Statistical Techniques for Language Recognition: An Introduction and Guide for Cryptanalysts

Ravi Ganesan
Security Planning and Research
Bell Atlantic
Beltsville, Maryland 20705

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

February 25, 1993

Abstract

We explain how to apply statistical techniques to solve several language-recognition problems that arise in cryptanalysis and other domains. Language recognition is important in cryptanalysis because, among other applications, an exhaustive key search of any cryptosystem from ciphertext alone requires a test that recognizes valid plaintext. Written for cryptanalysts, this guide should also be helpful to others as an introduction to statistical inference on Markov chains.

Modeling language as a finite stationary Markov process, we adapt a statistical model of pattern recognition to language recognition. Within this framework we consider four well-defined language-recognition problems: 1) recognizing a known language, 2) distinguishing a known language from uniform noise, 3) distinguishing unknown 0th-order noise from unknown 1st-order language, and 4) detecting non-uniform unknown language. For the second problem we give a most powerful test based on the Neyman-Pearson Lemma. For the other problems, which typically have no uniformly most powerful tests, we give likelihood ratio tests. We also discuss the chi-squared test statistic X^2 and the Index of Coincidence IC. In addition, we point out useful works in the statistics and pattern-matching literature for further reading about these fundamental problems and test statistics.

Keywords. Automatic plaintext recognition, categorical data, chi-squared test statistic, computational linguistics, contigency tables, cryptanalysis, cryptography, cryptology, hypothesis testing, Index of Coincidence, language recognition, likelihood ratio tests, Markov models of language, maximum likelihood estimators, statistical inference, statistical pattern recognition, statistics of language, weight of evidence.

*Most 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

In cryptanalysis, how can a computer program recognize when it has discovered all or part of the secret message?[1] For example, how can a program recognize character strings such as "Attack at dawn!", "DES@RT ST&RM", or "?tta????t d?wn" as fragments of intelligible messages? In the early days of cryptology, a human would perform these language-recognition tasks manually. This paper explains how to recognize language automatically with statistical techniques.

Recognizing "valid" plaintext is a crucial step in many cryptanalytic tasks. For example, this step can enable a cryptanalyst to carry out an exhaustive key search of any cryptosystem, given ciphertext only: For each candidate key, the cryptanalyst checks the candidate key by decrypting the given ciphertext under the candidate key and by inspecting the resulting candidate plaintext. Were corresponding plaintext also available, the cryptanalyst would simply accept the candidate key if the candidate plaintext and known plaintext matched. With ciphertext only, however, the cryptanalyst accepts the candidate key when the candidate plaintext appears to be a valid message.

Language recognition is also useful in searching for part of the secret key, since a portion of the key may produce a fragment of the message. For example, in his Crypt Breaker's Workbench, Baldwin [5] exploits language-recognition techniques to guess unknown wires in the Unix crypt cipher [74]. In addition, language recognition can enable a cryptanalyst to align adjacent columns of plaintext in transposition ciphers and to complete partial solutions of polyalphabetic substitution ciphers. Furthermore, in their new ciphertext-only attack on filter generators, Cain and Sherman [16, 17] apply a language-recognition subroutine to detect when they have discovered part of the initial fill. Related statistical techniques are also useful in breaking the Hagelin cryptograph [75] and various rotor machines [2]. Despite the importance of recognizing valid plaintext, the cryptologic literature provides little practical advice on how to automate this task.

Language recognition for cryptanalysis must deal with the following three constraints. First, cryptology is adversarial in nature. Therefore, the sender will not intentionally assist the cryptanalyst (*e.g.* through adaptive training exercises), and the sender might maliciously attempt to fool the cryptanalyst (*e.g.* by modifying the plaintext). Second, candidate plaintext can be short and incomplete. For example, the cryptanalyst might be able to decipher only a few isolated fragments of plaintext. Third, candidate plaintext is often contaminated with noise. For example, errors can appear in the candidate plaintext due to imperfect cryptanalysis or as a result of the sender injecting random bits into the plaintext. In addition, it must be possible to recognize valid plaintext even when language is unknown or broadly defined (*e.g.* any human or machine language).

In their solution of the $100,000 Decipher Puzzle, Baldwin and Sherman [6] devised a statistical technique for recognizing English messages. They compared the observed bigram frequencies in the candidate plaintext with those expected in a standard English message of the same length.[2] This engineering approach worked well for their application and motivated us to investigate the language-recognition problem more closely.

Although our motivation is cryptanalysis, language-recognition problems also arise in many non-cryptologic domains, including pattern recognition, voice recognition, optical character recognition, image analysis, submarine detection, and speaker identification. Therefore, we expect our work to

[1] We assume the reader is familiar with basics of cryptology—as explained by Beker and Piper [8], Denning [20], Rivest [76], or Simmons [81], for example. We also assume the reader is familiar with elementary statistics—as explained by Hoel [48] or Larsen and Marx [62].

[2] We use the term *k-gram* to refer to any sequence of exactly k letters. For $k = 1, 2, 3$, we call any such *gram* a *unigram*, *bigram*, or *trigram*, respectively.

apply in many of these other settings. For example, techniques described in this paper can enable a computerized telephone operator to determine what language the caller is speaking. Moreover, because of cryptology's challenging constraints, techniques that work well for cryptanalysis tend also to work well in other less demanding applications.

In this paper we answer the following questions. What is a useful framework in which to reason about language recognition, and what is a useful model of language? Within this framework, what does it mean—in a precise mathematical sense—to recognize language? In particular, what are some important well-defined language-recognition problems that arise in cryptanalytic practice? What are effective methods for solving each problem? What are useful criteria by which to evaluate these methods? Are there optimal techniques?

We approach language recognition from a statistical model of pattern recognition in which we model language as a finite stationary Markov process. Within this model, we take "valid messages" as strings generated from the Markov chain. We state four well-defined language-recognition problems and identify test statistics for each problem. Specifically, we derive likelihood ratio tests. For what we call Problem 2 (distinguishing a known language from uniform noise), the resulting test is optimal in the sense that it is a "most powerful test." In addition, we discuss the chi-squared test statistic X^2 and the Index of Coincidence IC. Along the way, we also point out additional approaches from the statistics literature.

Whereas this guide explains our framework, language model, problems, and test statistics, our companion experimental paper [29] examines how well these methods work in practice, especially when applied to natural language as opposed to the idealized language of the model. We hope our guide will be helpful to cryptanalysts and to others as an introduction to statistical techniques for language recognition.

The rest of this paper is organized as follows. Section 2 briefly reviews previous related work from the cryptologic literature. Section 3 explains how we apply a statistical model of pattern recognition to language recognition. Section 4 describes how we model language as a Markov chain. Section 5 states four well-defined plaintext-recognition problems and gives examples of how they apply in cryptanalytic practice. Section 6 reviews basic concepts and terminology from statistical inference, including the notion of a most powerful test and other criteria for evaluating test statistics. Section 7 defines and explains several tests statistics, including a most powerful test for Problem 2. Section 8 briefly discusses three practical variations of these techniques for dealing with noisy plaintext, short plaintext, and plaintext that consists of multiple strings. This section also gives pointers to further reading in the statistics and pattern-recognition literature and states several open questions. Finally, Section 9 summarizes our conclusions.

2 Previous Cryptologic Work

Previous research relevant to language recognition is scattered throughout several diverse disciplines, including statistics, pattern recognition, signal processing, computational linguistics, algorithms, and cryptology. Collectively, there is a large body of pertinent knowledge. But little of this prior art deals explicitly with the cryptanalytic perspective. In particular, in books and journals devoted to cryptology, only three methods have been previously suggested for recognizing valid plaintext: Sinkov's test and variations thereof by R. Anderson and by Baldwin and Sherman. In this section we briefly review these methods, together with another useful statistic known as the *Index of Coincidence (IC)*. For a guide to further readings in the statistics and pattern-recognition literature, see Section 8.2.

Sinkov's Statistic

The little that is written about language recognition for cryptanalysis is based on a log-weight statistic suggested in passing by Sinkov [82, pp. 76–77] in the context of breaking Vigenère ciphers. Specifically, for any sequence $X = x_1 x_2 \ldots x_n$ of n unigrams, the Sinkov statistic S_1 scores the candidate plaintext X by the formula $S_1(X) = \sum_{i=1}^{n} \ln p_{x_i}$, where for each i, p_{x_i} denotes the *a priori* unigram probability of the unigram x_i.[3] For example, if the *a priori* unigram probabilities of the letters 'D', 'E', 'S' are 0.044, 0.130, 0.063, respectively, then $S_1(\text{"DES"}) \approx -3.12 - 2.04 - 2.76 = -7.92$. Intuitively, the Sinkov statistic measures the likelihood of the observed plaintext.

Unfortunately, Sinkov neither explains the theoretical underpinnings of this remarkable statistic, characterizes its distribution, nor gives a decision procedure for accepting or rejecting candidate plaintexts on the basis of their S_1 values. Thus, with what confidence should we accept the plaintext "DES" as valid on the basis of a score of -7.92? The situation becomes more difficult if we must compare strings of different lengths because Sinkov does not explain how the distribution of his statistic changes with length, especially when applied to higher-order grams. As for how to accept or reject a candidate plaintext, Sinkov simply says to try all possibilities and to pick the one with the highest S_1 value. Although this procedure works for some applications, it is inadequate for applications that require on-line decisions. Furthermore, it is desirable to have a meaningful interpretation of the S_1 values.

Sinkov [82, p. 162] also points out that his statistic, applied to bigrams, is an effective way to measure how well two columns fit next to each other when anagramming columns of ciphertext in solving single-columnar transposition ciphers. But he does not address the interesting issue of whether the statistic should be computed using the *unconditional bigram probabilities* $p_{x_i x_{i+1}}$ (the *a priori* probability of the bigram $x_i x_{i+1}$) or using the *conditional bigram probabilities* $p_{x_i x_{i+1}|x_i}$ (the *posterior* probability of the bigram λx_{i+1} given that $\lambda = x_i$).[4] By default, Sinkov seems to suggest using unconditional bigram probabilities. This issue deals primarily with the question of whether the input is a sequence of *independent bigrams* from isolated fragments of text or a sequence of *dependent bigrams*, such as the "overlapping" bigrams "DE" and "ES" from the string "DES". To compute the likelihood of strings generated from our Markov model, for independent bigrams unconditional bigram probabilities should be used, and for dependent bigrams conditional bigram

[3] Throughout this paper, let $\ln = \log_e$.

[4] In anticipation of our Markov model, we shall also refer to these conditional probabilities as *transition probabilities*.

probabilities should be used.[5] Furthermore, the issue can be complicated by assumptions concerning the initial conditions of the string: using unigram initial conditions and conditional bigram probabilities on a string of length one yields the same effect as using unconditional bigram probabilities without initial conditions (see Section 8.1). When computed on bigrams with conditional bigram probabilities, we shall call the Sinkov statistic S_2.

Variations on Sinkov's Statistic

Toward the goal of separating vowels from consonants in substitution ciphers, Anderson [3] recommends a simple variation of the Sinkov statistic, which for any $1 \leq k \leq n$, operates on k-grams rather than on unigrams only. For any $1 \leq k \leq n$, Anderson computes $A_k = \prod_{i=1}^{N_k} \xi_i^{n/(kN_k)}$, where $N_k = n - k + 1$ is the number of k-grams in the candidate plaintext and ξ_i is the unconditional probability (not the transition probability) of the k-gram $x_i x_{i+1} \ldots x_{i+k-1}$.[6] Thus, for $k = 2$, $\ln A_2 = (n/(2N_2)) \sum_{i=1}^{N_2} \ln \xi_i \approx (S_1 + S_2)/2$, since $N_2 = n - 1$ and, for all $1 \leq i \leq n - 1$, $\xi_i = p_{x_i x_{i+1}} = p_{x_i} p_{x_i x_{i+1}|x_i}$. As with Sinkov, Anderson does not explain how to interpret his statistic.

Baldwin and Sherman [6] center and normalize S_2/N by computing $\hat{S}_2 = \sqrt{N}((S_2/N) - \mu)/\sigma$, where N is the number of bigrams in the candidate plaintext. The constants μ and σ are, respectively, the mean and standard deviation of S_2 applied to a randomly chosen bigram of English, with the bigram selected using unconditional bigram probabilities (see Section 7.5).[7] To interpret their statistic, these two cryptanalysts view it as a standard normal random variable, rejecting all values that fall outside some interval $[-\tau, \tau]$.[8] In solving the Decipher Puzzle, they arbitrarily selected $\tau = 4$. Although the central limit theorem [64] guarantees that $\hat{S}_2$ is asymptotically standard normal when applied to independent English bigrams (the special case of their application), Baldwin and Sherman say nothing about the distribution of $\hat{S}_2$ when applied to dependent bigrams. In addition, they provide no experimental evaluation of how well their method works in practice. After presenting our framework for language recognition, in Section 7 we will revisit the S_2 and $\hat{S}_2$ statistics from the new perspective of this framework.

The Index of Coincidence

The Index of Coincidence is another well-known tool for solving several related cryptanalytic problems. For unigrams, it is defined by $IC = \sum_{\lambda=1}^{m} f_\lambda(f_\lambda - 1)/(n(n-1))$, where n is the length of the candidate plaintext; m is the alphabet size; and for all letters $1 \leq \lambda \leq m$, f_λ is the observed frequency of letter λ. This statistic is an unbiased estimator of the *repeat rate* $\rho = \sum_{\lambda=1}^{m} p_\lambda^2$, the *a priori* probability that two randomly chosen unigrams from the language coincide. Intuitively, the IC measures the roughness of the distribution of characters in the candidate plaintext. According to Kahn [52, p. 376–382], the IC is due to Friedman [25].

---

[5] Although their input was a sequence of independent bigrams, Baldwin and Sherman [6] used conditional bigram probabilities and analyzed why their choice worked well for their application.

[6] There is a typographical mistake in Anderson's [3, p. 162] formula: using Anderson's notation, in his display equation the upper bound of the product should be $N - n + 1$ rather than $N - n - 1$.

[7] Baldwin and Sherman computed $\mu \approx -2.51$ and $\sigma \approx 0.98$ from the language statistics given by Beker and Piper [8].

[8] A random variable is *standard normal* if and only if it has a Gaussian distribution with mean 0 and variance 1.

Sinkov [82] points out two applications of the IC. First, as a measure of roughness of the distribution of observed ciphertext characters, the IC can help identify the unknown encryption scheme. Second, it can be similarly used to estimate the period of polyalphabetic substitution ciphers; for example, Beker and Piper [8] give a formula for doing so. Although Sinkov does not explicitly say so, one could also use the IC to help identify language types. For more about the IC, see Section 7.4.

3 A Framework for Language Recognition

We adopt a classical statistical framework in which plaintext recognition is performed in two steps:

Step 1 (off-line): Identify features of the base language from a base sample.

Step 2 (on-line): Compare features of the candidate plaintext with those of the base language.

The off-line step observes the base language through a *base sample* and extracts *features* from the base sample. Similarly, the on-line step observes the candidate plaintext from an *on-line sample* and extracts features from the on-line sample. In addition, the on-line step compares these features with those of the base language, and accepts or rejects the candidate plaintext by interpreting the value of a *test statistic* according to a *decision procedure*.

We find it helpful to view the role of base language and candidate plaintext as symmetrical: each is observed through samples consisting of character strings, and the same features are extracted from these samples. The base sample, however, is typically huge in comparison with the on-line sample. Furthermore, since the base sample is available off-line, the cryptanalyst can afford to spend a large amount of time preprocessing the base sample; by contrast, typically the on-line step must be performed quickly since it is calculated many times.

In our model, the extracted features are the frequency counts of the k-grams, for some k.[9] Hence, within our model, the only information from which the cryptanalyst can make a decision is the frequency counts of the base and on-line samples.

We view the test statistic as computing a "distance" between the features of the on-line sample and those of the base sample. Each of these samples defines a point in "feature space"—the space of all possible feature values (*e.g.* the space of bigram frequencies). Any candidate plaintext is considered valid if and only if it is sufficiently close in distance to the base language in this space. Thus, the decision procedure defines a region in feature space around the base sample in which all on-line samples are accepted. More generally, for classification problems involving two or more base languages, the decision procedure partitions feature space into regions, one corresponding to each base language.

Throughout this paper, we assume that the cryptanalyst has enough ciphertext to be able to distinguish valid messages from invalid messages. Thus, when applying language-recognition techniques in an exhaustive key search of any cipher, as a minimum information-theoretic requirement, we assume that the length of the ciphertext equals or exceeds the unicity distance of the cipher.[10]

[9] In our experimental work [29] we use $k = 2$.

[10] As formalized by Shannon [79], the *unicity distance* of a cipher is the minimum number of ciphertext characters required to guarantee that, for any cryptogram, the expected number of spurious decipherments is approximately zero.

For some ciphers, stronger assumptions may be needed to ensure further that the cryptanalyst can recognize valid plaintext efficiently.

For those applications when the cryptanalyst does not know the base language, we omit Step 1 and apply a statistic that detects some type of structure in the on-line sample. The next four sections explain how to perform the off-line and on-line steps.

4 A Markov Model of Language

We model any language as the set of finite strings that can be generated by a Markov chain, together with the probability distribution on these strings induced by the chain.[11] Markov chains provide a convenient, well-defined, well-understood model that captures useful statistical properties of language—especially the dependencies among successive characters of a string. By selecting an appropriate set of *parameters* (*i.e.* states, order, and transition probabilities), this model can be customized to represent a variety of languages, including human languages (*e.g.* French) and programming languages (*e.g.* C). For these reasons, and since Markov chains seem to work well in practice, we adopt this popular model. In the rest of this section, we explain our model of language; we explain how to estimate its parameters from a base sample; and we point out some sources of language statistics for English and for other natural languages.

What is a Markov Model?

We represent a *finite Markov chain* as a quadruple $(m, \mathcal{A}, \{Y_t\}, r)$, where $m \in \mathbb{Z}^+$ is the number of states, and $\mathcal{A} = \{a_1, a_2, \ldots, a_m\}$ is the *state space*.[12] Usually we associate the states with the letters of the alphabet. For each time $t \in \mathbb{Z}^+$, Y_t is a random variable that takes on a value in $\mathcal{A}$; this random variable describes the state of the chain at time t. The parameter $r \in \mathbb{Z}^+$ is the *order* of the chain, the maximum number of previous states on which each random variable Y_t depends. It is convenient to think of the chain as generating strings over $\mathcal{A}$ by outputting the values of Y_t.

For all $t > 1$, the distribution of Y_t given Y_{t-1} is defined by transition probabilities. For each $1 \le i, j \le m$, the *transition probability* $p_{i,j}$ is the *posterior* probability that the chain will move from state i to state j, given that the chain is in state i. For any Markov chain of order 1, the transition probabilities can be described by an $m \times m$ matrix. A more elaborate Markov model might also include an initial distribution of states. Unless otherwise noted, we shall assume that the Markov model for each base language is well-behaved in the technical sense that it is ergodic.[13] To construct a Markov model of a base language, it is necessary to select the state space, order, and transition probabilities.

Choosing the State Space

It is natural to associate the states with the letters that appear in the base language. But if the alphabet is large and contains many infrequent letters, it may be convenient to reduce the number of states in the Markov model by partitioning the alphabet into equivalence classes. For example, one

[11] Unless otherwise noted, we shall use the phrase *Markov chain* to mean a finite stationary Markov process. For a review of Markov chains, see Bhat [9], Billingsly [12], or Kemeny [56].

[12] Throughout this paper, we denote the set of positive integers by the symbol $\mathbb{Z}^+$.

[13] A Markov chain is *ergodic* if and only if it is stationary, recurrent, and aperiodic—see Bhat [9, pp. 47–48].

might lump all low-frequency letters into one equivalence class. Doing so ameliorates one difficulty that results from having a small number of observed grams for the infrequent transitions—albeit at the loss of some information: To interpret the X^2 statistic in the standard way, statisticians require that certain conditions be satisfied. For instance, one common condition—arguably too strong—is to require that the expected number of observations for each category (*i.e.* gram) be at least 5. For low-frequency grams, a long on-line sample is required to satisfy the condition; by condensing states, one can reduce the required length of the on-line sample.

Another simple way to condense states is to order the bigrams by decreasing bigram probabilities and to assign a state to each letter in the most probable bigrams. All remaining letters could be lumped into an additional state. An interesting and extreme example of condensing states is to model language with two states: vowel and consonant, as discussed by Bishop, Fienberg, and Holland [13, pp. 272–273].

Choosing the Order

Choosing the order of the model is an important decision. The higher the order, the more accurate the model—up to a point. For example, for English, a 2nd-order (trigram) model is more accurate than a 1st-order (bigram) model, which is more accurate than a 0th-order (unigram) model. But little, if anything, would likely be gained from using, say, a 20th-order model rather than a 15th-order model for English. Furthermore, higher-order models are more cumbersome than lower-order models, and the space required for their representations grows exponentially with order.

Sometimes, constraints of the application lead to a natural choice of order. In particular, in some cryptographic applications, high-order grams are not available. For example, Baldwin and Sherman [6] chose a 1st-order model in part because unigrams and bigrams were the only complete grams available to them. We note, however, that although Baldwin and Sherman did not do so, they could have used higher-order *incomplete* grams, such as "t?e", where the wild card '?' matches any single character.

Some readers may wonder, Why not simply recognize English by checking if the candidate plaintext contains words from an English dictionary? Indeed, dictionary methods are extremely powerful when they can be carried out. We chose not to pursue dictionary attacks because we are primarily interested in general-purpose statistical methods and because dictionary methods can be analyzed in the context of high-order Markov models. Moreover, it is not possible to carry out such attacks when high-order grams are not available from the candidate plaintext, as was the case with Baldwin and Sherman. In addition, the adversary can easily complicate such attacks by adding noise to the plaintext—though, even in the presence of noise, it is sometimes possible to carry out related pattern-word attacks.

Estimating the Transition Probabilities

Once the state space and order of the Markov model of the base language have been chosen, the transition probabilities of this chain can be estimated from the base sample. A straightforward way of estimating these probabilities is to use a maximum-likelihood estimator, as described for example by Bhat [9, p. 140]. Consider a 1st-order model, and let i, j be any letters. Having observed n_{ij} instances of the bigram ij in the base sample, the maximum-likelihood estimate of the transition probability p_{ij} is $\hat{p}_{ij} = n_{ij}/n_{i*}$, where n_{i*} denotes the number of observed bigrams that begin with the letter i. This process generalizes naturally to any order model.

In some applications it is helpful to adjust these maximum-likelihood estimates using a *shrinkage* technique, to move the estimated transition probabilities toward the uniform probability $1/m$. For example, the maximum-likelihood estimates can yield poor results when the the sample is *absolutely small* (*e.g.* $n_{ij} < 3$), or when the sample is *effectively small* (*e.g.* $n_{ij} \ll n_{i*}/m$) even though n_{i*} might be absolutely large. Shrinkage extends the applicability of the asymptotic theory to smaller sample sizes.

One shrinkage technique, known as *flattening*, computes the flattened estimate $\hat{p}^{\star}_{ij}$ of the transition probability p_{ij} by the formula

$$\hat{p}^{\star}_{ij} = \frac{n_{ij} + c}{n_{i*} + cm}, \tag{1}$$

where c is the so-called *flattening constant*. Two simple methods are to choose $c = 0.5$ or $c = 1/m$. Instead of directly adjusting the estimated parameters, it is possible to create an equivalent effect by appropriately modifying the test statistics. For a discussion and comparison of various flattening techniques, see Good [39, 40].

Language Statistics for Natural Languages

To carry out many tests described in this paper, it is necessary to know transition probabilities of the base language. Some practitioners may choose to estimate their own transition probabilities; others may prefer to use published language statistics, such as those reported by Solso, Juel, King, and Rubin [83, 84, 85, 86]. For additional sources of language statistics, see Shannon's study of the entropy and redundancy of English [80], Good's survey of the statistics of language [34, pp. 577–578], and cryptology texts by Beker and Piper [8], Denning [20], Friedman [26], and Kullbach [60].

In addition to computing unigram, bigram, and trigram frequencies, Solso and his associates also compute: versatility of unigrams, bigrams, and trigrams [83, 86]; positional frequency and versatility of unigrams and bigrams [84, 86]; and frequency and versatility of initial and final letters [85]. The *positional frequency* of a gram is the number of times the gram appears in the specified position (within all words); the *versatility* of a gram is the number of different words in which the gram appears. Solso and associates compute these statistics from word-frequency lists compiled from the Brown Corpus by Francis and Kučera [24, 59]. These positional frequencies and versatilities can be used in more elaborate models of language.

In practice, it is important to know how broadly to define valid plaintext. There are many different variations of each natural language. For example, *New York Times* sports English is different from the poetry English of e. e. Cummings. Some applications may need to recognize only Shakespearian English; other applications may need to recognize any language coarsely resembling English. To handle this concern, parameters of the base language and thresholds of the decision procedure must be chosen appropriately.

5 Four Language-Recognition Problems

We now formalize four plaintext-recognition problems as hypothesis-testing problems in statistical inference on Markov chains: 1) recognizing a known language, 2) distinguishing a known language from uniform noise, 3) distinguishing unknown 0th-order noise from unknown 1st-order language, and 4) detecting non-uniform unknown language. We also illustrate each problem with a concrete

example. These four well-defined problems abstract many of the practical language-recognition tasks faced in cryptanalysis.

5.1 Problem Statements

We state each problem as a hypothesis-testing problem in which we test some *null hypothesis* H_0 versus an *alternative hypothesis* H_1, given the features of the candidate plaintext. For each problem, the candidate plaintext X is one string of n characters, and the features extracted from X are the k-gram frequency counts, for some fixed k. Furthermore, for H_0 and H_1, we assume that X was produced by some Markov chain M with alphabet $\mathcal{A}$ and matrix of transition probabilities P_M, and we assume that the cryptanalyst knows an upper bound (*e.g.* 1) on the order of M. We define four problems by stating different hypotheses H_0 and H_1 concerning the parameters of M.

Problem 1 (Recognizing a Single Known Language).
Given candidate plaintext X, test the null hypothesis "X was produced by the known Markov chain B" versus the alternative hypothesis "X was produced by a different Markov chain of the same order." That is, $H_0 : P_M = P_B$, and $H_1 : P_M \neq P_B$, where P_M is the matrix of transition probabilities for the Markov chain that produced X, and P_B is a known matrix of transition probabilities.

Problem 2 (Distinguishing a Known Language from Uniform Noise).
Given candidate plaintext X, test the null hypothesis "X was produced by the known Markov chain B" versus the alternative hypothesis "X was produced by the uniform noise generator U." That is, $H_0 : P_M = P_B$, and $H_1 : P_M = P_U$, where P_U is the matrix of uniform transition probabilities.

Problem 3 (Distinguishing Unknown 0th-Order Noise from Unknown 1st-Order Language).
Given candidate plaintext X, test the null hypothesis "X was produced by some unknown 0th-order Markov chain" versus the alternative hypothesis "X was produced by some unknown 1st-order Markov chain." That is, $H_0 : \text{order}(M) = 0$, and $H_1 : \text{order}(M) = 1$.

Problem 4 (Detecting Non-Uniform Unknown Language).
Given candidate plaintext X, test the null hypothesis "X was produced by the uniform noise generator U" versus the alternative hypothesis "X was produced by a different Markov chain of a known order." That is, $H_0 : P_M = P_U$, and $H_1 : P_M \neq P_U$.

In Problems 1 and 2, H_0 represents the hypothesis that the plaintext is valid; in Problems 3 and 4, H_1 plays this role. As explained in Section 6.2, this reversal of roles between H_0 and H_1 affects how the cryptanalyst will select critical values for decision procedures.

To define the problems, it is necessary to specify what is valid plaintext. For this purpose, let B denote a specific 1st-order Markov chain with alphabet $\mathcal{A}$ and transition probabilities P_B. In Problems 1 and 2, chain B sharply defines the language of valid plaintext; for these two problems, we assume the cryptanalyst knows the order, alphabet, and transition probabilities of B. In Problem 3, we assume only that valid plaintext comes from some unknown 1st-order chain. In Problem 4, we

Table 1: Summary of our four language-recognition problems. Each problem is defined by different hypotheses H_0 and H_1 about the transitions probabilities P_M of the Markov chain M that produced the candidate plaintext. Here, P_B is a known matrix of transition probabilities, and P_U is the matrix of uniform transition probabilities.

Problem	H_0	H_1
1	$P_M = P_B$	$P_M \neq P_B$
2	$P_M = P_B$	$P_M = P_U$
3	$\text{order}(M) = 0$	$\text{order}(M) = 1$
4	$P_M = P_U$	$P_M \neq P_U$

take valid plaintext to be the unrestrictive alternative (within the Markov assumption) to uniform noise.

We must also specify what is invalid plaintext—the alternative to valid plaintext. To this end, let U be a Markov chain that generates uniform noise over $\mathcal{A}$; that is, U is a 0th-order Markov chain on alphabet $\mathcal{A}$ with uniform transition probabilities $P_U = [1/m]$. In Problems 2 and 4, chain U sharply defines the language of invalid plaintext. By contrast, in Problem 3, we assume only that invalid plaintext comes from some unknown, but not necessarily uniform, 0th-order chain. And in Problem 1, we take invalid plaintext to be the unrestrictive alternative (within the Markov assumption) to the known language B.

Implicit in each problem is the assumption that the candidate plaintext was produced by some Markov chain M with alphabet $\mathcal{A}$. Knowing the alphabet $\mathcal{A}$ is a mild assumption since this alphabet can be observed from the base and on-line samples. But assuming that the candidate plaintext came from some Markov chain is significant, since real language is not Markov and can only be approximated by a Markov language.

Since we nominally view each chain as a 1st-order chain, a clarification is needed for Problem 3. By testing if $\text{order}(M) = 0$, we test if the chain is also a 0th-order chain. Such a test is known as a test for *independence*.

Table 1 summarizes our four problems. Problem 1 differs from Problem 2 in that Problem 1 takes H_1 to be the unrestrictive alternative hypothesis $H_1 = \bar{H}_0$, whereas Problem 2 takes H_1 to be the simple, sharp alternative hypothesis that X was produced from the uniform noise generator U. Although it is not always possible to do so, it is preferable to formulate applications with Problem 2 rather than with Problem 1 because Problem 2 has an optimal test. Like Problem 1, Problem 4 has an unrestrictive alternative hypothesis $H_1 = \bar{H}_0$. Moreover, chain U can be viewed as a special case of the known chain B. But here the roles of valid plaintext and invalid plaintext are reversed. In this sense we view Problem 4 as the dual of Problem 1.

Discussion

Whenever possible the cryptanalyst should define her hypotheses as sharply as possible. Doing so makes maximum use of available information and facilitates the selection of the most appropriate test statistic. In particular, it is difficult to say much about the relative performance of test statistics on any problem any of whose hypotheses is broadly defined. Therefore, the cryptanalyst should try to avoid Problems 1 and 4, which have unrestrictive alternatives.

Zeroes in the matrices of transition probabilities require special attention. For example, consider Problem 1 and suppose the matrix P_B of known transition probabilities contains exactly d zeroes. For this situation, we assume that the parameter space Ω_{H_1} for the alternative hypothesis H_1 consists of all $m \times m$ matrices over $[0, 1]$, different from P_B, that contain exactly d zeroes and whose zeroes appear in the same positions as do the zeroes in P_B. Similarly, for the other problems, we assume that the zeroes in the matrices of transition probabilities in the parameter spaces Ω_{H_0} and Ω_{H_1} match. For more about zero transition probabilities, their difficulties, and how to cope with them, see Section 7.1.

Additional problems can be similarly defined, such as deciding if X was produced by a *stationary* Markov chain, determining the order of the chain that produced X, or classifying X among two or more known languages. For details about these addition problems, see T. Anderson and Goodman [4], Bhat [9], or Billingsly [11].

5.2 Examples

We present four examples to illustrate how the foregoing language-recognition problems arise in cryptanalytic practice. For each situation, the cryptanalyst selects one of the problems from the details of the application, defining her hypotheses as sharply as possible. In doing so, it is helpful to answer the following questions: What are the possible plaintext languages? What are their parameters? What are the alternatives to valid plaintext? For example, if the application is a key search, does the cryptanalyst know the statistical properties of strings produced by decrypting ciphertext under incorrect keys?

Example 1: Finding a C Program among Unknown Cleartext Files

The analyst seeks an important cleartext file written in the C programming language in a directory of unencrypted files of various unknown types. Problem 1 applies. Because of the predictable vocabulary of programming languages, a dictionary attack would also work especially well here.

Example 2: Key Search of DES with French Plaintext

This situation is typical for plaintext recognition. The cryptanalyst knows the plaintext language and the cipher. The plaintext language can be reasonably modeled with a Markov chain, and the DES cipher appears to produce random strings when applied to decrypt any ciphertext under incorrect keys. Problem 2 nicely models this situation.

Example 3: Key Search of a Beale Cipher with Unknown Plaintext

Baldwin and Sherman [6] encountered this problem in their solution of the Decipher Puzzle. In the Beale cipher (see Kahn [52, pp. 771–772]), each plaintext character is encrypted as any integer from a set of codenumbers corresponding to that character. Each codenumber is a position index of the plaintext character from an unknown keytext. For example, the keytext "BEETHOVEN" yields the codenumber set $\{2, 3, 8\}$ for the letter 'E' since 'E' appears in positions 2, 3, and 8 of the keytext. Neither the plaintext language nor the keytext language is known to the cryptanalyst.

Decryption under an incorrect key produces 0th-order noise from the keytext language, since such decryption has the effect of randomly indexing into the keytext. Typically, this noise is not

uniform since it usually follows the unigram frequencies of the keytext language. By contrast, the correct key produces a candidate plaintext from the plaintext language, which we assume has order greater than zero and can be modeled as a 1st-order language. Therefore, Problem 3 applies.

Problem 3 is also relevant when matching columns in solving transposition ciphers.

Example 4: Key Search of REDOC II with Unknown Plaintext

The REDOC II cryptosystem appears to produce random-looking strings when used to decrypt any ciphertext under incorrect keys. Therefore this situation is similar to Example 2, except the plaintext language is unknown. Hence Problem 4 could be used.

It would be preferable, however, if the cryptanalyst could find a way to use a multi-language variation of Problem 2 and thereby use a sharply defined alternative hypothesis. For example, suppose the cryptanalyst could make a list of possible plaintext languages that is certain to include the actual plaintext language. Then, the cryptanalyst could view her task as a classification problem among these languages and uniform noise.

6 Testing a Statistical Hypothesis

In each of the four plaintext-recognition problems we test a null hypothesis H_0 versus an alternative hypothesis H_1. We do so by computing some test statistic ψ, which depends on the candidate plaintext.[14] To interpret the value of this statistic, we define a *critical region*—a subset of image(ψ) for which we reject H_0. This testing process yields an effective way to solve our problems if the distribution of ψ given H_0 differs sufficiently from the the distribution of ψ given H_1. In this section we review the essentials of statistical hypothesis testing.

The rest of this section is organized as follows. Section 6.1 reviews some fundamental concepts and terminology from statistical inference. Section 6.2 explains how to select the critical region. And Section 6.3 discusses criteria for evaluating test statistics, including the notion of a uniformly most powerful test. The reader familiar with statistical inference may wish to skip to Section 7.[15]

6.1 Fundamental Concepts and Terminology

In testing a statistical hypothesis, the following three concepts play a crucial role: one-sided versus two-sided tests, simple hypothesis versus compound hypothesis, type-I error versus type-II error.

One-Sided versus Two-Sided Tests

In a *one-sided test*, the critical region is a single open or closed interval: for example, reject H_0 if and only if $\psi > \tau$, for some threshold τ. In a *two-sided test*, the critical region is defined by two such intervals: for example, reject H_0 if and only if $\psi < \tau_1$ or $\tau_2 < \psi$, for some thresholds $\tau_1 \leq \tau_2$. Whether it is possible or more appropriate to use a one-sided or two-sided test depends on the statistic and alternative hypothesis.

[14] The term *test statistic* refers to the random variable ψ; the term *test* refers the entire testing process.

[15] For a more detailed explanation of statistical hypothesis testing, see Lehmann [64], Rohatgi [77], and Wilks [93].

Simple versus Compound Hypotheses

We assume the distribution of ψ belongs to some known class of parameterized distributions $\mathcal{P} = \{P_\theta : \theta \in \Omega\}$, for some set of parameters Ω. For example, in Problem 1, Ω is the set of all possible transition probabilities for a Markov chain with the specified state space and order. Each hypothesis corresponds to some subset of Ω, and $\Omega = \Omega_{H_0} \cup \Omega_{H_1}$, where Ω_{H_0} and Ω_{H_1} are subsets of parameters corresponding to hypotheses H_0 and H_1, respectively.

A hypothesis is *simple* if it corresponds to a singleton subset of Ω; otherwise, it is *compound*. For example, in Problem 1, H_0 is simple and H_1 is compound.

Type I versus Type II Errors

For any critical region $\mathcal{C}$, two types of errors can arise. A *type-I error* occurs when the test rejects H_0 even though H_0 is true. A *type-II error* occurs when the test accepts H_0 when H_1 is true. When H_0 is simple, let $e_1 = \text{Prob}[\psi \in \mathcal{C} | H_0]$ denote the probability of a type-I error, with respect to $\mathcal{C}$. Similarly, when H_1 is simple, let $e_2 = \text{Prob}[\psi \notin \mathcal{C} | H_1]$ denote the probability of a type-II error. Thus for Problem 2, e_1 is the probability of rejecting valid plaintext, and e_2 is the probability of accepting uniform noise as plaintext.

When H_0 is compound, we view $e_1 : \Omega_{H_0} \rightarrow [0, 1]$ as a function of the parameters $\theta \in \Omega_{H_0}$. Specifically, for any $\theta \in \Omega_{H_0}$, $e_1(\theta) = \text{Prob}[\psi \in \mathcal{C} | H_0(\theta)]$, where $H_0(\theta)$ is the restriction of H_0 to parameter θ. Similarly, when H_1 is compound, we view $e_2 : \Omega_{H_1} \rightarrow [0, 1]$ as a function of the parameters $\theta \in \Omega_{H_1}$.

6.2 Selecting the Critical Region

The most appropriate method for selecting the critical region depends in part on the problem and on the application. For Problems 1 and 2, in which H_0 signifies valid plaintext, the cryptanalyst would typically carry out the following standard three-step procedure:

1. Decide whether to use a one-sided or two-sided test.
2. Choose a *critical level* $\alpha \in [0, 1]$, which specifies the maximum tolerable type-I error rate.
3. From this critical level and from the distribution of $\psi | H_0$, compute threshold(s) for ψ that satisfy $e_1 \leq \alpha$.

In this procedure, an important goal is to minimize e_2 subject to $e_1 \leq \alpha$. For Problems 1 and 2, we set the type-I error rate first, because in most applications it would be very unfortunate to overlook valid plaintext.

For Problems 3 and 4, in which H_1 signifies valid plaintext, the cryptanalyst might prefer to carry out a variation of this procedure in which she first sets the type-II error rate. In this variation, the cryptanalyst would choose threshold(s) for ψ, minimizing e_1 subject to $e_2 \leq \alpha$. For Problems 3 and 4, however, this variation presents the additional difficulty that H_1 is composite.

It is important to be aware that the *actual distribution* of ψ might differ from its *theoretical distribution*, since in practice the assumptions of the underlying model are usually not satisfied completely. For example, American English is not an ergodic 1st-order finite stationary Markov language. Similarly, the asymptotic distribution of ψ might differ from the distribution for the

finite-size messages that arise in practice. Consequently, the cryptanalyst should experiment to verify the actual error rates or to determine the thresholds empirically.

For applications where on-line decisions are not required, it may be helpful also to rank all candidate plaintexts by their ψ values, as suggested by Sinkov [82].

6.3 Criteria for Evaluating Test Statistics

There are three main criteria by which we evaluate test statistics: time and space complexity, quality of results, and implementation difficulty. Traditionally, statisticians have focused on quality of results, and more narrowly, on statistical power and robustness. In this section, we briefly explain the concepts of statistical power and robustness and the notion of a uniformly most powerful test. The question of how time and space relate to quality of statistical tests remains a fascinating active area of research today (see Section 8.3).

Statistical Power and Strength

When testing the null hypothesis H_0 versus the simple alternative H_1 at critical level α, the *power of ψ at level α* is $\beta = \text{Prob}[\psi \in \mathcal{C}|H_1]$, where $\mathcal{C}$ is the critical region. That is, $\beta = 1 - e_2$, where e_2 is the type-II error rate at critical level α. When H_1 is compound, we view $\beta : \Omega_{H_1} \rightarrow [0, 1]$ as a function of the parameters $\theta \in \Omega_{H_1}$.

In their experimental studies, Good and Crook [19, 44] define and apply a more general notion called strength, where the *strength* of a test statistic is a weighted average of its power, averaged over all parameters $\theta \in \Omega_{H_1}$, where the weights are given by a prior distribution of the parameters.

Most Powerful and Uniformly Most Powerful Tests

With respect to statistical power, there is a standard notion of an optimal test. When H_1 is simple, we say that a test is *most powerful at level α* if and only if its power at level α is at least as great as that for all other tests. When H_1 is compound, we say that a test is *uniformly most powerful* at level α if and only if it is a most powerful test at level α for all parameters $\theta \in \Omega_{H_1}$.

When both H_0 and H_1 are simple, for any critical level α, it is possible to construct a most powerful test at level α following the criteria of the Neyman-Pearson Lemma [64, p. 74]. In Section 7.2, we define such an optimal test for Problem 2. For compound H_1, however, typically no uniformly most powerful tests exist. When a uniformly most powerful test cannot be constructed, a common practice is to define likelihood ratio tests following the maximum-likelihood principle applied to the Neyman-Pearson criteria. In Section 7.3, we define such likelihood ratio tests for Problems 1, 3, and 4.

Robustness

Informally, a statistic is *robust* if its actual behavior does not deviate much from its theoretical behavior when the assumptions of the model are not fully satisfied. Thus, a robust test for recognizing Markov English should still work well when applied to real English. It is possible to construct formal models of robustness classes—for example, see Huber [50] and Poor [71].

7 Test Statistics and Decision Procedures: Solutions to Problems 1–4

We now present solutions to each of the four language-recognition problems defined in Section 5. For Problem 2, which has a sharply defined alternative hypothesis, we give a most powerful test based on the Neyman-Pearson Lemma [64]. For Problems 1, 3, and 4, which typically do not have uniformly most powerful tests, we give likelihood ratio tests derived from the maximum-likelihood principle applied to the Neyman-Pearson Lemma. We also discuss the X^2 and IC statistics. Furthermore, we point out the asymptotic distributions of these test statistics, and we explain two useful decision procedures based on approximate χ^2 and Gaussian interpretations.

7.1 Likelihood Ratio Tests

Likelihood ratio tests are natural, powerful statistical tests based on the log-likelihood ratio. Before deriving particular likelihood ratio test statistics, we briefly define our notation and review the log-likelihood ratio and its asymptotic distribution. For convenience, we define each test statistic with respect to a 1st-order (bigram) model of language; the generalization to higher-order models is straightforward.

Notation

We define each statistic in terms of the following convenient notation, which describes the observed frequency counts in the candidate plaintext and the constant parameters of the Markov model of the base language. The candidate plaintext is a sequence of n letters from an alphabet $\{1, \ldots, m\}$ of size m. This sequence forms $N = n-1$ overlapping bigrams. For each $1 \le i, j \le m$, let n_i denote the observed frequency of letter i in the candidate plaintext, and let n_{ij} denote the observed frequency of bigram ij. There are $n_{i*} = \sum_{j=1}^{m} n_{ij}$ bigrams beginning with the letter i, and $n_{*j} = \sum_{i=1}^{m} n_{ij}$ bigrams ending with the letter j.[16] The constants p_{ij} and p_i denote, respectively, the transition probability and state probability (*i.e.* unigram probability) from the Markov model of the base language.

From the frequency counts, we compute the *observed relative unigram frequencies* $\hat{p}_i = n_i/n$ and the *observed relative transition frequencies* $\hat{p}_{ij} = n_{ij}/n_{i*}$. Also, let $b_{ij} = p_i p_{ij}$ denote the corresponding *unconditional bigram probability* in the Markov model of the base language.

Finally, let $P_B = [p_{ij}]_{1 \le i,j \le m}$ denote the $m \times m$ matrix of transition probabilities from the Markov model B of the base language, and let $\hat{P}_X = [\hat{p}_{ij}]_{1 \le i,j \le m}$ denote the $m \times m$ matrix of observed relative transition frequencies from the candidate plaintext X. Similarly, let $P_U = [1/m]$ be the $m \times m$ matrix of uniform transition probabilities, and let $\hat{P}_0 = [\hat{p}_j]_{1 \le i,j \le m}$ be the $m \times m$ matrix of observed relative unigram frequencies from X.

The Log-Likelihood Ratio

Given any two competing hypotheses H_0 and H_1, we construct the *log-likelihood ratio*

[16] If the last plaintext letter is an i, then $n_{i*} = n_i - 1$; otherwise, $n_{i*} = n_i$. Similarly, if the first plaintext letter is a j, then $n_{*j} = n_j - 1$; otherwise, $n_{*j} = n_j$.

$$\Lambda(H_0/H_1 : X) = \ln \frac{\max_{\theta_0 \in \Omega_{H_0}} \text{Prob}[X|H_0(\theta_0)]}{\max_{\theta_1 \in \Omega_{H_1}} \text{Prob}[X|H_1(\theta_1)]}, \tag{2}$$

where X is the candidate plaintext, and $H_0(\theta_0)$ and $H_1(\theta_1)$ are, respectively, the restrictions of H_0 and H_1 to parameters θ_0 and θ_1. This ratio is sometimes called the *weight of evidence in favor of H_0 as opposed to H_1, provided by X*; for example, see Good [32, 38, 70].

For Problem 2, whose hypotheses are simple, this construction yields a most powerful test by the Neyman-Pearson Lemma. For Problems 1, 3, and 4, whose alternative hypotheses are compound, we compute the log-likelihood ratio, replacing the unknown parameters P_M with their maximum-likelihood estimates $\hat{P}_X$ computed from X. At least for Problems 1 and 4, each of which has one simple hypothesis, this construction is equivalent to the common practice of computing the log-likelihood ratio replacing the unknown parameters with those obtained by maximizing (for Problem 4, minimizing) the log-likelihood ratio. The resulting tests are called *likelihood ratio tests.*

To derive particular test statistics, it is necessary to compute the *likelihood function* $L_H(X) = \text{Prob}[X|H]$, or its natural logarithm, for various hypotheses H; this function computes the *a priori* probability that a candidate plaintext X yields the observed frequency counts $\{n_{ij}\}$ under the specified hypothesis H. For example, in Problem 1, H_0 is the hypothesis that the candidate plaintext was produced by a 1st-order Markov chain with known transition probabilities $P_M = P_B$. In this case, we write $L(P_B) = L_{H_0}(X)$ since it is traditional to view L as a function of the language parameters rather than of the plaintext. Ignoring initial conditions of the Markov chain, for Problem 1 we have

$$L(P_B) = \text{Prob}[X|P_M = P_B] = \prod_{i=1}^{m} p_{i1}^{n_{i1}} p_{i2}^{n_{i2}} \cdots p_{im}^{n_{im}}, \tag{3}$$

and therefore

$$\ln L(P_B) = \sum_{1 \le i,j \le m} n_{ij} \ln p_{ij}. \tag{4}$$

Although Bhat [9, p. 138] states that the frequency counts "can be considered as a sample ... from a multinomial distribution," due to linear dependencies among these counts (*e.g.* $|n_{i*} - n_{*i}| \le 1$), the frequency counts are not exactly so distributed.

Asymptotic Distribution of the Log-Likelihood Ratio

Under suitable regularity conditions, such as those described by Wilks [93], it is known that under H_0 the test statistic -2Λ has an asymptotically χ^2 distribution. This asymptotic distribution follows from a Taylor series expansion of the likelihood function around the known parameters of Ω_{H_0} (for compound H_0, around their maximum-likelihood estimates). Intuitively, the regularity conditions ensure in part, that under H_0, this Taylor series expansion exists in a suitable neighborhood within the parameter space Ω. The mysterious -2 factor stems from a constant factor in this expansion. In Section 7.5 we explain how to carry out this approximate χ^2 interpretation in practice, and we discuss two additional options for interpreting Λ.

In the asymptotically χ^2 distribution of -2Λ, the number of degrees of freedom is equal to the number of unknown parameters (*i.e.* transition probabilities) minus the number of constrained

parameters. These parameters can be constrained, for example, from the definition of a Markov chain (*e.g.* it must be true for each $1 \leq i \leq m$ that $\sum_{j=1}^{m} p_{ij} = 1$) and from zeroes in the matrix of known transition probabilities. As explained in Section 5.1, we assume that the zeroes in the matrices of transition probabilities in the parameter spaces Ω_{H_0} and Ω_{H_1} match.

Zero transition probabilities can cause theoretical and practical difficulties. The theoretical difficulty is that, unless the zeroes in the matrices of transition probabilities in the parameter spaces Ω_{H_0} and Ω_{H_1} match, such zeroes might violate the regularity conditions. The practical difficulty is that the cryptanalyst must deal with *impossible bigrams*, such as "QZ", that appear in candidate plaintext despite having zero transition probabilities in the theoretical model. How best to handle such impossible bigrams in practice depends on the application. One simple strategy, which avoids the theoretical difficulties, is to replace each zero transition probability with a small number, as did Baldwin and Sherman [6].

7.2 A Most Powerful Test for Problem 2

For Problem 2 (distinguishing a known language from uniform noise), Equations 2 and 4 yield the test statistic

$$\Lambda_2 = \ln \frac{L(P_B)}{L(P_U)} = \left(\sum_{1 \leq i,j \leq m} n_{ij} \ln p_{ij} \right) - \sum_{1 \leq i,j \leq m} n_{ij} \ln \frac{1}{m} = \left(\sum_{1 \leq i,j \leq m} n_{ij} \ln p_{ij} \right) + N \ln m. \qquad (5)$$

In Equation 5, and analogously for all summations in this section, we assume the sum is computed over only those indices i, j such that $p_{ij} \neq 0$. For this problem the parameter space $\Omega = \{P_B, P_U\}$ consists of exactly two points; consequently, the regularity conditions for $-2\Lambda_2$ to have an asymptotically χ^2 distribution do not make sense. Therefore, we suggest applying decision procedures based on a Gaussian interpretation of Λ_2/N, as explained in Section 7.5.

By the Neyman-Pearson Lemma and by the ergodicity of P_B, Λ_2 is an asymptotically most powerful test for Problem 2. Moreover, with initial conditions included, Λ_2 would become a most powerful test for Problem 2 (without the asymptotic qualification).

Equation 5 explains the Sinkov statistic $S_2(X) = \sum_{i=1}^{n} \ln p_{x_{ij}|x_i}$ defined in Section 2: By rearranging Sinkov's summation over all possible bigrams in the base language rather than over the observed bigrams in the candidate plaintext, we see that S_2 is simply the log-likelihood function

$$S_2 = \sum_{1 \leq i,j \leq m} n_{ij} \ln p_{ij}. \qquad (6)$$

Moreover, since the linear summand $N \ln m$ in Equation 5 does not affect the essential character of Λ_2 (S_2 is a monotonically decreasing function of Λ_2), the statistics Λ_2 and S_2 are equivalent.[17] We note further that the Sinkov statistic is closely related to concepts in information theory. For example, as observed by Bartlett [7, p. 88], when applied to unigrams, $\lim_{n \to \infty} S_1/n = (-1/\log_2 e)\, H(X)$, where $H(X)$ is the entropy of X.[18]

A similar construction also yields a most powerful test for classifying among two or more known languages.

[17] For a formal definition of when two statistics are *equivalent*, see Lehmann [64, p. 43].

[18] *Entropy* is a measure of uncertainty, expressed in bits—for details see Gallager [28]. Shannon [80] measured the entropy of printed English experimentally and found that it is approximately 1.0 bit/character.

7.3 Likelihood Ratio Tests for Problems 1, 3, and 4

For Problems 1, 3, and 4, we define likelihood ratio tests from Equations 2 and 4 by replacing the unknown parameters P_M with their maximum-likelihood estimates computed from the candidate plaintext.

A Likelihood Ratio Test for Problem 1

For Problem 1 (recognizing a known language), the maximum-likelihood criterion applied to Equations 2 and 4 yields the log-likelihood ratio

$$\Lambda_1 = \ln \frac{L(P_B)}{L(\hat{P}_X)} = \left(\sum_{1 \le i,j \le m} n_{ij} \ln p_{ij} \right) - \sum_{1 \le i,j \le m} n_{ij} \ln \hat{p}_{ij} = \sum_{1 \le i,j \le m} n_{ij} \ln \frac{p_{ij}}{\hat{p}_{ij}}, \tag{7}$$

where P_B is the matrix of known transition probabilities from the base language and $\hat{P}_X$ is the matrix of observed relative transition frequencies from the candidate plaintext X. Note that $\hat{P}_X$ is also the maximum-likelihood estimate computed from X of the parameters P_M. Since hypothesis H_1 is compound, in computing the log-likelihood ratio Λ_1 we replace the unknown parameters P_M corresponding to H_1 with $\hat{P}_X$.

From Equation 7, we define the related test statistic

$$ML = -2\Lambda_1 = 2 \sum_{1 \le i,j \le m} n_{ij} \ln \frac{\hat{p}_{ij}}{p_{ij}}. \tag{8}$$

Under hypothesis $H_0 : P_M = P_B$, the statistic ML has an asymptotically χ^2 distribution with $m(m-1) - d$ degrees of freedom, where d is the number of zero transition probabilities in the base language. To compute the degrees of freedom, note that there are $m^2 - d$ nonzero transition probabilities in P_B, among which there are m constraints that force each row sum of P_B to be 1. For more details, see Anderson and Goodman [4].

A Likelihood Ratio Test for Problem 3

For Problem 3 (distinguishing 0th-order noise from unknown 1st-order language), we obtain the log-likelihood ratio

$$\Lambda_3 = \ln \frac{L(\hat{P}_0)}{L(\hat{P}_X)} = \left(\sum_{1 \le i,j \le m} n_{ij} \hat{p}_j \right) - \sum_{1 \le i,j \le m} n_{ij} \hat{p}_{ij} = \sum_{1 \le i,j \le m} n_{ij} \ln \frac{\hat{p}_j}{\hat{p}_{ij}}, \tag{9}$$

where $\hat{P}_0$ is the matrix of observed relative unigram frequencies from the candidate plaintext. It is customary to use the related test statistic

$$IND = -2\Lambda_3 = 2 \sum_{1 \le i,j \le m} n_{ij} \ln \frac{\hat{p}_{ij}}{\hat{p}_j}, \tag{10}$$

which under hypothesis $H_0 : P_M = \hat{P}_0$, has an asymptotically χ^2 distribution with $(m-1)^2 - d$ degrees of freedom. This distribution has $m-1$ additional constraints beyond those for Problem 1.

A Likelihood Ratio Test for Problem 4

Similarly for Problem 4 (detecting non-uniform unknown language), we have the log-likelihood ratio

$$\Lambda_4 = \ln \frac{L(P_U)}{L(\hat{P}_X)} = \left(\sum_{1 \le i,j \le m} n_{ij} \ln(1/m) \right) - \sum_{1 \le i,j \le m} n_{ij} \ln \hat{p}_{ij} = -N \lg m - \sum_{1 \le i,j \le m} n_{ij} \ln \hat{p}_{ij}. \quad (11)$$

Under hypothesis $H_0 : P_M = P_U$, the related statistic $-2\Lambda_4$ has an asymptotically χ^2 distribution with $m(m-1)$ degrees of freedom. For the degrees of freedom, the situation is similar to that in Problem 1, except P_U has no zeroes.

7.4 Additional Test Statistics

The chi-squared test statistic X^2, and the Index of Coincidence IC introduced in Section 2, are useful for a variety of inference problems including Problems 1 and 4. We now explain these test statistics and point out a few selected other tests.

Chi-Squared Test Statistic

The well-known chi-squared test statistic

$$X^2 = \sum_{1 \le i,j \le m} \frac{(n_i \hat{p}_{ij} - n_i p_{ij})^2}{n_i p_{ij}} \quad (12)$$

is asymptotically equivalent to ML, as can be proven by a Taylor series expansion of the likelihood function L around the known parameters P_B. Thus, under hypothesis $H_0 : P_M = P_B$, the test statistic X^2 has an asymptotically χ^2 distribution with $m(m-1)-d$ degrees of freedom.[19] As mentioned in Section 4, certain conditions must be satisfied for the χ^2 interpretation of the X^2 statistic to be accurate. For example, some statisticians require that, for each $1 \le i,j \le m$, the expected number of observed bigrams ij be at least 5. For a discussion of such conditions, see Good, Gover, and Mitchell [45, p. 268].

Several embellishments of X^2 are possible. For example in Equation 12, the squaring operation loses information by treating observed positive and negative deviations from the expected value $n_i p_{ij}$ equivalently. For some applications, slightly better results can be obtained by using a variation of X^2, as derived by Lehmann [64], that takes advantage of the signs of these deviations. For independence problems related to Problem 3, Diaconis and Efron [21] propose a new interpretation of the X^2 statistic. And for Problem 4, Good [40], Good, Gover, and Mitchell [45], and Good and Crook [44] recommend using Cochran's continuity-adjusted X'^2 statistic in place of X^2.

The Index of Coincidence

Applied to bigrams, the Index of Coincidence is defined by

[19] We distinguish between the test statistic X^2 and the related random variable χ^2.

$$IC = \sum_{1 \le i,j \le m} \frac{n_{ij}(n_{ij}-1)}{N(N-1)}. \tag{13}$$

The IC is similar to X^2 in that both statistics are quadratic functions of the frequency counts. Good, Gover, and Mitchell [45] claim that the IC is especially useful when the conditions for using X^2 are not satisfied.

As for its distribution, Kullback [60, pp. 151–153] gives a lengthy formula for a related statistic from which the variance of IC can be calculated, and Good [42] describes how to compute this variance exactly using computer techniques. In addition, Good, Gover, and Mitchell [45] and Good and Crook [44] present experimental evidence that, under hypothesis $H_0 : P_M = P_U$ for Problem 4, when $N \ll t/12$, the statistic $N(N-1)IC/2$ has an approximately Poisson distribution with mean $N(N-1)/(2t)$, for $t = m^2$. For more about the IC, including an experimental evaluation of its application to Problem 4, see Good [36, 40, 42].

Selected other Tests

Other relevant statistics are also suggested in the statistics and computer science literature. For example, Good [40] introduces a non-Bayesian statistic G, based on his Bayes/Non-Bayes Compromise [43, 44]. Good claims this statistic is sometimes useful for Problem 4 for small samples when the X^2 statistic does not apply. Good and Crook [18, 19, 41] analyze the G statistic and compare it with the X^2 and log-likelihood statistics and with a Bayes factor F.

In his study of pseudorandom number generators, Knuth [57] describes many tests for non-randomness, including his so-called spectral test which interprets frequency counts in a geometric fashion. Knuth [57, p. 89] asserts, that for linear congruential generators, his test "is by far the most powerful test known."

Using ideas from digital signal processing, Feldman [23] proposes a new spectral test for non-randomness that interprets the power spectrum of finite strings. To evaluate his statistic, Feldman applies it to strings produced by short-round versions of the DES cryptosystem.

In addition, Dickey and Lientz [22] propose a Bayesian test for Markov order based on a weighted likelihood ratio.

7.5 Decision Procedures

To interpret values of these test statistics, the cryptanalyst must adopt some decision procedure. One option is to approximate the test statistic with an appropriate χ^2 distribution. Since many log-likelihood statistics have an asymptotically χ^2 distribution under the null hypothesis, using a χ^2 approximation is an attractive theoretical and practical option when the asymptotic theory applies. Good [33, p. 40] admonishes, however, that such χ^2 approximations "are at best asymptotic, and are usually inapplicable." A second option is to center and normalize the test statistic using a construction from the central limit theorem. This method yields a convenient practical decision procedure, which for some cases is theoretically justified. These two options enable the cryptanalyst to use standard tables in selecting thresholds for the critical regions. A third option is to compute critical regions based on experimentally-determined exact or approximate distributions. This option requires a precomputation, but often produces the best practical results. In the rest of this section, we explain how to carry out decision procedures based on the first two options.

Normalizing a Chi-Squared Statistic

Let ψ be any test statistic that has an approximately χ^2 distribution with ν degrees of freedom. For small values of ν, the cryptanalyst can interpret ψ using standard tables of the χ^2 distribution. For large values of ν, another method is usually needed since most tables do not give critical values for large values of ν. When $\nu > 100$, the cryptanalyst can center and normalize ψ by computing

$$\diamond\psi = \sqrt{2\psi} - \sqrt{2\nu - 1}, \tag{14}$$

as recommended by Trivedi [90, p. 593]. The resulting random variable $\diamond\psi$ can be interpreted as approximately standard normal.

For example, in our experiments [29] in which we interpret the *ML*, *IND*, and X^2 statistics using this normalization technique, ν is typically approximately 500. For such large degrees of freedom, it may be helpful to condense states, as described in Section 4.

Standard Normalization

Let Y be any test statistic that can be viewed as the sum of N identical random variables applied to N bigrams. For example, the S_2 statistic can be so viewed by organizing its sum over the candidate plaintext—as we did in Section 2—rather than over all possible bigrams of the base language. A simple and effective way to interpret Y is to center and normalize Y/N and to view the resulting random variable $\hat{Y}$ as approximately standard normal.

For example, Baldwin and Sherman [6] computed

$$\hat{S}_2 = \frac{(S_2/N) - \mu_S}{\sigma_S/\sqrt{N}}, \tag{15}$$

where μ_S and σ_S are, respectively, the mean and standard deviation of S_2 applied to a randomly chosen bigram, with the bigram chosen according to the unconditional bigram probabilities of the base language. Specifically, Baldwin and Sherman computed constants μ_S and $\sigma_S = \sqrt{\sigma_S{}^2}$ from the definitions of mean and variance as follows:

$$\mu_S = \sum_{1 \le i,j \le m} b_{ij} \ln p_{ij} \tag{16}$$

and

$$\sigma_S^2 = \sum_{1 \le i,j \le m} b_{ij} \left((\ln p_{ij}) - \mu_S \right)^2, \tag{17}$$

where for all $1 \le i, j \le m$, p_{ij} and b_{ij} are, respectively, the conditional and unconditional bigram probabilities of the bigram ij in the base language.

When the S_2 statistic is applied to independent bigrams from the base language, the central limit theorem [64] guarantees that the random variable $\hat{S}_2$ is asymptotically standard normal. However, when the S_2 statistic is applied to dependent bigrams, simple versions of the central limit theorem do not apply because the underlying random variable summands are not independent. In this dependent-bigram case, the more general central limit theorem for dependent random variables ensures only that $\hat{S}_2$ is asymptotically Gaussian with mean 0. For this case, the variance of S_2/N can be exactly computed from the covariance matrix of the frequency counts, which matrix is given

by Good [31]. Alternatively, the cryptanalyst can approximate this variance experimentally. In practice, as confirmed by our experiments [29], Equation 17 yields a good approximation of the variance of $\sqrt{N}((S_2/N) - \mu_S)$ for standard English.

Especially when the number of degrees of freedom is large, there may also be some practical value in devising similar Gaussian interpretations for X^2/N and for other test statistics.

8 Discussion

In Section 7 we presented solutions to our four language-recognition problems, assuming the input is a single unadulterated string of sufficient length. But in practice, a cryptanalyst must sometimes deal with candidate plaintext that is very short, is contaminated with noise, and consists of two or more separate strings. In this section, we briefly discuss techniques for dealing with these practical complications. In addition, we point out useful references in the statistics and pattern-recognition literature for further reading, and we raise some open problems.

8.1 Practical Variations

To cope with the aforementioned three practical complications, one could simply apply the standard methods described in Section 7, or one could choose special-purpose methods. This section describes variations of the standard methods for coping with these complications. First, we outline how to use Hidden Markov Models to deal with noisy plaintext. Second, we show how to exploit initial conditions to obtain better results for short candidate plaintexts. Third, we note that the problem of dealing with multiple input strings can be reduced to the problem of dealing with one longer input string. For an experimental evaluation of how well the standard methods work for short and noisy plaintext, see our companion paper [29].

Noisy Plaintext

In practical cryptanalysis, candidate plaintext is often contaminated with noise that injects, deletes, or modifies characters. Such noise can result, for example, from incomplete knowledge of the key, transmission errors, or errors in the original plaintext. Moreover, the sender might maliciously add noise to the plaintext. It is desirable for language-recognition techniques to be robust in the presence of such noise.

The problem of recognizing plaintext in the presence of noise is a discrete version of the signal-detection problem, which has been extensively studied. For example, van Trees [88, 89] gives a thorough engineering treatment of this problem, and Osteyee and Good [70] discuss the problem from the point of view of information theory and the weight of evidence. In the rest of this section, we briefly explain a general model of language for dealing with noisy plaintext and point out a few helpful references for learning about this model.

The *hidden Markov model (HMM)* is an especially useful model of noisy language. A HMM is a variation of the Markov model in which, at each state, the model randomly outputs any letter of the alphabet according to a specified probability distribution. By contrast, in a Markov chain, for each state the chain always outputs the single fixed letter corresponding to that state. For example, consider a 0th-order HMM of English for which 10% of the outputs are randomized by uniform noise. When this model is in the state corresponding to the letter 'Z', with probability 0.9

it outputs 'Z', and with probability 0.1 it ouputs a randomly chosen letter of the alphabet. Trivially, each Markov model is also a degenerate HMM.

Rabiner [72] reviews the hidden Markov model and its applications to speech recognition, including an explanation of the forward-backward procedure for computing the likelihood of an observed plaintext given a particular HMM. Juang and Rabiner [51] develop a distance measure for this model. As a recent practical example, Krishnamurthy, Moore, and Chung [58], who to enhance the performance of a biomedical instrument, develop maximum-likelihood techniques for interpreting Markov signals that are simultaneously contaminated with white Gaussian noise and a deterministic signal. In addition to modeling noisy plaintext, HMMs may also be helpful simply as a more general model of language.

Short Strings

Cryptanalysts must deal with short strings. Short strings can occur from short messages or from incomplete cryptanalysis, as can result from incomplete knowledge of the key or ciphertext. For example, in their solution of the Decipher Puzzle, Baldwin and Sherman [6] produced for each candidate key fragment a set of approximately ten independent bigrams. Moreover, when using language-recognition techniques as subroutines within more elaborate attacks against strong ciphers, often it is not possible to generate long substrings.

Short plaintext presents two problems. First, less information is available; for example, repeated grams are unlikely in very short messages. Second, the asymptotic distributions of the test statistics might not apply. On the other hand, for short strings it may be feasible to determine the distributions of test statistics exactly through computer simulations.

To make better use of available information, for short strings it is often beneficial to include the initial conditions of the Markov chain in the test statistics. For example, the cryptanalyst could take as her initial conditions the state probabilities. Alternatively, if the cryptanalyst knows that the first character of the candidate plaintext begins a word, then the cryptanalyst could take as her initial conditions frequencies for initial letters of words. For long strings, however, there is little value in using initial conditions because, for ergodic chains, the initial conditions do not affect the asymptotic behavior of the tests. For each test statistic based on the log-likelihood ratio, initial conditions can be simply incorporated by adding the appropriate summand, as shown by Bartlett [7] for example.

Another approach for dealing with small samples is given by Yakowitz [94], who motivated by river modeling problems, proposes a new class of tests for the order of a Markov chain.

Multiple Strings

Sometimes the candidate plaintext consists of a set of separate strings rather than of one long string. For example, multiple strings can result from incomplete cryptanalysis or from several messages that are written in the same language and encrypted with the same key. When each of the component strings is very short, it may be beneficial to combine the strings.

A simple and effective approach is to combine the observed frequency counts from all strings and to apply the usual tests: each string contributes a set of overlapping grams, but the strings are not concatenated. Anderson and Goodman [4, p. 105] explain that, for stationary Markov chains, this procedure is asymptotically equivalent to observing one long string. This procedure can be further refined by separately incorporating the initial conditions for each string.

For any component string that is long enough to be tested separately, it is usually advisable to test that string separately. Doing so helps limit the type-I errors caused by irregularities in any one component string. Alternatively, the cryptanalyst could test each component string separately in addition to testing the combined string.

8.2 Further Reading

As a guide to further reading on language recognition, we point out some of the relevant research papers and survey articles from the statistics literature. We also list selected works from the pattern-recognition literature that apply statistical techniques to voice recognition and to other related problems.

Statistical Inference on Markov Chains and Contingency Tables

Traditionally, mathematicians view language recognition as a problem in statistical inference on Markov chains. From this perspective, Anderson and Goodman [4] review standard tests based on asymptotic theory; most of these tests stem from the seminal work of Hoel and Peterson [49] and Bartlett [7], as refined by Hoel [47] and Good [30].[20] For a concise summary of this review, see Bhat [9, Chapter 5]. For an extensive survey of the asymptotic theory of statistical methods on Markov chains, see Billingsley [10, 11].

Another similar and more general view is to cast the problem as an inference problem on contingency tables. A *contingency table* is simply a table of data (*e.g.* observed k-gram frequencies) that can be analyzed under various assumptions about dependencies among the data. Agresti [1] reviews exact techniques for such problems, and Bishop, Fienberg, and Holland [13] provide a basic introduction to the related area of discrete multivariate analysis. McCloskey and Pittenger [68] give closed-form expressions for maximum-likelihood estimates that arise from testing if a multidimensional contingency table satisfies specified linear constraints that meet certain group-theoretic assumptions. In addition, from an information-theoretic perspective, Kullback, Kupperman, and Ku [61] review a variety of practical tests for inference on Markov chains and contingency tables.

A popular theme in statistical inference is to evaluate statistical tests and to compare them to other tests for solving various inference problems. For example, West and Kempthorne [92] compare the χ^2 and likelihood ratio tests in selected restrictive settings with unremarkable conclusions. To carry out such evaluations, many mathematicians resort to asymptotic theory to characterize the distributions of their statistics. To deal with non-asymptotic sizes, others perform computer simulations to compute the distributions exactly or approximately, as done for example in many papers in the *Journal of Statistical Computation and Simulation*.

C. Kelton and W. Kelton [54, 55] consider hypothesis-testing problems on Markov chains when the only observable features are unigram frequencies. For this scenario they propose and analyze several tests, including tests for 0th order, stationarity, and specific chains.

A review of prior statistics research on language recognition would be incomplete without mentioning the prolific work of I. J. Good [35, 37], who was Turing's chief statistical assistant during World War II. In his work on likelihood ratio tests [30] and on the frequency counts of Markov chains [31], Good refines some of the seminal results. Good [39] also analyzes methods for estimating language parameters, both with respect to various philosophical assumptions and with the purpose

[20] By "asymptotic theory," we mean the theory as the length of the plaintext sample tends to infinity.

of developing better practical techniques for small samples. Assuming a symmetric Dirichlet *prior* distribution on the parameters, Good [40, 41] and Crook and Good [18] extensively analyze several statistics for solving inference problems with composite alternative hypotheses. Through computer simulations, Good, Gover, and Mitchel [45]; Good and Crook [18, 19, 44]; and Good [42] compare several statistics. Based on his Bayes/Non-Bayes Compromise [43, 44], which interprets a Bayes factor as a traditional tail-area statistic, Good [40, 41, 18, 19] develops, analyzes, and advocates his G statistic. Good [34] also studies statistical models of natural language. Finally, his theory of the *weight of evidence* [32, 38] provides a practical rational approach to many cryptographic problems.

Pattern Recognition

By viewing language recognition as a form of pattern recognition, one can draw upon the many statistical techniques developed in artificial intelligence for automatic pattern recognition. Mendal and Fu [67] overview such statistical techniques. Niemann [69] also offers an introduction to this area, with an emphasis on the analysis of patterns from visual images and sound. For surveys on automatic pattern recognition and statistical techniques in pattern classification, see Kanal [53] and Ho and Agrawala [46]. Other techniques, such as neural networks, rule-based systems, and fuzzy logic, have also been tried in pattern recognition, but we shall focus here on the application of statistical techniques. In the rest of this section, we point out several instructive engineering projects that apply statistical pattern-recognition techniques.

In speech-recognition tasks, many researchers have extensively used the hidden Markov model (HMM), which we discuss in Section 8.1. For example, Tishby [87] applies a HMM to identify speakers. Ljolje and Levinson [65] and Lee and Hon [63] also apply this model to speech-recognition problems.

Using Bayesian techniques, Raviv [73] developed a program to recognize printed characters in legal text, and Valiveti and Oommen [91] present algorithms for classifying strings into known distributions. In addition, Lund and Lee [66] apply Wald's Sequential Probability Ratio Test (SPRT) to authenticate speakers, and Fukunaga and Hayes [27] study the effect of sample size on parameter estimates used in linear and quadratic classifiers.

8.3 Open Problems

Our study of language recognition raises several important questions involving: theoretical models of language, robustness of statistical tests when applied to real language, statistical properties of natural languages, the special nature of cryptanalysis, and theoretical statistics. We now discuss some of these questions.

Markov models of language are convenient and well understood, but are there better models for language recognition? For example, in cryptanalysis it might be helpful to make better use of available information by constructing a composite model that incorporates k-gram frequency counts, pattern words, word dictionaries, and other detailed language properties. Such a composite model might provide the statistical power of a high-order Markov model in a more versatile and efficient fashion. In addition, though apparently more difficult, it would be useful to take advantage of language semantics.

Although much is known about statistical techniques for language recognition in theoretical Markov models, little is known about how well such models and techniques work for real language. For $0 \leq r \leq 10$, how accurate is a r-th order model of English? What is the minimum order

required to achieve a good model of English? For each test statistic mentioned in this paper, what is its distribution when applied to real English and to other natural languages?

High-quality language processing requires detailed knowledge of the statistical properties of natural language. It would be helpful to have access to better databases of such knowledge. What types of language statistics are useful? How should they be computed? And how can they be made conveniently available?

Section 8.1 outlines some standard methods for dealing with plaintext that is short or noisy or that consists of multiple strings. But are there better ways to exploit the special nature of cryptanalysis? In addition, are there methods that exploit the fact that cryptanalytic problems typically come with a "promise" that some solution exists, and that they typically seek "a needle in the haystack" in the sense that they seek a low-probability event (*i.e.* the particular choice of a secret key) among exponentially many possibilities?

The power of "negative deductions" is well known to cryptanalysts. For example, in his Crypt Breaker's Workbench, Baldwin [5] rejects any candidate wire that implies any non-ASCII character. Yet negative deductions seem to be in conflict with robust decision making in the presence of noise. For example, it would be inflexible to reject a candidate English plaintext solely on the basis of observing the single impossible bigram "XZ". Is it possible to harness the power of negative deductions in language recognition while maintaining a sufficient level of robustness? From a statistical perspective, part of this question deals with how to treat situations in which parameters lie on the boundary of the parameter space (*e.g.* see Self and Liang [78])—but the traditional statistical model is not necessarily the most useful model for harnessing the power of negative deductions.

Finally, a fundamental challenge is to develop and to extend the theory of "optimal" statistical tests for time- and space-bounded computations, and to identify such optimal tests for language-recognition problems. For some relevant foundational work, see Blum and Goldreich [14], Boppana and Hirschfeld [15], and Yao [95]. This important area offers a synergistic opportunity for cooperation among statisticians, complexity theorists, and cryptologists.

9 Conclusion

In this introductory guide, we have shown how language recognition can be performed in a classical statistical model of pattern recognition using standard statistical tests based on Markov models of language. We identified four well-defined plaintext recognition problems, and we derived likelihood ratio tests for each of these problems. For Problem 2 (distinguishing a known language from uniform noise) we observed that the Sinkov test is optimal in the sense that it is a most powerful test. Along the way, we have identified useful references in the statistics and pattern-recognition literature for further reading.

For language-recognition problems which typically have no uniformly most powerful test, including Problems 1, 3, and 4, the theory of statistics offers no clear recommendation on what test to use. In particular, many tests for problems with composite alternative hypotheses are incomparable with regard to the standard notion of statistical power. What test is most appropriate depends on many factors, including: the application; the model; how well the model fits the application; the particular problem; the costs of various types of experimental outcomes; engineering, financial, and computational constraints; and the cryptanalyst's philosophical beliefs. Moreover, statisticians have been debating this question for decades and still have not reached a consensus,

except to agree that many tests are incomparable with respect to statistical power. And little is known about optimal statistical tests when time and space complexity are included in the evaluation criteria. Nevertheless, the test statistics described in this paper offer a reasonable starting point and are likely to yield good results for many applications.

If better methods exist for solving practical language-recognition problems, they are likely to exploit particular constraints of the application or to exploit alternative models. For example, if restrictions can be imposed on a composite hypothesis, such restrictions might lead to more effective tests.

Despite extensive knowledge about the behavior of test statistics when applied to the idealized models of Markov languages, little is known about their distributions when applied to natural languages. In our companion paper [29], we explore this question through computer experiments, using real and simulated English from the Brown Corpus [24, 59].

Language-recognition problems are important both in cryptanalysis and in other settings. For example, in our multi-lingual society it would be useful to have communication systems and natural-language interfaces that automatically recognize what language is being spoken. Similarly, by classifying the base language of unknown words and phrases, programs that read text aloud could intelligently guess how to pronounce such unknown phrases. Statistical techniques in language recognition provide powerful tools for solving these and other language-recognition problems.

At the beginning of this project we started with the three works on plaintext recognition that had been published in the cryptologic literature (see Section 2). This paper extends that knowledge by applying techniques from the theory of statistical inference on Markov chains. We hope our introduction and guide will be of interest and use to practitioners who wish to solve language-recognition problems.

Acknowledgments

We are very grateful to Peter Matthews for helpful comments and suggestions, and for pointing out works by Diaconis and Efron [21], Huber [50], McCloskey and Pittenger [68], and Self and Liang [78]. In addition, we thank Robert Baldwin, James Mayfield, Raymond Pyle (Bell Atlantic), and James Sasaki for editorial comments. At the beginning of the project, Harold P. Edmundson pointed out papers by Billingsley [10] and Hoel [47].

References

[1] Agresti, Alan, "A survey of exact inference for contingency tables," *Statistical Science*, **7**:1 (February 1992), 313–177.

[2] Andelman, Dov; and James Reeds, "On the cryptanalysis of rotor machines and substitution-permutation networks," *IEEE Transactions on Information Theory*, **IT-28**:4 (July 1982), 578–584.

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

[4] Anderson, T. W.; and Leo A. Goodman, "Statistical inference about Markov chains," *Annals of Mathematical Statistics*, **28** (1957), 89–110.

[5] Baldwin, Robert W., "Crypt Breaker's Workbench users manual," MIT Laboratory for Computer Science (October 1986), unpublished manuscript. 21 pages.

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

[7] Bartlett, M. S., "The frequency goodness of fit test for probability chains," *Proceedings of the Cambridge Philosophical Society*, **47** (1951), 86–95.

[8] Beker, Henry; and Fred Piper, *Cipher Systems: The Protection of Communications*, John Wiley (New York, 1982).

[9] Bhat, U. Narayan, *Elements of Applied Stochastic Processes*, John Wiley (New York 1984).

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

[11] Billingsley, Patrick, *Statistical Inference for Markov Processes*, University of Chicago Press (Chicago 1961).

[12] Billingsley, Patrick, *Probability and Measure*, John Wiley (1986).

[13] Bishop, Y.; S. Fienberg; and P. Holland, *Discrete Multivariate Analysis*, MIT Press (Cambridge, MA, 1975).

[14] Blum, Emanuel; and Oded Goldreich, "Towards a computational theory of statistical tests (Extended abstract)" in *Proceedings of the 33rd Annual IEEE Symposium on Foundations of Computer Science*, IEEE Press (1992), 406–416.

[15] Boppana; Ravi B.; and Rafael Hirschfeld, "Pseudorandom generators and complexity classes," *Advances in Computing Research*, vol. 5, edited by Silvio Micali, JAI Press (1989), 1–26.

[16] Cain, Thomas; and Alan T. Sherman, "Cryptanalysis of filter generators from ciphertext alone" (1993), in preparation.

[17] Cain, Thomas; and Alan T. Sherman, "How to break Gifford's Cipher" (1993), in preparation.

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

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

[20] Denning, Dorothy E. R., *Cryptography and Data Security*, Addison-Wesley (Reading, MA, 1983).

[21] Diaconis, Persi; and Bradley Efron, "Testing for independence in a two-way table: New interpretations of the chi-square statistic," *Annals of Statistics*, **13**:3 (September 1985), 845–874.

[22] Dickey, James M.; and Lientz, B. P., "The weighted likelihood ratio, sharp hypotheses about chances, the order of a Markov chain," *Annals of Mathematical Statistics*, **41**:1 (1970), 214–226.

[23] Feldman, Frank A., "A new spectral test for nonrandomness and the DES," *IEEE Transactions on Software Engineering*, **16**:3 (March 1990), 261–267.

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

[25] Friedman, William F., "The index of coincidence and its applications in cryptanalysis," Technical Paper, War Department, Office of the Chief Signal Officer, United States Government Printing Office (Washington, D.C., 1925). [Available through Aegean Park Press.]

[26] Friedman, William F.; and Lambros D. Callimahos, *Military Cryptanalytics, Part I, Volume 2*, Aegean Park Press.

[27] Fukunaga, Keinosuke; and Raymond R. Hayes, "Effects of sample size in classifier design," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**:8 (August 1989), 873–885.

[28] Gallager, Robert G., *Information Theory and Reliable Communication*, John Wiley (New York, 1968).

[29] Ganesan, Ravi; and Alan T. Sherman, "Statistical techniques for language recognition: An empirical study using real and simulated English" (1993), in preparation.

[30] Good, I. J., "The likelihood ratio test for Markov chains," *Biometrika*, **42**, parts 3–4 (December 1955), 531–533.

[31] Good, I. J., "The frequency count of a Markov chain and the transition to continuous time," *Annals of Mathematical Statistics*, **32**:1 (1961), 41–48.

[32] Good, I. J., "Weight of evidence, causality and false-alarm probabilities" in *Information Theory*, edited by Colin Cherry, Butterworths (1961), 125–136.

[33] Good, I. J., Comment on the paper, "Topics in the investigation of linear relations fitted by the method of least squares," by F. J. Anscombe, *Journal of the Royal Statistical Society*, **29**:1 (1967), 39–42.

[34] Good, I. J., "Statistics of Language: Introduction" in *Encyclopaedia of Linguistics, Information and Control*, Meetham, A. R., ed., Pergamon Press (Oxford, 1969), 567–581.

[35] Good, I. J., "Studies in the history of probability and statistics. XXXVII A.M. Turing's statistical work in World War II," *Biometrika*, **66**:2 (1979), 393–396.

[36] Good, I. J., Comment on Patil and Taillie's paper on diversity, *Journal of the American Statistical Association*, **77**:379 (September 1982), 561–563.

[37] Good, I. J. *Good Thinking: The Foundations of Probability and its Applications*, University of Minnesota Press (Minneapolis, 1983).

[38] Good I. J., *Probability and the Weighing of Evidence*, Charles Griffen (London, 1950).

[39] Good, Irving John, *The Estimation of Probabilities: An Essay on Modern Bayesian Methods*, MIT Press (Cambridge, MA, 1965).

[40] Good, I. J., "A Bayesian significance test for multinomial distributions," *Journal of the Royal Statistical Society*, **B 29**:3 (1967), 399–431.

[41] Good, I. J., "On the application of symmetric Dirichlet distributions and their mixtures to contigency tables," *Annals of Statistics*, **4**:6 (1976), 1159–1189.

[42] Good, I. J., "The fast calculation of the exact distribution of Pearson's Chi-squared and of the number of repeats within cells of a multinomial by using a fast Fourier transform," *Journal of Statistical Computation and Simulation*, **14**:1 (1981), 71–78.

[43] Good, I. J., "The Bayes/Non-Bayes compromise: A brief review," *Journal of the American Statistical Association*, **87**:419 (September 1992), 597–606.

[44] Good, Irving John; and James Flinn Crook, "The Bayes/non-Bayes compromise and the multinomial distribution," *Journal of the American Statistical Association*, **69**:347 (1974), 711–720.

[45] Good, I. J.; T. N. Gover; and G. J. Mitchell, "Exact distributions for X^2 and for the likelihood-ratio statistic for the equiprobable multinomial distribution," *Journal of the American Statistical Association*, **65**:329 (March 1970), 267–283.

[46] Ho, Yu-Chi; and Ashok K. Agrawala, "On pattern classification algorithms introduction and survey," *Proceedings of the IEEE*, **56**:12 (December 1968), 2101–2114.

[47] Hoel, Paul G., "A test for Markoff chains," *Biometrika*, **41**, parts 3–4 (1954), 430–433.

[48] Hoel, Paul G., *Introduction to Mathematical Statistics*, John Wiley (New York, 1963).

[49] Hoel, P. G.; and R. P. Peterson, "A solution to the problem of optimum classification," *Annals of Mathematical Statistics*, **20**:3 (September 1949), 433–437.

[50] Huber, Peter J., *Robust Statistics*, John Wiley (New York, 1981).

[51] Juang, B.-H.; and L.P. Rabiner, "A probabilistic distance measure for hidden Markov models," *AT&T Technical Journal*, **64**:2 (February 1985), 391–408.

[52] Kahn, David, *The Codebreakers: The Story of Secret Writing* MacMillan, (New York, 1967).

[53] Kanal, Laveen, N., "Automatic pattern recognition," Technical Report UMIACS-TR-91-11/CS-TR-2594, University of Maryland College Park (January 1991).

[54] Kelton, Christinas M. L.; and W. David Kelton, "Development of specific hypothesis tests for estimated Markov chains," *Journal of Statistical Computation and Simulation*, **23** (1985), 15–39.

[55] Kelton, W. David ; and Christinas M. L. Kelton, "Hypothesis tests for Markov process models estimated from aggregate frequency data," *Journal of the American Statistical Association*, **79**:388 (December 1984), 922–928.

[56] Kemeny, John G.; and J. Laurie Snell, *Finite Markov Chains*, Van Nostrand (Princeton, 1967).

[57] Knuth, Donald E., *Seminumerical Algorithms* in *The Art of Computer Programming*, vol. 2 (Reading, MA, 1981).

[58] Krishnamurthy, Vikram; John B. Moore; and Shin-Ho Chung, "Hidden Markov model signal processing in presence of unknown deterministic interferences," *IEEE Transactions on Automatic Control*, to appear.

[59] Kučera, Henry; and W. Nelson Francis W., *Computational Analysis of Present-Day American English*, Brown University Press (Providence, RI, 1967).

[60] Kullbach, Solomon, *Statistical Methods in Cryptanalysis*, Aegean Park Press (Laguna Hills, CA, 1976).

[61] Kullback, S.; M. Kupperman; and H. H. Ku, "Tests for contingency tables and Markov chains," *Technometrics*, **4**:4 (1962), 573–608.

[62] Larsen, Richard J.; and Morris L. Marx, *An Introduction to Mathematical Statistics and its Applications*, Prentice-Hall (Englewood Cliffs, 1981).

[63] Lee, Kai-Fu; and Hsiao-Wuen Hon, "Speaker-independent phone recognition using hidden Markov models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **37**:11 (November 1989), 1641–1648.

[64] Lehmann, E. L., *Testing Statistical Hypotheses*, Wiley (New York, 1986).

[65] Ljolje, Andrej; and Stephen E. Levinson, "Development of an acoustic-phonetic hidden Markov model for continuous speech recognition," *IEEE Transactions on Signal Processing*, **39**:1 (January 1991), 29–39.

[66] Lund, Michael A.; and C. C. Lee, "Wald's SPRT applied to speaker verification," *Processings of the 30th Annual Allerton Conference on Communication, Control, and Computing* (1993), to appear.

[67] Mendal, J. M.; and K. S. Fu, *Adaptive Learning and Pattern Recognition Systems*, Academic Press (New York, 1970).

[68] McCloskey, Joseph P.; and Arthur O. Pittenger, "Maximum likelihood estimates for linear models with linear constraints" in *Proceedings of the NSA Mathematical Sciences Meetings*, Nos. 6–8, edited by Robert L. Ward (January and October 1987), 137–158.

[69] Niemann, Heinrich, *Pattern Analysis and Understanding*, Springer-Verlag (New York, 1990).

[70] Osteyee, David Bridston; and Irving John Good, *Information, Weight of Evidence, the Singularity between Probability Measures and Signal Detection*, Springer-Verlag (Berlin, 1974).

[71] Poor, H. V., *An Introduction to Signal Detection and Estimation*, Springer-Verlag (New York, 1988).

[72] Rabiner, Lawrence R., "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, **77**:2 (February 1989), 257–286.

[73] Raviv, J., "Decision making in Markov chains applied to the problem of pattern recognition," *IEEE Transactions on Information Theory*, **IT-3**:4 (October 1967), 536–551.

[74] Reeds, J. A.; and P. J. Weinberger, "File security and the Unix system crypt command," *AT&T Bell Laboratories Technical Journal*, **63** (October 1984), 1673–1683.

[75] Rivest, Ronald L., "Statistical analysis of the Hagelin cryptograph," *Cryptologia*, **5**:1 (January 1981), 27–32.

[76] Rivest, Ronald L., "Cryptography" in *Handbook of Theoretical Computer Science*, vol. A, edited by Jan van Leeuwen, Elsevier/MIT Press (1990), Chapter 13, 717–755.

[77] Rohatgi, Vijay K., *statistical Inference*, John Wiley (New York, 1984).

[78] Self, Steven G.; and Kung-Yee Liang, "Asymptotic properties of maximum likelihood estimators and likelihood ratio tests under nonstandard conditions," *American Statistical Association*, **82**:398 (June 1987), 605–610.

[79] Shannon, Claude E., "Communication theory of secrecy systems," *Bell System Technical Journal*, **28** (October 1949), 656–715.

[80] Shannon, C. E., "Prediction and entropy of printed English," *Bell System Technical Journal*, **30**, (January 1951), 50–64.

[81] Simmons, Gustavus J., ed., *Contemporary Cryptology: The Science of Information Integrity*, IEEE Press (New York, 1992).

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

[83] Solso, Robert L.; Paul F. Barbuto, Jr.; and Connie L. Juel, "Bigram and trigram frequencies and versatilities in the English language," *Behavior Research Methods & Instrumentation*, **11**:5 (1979), 475–484.

[84] Solso, Robert L.; and Connie L. Juel, "Positional frequency and versatility of bigrams for two- through nine-letter English words," *Behavior Research Methods & Instrumentation*, **12**:3 (1980), 297–343.

[85] Solso, Robert L.; Connie Juel; and David C. Rubin, "The frequency and versatility of initial and terminal letters in English words," *Journal of Verbal Learning and Verbal Behavior*, **21** (1982), 220–235.

[86] Solso, Robert L.; and Joseph F. King, "Frequency and versatility of letters in the English language," *Behavior Research Methods & Instrumentation*, **8**:3 (1976), 283–286.

[87] Tishby, Naftali Z., "On the application of mixture AR hidden Markov models to text independent speaker recognition," *IEEE Transactions on Signal Processing*, **39**:3 (March 1991), 563–570.

[88] Trees, Harry L. van, *Detection, Estimation, and Linear Modulation Theory* in *Detection, Estimation, and Modulation Theory*, Part I, John Wiley (New York, 1968).

[89] Trees, Harry L. van, *Radar-Sonar Signal Processing and Gaussian Signals in Noise* in *Detection, Estimation, and Modulation Theory*, Part III, John Wiley (New York, 1971).

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

[91] Valiveti, R. S.; and B. J. Oommen, "Recognizing sources of random strings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**:4 (April 1991), 386–394.

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

[93] Wilks, S. S., *Mathematical Statistics*, John Wiley (New York, 1962).

[94] Yakowitz, Sidney J., "Small-sample hypothesis tests of Markov order, with application to simulated and hydrologic chains," *Journal of the American Statistical Association*, **71**:353 (March 1976), 132–136.

[95] Yao, A. C., "Computational information theory" in *Complexity in Information Theory*, Yaser S. Abu-Mostafa, ed., Springer (New York, 1988), 1–15.