Journal of Modern Applied Statistical Methods
May, 2007, Vol. 6, No. 1, 265-278

1538 – 9472/07/$95.00

A Fano-Huffman Based Statistical Coding Method

Aladdin Shamilov Senay Asma
Anadolu University, Turkey

Statistical coding techniques have been used for lossless statistical data compression, applying methods such as Ordinary, Shannon, Fano, Enhanced Fano, Huffman and Shannon-Fano-Elias coding methods. A new and improved coding method is presented, the Fano-Huffman Based Statistical Coding Method. It holds the advantages of both the Fano and Huffman coding methods. It is more easily applicable than the Huffman coding methods and it is more optimal than Fano coding method. The optimality with respect to the other methods is realized on the basis of English, German, Turkish, French, Russian and Spanish.

Key words: Fano-Huffman based statistical coding method, probability distribution of language, entropy, information, optimal code.

Introduction

Problem Statement

Huffman's algorithm is a well-known encoding method that generates an optimal prefix encoding scheme, in the sense that the average code word length is minimum. As opposed to this, Fano's method has not been used so much because it generates prefix encoding schemes that can be sub-optimal (Rueda & Oommen, 2004).

In this article, an improved coding method is presented, which has been named the Fano-Huffman Based Statistical Coding method and applications of this method. This method holds the both advantages of Fano and Huffman coding method. So, it is more easily applicable than the Huffman coding method and is more optimal than Fano coding method. The optimality of the mentioned coding method with respect to the other coding methods is realized on the basis of English, German, Turkish, French, Russian and Spanish.

The classical coding methods and the concept of optimality are described in the section titled Classical Coding Methods and Optimality.

An improved coding method, Fano-Huffman Based Coding Method by which encoding schemes, which are arbitrarily close to the optimum, can be easily constructed, is introduced in the section called Fano-Huffman Based Statistical Coding Method.

In the following section, the tables of constructed binary codes are given and comparisons of considered methods in sense of optimality are made.

In the conclusion, the interpretation of optimality of these results is made subject to classical coding methods and suggestions are given.

Aladdin Shamilov is a Professor in the Department of Statistics. Email: asamilov@anadolu.edu.tr. Senay Asma is a Research Assistant in the Department of Statistics. Email: senayyolacan@anadolu.edu.tr.

Overview

Assume that a source alphabet, $S = \{s_1, s_2, \ldots s_n\}$, whose probabilities of occurrence are $\mathrm{P} = \{\mathrm{p}_1, \mathrm{p}_2, \ldots, \mathrm{p}_n\}$, and a code alphabet, $A = \{a_1, a_2, \ldots a_r\}$ is given. The propose of this study is the generation of an encoding scheme, $\{s_i \to w_i\}$, in such a way

that $\bar{l} = \sum_{i=1}^{n} p_i l_i$ is minimized, where l_i is the length of w_i.

Information theory has important applications in probability theory, statistics and communication systems. Lossless encoding methods used to solve this problem include Huffman's algorithm (Huffman, 1952), Shannon's method (Shannon & Weaver, 1949), arithmetic coding (Sayood, 2000), Fano's method (Hankerson, Harris, & Johnson, 1998), enhanced Fano-based coding algorithm (Rueda & Oomen, 2004) etc. Adaptive versions of these methods have been proposed, and can be found in (Faller, 1973; Gallager, 1978; Hankerson et al., 1998; Knuth, 1985; Rueda, 2002; Sayood, 2000). The survey is necessarily brief as this is a well-reputed field.

Also, assume that the source is memoryless or zeroth-order, which means that the occurrence of the next symbol is independent of any other symbol that has occurred previously. Higher-order models include Markov models (Hankerson et al., 1998), dictionary techniques (Ziv & Lempel, 1977; Ziv & Lempel, 1978), prediction with partial matching (Witten, Moffat, & Bell, 1999), grammar based compression (Kieffer & Yang, 2000), etc., and the techniques introduced here are also readily applicable for such structure models.

Classical Coding Methods and Optimality

In this section, the fundamental steps of classical coding methods are described and the concept of optimality of codes is expounded.

Classical coding methods

Suppose that source alphabet (alphabet of language) $S = \{s_1, s_2, \ldots, s_n\}$ and its probability distribution $P = \{p_1, p_2, \ldots, p_n\}$ are given.

Ordinary Coding Method

This method requires the following steps:

(a) Determine number ℓ satisfying the inequality $\ell \geq \log_2 N$, where ℓ is the length of codeword and N is the the number of symbols in source alphabet;

(b) Enumerate letter ignoring the frequency;

(c) Convert numbers determined by (b) from base 10 to base 2 such that ℓ is the length of converted number (Roman, 1997).

Shannon Coding Method

Construction of Shannon Codes is provided by steps:

(a) Put $P = \{p_1, p_2, \ldots, p_n\}$ in ascending order $p_1 \geq p_2 \geq \ldots \geq p_n$;

(b) Calculate $\ell_i = \lceil \log_2 p_i^{-1} \rceil$ the length of codeword, i= 1, 2, ...,n;

(c) Let define dyadic fraction as $F_1 = 0$ and $F_k = \sum_{i=1}^{k-1} p_i$, $2 \leq k \leq n$. Then calculate F_i, i=1,2,...,n;

(d) Convert dyadic fraction F_i to binary form by using Koblitz's trick, then select first ℓ_i bits as a code corresponding to s_i (Hankerson et. al., 2003).

Fano Coding Method

This method involves the steps:

(a) Perform the probabilities of symbols in source alphabet in ascending order $p_1 \geq p_2 \geq \ldots \geq p_n$;

(b) Divide the set of symbols into two subsets such that the sum of the probabilities of occurrences of symbols in each subset are equal or almost equal. Then, assign a 0 to first subset and a 1 to second;

(c) Repeat step (a) until all subsets have a single element (Венцель, 1969).

Enhanced Fano Coding Method

This method proposed the following steps:

(a) Consider the source alphabet $S = \{s_1, s_2, \ldots, s_n\}$ whose probability distribution of occurrences is $P = \{p_1, p_2, \ldots, p_n\}$, where $p_1 \geq p_2 \geq \ldots \geq p_n$;

(b) Obtain $\phi : s_1 \to w_1, \ldots, s_n \to w_n$ the encoding scheme by Fano's method;

(c) Rearrange $w_1, w_2, \ldots, w_n$ into $w_1', w_2', \ldots, w_n'$ such that $\ell_i' \leq \ell_j'$ for all i < j, and simultaneously maintain $s_1, s_2, \ldots, s_n$ in the same order, to yield the encoding scheme: $\phi' : s_1 \to w_1', \ldots, s_n \to w_n'$ (Rueda & Oommen, 2004).

Huffman Coding Method

This method is bottom-up while the others are top-down. It can be explained more clearly as follows:

(a) Sort symbols of source alphabet in decreasing order of their probabilities;

(b) Merge the two least-probable letter into a single output whose probability is the sum of the corresponding probabilities;

(c) Go to step (a) if the number of remaining outputs is more than 2;

(d) Assign a 0 and a 1 arbitrarily as code words for the two remaining outputs;

(e) Append the current codeword with a 0 and a 1 to obtain the codeword the preceding outputs and repeat step (e) if an output is the result of the merger of two outputs in a preceding step. Stop if no output is preceded by another output in a preceding step (Aazhang, 2004).

Shannon-Fano-Elias Coding Method

This method can be explained by steps:

(a) Perform the source alphabet $S = \{s_1, s_2, \ldots, s_n\}$ whose probability distribution of occurrences is $P = \{p_1, p_2, \ldots, p_n\}$ and the order of probabilities isn't important;

(b) Obtain the cumulative distribution by the function $F(s) = \sum_{a \leq s} p(a)$;

(c) Consider modified cumulative distribution function $\overline{F}(s) = \sum_{a<s} p(a) + \frac{1}{2} p(s)$, where $\overline{F}(s)$ denotes the sum of probabilities of all symbols less than s plus half the probability of the symbols;

(d) Obtain the length of codeword by the formula $\ell_i(s) = \left\lceil \log \frac{1}{p(s)} \right\rceil + 1$, where $\lceil . \rceil$ denotes rounding up;

(e) Convert dyadic fraction $\overline{F}(s)$ to binary form by using Koblitz's trick such that the codeword has $\ell_i(s)$ bits (Cover & Thomas, 1991).

The concept of optimality of codes

There exists a uniquely decodable code whose codeword lengths are given by the sequence $\{l_i\}_{i=1}^n$ if Kraft inequality $\sum_{i=1}^{n} 2^{-l_i} \leq 1$ holds. Due to Kraft inequality (Cover, 1991), the conditions for optimal codes are as follows:

(a) The average codeword length $\overline{\ell} = \sum_{i=1}^{n} p_i \ell_i$ of an optimal code for a source S is greater than or equal to its entropy $H(S) = -\sum_{i=1}^{n} p_i \log_2 p_i$;

(b) The average codeword length $\bar{\ell}$ of an optimal code for a source S is strictly less than H(S)+1.

For source alphabet $S = \{s_1, s_2, \ldots, s_n\}$ whose probability distribution of occurrences is $P = \{p_1, p_2, \ldots, p_n\}$, the average codeword length is given by $\bar{\ell}$, and entropy of the source alphabet is given by $H(S)$.

Under these conditions, it is required to transmit as well as possible information by using codes consists of fewer bits. So, this problem can be considered as optimization problem which is consist of minimizing $\bar{\ell} = \sum_{i=1}^{n} p_i \ell_i$ subject to constraint $\sum_{i=1}^{n} D^{-l_i} \le 1$, where D is dimension of codebook, i.e. if the codebook is {0,1} then D=2 etc.

This problem is solved by using Langrange Multipliers, and the following result is obtained:

$$l_i^* = -\log_D p_i ; \tag{2.1}$$

$$\bar{l} = \sum_{i=1}^{n} p_i l_i^* = -\sum_{i=1}^{n} p_i \log_D p_i = H_D(S) ; \tag{2.2}$$

$$\bar{l} = H_D(S) . \tag{2.3}$$

But it isn't possible to find an interger number for codeword length that satisfies (2.1). For this reason, it is necessary to obtain the entropy lower bound (Cover & Thomas, 1991; Roman, 1997) satisfying the following inequality:

$$\bar{l} = \sum_{i=1}^{n} p_i l_i^* \ge H_D(S) . \tag{2.4}$$

Moreover, if S is a stationary stochastic process,

$$\bar{l} \to H(S) , \tag{2.5}$$

where H(S) is the entropy rate of the process.

Under the mentioned knowledge, the information per symbol (letter) is given by $I_{\inf/letter} = \frac{H(S)}{\bar{\ell}}$ and the optimality criteria for codes is considered as $I_{\inf/letter} \to 1$ (Венцель, 1969). Moreover, the optimality means that if the text is coded by an optimal coding method, the number of 1s and the number of 0s are nearly equal in sence of maximum entropy. Hence, the optimal codes means that they transmit nearly maximum information since 1s and 0s aren't always equal probable.

Fano-Huffman Based Statistical Coding Method

In this section, a new and improved coding method is proposed, which can be considered as a hybrid method that holds the both advantages of Fano and Huffman coding methods.

It is well known that Fano coding method is a suboptimal procedure for constructing a source code (Rueda & Oommen, 2004). In this method, the source symbols and their probabilities are sorted in a non-increasing order of the probabilities and then the set of symbols is divided into two subsets such that the sum of the probabilities of occurrences of symbols in each subset are equal or almost equal. The main advantage of this method is the division of the set of symbols. Because, it requires pure computations. Hence, the first goal of the improved coding method is to hold this advantage.

Huffman coding method is a optimal procedure (Cover & Thomas, 1991). In this method, the source symbols and their probabilities are also sorted in decreasing order and then the two least-probable symbols are merged into a single output whose probability is the sum of the corresponding probabilities. Thus, by this recursive procedure, the optimal Huffman codes are constructed. The advantage of this coding method is that the procedure is from bottom to top. In this way, the short code

words are attain to the symbols that occur frequently and long code words are attain to the symbols that occur rarely. This advantage of Huffman coding method constitutes the second goal of the improved coding method.

Considering the advantages of these two coding procedure a hybrid coding method is presented. So, the coding method is more easily applicable than the Huffman coding methods and is more optimal than Fano coding method. The codes performed by that coding method are prefix codes and satisfy the sibling property.

The Fano-Huffman based statistical coding method is now proposed in the following form:

(a) Perform the probabilities of symbols in source alphabet in ascending order $p_1 \geq p_2 \geq \ldots \geq p_n$;

(b) Choose k such that $\left| \sum_{i=1}^{k} p_i - \sum_{i=k+1}^{m} p_i \right|$ is minimized. This number k divides the source symbols into two sets of almost equal probability.

(c) Merge the two least-probable letter in each set into a single output whose probability is the sum of the corresponding probabilities;

(d) Go to step (c) if the number of remaining outputs is more than 2;

(e) Assign a 0 and a 1 arbitrarily as codewords for the two remaining outputs;

(f) Append the current codeword with a 0 and a 1 to obtain the codeword the preceding outputs and repeat step (e) If no output is preceded by another output in a preceding step merge the two least-probable subset into a single output whose probability is the sum of the corresponding probabilities;

(g) Stop if no output is preceded by another output in a preceding step.

Note that, according to step (b) due to size of source alphabet, the set of symbols can be divided into more subsets (2^n, n=1,2,...) of equal or almost equal probabilities.

The advantages of the proposed method arise from the comparisons of this method with the other aforesaid coding methods. The applications of this method and comparisons are given in the following section.

Tables, Computational Details and Comparisons

In this section, in order to indicate the advantages of our proposed method, Fano-Huffman Based statistical coding method, we compare it with the traditional coding methods. Various binary codes for English, German, Turkish, French, Russian and Spanish symbols are constructed in sense of optimality.

French, German, Spanish and English symbols (letters) are the Latin characters consisting of 26 letters which are given in Table 1a.

The probabilities of French, German and Spanish symbols (letters) were established in 1939 by Fletcher Pratt (Stephens, 2002; Pratt, 1939), the probabilities of English symbols (letters) were established by Nam Phamdo (2001) and they are given in Table 1b.

Table 1a. French, German, Spanish and English Symbols

A	B	C	D	E	F	G	H	I	J	K	L	M
a	B	c	d	e	F	g	h	i	j	k	l	m

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
n	O	p	q	r	s	t	u	v	w	x	y	z

Table 1b. Probabilities of French, German, Spanish and English Symbols

Symbols	English	French	German	Spanish
A	0.065174	0.08147	0.06506	0.12529
B	0.012425	0.00876	0.02566	0.01420
C	0.021734	0.03063	0.02837	0.04679
D	0.034984	0.04125	0.05414	0.05856
E	0.104144	0.17564	0.16693	0.13676
F	0.019788	0.00959	0.02044	0.00694
G	0.015861	0.01051	0.03647	0.01006
H	0.049289	0.00721	0.04064	0.00704
I	0.055809	0.07559	0.07812	0.06249
J	0.000903	0.00598	0.00191	0.00443
K	0.005053	0.00041	0.01879	0.00004
L	0.033149	0.05783	0.02825	0.04971
M	0.020212	0.02990	0.03005	0.03150
N	0.056451	0.07322	0.09905	0.06712
O	0.059630	0.05289	0.02285	0.08684
P	0.013765	0.02980	0.00944	0.02505
Q	0.000861	0.01361	0.00055	0.00875
R	0.049756	0.06291	0.06539	0.06873
S	0.051576	0.08013	0.06765	0.07980
T	0.072936	0.07353	0.06742	0.04629
U	0.022513	0.05991	0.03703	0.03934
V	0.008290	0.01557	0.01069	0.00895
W	0.017127	0.00020	0.01396	0.00023
X	0.001369	0.00350	0.00022	0.00221
Y	0.014598	0.00116	0.00032	0.00895
Z	0.000784	0.00072	0.01002	0.00523
#	0.191818	-	-	-

Turkish Source Alphabet consists of 29 symbols (letters). The capital and small letters of the Turkish Alphabet are given in Table 2a.

Probabilities of occurrence of Turkish symbols (letters) are given in Table 2b (Shamilov & Yolacan, 2005; Dalkilic & Dalkilic, 2002). Considered probabilities have been constituted from a corpus consist of words from many variety of fields, i. e. scientific articles, newspapers, poetics etc., 12.5 million characters in total.

Russian uses Cyrillic alphabet consisting of 32 symbols (letters) which are given in Table 3a. Probabilities of Russian symbols are given in Table 3b., where # denotes the space symbol (Венцель, 1969; Yaglom & Yaglom, 1966).

Table 2a. Turkish Source Alphabet

A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K
a	b	C	ç	d	e	f	g	ğ	h	ı	i	j	k

L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z
l	m	N	o	ö	p	r	s	ş	t	ı	ü	v	y	Z

Table 2b. Probabilities of Turkish Symbols

Letter	Frequency	Letter	Frequency	Letter	Frequency
A	0.1026	I	0.0444	R	0.0604
B	0.0237	İ	0.0723	S	0.0264
C	0.0084	J	0.0003	Ş	0.0157
Ç	0.0102	K	0.0407	T	0.0287
D	0.0400	L	0.0530	U	0.0284
E	0.0782	M	0.0320	Ü	0.0171
F	0.0038	N	0.0633	V	0.0087
G	0.0114	O	0.0214	Y	0.0295
Ğ	0.0092	Ö	0.0074	Z	0.0130
H	0.0096	P	0.0073	#	0.1329

Table 3a. Russian Symbols (Cyrillic alphabet)

А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О
а	б	в	г	д	е	ж	з	и	й	к	Л	м	н	о

П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ(Ь)	Ы	Э	Ю	Я
п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ(ь)	ы	э	ю	я

Table 3b. Probabilities of Russian Symbols

Symbols	Probabilities	Symbols	Probabilities
А	0.064	Р	0.041
Б	0.015	С	0.047
В	0.039	Т	0.056
Г	0.014	У	0.021
Д	0.026	Ф	0.002
Е	0.074	Х	0.009
Ж	0.008	Ц	0.004
З	0.015	Ч	0.013
И	0.064	Ш	0.006
Й	0.010	Щ	0.003
К	0.029	Ъ(Ь)	0.015
Л	0.036	Ы	0.016
М	0.026	Э	0.003
Н	0.056	Ю	0.007
О	0.095	Я	0.019
П	0.024	#	0.145

In order to construct binary codes for English, German, Turkish, French, Russian and Spanish, the classical coding methods are applied to considered source alphabets. Consequently, the constructed binary codes are given respectively in Tables 4-9. Moreover, Fano-Huffman Based statistical coding method is also applied to considered languages. Binary Codes constructed by Fano-Huffman based statistical coding are given in Table 10.

Table 4 Binary Codes for Probability Distrubution of English Symbols

English Alphabet	No	Ordinary Codes	S-F-E Codes	Ordered Alphabet	Shannon Codes	Fano Codes	Enhanced Fano Codes	Huffman Codes
A	0	00000	00001	#	000	000	000	001
B	1	00001	00010010	E	0011	001	001	100
C	2	00010	0001011	T	0100	010	010	0101
D	3	00011	000111	A	0101	0110	0110	0011
E	4	00100	00101	O	01101	0111	0111	0111
F	5	00101	0011111	N	01111	1000	1000	0000
G	6	00110	0100010	I	10001	1001	1001	1000
H	7	00111	010011	S	10011	1010	1010	0010
I	8	01000	010110	R	10100	10110	10110	1010
J	9	01001	011000010011	H	10110	10111	10111	0110
K	10	01010	011000011	D	11000	11000	11000	01101
L	11	01011	011001	L	11001	11001	11001	01011
M	12	01100	0110110	U	110100	11010	11010	01110
N	13	01101	011101	C	110110	110111	11100	11110
O	14	01110	100001	M	110111	110110	110110	011101
P	15	01111	10001111	F	111000	11100	110111	111101
Q	16	10000	100100011001	W	111001	111010	111010	011011
R	17	10001	100110	G	111011	111011	111011	001111
S	18	10010	101001	Y	1111000	111100	111100	101111
T	19	10011	10110	P	1111010	111101	111101	011111
U	20	10100	1100000	B	1111011	111110	111110	111111
V	21	10101	11000101	V	1111101	1111110	1111110	0111011
W	22	10110	1100100	K	11111100	11111110	11111110	01111011
X	23	10111	110001010110	X	1111111000	111111110	111111110	011111011
Y	24	11000	11001100	J	11111110100	1111111110	1111111110	1011111011
Z	25	11001	110011101100	Q	11111110110	11111111110	11111111110	0111111011
#	26	11010	1110	Z	11111110111	11111111111	11111111111	1111111011

Table 5. Binary Codes for Probability Distrubution of German Symbols

German Alphabet	No	Ordinary Codes	S-F-E Codes	Ordered Alphabet	Shannon Codes	Fano Codes	Enhanced Fano Codes	Huffman Codes
A	0	00000	00001	E	000	000	000	000
B	1	00001	0001001	N	0010	001	001	001
C	2	00010	0001101	I	0100	010	010	0100
D	3	00011	001001	S	0101	0110	0110	0010
E	4	00100	0100	T	0110	0111	0111	1010
F	5	00101	0101100	R	0111	1000	1000	0110
G	6	00110	011000	A	1000	1001	1001	0011
H	7	00111	011010	D	10011	1010	1010	0111
I	8	01000	01111	H	10101	10110	10110	0101
J	9	01001	10000100010	U	10110	10111	10111	01100
K	10	01010	1000011	G	10111	11000	11000	01110
L	11	01011	1000110	M	110001	11001	11001	11110
M	12	01100	1001010	C	110011	11010	11010	01011
N	13	01101	10100	L	110101	11011	11011	11011
O	14	01110	1011010	B	110111	11100	11100	01111
P	15	01111	10111000	O	111000	111010	111010	01101
Q	16	10000	101110011111	F	111010	111011	111011	011100
R	17	10001	11000	K	111011	111100	111100	111100
S	18	10010	11010	W	1111001	111101	111101	011111
T	19	10011	11100	V	1111011	1111100	1111100	011101
U	20	10100	111100	Z	1111101	1111101	1111101	111101
V	21	10101	11111000	P	1111110	1111110	1111110	0111111
W	22	10110	11111011	J	1111111100	11111110	11111110	01111111
X	23	10111	11111101001011	Q	11111111100	111111110	111111110	011111111
Y	24	11000	1111110100111	Y	111111111011	1111111110	1111111110	0111111111
Z	25	11001	11111110	X	1111111111001	1111111111	1111111111	1111111111

Table 6. Binary Codes for Probability Distrubution of Turkish Symbols

Turkish Alphabet	No	Ordinary Codes	S-F-E Codes	Ordered Alphabet	Shannon Codes	Fano Codes	Enhanced Fano Codes	Huffman Codes
A	0	00000	00001	#	000	000	000	001
B	1	00001	0001110	A	0010	001	001	000
C	2	00010	00100001	E	0011	0100	0100	0011
Ç	3	00011	00100011	İ	0101	0101	0101	0111
D	4	00100	001010	N	0110	0110	0110	0101
E	5	00101	00111	R	01110	0111	0111	0010
F	6	00110	0100001111	L	10000	1000	1000	0110
G	7	00111	01000101	I	10010	1001	1001	0100
Ğ	8	01000	01001000	K	10011	10100	10100	01011
H	9	01001	01001010	D	10100	10101	10101	01111
I	10	01010	010100	M	10110	10110	10110	01101
İ	11	01011	01100	Y	101110	10111	10111	01010
J	12	01100	0110100111111	T	101111	11000	11000	11010
K	13	01101	011011	U	110001	11001	11001	01110
L	14	01110	011110	S	110011	11010	11010	01100
M	15	01111	100001	B	110101	110110	110110	11100
N	16	10000	10010	O	110110	110111	110111	011011
O	17	10001	1001110	Ü	111000	111000	111000	011101
Ö	18	10010	101000001	Ş	111001	111001	111001	011110
P	19	10011	101000101	Z	1110100	111010	111010	111110
R	20	10100	101010	G	1110110	111011	111011	0111011
S	21	10101	1011011	Ç	1110111	1111000	1111000	0011111
Ş	22	10110	1011101	H	1111000	1111001	1111001	1011111
T	23	10111	1100000	Ğ	1111010	1111010	1111010	0111111
U	24	11000	1100100	V	1111011	1111011	1111011	1111111
Ü	25	11001	1100111	C	1111100	1111100	1111100	0111101
V	26	11010	11010001	Ö	11111011	1111101	1111101	1111101
Y	27	11011	1101011	P	11111101	1111110	1111110	01111011
Z	28	11100	11011100	F	111111101	11111110	11111110	011111011
#	29	11101	1110	J	111111111110	11111111	11111111	111111011

Table 7. Binary Codes for Probability Distrubution of French Symbols

French Alphabet	No	Ordinary Codes	S-F-E Codes	Ordered Alphabet	Shannon Codes	Fano Codes	Enhanced Fano Codes	Huffman Codes
A	0	00000	00001	E	000	000	000	00
B	1	00001	00010101	A	0010	001	001	0101
C	2	00010	0001101	S	0100	010	010	0001
D	3	00011	001001	I	0101	0110	0110	1001
E	4	00100	0011	T	0110	0111	0111	0011
F	5	00101	01010111	N	0111	1000	1000	1011
G	6	00110	01011010	R	1000	1001	1001	0111
H	7	00111	010111001	U	10011	1010	1010	0110
I	8	01000	01100	L	10101	1011	1011	0010
J	9	01001	011100011	O	10111	1100	1100	1010
K	10	01010	0111001001100	D	11001	11010	11010	01101
L	11	01011	011110	C	110101	11011	11011	01111
M	12	01100	1000010	M	110111	11100	11100	01110
N	13	01101	10010	P	111001	111010	111010	11110
O	14	01110	101000	V	1110110	111011	111011	011111
P	15	01111	1010110	Q	1111000	111100	111100	0111101
Q	16	10000	10110010	G	1111010	1111010	1111010	0011101
R	17	10001	10111	F	1111011	1111011	1111011	1011101
S	18	10010	11001	B	1111100	1111100	1111100	0111111
T	19	10011	11100	H	11111011	1111101	1111101	1111111
U	20	10100	111100	J	11111101	1111110	1111110	01111101
V	21	10101	11111101	X	111111101	11111110	11111110	011111101
W	22	10110	11111111000101	Y	1111111111	111111110	111111110	0111111101
X	23	10111	1111111110	Z	00000000001	1111111110	1111111110	01111111101
Y	24	11000	00000000001	K	000000000101	11111111110	11111111110	011111111101
Z	25	11001	000000000110	W	0000000001110	11111111111	11111111111	111111111101

Table 8. Binary Codes for Probability Distrubution of Russian Symbols

Russian Alphabet	No	Ordinary Codes	S-F-E Codes	Ordered Alphabet	Shannon Codes	Fano Codes	Enhanced Fano Codes	Huffman Codes
А	0	00000	00010	#	000	000	000	000
Б	1	00001	00010010	О	0010	001	001	001
В	2	00010	000110	Е	0011	0100	0100	011
Г	3	00011	00100000	А	0101	0101	0101	0010
Д	4	00100	0010001	И	0110	0110	0110	1010
Е	5	00101	00110	Т	01110	0111	0111	0111
Ж	6	00110	00111100	Н	01111	1000	1000	0011
З	7	00111	00111111	С	10001	1001	1001	0101
И	8	01000	01001	Р	10011	10100	10100	00100
Й	9	01001	01010010	В	10100	10101	10101	01100
К	10	01010	0101011	Л	10101	10110	10110	01110
Л	11	01011	011000	К	101101	10111	10111	01011
М	12	01100	0110100	М	101111	11000	11000	01101
Н	13	01101	011100	Д	110001	110010	11010	11101
О	14	01110	10000	П	110011	110011	110010	010100
П	15	01111	1001010	У	110100	11010	110011	110100
Р	16	10000	100111	Я	110101	110110	110110	011100
С	17	10001	101010	Ы	110111	110111	110111	011110
Т	18	10010	101101	З	1110000	111000	111000	001111
У	19	10011	1011111	Ъ(Ь)	1110010	111001	111001	101111
Ф	20	10100	1100001011	Б	1110100	111010	111010	011111
Х	21	10101	11000100	Г	1110110	111011	111011	011011
Ц	22	10110	110001011	Ч	1110111	111100	111100	111011
Ч	23	10111	11001000	Й	1111001	1111010	1111010	0111100
Ш	24	11000	110010100	Х	1111010	1111011	1111011	0111110
Щ	25	11001	1100101110	Ж	1111100	1111100	1111100	1111110
Ъ(Ь)	26	11010	11001101	Ю	11111010	1111101	1111101	0111111
Ы	27	11011	1101000	Ш	11111011	11111100	11111100	01111100
Э	28	11100	1101010001	Ц	11111101	11111101	11111101	11111100
Ю	29	11101	110101011	Щ	111111100	11111110	11111110	01111111
Я	30	11110	1101100	Э	111111110	111111110	111111110	011111111
#	31	11111	1110	Ф	111111111	111111111	111111111	111111111

Table 9. Binary Codes for Probability Distrubution of Spanish Symbols

Spanish Symbols	No	Ordinary Codes	S-F-E Codes	Ordered Alphabet	Shannon Codes	Fano Codes	Enhanced Fano Codes	Huffman Codes
A	0	00000	0001	E	000	000	000	000
B	1	00001	00100001	A	001	001	001	001
C	2	00010	001010	O	0100	010	010	0010
D	3	00011	001101	S	0101	0110	0110	1010
E	4	00100	0101	R	0110	0111	0111	0110
F	5	00101	011000101	N	0111	1000	1000	0100
G	6	00110	01100100	I	10010	1001	1001	1100
H	7	00111	011001101	D	10100	1010	1010	0101
I	8	01000	011011	L	10101	1011	1011	0011
J	9	01001	011110000	C	10111	11000	11000	1011
K	10	01010	0111100011111011	T	11001	11001	11001	0111
L	11	01011	011111	U	11010	11010	11010	01110
M	12	01100	100010	M	11011	11011	11011	01101
N	13	01101	10010	P	111001	11100	11100	01111
O	14	01110	10101	B	1110110	111010	111010	011110
P	15	01111	1011100	G	1111000	111011	111011	011111
Q	16	10000	10111100	Y	1111001	111100	111100	111111
R	17	10001	11000	V	1111010	1111010	1111010	0111110
S	18	10010	11011	Q	1111011	1111011	1111011	1111110
T	19	10011	111010	H	11111001	1111100	1111100	0011101
U	20	10100	111101	F	11111011	1111101	1111101	1011101
V	21	10101	11111010	Z	11111101	1111110	1111110	0111101
W	22	10110	11111100000110	J	11111110	11111110	11111110	01111101
X	23	10111	1111110001	X	111111111	111111110	111111110	011111101
Y	24	11000	11111101	W	0000000001000	1111111110	1111111110	0111111101
Z	25	11001	111111111	K	000000000101001	1111111111	1111111111	1111111101

Table 10. Binary Codes Constructed by Fano-Huffman Based Statistical Coding Method

Turkish Alphabet	Fano-Huffman based Codes for Turkish symbols	Russian Alphabet	Fano-Huffman based Codes for Russian symbols	English, French, German, Spanish Alphabet	Fano-Huffman based Codes for English symbols	Fano-Huffman based Codes for French symbols	Fano-Huffman based Codes for German symbols	Fano-Huffman based Codes for Spanish symbols
A	110	А	1101	A	0101	0001	0010	011
B	10111	Б	011100	B	111100	0001100	11110	111000
C	0100101	В	11000	C	110000	11100	01010	1110
Ç	0011001	Г	011010	D	11000	10000	0110	0010
D	01001	Д	10110	E	0001	11	001	101
E	0000	Е	0101	F	101000	1100000	001000	1011010
F	010000001	Ж	1010100	G	001100	0100000	00100	0110000
G	1000001	З	001100	H	00000	1001100	10000	0011010
Ğ	0111001	И	0011	I	1010	0101	111	1100
H	1011001	Й	1100000	J	1010000100	11000000	01101110	00111010
I	1111	К	01010	K	00000100	010101000000	101000	1110111010
İ	1000	Л	00100	L	10100	0110	11010	0110
J	110000001	М	00110	M	001000	01010	10100	01010
K	10001	Н	0010	N	0010	1000	011	0100
L	1011	О	111	O	1101	1110	000000	111
M	10101	П	000000	P	011100	11010	0101110	010000
N	0010	Р	10000	Q	0110000100	0000000	011101110	1011000
O	100001	С	1110	R	1110	0100	1100	1001
Ö	1100101	Т	1011	S	0110	1001	0101	0001
P	00000001	У	001000	T	1001	1101	1101	00000
R	1010	Ф	110100000	U	010000	0010	11000	01000
S	00111	Х	0010100	V	1000100	101100	0100000	0011000
Ş	011101	Ц	01111100	W	100100	110101000000	001110	0110111010
T	00011	Ч	111010	X	0010000100	001000000	1111101110	010111010
U	10011	Ш	00100000	Y	101100	1101000000	0111101110	1110000
Ü	000101	Щ	11111100	Z	1110000100	00101000000	1100000	1111010
V	1111001	Ъ(Ь)	101100	#	11			
Y	01101	Ы	110100					
Z	111101	Э	010100000					
#	100	Ю	0111100					
		Я	101000					
		#	001					

In order to determine the information per letter for considered alphabets due to the mentioned coding methods, the following stages are presented:

1) The entropy of each mentioned languages H(S) is calculated.

2) The codeword length of each codes shown in Tables 4-10 is obtained by counting the bits of the code words and thus average codeword length $\bar{\ell}$ is computed for each coding methods.

3) The information per letter $I_{inf/\,letter} = \frac{H(S)}{\bar{\ell}}$ is get for interpretation of optimality of codes.

The results of these stages are given in Table 11. As previously presented, the optimality criteria for codes is $I_{i/s} \to 1$. Obviously, it is seen from Table 11 that, binary codes constructed for each symbols of different alphabet by Fano-Huffman based statistical coding method is more optimal than Fano coding method and is as optimal as constructed by Huffman coding method but it is more easily applicable than Huffman coding method. Also, the improved coding method is more optimal than the others. Moreover, if a file is coded by Fano-Huffman based codes then the dimension of the file will be less than the files coded by the other considered coding methods. Hence, this means faster communication.

Table 11 Information per letter sent by constructed binary codes

Source Alphabet	Ordinary Codes (bits)	Shannon Codes (bits)	Fano Codes (bits)	Improved Fano Codes (bits)	Shannon Fano Elias Codes (bits)	Huffman Codes (bits)	Fano-Huffman based Codes (bits)
English	0.8145	0.8801	0.9834	0.9839	1.0792	0.9905	0.9888
Turkish	0.8732	0.9075	0.9937	0.9937	1.0955	0.9939	0.9939
French	0.7971	0.8885	0.9854	0.9854	1.0911	0.9899	0.9899
German	0.8190	0.9100	0.9901	0.9901	1.1083	0.9915	0.9901
Spanish	0.8032	0.9150	0.9909	0.9909	1.1161	0.9924	0.9916
Russian	0.8839	0.9085	0.9925	0.9936	1.2142	0.9936	0.9936

Conclusion

It is seen that, binary codes constructed by Fano-Huffman based statistical coding method carry information per letter as much as codes constructed by Huffman coding method. However, by this coding method the less subset you divide the more optimal codes you obtain. Thus, this result make Fano-Huffman based statistical coding method preferred coding methods as Huffman coding method for each of the considered languages. Fano-Huffman based statistical coding method takes less time than Huffman coding method to construct binary codes. However, it require more pure computation than Huffman coding method by means of dividing the source alphabet to subsets and this means faster coding.

As it is commonly known, operating system of computers based on American Standard Code for Information Interchange (ASCII) which is ordinary binary codes. Therefore, another main result from this study is the advantage of Fano-Huffman based codes rather than ASCII. Obviously, it can be concluded from this study that ordinary codes are not optimal because they have the highest average codeword length and the least information per letter. Hence, since ASCII codes are ordinary codes, the text coded by them will be larger in size contrary to Fano-Huffman based codes. So, ASCII codes are not preferred codes.

Consequently, Fano-Huffman based codes can be used in computer systems for data compression rather than ASCII for faster communication. Because, if a file is coded by Fano-Huffman based codes then the dimension of the file will be less than file coded by ASCII but it will transmit the same information by using codes consist of less bits.

References

Aazhang, B. (2004). http://cnx.rice.edu/content/m10176/latest/, Creative Commons.

Cover, T. M. & Thomas, J. A. (1991). *Elements of information theory*. USA: John Wiley & Sons, Inc.

Faller, N. (1973). *An adaptive system for data compression*. In 7th Asilomar conference on circuits, systems, and computers , 593–597.

Gallager, R. (1978). Variations on a theme by Huffman. *IEEE Transactions on Information Theory, 24*(6), 668–674.

Hankerson, D., Harris, G. A. & Johnson, P. D. (2003). *Introduction to information theory and data compression* (2nd ed.). Boca Raton, FL: Chapman & Hall/CRC Press.

Hankerson, D., Harris, G. & Johnson, P. (1998). *Introduction to information theory and data compression*. CRC Press.

Huffman, D. (1952). A method for the construction of minimum redundancy codes. *Proceedings of IRE, 40*(9), 1098–1101.

Kieffer, J. C., & Yang, E. (2000). Grammar-bassed codes: a new class of universal lossless source codes. *IEEE Transactions on Information Theory, 46*(3), 737–754.

Knuth, D. (1985). Dynamic Huffman coding. *Journal of Algorithms, 6*, 163–180.

Roman, S. (1997). *Introduction to Coding and Information Theory*. New York: Springer-Verlag.

Rueda, L. (2002). Advances in data compression and pattern recognition. PhD thesis, School of Computer Science, Carleton University, Ottawa, Canada.

Rueda, L. G. & Oommen B. J. (2004). A Nearly-Optimal Fano-Based Coding Algorithm. *Information Processing and Management, 40*, 257-268.

Sayood, K. (2000). *Introduction to data compression* (2nd ed.). Morgan Kaufmann.

Венцель, Е. С. (1969). Теория Вероятностей, Москва.

Pratt, F. (1939). *Secret and urgent: The story of codes and ciphers.* Blue Ribbon Books.

Phamdo, N. (2001). http://diwww.epfl.ch/mantra/CoursINFOII/Web/compression/english.html, State University of New York.

Stephens, D. (2002). http://www.santacruzpl.org/readyref/files/g-l/ltfrqsp.shtml, Santa Cruz Public Libraries, California.

Shamilov A. and Yolacan S. (2005). Various binary codes for probability distribution of Turkish letters. International Conference Ordered Statistical Data: Approximations, Bounds and Characterizations, pp.70 Izmir, Turkey.

Shannon, C. E., & Weaver, W. (1949). The mathematical theory of communications. University of Illinois Press.

Witten, I., Moffat, A. & Bell, T. (1999). Managing gigabytes: Compressing and indexing documents and images (2nd ed.). Morgan Kaufmann.

Yolacan, S. (2005). Statistical properties of different languages based on entropy and information theory. Anadolu University Graduate School of Sciences, Master of Science Thesis (at turkish), Eskisehir.

Ziv, J. & Lempel, A. (1977). A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory, 23*(3), 337–343.

Ziv, J. & Lempel, A. (1978). Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory, 25*(5), 530–536.