Slide 1

Potential Distinguishing Characteristics of Human Aural Pattern Recognition

James Rogers[†] and Marc D. Hauser[‡]

[†] Dept. of Computer Science
Earlham College
this work completed, in part, while at the
Radcliffe Institute for Advanced Study

[‡]Depts. of Psychology, Oganismic & Evolutionary
Biology and Biological Anthropolgy
Harvard University

Slide 2

We hypothesize that FLN only includes recursion and is the only uniquely human component of the faculty of language.

Hauser, Chomsky and Fitch, *Nature*, v. 298, 2002.

Slide 3

The Comparative Approach to Language Evolution

- Shared *vs.* unique
  - Homologous *vs.* analogous
- Gradual *vs.* saltational
- Continuity *vs.* exaption

Slide 4

Three Hypotheses

1. FLB is strictly homologous to animal communication
2. FLB is a derived, uniquely human adaptation for language
3. Only FLN is uniquely human

Empirical support for the comparative method

Slide 5

- Across species
- Domains other than (just) communication
- Spontaneous and trained behaviors

Contrasting $(AB)^n$ with A^nB^n

Slide 6

- Finite State *vs.* Context-Free
- $\{(\textbf{ding dong})^n\}$ *vs.* $\{\textbf{people}^n\ \textbf{left}^n\}$
- *vs.*
  $\{\textbf{those people who were left}(\textbf{by people who were left})^n\textbf{left}\}$
- *vs.*
  $\{\textbf{those people who were left}(\textbf{by people who were left})^{2n}\textbf{left}\}$

Slide 7

Dual Characterizations of Classes of Patterns

Descriptive characterizations

- Nature of the information about strings
- Independent of mechanism
- Support conclusions about abstract properties of mechanisms

Grammar- and automata-theoretic characterizations

- Concrete algorithm s
- Support reasoning about the structure of stringsets
- Guide experimental design

Slide 8

Strictly Local Stringsets

2-factors: $\mathcal{G}_{(AB)^n} = \{\rtimes A, AB, BA, B\ltimes\}$

$$\rtimes ABABAB \ltimes \qquad \rtimes A\overset{\frown}{BB}AB \ltimes$$

Strictly k-Local Definitions

$$\mathcal{G} \subseteq F_k(\{\rtimes\} \cdot \Sigma^* \cdot \{\ltimes\})$$

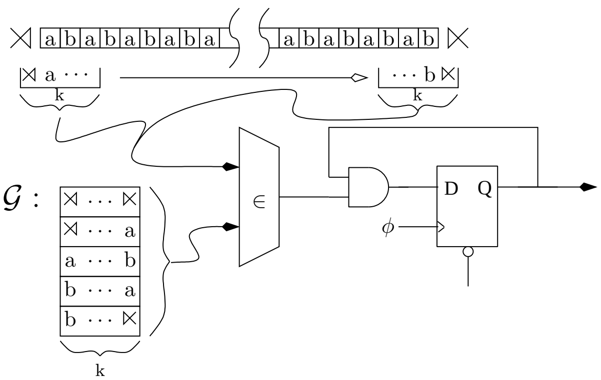
$$w \models \mathcal{G} \overset{\text{def}}{\Longleftrightarrow} F_k(\rtimes \cdot w \cdot \ltimes) \subseteq \mathcal{G}$$

$$L(\mathcal{G}) \overset{\text{def}}{=} \{w \mid w \models \mathcal{G}\}$$

Membership in an SL_k stringset depends only on the individual k-factors which do and do not occur in the string.

Scanners

Slide 9



Recognizing an SL_k stringset requires only remembering the k most recently encountered symbols.

Character of Strictly 2-Local Sets

Slide 10

Theorem (Suffix Substitution Closure):
A stringset L is strictly 2-local iff whenever there is a word x and strings w, y, v, and z, such that

$$w \cdot x \cdot y \in L$$
$$v \cdot x \cdot z \in L$$

then it will also be the case that

$$w \cdot x \cdot z \in L$$

Example:

The dog	·	likes	·	the biscuit	$\in L$
Alice	·	likes	·	Bob	$\in L$
The dog	·	likes	·	Bob	$\in L$

Probing the SL Boundary

Slide 11

$$\text{Some-B} \stackrel{\text{def}}{=} \{w \in \{A,B\}^* \mid |w|_B \geq 1\}$$

$$\begin{array}{ccccccl} A\ldots A & \cdot & \underbrace{A\ldots A}_{k-1} & \cdot & BA\ldots A & & \in \text{Some-B} \\ A\ldots AB & \cdot & \underbrace{A\ldots A}_{k-1} & \cdot & A\ldots A & & \in \text{Some-B} \\ \hline A\ldots A & \cdot & \underbrace{A\ldots A}_{k-1} & \cdot & A\ldots A & & \notin \text{Some-B} \end{array}$$

		In	Out
SL	$(AB)^n$	$(AB)^{i+j+1}$	$(AB)^i AA (AB)^j$
	$A^m B^n$	$A^{i+k} B^{j+l}$	$A^i B^j A^k B^l$
non-SL	Some-B	$A^i B A^j$	A^{i+j+1}

Locally k-Testable Stringsets

Slide 12

Some-B: $\neg(\neg \rtimes B \wedge \neg AB) \quad (= \rtimes B \vee AB)$

k-Expressions

$$\begin{array}{ccccc} f \in F_k(\rtimes \cdot \Sigma^* \ltimes) & & w \models f & \stackrel{\text{def}}{\iff} & f \in F_k(\rtimes \cdot w \cdot \ltimes) \\ \varphi \wedge \psi & & w \models \varphi \wedge \psi & \stackrel{\text{def}}{\iff} & w \models \varphi \text{ and } w \models \psi \\ \neg\varphi & & w \models \neg\varphi & \stackrel{\text{def}}{\iff} & w \not\models \varphi \end{array}$$

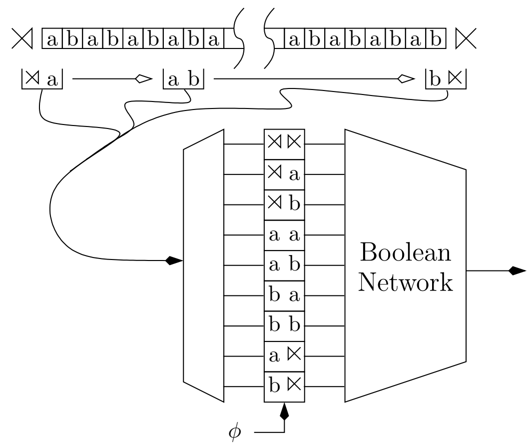
Locally k-Testable Languages (LT_k):

$$L(\varphi) \stackrel{\text{def}}{=} \{w \mid w \models \varphi\}$$

Membership in an LT_k stringset depends only on the set of k-Factors which occur in the string.

LT Automata

Slide 13



Recognizing an LT_k stringset requires only remembering which k-factors occur in the string.

Character of Locally Testable Sets

Slide 14

Theorem (k-Test Invariance):
A stringset L is Locally Testable iff

there is some k such that, for all strings x and y,

if $\rtimes \cdot x \cdot \ltimes$ and $\rtimes \cdot y \cdot \ltimes$ have exactly the same set of k-factors

then either both x and y are members of L or neither is.

Probing the LT Boundary

Slide 15

$$\text{Some-B} = \{w \in \{A, B\}^* \mid w \models \rtimes B \vee AB\} \quad (\in \text{LT}_2)$$

$$\text{One-B} \stackrel{\text{def}}{=} \{w \in \{A, B\}^* \mid |w|_B = 1\} \notin \text{LT}$$

$$A^k B A^k \in \text{One-B} \qquad\qquad A^k B A^k B A^k \notin \text{One-B}$$

$$F_k(\rtimes A^k B A^k \ltimes) = F_k(\rtimes A^k B A^k B A^k \ltimes)$$

		In	Out
LT	Some-B	$A^i B A^j$	A^{i+j+1}
non-LT	One-B	$A^i B A^{j+k+1}$	$A^i B A^j B A^k$

FO(+1) (Strings)

Slide 16

$$AABA \quad \models \quad (\forall x)[A(x) \vee B(x)] \wedge (\exists x)[B(x)]$$

$\langle \mathcal{D}, \triangleleft, P_\sigma \rangle_{\sigma \in \Sigma}$

$$AABA = \Big\langle \{0, 1, 2, 3\},\ \{\langle i, i+1 \rangle \mid 0 \leq i < 3\},\ \{0, 1, 3\}_A,\ \{2\}_B \Big\rangle$$

First-order Quantification (over positions in the strings)

$$x \triangleleft y \qquad w, [x \mapsto i, y \mapsto j] \models x \triangleleft y \stackrel{\text{def}}{\iff} j = i + 1$$

$$P_\sigma(x) \qquad w, [x \mapsto i] \models P_\sigma(x) \stackrel{\text{def}}{\iff} i \in P_\sigma$$

$$\varphi \wedge \psi \qquad \vdots$$

$$\neg\varphi \qquad \vdots$$

$$(\exists x)[\varphi(x)] \qquad w, s \models (\exists x)[\varphi(x)] \stackrel{\text{def}}{\iff} w, s[x \mapsto i] \models \varphi(x) \text{ for some } i \in \mathcal{D}$$

FO(+1)-Definable Stringsets: $L(\varphi) \stackrel{\text{def}}{=} \{w \mid w \models \varphi\}$.

Slide 17

Character of the FO(+1) Definable Stringsets

Definition 1 (Locally Threshold Testable) *A set L is* Locally Threshold Testable *(LTT) iff there is some k and t such that, for all $w, v \in \Sigma^*$:*

if for all $f \in F_k(\rtimes \cdot w \cdot \ltimes) \cup F_k(\rtimes \cdot v \cdot \ltimes)$

either $|w|_f = |v|_f$ or both $|w|_f \geq t$ and $|v|_f \geq t$,

then $w \in L \iff v \in L$.

Theorem 1 (Thomas) *A set of strings is First-order definable over $\langle \mathcal{D}, \triangleleft, P_\sigma \rangle_{\sigma \in \Sigma}$ iff it is* Locally Threshold Testable.

Membership in an FO(+1) definable stringset depends only on the multiplicity of the k-factors, up to some fixed finite threshold, which occur in the string.

Slide 18

Probing the LTT Boundary

$$\text{One-B} = \{w \in \{A, B\}^* \mid w \models (\exists x)[B(x) \wedge (\forall y)[B(y) \rightarrow x \approx y]\,]\}(\in \text{LTT})$$

$$\text{B-before-C} \stackrel{\text{def}}{=} \{w \in \{A, B, C\}^* \mid \text{at least one B precedes any C}\} \notin \text{LTT}$$

$A^kBA^kCA^k$ and $A^kCA^kBA^k$ have exactly the same number of occurrences of every k-factor.

		In	Out
LTT	One-B	A^iBA^{j+k+1}	$A^iBA^jBA^k$
non-LTT	B-before-C	$A^iBA^jCA^k$	$A^iCA^jBA^k$

FO(<) (Strings)

$$ABACA \models (\exists x)[\ C(x) \rightarrow (\exists y)[B(y) \wedge y \triangleleft^{+} x]\ ]$$

$\langle \mathcal{D}, \triangleleft, \triangleleft^{+}, P_{\sigma} \rangle_{\sigma \in \Sigma}$

First-order Quantification over positions in the strings

Slide 19

$x \triangleleft y$	$w, [x \mapsto i, y \mapsto j] \models x \triangleleft y$	$\stackrel{\text{def}}{\Longleftrightarrow}$	$j = i + 1$
$x \triangleleft^{+} y$	$w, [x \mapsto i, y \mapsto j] \models x \triangleleft^{+} y$	$\stackrel{\text{def}}{\Longleftrightarrow}$	$i < j$
$P_{\sigma}(x)$	$w, [x \mapsto i] \models P_{\sigma}(x)$	$\stackrel{\text{def}}{\Longleftrightarrow}$	$i \in P_{\sigma}$
$\varphi \wedge \psi$	$\vdots$		
$\neg\varphi$	$\vdots$		
$(\exists x)[\varphi(x)]$	$w, s \models (\exists x)[\varphi(x)]$	$\stackrel{\text{def}}{\Longleftrightarrow}$	$w, s[x \mapsto i] \models \varphi(x)]$ for some $i \in \mathcal{D}$

Locally Testable with Order (LTO$_k$)

Slide 20

LT$_k$ plus

$$\varphi \bullet \psi \qquad w \models \varphi \bullet \psi \stackrel{\text{def}}{\Longleftrightarrow} w = w_1 \cdot w_2, \quad w_1 \models \varphi \text{ and } w_2 \models \psi.$$

B-before-C: $(\rtimes B \vee AB \bullet \rtimes C \vee AC) \vee \neg(\rtimes C \vee AC \vee BC)$

Definition 2 (Star-Free Set) *The class of* Star-Free Sets *(SF) is the smallest class of languages satisfying:*

- $\emptyset \in SF$, $\{\varepsilon\} \in SF$, *and* $\{\sigma\} \in SF$ *for each* $\sigma \in \Sigma$.
- *If* $L_1, L_2 \in SF$ *then:* $L_1 \cdot L_2 \in SF$,
  $L_1 \cup L_2 \in SF$,
  $\overline{L_1} \in SF$.

Theorem 2 (McNauthton and Papert) *A set of strings is First-order definable over* $\langle \mathcal{D}, \triangleleft, \triangleleft^{+}, P_{\sigma} \rangle_{\sigma \in \Sigma}$ *iff it is* Star-Free.

Character of FO(<) Definable Sets

Slide 21

Theorem (McNaughton and Papert):
A stringset L is definable by a set of First-Order formulae over strings iff it is recognized by a finite-state automaton that is *non-counting* (that has an *aperiodic* syntactic monoid), that is, iff:

> there exists some $n > 0$ such that
>
> > for all strings u, v, w over Σ
> >
> > > if $uv^n w$ occurs in L
> > >
> > > then $uv^{n+i}w$, for all $i \geq 1$, occurs in L as well.

E.g.

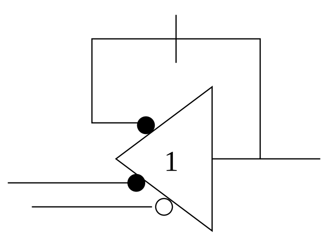
$$\frac{\textbf{those people who were left (by people who were left)}^{n}\ \textbf{left} \quad \in L}{\textbf{those people who were left (by people who were left)}^{n+1}\ \textbf{left} \quad \in L}$$

A Characterization via ANNs

Slide 22

Binary valued Artificial Neural Nets

> Buzzer-free: no inhibitory feedback.
>
> Almost loop-free: no loops including more than one neuron or delay greater than one.



Theorem (McNaughton and Papert):
A stringset L is definable by a set of First-Order formulae over strings iff it is representable by a buzzer-free, almost loop-free ANN.

Slide 23

Probing the LT0 Boundary

B-before-C $= \{w \in \{A, B\}^* \mid w \models (\exists x)[C(x) \rightarrow (\exists y)[B(y) \wedge y < x]\,]\}(\in \text{LTO})$

$$\text{Even-B} \stackrel{\text{def}}{=} \{w \in \{A, B\}^* \mid |w|_B = 2i,\, 0 \leq i\} \notin \text{LTT}$$

$$A^i B^n B^n \in \text{Even-B} \quad \text{but} \quad A^i B^{n+1} B^n \notin \text{Even-B}$$

		In	Out
LTO	B-before-C	$A^iBA^jCA^k$	$A^iCA^jBA^k$
non-LTO	Even-B	B^{2i}	B^{2i+1}

Slide 24

MSO (Strings)

$\langle \mathcal{D}, \triangleleft, \triangleleft^+, P_\sigma \rangle_{\sigma \in \Sigma}$

First-order Quantification (positions)

Monadic Second-order Quantification (sets of positions)

$\triangleleft^+$ is MSO-definable from $\triangleleft$.

MSO Example

$$(\exists X_0)[\quad (\forall x)[(\exists y)[y \triangleleft x] \vee X_0(x)] \wedge \\ (\forall x,y)[x \triangleleft y \rightarrow (X_0(x) \leftrightarrow \neg(X_0(y))] \wedge \\ (\forall x)[(\exists y)[x \triangleleft y] \vee \neg(X_0(x))] \quad]$$

Slide 25

c	b	a	b	c	b
X_0	$\overline{X_0}$	X_0	$\overline{X_0}$	X_0	$\overline{X_0}$

Theorem 3 (Chomsky Shützenberger) *A set of strings is Regular iff it is a homomorphic image of a Strictly 2-Local set.*

Definition (**Nerode Equivalence**) *Two strings w and v are* Nerode Equivalent *with respect to a stringset L over Σ (denoted $w \equiv_L v$) iff for all strings u over Σ, $wu \in L \Leftrightarrow vu \in L$.*

Slide 26

Theorem 4 (Myhill-Nerode) *A stringset L is recognizable by a FSA (over strings) iff $\equiv_L$ partitions the set of all strings over Σ into finitely many equivalence classes.*

Theorem 5 (Büchi, Elgot) *A set of strings is MSO-definable over $\langle \mathcal{D}, \triangleleft, \triangleleft^+, P_\sigma \rangle_{\sigma \in \Sigma}$ iff it is regular.*

Probing the FS Boundary

Slide 27

$$\text{Even-B} \stackrel{\text{def}}{=} \{w \in \{A,B\}^* \mid |w|_B = 2i,\ 0 \le i\} \in \text{FS}$$

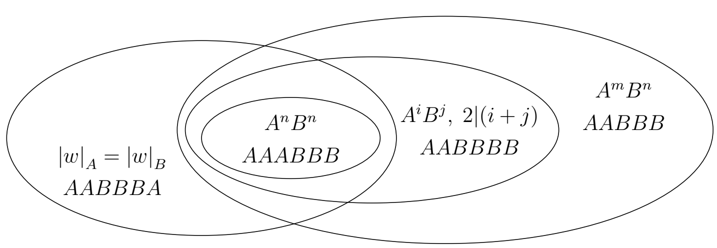
$$\{A^n B^n \mid n > 0\} \notin \text{FS}$$

$$w \equiv_{A^n B^n} v \quad \Leftrightarrow \quad w, v \notin \{A^i B^j \mid i, j \ge 0\} \text{ or } |w|_A - |w|_B = |v|_A - |v|_B\,.$$

		In	Out
FS	Even-B	B^{2i}	B^{2i+1}
non-FS	A^nB^n	A^nB^n	$A^{n-1}B^{n+1}$

Testing A^nB^n

Slide 28



Context-Free

Slide 29

Additional structure — not finitely bounded

$A^n B^n$

$D_1 = |w|_A = |w|_B$, properly nested

$D_2 = |w|_A = |w|_B$ and $|w|_C = |w|_D$, properly nested.

Subregular Hierarchy over Trees

$$CFG = SL_2 \; < \; LT \; < \; FO(+1) \; < \; FO(<) \; < \; MSO = \text{FSTA}$$

Conclusions

Slide 30

FLT support for aural pattern recognition experiments

Model-theoretic characterizations
– very general methods for describing patterns
– provide clues to nature of cognitive mechanisms
– independent of *a priori* assumptions

Grammar- and Automata-theoretic characterizations
– provide information about nature of stringsets
– minimal pairs

Sub-regular hierarchy

- broad range of capabilities weaker than human capabilities
- characterizations in terms of plausible cognitive attributes

References

Beauquier, D., and Jean-Eric Pin. 1991. Languages and scanners. *Theoretical Computer Science* 84:3–21.

Büchi, J. Richard. 1960. Weak second-order arithmetic and finite automata. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik* 6:66–92.

Chomsky, Noam, and M. P. Schützenberger. 1963. The algebraic theory of context-free languages. In *Computer programming and formal systems*, ed. P. Braffort and D. Hirschberg, Studies in Logic and the Foundations of Mathematics, 118–161. Amsterdam: North-Holland, 2nd (1967) edition.

Elgot, Calvin C. 1961. Decision problems of finite automata and related arithmetics. *Transactions of the American Mathematical Society* 98:21–51.

McNaughton, Robert, and Seymour Papert. 1971. *Counter-free automata*. Cambridge, MA: MIT Press.

Perrin, Dominique, and Jean-Eric Pin. 1986. First-Order logic and Star-Free sets. *Journal of Computer and System Sciences* 32:393–406.

Rogers, James. 2003. wMSO theories as grammar formalisms. *Theoretical Computer Science* 293:291–320.

Thomas, Wolfgang. 1982. Classifying regular events in symbolic logic. *Journal of Computer and Systems Sciences* 25:360–376.