A Markov model of the Indus script

Rajesh P. N. Rao[a,1], Nisha Yadav[b,c], Mayank N. Vahia[b,c], Hrishikesh Joglekar[d], R. Adhikari[e], and Iravatham Mahadevan[f]

[a]Department of Computer Science and Engineering, University of Washington, Seattle, WA 98195; [b]Department of Astronomy and Astrophysics, Tata Institute of Fundamental Research, Mumbai 400005, India; [c]Centre for Excellence in Basic Sciences, Mumbai 400098, India; [d]14, Dhus Wadi, Laxminiketan, Thakurdwar, Mumbai 400002, India; [e]Institute of Mathematical Sciences, Chennai 600113, India; and [f]Indus Research Centre, Roja Muthiah Research Library, Chennai 600113, India

Communicated by Roddam Narasimha, Jawaharlal Nehru Centre for Advanced Scientific Research, Bangalore, India, June 15, 2009 (received for review December 26, 2008)

Although no historical information exists about the Indus civilization (flourished *ca*. 2600–1900 B.C.), archaeologists have uncovered about 3,800 short samples of a script that was used throughout the civilization. The script remains undeciphered, despite a large number of attempts and claimed decipherments over the past 80 years. Here, we propose the use of probabilistic models to analyze the structure of the Indus script. The goal is to reveal, through probabilistic analysis, syntactic patterns that could point the way to eventual decipherment. We illustrate the approach using a simple Markov chain model to capture sequential dependencies between signs in the Indus script. The trained model allows new sample texts to be generated, revealing recurring patterns of signs that could potentially form functional subunits of a possible underlying language. The model also provides a quantitative way of testing whether a particular string belongs to the putative language as captured by the Markov model. Application of this test to Indus seals found in Mesopotamia and other sites in West Asia reveals that the script may have been used to express different content in these regions. Finally, we show how missing, ambiguous, or unreadable signs on damaged objects can be filled in with most likely predictions from the model. Taken together, our results indicate that the Indus script exhibits rich synactic structure and the ability to represent diverse content. both of which are suggestive of a linguistic writing system rather than a nonlinguistic symbol system.

ancient scripts | archaeology | linguistics | machine learning | statistical analysis

The Indus (or Harappan) civilization flourished from *ca*. 2600 to 1900 B.C. in a vast region spanning what is now Pakistan and northwestern India. Its trade networks stretched to the Persian Gulf and the Middle East. The civilization emerged from the depths of antiquity in the late 19th century, when General Alexander Cunningham (1814–1893) visited a site known as Harappa and published a description (1), including an illustration of a tiny seal with characters in an unknown script. Since then, much has been learned about the Indus civilization through the work of archaeologists (see refs. 2 and 3 for reviews), but the script still remains an enigma.

More than 3,800 inscriptions in the Indus script have been unearthed on stamp seals, sealings, amulets, small tablets, and ceramics (see Fig. 1*A* for examples). Although there have been >60 claimed decipherments (4), none of these has been widely accepted by the community. Several obstacles to decipherment have been identified (4), including the lack of any bilinguals, the brevity of the inscriptions (the average inscription is ≈5 signs long), and our almost complete lack of knowledge of the language(s) used in the civilization.

Given these formidable obstacles to direct decipherment, we propose instead the analysis of the script's syntactical structure using techniques from the fields of statistical pattern analysis and machine learning (5). It is our belief that such an approach could provide valuable insights into the grammatical structure of the script, paving the way for a possible eventual decipherment. As a first step in this endeavor, we present here results obtained from analyzing the sequential structure of the Indus script using a simple type of probabilistic graphical model (6) known as a Markov model (7). Markov models assume that the current "state" (e.g., symbol in a text) depends only on the previous state, an assumption that renders learning and inference over sequences tractable. Although this assumption is a simplification of the complex sequential structure seen in languages, Markov models have proved to be extremely useful in analyzing a range of sequential data, including speech (8) and natural language (9). Here, we apply them to the Indus script. A major goal of this article is to provide an exposition of Markov models for those unfamiliar with them and to illustrate their usefulness in the study of an undeciphered script. We leave the application of more advanced higher-order models and grammar induction methods to other papers (10) and future work.

Markov Models for Analyzing the Indus Script. A Markov model (also called a Markov chain) (7, 11, 12) consists of a finite set of N "states" $s_1, s_2, \ldots, s_N$ (e.g., the states could be the signs in the script) and a set of conditional (or transition) probabilities $P(s_i|s_j)$ that determine how likely it is that state s_i follows state s_j. There is also a set of prior probabilities $P(s_i)$ that denote the probability that state s_i starts a sequence. Fig. 1*B* shows an example of a "state diagram" for a Markov model with 3 states labeled A, B, and #, where A and B denote letters in a language, and # denotes the end of a text. Fig. 1*B* also shows the prior probabilities $P(s_i)$ and the transition probabilities $P(s_i|s_j)$, picked arbitrarily here for the purposes of illustration. Some example sequences generated by this Markov model are BAAB, ABAB, B, etc. (the terminal sign # is not shown). Texts that are not generated by this Markov model include all texts that contain a repetition of B (…BB…) and all texts that end in A, because these are precluded by the transition probability table. A more realistic example would be a Markov model for English texts involving the 26 letters of the alphabet plus space. In this case, the transition probability table (or matrix) would be of size 27×27. In the matrix, we would expect, for example, higher probabilities for the letter "s" to be immediately followed by letters such as "e," "o," or "u," than letters such as "x" or "z," because of the morphological structure of words in English. A Markov model learned from a corpus of English texts would capture such statistical regularities.

Markov models [and their variants, hidden Markov models (HMMs)] are special cases of a more general class of probabilistic models known as graphical models (6). Graphical models can be used to model complex relationships between states, including higher-order dependencies such as the dependence of

Author contributions: R.P.N.R., N.Y., M.N.V., H.J., and R.A. designed research; R.P.N.R., N.Y., H.J., and R.A. performed research; I.M. contributed new reagents/analytic tools; R.P.N.R., N.Y., M.N.V., H.J., and R.A. analyzed data; and R.P.N.R., M.N.V., and R.A. wrote the paper.

The authors declare no conflict of interest.

Freely available online through the PNAS open access option.

[1]To whom correspondence should be addressed. E-mail: rao@cs.washington.edu. .

This article contains supporting information online at www.pnas.org/cgi/content/full/0906237106/DCSupplemental

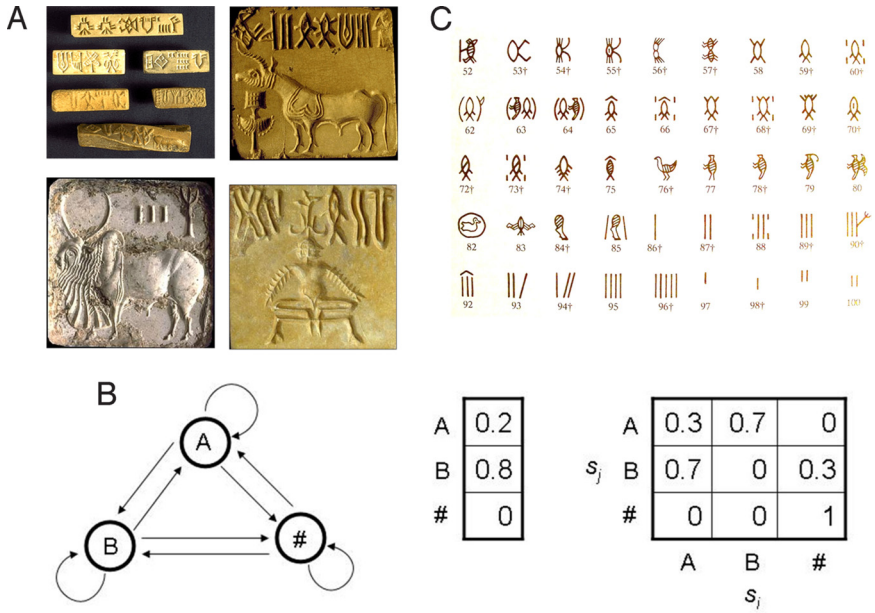


Fig. 1. The Indus script and Markov models. (*A*) Examples of Indus inscriptions on seals and tablets (*Upper Left*) and 3 square stamp seals (image credit: J. M. Kenoyer, Harappa.com). Note that inscriptions appear reversed on seals. (*B*) Example of a simple Markov chain model with 3 states. (*C*) Subset of Indus signs from Mahadevan's list of 417 signs (13).

a symbol on the past N-1 symbols [equivalent to N-gram models in language modeling (9)]. In this article, we focus on first-order Markov models. In other work (10), we have compared N-gram models for Indus texts using the information-theoretic measure of "perplexity" (9). We have found that the bulk of the perplexity in the Indus corpus can be captured by a bigram ($n = 2$) model (or equivalently, a first-order Markov model), supporting the usefulness of such models in analyzing the Indus script.

We focus in this article on 3 applications of Markov models to analyzing the Indus script: (*i*) Sampling: We show how new sample texts can be generated by randomly sampling from the prior probability distribution $P(s_i)$ to obtain a starting sign (say S_1) and then sampling from the transition probabilities $P(s_i|S_1)$ to obtain the next sign S_2, and so on. The string generation process can be terminated by assuming an end of text (EOT) sign that transitions to itself with probability one. (*ii*) Likelihood computation: If $\boldsymbol{x}$ is a string of length L and is of the form $x_1x_2\ldots x_L$, where each x_i is a sign from the list of signs, the likelihood that $\boldsymbol{x}$ was generated by a Markov model M can be computed as: $P(\mathbf{x}|M) = P(x_1x_2\ldots x_L|M) = P(x_1)P(x_2|x_1)P(x_3|x_2)\ldots P(x_L|x_{L-1})$. The likelihood of a string under a particular model tells us how closely the statistical properties of the given string match those of the original strings used to learn the given model. Thus, if the original strings were generated according to the statistical properties of a particular language, the likelihood is useful in ascertaining whether a given string might have been generated by the same language. (*iii*) Filling in missing signs: Given a learned Markov model M and an input string $\boldsymbol{x}$ with some known signs and some missing signs, one can estimate the most likely complete string $\boldsymbol{x}^*$ by calculating the most probable explanation (MPE) for the unknown parts of the string (see *Methods* for details).

The Indus Script. A prerequisite for understanding the sequential structure of a script is to identify and isolate its basic signs. This task is particularly difficult in the case of an undeciphered script such as the Indus script, because there is considerable variability in the rendering of the signs, making it difficult to ascertain whether 2 signs that look different are stylistic variants of the same sign or 2 independent signs. After an analysis of the positional statistics of variant signs in a corpus of known inscriptions (*ca*. 1977), Mahadevan arrived at a list of 417 independent signs in his concordance (13) [Parpola used similar methods to estimate a slightly shorter list of 386 signs (14)]. We used this list of 417 signs as the basis for our study. Fig. 1*C* shows a subset of signs from this list.

Barring a few exceptions (see p. 14 in ref. 13), the writing direction is generally accepted to be predominantly from right to left (i.e., left to right in seals and right to left in the impressions). There exists convincing external and internal evidence supporting this hypothesis (e.g., refs. 4, 13, and 14). We consequently learned the sequential structure of the Indus texts based on a right-to-left reading of their signs, although a Markov model could equally well be learned for the opposite direction.

Results

Markov Model of Indus Texts. The learned Markov model provides several interesting insights into the nature of the Indus script. First, examining the learned prior probabilities $P(s_i)$ provides information about how likely it is that a particular sign s_i starts a text. Fig. 2*B* shows this probability for the 10 most frequently occurring signs (Fig. 2*A*) in the corpus of Indus inscriptions.

An examination of Fig. 2*B* reveals that certain frequently occurring signs such as ⟐ and ⊗ (signs numbered 3 and 10 in Fig. 2*B*) are much more likely to start a text than the others. On the other hand, certain signs such as ⋃ (the most frequent sign in the corpus) and ⍭ are highly unlikely to start a text (they are, in fact, highly likely to end a text; see Fig. 3*C* and Table 1). These observations have also been made by others (13–16).

From the learned values for $P(s_i)$, one can also extract the 10 most likely signs to start a text and the 10 least likely signs to do so (among signs occurring at least 20 times in the dataset) (Fig. 3*A*). These results suggest that some signs that look similar, such as | and ᛁ, subserve different functions within the script and thus cannot be considered variants of the same sign.

The matrix of transition probabilities $P(s_i|s_j)$ learned from data is shown in Fig. 2*C* (only the portion corresponding to the 10 most frequent signs is shown). The learned transition prob-

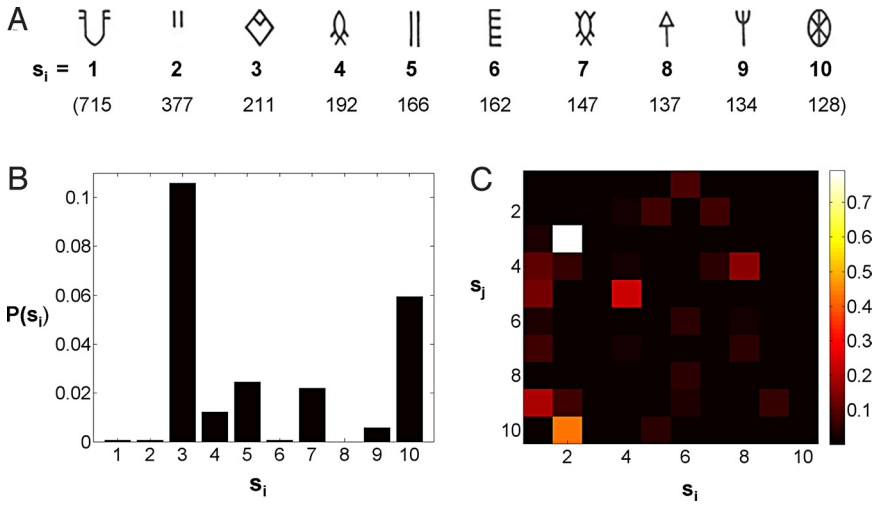


Fig. 2. Learned Markov model for the Indus script. (*A*) The 10 most frequently occurring Indus signs in the dataset. The numbers within parentheses in the bottom row are the frequencies of occurrence of each sign depicted above in the dataset used to train the model. (*B*) Prior (starting) probabilities $P(s_i)$ learned from data, shown here for the 10 most frequently occurring Indus signs depicted in *A*. (*C*) Matrix of transition probabilities $P(s_i|s_j)$ learned from data. A 418 × 418 matrix was learned but to aid visualization, only the 10 × 10 portion corresponding to the 10 most frequent signs is shown.

abilities for Indus sign pairs can be shown to be significant, in the sense that the null hypothesis for independence can be rejected (see ref. 10 for details). To interpret the transition probability matrix, consider the entry with the highest value (colored white in Fig. 2*C*). This value (≈0.8) corresponds to the probability $P(s_i = 2|s_j = 3)$ and indicates that the sign ◇ is followed by the sign 𝍐 with a very high probability. Other highly probable pairs and longer sequences of signs are shown in Fig. 3*B*.

The transition matrix also tells us which signs are most likely to end a text (these are signs that have a high probability of being followed by the EOT symbol). Fig. 3*C* (top row) shows 10 signs that occur at least 20 times in the dataset, which have high probabilities (in the range 0.87–0.39) of terminating a text. Fig. 3*C* (bottom row) shows 10 frequently occurring signs (occurring at least 20 times in the dataset) that have the least probability of terminating a text.

The diagonal of the transition matrix [representing $P(s_i|s_i)$] tells us how likely it is that a given sign follows itself (i.e., repeats). Fig. 3*D* shows 10 signs with the highest probability of repeating. The sign 𓀠 occurs in only one inscription where it occurs as a pair, whereas the sign 𓀀 occurs in 8 inscriptions, 6 times as a pair. More interestingly, the frequently occurring sign 𓏏 occurs as a pair in 33 of the 58 inscriptions in which it occurs. The presence of such repeating symbols in the Indus texts puts strong constraints on their semantic interpretation, because the interpretation has to remain intelligible when repeated.

The dark regions in the transition matrix in Fig. 2*C* indicate a transition probability that is near zero. An inspection of the transition matrix reveals that for a given sign, there exists a specific subset of signs that have a high probability of following that sign, with the rest of the entries of the transition matrix being close to zero. This suggests that the Indus texts exhibit an intermediate degree of flexibility between random and rigid symbol order: there is some amount of flexibility in what the next sign can be, but not too much flexibility. This intermediate degree of flexibility (or randomness) of strings is also exhibited by most natural languages. A comparison of the Indus script with other linguistic and nonlinguistic systems using the measure of conditional entropy to quantify this flexibility is given in ref. 17 (see also ref. 18). One way in which such structure in the transition matrix could arise is from grammar-like rules that allow only a specific subset of signs to follow any given sign. The existence of such rules makes it more likely that the Indus texts encode linguistic information, rather than being random or rigid juxtapositions of emblems or religious and political symbols (19).

Table 1 provides additional support for the hypothesis that sequences of signs in the Indus texts may be governed by specific rules. Table 1 shows that for each of the 10 most frequently occurring signs in the dataset, only a small set of signs has a high

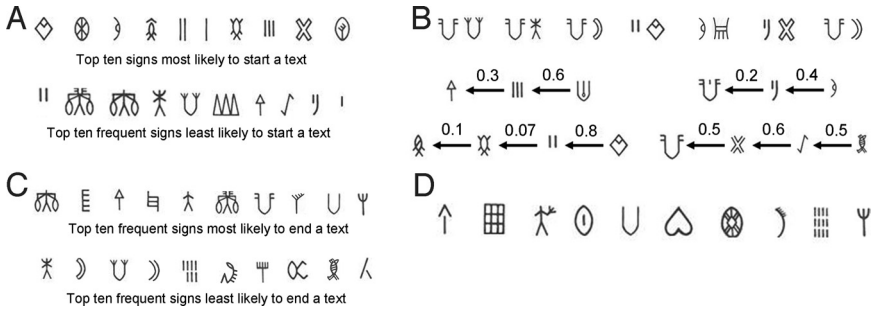


Fig. 3. Some characteristics of the Indus script extracted from the learned Markov model. (*A*) Signs most likely and least likely to start an Indus text. (*B*) Highly probable pairs (top row) and longer sequences of Indus signs predicted by the learned transition matrix. The numbers above the arrows are the learned probabilities for the transitions indicated by the arrows. (*C*) Signs most likely and least likely to end an Indus text. (*D*) Top 10 signs with the highest probability of repeating (sign following itself).

Table 1. Signs that tend to be followed by or follow each of the top 10 frequently occurring signs

Preceding signs (in order of decreasing probability)	Sign	High probability successor signs (in order of decreasing probability)
(0.92, 0.89, 0.87, 0.62, 0.55)		EOT (0.74, 0.08, 0.07)
(0.80, 0.43, 0.35, 0.18, 0.14)		(0.07, 0.07, 0.06, 0.06, 0.04, 0.04, 0.04, 0.04)
(0.03, 0.02, 0.02)		(0.80, 0.09, 0.04, 0.03, 0.01)
(0.53, 0.23, 0.16, 0.13, 0.05, 0.05)		(0.16, 0.10, 0.08, 0.06, 0.04)
(0.43, 0.07, 0.06, 0.05, 0.05)		(0.23, 0.13, 0.10, 0.04, 0.04)
(0.16, 0.15, 0.12, 0.08)		(0.84, 0.04, 0.03, 0.02)
(0.22, 0.08, 0.08, 0.07, 0.07)		(0.10, 0.08, 0.06, 0.05, 0.04, 0.04, 0.03)
(0.41, 0.3, 0.16, 0.14, 0.13, 0.05)		(0.83, 0.05, 0.02)
(0.38, 0.23, 0.14, 0.13, 0.11)		(0.4, 0.19, 0.06, 0.05, 0.05)
(0.13, 0.03, 0.02, 0.02, 0.02)		(0.43, 0.15, 0.10, 0.07, 0.04)

Note: The table assumes a right to left reading of the texts. The number below each sign is the value from the transition probability matrix for the corresponding sign preceding (left column) or following (right column) a given sign (center column). Only signs that occur 20 or more times in the dataset are included in this table.

probability of following the given sign or being followed by it. The possibility of grammar-like rules is further strengthened by the observation that many of these sets of high probability signs exhibit similarities in visual appearance: for example, among the group of signs with a high probability of following the sign ||| are the signs, , , , and , as well as and . Other such examples can be observed in Table 1 and in the full transition matrix, hinting at distinct syntactic regularities underlying the composition of Indus sign sequences.

Generating New Sample Indus Texts from the Learned Model. The Markov model learned from the Indus inscriptions acts as a "generative" model, in that one can sample from it and obtain new sequences of signs that conform to the sequential statistics of the original inscriptions (albeit limited to pairwise statistics). Fig. 4*A* provides an example of a new text obtained by sampling the learned model, along with the closest matching text in the original corpus. The closest match was computed using the "string edit distance" between strings, which measures the number of additions, deletions, and replacements needed to go from one string to the other. The generated text in Fig. 4*A* is not identical to the closest matching Indus text but differs from it in 2 interesting ways. First, the symbol occurs as the starting symbol instead of . An examination of the transition matrix reveals that both and have a high probability of being followed by the sign . Second, the sample text contains the sign instead of in the same position. This suggests that and may have similar functional roles, given that they occur within similar contexts.

Fig. 4*B* (top row) gives another example of a new generated Indus text and 2 closest matching texts from the Indus dataset of inscriptions. Once again, based on their interchangeability in these and other texts, one may infer that the signs , , and share similar functional characteristics in terms of where they may appear within texts.

Filling in Incomplete Indus Inscriptions. Many of the inscribed objects excavated at various Indus sites are damaged, resulting in inscriptions that contain missing or illegible signs. To ascertain whether the model trained on complete texts could be used to fill in the missing portion of incomplete inscriptions, we first generated an artificial dataset of "damaged" inscriptions by taking complete inscriptions from the Indus dataset and obliterating one or more signs. Fig. 4*C* (top row) shows an example of one such inscription. The complete inscription (middle row) predicted by the Markov model using the MPE method matched a preexisting Indus inscription (bottom row). A detailed cross-validation study of filling in performance revealed that a first-order Markov model can correctly predicted deleted signs with ≈74% accuracy (10), which is encouraging given that only pairwise statistics were learned. Further improvement in performance can be expected with higher-order models.

Fig. 4*D* (top row) shows an actual Indus text with missing signs (from ref. 14). The middle row shows the completed text generated by the MPE method, with the closest matching Indus

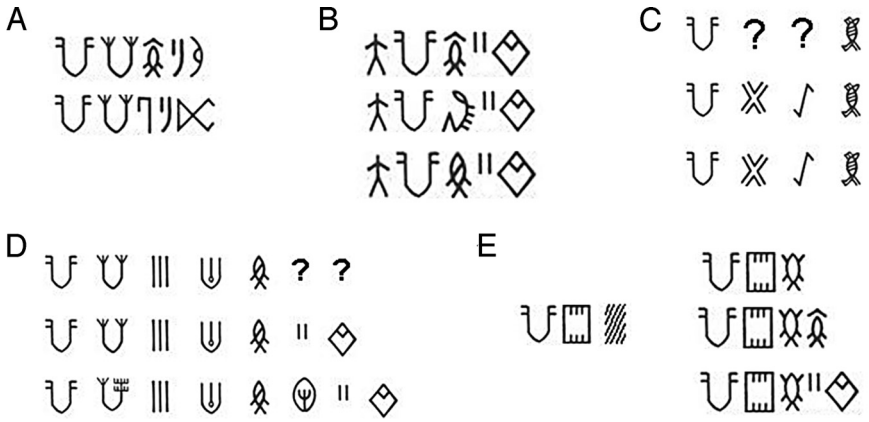
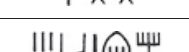


Fig. 4. Generating new Indus texts and filling in missing signs. (*A*) (*Upper*) New sequence of signs (read right to left) generated by sampling from the learned Markov model. (*Lower*) Closest matching actual Indus inscription from the corpus used to train the model. (*B*) Inferring functionally related signs. A sample from the Markov model (top) is compared with 2 closest matching inscriptions in the corpus, highlighting signs that function similarly within inscriptions. (*C*) Filling in missing signs in a known altered text. (*Top*) Inscription obtained by replacing 2 signs in a complete inscription with blanks (denoted by ?). (*Middle*) The MPE output. (*Bottom*) Closest matching text in the corpus. (*D*) Filling in an actual incomplete Indus inscription. (*Top*) An actual Indus inscription (from ref. 14, figure 4.8) with 2 signs missing. (*Middle*) MPE output. (*Bottom*) Closest matching complete Indus text in the corpus. (*E*) Filling in of another actual incomplete inscription from ref. 13. (*Left*) Text with an unknown number of missing signs (hashed box). (*Right*) Three complete texts of increasing length predicted by the model. The first and third texts actually exist in the corpus.

Table 2. Likelihood of West Asian Texts compared with Indus valley texts

West Asian Text (from [13])	Likelihood
	0
	3.11×10^{-10}
	7.09×10^{-8}
	6.34×10^{-14}
	0
	1.13×10^{-11}
	1.22×10^{-12}
	2.87×10^{-17}
Indus valley held-out texts (median)	1.12×10^{-7}

Note: Only complete and unambiguous West Asian texts from ref. 13 are included in this table. Two texts have a likelihood of zero, because they each contain a symbol not occurring in the training dataset used to learn the model. The last row shows for comparison the median likelihood for a randomly selected set of 100 texts originating from within the Indus valley, which were held out and not used for training the Markov model (these 100 texts had the same average length as the West Asian texts).

text at the bottom. The closest matching text differs from the generated text in 2 ways: the "modifier" has been inserted and the sign is replaced by a visually similar sign . The text shown at the left of Fig. 4*E* is another actual Indus inscription with an unknown number of signs missing (from ref. 13). The 3 texts shown at the right are MPE outputs assuming 1, 2, or 3 signs are missing. The first and third MPE texts actually occur in the Indus corpus, whereas the middle text contains the frequently occurring pair . Additional examples of filling in of damaged texts are given in supporting information (SI) Table S1.

Testing the Likelihood of Indus Inscriptions. The likelihood of a particular sequence of Indus signs with respect to the learned Markov model tells us how likely it is that the sign sequence belongs to the putative language encoded by the Markov model. We found that altering the order of signs in an existing Indus text typically caused the likelihood of the text to drop dramatically (*SI Text* and Fig. S1), supporting the hypothesis that the Indus texts are subject to specific syntactic rules determining the sequencing of signs.

Applying this analysis to Indus texts discovered outside the Indus valley, for example, in Mesopotamia and other sites in West Asia, we found that the likelihoods of most of these inscriptions are extremely low compared with their counterparts found in the Indus valley (Table 2). Indeed, the median value of likelihoods for the West Asian texts is 6.40×10^{-13}, which is $\approx$100,000 times less than the median value of 1.12×10^{-7} obtained for a random set of 100 texts of Indus valley origin that were excluded from the training set for comparison purposes. These findings suggest the intriguing possibility that the Indus script may have been used to represent a different language or subject matter by Indus traders conducting business in West Asia or West Asian traders sending goods back to the Indus valley. Such a possibility was earlier suggested by Parpola, who noted that the West Asian texts often contain unusual sign combinations (14). Our results provide a quantitative basis for this possibility. The low likelihoods arise from the fact that many of the West Asian texts in Table 2 contain sign combinations such as , , , , and that never appear in any texts found in the Indus valley, even though the signs themselves occur frequently in other combinations in the Indus valley texts.

Discussion

A number of researchers have made observations regarding sequential structure in the Indus script, focusing on frequently occurring pairs, triplets, and other groups of signs (13–15, 20). Koskenniemi suggested the use of pairwise frequencies of signs to construct syntax trees and segment texts, with the goal of eventually deriving a formal grammar (20). More recently, Yadav, Vahia, and colleagues (15, 16) have performed statistical analyses of the Indus texts, including explicit segmentation of texts based on most frequent pairs, triplets, and quadruplets.

In this article, we provide an investigation of sequential structure in the Indus script based on Markov models. An analysis of the transition matrix learned from a corpus of Indus texts provided important insights into which signs tend to follow particular signs and which signs do not. The transition matrix also provides a probabilistic basis for extracting common sequences and subsequences of signs in the Indus texts. We demonstrated how the learned Markov model can be used to generate new sample texts, revealing groups of signs that tend to function similarly within a text. The approach can also be used to fill in missing portions of illegible and incomplete Indus inscriptions based on the corpus of complete inscriptions. Finally, a comparison of the likelihood of Indus inscriptions discovered in West Asian sites with those from the Indus valley suggests that many of the West Asian inscriptions may represent subject matter different from Indus valley inscriptions.

Our results appear to favor the hypothesis that the Indus script represents a linguistic writing system. Our Markov analysis of sign sequences, although restricted to pairwise statistics, makes it clear that the signs do not occur in a random manner within inscriptions but appear to follow certain rules: (*i*) some signs have a high probability of occurring at the beginning of inscriptions whereas others almost never occur at the beginning; and (*ii*) for any particular sign, there are signs that have a high probability of occurring after that sign and other signs that have negligible probability of occurring after the same sign. Furthermore, signs appear to fall into functional classes in terms of their position within an Indus text, where a particular sign can be replaced by another sign in its equivalence class. Such rich syntactic structure is hard to reconcile with a nonlinguistic system. Additionally, our finding that the script may have been versatile enough to represent different subject matter in West Asia argues against the claim that the script merely represents religious or political symbols. Other arguments in favor of the linguistic hypothesis for the Indus script are provided by Parpola (21).

Our study suffers from some shortcomings that could be addressed in future work. First, our first-order Markov model captures only pairwise dependencies between signs, ignoring important longer-range dependencies. Higher-order Markov models (10) and other types of probabilistic graphical models (6) would allow more accurate modeling of such dependencies. A second potential shortcoming is our use of an Indus corpus of texts from 1977 (13). New texts and signs have since been discovered, and new sign lists have been suggested with up to 650 signs (22). However, the types of new material that have been

discovered are in the same categories as the texts in the 1977 corpus, namely, more seals, tablets, etc., exhibiting syntactic structure similar to that analyzed in this article. Also, the new signs that have been discovered are still far outnumbered by the most commonly occurring signs in the 1977 corpus, most of which also occur frequently in the newly discovered material. Therefore, we expect the variations in sign frequencies due to the new material to only slightly change the conditional probabilities in the Markov model. Nevertheless, a more complete analysis with all known texts and new sign lists remains an important objective for the future. Additionally, our analysis combined data from different geographical locations. A more detailed site-by-site analysis could shed light on the interesting question of whether there are differences in the sequential patterning of signs across regions.

In summary, the results we have presented strongly suggest the existence of specific rules governing the sequencing of Indus signs in a manner that is indicative of an underlying grammar. A formidable, but perhaps not insurmountable, challenge for the future (also articulated in refs. 14, 15, and 20) is to apply statistical and machine learning techniques to infer a grammar directly from the corpus of available Indus inscriptions.

Methods

We applied our Markov model analysis techniques to a subset of Indus texts extracted from Mahadevan's 1977 concordance (13). This dataset, called EBUDS (15), excludes all texts from Mahadevan's concordance containing ambiguous or missing signs and all texts having multiple lines on a single side of an object. In the case of duplicates of a text, only one copy is kept in the dataset. This resulted in a dataset containing 1,548 lines of text, with 7,000 sign occurrences. We used Mahadevan's list of 417 signs plus an additional EOT sign to denote the end of each text. Signs were fed to the model from right to left in each line of text, ending in the EOT sign.

Learning a Markov Model from Data. The parameters of a Markov model include the prior probabilities $P(s_i)$ and the transition probabilities $P(s_i|s_j)$. A simple method for computing these probabilities is counting frequencies, e.g., $P(s_i)$ is set equal to the number of times sign s_i begins a text divided by the total number of texts. This can be shown to be equivalent to maximum likelihood estimation of the parameters (9). However, such an estimate, especially for the transition probabilities $P(s_i|s_j)$, can yield poor estimates when the dataset is small, as is the case with the Indus script, because many pairs of signs may not occur in the small dataset at all, even though their actual probabilities may be nonzero. There has been extensive research on ''smoothing'' techniques, which assign small probabilities to unseen pairs based on various heuristic principles (see chapter 6 in ref. 9 for an overview). For the results in this article, we used modified Kneser–Ney smoothing (23) (based on ref. 24), a technique that has been shown to outperform other smoothing techniques on a number of benchmark datasets (23).

Filling in Missing Signs. Let $\boldsymbol{x} = x_1x_2\ldots x_L$ be a string of length L where each x_i is a random variable whose value can be any sign from the list of signs. Let X denote the set of x_i for which the values are given and Y the set of x_i with values missing. For a general graphical model M, the MPE for the missing variables Y given the ''evidence'' X is computed as $Y^* = \arg\max_Y P(Y|X,M)$. For the learned Markov model, $M = (\alpha, \pi)$, where $\alpha_i = P(s_i)$ and $\pi_{ij} = P(s_i|s_j)$. For a Markov model, we can compute the MPE $Y^* = \arg\max_Y P(Y|X, \alpha, \pi)$ using a version of the ''Viterbi algorithm'' (25) (see refs. 8 and 12 for algorithmic details).

ACKNOWLEDGMENTS. This work was supported by the Packard Foundation, the Sir Jamsetji Tata Trust, the Simpson Center for the Humanities at the University of Washington (UW), the UW College of Arts and Sciences, the UW College of Engineering and Computer Science and Engineering Department, and the Indus Research Centre, Chennai, India.

1. Cunningham A (1875) *Archaeological Survey of India Report for the Year 1872-73* (Archaeological Survey of India, Calcutta, India).
2. Kenoyer JM (1998) *Ancient Cities of the Indus Valley Civilisation* (Oxford Univ Press, Oxford, UK).
3. Possehl GL (2002) *The Indus Civilisation* (Alta Mira Press, Walnut Creek, CA).
4. Possehl GL (1996) *The Indus Age: The Writing System* (University of Pennsylvania Press, Philadelphia).
5. Bishop C (2008) *Pattern Recognition and Machine Learning* (Springer, Berlin).
6. Jordan MI (2004) Graphical models. *Stat Sci (Special Issue on Bayesian Statistics)* 19:140–155.
7. Markov AA (1906) Extension of the law of large numbers to dependent quantities (in Russian). *Izv Fiz-Matem Obsch Kazan Univ (2nd Ser)* 15:135–156.
8. Jelenik F (1997) *Statistical Methods for Speech Recognition* (MIT Press, Cambridge, MA).
9. Manning C, Schütze H (1999) *Foundations of Statistical Natural Language Processing* (MIT Press, Cambridge, MA).
10. Yadav N, et al. (2009) Statistical analysis of the Indus script using n-grams. arxiv: 0901.3017 (available at http://arxiv.org/abs/0901.3017).
11. Drake AW (1967) *Fundamentals of Applied Probability Theory* (McGraw–Hill, New York).
12. Rabiner LR (1989) A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proc IEEE* 77:257–286.
13. Mahadevan I (1977) *The Indus Script: Texts, Concordance, and Tables* (Memoirs of Archaeological Survey of India, New Delhi, India).
14. Parpola A (1994) *Deciphering the Indus script* (Cambridge Univ Press, Cambridge, UK).
15. Yadav N, Vahia MN, Mahadevan I, Joglekar H (2008) A statistical approach for pattern search in Indus writing. *International Journal of Dravidian Linguistics* 37:39–52.
16. Yadav N, Vahia MN, Mahadevan I, Joglekar H (2008) Segmentation of Indus texts. *International Journal of Dravidian Linguistics* 37:53–72.
17. Rao RPN, et al. (2009) Entropic evidence for linguistic structure in the Indus script. *Science* 324:1165.
18. Schmitt AO, Herzel H (1997) Estimating the entropy of DNA sequences. *J Theor Biol* 188:369.
19. Farmer S, Sproat R, Witzel M (2004) The collapse of the Indus-script thesis: The myth of a literate Harappan civilization. *Electronic Journal of Vedic Studies* 11:19.
20. Koskenniemi K (1981) Syntactic methods in the study of the Indus script. *Studia Orientalia* 50:125–136.
21. Parpola A (2008) in *Airavati: Felicitation Volume in Honor of Iravatham Mahadevan* (www.varalaaru.com, Chennai, India), pp 111–131.
22. Wells BK (2009) PhD dissertation (Harvard University, Cambridge, MA).
23. Chen SF, Goodman J (1995) An emperical study of smoothing techniques for language modeling. Harvard University Computer Sci. Technical Report TR-10-98.
24. Kneser R, Ney H (1995) in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol 1, pp 181–184.
25. Viterbi AJ (1967) Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory* 13:260–269.