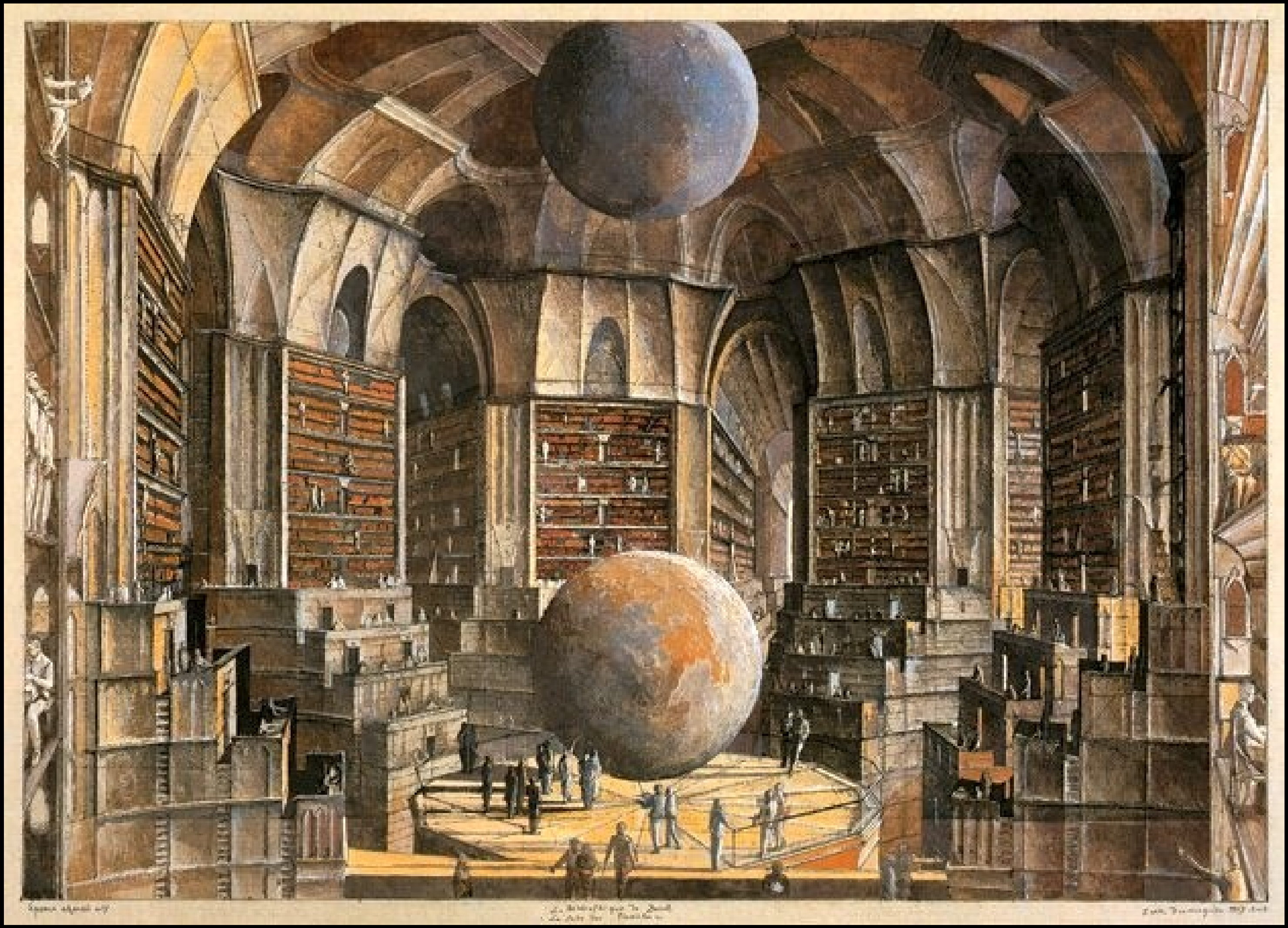
One model for the learning of language



Steve Piantadosi & Yuan Yang
UC Berkeley
Psychology & Neuroscience

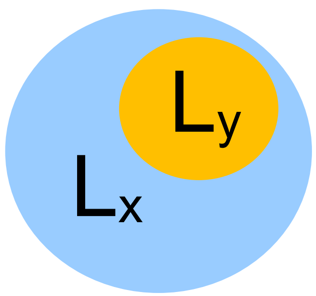
Outline

- **Overview of learnability & formal languages**
- **Learning model**
  - Simple formal languages
  - Artificial language learning
  - Simplified English CFG
- **Three related lines of ongoing work**
  - Human experiments
  - Recursion in monkeys and human groups
  - Algorithm learning in indigenous Amazonians

Gold's learnability result

- **Gold (1967)** showed that positive evidence is not enough for learners to necessarily identify their parent's target grammar (see Johnson 2004)

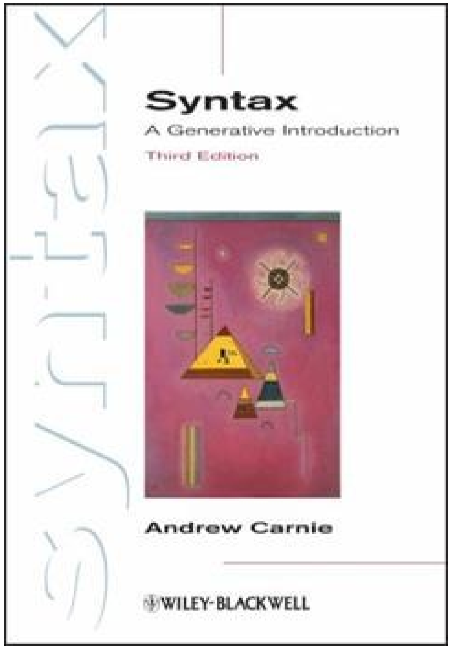
<u>Subset problem</u>



<u>Gold property</u>

$$L_1 \subset L_2 \subset L_3 \subset \ldots \subset L_\infty$$

- **Gold's theorem motivated a lot of theorizing about linguistic nativism,** averaging a citation every 4 days since 1967!
(e.g. Wexler & Culicover's *Formal Principles of Language Acquisition)*



6.3 The Logical Problem of Language Acquisition

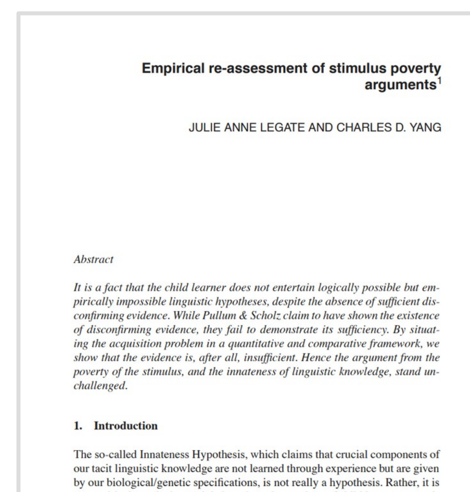
What follows is a fairly technical proof of the idea that parts of our linguistic system are at least plausibly construed as an innate, in-built system. If you aren't interested in this proof (and the problems with it), then you can reasonably skip ahead to section 6.4.

The argument in this section is that a productive system like the rules of Language probably could not be learned or acquired. Infinite systems are in principle, given certain assumptions, both unlearnable and unacquirable. Since we'll show that syntax is an infinite system, we shouldn't have been able to acquire it. So it follows that it is built in. The argument presented here is based on an unpublished paper by Alec Marantz, but is based on an argument dating back to at least Chomsky (1965).

First here's a sketch of the proof, which takes the classical form of an argument by modus ponens:

> *Premise (i)*: Syntax is a productive, recursive and infinite system.
> <u>*Premise (ii):* Rule-governed infinite systems are unacquirable.</u>
> *Conclusion*: Therefore syntax is an unacquirable system. Since we have such a system, it follows that at least parts of syntax are innate.

The so-called Innateness Hypothesis, which claims that crucial components of our tacit linguistic knowledge are not learned through experience but are given by our biological/genetic specifications, is not really a hypothesis. Rather, it is an empirical conclusion mainly based on observations of child language acquisition, one of which is now known as the *Argument from the Poverty of Stimulus* (APS).



Empirical re-assessment of stimulus poverty arguments[1]

JULIE ANNE LEGATE AND CHARLES D. YANG

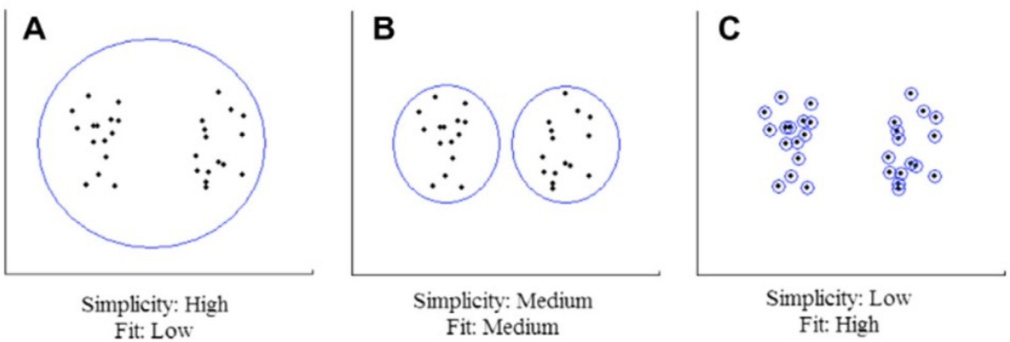
Abstract

It is a fact that the child learner does not entertain logically possible but empirically impossible linguistic hypotheses, despite the absence of sufficient disconfirming evidence. While Pullum & Scholz claim to have shown the existence of disconfirming evidence, they fail to demonstrate its sufficiency. By situating the acquisition problem in a quantitative and comparative framework, we show that the evidence is, after all, insufficient. Hence the argument from the poverty of the stimulus, and the innateness of linguistic knowledge, stand unchallenged.

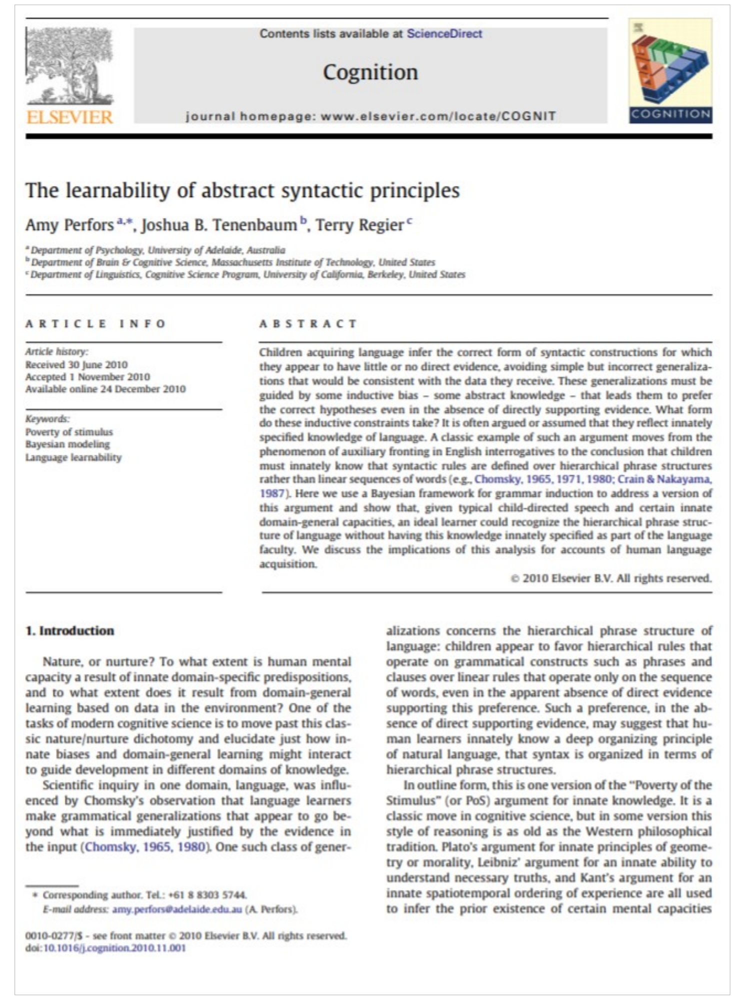
1. Introduction

The so-called Innateness Hypothesis, which claims that crucial components of our tacit linguistic knowledge are not learned through experience but are given by our biological/genetic specifications, is not really a hypothesis. Rather, it is

Child-directed speech supports hierarchical structure



Child-directed speech would lead an ideal learner to choose a hierarchical grammar over alternatives.



Contents lists available at ScienceDirect

Cognition

journal homepage: www.elsevier.com/locate/COGNIT

The learnability of abstract syntactic principles

Amy Perfors [a,*], Joshua B. Tenenbaum [b], Terry Regier [c]

[a] *Department of Psychology, University of Adelaide, Australia*
[b] *Department of Brain & Cognitive Science, Massachusetts Institute of Technology, United States*
[c] *Department of Linguistics, Cognitive Science Program, University of California, Berkeley, United States*

ARTICLE INFO

Article history:
Received 30 June 2010
Accepted 1 November 2010
Available online 24 December 2010

Keywords:
Poverty of stimulus
Bayesian modeling
Language learnability

ABSTRACT

Children acquiring language infer the correct form of syntactic constructions for which they appear to have little or no direct evidence, avoiding simple but incorrect generalizations that would be consistent with the data they receive. These generalizations must be guided by some inductive bias – some abstract knowledge – that leads them to prefer the correct hypotheses even in the absence of directly supporting evidence. What form do these inductive constraints take? It is often argued or assumed that they reflect innately specified knowledge of language. A classic example of such an argument moves from the phenomenon of auxiliary fronting in English interrogatives to the conclusion that children must innately know that syntactic rules are defined over hierarchical phrase structures rather than linear sequences of words (e.g., Chomsky, 1965, 1971, 1980; Crain & Nakayama, 1987). Here we use a Bayesian framework for grammar induction to address a version of this argument and show that, given typical child-directed speech and certain innate domain-general capacities, an ideal learner could recognize the hierarchical phrase structure of language without having this knowledge innately specified as part of the language faculty. We discuss the implications of this analysis for accounts of human language acquisition.

1. Introduction

Nature, or nurture? To what extent is human mental capacity a result of innate domain-specific predispositions, and to what extent does it result from domain-general learning based on data in the environment? One of the tasks of modern cognitive science is to move past this classic nature/nurture dichotomy and elucidate just how innate biases and domain-general learning might interact to guide development in different domains of knowledge.

Scientific inquiry in one domain, language, was influenced by Chomsky's observation that language learners make grammatical generalizations that appear to go beyond what is immediately justified by the evidence in the input (Chomsky, 1965, 1980). One such class of generalizations concerns the hierarchical phrase structure of language: children appear to favor hierarchical rules that operate on grammatical constructs such as phrases and clauses over linear rules that operate only on the sequence of words, even in the apparent absence of direct evidence supporting this preference. Such a preference, in the absence of direct supporting evidence, may suggest that human learners innately know a deep organizing principle of natural language, that syntax is organized in terms of hierarchical phrase structures.

In outline form, this is one version of the "Poverty of the Stimulus" (or PoS) argument for innate knowledge. It is a classic move in cognitive science, but in some version this style of reasoning is as old as the Western philosophical tradition. Plato's argument for innate principles of geometry or morality, Leibniz' argument for an innate ability to understand necessary truths, and Kant's argument for an innate spatiotemporal ordering of experience are all used to infer the prior existence of certain mental capacities

* Corresponding author. Tel.: +61 8 8303 5744.
E-mail address: amy.perfors@adelaide.edu.au (A. Perfors).

0010-0277/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.cognition.2010.11.001

Child-directed speech supports hierarchical structure

Log prior, likelihood, and posterior probabilities of each hand-designed grammar for each level of evidence. Because numbers are negative, smaller absolute values correspond to higher probability. If two grammars have log probabilities that differ by n, their actual probabilities differ by e^n; thus, the best hierarchical phrase-structure grammar CFG-L is e^{101} ($\sim 10^{43}$) times more probable than the best linear grammar REG-M. Bold values indicate the highest posterior score at each level.

Corpus	Probability	FLAT	REG-N	REG-M	REG-B	1-ST	CFG-S	CFG-L
Level 1	Prior	−99	−148	−124	−117	−94	−155	−192
	Likelihood	−17	−20	−19	−21	−36	−27	−27
	Posterior	−**116**	−168	−143	−138	−130	−182	−219
Level 2	Prior	−630	−456	−442	−411	−201	−357	−440
	Likelihood	−134	−147	−157	−162	−275	−194	−177
	Posterior	−764	−603	−599	−573	−**476**	−551	−617
Level 3	Prior	−1198	−663	−614	−529	−211	−454	−593
	Likelihood	−282	−323	−333	−346	−553	−402	−377
	Posterior	−1480	−986	−947	−875	−**764**	−856	−970
Level 4	Prior	−5839	−1550	−1134	−850	−234	−652	−1011
	Likelihood	−1498	−1761	−1918	−2042	−3104	−2078	−1956
	Posterior	−7337	−3311	−3052	−2892	−3338	−**2730**	−2967
Level 5	Prior	−10,610	−1962	−1321	−956	−244	−732	−1228
	Likelihood	−2856	−3376	−3584	−3816	−5790	−3917	−3703
	Posterior	−13,466	−5338	−4905	−4772	−6034	−**4649**	−4931
Level 6	Prior	−67,612	−5231	−2083	−1390	−257	−827	−1567
	Likelihood	−18,118	−24,454	−25,696	−27,123	−40,108	−27,312	−26,111
	Posterior	−85,730	−29,685	−27,779	−28,513	−40,365	−28,139	−**27,678**

More optimistic results about positive evidence

- **Positive evidence can lead you to the correct answer out of all computations** (Chater & Vitanyi, 2007).

Available online at www.sciencedirect.com
ScienceDirect
Journal of Mathematical Psychology 51 (2007) 135–163
Journal of Mathematical Psychology
www.elsevier.com/locate/jmp

'Ideal learning' of natural language: Positive results about learning from positive evidence

Nick Chater[a,*], Paul Vitányi[b]

[a] *Department of Psychology, University College, London WC1E 6BT, UK*
[b] *Centrum voor Wiskunde en Informatica, Amsterdam, The Netherlands*

Received 25 November 2005; received in revised form 5 September 2006
Available online 26 December 2006

Abstract

Gold's [1967. Language identification in the limit. *Information and Control*, *16*, 447–474] celebrated work on learning in the limit has been taken, by many cognitive scientists, to have powerful negative implications for the learnability of language from positive data (i.e., from mere exposure to linguistic input). This provides one, of several, lines of argument that language acquisition must draw on other sources of information, including innate constraints on learning. We consider an 'ideal learner' that applies a Simplicity Principle to the problem of language acquisition. The Simplicity Principle chooses the hypothesis that provides the briefest representation of the available data—here, the data are the linguistic input to the child. The Simplicity Principle allows learning from positive evidence alone, given quite weak assumptions, in apparent contrast to results on language learnability in the limit (e.g., Gold, 1967). These results provide a framework for reconsidering the learnability of various aspects of natural language from positive evidence, which has been at the center of theoretical debate in research on language acquisition and linguistics.

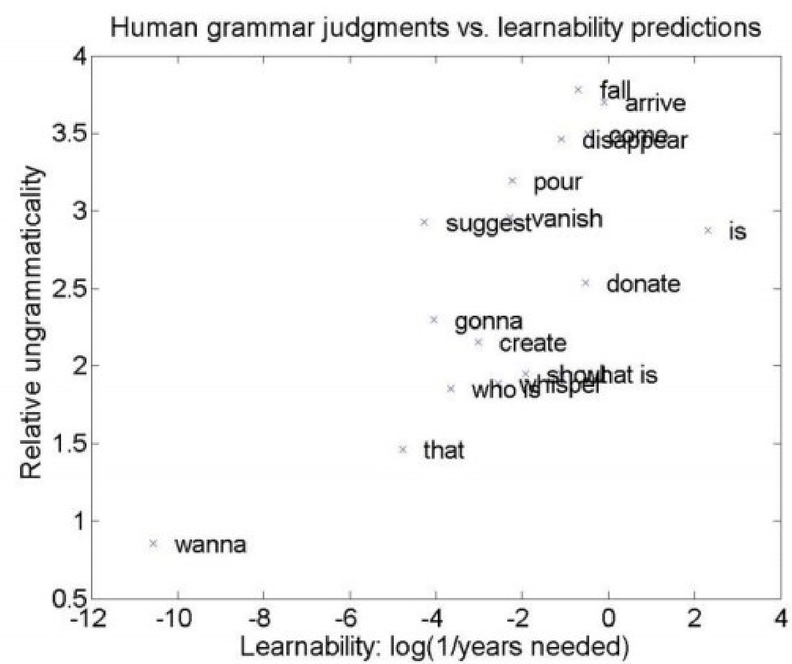
Keywords: Learnability; Language acquisition; Algorithmic complexity; Kolmogorov; Identification in the limit; Formal languages

1. Introduction

Language acquisition involves the rapid mastery of linguistic structure of astonishing complexity based on an input that appears noisy and partial. How can such an impoverished stimulus support such impressive learning? One influential line of argument is that it cannot—this "poverty of the stimulus" argument (Chomsky, 1980) is typically used to argue that language acquisition is guided by innate knowledge of language, often termed "universal grammar," that the child brings to bear on the learning problem (e.g., Chomsky, 1965, 1980; Hoekstra & Kooij, 1988). This type of argument for universal grammar is of central importance for the study of human language and language acquisition (e.g., Crain & Lillo-Martin, 1999; Hornstein & Lightfoot, 1981).

How can the poverty of the stimulus argument be assessed? At an abstract level, a natural approach is to attempt to somehow define an "ideal" language learner, which lacks universal grammar, but that can make the best use of the linguistic evidence that the child is given. If it were possible to show that this ideal learner is unable to learn language from the specific linguistic data available to the child, then we might reasonably conclude that some innate information must be available. Indeed, a second step, although not one we will consider in this paper, would be to attempt to prove that the ideal language learner, when provided with some appropriate innate information, is then able to learn language successfully from data of the sort available to the child.

Clearly, the task of constructing such an ideal language learner is a formidable one. We might reasonably suspect that the project of finding an optimal way of learning language is inherently open-ended; and our present understanding both of the mechanisms of human learning, and



Hsu, Chater, Vitanyi (2011)

ELSEVIER

Available online at www.sciencedirect.com
ScienceDirect

Journal of Mathematical Psychology 51 (2007) 135–163

Journal of Mathematical Psychology
www.elsevier.com/locate/jmp

'Ideal learning' of natural language: Positive results about learning from positive evidence

Nick Chater[a,*], Paul Vitányi[b]

[a]Department of Psychology, University College, London WC1E 6BT, UK
[b]Centrum voor Wiskunde en Informatica, Amsterdam, The Netherlands

Received 25

Abstract

Gold's [1967. Language identification in the
been taken, by many cognitive scientists, to ha
from mere exposure to linguistic input). This p
sources of information, including innate const
problem of language acquisition. The Simplicit
data—here, the data are the linguistic input t
quite weak assumptions, in apparent contrast
framework for reconsidering the learnability o
of theoretical debate in research on language

Keywords: Learnability; Language acquisition; Algo

1. Introduction

Language acquisition involves the ra
linguistic structure of astonishing comple
input that appears noisy and partial. H
impoverished stimulus support such imp
One influential line of argument is tha
"poverty of the stimulus" argument (Cl
typically used to argue that language acq
by innate knowledge of language, often t
grammar," that the child brings to bear
problem (e.g., Chomsky, 1965, 1980; Ho
1988). This type of argument for univers
central importance for the study of hum
language acquisition (e.g., Crain & Lil
Hornstein & Lightfoot, 1981).
How can the poverty of the stimul
assessed? At an abstract level, a natura

INFORMATION AND CONTROL 7, 1–22 (1964)

A Formal Theory of Inductive Inference. Part I*†

R. J. Solomonoff

Rockford Research Institute, Inc., Cambridge, Massachusetts

1. S

In Part I, four ostensibly diff
are presented, in which the proble
very long sequence of symbols—p
mation to be used in the inducti
induction can be put in this form.
Some strong heuristic argumen
lence of the last three models. O
Bayes formulation, in which a p
quences of symbols on the basis o
Turing machine that are required
output.
Though it seems likely, it is no
models is equivalent to the other
Few rigorous results are presen
of the properties of these models
sistency and meaningfulness, of
exact nature of the Turing machi
predictions in comparison to those
In Part II these models are appl

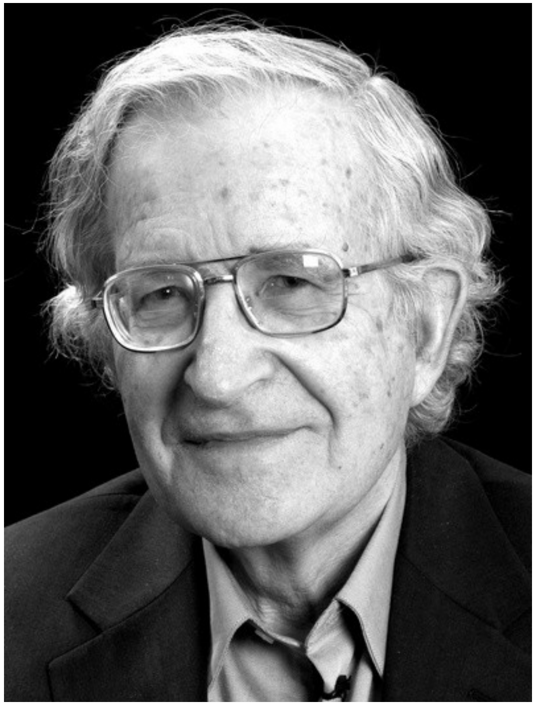
* This research was supported by A
tract No. AF 49(638)-376, Grant No. A
NIH Grant No. GM 11021-01.
† A paper given at the Conferenc
held at the California Institute of T
subject of Zator Technical Bulletin N
3.1 to 3.4 first appeared in Zator Tec
1960 and January 1961, respectively. S
tations of Zator Technical Bulletins
respectively.

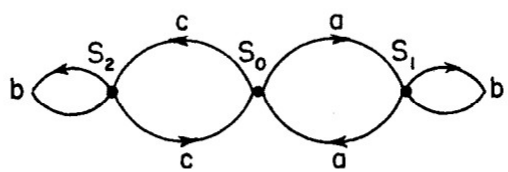
Outline

- **Overview of learnability & formal languages**

- **Learning model**
  - Simple formal languages
  - Artificial language learning
  - Simplified English CFG

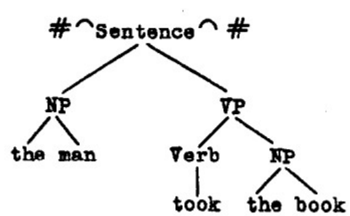
- **Three related lines of ongoing work**
  - Human experiments
  - Recursion in monkeys and human groups
  - Algorithm learning in indigenous Amazonians





$$\Sigma : \Sigma_1, \ldots, \Sigma_n$$

$$F: \quad X_1 \to Y_1$$
$$\vdots$$
$$X_m \to Y_m$$



$$t_p^*(Y_1, \ldots, Y_4) = Y_1 - Y_2{}^\frown be{}^\frown en - Y_3 - by{}^\frown Y_1$$

$$t_p^*(the{}^\frown man, past, eat, the{}^\frown food) =$$
$$the{}^\frown food - past{}^\frown be{}^\frown en - eat - by{}^\frown the{}^\frown man.$$

THREE MODELS FOR THE DESCRIPTION OF LANGUAGE*

Noam Chomsky
Department of Modern Languages and Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts

Abstract

We investigate several conceptions of linguistic structure to determine whether or not they can provide simple and "revealing" grammars that generate all of the sentences of English and only these. We find that no finite-state Markov process that produces symbols with transition from state to state can serve as an English grammar. Furthermore, the particular subclass of such processes that produce n-order statistical approximations to English do not come closer, with increasing n, to matching the output of an English grammar. We formalize the notions of "phrase structure" and show that this gives us a method for describing language which is essentially more powerful, though still representable as a rather elementary type of finite-state process. Nevertheless, it is successful only when limited to a small subset of simple sentences. We study the formal properties of a set of grammatical transformations that carry sentences with phrase structure into new sentences with derived phrase structure, showing that transformational grammars are processes of the same elementary type as phrase-structure grammars; that the grammar of English is materially simplified if phrase structure description is limited to a kernel of simple sentences from which all other sentences are constructed by repeated transformations; and that this view of linguistic structure gives a certain insight into the use and understanding of language.

1. Introduction

There are two central problems in the descriptive study of language. One primary concern of the linguist is to discover simple and "revealing" grammars for natural languages. At the same time, by studying the properties of such successful grammars and clarifying the basic conceptions that underlie them, he hopes to arrive at a general theory of linguistic structure. We shall examine certain features of these related inquiries.

The grammar of a language can be viewed as a theory of the structure of this language. Any scientific theory is based on a certain finite set of observations and, by establishing general laws stated in terms of certain hypothetical constructs, it attempts to account for these observations, to show how they are interrelated, and to predict an indefinite number of new phenomena. A mathematical theory has the additional property that predictions follow rigorously from the body of theory. Similarly, a grammar is based on a finite number of observed sentences (the linguist's corpus) and it "projects" this set to an infinite set of grammatical sentences by establishing general "laws" (grammatical rules) framed in terms of such hypothetical constructs as the particular phonemes, words, phrases, and so on, of the language under analysis. A properly formulated grammar should determine unambiguously the set of grammatical sentences.

General linguistic theory can be viewed as a metatheory which is concerned with the problem of how to choose such a grammar in the case of each particular language on the basis of a finite corpus of sentences. In particular, it will consider and attempt to explicate the relation between the set of grammatical sentences and the set of observed sentences. In other words, linguistic theory attempts to explain the ability of a speaker to produce and understand new sentences, and to reject as ungrammatical other new sequences, on the basis of his limited linguistic experience.

Suppose that for many languages there are certain clear cases of grammatical sentences and certain clear cases of ungrammatical sequences, e.g., (1) and (2), respectively, in English.

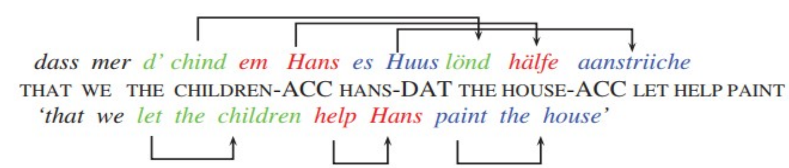
(1) John ate a sandwich
(2) Sandwich a ate John.

In this case, we can test the adequacy of a proposed linguistic theory by determining, for each language, whether or not the clear cases are handled properly by the grammars constructed in accordance with this theory. For example, if a large corpus of English does not happen to contain either (1) or (2), we ask whether the grammar that is determined for this corpus will project the corpus to include (1) and exclude (2). Even though such clear cases may provide only a weak test of adequacy for the grammar of a given language taken in isolation, they provide a very strong test for any general linguistic theory and for the set of grammars to which it leads, since we insist that in the case of each language the clear cases be handled properly in a fixed and predetermined manner. We can take certain steps towards the construction of an operational characterization of "grammatical sentence" that will provide us with the clear cases required to set the task of linguistics significantly.

*This work was supported in part by the Army (Signal Corps), the Air Force (Office of Scientific Research, Air Research and Development Command), and the Navy (Office of Naval Research), and in part by a grant from Eastman Kodak Company.

Formal and natural languages

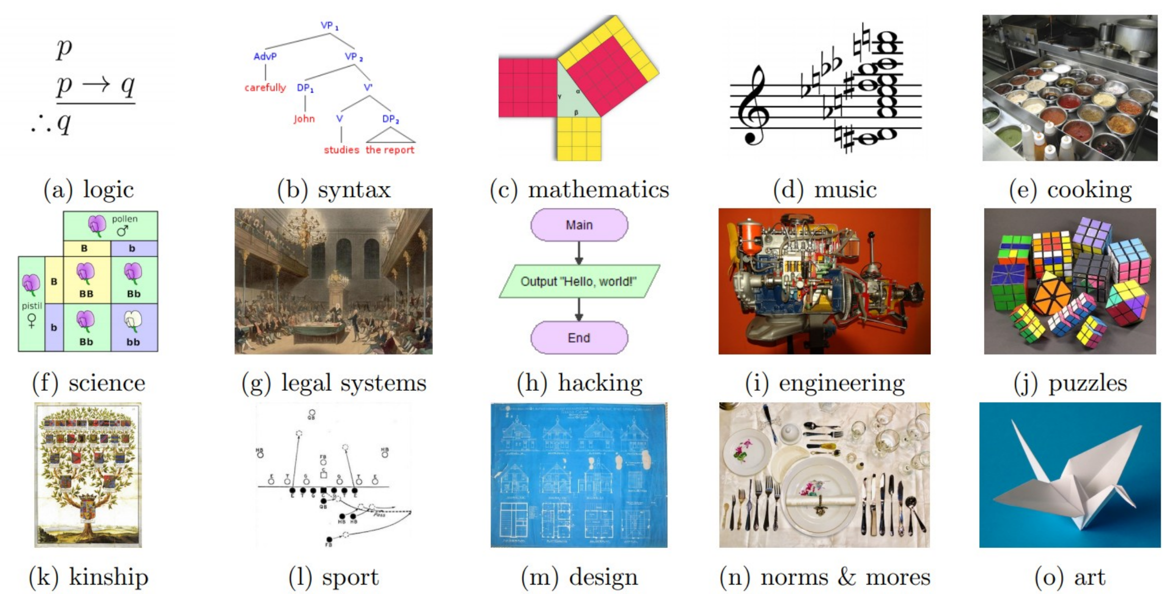
- **Patterns in natural language correspond to different formal languages & require distinct computational resources** (see Jäger & Rogers 2012)
  - $\mathbf{a^n}$ – {a,aa,aaa,aaaa,….}
    The tall, angry, young, giraffe …
  - $\mathbf{(ab)^n}$ – {ab, abab, ababab, …}
    Bring two boats, three cups, six accordions, …
  - $\mathbf{a^n b^n}$ – {ab, aabb, aaabbb, …}
    If If Ted cried then John was sad then John is empathetic.
  - **w w** – {abcabc, accbaaccba, bbbb, ...}
  - $\mathbf{a^n b^m c^n d^m}$ – {abcd, abbcdd, aabccd, …}



- **These examples face all the problems we started with** – subset problem, infinite productivity, gold non-learnability, etc.

Working hypothesis

- **Learning operates over a Turing-complete space**
  - **Learning is like programming** – learners combine existing operations in new ways to form generative models of data
  - **More input data** drives revision, improvement of programs, justifying additional complexity.



Trends in Cognitive Sciences

CellPress REVIEWS

Opinion

The Child as Hacker

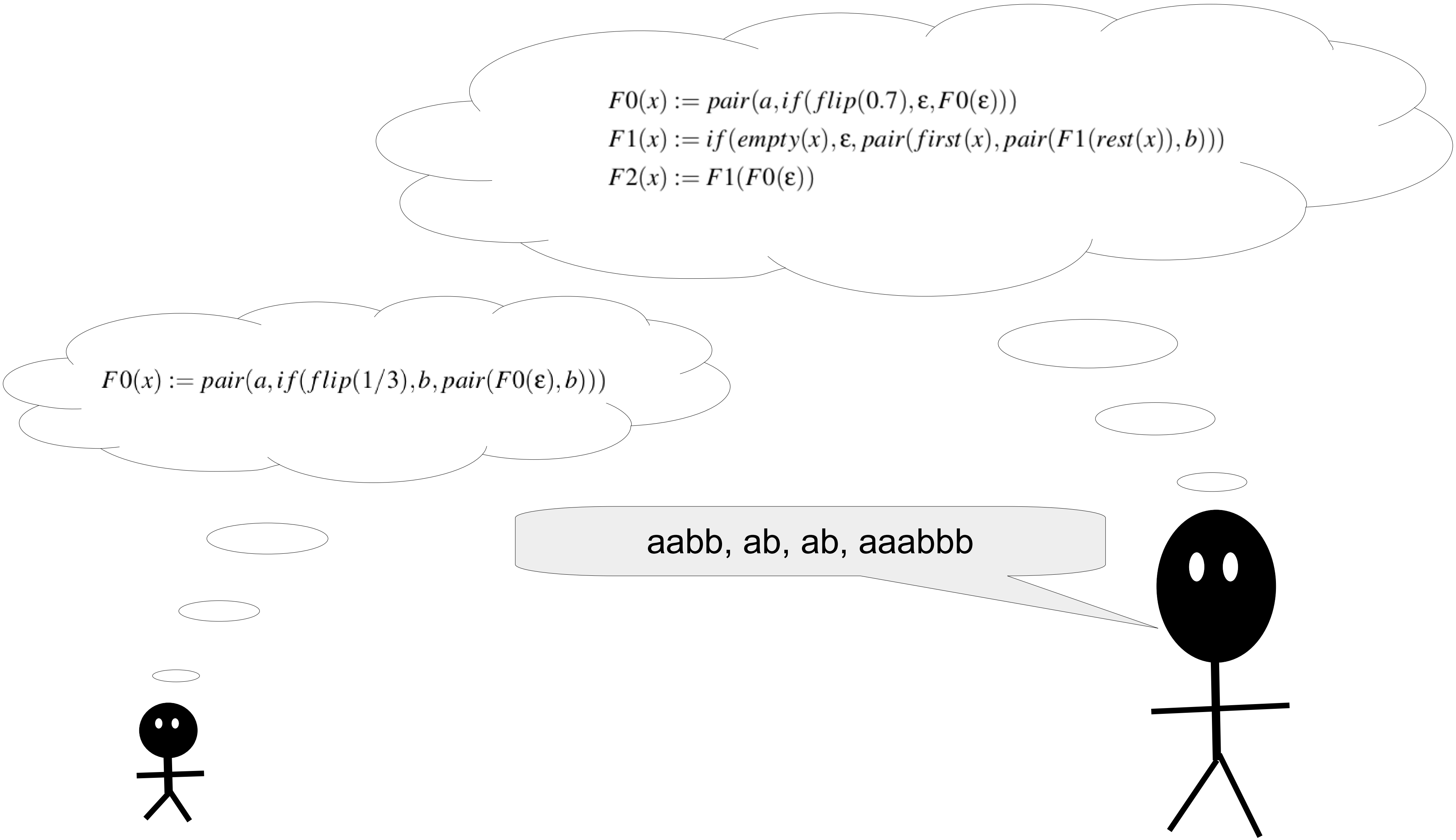
Joshua S. Rule,[1,*] Joshua B. Tenenbaum,[1] and Steven T. Piantadosi[2]

The scope of human learning and development poses a radical challenge for cognitive science. We propose that developmental theories can address this challenge by adopting perspectives from computer science. Many of our best models treat learning as analogous to computer programming because symbolic programs provide the most compelling account of sophisticated mental representations. We specifically propose that children's learning is analogous to a particular style of programming called hacking, making code better along many dimensions through an open-ended set of goals and activities. By contrast to existing theories, which depend primarily on local search and simple metrics, this view highlights the many features of good mental representations and the multiple complementary processes children use to create them.

Highlights

Hacking as a Metaphor for Learning in Cognitive Development

Cartoon of learning setup



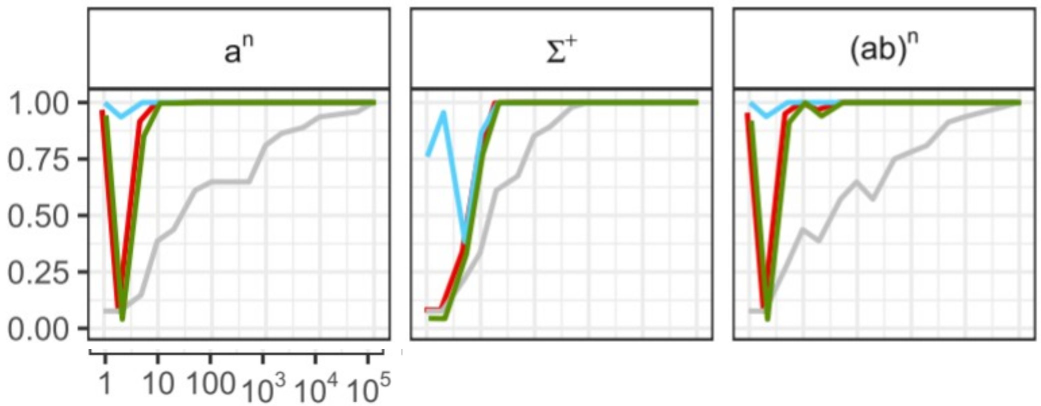
Outline

- **Overview of learnability & formal languages**

- **Learning model**
  - Simple formal languages
  - Artificial language learning
  - Simplified English CFG

- **Three related lines of ongoing work**
  - Human experiments
  - Recursion in monkeys and human groups
  - Algorithm learning in indigenous Amazonians

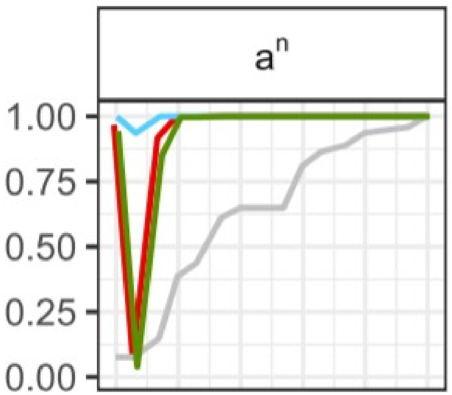
Learning generative programs can be fast and easy



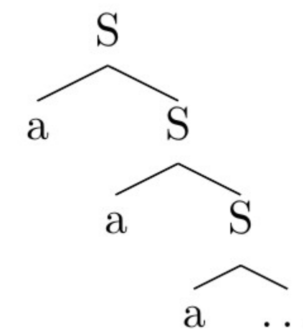
(approximated on top strings)

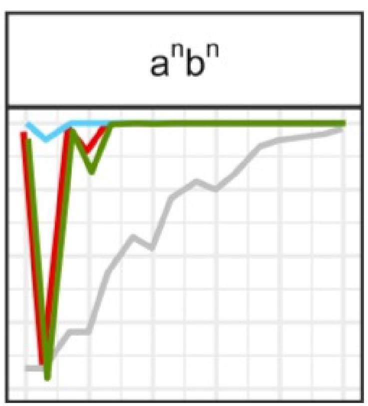
The model builds distinct generative models

Learning curve	Learned program	Structure

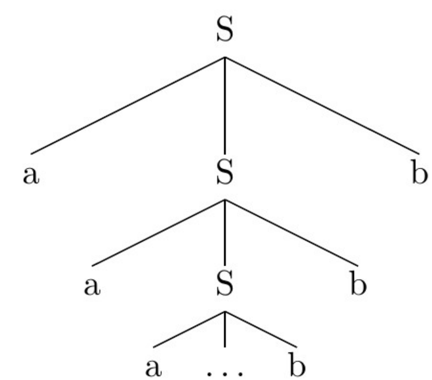



```
def F():
    if flip():
        ”a”
    else:
        return F()+”a”
```

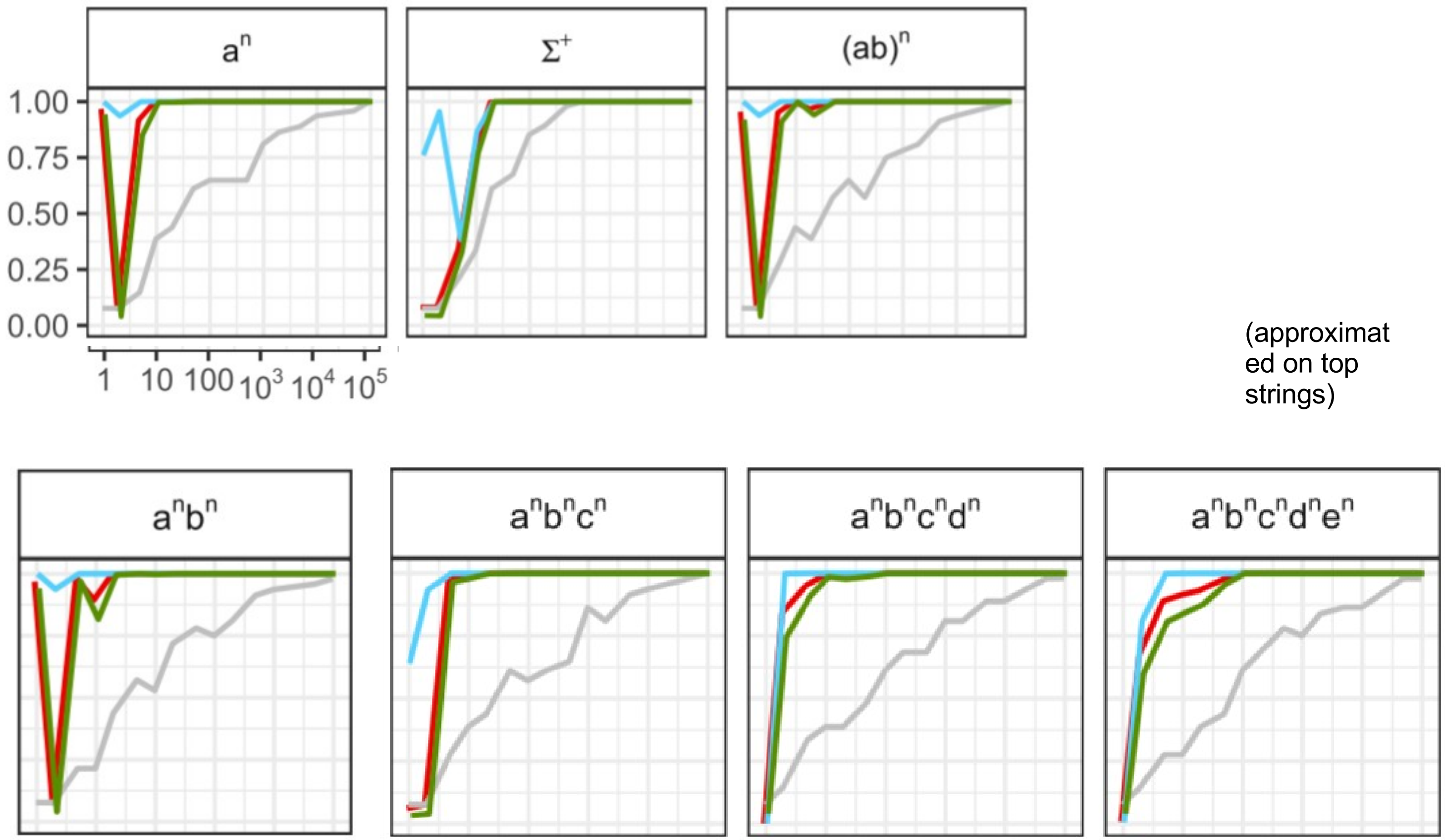




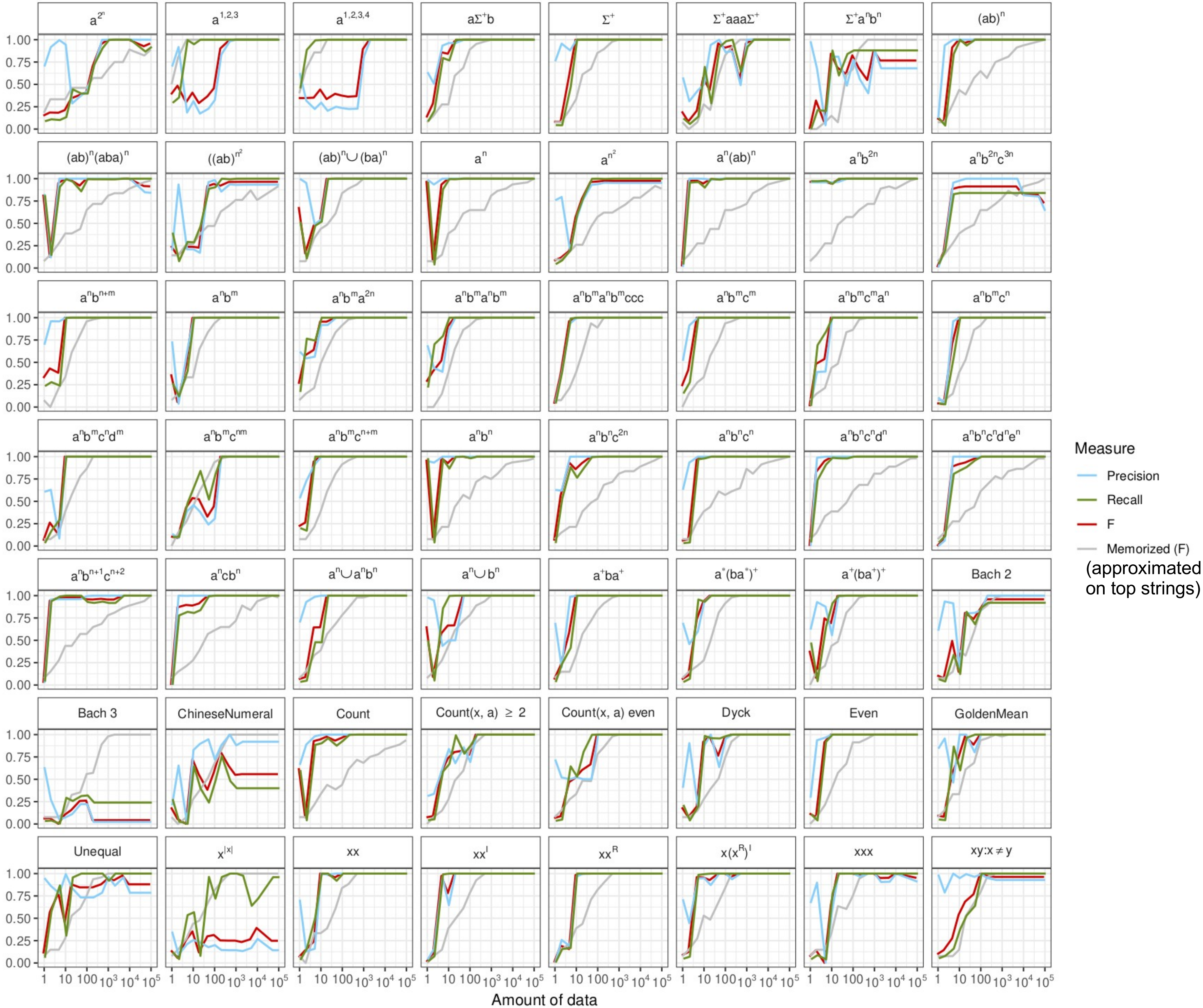

```
def F():
    if flip():
        ”ab”
    else:
        return “a”+F()+”b”
```



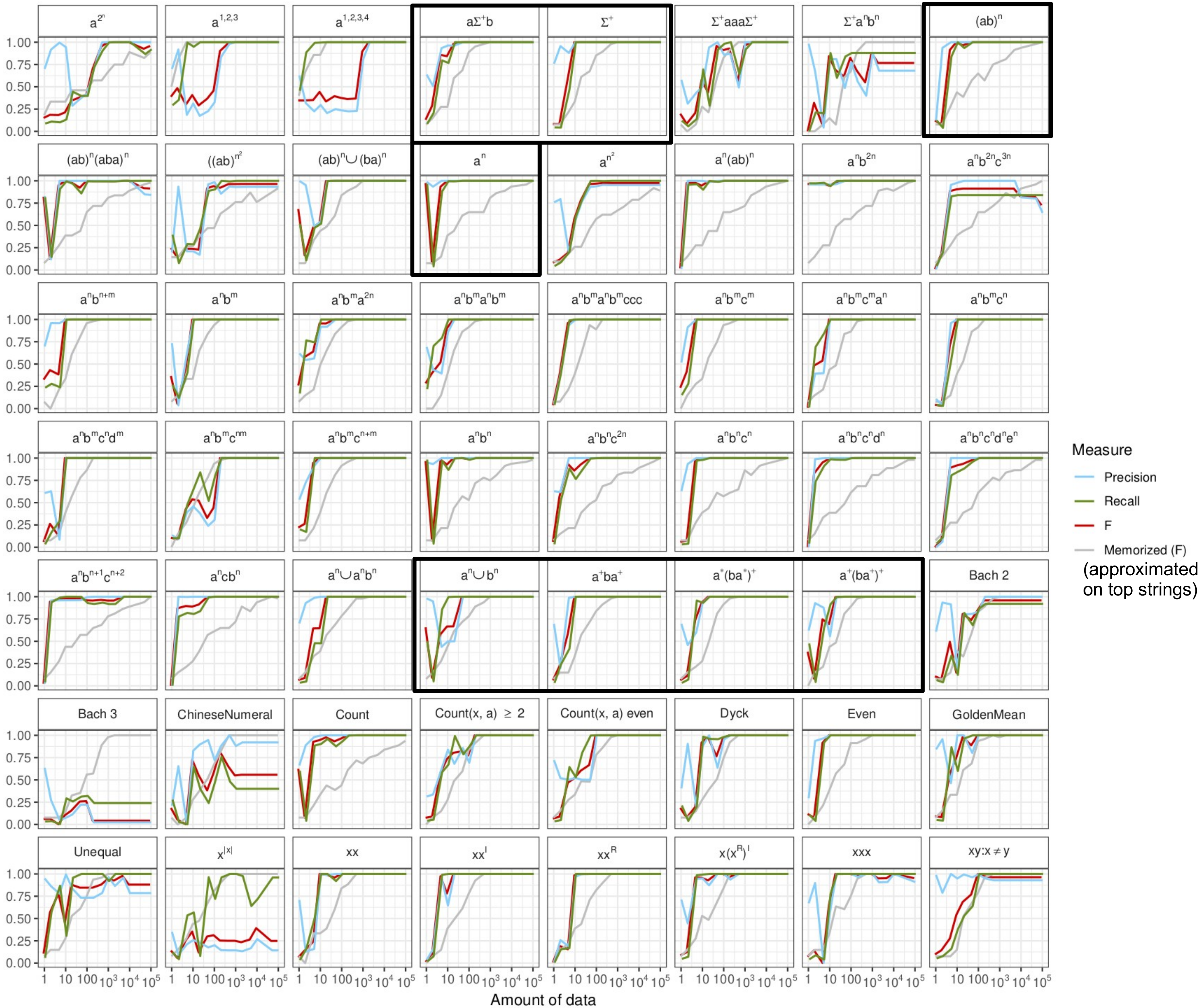
Learning generative programs can be fast and easy



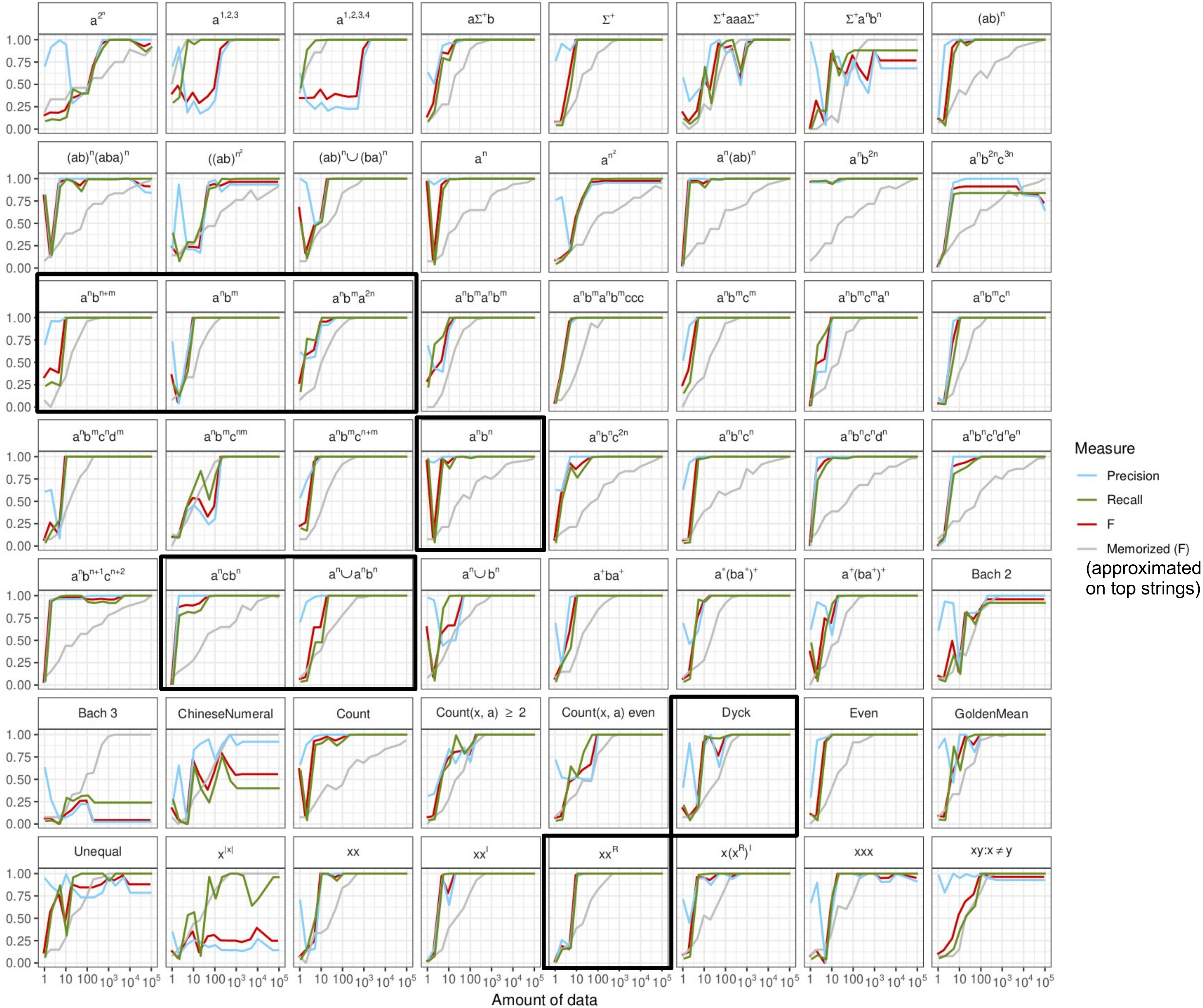
(approximat
ed on top
strings)



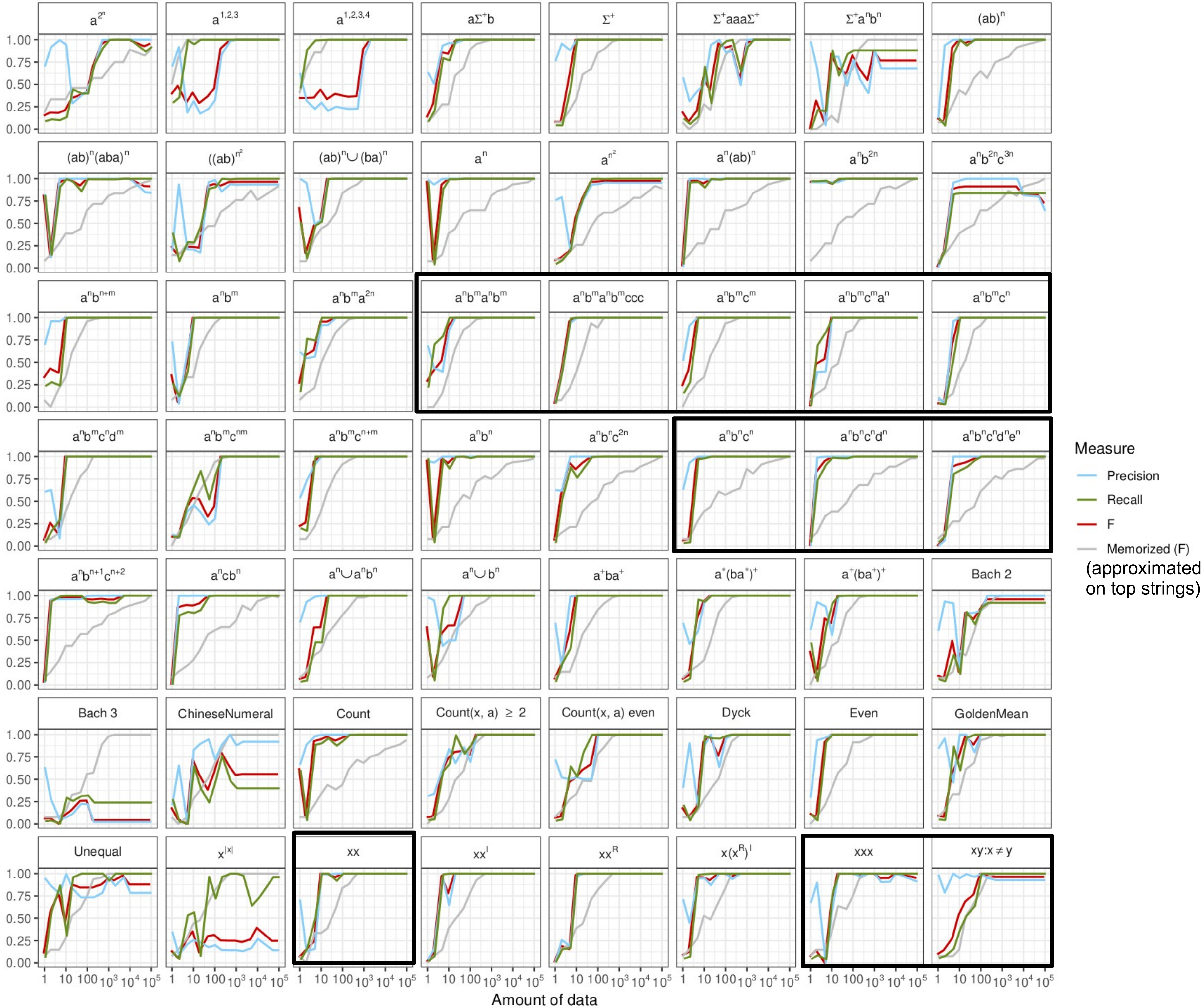
a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^+b$
Σ^+
$\Sigma^+aaa\Sigma^+$
$\Sigma^+a^nb^n$
$(ab)^n$
$(ab)^n(aba)^n$
$((ab)^{n^2}$
$(ab)^n\cup(ba)^n$
a^n
a^{n^2}
$a^n(ab)^n$
a^nb^{2n}
$a^nb^{2n}c^{3n}$
a^nb^{n+m}
a^nb^m
$a^nb^ma^{2n}$
$a^nb^ma^nb^m$
$a^nb^ma^nb^mccc$
$a^nb^mc^m$
$a^nb^mc^ma^n$
$a^nb^mc^n$
$a^nb^mc^nd^m$
$a^nb^mc^{nm}$
$a^nb^mc^{n+m}$
a^nb^n
$a^nb^nc^{2n}$
$a^nb^nc^n$
$a^nb^nc^nd^n$
$a^nb^nc^nd^ne^n$
$a^nb^{n+1}c^{n+2}$
a^ncb^n
$a^n\cup a^nb^n$
$a^n\cup b^n$
a^+ba^+
$a^*(ba^*)^+$
$a^+(ba^+)^+$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) $\geq$ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^I
xx^R
$x(x^R)^I$
xxx
$xy{:}x\neq y$
0.00 0.25 0.50 0.75 1.00
1 10 100 10^3 10^4 10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)



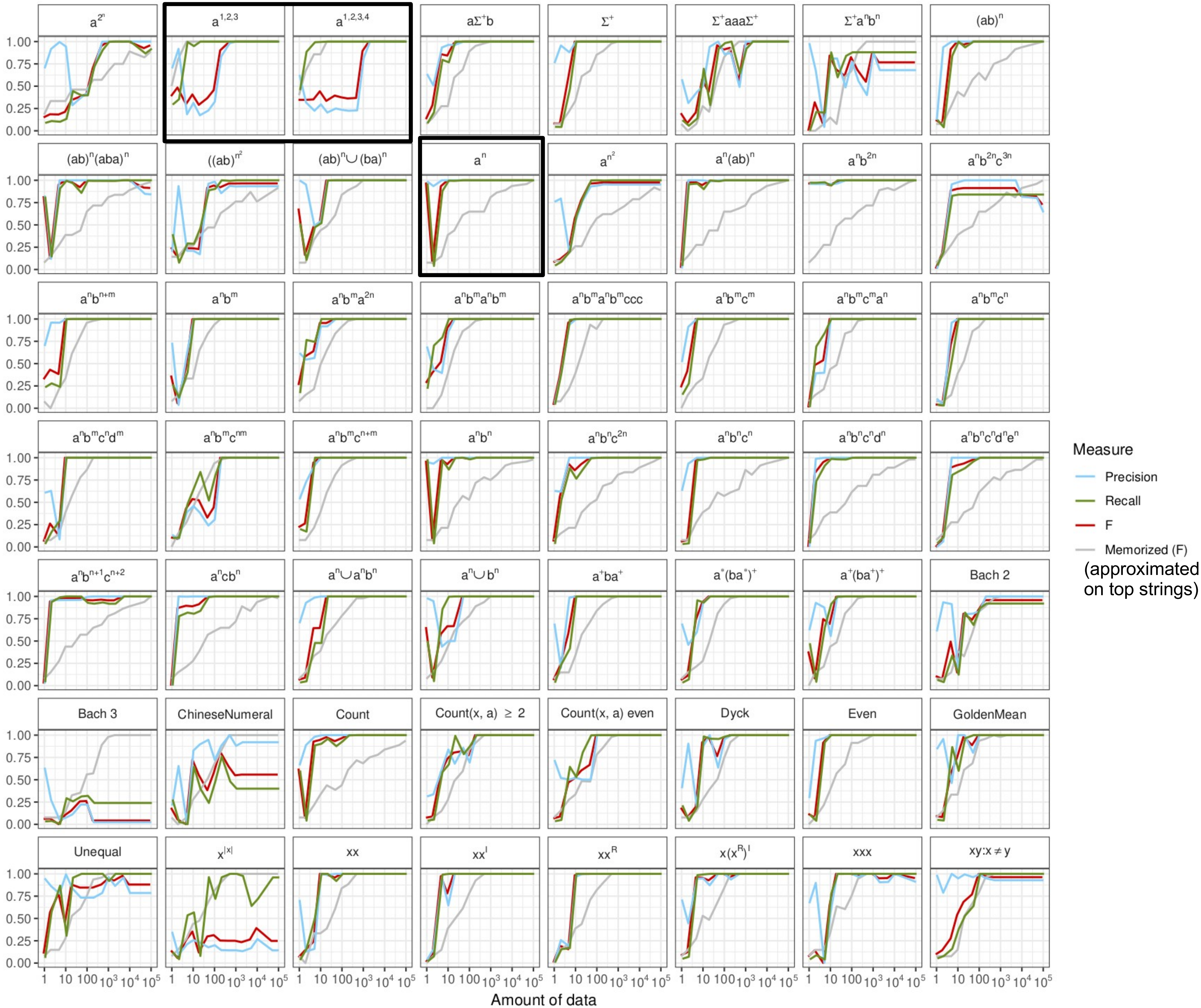
a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^+b$
Σ^+
$\Sigma^+aaa\Sigma^+$
$\Sigma^+a^nb^n$
$(ab)^n$
$(ab)^n(aba)^n$
$((ab)^n)^2$
$(ab)^n \cup (ba)^n$
a^n
a^{n^2}
$a^n(ab)^n$
a^nb^{2n}
$a^nb^{2n}c^{3n}$
a^nb^{n+m}
a^nb^m
$a^nb^ma^{2n}$
$a^nb^ma^nb^m$
$a^nb^ma^nb^mccc$
$a^nb^mc^m$
$a^nb^mc^ma^n$
$a^nb^mc^n$
$a^nb^mc^nd^m$
$a^nb^mc^{nm}$
$a^nb^mc^{n+m}$
a^nb^n
$a^nb^nc^{2n}$
$a^nb^nc^n$
$a^nb^nc^nd^n$
$a^nb^nc^nd^ne^n$
$a^nb^{n+1}c^{n+2}$
a^ncb^n
$a^n \cup a^nb^n$
$a^n \cup b^n$
a^+ba^+
$a^*(ba^*)^+$
$a^+(ba^+)^+$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) $\geq$ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^I
xx^R
$x(x^R)^I$
xxx
$xy : x \neq y$
0.00
0.25
0.50
0.75
1.00
1
10
100
10^3
10^4
10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)



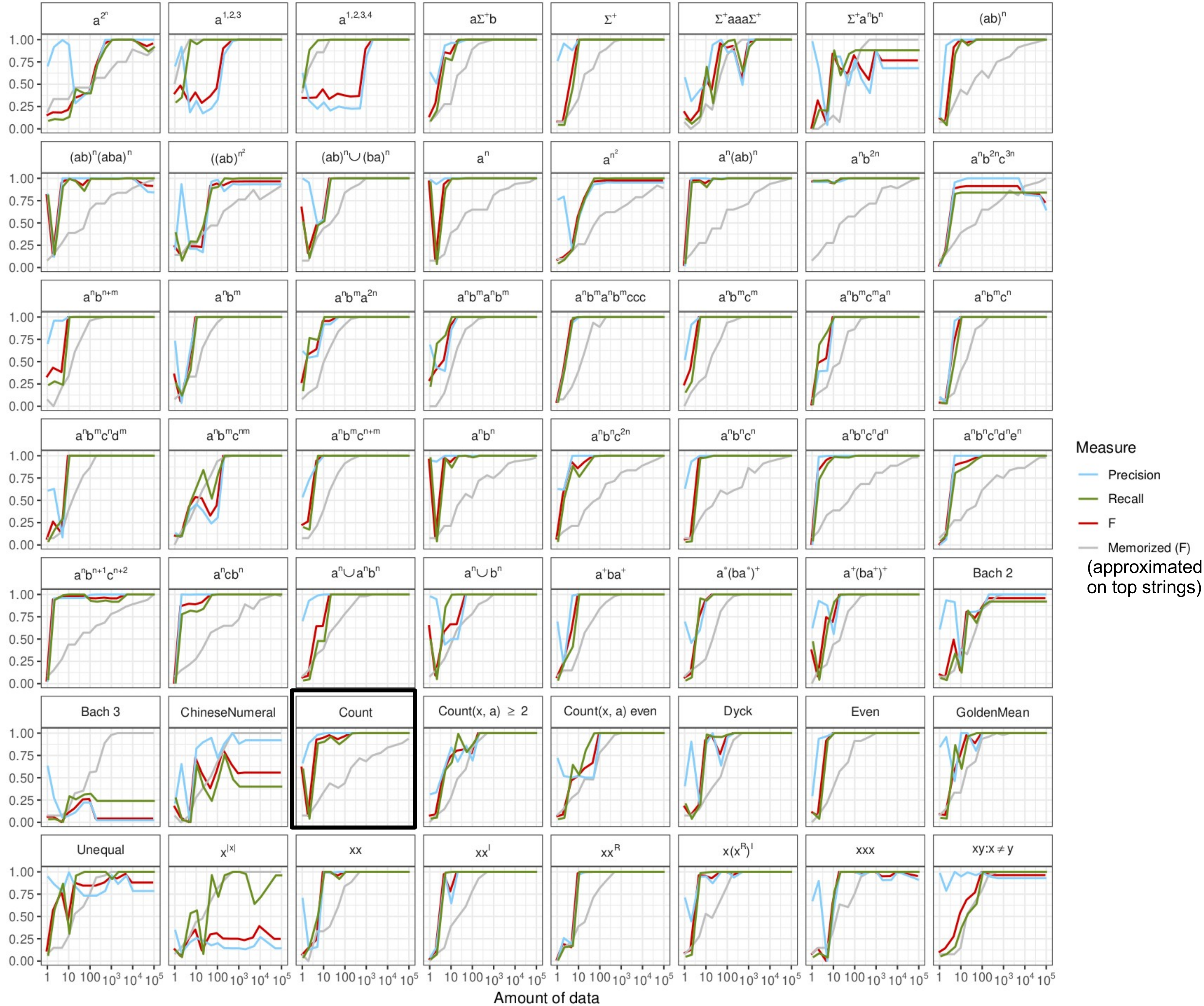
a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^+b$
Σ^+
$\Sigma^+aaa\Sigma^+$
$\Sigma^+a^nb^n$
$(ab)^n$
$(ab)^n(aba)^n$
$((ab)^{n^2}$
$(ab)^n \cup (ba)^n$
a^n
a^{n^2}
$a^n(ab)^n$
a^nb^{2n}
$a^nb^{2n}c^{3n}$
a^nb^{n+m}
a^nb^m
$a^nb^ma^{2n}$
$a^nb^ma^nb^m$
$a^nb^ma^nb^mccc$
$a^nb^mc^m$
$a^nb^mc^ma^n$
$a^nb^mc^n$
$a^nb^mc^nd^m$
$a^nb^mc^{nm}$
$a^nb^mc^{n+m}$
a^nb^n
$a^nb^nc^{2n}$
$a^nb^nc^n$
$a^nb^nc^nd^n$
$a^nb^nc^nd^ne^n$
$a^nb^{n+1}c^{n+2}$
a^ncb^n
$a^n \cup a^nb^n$
$a^n \cup b^n$
a^+ba^+
$a^*(ba^*)^+$
$a^+(ba^+)^+$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) $\geq$ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^I
xx^R
$x(x^R)^I$
xxx
$xy{:}x \neq y$
0.00
0.25
0.50
0.75
1.00
1
10
100
10^3
10^4
10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)



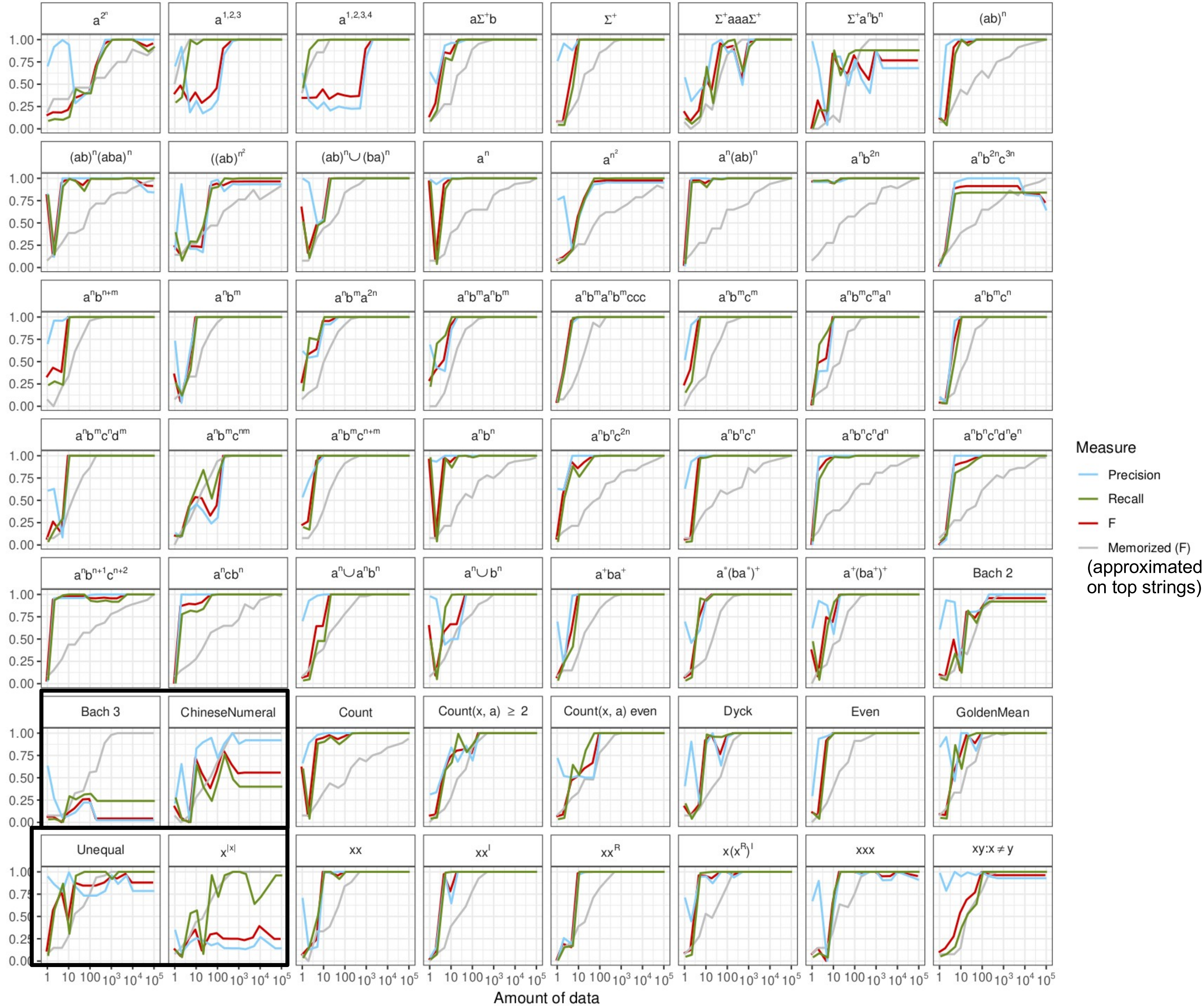
a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^+b$
Σ^+
$\Sigma^+aaa\Sigma^+$
$\Sigma^+a^nb^n$
$(ab)^n$
$(ab)^n(aba)^n$
$((ab)^n)^2$
$(ab)^n \cup (ba)^n$
a^n
a^{n^2}
$a^n(ab)^n$
a^nb^{2n}
$a^nb^{2n}c^{3n}$
a^nb^{n+m}
a^nb^m
$a^nb^ma^{2n}$
$a^nb^ma^nb^m$
$a^nb^ma^nb^mccc$
$a^nb^mc^m$
$a^nb^mc^ma^n$
$a^nb^mc^n$
$a^nb^mc^nd^m$
$a^nb^mc^{nm}$
$a^nb^mc^{n+m}$
a^nb^n
$a^nb^nc^{2n}$
$a^nb^nc^n$
$a^nb^nc^nd^n$
$a^nb^nc^nd^ne^n$
$a^nb^{n+1}c^{n+2}$
a^ncb^n
$a^n \cup a^nb^n$
$a^n \cup b^n$
a^+ba^+
$a^*(ba^*)^+$
$a^+(ba^+)^+$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) $\geq$ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^I
xx^R
$x(x^R)^I$
xxx
xy:x $\neq$ y
0.00
0.25
0.50
0.75
1.00
1
10
100
10^3
10^4
10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated
on top strings)



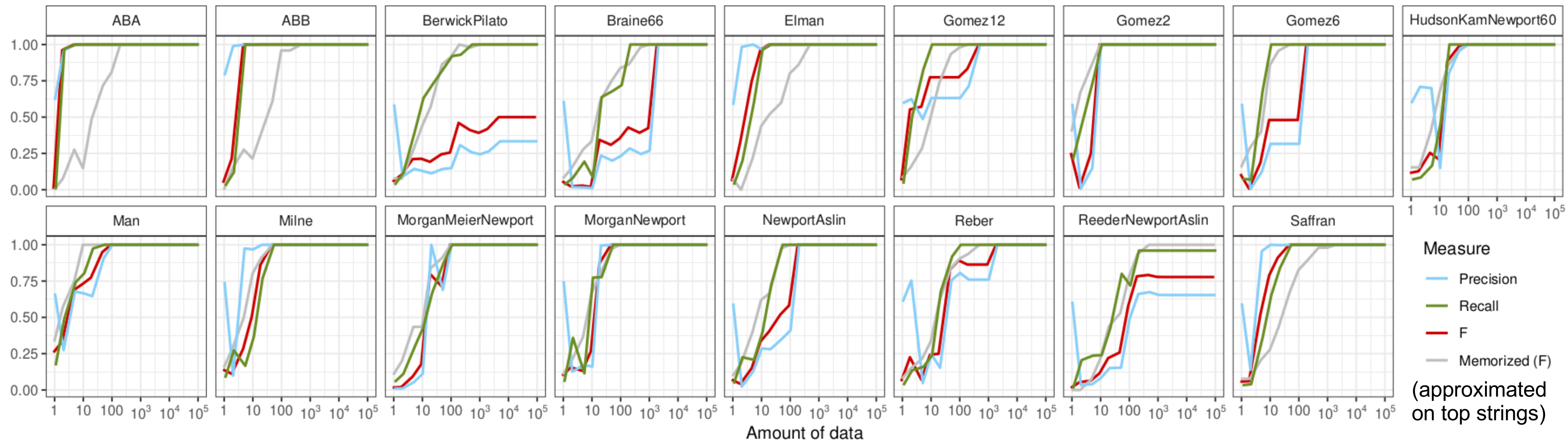
a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^+b$
Σ^+
$\Sigma^+aaa\Sigma^+$
$\Sigma^+a^nb^n$
$(ab)^n$
$(ab)^n(aba)^n$
$((ab)^{n^2}$
$(ab)^n\cup(ba)^n$
a^n
a^{n^2}
$a^n(ab)^n$
a^nb^{2n}
$a^nb^{2n}c^{3n}$
a^nb^{n+m}
a^nb^m
$a^nb^ma^{2n}$
$a^nb^ma^nb^m$
$a^nb^ma^nb^mccc$
$a^nb^mc^m$
$a^nb^mc^ma^n$
$a^nb^mc^n$
$a^nb^mc^nd^m$
$a^nb^mc^{nm}$
$a^nb^mc^{n+m}$
a^nb^n
$a^nb^nc^{2n}$
$a^nb^nc^n$
$a^nb^nc^nd^n$
$a^nb^nc^nd^ne^n$
$a^nb^{n+1}c^{n+2}$
a^ncb^n
$a^n\cup a^nb^n$
$a^n\cup b^n$
a^+ba^+
$a^*(ba^*)^+$
$a^+(ba^+)^+$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) ≥ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^I
xx^R
$x(x^R)^I$
xxx
$xy{:}x\neq y$
1.00
0.75
0.50
0.25
0.00
1 10 100 10^3 10^4 10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)

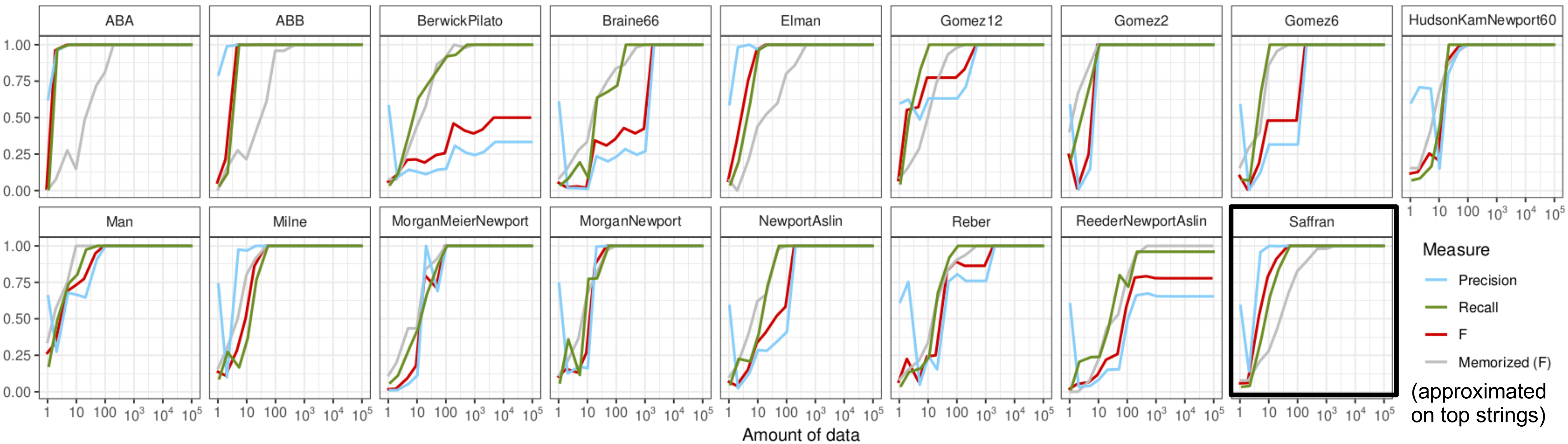


a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^+b$
Σ^+
$\Sigma^+aaa\Sigma^+$
$\Sigma^+a^nb^n$
$(ab)^n$
$(ab)^n(aba)^n$
$((ab)^{n^2}$
$(ab)^n \cup (ba)^n$
a^n
a^{n^2}
$a^n(ab)^n$
a^nb^{2n}
$a^nb^{2n}c^{3n}$
a^nb^{n+m}
a^nb^m
$a^nb^ma^{2n}$
$a^nb^ma^nb^m$
$a^nb^ma^nb^mccc$
$a^nb^mc^m$
$a^nb^mc^ma^n$
$a^nb^mc^n$
$a^nb^mc^nd^m$
$a^nb^mc^{nm}$
$a^nb^mc^{n+m}$
a^nb^n
$a^nb^nc^{2n}$
$a^nb^nc^n$
$a^nb^nc^nd^n$
$a^nb^nc^nd^ne^n$
$a^nb^{n+1}c^{n+2}$
a^ncb^n
$a^n \cup a^nb^n$
$a^n \cup b^n$
a^+ba^+
$a^*(ba^*)^+$
$a^+(ba^+)^+$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) $\geq$ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^I
xx^R
$x(x^R)^I$
xxx
$xy{:}x \neq y$
0.00
0.25
0.50
0.75
1.00
1
10
100
10^3
10^4
10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)



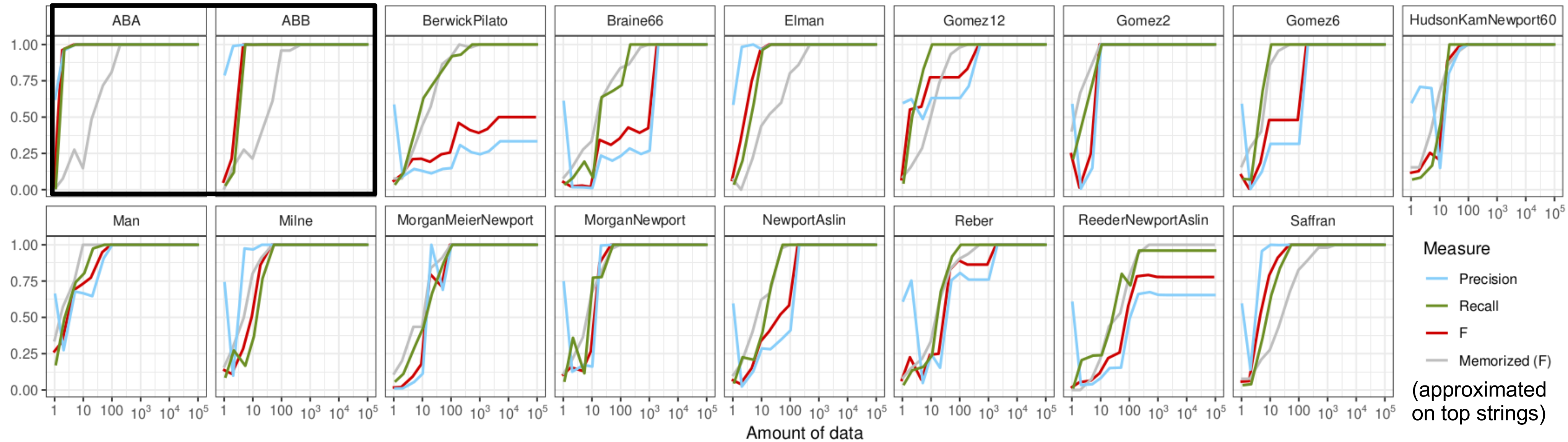
a^{2^n}
$a^{1,2,3}$
$a^{1,2,3,4}$
$a\Sigma^{+}b$
Σ^{+}
$\Sigma^{+}aaa\Sigma^{+}$
$\Sigma^{+}a^{n}b^{n}$
$(ab)^{n}$
$(ab)^{n}(aba)^{n}$
$((ab)^{n})^{2}$
$(ab)^{n} \cup (ba)^{n}$
a^{n}
a^{n^2}
$a^{n}(ab)^{n}$
$a^{n}b^{2n}$
$a^{n}b^{2n}c^{3n}$
$a^{n}b^{n+m}$
$a^{n}b^{m}$
$a^{n}b^{m}a^{2n}$
$a^{n}b^{m}a^{n}b^{m}$
$a^{n}b^{m}a^{n}b^{m}ccc$
$a^{n}b^{m}c^{m}$
$a^{n}b^{m}c^{m}a^{n}$
$a^{n}b^{m}c^{n}$
$a^{n}b^{m}c^{n}d^{m}$
$a^{n}b^{m}c^{nm}$
$a^{n}b^{m}c^{n+m}$
$a^{n}b^{n}$
$a^{n}b^{n}c^{2n}$
$a^{n}b^{n}c^{n}$
$a^{n}b^{n}c^{n}d^{n}$
$a^{n}b^{n}c^{n}d^{n}e^{n}$
$a^{n}b^{n+1}c^{n+2}$
$a^{n}cb^{n}$
$a^{n} \cup a^{n}b^{n}$
$a^{n} \cup b^{n}$
$a^{+}ba^{+}$
$a^{*}(ba^{*})^{+}$
$a^{+}(ba^{+})^{+}$
Bach 2
Bach 3
ChineseNumeral
Count
Count(x, a) $\geq$ 2
Count(x, a) even
Dyck
Even
GoldenMean
Unequal
$x^{|x|}$
xx
xx^{I}
xx^{R}
$x(x^{R})^{I}$
xxx
$xy : x \neq y$
0.00
0.25
0.50
0.75
1.00
1
10
100
10^3
10^4
10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)


ABA
ABB
BerwickPilato
Braine66
Elman
Gomez12
Gomez2
Gomez6
HudsonKamNewport60
Man
Milne
MorganMeierNewport
MorganNewport
NewportAslin
Reber
ReederNewportAslin
Saffran
0.00
0.25
0.50
0.75
1.00
1
10
100
10^3
10^4
10^5
Amount of data
Measure
Precision
Recall
F
Memorized (F)
(approximated on top strings)



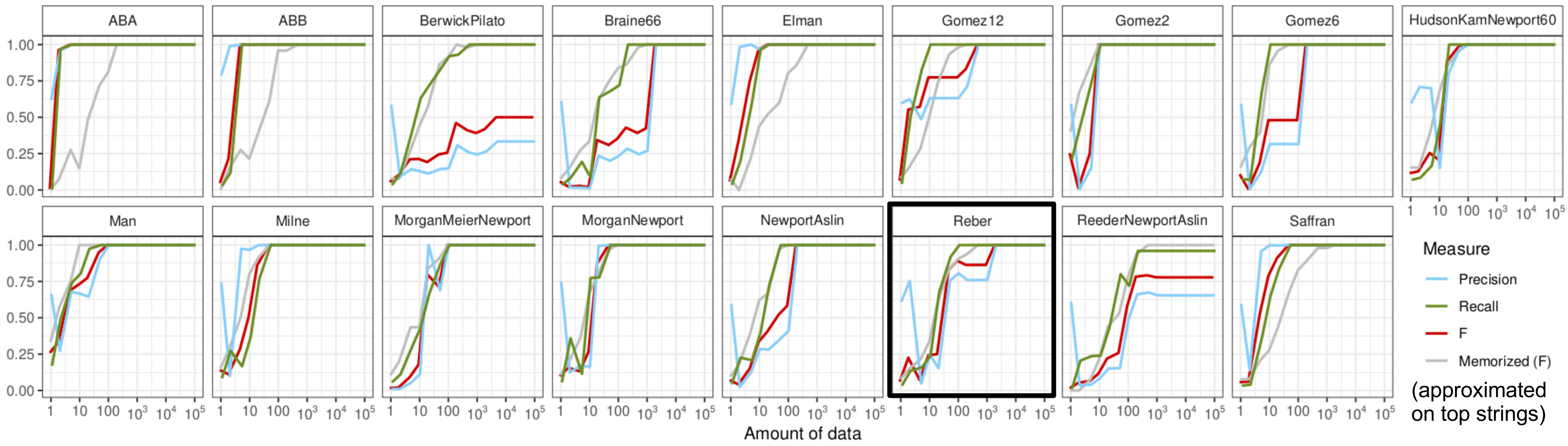
tupirogolabubidakupadotigolabubidakutupiropadoti

tupirogolabubidakupadotigolabubidakutupiropadoti

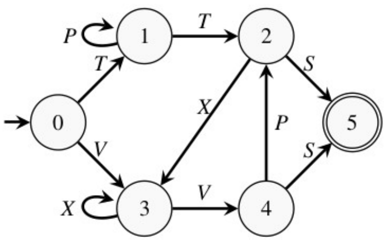


tupirogolabubidakupadotigolabubidakutupiropadoti

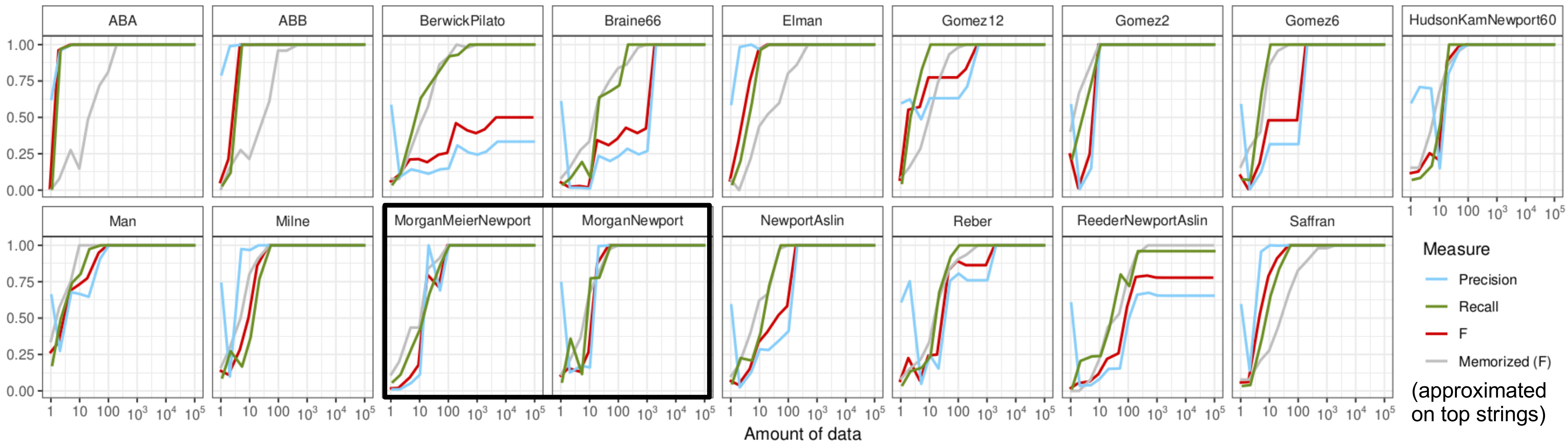
tupirogolabubidakupadotigolabubidakutupiropadoti



tupirogolabubidakupadotigolabubidakutupiropadoti
tupirogolabubidakupadotigolabubidakutupiropadoti

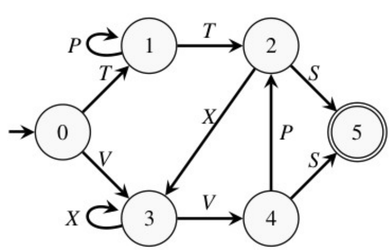


Reber



tupirogolabubidakupadotigolabubidakutupiropadoti

tupirogolabubidakupadotigolabubidakutupiropadoti



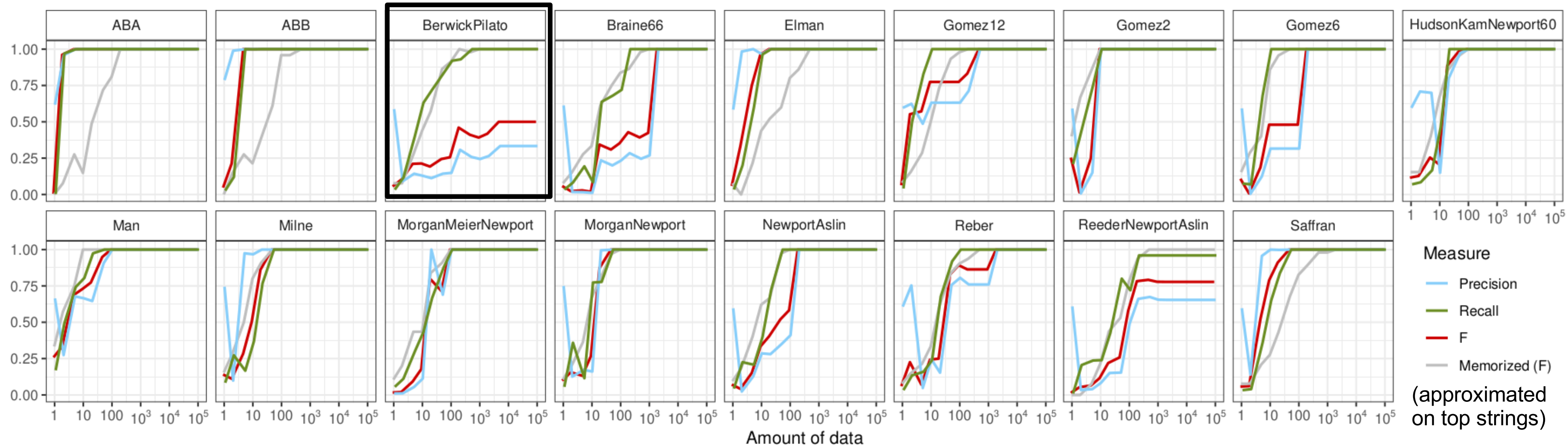
Reber

Morgan & Newport

$$S \to AP\ BP\ (CP)$$
$$AP \to a\ (D)$$
$$BP \to CP\ f \mid e$$
$$CP \to c\ (g)$$

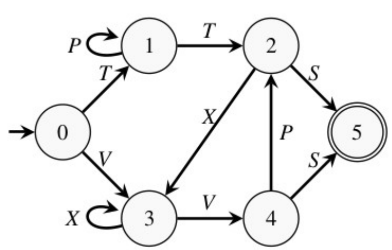
Morgan, Meier, & Newport

$$S \to AP\ BP\ (CP)$$
$$AP \to o\ a\ (d)$$
$$BP \to a\ CP\ f \mid u\ e$$
$$CP \to i\ c\ (g).$$

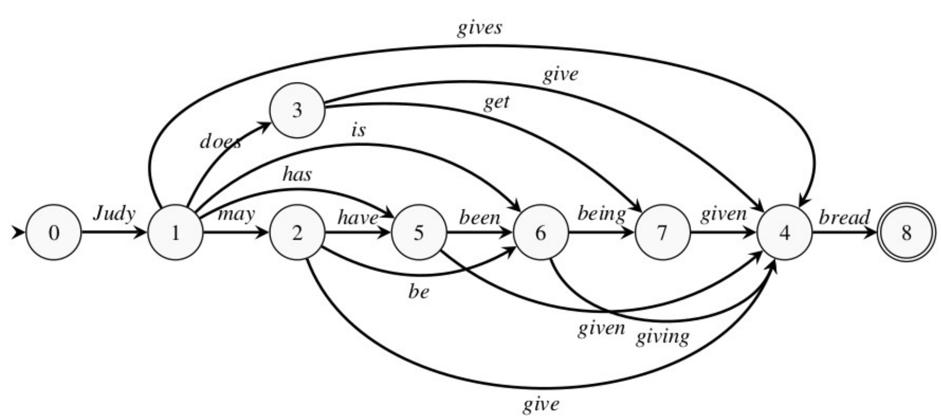


tupirogolabubidakupadotigolabubidakutupiropadoti

tupiro golabu bidaku padoti golabu bidaku tupiro padoti



Reber



Berwick & Pilato

Morgan & Newport

$S \rightarrow AP\ BP\ (CP)$

$AP \rightarrow a\ (D)$

$BP \rightarrow CP\ f \mid e$

$CP \rightarrow c\ (g)$

Morgan, Meier, & Newport

$S \rightarrow AP\ BP\ (CP)$

$AP \rightarrow o\ a\ (d)$

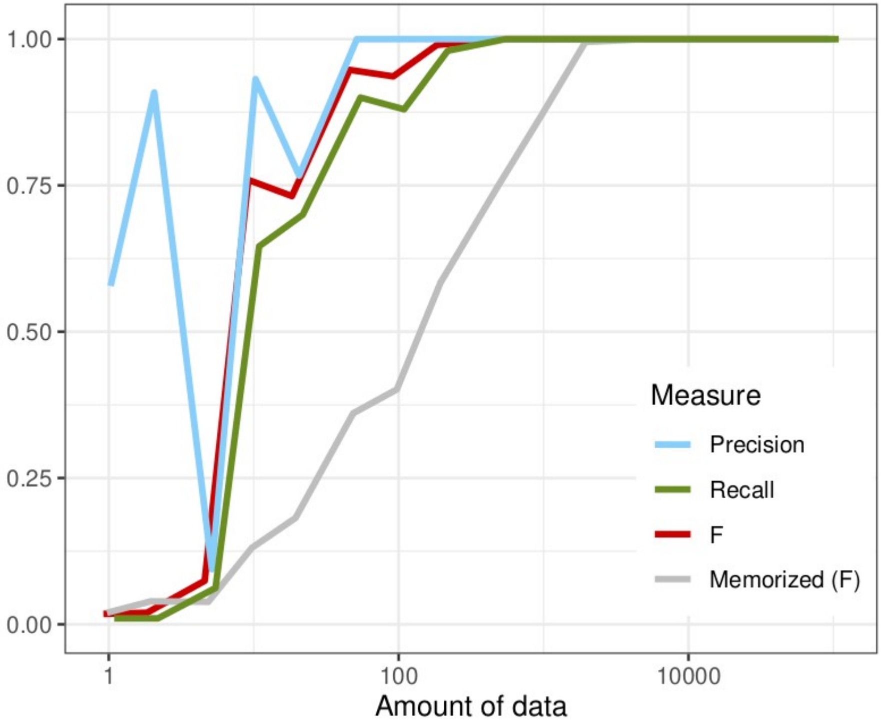
$BP \rightarrow a\ CP\ f \mid u\ e$

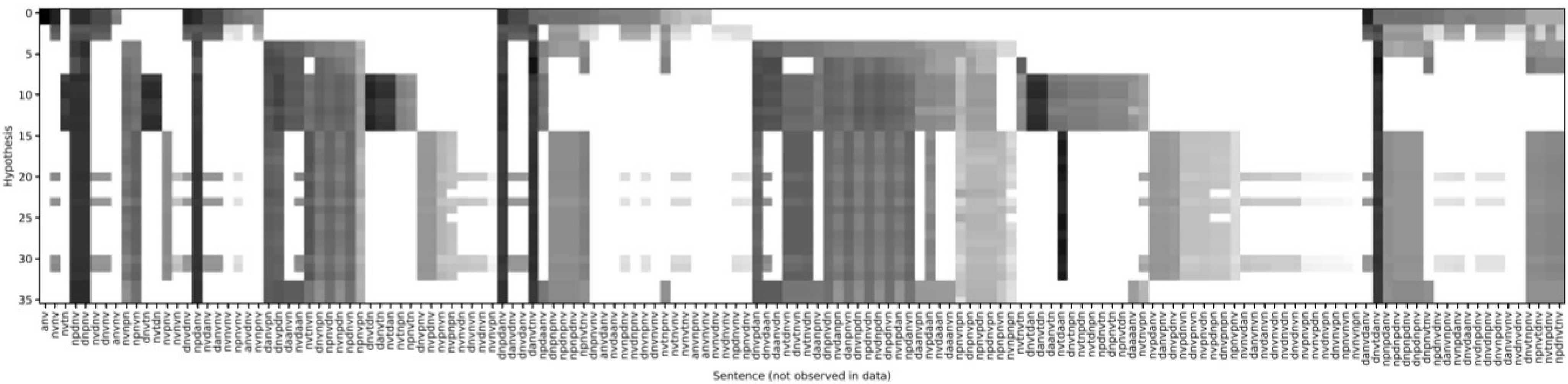
$CP \rightarrow i\ c\ (g).$

Moving toward natural language

- **Let's give the learning model data from a LING-101 CFG,** including a few kinds of structures – linear dependencies, tail recursion in AP, recursion in S, PP, etc.

$$S \rightarrow NP\,VP$$
$$NP \rightarrow n \mid d\,n \mid d\,AP\,n \mid NP\,PP$$
$$AP \rightarrow a \mid a\,AP$$
$$VP \rightarrow v \mid v\,NP \mid v\,t\,S \mid VP\,PP$$
$$PP \rightarrow p\,NP$$





Hypothesis
Sentence (not observed in data)

Learning is *much* more powerful than POS has claimed

- **Hierarchical structure**
  (Perfors, Tenenbaum & Regier 2011)
- **Language identification**
  (Chater & Vitanyi 2007, Yang & Piantadosi 2022)
- **Phonology textbook problems**
  (Ellis 2020)
- **Compositional semantics**
  (Kwiatkowski et al. 2010)
- **Island constraints**
  (Wilcox, Futrell, Levy 2021)
- **Linguistic features, structural generalizations**
  (Warstadt et al. 2020, Warstadt & Bowman 2020)
- **Binding theory / c-command (programs on trees)**
  (Gorensten & Piantadosi, in prep)

Modeling summary

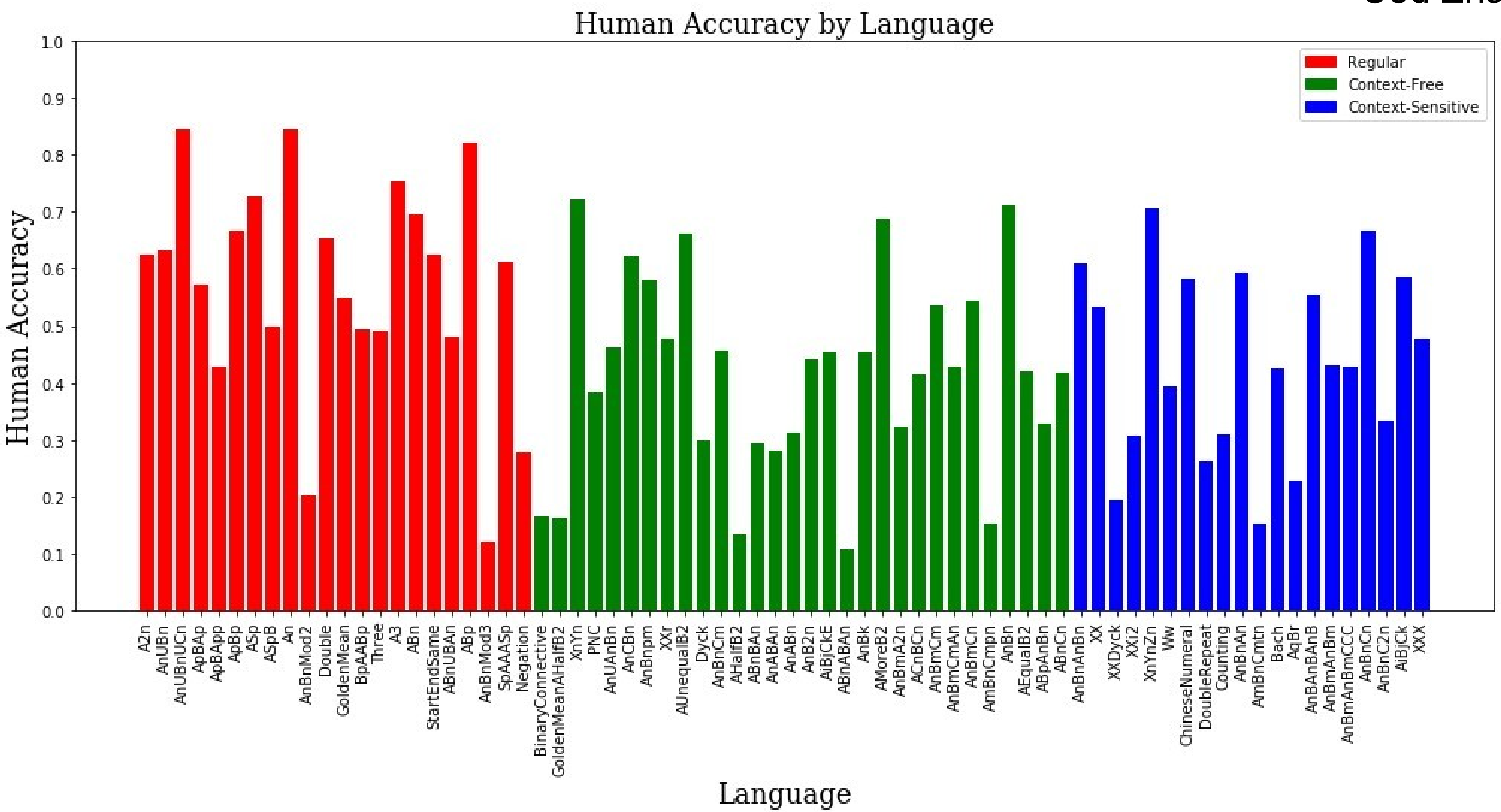
- **Idealized learners can <u>construct</u> computational devices to generate key structures in natural language.**
- **Children don't <u>need</u> language-specific representations or biases to solve the learnability problem**
(though other evidence might make us think those are real)
- **Complex, structured generalizations, infinite productivity, fast learning of latent generative processes** – all are natural tendencies of learning systems that work over computations.

Outline

- **Overview of learnability & formal languages**
- **Learning model**
  - Simple formal languages
  - Artificial language learning
  - Simplified English CFG
- **Three related lines of ongoing work**
  - Human experiments
  - Recursion in monkeys and human groups
  - Algorithm learning in indigenous Amazonians

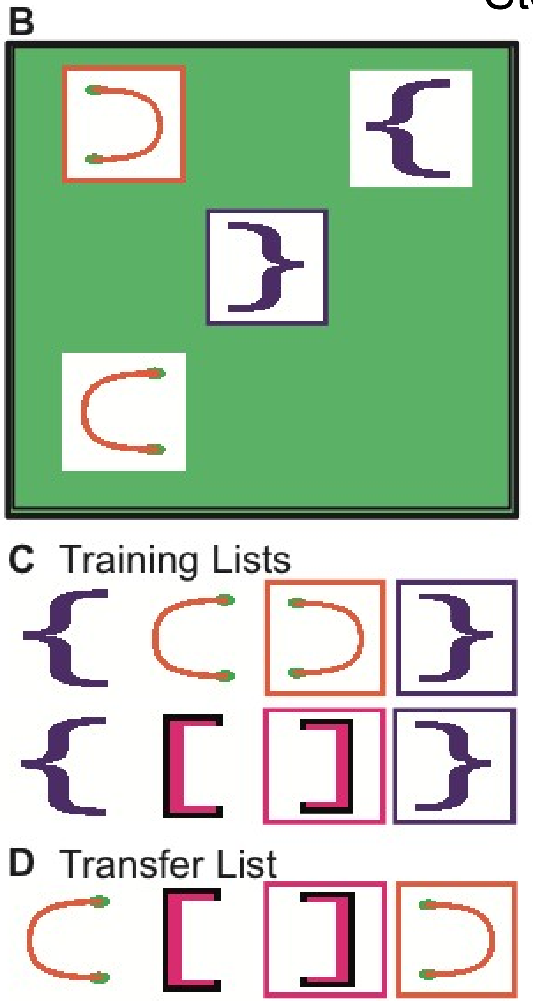
#1 – Formal language experiments

Ced Zhang



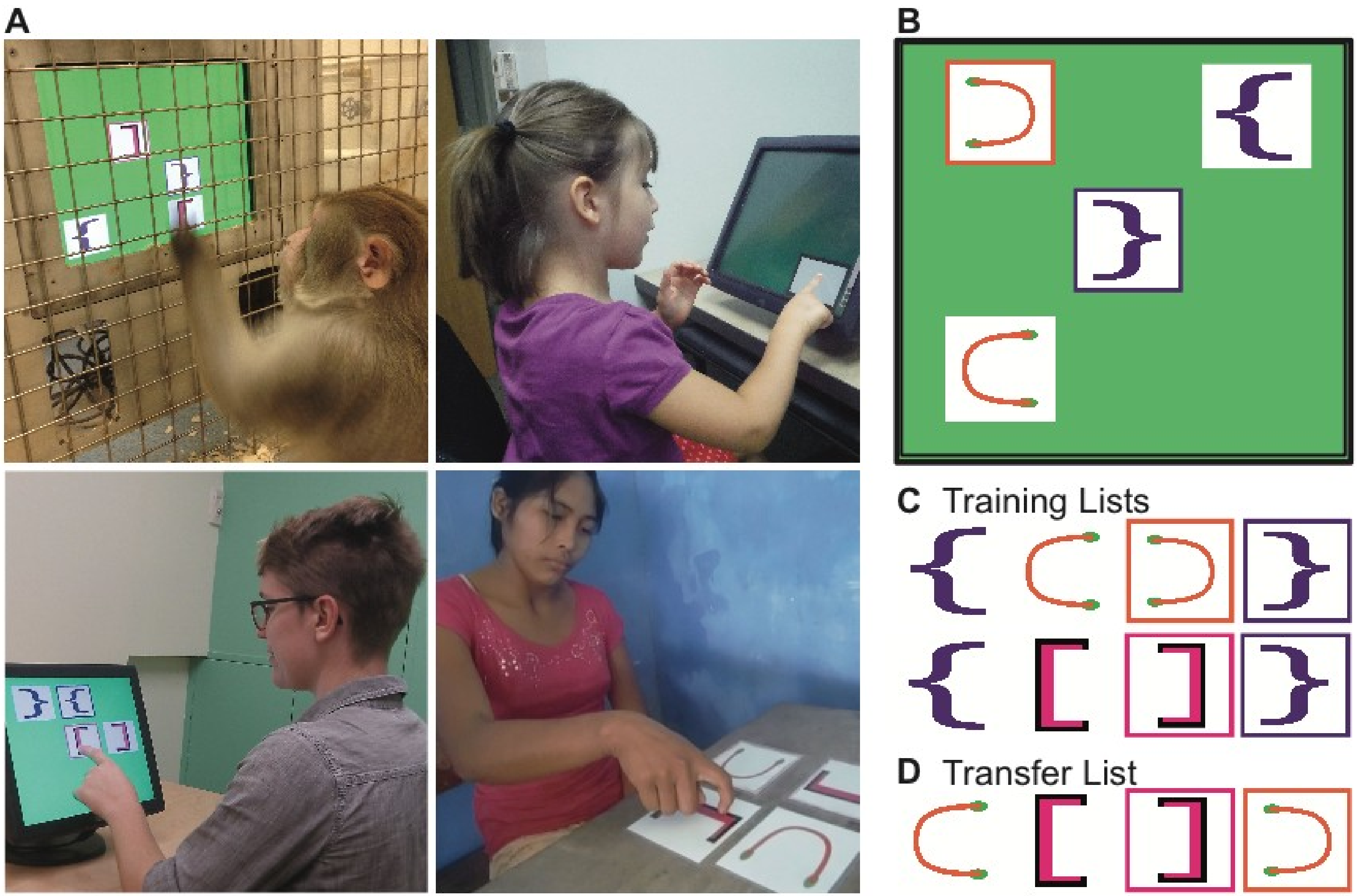
#2 – Bias for recursion
(Ferrigno, Cheyette, Piantadosi, & Cantlon 2020)

Steve Ferrigno

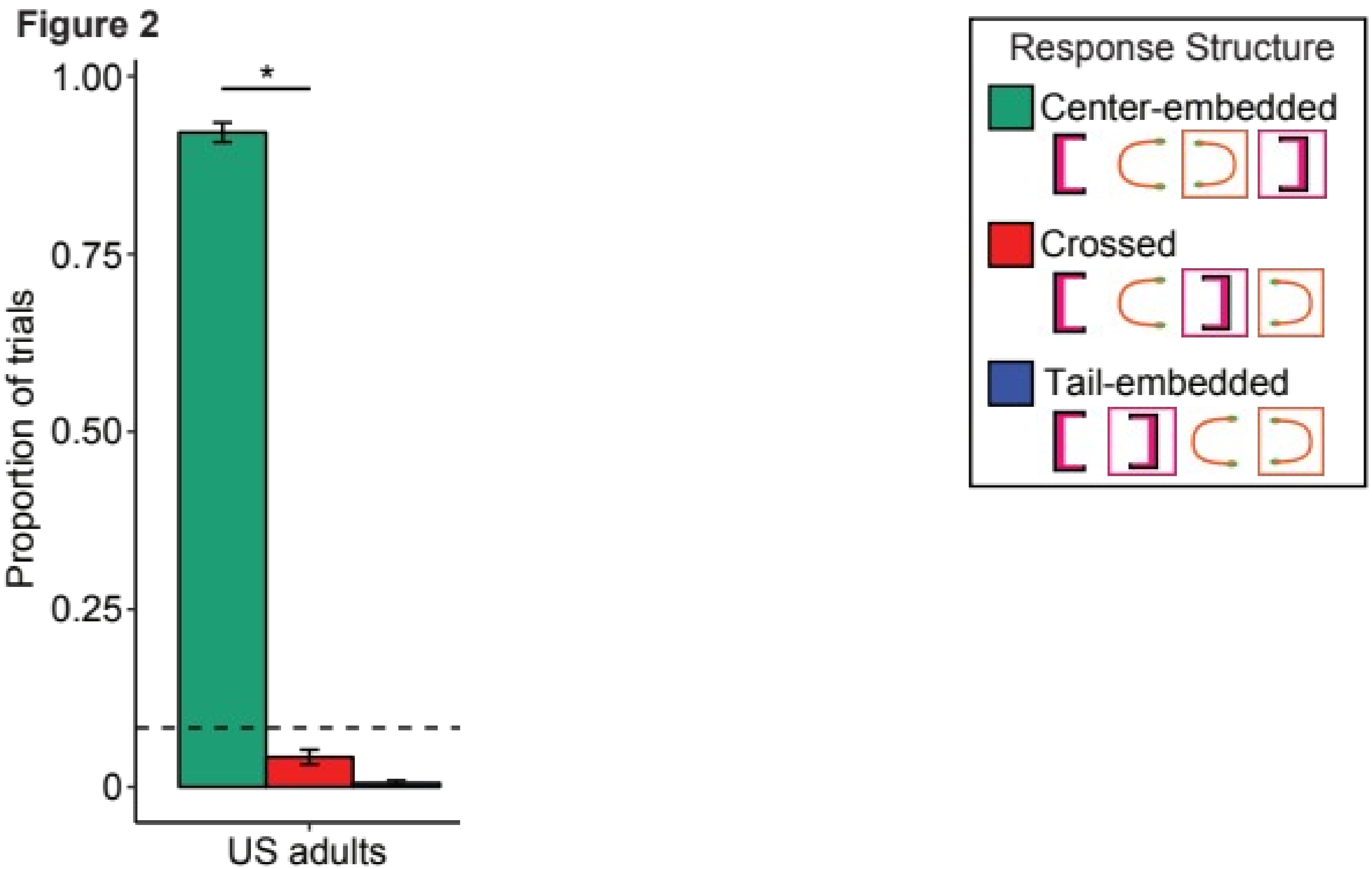


#2 – Bias for recursion

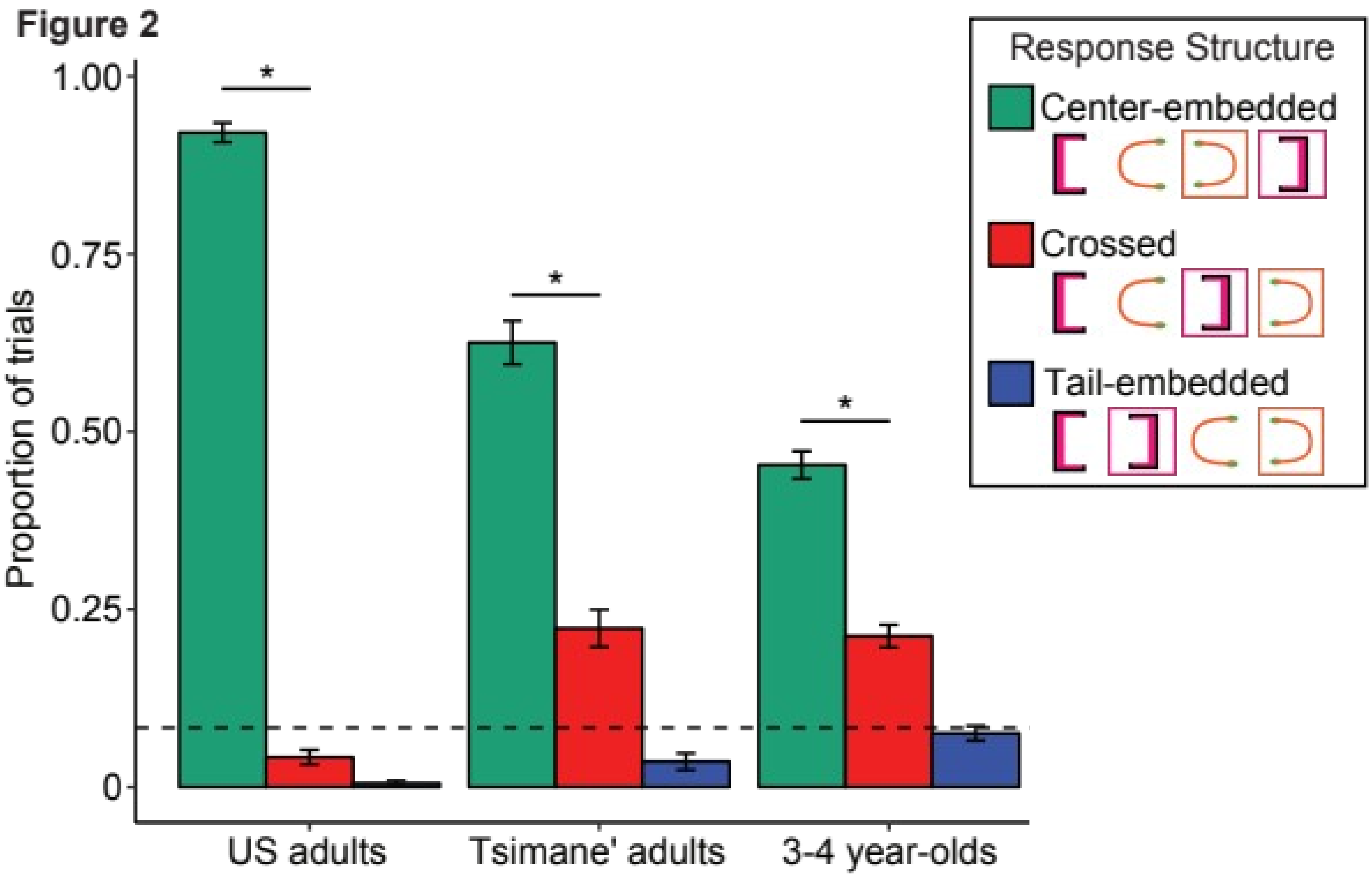
(Ferrigno, Cheyette, Piantadosi, & Cantlon 2020)



#2 – Bias for recursion
(Ferrigno, Cheyette, Piantadosi, & Cantlon 2020)

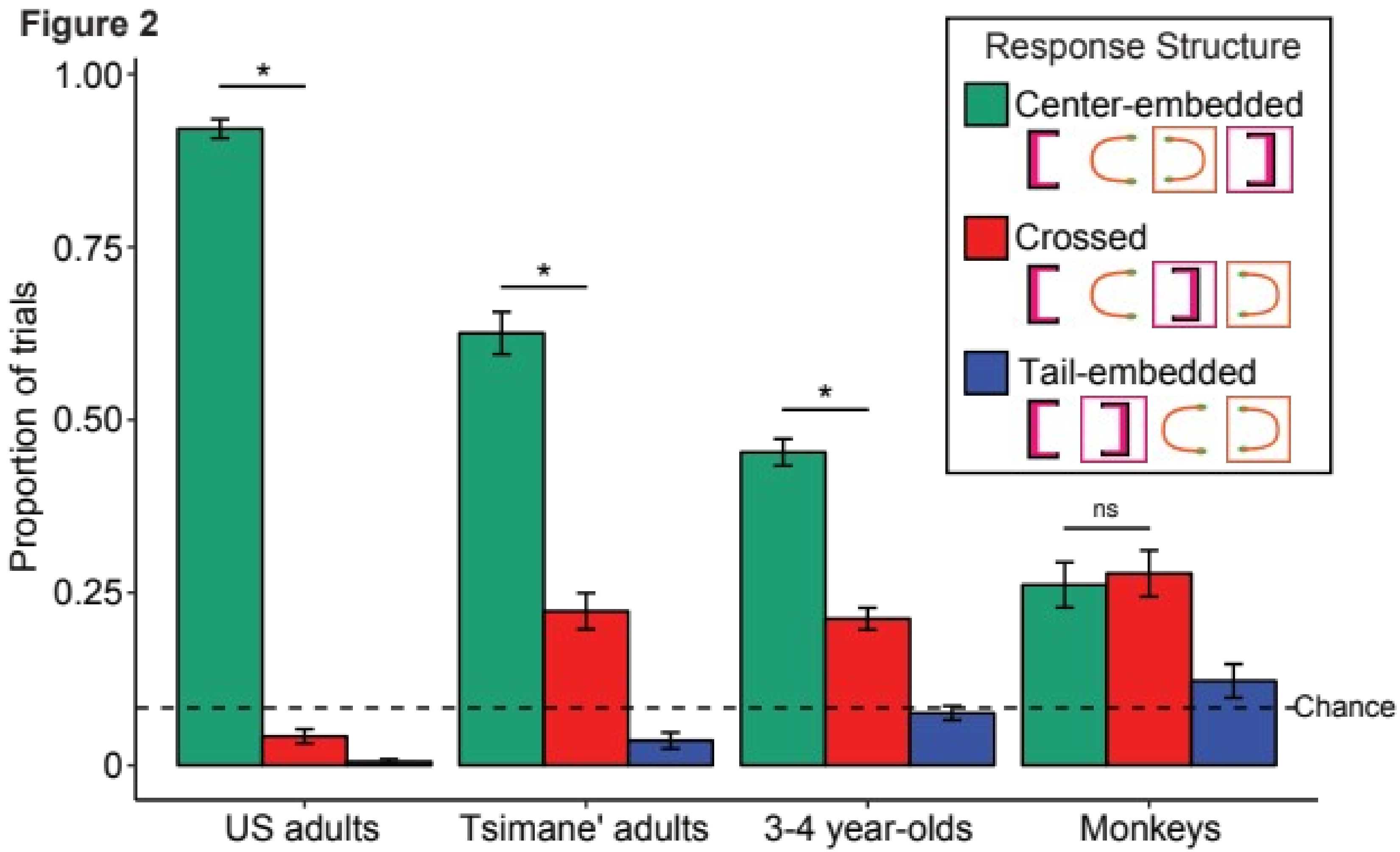


#2 – Bias for recursion
(Ferrigno, Cheyette, Piantadosi, & Cantlon 2020)

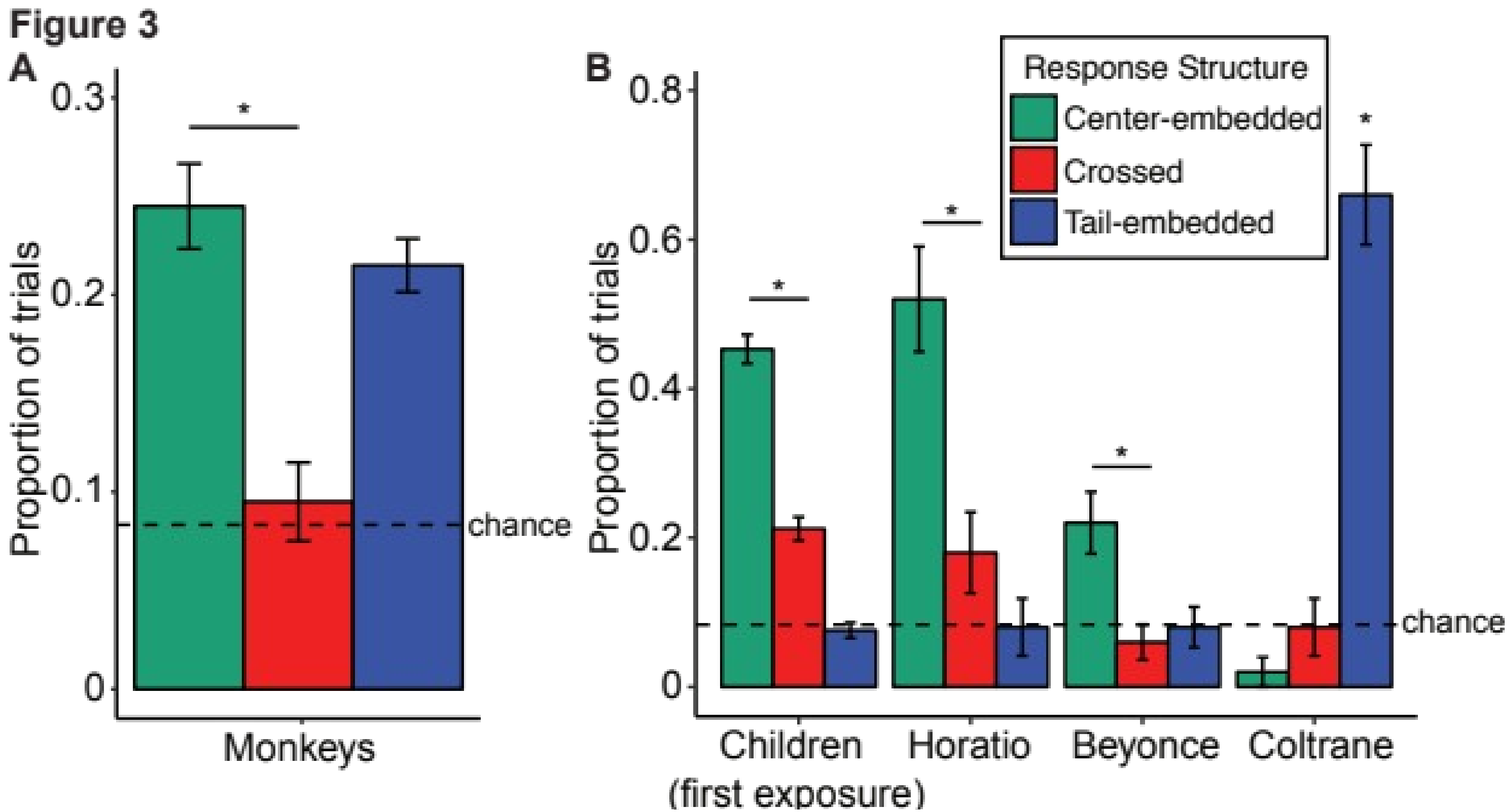


#2 – Bias for recursion

(Ferrigno, Cheyette, Piantadosi, & Cantlon 2020)

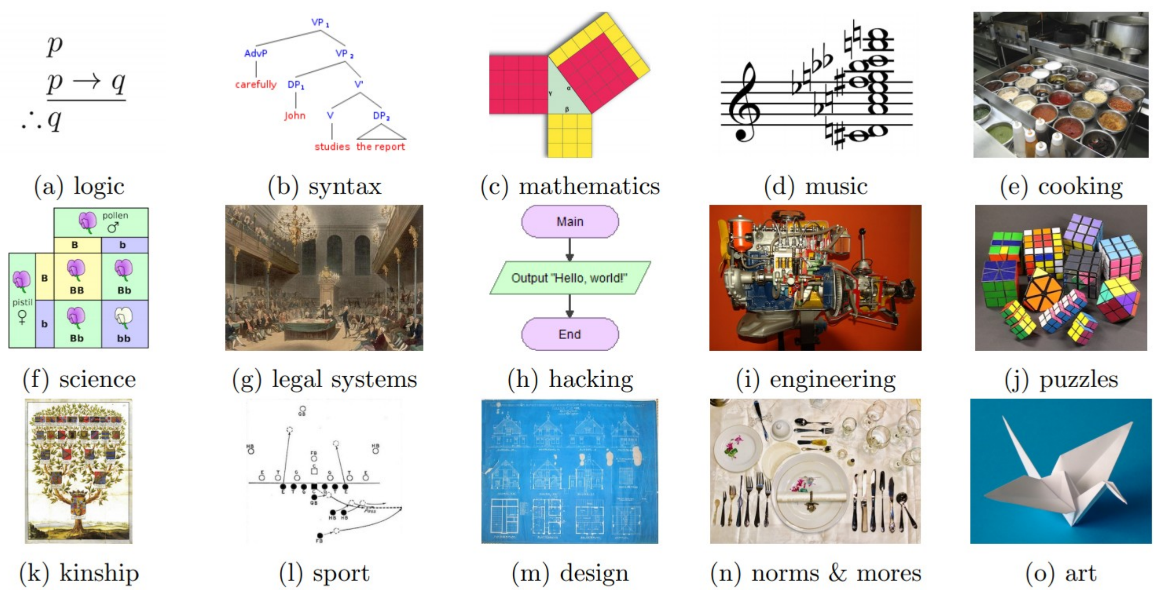


#2 – Bias for recursion
(Ferrigno, Cheyette, Piantadosi, & Cantlon 2020)



Working hypothesis

- **Learning operates over a Turing-complete space**
  - **Learning is like programming** – learners combine existing operations in new ways to form generative models of data
  - **More input data** drives revision, improvement of programs, justifying additional complexity.



Trends in Cognitive Sciences

CellPress REVIEWS

Opinion

The Child as Hacker

Joshua S. Rule,[1,*] Joshua B. Tenenbaum,[1] and Steven T. Piantadosi[2]

The scope of human learning and development poses a radical challenge for cognitive science. We propose that developmental theories can address this challenge by adopting perspectives from computer science. Many of our best models treat learning as analogous to computer programming because symbolic programs provide the most compelling account of sophisticated mental representations. We specifically propose that children's learning is analogous to a particular style of programming called hacking, making code better along many dimensions through an open-ended set of goals and activities. By contrast to existing theories, which depend primarily on local search and simple metrics, this view highlights the many features of good mental representations and the multiple complementary processes children use to create them.

Highlights

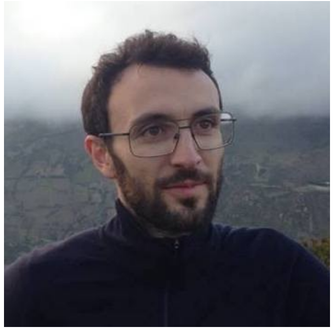
Programs provide our best general-purpose representations for human knowledge, inference, and planning; human learning is thus increasingly modeled as program induction, learning programs from data.

Many formal models of learning as program induction reduce to a stochastic search for concise descriptions of data. Actual human programmers and learners are significantly more complex, using many processes to optimize complex and frequently changing objectives.

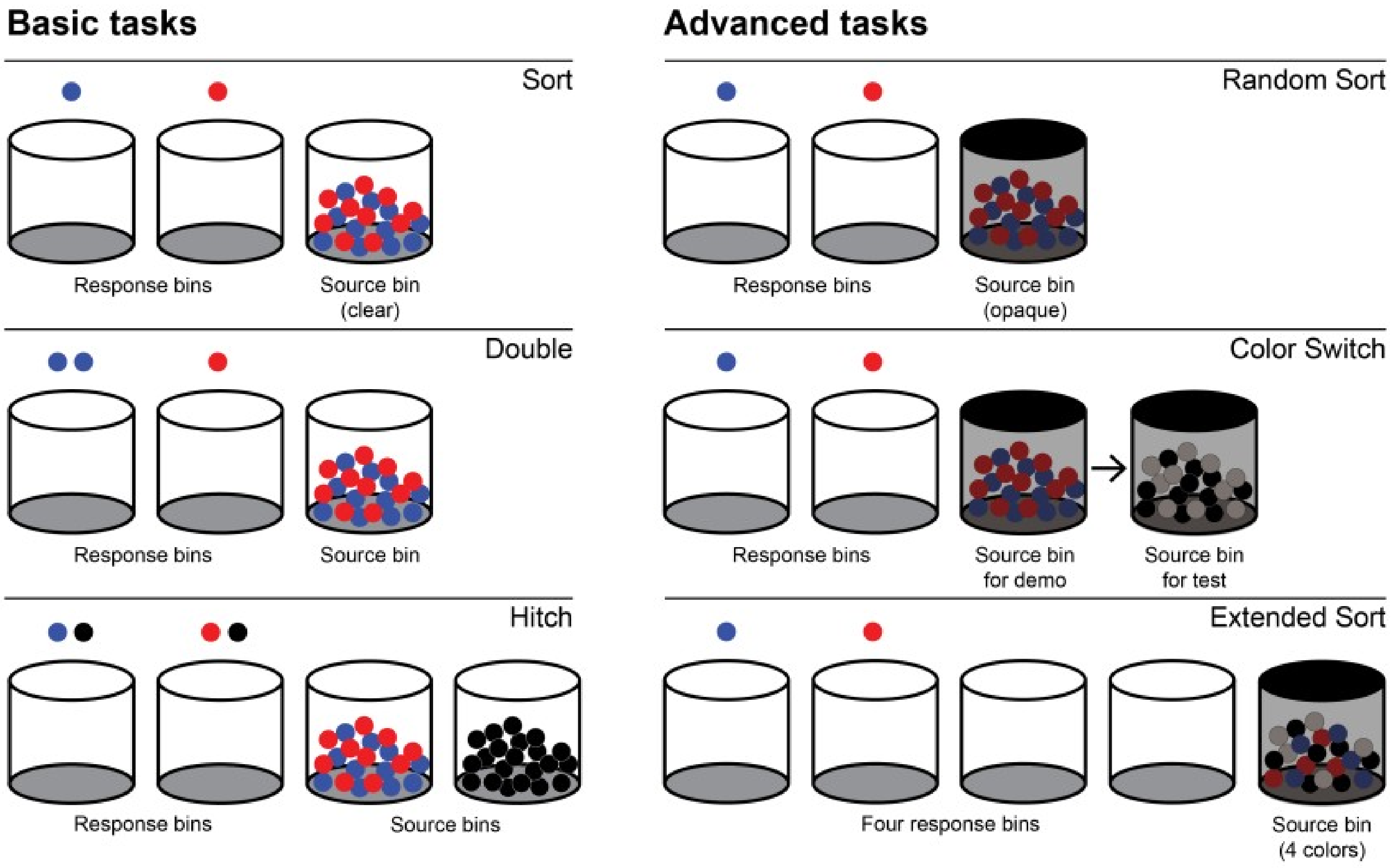
Hacking as a Metaphor for Learning in Cognitive Development

Human cognitive development is qualitatively unique. Though humans are born unusually helpless, they amass an unparalleled cognitive repertoire, including: intuitive theories for domains like physics and biology, formal theories in mathematics and science, language comprehension and production, and complex perceptual and motor skills. Children also learn to learn, producing

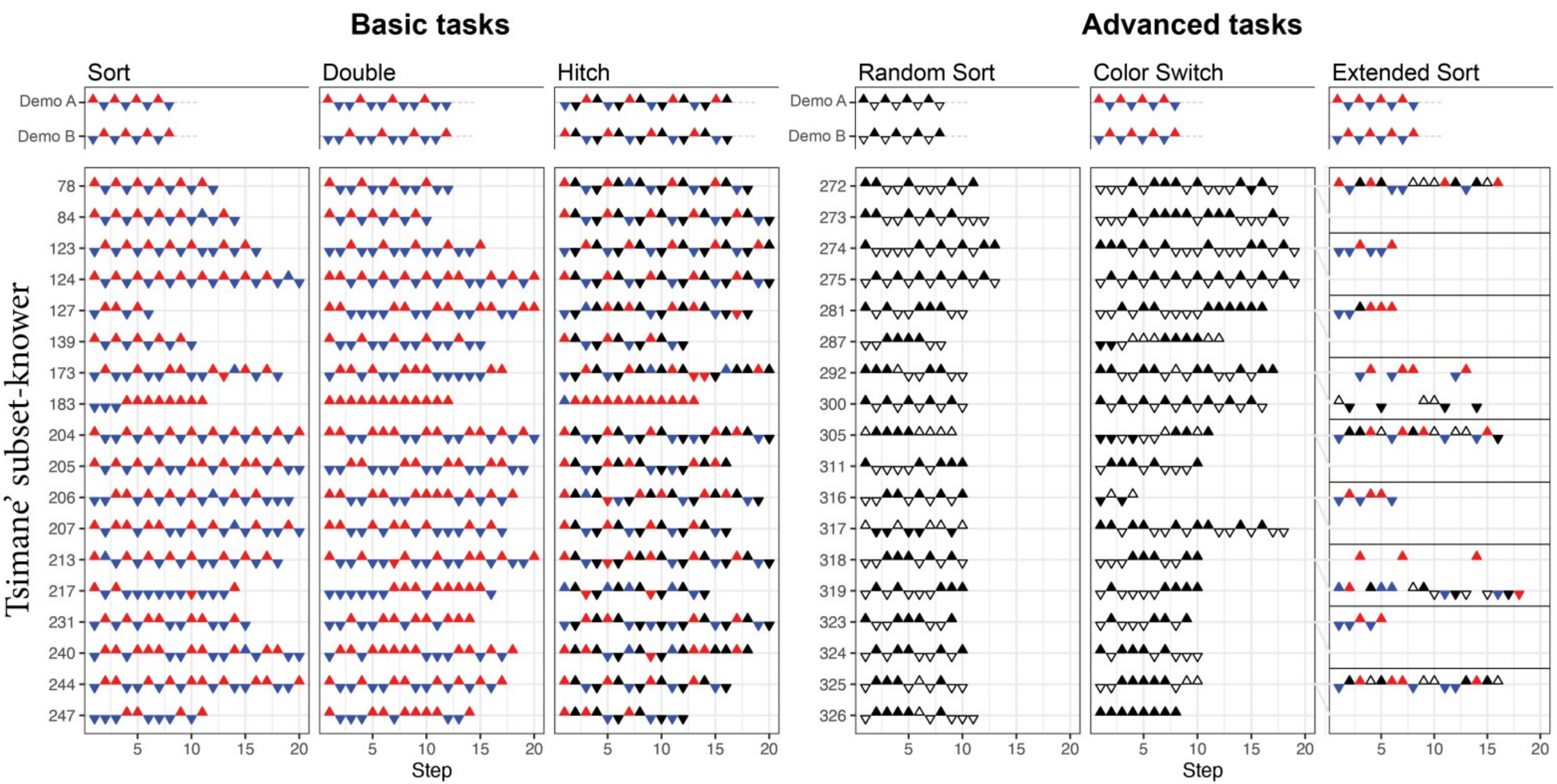
#3 – Program induction in child learners



Ben Pitt



#3 – Program induction in child learners



Thank you

Thanks to my coauthors & contributors on this and related work: Yuan (Leon) Yang, Ced Zhang, Josh Rule, Steve Ferrigno, Sam Cheyette, Jessica Cantlon, Ben Pitt, Charlene Gallardo, Bin Li

Grant support:

NSF DRL 2000759

NSF DRL 1760874

NIH 1R01HD085996

Code is all available in our lab's program induction library, Fleet: https://github.com/piantado/Fleet/

Berkeley
UNIVERSITY OF CALIFORNIA