes," *I.E.E.E. Trans. Inform. Theory* 46, 2434-2443, 2000.

[6] P. Elias, "Predictive coding," *I.R.E. Trans. Inform. Theory* IT-1, 16-33, March 1955.

[7] B. P. Tunstall, "Synthesis of noiseless compression codes," Ph.D. dissertation, Georgia Inst. Technol., Atlanta, GA, 1967.

[8] G. L. Khodak, "Delay-redundancy relation of VB-encoding," All-union Conference on Theoretical Cybernetics. Novobirsk, 1969. (Russian)

[9] R. Krichevsky, *Universal Compression and Retrieval,* Kluwer, Dordrecht 1994.

[10] T. J. Tjalkens and F. M. J. Willems, "Variable to fixed-length codes for Markov sources," *I.E.E.E. Trans. Inform. Theory* IT-33, 246-257, 1987.

[11] S. A. Savari and R. G. Gallager, "Generalized Tunstall codes for sources with memory," *I.E.E.E. Trans. Inform. Theory* 43, 658-668, 1997.

[12] F. Jelinek and K. S. Schneider, "On variable-length-to-block coding," *I.E.E.E. Trans. Inform. Theory* IT-18, 765-774, 1972.

[13] J. L. Massey, "The entropy of a rooted tree with probabilities," I.S.I.T., 1983.

[14] R. G. Gallager, *Discrete Stochastic Processes*, Kluwer, Boston 1996.

[15] F. Jelinek and G. Longo, "Algorithms for source coding" in *Coding and Complexity*, G. Longo, Ed. Springer Verlag, Wien, 1975.

[16] R. G. Gallager, "Variations on a theme by Huffman," *I.E.E.E. Trans. Inf. Thy.*, IT-24, 668-674, 1978.

[17] W. Szpankowski, *Average Case Analysis of Algorithms on Sequences*, Wiley, New York, 2001.