To appear in Cryptologia.

Statistical Techniques for Language Recognition: An Empirical Study Using Real and Simulated English

Ravi Ganesan
Center for Excellence for Electronic Commerce
Bell Atlantic
Silver Spring, Maryland 20904

Alan T. Sherman*
Computer Science Department
University of Maryland Baltimore County
Baltimore, Maryland 21228-5398

September 28, 1994
(revised June 30, 1994)

Abstract

Computer experiments compare the effectiveness of five test statistics at recognizing and distinguishing several types of real and simulated English strings. These experiments measure the statistical power and robustness of the test statistics X^2, *ML*, *IND*, S, and IC when applied to samples of everyday American English from the Brown Corpus and *Wall Street Journal* and to simulated English generated from 1st-order Markov models based on these samples. An empirical approach is needed because the asymptotic theory of statistical inference on Markov chains does not apply to short strings drawn from natural language. Here, X^2 is the chi-squared test statistic; *ML* is a likelihood ratio test for recognizing a known language; *IND* is a likelihood ratio test for distinguishing unknown 0th-order noise from unknown 1st-order language; S is a log-likelihood function that is a most-powerful test for distinguishing a known language from uniform noise; and IC is the index of coincidence. The test languages comprise four types of real English, two types of simulated 1st-order English, and three types of noise.

Two experiments characterize the distributions of these test statistics when applied to nine test languages, presented as strings of different lengths and contaminated with various amounts of noise. Experiment 1 varies the length of the string from 2 to 2^{17} characters. Experiment 2 adds uniform noise to samples of three fixed lengths (2^4, 2^7, 2^{10}), with the amount of added noise ranging from 0% to 100%. These experiments assess the performance of the test statistics under realistic cryptographic constraints.

Using graphs and tables of observed statistical power, we compare the effectiveness of the test statistics at distinguishing various pairs of languages at several critical levels. Although no statistic dominated all others for all critical levels and string lengths, each test performed well at its designated task. For distinguishing a known type of English from uniform noise at critical levels 0.1 through 0.0001, X^2 attained the highest power, with *ML* and S also performing well. For distinguishing uniform noise from a known type of English at the same critical levels, *ML*

*Part of this work was carried out while Sherman was a member of the Institute for Advanced Computer Studies, University of Maryland College Park.

1 Introduction

Automatic language recognition plays an important role in cryptanalysis, speaker identification, document processing, and other pattern-recognition tasks. To carry out such language-recognition tasks, the theory of statistics offers useful models and test statistics.[1] For example, in our companion introductory guide [16], we explain how to solve four well-defined language-recognition problems using likelihood ratio tests and other standard statistical techniques, when languages are modeled as Markov chains. Although the theory of statistical inference has much to say about the asymptotic performance of many test statistics when applied to the idealized languages of Markov models, this theory says little, if anything, about the performance of test statistics on short strings (*e.g.* 4 to 100 characters) drawn from natural languages such as English. Yet many practitioners must deal with such input strings.

This paper presents and analyzes results of computer experiments that characterize the distributions of five test statistics when applied to several types of real and simulated English strings of various lengths. We focus on statistical approaches to language recognition and empirically analyze the performance of five test statistics for recognizing and distinguishing nine test languages under realistic practical constraints. Specifically, we answer the following questions. What is the actual distribution of each statistic when applied to several types of real English? How do these distributions vary with the length of the input string? For each test statistic, how does its distribution on real English compare with that on simulated English generated from a 1st-order Markov model? How are these distributions affected when the input is contaminated with noise? How effective is each test statistic at distinguishing various pairs of languages? And, what helpful advice can be offered to practitioners who wish to solve language-recognition problems?

To answer these questions we perform two experiments: Experiment 1 characterizes the distributions of five test statistics when applied to strings from nine types of language, with strings ranging in length from 2 to 2^{17} characters. Experiment 2 characterizes the distributions of these statistics when applied to strings of three fixed lengths (2^4, 2^7, 2^{10}) from the same languages, when the strings are contaminated with various amounts of uniform noise. To compare the effectiveness of the test statistics at distinguishing various pairs of languages, we compute the power of these statistics at several critical levels.

Each experiment studies the five test statistics X^2, ML, IND, S, IC, and normalized versions thereof, defined in Section 2. We apply these test statistics to strings randomly selected from large samples of text drawn from nine test languages. Our nine test languages comprise four types of real English, two corresponding 1st-order Markov models of English, and three alternatives to English consisting of uniform noise, a non-uniform 0th-order noise, and a simple repeating pattern. Parameters for each Markov model, including the two 1st-order models of English and the non-uniform 0th-order noise, are computed from two of the samples of real English. We drew our four samples of real English from the *Wall Street Journal* and from the Brown Corpus, which is a well-known collection of everyday American English, assembled during 1963–1964 at Brown University under the direction of Francis, and analyzed by Kučera and his colleagues [13, 26].

Much is known about the asymptotic theory of statistic inference on Markov chains. For example, Anderson and Goodman [2]; Billingsly [6, 7]; and Kullback, Kupperman, and Ku [29] survey the literature. In addition, a few experiments have been carried out to determine the exact

[1] We assume the reader is familiar with elementary statistics—as explained by Larsen and Marx [30], for example. For a more advanced review of hypothesis testing, see Lehmann [31].

For each of our four types of real English, we estimated transition probabilities from a base sample of over 500,000 characters, using straightforward maximum-likelihood estimates as described in Section 3.2. Throughout we work with Markov models with $m = 27$ states corresponding to the characters 'A'–'Z' and blank. Although we do not experiment with case or special characters, such additional information may prove useful in more elaborate models of language.

Notation

Each of the test statistics X^2, ML, and S is defined in terms of the transition probabilities $[p_{ij}]_{1\le i,j\le m}$ of the base language and the observed bigram frequency counts $\{n_{ij}\}_{1\le i,j\le m}$ of the candidate string; the other statistics IND and IC are defined from these frequency counts alone. Here, $m = 27$ is the number of states in our Markov model. The candidate string of n characters forms $N = n - 1$ overlapping bigrams. For each $1 \le i, j \le m$, let n_i denote the observed frequency of letter i. There are $n_{i*} = \sum_{j=1}^{m} n_{ij}$ bigrams beginning with the letter i. For all $1 \le i \le m$, either $n_i = n_{i*}$ or $n_i = n_{i*} + 1$.

From these frequency counts, define the *observed relative unigram frequencies* $\hat{p}_i = n_i/n$ and the *observed relative transition frequencies* $\hat{p}_{ij} = n_{ij}/n_{i*}$. In addition, let $b_{ij} = p_i p_{ij}$ denote the *unconditional bigram probability* of the bigram ij in the Markov model of the base language, where p_i denotes the steady-state state probability (*i.e.* unigram probability) from the Markov model of the base language.

Test Statistics X^2, ML, IND and Normalized Variations $\diamond X^2$, $\diamond ML$, $\diamond IND$

The asymptotically equivalent test statistics X^2 and ML are defined by

$$X^2 = \sum_{1\le i,j\le m} \frac{(n_i\hat{p}_{ij} - n_i p_{ij})^2}{n_i p_{ij}} \tag{1}$$

and[5]

$$ML = 2 \sum_{1\le i,j\le m} n_{ij} \ln \frac{\hat{p}_{ij}}{p_{ij}}. \tag{2}$$

When the base and candidate languages are generated by the same Markov chain, X^2 and ML have an asymptotically χ^2 distribution with $\nu_1 = m(m-1) - d$ degrees of freedom, where d is the number of zero transition probabilities from the base language. In each experiment we took the base language to be language BCa (see Section 3.1), for which $d = 168$ and thus $\nu_1 = 27(27-1) - 168 = 534$. In Equations 1, 2 and throughout, we adopt the convention that all summations are computed over only those indices i, j such that $p_{ij} \neq 0$.[6]

Similarly, when the candidate language is 0th order, the likelihood ratio test statistic

$$IND = 2 \sum_{1\le i,j\le m} n_{ij} \ln \frac{\hat{p}_{ij}}{\hat{p}_j} \tag{3}$$

[5]Throughout let $\ln = \log_e$ and $\lg = \log_2$.

[6]How to treat zero transition probabilities can be important. For more elaborate ways of treating zero transition probabilities, see Good [19], Church and Gale [9], Davies and Ganesan [12], and Levin and Reeds [32].

Anderson's variation of S for bigrams is

$$A = \prod_{1 \le i,j \le m} b_{ij}^{n_{ij}(n/(2N))}. \tag{8}$$

For convenience, we worked with the equivalent statistic $\ln A$.

The Index of Coincidence IC and Normalized Variation $\hat{IC}_\star$

The index of coincidence IC for bigrams is defined by

$$IC = \sum_{1 \le i,j \le m} \frac{n_{ij}(n_{ij}-1)}{N(N-1)}. \tag{9}$$

Good and his colleagues [23, 24] present experimental evidence that, when applied to uniform noise and when $N \ll m^2/12$ (for us, $m^2/12 = 27^2/12 \approx 61$), the statistic $IC_\star = N(N-1)IC/2$ has an approximately Poisson distribution with mean $\mu_\star = N(N-1)/(2m^2)$. For more about the IC and its distribution, see Good [22, 20, 21] and Kullback [27, pp. 151–153].

To express IC on a convenient scale, we center and normalize $IC_\star$ by

$$\hat{IC}_\star = \frac{IC_\star - \mu_\star}{\sqrt{\mu_\star}}, \tag{10}$$

since the mean and variance in the Poisson distribution are the same. When the test statistic $IC_\star$ is approximately Poisson, and when $\mu_\star$ is sufficiently large (*e.g.* $\mu_\star \ge 5$), the related test statistic $\hat{IC}_\star$ is approximately standard normal. However, since the condition $\mu_\star \ge 5$ is satisfied only when $N \ge 86$, we do not recommend using a standard normal interpretation of $\hat{IC}_\star$ for short strings. When computing IC with large n_{ij}, to reduce the chance of arithmetic overflow, it is helpful not to distribute the factor $1/(N(N-1))$ outside of the summation.

3 Experimental Methods

We perform two experiments to determine how each of the test statistics X^2, ML, IND, S, and IC performs when applied to real and simulated English under realistic practical constraints. This section describes our two experiments in detail, focusing on our methods. In doing so, we explain our experimental toolkit, our test languages, how we preprocessed our base samples of these languages, how we carried out each experiment, and how we analyzed the results.

3.1 The Nine Test Languages

To carry out a run of any experiment on any test statistic, up to two languages must be specified: a *candidate language* from which candidate strings are drawn, and a *base language* (if needed) whose known transition probabilities are used in some test statistics. We ran our experiments using the nine test languages described in Table 1, each of which is defined by a sample file of between 500,000 and 900,000 characters. These nine test languages comprise four types of everyday American English (BCa, BCf, BCg, WSJ1), two types of simulated English (1st-order BCa, 1st-order BCf), and three types of noise (0th-order BCa, uniform noise, "er" repeated).

Table 1: The nine test languages. Each test statistic is applied to randomly generated strings from a sample file for each of these test languages. These nine languages consist of four types of American English (BCa, BCf, BCg, WSJ1), two types of simulated English (1st-order BCa, 1st-order BCf), and three types of noise (0th-order BCa, uniform noise, "er" repeated). The simulated languages 1st-order BCa and 1st-order BCf are 1st-order Markov models of BCa and BCf, respectively. For test statistics that require a known base language, we always used BCa as the base language.

Language Type	Source	Description	Length (chars)	Sample
English				
BCa	Brown Corpus file a0144	press reportage	508,567	THE FULTON COUNTY GRAND JURY SAID FRIDAY AN INVESTIGATION OF
BCf	Brown Corpus file f0148	popular lore	556,753	IN AMERICAN ROMANCE ALMOST NOTHING RATES HIGHER THAN WHAT THE
BCg	Brown Corpus file g0175	Belles lettres, biographies, essays	874,601	NORTHERN LIBERALS ARE THE CHIEF SUPPORTERS OF CIVIL RIGHTS AN
WSJ1	*Wall Street Journal* file wsj1	press reportage	763,770	PIERRE VINKEN YEARS OLD WILL JOIN THE BOARD AS A NONEXECUTIVE
Simulated English				
1st-order BCa	computer generated	1st-order Markov model using BCa transition probabilities	550,002	AN H QUND CHONEND S TISTILOFOLART TOR BUTHEY KATE OFIMIS BUDO
1st-order BCf	computer generated	1st-order Markov model using BCf transition probabilities	550,005	AN H RAPE CHOMEND S UPRKIMOFREART URO BUTHE PLATHORAMES PE BE
Not English				
0th-order BCa	computer generated	0th-order Markov model using BCa transition probabilities	550,002	AN I SAOE EJVHGIDTTW OTNNHRBNEAPT RR EW IESRNATETPBGIOTRDYDI
uniform noise	computer generated	0th-order Markov model using uniform transition probabilities	600,001	BMYJZRANFZFKVHHJDSTVYNUMMIQBNF BPSZYRQZDVXJFSRMAUEUPBHIOUQDWDJ
er repeated	computer generated	"ererer..."	628,860	ERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERERER

was given by a file of transition probabilities estimated from the clean BCa sample. Each run of Experiment 1 applies all specified test statistics to the same strings generated from the given candidate language.

We apply one test statistic to one candidate language as follows. For each integer $1 \leq e \leq 17$, we select $h = 100$ candidate sample strings of length $n = 2^e$ characters ($N = n - 1$ bigrams) from the candidate language file. Thus, the candidate strings range in length from 2 to $2^{17} = 131,072$ characters. Each string is selected independently, with uniform distribution from the set of all possible substrings of length n from the specified language file. The test statistic is computed on each of the h strings. To summarize the distribution of the test statistic on the specified language, for each string length, Experiment 1 outputs the sample mean and sample standard deviation computed from the observed h values of the test statistic.[8] Optionally, when we desired a more detailed characterization of the distribution of the test statistic, we output a list of these values and viewed them as a histogram.

We arbitrarily chose the values 100 and 17 with the intent of achieving a sufficiently large sample size and of ensuring that asymptotic behavior of the test statistics would be apparent by our maximum string lengths. After performing a complete set of experiments with $h = 100$, we ran a smaller set with $h = 10,000$ and $1 \leq \lg n \leq 11$ for the purpose of improving the accuracy of our power calculations. All power calculations for Experiments 1 and 2 are based on sample size $h = 10,000$.

It is possible for the sample strings to overlap. Overlap is unlikely for short strings; substantial overlap is likely for huge strings. To prevent potential run-time arithmetic overflow errors in calculating any test statistic, all probabilities and relative frequencies less than 10^{-6} were treated as zeroes.

3.4 Experiment 2: Noisy Plaintext

Experiment 2 is a variation of Experiment 1 that empirically characterizes the distribution of each of our five test statistics, when applied to strings contaminated with various amounts of uniform noise. As with Experiment 1, we apply each test statistic to sample strings drawn from each of our nine test languages. Unlike Experiment 1, however, for simplicity we work only with strings of three fixed lengths: $2^4 = 16$, $2^7 = 128$, $2^{10} = 1,024$. The purpose of this experiment is to characterize the actual distributions of our test statistics when applied to everyday English under noisy conditions, which are common in practical language-recognition applications.

Input to Experiment 2 consists of a list of test statistics, a base language, and a candidate language. For each test statistic, Experiment 2 outputs the sample mean and sample standard deviation of $h = 100$ values of the test statistic computed for each string length at each of several noise levels. We tried all noise levels $0 \leq w \leq 100$ in increments of 5. As with Experiment 1, we worked with the test statistics $\diamond X^2$, $\diamond ML$, $\diamond IND$, S, IC, and $\ln A$, and we always chose the base language to be BCa.

For each string length n and noise level $0 \leq w \leq 100$, Experiment 2 selects h strings at random from the given candidate language file. Each character of each string is processed independently, and with probability $w/100$ replaced with a randomly chosen character (possibly the same character). The replacement character is chosen independently with uniform distribution over our 27-character alphabet. Each specified test statistic is applied to the same strings.

[8] Specifically, we computed sample standard deviation as the square root of the unbiased sample variance.

package [34]. In particular, we used $\Phi^{-1}(0.1) = 1.28155$, $\Phi^{-1}(0.01) = 2.32635$, $\Phi^{-1}(0.001) = 3.09023$, and $\Phi^{-1}(0.0001) = 3.71902$.

3.6 Our Language-Recognition Toolkit

To carry out our experiments, we designed and implemented a toolkit of C programs running under the DEC Ultrix and SGI Irix operating systems. These programs include routines to remove extraneous characters from text files, to count k-gram frequencies, to simulate any Markov chain, and to display data. Particular experiments are run by a program that selects strings from a text file at random and applies tests to these strings. All experiments were carried out on 32-bit machines using double-precision arithmetic operators of the C programming language. To select sample strings at random from each candidate language, to generate random noise, and to drive our our Markov chain simulators, we used the pseudorandom number generator "random()". We also made extensive use of several standard data manipulation programs, including the Awk text-processing language and the ACE/gr interactive graphing program.

4 Results

We now describe and compare the distributions of our test statistics when applied to strings of various lengths drawn from the nine test languages. Specifically, we summarize and explain the results of Experiments 1 and 2 and of our power calculations. Finally, we discuss some issues raised in these findings. Throughout, the base language is BCa.

Complete experimental results are given in our technical report [17] and its appendices. In particular, Appendix B lists detailed results for Experiment 1 on plaintext length: for each test statistic, there is a table of sample means and standard deviations of the test statistic applied to each test language for various string lengths. The amount of data is overwhelming, and it is difficult to intuit phenomena from these numerical tables. Therefore, in this section, we selectively present highlights of our data through several graphs, histograms, and tables of statistical power. Additional similar figures and tables are included in the supplemental appendices [17].

4.1 Results of Experiment 1 (plaintext length)

We describe the distributions of the test statistics in histograms and in graphs of mean $\pm$ standard deviation. The histograms provide thorough descriptions of the distributions at fixed string lengths; the graphs summarize this information showing the effect of string length.

Histograms

Figures 1–6 describe the distributions of the statistics S/N, IC, X^2, ML, and IND when applied to BCa English, 1st-order BCa English, 0th-order BCa, and uniform noise. For example, for each of these four test languages, Figure 1 gives histograms of 100 observed values of S/N computed from randomly selected strings of length $n = 1024$, and Figure 2 gives histograms of S/N for $n = 16$. Figures 3–6 present similar histograms of IC, X^2, ML, and IND for $n = 1024$. These histograms reveal the shape, dispersion, and separation of the distributions.

As explained in Section 3.1, BCa is a type of real English; 1st-order BCa is an idealized model of BCa; and 0th-order BCa and uniform noise are alternatives to English. When distinguishing

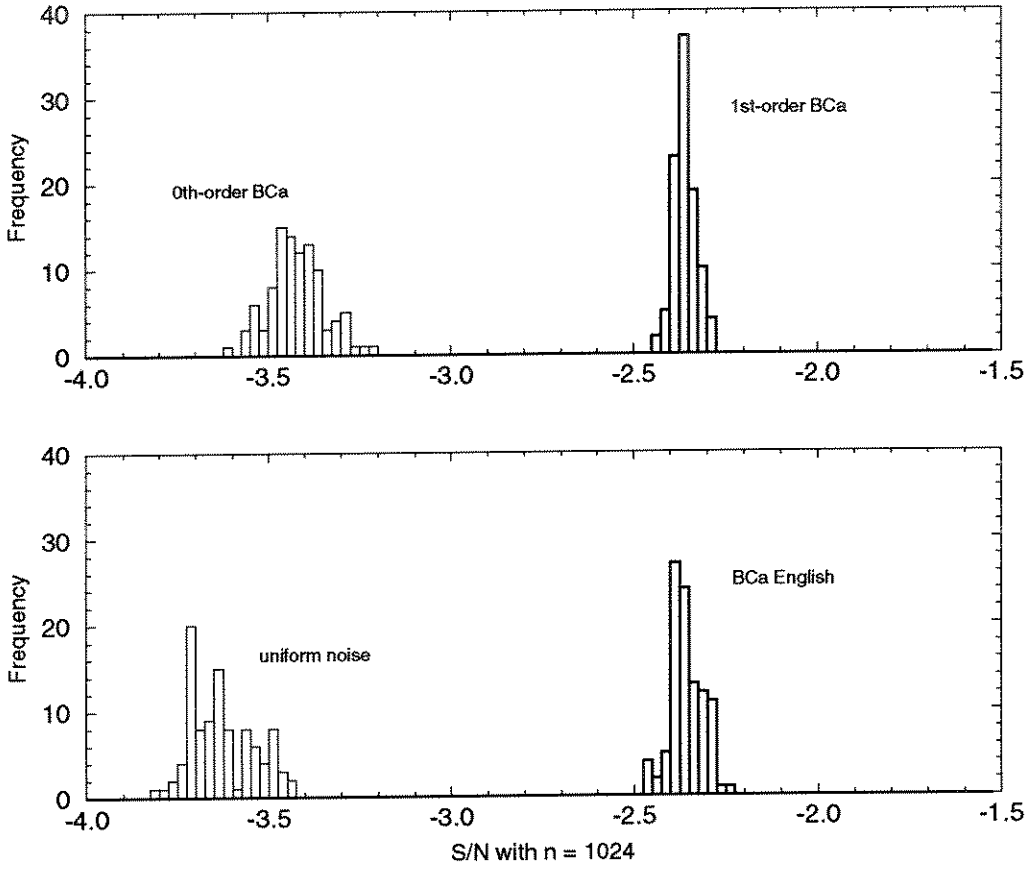

Figure 1: Four histograms of values of the statistic S/N computed on 100 randomly chosen strings of length $n = 1024$ drawn from BCa English, 1st-order BCa English, 0th-order BCa noise, and uniform noise, respectively. For example, for uniform noise, 20 of the chosen strings had S/N values between -3.700 and -3.725. Base language is BCa.

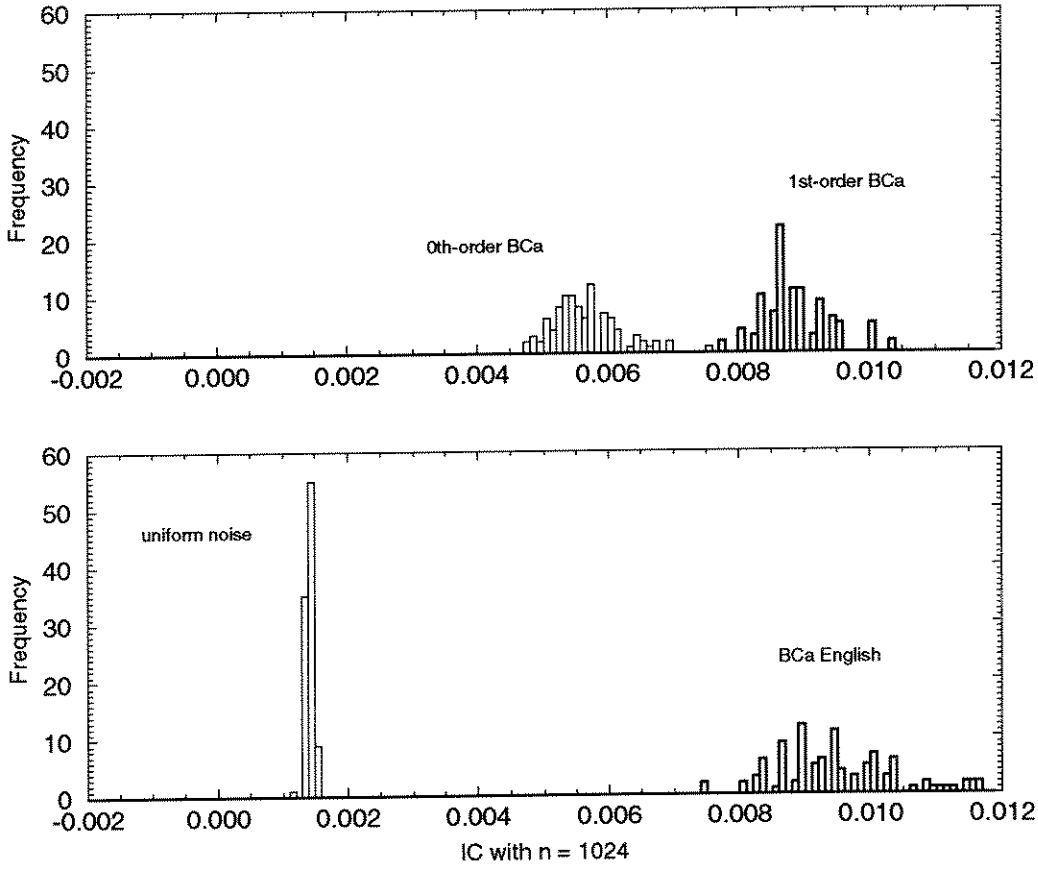

Figure 3: Four histograms of values of the statistic IC computed on 100 randomly chosen strings of length $n = 1024$ drawn from BCa English, 1st-order BCa English, 0th-order BCa noise, and uniform noise, respectively.

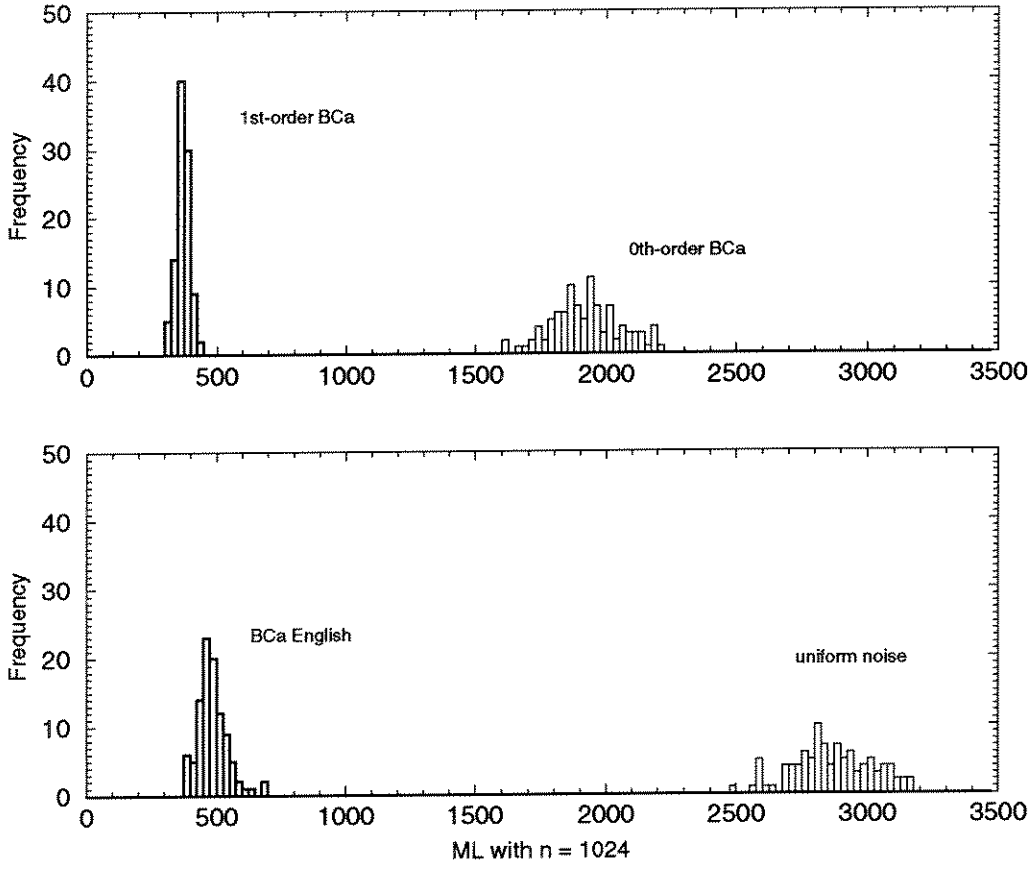

Figure 5: Four histograms of values of the unnormalized statistic ML computed on 100 randomly chosen strings of length $n = 1024$ drawn from BCa English, 1st-order BCa English, 0th-order BCa noise, and uniform noise, respectively. Base language is BCa.

Means and Standard Deviations

To examine the effects of string length on the test statistics, it is convenient to summarize the distributions in terms of their first two moments. We now do so, focusing on the statistics $\hat{S}$, IC, and $\diamond X^2$. Figures 7, 8, and 9 show, respectively, how the means of $\hat{S}$, IC, and $\diamond X^2$ vary with string length for all nine test languages. Figures 10 and 11 show the observed means and standard deviations for $\hat{S}$ and $\hat{IC}_\star$ when applied to BCa English and uniform noise; these standard deviations are crucial in comparing the effectiveness of the test statistics.

Figure 7 illustrates that, for sufficiently large strings, the mean of $\hat{S}$ differs for most of the test languages. For example, the three alternatives to English (uniform noise, 0th-order BCa, and "er" repeated) rapidly diverge from the other test languages. The significance of these differences is established by the standard deviations, as depicted in Figure 10 and given in Appendix B. By $n = 256$, the mean of $\hat{S}$ separates the remaining test languages into three groups: real and simulated BCa English, WSJ1, and the remaining three types of real and simulated English. Figure 8 shows a similar trend for IC. The IC, however, had a more chaotic behavior for very short strings—perhaps because very short strings are unlikely to have repeated bigrams. Also, the IC more cleanly separated uniform noise from 0th-order BCa than did $\hat{S}$, which is useful to know for cryptanalytic applications that yield unigrams only.

Figures 10 and 11 illustrate the behavior of $\hat{S}$ and $\hat{IC}_\star$ at distinguishing BCa English from uniform noise at various string lengths. Although both statistics eventually perfectly separate these two languages, $\hat{S}$ was relatively more effective at small lengths, as is quantified in our power calculations (see Figure 13). In addition, as observed in Figure 3, Figure 11 shows that $\hat{IC}_\star$ has a smaller standard deviation on uniform noise than on BCa English.

Our normalizations $\diamond X^2$ and $\hat{S}$ of X^2 and S, respectively, were intended to produce statistics whose distributions are approximately standard normal when applied to 1st-order BCa (and BCa). Similarly, $\hat{IC}_\star$ was intended to have an approximately standard normal distribution when applied to uniform noise (see Section 2). Figures 9–11 show that this intent was achieved fairly well for $\hat{IC}_\star$ and $\hat{S}$ but not for $\diamond X^2$, except when $\diamond X^2$ was applied to long strings of 1st-order BCa. Nevertheless, as proven in the power tables, the relative separations of the means of $\diamond X^2$ on most languages are indeed significant. Thus, when interpreting the values of $\diamond X^2$, it is important to use experimentally-determined thresholds. Similar statements also apply to $\diamond ML$ and $\diamond IND$.

As shown in Figure 10 and Appendix B [17], although $\hat{S}$ is approximately standard normal on BCa, its standard deviation increases slightly with string length throughout. We believe this slight increase in the standard deviation of $\hat{S}$ results from $\hat{S}$ being applied to dependent bigrams and from the fact that BCa English is not a 1st-order Markov chain. By contrast, when applied to 1st-order BCa, the standard deviation of $\hat{S}$ remains approximately constant and slightly less than 1. For huge strings, however, the standard deviation of $\hat{S}$ on 1st-order BCa decreases slightly, we believe reflecting the bias of selecting the sample strings from the base sample.

Table 8 of Appendix B [17] reveals some interesting properties of the $\diamond X^2$ statistic. First, from these means and standard deviations, it is apparent that a standard normal interpretation of $\diamond X^2$ is not useful, except for recognizing long strings of 1st-order BCa. For example, the observed means of $\diamond X^2$ on BCa lie in the interval $[-5, 5]$ only for string lengths $128 \leq n \leq 1024$, even though BCa is the base language and the candidate strings were drawn from the same base sample used to compute the BCa transition probabilities. Even for 1st-order BCa, the values of $\diamond X^2$ fall outside of $[-5, 5]$ until $n \geq 128$. Second, for long strings, the values of $\diamond X^2$ diverge for all languages except

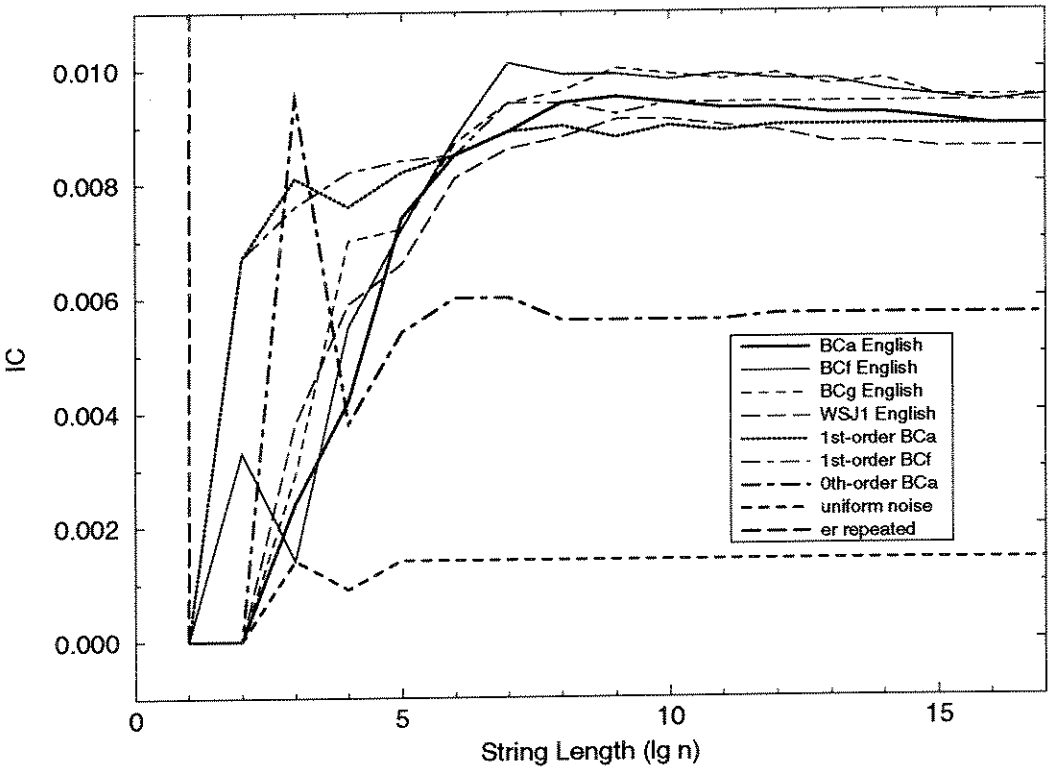

Figure 8: Sample means of the statistic IC computed on 100 randomly chosen strings of various lengths drawn from each of the nine test languages, respectively.

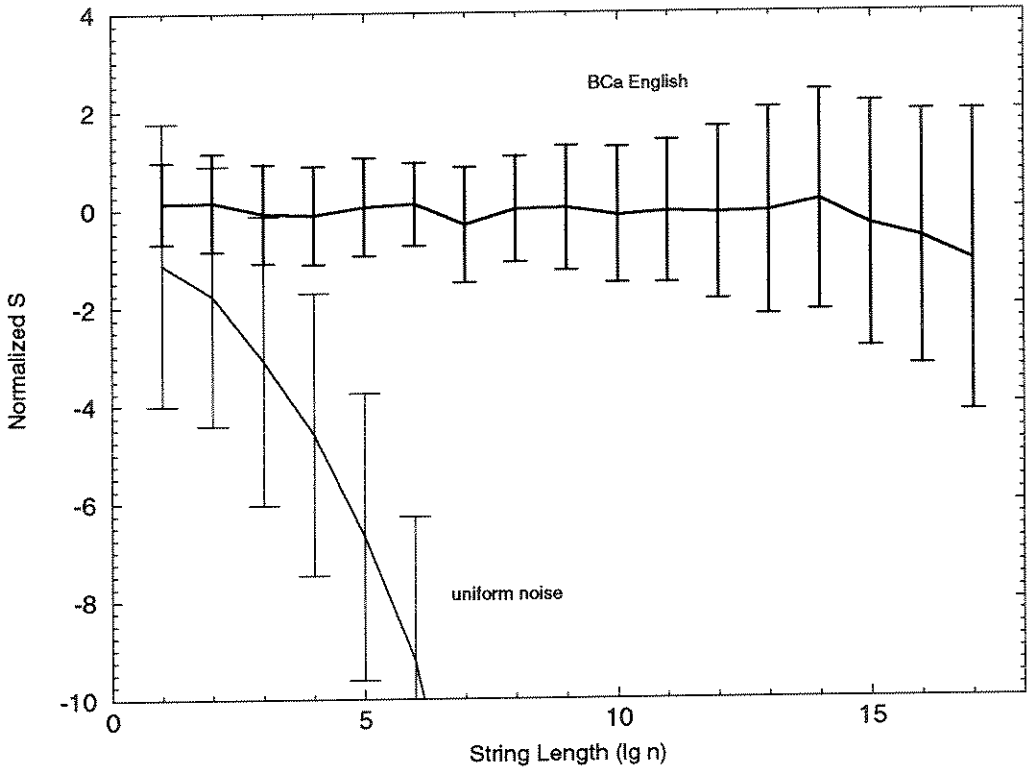

Figure 10: Sample means and standard deviations of the statistic $\hat{S}$ computed on 100 randomly chosen strings of various lengths drawn from BCa English and uniform noise, respectively. Base language is BCa.

4.2 Results of Experiment 2 (noisy plaintext)

Figure 12 highlights the results of Experiment 2 by illustrating the behavior of $\hat{S}$ at distinguishing BCa English from uniform noise in the presence of noise. Specifically, this figure shows the first two moments of $\hat{S}$ when applied to strings of length $n = 1024$ drawn from BCa English and uniform noise with various amounts of added uniform noise. In this figure, the curve for uniform noise serves as a reference line, which could have been drawn using data from Experiment 1 or from theoretically computed values. Similar figures in our technical report [17] show the behavior of the other statistics.

Comparing Figure 12 with the corresponding one for $\hat{IC}_\star$ reveals two interesting differences. First, at $n = 1024$, $\hat{IC}_\star$ is able to distinguish BCa English from uniform noise in the presence of higher amounts of noise than is $\hat{S}$. For example, the standard deviation bars for $\hat{S}$ in Figure 12 first overlap at 65% added noise, whereas those for $\hat{IC}_\star$ first overlap at 80% added noise. This behavior is quantified in our power calculations (see Figure 15). Second, with increasing noise levels, $\hat{IC}_\star$ on noisy BCa converges at a faster rate to $\hat{IC}_\star$ on uniform noise than does $\hat{S}$, perhaps reflecting the quadratic nature of the IC formula.

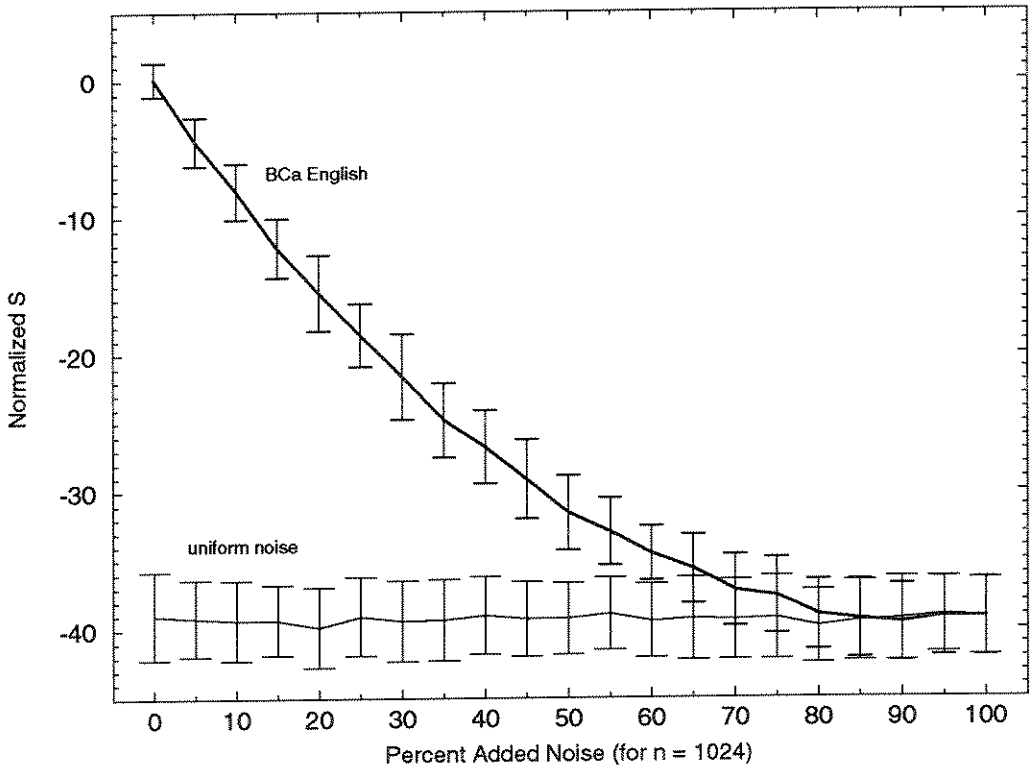

Figure 12: Sample means and standard deviations of the statistic $\hat{S}$ computed on 100 randomly chosen strings of length $n = 1024$ drawn from BCa English, with various amounts of added uniform noise. A corresponding curve for uniform noise is also drawn as a reference line. Base language is BCa.

that $\diamond X^2$ outperformed S given that S is an asymptotically most powerful test for distinguishing BCa English from uniform noise. Perhaps the relatively better performance of $\diamond X^2$ over S at $n < 128$ is explained by the small string length. Even for the short string length $n = 8$ at critical level 0.001, $\diamond X^2$ achieved a reasonably high power of over 0.79. Here and throughout, Anderson's statistic performed unimpressively—usually worse and never better than S (except for $n = 2$). But for each critical level and for all sufficiently long strings ($n \geq 1024$), all statistics performed indistinguishably with perfect power.

In Figure 14 and Table 3, $\diamond ML$ dominated the other statistics at distinguishing uniform noise from BCa English, for string lengths $n > 8$. The statistics IC, and $\diamond IND$, $\diamond X^2$, and S also performed relatively well. The strong performance of IC and $\diamond IND$ was expected: IC is designed to recognize uniform noise, and IND is designed to distinguish 0th-order language (which includes uniform noise) from 1st-order language (which we use to model BCa). We were surprised, however, that $\diamond ML$ outperformed IC since ML is designed to recognize BCa. We attribute this relatively better performance of $\diamond ML$ over IC to the fact that $\diamond ML$ depends on BCa transition probabilities whereas IC does not. As happened in Figure 13, for all critical levels and for all sufficiently long strings ($n \geq 1024$), all statistics performed indistinguishably with perfect power.

When distinguishing uniform noise from BCa English, the relative performance of the statistics varied depending on the critical level. For example, $\diamond X^2$ and S outperformed IC at critical level 0.1 but not at level 0.001, with the greatest difference in power between IC and S occurring at short string lengths and low critical levels. By contrast, when distinguishing BCa English from uniform noise, raising the critical level increased the power of all statistics but had little effect on their relative performance.

For test statistics $\diamond X^2$, $\diamond ML$, S, and $\ln A$, for the same critical level, as high or higher power was attained by distinguishing BCa English from uniform noise than by distinguishing uniform noise from BCa English. In particular, for short strings ($n < 16$) and all critical levels, significantly higher power was so attained using the best statistic for each problem ($\diamond X^2$ and $\diamond ML$, respectively). The implication to the practitioner, however, depends on the application. For example, when recognizing valid plaintext in cryptanalysis, typically the cryptanalyst will set the threshold on the basis of minimizing the chance of overlooking valid plaintext. Thus, when distinguishing BCa English from uniform noise, the cryptanalyst will start with a desired critical level. But when distinguishing uniform noise from BCa English, the cryptanalyst will pick a critical level that achieves a desired power. Therefore, for this application, the choice of which of these two hypothesis testing problems to use cannot be decided from Figures 13 and 14 alone. In addition, this choice will also depend on other factors, such as the cryptanalyst's degree of belief in what plaintext language was used.

For test statistics IC and $\diamond IND$, however, higher power was attained when distinguishing uniform noise from BCa English. This behavior results from the fact that each of IC and $\diamond IND$ has a much smaller variance on uniform noise than on BCa English, as observed in the histograms.

In all of our power calculations, as expected, the equivalent statistics S, S/N, and $\hat{S}$ achieved nearly identical powers (typically through at least three decimal places). For this reason, Tables 2–5 do not list powers for S/N or $\hat{S}$. We attribute the minor differences in their powers to experimental approximations, possibly due to approximations in computing tail areas of the normal distribution and our method for estimating observed power. Similarly, the equivalent statistics IC and $\hat{IC}_\star$ also achieved very close statistical powers.

We also computed observed powers for distinguishing 0th-order BCa from 1st-order BCa (see Figure 16 and Table 4 from our technical report [17]). At all critical levels, the statistics S and

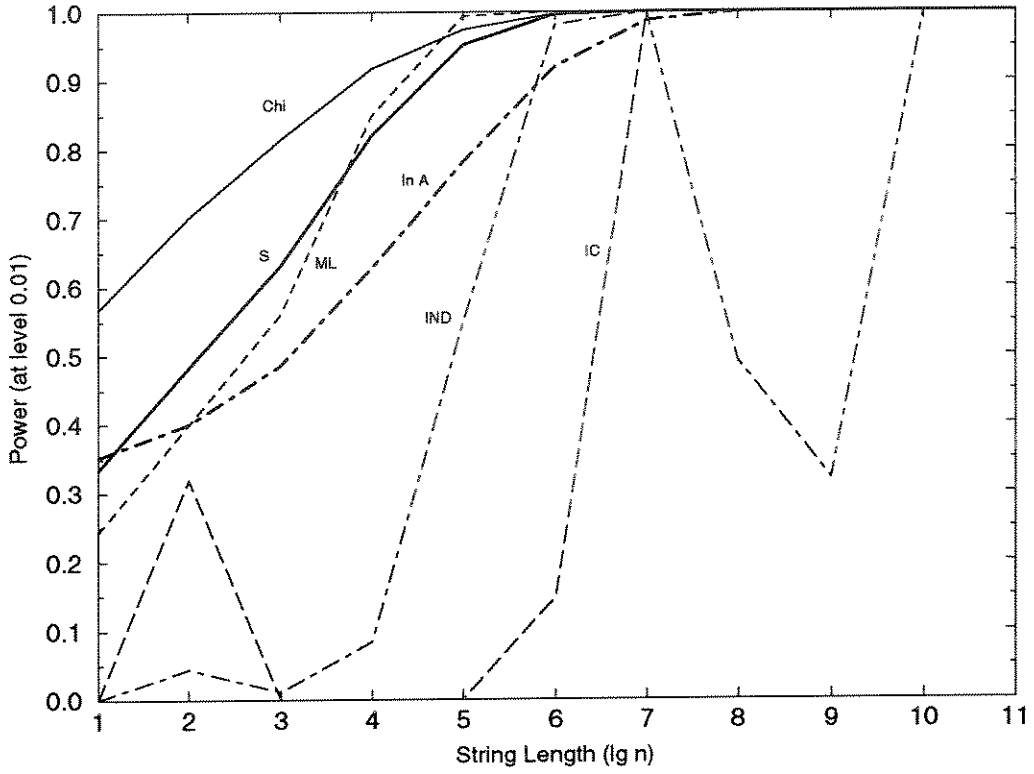

Figure 13: Power of the test statistics $\diamond X^2$, $\diamond ML$, $\diamond IND$, S, IC, and $\ln A$ at distinguishing BCa English from uniform noise at critical level 0.01 for strings of various lengths. Power was approximated from 10,000 randomly chosen strings at each length. Base language is BCa.

Table 2: Power of six test statistics at distinguishing BCa English from uniform noise at critical levels 0.1, 0.01, 0.001, 0.0001 for string lengths $n = 16, 32, 64$. Base language is BCa.

$\lg n$	Critical Level	$\diamond X^2$	$\diamond ML$	$\diamond IND$	S	IC	$\ln A$
4	0.1000	0.9350	0.9462	0.3834	0.9009	0.0477	0.7452
4	0.0100	0.9183	0.8509	0.0839	0.8214	0.0000	0.6281
4	0.0010	0.9042	0.7338	0.0150	0.7433	0.0000	0.5333
4	0.0001	0.8914	0.6110	0.0024	0.6677	0.0000	0.4536
5	0.1000	0.9823	0.9991	0.8894	0.9786	0.3168	0.8667
5	0.0100	0.9747	0.9940	0.5501	0.9523	0.0002	0.7831
5	0.0010	0.9676	0.9804	0.2494	0.9200	0.0000	0.7062
5	0.0001	0.9607	0.9549	0.0906	0.8828	0.0000	0.6349
6	0.1000	0.9981	1.0000	0.9996	0.9989	0.9988	0.9588
6	0.0100	0.9969	1.0000	0.9815	0.9966	0.1466	0.9205
6	0.0010	0.9956	1.0000	0.8723	0.9927	0.0000	0.8787
6	0.0001	0.9942	1.0000	0.6390	0.9869	0.0000	0.8341

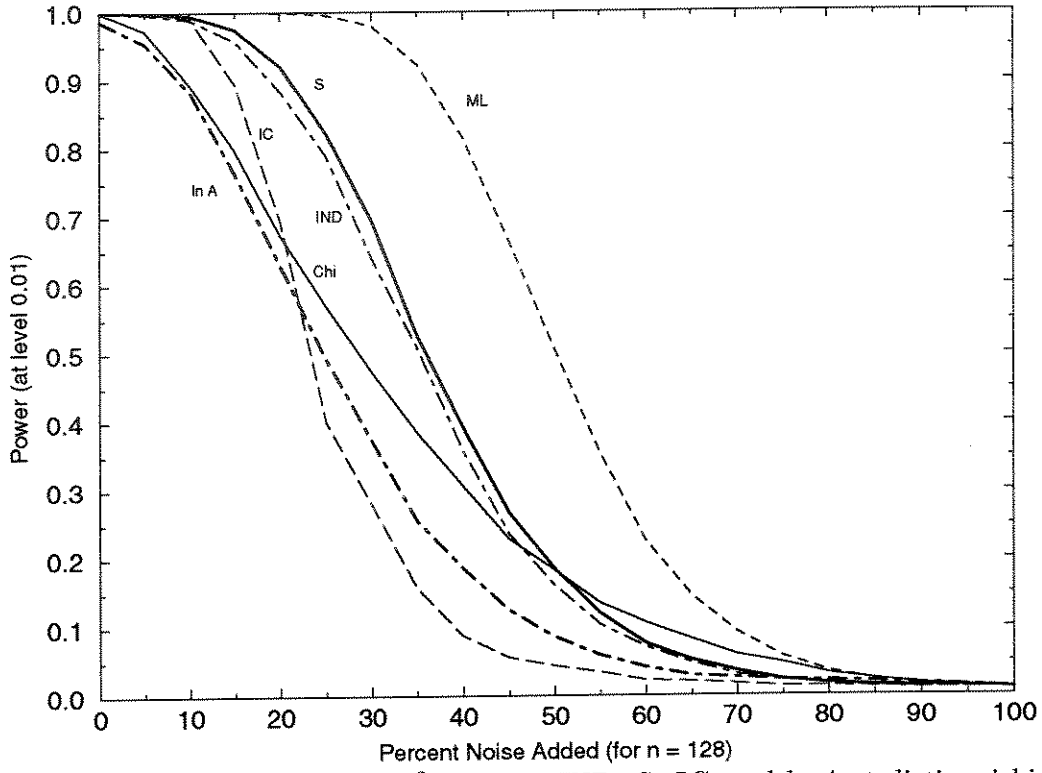

Figure 15: Power of the test statistics $\diamond X^2$, $\diamond ML$, $\diamond IND$, S, IC, and $\ln A$ at distinguishing BCa English from uniform noise at critical level 0.01 with various amounts of added uniform noise for strings of length $n = 128$. Power was approximated from 10,000 randomly chosen strings at each noise level. Base language is BCa.

Table 4: Power of six test statistics at distinguishing BCa English from uniform noise at critical levels 0.1, 0.01, 0.001, 0.0001 for strings of length $n = 128$ at noise levels 10%, 20%, and 30%. Base language is BCa.

% Added Noise	Critical Level	$\diamond X^2$	$\diamond ML$	$\diamond IND$	S	IC	$\ln A$
10	0.1000	0.9754	1.0000	1.0000	0.9993	1.0000	0.9557
10	0.0100	0.8938	1.0000	0.9893	0.9956	0.9919	0.8840
10	0.0010	0.7645	1.0000	0.8293	0.9859	0.1807	0.7951
10	0.0001	0.6132	1.0000	0.4372	0.9672	0.0001	0.6981
20	0.1000	0.8907	1.0000	0.9988	0.9836	1.0000	0.8276
20	0.0100	0.6735	0.9997	0.8845	0.9212	0.6952	0.6301
20	0.0010	0.4519	0.9950	0.4412	0.8122	0.0157	0.4540
20	0.0001	0.2774	0.9709	0.1047	0.6745	0.0000	0.3141
30	0.1000	0.7687	0.9993	0.9853	0.9005	0.9897	0.6367
30	0.0100	0.4733	0.9797	0.6396	0.6948	0.2796	0.3735
30	0.0010	0.2569	0.8872	0.1653	0.4772	0.0034	0.2078
30	0.0001	0.1281	0.6999	0.0193	0.3002	0.0000	0.1115

4.4 Discussion

We now discuss several issues raised by our power calculations.

• First, in Figures 13 and 14, the power curve for $\diamond IND$ has an unusual downward spike at string lengths $n = 256, 512$. In addition, the IC power curve in Figure 13 has an upward spike at $n = 4$. The $\diamond IND$ spikes are most prominent when distinguishing BCa from uniform noise (a task for which *IND* was not intended), and absent when distinguishing 0th-order BCa from 1st-order BCa (a task for which *IND* was explicitedly designed). When distinguishing uniform noise from BCa, the $\diamond IND$ spikes are present but less pronounced.

When we had first observed the $\diamond IND$ spikes from our initial run of Experiment 1, we had suspected that they might be statistical anomalies due to our small sampling size of $h = 100$. But the spikes reappeared when we repeated Experiment 1 with $h = 10,000$. Moreover, from the repeatability of the phenomenon, we cannot simply attribute the spikes to statistical anomalies or to weak pseudorandom number generators. Similarly, given the short and moderate lengths of the affected strings, we cannot attribute the spikes to overlapping strings. From the means of $\diamond IND$ in Table 10 of Appendix B [17], however, we find a partial explanation: For $3 < n < 512$, the observed means for $\diamond IND$ are less on BCa than on uniform noise, but for $n \geq 512$, the means are greater on BCa than on uniform noise. The $\diamond IND$ spikes correspond to this "crossing of the means" at $n = 512$. We find this phenomenon puzzling, and we do not have a good explanation for the isolated IC spike.

• Second, it would be interesting to compare our power results with previous related power calculations from other researchers. Unfortunately, we are not aware of any such prior work that would permit a direct comparison. We can, however, make some informal comparisons with the work of Baldwin and Sherman [3], and with that of Davies and Ganesan [12].

In their solution of the Decipher Puzzle, Baldwin and Sherman recognized standard English versus 0th-order English with the $\hat{S}$ statistic using a 26-state 1st-order model of English with transition probabilities published by Beker and Piper [4]. Each of their input strings consisted of a sequence of approximately ten independent bigrams, and they rejected strings for which $|\hat{S}| > 4$. Thus, they worked with at a critical level less than 0.0001. Although Baldwin and Sherman did not compute power, they found that their test worked "very well" in practice for their application. By contrast, when distinguishing BCa from uniform noise at level 0.001 for strings of dependent bigrams, we observed the power of S to be 0.5256 for $n = 8$ and 0.7433 at $n = 16$. Thus, our power calculations seem somewhat pessimistic in comparison to the experience of Baldwin and Sherman.

In their BApasswd checker, Davies and Ganesan used an equivalent variation of the S statistic in a 27-state 2nd-order model, estimating their own transition probabilities with the Good-Turing method [19]. Davies and Ganesan rejected bad eight-letter passwords using experimentally-determined thresholds. Although they did not compute power, it is possible to estimate power from their reported data for distinguishing uniform noise from valid English. For example, consider their dictionary file BP6 (bad passwords) as valid English, and consider their file GP1 (good passwords) of randomly-generated passwords with the letters 'A'–'Z' as uniform noise. According to their Figure 7 [12], using their threshold, their test accepted 96.26% of the random GP1 passwords as noise (corresponding to a critical level of 0.0374), while accepting only 12.29% of the English words in BP6 as noise (corresponding to a power of 0.8781). By contrast, when distinguishing uniform noise from BCa at critical level 0.1, we observed the power of S to be 0.3676 at $n = 8$ and 0.9997 at $n = 16$. Thus, our power calculations also seem pessimistic in comparison to the experimence of

0.0001. Even at critical level 0.0001 and short string length $n = 8$, X^2 attained a reasonable power of approximately 0.77.

3. For distinguishing uniform noise from BCa English, *ML* had the overall best performance, with *IC*, X^2, S, and *IND* also performing well. At critical levels 0.01 through 0.0001 *IC* attained higher power than did X^2 (and than did *ML* for $n \leq 8$), but at critical level 0.1 X^2 performed better than did *IC*. For string lengths $n \leq 128$, we recommend using *ML* for critical levels 0.1 through 0.0001. For $n > 128$, each of *ML*, *IC*, X^2, S worked perfectly at these critical levels. At $n = 32$ and critical level 0.1, *ML* achieved a power of over 0.99.

4. For distinguishing BCa English from uniform noise using strings of length $n = 128$ corrupted with uniform noise, *ML* outperformed the other statistics at all critical levels. At critical levels 0.01 through 0.0001, S had the overall second-best performance. We recommend using *ML* for this problem. At $n = 128$ and critical level 0.01, the power of *ML* remained above 0.8 through noise level 40%. For noise levels 0%–15%, *ML* had attained power greater than 0.99 at critical levels 0.1 through 0.0001; for noise levels 70%–100%, all statistics had power less than 0.5 at these critical levels.

5. For distinguishing uniform noise from BCa English using strings of length $n = 128$ corrupted with uniform noise, *ML* had the overall best performance, with *IC* and *IND* also performing well. We recommend using *ML*. The performance of *ML* on this problem was very similar to, and slightly better than, its performance at distinguishing uniform noise from BCa English under noisy conditions.

6. The S statistic outperformed Anderson's variation of it, except when distinguishing BCa English from uniform noise at string length $n = 2$. Therefore, we do not recommend Anderson's variation.

7. For $n < 128$, our four types of real English (BCa, BCf, BCg, and WSJ1) had similar means for all statistics. For longer strings, the statistics could distinguish BCa, BCf, and WSJ1 on the basis of their means.

8. Strict standard normal interpretations of the normalized statistics $\diamond X^2$, $\diamond ML$, and $\diamond IND$ do not apply, except when recognizing long strings of a known simulated 1st-order language. For best results, use experimentally-determined thresholds for all statistics, including $\hat{S}$ and $\hat{IC}_\star$. These thresholds can be computed as explained in Section 3.5.

9. As expected, the performance of all statistics on BCa English closely matched their performance on simulated 1st-order BCa English. In this sense, our 1st-order statistics are robust with respect to our 1st-order model. Nevertheless, minor differences can be seen in their histograms (*e.g.* 1st-order language produced more symmetrical and bell-shaped distributions). In addition, to human observers, our 1st-order BCa strings do not closely resemble real English.

Complete experimental data, including additional descriptive graphs and power calculations, are given in the supplemental appendices of our technical report [17].

Using an exploratory, descriptive approach, we have exposed general trends and uncovered interesting phenomena. We consider these trends and phenomena more important than our particular numerical results, since models and languages vary with the application, and since our power

Acknowledgments

We are grateful to Peter Matthews for helpful remarks and suggestions. In addition, we thank James Mayfield for providing us with a copy of the Brown Corpus, and we thank Rita M. Doerr for answering some of our questions about this corpus. Thanks also to Robert Baldwin, Thomas Cain, James Mayfield, Bryan Olson, Raymond Pyle (Bell Atlantic), and James Reeds for comments. We are also grateful to Tomoko Shimakawa for computing several tail areas using SAS. Computer work was carried out on a DECstation 5000/200 and on a Silicon Graphics Iris/Indigo at the University of Maryland Baltimore County, and on a SUN Sparcstation at Bell Atlantic.

References

[1] Anderson, Roland. April 1989. Recognizing complete and partial plaintext. *Cryptologia*, 13(2): 161–166.

[2] Anderson, T. W. and Leo A. Goodman. 1957. Statistical inference about Markov chains. *Annals of Mathematical Statistics*, 28: 89–110.

[3] Baldwin, Robert W. and Alan T. Sherman. 1990. How we solved the $100,000 Decipher Puzzle: (16 hours too late). *Cryptologia*, 14(3): 258–284.

[4] Beker, Henry and Fred Piper. 1982. *Cipher Systems.* New York: John Wiley.

[5] Bhat, U. Narayan. 1984. *Elements of Applied Stochastic Processes.* New York: John Wiley.

[6] Billingsley, Patrick. 1961. *Statistical Inference for Markov Processes.* Chicago: University of Chicago Press.

[7] Billingsley, Patrick. 1961. Statistical methods in Markov chains. *Annals of Mathematical Statistics*, 32(1): 12–40.

[8] Callimahos, Lambros D. and William F. Friedman. 1959. *Military Cryptanalytics Part II*, Volume 1. Washington, DC: United States Government. [Available through Aegean Park Press, Laguna Hills, CA.]

[9] Church, Kenneth W. and William A. Gale. 1991. A comparison of the enhanced Good-Turing and deleted estimation methods for estimating probabilities of English bigrams. *Computers, Speech, and Language*, 5(1).

[10] Crook, J. F. and I. J. Good. 1980. On the application of symmetric Dirichlet distributions and their mixtures to contingency tables, Part II. *Annals of Statistics*, 8(6): 1198–1218.

[11] Crook, James Flinn and Irving John Good. 1982. The powers and strengths of tests for multinomials and contingency tables. *Journal of the American Statistical Association*, 77(380): 793–802.

[12] Davies, Chris I. and Ravi Ganesan. September 1993. BApasswd: A new proactive password checker. *Proceedings of the 16th National Computer Security Conference.* 1–15.

[13] Francis, W. Nelson and Henry Kučera; with the assistance of Andrew W. Mackie. 1982. *Frequency Analysis of English Usage: Lexicon and Grammar.* Boston: Houghton-Mifflin.

[14] Friedman, William F. 1925. The index of coincidence and its applications in cryptanalysis. Technical Paper, War Department, Office of the Chief Signal Officer. Washington, DC: United States Government Printing Office. [Available through Aegean Park Press, Laguna Hills, CA.]

[35] Schrift, A. W. and A. Shamir. 1993. Universal tests for nonuniform distributions. *Journal of Cryptology*, 6(3): 119–133.

[36] Sinkov, Abraham. 1966. *Elementary Cryptanalysis: A Mathematical Approach*, New Mathematical Library No. 22. Washington D.C.: The Mathematical Association of America.

[37] Trivedi, Kishor Shridharbhai. 1982. *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*, Englewood Cliffs, NJ: Prentice Hall.

[38] West, Eric N. and Oscar Kempthorne. 1972. A comparison of the Chi^2 and likelihood ratio tests for composite alternatives. *Journal of Statistical Computation and Simulation*, 1: 1–33.

About the Authors

Ravi Ganesan is currently Manager of the Center of Excellence for Electronic Commerce at at Bell Atlantic. In this position he leads teams which are responsible for the implementation of Electronic Data Interchange projects and which seek to research, develop and introduce innovative electronic commerce services. He is also responsible for the invention and development of several new security tools. He is an organizer and co-Program Chair of the ACM Conference on Computer and Communications Security. Ravi is a member of the IEEE, ACM, and *Phi Kappa Phi*. He holds a Bachelor of Engineering from Anna University and a Masters from the University of Maryland Baltimore County, both in computer science, and is currently a doctoral student in the Department of Computer Science at The Johns Hopkins University. His current research interests are in computer security, approximation algorithms, and heuristic search.

Alan T. Sherman is an assistant professor of computer science at the University of Maryland Baltimore County. He received his Ph.D. in computer science from the Massachusetts Institute of Technology in February 1987, his S.M. in electrical engineering and computer science from the Massachusetts Institute of Technology in June 1981, and his Sc.B. in mathematics, *magna cum laude*, from Brown University in June 1978. Sherman is a member of ACM, AMS, IACR, IEEE, SIAM, *Phi Beta Kappa*, and *Sigma Xi*. His main research areas are discrete algorithms and cryptology.