Computers Math. Applic. Vol. 24, No. 7, pp. 77–87, 1992
Printed in Great Britain.

0097-4943/92 $5.00 + 0.00

MATHEMATICAL MODELING OF EMPIRICAL LAWS IN COMPUTER APPLICATIONS: A CASE STUDY

YE-SHO CHEN
Quantitative Business Analysis Department, Louisiana State University
Baton Rouge, LA 70803, U.S.A.
PETE CHONG
Department of Computer Science, Southeastern Louisiana University
Hammond, LA 70402, U.S.A.

(Received July 1991 and in revised form November 1991)

Abstract—A major difficulty in using empirical laws in computer applications is the estimation of parameters. In this paper, we argue that the difficulty arises from the misuse of goodness-of-fit tests. As an alternative, we suggest the use of Simon's theory of model building and apply the theory to examine the two well-known laws of Zipf. As a result, we show that the Simon-Yule model of text generation derives two formulations of Zipf's law: (1) the frequency-count distribution proposed by Booth, and (2) the frequency-rank distribution proposed by Mandelbrot. A further significant contribution of the paper is that it provides a theoretical foundation for the estimation of parameters associated with the two laws of Zipf.

1. INTRODUCTION

Various empirical laws have been reported since the invention of computers. A classical example is the two empirical laws of Zipf discovered through the use of the first generation computers. In his book, *Human Behavior and the Principle of Least Effort*, Zipf [1] stated that "if one takes the words making up an extended body of text and ranks them by frequency of occurrence, then the rank r multiplies by its frequency of occurrence, $g(r)$, will be approximately constant." This first law of Zipf focuses mainly on words of high frequency. In contrast, Zipf's second law [1] was motivated by two remarkable phenomena associated with words of very low frequency of occurrence. If we observe and analyze the frequency of different words in long sequences of text and count $f(n)$ as the number of words appearing n times, then the ratio of the number of words occurring once, $f(1)$, to the number of different words in the text is approximately a constant 0.5; also, the values of $f(n)/f(1)$, $n = 1, 2, 3, 4, 5$, show an approximate pattern of 1, 0.33, 0.17, 0.10 and 0.07.

Another example of empirical laws in computer applications is Grosch's law of economics of scale in computers. Grosch [2, p. 310] stated that "there is a fundamental rule ... giving added economy only as the square root of the increase in speed—that is, to do a calculation ten times as cheaply, you must do it one hundred times as fast." That is, the costs of computer systems increase at a rate equivalent to the square root of their power. Grosch's law has triggered the interest of scientific community to examine the validity and implications of the law [3–5].

The above-mentioned empirical laws are examples of extreme hypotheses, as defined by Herbert Simon [6, p. 109]: "assertions that a particular specific functional relation holds between the independent and the dependent variable." Traditional methods of examining the extreme hypotheses are the standard tests of hypotheses, e.g., the goodness-of-fit test. As pointed out by Simon [6, pp. 4–5] this is a questionable approach, because the necessary statistical theory for making a rigorous test is not available.

The first author's work was supported by a grant from the College of Business Administration at Louisiana State University. Both authors thank the referees for their valuable comments on the early draft of the paper.

Instead, Simon [7] proposed a more constructive alternative to standard probabilistic and statistical tests of fit. He suggested that research should focus on the underlying mechanisms that can (1) explain the simplest form of the empirical law as a first approximation, and (2) look for an additional mechanism that could be incorporated into the theory so as to lead to a better second approximation. Based on his theory of modeling, Simon [8] proposed a simple generating mechanism to explain a class of skew distributions associated with the sizes of business firms. Several refinements of the simple generating mechanism were conducted later [6] to better model the real world.

In this paper, we apply Simon's simple generating mechanism to the mathematical modeling of Zipf's laws. As a prelude to examining the paper, Section 2 briefly reviews applications of Zipf's laws in computer applications and the associated problems. In Section 3, we introduce the Simon's generating mechanism. In Section 4, we apply Simon's model to examine Zipf's second law. Examination of Zipf's first law is discussed in Section 5. Implications of the study for Zipf's laws in computer applications are discussed in Sections 6 and 7. Finally, in Section 8, we summarize the findings and propose the future research, including a promising approach to the mathematical modeling of Grosch's law.

2. ZIPF'S LAWS: APPLICATIONS AND PROBLEMS

The first law of Zipf has been applied to many areas of computer science. A classical example is Shannon's study on the vocabulary size of a language implied by Zipf's first law [9]. However, Zipf's observation only revealed a crude approximation of the phenomenon, since its simplicity cannot explain the concavity to the origin [10], as is usually the case with the empirical log-log distributions. Several new formulations of Zipf's law have been proposed. For example, in a study of the frequency of access of sequential searching, Knuth [11] pointed out that there were two more appropriate substitutes for Zipf's first law: Heising's 80-20 rule [12] and the word frequency data by Schwartz [13]. The most general formulation is perhaps the one proposed by Mandelbrot [14]:

$$g(r) = a\,(r+b)^c, \qquad r = 1, 2, 3, \ldots, \tag{1}$$

where $a > 0$, $c < 0$, and $b > -1$. The formulation is generally referred to as Mandelbrot's law of word frequency. Recently, this formulation has been applied to secondary key indexing by Samson and Bendell [15], and to program complexity measures by Shooman [16].

Based on his first law, Zipf derived a formulation which he stated as the second law [1]. Booth [17] argued that Zipf's formulation is only partially true and proposed a more general form as follows:

$$f(n) = a'\left(n^{c'} - (n+1)^{c'}\right), \qquad n = 1, 2, 3, \ldots, \tag{2}$$

where $a' > 0$ and $c' < 0$. The formulation is referred to as Booth's law of word frequency. Some of the recent applications of this formulation include indexed file performance evaluation by Fedorowicz [18] and automatic text analysis by Pao [19].

A major difficulty in using a Zipfian distribution is the estimation of the formulation parameters, e.g., a, b, c in (1) and a', c' in (2). The difficulty arises from the use of goodness-of-fit tests as a tool for judging the appropriateness of estimated parameters. A goodness-of-fit procedure is a statistical test of a hypothesis that the sampled population is distributed in a specific way [20]. There are several statistics used for a goodness-of-fit test. Among those test statistics used, the chi-square test is probably the most common one. Underlying the chi-square procedure, however, is the crucial assumption that the sample is *random* [20], i.e., the observations are independently and identically distributed. Unfortunately, the time-series observations, such as the data from a literary text, may have substantial dependence in practice.

The chi-square test also has the problem of its practice of combining classes. Coile [21] argued that this combining of classes is undesirable and suggested the use of the Kolmogorov-Smirmov one-sample test as a much more powerful test. Several authors followed Coile's suggestion and used the Kolmogorov-Smirmov test as a goodness-of-fit tool [22]. Like the chi-square test, it seems to be a misuse of the Kolmogorov-Smirmov test to the data related to Zipf's law. First,

the crucial assumption of the test is that the sample is *random*. Second, the test is conservative (i.e., not exact), since the data are discrete, not continuous [20].

A formal study of the effect of dependency on conventional tests of fit is conducted by Gleser and Moore [23]. The significant contribution of their paper is that "confounding of positive dependence with lack of fit is a general phenomenon in the use of omnibus tests of fit." The finding suggests that it is inappropriate to use the traditional tests of fit to model Zipf's law.

3. THE SIMON-YULE MODEL OF TEXT GENERATION

In terms of Zipf's laws, Simon's generating mechanism for the class of skew distributions in the sizes of business firms is introduced in this section. According to Simon, the process of text generation can be described as a stochastic process. The stochastic process by which words are chosen to be included in a written text is a two-fold process. Words are selected by an author by processes of association (i.e., sampling earlier segments of his word sentences) and imitation (i.e., sampling from other works, by himself or other authors). Simon's selection processes are stated in the following assumptions, where $f(n,t)$ is the number of different words that have occurred exactly n times in the first t words [6, pp. 28–29]. (Note that these mathematical notations are modified to be consistent with those used in the paper.)

ASSUMPTION 1. *The probability that the* $(t+1)^{\text{th}}$ *word has already appeared exactly* n *times is proportional to* $n f(n,t)$*—that is, to the total number of occurrences of all the words that have appeared exactly* n *times.*

ASSUMPTION 2. *There is a constant probability,* α*, that the* $(t+1)^{\text{th}}$ *word will be a new word—a word that has not occurred in the first* t *words.*

Based on the two assumptions, Simon derived

$$h(n) = \rho B(n, \rho + 1), \qquad n = 1, 2, 3, \ldots, \tag{3}$$

where $h(n)$ is the *expected relative frequency* of words appearing n times, $\rho = \frac{1}{1-\alpha}$ and $B(n, \rho+1)$ is the beta function with parameters n and $\rho + 1$. Since Yule [24] has derived the same equation in a study of a biological problem, predating the modern theory of stochastic process, Simon calls the last equation a Yule distribution. Simon's approach is frequently cited as the Simon-Yule model of text generation.

In the following sections, we will use Equation (3) as the backbone of the discussions and refer it as the Simon-Yule distribution.

4. THE SIMON-YULE DISTRIBUTION AND BOOTH'S LAW OF WORD FREQUENCY

Booth's [17] law, as shown in Equation (2), was originally proposed to model the incidence of low frequency words in English text, where $c' \approx -1$ as was indicated in his empirical data. A problem with Booth's formulation is that it is derived from "a rather arbitrary assumption" [17, p. 392]. In this section, we show that Booth's law can be derived from Simon-Yule's distribution shown in (3).

An equivalent form of Equation (3) is [6, p. 69]:

$$h(n) = \prod_{i+1}^{n-1} \frac{i}{i+\rho} \frac{\rho}{n+\rho}, \qquad n = 1, 2, 3, \ldots. \tag{4}$$

Let v be the total number of different words in the given text and $f(n)$ be the *expected frequency* of words appearing n times in the same text, then Equation (4) implies

$$f(n) = v \prod_{i=1}^{n-1} \frac{i}{i+\rho} \frac{\rho}{n+\rho}, \qquad n = 1, 2, 3, \ldots. \tag{5}$$

If $\rho = 1$, then

$$f(n) = v\left(n^{-1} - (n+1)^{-1}\right), \qquad n = 1, 2, 3, \ldots. \tag{6}$$

Equation (2) reduces to (6) when $a' = v$ and $c' = -1$.

Furthermore, let $F(n) = \sum_{k=n}^{\infty} f(k)$,*then* $F(n) = v\,\rho\,B(n,\rho)$, and we have the following lemma.

LEMMA 1. *If* $\rho \approx 1$, *the* $F(n) \approx v\,\rho\,\Gamma(\rho)\,n^{-\rho}, n = 1, 2, 3, \ldots.$

PROOF. From Titchmarsh [25, p. 58], we have

$$B(n,\rho) = \Gamma(\rho)\,n^{-\rho} - \int_0^{\infty} \left[t^{\rho-1} - \left(1 - e^{-t}\right)^{\rho-1}\right] e^{-nt}\,dt. \tag{7}$$

The second term on the right hand side of the equation is approximately zero if $\rho \approx 1$. Thus, $F(n) \approx v\,\rho\,\Gamma(\rho)\,n^{-\rho}$. Since $f(n) = F(n) - F(n+1)$, we have the following approximation:

$$f(n) = v\,\rho\,\Gamma(\rho)\left(n^{-\rho} - (n+1)^{-\rho}\right), \qquad n = 1, 2, 3, \ldots. \tag{8}$$

Equations (2) and (8) are the same, when $a' = v\,\rho\,\Gamma(\rho)$ and $c' = -\rho$.

In summary, we show that the Simon-Yule model of text generation provides a theoretical justification for the parameter estimation for the formulation proposed by Booth [17]. That is, we obtain the following theorem.

THEOREM 1. *If* $\rho \approx 1$, *then Booth's law of word frequency (Equation (2)) can be derived from the Simon-Yule distribution (Equation (3)) with the parameters* $a' = v\,\rho\,\Gamma(\rho)$ *and* $c' = -\rho$.

5. THE SIMON-YULE DISTRIBUTION AND MANDELBROT'S LAW OF WORD FREQUENCY

To estimate the parameters associated with Mandelbrot's law of word frequency, i.e., Equation (1), we need to introduce the index approach first.

5.1. Frequency-Count and Frequency-Rank Distributions: An Index Approach

Zipf-type data on word occurrences are based on observations of four entities: (a) the word count, n, that is, the number of occurrences of a certain word contained in a text; (b) the count frequency, $f(n)$, or the number of words of each count; (c) the word rank, r, that is, the cumulative frequency of words of the same or greater count; and (d) the rank frequency, $g(r)$, or the number of words of the same rank. Two approaches are taken with Zipf's law: (a) frequency-count and (b) frequency-rank. Booth's [17] formulation, shown in (2), represents a general frequency-count distribution. Mandelbrot's [14] formulation, shown in (1), shows a general frequency-rank distribution.

Two problems are associated with the traditional approaches in bridging the frequency-count and the frequency-rank distributions. First, they assume that the independent variable n runs from one to infinity. Second, they assume that the ns are consecutive without any "jump" or gaps. Typically, the observed values of ns, beyond the first small values, will "jump" to larger values in progressively larger steps; that is, they contain "gaps" and do not run consequently from one to infinity. Realizing the first problem, some authors [22] limited the running of n from one to $n_{\max}$, the maximum value of n. However, the second problem is still not recognized. The index approach avoids the two problems and gives a clearer understanding of the frequency-count approach and the frequency-rank approach.

We introduce the notion of an index i, $i = 1, 2, \ldots, m$, and let n_i and r_i denote the i^{th} different observed value of count and rank, respectively, so that $n_{i+1} > n_i$ and $r_{i+1} > r_i$. Let $f_a(n_i)$ and $F_a(n_i)$ denote the actual number of words with a count of exactly n_i and not less than n_i, respectively. Also, let $g_a(r_i)$ be the actual frequency for words with rank r_i. The data in Table 1 are taken from Zweben [26], who had analyzed a sample of algorithms from *Communications of the ACM*. Note that when several words have the same count, they are assumed to have the same

Table 1. The Maximal-Rank and Random-Rank approaches of Zweben's data [26].

					Maximal-Rank Approach		Random-Rank Approach	
A	B	C	D	E	F	G	H	I
i	n_i	$f_a(n_i)$	$n_i f_a(n_i)$	$F_a(n_i)$	r_i	$g_a(r_i)$	r	$g(r)$
1	1	8	8	29	1	15	1	15
2	2	6	12	21	2	12	2	12
3	3	1	3	15	4	8	3	8
4	4	6	24	14	5	7	4	8
5	5	3	15	8	8	5	5	7
6	7	1	7	5	14	4	6	5
7	8	2	16	4	15	3	7	5
8	12	1	12	2	21	2	8	5
9	15	1	15	1	29	1	9	4
							10	4
Sum		29	112				11	4
							11	4
							12	4
Column A = index i, $i = 1, 2, \ldots$; $m = 9$ in this case.							13	4
							14	4
Column B = number of occurrences.							15	3
							16	2
Column C = frequency of n_i.							17	2
							18	2
Column D = Column B * Column C.							19	2
							20	2
Column E = $\sum_{k=1}^{m} f(n_k)$.							21	2
							22	1
Column F = the maximal-rank.							23	1
							24	1
Column G = frequency of r_i.							25	1
							26	1
Column H = the random-rank.							27	1
							28	1
Column I = frequency of r.							29	1

maximal-rank, which is the largest possible rank [10]. Other ranking methods use minimum-rank, average-rank, and random-rank. Zipf [1] assigned the random-rank to all words with the same frequency of occurrence. In general, there are three advantages in using maximal-rank versus random-rank [10]. In some applications, however, the random-rank approach might be necessary. (For example, see Section 7.2.)

The index notations indicate the following relationships between r, n, F_a, and g_a (for more details, see [10]).

$$r_i = F_a(n_{m-i+1}), \tag{9}$$

and

$$g_a(r_i) = n_{m-i+1}. \tag{10}$$

For examples, in Table 1, $r_3 = F_a(n_7) = 4$ and $g_a(r_3) = n_7 = 8$.

5.2. Bridging the Simon-Yule Distribution and Mandelbrot's Law of Word Frequency

In this section, we show that Equations (9) and (10) provide a bridge between the frequency-count distribution of the Simon-Yule model and the frequency-rank distribution of Mandelbrot. We show a frequency-rank distribution (Equation (18)) of the Simon-Yule model through the index approach. By way of the distribution, we provide a theoretical justification for the estimation of the parameters associated with Mandelbrot's law of word frequency. The justification is shown in the following theorem.

THEOREM 2. *A more realistic formulation of Mandelbrot's law of word frequency (Equation (18)) can be derived from the Simon-Yule distribution (Equation (3)). The three parameters of (18) can be estimated as follows:*

$$a = (\nu\,\rho\,\Gamma(\rho))^{1/\rho},$$
$$b = \frac{1}{m}\sum_{i=1}^{m}(-\varepsilon(n_i)),$$
$$c = -\frac{1}{\rho}.$$

PROOF. Consider $f(n)$, the expected frequency of words appearing n times derived from the Simon-Yule model, and $f_a(n)$, the actual frequency of words with a count of n words. Note that $f_a(n) = 0$ if n is not in the index set $\{n_1, n_2, \ldots, n_m\}$. Let $e(n)$ be the deviation between the actual frequency and expected frequency of words appearing n times, then

$$f_a(n) = f(n) + e(n), \qquad n = 1,2,3,\ldots, \tag{11}$$

and

$$F_a(n) = F(n) + \varepsilon(n), \qquad n = 1,2,3,\ldots, \tag{12}$$

where $\varepsilon(n) = \sum_{k=n}^{\infty} e(k)$.

By using the lemma in Section 4, we have that for $\rho \approx 1$, an approximately equivalent form of Equation (12) is:

$$F_a(n) = v\,\rho\,\Gamma(\rho)\,n^{-\rho} + \varepsilon(n), \qquad n = 1,2,3,\ldots. \tag{13}$$

Since in the real data, we are only interested in the index set $\{n_1, n_2, \ldots, n_m\}$, we rewrite (13) as

$$F_a(n_i) = v\,\rho\,\Gamma(\rho)\,n_i^{-\rho} + \varepsilon(n_i), \qquad i = 1,2,\ldots,m, \tag{14}$$

which is equivalent to

$$n_i = (v\,\rho\,\Gamma(\rho))^{1/\rho}\,(F_a(n_i) - \varepsilon(n_i))^{-1/\rho}, \qquad 1,2,\ldots,m. \tag{15}$$

Equation (15) can be written as

$$n_{m-i+1} = (v\,\rho\,\Gamma(\rho))^{1/\rho}\,(F_a(n_m - i + 1) - \varepsilon(n_m - i + 1))^{-1/\rho}, \qquad i = 1,2,\ldots,m.$$

Substituting the last equation into (10), we have

$$g_a(r_i) = (v\,\rho\,\Gamma(\rho))^{1/\rho}\,(F_a(n_m - i + 1) - \varepsilon(n_{m-i+1}))^{-1/\rho}, \qquad i = 1,2,\ldots,m.$$

From (9), we have

$$g_a(r_i) = (v\,\rho\,\Gamma(\rho))^{1/\rho}\,(r_i - \varepsilon(n_{m-i+1}))^{-1/\rho}, \qquad i = 1,2,\ldots,m. \tag{16}$$

The last term of Equation (7) is approximately zero if n is large, which implies that $F(n) \approx v\,\rho\,\Gamma(\rho)\,n^{-\rho}$ when n is large. Thus, even without the assumption of $\rho \approx 1$, Equation (15) holds approximately for large n_is. This further implies that Equation (16) holds approximately for small r_is. Since only small r_is have significant impact on the distribution of (16), we conclude that Equation (16) holds for $\rho > 0$.

If we set $a = (v\,\rho\,\Gamma(\rho))^{1/\rho}$, $b_i = -\varepsilon(n_{m-i+1})$, $c = -1/\rho$, and define b as the average of all b_is, then we can rewrite Equation (16) as

$$g_a(r_i) = a\,(r_i + b)^c, \qquad i = 1,2,\ldots,m. \tag{17}$$

Equation (17) shows a maximal-rank representation of Equation (1). Based on Table 1, a random-rank version of Equation (17) could be represented as

$$g_a(r) = a\,(r + b)^c, \qquad r = 1,2,\ldots,r_{\max}, \tag{18}$$

where $r_{\max} = \sum_{i=1}^{m} f_a(n_i)$ is the total number of different words in the text. ∎

Equation (18) provides a more realistic formulation of Mandelbrot's law of word frequency. (Note that the parameters b and c play an important role in the shape of the distribution [10].)

6. SOME FACTS FOR APPLYING THE SIMON-YULE DISTRIBUTION

So far, we have shown that the Simon-Yule distribution provides a theoretical justification for both laws of word frequency proposed by Booth and Mandelbrot. Our next step is to show how to apply the Simon-Yule distribution in computer science. As indicated in Section 2, before continuing, we need to address two important topics: (1) the appropriateness of the two underlying assumptions, and (2) the estimation of the parameter ρ.

6.1. Appropriateness of the Two Assumptions

According to Simon [6, p. 3], the meaning of "appropriateness" is twofold. First, the assumptions must be chosen so that the theory fits the striking phenomena. As we have shown in Sections 4 and 5, the two assumptions of the Simon-Yule model satisfy this criteria. Second, the assumptions shall be plausible, i.e., they agree with known facts or are consistent with the empirical data. Regarding the testing of consistency, Simon [6, p. 145] argues that there is no exact theory available to make definite statements about "how good" the fits are. The main idea [6, p. 111] is that "we are interested in knowing what part of the variance of the data is explained by the theory, and how the remaining variance can be accounted for by successive approximations, rather than in testing whether the variance can be proved to be statistically significant." Successive refinements of the two assumptions are discussed in Section 8.

The first assumption of the Simon-Yule model is equivalent to Gibrat's law of proportionate effect [6, p. 4], which states that the expected percentage rate of growth in size is independent of the current size. The plausibility of Gibrat's law of proportionate effect can be examined several ways [6, p. 138]. Simon [6, p. 145] gave a simple and direct method to test Gibrat's law. Without loss of generality, the methodology may be applied to the language of text generation as follows: Construct on a logarithmic scale a scatter diagram of word frequencies for the beginning and the end of the time interval in question. If the data points fall roughly on a straight line of slope +1 on the log- log scale, then the underlying assumption holds.

The plausibility of Assumption 2 can be checked by examining time series data on rates of entry of words. For example, let $f(1,t)$ be the number of words that have occurred exactly once each in the first t words of a text, and let W_t be the total number of different words in the first t words of the same text. Then, the time series pattern of $\{f(1,t)/W_t\}$ may reveal the rates of entry of new words. Simon and Van Wormer [27] find that the steady-state distribution shown in (3) is rather robust with respect to a slowly decreasing rate of entry of new words, which is true in most applications. The finding implies that the second assumption holds if the time series $\{f(1,t)/W_t\}$ does not significantly deviate from a constant.

6.2. Estimation of the Parameter ρ

Let $h_a(n)$ be the *actual relative frequency* of words appearing n times in the sample text, $H_a(n) = \sum_{k=n}^{\infty} h_a(k)$, and $H(n) = \sum_{k=n}^{\infty} h(k)$, then a maximum likelihood estimate of ρ [6, p. 72] is:

$$\sum_{k=1}^{\infty} h(k) \frac{H_a(k)}{H(k)} = 1. \tag{19}$$

Ijiri and Simon [6, p. 72] noted that there is a unique solution ρ^* to Equation (19), and ρ^* makes the expected value of the ratio of actual and theoretical cumulative frequencies equal to 1.

7. APPLICATIONS OF THE SIMON-YULE DISTRIBUTION IN COMPUTER SCIENCE

As indicated in Section 2, there are two types of computer science applications where the Simon-Yule distribution is useful. The first type relates to Booth's law of word frequency. The second type is associated with Mandelbrot's law of word frequency.

7.1. An Application of Booth's Law of Word Frequency

Fedorowicz [18] recently applied Booth's law of word frequency to indexed file performance evaluation. In particular, she shows an access time model in an index file environment. There, Equation (2) is incorporated to depict the storage requirement for this database system. Terms derived from (2) are also used in the access time model. In addition, Booth's law of word frequency is useful in the estimation of mean and variance of the random variable P_i, the number of postings for the i^{th} key. The variable has significant impact on the access time model of Fedorowicz. To see why, we need some basic knowledge of the inverted file structure.

Figure 1 shows three separate files composing an inverted file environment. Each entry of the Index File consists of one access key, the starting location of the key in the Postings File, and the number of postings corresponding to the key. When a search with multikey is performed, each key's corresponding record number(s) in the Header File are obtained and compared with respect to the Boolean operation(s) joining the search keys. For example, suppose a multikey search, A and CAT, is issued. For key A, we identify six record numbers (12, 35, 61, 110, 300 and 561) in the Header File. For key CAT, there are three record numbers (61, 1003 and 2101). With the Boolean operator "AND," we compare the nine record numbers and obtain record 61 from the Header File.

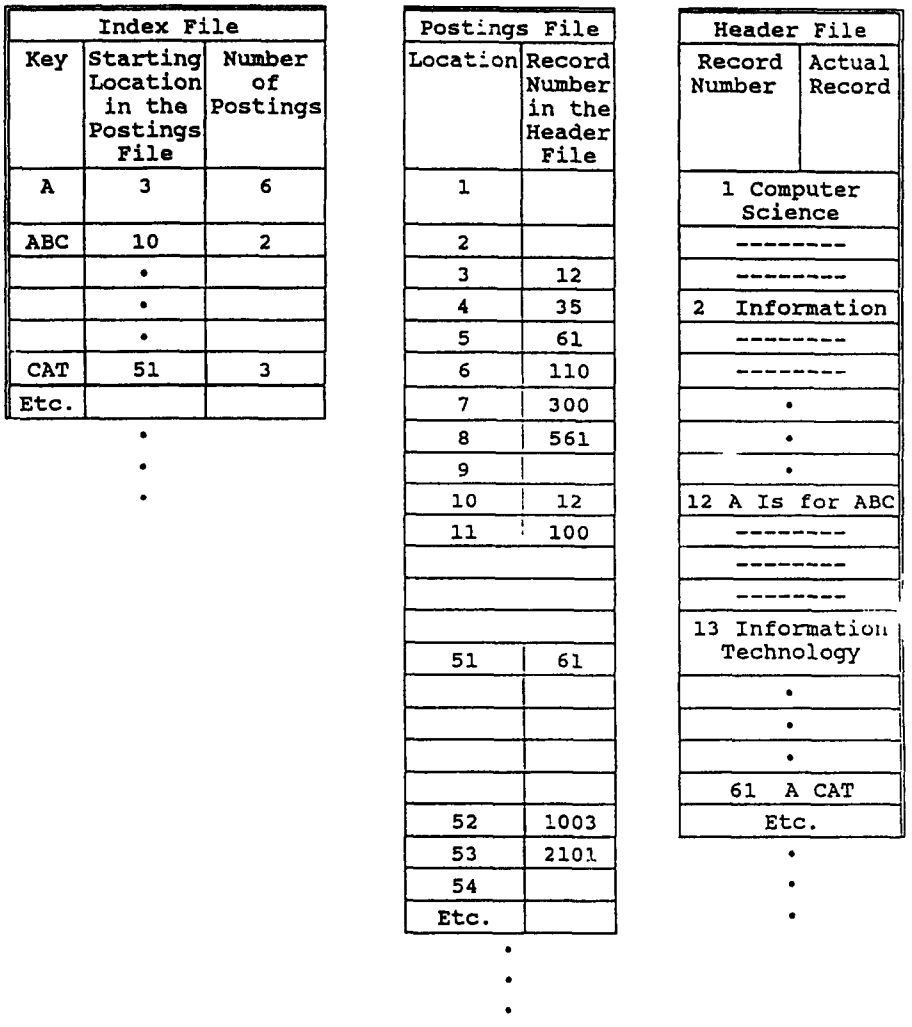

Figure 1. An inverted file environment.

As we can see, the number of comparisons, and hence the access time, for the search is proportional to the size of P_1 and P_2. This relationship is formally presented in Fedorowicz's

article [18, p. 92], which states that the number of comparisons for two keys is

$$(P_1 + P_2 - 1)\,(\mathrm{OR}_1 + \beta\,\mathrm{AND}_1)\,, \tag{20}$$

where $\mathrm{OR}_i\,(\mathrm{AND}_i)$ is a 0–1 variable indicating whether the i^{th} Boolean operation is an OR (AND), and β is an optimization factor when intersecting two blocks of Boolean operation AND. To estimate the mean and variance for (20), we need the following properties of the Simon-Yule distribution derived by Ijiri and Simon [6, pp. 69–70]: The mean and variance of the Simon-Yule distribution are:

$$\mu = \frac{\rho}{\rho - 1}, \qquad \text{if and only if } \rho > 1,$$

and

$$\sigma^2 = \rho\rho^2(\rho - 2)(\rho - 1)^2, \qquad \text{if and only if } \rho > 2.$$

If we assume that P_1 and P_2 are independent and identically distributed, then the mean and variance for (20) are

$$\left(\frac{2\rho}{\rho - 1}\right)(\mathrm{OR}_1 + \alpha\,\mathrm{AND}_1)\,, \qquad \text{if } \rho > 1,$$

and

$$\frac{2\rho}{(\rho - 2)(\rho - 1)^2}\,(\mathrm{OR}_1 + \alpha\,\mathrm{AND}_1)^2\,, \qquad \text{if } \rho > 2,$$

respectively.

7.2. An Application of Mandelbrot's Law of Word Frequency

A typical application of Mandelbrot's law of word frequency is shown in Knuth's book *Sorting and Searching* [11]. In particular, he pointed out that Zipf's first law could be useful in the areas of access frequency of sequential searching and a "self-organizing" file searching. Knuth was aware of the inappropriateness of Zipf's original formulation, so he suggested two substitutes that are more appropriate [12,13] for Zipf's first law. Interestingly, the two substitutes are all special cases of Equation (1) with $b = 0$. Knuth's findings are cited and used frequently in the computer science community. For example, Wiederhold [28] applied Knuth's results to compression of data. However, the essential questions concerning Zipf's first law still remain. For example, why and when does the law hold?, and how to estimate the parameters? As we can see, the discussions in Sections 5 and 6 provide solid background to answer those questions. We will focus our study on the frequency of access of sequential searching. For the convenience of reference, we follow the notations used by Knuth [11].

A sequential search [11, p. 393] begins at the beginning of a file and goes through each record until a desired record is found or the end of the file is reached. Suppose there are N records in the file, and let p_i be the probability that record i will occur, then

$$\sum_{i=1}^{N} p_i + q = 1,$$

where q is the probability that the record is not in the file. Supposing that the file is sufficiently large and we can reasonably assume that $q = 0$, then $\sum_{i=1}^{N} p_i = 1$. Let $\overline{C}_N$ be the expected number of comparisons to search a record, then $\overline{C}_N = \sum_{n=1}^{N} i\,p_i$. Suppose we are able to arrange the records in any order we desire, then $\overline{C}_N$ reaches its minimum if $p_1 \geq p_2 \geq \cdots \geq p_N$. In other

words, the records are arranged by descending frequency of access, such that the most frequently used records appear close to the beginning.

If the two assumptions of the Simon-Yule model hold, then an appropriate distribution for p_i, $i = 1, 2, \ldots, N$, would be

$$p_i = a(i+b)^c, \qquad i = 1, 2, \ldots, N, \tag{21}$$

where $a = (v\rho\Gamma(\rho))^{1/\rho}$, $b = 1/m\sum_{i=1}^{M}(-\varepsilon(n_i))$, and $c = -1/\rho$. We have immediately

$$\overline{C}_N = \sum_{i=1}^{N} a\, i\, (i+b)^c. \tag{22}$$

Note that Equation (21) is equivalent to Equation (18).

8. CONCLUSIONS AND FUTURE RESEARCH

Empirical laws discovered in computer applications are of special interest to the scientific community. However, due to lack of an appropriate statistical theory, there have been debates regarding the validity of the laws. In this paper, we suggest the use of Simon's modeling process for the mathematical modeling of empirical laws. The process is applied to examining the two well-known empirical laws of Zipf. Three significant contributions can be identified: (1) we show that the Simon-Yule model of text generation derives a frequency-count distribution which is consistent with the distribution proposed by Booth; (2) we show that Mandelbrot's formulation of Zipf's first law can be derived from the Simon-Yule model; and (3) we provide a theoretical foundation for the estimation of parameters of Zipf's laws without using the troublesome goodness-of-fit tests adopted by traditional modelers.

As noted in Section 1, the basic model of Simon-Yule is only a first approximation to the striking features of Zipfian data. Our next step is to examine the two assumptions described in Section 3 and to look for an additional explanatory variable that could be incorporated into the generating mechanism so as to lead to a better second approximation of the empirical data. The process of successive refinements was conducted by Simon and his colleagues from 1955 to 1977. Instead of doing research on word frequencies, they focused on the sizes of business firms. Eleven papers were collected in the monograph: *Skew Distributions and the Sizes of Business Firms* [6]. The modifications were based on empirical data and supported by economic theory. The two main refinements are: autocorrelated growth of firms, and mergers and acquisitions. The two main refinements increase the realism of the model and enable the policy makers to show the effect of public policy on the size of firms.

Besides word frequencies and business firms, the Simon-Yule model can also be applied to a number of different sets of highly skewed distributions including [6, pp. 39–50] scientific publications, city sizes, income distribution, and biological species. The accumulated knowledge developed for the sizes of business firms can be applied to other skewed distribution functions to provide a better and richer understanding of the phenomena. With regard to Zipf's laws, consider, as an example, Simon's refinement assumptions on business mergers and acquisitions. A possible relevance of the assumptions to Zipf's laws is the hypotheses about how the forming and dissolution of joint words could affect text output. Empirical data are necessary to support the hypotheses.

Besides Zipf's laws, the principle of Simon's modeling process can be applied to the mathematical modeling of other empirical laws in computer applications. Consider, for example, the modeling of Grosch's law. It is well known that the advent of micro- and mini-computers has greatly complicated the issue. However, the law is still valid if mainframes are considered alone. Instead of using the traditional regression analysis, Simon's modeling process [7] suggests the following steps to study Grosch's law:

(1) Begin with raw data, not regression models.
(2) Draw simple generalizations from striking features of the data.
(3) Find limiting conditions by manipulating the influential variables.
(4) Construct simple mechanisms to explain Steps (2) and (3).

(5) Propose the explanatory theories that go beyond Step 4 and make experiments for new empirical observations.

A study based on the suggested process is in progress.

References

1. G.K. Zipf, *Human Behavior and the Principal of Least Effort*, Addison-Wesley, Cambridge, MA, (1949).
2. H.R.J. Grosch, High speed arithmetic: The digital computer as a research tool, *Journal of the Optical Society of America* **43** (4), 306–310 (April 1953).
3. P. Ein-Dor, Grosch's law re-revisited: CPU power and the cost of computation, *Communications of the ACM* **28** (2), 142–151 (February 1985).
4. Y.M. Kang, Computer hardware performance: Production and cost function analysis, *Communications of the ACM* **32** (5), 586–592 (May 1989).
5. H. Mendelson, Economics of scale in computing: Grosch's law revisited, *Communications of the ACM* **30** (12), 1066–1072 (December 1987).
6. Y. Ijiri and H.A. Simon, *Skew Distributions and the Sizes of Business Firms*, North-Holland, New York, (1977).
7. H.A. Simon, On judging the plausibility of theories, In *Logic, Methodology and Philosophy of Sciences, Vol. III*, (Edited by B. van Rootselaar and J.F. Staal), North-Holland, Amsterdam, (1968).
8. H.A. Simon, On a class of skew distribution functions, *Biometrika* **42** (3/4), 425–440 (1955).
9. C.E. Shannon, Prediction and entropy of printed English, *Bell Syst. Tech. J.* **30**, 50–64 (January 1951).
10. Y.S. Chen and F.F. Leimkuhler, Analysis of Zipf's law: An index approach, *Information Processing and Management* **23** (3), 171–182 (1987).
11. D.E. Knuth, *The Art of Computer Programming, Vol. 3—Sorting and Searching*, Addison Wesley, Reading, MA, (1973).
12. W.P. Heising, Note on random addressing techniques, *IBM Sys. J.* **2** (2), 112–116 (June 1963).
13. E.S. Schwartz, A dictionary for minimum redundancy encoding, *JACM* **10** (4), 413–439 (October 1963).
14. B. Mandelbrot, An information theory of the statistical structure of language, In *Proceedings of the Symposium on Applications of Communications Theory*, London, September 1952, pp. 486–500, Butterworths, London, (1953 486-500).
15. W.B. Samson and A. Bendell, Rank order distributions and secondary key indexing, *The Computer Journal* **28** (3), 309–312 (1985).
16. M.L. Shooman, *Software Engineering: Design, Reliability, and Management*, McGraw-Hill, New York, (1983).
17. A.D. Booth, A law of occurrences for words of low frequency, *Information and Control* **10** (4), 386–393 (1967).
18. J. Fedorowicz, Database performance evaluation in an indexed file environment, *ACM Transactions on Database Systems* **12** (1), 85–110 (March 1987).
19. M.L. Pao, Automatic text analysis based on transition phenomena of word occurrences, *Journal of the American Society for Information Science* **29** (3), 121–124 (1978).
20. W.J. Conover, *Practical Nonparametric Statistics*, 2nd ed., John Wiley & Sons, New York, (1980).
21. R.C. Coile, Lotka's frequency distribution of scientific productivity, *Journal of the American Society for Information Science* **28** (6), 366–370 (November 1977).
22. J. Tague and P. Nicholls, The maximal value of a Zipf size variable: Sampling properties and relationship to other parameters, *Information Processing & Management* **23** (3), 155–170 (1987).
23. C.J. Gleser and D.S. Moore, The effect of dependence on Chi-Squared and empiric distribution tests of fit, *The Annals of Statistics* **11** (4), 1100–1108 (1983).
24. G.U. Yule, A mathematical theory of evolution, based on the conclusions of Dr. J.C. Willis, F.R.S., *Philosophical Transactions of the Royal Society of London*, Series B **213**, 21–87 (1924).
25. E.C. Titchmarsh, *The Theory of Functions*, 2nd ed., Clarendon Press, Oxford, (1939).
26. S.H. Zweben, A study of the physical structure of algorithms, *IEEE Trans. on Soft Engineering* **56-3** (3), 250–258 (May 1977).
27. H.A. Simon and T.A. Van Wormer, Some Monte Carlo estimates of the Yule distribution, *Behavioral Science* **8**, 203–210 (July 1963).
28. G. Wilderhold, *File Organization for Database Design*, McGraw-Hill, New York, (1987).