UC Merced
Proceedings of the Annual Meeting of the Cognitive Science Society

Title
Finding structure in logographic writing with library learning

Permalink
https://escholarship.org/uc/item/99c771pj

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 46(0)

Authors
Jiang, Guangyuan
Hofer, Matthias
Mao, Jiayuan
et al.

Publication Date
2024

Copyright Information
This work is made available under the terms of a Creative Commons Attribution License, available at https://creativecommons.org/licenses/by/4.0/

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

Finding structure in logographic writing with library learning

Guangyuan Jiang[1,3,⋆], **Matthias Hofer**[1], **Jiayuan Mao**[2], **Lionel Wong**[1], **Joshua B. Tenenbaum**[1,2], and **Roger P. Levy**[1]
[1]MIT BCS [2]MIT CSAIL [3]Peking University
⋆correspondence to jianggy@mit.edu

Abstract

One hallmark of human language is its combinatoriality—reusing a relatively small inventory of building blocks to create a far larger inventory of increasingly complex structures. In this paper, we explore the idea that combinatoriality in language reflects a human inductive bias toward representational efficiency in symbol systems. We develop a computational framework for discovering structure in a writing system. Built on top of state-of-the-art library learning and program synthesis techniques, our computational framework discovers known linguistic structures in the Chinese writing system and reveals how the system evolves towards simplification under pressures for representational efficiency. We demonstrate how a library learning approach, utilizing learned abstractions and compression, may help reveal the fundamental computational principles that underlie the creation of combinatorial structures in human cognition, and offer broader insights into the evolution of efficient communication systems.

Keywords: language learning; evolution; phonology; sketch understanding; bayesian modeling

Introduction

Human language is fundamentally combinatorial, reusing a relatively small inventory of building blocks to create a far larger inventory of increasingly complex structures. This structure is deeply rooted and evident in language's earliest written records, such as cuneiform and oracle bone scripts (Figure 1A,C). It manifests at multiple levels of structure, from phonemes to words and sentences, and across various linguistic modalities—spoken, signed, and written—illustrating a universal trait of human communication systems (Hockett & Hockett, 1960; Zuidema & De Boer, 2018).

In this paper, we explore the idea that combinatoriality in language reflects a human inductive bias toward representational efficiency in symbol systems. Research has recognized that the evolution and structure of these systems are profoundly influenced by an inductive bias towards representational efficiency (Kirby, Tamariz, Cornish, & Smith, 2015; Gibson et al., 2019), which combinatoriality offers (Kirby & Tamariz, 2022). Empirical support for this notion has come from laboratory experiments (Verhoef, Kirby, & De Boer, 2014; Little, Eryılmaz, & De Boer, 2017; Hofer & Levy, 2019), as well as qualitative analyses of emerging sign languages (Sandler, Aronoff, Meir, & Padden, 2011) and logographic writing systems, posited to have evolved towards increasing levels of simplicity (Sampson, 1985). Representational efficiency-based methods have been used for morphology (Goldsmith, 2001) and syntax (Carroll & Charniak, 1992; Kim, Dyer, & Rush, 2019), but structure discovery for these levels of linguistic representation has proven technically challenging.

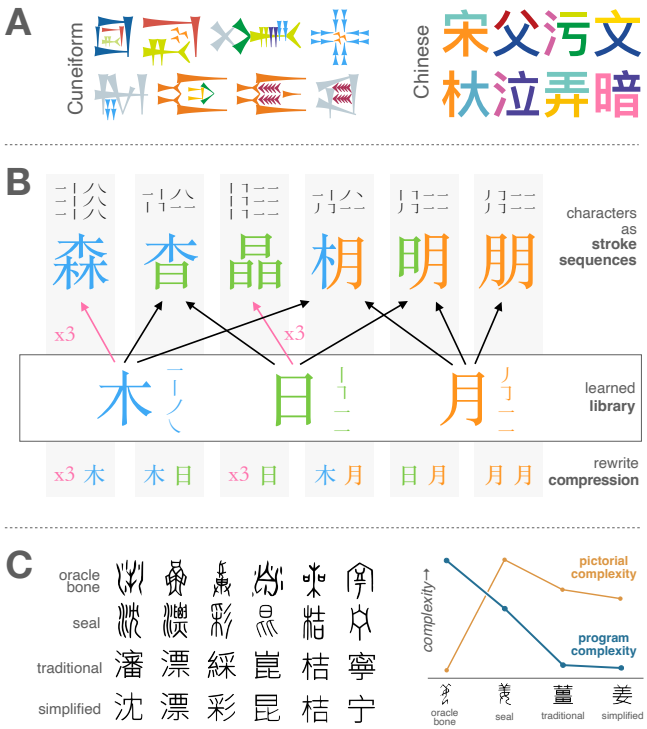

Figure 1: **An overview of our library learning model for writing systems**: **(A)** Parts are frequently reused (marked in the same color) within and across characters in the multiple logographic writing systems (e.g., Cuneiform, Chinese). **(B)** In our library learning model, we represent characters as stroke sequences. Learned library functions identify and represent reused parts (e.g., 木) and relations (e.g., x3, repeating three times), leading to program compression and the discovery of structures. **(C)** The model scales to study the multiple scripts in the Chinese writing system across time, revealing trends and adaptations in the use of radicals and other elements.

Here, we develop an efficiency-based structure discovery method for a complex logographic writing system, namely Chinese orthography, using the framework of **library learning** from the program synthesis literature (Ellis, Albright, Solar-Lezama, Tenenbaum, & O'Donnell, 2022; Bowers et al., 2023).

In L. K. Samuelson, S. L. Frank, M. Toneva, A. Mackey, & E. Hazeltine (Eds.), *Proceedings of the 46th Annual Conference of the Cognitive Science Society.* ©2024 The Author(s). This work is licensed under a Creative Commons Attribution 4.0 International License (CC BY).

The Chinese orthography presents a unique opportunity for studying combinatorial structure due to their long evolutionary history (of over 3,000 years, see, e.g., Kane, 2006) and the frequent reuse of graphical elements (hundreds of radicals) within individual characters and across the writing system. The library learning framework iteratively identifies recurring patterns and stores them in a library of abstractions (Figure 1B). The characters themselves are then redescribed by reference to these abstractions, increasing the overall concision of representation of the character inventory. Applying library learning to the Chinese writing system offers two related opportunities: to validate whether library learning recovers its radical combinatorial structure, and to use library learning to analyze the changes in the system over time.

Reflecting these two opportunities, our investigation consists of two parts. Part I develops a library learning model as a candidate hypothesis about human inductive biases for combinatorial structure. We validate the model by showing that it successfully rediscovers known linguistic structures, such as radicals, and characters' hierarchical decomposition, of the Chinese writing system in its simplified form.

Part II extends this analysis diachronically, investigating the evolution of Chinese scripts over several representative historical stages. Previous work in emerging sign languages has suggested the hypothesis that languages evolve towards simplification under pressures for representational efficiency (Motamedi, Schouwstra, Smith, Culbertson, & Kirby, 2019; Brentari & Goldin-Meadow, 2017). Here, the ancient Chinese writing system allows us to test this hypothesis at a larger scale. However, providing quantitative evidence presents significant challenges, especially when it comes to describing the graphical components. A recent study by Han, Kelly, Winters, and Kemp (2022), for instance, aimed to evaluate the evolution of the Chinese writing system using a simple measure of pictorial complexity that does not account for internal combinatorial structure within characters, and reported findings that ostensibly conflict with the expected trend toward the hypothesis and previous qualitative arguments in the Chinese writing system (Woon, 1987; Sampson, 1985; Wang, 1973; Geda, Zhao, & Baldauf, 2010). Our result challenges the pictorial complexity and supports earlier qualitative results, suggesting a trend towards simplification over time.

Our findings demonstrate how a library learning approach, utilizing learned abstractions and compression, may help reveal the fundamental computational principles that underlie the creation of combinatorial structures in human cognition, and offer broader insights into the evolution of efficient communication systems.

Part I: A library learning model for the Chinese writing system

In this section, we build a library learning-based computational model for writing systems and present studies on the most widely used script of the Chinese writing system: simplified Chinese. Our goal is to leverage library learning to reverse-engineer the core human inductive biases for combinatorial structure in language. We will show that our library learning model can accurately capture the core structural aspects of simplified Chinese, aligned with prior empirical findings and theories about hierarchical decomposition and radicals in the Chinese language.

Methods

We start by describing the computational model designed to learn library functions for the Chinese writing system. The high-level goal of the model is to find efficient representations (good compression in terms of minimal description length, or MDL) of all the characters in a writing system.

Defining a DSL. We start with defining a domain-specific language (DSL) for the Chinese writing system. It enables us to represent individual characters as sequences of primitive strokes. This approach is founded on three crucial insights: firstly, Chinese characters consist of strokes; second, these strokes can be classified into a finite number of basic types; and third, the spatial relationships between consecutive strokes are generally consistent.

Our base DSL $\mathcal{L}_{base}$ (visualized in Figure 2 left) consists of 33 stroke primitives {`T`, `N`, `H`, ..., `HZZP`} (six basic strokes + 27 turning strokes, as defined in Chinese language authoritative standard (Fu, Liu, Wang, & Wang, 2011)) and one helper primitive `list` indicating the beginning of the stokes sequence[1]. Each stroke primitive in the DSL represents a stroke type by its visual form and how it is written. For modern Chinese characters, we can uniquely encode every character to a LISP-style program p_{char} of the stroke sequence based on canonical stroke orderings (Fu, Wang, & He, 2020). For example, for the two-character word 认知, which means "cognition". The corresponding program encoding each characters are: $p_{认}$:= `(list D HZT SP N)` and $p_{知}$:= `(list P H H SP N S HZ H)`.

Objectives for library learning and the measure of program complexity. We formalize the process of learning an efficient representation of a writing system as a library learning problem. At a high level, the library learning algorithm identifies instances of reuse and abstraction from the dataset, facilitating efficient data compression. Our algorithm takes as input a set of (literal) character programs $P_{\mathcal{L}_{base}}(\mathcal{W}) = \{p_{认}, p_{知}, \dots\}$, with individual programs p represented with primitives in $\mathcal{L}_{base}$. The algorithm iteratively infers an optimal library of functions $\mathcal{L}_*$ that includes reusable components (abstractions) learned from the programs in addition to the base DSL. The programs set $P_{\mathcal{L}_{base}}(\mathcal{W})$ rewritten with the new library $\mathcal{L}$ yield $P_{\mathcal{L}}(\mathcal{W}) := \{\textsc{Rewrite}(p, \mathcal{L}) \mid p \in P_{\mathcal{L}_{base}}(\mathcal{W})\}$, where $\textsc{Rewrite}(p, \mathcal{L})$, informally, rewrites the program p in the most compact way using functions from the library $\mathcal{L}$.

[1] Following conventions in λ-calculus, `(list A B)` represents a list of two elements, and `((list A B) C)` is equivalent to a (flattened) list representation `(list A B C)`. We do not consider nested lists in this paper.

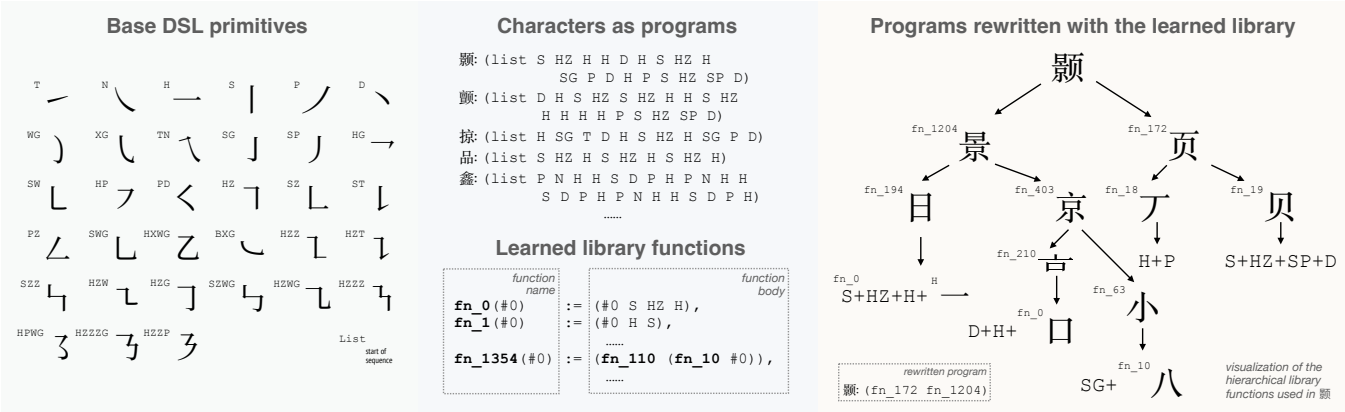

Figure 2: **(Left)**: Primitives used in the base DSL $\mathcal{L}_{base}$, including 33 stroke primitives and one `list` symbol. **(Middle)**: Character programs represented in $\mathcal{L}_{base}$ and the library functions learned by the model. **(Right)**: Visualization of an example character 颢's hierarchical decomposition discovered, represented as a tree of library functions.

How do we know which library is optimal? When would the compression algorithm stop? The optimal library $\mathcal{L}_*$ for all character programs $P_{\mathcal{L}_{base}}$ in a writing system should have a maximum reduction in description length after being rewritten with $\mathcal{L}_*$, while keeping the $\mathcal{L}_*$ size overhead small. Following previous program synthesis approaches in bootstrap library learning (i.e., DreamCoder (Ellis et al., 2022) and Stitch (Bowers et al., 2023)), we define the program (MDL) complexity of writing systems as the description length of the optimal library and the individual programs rewritten with the optimal library. This aligns with the concept of minimum description length (MDL): the description length of the shortest program that can generate the targeted data.

Formally, we define the description length $\text{DL}_{\mathcal{L}}(\mathcal{W})$ of a writing system $\mathcal{W}$ under a specific library $\mathcal{L}$, the description length of the library itself $\text{DL}(\mathcal{L})$, and the program (MDL) complexity of the writing system, represented as $C(\mathcal{W})$ as follows:

$$\begin{aligned}
\text{DL}_{\mathcal{L}}(\mathcal{W}) &= \overbrace{\sum_{p \in P_{\mathcal{L}_{base}}(\mathcal{W})} \text{DL}(\textsc{Rewrite}(p, \mathcal{L}))}^{\text{description length of the rewritten characters}} + \overbrace{\text{DL}(\mathcal{L})}^{\text{description length of the library}} \\
\text{DL}(\mathcal{L}) &= \sum_{\texttt{fn} \in \mathcal{L}} \text{DL}(\textsc{Body}(\texttt{fn})) \\
C(\mathcal{W}) &= \min_{\mathcal{L}} \text{DL}_{\mathcal{L}}(\mathcal{W})
\end{aligned} \tag{1}$$

Here, formally, $\textsc{Rewrite}(\cdot,\cdot)$ is an operation that seeks the shortest representation (in terms of DL) to rewrite the literal program using functions from the new library. $\textsc{Body}(\cdot)$ is the program string that defines a library function's body. $\text{DL}(\cdot)$ is a measure of the description length of a program string (which generally reflects the number of functions used); we leverage the exact λ-calculus-based measure of program size used in Stitch and DreamCoder, specifically the cost$(\cdot)$ defined in Bowers et al. (2023).

Iterative learning of the library. The library learning model's goal is to find the most compressive and concise $\mathcal{L} = \mathcal{L}_*$ for the writing system (i.e., minimizing $\text{DL}_{\mathcal{L}}(\mathcal{W})$).

Specifically, as we are running large-scale library learning on thousands of character programs, we leverage the Stitch algorithm (Bowers et al., 2023) for efficiently discovering library functions. Based on state-of-the-art program synthesis techniques, Stitch iteratively performs top-down searches to discover λ-abstractions (Alama & Korbmacher, 2023) as library functions, thereby compressing the programs. Formally, it gradually grows $\mathcal{L}_{base}$ to find $\mathcal{L}_*$ by minimizing $\text{DL}_{\mathcal{L}}(\mathcal{W})$ (as shown in eq. (1)). Compared to the DreamCoder compression algorithm, it is three orders of magnitude faster, making it tractable for us to examine the entire writing system.

Here, a minimal "writing system" containing three characters $\mathcal{W} = \{旦, 见, 日\}$ will be used as a working example. The initial programs $P_{\mathcal{L}_{base}}(\mathcal{W}) = \{p_{旦}, p_{见}, p_{日}\}$ are defined as follows:

$$\begin{aligned}
p_{旦} &:= \texttt{(list S HZ H H H)} \\
p_{见} &:= \texttt{(list S HZ SP SWG)} \\
p_{日} &:= \texttt{(list S HZ H H)}
\end{aligned} \tag{2}$$

In the first iteration, the algorithm discovers a (largest) reusable part across all three programs `(list S HZ)`, this part is thus added to the $\mathcal{L}_{base}$ as a library function, yielding a new library $\mathcal{L}_1 := \mathcal{L}_{base} \cup \{\texttt{fn_0}\}$, where $\textsc{Body}(\texttt{fn_0}) =$ `(list S HZ)`. The character programs in the set $P_{\mathcal{L}_1}(\mathcal{W})$, rewritten with the new library $\mathcal{L}_1$, are as follows:

$$\begin{aligned}
\textsc{Rewrite}(p_{旦}, \mathcal{L}_1) &= \texttt{(fn_0 H H H)} \\
\textsc{Rewrite}(p_{见}, \mathcal{L}_1) &= \texttt{(fn_0 SP SWG)} \\
\textsc{Rewrite}(p_{日}, \mathcal{L}_1) &= \texttt{(fn_0 H H)}
\end{aligned} \tag{3}$$

In the second iteration, the algorithm discovers `(fn_0 H H)` that is reused and adds it to the library, resulting $\mathcal{L}_2 := \mathcal{L}_1 \cup \{\texttt{fn_1}\}$, where $\textsc{Body}(\texttt{fn_1}) =$ `(fn_0 H H)`.

$$\begin{aligned}
\textsc{Rewrite}(p_{旦}, \mathcal{L}_2) &= \texttt{(fn_1 H)} \\
\textsc{Rewrite}(p_{见}, \mathcal{L}_2) &= \texttt{(fn_0 SP SWG)} \\
\textsc{Rewrite}(p_{日}, \mathcal{L}_2) &= \texttt{fn_1}
\end{aligned} \tag{4}$$

In this example, $\mathcal{L}_2$ is the optimal library ($\mathcal{L}_*$) for this 旦见

Figure 3: **Visualization of the aligned MoE radical–library function pairs.** 201 radicals from the MoE radicals set are colored in blue, corresponding library functions are colored in gray on the top left (we omit the `fn_` prefix for brevity). Our model discovered most of the expert-defined radicals (93.0%).

日 writing system. It is also a hierarchical library, as `fn_1` is defined based on other library functions `fn_0`. The program strings in $P_{\mathcal{L}_*}(\mathcal{W})$, rewritten with the optimal library $\mathcal{L}_*$, lead to a more concise (compressed) description of the writing system (eq. (4)).

Data. A total of 6,596 simplified Chinese characters were collected, including their canonical stroke decompositions and orderings, from the Han character library (Cjklib Developers, 2012). This dataset covers a majority of the official set of simplified Chinese characters (6,596 out of 6,763) specified in the China National Institute of Standardization (1980) standard.

Results

Learned library captures reuse and hierarchical decomposition in the Chinese writing system. Inspired by previous work on using library learning to discover geometric shape patterns (Sablé-Meyer, Ellis, Tenenbaum, & Dehaene, 2022) and object structures (Wong et al., 2022), we applied our model to analyze the library functions it learned from 6,596 simplified Chinese characters.

Our model discovered a set of hierarchical library functions for compressing the writing system. In addition to the 34 primitives defined in the base DSL $\mathcal{L}_{base}$, 1,805 learned library functions were found in the optimal library $\mathcal{L}_*$, resulting in a library size of $|\mathcal{L}_*| = 1,839$. Among all the new library functions learned, 1,717 (95.1%) were hierarchically defined, meaning that they were not only composed of base DSL primitives but also incorporated other newly learned library functions (see Table 1 for examples).

This hierarchical organization facilitated significant compression of the writing system, achieving an overall compression rate of 4.16×, defined by $\mathrm{DL}_{\mathcal{L}_{base}}(\mathcal{W})/\mathrm{DL}_{\mathcal{L}_*}(\mathcal{W})$. In terms of individual characters, the program description length was drastically reduced. On average, the number of functions required to encode an individual character was 1/5.63 of the original amount. As shown in the right of Figure 2, 颢 (pronounced as: *hào*, meaning *bright sunlight*) is represented by a program containing only two functions in $\mathcal{L}_*$ compared to the 19 functions required in $\mathcal{L}_{base}$ before compression. Delving into the two functions learned to encode 颢, `fn_1204` resembles 景 and `fn_172` 页, which are both commonly used in several other characters (e.g., 惊, 影). The `fn_1204` (景) and `fn_172` (页) were also hierarchically defined on other functions (i.e., 日, 京, 贝). This contributes to an explicit hierarchical representation that captures reuse and structure at a system level for simplified Chinese.

The example of 颢 intuitively demonstrates that our model's learned hierarchical decomposition closely resembles the way humans cognitively break down structures. To assess the validity of these decompositions, we compared the model's predictions against a gold standard from the Han character library. The comparison, based on 3,052 characters, revealed that the model's hierarchical representations achieved an overall recall rate of 76.3% and an F_1 score of 61.6 over the spans of the parsed trees (results are shown in Table 2). It is important to emphasize the significance of the recall (true positive) rate in this context, as our model prioritizes maximum compression, potentially uncovering more fine-grained patterns of reuse than the ones typically recognized by humans. These findings suggest that our model effectively captures the intrinsic hierarchical organization of simplified Chinese characters.

Library function discovered	#Uses (percentile)	Semantic	Example usage
`fn_0(#0) := (#0 S HZ H)`	3342 (100%)	口	口品扣亩
`fn_23(#0) := (#0 SP N)`	418 (99%)	人	人仄灭认
`fn_48(#0) := (#0 H SG)`	220 (98%)	夫	撵潜窥肤
`fn_105(#0) := (fn_23 (fn_2 #0))`	83 (95%)	寸	夺守尊将
`fn_209(#0) := (fn_66 (fn_0 #0))`	38 (90%)	兄	兑兕党况
`fn_415(#0) := (fn_20 (fn_3 #0))`	15 (80%)	喜	僖嘻嬉喜
`fn_776() := (fn_128 fn_274)`	6 (60%)	比	皆比毕毖
`fn_1624() := (fn_1233 T)`	2 (20%)	禺	愚遇

Table 1: **Examples of learned library functions**, ordered according to their frequency of usage. `(#·)` indicates the parameter a function takes in.

Model	F_1	Precision	Recall	Exact match
Library learning	61.6	51.7	76.3	6.9
Baselines				
– Balanced binary tree	34.4	26.5	48.6	2.2
– Random binary tree	28.5	22.0	40.3	1.1
– Left-branching tree	30.8	23.8	43.6	0.5
– Right-branching tree	36.0	27.8	50.9	0.4

Table 2: **Quantitative evaluation of the learned structure.** Similar to the evaluations used in constituency parsing (Black et al., 1991), we report F_1 scores, precisions, recalls, and exact match (%) rates with respect to a gold standard of Chinese character decomposition.

Library functions resemble expert-defined radicals. Prior research has shown that computational models of natural languages as hierarchical programs resemble linguists' theories in morphophonology (Goldsmith, 2001; Ellis et al., 2022). Can library learning models similarly uncover the structural theories underlying the Chinese language? In this section, we analyze

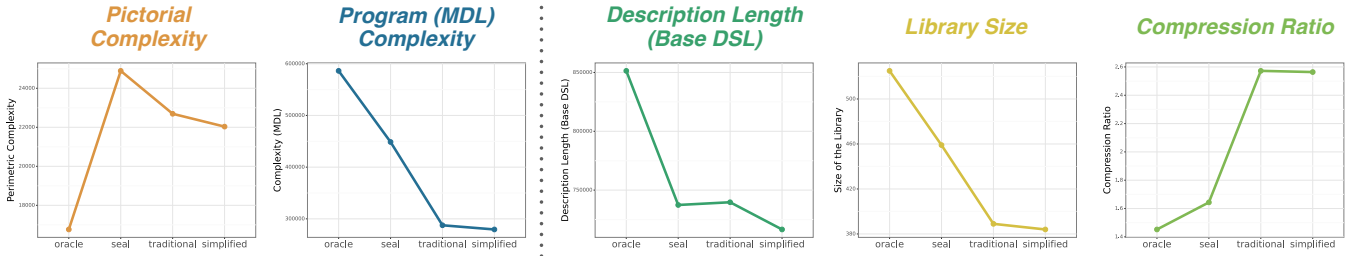

Figure 4: **Changes in the quantifiable metrics over time.** We visualize **Left:** pictorial complexity (following Han et al. (2022)), program (MDL) complexity calculated by our model $C(\mathcal{W})$; **Right:** description length under the base library $\mathrm{DL}_{\mathcal{L}_{base}}(\mathcal{W})$, learned library size $|\mathcal{L}_*|$, and compression ratio $\mathrm{DL}_{\mathcal{L}_{base}}(\mathcal{W})/\mathrm{DL}_{\mathcal{L}_*}(\mathcal{W})$ for the four scripts (oracle bone, seal, traditional, and simplified) respectively.

the library functions learned by the model and compare them to expert-defined radicals in simplified Chinese.

Radicals are the graphical components that frequently occur in Chinese characters and have been used for indexing characters in dictionaries for almost two thousand years. Several seminal books have defined radical sets throughout history, notably, the Shuowen Jiezi Radicals (說文部首, (Duan, 1821)), the Kangxi Radicals (康熙部首, (Hanlin Academy, 1885)), and the Table of Indexing Chinese Character Components (汉字部首表, (Zhang & Wang, 2009)) developed by the Ministry of Education (MoE) in China. These radical sets were designed for different writing systems and varied in size. In the scope of this work, we leverage the MoE radicals set, as it is the only well-recognized radicals definition for the simplified Chinese writing system.

The MoE radicals set contains 201 radicals: all radicals are parts of Chinese characters, and every character in Chinese has at least one identifiable radical from the MoE radicals set. Intuitively, these radicals can be interpreted as the most frequent co-occurring components discovered by experts in the simplified Chinese writing system and are designed to be easy to identify. Hence, our hypothesis is that our computational model would be able to rediscover most MoE radicals from the character programs.

By comparing the library functions learned by the model and the MoE radicals set, we found that our model discovered 187 (93.0%) radicals in the MoE set. We illustrate the aligned MoE radical and library function pairs in Figure 3.

Part II: Complexity analysis across time

Our results so far indicate that library learning effectively discovers structural patterns that link character representational efficiency and compression to their widely established prescriptive decompositions. This alignment suggests that the model may accurately reflect human inductive biases toward combinatorial structure. If this premise holds, the model should reveal a gradual simplification as systems adapt to these biases in cultural evolution (Smith, Kirby, & Brighton, 2003).

To test this prediction, the second part of our study presents a diachronic analysis of the Chinese writing system. To our knowledge, the only attempt to date to quantify script complexity of the Chinese at different historic stages comes from Han et al. (2022), which used a simple measure of pictorial complexity. Contrary to expectations, their findings did not support a trend towards simplification (Figure 4 left). However, the reliance on pictorial complexity, which fails to take into account systematic reuse across and within characters, could inadvertently inflate perceived character complexity. By contrast, our model assesses writing system complexity on a holistic level rather than at the individual character level, capitalizing on the advantages of systematic compression through part reuse, and may thus provide a more accurate picture of writing system complexity.

Methods and data

Data used for oracle bone, seal, traditional, and simplified. To test whether the Chinese writing system has indeed become simpler when taking combinatorial reuse into account, we analyze Chinese scripts at four representative historical stages (Kane, 2006): oracle bone (∼1300 B.C.E), seal (∼200 B.C.E), traditional (∼5th century–present), and simplified (1956–present). 754 aligned characters were retrieved in each of these four stages of the Chinese writing system. For seal scripts, we used the correspondence data from the Unicode Seal Script Encoding Project (TCA & China, 2022). For oracle bone scripts, the 殷墟甲骨文字詞表 (*Yin Ruins Oracle Bone Script Lexicon*, N. Chen, 2010, 2012) was utilized for both the correspondence data and character shapes.

Next, we obtained the program representations for all four scripts. For traditional and simplified Chinese, we adopted stroke decompositions from the Han character library. Since there were no off-the-shelf stroke decompositions for seal and oracle bone characters. Thus, we manually labeled all strokes with a graphics tablet. Next, following Lake, Salakhutdinov, Gross, and Tenenbaum (2011) on defining stroke library, the recorded stroke trajectories were fitted into 2D Cubic Bézier curves and discretized the strokes into 33 primitives with K-means clustering, resulting in DSLs of the same size across the four scripts. Finally, we apply the same compression objectives and the iterative library learning algorithm as defined in Part I.

Data used for traditional and simplified comparison. 3,762 traditional-simplified Chinese character pairs were collected from 標準字與簡化字對照手冊 (*Comparison Manual of Traditional and Simplified Chinese Characters,* Ministry of Education, ROC, 2011) and the Han character library.

Results

The Chinese writing system has been simplified over time. We applied the library learning model to the four scripts (oracle bone, seal, traditional, and simplified) of the Chinese writing system, respectively, and compared the program complexity metrics as defined in eq. (1). The results for the writing system's complexity $C(\mathcal{W})$, literal description length $\mathrm{DL}_{\mathcal{L}_{base}}$ (written in the base DSL without library learning), learned library size $|\mathcal{L}_*|$, and compression rate $\mathrm{DL}_{\mathcal{L}_{base}}(\mathcal{W})/\mathrm{DL}_{\mathcal{L}_*}(\mathcal{W})$ are shown in Figure 4.

We observed a non-monotonic change in the literal description lengths with an overall trend of decreasing. The literal description length is practically a measurement of the total number of strokes used in the writing system. Oracle bone scripts were observed the largest literal description lengths, and there was an increase from the transition from seal script to traditional Chinese.

To validate the simplification hypothesis we proposed earlier in the paper, we analyzed the complexity change over time. As shown in Figure 4 left, on the contrary to pictorial complexity measurement based on perimetric calculation used in drawing Han et al. (2022)'s conclusion (a complexity ranking of seal > traditional > simplified > oracle bone), we generally found that the writing system's program complexity $C(\mathcal{W})$ has shown a monotonic decrease across time (oracle bone > seal > traditional > simplified), confirming earlier empirical arguments (Woon, 1987; Sampson, 1985; Wang, 1973; Geda et al., 2010).

Simplified Chinese is simpler but less systematic compared to traditional Chinese. The "simplification" from traditional to simplified Chinese was carried out as a deliberate process by the People's Republic of China between 1955 and 1986. Initially designed to ease the difficulty of learning and improve literacy by reducing strokes in individual characters (P. Chen, 1999; Rohsenow, 2004), this process may have disrupted established systematicity and lead to a loss of established semantic-phonetic and graphic patterns (Handel, 2013; Zhao & Baldauf, 2011). It has been suggested that this may have contributed to recognition and learning behavior differences (Liu, Chuk, Yeh, & Hsiao, 2016; McBride-Chang, Chow, Zhong, Burgess, & Hayward, 2005).

Here, our model allows us to directly assess the systematicity of the traditional and simplified Chinese scripts. From the MDL perspective, systematic-patterned data should be more compressive. Therefore, the compression ratio can be viewed as a proxy for the systematicity of a script. By applying our model to the 3,762 aligned characters on the two scripts separately, we found (Figure 5) that although traditional has a noticeably larger description length under the base DSL than simplified (+16%), the gap was narrowed after compression (+4%). Further analyzing the compression ratio, we observed that the traditional Chinese yielded a higher compression rate than the simplified Chinese, suggesting the simplification process from traditional to simplified Chinese did break part of the systematicity from a computational standpoint.

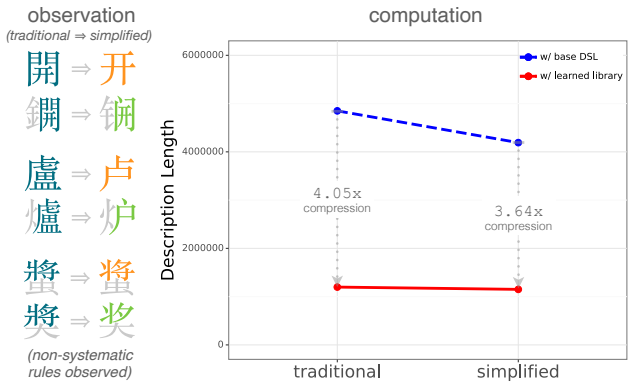

Figure 5: **Comparison of the compression ratio between traditional and simplified Chinese.** The traditional Chinese script is more compressible than simplified Chinese on the 3,762 aligned characters at a larger scale.

Discussion

In this work, we develop a library learning-based computational model, positing it as a framework for understanding the inductive biases behind the emergence and evolution of combinatorial structures in human language. The model is centered on the idea that combinatoriality develops from a MDL perspective of representational efficiency, both by discovering inventories of reusable parts and by compressing the language using those parts. This validity of this approach is demonstrated in the first part, where applying our model to the Chinese writing system uncovers known linguistic primitives and character decompositions, that align with intuitive human understandings of these languages. Results from the second part reveal that these inductive biases, when applied over time, lead to the development of increasingly simple and efficient systems, as demonstrated in an analysis of the Chinese writing system across several stages of historic development.

There are multiple challenges to refining this computational framework. For instance, we are using canonical decompositions of modern Chinese scripts (i.e., traditional and simplified) but manually-parsed sequences for ancient scripts (i.e., oracle bone and seal), which could lead to less systematic representations of ancient scripts. Meanwhile, the stroke-based representation may not faithfully recover spatial layouts. These issues can be addressed by using unsupervised image parsing algorithms and including spatial templates (Lake, Salakhutdinov, & Tenenbaum, 2015; Hu, Wu, & Zhu, 2011).

In future work, we intend to further scale our model to analyze the systematic structure to not only forms but a broader range of form-meaning mappings. We hope our work can provide insights into how to build a computational model to reverse-engineer how compression, efficiency, and structure shape human languages (Kirby et al., 2015; Gibson et al., 2019; Tamariz & Kirby, 2015; Zaslavsky, Kemp, Regier, & Tishby, 2018). More broadly, how large-scale library learning can contribute to theories of the evolution of efficient communicative systems and modeling human cognitive representations of abstractions.

Acknowledgments

We thank Yixin Zhu (Peking University), Robert Hawkins (UW-Madison), Judith Fan (Stanford), and Kevin Ellis (Cornell) for early discussions on this topic. Maddy Bowers, Alex Lew, Peng Qian (MIT), Yi Bai (Zi-tools) for many valuable discussions and contributions. We thank anonymous reviewers for their valuable comments.

MH, JM, LW, and JBT are supported by MIT Quest for Intelligence, AFOSR Grant #FA9550-19-1-0269, the MIT-IBM Watson AI Lab, ONR Science of AI, and Simons Center for the Social Brain.

References

Alama, J., & Korbmacher, J. (2023). The Lambda Calculus. In E. N. Zalta & U. Nodelman (Eds.), *The Stanford encyclopedia of philosophy* (Winter 2023 ed.). Metaphysics Research Lab, Stanford University. https://plato.stanford.edu/archives/win2023/entries/lambda-calculus/.

Black, E., Abney, S., Flickinger, D., Gdaniec, C., Grishman, R., Harrison, P., ... others (1991). A procedure for quantitatively comparing the syntactic coverage of english grammars. In *Speech and natural language: Proceedings of a workshop held at pacific grove, california, february 19-22, 1991*.

Bowers, M., Olausson, T. X., Wong, L., Grand, G., Tenenbaum, J. B., Ellis, K., & Solar-Lezama, A. (2023). Top-down synthesis for library learning. *Proceedings of the ACM on Programming Languages*, *7*(POPL), 1182–1213.

Brentari, D., & Goldin-Meadow, S. (2017). Language emergence. *Annual Review of Linguistics*, *3*, 363–388.

Carroll, G., & Charniak, E. (1992). Two experiments on learning probabilistic dependency grammars from corpora. In *Working notes of the aaai workshop statistically-based nlp techniques, 1992* (pp. 1–13).

Chen, N. (2010). *Yinxu jiaguwen moshi quanbian diyi juan [The complete compilation of rubbings and interpretations of oracle bone inscriptions from the yin ruins, volume one]*. Beijing: Xianzhuang shuju.

Chen, N. (2012). *Yinxu jiaguwen zicibiao [Yin ruins oracle bone script lexicon]*. (available at: https://www.xianqin.org/paper/Chennianfu_OBI_Table_20120409.pdf (Dec. 2023))

Chen, P. (1999). *Modern Chinese: History and sociolinguistics*. 110 Midland Avenue, Port Chester, NY 10573-4390: Cambridge University Press.

China National Institute of Standardization. (1980). *GB 2312-1980: Code of Chinese graphic character set for information interchange–primary set* (Chinese National Standard No. GB 2312-1980). Beijing, China: Standards Press of China.

Cjklib Developers. (2012). *cjklib - Han character library*. https://cjklib.readthedocs.io/en/latest/.

Duan, Y. (1821). *Shuowen jiezi zhu [Explanation to Shuowen Jiezi]*. China: Qi Ye Yan Xiang Tang.

Ellis, K., Albright, A., Solar-Lezama, A., Tenenbaum, J. B., & O'Donnell, T. J. (2022). Synthesizing theories of human language with bayesian program induction. *Nature Communications*, *13*(1), 5024.

Fu, Y., Liu, L., Wang, C., & Wang, D. (2011). *Chinese character turning stroke standard of GB 13000.1 character set* (Chinese National Standard No. GF 2001-2001). Beijing, China: Minisitry of Education of PRC and State Language Commission.

Fu, Y., Wang, M., & He, R. (2020). *Storke orders of the commonly used standard Chinese characters* (Chinese National Standard No. GF 0023-2020). Beijing, China: Minisitry of Education of PRC and State Language Commission.

Geda, K., Zhao, S., & Baldauf, R. B. (2010). Planning Chinese characters: Reaction, evolution or revolution? *Language in Society*, *39*(2), 291-292.

Gibson, E., Futrell, R., Piantadosi, S. P., Dautriche, I., Mahowald, K., Bergen, L., & Levy, R. (2019). How efficiency shapes human language. *Trends in Cognitive Sciences*, *23*(5), 389–407.

Goldsmith, J. (2001). Unsupervised learning of the morphology of a natural language. *Computational linguistics*, *27*(2), 153–198.

Han, S. J., Kelly, P., Winters, J., & Kemp, C. (2022). Simplification is not dominant in the evolution of Chinese characters. *Open Mind*, *6*, 264–279.

Handel, Z. (2013). Can a logographic script be simplified? lessons from the 20th century Chinese writing reform informed by recent psycholinguistic research. *Scripta*, *5*, 21–66.

Hanlin Academy. (1885). *Kangxi zidian [Kangxi dictionary]*. China: Tongwen Shuju.

Hockett, C. F., & Hockett, C. D. (1960). The origin of speech. *Scientific American*, *203*(3), 88–97.

Hofer, M., & Levy, R. P. (2019). Iconicity and structure in the emergence of combinatoriality. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 41).

Hu, W., Wu, Y. N., & Zhu, S.-C. (2011). Image representation by active curves. In *2011 international conference on computer vision* (pp. 1808–1815).

Kane, D. (2006). *The Chinese language: Its history and current usage*. Tuttle Publishing.

Kim, Y., Dyer, C., & Rush, A. M. (2019). Compound probabilistic context-free grammars for grammar induction. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 2369–2385).

Kirby, S., & Tamariz, M. (2022). Cumulative cultural evolution, population structure and the origin of combinatoriality in human language. *Philosophical Transactions of the Royal Society B*, *377*(1843), 20200319.

Kirby, S., Tamariz, M., Cornish, H., & Smith, K. (2015). Compression and communication in the cultural evolution of linguistic structure. *Cognition*, *141*, 87–102.

Lake, B. M., Salakhutdinov, R., Gross, J., & Tenenbaum, J. (2011). One shot learning of simple visual concepts. In

Proceedings of the annual meeting of the cognitive science society (Vol. 33).

Lake, B. M., Salakhutdinov, R., & Tenenbaum, J. B. (2015). Human-level concept learning through probabilistic program induction. *Science*, *350*(6266), 1332–1338.

Little, H., Eryılmaz, K., & De Boer, B. (2017). Signal dimensionality and the emergence of combinatorial structure. *Cognition*, *168*, 1–15.

Liu, T., Chuk, T. Y., Yeh, S.-L., & Hsiao, J. H. (2016). Transfer of perceptual expertise: The case of simplified and traditional Chinese character recognition. *Cognitive Science*, *40*(8), 1941–1968.

McBride-Chang, C., Chow, B. W., Zhong, Y., Burgess, S., & Hayward, W. G. (2005). Chinese character acquisition and visual skills in two Chinese scripts. *Reading and Writing*, *18*, 99–128.

Ministry of Education, ROC. (2011). *Biaozhun zi yu jianhua zi duizhao shouce [Standard characters and simplified characters: A comparison manual]*. Taipei: Ministry of Education, Republic of China (Taiwan).

Motamedi, Y., Schouwstra, M., Smith, K., Culbertson, J., & Kirby, S. (2019). Evolving artificial sign languages in the lab: From improvised gesture to systematic sign. *Cognition*, *192*, 103964.

Rohsenow, J. S. (2004). Fifty years of script and written language reform in the p.r.c. In M. Zhou & H. Sun (Eds.), *Language policy in the people's republic of china: Theory and practice since 1949* (pp. 21–43). Dordrecht: Springer Netherlands.

Sablé-Meyer, M., Ellis, K., Tenenbaum, J., & Dehaene, S. (2022). A language of thought for the mental representation of geometric shapes. *Cognitive Psychology*, *139*, 101527.

Sampson, G. (1985). Writing systems. *London, UK: H utchinson*.

Sandler, W., Aronoff, M., Meir, I., & Padden, C. (2011). The gradual emergence of phonological form in a new language. *Natural language & linguistic theory*, *29*, 503–543.

Smith, K., Kirby, S., & Brighton, H. (2003). Iterated learning: A framework for the emergence of language. *Artificial life*, *9*(4), 371–386.

Tamariz, M., & Kirby, S. (2015). Culture: copying, compression, and conventionality. *Cognitive Science*, *39*(1), 171–183.

TCA, & China. (2022). *Unicode WG2 Doc N5191: THX Shuowen properties table.* https://www.unicode.org/wg2/docs/n5191-THX-Properties-Table.pdf.

Verhoef, T., Kirby, S., & De Boer, B. (2014). Emergence of combinatorial structure and economy through iterated learning with continuous acoustic signals. *Journal of Phonetics*, *43*, 57–68.

Wang, W. S. (1973). The Chinese language. *Scientific American*, *228*(2), 50–63.

Wong, C., McCarthy, W. P., Grand, G., Friedman, Y., Tenenbaum, J., Andreas, J., ... Fan, J. E. (2022). Identifying concept libraries from language about object structure. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 44).

Woon, W. L. (1987). *Chinese writing: Its origin and evolution* (Vol. 2). University of East Asia.

Zaslavsky, N., Kemp, C., Regier, T., & Tishby, N. (2018). Efficient compression in color naming and its evolution. *Proceedings of the National Academy of Sciences*, *115*(31), 7937–7942.

Zhang, S., & Wang, M. (2009). *The table of indexing Chinese character commponent* (Chinese National Standard No. GF 0011-2009). Beijing, China: Minisitry of Education of PRC and State Language Commission.

Zhao, S., & Baldauf, R. B. (2011). Simplifying Chinese characters: Not a simple matter. In J. Fishman & O. Garcia (Eds.), *Handbook of language and ethnic identity volume 2: The success-failure continuum in language and ethnic identity efforts* (pp. 168–179). Oxford University Press.

Zuidema, W., & De Boer, B. (2018). The evolution of combinatorial structure in language. *Current Opinion in Behavioral Sciences*, *21*, 138–144.