Cognitive Complexity of Linguistic Patterns

Artificial Grammar Learning Workshop
Max Planck Institute for Psycholinguistics
23–24 November 2010

Slide 1

James Rogers
Dept. of Computer Science
Earlham College
jrogers@cs.earlham.edu

http://cs.earlham.edu/~jrogers/slides/agl.ho.pdf

This work completed, in part, while in residence at the
Radcliffe Institute for Advanced Study

Slide 2

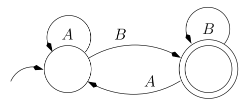
Joint work with:

- Geoffrey K. Pullum
  School of Philosophy, Psychology and Language Sciences
  University of Edinburgh
- Marc D. Hauser
  Depts. of Psychology, Organismic & Evolutionary Biology and Biological Anthropology
  Harvard University
- Jeffery Heinz
  Dept. of Linguistics and Cognitive Science
  University of Delaware
- Gil Bailey, Matt Edlefsen, Magaret Fero, Molly Visscher, David Wellcome, Aaron Weeden and Sean Wibel
  Dept. of Computer Science
  Earlham College

Cognitive Complexity of Simple Patterns

Sequences of 'A's and 'B's which end in 'B':

$$S_0 \longrightarrow AS_0,\ S_0 \longrightarrow BS_0,\ S_0 \longrightarrow B$$

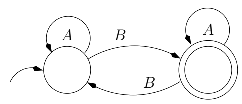


$(A+B)^*B$

Slide 3

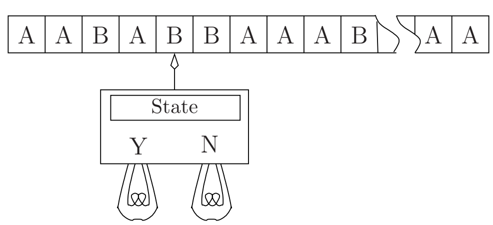
Sequences of 'A's and 'B's which contain an odd number of 'B's:

$$S_0 \longrightarrow AS_0,\ S_0 \longrightarrow BS_1,\ S_0 \longrightarrow B,$$
$$S_1 \longrightarrow AS_1,\ S_1 \longrightarrow BS_0,\ S_1 \longrightarrow A$$

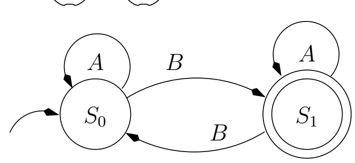


$(A^*BA^*BA^*)^*A^*BA^*$

Finite State Automata and Regular Grammars



Slide 4

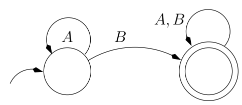


$$S_0 \longrightarrow AS_0,\ S_0 \longrightarrow BS_1,\ S_0 \longrightarrow B,$$
$$S_1 \longrightarrow AS_1,\ S_1 \longrightarrow BS_0,\ S_1 \longrightarrow A$$

Some More Simple Patterns

Sequences of 'A's and 'B's which contain at least one 'B':

$$S_0 \longrightarrow AS_0,\ S_0 \longrightarrow BS_1,\ S_0 \longrightarrow B,$$
$$S_1 \longrightarrow AS_1,\ S_1 \longrightarrow BS_1,\ S_1 \longrightarrow A,\ S_1 \longrightarrow B$$

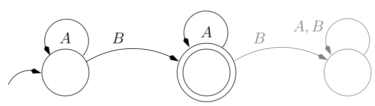


$$A^*B(A+B)^*$$

Slide 5

Sequences of 'A's and 'B's which contain exactly one 'B':

$$S_0 \longrightarrow AS_0,\ S_0 \longrightarrow BS_1,\ S_0 \longrightarrow B,$$
$$S_1 \longrightarrow AS_1,\ S_1 \longrightarrow A$$



$$A^*BA^*$$

Cognitive Complexity from First Principles

What kinds of distinctions does a cognitive mechanism need to be sensitive to (attend to) in order to classify an event with respect to a pattern?

Slide 6

Reasoning about patterns

- What objects/entities/things are we reasoning about?
- What relationships between them are we reasoning with?

Some Assumptions about (Proto-)Linguistic Behaviors

Slide 7

- Perceive/process/generate linear sequence of (sub)events
- Can model as strings—linear sequence of abstract symbols
  - Positions—Discrete linear order (initial segment of $\mathbb{N}$).
  - Labeled with alphabet of events

    Partitioned into subsets, each the set of positions at which a particular event occurs.

Dual characterizations of complexity classes

Slide 8

Computational classes

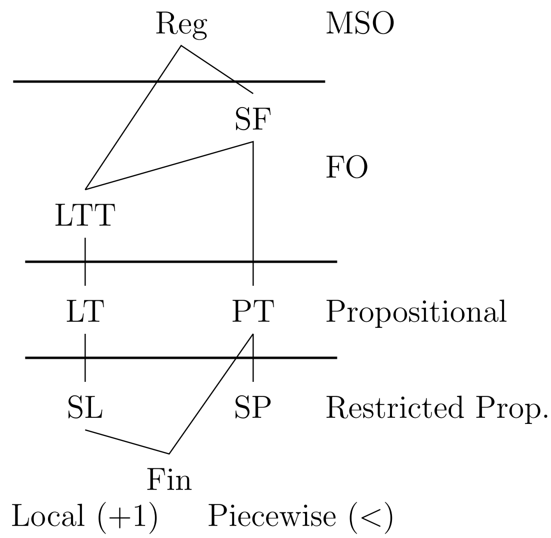
- Characterized by abstract computational mechanisms
- Equivalence between mechanisms
- Means to determine structural properties of stringsets

Descriptive classes

- Characterized by the nature of information about the properties of strings that determine membership
- Independent of mechanisms for recognition
- Subsume wide range of types of patterns

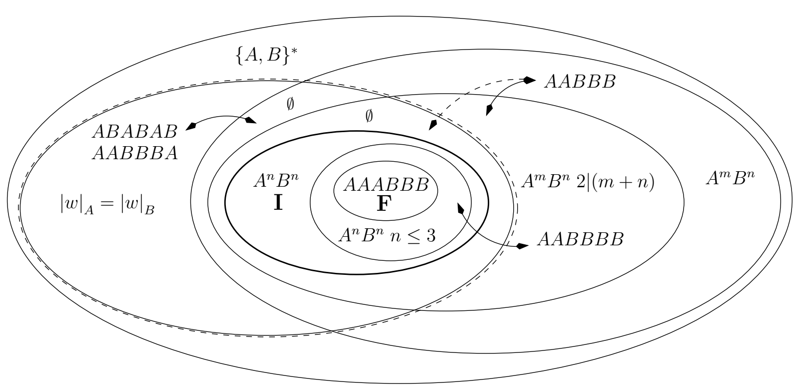
Local and Piecewise Hierarchies

Slide 9



Stringset inference experiments

Slide 10



$I = \{A^nB^n \mid n \geq 1\}$ $\quad F = \{AAABBB\}$ $\quad D = ?$

Slide 11

Formal Issues for AGL Experiments

Design

- Identifying relevant classes of patterns
- Finding minimal pairs of stringsets
- Finding sets of stimuli that distinguish those stringsets

Interpretation

- Identifying the class of patterns subject has generalized to
- Inferring the properties of the cognitive mechanism involved
  - properties common to all mechanisms capable of identifying that class of patterns

Slide 12

Inferences from AGL experiments

Subject successfully generalizes a pattern in a given complexity class:

- Mechanism is sensitive to features characteristic of class.
- Does not imply that subject can generalize every pattern in that class.
  - Other processing factors may interfere.

Subject consistently fails to generalize patterns in a given class:

- Suggests mechanism is not sensitive to features characteristic of class.
- Inability to generalize may be due to interfering factors.
  - Complexity of patterns properly in class may exceed other limitations of processing.

Slide 13

Assumptions

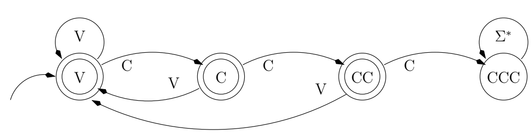
- Inferred set is not arbitrary
- Principle determining membership is structural
- Inference exhibits some sort of minimality

Slide 14

Yawelmani Yokuts (Kissberth'73)

$$\star\, CCC$$

$$\overline{\Sigma^* CCC \Sigma^*}$$



$CCVCVVCVCCVCVCVCCVCCVVVCV$

$CCVCVVCVCCC^{\star}VCVCVCCVCCVVVCV$

Adjacency—Substrings

Definition 1 (k-Factor)
v is a factor *of w if $w = uvx$ for some $u, v \in \Sigma^*$.*

*v is a k-*factor *of w if it is a factor of w and $|v| = k$.*

$$F_k(w) \stackrel{def}{=} \begin{cases} \{v \in \Sigma^k \mid (\exists u, x \in \Sigma^*)[w = uvx]\} & \textit{if } |w| \geq k, \\ \{w\} & \textit{otherwise.} \end{cases}$$

Slide 15

$F_k(w)$ is the set of length k substrings (contiguous) of w (or just w itself if length of $w < k$).

$$ABABAB$$

$$F_2(ABABAB) = \{AB, BA\}$$
$$F_7(ABABAB) = \{ABABAB\}$$

Strictly Local Stringsets—SL

Strictly k-Local Definitions

$$\mathcal{G} \subseteq F_k(\{\rtimes\} \cdot \Sigma^* \cdot \{\ltimes\})$$
$$w \models \mathcal{G} \stackrel{\text{def}}{\iff} F_k(\rtimes \cdot w \cdot \ltimes) \subseteq \mathcal{G}$$
$$L(\mathcal{G}) \stackrel{\text{def}}{=} \{w \mid w \models \mathcal{G}\}$$

Slide 16

A stringset L is Strictly k-Local iff membership depends solely on the k-factors that are permitted.

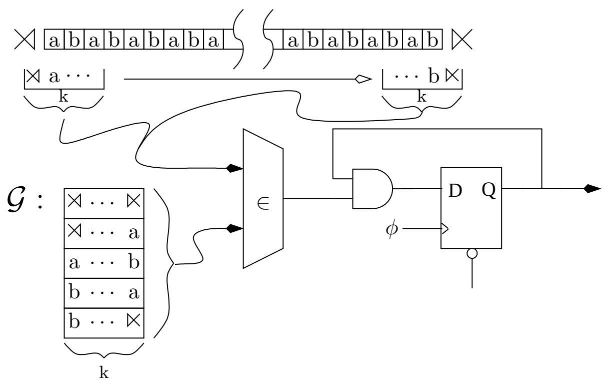
$$\mathcal{G}_{(AB)^n} = \{\rtimes A, AB, BA, B\ltimes\}$$

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

Membership in an SL_k stringset depends only on the individual k-factors which actually occur in the string.

Scanners

Slide 17



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

Character of Strictly k-Local Sets

Slide 18

Theorem (Suffix Substitution Closure):
A stringset L is strictly k-local iff whenever there is a string x of length $k-1$ and strings w, y, v, and z, such that

$$\begin{array}{ccccccc} w & \cdot & \overbrace{x}^{k-1} & \cdot & y & \in L \\ v & \cdot & x & \cdot & z & \in L \end{array}$$

then it will also be the case that

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

Slide 19

Examples of Suffix Substitution

$$\begin{array}{rcccll} \text{The dog} & \cdot & \text{likes} & \cdot & \text{the biscuit} & \in L \\ \text{Alice} & \cdot & \text{likes} & \cdot & \text{Bob} & \in L \\ \hline \text{The dog} & \cdot & \text{likes} & \cdot & \text{Bob} & \in L \end{array}$$

But:

$$\begin{array}{rcccll} \text{The dog} & \cdot & \text{likes} & \cdot & \text{the biscuit} & \in L \\ \text{Bob, Alice} & \cdot & \text{likes} & \cdot & \varepsilon & \in L \\ \hline \star\text{The dog} & \cdot & \text{likes} & \cdot & \varepsilon & \notin L \end{array}$$

Slide 20

SL Hierarchy

Definition 2 (SL)
A stringset is Strictly k-Local *if it is definable with an* SL_k *definition.*

A stringset is Strictly Local *(in SL) if it is* SL_k *for some* k.

Theorem 1 (SL-Hierarchy)

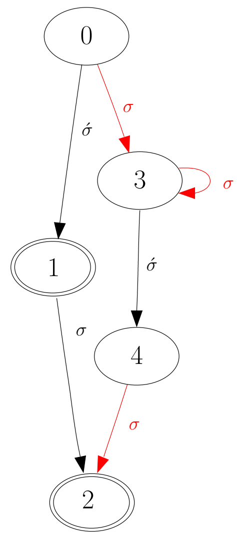
$$SL_2 \subsetneq SL_3 \subsetneq \cdots \subsetneq SL_i \subsetneq SL_{i+1} \subsetneq \cdots \subsetneq SL$$

Every Finite stringset is SL_k for some k: Fin $\subseteq$ SL.

There is no k for which SL_k includes all Finite languages.

Alawa

Slide 21

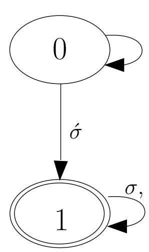


$\rtimes\sigma$	σ	$\acute{\sigma}\sigma\ltimes$
$\rtimes\acute{\sigma}$	σ	$\ltimes$
$\star\ \rtimes\sigma$	σ	$\ltimes$

$$\mathcal{G}_{\text{Alawa}} = \{\quad \rtimes\sigma\sigma,\ \rtimes\sigma\acute{\sigma},\ \rtimes\acute{\sigma}\sigma,\ \sigma\sigma\sigma,\ \sigma\sigma\acute{\sigma},\ \sigma\acute{\sigma}\sigma,\ \rtimes\acute{\sigma}\ltimes,\ \acute{\sigma}\sigma\ltimes \quad\}$$

Some syllable must get primary stress

Slide 22



$\rtimes\sigma_1$	$\overbrace{\sigma_0\cdots\sigma_0}^{k-1}$	$\acute{\sigma_2}\ltimes$
$\rtimes\acute{\sigma_2}$	$\overbrace{\sigma_0\cdots\sigma_0}^{k-1}$	$\sigma_1\ltimes$
$\star\ \rtimes\sigma_1$	$\overbrace{\sigma_0\cdots\sigma_0}^{k-1}$	$\sigma_1\ltimes$

Cognitive interpretation of SL

Slide 23

- Any cognitive mechanism that can distinguish member strings from non-members of an SL_k stringset must be sensitive, at least, to the length k blocks of events that occur in the presentation of the string.
- If the strings are presented as sequences of events in time, then this corresponds to being sensitive, at each point in the string, to the immediately prior sequence of $k-1$ events.
- Any cognitive mechanism that is sensitive *only* to the length k blocks of events in the presentation of a string will be able to recognize *only* SL_k stringsets.

Strictly Local Stress Patterns

Slide 24

Heinz's Stress Pattern Database (ca. 2007)—109 patterns

9 are SL_2	Abun West, Afrikans, . . . Cambodian,. . . Maranungku
44 are SL_3	Alawa, Arabic (Bani-Hassan),. . .
24 are SL_4	Arabic (Cairene),. . .
3 are SL_5	Asheninca, Bhojpuri, Hindi (Fairbanks)
1 is SL_6	Icua Tupi
28 are not SL	Amele, Bhojpuri (Shukla Tiwari), Arabic Classical, Hindi (Keldar), Yidin,. . .

72% are SL, all $k \leq 6$. 49% are SL_3.

Slide 25

Probing the SL boundary

$$(AB)^n = L(\{\rtimes A, AB, BA, B\ltimes\}) \in \mathrm{SL}_2$$

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

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

		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}

Slide 26

Locally k-Testable Stringsets

Boolean combinations of SL_k stringsets

k-Expressions

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

Locally k-Testable Languages (LT_k):

$$L(\varphi) \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid w \models \varphi\}$$

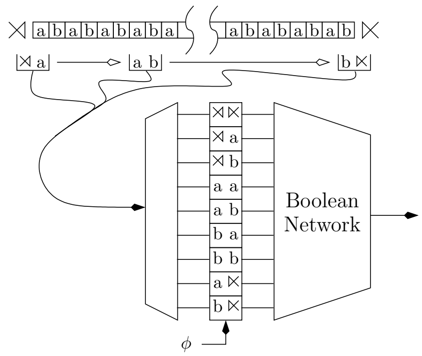
$$\text{Some-}B = L(\rtimes B \lor AB) \qquad (= L(\neg(\neg\rtimes B \land \neg AB)))$$

LT stringsets are those definable in Propositional Logic with k-factors as atomic formulae.

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

Slide 27

LT Automata



Recognizing an LT$_k$ stringset requires only remembering which k-factors occur in the string.

Slide 28

Character of Locally Testable sets

Theorem 2 (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.*

$$w \equiv^L_k v \stackrel{\text{def}}{\iff} F_k(\rtimes w \ltimes) = F_k(\rtimes v \ltimes).$$

LT$_k$ stringsets do not distinguish strings that have the same set of k-factors.

LT Hierarchy

Definition 3 (LT)
A stringset is k-Locally Testable *if it is definable with an* LT_k*-expression.*

Slide 29

A stringset is Locally Testable *(in LT) if it is* LT_k *for some* k*.*

Theorem 3 (LT-Hierarchy)

$$LT_2 \subsetneq LT_3 \subsetneq \cdots \subsetneq LT_i \subsetneq LT_{i+1} \subsetneq \cdots \subsetneq LT$$

Examples of k-Test Invariance

Some syllable gets primary stress is LT$_1$

$w \in \text{Some-}\acute{\sigma} \Leftrightarrow \acute{\sigma} \in F_1(\rtimes\cdot w\cdot\ltimes) \qquad \text{Some-}\acute{\sigma} = \{w \in \{A, B\}^* \mid w \models \acute{\sigma}\}$

No more than one syllable gets primary stress is not LT (not LT$_k$ for any k)

Slide 30

$$\begin{aligned} & F_k(\rtimes \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \acute{\sigma} \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \ltimes) \\ = \quad & F_k(\rtimes \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \acute{\sigma} \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \acute{\sigma} \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \ltimes) \end{aligned}$$

But

$$\begin{aligned} \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \acute{\sigma} \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \quad & \in \quad \text{OnlyOne-}\acute{\sigma} \\ \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \acute{\sigma} \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \cdot \acute{\sigma} \cdot \overbrace{\sigma\cdots\sigma}^{k-1} \quad & \notin \quad \text{OnlyOne-}\acute{\sigma} \end{aligned}$$

Cognitive interpretation of LT

Slide 31

- Any cognitive mechanism that can distinguish member strings from non-members of an LT_k stringset must be sensitive, at least, to the set of length k blocks of events that occur in the presentation of the string—both those that do occur and those that do not.
- If the strings are presented as sequences of events in time, then this corresponds to being sensitive, at each point in the string, to the length k blocks of events that occur at any prior point.
- Any cognitive mechanism that is sensitive *only* to the set of length k blocks of events in the presentation of a string will be able to recognize *only* LT_k stringsets.

Probing the LT boundary

Slide 32

$$\text{Some-B} = L(\rtimes B \vee AB) \quad \in \mathrm{LT}_2$$

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

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

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

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

Slide 33

FO(+1) (Strings)

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

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

First-order Quantification (over positions in the strings)

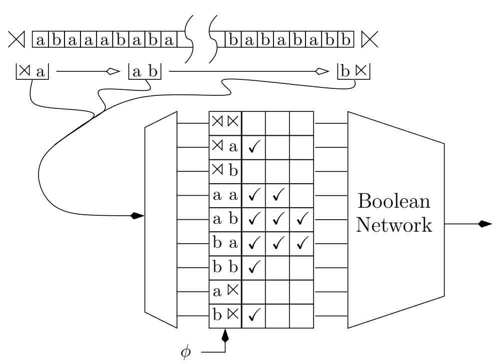
$$\begin{array}{lrcl}
x \triangleleft y & w, [x \mapsto i, y \mapsto j] \models x \triangleleft y & \stackrel{\text{def}}{\Longleftrightarrow} & j = i + 1 \\
P_\sigma(x) & w, [x \mapsto i] \models P_\sigma(x) & \stackrel{\text{def}}{\Longleftrightarrow} & i \in P_\sigma \\
\varphi \wedge \psi & \vdots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

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

$$\text{One-}\acute{\sigma} = L((\exists x)[\acute{\sigma}(x) \wedge (\forall y)[\acute{\sigma}(y) \rightarrow x \approx y]\,])$$

Slide 34

LTT Automata



Character of the FO(+1) Definable Stringsets

Slide 35

Definition 4 (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 4 (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 which occur in the string, up to some fixed finite threshold t.

Examples of Local Threshold Testability

Slide 36

One-$\acute{\sigma}$ is LTT$_{1,2}$

$$w \in \text{One-}\acute{\sigma} \Leftrightarrow |w|_{\acute{\sigma}} = 1 \qquad (\text{and not } |w|_{\acute{\sigma}} \geq 2)$$

First heavy syllable gets primary stress is not LTT (LTT$_{k,t}$ for any k or t)

$$\begin{array}{rl} & F_k(\rtimes \cdot \overbrace{L \cdots L}^{k-1} \cdot \acute{H} \cdot \overbrace{L \cdots L}^{k-1} \cdot H \cdot \overbrace{L \cdots L}^{k-1} \cdot \ltimes) \\ = & F_k(\rtimes \cdot \overbrace{L \cdots L}^{k-1} \cdot H \cdot \overbrace{L \cdots L}^{k-1} \cdot \acute{H} \cdot \overbrace{L \cdots L}^{k-1} \cdot \ltimes) \end{array}$$

Another example of non-LTT

There must be an even number of heavy syllables $\notin$ LTT

$$|\rtimes\cdot\underbrace{\overbrace{L\cdots L}^{k-1}\cdot H}_{t}\cdot\overbrace{L\cdots L}^{k-1}\cdot\ltimes|_H \quad\geq\quad t$$

$$|\rtimes\cdot\underbrace{\overbrace{L\cdots L}^{k-1}\cdot H}_{t}\cdot\overbrace{L\cdots L}^{k-1}\cdot H\cdot\overbrace{L\cdots L}^{k-1}\cdot\ltimes|_H \quad\geq\quad t$$

Slide 37

But

$$\underbrace{\overbrace{L\cdots L}^{k-1}\cdot H}_{t}\cdot\overbrace{L\cdots L}^{k-1} \quad\in\quad \text{Even-}H$$

$$\Leftrightarrow$$

$$\underbrace{\overbrace{L\cdots L}^{k-1}\cdot H}_{t}\cdot\overbrace{L\cdots L}^{k-1}\cdot H\cdot\overbrace{L\cdots L}^{k-1} \quad\notin\quad \text{Even-}H$$

Cognitive interpretation of FO(+1)

Slide 38

- Any cognitive mechanism that can distinguish member strings from non-members of an FO(+1) stringset must be sensitive, at least, to the multiplicity of the length k blocks of events, for some fixed k, that occur in the presentation of the string, distinguishing multiplicities only up to some fixed threshold t.
- If the strings are presented as sequences of events in time, then this corresponds to being able count up to some fixed threshold.
- Any cognitive mechanism that is sensitive *only* to the multiplicity, up to some fixed threshold, (and, in particular, not to the order) of the length k blocks of events in the presentation of a string will be able to recognize *only* FO(+1) stringsets.

Probing the FO(+1) boundary

$$\text{One-}B = L((\exists x)[B(x) \wedge (\forall y)[B(y) \rightarrow x \approx y]]) \in \text{LTT}$$

$$\text{No-}B\text{-after-}C \stackrel{\text{def}}{=} \{w \in \{A, B, C\}^* \mid \text{no B follows any C}\} \quad \notin \text{LTT}$$

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

Slide 39

		In	Out
FO(+1)	One-B	A^iBA^{j+k+1}	$A^iBA^jBA^k$
non-FO(+1)	No-B-after-C	$A^iBA^jCA^k$	$A^iCA^jBA^k$
		$A^iBA^jBA^k$	
		$A^iCA^jCA^k$	

Long-Distance Dependencies

Sarcee sibilant harmony:

[-anterior] sibilants do not occur after [+anterior] sibilants

a. /si-tʃiz-aʔ/ → **ʃ**í**tʃ**íd**z**àʔ ‘my duck’

b. /na-s-ɣatʃ/ → nā**ʃ**ɣát**ʃ** ‘I killed them again’

c. cf. ⋆**s**í**tʃ**íd**z**àʔ

Slide 40

$$\overline{\Sigma^* \cdot [+] \cdot \Sigma^* \cdot [\text{-}] \cdot \Sigma^*}$$

Samala (Chumash) sibilant harmony:

[-anterior] sibilants do not occur in the same word as [+anterior] sibilants

[**ʃ**tojonowonowa**ʃ**] ‘it stood upright’ *[**ʃ**tojonowonowa**s**]

$$\overline{(\Sigma^* \cdot [+] \cdot \Sigma^* \cdot [\text{-}] \cdot \Sigma^*) + (\Sigma^* \cdot [\text{-}] \cdot \Sigma^* \cdot [+] \cdot \Sigma^*)}$$

Complexity of Sibilant Harmony

(Samala and Sarcee)

Symmetric sibilant harmony is LT

Slide 41

$$\neg([+] \wedge [-])$$

Asymmetric sibilant harmony is not FO$(+1)$

$$\rtimes w\,[-]\,w\,[+]\,w\ltimes$$
$$\equiv^{L}_{k,t}$$
$$\star\ \rtimes w\,[-]\,w\,[+]\,w\,[-]\,w\ltimes$$

Precedence—Subsequences

Definition 5 (Subsequences)

$$v \sqsubseteq w \overset{def}{\Longleftrightarrow} v = \sigma_1 \cdots \sigma_n \ \mathit{and}\ w \in \Sigma^* \cdot \sigma_1 \cdot \Sigma^* \cdots \Sigma^* \cdot \sigma_n \cdot \Sigma^*$$
$$P_k(w) \overset{def}{=} \{v \in \Sigma^k \mid v \sqsubseteq w\}$$
$$P_{\leq k}(w) \overset{def}{=} \{v \in \Sigma^{\leq k} \mid v \sqsubseteq w\}$$

Slide 42

$P_{\leq k}(w)$ is the set of subsequences (not necessarily contiguous) of length $\leq k$ occurring in w.

A A B A C A

AA, AB, BA, AC, CA
AB, AA, BC
AA, AC, BA
AC, AA
AA

$$P_2(AABACA) = \{AA, AB, AC, BA, BC, CA\}$$
$$P_{\leq 2}(AABACA) = \{\varepsilon, A, B, C, AA, AB, AC, BA, BC, CA\}$$

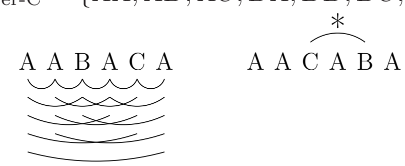
Strictly Piecewise Stringsets—SP

Strictly k-Piecewise Definitions

$$\mathcal{G} \subseteq \Sigma^{\leq k}$$
$$w \models \mathcal{G} \overset{\text{def}}{\iff} P_{\leq k}(w) \subseteq P_{\leq k}(\mathcal{G})$$
$$L(\mathcal{G}) \overset{\text{def}}{=} \{w \in \Sigma^* \mid w \models \mathcal{G}\}$$

Slide 43

$$\mathcal{G}_{\text{No-B-after-C}} = \{AA, AB, AC, BA, BB, BC, CA, CC\}$$



Membership in an SP$_k$ stringset depends only on the individual ($\leq k$)-subsequences which occur in the string.

Character of the Strictly k-Piecewise Sets

Theorem 5 *A stringset L is Strictly k-Piecewise Testable iff, for all $w \in \Sigma^*$,*

$$P_{\leq k}(w) \subseteq P_{\leq k}(L) \Rightarrow w \in L$$

Consequences:

Slide 44

Prefix & Suffix Closure:	$wv \in L \Rightarrow w, v \in L$
Subsequence Closure:	$w\sigma v \in L \Rightarrow wv \in L$
Unit Strings:	$P_1(L) \subseteq L$
Empty String:	$L \neq \emptyset \Rightarrow \varepsilon \in L$

A stringset L is SP$_k$ iff every subsequence of any string in L is also in L.

SP Hierarchy

Definition 6 (SP)
A stringset is Strictly k-Piecewise *if it is definable with an* SP_k *definition.*

A stringset is Strictly Piecewise *(in SP) if it is* SP_k *for some* k.

Theorem 6 (SP-Hierarchy)

Slide 45

$$SP_2 \subsetneq SP_3 \subsetneq \cdots \subsetneq SP_i \subsetneq SP_{i+1} \subsetneq \cdots \subsetneq SP$$

SP is incomparable (wrt subset) with the Local Hierarchy

$$\begin{array}{rl} \mathrm{SP}_2 \not\subseteq \mathrm{FO}(+1) & \text{No-}B\text{-after-}C \in \mathrm{SP}_2 - \mathrm{FO}(+1) \\ \mathrm{SL}_2 \not\subseteq \mathrm{SP} & (AB)^n \in \mathrm{SL}_2 - \mathrm{SP} \\ \mathrm{SP}_2 \cap \mathrm{SL}_2 \neq \emptyset & A^m B^n \in \mathrm{SP}_2 \cap \mathrm{SL}_2 \\ \mathrm{Fin} \not\subseteq \mathrm{SP} & \{A\} \in \mathrm{Fin} - \mathrm{SP} \end{array}$$

Cognitive interpretation of SP

- Any cognitive mechanism that can distinguish member strings from non-members of an SP_k stringset must be sensitive, at least, to the length k (not necessarily consecutive) sequences of events that occur in the presentation of the string.

Slide 46

- If the strings are presented as sequences of events in time, then this corresponds to being sensitive, at each point in the string, to up to $k-1$ events distributed arbitrarily among the prior events.
- Any cognitive mechanism that is sensitive *only* to the length k sequences of events in the presentation of a string will be able to recognize *only* SP_k stringsets.

Probing the SP boundary

$$\text{No-}B\text{-after-}C \in \mathrm{SP}_2$$

$$B\text{-before-}C \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid \text{Some B occurs prior to any C}\} \quad \notin \mathrm{SP}$$

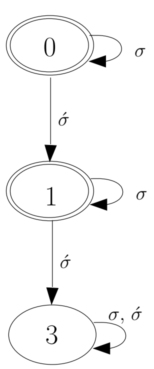
$$AABACA \in \text{B-before-C}, \quad AACA \sqsubseteq AABACA, \quad AACA \notin \text{B-before-C}$$

Slide 47

		In	Out
SP	No-B-after-C	$A^iBA^jCA^k$	$A^iCA^jBA^k$
		$A^iBA^jBA^k$	
		$A^iCA^jCA^k$	
	A^mB^n	$A^{i+k}B^{j+l}$	$A^iB^jA^kB^l$
non-SP	B-before-C	$A^iBA^jCA^k$	$A^iCA^jBA^k$
			$A^iCA^jCA^k$
	$(AB)^n$	$(AB)^{i+j+1}$	$(AB)^iAA(AB)^j$

No more than one syllable gets primary stress

Slide 48



$$\rtimes \overbrace{\sigma_0 \cdots \sigma_0}^{k-1} \acute{\sigma}_1 \overbrace{\sigma_0 \cdots \sigma_0}^{k-1} \ltimes$$

$$\equiv^L_k$$

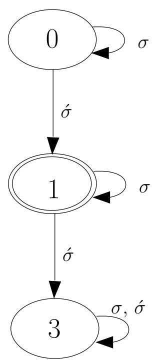
$$\rtimes \overbrace{\sigma_0 \cdots \sigma_0}^{k-1} \acute{\sigma}_1 \overbrace{\sigma_0 \cdots \sigma_0}^{k-1} \acute{\sigma}_1 \overbrace{\sigma_0 \cdots \sigma_0}^{k-1} \ltimes$$

$$\overline{\{\acute{\sigma}\acute{\sigma}\}}$$

$$\text{NoMoreThanOne-}B \in \{\mathrm{SP} - \mathrm{LT}\}$$

Slide 49

Exactly one syllable gets primary stress, reprise



$$\rtimes\overbrace{\sigma_0\cdots\sigma_0}^{k-1}\acute{\sigma}_1\overbrace{\sigma_0\cdots\sigma_0}^{k-1}\ltimes$$

$$\equiv_k^L$$

$$\rtimes\overbrace{\sigma_0\cdots\sigma_0}^{k-1}\acute{\sigma}_1\overbrace{\sigma_0\cdots\sigma_0}^{k-1}\acute{\sigma}_1\overbrace{\sigma_0\cdots\sigma_0}^{k-1}\ltimes$$

$$\grave{\sigma}\grave{\sigma}\sqsubseteq\grave{\sigma}\acute{\sigma}\grave{\sigma}$$

$$\text{One-}B\notin\text{LT}\qquad\text{One-}B\notin\text{SP}$$

$$\text{Some-}B\in\text{LT}\qquad\text{NoMoreThanOne-}B\in\text{SP}$$

$$\text{One-}B=\text{Some-}B\cap\text{NoMoreThanOne-}B$$

One-B is the co-occurence of LT and SP constraints.

Slide 50

k-Piecewise Testable Stringsets

PT$_k$-expressions

$$\begin{array}{ccc} p\in\Sigma^{\leq k} & w\models p\overset{\text{def}}{\iff}p\sqsubseteq w \\ \varphi\wedge\psi & w\models\varphi\wedge\psi\overset{\text{def}}{\iff}w\models\varphi\text{ and }w\models\psi \\ \neg\varphi & w\models\neg\varphi\overset{\text{def}}{\iff}w\not\models\varphi \end{array}$$

k-Piecewise Testable Languages (PT$_k$):

$$L(\varphi)\overset{\text{def}}{=}\{w\in\Sigma^*\mid w\models\varphi\}$$

$$B\text{-before-}C=L(\neg C\vee BC)\qquad(=L(C\rightarrow BC))$$

Membership in a PT$_k$ stringset depends only on the set of ($\leq k$)-subsequences which occur in the string.

Slide 51

Character of Piecewise Testable sets

Theorem 7 (k-Subsequence Invariance) *A stringset L is Piecewise Testable iff*

> *there is some k such that, for all strings x and y,*
>
> > *if x and y have exactly the same set of $(\leq k)$-subsequences*
> >
> > *then either both x and y are members of L or neither is.*

$$w \equiv_k^P v \overset{\text{def}}{\iff} P_{\leq k}(w) = P_{\leq k}(v).$$

$$B\text{-before-}C = \bigcup \{[w]_2^P \mid w \in \{A, B\}^*,\ w \models (C \to BC) \text{ and } |w| \leq 6\}.$$

PT$_k$ stringsets do not distinguish strings that have the same set of ($\leq k$)-subsequences.

Slide 52

PT Hierarchy

Definition 7 (SP)
*A stringset is k-*Piecewise Testable *if it is definable with an PT_k definition.*

A stringset is Piecewise Testable *(in PT) if it is PT_k for some k.*

Theorem 8 (PT-Hierarchy)

$$PT_2 \subsetneq PT_3 \subsetneq \cdots \subsetneq PT_i \subsetneq PT_{i+1} \subsetneq \cdots \subsetneq PT$$

Slide 53

PT, SP and the Local Hierarchy

$$\begin{array}{rl}
\mathrm{SP}_k \subsetneq \mathrm{PT}_k & \\
\mathrm{SP}_{k+1} \not\subseteq \mathrm{PT}_k & \\
\mathrm{PT}_2 \not\subseteq \mathrm{SP} & B\text{-before-}C, \text{One-}B \in \mathrm{PT}_2 - \mathrm{SP} \\
\mathrm{PT}_2 \not\subseteq \mathrm{FO}(+1) & \text{No-}B\text{-after-}C \in \mathrm{PT}_2 - \mathrm{FO}(+1) \\
\mathrm{SL}_2 \not\subseteq \mathrm{PT} & (AB)^n \in \mathrm{SL}_2 - \mathrm{PT} \\
\mathrm{PT}_2 \cap \mathrm{SL}_2 \neq \emptyset & A^m B^n \in \mathrm{PT}_2 \cap \mathrm{SL}_2
\end{array}$$

$\mathrm{Fin} \subseteq \mathrm{SP}$:

$$\Sigma^* = L(\varepsilon), \quad \emptyset = L(\neg\varepsilon), \quad \{\varepsilon\} = L(\bigwedge_{\sigma\in\Sigma}[\neg\sigma]),$$

$$\{w\} = L(w \wedge \bigwedge_{p\in\Sigma^{|w|+1}} [\neg p])$$

$$\{w_1, \ldots, w_n\} = L(\bigvee_{1\leq i\leq n} [w_i \wedge \bigwedge_{p\in\Sigma^{|w_i|+1}} [\neg p]])$$

Slide 54

Cognitive interpretation of PT

- Any cognitive mechanism that can distinguish member strings from non-members of an PT_k stringset must be sensitive, at least, to the set of length k subsequences of events that occur in the presentation of the string—both those that do occur and those that do not.
- If the strings are presented as sequences of events in time, then this corresponds to being sensitive, at each point in the string, to the set of all length k subsequences of the sequence of prior events.
- Any cognitive mechanism that is sensitive *only* to the set of length k subsequences of events in the presentation of a string will be able to recognize *only* PT_k stringsets.

Probing the PT boundary

$$B\text{-before-}C, \text{One-}B \in \text{PT}_2$$

$$(AB)^n \notin \text{PT}$$

$$(AB)^k \in (AB)^n \qquad\qquad (AB)^k A \notin (AB)^n$$

$$P_k((AB)^k A) = P_k((AB)^k)$$

Slide 55

		In	Out
PT	B-before-C	$A^iBA^jCA^k$	$A^iCA^jBA^k$
			$A^iCA^jCA^k$
	One-B	A^iBA^{j+k+1}	$A^iBA^jBA^k$
non-PT	$(AB)^n$	$(AB)^{i+j+1}$	$(AB)^iAA(AB)^j$

First-Order(<) definable stringsets

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

First-order Quantification over positions in the strings

Slide 56

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

Slide 57

PT, FO(+1) and FO(<)

Theorem 9 *$PT \subsetneq FO(<)$.*

$$\sigma_1 \cdots \sigma_n \sqsubseteq w \Leftrightarrow (\exists x_1, \ldots, x_n)[\bigwedge_{1 \leq i < j \leq n} [x_i \triangleleft^+ x_j] \wedge \bigwedge_{1 \leq i \leq n} [P_{\sigma_i}(x_i)]$$

Theorem 10 *$FO(+1) \subsetneq FO(<)$.*

+1 is FO definable from <:

$$x \triangleleft y \equiv x \triangleleft^+ y \wedge \neg(\exists z)[x \triangleleft^+ z \wedge z \triangleleft^+ y]$$

$$\text{No-}B\text{-after-}C \subseteq \text{FO}(<) - \text{FO}(+1)$$

$$(AB)^n \subseteq \text{FO}(<) - \text{PT}$$

Slide 58

Star-Free stringsets

Definition 8 (Star-Free Set) *The class of* Star-Free Sets *(SF) is the smallest class of languages satisfying:*

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

PT and LT with Order

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

LTO_k is LT_k plus $\varphi \bullet \psi$

$$\text{No-}B\text{-after-}C = L((\neg C) \bullet (\neg B)) \in \mathrm{LTO}$$

Slide 59

PTO_k is PT_k plus $\varphi \bullet \psi$
Let:

$$\varphi_{A^{=i}} = A^i \wedge \bigwedge_{p \in \Sigma^{i+1}} [\neg p], \qquad \varphi_{\Sigma^*} = \varepsilon$$

Then:

$$\begin{aligned}(AB)^n \quad = \quad & L(\neg(\varphi_{B^{=1}} \bullet \varphi_{\Sigma^*}) \wedge \neg(\varphi_{\Sigma^*} \bullet \varphi_{A^{=1}}) \wedge \\ & \neg(\varphi_{\Sigma^*} \bullet \varphi_{A^{=2}} \bullet \varphi_{\Sigma^*}) \wedge \neg(\varphi_{\Sigma^*} \bullet \varphi_{B^{=2}} \bullet \varphi_{\Sigma^*})) \quad \in \mathrm{PTO}\end{aligned}$$

PTO, LTO and SF

Theorem 12

$$PTO = SF = LTO$$

SF $\subseteq$ PTO, SF $\subseteq$ LTO

Slide 60

Fin $\subseteq$ PTO, Fin $\subseteq$ LTO and both are closed under concatenation, union and complement.

LTO $\subseteq$ PTO $\subseteq$ SF

Concatenation is FO($<$) definable.

Character of FO(<) definable sets

Theorem 13 (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

Slide 61

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.

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

Cognitive interpretation of FO(<)

- Any cognitive mechanism that can distinguish member strings from non-members of an FO(<) stringset must be sensitive, at least, to the sets of length k blocks of events, for some fixed k, that occur in the presentation of the string when it is factored into segments, up to some fixed number, on the basis of those sets with distinct criteria applying to each segment.

Slide 62

- If the strings are presented as sequences of events in time, then this corresponds to being able to count up to some fixed threshold with the counters being reset some fixed number of times based on those counts.
- Any cognitive mechanism that is sensitive *only* to the sets of length k blocks of events in the presentation of a string once it has been factored in this way will be able to recognize *only* FO(<) stringsets.

Probing the FO(<) boundary

$$BB\text{-before-}C \in \mathrm{FO}(<)$$

$$\text{Even-}B \stackrel{\text{def}}{=} \{w \in \{A,B\}^* \mid |w|_B = 2i,\, 0 \le i\} \notin \mathrm{FO}(<)$$

Slide 63

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

		In	Out
FO(<)	BB-before-C	$A^iBBA^{j+k}CA^l$	$A^iCA^{j+k}BBA^l$
			$A^iBA^jBA^kCA^l$
non-FO(<)	Even-B	B^{2i}	B^{2i+1}

MSO definable stringsets

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

Slide 64

First-order Quantification (positions)

Monadic Second-order Quantification (sets of positions)

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

MSO example

$$
\begin{aligned}
(\exists X_0, X_1)[\quad & (\forall x)[(\exists y)[y \triangleleft x] \vee X_0(x)] \wedge \\
& (\forall x, y)[\neg(X_0(x) \wedge X_1(x))] \wedge \\
& (\forall x, y)[x \triangleleft y \rightarrow (X_0(x) \leftrightarrow X_1(y)] \wedge \\
& (\forall x)[(\exists y)[x \triangleleft y] \vee X_1(x)] \qquad\qquad]
\end{aligned}
$$

Slide 65

a	b	b	a	b	a
X_0		X_0		X_0	
	X_1		X_1		X_1

Theorem 14 (Chomsky Schützenberger) *A set of strings is Regular iff it is a homomorphic image of a Strictly 2-Local set.*

Definition 9 (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 66

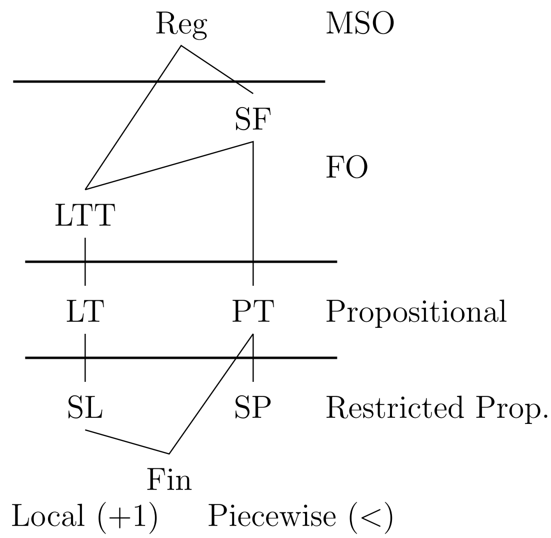
Theorem 15 (Myhill-Nerode) *A stringset L is recognizable by a FSA (over strings) iff $\equiv_L$ partitions the set of all strings over Σ into finitely many equivalence classes.*

Theorem 16 (Medvedev, 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.*

Theorem 17 *$MSO = \exists MSO$ over strings.*

Slide 67

Local and Piecewise Hierarchies



Slide 68

Cognitive interpretation of Finite-state

- Any cognitive mechanism that can distinguish member strings from non-members of a finite-state stringset must be capable of classifying the events in the input into a finite set of abstract categories and are sensitive to the sequence of those categories.
- Subsumes *any* recognition mechanism in which the amount of information inferred or retained is limited by a fixed finite bound.
- Any cognitive mechanism that has a fixed finite bound on the amount of information inferred or retained in processing sequences of events will be able to recognize *only* finite-state stringsets.

Slide 69

Probing the FS boundary

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

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

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

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

Slide 70

Non-FS classes

Additional structure — not finitely bounded

A^nB^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

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

FLT support for AGL 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

Slide 71

- provide information about nature of stringsets
- minimal pairs

Sub-regular hierarchies

- broad range of capabilities weaker than human capabilities
- characterizations in terms of plausible cognitive attributes
- relevant as long as generalizations are based on structure of strings

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.

García, Pedro, and José Ruiz. 1996. Learning k-piecewise testable languages from positive data. In *Grammatical Interference: Learning Syntax from Sentences*, ed. Laurent Miclet and Colin de la Higuera, volume 1147 of *Lecture Notes in Computer Science*, 203–210. Springer.

Heinz, Jeffrey. 2007. The inductive learning of phonotactic patterns. Doctoral Dissertation, University of California, Los Angeles.

Heinz, Jeffrey. 2008. Learning long distance phonotactics. Submitted manuscipt.

Heinz, Jeffrey. to appear. On the role of locality in learning stress patterns. *Phonology* .

Kontorovich, Leonid, Corinna Cortes, and Mehryar Mohri. 2006. Learning linearly separable languages. In *The 17th International Conference on Algorithmic Learning Theory (ALT 2006)*, volume 4264 of Lecture Notes in Computer Science, 288–303. Springer, Heidelberg, Germany.

Lothaire, M., ed. 2005. *Applied combinatorics on words*. Cambridge University Press, 2nd edition.

McNaughton, R., and S. Papert. 1971. *Counter-free automata*. 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.

Rogers, James, Jeffery Heinz, Gil Bailey, Matt Edlefsen, Molly Visscher, David Wellcome, and Sean Wibel. 2009. On languages piecewise testable in the strict sense. In *Preproceedings of 11th Meeting on Mathematics of Language*. Bielefeld, Germany. To Appear.

Rogers, James, and Geoffrey Pullum. 2007. Aural pattern recognition experiments and the subregular hierarchy. In *Proceedings of 10th Mathematics of Language Conference*, ed. Marcus Kracht, 1–7. University of California, Los Angeles.

Simon, Imre. 1975. Piecewise testable events. In *Automata Theory and Formal Languages: 2nd Grammatical Inference conference*, 214–222. Berlin ; New York: Springer-Verlag.

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