



# Journal of Language Modelling

VOLUME 9 ISSUE 1  
JUNE 2021



*Institute of Computer Science  
Polish Academy of Sciences  
Warsaw*



# Journal of Language Modelling

VOLUME 9 ISSUE 1  
JUNE 2021

## Editorials

Introduction to the special issue on simplicity in grammar learning 1  
*Roni Katzir, Timothy J. O'Donnell, Ezer Rasin*

## Articles

Simplicity and the form of grammars 5  
*Noam Chomsky*

Approaching explanatory adequacy in phonology  
using Minimum Description Length 17  
*Ezer Rasin, Iddo Berger, Nur Lan, Itamar Shefi, Roni Katzir*

Modelling a subregular bias in phonological learning  
with Recurrent Neural Networks 67  
*Brandon Prickett*

Investigating the effects of i-complexity and e-complexity  
on the learnability of morphological systems 97  
*Tamar Johnson, Kexin Gao, Kenny Smith,  
Hugh Rabagliati, Jennifer Culbertson*

Typology emerges from simplicity in representations and learning 151  
*Dakotah Lambert, Jonathan Rawski, Jeffrey Heinz*



JOURNAL OF  
LANGUAGE MODELLING

ISSN 2299-8470 (electronic version)

ISSN 2299-856X (printed version)

<http://jlm.ipipan.waw.pl/>

MANAGING EDITOR

*Adam Przepiórkowski* IPI PAN

GUEST EDITORS OF THIS SPECIAL ISSUE

*Roni Katzir* Tel Aviv University, Tel Aviv, Israel

*Timothy J. O'Donnell* McGill University, Montréal, Canada

*Ezer Rasin* Tel Aviv University, Tel Aviv, Israel

SECTION EDITORS

*Elżbieta Hajnicz* IPI PAN

*Agnieszka Mykowiecka* IPI PAN

*Marcin Woliński* IPI PAN

STATISTICS EDITOR

*Łukasz Dębowski* IPI PAN



Published by IPI PAN

Institute of Computer Science, Polish Academy of Sciences  
ul. Jana Kazimierza 5, 01-248 Warszawa, Poland

Circulation: 50 + print on demand

Layout designed by Adam Twardoch.

Typeset in X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X using the typefaces: *Playfair*  
by Claus Eggers Sørensen, *Charis SIL* by SIL International,  
*JLM monogram* by Łukasz Dziedzic.

*All content is licensed under  
the Creative Commons Attribution 4.0 International License.*

<http://creativecommons.org/licenses/by/4.0/>

## EDITORIAL BOARD

*Steven Abney* University of Michigan, USA

*Ash Asudeh* Carleton University, CANADA;  
University of Oxford, UNITED KINGDOM

*Chris Biemann* Technische Universität Darmstadt, GERMANY

*Igor Boguslavsky* Technical University of Madrid, SPAIN;  
Institute for Information Transmission Problems,  
Russian Academy of Sciences, Moscow, RUSSIA

*António Branco* University of Lisbon, PORTUGAL

*David Chiang* University of Southern California, Los Angeles, USA

*Greville Corbett* University of Surrey, UNITED KINGDOM

*Dan Cristea* University of Iași, ROMANIA

*Jan Daciuk* Gdańsk University of Technology, POLAND

*Mary Dalrymple* University of Oxford, UNITED KINGDOM

*Darja Fišer* University of Ljubljana, SLOVENIA

*Anette Frank* Universität Heidelberg, GERMANY

*Claire Gardent* CNRS/LORIA, Nancy, FRANCE

*Jonathan Ginzburg* Université Paris-Diderot, FRANCE

*Stefan Th. Gries* University of California, Santa Barbara, USA

*Heiki-Jaan Kaalep* University of Tartu, ESTONIA

*Laura Kallmeyer* Heinrich-Heine-Universität Düsseldorf, GERMANY

*Jong-Bok Kim* Kyung Hee University, Seoul, KOREA

*Kimmo Koskenniemi* University of Helsinki, FINLAND

*Jonas Kuhn* Universität Stuttgart, GERMANY

*Alessandro Lenci* University of Pisa, ITALY

*Ján Mačutek* Comenius University in Bratislava, SLOVAKIA

*Igor Mel'čuk* University of Montreal, CANADA

*Glyn Morrill* Technical University of Catalonia, Barcelona, SPAIN

*Stefan Müller* Freie Universität Berlin, GERMANY

*Mark-Jan Nederhof* University of St Andrews, UNITED KINGDOM

*Petya Osenova* Sofia University, BULGARIA

*David Pesetsky* Massachusetts Institute of Technology, USA

*Maciej Piasecki* Wrocław University of Technology, POLAND

*Christopher Potts* Stanford University, USA

*Louisa Sadler* University of Essex, UNITED KINGDOM

*Agata Savary* Université François Rabelais Tours, FRANCE

*Sabine Schulte im Walde* Universität Stuttgart, GERMANY

*Stuart M. Shieber* Harvard University, USA

*Mark Steedman* University of Edinburgh, UNITED KINGDOM

*Stan Szpakowicz* School of Electrical Engineering  
and Computer Science, University of Ottawa, CANADA

*Shravan Vasishth* Universität Potsdam, GERMANY

*Zygmunt Vetulani* Adam Mickiewicz University, Poznań, POLAND

*Aline Villavicencio* Federal University of Rio Grande do Sul,  
Porto Alegre, BRAZIL

*Veronika Vincze* University of Szeged, HUNGARY

*Yorick Wilks* Florida Institute of Human and Machine Cognition, USA

*Shuly Wintner* University of Haifa, ISRAEL

*Zdeněk Žabokrtský* Charles University in Prague, CZECH REPUBLIC

# Introduction to the special issue on simplicity in grammar learning

Roni Katzir<sup>1</sup>, Timothy J. O'Donnell<sup>2</sup>, and Ezer Rasin<sup>1</sup>

<sup>1</sup> Tel Aviv University

<sup>2</sup> McGill University

Simplicity has long been central to philosophy of science, at least in the sense that all things being equal, a more parsimonious theory is better than a more complex one. In modern linguistics simplicity has played a particularly prominent role, with explicit discussion in Chomsky 1951, 1965, Halle 1962, Chomsky and Halle 1968, and much subsequent work. The prominence of simplicity in linguistic theory reflects the importance of learning in this domain: children acquiring a language must choose between many different grammars compatible with the input data, and an intriguing possibility is that their choice, perhaps like that of the scientist, is affected by considerations of simplicity.

The present special issue considers the place of simplicity in grammar learning, focusing on recent computational and theoretical linguistic work but very much building on earlier foundations. In addition to discussing the use of simplicity, the papers in this collection touch on some of the challenges involved in turning simplicity from a guiding intuition into a concrete tool. For example, to what extent would such a tool be limited by the observation that simplicity is always stated with respect to a specific frame of reference? Which, if any, of the various notions of simplicity that have been proposed could support successful grammar learning, and would such a notion adequately model how children generalize from the primary linguistic data? Do observed typological generalizations regarding simplicity in linguistic systems arise from general considerations of stability of simple grammars under repeated iterations of learning across generations, or is there (also) a direct pressure for simplicity? Our hope is

that the papers in this issue both advance the understanding of these questions and serve as an invitation to debate them further.

The paper by Chomsky discusses a fundamental observation in treatments of simplicity: that simplicity must always be stated with respect to a concrete frame of reference. This observation highlights an arbitrariness or subjectivity that might be seen as an obstacle to the scientific use of simplicity. A striking insight of early generative grammar, however, was that in the hands of the cognitive scientist this frame-dependence is in fact an asset rather than a liability. In particular, the very dependence on a frame of reference that makes simplicity subjective also makes it possible to reason from typological and acquisitional generalizations to underlying representational frameworks, providing evidence for those frameworks that make the observed generalizations simple. The paper situates several major theoretical developments in generative linguistics – ranging from early work to very recent additions, and including theorizing about the evolution of universal grammar itself – within the context of simplicity-based considerations.

The paper by Rasin, Berger, Lan, Shefi, and Katzir discusses the right notion of simplicity for grammar learning in light of different notions that have been proposed in the literature. In particular, it considers both grammar simplicity, as in the evaluation metric of early generative grammar, and simplicity of describing the data, which is closely related to the Subset Principle and to Maximum Likelihood approaches. The paper concludes that neither notion is adequate on its own but that a notion that combines them in a certain way – as in Solomonoff's theory of induction, Kolmogorov Complexity, and the principle of Minimum Description Length (MDL) – is adequate and could provide the child with a criterion for comparing hypotheses that seems to match linguistic intuitions in various cases. It illustrates the use of this criterion with an implemented MDL learner for phonological rule systems, reporting simulation results on small corpora that present well-known morpho-phonological challenges from the literature.

The paper by Prickett takes a different perspective on simplicity and grammar learning by associating complexity with layers in the hierarchy of formal languages. The paper considers evidence from experiments of artificial grammar learning that suggests a bias for hy-



potheses that reside on lower rungs of the hierarchy over hypotheses that are higher and require greater weak generative power. The paper provides evidence that the implicit learning bias of a specific recurrent neural network is compatible with this kind of preference.

The paper by Johnson, Gao, Smith, Rabagliati, and Culbertson looks at simplicity in morphological paradigms in light of a distinction due to Ackerman and Malouf (2013) between e(nergetic) and i(ntegrative) complexity. E-complexity tracks the number of surface distinctions in a paradigm and varies greatly across paradigms. I-complexity is a measure of how predictable the elements of a paradigm are from the form of a representative element and has been argued by Ackerman and Malouf to be consistently low across paradigms. As Ackerman and Malouf note, the typological generalization might be related to learning, though this relation could in principle be indirect and arise from the fact that predictable paradigms might be more stable to intergenerational transmission than unpredictable paradigms. Using both an artificial grammar learning experiment and simulations with RNNs, and using specific information-theoretic formulations of e- and i-complexity, Johnson et al. ask whether there is a more direct learning preference for simpler paradigms. They indeed report such a preference but conclude that it is greater for the typologically variable e-complexity than for the typologically stable i-complexity. They also explore the relation between the two notions of complexity across artificially-generated paradigms and find an inverse relation between the two.

The final paper in this special issue, by Lambert, Rawski, and Heinz, looks at grammar learning through the prism of yet another notion of complexity: that of resource (specifically, space and time) complexity. The paper provides a systematic exploration of representations and learning algorithms that vary in terms of their resource complexity, drawing a connection between the possible combinations of representations and algorithms on the one hand and the subregular hierarchy in phonological typology (Heinz and subsequent work) on the other hand.

## REFERENCES

- Farrell ACKERMAN and Robert MALOUF (2013), Morphological Organization: The Low Conditional Entropy Conjecture, 89(3):429–464.
- Noam CHOMSKY (1951), Morphophonemics of Modern Hebrew, Master's thesis, University of Pennsylvania.
- Noam CHOMSKY (1965), Aspects of the Theory of Syntax, MIT Press, Cambridge, MA.
- Noam CHOMSKY and Morris HALLE (1968), The Sound Pattern of English, Harper and Row Publishers, New York.
- Morris HALLE (1962), Phonology in generative grammar, Word, 18(1/2):54–72.

Roni Katzir

© 0000-0002-0241-1896  
rkatzir@tauex.tau.ac.il

Department of Linguistics  
and Sagol School of Neuroscience  
Tel Aviv University  
Tel Aviv, Israel 6997801

Timothy J. O'Donnell

© 0000-0000-0000-0000  
timothy.odonnell@mcgill.ca

Department of Linguistics  
McGill University  
Montréal, Canada

Ezer Rasin

© 0000-0001-8980-5566  
rasin@tauex.tau.ac.il

Department of Linguistics  
Tel Aviv University  
Tel Aviv, Israel 6997801

Roni Katzir, Timothy J. O'Donnell, and Ezer Rasin (2021), Introduction to the special issue on simplicity in grammar learning, Journal of Language Modelling, 9(1):1–4

https://dx.doi.org/10.15398/jlm.v9i1.280

This work is licensed under the Creative Commons Attribution 4.0 Public License.

<http://creativecommons.org/licenses/by/4.0/>

# Simplicity and the form of grammars

*Noam Chomsky*

Massachusetts Institute of Technology  
University of Arizona

## ABSTRACT

The goal of theory construction is explanation: for language, theory for particular languages (grammar) and for the faculty of language FoL (the innate endowment for language acquisition). A primitive notion of simplicity of grammars is number of symbols, but this is too crude. An improved measure distinguishes grammars that capture genuine properties of language from those that do not. The theory of FoL must meet the empirical conditions of learnability (under extreme poverty of stimulus), and evolvability (given the limited but not insignificant evidence available). Recent work provides promising insights into how these twin conditions may be satisfied.

*Keywords:*  
*simplicity,*  
*explanation,*  
*evaluation,*  
*grammar, faculty*  
*of language,*  
*learnability,*  
*evolvability,*  
*externalization*

There is a close relation between the two concepts in the title – which also happens to be the title of the first talk I gave as a graduate student and the topic of my first paper *Morphophonemics of Modern Hebrew* (MMH; Chomsky 1949/1951)<sup>1</sup> – concerns that have remained salient for me to the present. The relation becomes clear when we consider the goals of the theory of language. Pursuing the relation more

---

<sup>1</sup> An improved 1951 version was published in 1979. I bring up this text, a student paper not intended for publication, because it is the first extensive study of these topics, and the last at any such level of detail. It soon became obvious that the effort was far too ambitious though the general concerns persisted in new forms, even some of the measures of simplicity outlined, as discussed below.

closely gives a good deal of insight into the nature and development of linguistic theory, and also provides a more principled basis for elements of common practice.

As in other domains, the primary goal of theories of language is to explain in the best way the data that constitute the subject matter of the theory, along with determining just what is the relevant subject matter.<sup>2</sup> The concept “best way” is traditionally (and plausibly) understood in terms of simplicity/economy. And when spelled out, these notions are necessarily relative to the formal nature of the system under consideration.

In his investigations of these topics, Nelson Goodman – with whom I was studying at the time – observed that “The motives for seeking economy in the basis of a system are much the same as the motives for constructing the system itself”; “To seek truth is to seek a true system, and to seek system at all is to seek simplicity” (Goodman 1943, 1955).

From a somewhat different perspective, Herman Weyl drew essentially the same conclusions: “The assertion that nature is governed by strict laws is devoid of all content if we do not add the statement that it is governed by mathematically simple laws... That the notion of law becomes empty when an arbitrary complication is permitted was already pointed out by Leibniz in his *Metaphysical Treatise*... The astonishing thing is not that there exist natural laws, but that the further the analysis proceeds..., the finer the elements to which the phenomena are reduced, the simpler – and not the more complicated, as one would originally expect – the fundamental relations become and the more exactly do they describe the actual occurrences” (Weyl 1932, cited by Roberts and Watumull 2015).

In a similar vein, Galileo held that nature is simple and it is the task of the scientist to demonstrate that in particular cases – a quasi-empirical claim, but so powerfully verified over the centuries that it is fair to adopt the precept.

There are many similar observations by distinguished figures, for good reasons. If we are serious about linguistic theory we can hardly

---

<sup>2</sup>Not given *a priori*. Just what constitutes a language *L* is in part a matter of decision. What data belong to *L* is theory-driven, a familiar matter.

ignore the question of finding a way to measure its simplicity, which will, transparently, depend on the form it assumes.

For language, there is an additional reason to suppose that the basic system is quite simple. There is mounting evidence that the core elements of the faculty of language (FoL) emerged pretty much along with modern humans and haven't changed since, hence emerged rather suddenly in evolutionary time (Berwick and Chomsky 2016; Huybregts 2017). If so, one would expect that they would have assumed a simple form.

The task of finding the simplest theory for language is posed at two levels: for the theory of each language (its grammar), and for the theory of FoL (UG, in contemporary terminology). FoL provides the framework within which each language develops much as the general faculty of human vision does for each individual visual system, allowing considerable variation as classic experimental work has shown. FoL must satisfy at least what has been called the Basic Property of language: it must provide mechanisms for a language to generate an unbounded array of hierarchically structured expressions in a form that can be interpreted at two interfaces with external systems, at the conceptual-intentional level CI for expression of thought and at the sensorymotor level SM for externalization in some medium, typically sound. There are important asymmetries to which we return.

More generally, UG must satisfy the condition of “explanatory adequacy,” answering the question how a particular language can in principle be acquired from the data available (Chomsky 1965). To do so, UG must specify the “search space,” the class of possible languages PL, along with a selection procedure SP that selects the correct grammar (or set of grammars) for each language given relevant data.

These conditions become far more restrictive if we take a language to be a property of the organism in accord with the “Biolinguistic Program” BL, Massimo Piattelli-Palmarini's term for the evolving discipline.<sup>3</sup> This was a departure from standard views,<sup>4</sup> and partially remains so. While sometimes regarded as contentious, it seems to me that the legitimacy of the BL approach is obvious to the point of

---

<sup>3</sup>Ibid. Lenneberg (1967), the classic exposition.

<sup>4</sup>For a sample, see Chomsky (2013).

truism.<sup>5</sup> If so, adoption of it raises no issue of substance but only one of decision as to which concept of language we choose to consider, so we can put it aside.

Adopting BL, explanatory adequacy requires the further condition that PL-SP be *feasible*. They must provide a realistic abstract account of language acquisition on the basis of the Primary Linguistic Data. In particular, they must account for the huge gap between the data available and what the child knows. It was recognized from the early days of work on generative grammar that this problem of Poverty of Stimulus is enormous, and later investigations of what is known by a very young child along with statistical study of the sparsity of data available have revealed that the problem is far more severe even than what had been assumed.<sup>6</sup> Accordingly, PL-SP must be sharply constrained.

Whether our concern is feasibility and BL or the weaker notion of just explanation, the next problem is to spell out what we mean by “simplicity.” For Goodman (1943), as the quote above indicates, the answer reduced (mainly) to minimal number of primitives as the basis for the constructional system under consideration. MMH was an attempt to explore these ideas over a broader range. Language provides interesting cases, and the subject matter for MMH was a natural choice: the data are readily available and sufficiently intricate to require richer notions of simplicity. Much richer, it became clear as study of the dual problems of theory construction for language proceeded.

The form of grammars in MMH is a system of rewriting rules with the conventional interpretation: the rule  $X \rightarrow Y$  maps  $AXB$  to  $AYB$ .<sup>7</sup> Exploring ways to measure simplicity, we can begin with the most obvious idea: take SP to rank grammars by the number of symbols they contain. While that seems a natural measure, it quickly becomes clear that it is seriously inadequate. One reason is that the measure does

---

<sup>5</sup> Further, I think it can be argued that other concepts tacitly presuppose it.

<sup>6</sup> See Yang *et al.* 2017.

<sup>7</sup> MMH included a rudimentary syntax with optional unordered rules (basically what became phrase structure grammar) and a complex morphophonemics – part of externalization in current terms – with obligatory ordered rules. The reasons for such distinctions only became clear much later; see below.

not distinguish between rule systems that express legitimate linguistic generalizations from others that do not do so.<sup>8</sup>

Suppose for example that we have the rule sets (1), (1'):

- (1) a.  $X \rightarrow YWB$
- b.  $X \rightarrow YW$
- (1') a.  $X \rightarrow BWY$
- b.  $X \rightarrow YW$

(1) expresses an expected configuration: *B* is optional in the context  $YW\_.$ <sup>9</sup> (1') in contrast expresses no legitimate generalization. But the number of symbols in each is 7. Counting symbols is clearly too crude a measure. In MMH the distinction is captured with a notational transformation taken to be part of the simplicity measure of UG, mapping the rules of (1) to (2):

- (2)  $X \rightarrow YW(B)$

The notation is interpreted as: *B* is optional in the context  $YW\_.$  No similar notation is provided by UG for (1'), not considered a legitimate generalization – an empirical assumption about language, as noted, but well confirmed. Under this notational transformation, the simplicity measure of (1) is 4, capturing the intended distinction.

Consider a more complex configuration, very commonly found. Suppose that  $X \rightarrow Y$  before *A* and elsewhere  $X \rightarrow Z$ .<sup>10</sup> The set of rewriting rules is (3):

- (3) a.  $XA \rightarrow YA$
- b.  $XW \rightarrow ZW$  (for  $W \neq A$ )

---

<sup>8</sup>What is taken to be a “legitimate linguistic generalization” is an empirical hypothesis, subject to testing by examination of languages and by direct experiment.

<sup>9</sup>E.g., the parenthesized optional element in such phrase structure grammar configurations as  $VP \rightarrow V_r NP (PP)$  “read the book (in the library).” In contrast, we do not expect to find the rule set  $VP \rightarrow PP NP V_r$ ,  $VP \rightarrow V_r NP$  (irrelevantly, such outcomes might result from a series of rules). Or more marginally, “all (of) the men,” “cyclic(al) rules,” [sinj(g)r] (with *g* missing in some dialects, yielding a *singer/finger* contrast).

<sup>10</sup>Voicing assimilation of final consonants as in *wife-wives*, sets of irregular verbs, and innumerable other cases.

The parenthesized phrase must be spelled out, listing all cases of  $W \neq A$ . The list is infinite, but even if we impose some sharp restriction on  $W$ , the list is very long, and the number of symbols in the expanded version of (3) gives a completely wrong simplicity measure for a configuration that should be highly valued.

What is clearly the right answer requires several steps that are of more general significance.<sup>11</sup>

First, we have to distinguish obligatory from optional rules, and ordered from unordered rules. For the syntax – mapping to CI – the normal case is unordered and optional, if such rules exist at all; they may not (see below). For externalization to SM, the normal case of rule systems (which are quite complex) is ordered and obligatory. The configuration (3) falls within externalization.<sup>12</sup>

With these conventions in place, we can introduce the notational transformation of (4), interpreted as (3):

- (4)     a.  $X \rightarrow Y / \_A$   
          b.  $X \rightarrow Z$

The simplicity measure is small, as it should be for this legitimate generalization. The rules (3)–(4) state that  $X$  becomes  $Y$  before  $A$ , and becomes  $Z$  elsewhere. This device is the familiar “elsewhere condition”: first list the exceptions, then the general rule for everything else.<sup>13</sup>

From the early inquiries into generative grammar it was found that rule ordering was still more intricate: with cyclic application of rules and implicational relations, grammars are greatly simplified and (accordingly) yield deeper explanations, while also providing the ba-

---

<sup>11</sup> These steps were all taken in MMH and commonly adopted in later work in generative grammar.

<sup>12</sup> NB: “normal.” There are some exceptions on the periphery, like free variation. There are interesting questions about the tacit choices here but they do not bear on the main points about simplicity and general architecture of grammar, so I will put them aside.

<sup>13</sup> The elsewhere condition, which may trace back to classical India, has been widely used in practice. It is also a core element of Charles Yang’s tolerance principle, which has been highly successful in explaining when rules are productive and establishing a firm core–periphery distinction. See Yang *et al.* 2017, and for more extensive analysis Yang 2016.



sis for compositionality of semantic interpretation, matters I will put aside here (Chomsky *et al.* 1956; Chomsky and Halle 1968).

The notations and conventions in MMH, now common, provide a reasonable step towards a feasible evaluation procedure: the simplicity measure of a rule system is the number of symbols under the conventions and notational transformations that capture legitimate linguistic generalizations – all expressing empirical hypotheses about language.

In MMH, the main problem was to find the simplest ordering of rules, which was quite deep. In those hand-computation days, the task was impossible, so the analysis was restricted to finding a relative minimum: a particular ordering with a lower measure (higher valued) than any re-ordering of adjacent rules. The exercise illustrates some of the problems of constructing UG, tasks challenging enough that they have rarely been undertaken on any large scale.<sup>14</sup> Note that the complexities arise primarily (perhaps completely) in externalization, a matter to which we return.

All of this is only the beginning, however. Another aspect of the quest for feasibility is restricting the search space PL and constraining the selection procedure SP. These topics have been the main concern of the study of *narrow syntax*,<sup>15</sup> generation of structures at the CI interface. The topic is too rich to review here. I will briefly mention only a few stages, keeping to one course of development, which I think is on the right track.

Early generative grammar assumed that two systems of rules interact: Phrase Structure Grammar PSG and Transformational Grammar TG. Both were relatively unconstrained, yielding a huge search space, remote from any hope for feasibility. Serious efforts to restrict the search space began in the early 1960s. It was quickly recognized that PSG permitted far too many options; there was, for example, noth-

---

<sup>14</sup> One of the last cases I know of is *Sound Pattern of English* (Chomsky and Halle 1968). Later study of externalization, the primary locus of these issues, took a different course that ignores the questions, and as far as I can see, cannot accommodate them. See Chomsky 1995, p. 380.

<sup>15</sup> Broadly construed, syntax incorporates all internal symbolic computation, including externalization to phonetic form and logical syntax, often called formal semantics. For reasons discussed elsewhere, human language may not have semantics in the technical sense based on reference/denotation.

ing to bar the vast array of “crazy rules” such as  $NP \rightarrow V PP$ , and the symbols used were themselves illegitimate, tacitly incorporating structural relations that must be spelled out (why  $NP$ ?). PSG was therefore abandoned in favor of X-bar theory, sharply restricting the options for grammars.<sup>16</sup>

Though it wasn’t recognized at the time, X-bar theory had rich consequences, some not explored seriously until recent years. Unlike PSG, X-bar theory (also TG as it developed) yields pure structures, without linear order or other organization. Hence resort to X-bar theory introduces a sharp distinction between (i) narrow syntax, consisting of X-bar theory and TG and yielding CI representations, and (ii) externalization of syntactic structure to the sensorimotor system SM (typically phonetic form PF). As noted, externalization appears to be the locus of the apparent complexity, variety, and mutability of language – not surprisingly. Externalization relates two systems that are entirely independent, both in character and evolutionary history: language proper and SM. Establishing that relation is a complex cognitive process that can be carried out in many ways. In particular, it must deal with the mismatch between narrow syntax, a system of pure structure, and SM, which imposes a requirement of linear order for reasons that have nothing to do with language.<sup>17</sup> There must, it seemed, be a “head parameter” that each  $L$  has to set one way or another (V-Object for English, Object-V for Japanese, etc.). Along with other work of the 1970s, including radical simplification of TG, that led to a new conception of the form of language, the Principles and Parameters (P&P) framework, with fixed principles of UG that determine PL and parameters that have to be set in acquisition of language, the latter restricted largely to externalization (perhaps completely, we might someday learn).

The problems of simplicity of grammars and of UG are accordingly reshaped. A crucial problem is to find a feasible search process through the set of parameters, and to determine their status: how did they evolve? How are they captured in UG and stored in the brain?

---

<sup>16</sup> Too far, it was later realized. See Chomsky (2013), opening directions I will put aside though they bear directly on explanatory adequacy and simplicity.

<sup>17</sup> Sign language, using the options available in visual space, permits somewhat different devices.

The “head parameter” suggests possible answers. It is, strictly speaking, not a parameter. It is not part of UG, did not evolve, and is not internally stored. Rather, it expresses a mismatch between two independent systems: language proper and SM. The mismatch must be resolved in acquisition, but is not part of grammar.

Recent work by Ian Roberts (Roberts 2019), supported by rich empirical evidence from a wide range of typologically different languages, suggests a radical solution to these problems. It provides a feasible search procedure and concludes that parameters altogether are not part of UG (hence did not evolve and are not stored) but rather emerge in the course of acquisition in predictable ways.

Meanwhile work in the “Minimalist Program” has subjected the principles of the P&P systems to much closer analysis, reducing generation in narrow syntax to the simplest combinatorial operation (binary set-formation, called “merge”). That step turns out to incorporate and unify earlier proposals and to yield solutions to long-standing puzzles and new ones discovered along the way, along with suggestions as to how language evolved.<sup>18</sup> One conclusion reverses the general view (mine included) concerning compositionality and displacement: that compositionality (provided by PSG and its descendants) is unproblematic and displacement (handled by TG) is a curious “imperfection” of language that has to be explained away somehow. It turns out that the opposite is true. Displacement is the simplest and unproblematic case, and composition beyond displacement requires an explanation in terms of special properties of language. All of these developments bear directly on our topic here.

Without further elaboration, even a brief review of the course of research in generative grammar since its modern origins reveals that the concepts of simplicity and form of grammar have been closely related throughout, that measuring simplicity is an essential task and is no simple matter, and that inquiry into this relation has led to substantial insight into the general nature of language, with a promise of more to come.

---

<sup>18</sup> See note 3. For more general discussion, see Chomsky (2015).

## REFERENCES

- Robert C. BERWICK and Noam CHOMSKY (2016), *Why Only Us: Language and Evolution*, MIT Press, Cambridge, MA.
- Noam CHOMSKY (1949/1951), *Morphophonemics of Modern Hebrew*, Master's thesis, University of Pennsylvania.
- Noam CHOMSKY (1965), *Aspects of the Theory of Syntax*, MIT Press, Cambridge, MA.
- Noam CHOMSKY (1979), *Morphophonemics of Modern Hebrew*, Garland Publishing, Inc., New York and London.
- Noam CHOMSKY (1995), *The Minimalist Program*, MIT Press, Cambridge, MA.
- Noam CHOMSKY (2013), Problems of projection, *Lingua*, 130:33–49.
- Noam CHOMSKY (2015), *What Kind of Creatures are We?*, Columbia University Press.
- Noam CHOMSKY and Morris HALLE (1968), *The Sound Pattern of English*, Harper and Row Publishers, New York.
- Noam CHOMSKY, Morris HALLE, and Fred LUKOFF (1956), On accent and juncture in English, in M. HALLE, H. LUNT, and H. MACLEAN, editors, *For Roman Jakobson*, pp. 65–80, Mouton.
- Nelson GOODMAN (1943), On the simplicity of ideas, *The Journal of Symbolic Logic*, 8(4):107–121.
- Nelson GOODMAN (1955), Axiomatic measurement of simplicity, *The Journal of Philosophy*, 52(24):709–722.
- MAC Riny HUYBREGTS (2017), Phonemic clicks and the mapping asymmetry: how language emerged and speech developed, *Neuroscience & Biobehavioral Reviews*, 81:279–294.
- Eric LENNEBERG (1967), *Biological Foundations of Language*, volume 68, Wiley New York.
- Ian ROBERTS (2019), *Parameter Hierarchies and Universal Grammar*, Oxford University Press, USA.
- Ian ROBERTS and Jeffrey WATUMULL (2015), Leibnizian Linguistics, in Ángel J. GALLEGO and Dennis OTT, editors, *50 Years Later: Reflections on Chomsky's Aspects*, pp. 211–222, MIT Working Papers in Linguistics.
- Hermann WEYL (1932), *The Open World: Three Lectures on the Metaphysical Implications of Science*, Yale University Press, New Haven.
- Charles YANG (2016), *The Price of Linguistic Productivity: How Children Learn to Break the Rules of Language*, MIT Press, Cambridge, MA.


*Simplicity and the form of grammars*

Charles YANG, Stephen CRAIN, Robert C BERWICK, Noam CHOMSKY, and Johan J BOLHUIS (2017), The growth of language: Universal Grammar, experience, and principles of computation, *Neuroscience & Biobehavioral Reviews*, 81:103–119.


*Noam Chomsky*

Massachusetts Institute of Technology, Cambridge, MA, USA  
University of Arizona, Tucson, AZ, USA

Noam Chomsky (2021), *Simplicity and the form of grammars*, *Journal of Language Modelling*, 9(1):5–15

 <https://dx.doi.org/10.15398/jlm.v9i1.257>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

 <http://creativecommons.org/licenses/by/4.0/>



# Approaching explanatory adequacy in phonology using Minimum Description Length

*Ezer Rasin, Iddo Berger, Nur Lan, Itamar Shefi, and Roni Katzir*  
Tel Aviv University

## ABSTRACT

A linguistic theory reaches explanatory adequacy if it arrives at a linguistically-appropriate grammar based on the kind of input available to children. In phonology, we assume that children can succeed even when the input consists of surface evidence alone, with no corrections or explicit paradigmatic information – that is, in learning from *distributional evidence*. We take the grammar to include both a lexicon of underlying representations and a mapping from the lexicon to surface forms. Moreover, this mapping should be able to express optionality and opacity, among other textbook patterns. This learning challenge has not yet been addressed in the literature. We argue that the principle of Minimum Description Length (MDL) offers the right kind of guidance to the learner – favoring generalizations that are neither overly general nor overly specific – and can help the learner overcome the learning challenge. We illustrate with an implemented MDL learner that succeeds in learning various linguistically-relevant patterns from small corpora.

*Keywords:*  
*learning,*  
*phonology,*  
*opacity,*  
*optionality,*  
*Minimum*  
*Description Length*

As part of language acquisition, the child needs to acquire many different aspects of the morpho-phonology of their language. If the child is learning English, for example, they will need to learn that in ‘cats’, pronounced [k<sup>h</sup>æts], the aspiration of the initial [k] and the voicelessness of the final [s] are no accident: in English, voiceless stops such as [k] are always aspirated in this position (roughly, syllable-initially in a stressed syllable), and the expression of the plural morpheme is always the voiceless [s] after a voiceless stop such as [t]. Thus, the child will need to learn that imaginable forms such as [kæts] or [k<sup>h</sup>ætz] are not possible in the language. These pieces of knowledge come from a very large – possibly unbounded – set of possible choices that languages can make and that children must be able to acquire. Moreover, children are capable of acquiring at least some linguistic knowledge of this kind from distributional cues alone, without access to analyzed forms or paradigms and without negative evidence. The result is a nontrivial learning task that is challenging even in relatively simple cases such as deterministic, surface-true phonotactics (as in the aspiration pattern of English) or alternations providing useful information (such as the voicing pattern concerning the /z/ suffix in English). The learning challenge is even more pronounced in cases of optional phonological processes and of opaque interactions of phonological processes. A theory that addresses this challenge can be said to have reached explanatory adequacy (Chomsky 1965). To date, no general solution to this challenge has been provided in the literature.

In this paper, we propose a response to the learning challenge in terms of a certain kind of simplicity metric. The simplicity metric will follow the principle of Minimum Description Length (MDL; Rissanen 1978), which incorporates both the idea of grammar simplicity (as in the evaluation metric of early generative phonology) and that of restrictiveness (or how easy it is for the grammar to capture the data). The representational framework that we use for our discussion will be that of rule-based phonology, which offers a particularly direct handle on the representation of both optionality and opacity. We wish to emphasize, however, that our focus in this paper is the learning approach – namely, the MDL metric – and how it guides the learner given a rep-



representational framework rather than the representational framework itself. In order to illustrate how the MDL metric can guide the learner toward appropriate hypotheses, we present several simulations that start with a small corpus of unanalyzed surface forms – generated from artificial grammars based on morpho-phonological patterns in various languages – and arrive at a full grammar including a lexicon of underlying representations (URs), a morphological segmentation of forms into morphemes and their attachment possibilities, and different kinds of phonological rules (both obligatory and optional) and their ordering (including both transparent and opaque interactions). While it might seem that these different aspects of morpho-phonological knowledge call for a fragmented learning approach, with specialized learners for the different sub-tasks, we will show how the MDL evaluation metric allows all of them to be acquired in a unified way.

We start, in Section 2, by reviewing the challenge of explanatory adequacy in phonology. In Section 3, we present the MDL metric in the context of rule-based phonology and specify a concrete set of representations for phonological grammars and their MDL costs. In Section 4, we present proof-of-concept learning simulations with optionality, rule interaction (including opacity), and interdependent phonology and morphology. Section 5 discusses previous work on learning in phonology and its relation to the goals of this paper. Section 6 concludes the paper.

## EXPLANATORY ADEQUACY IN PHONOLOGY

2

An explanatorily adequate linguistic theory accounts for how the child arrives at a descriptively-adequate grammar based on the primary linguistic data (Chomsky 1965, pp. 25–27). The present paper focuses on this learning challenge in phonology. In Section 3, we argue that combining a suitable theory of phonological representations with the general principle of MDL goes beyond all other proposals in the literature in terms of approaching the goal of explanatory adequacy. Before that,

in the present section, we briefly outline certain aspects of explanatory adequacy in phonology that will be important for evaluating our claim below.

First, we follow Calamaro and Jarosz (2015) in assuming that children can acquire significant aspects of phonological knowledge from distributional evidence alone (that is, from surface forms alone, without systematic negative evidence, direct information about underlying representations, or other kinds of assistance). To be sure, children are also exposed to a great deal of other information, including contextual cues as to the meanings of words. Calamaro and Jarosz's (2015) assumption, which we adopt here, is simply that children can succeed in phonological learning even when such additional information is not present. Some support for this view comes from experimental work that provides evidence for children's ability to acquire key aspects of morpho-phonology, including segmentation (Saffran *et al.* 1996), allomorphy (Gerken *et al.* 2005), and phonological alternations (White *et al.* 2008), all from distributional evidence. We note, in addition, that non-distributional information such as morpheme meanings is more limited in its ability to assist phonological learning than is often assumed in the phonological learning literature. A common assumption made in the literature is that semantic information can teach the learner about the existence of phonological processes. On this common view, when the learner encounters two morphemes with different phonological surface forms that have exactly the same meaning, the learner knows that a phonological process is responsible for the surface difference between them. Semantics is therefore assumed to take the learner a long way towards learning the phonological grammar. We believe that this view overestimates the utility of semantics for the learner because it mistakenly ignores the possibility that two morphemes with the same meaning are not related through phonological processes: namely, it ignores the possibility of suppletion, where two semantically identical forms are stored separately in the lexicon, without being derived from a common lexical entry through any phonological process. Since nobody tells the learner when suppletion is involved, the learner has to figure out the existence of phonological processes itself. We assume that an explanatorily adequate theory needs to account for this aspect of learning as well. However, a more complete characterization of the evidence that children base

their learning on, both in lab settings and during acquisition, awaits further work.

Second, we assume that children can acquire their phonological knowledge even in the face of nontrivial dependencies between morphological segmentation and phonological processes, and we assume that underlying representations may be abstract, in the sense of differing from surface forms even in the absence of conclusive evidence from alternations. Moreover, we take the phonological knowledge that children attain to involve various textbook properties such as opacity and optionality. We discuss each of these aspects of phonological knowledge and learning in turn.

Dependencies between morphological segmentation and phonological processes exist in many affixes and alternations across languages. Vowel harmony in Turkish provides a particularly clear illustration. Focusing on stems such as *ip* ‘rope’ and *kız* ‘girl’ and on the suffixes for the genitive and the plural, the child’s input might consist of surface forms such as *ipler*, *kızlar*, *ipin*, and *kızın*. If the child already knows that vowel harmony applies within such forms, they can undo it and reason that *ler* and *lar* might be underlyingly identical (and similarly for *in* and *ın*). This, in turn can guide the child toward the correct morphological segmentation of the forms:

		‘rope’	‘girl’
(1)	Plural	<i>ip-ler</i>	<i>kız-lar</i>
	Genitive	<i>ip-in</i>	<i>kız-ın</i>

Similarly, if the child already knows the morphological decomposition of these forms, they can reason about the relation of *ler* and *lar* (and similarly for *in* and *ın*), which can guide the child toward a discovery of vowel harmony. However, if the child does not yet know either about the process of vowel harmony or about the morphological decomposition of the surface forms, they will face the challenge of discovering both despite the bidirectional dependencies between the two.

Abstract URs are URs that differ from their surface forms despite insufficient evidence for the discrepancy from alternations. The extent to which URs may be abstract was a matter of much debate in early generative phonology. More recently, abstractness has been argued for

by Alderete and Tesar (2002), McCarthy (2005), and Nevins and Vaux (2007), among others (see also discussion in Krämer 2012). Here, we will assume, conservatively, that abstractness is possible, illustrating with a schematic example, based on an example from Alderete and Tesar (2002), which was in turn modeled after the interaction of stress and epenthesis in Yimas. In this example, stress in bisyllabic words is generally initial, but there are some words, in all of which the first vowel is [i], where stress falls on the second syllable. The following table, showing three possible (and different) words and one impossible form, illustrates:

		Initial vowel = i	Initial vowel = a
(2)	Initial stress	píkut	pákut
	Pen-initial stress	pikút	*pakút

A familiar kind of analysis would posit a pattern of initial stress, where an unstressed initial [i] is always epenthetic:

(3) /pkut/ → |pkút| → [pikút]

According to Alderete and Tesar (2002), however, this generalization is acquired without support from alternations.

Finally, the acquired phonological knowledge should capture speakers' intuitions not just in simple cases but also in more complex patterns, of which we focus here on two: optionality and opacity. An example of optionality is the process of liquid deletion in French, analyzed by Dell (1981) and discussed in some detail below, which allows a word-final liquid to optionally delete in certain environments (as in [tabl]~[tab] for 'table'). An example of opacity is the counter-feeding interaction between nasal deletion and cluster simplification in Catalan (Mascaró 1976). As the following illustrates, word-final nasals sometimes delete in Catalan, as do post-nasal word-final stops, but while the latter process creates an appropriate environment for the former, cluster simplification does not lead to nasal deletion:

(4) kuzí ~ kuzín-s      'cousin.SG ~ cousin.PL'  
 kəlén ~ kəlént̪-ə      'hot.MASC ~ hot.FEM'

To summarize, we take the following to be requirements of any theory that achieves explanatory adequacy in the domain of phonology. It should allow for learning from distributional evidence alone. It

should support the joint learning of morphological segmentation and phonological processes and the learning of abstract URs. And it should handle complex patterns such as optionality and opacity. To be sure, this is just a starting point; we certainly do not wish to suggest that these requirements are all there is to learning in phonology. However, we do believe that it is a meaningful starting point that is relevant for the evaluation of any theory that aims at explanatory adequacy in phonology.

In Sections 3 and 4 below we show that the MDL principle, when coupled with a suitable representational framework (for concreteness, we will use rule-based phonology), favors hypotheses that seem appropriate with respect to the different aspects of the learning challenge considered here. This makes MDL a promising candidate for the child's learning criterion. In Section 5 we argue that other approaches in the literature on learning in phonology have yet to address central aspects of the learning challenge.

## THE PRESENT WORK

3

The current section presents the assumptions behind our learning model. One general assumption that we make is that the child chooses between competing grammars using some kind of evaluation metric. We start, in Section 3.1, by considering two evaluation metrics from the literature – the evaluation metric of the *Sound Pattern of English* (SPE; Chomsky and Halle 1968, p. 334), which aims for grammar economy, and the subset principle, which aims for restrictiveness – in the context of acquiring a single optional phonological rule. We will see that in order to acquire the relevant rule, the child cannot follow grammar economy alone or restrictiveness alone but must instead balance between the two. This balancing of economy and restrictiveness is the essence of the MDL evaluation metric, and while we motivate it here using one simple rule, the very same metric will serve as a good guide for learning whole (though at present artificial) phonological grammars, including the lexicon, the morphological segmentation of forms into stems and affixes, a variety of phonological rules, and both transparent and opaque rule interactions. In order to use the MDL

evaluation metric as a part of an actual phonological learner, we need to adopt explicit representations for phonological grammars. We do this in Section 3.2, where we present the concrete representations we assume and the costs they induce in terms of MDL. Section 3.3 presents a search procedure that will allow us to turn the MDL metric into a full learner, and while our focus in this paper is the MDL metric rather than the full learner, it is through reporting simulations with the learner that we will be able to best illustrate the kind of guidance provided by MDL (in Section 4).

### 3.1

#### *The MDL criterion*

French has an optional process of liquid-deletion word-finally following an obstruent (Dell, 1981). The French-learning child, then, might be exposed to surface forms such as [tabl] and [tab] for ‘table’ and [katr] and [kat] for ‘four’ (but only [gar] and not \*[ga] for ‘train station’, since its liquid does not appear in the right environment for deletion). Suppose that the child uses a simplicity metric such as the one in SPE, which optimizes grammar economy. Restricting our attention here and below to grammars that are licensed by Universal Grammar (UG) and using  $|G|$  to notate the length of a grammar  $G$ , we can state this metric as follows:<sup>1</sup>

- (5) SPE EVALUATION METRIC: If  $G$  and  $G'$  can both generate the data  $D$ , and if  $|G| < |G'|$ , prefer  $G$  to  $G'$

To see how we can use (5), we need to be precise about how  $|\cdot|$  is measured. Anticipating our discussion below, it will be convenient to think of grammars as sitting in computer memory according to a given encoding scheme – a scheme that is provided by UG – with  $|G|$  being the number of bits taken up by  $G$ . In Section 3.2 we will present the details of one specific encoding scheme and show how  $|G|$  is measured within it. For now, however, we will set aside such details as we build toward the MDL criterion.

---

<sup>1</sup>Here and below the grammar  $G$  will be taken to be not just the phonological rules and their ordering but also the lexicon. Thus, by saying that a grammar  $G$  generates the data  $D$ , we mean that every string in  $D$  can be derived as a licit surface form from some UR in the lexicon and the ordered phonological rules.

Early on, the child will store a separate UR for each surface form of the alternating pairs: both /tabl/ and /tab/ for ‘table’; both /katr/ and /kat/ for ‘four’; both /arbr/ and /arb/ for ‘tree’; and so on (along with a single /gar/ for ‘train station’). After seeing a few additional alternating pairs of this kind, however, (5) will lead the child to conclude that for each such pair there is just one UR – /tabl/ for ‘table’, /katr/ for ‘four’, /arbr/ for ‘tree’, and so on – and that an optional phonological rule such as the following applies (where *L* stands for *liquid*):<sup>2</sup>

(6)  $L \rightarrow \emptyset$  (optional)

The rule in (6) adds complexity to the grammar, but this complexity is more than offset by the savings obtained by the elimination of all the *L*-less forms from the lexicon. Consequently, the overall size of the grammar is shorter using (6), and (5) will favor the new grammar.

As mentioned above, however, the actual process of *L*-deletion in French is somewhat more specific than (6) suggests: *L* may be deleted, but only in certain contexts. A more appropriate rule is the following, in which *L*-deletion is restricted to word-final environments following an obstruent:

(7)  $L \rightarrow \emptyset$  /[-son]\_\_# (optional)

And unfortunately, as pointed out by Dell (1981), a child using (5) will fail to acquire the appropriate context for the application of the rule. That is, the child will prefer (6) to the more appropriate (7). This is so since (a) both a grammar *G* using the unrestricted (6) and a grammar *G'* using the restricted (7) can generate the data; and (b) *G* is shorter than *G'* (since specifying the context in (7) adds to the grammar’s length). By the SPE evaluation metric in (5), the child will prefer *G* to *G'*, which is the wrong result. For example, a child using *G* will

---

<sup>2</sup>An even simpler grammar is one in which the lexicon includes just one, empty UR and in which any segment can be inserted by an optional rule. Such a grammar would be an extreme example of a very simple but wildly overgenerating grammar, and we could have used it instead of (6) to illustrate the perils of minimizing  $|G|$  alone in our discussion below. In the interest of keeping the presentation focused on deletion processes, however, we set this grammar aside and start from (6).

erroneously rule in  $L$ -deleted forms such as \*[ga] for /gar/.<sup>3</sup> Moreover, the child will never recover from this error: since the child sees only positive evidence, they will never be forced to leave the simpler but overly inclusive  $G$ .

The problem is quite general, as discussed by Braine (1971) and Baker (1979), and goes well beyond phonology: a child guided solely by a preference for grammar economy, as in the SPE evaluation metric in (5), will fail to learn the contexts for optional rules. Just as in the example of optional  $L$ -deletion, a grammar  $G$  in which an optional rule  $R$  has no context will generally be both simpler and more inclusive than a minimal variant  $G'$  in which the optional rule does have a context. If  $G'$  is the correct grammar, both grammars will be able to generate the input data:  $G'$  since it is the correct grammar, and  $G$  since its *language* – that is, the set of all licit forms according to the lexicon and rules of  $G$  – is a superset of the language of  $G'$ . By (5), then, the child will incorrectly prefer the simpler  $G$  to  $G'$  and – since the child will not receive negative evidence – will never recover from this error.

One solution to this predicament – the one advocated by Dell (1981) and adopted in much later work – is to change the evaluation metric from one that favors simple grammars to one that favors restrictive ones, where restrictiveness is captured in terms of subsethood:  $G$  is more restrictive than  $G'$  if its language is a subset of the language of  $G'$ .<sup>4</sup> This solution, also known as the *subset principle* (Berwick

---

<sup>3</sup>In fact, as mentioned in footnote 2, a preference for grammar economy will lead the learner to even more extreme solutions if left unchecked. In particular, consider a grammar (as in footnote 2) that has an optional epenthesis rule for each segment that appears in the data and a lexicon that consists only of the empty string. Such a grammar can generate the data and is extremely short to state. Unless it is blocked by some other principle, this grammar will be preferred by (5) to both  $G$  and  $G'$ .

<sup>4</sup>Other ways of cashing out the informal idea of restrictiveness have been proposed in the literature. Within Optimality Theory (Prince and Smolensky 1993), for example, restrictiveness is often interpreted as subsethood not of the languages of the original grammars  $G$  and  $G'$  but rather of the languages of variants of  $G$  and  $G'$  in which the lexicon is replaced with the set  $\Sigma^*$  of all possible strings over the alphabet  $\Sigma$  in which the lexicon is written (see Smolensky 1996). The MDL metric, which we will present and argue for below, implements restrictive-



1985; Wexler and Manzini 1987; Hale and Reiss 2003, 2008), directs the learner to never choose a grammar for a superset language when a grammar for a proper subset is compatible with the data:<sup>5</sup>

- (8) SUBSET EVALUATION METRIC: If  $G$  and  $G'$  can both generate the data  $D$ , and if the language of  $G$  is a proper subset of the language of  $G'$ , prefer  $G$  to  $G'$

A child following (8) will always choose from among the grammars sanctioned by UG and whose language is compatible with the data a grammar whose language is minimal in terms of subsethood. Such a child will therefore avoid the overgeneralization problem. In the case of optional  $L$ -deletion in French, the grammar with the unrestricted (6) generates a language that is a strict superset of the one with the restricted (7), and both grammars generate the data  $D$ ; consequently, the unrestricted (6) will be rejected and the restricted (6) chosen, which is the correct result.

While choosing correctly between (6) and (7), the subset principle gives rise to a problem of undergeneralization – the mirror image of the overgeneralization problem of the SPE simplicity metric – and does not offer a general solution for learning. To see the problem in the case of French  $L$ -deletion, consider the situation of a learner who has heard a surface form such as [sabl] but, accidentally, has not yet heard its  $L$ -elided variant [sab] (both for the UR /sabl/ ‘sand’). If the learner has heard sufficiently many other pairs differing only in whether they have a final liquid, we would expect them to adopt (7), even if for /sabl/ only one member of the pair has been observed so far. That is, we would like the learner to generalize beyond the data in this case. But if the learner follows the subset principle, this will not be possible: with (7), the language will include also the  $L$ -deleted form [sab], which makes the language a strict superset of the language of a grammar that does not generate [sab]. One example of such an

---

ness in yet another way, by comparing how easy it is to specify the actual input data using  $G$  and  $G'$ : if the data can be more easily specified using  $G$  than using  $G'$ , then  $G$  is the more restrictive grammar of the two.

<sup>5</sup>As Baker (1979) notes, Braine's (1971) alternative to the SPE evaluation metric, while stated in procedural terms, has a similar effect to a restrictiveness metric.

overly restrictive grammar is one without any deletion rules and with a lexicon that has separate URs for each of the  $L$ -variants that have been seen in the input data. For a learner that follows the subset principle, the only way to avoid such an overly restrictive grammar is if it is not licensed by UG. On most theories of UG, however, a memorizing and overly specific grammar is perfectly capable of being represented. Consequently, the learner will fail to choose the correct and more permissive (7). In other words, as long as UG makes available overly restrictive grammars, a single accidental gap is enough to prevent a learner following the subset principle from making what seems like a reasonable generalization.

We have seen that minimizing  $|G|$ , as in the SPE evaluation metric, makes the child generalize; when left unchecked, however, it leads to overgeneralization. Meanwhile, restrictiveness (as in the subset principle) protects from overgeneralization, but on its own prevents useful generalizations. It seems sensible, then, to try to balance the two principles against each other: look for a grammar that is both reasonably small and reasonably restrictive. This is exactly the idea behind Minimal Description Length (MDL; Rissanen 1978), which we will adopt here.<sup>6</sup> To make it work, however, we need to specify how we quantify both grammar size and restrictiveness and how the two are balanced. The insight of MDL – building on the work of Solomonoff (1964a,b), Kolmogorov (1965), and Chaitin (1966) – is that we can think of restrictiveness as another simplicity criterion and combine it naturally with grammar economy. As above, for grammar economy we will consider  $G$  as sitting in computer memory according to a given encoding – as specified by UG – and measure  $|G|$  in terms of how many bits the storage of  $G$  takes up. Restrictiveness, meanwhile, will be thought of in terms of how simple it is to describe the data,  $D$ , given the grammar,  $G$ . We will use the notation  $D : G$ , somewhat loosely, for the shortest description of  $D$  given  $G$  (loosely because there might be multiple such shortest descriptions), and we will notate the length of the shortest description of  $D$  given  $G$  as  $|D : G|$ .<sup>7</sup> To see how  $|D : G|$

---

<sup>6</sup> See also the closely related idea of Minimal Message Length of Wallace and Boulton (1968).

<sup>7</sup> In what follows, we will consider  $D$  to be the actual data sequence that the learner is exposed to. Consequently,  $D : G$  will be the description of those

is measured given a grammar  $G$ , consider again the case of optional  $L$ -deletion. Suppose that the learner has acquired a lexicon with the single UR /tabl/ and an optional rule such as (6) or (7). To describe an instance of the surface form [tabl] or the surface form [tab], we need to first specify the UR /tabl/ and then specify whether  $L$ -deletion has applied (for [tab]) or not (for [tabl]). Specifying the UR /tabl/ involves a choice from among the URs. In general, the greater the number of URs from which we choose, the longer the specification of the UR we have selected. A convenient way of specifying such choices – and one that will allow us to directly balance the length of  $D : G$  against that of the grammar  $G$  – is using bits. A single bit encodes one binary choice, and as the number of bits grows, the number of choices that can be stated grows (exponentially) with it. For example, if there are just two possible URs, we can specify the choice using one bit. With four URs in the lexicon, we now need about two bits to specify each choice. And so on.<sup>8</sup> The optional  $L$ -deletion rule requires the further specification of whether it applied or not, which can be stated as one additional bit (perhaps 0 to specify that the rule did not apply and 1 to specify that it did). These specifications for the different surface forms in the input data  $D$  are accumulated to provide the complete  $D : G$ , the encoding of the specific input data  $D$  given the grammar  $G$ .

---

actual input tokens given the grammar. This choice is made for concreteness and in order to keep the presentation simple. A different possibility would be to abstract away from individual tokens and consider only the types – that is, the distinct surface forms – rather than the tokens. It is also possible to define the restrictiveness factor  $|D : G|$  in terms of a combined measure of types and tokens. We will not attempt to investigate these choices and their implications for learning within this paper (see Goldwater *et al.* 2006, Endress and Hauser 2011, and Yang 2016 for relevant discussion).

<sup>8</sup>Exactly how many bits are needed for each choice will depend on the specific grammar  $G$ , relative to which the choices are made. In Section 3.2 we show how  $D : G$  is stated relative to the grammars presented in that section. For similar considerations regarding the measurement of  $|G|$  and  $|D : G|$  in bits but within constraint-based phonology see Rasin and Katzir 2016. We further note that the number of bits used for a given choice point need not be uniform. In general, the optimal cost of each choice  $x$  in bits will be  $-\lg P(x)$  (that is, minus the logarithm base two of the probability of  $x$ ). A fixed number of bits per choice point is optimal only if the probability distribution at each choice point is uniform.

We can now see how the motivation for restricting the context for optional  $L$ -deletion can be stated in terms of simplicity. If  $L$ -deletion were not optional – if it always applied or if it never applied – the final bit would have been unnecessary for the specification of the relevant surface forms: selecting a UR would have fully determined the surface form. For URs like /tabl/ and /katr/,  $L$ -deletion is optional, and the extra bit of the appropriate rule cannot be avoided. But for /gar/  $L$ -deletion never applies, so paying an extra bit for each occurrence is an unnecessary expense. The unrestricted (6) forces us to pay this unnecessary expense: the optional rule is applicable whenever a UR is chosen that contains liquids (and for each occurrence of a liquid within such a UR), including URs such as /gar/ that do not allow for  $L$ -deletion, so a bit specifying whether the rule applies is always required, leading to  $D : G$  that is longer than needed. The more restrictive (7), on the other hand, makes us pay the extra bit only when an appropriate UR such as /tabl/ is chosen but not when /gar/ is chosen. Consequently, (7) leads to a shorter  $D : G$ .

Having recast the notion of restrictiveness in terms of simplicity (specifically, the simplicity of  $D : G$ ), we can directly combine it with simplicity of grammar: instead of minimizing  $|G|$  alone, as in the SPE evaluation metric, we can now minimize the sum of the two quantities,  $|G| + |D : G|$ , thus balancing between the goal of a simple, general grammar and a restrictive one.

- (9) MDL EVALUATION METRIC: If  $G$  and  $G'$  can both generate the data  $D$ , and if  $|G| + |D : G| < |G'| + |D : G'|$ , prefer  $G$  to  $G'$

Combining grammar economy with restrictiveness in terms of the subset principle as stated in (8) is a nontrivial challenge. Combining it with the reformulation of restrictiveness in terms of  $|D : G|$ , on the other hand, is straightforward, as (9) shows. Moreover, the MDL quantity  $|G| + |D : G|$  has a direct interpretation in terms of quantities that are arguably available to the learner, as discussed in Katzir 2014 and Rasin and Katzir 2020. Grammars are stored in memory according to the specifications provided by UG, and  $|G|$  is therefore simply the amount of memory required to store  $G$  using this specification. As for  $|D : G|$ , any given grammar  $G$  considered by the learner and compatible with  $D$  can presumably be used to parse  $D$ , and if this parse is stored in memory, its storage space is  $|D : G|$ . This makes  $|G| + |D : G|$

nothing more than the overall storage space used for keeping  $G$  and its (shortest) parse of  $D$  in memory. This makes MDL a natural evaluation criterion that uses only quantities that are available to the learner with minimal stipulation beyond what is already needed to represent grammars and use them to parse the data.<sup>9</sup>

Let us now return to the  $L$ -deletion example and see how MDL leads to an adequate level of generalization. As discussed above, storing a single UR for pairs like [tabl]/[tab] and [katr]/[kat] will shorten  $|G|$  sufficiently (given a large enough number of such pairs) to justify adding an optional rule of  $L$ -deletion to  $G$ , just as with the SPE evaluation metric. As for the precise form of the rule, the simultaneous consideration of both  $|G|$  and  $|D : G|$ , as in (9), will mean that the more complex rule in (7) will eventually be chosen over the unrestricted (6), despite its increased  $|G|$ . The reason is that after sufficiently many instances of words like [gar] have been encountered, the savings in terms of  $|D : G|$  obtained with (7) – since no bit will need to be spent when a UR such as /gar/ is chosen – will more than outweigh the increase in  $|G|$ . Figure 1 illustrates. The MDL metric in (9) thus allows the child to generalize but protects them from overgeneralizing.

Note that, differently from the case of restrictiveness-only (as in the subset principle), the MDL metric has the means to generalize beyond the data even in the face of certain gaps in the input. Consider again the situation of a learner who has heard the form [sabl] but has not (yet) heard its  $L$ -deleted variant [sab]. We saw earlier how this kind of gap in the input data will prevent a restrictiveness-only learner from generalizing correctly. For an MDL learner (that is, a learner that relies on the MDL metric to choose between hypotheses), the added restrictiveness of ruling out [sab] is weighed against the added complexity in stating a grammar that does that while still accounting for

---

<sup>9</sup>A reviewer suggests combining  $|G|$  not with  $|D : G|$  but rather with  $|L(G)|$ , the cardinality of the language of  $G$ . We note, however, that using  $|L(G)|$  as a proxy for restrictiveness will only be useful when the language of the target grammar is finite, and this assumption is problematic even within morpho-phonology due the possibility of unbounded processes of affixation. And even if the languages under consideration are assumed to be finite, computing  $|L(G)|$  strikes us as significantly more challenging than using  $|D : G|$ , a quantity that as just discussed is presumably already available to the learner.

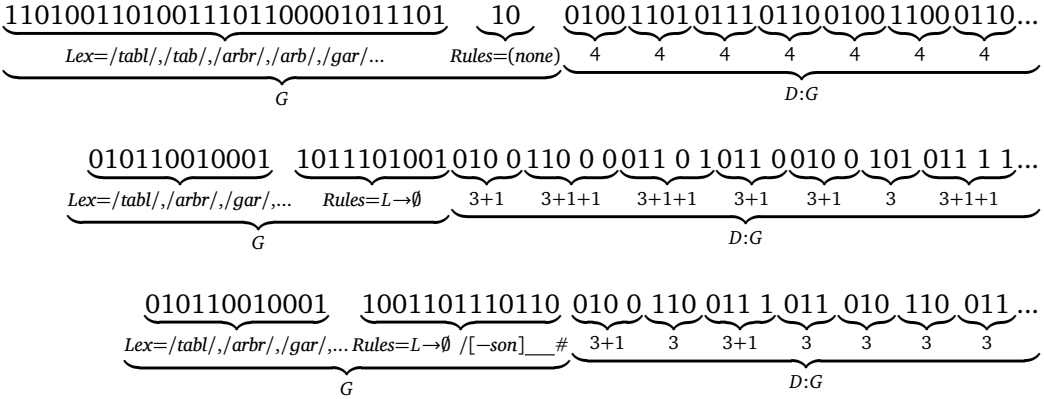


Figure 1: Schematic illustration of three hypotheses. (The order of URs in the lexicon and of tokens in  $D : G$  are unrelated.) Introducing a naive lexicon (*top*), in which [tabl] and [tab] have distinct URs results in a complex grammar. Capturing optional  $L$ -deletion with (6) allows the grammar to be simplified (*middle*): the complexity of the rule is outweighed by the savings of eliminating unnecessary URs. Moreover, since there are now fewer URs than with the naive lexicon, each UR can be specified more succinctly. However, an additional bit is needed for specifying the actual surface form of each occurrence of  $L$  in a UR (for each surface token of that UR). Finally, restricting the context of  $L$ -deletion, using (7), allows us to limit the extra bit to just those URs that require it (*bottom*):  $/tabl/$  but not  $/gar/$

both [tabl] and [tab]. In the present case, a grammar that rules out [sabl] will be quite complex: it might dispense with  $L$ -deletion and resort to memorizing each observed surface form using a separate UR; or it might state a highly involved rule (or system of rules) that license  $L$ -deletion in those forms where both variants of a pair has been observed. Either way, the result will be a complex grammar that does not justify the minimal savings obtained by not having to specify whether  $L$ -deletion has applied for the single occasion when the UR  $/sabl/$  was chosen. (This is very different from the case of [gar], where preventing inappropriate  $L$ -deletion involved only a slight increase in grammar size, and where there were sufficiently many relevant instances of  $L$  in non-deleting environments to justify the added complexity.) Consequently, the accidental gap arising from seeing an occurrence of [sabl] without an instance of [sabl] will not prevent the MDL learner from keeping the rule of  $L$ -deletion in (9), thus generalizing beyond the data, which seems to be the correct result.

Suppose now that the learner sees not just one instance of [sabl] but rather many instances, still without any instance of [sab]. In this case, the absence of [sab] will start looking less like an accident of the specific data sequence seen so far and more like a systematic fact of French that needs to be captured. The MDL learner allows us to make this intuition precise: with sufficiently many occurrences of [sabl], the extra bit that is needed to state for each occurrence that /sabl/ does not undergo optional *L*-deletion results in an increase to  $|D : G|$  that is big enough to justify blocking *L*-deletion for this UR. How exactly *L*-deletion is blocked will depend on the representations available to the learner. For example, if these representations offer a general way to mark exceptions to rules, the learner might choose to mark /sabl/ as an exception to *L*-deletion. If such a method is not available, the learner might choose to block *L*-deletion in a more *ad hoc* way. For example, the learner might decide to add a special segment at the end of the UR (e.g., storing the relevant UR as /sablx/), thus preventing the *L* under consideration from appearing in the right context for deletion, along with a rule that deletes that special segment and is ordered after *L*-deletion.

Before proceeding, we note that in the discussion above we assumed that the input to the learner is a sequence of surface forms of words in isolation. If further information is available to the learner, such as the order of words in sentences or representations of scenes in which words are uttered, the decision of the learner regarding which forms to collapse using phonological rules can change. For example, a learner considering a small portion of the English lexicon containing ‘spare’, ‘pear’, ‘spit’, ‘pit’, ‘stick’, ‘tick’, and similar pairs might mistakenly collapse these pairs with the aid of an optional rule of [s]-deletion before [p] word-initially. By considering not just words in isolation but also the linguistic and extra-linguistic contexts in which they appear, however, an MDL learner will be justified in moving to a more complex grammar that does not collapse the relevant pairs but rather represents them using distinct URs in the lexicon.

The balancing of economy and restrictiveness has made MDL – and the closely related Bayesian approach to learning – helpful across a range of grammar induction tasks, in works such as Horning (1969), Berwick (1982), Ellison (1994), Rissanen and Ristad (1994), Stolcke (1994), Grünwald (1996), de Marcken (1996), Brent (1999),

and Clark (2001), among others.<sup>10</sup> Recently, Rasin and Katzir (2016) have used MDL to show how phonological grammars can be acquired distributionally within constraint-based phonology, and Rasin and Katzir (2018, 2020) have discussed the acquisition of abstract URs using MDL. The present work extends this approach, using rule-based phonology as a concrete representational framework. In particular, we will show how the same MDL metric that supported the correct generalization in the case of the optional rule of *L*-deletion in French, as discussed above, will support the acquisition of whole phonological grammars, including the lexicon, the segmentation of forms into stems and affixes, a variety of phonological rules, and both transparent and opaque rule interactions. The simulations illustrating the use of MDL for the acquisition of phonological grammars – at present, using small corpora generated from artificial grammars – will be presented in Section 4. Before that, in the remainder of the present section, we describe the phonological representations that we assume, in order to make explicit their contribution to the MDL score, and we describe the search procedure we use to traverse the space of possible grammars.

### 3.2

#### *Representations*

As is standard, we assume that segments in phonological rules are represented not atomically but as feature bundles.<sup>11</sup> For convenience, each simulation below works with a feature table that makes distinctions that are relevant to the phenomenon at hand, but we remain agnostic here as to whether learners start with a large innate table or acquire language-specific tables at an earlier stage. To illustrate, the feature table in Table 1 will be used for those simulations that are based on English.

---

<sup>10</sup>MDL and Bayesian grammar induction are almost equivalent. There are some differences, such as MDL's use of the shortest encoding of *D* given *G*, which corresponds to the maximal probability of a parse of *D* given *G*, while Bayesian learning marginalizes over all parses. As far as we can tell, however, such differences are irrelevant to the examples discussed here, and we will treat MDL and Bayesian inference as essentially the same for the purposes of this paper.

<sup>11</sup>In principle, the same holds also for the lexicon, though in the implementation reported here, the representation of segments in the lexicon does not explicitly use feature bundles.



Table 1: Feature table

	<i>cons</i>	<i>voice</i>	<i>cont</i>	<i>coronal</i>	<i>low</i>	<i>high</i>	<i>back</i>	<i>son</i>	<i>lateral</i>	<i>labial</i>	<i>strident</i>
d	+	+	+	-	-	-	-	-	-	-	-
t	+	-	+	-	-	-	-	-	-	-	-
z	+	+	+	+	-	-	-	-	-	-	+
s	+	-	+	+	-	-	-	-	-	-	+
g	+	+	-	-	-	-	-	-	-	-	-
k	+	-	-	-	-	-	-	-	-	-	-
b	+	+	-	-	-	-	-	-	-	+	-
p	+	-	-	-	-	-	-	-	-	+	-
m	+	+	-	-	-	-	-	+	-	+	-
n	+	+	+	-	-	-	-	+	-	-	-
r	+	+	+	+	-	-	-	+	-	-	-
l	+	+	+	+	-	-	-	+	+	-	-
a	-	+	+	+	+	-	+	+	-	-	-
o	-	+	+	+	-	-	+	+	-	-	-
e	-	+	+	+	-	-	-	+	-	-	-
i	-	+	+	+	-	+	-	+	-	-	-
u	-	+	+	+	-	+	+	+	-	-	-

Phonological rules

3.2.1

Feature bundles based on feature tables such as the one in Table 1 are used to state the phonological rules. The general form of rules is as follows, where  $A, B$  are feature bundles or  $\emptyset$ ;  $X, Y$  are (possibly empty) sequences of feature bundles; and *optional?* is a boolean variable specifying whether the rule is obligatory or optional (Figure 2).

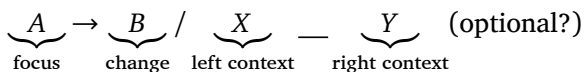


Figure 2:  
Rule format

The following, for example, is an optional phonological rule of vowel harmony that fronts a vowel before another front vowel when the two are separated by arbitrarily many consonants, stated in text-book notation in (10a) and in string notation (more convenient for the purposes of the conversion to bits below, and using various delimiters, marked with # with certain subscripts and discussed shortly) in (10b).

- (10) Vowel harmony rule  
 a. Textbook notation

$$[-cons] \rightarrow [-back] / \_ [+cons]^* \begin{bmatrix} -cons \\ -back \end{bmatrix} \text{ (optional)}$$

- b. String notation

$$-cons\#_{rc} -back\#_{rc}\#_{rc} +cons*\#_b -cons\#_f -back\#_{rc}1\#_{rc}$$

As discussed informally in Section 3.1 above, determining both  $|G|$  and  $|D : G|$  for purposes of MDL is done in bits, where each bit represents a single binary choice. In the simple representations that we use in this paper, all possible outcomes at any particular choice point (whether binary or otherwise) are treated as equally easy to encode. For purposes of presentation, we will first discuss a particularly simple representation in which at any given choice point, the different outcomes are not just equally easy on average to encode but actually have fixed, equal length codes. This will allow us to discuss the various encodings in terms of fixed conversion tables in which if there are  $n$  possible outcomes, each will be assigned a code whose length in bits is  $\lceil \lg n \rceil$  (that is, the logarithm base two of  $n$ , rounded up to the closest integer). In our actual simulations, presented in Section 4, we will deviate from the encoding presented below by allowing non-integral code lengths, taking  $\lg n$  rather than  $\lceil \lg n \rceil$  as the code length for an  $n$ -ary choice point.<sup>12</sup>

Within the simplified representational framework just described, determining the length in bits of a single phonological rule for the purposes of MDL is done by using a conversion table that states the codes for the possible elements within phonological rules. An example of a possible conversion table appears in Table 2.

---

<sup>12</sup>The reason for this change is that the encoding used in the current section, using  $\lceil \lg n \rceil$ , is highly sensitive to changes in which the number of outcomes at a given choice point crosses a power of 2 (which is where  $\lceil \lg n \rceil$  changes). By taking  $\lg n$  instead of  $\lceil \lg n \rceil$ , this unhelpful sensitivity to powers of 2 is avoided. On the other hand, using conversion tables with fixed code lengths, corresponding to  $\lceil \lg n \rceil$ , allows us to keep the presentation considerably simpler than if we had to discuss  $\lg n$  in terms of code lengths. We therefore keep the presentationally simpler  $\lceil \lg n \rceil$  for the current section and the more robust  $\lg n$  for the actual simulations.

Symbol	Code	Symbol	Code
$\#_f$ (feature)	0000	cons	0110
$\#_b$ (bundle)	0001	voice	0111
$\#_{rc}$ (rule component)	0010	velar	1000
+	0011	back	1001
-	0100	...	...
*	0101		

Table 2:  
Conversion table  
for rules

Using the conversion table in Table 2, we can encode the phonological rule of vowel harmony (in (10) above) by converting each element in the string representation in (10b) into bits according to Table 2 and concatenating the codes. To ensure unique readability, we use delimiters to mark the end of the description of features within a feature bundle ( $\#_f$ ), feature bundles within the left and right contexts of a rule ( $\#_b$ ), and the rule’s components ( $\#_{rc}$ ; in terms of the notation in Figure 2, an occurrence of  $\#_{rc}$  occurs after each of A, B, X, Y, and optional?). The following is the result, and its length is 73 bits:

(11) Vowel harmony rule (bit representation):

$$\begin{array}{cccccccccccccccc}
 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\
 \underbrace{\hspace{1.5em}}_{-} & \underbrace{\hspace{1.5em}}_{cons} & \underbrace{\hspace{1.5em}}_{\#_{rc}} & \underbrace{\hspace{1.5em}}_{-} & \underbrace{\hspace{1.5em}}_{back} & \underbrace{\hspace{1.5em}}_{\#_{rc}} & \underbrace{\hspace{1.5em}}_{\#_{rc}} & \underbrace{\hspace{1.5em}}_{\#_{rc}} & \underbrace{\hspace{1.5em}}_{+} & \underbrace{\hspace{1.5em}}_{cons} & \underbrace{\hspace{1.5em}}_{*} & \underbrace{\hspace{1.5em}}_{\#_b}
 \end{array}$$
  

$$\begin{array}{cccccccccccc}
 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
 \underbrace{\hspace{1.5em}}_{-} & \underbrace{\hspace{1.5em}}_{cons} & \underbrace{\hspace{1.5em}}_{\#_f} & \underbrace{\hspace{1.5em}}_{-} & \underbrace{\hspace{1.5em}}_{back} & \underbrace{\hspace{1.5em}}_{\#_{rc}} & \underbrace{\hspace{1.5em}}_{1} & \underbrace{\hspace{1.5em}}_{\#_{rc}}
 \end{array}$$

A phonological rule system is a sequence of phonological rules. Since the encoding described above allows us to determine from the bit representation where each rule ends, we can specify a phonological rule system by concatenating the encodings of the individual rules while maintaining unique readability with no further delimiters. The ordering of the rules is the order in which they are specified, from left to right. At the end of the entire rule system another  $\#_{rc}$  is added.

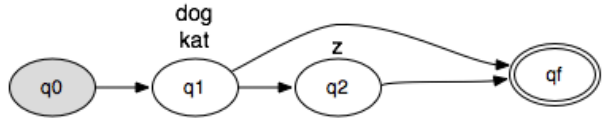
Lexicon

3.2.2

The lexicon contains the UR of each morpheme. Since morphemes combine selectively and in specific orders, some information about morpheme combinations must be encoded. We encode this information using Hidden Markov Models (HMMs), where morphemes are

listed in the emission table for specific states, and the possible combinations are defined by state transitions. A simple example is provided in Figure 3.

Figure 3:  
An HMM representation  
of a lexicon



The HMM in Figure 3 defines a lexicon with two kinds of morphemes: the stems /dog/ and /kat/, and the optional suffix /z/. As with rules, description length is not calculated directly for the standard, graphical notation of the HMM but rather for a bit-string form. As before, we start with an intermediate string representation for the HMM, as presented in Figure 4 (derived from the concatenation of the string representations for the different states, as listed in Table 3; the delimiter #<sub>s</sub> marks the end of the list of outgoing edges from a state and #<sub>w</sub> marks the end of each emitted word; another #<sub>w</sub> is added at the end of each state). Within the simplified representational framework described earlier, we convert the string to a bit-string using a conversion table, as in Table 4. As before, all choices at a given point are uniform, with the same code length for all possible selections at that point ( $\lceil \lg n \rceil$  if there are  $n$  possible choices). As discussed above, the actual simulations presented in Section 4 use  $\lg n$  rather than  $\lceil \lg n \rceil$  as the code length.

Table 3:  
String representations  
of HMM states

State	Encoding string
q <sub>0</sub>	q <sub>0</sub> q <sub>1</sub> # <sub>s</sub> # <sub>w</sub>
q <sub>1</sub>	q <sub>1</sub> q <sub>2</sub> q <sub>f</sub> # <sub>s</sub> dog# <sub>w</sub> kat# <sub>w</sub> # <sub>w</sub>
q <sub>2</sub>	q <sub>2</sub> q <sub>f</sub> # <sub>s</sub> z# <sub>w</sub> # <sub>w</sub>

Figure 4:  
String representation  
of an HMM

q<sub>0</sub>q<sub>1</sub>#<sub>s</sub>#<sub>w</sub>#<sub>w</sub>q<sub>1</sub>q<sub>2</sub>q<sub>f</sub>#<sub>s</sub>dog#<sub>w</sub>kat#<sub>w</sub>#<sub>w</sub>q<sub>2</sub>q<sub>f</sub>#<sub>s</sub>z#<sub>w</sub>#<sub>w</sub>

State	Code	Segment	Code
$\#_s$	000	$\#_w$	0000
$q_0$	001	a	0001
$q_1$	010	k	0010
$q_2$	011	d	0011
$q_f$	100	...	...

Table 4:  
Conversion table for HMM

Data given the grammar

3.2.3

Turning to the encoding of the data given the grammar,  $D : G$ , recall that the generation of a surface form involves concatenating several morphemes in a specific order and applying a sequence of phonological rules. Given the grammar as described above, specifying a surface form will therefore involve: (a) specifying the sequence of morphemes (as a sequence of choices within the lexicon, repeatedly stating the code for a morpheme according to the table in the current state followed by the code to make the transition to the next state); and (b) specifying the code for each application of an optional rule. Note that obligatory rules do not require any statement to make them apply.

Given a surface form, we need to determine the best way to derive it from the grammar in terms of code length. A naive approach to this parsing task would be to try all the ways to generate a surface form from the grammar. Even with simple grammars, however, this approach can be unfeasible. Instead, we compile the lexicon and the rules into a weighted finite-state transducer (FST) that allows us to obtain the best derivation using dynamic programming. The compilation of the rules relies on Kaplan and Kay (1994), and the FST is created by combining the rules with the HMM representing the lexicon using transducer composition.

Let us illustrate the encoding of best derivations in the case of the form  $[k^h\text{æts}]$  – actually, of the simpler  $[k\text{æts}]$  – using the FSTs for two simple grammars. First, consider the FST in Figure 5, which corresponds to a grammar with the lexicon in Figure 6 and no phonological rules. Using this FST, encoding the word  $[k^h\text{æts}]/[k\text{æts}]$  requires 16 bits. The initial transition from  $q_0$  to  $q_1$  is deterministic and costs zero bits. After that, each of the four segments costs four bits: three bits to specify the segment itself (since there are eight outgoing edges

Figure 5:  
Naive FST

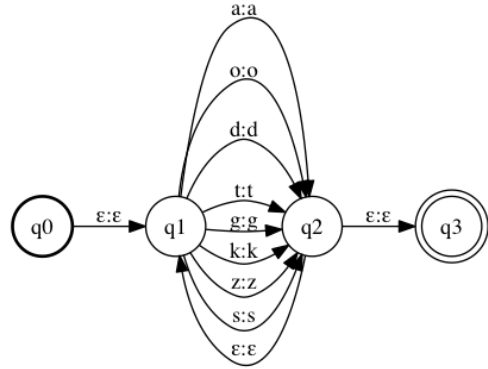


Figure 6:  
Lexicon corresponding  
to the naive FST

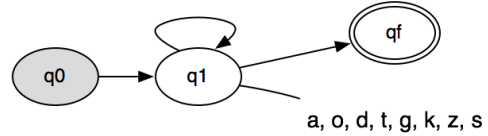


Figure 7:  
Encoding of a surface form  
using the naive FST

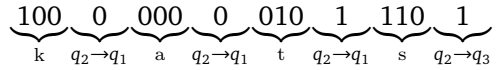


Table 5:  
Conversion table  
for naive FST

State $q_0$		State $q_1$		State $q_2$	
Arc	Code	Arc	Code	Arc	Code
$(-,q_1)$	$\epsilon$	$(a,q_2)$	000	$(-,q_1)$	0
		$(o,q_2)$	001	$(-,q_3)$	1
		$(t,q_2)$	010		
		$(d,q_2)$	011		
		...	...		

from  $q_1$ ) followed by one bit to specify the transition from  $q_2$  (loop back to  $q_1$  or proceed to  $q_3$ ). The encoding, using the conversion table in Table 5, is in Figure 7.<sup>13</sup>

<sup>13</sup>Specifying  $[k^h\text{æts}]$  requires handling the aspiration of the initial segment. Since the relevant rule is obligatory, the same number of bits is required as for  $[k\text{æts}]$ , though the FST is slightly more complex.

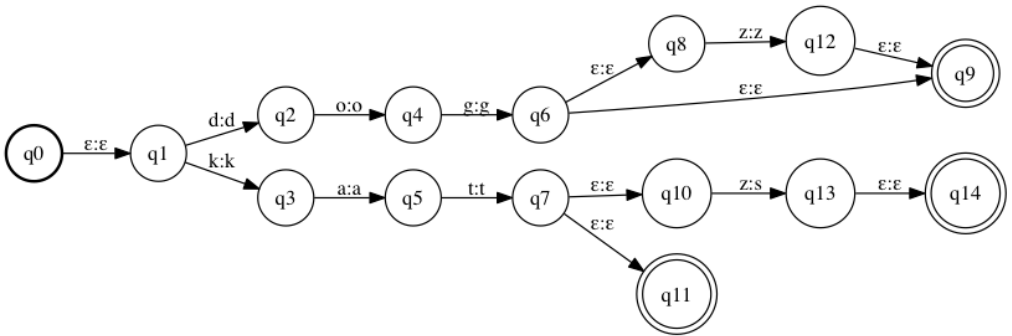


Figure 8: A more complex FST

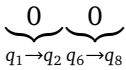


Figure 9:

Encoding of a surface form using the more complex FST

Consider now the more complex FST in Figure 8, which corresponds to a grammar with the lexicon in Figure 3 and the English voicing assimilation rule. This FST corresponds to a more restrictive grammar: differently from the simpler FST in Figure 5, the present FST can only generate a handful of surface forms. Consequently, the present FST offers a shorter  $D : G$ . Specifically, since specifying  $[k^hæts]/[kæts]$  requires making only two choices in the FST, both of them binary, it allows us to encode the relevant string using only 2 bits, as in Figure 9.

### Search

### 3.3

Above we saw how encoding length,  $|G| + |D : G|$ , is derived for any specific hypothesis  $G$ . In order to use it for learning, the learner can search through the space of possible hypotheses provided by UG and look for a hypothesis that minimizes encoding length. We do not wish to make any claims about the search that the human learner might perform: our only claim in this paper concerns the MDL evaluation metric as a promising guide in comparing hypotheses. However, in order to show how this metric can guide the learner not just in the minimal comparisons discussed above but also when the learner faces a large space of possible hypotheses, we must combine the metric with

some search procedure. Since the hypothesis space is big – infinitely so in principle – an exhaustive search is out of the question, and a less naive option must be used. For concreteness, we adopt a genetic algorithm (GA), a general strategy that supports searching through complicated spaces that involve multiple local optima (Holland 1975).

The search starts with a random population of hypotheses that are generated by randomly selecting a lexicon and a set of ordered rules for each hypothesis. Individual hypotheses are selected for the next generation based on their fitness. The fitness of a hypothesis  $G$  equals  $|G|+|D:G|$ , the encoding length derived for it. Once a set of hypotheses is selected for the next generation, each pair of hypotheses is crossed-over to produce two offspring which replace their parents, and each offspring undergoes a random mutation to either its lexicon or its rule set. The simulation ends after a specified number of generations. The fittest hypothesis in the last generation is reported below as the final grammar.<sup>14</sup>

## 4

## SIMULATIONS

The present section provides several simulations in which the MDL learner described in Section 3 is faced with unanalyzed data exhibiting various linguistically-relevant patterns.<sup>15</sup> We are not able to test the learner on real-life corpora at this point: both the size of the relevant part of the search space and the time it takes to parse each hypothesis during the search grow rapidly with the size and complexity of the corpus. Instead, we provide a proof-of-concept demonstration, using small datasets generated by artificial grammars that incorporate phonologically interesting dependencies. We return to this matter in Section 6. To simulate a larger corpus, we multiply  $|D:G|$  by 10 in the simulations reported below (the effect is similar to presenting the learner with each word 10 times). The one exception to the multiplication of  $|D:G|$  by 10 is the simulations in Section 4.1 for which we use

---

<sup>14</sup>For a detailed discussion of the search procedure see Lan (2018).

<sup>15</sup>The code for the simulations is available at [https://github.com/taucompling/morphophonology\\_spe](https://github.com/taucompling/morphophonology_spe).



different multipliers, as discussed below. Also with the exception of Section 4.1, each simulation allowed for between 1 and 5 states in the HMM, between 0 and 5 phonological rules, and between 0 and 2 feature bundles in both the left context and the right context of each rule.

Section 4.1 illustrates our learner's acquisition of optionality, using a dataset based on the case of optional French *L*-deletion discussed above. Section 4.2 uses a dataset based on /-z/-affixation in English to illustrate the joint acquisition of affixation and phonological processes. Section 4.3 extends the results of Section 4.2 by showing how the learner can acquire two rules and their ordering in the case of transparent rule interaction. Section 4.3 modifies the English-based dataset to one that involves counterbleeding opacity and shows that the MDL learner succeeds in this case as well. Section 4.5 shows that the MDL learner succeeds on a case of counterfeeding opacity modeled after the interaction of two processes in Catalan.

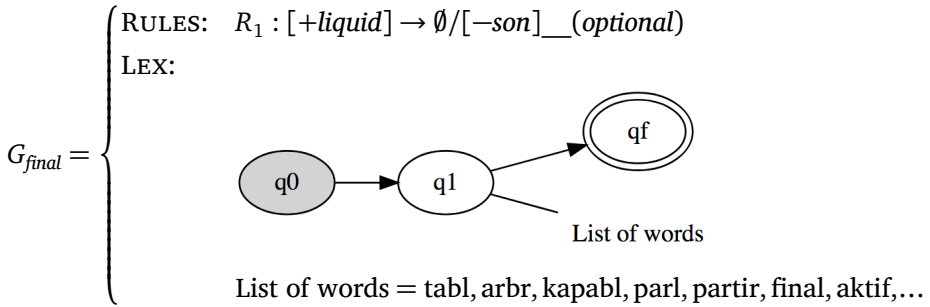
### *Optionality*

### 4.1

The first dataset shows a pattern modeled after French *L*-deletion (Dell, 1981) and is designed to test the learner on the problem of restricted optionality. As discussed in Section 3.1, the challenge for the learner is to strike the right balance between economy and restrictiveness. The learner needs to generalize beyond the data and conclude that for each pair like [tab]–[tabl] there is a single UR, and that a rule of *L*-deletion optionally applies. But the learner must not overgeneralize and should restrict *L*-deletion to only apply after obstruents, despite the added complexity of specifying the restricted environment in the description of the rule.

The data presented to the learner in the present simulation consisted of 91 words, including 33 collapsible pairs (since the task in our simulations is the acquisition of a grammar from distributional evidence alone, from the learner's perspective the data are an unstructured sequence of surface forms: the learner does not know that surface forms like [tab] and [tabl] are related in any way). A sample of the data is given in (12).

(12) tab, tabl, arb, arbr, kapab, kapabl, parl, partir, final, aktif, ...



Description length:  $|G_{final}| + |D:G_{final}| = 29,100.4 + 30,153.8 = 59,254.3$

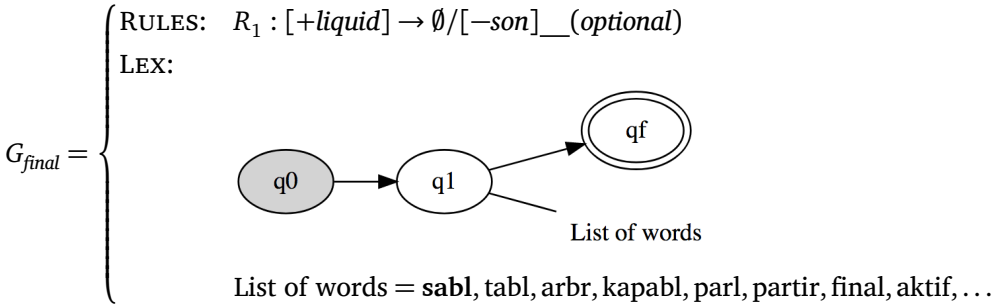
Figure 10: Final grammar for the French optionality simulation. The grammar includes the restricted *L*-deletion rule and forms like /tabl/ without their *L*-deleted counterparts (like /tab/). Here and below all scores are rounded to the first decimal place

The parameters for the present simulation were different from those for the other simulations reported in this paper (and mentioned above). In the present simulation, the encoding length of the data given the grammar was multiplied by 50, and the encoding length of the HMM was multiplied by 20. The simulation also allowed only one state in the HMM, between 0 and 2 phonological rules, and up to one feature vector in the left context and in the right context of each rule. We tried running the simulation also with the usual parameters, but the search did not converge. At present, we are not sure whether this is because the search was difficult in this case or because of something more significant.

The learner induced the correct optional rule and converged on the target lexicon (Figure 10). Compared to the final (correct) grammar, the over-generating hypothesis has a shorter grammar but a longer  $D:G$ , leading to an overall longer description:

- (13) a. Correct Hypothesis:
- $R_1 : [+liquid] \rightarrow \emptyset / [-son] \_ \_ \text{ (optional)}$
  - Description length:  
 $|G| + |D:G| = 29,100.4 + 30,153.8 = 59,254.3$
- b. Over-generating Hypothesis:
- $R_1 : [+liquid] \rightarrow \emptyset / \_ \_ \text{ (optional)}$
  - Description length:  
 $|G| + |D:G| = 29,092.9 + 32,853.8 = 61,946.7$

In Section 3.1 we discussed the undergeneralization problem for restrictiveness-only learning principles like the subset principle. We mentioned a scenario in which a learner has heard a surface form such as [sabl] but, accidentally, has not yet heard its *L*-elided variant [sab]. We noted that, while we would expect the human learner to generalize and learn *L*-deletion in the face of a single accidental gap, the subset principle predicts that *L*-deletion would be avoided. The MDL principle, on the other hand, predicts generalization. We ran another simulation of French using a variant of the corpus in (12) in which [sabl] was added without its *L*-elided variant [sab]. As expected, the learner generalized correctly and converged on the hypothesis in Figure 11 which includes the *L*-deletion rule and a variant of the lexicon that also contains /sabl/.



Description length:  $|G_{final}| + |D : G_{final}| = 29,517.5 + 30,610.1 = 60,127.6$

Figure 11: Final grammar for a variant of the French-optionalty simulation with an occurrence of [sabl] in the data but no occurrences of [sab]. The grammar includes the *L*-deletion rule which can generate the unattested [sab] as an output of /sabl/

*Joint learning of morphology and phonology*

4.2

Our next simulation demonstrates the learner’s ability to perform joint learning of morphology and a single phonological rule. Other works in the literature that perform joint learning of this kind include Naradowsky and Goldwater (2009) and (in a framework of constraint-based phonology) Rasin and Katzir (2016). After establishing this baseline,

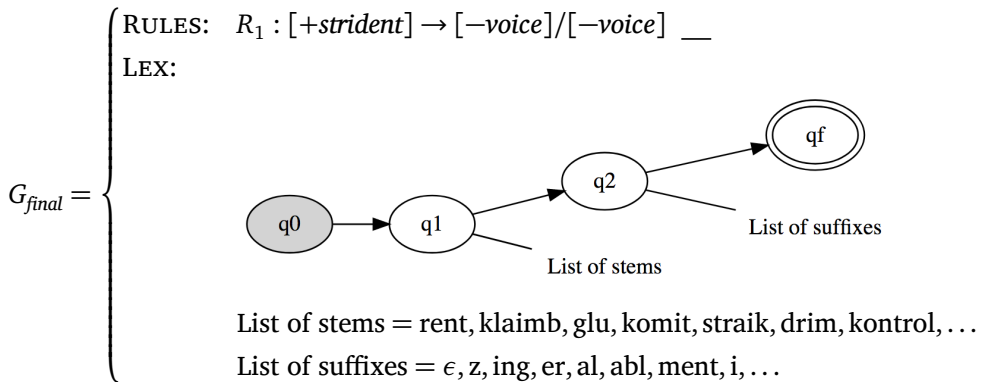
we will proceed, in the following sections, to the joint learning of morphology and rule interaction, a task that, as discussed in Section 5, has not been accomplished in previous work. In the present simulation, the learner’s tasks are to decompose the unanalyzed surface forms into a lexicon of underlying morphemes and to learn the relevant phonological rule.

Our example is modeled after English voicing assimilation where, as discussed in Section 1, the suffix /z/ becomes voiceless following a voiceless consonant. The learner was presented with 250 words generated by creating all combinations of 25 verbal stems with 10 suffixes (including the null suffix) and applying voicing assimilation.<sup>16</sup> A sample of the data is provided in (14).

stem\suffix	∅	-z	-ing	-er	...
rent	rent	rents	renting	renter	
(14) kontrol	kontrol	kontrolz	kontrolling	kontroler	
glu	glu	gluz	gluing	gluer	
...					

The simulation converged on the grammar in Figure 12, which contains the correct rule and segmented lexicon. Given this grammar, generating a surface form requires first choosing a stem (out of 25 stems, at a cost of  $\lg 25$  bits), then choosing a suffix (out of 10 suffixes, at a cost of  $\lg 10$  bits), which makes a total of  $\lg 25 + \lg 10 \approx 7.96$  bits for encoding each surface form. For comparison, consider the minimally-different alternative hypothesis in (15) that fails to learn the voicing-assimilation rule and stores both -z and -s as suffixes without collapsing them into a single UR. The hypothesis in (15) has a slightly smaller  $|G|$ : it stores an additional suffix in the lexicon (-s) but saves some space by omitting the rule. On the other hand, (15) over-generates. Any stem can be suffixed by either -z or -s regardless of the voicing of its final consonant. Thus, for example, both [rents] and [rentz] can be generated from the stem /rent/. This over-generation

<sup>16</sup>When attached to verbs, as in our simulation, the suffix /z/ marks the 3rd person singular in present tense. Since at present we do not model part-of-speech categories, our presentation of voicing assimilation will not distinguish this suffix from the nominal plural marker /z/.



Description length:  $|G_{final}| + |D : G_{final}| = 837.1 + 19,914.5 = 20,751.6$

Figure 12: Final grammar for the joint learning simulation. The grammar includes the voicing assimilation rule and a segmented lexicon with the UR /-z/ from which both surface [-z] and [-s] can be derived

translates into a larger  $|D : G|$ : with the additional suffix, encoding any surface form given (15) now requires choosing a suffix out of 11 suffixes, so the total cost per surface form is  $\lg 25 + \lg 11 \approx 8.1$  bits. Compared to the target hypothesis in Figure 12, the added cost of encoding each surface form given (15) is small ( $\approx 0.14$  bits), but it accumulates over the entire corpus and ends up outweighing the slight advantage that (15) has in terms of  $|G|$ . Overall, then, the target hypothesis in Figure 12 wins due to a smaller combined  $|G| + |D : G|$ .

(15) Over-generating Hypothesis:

- Rules:  $\emptyset$
- List of suffixes = z, s, ...
- Description length:  $|G| + |D : G| = 804.4 + 20,258.2 = 21,062.6$

In the simplified setting we have considered here, the corpus includes all combinations of 25 stems and 10 suffixes (a total of 250 words). This means, for example, that a hypothesis that simply memorizes the data (without performing any segmentation or learning any rules) would be as successful as the target hypothesis in terms of tightness of fit to the data, as both hypotheses generate precisely the same

set of forms. In terms of  $|D : G|$ , encoding each surface form given the memorizing hypothesis would require choosing one out of 250 words in the lexicon at a cost of  $\lg 250$  bits. Since  $\lg 250 = \lg 25 + \lg 10$ , this cost is identical to the cost given the target hypothesis. Despite the tie in the value for  $|D : G|$ , the target hypothesis wins due to its strictly smaller  $|G|$ . In a more realistic setting, the corpus will typically contain gaps, which would give the memorizing hypothesis an advantage in terms of  $|D : G|$ . For example, if five stem + suffix combinations (e.g., [kontrol-er]) are missing from the corpus, encoding a surface form given the memorizing hypothesis would cost  $\lg 245$  bits, compared to an unchanged cost of  $\lg 250$  for the target hypothesis (which can generate the five unattested combinations). As the data  $D$  grows, this wastefulness of the target hypothesis in terms of  $|D : G|$  would accumulate and at some point outweigh the savings in the lexicon obtained by segmenting  $D$ . To estimate the effect of an increase in  $D$ , we created a variant of the data in (14) by omitting five words chosen at random, and we calculated different values for  $|G| + |D : G|$  while varying the multiplier for  $|D : G|$ . We found that when the multiplier for  $|D : G|$  exceeds 1,039, the target hypothesis loses to the memorizing hypothesis in terms of the combined  $|G| + |D : G|$ . We re-ran the simulation several times with the gapped corpus using each of the following multipliers for  $|D : G|$ : 10, 100, 1,000, 10,000, and 100,000. The simulation converged on the target hypothesis in Figure 12 in all cases. At least for the cases of the multipliers 10,000 and 100,000, this means that the simulation converged on a sub-optimal hypothesis. Since this is an accident of the search procedure, whose modeling is not our focus in this paper (as mentioned in Section 3.3), we leave attempts to optimize the results with larger multipliers to a separate occasion.

## 4.3

*Rule ordering*

Rule-based phonology accounts for the interaction of phonological processes through rule ordering. In English, as we have seen, voicing assimilation devoices the suffix /-z/ when preceded by a voiceless obstruent. Epenthesis inserts the vowel [ɪ] between two sibilants (as in [glæsɪz], ‘glasses’). To derive forms such as [glæsɪz], where voicing

assimilation does not apply and the suffix remains voiced, epenthesis is ordered before assimilation. When epenthesis applies to the UR /glæs-z/, it *bleeds* assimilation by disrupting the adjacency between the suffix and the preceding consonant, rendering assimilation inapplicable. The opposite ordering would have derived the incorrect form \*[glæsis], as demonstrated in (16):

- (16) a. Good: epenthesis before assimilation

	/glæs-z/
Epenthesis	glæsɪz
Assimilation	–
	[glæsɪz]

- b. Bad: assimilation before epenthesis

	/glæs-z/
Assimilation	glæss
Epenthesis	glæsis
	*[glæsis]

Our next dataset was generated by an artificial grammar modeled after the interaction of voicing assimilation and epenthesis in English. The learner was presented with 250 words generated by creating the same combinations of stems and suffixes as in the previous section and applying epenthesis (17a) and voicing assimilation (17b), in this order. A sample of the data is provided in (18). The learner converged on the expected lexicon and on the two rules – epenthesis ( $R_1$ ) and assimilation ( $R_2$ ) – and their correct ordering (Figure 13).

- (17) Rules

- a. Rule 1: [i]-epenthesis between stridents
- b. Rule 2: Progressive assimilation with [–voice] spreading to an adjacent segment

stem\suffix	∅	-z	-ing	-er	...
rent	rent	rents	renting	renter	
klaimb	klaimb	klaimbz	klaimbing	klaimber	
kros	kros	krosiz	krosing	kroser	
...					

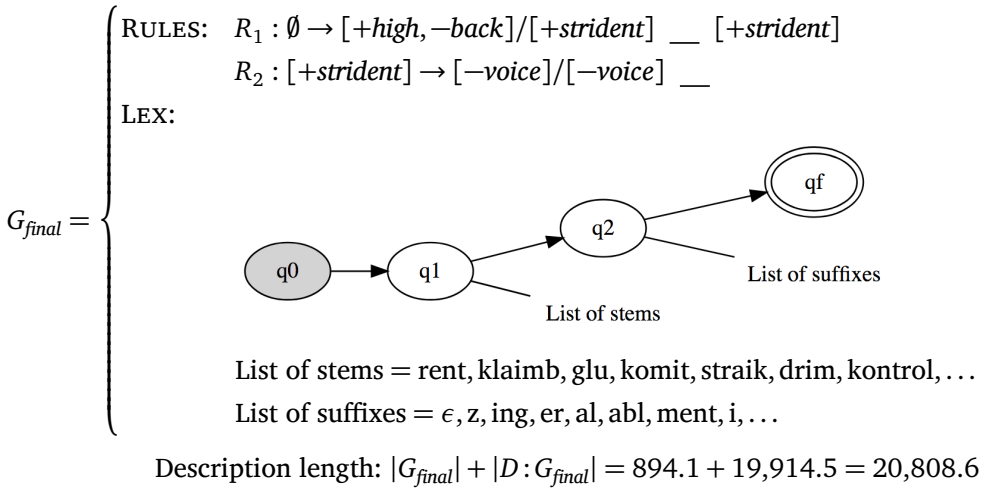


Figure 13: Final grammar for the rule-ordering simulation. The grammar includes epenthesis and voicing assimilation, in this order, and a segmented lexicon

#### 4.4

#### Counterbleeding opacity

The term *opacity* is used to describe rules whose effect is obscured on the surface, often because of an interaction with another rule (Kiparsky 1971, Baković 2011). One type of opacity called *counterbleeding* in the literature results when a rule  $R_2$  removes the conditions for the application of another rule  $R_1$  which has applied earlier in the derivation.  $R_1$  is opaque since its environment of application is missing on the surface.

Our next dataset was designed to test the learner on the problem of counterbleeding opacity. We used two rules modeled after English epenthesis and voicing assimilation and changed the order such that assimilation was ordered first:

- (19) Rules
- a. Rule 1: Progressive assimilation with  $[-voice]$  spreading to an adjacent segment
  - b. Rule 2:  $[i]$ -epenthesis between stridents

The result is that feature spreading takes place even between segments that are separated by an epenthetic vowel on the surface.



Examples of natural languages that reportedly show a similar interaction between feature spreading and epenthesis are some varieties of English and Armenian, as reported in Vaux (2016), and Iraqi Arabic, as reported in Kiparsky (2000, citing Erwin, 1963).

As shown in (20), the opposite rule ordering would lead to the wrong result. Given the correct order, epenthesis applies after assimilation, rendering assimilation opaque: the first consonant of the suffix undergoes assimilation but is preceded by the epenthetic vowel on the surface.

(20) Voicing assimilation crucially precedes epenthesis

a. Good: assimilation before epenthesis

	/glæs-z/
Assimilation	glæss
Epenthesis	glæsis
	[glæsis]

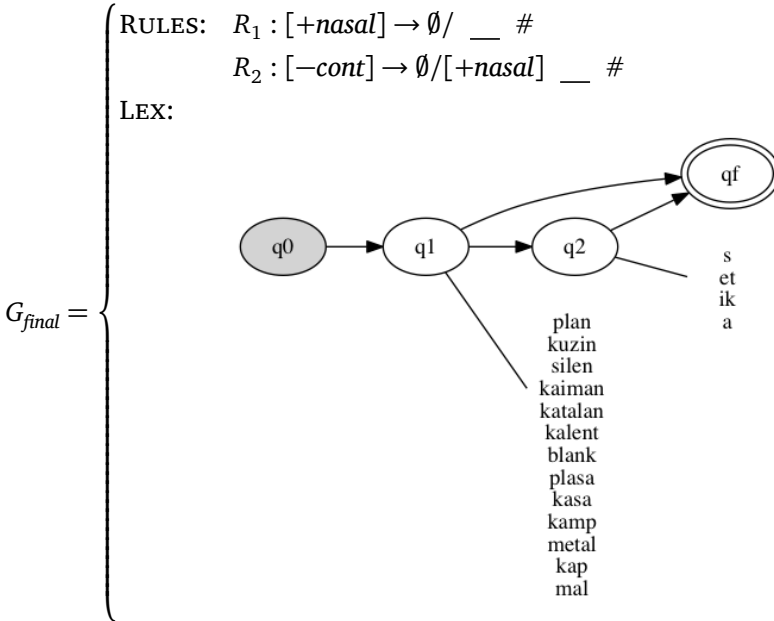
b. Bad: epenthesis before assimilation

	/glæs-z/
Epenthesis	glæsɪz
Assimilation	-
	*[glæsɪz]

For this simulation, the dataset was generated by taking the same combinations of 25 stems and 10 suffixes as before and applying voicing assimilation and epenthesis, in this order. A sample of the data is provided in (21). The learner converged on the expected lexicon and on the two rules – assimilation ( $R_1$ ) and epenthesis ( $R_2$ ) – and their correct ordering (Figure 14).

	stem\suffix	∅	-z	-ing	-er	...
	rent	rent	rents	renting	renter	
(21)	kontrol	kontrol	kontrolz	kontroling	kontroler	
	kros	kros	krosis	krosing	kroser	
	...					





Description length:  $|G_{final}| + |D : G_{final}| = 1093.9 + 14,563.1 = 15,657.1$

Figure 15: Final grammar for the counterfeeding opacity simulation. The grammar includes final-nasal deletion and cluster simplification (in this order) and a segmented lexicon

(23) Rules

- a. Rule 1: Delete a nasal word-finally
- b. Rule 2: Delete a word-final stop following a nasal

stem\suffix	$\emptyset$	-s	-et	...
(24) kalent	kalen	kalents	kalentet	
kuzin	kuzi	kuzins	kuzinet	
...				

The learner converged on a segmented lexicon and on the two rules – final-nasal deletion ( $R_1$ ) and cluster simplification ( $R_2$ ) – and their correct ordering, as in Figure 15. There was one difference between the final result and the grammar used to generate the corpus. The rule of cluster simplification induced by the learner deletes stops

in a broader environment: after any non-continuant consonant rather than only after nasals. Since all word-final consonant-stop clusters in our corpus were nasal-stop clusters, multiple left contexts for cluster simplification were consistent with the data, including left contexts that specify nasal consonants ([+nasal]), any non-continuants ([−cont]), or any consonants ([+cons]). The statements of these three left contexts are equally simple under our current representations, so the learner is expected to choose between them arbitrarily given this corpus.

5

## EXPLANATORY ADEQUACY AND PREVIOUS WORK ON LEARNING IN PHONOLOGY

We presented a learner that uses the MDL evaluation metric, which minimizes  $|G| + |D : G|$ , to jointly learn morphology and phonology within a rule-based framework. This learner is fully distributional, working from unanalyzed surface forms alone – without access to paradigms or negative evidence – to obtain the URs in the lexicon, the possible morphological combinations, and the ordered phonological rules. It acquires both allophonic rules and alternations and handles both optionality and rule interaction, including instances of opacity. By accomplishing all of these tasks, the learner goes beyond previous work in terms of its ability to address the challenge of explanatory adequacy discussed in Section 2: arriving at a descriptively-adequate grammar based on primary linguistic data.

In this section, we review prominent proposals from past work on learning in phonology and show that they have not gone as far in terms of achieving explanatory adequacy. This is because previous learners either do not work with what we take to be the primary linguistic data (e.g., by assuming that the child is given direct information about URs) or because they do not arrive at a full phonological grammar (e.g., by not acquiring opacity). To make the comparison easier, we will focus on five components of the learning challenge: learning from distributional evidence alone, learning segmentation simultaneously with phonology, learning opacity, learning optionality, and learning

Table 6: Some prominent proposals from past work on learning in phonology and their ability to address five learning challenges

Theory ↓	Distributional evidence	Simultaneous segmentation	Opacity	Optionality	Abstract URs
1) Constraint reranking	✗	✗	?	✓	✗
2) Reranking + Free Ride	✗	✗	?	✓	✗
3) MaxEnt + OT	✗	✗	✓	?	?
4) Dist. alt. learner	✓	✗	✗	✗	✗
5) MaxLikelihood + OT		* (see discussion below)			
6) Lexicon Entropy		* (see discussion below)			

abstract URs. Each of the learners we discuss fails on at least one of those components, as summarized in Table 6 (and as discussed in the rest of this section).

We first consider constraint reranking algorithms (row 1 in Table 6), a family of learning algorithms for OT that include the proposals by Tesar (1995, 2014), Tesar and Smolensky (1998), Boersma and Hayes (2001), Prince and Tesar (2004), and much related work. These proposals assume that URs are given to the learner in advance or that the learner is exposed to surface forms already segmented into morphemes, along with the information of which surface morphemes come from the same UR. Therefore, these works do not address the challenge of learning from distributional evidence and the challenge of learning segmentation simultaneously with the phonology.

Another shortcoming of the constraint-reranking proposals just mentioned is that they assume that, in the absence of direct evidence from alternations, URs are identical to their corresponding surface forms. Hence, they do not address the challenge of learning abstract URs. An attempt to address this problem was made by McCarthy (2005), who proposed to extend constraint reranking algorithms with the Free Ride Principle, a learning principle that aims to deal with some cases of abstract URs (row 2 in Table 6). This principle allows using information from alternations to infer non-identical URs for non-alternating forms. While addressing some cases of abstract-UR learning, McCarthy’s algorithm does not offer constraint reranking algorithms a handle on cases of abstract URs where there is no sup-

porting evidence from alternations at all, as in Alderete and Tesar's (2002) stress-epenthesis example. See Rasin and Katzir 2018 for further discussion.

Another family of learners in the OT literature are the so-called MaxEnt learners (Goldwater and Johnson 2004, Nazarov and Pater 2017, and O'Hara 2017, among others), which rely on the principle of Maximum Entropy as an evaluation metric (row 3 in Table 6). These learners receive morphologically-segmented surface forms, as well as information about which surface morphemes come from the same UR. Hence, like constraint reranking algorithms, they do not address the challenges of learning from distributional evidence alone and learning segmentation simultaneously with the phonology.

Similarly to the present proposal, the distributional alternation learner of Calamaro and Jarosz (2015) learns phonological rules – both allophony and alternations – in a fully distributional way (row 4 in Table 6). Since their learner is closer to our goals than the previous learners are, we discuss it here in more detail. The proposal extends the allophonic learner of Peperkamp *et al.* (2006). Peperkamp *et al.* detect maximally dissimilar contexts as hints for allophonic distribution. For example, [æ] and [ã] are allophones in English, and the contexts that they can appear in are very different: [ã] can only appear before a nasal consonant, while [æ] can only appear elsewhere. Peperkamp *et al.* provide a statistical score that identifies such dissimilarities in the contexts in which two segments can appear; when two segments have highly dissimilar contexts, they are considered to be potential allophones.<sup>17</sup> Calamaro and Jarosz (2015) look to extend Peperkamp *et al.*'s (2006) model beyond allophony, in order to account for neutralization processes. The challenge, given Peperkamp *et al.*'s dissim-

---

<sup>17</sup>This raises well-known issues with phonemics, such as the fact that, in English, [h] and [ŋ] are in complementary distribution but are not phonemically related. And indeed, Peperkamp *et al.* encounter many false positives (a problem that is exacerbated by the fact that their model does not require full complementary distribution). Echoing early structuralist proposals, they propose that complementarity should be combined with requirements of phonological similarity. As discussed by Chomsky (1964, p. 85), such requirements do not resolve the problem for phonemic analysis.

ilarity score, is that neutralization involves segments whose possible contexts may have a significant overlap. Consider, for example, a language like Dutch that has final devoicing. In such a language, [t] and [d] might contrast everywhere except for the context \_\_#; a global score of contextual dissimilarity will consequently treat [t] and [d] as quite similar and fail to relate them to one another. In order to overcome this challenge, Calamaro and Jarosz consider contextualized distributional dissimilarity: for a given context  $X\_Y$  and two potential alternants  $A$  and  $B$ , they compute a dissimilarity score for the triple  $\langle X\_Y, A, B \rangle$  by comparing the probability of the context  $X\_Y$  given  $A$  and given  $B$ . These dissimilarity scores are summed for the context and for the featural change over all pairs  $A$  and  $B$  that have that change, thus allowing for generalization in terms of the change. A further extension introduces generalization over contexts (subject to two special conditions). In terms of comparison with the present proposal, Calamaro and Jarosz's model faces two challenges that, as far as we can tell, are hard to address within the framework of distribution comparison that they adopt. First, their model does not handle rule orderings. This gap is particularly difficult to bridge in the case of opaque rule interactions, where surface distributions obscure the correct context for rule application. The second challenge to Calamaro and Jarosz's model concerns optionality. When a rule is optional, the distribution of  $A$  and  $B$  can be similar in all contexts, so a dissimilarity detector will fail to identify the rule.

Other learners close to our goals include Jarosz's (2006, 2009) Maximum Likelihood OT learner and Riggle's (2006) Lexicon Entropy OT learner (rows 5 and 6 in Table 6). Both learners rely on evaluation metrics rather than on a procedural approach to acquire an OT ranking and URs. Differently from MDL, however, these evaluation metrics do not balance economy and restrictiveness and thus lead to overgeneralization and undergeneralization problems of the kinds discussed earlier in Section 3. These problems for Maximum Likelihood and Lexicon Entropy have been discussed in detail in Rasin and Katzir 2016.

Of the other learners proposed in the literature, our learner is closest to those proposed by Goldwater and Johnson (2004), Goldsmith (2006), Naradowsky and Goldwater (2009), and Rasin and Katzir (2016), all of which are fully distributional phonological learners that

rely on the same kind of balanced evaluation metric as the present paper. The first three learn rule-based morpho-phonology, while the fourth learns constraint-based phonology.<sup>18</sup> Goldwater and Johnson's (2004) algorithm starts with a morphological analysis based on Goldsmith's (2001) MDL-based learner and then searches for phonological rules that lead to an improved grammar, where the improvement criterion is Bayesian. Goldsmith's (2006) learner follows a similar path but uses MDL also for the task of phonological learning. Naradowsky and Goldwater's (2009) learner is a variant of Goldwater and Johnson's (2004) learner with joint learning of morphology and phonology, thus addressing (similarly to the present learner) the interdependency of phonology and morphology. As originally presented, all three learners can acquire rules only at morpheme boundaries and generalize only with respect to  $X\_Y$  and not with respect to  $A$  and  $B$ .<sup>19</sup> They are also aimed at obligatory rules and do not handle rule interaction. Rasin and Katzir (2016) propose an MDL-based learner for Optimality Theory that can learn the URs, constraint ranking, and also the constraints themselves, from distributional evidence alone. That learner has not yet been shown to acquire opacity. One way of interpreting our simulations above is as showing that the limitations of all these balanced distributional learners are not essential within this framework and that MDL can support the acquisition of allophony, generalizations over both the context and the change (in the case of rule-based phonology), optionality, and opacity.

## 6

## DISCUSSION

We argued that the MDL metric can adequately guide the child in choosing between competing hypotheses while learning phonology.

---

<sup>18</sup>Naradowsky and Goldwater (2009) target orthographic rules rather than phonology, but the difference is immaterial. Other balanced learners proposed in the literature, which are not fully distributional, include Cotterell *et al.* (2015) and Ellis and O'Donnell (2017).

<sup>19</sup>By limiting the kinds of rule that can be learned, these learners are similar to the procedural rule-based learners of Johnson (1984), Albright and Hayes (2002, 2003), and Simpson (2010).



We illustrated this with an implemented MDL-based learner for the unsupervised learning of rule-based morpho-phonological grammars. The generality of the MDL metric has allowed the learner to simultaneously perform morphological segmentation and acquire complete grammars, including URs and ordered rules, and including transparent and opaque rule interactions, as well as optional rules. By doing that, the learner is the first learner we know of that acquires opacity and optionality – basic textbook patterns that any theory of learning will have to address – from distributional evidence alone.<sup>20</sup> More generally, the learner goes beyond the phonological learning literature – including both rule-based and constraint-based learners – in its ability to address the challenge of explanatory adequacy. Previous proposals have not gone as far because they either rely on richer input data than children require or do not return a full, descriptively-adequate grammar. In particular, by learning from distributional evidence alone, the learner differs from many proposals in the literature on phonological learning which assume that the learner is given systematic paradigmatic information, information about URs, or even the URs themselves. The ability of our learner to acquire opaque rule interactions and optional rules distinguishes it from other learners that are limited to transparent process interactions or deterministic processes.

While the present work goes beyond the literature in terms of the challenge of explanatory adequacy in phonology, the simulation results we presented use corpora that are smaller than corpora used by some previous learners. In this respect the present work is in line with Chomsky's view (Chomsky 1965, p. 26), which prioritizes the comparison of learning theories based on their success on explanatory adequacy rather than on their ability to apply to large datasets:

*“Clearly, it would be utopian to expect to achieve explanatory adequacy on a large scale in the present state of linguistics. Never-*

---

<sup>20</sup> To be clear, the ability of the learner to acquire opacity does not necessarily rely on its use of a rule-based formalism. For example, as noted by Baković (2011), rule-based phonology does not necessarily offer a uniform improvement over Optimality Theory in terms of its account of known opaque patterns. Since the MDL metric is general, it could in principle support the acquisition of opaque patterns using a variety of formalisms, as long as these formalisms are capable of representing these patterns.

*theless, considerations of explanatory adequacy are often critical for advancing linguistic theory. Gross coverage of a large mass of data can often be attained by conflicting theories; for precisely this reason it is not, in itself, an achievement of any particular theoretical interest or importance.”*

Still, an investigation of how well the MDL metric can extend to larger, more realistic corpora remains an important task that the present work has not addressed. A central part of this task is a study of the optimization procedure to see where it adequately navigates the highly complex search space and where it fails. The present work, with its focus on the MDL metric rather than the search barely starts to probe the behavior of the optimization procedure. We have to leave the examination of this question to future work.

As mentioned in Section 3.1, the simple and very general MDL metric compares hypotheses in terms of two readily available quantities: the storage space required for the current grammar and the storage space required for the current grammar’s best parse of the grammar. It has been argued recently that this approach has cognitive plausibility as a null hypothesis for language learning in humans and that it offers a reasonable framework for the comparison of different representational choices in terms of predictions about learning (see Katzir 2014, Katzir *et al.* 2020, and Rasin and Katzir 2020). From an empirical perspective, Pycha *et al.* (2003) have provided evidence that simplicity plays a central role in the acquisition of phonological rules.<sup>21</sup> If correct, the present work is a step toward a cognitively plausible learner for rule-based morpho-phonology, and its predictions can be compared with those of MDL or Bayesian learners for other representation choices such as Rasin and Katzir’s (2016) MDL learner for constraint-based phonology. We leave the investigation of such predictions for future work.

---

<sup>21</sup> See also Moreton and Pater (2012a,b) for simplicity in phonological learning (though see Moreton *et al.* 2017 for an argument that phonotactic and concept learning are guided by something closer to a Maximum Entropy model rather than by simplicity), and see Goodman *et al.* (2008) and Orbán *et al.* (2008), among others, for empirical evidence for balanced learning elsewhere in cognition.

## REFERENCES

- Adam ALBRIGHT and Bruce HAYES (2002), Modeling English past tense intuitions with minimal generalization, in *Proceedings of the ACL-02 workshop on Morphological and phonological learning-Volume 6*, pp. 58–69, Association for Computational Linguistics.
- Adam ALBRIGHT and Bruce HAYES (2003), Rules vs. analogy in English past tenses: a computational/experimental study, *Cognition*, 90(2):119–161, doi:[http://dx.doi.org/10.1016/S0010-0277\(03\)00146-X](http://dx.doi.org/10.1016/S0010-0277(03)00146-X).
- John ALDERETE and Bruce TESAR (2002), Learning covert phonological interaction: an analysis of the problem posed by the interaction of stress and epenthesis, Technical Report RuCCS-TR-72, Rutgers Center for Cognitive Science, Piscataway, NJ.
- Carl L. BAKER (1979), Syntactic theory and the projection problem, *Linguistic Inquiry*, 10(4):533–581.
- Eric BAKOVIĆ (2011), Opacity and ordering, in *The Handbook of Phonological Theory, Second Edition*, pp. 40–67, Wiley-Blackwell.
- Robert C. BERWICK (1982), *Locality principles and the acquisition of syntactic knowledge*, Ph.D. thesis, MIT, Cambridge, MA.
- Robert C. BERWICK (1985), *The acquisition of syntactic knowledge*, MIT Press, Cambridge, Massachusetts.
- Paul BOERSMA and Bruce HAYES (2001), Empirical Tests of the Gradual Learning Algorithm, *Linguistic Inquiry*, 32:45–86.
- Martin D. S. BRAINE (1971), On Two Types of Models of the Internalization of Grammars, in D. J. SLOBIN, editor, *The Ontogenesis of Grammar*, pp. 153–186, Academic Press.
- Michael BRENT (1999), An efficient, probabilistically sound algorithm for segmentation and word discovery, *Computational Linguistics*, 34(1–3):71–105.
- Shira CALAMARO and Gaja JAROSZ (2015), Learning General Phonological Rules From Distributional Information: A Computational Model, *Cognitive Science*, 39(3):647–666, doi:10.1111/cogs.12167.
- Gregory J. CHAITIN (1966), On the Length of Programs for Computing Finite Binary Sequences, *Journal of the ACM*, 13:547–569.
- Noam CHOMSKY (1964), *Current issues in linguistic theory*, Mouton & Company.
- Noam CHOMSKY (1965), *Aspects of the theory of syntax*, MIT Press, Cambridge, MA.
- Noam CHOMSKY and Morris HALLE (1968), *The Sound Pattern of English*, Harper and Row Publishers, New York.

- Alexander CLARK (2001), *Unsupervised Language Acquisition: Theory and Practice*, Ph.D. thesis, University of Sussex.
- Ryan COTTERELL, Nanyun PENG, and Jason EISNER (2015), Modeling Word Forms Using Latent Underlying Morphs and Phonology, *Transactions of the Association for Computational Linguistics*, 3:433–447.
- Carl DE MARCKEN (1996), *Unsupervised Language Acquisition*, Ph.D. thesis, MIT, Cambridge, MA.
- François DELL (1981), On the learnability of optional phonological rules, *Linguistic Inquiry*, 12(1):31–37.
- Kevin ELLIS and Timothy O'DONNELL (2017), Inducing phonological rules: Perspectives from Bayesian program learning, Presented at the MIT Workshop on Simplicity in Grammar Learning.
- Timothy Mark ELLISON (1994), *The machine learning of phonological structure*, Ph.D. thesis, University of Western Australia.
- Ansgar D. ENDRESS and Marc D. HAUSER (2011), The influence of type and token frequency on the acquisition of affixation patterns: Implications for language processing., *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 37(1):77–95.
- Wallace M. ERWIN (1963), *A short reference grammar of Iraqi Arabic*, Georgetown University Press.
- Louann GERKEN, Rachel WILSON, and William LEWIS (2005), Infants can use distributional cues to form syntactic categories, *Journal of Child Language*, 32(2):249–268.
- John GOLDSMITH (2001), Unsupervised Learning of the Morphology of a Natural Language, *Computational Linguistics*, 27(2):153–198.
- John GOLDSMITH (2006), An Algorithm for the Unsupervised Learning of Morphology, *Natural Language Engineering*, 12(3):1–19.
- Sharon. GOLDWATER, Thomas L. GRIFFITHS, and Mark JOHNSON (2006), Interpolating between types and tokens by estimating power-law generators, *Advances in neural information processing systems*, 18:459.
- Sharon GOLDWATER and Mark JOHNSON (2004), Priors in Bayesian Learning of Phonological Rules, in *7th Annual Meeting of the ACL Special Interest Group on Computational Phonology*, pp. 35–42.
- N.D. GOODMAN, J.B. TENENBAUM, J. FELDMAN, and T.L. GRIFFITHS (2008), A Rational Analysis of Rule-Based Concept Learning, *Cognitive Science*, 32(1):108–154.
- Peter GRÜNWARD (1996), A Minimum Description Length Approach to Grammar Inference, in Stefan WERMTER, Ellen RILOFF, and Gabriele SCHELER, editors, *Connectionist, Statistical and Symbolic Approaches to Learning for Natural*

*Language Processing*, Springer Lecture Notes in Artificial Intelligence, pp. 203–216, Springer.

Mark HALE and Charles REISS (2003), The Subset Principle in phonology: why the tabula can't be rasa, *Journal of Linguistics*, 39:219–244.

Mark HALE and Charles REISS (2008), *The phonological enterprise*, Oxford University Press.

John H. HOLLAND (1975), *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.*, University of Michigan Press.

James HORNING (1969), *A Study of Grammatical Inference*, Ph.D. thesis, Stanford University.

Gaja JAROSZ (2006), *Rich Lexicons and Restrictive Grammars – Maximum Likelihood Learning in Optimality Theory*, Ph.D. thesis, Johns Hopkins University, Baltimore, Maryland.

Gaja JAROSZ (2009), Restrictiveness in Phonological Grammar and Lexicon Learning, in *Proceedings from the Annual Meeting of the Chicago Linguistic Society*, volume 43, pp. 125–139, Chicago Linguistic Society.

Mark JOHNSON (1984), A Discovery Procedure for Certain Phonological Rules, in *Proceedings of 10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*, pp. 344–347.

Ronald M. KAPLAN and Martin KAY (1994), Regular Models of Phonological Rule Systems, *Computational Linguistics*, 20(3):331–378.

Roni KATZIR (2014), A Cognitively Plausible Model for Grammar Induction, *Journal of Language Modelling*, 2(2):213–248.

Roni KATZIR, Nur LAN, and Noa PELED (2020), A note on the representation and learning of quantificational determiners, in Michael FRANKE *et al.*, editors, *Proceedings of Sinn und Bedeutung 24*, volume 1, pp. 392–410.

Paul KIPARSKY (1971), Historical linguistics, in W. O. DINGWALL, editor, *A Survey of Linguistic Science*, pp. 576–642, University of Maryland Linguistics Program, College Park.

Paul KIPARSKY (2000), Opacity and Cyclicity, *The Linguistic Review*, 17(2–4):351–366, doi:10.1515/tlir.2000.17.2-4.351.

Andrei Nikolaevic KOLMOGOROV (1965), Three Approaches to the Quantitative Definition of Information, *Problems of Information Transmission (Problemy Peredachi Informatsii)*, 1:1–7, republished as Kolmogorov (1968).

Andrei Nikolaevic KOLMOGOROV (1968), Three Approaches to the Quantitative Definition of Information, *International Journal of Computer Mathematics*, 2:157–168.

- Martin KRÄMER (2012), *Underlying Representations*, Cambridge University Press, Cambridge, UK.
- Nur LAN (2018), *Learning morpho-phonology using the Minimum Description Length Principle and a Genetic Algorithm*, Master's thesis, Tel Aviv University.
- Joan MASCARÓ (1976), *Catalan Phonology and the Phonological Cycle*, Ph.D. thesis, MIT.
- John J. MCCARTHY (2005), Taking a free ride in morphophonemic learning, *Catalan Journal of Linguistics*, 4:19–56.
- Elliott MORETON and Joe PATER (2012a), Structure and Substance in Artificial-phonology Learning, Part I: Structure, *Language and Linguistics Compass*, 6(11):686–701.
- Elliott MORETON and Joe PATER (2012b), Structure and Substance in Artificial-Phonology Learning, Part II: Substance, *Language and Linguistics Compass*, 6(11):702–718.
- Elliott MORETON, Joe PATER, and Katya PERTSOVA (2017), Phonological Concept Learning, *Cognitive Science*, 41(1):4–69.
- Jason NARADOWSKY and Sharon GOLDWATER (2009), Improving Morphology Induction by Learning Spelling Rules, in *IJCAI*, pp. 1531–1536.
- Aleksei NAZAROV and Joe PATER (2017), Learning opacity in Stratal Maximum Entropy Grammar, *Phonology*, 34(2):299–324.
- Andrew NEVINS and Bert VAUX (2007), Underlying representations that do not minimize grammatical violations, in Sylvia BLAHO, Patrik BYE, and Martin KRÄMER, editors, *Freedom of analysis?*, pp. 35–61, Mouton de Gruyter.
- Charlie O'HARA (2017), How abstract is more abstract? Learning abstract underlying representations, *Phonology*, 34(2):325–345.
- Gergő ORBÁN, József FISER, Richard N. ASLIN, and Máté LENGYEL (2008), Bayesian learning of visual chunks by human observers, *Proceedings of the National Academy of Sciences*, 105(7):2745–2750.
- Sharon PEPPERKAMP, Rozenn Le CALVEZ, Jean-Pierre NADAL, and Emmanuel DUPOUX (2006), The acquisition of allophonic rules: Statistical learning with linguistic constraints, *Cognition*, 101(3):B31–B41, doi:<http://dx.doi.org/10.1016/j.cognition.2005.10.006>.
- Alan PRINCE and Paul SMOLENSKY (1993), *Optimality Theory: Constraint Interaction in Generative Grammar*, Technical report, Rutgers University, Center for Cognitive Science.
- Alan PRINCE and Bruce TESAR (2004), Learning phonotactic distributions, in René KAGER, Joe PATER, and Wim ZONNEVELD, editors, *Constraints in phonological acquisition*, pp. 245–291, Cambridge University Press.

Anne PYCHA, Pawel NOWAK, Eurie SHIN, and Ryan SHOSTED (2003), Phonological rule-learning and its implications for a theory of vowel harmony, in *Proceedings of the 22nd West Coast Conference on Formal Linguistics*, pp. 101–114, Cascadilla Press, Somerville, MA.

Ezer RASIN and Roni KATZIR (2016), On evaluation metrics in Optimality Theory, *Linguistic Inquiry*, 47(2):235–282, doi:10.1162/ling\_a\_00210.

Ezer RASIN and Roni KATZIR (2018), Learning abstract underlining representations from distributional evidence, in S. HUCKLEBRIDGE and M. NELSON, editors, *Proceedings of NELS 48*, pp. 283–290, doi:10.1017/S0022226720000146.

Ezer RASIN and Roni KATZIR (2020), A Conditional Learnability Argument for Constraints on Underlying Representations, *Journal of Linguistics*, 56(4):745–773.

Jason RIGGLE (2006), Using entropy to learn OT grammars from surface forms alone, in *Proceedings of the 25th West Coast Conference on Formal Linguistics*, pp. 346–353.

Jorma RISSANEN (1978), Modeling by Shortest Data Description, *Automatica*, 14:465–471.

Jorma RISSANEN and Eric Sven RISTAD (1994), Language Acquisition in the MDL Framework, in *Language computations: DIMACS Workshop on Human Language, March 20-22, 1992*, p. 149, Amer Mathematical Society.

Jenny R. SAFFRAN, Elissa L. NEWPORT, and Richard N. ASLIN (1996), Statistical learning by 8-month old infants, *Science*, 274:1926–1928.

Marc SIMPSON (2010), *From alternations to ordered rules: A system for learning Derivational Phonology*, Master's thesis, Concordia University, Montreal.

Paul SMOLENSKY (1996), The initial state and 'richness of the base' in Optimality Theory, Technical Report JHU-CogSci-96-4, Johns Hopkins University.

Ray J. SOLOMONOFF (1964a), A formal theory of inductive inference, part I, *Information and Control*, 7(1):1–22.

Ray J. SOLOMONOFF (1964b), A formal theory of inductive inference, part II, *Information and Control*, 7(2):224–254.

Andreas STOLCKE (1994), *Bayesian Learning of Probabilistic Language Models*, Ph.D. thesis, University of California at Berkeley, Berkeley, California.

Bruce TESAR (1995), *Computational optimality theory*, Ph.D. thesis, University of Colorado.

Bruce TESAR (2014), *Output-Driven Phonology*, Cambridge University Press.

Bruce TESAR and Paul SMOLENSKY (1998), Learnability in Optimality Theory, *Linguistic Inquiry*, 29(2):229–268.

Bert VAUX (2016), Can epenthesis counterbleed assimilation?, talk presented at NAPhC 9, Concordia University, May 7–8, 2016.

Christopher S. WALLACE and David M. BOULTON (1968), An Information Measure for Classification, *Computer Journal*, 11(2):185–194.

Kenneth WEXLER and Rita M. MANZINI (1987), Parameters and Learnability in Binding Theory, in Thomas ROEPER and Edwin WILLIAMS, editors, *Parameter Setting*, pp. 41–76, D. Reidel Publishing Company, Dordrecht, The Netherlands.

Katherine S. WHITE, Sharon PEPPERKAMP, Cecilia KIRK, and James L. MORGAN (2008), Rapid acquisition of phonological alternations by infants, *Cognition*, 107(1):238–265.

Charles YANG (2016), *The Price of Linguistic Productivity: How Children Learn to Break the Rules of Language*, MIT Press.

*Ezer Rasin*

Ⓔ 0000-0001-8980-5566  
rasin@tauex.tau.ac.il

*Iddo Berger*

Ⓔ 0000-0003-1117-1166  
iddoberger@gmail.com

*Nur Lan*

Ⓔ 0000-0003-0712-4236  
nurlan@mail.tau.ac.il

*Itamar Shefi*

Ⓔ 0000-0001-7534-3006  
itamarshefi@gmail.com

Department of Linguistics  
Tel Aviv University  
Tel Aviv, Israel 6997801

*Roni Katzir*

Ⓔ 0000-0002-0241-1896  
rkatzir@tauex.tau.ac.il

Department of Linguistics  
and Sagol School of Neuroscience  
Tel Aviv University  
Tel Aviv, Israel 6997801

Ezer Rasin, Iddo Berger, Nur Lan, Itamar Shefi, and Roni Katzir (2021), *Approaching explanatory adequacy in phonology using Minimum Description Length*, *Journal of Language Modelling*, 9(1):17–66

Ⓓ <https://dx.doi.org/10.15398/jlm.v9i1.266>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

ⒸⒻ <http://creativecommons.org/licenses/by/4.0/>



# Modelling a subregular bias in phonological learning with Recurrent Neural Networks

*Brandon Prickett*

University of Massachusetts Amherst

## ABSTRACT

A number of experiments have demonstrated what seems to be a bias in human phonological learning for patterns that are simpler according to Formal Language Theory (Finley and Badecker 2008; Lai 2015; Avcu 2018). This paper demonstrates that a sequence-to-sequence neural network (Sutskever *et al.* 2014), which has no such restriction explicitly built into its architecture, can successfully capture this bias. These results suggest that a bias for patterns that are simpler according to Formal Language Theory may not need to be explicitly incorporated into models of phonological learning.

*Keywords:*  
*neural networks,*  
*learning bias,*  
*Formal Language*  
*Theory,*  
*phonology*

## INTRODUCTION

1

*Formal Language Theory* (FLT; Chomsky 1956) describes how complex a pattern is in terms of the computational machinery needed to represent it. The framework was originally designed to demonstrate that natural language syntax was more complex than the set of *Regular* patterns (i.e., those that could be represented using finite state machines). However, Johnson (1972) showed that all known phonological mappings could be considered, at most, *Regular* (see also Kaplan and Kay

1994). Recent work has supported this finding, arguing that phonological learning must be categorically limited to patterns that can be characterized as *Subregular* (i.e., belonging to specific classes of patterns that can be represented with less expressive power than that of a finite state machine; Heinz 2010; Heinz and Idsardi 2011). One piece of evidence for this hypothesis is a series of experimental results that show humans being biased against learning certain patterns that seem to be too complex according to FLT-based metrics (Finley and Badecker 2008; Lai 2015; Finley 2017; Avcu 2018).

For example, Finley and Badecker (2008) showed that their participants were biased against learning *Majority Rule Harmony* (also known as *Majority Rules*; Lombardi 1999; Bakovic 2000), an unattested phonological process that is more complex than the set of Regular mappings. Later experimental work went on to show that people were also biased against learning some Subregular patterns (Lai 2015; Avcu 2018; McMullin and Hansson 2019), providing evidence that the phonological grammar might be limited to even simpler levels of the FLT hierarchy, such as those that can be characterized as *Strictly Local* and *Tier-based Strictly Local* (TSL; Heinz *et al.* 2011).<sup>1</sup> The former level of complexity includes any pattern that bans a finite set of substrings from occurring in a word, while the latter does so over a tier of segments (i.e., certain segments can be ignored by the pattern).

An example of a Strictly Local pattern that commonly occurs in natural language is the restriction banning voiceless sounds after nasals (henceforth \*NÇ; Pater, 1999). This pattern is Strictly Local since it bans any word containing the finite set of strings that result from combining all nasals with all voiceless sounds (e.g. [nt], [np], [mt], [mp], etc.). TSL patterns are also common in phonology and are typically called *harmony* (see Rose and Walker 2011 for an overview), since many of them cause a subset of segments in a word to agree in their value for some feature.<sup>2</sup> For example, Navajo contains a har-

---

<sup>1</sup>*Strictly Piecewise* has also been suggested as an appropriate level of complexity to describe phonological patterns (Heinz 2010); however, see McMullin (2016) and Lamont (2018, 2019a) for arguments against this.

<sup>2</sup>Long-distance dissimilation patterns (i.e., patterns in which sounds must disagree in their value for a feature; Bennett 2015), are rarer in natural language but are also Tier-based Strictly Local.

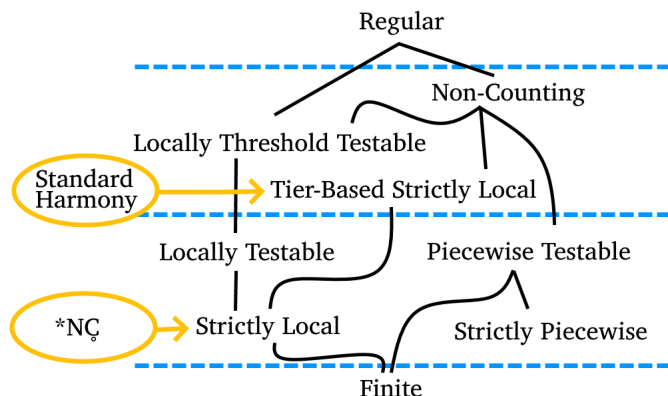


Figure 1: The Subregular Hierarchy (Heinz 2018), with examples of Strictly Local and TSL patterns given. Dashed blue lines indicate different orders of logic. Solid black lines indicate subset relationships

mony pattern in which all sibilants (e.g., [s] and [ʃ]) within a word have to agree in their value for the feature [anterior] (Sapir and Hoijer 1967). This means that on the sibilant tier, the strings [sʃ] and [ʃs] are banned, since [s] is [+anterior] but [ʃ] is [−anterior]. Any sounds that are not sibilants are irrelevant to the pattern. A word like \*[saʃ] would not be allowed, since its sibilant tier would exclude [a] and only include the banned sequence \*[sʃ]. Figure 1 shows the full Subregular Hierarchy and where each of these two types of patterns are located in it.

While a considerable amount of work has been done to explain phonological typology and learning in terms of these FLT-based criteria, little work has been done to computationally model the experimental results that support a bias for Subregular patterns.<sup>3</sup> Here, I will show that the biases observed in past FLT-related experiments can emerge from the learning process of a relatively generic learner, namely a sequence-to-sequence neural network, which has the expressive power to represent both Subregular and Supraregular patterns (Siegelmann 1999). Since the network has no explicit, FLT-related biases built into its architecture, this provides evidence that such a

<sup>3</sup>Note that most of the literature involving FLT and learning (e.g., Chandlee *et al.* 2015; Jardine and Heinz 2016, among others) does not have an explicit hypothesis for how such learning algorithms can be used to make predictions for artificial language learning experiments. Instead, such work tends to focus on whether formally defined classes of languages are learnable at all, given certain kinds of training data.

bias may not need to be added to theories of phonological acquisition.

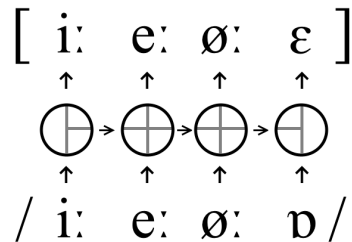
The paper is structured as follows: Section 2 introduces the neural network model that I will be using, Section 3 focuses on simulating experimental results regarding Majority Rule Harmony (Lombardi 1999; Bakovic 1999; Finley and Badecker 2008), Section 4 focuses on doing the same for experiments that involve First-Last Assimilation (Lai 2015; Avcu 2018), and Section 5 concludes.

## 2 MODELLING PHONOLOGICAL LEARNING WITH NEURAL NETWORKS

Neural networks have been used to model linguistic patterns since at least Rumelhart and McClelland (1986) and were quickly applied to the domain of phonology by Touretzky (1989) and Touretzky and Wheeler (1990). Hare (1990) first used *recurrent* neural networks (Jordan 1986; Elman 1990) to capture Hungarian vowel harmony, demonstrating that this architecture could be particularly useful for learning phonological mappings. Recurrent neural networks treat a stimulus as being made up of multiple timesteps, each of which the model processes separately. At each timestep, the model has connections that lead to the output layer and to the next step in time. These connections that feed into future timesteps are called recurrent and give the model a kind of memory as it walks through the full stimulus. This is illustrated in Figure 2 for Hungarian vowel harmony.

Figure 2:

Illustration of a recurrent neural network. Circles represent the hidden recurrent layer processing each timestep, black arrows represent groups of connections, grey lines represent the internal structure of the layer, and IPA symbols represent feature vectors corresponding to each segment

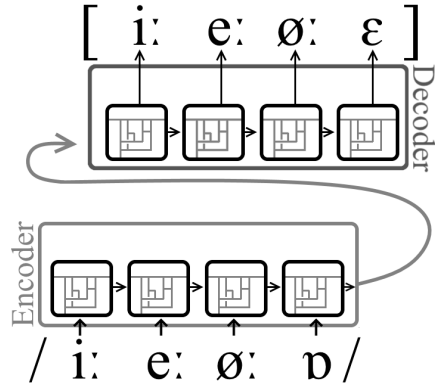


The use of such simple recurrent networks was later expanded to model other phonological phenomena, such as voicing assimilation (Gasser and Lee 1992) and phonotactic learning biases (Doucette 2017). However, these simple networks have been critiqued for their inability to generalise in a human-like way (Gasser 1993; Marcus *et al.* 1999) and for being too myopic (Alderete and Tupper 2018), since they have no ability to look ahead in their input sequence. There are a number of other reasons to suspect that simple recurrent networks would not be able to handle the full wealth of phonological phenomena – for example, their dependency on input and output lengths being equal (Sutskever *et al.* 2014).

Most of these issues are solved by the neural network architecture used in this paper, *sequence-to-sequence networks* (henceforth Seq2Seq; Sutskever *et al.* 2014). Seq2Seq networks were originally designed for machine translation and are meant to handle the fact that different languages often use different numbers of words to express the same idea. For example, a sentence like “No, I am your father” could be translated to Spanish as “No, soy tu padre,” which has one less word. Seq2Seq networks deal with this by processing sequences in the input with a recurrent network called the *encoder* which is connected to a separate network, called the *decoder*, via its hidden layer connections. This processed data is then unpacked by the decoder into an output sequence whose length is independent of the length of the input.

This design also makes Seq2Seq networks well suited for modelling morphological and phonological patterns (e.g., Kirov and Cotterell 2018; Prickett *et al.* 2018; Prickett 2019), since these often involve mapping between forms of different lengths. For the simulations presented in this paper, words are represented as sequences of sounds, where sounds are vectors of real-numbered features that range from 0 to 1. In the input, which represents the underlying form, standard phonological features are used (like [voice] or [back]), with 0 and 1 corresponding to [–] and [+], respectively. In the output, which represents the surface representation, the network has a binary classifier for each feature that gives the model’s estimated probability for how likely that feature is to have a positive value, given the underlying representation (UR) in its input. This is illustrated in Figure 3 using the same Hungarian example as above, with the feature vectors in the

Figure 3:  
 An example of how Hungarian vowel harmony might be handled by a Seq2Seq network. The IPA symbols shown at the top and bottom of the figure represent the model’s output and input, respectively, and stand in for vectors of real-numbered feature values. Black squares are Gated Recurrent Units and black arrows are sets of connections. The grey arrow shows the encoder’s hidden layer activations being passed to the decoder



input and the most probable sets of feature values in the output being represented using IPA symbols.

The network presented here also uses *Gated Recurrent Units* (GRU; Cho *et al.* 2014) which were designed to solve another issue with simple recurrent networks: *vanishing gradients* (Bengio *et al.* 1994), which can prohibit a network from learning long-distance dependencies. While none of the patterns I investigate have dependencies that are long enough to be affected by this phenomenon, GRU units are relatively standard in the Seq2Seq literature and I leave it to future work to see whether they are necessary for capturing the results presented here. Similarly, in all of my simulations, the network’s weights were optimized using Adam (Kingma and Ba 2015), a standard algorithm for training neural networks, but one that is likely not necessary to produce the results that I observed. The loss function used for optimization was the sum of binary cross entropy over all of the binary feature classifiers in the output and weight updates were made after seeing each word in training (i.e. batch sizes were equal to 1, sometimes called *online learning* in the phonological literature).

A final aspect of the model’s architecture worth noting is *attention* (Bahdanau *et al.* 2015). This gives the model’s decoder additional access to information from the input sequence by allowing it to see the decoder’s hidden-state activations. Attention has been shown to encourage human-like generalization in Seq2Seq networks (Nelson *et al.* 2020). Some pilot simulations without attention suggested that it helped the model generalise better in the simulations presented here.

## MAJORITY RULE HARMONY

3

*Background*

3.1

Majority Rule Harmony is a pattern predicted by some constraint-based theories of assimilation in which the number of segments in a word's underlying representation (UR) with a particular feature value determines what the value of that feature will be throughout the surface representation (SR) of the word (Lombardi 1999; Bakovic 1999). For example, if a UR has two [–anterior] segments and only one [+anterior] segment (e.g. /saʃaʃ/), then the surface representation of the word would assimilate all of the sounds to be [–anterior] (e.g. [ʃaʃaʃ]). Conversely, if a UR has two [+anterior] sounds and only a single [–anterior] one (e.g. /sasaʃ/), the surface form would instead assimilate all of the sibilants to be [+anterior] (e.g. [sasas]). Since Majority Rule requires a potentially unbounded amount of memory (i.e. enough memory to keep track of the quantities for each feature value), it cannot be represented with a finite state transducer and is more complex than the set of Regular functions (Heinz and Lai 2013).<sup>4</sup>

Finley and Badecker (2008) tested whether humans were biased against Majority Rule. They did this by training participants on a language that was ambiguous between Majority Rule Harmony and a more standard, attested harmony pattern (henceforth *Attested Harmony*), in which the value of the relevant feature in the SR was determined by the value of that feature in either the leftmost or rightmost segment of the UR (see Rose and Walker 2011, for more on the kinds of harmony patterns that are common in natural language). Directional harmony mappings like this are Subregular, since determining how a vowel will surface only depends on local information in the input and

---

<sup>4</sup>Since TSL only defines a set of languages (i.e. phonotactic restrictions on SRs) and not a set of functions (i.e. UR→SR mappings), standard harmony patterns (when represented as transformations) are *Output Tier-based Strictly Local* (Burness and McMullin 2019), a subset of Regular *functions*. See Lamont (2019b) for more on this distinction between mappings and phonotactics and its relevance to complexity in phonology.

output (Chandlee 2014; Chandlee *et al.* 2014, 2015; Graf and Mayer 2018; Burness and McMullin 2019).

Participants in the experiment were exposed to stimuli meant to represent underlying forms like /kupoki/, with both [+back] and [–back] vowels present in a single word. Crucially, the minority vowel (/i/ in this case, since it is [–back] while /o/ and /u/ are both [+back]) always occurred on the same side of the word in training. After being given each “underlying” form, participants would then be exposed to a stimulus representing the “surface” form it mapped to (e.g., [kupoku] for the example above). The mapping /kupoki/→[kupoku] could then be analysed by the participants in two ways: either Attested Harmony, where the [back] value of the final vowel changed because the leftmost vowel in the word was [+back], or Majority Rule Harmony, where the word-final /i/ changed because the majority of vowels in the underlying form were [+back].

After being exposed to a number of these ambiguous mappings, participants were asked to choose between mappings that were unambiguous between Majority Rule and Attested Harmony.<sup>5</sup> For example, they might be given /kupeki/ and need to choose between mapping it to [kupoku] (the Attested Harmony candidate) or [kipeki] (the Majority Rule candidate). If participants chose between the options at chance, it would suggest that they had no preference for either pattern. However, if they chose one significantly more often than the other, it would suggest that they were biased toward learning that pattern. Finley and Badecker (2008) found that their participants were significantly more likely to generalise in a way that adhered to Attested Harmony. That is, when choosing to either apply an Attested Harmony or Majority Rule mapping to items that were unambiguous between the two patterns, participants only applied the latter in approximately 20% of trials. This suggests that in the face of ambiguous training, the participants learned the Attested Harmony pattern – which Finley and Badecker (2008) interpreted as

---

<sup>5</sup>Thanks to a reviewer for pointing out that these forms are only unambiguous as to which of the two patterns of interest they adhere to. A number of other analyses could be used to account for both sets of words, such as a bidirectional harmony process for the Majority Rule items (where the value of [back] spreads outward from the middle vowel).



evidence of a bias against learning Supraregular patterns like Majority Rule.

Simulations

3.2

To see whether the behaviour observed by Finley and Badecker (2008) is mirrored by a Seq2Seq network, I simulated their experiment using the architecture described in Section 2. The model was exposed to the same types of training data that Finley and Badecker (2008) gave their participants, which was ambiguous between Majority Rule and Attested Harmony. Since only the vowels were relevant to the patterns in this experiment, all consonants were removed. Other than this difference, the model was exposed to the same underlying and surface forms that the experiment participants were given. These are shown in Table 1 and the features used in all the simulations presented in this subsection are shown in Table 2.

All simulations consisted of 15 repetitions using this training data, with randomly initialized weights at the start of learning, and 300 full passes through the training data (i.e., 300 epochs). At each epoch, the

Underlying Representation	Surface Representation
/o u i/	[o u u]
/e i o/	[e i e]
/u o i/	[u o u]
/i e o/	[i e e]
/o u e/	[o u o]
/u o e/	[u o o]
/e i u/	[e i i]
/i e u/	[i e i]

Table 1:  
Training data for Majority Rule simulations

	[back]	[high]
i	–	+
u	+	+
e	–	–
o	+	–

Table 2:  
Features for Majority Rule simulations

Table 3:  
Test Data for Majority Rule  
simulations. Model was given a UR as  
input (shown in the leftmost column)  
and assigned probabilities to each  
output choice (shown in the center  
and rightmost columns)

UR	Attested Harmony SR	Majority Rule SR
/o i e/	[o u o]	[e i e]
/o e i/	[o o u]	[e e i]
/u i e/	[u u o]	[i i e]
/u e i/	[u o u]	[i e i]
/i o u/	[i e i]	[u o u]
/i u o/	[i i e]	[u u o]
/e o u/	[e e i]	[o o u]
/e u o/	[e i e]	[o u o]

model was presented with the same kind of crucial forced choices that Finley and Badecker (2008) gave their participants in the experiment’s test phase (shown in Table 3).

The conditional probability that the model assigned to each choice, given a particular UR, was calculated using the equation defined in Equation 2, based on Luce (1959), where  $pr(UR_i \rightarrow SR_j)$  is found using Equation 1, and where  $f_{ij}$  stands for feature  $j$  in segment  $s_i$  of the relevant SR.

$$(1) \quad pr(UR \rightarrow SR) = \prod \prod pr(f_{ij}|UR)$$

$$(2) \quad pr(UR_i \rightarrow SR_1 | SR_1 \text{ or } SR_2) = \frac{pr(UR_i \rightarrow SR_1)}{pr(UR_i \rightarrow SR_1) + pr(UR_i \rightarrow SR_2)}$$

Results for these forced choice estimates were averaged over stimulus types and repetitions, and these averages are shown for each epoch in Figure 4. Figure 5 gives the 50th epoch in more detail, for results that are more visually comparable to the ones presented by Finley and Badecker (2008).

These results show that throughout learning, the model prefers choices that are consistent with Attested Harmony, even though it has been trained on data that is ambiguous between the two patterns. This difference reaches statistical significance for a range of epochs (including the 50th epoch), meaning that the bias in humans observed by Finley and Badecker (2008) can be captured by the model.

To further test the model’s biases in regards to Majority Rule Harmony, I also ran a simulation that does not correspond to Finley and

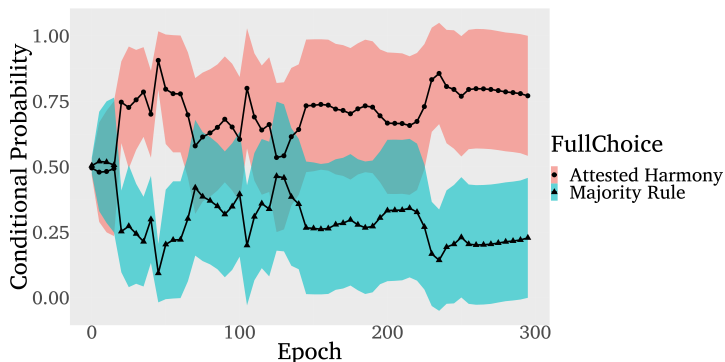


Figure 4:  
Forced choice probabilities at each epoch in learning for the simulations of Finley and Badecker (2008). Coloured regions show 95% confidence intervals

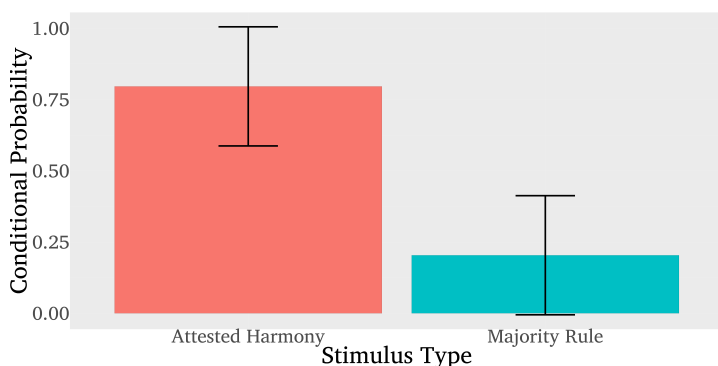


Figure 5:  
Forced choice probabilities for the 50th epoch of training in the simulation of Finley and Badecker (2008). Error bars show 95% confidence intervals

Badecker's (2008) experiment. Rather than using a generalization-based design, in this simulation, multiple, unambiguous languages were used in training. Additional data points were added to the training data in Table 1 to disambiguate the two patterns of interest. The data for unambiguous versions of Majority Rule Harmony and Attested Harmony are shown in Tables 4 and 5.

The model was trained on these unambiguous versions of Attested Harmony and Majority Rule and the cross entropy and accuracy were recorded at each epoch. Accuracy was estimated by feeding the model each of the underlying forms in the training data, sampling from the probabilities it produced in the output to create surface forms, and finding the proportion of those surface forms that were perfectly produced in that epoch's sample. The learning curves created from these results (averaged over 15 repetitions) are shown in Figure 6.

These results show that for small portions of the learning curve, Attested Harmony's average accuracy is marginally higher than

Table 4:  
Training data for the unambiguous  
Majority Rule language, based on the  
ambiguous data from Finley and  
Badecker (2008). Bolded cells show  
which data are unambiguous

Underlying Representation	Surface Representation
/o u i/	[o u u]
/e i o/	[e i e]
/u o i/	[u o u]
/i e o/	[i e e]
/o u e/	[o u o]
/u o e/	[u o o]
/e i u/	[e i i]
/i e u/	[i e i]
/o i e/	[e i e]
/o e i/	[e e i]
/u i e/	[i i e]
/u e i/	[i e i]
/i o u/	[u o u]
/i u o/	[u u o]
/e o u/	[o o u]
/e u o/	[o u o]

Table 5:  
Training data for the unambiguous  
Attested Harmony language, based on  
the ambiguous data from Finley  
and Badecker (2008). Bolded cells  
show which data are unambiguous

Underlying Representation	Surface Representation
/o u i/	[o u u]
/e i o/	[e i e]
/u o i/	[u o u]
/i e o/	[i e e]
/o u e/	[o u o]
/u o e/	[u o o]
/e i u/	[e i i]
/i e u/	[i e i]
/o i e/	[o u o]
/o e i/	[o o u]
/u i e/	[u u o]
/u e i/	[u o u]
/i o u/	[i e i]
/i u o/	[i i e]
/e o u/	[e e i]
/e u o/	[e i e]

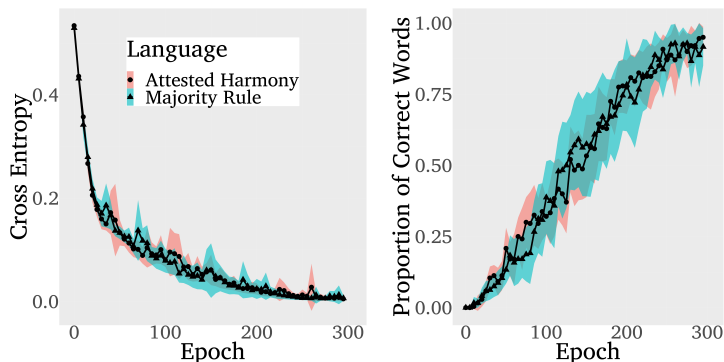


Figure 6: Learning curves for Majority Rule and Attested Harmony in the simulations using unambiguous versions of the language from Finley and Badecker (2008). Chance performance for the plot on the right would be considerably lower than 0.1, since the model assigns probabilities to each feature value in each segment. Coloured regions show 95% confidence intervals

Majority Rule's, but this difference is not a reliable one. There also seems to be a small, statistically marginal difference between the loss curves for the two patterns, but this effect is even less consistent throughout learning. Assuming that the small, artificial languages used here adequately represented each of the languages, this suggests that if the model does have a bias for Subregular patterns in its learning from unambiguous data, the effect size of this bias is too small to see in just 15 repetitions.

## FIRST-LAST ASSIMILATION

4

### *Background*

4.1

*First-Last Assimilation* is a hypothetical phonotactic restriction in which the first and last segment of a word must agree in some feature value, while the intervening sounds are ignored (Lai 2015). For example, if the feature that needed to agree was [anterior], the word [saʃas] would be allowed, but the word \*[saʃaʃ] would not be. Lai (2015) argued that there are reasonable diachronic origins for such a pattern,

since the beginning and end of a word are perceptually salient positions. She went on to argue that the absence of such a pattern in the phonological typology could be due to its FLT-based complexity.

While First-Last Assimilation is Subregular, it belongs to the *Locally Testable* region, which is more complex than TSL, in terms of the logic needed to define the crucial parts of the pattern. That is, sets of sequences are necessary to describe words banned by First-Last Assimilation (i.e. “words with either [#s] and [j#] or [#j] and [s#] are banned”), which is never true for TSL patterns.

Two studies have shown that people have biases against learning First-Last Assimilation. Lai (2015) trained participants on either a standard sibilant harmony pattern (henceforth, *Attested Harmony*) or First-Last Assimilation by having them listen to and then repeat words adhering to the pattern they were assigned to. In the testing phase of the experiment, participants were asked to judge which word was more likely to belong to the language they were trained on in three types of forced choice:<sup>6</sup>

- i. a choice between a word that was allowed in both patterns (e.g. [sasakas], denoted as FL/AH below) and a word that was only allowed in First-Last Assimilation (e.g. [saʃakas], denoted as FL/\*AH below),
- ii. a choice between a word that was allowed in both patterns and a word that was banned by both (e.g. [sasakaʃ], denoted as \*FL/\*AH below),
- iii. a choice between a word that was only allowed in First-Last Assimilation and one that was banned by both.

Participants who learned an *Attested Harmony* pattern would be expected to choose words that were allowed by both patterns when presented with choices (i) and (ii), but should choose at random for choice (iii). This is because choice (iii) forces participants to choose between two words that are both banned by the *Attested Harmony* pattern. Participants who learned a First-Last Assimilation pattern would

---

<sup>6</sup> While there are more than three logically possible forced choice options, including words that were only allowed in *Attested Harmony* would have been impossible. This is because all words that are allowed in *Attested Harmony* are also allowed in First-Last Assimilation.

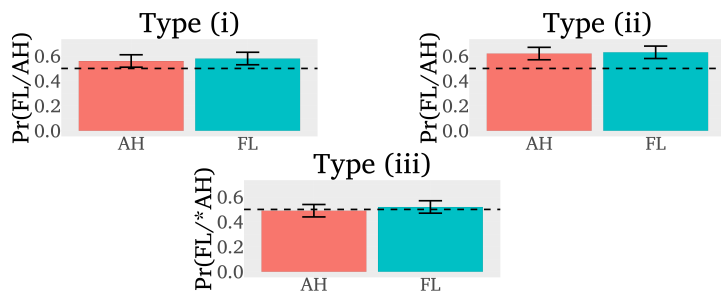


Figure 7: Results adapted from Figures 2–4 in Lai (2015). The x-axis shows which pattern participants were trained on. Type labels are mine, with “FL” standing for First-Last Assimilation, “AH” for Attested Harmony, and “\*” indicating an option not being allowed in a given pattern. Note that Lai (2015) used the term “Standard Harmony”/“SH” for the pattern I’m calling “Attested Harmony”/“AH”

be expected to choose at chance for choice (i), since both choices are grammatical according to First-Last Assimilation. For choice (ii), they would be expected to choose words that are allowed by both patterns, and for choice (iii) they should choose the words that are only allowed by First-Last Assimilation.

However, participants trained on First-Last Assimilation in Lai’s (2015) experiment did not behave as expected. Her results (reproduced in Figure 7) showed that participants in both language conditions behaved as if they had learned Attested Harmony.

Specifically, when presented with choices (i) and (ii), participants in both conditions chose items that were grammatical in both languages significantly more than chance, showing that they preferred items in which Attested Harmony was not violated. However, when presented with choice (iii), participants performed at chance, demonstrating that they had no preference between items that violated First-Last Assimilation and those that did not. This shows that they failed to learn First-Last Assimilation when trained on the pattern, and instead learned the Attested Harmony pattern. These results are what one would expect if there were a categorical restriction banning the acquisition of phonological patterns that are more complex than TSL.

Avcu (2018) ran another artificial language learning experiment to test for a bias against First-Last Assimilation. Participants received the same training as Lai’s (2015) study; however in testing, they were asked to make a different kind of choice. Instead of choosing between

two words, participants judged whether they thought each test stimulus (some of which followed the pattern from training and some which did not) belonged to the language they had just learned. This allowed Avcu (2018) to analyse participant responses using *Signal Detection Theory* (Green and Swets 1966) and provided a measure of how sensitive individuals were to whether a word belonged to the language they were assigned. The results showed that participants in both language conditions were better than chance at performing this discrimination task, but that those who learned Attested Harmony performed significantly better. Since Avcu's (2018) participants were less successful at learning First-Last Assimilation than its more standard counterpart, these results also support the idea of a bias for patterns that are simpler according to FLT.

## 4.2

### *Simulations*

To see if an explicit, FLT-related bias is needed to capture the results that Lai (2015) and Avcu (2018) observed in human learning, I ran a simulation using a Seq2Seq network.<sup>7</sup> The training and testing data that the model received were identical to the stimuli used by Lai (2015), except that all vowels were removed from the model's representations (as they were irrelevant to the patterns of interest).

Since Lai's (2015) participants were not exposed to the underlying forms for any of the stimuli, all training and testing data for the model assumed that underlying forms were identical to their corresponding surface forms (see Prince and Tesar 2004, for a similar approach to phonotactic learning). While this data represents an identity mapping, the fact that neural networks cannot perfectly learn such a mapping (Tupper and Shahriari 2016) means that the model must learn alternative ways to optimize its objective function, such as acquiring the phonotactic patterns present in the language (see Kurtz 2007, for a similar approach using a different neural network architecture). The

---

<sup>7</sup>Thanks to a reviewer for pointing me toward similar work in the domain of syntax: Ravfogel *et al.* (2019) show that a neural network, when trained on data that is ambiguous between an agreement pattern analogous to First-Last Assimilation and a pattern that involves more local agreement, the network generalises in a way that suggests it learned the latter.



Surface Representation
[ʃ s k ʃ]
[s ʃ k s]
[ʃ k s ʃ]
[s k ʃ s]
[ʃ ʃ k ʃ]
[s s k s]
[ʃ k ʃ ʃ]
[s k s s]

Table 6:  
Training data for First-Last Assimilation language in the simulations of Lai (2015). The input and output to the model was identical for all data

Surface Representation
[ʃ ʃ k ʃ]
[s s k s]
[ʃ k ʃ ʃ]
[s k s s]
[ʃ ʃ k ʃ]
[s s k s]
[ʃ k ʃ ʃ]
[s k s s]

Table 7:  
Training data for Attested Harmony language in the simulations of Lai (2015). The input and output to the model was identical for all data

	[anterior]	[sibilant]
s	+	+
ʃ	–	+
k	–	–

Table 8:  
Features and segments used in Lai (2015) simulations

training data for First-Last Assimilation and Attested Harmony are shown in Tables 6 and 7, respectively. Additionally, the features used to represent the segments in both patterns are shown in Table 8.

Simulations consisted of 15 repetitions in each language condition, with randomly initialized weights at the start of learning, and 300 passes through the full data set. At each epoch of training, the model’s cross entropy and accuracy were measured. Accuracy was estimated by feeding the model each of the forms in the training data as input, sampling from the probabilities it produced in its output to create surface forms, and finding the proportion of those surface forms

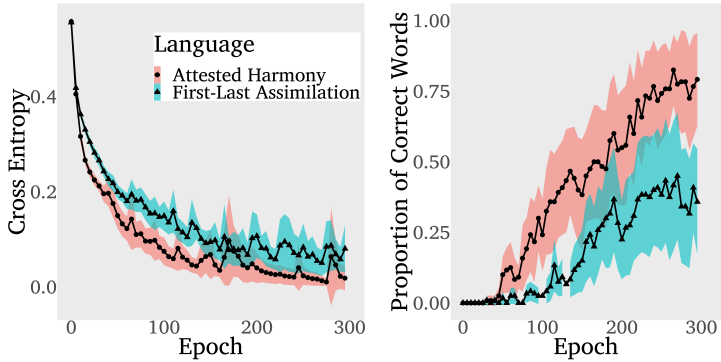


Figure 8: Learning curves for First-Last Assimilation and Attested Harmony in the simulations of Lai (2015). Chance performance for the plot on the right would be considerably lower than 0.1, since the model assigns probabilities to each feature value in each segment. Coloured regions show 95% confidence intervals

that matched their input in that epoch’s sample. Learning curves showing both of these metrics are given in Figure 8.

The curves in Figure 8 show that Attested Harmony is learned consistently faster than First-Last Assimilation. This difference is significant for considerable portions of learning in both the model’s loss and accuracy. These results are most comparable to those reported by Avcu (2018), since the model’s performance is higher than chance for both patterns, but significantly better for Attested Harmony.

To compare the model’s learning to the results in Lai (2015), the network was given a forced-choice task similar to the one described in Section 3.2, with the test data given in Table 9.

Since the patterns here were phonotactic (rather than mappings), there was no shared UR between the two choices. That is, the conditional probability that the model assigned to each choice was just a normalized probability for each of the two SRs mapping to themselves, as shown in Equation 3.

$$(3) \quad pr(SR_1|SR_1 \text{ or } SR_2) = \frac{pr(SR_1 \rightarrow SR_1)}{pr(SR_1 \rightarrow SR_1) + pr(SR_2 \rightarrow SR_2)}$$

The relevant conditional probabilities were averaged over stimulus types and repetitions, and are shown in Figure 9 and Figure 10 for the model that was trained on First-Last Assimilation and the model that was trained on Attested Harmony, respectively.

Modelling a subregular bias in phonological learning

FL/*AH Choice	*FL/*AH Choice
[s k ʃ s]	[ʃ k ʃ s]
[ʃ s k ʃ]	[ʃ s k s]
[s ʃ k s]	[s ʃ k ʃ]
[ʃ k s ʃ]	[s k s ʃ]

FL/AH Choice	*FL/*AH Choice
[s k s s]	[s k s ʃ]
[ʃ ʃ k ʃ]	[s ʃ k ʃ]
[ʃ k ʃ ʃ]	[ʃ k ʃ s]
[s s k s]	[ʃ s k s]

FL/AH Choice	FL/*AH Choice
[ʃ ʃ k ʃ]	[ʃ s k ʃ]
[s k s s]	[s k ʃ s]
[s s k s]	[s ʃ k s]
[ʃ k ʃ ʃ]	[ʃ k s ʃ]

Table 9:

Test data for First-Last Assimilation simulations. Probabilities for each form in the left column were normalized with their corresponding item in the right column. These normalized probabilities were then used to simulate the model's performance on the forced-choice task from Lai (2015)

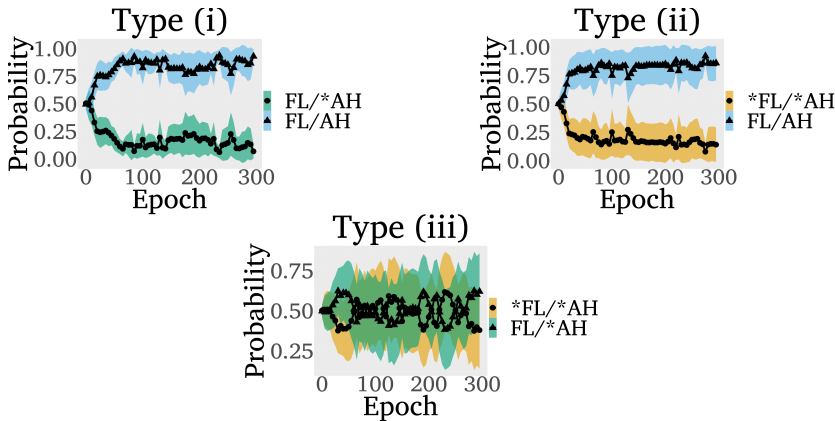


Figure 9: Forced choice probabilities at each epoch in learning for the First-Last Assimilation language. Coloured regions show 95% confidence intervals

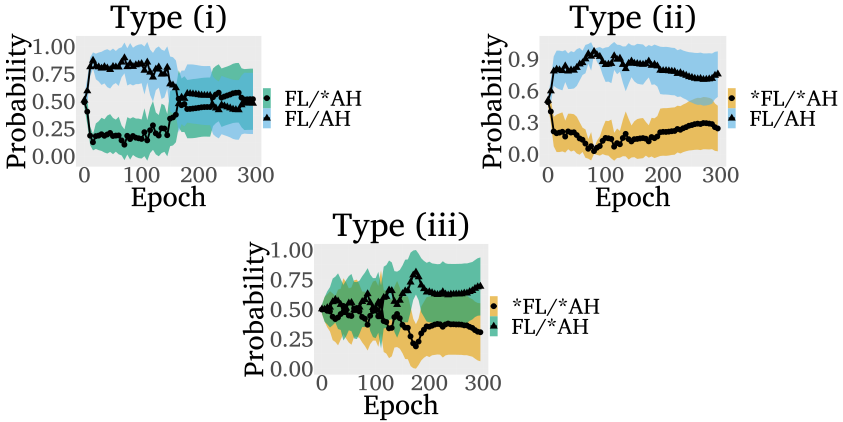


Figure 10: Forced choice probabilities at each epoch in learning for the Attested Harmony Language. Coloured regions show 95% confidence intervals

These results show that the Seq2Seq model, like the human participants in Lai (2015), behaved in a way that was consistent with Attested Harmony, even when trained on data that unambiguously followed the First-Last Assimilation pattern. That is, regardless of the model’s training data, it chose at chance between words that were banned by Attested Harmony, even when one of those words adhered to First-Last Assimilation (with the only exception to this behaviour being a small number of epochs in the Attested Harmony condition). This is shown in the results for choice (iii). By itself, this only shows that the model did not learn First-Last Assimilation. However, choices (i) and (ii) both show that the models acquired Attested Harmony, since words adhering to this pattern are consistently given more probability than words banned by it for most of the acquisition process.<sup>8</sup> To show these results in a way that is more visually comparable to the results reported in Lai (2015), the model’s estimates for the 100th

<sup>8</sup>Although, note that toward the end of learning, the model trained on the attested pattern begins to choose at chance in all three of the choice types. This could be due to the model eventually learning to faithfully map the segments in the input in those cases. While this approximates an identity mapping for the segments that were present in the training, it would not be a true identity mapping, since neural networks trained with algorithms like Adam cannot capture identity-based functions (Tupper and Shahriari 2016).

## Modelling a subregular bias in phonological learning

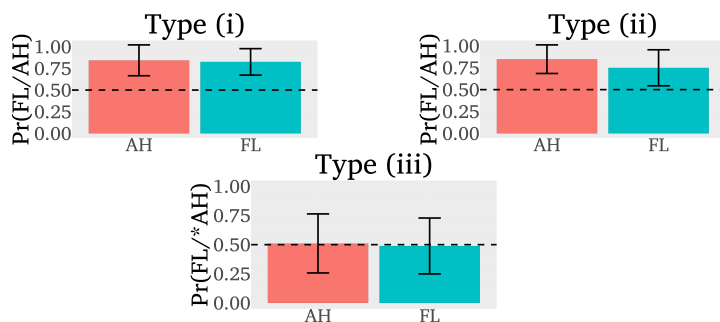


Figure 11: Forced choice probabilities for the 100th epoch of training in both the First-Last Assimilation language and the Attested Harmony Language. The dashed line shows chance and the error bars show 95% confidence intervals. As in Figure 7, “FL” stands for First-Last Assimilation, “AH” stands for Attested Harmony, and “\*” indicates an option not being allowed in a given pattern

epoch in each language, which was a relatively representative point in each language’s learning curve, are shown in Figure 11.

## DISCUSSION

5

### *Why can the Seq2Seq network capture these biases?*

5.1

In this paper, I showed that the apparent FLT-related bias observed in past artificial language learning experiments could be modeled by a recurrent neural network with no FLT-based restrictions built into its architecture. But the question of why these biases exist has not been addressed. One reason for the model’s bias against Majority Rule Harmony could be its inability to count. Weiss *et al.* (2018) showed that GRU units, like the one used in the hidden layer of the neural network I tested, prohibit a model from acquiring the ability to count (as opposed to simple recurrent networks and networks with LSTM units, which were able to learn counting-based patterns). Since Majority Rule Harmony requires counting the occurrences of a particular feature value in the input, this could explain the model’s preference

for learning an Attested Harmony pattern in the face of ambiguous data.

Another relevant factor is the locality bias (sometimes also called “sequentiality”; Battaglia *et al.* 2018) present in all recurrent network architectures. This is a bias for patterns that involve local dependencies, originating from the fact that recurrent connections have a finite amount of memory with which to store information across time. Past results on syntactic patterns have shown that this bias can cause RNNs to learn a local agreement pattern when given ambiguous evidence between that and a non-local one (Ravfogel *et al.* 2019). Similarly, McCoy *et al.* (2020) showed that Seq2Seq neural networks similar to the one used here were more likely to learn syntactic patterns that depended on linear order, which typically involves more local dependencies, than patterns that depended on hierarchical structure, which typically involves longer distance dependencies. Since First-Last Assimilation also involves non-local dependencies (i.e. two arbitrarily distant first and last segments), the network could have struggled to keep track of the relevant feature values in its recurrent connections when acquiring that pattern.<sup>9</sup>

## 5.2

### *Future work*

This paper has shown that three experiments that found evidence supporting an FLT-based bias in humans (Finley and Badecker 2008; Lai 2015; Avcu 2018) can be simulated using a Seq2Seq recurrent neural network. Future work should continue to explore the phonological learning biases present in both humans and computational models. For example, one phonological pattern that was not discussed here but which the literature has discussed in detail is *Sour Grapes Harmony* (Bakovic 2000; Wilson 2003). *Sour Grapes* is identical to Standard Harmony, except when a segment that blocks the harmony process is

---

<sup>9</sup>The difference between local and non-local dependencies has been thoroughly explored in the statistical learning literature as well (e.g., Newport and Aslin 2004), and simulations of such statistical learning experiments with RNNs have been performed (see, e.g., Farkaš 2008). I leave exploring the relationship between these experiments and those that have been used to support FLT-based biases in phonology to future work.

present in a word. When this happens, any changes that would have occurred up to the blocker are prevented from occurring at all. Like First-Last Assimilation and Majority Rule, Sour Grapes is unattested in natural language and more complex than the Tier-based Strictly Local region of the Subregular Hierarchy (O'Hara and Smith 2019; Lamont 2019b).

Another avenue for future work is using more realistic artificial languages. In all of the experiments simulated here, word length was kept constant. When testing the effects of formal complexity on human learning, generalization to novel lengths has been shown to be crucial in understanding human bias (Westphal-Fitch *et al.* 2018). Further research that makes use of variable lengths in its training and testing data could shed light on whether humans display an FLT-based bias under these more realistic conditions.

Researchers should also explore how the predictions about human learnability made by FLT and neural networks differ. For example, certain Context-Sensitive patterns are easier for neural networks and humans to learn than corresponding Context-Free patterns (Li *et al.* 2013; Westphal-Fitch *et al.* 2018), despite the fact that Context-Sensitive is more complex according to FLT. Exploring whether mismatches like this occur in phonological patterns could shed more light on how psychologically real FLT-based complexity is.

Understanding better *why* the neural network is able to capture these results and what representations it learns while doing so is another important next step. While the interpretability of recurrent networks has primarily been explored in the context of syntactic patterns and language modelling (see, e.g., Alishahi *et al.* 2019, for a review), some recent work on phonological patterns has shown promising results in this direction (Nelson *et al.* 2020; Smith *et al.* 2021) and these techniques could likely be applied to the networks used here.

Finally, a number of choices about the model I used were made somewhat arbitrarily: the number of hidden states in each layer, the use of GRU instead of a different kind of recurrent layer in the model, the use of attention, *et cetera*. Changing any one of these would likely have an effect on the model's ability to capture the experiment results investigated in Section 3 and Section 4, and I leave exploring the consequences of such changes to future work.

5.3 *The relationship between FLT and other complexity metrics*

The Subregular Hierarchy is not the only way of measuring complexity that has been used in phonological research. Feature counting (Chomsky and Halle 1968), Minimum Description Length (Rasin and Katzir 2016), and various other methods (e.g. Moreton *et al.* 2017) have been used to characterize the complexity of phonological patterns. While these other methods are related to FLT, they are not perfectly correlated with it. For example, a feature-counting complexity metric would find a pattern banning all voiced sounds at the end of words (i.e., \*[+voice]#) to be simpler than a pattern banning voiced, velar stops in that context (i.e., \*[+voice, Dorsal]#). However, according to FLT, these patterns would both be Strictly Local, with no difference in complexity. Exploring the relationship between FLT and these other metrics is outside the scope of this paper; however future work should investigate what formalizations of complexity best predict both human behavior and linguistic typology (see, e.g., Moreton and Pater 2012).

5.4 *Conclusions*

Past work has explained phonological typology using an explicit, categorical restriction that prohibits the acquisition of patterns that are too complex according to the Subregular Hierarchy. Evidence for this hypothesis includes a series of experiments that showed humans being affected by an apparent FLT-based bias in an artificial language learning context (Finley and Badecker 2008; Lai 2015; Avcu 2018).

The results in this paper challenge the idea that a categorical, explicit bias like this is needed to capture phonological learning, since a Seq2Seq neural network with the expressive power to represent Supraregular patterns was able to capture these experimental results. While FLT can be useful for describing phonological typology, these results suggest that an explicit FLT-based bias may not be needed in models of phonological learning.



## ACKNOWLEDGEMENT

The author would like to thank Joe Pater, Gaja Jarosz, John Kingston, Mohit Iyyer, Katya Pertsova, and Andrew Lamont for helpful discussion. Thanks also to the reviewers for their valuable feedback.

## REFERENCES

- John ALDERETE and Paul TUPPER (2018), Connectionist Approaches to Generative Phonology, *The Routledge Handbook of Phonological Theory*. Routledge.
- Afra ALISHAHI, Grzegorz CHRUPAŁA, and Tal LINZEN (2019), Analyzing and Interpreting Neural Networks for NLP: A Report on the First BlackboxNLP Workshop, *arXiv preprint arXiv:1904.04063*.
- Enes AVCU (2018), Experimental Investigation of the Subregular Hypothesis, in *Proceedings of the 35th West Coast Conference on Formal Linguistics*, pp. 77–86.
- Dzmitry BAHDANAU, Kyunghyun CHO, and Yoshua BENGIO (2015), Neural Machine Translation by Jointly Learning to Align and Translate, in *3rd International Conference on Learning Representations, Conference Track Proceedings*.
- Eric BAKOVIC (1999), Assimilation to the Unmarked, *University of Pennsylvania Working Papers in Linguistics*, 6(1):2.
- Eric BAKOVIC (2000), *Harmony, dominance and control*, PhD Thesis, Rutgers University.
- Peter W BATTAGLIA, Jessica B HAMRICK, Victor BAPST, Alvaro SANCHEZ-GONZALEZ, Vinicius ZAMBALDI, Mateusz MALINOWSKI, Andrea TACCHETTI, David RAPOSO, Adam SANTORO, Ryan FAULKNER, *et al.* (2018), Relational Inductive Biases, Deep Learning, and Graph Networks, *arXiv preprint arXiv:1806.01261*.
- Yoshua BENGIO, Patrice SIMARD, and Paolo FRASCONI (1994), Learning Long-term Dependencies with Gradient Descent is Difficult, *IEEE Transactions on Neural Networks*, 5(2):157–166.
- Wm G BENNETT (2015), *The phonology of Consonants: Harmony, Dissimilation and Correspondence*, Cambridge University Press.

Phillip BURNES and Kevin MCMULLIN (2019), Efficient Learning of Output Tier-based Strictly 2-local Functions, in *Proceedings of the 16th Meeting on the Mathematics of Language*, pp. 78–90.

Jane CHANDLEE (2014), *Strictly Local Phonological Processes*, PhD Thesis, University of Delaware.

Jane CHANDLEE, Rémi EYRAUD, and Jeffrey HEINZ (2015), Output Strictly Local Functions, in *14th Meeting on the Mathematics of Language*, pp. 112–125.

Jane CHANDLEE, Rémi EYRAUD, and Jeffrey HEINZ (2014), Learning Strictly Local Subsequential Functions, *Transactions of the Association for Computational Linguistics*, 2:491–504.

Kyunghyun CHO, Bart VAN MERRIËNBOER, Dzmitry BAHDANAU, and Yoshua BENGIO (2014), On the Properties of Neural Machine Translation: Encoder–Decoder Approaches, in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pp. 103–111, Association for Computational Linguistics.

Noam CHOMSKY (1956), Three Models for the Description of Language, *IRE Transactions on Information Theory*, 2(3):113–124.

Noam CHOMSKY and Morris HALLE (1968), *The Sound Pattern of English*, Harper & Row.

Amanda DOUCETTE (2017), Inherent Biases of Recurrent Neural Networks for Phonological Assimilation and Dissimilation, in *Proceedings of the 7th Workshop on Cognitive Modeling and Computational Linguistics*, pp. 35–40.

Jeffrey L. ELMAN (1990), Finding Structure in Time, *Cognitive science*, 14(2):179–211.

Igor FARKAŠ (2008), Learning Nonadjacent Dependencies with a Recurrent Neural Network, in *International Conference on Neural Information Processing*, pp. 292–299, Springer.

Sara FINLEY (2017), Locality and Harmony: Perspectives from Artificial Grammar Learning, *Language and Linguistics Compass*, 11(1):1–16.

Sara FINLEY and William BADECKER (2008), Analytic biases for vowel harmony languages, in *West Coast Conference on Formal Linguistics*, volume 27, pp. 168–176.

Michael GASSER (1993), *Learning Words in Time: Towards a Modular Connectionist Account of the Acquisition of Receptive Morphology*, Indiana University, Department of Computer Science.

Michael GASSER and Chan-Do LEE (1992), Networks that Learn about Phonological Feature Persistence, in *Connectionist Natural Language Processing*, pp. 349–362, Springer.

- Thomas GRAF and Connor MAYER (2018), Sanskrit n-Retroflexion is Input-Output Tier-Based Strictly Local, in *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pp. 151–160.
- David Marvin GREEN and John A. SWETS (1966), *Signal Detection Theory and Psychophysics*, volume 1, Wiley.
- Mary HARE (1990), The Role of Trigger-target Similarity in the Vowel Harmony Process, in *Annual Meeting of the Berkeley Linguistics Society*, volume 16, pp. 140–152.
- Jeffrey HEINZ (2010), Learning Long-distance Phonotactics, *Linguistic Inquiry*, 41(4):623–661.
- Jeffrey HEINZ (2018), The computational nature of phonological generalizations, in *Phonological typology*, pp. 126–195, De Gruyter Mouton.
- Jeffrey HEINZ and William IDSARDI (2011), Sentence and Word Complexity, *Science*, 333(6040):295–297.
- Jeffrey HEINZ and Regine LAI (2013), Vowel Harmony and Subsequentiality, in *Proceedings of the 13th Meeting on the Mathematics of Language (MoL 13)*, pp. 52–63.
- Jeffrey HEINZ, Chetan RAWAL, and Herbert G TANNER (2011), Tier-based Strictly Local Constraints for Phonology, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pp. 58–64, Association for Computational Linguistics.
- Adam JARDINE and Jeffrey HEINZ (2016), Learning Tier-based Strictly 2-local Languages, *Transactions of the Association for Computational Linguistics*, 4:87–98.
- C. Douglas JOHNSON (1972), *Formal Aspects of Phonological Description*, Mouton & Co. N.V.
- Michael I. JORDAN (1986), Serial Order: A Parallel Distributed Processing Approach., Technical report, University of California, San Diego.
- Ronald M. KAPLAN and Martin KAY (1994), Regular Models of Phonological Rule Systems, *Computational Linguistics*, 20(3):331–378.
- Diederik P. KINGMA and Jimmy BA (2015), Adam: A Method for Stochastic Optimization, in *3rd International Conference on Learning Representations, Conference Track Proceedings*.
- Christo KIROV and Ryan COTTERELL (2018), Recurrent Neural Networks in Linguistic Theory: Revisiting Pinker and Prince (1988) and the Past Tense Debate, *Transactions of the Association for Computational Linguistics*, 6:651–665.
- Kenneth J. KURTZ (2007), The Divergent Autoencoder (DIVA) Model of Category Learning, *Psychonomic Bulletin & Review*, 14(4):560–576.

- Regine LAI (2015), Learnable vs. Unlearnable Harmony Patterns, *Linguistic Inquiry*, 46(3):425–451.
- Andrew LAMONT (2018), Precedence is Pathological: The Problem of Alphabetical Sorting, *Proceedings of the 36th West Coast Conference on Formal Linguistics*, pp. 243–249.
- Andrew LAMONT (2019a), Majority Rule in Harmonic Serialism, in *Proceedings of the Annual Meetings on Phonology*, volume 7.
- Andrew LAMONT (2019b), Sour Grapes is Phonotactically Complex, Linguistic Society of America, 2019 Annual Meeting.
- Feifei LI, Shan JIANG, Xiuyan GUO, Zhiliang YANG, and Zoltan DIENES (2013), The Nature of the Memory Buffer in Implicit Learning: Learning Chinese Tonal Symmetries, *Consciousness and cognition*, 22(3):920–930.
- Linda LOMBARDI (1999), Positional Faithfulness and Voicing Assimilation in Optimality Theory, *Natural Language & Linguistic Theory*, 17(2):267–302.
- R. Duncan LUCE (1959), *Individual Choice Behavior*, Dover Publications.
- Gary MARCUS, Sugumaran VIJAYAN, S. Bandi RAO, and Peter M. VISHTON (1999), Rule Learning by Seven-month-old Infants, *Science*, 283(5398):77–80.
- R. Thomas MCCOY, Robert FRANK, and Tal LINZEN (2020), Does Syntax Need to Grow on Trees? Sources of Hierarchical Inductive Bias in Sequence-to-sequence Networks, *Transactions of the Association for Computational Linguistics*, 8:125–140.
- Kevin MCMULLIN and Gunnar Ólafur HANSSON (2019), Inductive Learning of Locality Relations in Segmental Phonology, *Laboratory Phonology: Journal of the Association for Laboratory Phonology*, 10(1).
- Kevin James MCMULLIN (2016), *Tier-based Locality in Long-distance Phonotactics: Learnability and Typology*, Ph.D. thesis, University of British Columbia.
- Elliott MORETON and Joe PATER (2012), Structure and Substance in Artificial-phonology Learning, Part I: Structure, *Language and Linguistics Compass*, 6(11):686–701.
- Elliott MORETON, Joe PATER, and Katya PERTSOVA (2017), Phonological Concept Learning, *Cognitive science*, 41(1):4–69.
- Max NELSON, Hossep DOLATIAN, Jonathan RAWSKI, and Brandon PRICKETT (2020), Probing RNN Encoder-decoder Generalization of Subregular Functions using Reduplication, *Proceedings of the Society for Computation in Linguistics*, 3(1):31–42.
- Elissa L. NEWPORT and Richard N. ASLIN (2004), Learning at a Distance I. Statistical Learning of Non-adjacent Dependencies, *Cognitive psychology*, 48(2):127–162.

- Charlie O'HARA and Caitlin SMITH (2019), Computational Complexity and Sour-Grapes-like Patterns, in *Proceedings of the Annual Meetings on Phonology*, volume 7.
- Brandon PRICKETT (2019), Learning Biases in Opaque Interactions, *Phonology*, 36(4):627–653.
- Brandon PRICKETT, Aaron TRAYLOR, and Joe PATER (2018), Seq2Seq Models with Dropout can Learn Generalizable Reduplication, in *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pp. 93–100.
- Alan PRINCE and Bruce TESAR (2004), Learning Phonotactic Distributions, *Constraints in phonological acquisition*, pp. 245–291.
- Ezer RASIN and Roni KATZIR (2016), On Evaluation Metrics in Optimality Theory, *Linguistic Inquiry*, 47(2):235–282.
- Shauli RAVFOGEL, Yoav GOLDBERG, and Tal LINZEN (2019), Studying the Inductive Biases of RNNs with Synthetic Variations of Natural Languages, in *Proceedings of NAACL-HLT*, pp. 3532–3542.
- Sharon ROSE and Rachel WALKER (2011), Harmony Systems, *The handbook of phonological theory*, 2:240–290.
- DE RUMELHART and JL MCCLELLAND (1986), On Learning the Past Tenses of English Verbs, in JL MCCLELLAND and DE RUMELHART, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 2: Psychological and Biological Models, pp. 216–271, The MIT Press.
- Edward SAPIR and Harry HOIJER (1967), *The Phonology and Morphology of the Navaho Language*, University of California Press.
- Hava T. SIEGELMANN (1999), *Neural Networks and Analog Computation: Beyond the Turing Limit*, Springer Science & Business Media.
- Caitlin SMITH, Charlie O'HARA, Eric ROSEN, and Paul SMOLENSKY (2021), Emergent Gestural Scores in a Recurrent Neural Network Model of Vowel Harmony, *Proceedings of the Society for Computation in Linguistics*, 4(1):61–70.
- Ilya SUTSKEVER, Oriol VINYALS, and Quoc V. LE (2014), Sequence to Sequence Learning with Neural Networks, in *Advances in Neural Information Processing Systems*, pp. 3104–3112.
- David S. TOURETZKY (1989), Towards a Connectionist Phonology: The “Many Maps” Approach to Sequence Manipulation, in *Proceedings of the 11th Annual Conference of the Cognitive Science Society*, pp. 188–195.
- David S. TOURETZKY and Deirdre W. WHEELER (1990), A Computational Basis for Phonology, in *Advances in Neural Information Processing Systems*, pp. 372–379.

Paul TUPPER and Bobak SHAHRIARI (2016), Which Learning Algorithms Can Generalize Identity-Based Rules to Novel Inputs?, *arXiv preprint arXiv:1605.04002*.

Gail WEISS, Yoav GOLDBERG, and Eran YAHAV (2018), On the Practical Computational Power of Finite Precision RNNs for Language Recognition, in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 740–745.

Gesche WESTPHAL-FITCH, Beatrice GIUSTOLISI, Carlo CECCHETTO, Jordan Scott MARTIN, and W. Tecumseh FITCH (2018), Artificial Grammar Learning Capabilities in a Visual Task Match Requirements for Linguistic Syntax, *Frontiers in psychology*, 9:1210.

Colin WILSON (2003), Analyzing Unbounded Spreading with Constraints: Marks, Targets, and Derivations, *Unpublished manuscript, University of California, Los Angeles*.

*Brandon Prickett*

© 0000-0001-9217-2130


bprickett@umass.edu

University of Massachusetts Amherst

Brandon Prickett (2021), *Modelling a subregular bias in phonological learning with Recurrent Neural Networks*, *Journal of Language Modelling*, 9(1):67–96

doi <https://dx.doi.org/10.15398/jlm.v9i1.251>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

cc  <http://creativecommons.org/licenses/by/4.0/>

# Investigating the effects of i-complexity and e-complexity on the learnability of morphological systems

Tamar Johnson<sup>1</sup>, Kexin Gao<sup>1</sup>, Kenny Smith<sup>1</sup>, Hugh Rabagliati<sup>2</sup>,  
and Jennifer Culbertson<sup>1</sup>

<sup>1</sup> Centre for Language Evolution, University of Edinburgh

<sup>2</sup> Department of Psychology, University of Edinburgh

## ABSTRACT

Research on cross-linguistic differences in morphological paradigms reveals a wide range of variation on many dimensions, including the number of categories expressed, the number of unique forms, and the number of inflectional classes. However, in an influential paper, Ackerman and Malouf (2013) argue that there is one dimension on which languages do not differ widely: in predictive structure. Predictive structure in a paradigm describes the extent to which forms predict each other, called i-complexity. Ackerman and Malouf (2013) show that although languages differ according to measure of surface paradigm complexity, called e-complexity, they tend to have low i-complexity. They conclude that morphological paradigms have evolved under a pressure for low i-complexity. Here, we evaluate the hypothesis that language learners are more sensitive to i-complexity than e-complexity by testing how well paradigms which differ on only these dimensions are learned. This could result in the typological findings Ackerman and Malouf (2013) report if even paradigms with very high e-complexity are relatively easy to learn, so long as they have low i-complexity. First, we summarize a recent work by Johnson *et al.* (2020) suggesting that both neural networks and human learners may

*Keywords:*  
*morphological complexity, learning, neural networks, typology*

actually be more sensitive to e-complexity than i-complexity. Then we build on this work, reporting a series of experiments which confirm that, indeed, across a range of paradigms that vary in either e- or i-complexity, neural networks (LSTMs) are sensitive to both, but show a larger effect of e-complexity (and other measures associated with size and diversity of forms). In human learners, we fail to find any effect of i-complexity on learning at all. Finally, we analyse a large number of randomly generated paradigms and show that e- and i-complexity are negatively correlated: paradigms with high e-complexity necessarily show low i-complexity. We discuss what these findings might mean for Ackerman and Malouf's hypothesis, as well as the role of ease of learning versus generalization to novel forms in the evolution of paradigms.

1

## INTRODUCTION

Languages differ widely in their morphological systems, including substantial variation in their inflectional paradigms; some languages do not use morphology to mark grammatical information at all (e.g. Mandarin) whereas others make use of inflectional morphology to mark dozens of grammatical functions (e.g. Arabic). Intuitively, this kind of variation should have an effect on how easy or difficult it is to learn a morphological system – the more inflected forms for each lexeme there are, the more difficult learning should be. Indeed, using the size of an inflectional paradigm is a common method for measuring morphological complexity, for example by counting the number of potential inflections a verb or a noun can be marked with (e.g. Shosted 2006; Bickel and Nichols 2013). In addition to the number of inflectional categories, the size of a morphological system is also impacted by the number of inflection classes, i.e. different realizations for the same morphosyntactic or morphosemantic distinction across groups of lexemes (Aronoff 1994; Corbett 2009), which has also been claimed to be a source of complexity in morphological systems (e.g. Baerman *et al.* 2010; Ackerman and Malouf 2013). These aspects of morphological complexity, which pertain to the size of a morphological sys-



tem, are all referred to as enumerative complexity or e-complexity (e.g. Ackerman and Malouf 2013; Meinhardt *et al.* 2019).

Recently, another measure of the complexity of morphological paradigms has been suggested, referred to as integrative complexity, or i-complexity. I-complexity refers to the organization of the inflected forms in the paradigm and the relations between the forms that such organization generates; in paradigms with low i-complexity, forms are predictive of one another (e.g. Blevins 2006; Ackerman and Malouf 2013). Proponents of this measure suggest that i-complexity reflects the difficulty speakers face in generating forms they have not previously encountered, based on known forms of the same lexeme (the Paradigm Cell Filling Problem, Ackerman and Malouf 2013, 2015). Predictive structure in a morphological system can be seen in Table 1 below, which shows the Russian nominal inflection paradigm. This paradigm has four inflectional classes, and inflections for two number categories and six case categories. The nominative singular *-o* is predictive of all the other case forms (i.e. if you know that a given noun takes *-o* in the nominative singular you can predict its inflection in any other combination of case and number); in contrast, the nominative plural *-i* is less predictive, since nouns which take that inflection show variation in inflectional marking elsewhere.

Crucially, Ackerman and Malouf (2013) observe that across natural language paradigms, while the size or e-complexity vary widely, i-complexity is consistently low. Further they show that high

Table 1: Russian nominal inflection paradigm (phonological transcription). Nouns fall into one of 4 inflection classes (rows) which show different patterns of inflection; nouns are inflected for number (SG=singular, PL=plural) and case (NOM=nominative, ACC=accusative, GEN=genitive, DAT=dative, LOC=locative, INS=instrumental)

	SG						PL					
	NOM	ACC	GEN	DAT	LOC	INS	NOM	ACC	GEN	DAT	LOC	INS
noun class 1	-o	-o	-a	-u	-e	-om	-a	-a	∅	-am	-ax	-am'i
noun class 2	∅	∅	-a	-u	-e	-om	-i	-i	-ov	-am	-ax	-am'i
noun class 3	-a	-u	-i	-e	-e	-oj	-i	-i	∅	-am	-ax	-am'i
noun class 4	∅	∅	-i	-i	-i	-ju	-i	-i	-ej	-am	-ax	-am'i

e-complexity paradigms tend to have low i-complexity. They conclude that i-complexity is therefore a primary measure of complexity which shapes the types of morphological paradigms attested cross-linguistically.

Ackerman and Malouf (2015) further suggest that the pressure for low i-complexity shapes languages through the dynamics of language change. Specifically, during language use, low i-complexity may assist language users in solving the Paradigm Cell Filling Problem, and further, errors language users make when generalizing to unknown forms may be i-complexity-reducing. This idea is also compatible with the general hypothesis that languages evolve to maximise learnability (e.g. Deacon 1997; Kirby 2002; Christiansen and Chater 2008; Kirby *et al.* 2008; Culbertson and Kirby 2016). In this case, a learning bias against high i-complexity paradigms would drive i-complexity down over generations of learners. If i-complexity affects learning and use more than other aspects of complexity, then the former might end up being constrained across languages, while the latter may vary quite freely. That said, from this perspective the substantial variation in languages' e-complexity that Ackerman and Malouf (2013) observe is on its face surprising. We might reasonably expect that higher e-complexity also poses challenges for language learners; and the existence of languages with large morphological paradigms and numerous inflectional classes in particular is puzzling.

Here we compare how different sources of morphological complexity affect learnability of inflectional paradigms. We focus on the two types of measures described above: e-complexity as reflected in the number of inflection classes in a paradigm and the distribution of their forms, and i-complexity as reflected in the predictability of forms in a paradigm based on other parts of the paradigm. We also investigate how these interact with the number of different markers in the system, another aspect of the e-complexity of the paradigm, and different types of syncretism. Syncretism is a phenomenon in which different cells in an inflectional paradigm are realized by the same phonological form. Whether the same phonological form marks semantically related meanings or is accidental homonymy, has been suggested to affect the learning of the forms (e.g. Baerman *et al.* 2005; Pertsova 2012; Maldonado and Culbertson 2019). For example, in Table 1, *-o* is used for semantically related forms – class 1 nouns which

differ in case. However, *-a* can be considered accidental homophony as it is used across different classes for different cases.

The paper proceeds as follows. We first outline more precisely how *e-* and *i-*complexity are calculated in this study. We then discuss previous work aimed at providing empirical evidence for the link between *i-*complexity and learning of morphological paradigms. This work has highlighted the role of predictive structure in producing novel inflections, i.e. generalization. In Section 2 we report a series of experiments using LSTM neural network and human learners testing the related hypothesis that low *i-*complexity provides a more general facilitatory effect on learning than *e-*complexity, including facilitating the retrieval of already-encountered forms early in learning. While the biases of human learners are obviously of primary interest in understanding the pressures that shape human language, we use neural networks as a convenient model of an ‘ideal learner’. Testing such a learner serves to provide proof-in-principle for whether *i-*complexity can affect learnability and whether its influence is greater than other types of morphological complexity. For both human and network learners we see similar results, contrary to the hypothesis above; *e-*complexity generally impacts learning more than *i-*complexity. Finally, in Section 3 we explore the relationship between the *i-* and *e-*complexity by generating a large number of random paradigms with different values of these two measures. Here we find that *i-*complexity and *e-*complexity are highly negatively correlated: as the number of distinct forms increases, the implicative structure between forms also necessarily increases. Furthermore, the range of *e-*complexity values is also necessarily higher than the range of *i-*complexity values for paradigms of the same size. These findings suggest that the observations made by Ackerman and Malouf (2013) concerning morphological paradigms may stem in part from the nature of the measures rather than pressures (e.g. inductive or usage biases) that are specially attuned to *i-*complexity.

### *Measuring i-complexity and e-complexity*

1.1

Here we adopt methods for calculating *i-*complexity outlined in Ackerman and Malouf (2013). The *i-*complexity of inflectional paradigms

is measured using the information-theoretic notion of entropy (Shannon 1963), specifically the averaged conditional entropy of forms in the paradigm. The conditional entropy of a pair of grammatical functions  $X, Y$  in the paradigm is presented in (1) below. Here  $P(x, y)$  indicates the joint probability of the two grammatical functions in the paradigm being realized as forms  $x$  and  $y$ , respectively;  $P(y|x)$  indicates the conditional probability of  $Y$  being realized as  $y$ , given that  $X$  is realized as  $x$ . Conditional entropy  $H(Y|X)$  quantifies the uncertainty associated with the value of  $Y$  given the value of  $X$ . For example, looking at the Russian nominal inflection paradigm in Table 1, let  $Y$  be the set of forms realizing SG.NOM, [-o,  $\emptyset$ , -a,  $\emptyset$ ], and  $X$  be the set of forms realizing SG.DAT, [-u, -u, -e, -i]. The conditional entropy of SG.NOM given the form in SG.DAT would represent the uncertainty associated with the form in SG.NOM, when the form realizing SG.DAT for the same lexeme is known.

$$(1) \quad H(Y|X) = - \sum_{x \in X} \sum_{y \in Y} P(x, y) \log_2 P(y|x)$$

A paradigm's total i-complexity is the averaged conditional entropy over all pairs of grammatical functions in the paradigm, as in (2),

$$(2) \quad \frac{\sum_{Y \in G} \sum_{X \in G} H(X|Y)}{N_G(N_G - 1)},$$

where  $G$  is the set of grammatical functions in the paradigm and  $N_G$  is their total number.<sup>1</sup>

Although Ackerman and Malouf (2013) do not explicitly suggest a measure for e-complexity, we adopt here their average cell entropy as a measure for e-complexity. The cell entropy, defined in (3) below, captures the number of inflection classes and the number of different variants to mark each grammatical function (e.g. combinations of number and case in the Russian nominal inflection paradigm above). Intuitively, grammatical functions that are realized with a large set

---

<sup>1</sup>Note that this is not the only way of calculating i-complexity. For alternative formulations, see Malouf (2017) as well as Bonami and Beniamine (2016) and Sims and Parker (2016), who propose alternative formulations which are less dependent on linguist-constructed paradigms.

of optional forms, or do not have a dominant/frequent variant, have higher cell entropy. The difference between these two measures rests in the extent to which they take into account the inter-predictability of forms across the paradigm. I-complexity is specifically defined to measure the degree to which one form can be guessed based on another form, in any other cell of the paradigm. In other words, it critically involves predicting the form of a lexeme in some grammatical function based on the form of that lexeme in a different grammatical function. By contrast, average cell entropy is only defined in terms of a single grammatical function, i.e. it is based on what one can predict from the form of other lexemes for that grammatical function. Average cell entropy is thus suitable for measuring what is crucially different about e-complexity as compared to i-complexity.<sup>2</sup> For example, Ackerman and Malouf (2013) illustrate at their claim that paradigms tend to have low i-complexity but vary in their e-complexity using the average conditional entropy and average cell entropy, respectively.

$$(3) \quad H(X) = - \sum_{x \in X} P(x) \log_2 P(x)$$

---

<sup>2</sup>We further discuss the relationship between average cell entropy and another common measures of e-complexity, number of forms in the paradigm, in Section 3. In general, we prefer average cell entropy over simply counting the number of forms in the paradigm, or number of forms for a given grammatical function, because the entropy-based measure also accounts for the frequency with which forms are used across a grammatical function. For example, in the Russian paradigm above, SG.GEN and SG.LOC both are expressed with two affixes, but the skewed distribution over those two affixes for SG.LOC reduces uncertainty (the appropriate affix is more likely to be *-e* than *-i*), which the entropy-based measure captures. However, it should be noted that Malouf (p.c.) has suggested that the number of forms, but not average cell entropy, should be considered a measure of e-complexity. They argue this based on the fact that average cell entropy, like the measure of i-complexity we use, also reflects predictive relationships within the paradigm (just not across grammatical forms for a given lexeme). We would argue against this interpretation, since the number of forms – an uncontroversial measure of e-complexity – can also be considered predictive in this way, as it affects how well a form can be predicted based on knowledge of all the forms in the paradigm. Put another way, a paradigm with fewer forms makes any given form easier to guess.

E-complexity is measured as the averaged cell entropy over all grammatical functions in a paradigm as in (4),

$$(4) \quad \frac{\sum_{X \in G} H(X)}{N_G},$$

where  $G$  is the set of grammatical functions in the paradigm and  $N_G$  is their total number.

## 1.2 *Previous work investigating the effects of complexity on morphological learnability*

As mentioned above, Ackerman and Malouf (2013) find evidence that while morphological paradigms differ widely in their e-complexity, the range of i-complexity values appears to be more constrained. They calculate both e- and i-complexity for inflectional paradigms in a set of 10 geographically and genetically varying languages. The e-complexity values they report (as measured by average cell entropy) ranged between 0.78 and 4.9 bits, while their i-complexity values were under 1 bit across the board.<sup>3</sup> A simulation analysis performed on one of the languages exhibiting high e-complexity (Chiquihuitlàn Mazatec) showed that the i-complexity of the actual paradigm was lower than the i-complexity values for random permutations of that paradigm. This suggests that the inflectional paradigms of natural languages may be organized in such a way as to minimize their i-complexity. How might this come about? One possibility is that low i-complexity facilitates solving the Paradigm Cell Filling Problem (Ackerman *et al.* 2009; Ackerman and Malouf 2015), i.e. it makes it easier to determine the correct form for novel inflection. This generalization-based mechanism could lead to lower i-complexity: assuming individuals are frequently required to produce novel inflections (i.e. generate the inflectional form associated with grammatical function  $Y$  for a lexeme which they have only seen inflected for grammatical function  $X$ ), and

---

<sup>3</sup>The relationship between e-complexity and i-complexity found by Ackerman and Malouf (2013) is also reported in Cotterell *et al.* (2019), using different measures of both e- and i-complexity (the latter based on forms drawn from corpora rather than paradigms posited by linguists, cf. Bonami and Beniamine 2016; Sims and Parker 2016).

assuming they exploit predictive relationships between grammatical functions as captured by i-complexity, paradigms with low i-complexity will be relatively stable whereas paradigms with high i-complexity (i.e. where prediction from the form for function *X* to the form for function *Y* is not possible) will tend to change. Specifically, they might be expected to change in ways which reduce i-complexity since learners might actually introduce errors which reflect predictive relationships when attempting to generalise.

Seyfarth *et al.* (2014) tested the Ackerman *et al.* (2009) hypothesis that i-complexity has an effect on the ability of human learners to solve the Paradigm Cell Filling Problem. They compared the ability of human learners to predict novel inflected forms in low vs. high i-complexity input. They trained participants on an artificially constructed nominal inflectional paradigm in which nouns were marked for three grammatical numbers (singular, dual and plural) according to one of two noun classes (Table 2a). In the test phase, they asked participants to generate inflected forms for a novel lexeme given that lexeme’s inflected form in another grammatical number. In some trials, the required form could be predicted from the given form (predictive trials), while in others it could not be (non-predictive trials). In Table 2a, for example, being prompted with a novel singular form marked with *-yez* allows the learner to predict what form the lexeme

(a) Paradigm with two noun classes  
(their Experiment 1)

	Singular	Dual	Plural
noun class 1	-yez	-cav	-lem
noun class 2	-taf	-guk	-lem

Table 2:  
Artificially constructed  
nominal inflection paradigms  
used in Seyfarth *et al.* (2014)

(b) Paradigm with three noun classes  
(their Experiment 2)

	Singular	Dual	Plural
noun class 1	-taf	-guk	-lem
noun class 2	-yez	-cav	-lem
noun class 3	-yez	-cav	-nup

takes in the dual (-*cav*). However, knowing the form in plural is not predictive of the form in dual. They found that participants' performance differed across predictive and non-predictive trials, showing that learners were indeed able to use the predictive structure to generate a correct novel form when it was available. In a second experiment, Seyfarth *et al.* (2014) tested whether predictive information facilitated generalization to novel stems in a larger paradigm (Table 2b). They found that learners made less use of predictive information in this larger paradigm: learners tended to inflect novel stems with the most frequent marker (e.g. they used the suffix -*cav* to mark dual regardless of class). Notably, while predictive relations between forms in the paradigm is captured by *i*-complexity, suffix frequency is captured by our measure of *e*-complexity. Therefore, these results suggest that *e*-complexity may also influence how learners generalize to novel forms.

The Seyfarth *et al.* (2014) study simulates a case in which language learners have to generalize from the paradigm they have learned to inflect a novel stem for one grammatical feature based on exposure to that stem inflected for a different grammatical feature. For example, they might be required to inflect a stem for dual when they had only seen that stem inflected in the singular. They show that, in this case, learners are indeed able to use this predictive structure to predict the novel form. Johnson *et al.* (2020) replicate these results with LSTM networks, showing that the networks are able to use the predictive relations between forms in the paradigm to generalize to novel wordforms. However, generalizing to completely novel forms is an extreme case of a much more general problem that language learners face. In addition to generalizing to completely novel forms, learners must generate (or retrieve) forms which may have been encountered but have not yet been robustly acquired. Our hypothesis is that if low *i*-complexity facilitates solving the Paradigm Cell Filling Problem, i.e. using familiar forms to predict new forms, it should, in principle, facilitate learning forms under low exposure as well; learners can use the same strategy they use when generalizing to completely novel stems to help generate (or retrieve) low frequency forms that are not fully memorized.

Here we test this hypothesis, comparing the effects of *e*- and *i*-complexity on the learnability of morphological paradigms. We systematically manipulate *i*-complexity and *e*-complexity, holding other



potential differences among paradigms (e.g. number of forms) constant. In Section 2, we use an artificial language learning task to train and test LSTM neural networks and human participants on four inflectional paradigms with varying values of i- and e-complexity. To test the effect of i-complexity on speed and final attainment of learning, we test how well LSTMs and human learners are able to generate forms they are trained on over the course of learning. Data from these experiments, in combination with results from Seyfarth *et al.* (2014), will provide evidence concerning the mechanism by which i-complexity might shape paradigms over time. Specifically, whether the pressure for low i-complexity suggested by Ackerman and Malouf (2013, 2015) comes solely from how it affects generalization to novel forms, or from a more general facilitatory effect on learning, including retrieval of encountered forms. Moreover, comparing the effects of e- and i-complexity on learning will potentially provide corroborating evidence for the hypothesis that i-complexity rather than e-complexity shapes morphological paradigms. To preview, we find that the LSTM neural networks exhibit different learning rates for paradigms with different values of i-complexity, however the effect of variations in e-complexity is larger. Results from the task with human learners reveal an effect of e-complexity but not i-complexity on learning.

TESTING THE EFFECTS  
OF E- AND I-COMPLEXITY  
IN HUMAN LEARNERS  
AND LSTM NEURAL NETWORKS

2

Johnson *et al.* (2020) report a series of artificial language learning experiments with human learners and Long Short Term Memory (LSTM, Hochreiter and Schmidhuber 1997) neural networks. Learners and networks were trained on one of two nominal inflectional paradigms which were matched in e-complexity but differed in i-complexity: one with low i-complexity and one with high(er) i-complexity. They found evidence that the low i-complexity paradigm was learned faster by

LSTMs, but there was no clear effect of i-complexity for human learners. In a second series of experiments, manipulating both e- and i-complexity, e-complexity was shown to better predict learnability for both LSTMs and human learners. However, in Johnson *et al.* (2020), learning was staged, i.e. learners were first exposed to all forms in one grammatical function (singular), then forms in a second grammatical function were added (singular and plural), and finally forms in the last grammatical function were added (singular, plural, and dual). This was done to increase the chances of finding an effect of i-complexity; in low i-complexity paradigms, the dual forms could be predicted from the singular. Here, we explore more realistic, unstaged learning: presentation of forms is fully random and learners are exposed to all forms in the paradigm from the beginning. In contrast to Johnson *et al.* (2020), we also measure the overall accuracy of learning all inflected forms in the paradigm, rather than focusing only on learning of forms in one grammatical number. Replicating these results with unstaged learning is important, since our objective is to compare different types of complexity and their effects on learning. The learning regime should therefore be neutral in terms of enhancing or reducing the probability that learners would be affected by one measure or another. Furthermore, we take this as a starting point to investigate a wider range of differences in e- and i-complexity across paradigms, and therefore the privileged role of one specific portion of the paradigm (e.g. the singular in the staged learning design) will not hold across these more diverse paradigms.

Artificial language learning tasks allow us to create languages that differ only in the aspect we are interested in testing, while controlling for all other aspects of the language. This allows us to test the effects of i- and e-complexity on learning without confounds from other aspects of the paradigm and language such as the size of the paradigm, number of unique forms and number of words in each noun class. Another advantage of artificial languages paradigms is that since they are small compared to natural languages, they can generally be learned to a reasonably high proficiency over the course of a single short session. While they do not reflect the full complexity of natural languages learned in natural settings, artificial language paradigms are widely used in research on language acquisition, including to investigate learning biases (e.g. Wonnacott and Newport 2005; Hudson Kam and

Newport 2009; Moreton and Pater 2012; Fedzechkina *et al.* 2012, and many others). Moreover, studies using artificial learning paradigms show correspondence between lab-based learning biases and typology (e.g. see for reviews Culbertson *et al.* 2012; Culbertson and Newport 2015).

We use LSTM networks as a supplement to human learners as an additional means of testing the relative impact of i-complexity and e-complexity on paradigm learning. LSTM networks are powerful learning devices, and various recent studies show that they can be capable of extracting and using relevant linguistic information in sequence processing tasks. For example, Linzen *et al.* (2016) show that LSTM networks can in some cases predict long-distance subject-verb number agreement, in the presence of other potential agreement triggers (often called attractors) intervening between the subject and verb; Gulordava *et al.* (2018) show that LSTMs trained on four different languages can often accurately predict subject-verb agreement even when they are not trained specifically on that task; Futrell *et al.* (2019) show that surprisal scores of LSTMs (a measure of processing cost) paralleled preferences of human participants on grammatical judgments task differentiating word-order alternations.

Here, we use LSTMs as a convenient ‘ideal learner’, to provide evidence that i-complexity can in principle influence paradigm learnability for at least one learning model. This is particularly useful in circumstances where (as turns out to be the case here) human data provides little evidence of an effect of i-complexity. The LSTM models allow us to show that this is not an intrinsic limitation to the way in which we set up our learning task – we find that i-complexity does influence learning in LSTMs trained on the same paradigms. Crucially, we can then show that, even for a class of learners sensitive to i-complexity, those effects are smaller than the effects of e-complexity. Finally, directly comparing performance of LSTMs and humans on a matched task opens up the possibility that, to the extent that they show similar patterns of performance, LSTMs could be used as a convenient tool to quickly generate predictions to be tested in further human experiments on paradigm learning. In other words, if these models reliably produce a similar pattern of results to human learners then they could potentially also be used to extrapolate to paradigms that are hard to test with human learners under controlled circumstances, e.g. learn-

ing of very large paradigms or paradigms inflecting over very large lexicons.

## 2.1

*Target paradigms*

We use four artificially constructed inflectional paradigms, similar in size and design to the ones used in Seyfarth *et al.* (2014) and Johnson *et al.* (2020). The same paradigms were used for both neural networks and human participants. The paradigms consist of nine CVC nouns (*gob, tug, sov, kut, pid, tal, dar, ler, mip*), randomly paired with meanings for human participants (see Section 2.3 below). The nouns were randomly allocated to three classes (for each run of the network, or each human participant), and each class was inflected for three numbers: singular, dual and plural. Inflectional markers were seven VC monosyllabic suffixes (*-op, -oc, -um, -ib, -el, -ek, -at*). These inflectional markers were randomly allocated to cells in each paradigm, such that the four paradigms were always structured as in Table 3 below but with a different mapping of affixes to cells for each human participant.

As summarized in Table 3, the paradigms differ either in i-complexity or e-complexity, holding the other constant. We also hold constant all other aspects of the paradigms: the paradigms are matched in

Table 3: Four target paradigms differing either in i-complexity or e-complexity values. The low i-complexity, low e-complexity (low-i/low-e) and high i-complexity, low e-complexity (high-i/low-e) paradigms differ in i-complexity only. The two remaining low-i/high-e paradigms have low i-complexity but have higher e-complexity; these paradigms also differ in the type of syncretism pattern (within class or across class)

		e-complexity	
		Low (1.141 bits)	High (1.363 bits)
i-complexity	Low (0.222 bits)	low-i/low-e	low-i/high- $e_{within}$ low-i/high- $e_{across}$
	High (0.444 bits)	high-i/low-e	

Effects of *i-* and *e-* complexity on morphological learning

Table 4: Example paradigms for each type tested. See Table 3 for high-level descriptions of each type. Colored cells highlight distinct paradigm structures: in low-*i*/low-*e* (a), singular *-op* predicts dual *-um*; in high-*i*/low-*e* (b), singular does not predict dual; in both low-*i*/high-*e* paradigms (c,d), the singular form which occurs most frequently is reused for plural elsewhere in the paradigm (syncretism) – either in one of the classes with that form in the singular (c low-*i*/high-*e*<sub>within</sub>), or in a different class (d low-*i*/high-*e*<sub>across</sub>)

(a) low- <i>i</i> /low- <i>e</i>				(b) high- <i>i</i> /low- <i>e</i>			
	Singular	Dual	Plural		Singular	Dual	Plural
noun class 1	-op	-um	-ib	noun class 1	-op	-um	-ib
noun class 2	-at	-oc	-el	noun class 2	-at	-um	-el
noun class 3	-op	-um	-ek	noun class 3	-op	-oc	-ek

(c) low- <i>i</i> /high- <i>e</i> <sub>within</sub>				(d) low- <i>i</i> /high- <i>e</i> <sub>across</sub>			
	Singular	Dual	Plural		Singular	Dual	Plural
noun class 1	-op	-um	-op	noun class 1	-op	-um	-el
noun class 2	-at	-ib	-el	noun class 2	-at	-ib	-op
noun class 3	-op	-oc	-ek	noun class 3	-op	-oc	-ek

terms of number of distinct affixes and number of inflectional classes, and they feature the same three-way number distinction. The low *i*-complexity, low *e*-complexity (low-*i*/low-*e*) and high *i*-complexity, low *e*-complexity (high-*i*/low-*e*) paradigms differ in their *i*-complexity (0.222 vs. 0.444 bits) while their *e*-complexity is kept constant (1.141 bits). The key difference between the two paradigms is that in the low-*i*/low-*e* paradigm, knowing the singular affix of a word (e.g. *-op* in Table 4a), predicts the dual affix (e.g. *-um*). This is not the case in the high-*i*/low-*e* paradigm (in Table 4b the singular *-op* does not uniquely determine the form of the dual). The remaining two paradigms (Table 4c, d) both have low *i*-complexity (0.222 bits) but higher *e*-complexity (1.363 bits). In general, higher *e*-complexity here means having distinct dual forms for each class, which results in higher uncertainty across forms relative to the low *e*-complexity paradigms. *I*-complexity is kept constant and low in these two paradigms since both the plural and dual forms are predictive of each other as well as

the forms in singular. However, increasing e-complexity while keeping the number of markers constant requires *syncretism* in the paradigm; a single affix is used to mark different grammatical functions. In order to additionally explore how syncretism affects learning, here we generated two different syncretism patterns: within class syncretism (low-i/high- $e_{within}$ ) and across class syncretism (low-i/high- $e_{across}$ ). In both low-i/high-e paradigms, the singular form is the same for classes 1 and 3 (e.g. -op in the example paradigm in Table 4c, d). In the low-i/high- $e_{within}$  the syncretic form is reused as a plural in class 1 (Table 4c). In the low-i/high- $e_{across}$  the syncretic form is reused as a plural marker for class 2 (Table 4d), crucially, not one of the classes which use this form in the singular. Previous work on morphological paradigms suggests that this difference in syncretism type could affect learning in human learners (e.g. Baerman *et al.* 2005; Pertsova 2012; Maldonado and Culbertson 2019), therefore we test both paradigm types.

Note that we do not include a paradigm with high i-complexity *and* high e-complexity. This is not actually possible: there is no way to distribute markers such that both measures of complexity are high without changing the number of markers in the paradigm. We discuss this further below.

As mentioned above, in Johnson *et al.* (2020), exposure to forms from a paradigm was *staged*: input initially contained only singular forms, then singular and plural forms, then singular, plural, and dual forms. This was designed to highlight the implicative structure of low i-complexity paradigms. However, it is also rather unrealistic in that exposure in natural language is unlikely to be staged in this way, or at least not so rigidly staged. Here, we expose learners to forms drawn at random from the entire paradigm. Therefore, we test whether having low vs. high values of i- or e-complexity is beneficial when learners have not always learned predictive forms first. We compared speed and accuracy of learning all forms in the language across all four conditions.

## 2.2

### *Experiment 1: LSTM neural networks*

Neural networks are computational models which approximate a function linking the network's input with its desired output. The

model consists of several layers of nodes interconnected by associative weights. Given a dataset of input-output pairs, the model tries to learn the optimal setting of these weights to correctly transform an input into its corresponding output. Updating the weights to better approximate the input-output function is done by searching for weights that minimize the *loss function* of the network, which measures how close the network's output is to the true output. Different algorithms are used for this search. A common algorithm is (*stochastic*) *gradient descent*. Intuitively, the network generates an output through a forward pass from the input layer to the output layer, after which the loss function calculates the difference between the predicted and the target values. Then, in a backward pass, the loss function is used to compute an error gradient with respect to each weight and the network's weights are updated in the direction of the greatest descent so as to reduce this error.

*Recurrent* neural networks (RNNs) overcome a limitation of simple neural networks fundamental to language tasks; simple neural networks are not sensitive to the 'context' of the current input or, in other words, how previous inputs affect the correct output for the current input. RNNs overcome this limitation by having 'short term memory' through looping back the output or hidden layer activations previously produced for earlier inputs (Elman 1990; Jordan 1997; Elman 1991). This allows the network to adjust the output for the current input according to previous inputs. The extent to which previous states of the network affect the current state is also determined by weights updated through the backpropagation process.

*Long Short Term Memory* (LSTM) networks are an extension of recurrent neural networks introduced by Hochreiter and Schmidhuber (1997) in order to improve learning of longer temporal dependencies. Practically, LSTMs add an element of 'long term memory' to networks by allowing the network to control the influence of current and previous inputs during the process of activation propagation, using 'weighted gates' in the networks. Like activation weights, these gates are optimized during training to determine what information is stored or passed along and therefore allowed to influence subsequent inputs. This allows LSTMs to make better use of sequential information, including learning non-adjacent dependencies.

LSTMs therefore offer a powerful but convenient general-purpose learning mechanism for language based tasks. Here we use LSTMs to process relatively short sequences: networks are presented with stems and grammatical features and produce an inflectional affix, and we train models on the target paradigms which differ in either their i-complexity or e-complexity.

### 2.2.1

#### Network structure

We trained and tested LSTM networks using the Keras package in Python (Chollet *et al.* 2015). In this task, the model gets as input a sequence containing the noun’s stem and an extra character indicating the grammatical number of the object (1 for singular, 2 for dual and 3 for plural). For example, the string *mip3* indicates the noun with the stem *mip* in plural. The model’s task is to output the correct affix for this wordform, according to the paradigm it is trained on. An overview of the network structure is given in Figure 1. The network has 7 output units, one for each of the 7 affixes in the target paradigms. Input stem + number sequences are encoded as one-hot vectors. i.e. every character used in the language is represented as a vector of zeroes (with length equal to the total set of characters, 27) with ‘1’ in a different index uniquely identifying it. We trained the model with a range of embedding vectors dimensionalities for the input layer and LSTM hidden layer dimensionalities (from 5-dimensional embedding vectors and 5-unit layer (542 parameters) to 50 (14,657 parameters), with increases of 5 units). The state of the LSTM at the end of the input string is fed into a ‘softmax’ function to produce a one-hot encoding representing the output affix for this stem + number input (i.e. the network’s task it to learn a 7-way categorical classification of the input sequences). The network was optimized using Stochastic Gradient Descent (SGD) with learning rate of 0.1, batch size of 32, and no dropout.<sup>4</sup> Initial weights were randomly generated, according to a

---

<sup>4</sup>In addition to the various network sizes reported in the main paper, we also ran variants of the model with a range of learning rates, using both SGD and Adam (Kingma and Ba 2014) optimizers. Detailed results are presented in the Appendix. Note that the overall conclusions discussed in the main text remain unchanged across these variants.



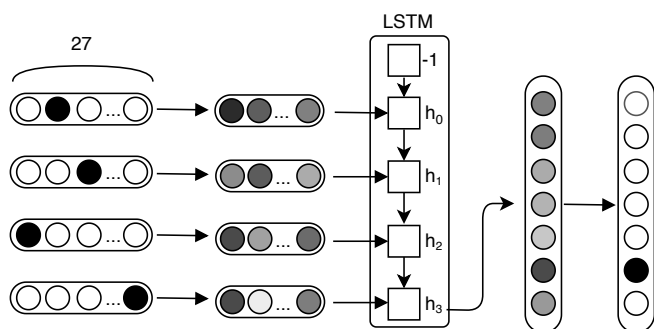


Figure 1: A diagram of the recurrent neural network: the input layer receives a string of four characters (stem + grammatical number), each coded as a one-hot vector of the length of the different characters used in the language (27). The input vectors are embedded and the embeddings are transferred to a hidden layer with 5–50 LSTM units. Output from the LSTM units ( $h_3$ ) is then transferred to an output layer with seven options, representing the seven suffixes in the language. Using a softmax function, the output is converted to a one-hot vector, representing the suffix the network selected for this input

‘glorot\_uniform’ function (sampling from a uniform distribution in the range of  $[-x, +x]$ , where  $x$  is a function of the size of the network).

For each paradigm and set of hyperparameters, 50 runs were produced. In each run, the lexical items were randomly assigned to noun classes and the model was trained and tested on input-output pairs across 900 epochs. In each epoch, the network is trained and tested on all 27 wordforms in the language (9 stems marked for singular, dual and plural). The test set in this task is identical to the training set – we are not testing the capacity of the network to generalize, but rather the overall accuracy and speed with which it learns the mapping from stem + number input to the appropriate affix output.<sup>5</sup>

## Results

### 2.2.2

We measured the average accuracy of the networks in producing the correct affix for all wordforms in the target paradigm over epochs (averaged over 50 runs for each combination of target paradigm and

<sup>5</sup> As discussed above, this task differs from that used in Seyfarth *et al.* (2014), who focus on generalizing to unknown forms.

network size). For simplicity, we first collapse the two low-i/high-e paradigms in these graphs, and deal with the effect of syncretism separately below. Figure 2 presents network learning trajectories for these three paradigm types.

The same trend is seen across different network sizes. While 900 epochs is sufficient for all paradigms to be learned perfectly, even for the smallest networks, the low-i/low-e paradigm type is learned most rapidly. Networks trained on the high-i/low-e paradigm type show a similar but slightly slower learning trajectory. Networks trained on the low-i/high-e paradigm types show the slowest learning, with accuracy increasing markedly later in training than the other paradigms.<sup>6</sup>

Since we are interested in the effect of i- and e-complexity on the difficulty of learning the paradigm, rather than whether the language is eventually learnable or not (all of our paradigms were eventually learned with 100% accuracy given sufficient training), we compare the *summed accuracy* (i.e. the sum of the epoch-by-epoch accuracies) of the networks trained on the different languages. The summed accuracy reflects both the speed of learning the language and the accuracy throughout learning. For example, in the results shown in Figure 2, where all networks eventually reach ceiling, networks which learn more rapidly will have a higher summed accuracy reflecting the faster pick-up in accuracy over epochs. Other measures of learning speed are possible, e.g. the mean number of epochs to reach 100% accuracy; we prefer mean summed accuracy because it relates more obviously to the different shapes of the curve we see in Figure 2, and is still interpretable for network parameterisations that do not result in convergence to 100% accuracy.

---

<sup>6</sup>We looked at the errors made by the LSTMs at epochs 1–150 (when the neural networks show a plateau in learning). At this stage in learning, the networks use only two out of the seven possible affixes as an output. This likely reflects a local minimum in the loss function, meaning that the LSTM ‘found’ a partial solution that maximizes its output accuracy. Each input string is classified with one of those two affixes solely according to the number indicating the grammatical number at the end of the input string so that all singulars take one affix (one of the affixes that mark singular), and all dual and plural inputs are marked with another affix (one of the affixes that mark either dual or plural). After around 150 epochs, the networks start using additional affixes, which is then reflected by a jump in performance.

*Effects of i- and e-complexity on morphological learning*

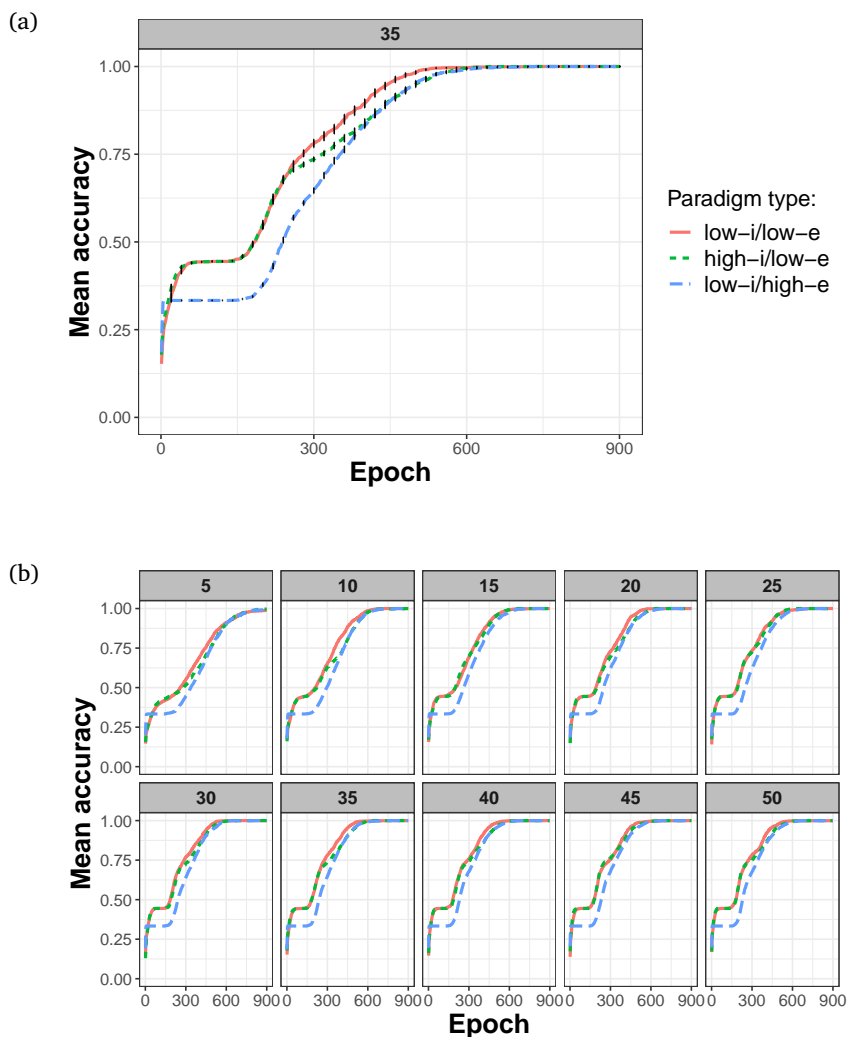


Figure 2: Network learning trajectories. (a) results for one network size (35 cells), with error bars indicating standard error every 10 epochs, (b) results for all network sizes tested (facet titles give network size in number of cells). Networks trained on low-i/low-e and high-i/low-e paradigm types show similar learning trajectories, while networks trained on low-i/high-e paradigms show lower accuracy levels. Results from models with further learning rates for both SGD and Adam optimizers show similar patterns for most cases, and we never see the opposite trend of lower accuracies for the high-i/low-e condition (see the Appendix for detailed results)

Figure 3:  
Summed accuracy over the 900 epochs of the networks trained on each of the three paradigm types across different sizes of the network. Error bars represent standard error. Note that the two low-i/high-e paradigms are collapsed here

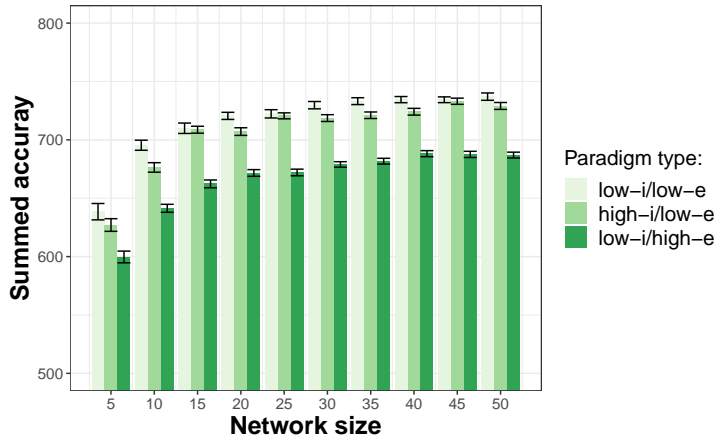


Figure 3 shows the summed accuracy of the networks trained on each paradigm type across different network sizes. To determine whether these differences between network learning trajectories are significant, we ran a linear mixed-effect regression model<sup>7</sup> predicting the summed accuracy of the network across all epochs based on fixed effects of paradigm type (low-i/low-e, high-i/low-e, low-i/high-e), size of the network, and their interaction. In addition to these fixed effects, we also included random intercepts for each run of a network. Network size was mean centred. Paradigm type was Helmert-coded to test our predictions about the relative levels of accuracy across paradigms. Based on results from Johnson *et al.* (2020) we predict low-i/low-e to be the easiest, therefore this was set as the baseline. The model compares the baseline to the next level, high-i/low-e, then the mean of these two levels is compared to the third level, low-i/high-e. The first contrast, therefore, tests the effect of i-complexity and the second tests the effect of e-complexity. The model revealed a significant effect of network size on summed accuracy ( $\beta = 1.63$ ,  $sd = 0.049$ ,  $t = 32.83$ ,  $p < 0.001$ ), suggesting that larger networks learn the languages faster. Critically, the model also revealed a significant effect of both i-complexity ( $\beta = -4.48$ ,  $sd = 0.9$ ,  $t = -4.68$ ,  $p < 0.001$ ) and e-complexity ( $\beta = -10.61$ ,  $sd = 0.52$ ,  $t = -20.23$ ,  $p < 0.001$ )

<sup>7</sup>All models reported here were run using the lme4 (Bates *et al.* 2014) and lmerTest (Kuznetsova *et al.* 2017) packages in R.

on summed accuracy. These results suggest that measures of paradigm complexity based on implicative structure (i-complexity) and on number and distribution of forms (e-complexity) both impact ease of learning in LSTM neural networks. Note that while both effects are significant, the estimated effect size for the effect of e-complexity is larger than the estimate effect of i-complexity, suggesting the e-complexity manipulation had a larger effect than our i-complexity manipulation; this difference in effect sizes can be seen in the timecourses in Figure 2 and in Figure 3.

### Type of Syncretism

### 2.2.3

Recall that we included two types of low-i/high-e paradigms: one in which syncretism was within class, and one where it was across class (see Table 4). In general, cross-class syncretism can affect both i-complexity and e-complexity, but for our paradigms neither i-complexity nor e-complexity distinguish between syncretism types; the two paradigm types have the same values for both measures. Figure 4 shows network learning trajectories with these two paradigm types plotted separately. Across different network sizes, the paradigm type with cross-class syncretism appears to be learned slower, in line with previous work (e.g. Pertsova 2012; Maldonado and Culbertson 2019).

Summed accuracies of networks trained on low-i/high- $e_{within}$  and low-i/high- $e_{across}$  paradigms (averaged over the 50 runs of the model) across different network sizes are presented in Figure 5. We ran a linear mixed-effect regression model predicting summed accuracy by paradigm type (within-class syncretism vs. across-class syncretism), network size and their interaction. In addition to these fixed effects, the model included random intercepts for each run of a network. Paradigm type was dummy coded, with within-class syncretism coded as the reference group. Network size was mean centred. The model revealed a significant effect for the network size, increasing the learning accuracy for larger neural networks ( $\beta = 1.45$ ,  $sd = 0.09$ ,  $t = 15.9$ ,  $p < 0.001$ ). Critically, the model also revealed a significant effect of paradigm type ( $\beta = -34.37$ ,  $sd = 1.84$ ,  $t = -18.62$ ,  $p < 0.001$ ), suggesting that paradigms with across-class syncretism are learned slower by the neural networks.

Since the type of syncretism was found to affect learning, we conducted an additional analysis to determine whether the effect of

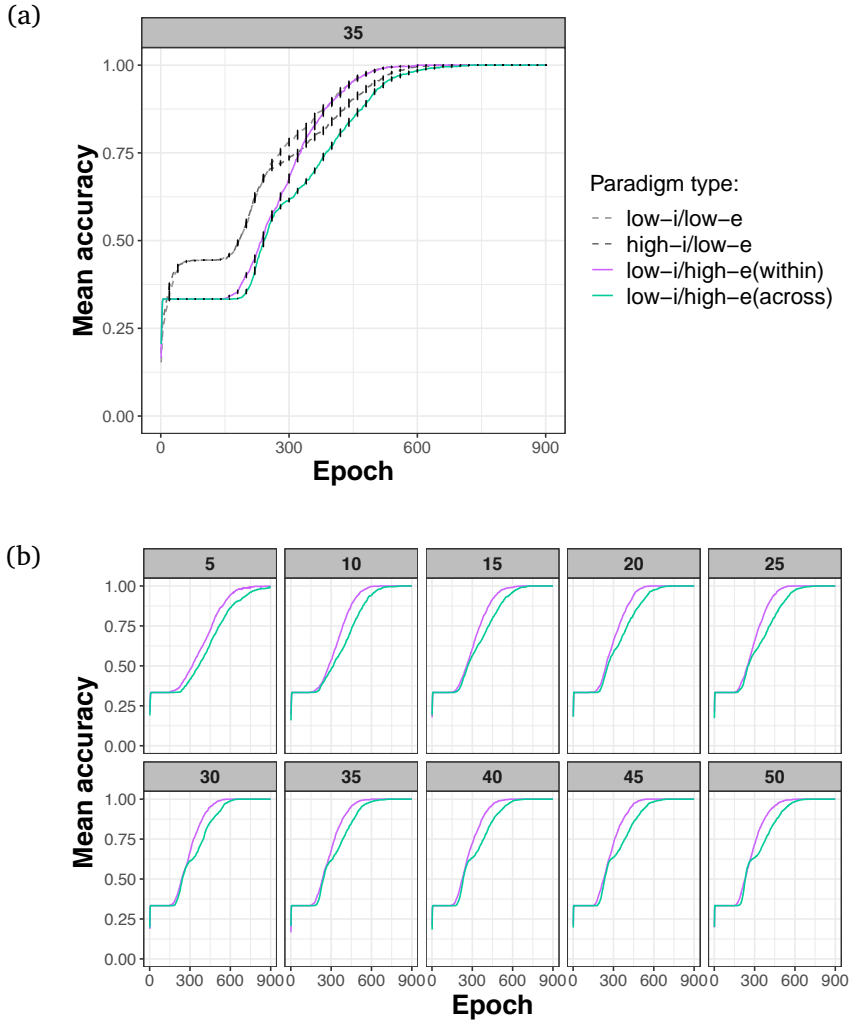


Figure 4: Network learning trajectories with low-i/high- $e_{within}$  and low-i/high- $e_{across}$  paradigms plotted separately. Trajectories for networks trained on low-i/low-e and high-i/low-e paradigms presented in grey (dashed lines) for comparison. (a) results for one network size (35 cells), with error bars indicating standard error every 10 epochs. (b) results for all network sizes tested (facet titles give network size in number of cells). Networks trained on paradigms with cross-class syncretism show slower learning

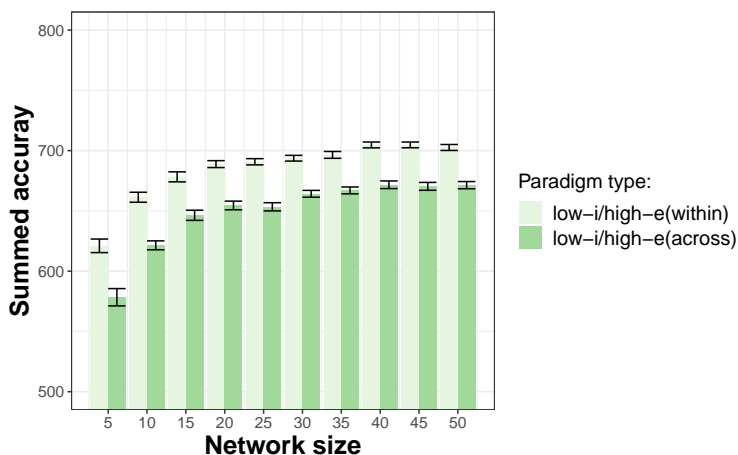


Figure 5: Summed accuracy over the 900 epochs of networks trained on low-*i*/high- $e_{within}$  and low-*i*/high- $e_{across}$  paradigms across different network sizes. Error bars represent standard error. Across all network sizes the paradigm type with across-class syncretism is learned slower

*e*-complexity was entirely driven by the low-*i*/high- $e_{across}$ , or whether this effect is found regardless of syncretism type. We ran a linear mixed-effect regression model predicting summed accuracy by paradigm type and network size (mean centred), with random effects as specified for previous models. Paradigm type was dummy coded with low-*i*/low-*e* as the reference group. The model revealed a significant effect of network size ( $\beta = 1.61$ ,  $sd = 0.09$ ,  $t = 17.25$ ,  $p < 0.001$ ). In addition, the model revealed a significant difference between low-*i*/low-*e* and both low-*i*/high-*e* paradigm types (low-*i*/high- $e_{within}$ :  $\beta = -31.3$ ,  $sd = 1.89$ ,  $t = -16.52$ ,  $p < 0.001$ , low-*i*/high- $e_{across}$ :  $\beta = -65.67$ ,  $sd = 1.89$ ,  $t = -34.67$ ,  $p < 0.001$ ). This confirms the generality of the effect of *e*-complexity on learning; regardless of the type of syncretism, paradigms with high *e*-complexity are learned more slowly than languages with low *e*-complexity, even when all other aspects of the paradigm (*i*-complexity, but also number of inflections, number of inflectional classes, etc.) are held constant. As before, there was also a significant difference between low-*i*/low-*e* and high-*i*/low-*e* ( $\beta = -8.96$ ,  $sd = 1.89$ ,  $t = -4.73$ ,  $p < 0.001$ ).

To summarize, here we trained LSTM neural networks on one of four nominal inflectional paradigms which differed in either *i*-complexity or *e*-complexity. The results of our simulation experiments showed that both measures of complexity affect learning in these networks, with more complex paradigms being learned more

slowly. We also found that type of syncretism mattered: networks more readily learned syncretic forms which targeted cells within a class rather than across class. These effects were not necessarily all of equal strength: effects of i-complexity were weaker than the effects of e-complexity and syncretism type. The effect size of e-complexity on the network's accuracy was four times larger than the effect of i-complexity (estimated  $\beta$  values of  $-31.3$  in the case of within-class syncretism and  $-65.67$  in the case of across-class syncretism vs.  $-8.96$  for the effect of increased i-complexity). In sum, our neural network simulations show that, in principle, i-complexity can affect learning morphological paradigms. This complements earlier results for human learners and LSTMs (Seyfarth *et al.* 2014; Johnson *et al.* 2020) showing that low i-complexity facilitates generalisation to novel forms. Importantly however, our results also provide evidence that e-complexity has a stronger effect on learning. In the next section, we turn to human learners. Johnson *et al.* (2020) found that i-complexity only weakly affected human learning, even in a staged paradigm intended to maximise the effects of i-complexity. Here we will compare the effects of i- and e-complexity to see whether indeed e-complexity plays a stronger role in determining ease of learning for humans when learning is not staged.

## 2.3

### *Experiment 2: human learners*

#### 2.3.1

##### Materials

The same artificially constructed paradigms described in Table 4 were used to train and test human participants. Participants were exposed to the word forms in the language together with meanings. Stems referred to a set of simple objects: lemon, cow, tomato, bicycle, horse, clock, pigeon, mug and pear. Visual stimuli were identical to those used in Johnson *et al.* (2020). Singular nouns corresponded to a single object, dual corresponded to two objects, and plural ranged from 3 to 12 objects (selected randomly). See Figure 6 for an example plural trial. Objects in the language were divided into the three noun classes so that every noun class had one animate object (cow/pigeon/horse), one edible object (tomato/lemon/pear) and one other (clock/bicycle/mug). This was done to ensure that noun class



membership could not be determined based solely on semantic features. All stems and markers were randomly assigned to meanings for each participant.

### Participants

2.3.2

144 self-reported native English speakers participants were recruited via Amazon's Mechanical Turk crowd-sourcing platform. They were compensated \$6 for their participation and the experiment lasted 53 minutes on average (min = 19, max = 166, mode = 41). We recruited participants who possessed an Mturk qualification indicating that they were based in the US. Participants were allocated randomly to each of the four paradigms. We excluded from the final dataset 22 participants who did not complete the experiment,<sup>8</sup> thus the final dataset consisted of 120 participants: low-i/low-e (29); high-i/low-e (31); low-i/high-*e<sub>within</sub>* (28); low- i/high-*e<sub>across</sub>* (31).

### Procedure

2.3.3

Participants learned the language via trial and error. On each trial, a picture (featuring 1–12 instances of a single object) was presented on the screen together with a set of possible labels, as in Figure 6. Participants were asked to choose the correct label after which they received feedback on their answer. If their answer was incorrect, they were presented with the correct form. The set of possible labels consisted of all combinations of the correct stem with all the suffixes in the paradigm. The task was divided into 3 identical blocks of 108 trials each: in every block, participants were exposed to all stems inflected in each of the three grammatical numbers (27 wordforms), 4 times each. The order of trials was randomized in each block. Participants were allowed a self-paced break between blocks; they were presented with a screen announcing the end of the block and were asked to click on 'continue' to complete the next block of trials. Participants' answers on each trial were recorded and their overall accuracy was measured to test the effects of i-complexity and e-complexity on paradigm learnability.

---

<sup>8</sup> Participants who did not complete the experiment and who contacted us were paid according to the proportion of trials they completed.

(a)

Score: 60, Trial: 62/108



- 
- 
- 
- 
- 
- 
- 

Which word matches the picture?

(b)

Score: 60, Trial: 62/108



- 
- 
- 
- 
- 
- 
- 

The correct word is **kutel**

(c)

Score: 80, Trial: 84/108



- 
- 
- 
- 
- 
- 

Well done!

Figure 6: Example plural trial. (a) A picture is presented and participants are asked to choose the correct label from a set of options. (b), (c) Participants receive feedback on their answer, including the correct label. (b) Negative feedback following trial shown in (a). (c) Positive feedback following plural trial with a different number of objects

Figure 7 shows learning trajectories for each paradigm type, here with low-*i*/high-*e* paradigm types (which differed in syncretism type) collapsed. Participants' learning trajectories are non-linear but less complex than the learning curves of the LSTMs and can be described using quadratic polynomial curves (as in Figure 7). Therefore, we used logistic growth curve analysis (Mirman 2017) to analyse the effect of *i*-complexity and *e*-complexity on learning over trials. The model predicted accuracy by paradigm type and trial number. In addition to these fixed effects, the model also included by-participant intercepts and random slopes for trial number. Paradigm type was Helmert-coded

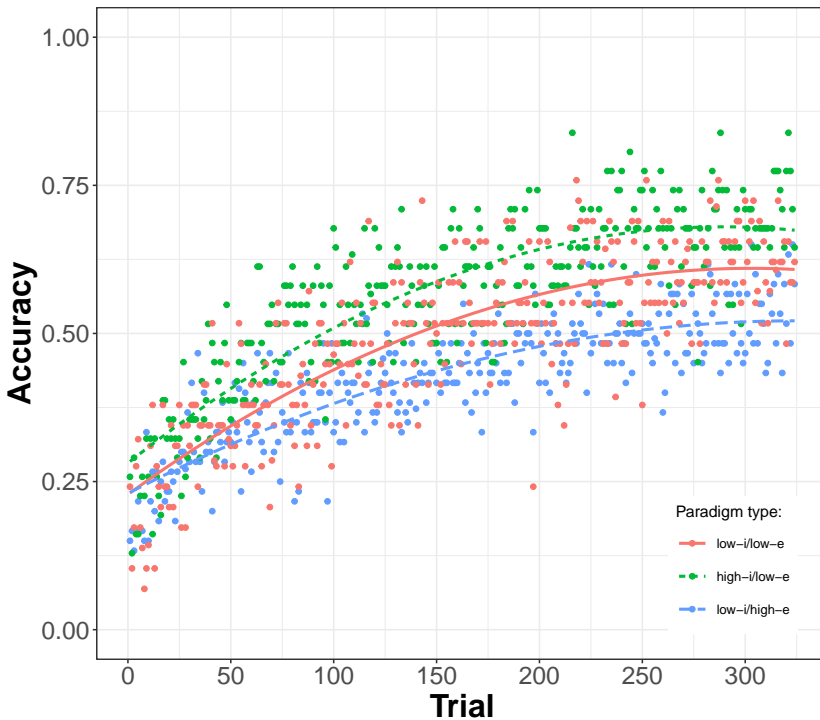


Figure 7: Mean accuracy by trial for each of the three paradigm types (collapsing the two low-*i*/high-*e* paradigms). Points indicate the average accuracy across participants for each trial. Lines show quadratic polynomial curves predicting accuracy by trial number for each paradigm type. Learning is worst for the low-*i*/high-*e* and best for the high-*i*/low-*e* paradigms

as in Experiment 1. Learning curves (accuracy over trials) were modelled with second-order orthogonal polynomials. The model revealed no significant effect of i-complexity ( $\beta = 0.2$ ,  $sd = 0.15$ ,  $z = 1.29$ ,  $p = 0.19$ ), but a significant effect of e-complexity ( $\beta = -0.16$ ,  $sd = 0.07$ ,  $z = -2.18$ ,  $p = 0.028$ ): participants trained on one of two low e-complexity paradigms learned better than participants trained on a high e-complexity paradigm. There was also a significant effect of trial in both the linear ( $\beta = 9.9$ ,  $sd = 0.87$ ,  $z = 11.3$ ,  $p < 0.001$ ) and quadratic ( $\beta = -2.23$ ,  $sd = 0.43$ ,  $z = -5.16$ ,  $p < 0.001$ ) terms, indicating that across trials, overall accuracy increased, but curves became less steep over time. These results provide clear evidence of the effect of e-complexity on human learning of inflectional paradigms. However, our results fail to show any effect of i-complexity. The data are noisy, but the numerical trend is in fact in the wrong direction – the high-i/low-e paradigm is learned numerically better than the low-i/low-e paradigm.

One plausible strategy, which would be consistent with the results showing an effect of e-complexity and no evidence for an effect of i-complexity, is simply to choose the most frequent form for each grammatical number, ignoring class membership for each stem. This strategy would result in higher accuracy in the low e-complexity conditions (where there is a frequent form for both the singular and the dual, see Table 4) but would yield lower accuracy in the high e-complexity conditions (where there is a frequent form in singular only). However, a closer look at our participants' responses, and the rates with which they chose the frequent form for each grammatical number, show that this is probably not the case; participants (as a group) do not choose the frequent form for a specific number more than its actual probability with which it appears (66% of the trials with this grammatical number). Participants in the low-i/low-e condition on average chose the frequent form of a grammatical number in 64.9% of the relevant trials, and participants in the high-i/low-e condition chose the frequent form of a grammatical number in 66.5% of the relevant trials. These results suggest that participants are probability matching (e.g. Hudson Kam and Newport 2005, 2009); participants match the probability of the form in their responses to its actual probability in the language rather than simply choosing the most frequent form for each grammatical number. Therefore, there is an advantage to the

skewed distribution of forms in low e-complexity paradigms that facilitates learning the paradigm even if participants do not simply select the most frequent form.

Type of syncretism

2.3.5

As with the LSTMs, we further tested whether there was a difference in learning for the two paradigms differing in syncretism type. We ran a separate logistic growth curve model predicting accuracy by paradigm type (within-class syncretism vs. across-class syncretism, sum coded) and trial number, with by-participant intercepts and random slopes for trial number. Here as well, learning curves (accuracy over trials) were modelled with second-order orthogonal polynomials. The model revealed no significant effect of syncretism type ( $\beta = -0.019$ ,  $sd = 0.15$ ,  $z = -0.127$ ,  $p = 0.89$ ). As before, the model revealed a significant effect of trial in both the linear ( $\beta = 8.06$ ,  $sd = 1.19$ ,  $z = 6.9$ ,  $p < 0.001$ ) and quadratic ( $\beta = 8.06$ ,  $sd = 1.19$ ,  $z = 6.9$ ,  $p < 0.001$ ) terms, indicating that across trials, overall accuracy increased, but curves became less steep over time. The results do not provide any evidence for differences in learnability of morphological paradigms with across-class as compared to within-class syncretism in human learners. There is therefore no reason to suspect that the effect found above of e-complexity in human learners is driven by differences in learnability across types of syncretism.

EXPLORING THE RELATIONSHIP  
BETWEEN I- AND E-COMPLEXITY  
WITH RANDOM PARADIGMS

3

Results from simulations with LSTM neural networks and behavioural experiments with human learners both suggest that e-complexity has a robust effect on learning of inflectional paradigms. By contrast, the effect of i-complexity was present but weaker in neural networks and absent in human learners. This suggests that i-complexity is not the primary determinant of learnability – e-complexity, at least how we have measured it here, has a much larger impact on how well learners

are able to generate (or retrieve) forms they have been exposed to. It may be that the beneficial effects of low i-complexity largely derive from its facilitating effect on generalisation (as suggested by Ackerman and Malouf 2015).

Ackerman and Malouf's (2013) Low I-complexity Conjecture for natural languages is based on the observation that, across a sample of natural languages, a relatively wide range of e-complexity values was found, but the range of i-complexity values was much more narrow. From this Ackerman and Malouf (2013) concluded that e-complexity in natural morphological paradigms is relatively free to vary and can be high as long as i-complexity stays low. However, as we have already mentioned, these two measures are not independent of one another: it was not possible for us to construct a paradigm with both high e-complexity and high i-complexity (while keeping the number of forms constant). In this section we explore the relationship between i- and e-complexity by looking at their values across 1000 randomly generated paradigms. To preview, we find an inverse correlation between i- and e-complexity which is in line with the pattern Ackerman and Malouf (2013) observe. This suggests that the Low I-complexity Conjecture is not necessarily a result of language change, i.e. it may not be driven purely from usage errors or learnability pressure. We also test the learnability of this set of 1000 paradigms with LSTM neural networks to show how these two measures relate to learning across a wider range of paradigms than we covered in Experiments 1 and 2.

### 3.1

#### *Generating random paradigms*

We generated 1000 random inflectional paradigms expressing the same three grammatical numbers (singular, dual and plural) across three noun classes, as in the paradigms tested above. The paradigms were generated by randomly assigning affixes to the nine cells with replacement, i.e. allowing affixes to repeat. Therefore, paradigms also vary randomly in number of unique affixes. Generated paradigms had between three and eight affixes, with most paradigms (42%) including six unique affixes. For each randomly generated paradigm, we calculated i- and e-complexity. I-complexity varied between 0 and 0.667

bits with a mean value of 0.201 bits. E-complexity varied between 0.528 and 1.585 bits with a mean value of 1.36 bits.

*Quantifying the relationship  
between i- and e-complexity in random paradigms*

3.2

We first explored the relationship between these three dimensions of variation (*i*-complexity, *e*-complexity, number of distinct affixes) in the 1000 randomly generated paradigms. Figure 8 shows the distribution of *i*-complexity and *e*-complexity values across paradigms, with average number of markers indicated by color. As suggested by the figure, *i*-complexity is strongly negatively correlated with *e*-complexity ( $r = -0.92$ ,  $t(998) = -73.8$ ,  $p < 0.001$ ). In other words, paradigms with high *i*-complexity tend to have low *e*-complexity, and vice versa. To explore the relationship between these complexity measures and the number of the unique affixes in the paradigm, we ran additional correlation tests. While *e*-complexity is positively correlated with the number of markers in the paradigm, ( $r = 0.44$ ,  $t(998) = 15.62$ ,  $p < 0.001$ ), *i*-complexity is negatively correlated with it ( $r = -0.38$ ,

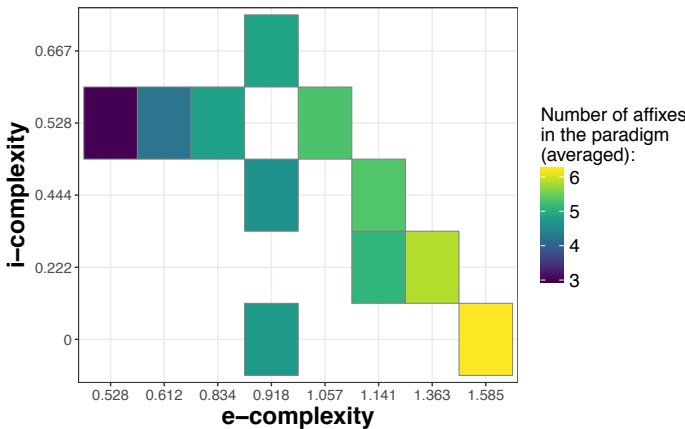


Figure 8: Distribution of randomly generated paradigms in terms of *i*- and *e*-complexity. Colour represents the average number of markers for paradigms with specific *i*- and *e*-complexity values. No paradigms have high *i*-complexity *and* high *e*-complexity. Paradigms with high *i*-complexity and low *e*-complexity have on average fewer markers while paradigms with low *i*-complexity and high *e*-complexity have more

$t(998) = -13.1, p < 0.001$ ): as the number of distinct forms increases, the implicative structure between forms increases. For example, if every cell in the paradigm is expressed by a unique form, then each form will perfectly predict every other form.

Since both i-complexity and e-complexity correlate with the number of markers in the paradigm, we further analysed the subset of random paradigms with the most frequently generated number of markers (six). We tested the relationship between i-complexity and e-complexity for these paradigms (423 paradigms), again confirming the negative correlation ( $r = -0.94, t(421) = -59.24, p < 0.001$ ). Table 5 presents two randomly-generated example paradigms with six markers which illustrates how the negative correlation between i-complexity and e-complexity arises from the organization of markers in the paradigm, even when the number of markers in the paradigm is held constant. Paradigms in which a grammatical function is marked with the same marker across inflection classes tend to have lower e-complexity (there is a more frequent form marking this grammatical function) and higher i-complexity (forms in this grammatical function are less likely to predict other forms in the paradigm).

The strong negative correlation between i-complexity and e-complexity has clear implications for how Ackerman and Malouf's (2013) findings should be interpreted. They show that across a sample of morphological paradigms in ten languages, e-complexity reaches relatively high values (a maximum of 4.9 bits for Mazatec), while i-complexity stays relatively constant (between 0 and 1.1 bits). However, randomly generating paradigms of a fixed shape results in a similar distribution: e-complexity varies more than i-complexity,<sup>9</sup> and when a paradigm has high e-complexity, it will necessarily also have low i-complexity. Ackerman and Malouf's (2013) findings may therefore at least partly reflect the nature of the relationship between these two

---

<sup>9</sup>Note however, that the paradigms generated here were matched in size to the paradigms used in Section 2 (3 inflectional classes and 3 grammatical functions); it could be that for much larger paradigms, such as found in natural languages, randomly generating the paradigms would result in higher i-complexity than values that can actually be found in natural languages (as suggested by the simulation with Chiquihuitlàn Mazatec done by Ackerman and Malouf 2013).



Table 5: Two example paradigms (with affixes indicated by integers) with six unique markers illustrating the inverse correlation between i-complexity and e-complexity when number of markers is constant: (a) has relatively high e-complexity (1.58 bits) and low i-complexity (0 bits), while (b) has relatively low e-complexity (0.83 bits) and relatively high i-complexity (0.52 bits). In (a) there are three different ways to mark each grammatical function (hence high e-complexity), and forms in all grammatical functions are predictive of all other forms (hence low i-complexity). In (b), on the other hand, there is only one realization for marking the plural number and two for marking dual (hence lower e-complexity), but in this organization the plural form is not predictive of forms in any other grammatical function and forms in dual do not fully predict the singular (hence higher i-complexity)

(a)

	Singular	Dual	Plural
noun class 1	6	5	6
noun class 2	8	1	3
noun class 3	5	7	7

(b)

	Singular	Dual	Plural
noun class 1	2	6	8
noun class 2	4	0	8
noun class 3	1	6	8

measures rather than anything specific to the dynamics of language change.

*The effects of i- and e-complexity  
on LSTM neural networks*

3.3

The learning results presented in Section 2 already suggest that i-complexity has less impact on learning than e-complexity in networks, and possibly no impact in humans. To strengthen this conclusion, we also test how the 1000 randomly generated paradigms described above are learned using LSTM neural networks with the same architecture

and parameters described in Section 2.2.1. Since the effects we found above held across networks of different sizes, here we only used networks of size 25 (4,656 parameters). We generated 50 different runs for each paradigm. In each run the initial weights of the network were randomly generated. As before, stems were randomly assigned into one of the three noun classes. Below we analyse accuracy in each epoch as well as the summed accuracy across epochs.

## 3.3.1

## Results

Figure 9 shows the learning trajectories of the neural networks in choosing the correct affix for lexemes, both by the i-complexity of the paradigm, and by its e-complexity.

To test how varying values of i-complexity and e-complexity affect learning, we ran a linear mixed-effects regression model predicting summed accuracy by paradigm i-complexity, paradigm e-complexity, the number of different affixes in the paradigm, and their interactions.

Summed accuracy was divided by 900 (number of epochs) to get the proportional summed accuracy, ranging from 0 to 1. I-complexity and e-complexity were scaled and number of markers was centred such that estimates for the effects of i-complexity or e-complexity reflect their effect on learning when the number of affixes equals the mean value (six affixes). In addition to these fixed effects, the model included random intercepts for different runs of the network (recall that network size was held constant).

The model revealed a significant effect of both i-complexity ( $\beta = -0.0093$ ,  $t(49992) = -9.96$ ,  $p < 0.001$ ) and e-complexity ( $\beta = -0.04$ ,  $t(49992) = -40.66$ ,  $p < 0.001$ ). These results replicate our initial findings with only four paradigms: increasing either the i-complexity or e-complexity of the paradigm leads to slower learning. Note that this holds even though, as discussed above, i-complexity and e-complexity have a strong inverse correlation ( $r = -0.94$ ). Importantly, as before the effect size of e-complexity is much higher than the effect size of i-complexity ( $-0.04$  vs.  $-0.009$ ; approximately 4 times greater), suggesting a stronger effect of e-complexity on learning.

The model also reveals a significant effect of number of affixes ( $\beta = 0.007$ ,  $t(49992) = 18.51$ ,  $p < 0.001$ ). Surprisingly, this effect

Effects of *i*- and *e*-complexity on morphological learning

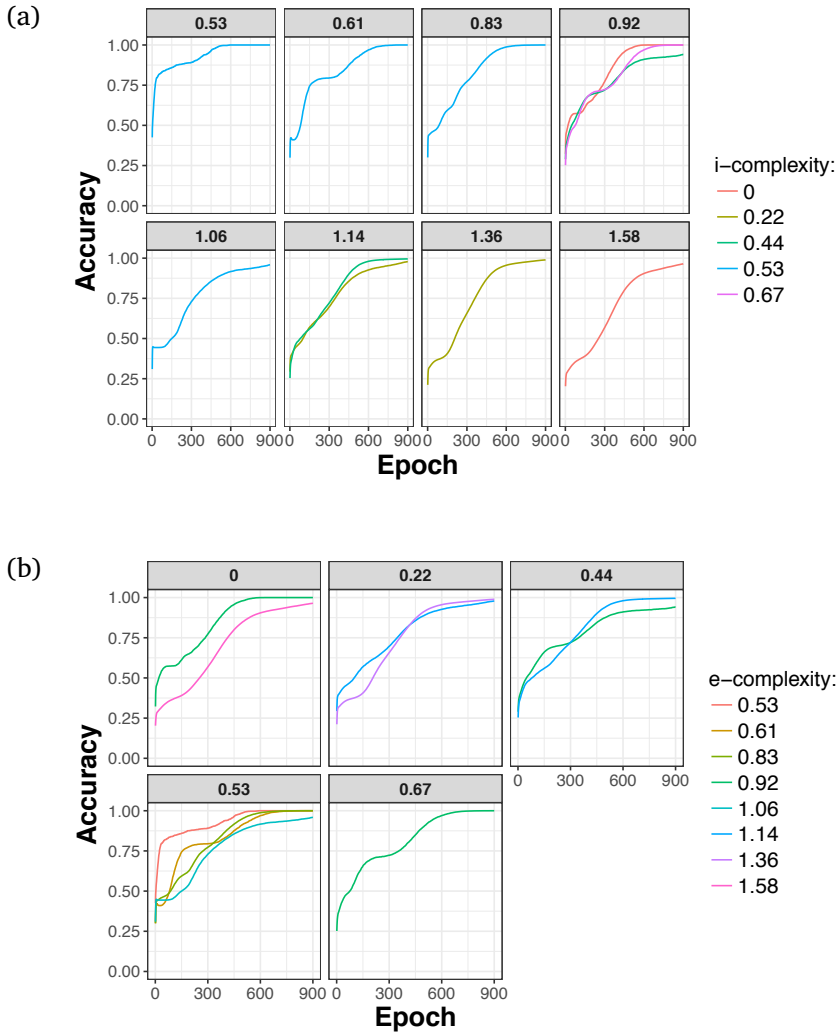


Figure 9: Network learning trajectory for paradigms varying in *i*-complexity and *e*-complexity values. (a) *i*-complexity varying by colour (facet titles showing *e*-complexity in bits). (b) *e*-complexity varying by colour (facet titles showing *i*-complexity in bits). Note that, as discussed above, for some values of *e*-complexity, the random paradigms do not vary in *i*-complexity. In these cases, only one learning curve is shown (e.g. for *e*-complexity of 0.53 bits, there are only paradigms with *i*-complexity of 0.53 bits). Differences in *e*-complexity produce higher variability in network learning trajectories (b) compared to differences in *i*-complexity (a)

is positive: more unique affixes appears to facilitate learning. However, a closer look at paradigms with the same i- and e-complexity and the same number of markers reveals a potential confounding factor, namely syncretism type. Table 6 shows an example of two of the random paradigms (labelled (a) and (b)), both of which have i-complexity of 0 bits, e-complexity of 1.58 bits, and 5 unique affixes (represented by numbers). While the proportional summed accuracy for paradigm (a) is 0.538, for paradigm (b) it is 0.87.

In paradigm (a), markers are distributed such that there is syncretism targeting forms across different noun classes. For example, the affix 1 marks singular for noun class 1, but plural for noun class 3. On the other hand, syncretic affixes in paradigm (b) are largely within noun classes. For example, the affix 1 marks singular and plural for noun class 1. There is one case of across-class syncretism in paradigm (b) – the affix 8 marks dual for noun class 1 but plural for noun class 3 – whereas in paradigm (a) there are 4 such cases. The learnability disadvantage for across-class syncretism is expected

Table 6: Two example paradigms (with affixes indicated by integers) differing only in their degree of cross-class syncretism: (a) shows only across-class syncretism, while (b) shows mostly within-class syncretism. For both paradigms i-complexity (0 bits), e-complexity (1.58 bits) and number of markers (5 markers) are matched. Paradigm (b) is learned more accurately by our networks

(a)			
	Singular	Dual	Plural
noun class 1	1	2	8
noun class 2	8	3	5
noun class 3	3	8	1

(b)			
	Singular	Dual	Plural
noun class 1	1	8	1
noun class 2	0	5	0
noun class 3	2	2	8

based on the previous results reported above. However, it turns out to lead to the unexpected apparent advantage for paradigms with more unique affixes, since paradigms with fewer affixes will tend to have more across-class syncretic forms in our design. We added number of across-class syncretic forms (centred) as a predictor in our previous regression model, including its interaction with the original predictors. This model again reveals a significant effect of i-complexity ( $\beta = -0.0086$ ,  $t(49992) = -9.12$ ,  $p < 0.001$ ) and e-complexity ( $\beta = -0.024$ ,  $t(49992) = -23.42$ ,  $p < 0.001$ ). The model also reveals a significant *negative* effect of number of affixes ( $\beta = -0.034$ ,  $t(49992) = -91.4$ ,  $p < 0.001$ ), and a significant effect of the number of across-class syncretic forms ( $\beta = -0.039$ ,  $t(49992) = -151.1$ ,  $p < 0.001$ ). Here, both of these effects are in the expected direction: having more unique affixes or having more across-class syncretic forms both lead to slower learning.

## DISCUSSION

4

In this study, we compared how different features of morphological paradigms affect learnability of morphological systems. Specifically, we compared measures reflecting the number of inflection classes in the paradigm and the number of different variants to mark each inflection (e-complexity), measures capturing the implicative structure of the paradigm and the extent to which forms in the paradigm predict each other (i-complexity), number of affixes used in the paradigm, and type of syncretism (within versus across class). We tested the effects of these features on learning inflection paradigms with human participants and with recurrent neural networks (LSTMs). In Section 2 we compared the learnability of four artificially constructed nominal inflection paradigms differing either in e- or i-complexity. We found that changing the i-complexity of the paradigm had an effect on learning only in LSTMs but did not show an effect on learning in human participants. By contrast, e-complexity was found to have a stronger effect on learning in LSTMs relative to i-complexity and low e-complexity was beneficial for human learners. These results replicate the effects reported in Johnson *et al.* (2020) and extend them to a more realistic

learning scenario where input includes all forms at all stages (rather than restricting early input to predictive forms).

It is worth noting that the differences in *i*-complexity between our low- and high-complexity paradigms were not very large – the difference is 0.222 bits. It could be that larger differences in *i*-complexity values would reveal a larger effect on learning. However even this difference corresponds to complete predictability of the dual given the singular in the low complexity paradigm, compared to at best 66% predictability in the high complexity paradigm. In other words, while the difference as measured in bits is small, the difference in probability of correct prediction in the paradigm is large. Furthermore, the same size difference in *e*-complexity values did reveal a significant effect on learning. Testing more extreme values of *i*-complexity and *e*-complexity is, in principle, possible, but would necessitate training participants on much larger inflectional paradigms. This is challenging with human participants, since our experiment was already at the upper end of what we believe participants will tolerate in a single sitting; using the same methods for larger paradigms would probably necessitate a multi-day experiment.<sup>10</sup>

Type of syncretism was also found to be predictive of learning in LSTMs; a paradigm with across-class syncretism in which the same affix is used to mark two different categories (e.g. singular and plural) for lexemes from separate inflection classes was learned slower than a paradigm with within-class syncretism, where the same affix is used to mark different numbers for lexemes within the same inflection class. This effect of syncretism on learning in LSTMs was seen both in Section 2, with the two example paradigms differing by types of syncretism, and in Section 3, when training the neural networks on paradigms with varying number of across-class syncretic forms. These results are compatible with studies with human learners showing that certain types of syncretism patterns are easier to learn than

---

<sup>10</sup> It is also worth noting that we only tested adult learners, and thus the scenario is most similar to adult L2 acquisition. It is of course possible that child L2 learners might behave differently, or that the effect of *i*-complexity is only relevant for first language acquisition. Although we have no specific reason to believe this is the case, one could, in principle, investigate child learners using the kind of study we have reported here.

others (e.g. Pertsova 2012; Maldonado and Culbertson 2019). However, in our experiment with human learners, there was no effect of type of syncretism. Given the different results in the LSTMs and human learners, these mixed results call for a more systematic investigation into the effects of syncretism type on learning morphological paradigms.

Recall that Ackerman and Malouf (2015) suggested that morphological paradigms come to have restricted values of i-complexity through the process by which language users solve the Paradigm Cell Filling Problem for unknown forms. In other words, the mechanism by which i-complexity is kept low in natural language is generalization, rather than learning more generally. In Johnson *et al.* (2020), we tested the effect of i-complexity on generalization with LSTMs, and our results there match Ackerman and Malouf's prediction: we saw a clear generalization advantage for low i-complexity paradigms. Together with our finding that i-complexity does not robustly affect paradigm learning in the absence of generalization to completely novel forms, these results suggest that i-complexity may indeed influence how paradigms evolve, but primarily (or perhaps even solely) through its impact on generalisation.

However, this interpretation is made somewhat less plausible by the results from Section 3 investigating randomly generated paradigms. These results suggest that the low i-complexity that Ackerman and Malouf (2013) observed may to some extent reflect an intrinsic relationship between the two measures. Specifically, we found that for randomly-generated paradigms, e-complexity and i-complexity are strongly negatively correlated; crucially, there were no paradigms with both high e-complexity and high i-complexity (Figure 8). Moreover, the ranges of values the two measures exhibited were different, with lower and less varied values of i-complexity (0 to 1.667 bits) than the values of e-complexity (0.528 to 1.585 bits). Following these results from Section 3, we would therefore *expect* to find similar trends in natural languages, as indeed shown in Ackerman and Malouf (2013). Any typological observation deviating from this trend would call for a theoretical explanation.

In addition to manipulating e- and i-complexity, the number of affixes used in the random paradigms was not fixed and varied randomly from 3 to 8 affixes. This allowed us to test the effect of the number of

affixes on morphological learning by the networks and to explore the relationship between this aspect of the paradigm and the two complexity measures. The number of affixes was found to positively correlate with e-complexity and to negatively correlate with i-complexity; an inflectional paradigm with low i-complexity is more likely to have a high number of affixes and to be more e-complex. Note that this gives support to our decision to use average cell entropy to measure e-complexity in this study; it is positively correlated with number of affixes in the paradigm, a common measure for e-complexity in the literature, in randomly generated paradigms.

The high inverse correlation between e-complexity and i-complexity was also found when looking at a subset of paradigms with the same number of unique affixes (six). Together with the previous finding, showing that both e-complexity and i-complexity correlate with number of affixes, these results suggest that the inverse correlation between i-complexity and e-complexity derives from both the number of affixes in the paradigm, and from the way the affixes are organized in the paradigm; intuitively, when there is a frequent form with which a grammatical function is realized across noun classes, the entropy of this grammatical function is reduced and thus the overall e-complexity is likely to be lower. However, forms in this grammatical function are less likely to predict other forms in the paradigm and therefore its overall i-complexity is likely to be high. This is more likely to occur with low number of unique affixes in the paradigm, but the relationship between e- and i-complexity can be seen even when controlling for number of affixes.

Finally, generating the random paradigms also enabled us to test the effect of e- and i-complexity on learning with LSTM networks for a larger range of values of the two measures, as opposed to the specific values we tested in Section 2. Again, we found that both e-complexity and number of affixes strongly predict learnability of the paradigm. I-complexity was also found to predict the learnability of the paradigm, but with a much smaller effect size ( $-0.0086$  vs.  $-0.024$  for e-complexity).

The strong effect of e-complexity (measured as average cell entropy) on the learnability of morphological paradigms found here suggests that the frequency of forms play an important role in the learnability of the paradigm. This is a further evidence for the pervasive-



ness of the effects of frequency on language learning (e.g. Ambridge *et al.* 2015). In the context of inflectional complexity, Sims and Parker (2016) suggest that in addition to implicative structure (i-complexity), type frequency of inflection classes also plays a role in reducing the complexity of the paradigm. In our experiments, type frequency of all noun classes was kept constant (with three words per noun class), but our results support the general claim that the frequency of elements in the paradigm plays a role in inferring the correct inflected form for a lexeme.

To summarize, our findings suggest that a number of factors affect the learnability of inflection paradigms. However, these factors do not all play equal roles in determining ease of learning. The i-complexity of a paradigm does affect learning, at least in neural networks. But it is a relatively weak predictor of learnability relative to e-complexity (and number of unique affixes). Moreover, all paradigm features examined here were found to be interdependent, most crucially e- and i-complexity. This suggests that conclusions about the contribution of different types of complexity to natural language paradigms must take into account how measures of complexity relate to one another; observing measures independently can lead to potentially misleading conclusions about how different types of complexity might shape language.

Lastly, it is worth returning to the observation that e-complexity varies widely in morphological paradigms across languages. Since our findings show that e-complexity better predicts the learnability of the paradigm, all other things being equal, paradigms with low e-complexity should be preferred. Of course, learnability is not the only factor shaping linguistic systems: languages are used for communication, and linguistic systems have been claimed to reflect a trade-off between inductive biases (e.g. for simplicity) and pressure from communication (e.g. minimizing ambiguity, Kemp and Regier 2012). This trade-off has been shown in a variety of linguistic domains, where natural languages show a near-optimal balance between these two pressures (e.g. Regier *et al.* 2015; Xu *et al.* 2016; Zaslavsky *et al.* 2020). Evidence for this trade-off has also been found in experimental studies manipulating the relative importance of learning and communication (e.g. Silvey *et al.* 2015; Kirby *et al.* 2015; Motamedi *et al.* 2019). Since we showed here that e-complexity correlates positively with a num-

ber of distinct forms in the paradigm (i.e. distinctions in the lexicon), morphological paradigms with high e-complexity could in principle reflect a balance between the communicative needs of speakers and the inductive biases of learners. Relatedly, it may be that e-complexity interacts with frequency effects coming from other aspects of the morphological paradigm and the lexicon. E-complexity captures the distribution of forms for each grammatical number, and thus reflects only the frequency of a specific aspect of the morphological paradigm. It is possible however that paradigms with high e-complexity have other means for reducing learning-relevant complexity, e.g. through skewed distribution of other aspects of the paradigm (e.g. type/token frequencies of inflection classes or frequencies of forms of grammatical functions in the paradigm).

5

CONCLUSIONS

On the surface, natural languages exhibit a huge range of variation in terms of their inflectional paradigms; some languages have relatively little morphology, and others have large morphological paradigms with many inflectional classes, expressing many grammatical categories. How such large paradigms are acquired, and by extension how they persist across generations of learners is thus something of a mystery. A recent influential conjecture is that predictive structure is a shared feature of large paradigms one finds in natural languages (Ackerman and Malouf 2013). One possibility is that this predictive structure influences how languages change over time: inflectional paradigms have evolved under a pressure for low i-complexity (a measure of predictive structure in paradigms), rather than a pressure for low e-complexity (a measure of paradigm size). Here we presented results from a series of experiments with neural networks and human learners which muddy this picture. First, we find relatively small effects of i-complexity on learning, but robust effects of e-complexity. Further, we find that in randomly generated paradigms, e-complexity and i-complexity are negatively correlated; roughly speaking, as paradigms get bigger, they will necessarily have more predictive structure. Although it may well be that learners use predictive structure

when it's all they have to go on, our findings therefore suggest that pressure from learning should tend to favour low e-complexity rather than low i-complexity.

APPENDIX 6

*Exploring hyperparameters space* 6.1

For the LSTM model presented in Section 2.2 we explored further hyperparameters in addition to the parameter settings specified in the main text. We explored two optimizers, SGD and Adam (Kingma and Ba 2014). We used these two optimizers with networks of two hidden and embedding dimensions (5 and 25), trained with four different learning rates. Since we were interested in the cases where the networks fully learned the forms in the language by the end of 900 epochs, the explored learning rates differed across optimizers; for models optimized with SGD, we explored learning rates of 0.05, 0.1, 0.15 and 0.2. For models optimized with Adam, where learning was more rapid, we explored learning rates of 0.0005, 0.001, 0.0015 and 0.002.

Results are presented in Figures 10–13, and a summary of the mean summed accuracy for all combinations of hyperparameters is presented in Tables 7, 8 below. Results from all models optimized with SGD show small effects of i-complexity compared to effects of e-complexity, regardless of the learning rate of the network. Models optimized with Adam show a similar trend for the very low learning rates, but for the rest of the models there is no difference between the conditions. Crucially, none of the hyperparameters combinations we explored showed the opposite picture where i-complexity has a stronger effect on learning than e-complexity.

These results show that for this space of hyperparameters, all models replicate the results presented in Section 2.2, namely that in cases where i-complexity has an effect on learning the paradigm, the effect is smaller than the effect of e-complexity.

Table 7: Summary of mean of summed accuracy of the model runs optimized with SGD with combinations of hidden and embedding dimensions (5, 25) and learning rates (0.05, 0.1, 0.15, 0.2). Standard deviations are presented in brackets

		5				25			
		0.05	0.1	0.15	0.2	0.05	0.1	0.15	0.2
SGD	low-i	439.6	637.0	724.4	761.7	560.6	722.2	784.4	811.1
	/low-e	(48.7)	(47.0)	(32.2)	(22.9)	(35.1)	(20.0)	(15.6)	(10.82)
	high-i	440.5	629.0	724.3	765.6	538.5	722.3	782.9	808.0
	/low-e	(50.3)	(49.2)	(30.8)	(21.6)	(27.1)	(16.9)	(12.7)	(11.4)
	low-i	367.9	594.5	690.8	743.1	466.8	674.9	750.9	787.7
	/high-e	(41.4)	(51.0)	(33.5)	(21.4)	(41.4)	(26.5)	(18.1)	(13.4)

Table 8: Summary of mean of summed accuracy of the model runs optimized with Adam with combinations of hidden and embedding dimensions (5, 25) and learning rates (0.0005, 0.001, 0.0015, 0.002). Standard deviations are presented in brackets

		5				25			
		0.0005	0.001	0.0015	0.002	0.0005	0.001	0.0015	0.002
Adam	low-i	483.5	678.7	747.9	786.9	786.7	827.9	849.7	860.8
	/low-e	(58.7)	(35.2)	(24.2)	(18.3)	(13.8)	(8.5)	(7.1)	(5.2)
	high-i	512.1	680.3	751.4	787.7	762.2	827.3	847.3	858.2
	/low-e	(44.8)	(28.8)	(21.7)	(13.5)	(14.6)	(7.5)	(5.9)	(4.9)
	low-i	469.3	670.2	742.9	782.3	746.6	814.6	840.1	852.5
	/high-e	(40.9)	(32.0)	(20.11)	(13.0)	(11.4)	(5.9)	(3.8)	(3.3)

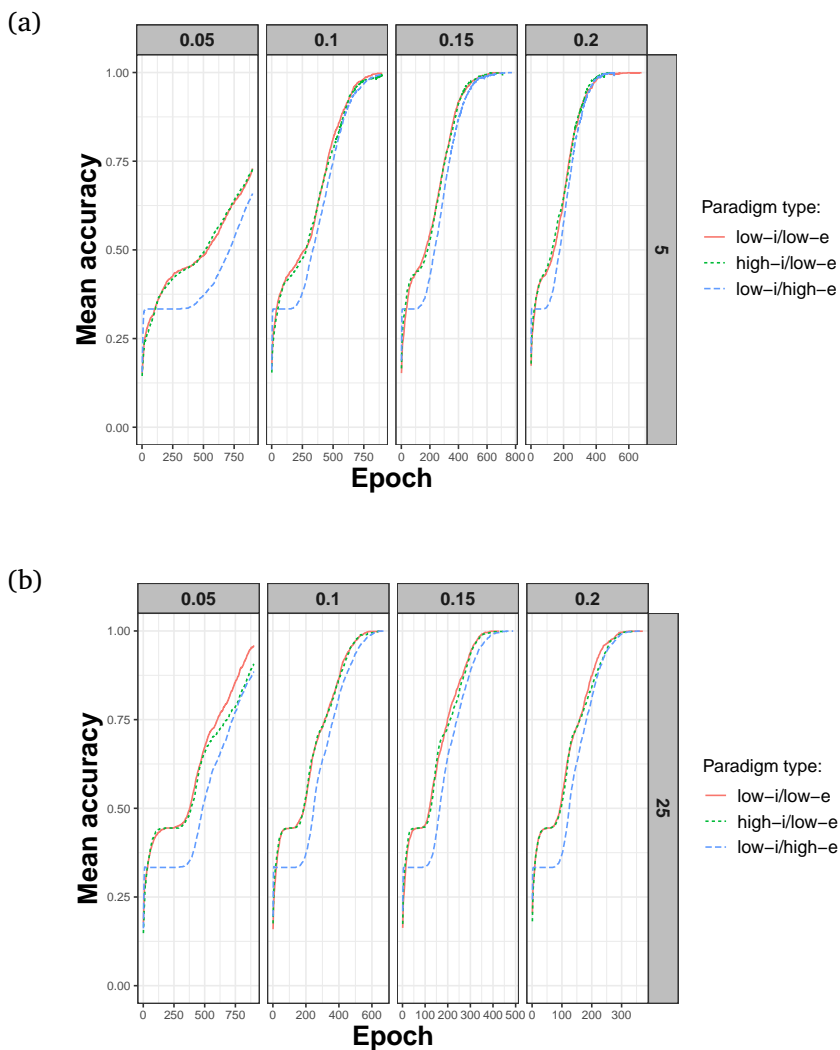


Figure 10: Learning trajectories of networks with two embedding and hidden layer dimensionalities; (a) networks with 5-dimensional embedding vectors and hidden layer, (b) networks with 25-dimensional embedding vectors and hidden layer, trained with different learning rates (columns), and optimized with SGD. X axis shows number of epochs up to perfect learning of the forms in the language (differs across learning rates and network dimensions)

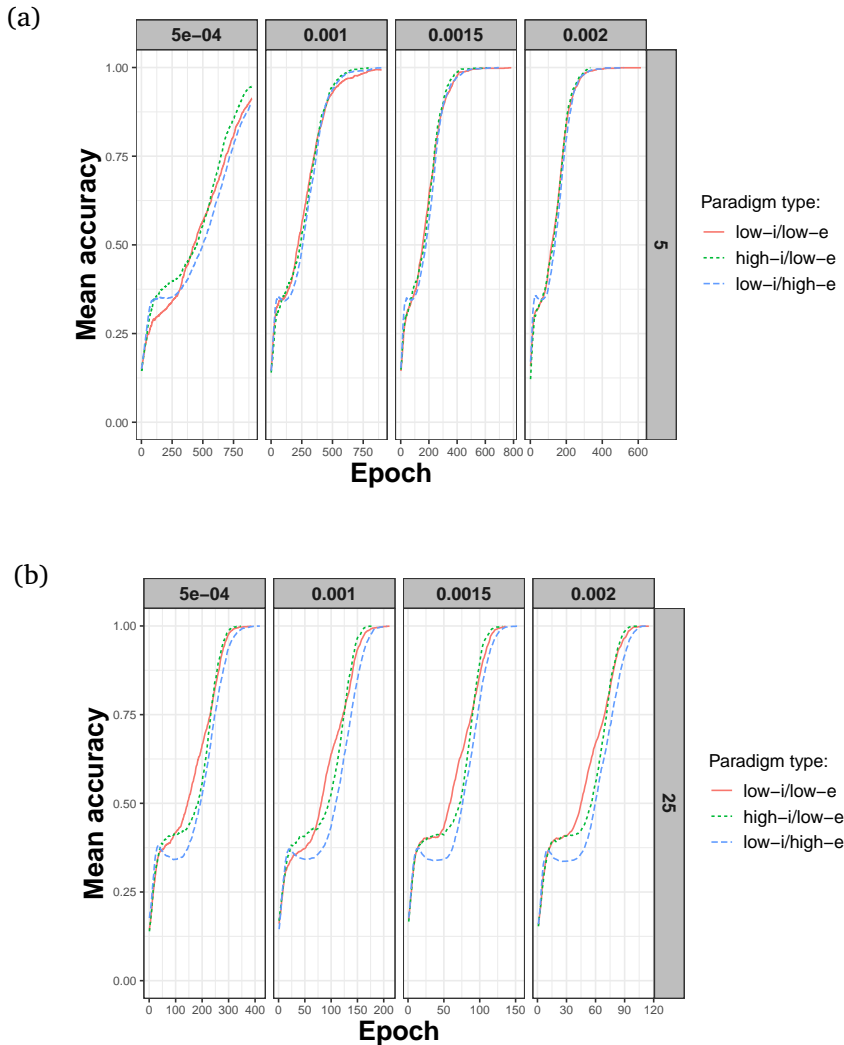


Figure 11: Learning trajectories of networks with two embedding and hidden layer dimensionalities; (a) networks with 5-dimensional embedding vectors and hidden layer, (b) networks with 25-dimensional embedding vectors and hidden layer, trained with different learning rates (columns), and optimized with Adam. X axis shows number of epochs up to perfect learning of the forms in the language (differs across learning rates and networks dimensions)

*Effects of i- and e-complexity on morphological learning*

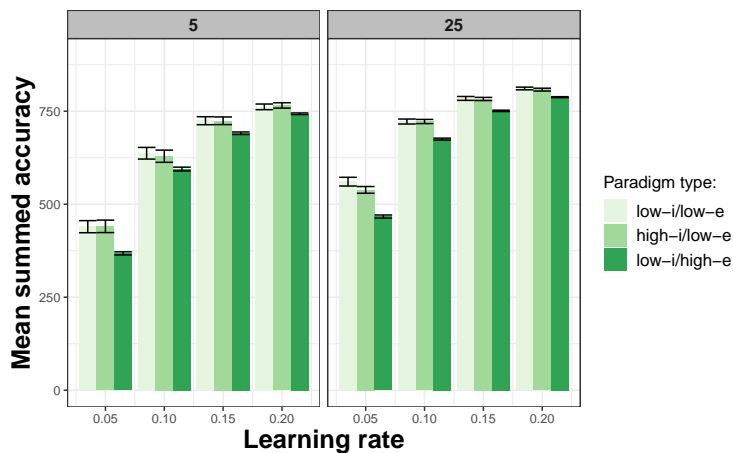


Figure 12: Summed accuracy over the 900 epochs of the networks trained on each of the three paradigm types for models with different learning rates ( $x$  axis) and for models with different dimensions (columns) optimized with SGD

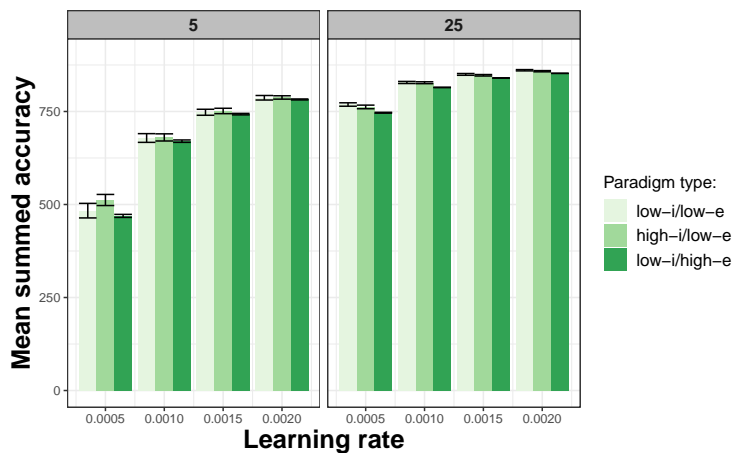


Figure 13: Summed accuracy over the 900 epochs of the networks trained on each of the three paradigm types for models with different learning rates ( $x$  axis) and for models with different dimensions (columns) optimized with Adam

## REFERENCES

- Farrell ACKERMAN, James P. BLEVINS, and Robert MALOUF (2009), Parts and wholes: Patterns of relatedness in complex morphological systems and why they matter, in James P. BLEVINS and Juliette BLEVINS, editors, *Analogy in grammar: Form and acquisition*, pp. 54–82, Oxford University Press, Oxford.
- Farrell ACKERMAN and Robert MALOUF (2013), Morphological organization: The low conditional entropy conjecture, *Language*, 89(3):429–464.
- Farrell ACKERMAN and Robert MALOUF (2015), The No Blur Principle effects as an emergent property of language systems, in *Proceedings of the annual meeting of the Berkeley Linguistics Society*, volume 41, pp. 1–14.
- Ben AMBRIDGE, Evan KIDD, Caroline F. ROWLAND, and Anna L. THEAKSTON (2015), The ubiquity of frequency effects in first language acquisition, *Journal of Child Language*, 42(2):239–273.
- Mark ARONOFF (1994), *Morphology by itself: Stems and inflectional classes*, MIT Press.
- Matthew BAERMAN, Dunstan BROWN, and Greville G. CORBETT (2005), *The syntax-morphology interface: A study of syncretism*, Cambridge University Press.
- Matthew BAERMAN, Dunstan BROWN, and Greville G. CORBETT (2010), Morphological complexity: a typological perspective, <https://www.researchgate.net/publication/266215146>, unpublished manuscript, University of Surrey.
- Douglas BATES, Martin MÄCHLER, Ben BOLKER, and Steve WALKER (2014), Fitting linear mixed-effects models using lme4, *arXiv preprint arXiv:1406.5823*.
- Balthasar BICKEL and Johanna NICHOLS (2013), Inflectional synthesis of the verb, in Matthew S. DRYER and Martin HASPELMATH, editors, *The world atlas of language structures online*, Max Planck Institute for Evolutionary Anthropology, <https://wals.info/chapter/22>.
- James P. BLEVINS (2006), Word-based morphology, *Journal of Linguistics*, 42(3):531–573.
- Olivier BONAMI and Sacha BENIAMINE (2016), Joint predictiveness in inflectional paradigms, *Word Structure*, 9(2):156–182.
- François CHOLLET et al. (2015), keras, <https://keras.io>.
- Morten H. CHRISTIANSEN and Nick CHATER (2008), Language as shaped by the brain, *The Behavioral and Brain Sciences*, 31(5):489–509.
- Greville G. CORBETT (2009), Suppletion: Typology, markedness, complexity, in Patrick O. STEINKRÜGER and Manfred KRIFKA, editors, *On inflection*, p. 40, Mouton de Gruyter.



- Ryan COTTERELL, Christo KIROV, Mans HULDEN, and Jason EISNER (2019), On the complexity and typology of inflectional morphological systems, *Transactions of the Association for Computational Linguistics*, 7:327–342.
- Jennifer CULBERTSON and Simon KIRBY (2016), Simplicity and specificity in language: Domain-general biases have domain-specific effects, *Frontiers in psychology*, 6:1964.
- Jennifer CULBERTSON and Elissa L. NEWPORT (2015), Harmonic biases in child learners: In support of language universals, *Cognition*, 139(6):71–82.
- Jennifer CULBERTSON, Paul SMOLENSKY, and Géraldine LEGENDRE (2012), Learning biases predict a word order universal, *Cognition*, 122(3):306–329.
- Terrence William DEACON (1997), *The symbolic species: The co-evolution of language and the brain*, Allen Lane the Penguin Press.
- Jeffrey L. ELMAN (1990), Finding structure in time, *Cognitive Science*, 14(2):179–211.
- Jeffrey L. ELMAN (1991), Distributed representations, simple recurrent networks, and grammatical structure, *Machine Learning*, 7(2):195–225.
- Maryia FEDZECHKINA, T. Florian JAEGER, and Elissa L. NEWPORT (2012), Language learners restructure their input to facilitate efficient communication, *Proceedings of the National Academy of Sciences*, 109(44):17897–17902.
- Richard FUTRELL, Ethan WILCOX, Takashi MORITA, Peng QIAN, Miguel BALLESTEROS, and Roger LEVY (2019), Neural language models as psycholinguistic subjects: Representations of syntactic state, in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pp. 32–42.
- Kristina GULORDAVA, Piotr BOJANOWSKI, Edouard GRAVE, Tal LINZEN, and Marco BARONI (2018), Colorless green recurrent networks dream hierarchically, in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pp. 1195–1205.
- Sepp HOCHREITER and Jürgen SCHMIDHUBER (1997), Long short-term memory, *Neural Computation*, 9(8):1735–1780.
- Carla L. HUDSON KAM and Elissa L. NEWPORT (2005), Regularizing unpredictable variation: The roles of adult and child learners in language formation and change, *Language Learning and Development*, 1(2):151–195.
- Carla L HUDSON KAM and Elissa L NEWPORT (2009), Getting it right by getting it wrong: When learners change languages, *Cognitive Psychology*, 59(1):30–66.
- Tamar JOHNSON, Jennifer CULBERTSON, Hugh RABAGLIATI, and Kenny SMITH (2020), Assessing integrative complexity as a predictor of morphological learning using neural networks and artificial language learning,

<https://psyarxiv.com/yngw9/>, unpublished manuscript, University of Edinburgh.

Michael I. JORDAN (1997), Serial order: A parallel distributed processing approach, in John W. DONAHOE and Vivian PACKARD DORSEL, editors, *Neural-network models of cognition*, pp. 471–495, Elsevier.

Charles KEMP and Terry REGIER (2012), Kinship categories across languages reflect general communicative principles, *Science*, 336(6084):1049–1054.

Diederik P. KINGMA and Jimmy BA (2014), Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980*.

Simon KIRBY (2002), Learning, bottlenecks and the evolution of recursive syntax, in Ted BRISCOE, editor, *Linguistic evolution through language acquisition*, pp. 173–204, Cambridge University Press.

Simon KIRBY, Hannah CORNISH, and Kenny SMITH (2008), Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language, *Proceedings of the National Academy of Sciences*, 105(31):10681–10686.

Simon KIRBY, Monica TAMARIZ, Hannah CORNISH, and Kenny SMITH (2015), Compression and communication in the cultural evolution of linguistic structure, *Cognition*, 141:87–102.

Alexandra KUZNETSOVA, Per B. BROCKHOFF, Rune HB CHRISTENSEN, et al. (2017), lmerTest package: tests in linear mixed effects models, *Journal of Statistical Software*, 82(13):1–26.

Tal LINZEN, Emmanuel DUPOUX, and Yoav GOLDBERG (2016), Assessing the ability of LSTMs to learn syntax-sensitive dependencies, *Transactions of the Association for Computational Linguistics*, 4:521–535.

Mora MALDONADO and Jennifer CULBERTSON (2019), Something about "us": Learning first person pronoun systems, in *Proceedings of the 41st annual meeting of the Cognitive Science Society*, pp. 749–755.

Robert MALOUF (2017), Abstractive morphological learning with a recurrent neural network, *Morphology*, 27(4):431–458.

Eric MEINHARDT, Rob MALOUF, and Farrell ACKERMAN (2019), Morphology gets more and more complex, unless it doesn't, <https://www.researchgate.net/publication/333194657>, unpublished manuscript, San Diego State University and University of California San Diego.

Daniel MIRMAN (2017), *Growth curve analysis and visualization using R*, CRC Press, first edition. edition.

Elliott MORETON and Joe PATER (2012), Structure and substance in artificial-phonology learning, part I: Structure, *Language and Linguistics Compass*, 6(11):686–701.

- Yasamin MOTAMEDI, Marieke SCHOUWSTRA, Kenny SMITH, Jennifer CULBERTSON, and Simon KIRBY (2019), Evolving artificial sign languages in the lab: From improvised gesture to systematic sign, *Cognition*, 192:103964–103964.
- Katya PERTSOVA (2012), Logical complexity in morphological learning: Effects of structure and null/overt affixation on learning paradigms, in *Annual meeting of the Berkeley Linguistics Society*, volume 38, pp. 401–413.
- Terry REGIER, Charles KEMP, and Paul KAY (2015), Word meanings across languages support efficient communication, in Brian MACWHINNEY and William O'GRADY, editors, *The handbook of language emergence*, pp. 237–263, John Wiley & Sons, Inc.
- Scott SEYFARTH, Farrell ACKERMAN, and Robert MALOUF (2014), Implicative organization facilitates morphological learning, in *Annual meeting of the Berkeley Linguistics Society*, volume 40, pp. 480–494.
- Claude Elwood SHANNON (1963), *The mathematical theory of communication*, University of Illinois Press.
- Ryan K SHOSTED (2006), Correlating complexity: A typological approach, *Linguistic Typology*, 10(1):1–40.
- Catriona SILVEY, Simon KIRBY, and Kenny SMITH (2015), Word meanings evolve to selectively preserve distinctions on salient dimensions, *Cognitive Science*, 39(1):212–226.
- Andrea D SIMS and Jeff PARKER (2016), How inflection class systems work: On the informativity of implicative structure, *Word Structure*, 9(2):215–239.
- Elizabeth WONNACOTT and Elissa L. NEWPORT (2005), Novelty and regularization: The effect of novel instances on rule formation, in *BUCLD 29: Proceedings of the 29th annual Boston University conference on language development*, pp. 663–673.
- Yang XU, Terry REGIER, and Barbara C MALT (2016), Historical semantic chaining and efficient communication: The case of container names, *Cognitive Science*, 40(8):2081–2094.
- Noga ZASLAVSKY, Charles KEMP, Naftali TISHBY, and Terry REGIER (2020), Communicative need in colour naming, *Cognitive Neuropsychology*, 37(5-6):312–324.

*Tamar Johnson*

© 0000-0003-1071-6750  
tamar.johnson@unige.ch

*Kexin Gao*

kexin.gao@hotmail.com

*Kenny Smith*

© 0000-0002-4530-6914  
kenny.smith@ed.ac.uk

*Jennifer Culbertson*

© 0000-0002-1737-6296  
jennifer.culbertson@ed.ac.uk

Centre for Language Evolution,  
University of Edinburgh,  
Edinburgh, Scotland, United Kingdom

*Hugh Rabagliati*


© 0000-0001-9828-5857  
hugh.rabagliati@ed.ac.uk

Department of Psychology,  
University of Edinburgh,  
Edinburgh, Scotland, United Kingdom

Tamar Johnson, Kexin Gao, Kenny Smith, Hugh Rabagliati, and Jennifer Culbertson (2021), *Investigating the effects of i-complexity and e-complexity on the learnability of morphological systems*, *Journal of Language Modelling*, 9(1):97–150

doi <https://dx.doi.org/10.15398/jlm.v9i1.259>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

cc  <http://creativecommons.org/licenses/by/4.0/>

# Typology emerges from simplicity in representations and learning

*Dakotah Lambert*<sup>1</sup>, *Jonathan Rawski*<sup>2</sup>, and *Jeffrey Heinz*<sup>1</sup>

<sup>1</sup> Stony Brook University

<sup>2</sup> San José State University

## ABSTRACT

We derive well-understood and well-studied subregular classes of formal languages purely from the computational perspective of algorithmic learning problems. We parameterise the learning problem along dimensions of representation and inference strategy. Of special interest are those classes of languages whose learning algorithms are necessarily not prohibitively expensive in space and time, since learners are often exposed to adverse conditions and sparse data. Learned natural language patterns are expected to be most like the patterns in these classes, an expectation supported by previous typological and linguistic research in phonology. A second result is that the learning algorithms presented here are completely agnostic to choice of linguistic representation. In the case of the subregular classes, the results fall out from traditional model-theoretic treatments of words and strings. The same learning algorithms, however, can be applied to model-theoretic treatments of other linguistic representations such as syntactic trees or autosegmental graphs, which opens a useful direction for future research.

*Keywords:*  
*model theory,*  
*subregularity,*  
*grammatical*  
*inference, formal*  
*language theory,*  
*phonology,*  
*learning*  
*complexity*

This paper presents an analysis supporting the view that the computational simplicity of learning mechanisms has considerable impact on the types of patterns found in natural languages.

First, we derive well-understood and well-studied subregular classes of formal languages purely from the computational perspective of algorithmic learning problems. We present a family of four learning algorithms, generalizing the String Extension learners in Heinz (2010b). We show that these algorithms, over different data structures, naturally structure the subregular Hierarchy of language classes purely by difficulty of learning. We show that the simplest classes of languages in these hierarchies are precisely the ones whose learning algorithms use the least computational resources, in particular space complexity. In fact, these are the only ones that are not prohibitively expensive to learn. A reasonable prediction is that learned natural language patterns would be most similar to patterns in the simplest of these classes, and this expectation is supported by previous typological and linguistic research in the domain of phonology.

The second result is that we introduce *linear-time* learning algorithms for some subregular classes, a further restriction of the typology beyond space-efficiency. As we explain, these algorithms are helpful in certain cases and not so helpful in others, depending on the extent to which the target patterns interact with other constraints. At issue is that a set of data points which may be helpful in identifying one constraint do not occur because they also happen to violate another. A virtue of this analysis is that we can identify precisely the situations where the linear-time learning algorithms can be applied.

Our third result is that the learning algorithms presented here are completely agnostic to choice of linguistic representation. These learning algorithms essentially parameterise the learning problem in two ways: the structural knowledge salient to the learner (the representation), and the way the learner collects and combines this structural information to derive sets of acceptable and unacceptable linguistic structures. In the case of the subregular classes of formal languages, the results emerge from traditional model-theoretic treatments of words and strings on the representational side and how the

combinatorics of the grammars relate to kinds of logical languages on the other side.

Since the algorithms are agnostic to the representations, the same learning algorithms can be applied to model-theoretic treatments of other linguistic representations such as syntactic tree structures or autosegmental graphs. Of course, the real-life learning problem is complicated by the fact that language learners do not have direct access to linguistic structures like trees. Nonetheless the generality of these learning algorithms means the real-life learning problems may be reduced to these algorithms coupled with appropriate parsing mechanisms.

### *Priors in language learning*

1.1

Language acquisition succeeds despite sparse, underdetermined, Zipf-distributed input, compounded by a lack of invariance in the signal – the so-called poverty of stimulus (Yang 2013). This holds across all domains of language, from phonological to syntactic induction.

It is uncontroversial that *some* bias or innate component restricts a learner’s hypothesis space regardless of its strategy to solve this induction problem, often referred to as Universal Grammar (Nowak *et al.* 2002). The question is its nature. How is it rich, and how is it poor?

Data-driven statistical learning does not change this basic calculus. One reason is that children often learn language in ways that defy adult distributions (Legate and Yang 2002). Another is that induction from a data distribution without a prior may only recapitulate the training data (Fodor and Pylyshyn 1988; Mitchell 1982, 2017), and cannot generalize. Without a lens in which linguistic experience is viewed, even the input distribution cannot be recovered, simply because distributions are based on the structure of their parameters (Lapin and Shieber 2007). Consequently, the nontrivial open question central to learnability research in linguistics instead concerns the characteristics of this additional prior knowledge or bias such that learners *generalize* from limited experience (Rawski and Heinz 2019). This point is not specific to language. Any cognitive theory requires carefully constructed computational restrictions on the hypothesis space in order to be tractable and analytically verifiable (van Rooij and Baggio 2021).

Recent typological and experimental work highlights the Regular region of languages as a sufficient structural bound on computational expressivity for phonological and morphological grammars. This Regular characterization has been extended to syntactic distributions when the data structure characterizing the computational trace is formulated as a tree rather than strings, which enforce syntactic membership in the Mildly Context-Sensitive class of languages (Kobele 2011; Graf 2011). However, the Regular class is not learnable under various learning scenarios including identification in the limit from positive data, and the Probably Approximately Correct (PAC) framework (Gold 1967; Valiant 1984; de la Higuera 2010). Additionally, the range of distributions present in phonology and morphology that sit in the Regular region do not require the full complexity of Regular power (Heinz 2018; Chandlee 2017).

For these reasons, phonological constraints are hypothesised to inhabit structured subclasses of the Regular languages, lumped under the term subregular (Heinz 2010a, 2018). Various connections between logic, formal languages and automata defining these classes have been explored in great detail. These characterizations build on two classical results in formal language theory: Büchi's monadic second-order characterization of the Regular languages (1960), and the first-order characterization by McNaughton and Papert (1971) of the Star-Free languages, which are also characterized by aperiodic deterministic finite-state automata (Schützenberger 1965). Refinements of these results from logical, automata-theoretic, and algebraic viewpoints have defined the Local and Piecewise hierarchies (Rogers *et al.* 2012). Linguistically, these refinements have garnered interest since the morphological and phonological typology correlates with these refinements, favouring the weakest subclasses in the subregular hierarchy. Experimental work also favours this characterization (Finley 2008; Lai 2015; McMullin and Hansson 2019). Our learning algorithms can be applied to model-theoretic treatments of other linguistic representations such as syntactic trees or autosegmental graphs, which opens a useful direction for future research.



This paper proceeds as follows. Section 2 defines a general model-theoretic treatment of linguistic representations, and analyzes several types of linguistic structures based on different model signatures. Section 3 defines a typology of online learning algorithms and derives the subregular language classes, hierarchically organised by space complexity. Section 4 characterizes this space of algorithms according to time complexity, and picks out of the least space-intensive subregular classes those that can be learned in linear time. Section 5 characterizes interactions of constraints defined in and between these classes. Section 6 discusses model signatures for other linguistic representations. Section 7 describes related work. Section 8 concludes with future directions.

## MODEL THEORIES

This section will introduce the structural representations that the learning algorithms will work over. We will first discuss a general notion of structural information, and use it to derive a notion of *substructures*. In contrast to previous approaches, this will allow us to describe several distinct representations of words in a uniform way. Structural information is defined relationally in terms of model theory. Finite model theory provides a unified ontology and a vocabulary for representing many kinds of objects, by considering them as *relational structures* (see Libkin 2004 for a thorough introduction). This allows flexible but precise definitions of the structural information in an object, by explicitly defining its parts and the relations between them. This makes model-theoretic representations a powerful tool for analyzing the information characterizing a certain structure. This application of model theory is nothing new. It has been applied to syntax by Johnson (1988), King (1989), and Rogers (1998), to phonology by Potts and Pullum (2002), Rogers *et al.* (2012), and Strother-Garcia (2019), and to tonal systems and autosegmental representations by Jardine (2017a), Jardine *et al.* (2021), and Oakden (2020).

The discussion of this section is organized around different notions of order: successor, precedence, and relativized successor. The successor and precedence orders give rise to the Local and Piecewise branches of the subregular hierarchy, and the relativized successor gives rise to the Tier-Based Local branch. We assume some familiarity with these classes. Because this presentation focuses on deriving these subregular classes from a model-theoretic and learning perspective, we postpone most references to these classes and related work to Section 7.

A relational structure in general is a set of domain elements,  $\mathcal{D}$ , which is augmented with a set of relations of arbitrary arity,  $R_i \subseteq \mathcal{D}^{n_i}$ . The relations provide information about the domain elements. The *model signature*  $\mathcal{M} = \langle \mathcal{D}; R_i \rangle$  collects these parts and defines the nature of the structure in terms of the information in the model. Let  $w$  be a string over some alphabet  $\Sigma$ . Then a model for a word  $w$  is a structure:

$$\mathcal{M}_{\Sigma}^{R_i}(w) := \langle \mathcal{D}_w; R_i, \sigma_w \rangle_{\sigma \in \Sigma}$$

where  $\mathcal{D}_w$  is isomorphic to an initial segment  $\langle 1, \dots, |w| \rangle$  of the non-zero natural numbers and represents the positions in  $w$ , and each  $\sigma_w$  is a unary relation that holds for all and only those positions at which  $\sigma$  occurs. Note that it is assumed that the set  $\{\sigma_w\}_{\sigma \in \Sigma}$  is a partition of  $\mathcal{D}_w$ .<sup>1</sup> Without loss of generality, consider an alphabet  $\Sigma = \{s, \int, \acute{a}, \grave{a}\}$ , which represent two types of sibilants and a vowel with either low or high tone. Strings are combinations of these symbols at certain events, like the word ‘sásàfá’.

The remaining  $R_i$  are the other salient relations, which are used to define order in a particular structure. One model signature for strings, called the *precedence model*, is given as

$$\mathcal{M}^<(w) = \langle \mathcal{D}_w; <_w, s_w, \int_w, \acute{a}_w, \grave{a}_w \rangle.$$

This model says that for every symbol  $\sigma$  in alphabet  $\Sigma$ , there is a unary relation  $R_{\sigma}$  in  $\mathcal{R}$  that can be thought of as a labelling relation for that symbol. For our set  $\Sigma = \{s, \int, \acute{a}, \grave{a}\}$ ,  $\mathcal{R}$  includes the unary relations  $R_s$ ,

---

<sup>1</sup>One can convert a model in which multiple unary relations may apply to a given domain element into a partitioned normal form by simply replacing these unary relations with their powerset.

$R_f$ ,  $R_{\grave{a}}$ , and  $R_{\acute{a}}$ . It also defines a binary relation ( $x < y$ ), the general precedence relation on the domain  $\mathcal{D}$ . A visual of the word model for ‘sásàfá’ under this signature is given in Figure 1.

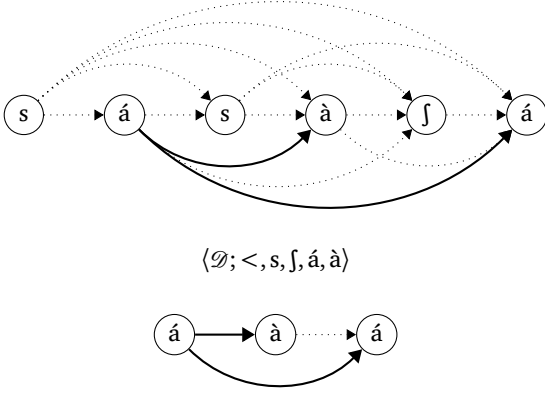


Figure 1:  
The general precedence model of ‘sásàfá’, along with the 3-factor ‘áàá’. Each edge defined by the relation is pictured, while the thick solid edges designate those that form the window from which this 3-factor is derived

The general precedence relation describes a notion of structural information purely in terms of whether a node precedes another one. While the information that, say, the last element in a string comes after the first is immediately accessible from the model, this distinction collapses the notions of immediate and general structural adjacency. Building on this precedence relation we can derive different types of relational structure. These refine the model of a word to describe immediate, relativized, or multiply-relativized adjacency.

Perhaps we would like to consider only immediately adjacent elements. Rather than a general precedence relation  $<$ , we may consider an immediate precedence, or successor, relation  $\triangleleft$ . The standard successor relation is the transitive reduction of the precedence relation and is first-order definable from the latter as follows:

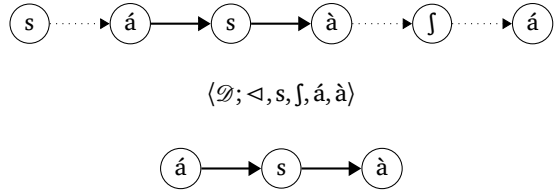
$$x \triangleleft y := x < y \wedge (\forall z)[x < z \Rightarrow y \leq z].$$

This relation gives a different word model, where elements are arranged according to immediate adjacency, commonly called the *successor model*. The signature for this model is given as

$$\mathcal{M}^{\triangleleft}(w) = \langle \mathcal{D}_w; \triangleleft_w, s_w, f_w, \acute{a}_w, \grave{a}_w \rangle.$$

A visual of the successor word model for the word ‘sásàfá’ is given in Figure 2.

Figure 2:  
The immediate successor model  
of ‘sàsàfá’, along with its 3-factor ‘àsà’

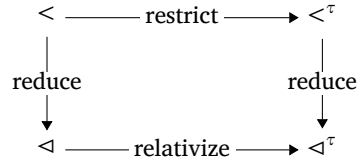


The general precedence relation can alternatively be refined to discuss a form of immediate adjacency relativized to certain unary relations in the signature. In particular, we can form relations between subsets of the alphabet, commonly called a *tier-alphabet*. For example, we may want to discuss the relations between only the sibilant elements present in a word, to the exclusion of all others. Similarly to how the successor relation is derived, we can restrict the precedence relation to the intended tier-alphabet  $\tau$  and first-order define a similar tier-successor relation  $<^\tau$ :

$$x <^\tau y := \tau(x) \wedge \tau(y) \wedge x < y \wedge (\forall z)[(\tau(z) \wedge x < z) \Rightarrow y \leq z].$$

Figure 3 depicts the relationships among these ordering relations.

Figure 3:  
Relationships between the general  
precedence relation and others  
first-order definable from it



Adjusting the model signature appropriately, shown below, we get a tier-based notion of structure, shown visually in Figure 4.

$$\mathcal{M}^{<^{\{s,f\}}}(w) = \langle \mathcal{D}_w; <_w^{\{s,f\}}, s_w, f_w, \acute{a}_w, \grave{a}_w \rangle.$$

Because the unary relations partition the domain elements, we can create a tier-adjacency relation for each element of the powerset of these relations. This merely amounts to adding tier-adjacency relations to the model signature to create a multi-tier signature. A model of the multi-tier relations is shown in Figure 5.

$$\mathcal{M}^{<^{\{s,f\}}, <^{\{\acute{a},\grave{a}\}}}(w) = \langle \mathcal{D}_w; <_w^{\{s,f\}}, <_w^{\{\acute{a},\grave{a}\}}, s_w, f_w, \acute{a}_w, \grave{a}_w \rangle.$$

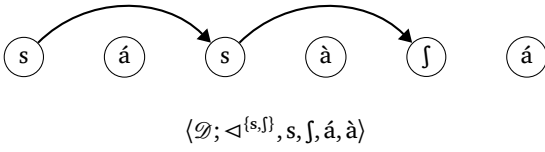


Figure 4:  
The tier-successor model of ‘sásàfá’ relativized over the set  $\tau = \{s, f\}$ , along with its only 3-factor ‘sfs’

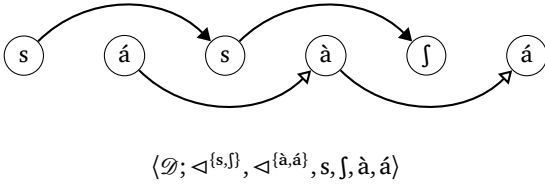
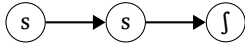


Figure 5:  
The multi-tier-successor model of ‘sásàfá’ relativized over the sets  $\{s, f\}$  and  $\{\grave{a}, \acute{a}\}$ , along with its only two 3-factors, ‘sfs’ and ‘ááá’



These four model signatures are by no means the only relational word models that may be considered. However, for the purposes of this paper we restrict ourselves to these signatures. Additionally, the definability of these signatures from other signatures leads to a general ability to define a notion of substructure, which we cover below.

### Windows and factors

2.1

Now that we have a general model-theoretic notion of structure, we would like a way to define certain parts of each structure, each of which is a structure in itself defined by the signature. Here, we generalize the method of Lambert and Rogers (2020) in defining these restrictions on models.

In order to pick out the subparts of a word model, we first pick out sets of elements that will define the substructure. Given a homogeneous relation  $R$  of arity  $a$ , the set

$$\mathcal{W}_a^R(m) := \left\{ \left\{ \langle x_i^i, x_{i+1}^{i+1} \rangle : 1 \leq i < a \right\} : \langle x_1, \dots, x_a \rangle \in R_m \right\}$$

is the set of  $a$ -windows over  $R$  in the context of the model  $m$ . These are merely directed acyclic graphs (represented by their edge sets alone)

constructed from the relations in  $R$ , such that each instance of a given domain element in the tuple is represented by a distinct node in the window, rather than merging all instances into a single node. Concretely, if 1 were a domain element and  $\langle 1, 1 \rangle$  an element of the relation, the corresponding 2-window would have two distinct nodes, both labelled by an index and the domain element 1:  $\langle 1^1, 1^2 \rangle$ . The set of windows of length greater than  $a$  is defined inductively by

$$\begin{aligned} \mathcal{W}_{k+1}^R(m) := & \left\{ A \cup \langle x_{a-1}^{j_{a-1}}, x_a^{k+1} \rangle : A \in \mathcal{W}_k^R(m) \text{ and } \langle x_1, \dots, x_a \rangle \in R \right. \\ & \text{and } \{j_1, \dots, j_{a-1}\} \subseteq \{1, \dots, k\} \\ & \text{and } \{ \langle x_i^{j_i}, x_{i+1}^{j_{i+1}} \rangle : 1 \leq i < a-1 \} \subseteq A \\ & \text{and } (\exists y, \ell) [ \langle x_{a-1}^{j_{a-1}}, y^\ell \rangle \in A \text{ or } \langle y^\ell, x_{a-1}^{j_{a-1}} \rangle \in A ] \\ & \left. \text{and } (\forall j_a \in \{1, \dots, k\}) [ \langle x_{a-1}^{j_{a-1}}, x_a^{j_a} \rangle \notin A ] \right\}. \end{aligned}$$

This means that for each  $k$ -window, we find a linear subgraph (a path) that maps to the initial  $a-1$  domain elements of one of the  $a$ -tuples that comprise  $R$  and add an edge from the final node of this path to a newly constructed node representing the final domain element from that tuple. The conditions are arranged in such a way that each iteration actually adds a new step to the path rather than simply repeating an older step, while still allowing cycles to be taken arbitrarily many times. Each of these larger windows can then be thought of as a graph of positions that are formed from a set of overlapping  $a$ -windows, which in turn are merely representations of tuples in the relation  $R$ . However, we may also wish to discuss a window which is of shorter length than the arity of the relation that defines it. To do so, we simply state that any connected subgraph of a window is itself a window.

For a given window  $x$  of a word model  $m$ , we define *the factor at  $x$*  (written  $\llbracket x \rrbracket_m$ ) as the restriction of  $m$  to the domain elements that occur in  $x$ . This lets us define the set of all  $k$ -factors of  $m$  as follows:

$$\mathcal{F}_k^R(m) := \{ \llbracket x \rrbracket_m : x \in \mathcal{W}_k^R(m) \}.$$

Note that a window is distinct from a factor in that the former is a graph of positions while the latter describes a word model whose domain consists of only a certain set of positions.

As example, consider the tier successor model of the word ‘sásàǎ́’ as above. Consider a 3-window  $x$  which contains all and only the domain elements  $\{1, 3, 5\}$ . Here, the restriction of the word model that defines this 3-factor is

$$\llbracket x \rrbracket_m = m \upharpoonright x = \langle \{1, 3, 5\}; \{\langle 1, 3 \rangle, \langle 3, 5 \rangle\}, \{1, 3\}, \{5\}, \emptyset, \emptyset \rangle.$$

Similar examples can be seen above in Figures 1–5. Various parallels emerge. The precedence word model contains a strict superset of the factors of every other word model we have considered. The tier-based and multi-tier-based word models have ‘sfs’ as a 3-factor, but the immediate successor model does not. On the other hand, ‘ásà’ is a 3-factor of only the precedence and immediate successor models. Only the precedence and multi-tier successor models have both ‘sfs’ and ‘ààà’ (a sequence of High-Low-High tone vowels) as 3-factors.

### *Anchored word models*

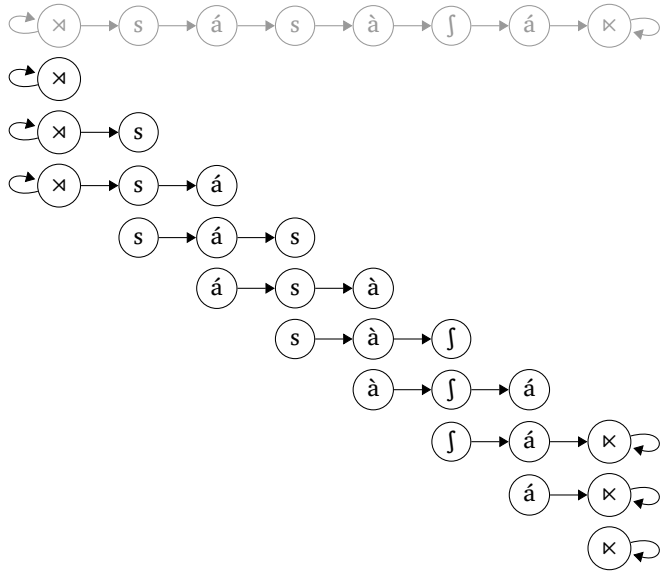
2.2

The word models considered up to this point do not encode domain boundaries explicitly. However, many prior treatments, including that of Lambert and Rogers (2020), explicitly assume such boundaries. One approach that has not been explicitly considered in this prior work is a model whose string yield is biinfinite. Here, left and right boundary symbols (labelled  $\bowtie$  and  $\bowtie$ , respectively) exist in the model and both participate in and are self-related under any ordering relations. This approach naturally captures words shorter than  $k$  symbols in its concept of a  $k$ -factor, without having to consider a union of smaller factor widths. The successor model for ‘sásàǎ́’ is shown along with each of its 3-factors in Figure 6.

The learning algorithms that we consider in this work are not bound to any particular model signature. Thus, we may consider the standard word models as shown in, for example, Figure 1, or we might consider these anchored word models.

This section has shown concretely how relational structures provide a uniform language for describing the structural information in representations of words. In this way, the differences between distinct subregular classes are isolated according to the relevant structural information. Also, the models considered in this section are just some

Figure 6:  
 The anchored word model under successor for ‘sásàfá’ along with each of its 3-factors. Note that every factor that includes a boundary symbol has an infinite yield. Those factors shorter than 3 symbols are formed from windows of length 3 that repeat the boundary symbols



of many models. The contents of each model signature clarify precisely what structural information the learner has immediate access to when making inferences during learning, described by the notion of a  $k$ -factor, and which it must computationally infer. For example, the non-local information that is immediately present in the precedence model requires more work in the successor model, its transitive reduct. These properties are encoded into the grammars being learned, and directly carve out the properties of classes of languages that result from a particular learning algorithm inferring such structures.

### 3 SPACE COMPLEXITY AND THE SUBREGULAR GRID

This section will introduce and examine four learning algorithms—algorithms I, II, III, and IV—where stringsets are learned in the limit from arbitrary positive data. Indeed, we will only be considering subclasses of a style of learning algorithm presented by Heinz (2010b, expanded upon by Heinz *et al.* 2012). We show that of these subclasses,



some require substantially more space than others to properly account for the distinctions that must be made in the course of learning, and we argue that this alone would cause linguistic typology to tend toward the simpler, less space-intensive classes.

First we briefly discuss some background from learning theory. Generally, our presentation follows the style of Gold (1967). While issues with this theoretical framework have been pointed out (Johnson 2004; Clark and Lappin 2011), these criticisms stem from misunderstandings (see Heinz 2016, and references therein). Gold's framework is the basis for much influential work on learning formal languages (Jain *et al.* 1999; Nowak *et al.* 2002; Niyogi 2006; de la Higuera 2010; Clark and Lappin 2011).

More importantly, however, the algorithms we present here are largely independent of the particular learning framework that we use to evaluate their behaviour. They can be studied with respect to the various identifiability-in-the-limit paradigms of Gold, but they can also be studied with respect to other paradigms (Mohri *et al.* 2012). For example, all of the algorithms presented here are not only identifiable in the limit from positive data in polynomial time, they are also PAC-learnable.<sup>2</sup> While the assumptions of PAC learning, including the use of negative evidence and approximate identification, seem to make the learning problem easier, in fact the conclusions show the learning problem is harder. For example, the finite languages, learnable in the limit from positive data, are not PAC learnable. Interestingly, not all PAC-learning algorithms even require negative evidence. The standard textbook examples of rectangles (Kearns and Vazirani 1994) and rays (Anthony and Biggs 1992) only use positive data just like our algorithms here. Despite these differences, both frameworks focus the learning problem on generalization which has led some researchers to provide a unified analysis of these different frameworks (Niyogi 2006). Nonetheless, irrespective of the framework, we demonstrate that the space complexity requirements are severe for algorithms III and IV, but not for algorithms I and II.

It is important to note that, while we present only four algorithms here that are sufficient to learn the well-understood subregular classes

---

<sup>2</sup>This is because when the parameters  $k$  (and  $t$ ) are fixed, the defined class has a finite VC dimension (since the class has finite cardinality) (Vapnik 1995).

under consideration, these are not the only possible algorithms. Others do exist and may well meet the criteria for these learning frameworks. The complexity results here are general, applying to any algorithm that can learn the classes, simply because they are based on the kinds of distinctions that must be made.

First, we describe the general learning setup. Let  $L$  be a set of strings drawn from  $\Sigma^*$  and let  $L_\odot$  represent  $L$  with an adjoined element  $\odot$ . An online learner is a function  $\varphi: \mathcal{G} \times L_\odot \rightarrow \mathcal{G}$ , where  $\mathcal{G}$  is some kind of grammar representation, a mechanism by which one can decide whether a given string is in  $L$ . In other words, an online learner begins with some guess as to what the grammar might be and updates this guess for each input word. Let  $\mathcal{L}: \mathcal{G} \rightarrow \mathcal{P}(\Sigma^*)$  be the function that maps a grammar to its extensions, the set of strings it represents. Two grammars  $G_1$  and  $G_2$  are equivalent ( $G_1 \equiv G_2$ ) iff they are extensionally equal, that is,  $\mathcal{L}(G_1) = \mathcal{L}(G_2)$ .

A text for  $L$  is a function  $t: \mathbb{N} \rightarrow L_\odot$ , a sequence of strings drawn from  $L$  or pauses in which data does not appear. Following traditional mathematical notation for sequences, we use  $t_n$  to represent  $t(n)$ . If  $\emptyset$  represents an initial guess at what the grammar might be, then the recursively-defined sequence

$$a_n(t) := \begin{cases} \emptyset & \text{if } n = 0 \\ \varphi(a_{n-1}, t_n) & \text{otherwise.} \end{cases}$$

represents the learning trajectory over a given text. Then given a text  $t$  for a language  $L$ , we say that a learning algorithm  $\varphi$  *converges* on  $t$  iff there is some  $i \in \mathbb{N}$  such that for all  $j > i$  it holds that  $a_j(t) \equiv a_i(t)$ . If for every possible text  $t$  over  $L$  it is the case that  $\varphi$  converges on  $t$  and  $\mathcal{L}(\lim_{n \rightarrow \infty} a_n(t)) = L$ , then we say  $\varphi$  converges on  $L$ . As a second lift, if for every stringset  $L$  in a class  $\mathbb{L}$  it is the case that  $\varphi$  converges on  $L$ , then we say  $\varphi$  converges on  $\mathbb{L}$ .

### 3.1

#### *String extension learning*

Heinz (2010b, expanded by Heinz *et al.* 2012) defined string extension learning, a general notion of learning from gathered substructures. Originally treated only as a batch learner, the online definition is trivial to derive. Given a function  $f: \mathcal{M} \rightarrow \mathcal{S}$  that extracts informational

content from a word model, where  $\mathcal{S}$  represents some notion of structural content, along with a combinator  $\oplus: \mathcal{G} \times \mathcal{S} \rightarrow \mathcal{G}$  that somehow informs the grammar of these structures, we define

$$\varphi(G, w) := \begin{cases} G & \text{if } w = \odot \\ G \oplus f(\mathcal{M}(w)) & \text{otherwise.} \end{cases}$$

In a simple case,  $\mathcal{G}$  and  $\mathcal{S}$  will be the same type, and  $\oplus$  will simply be set union, but this is not a necessary requirement.

Although the present discussion has been contextualized in the presence of a complete text, the algorithms can only ever operate on a finite sample. No infinite complete presentation is ever needed or even available. The analysis with complete texts guarantees that no matter the order of the input there is always some finite point in time, some finite sample, at which point every piece of informational content that could occur has occurred, and the algorithm will converge exactly to the target grammar (Heinz *et al.* 2012). For samples that do not meet this criterion, the smallest stringset in the target class that is consistent with the data will be learned instead of the target stringset itself (Heinz *et al.* 2012).

The space required by any string extension learning algorithm is bounded below by the output grammar size. This is dependent on the type of information that the grammar must retain. For the subsequent discussion, no additional space is necessary, so all that is relevant is the size of the grammar representation. Generally the worst case is when the target language is  $\Sigma^*$  and every factor, set, or multiset will need to be observed and stored.

### Learning with factors

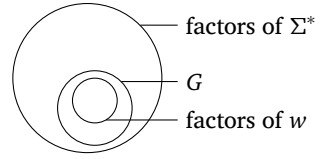
### 3.1.1

This simple case is exemplified by a learner that makes distinctions only between permitted and nonpermitted factors. This learner is parameterised by a factor width  $k$ . We have  $\mathcal{G} = \mathcal{S} = \mathcal{P}(\Sigma^k)$  and  $G \oplus S = G \cup S$ . The information extraction function is

$$f(m) := \mathcal{F}_k(m).$$

Upon convergence, a word  $w$  is accepted iff all of its factors occur in  $G$  as shown in Figure 7.

Figure 7:  
Grammars returned by Algorithm II accept all and only those strings  $w$  whose factors are all in  $G$

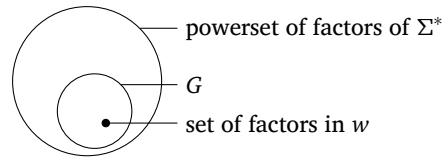


Since the grammar only needs to maintain a single merged set of attested factors, the space complexity for this class of learner is  $\mathcal{O}(|\Sigma|^k)$ . This will be referred to as Algorithm II. A variant, which will be Algorithm I, will be discussed in Section 3.1.4.

### 3.1.2 Learning with sets

The primary difference between learning with factors and learning with sets thereof is the grammar augmentation combinator. Rather than set union,  $G \oplus S = G \cup S$ , we have set insertion,  $G \oplus S = G \cup \{S\}$ . This of course means that  $\mathcal{G}$  and  $\mathcal{S}$  are no longer equal, with  $\mathcal{G}$  being the powerset  $\mathcal{P}(\mathcal{S})$ , adding a layer of structure. Upon convergence, a word  $w$  is accepted iff its set of factors is an element of  $G$  as shown in Figure 8. Since a given grammar is in this case a set of sets of factors, with this larger grammar the space complexity is  $\mathcal{O}(2^{|\Sigma|^k})$ . These set-based classes can make more distinctions than the purely factor-based classes, but this power comes at a cost. This is Algorithm III.

Figure 8:  
Grammars returned by Algorithm III accept all and only those strings  $w$  whose set of factors is an element of  $G$



### 3.1.3 Learning with multisets

A set is simply a structure that contains for each possible element a Boolean value describing whether or not that element is included. Given the natural isomorphism between the Booleans and the subset of  $\mathbb{N}$  consisting of 0 and 1, one might consider a natural expansion of this structure which denotes number of occurrences saturating not at 1 but at some arbitrary value  $t$ . (In other words,  $t$  is the largest number one can count to.) We can learn classes in which well-formedness is characterized by the saturating multisets of factors in a word as follows. With

$\mathcal{S} = \mathcal{P}(\Sigma^k \times \mathbb{N}_t)$  and  $\mathcal{G} = \mathcal{P}(\mathcal{S})$ , we can maintain from the set-based learner the augmentation combinator where  $G \oplus S = G \cup \{S\}$ . However, the function that extracts informational content must be modified to include the  $t$ -counts associated with a given factor as follows

$$f(m) := \{\llbracket x \rrbracket_m : x \in \mathcal{W}_k(m)\}_t.$$

The notation  $\{\dots\}_t$  represents a multiset that saturates at a count of  $t$ . Note that this parallels the window-based definition of factors in a model, except that a saturating multiset is formed rather than merely a set. Upon convergence, a word  $w$  is accepted iff its saturating multiset of factors is an element of  $G$  as shown in Figure 9. The space complexity here is much like that of Algorithm III, except that the base of the exponent is changed to correspond with the number of values each factor may be associated with:  $\mathcal{O}((t+1)^{|\Sigma|^k})$ . This is Algorithm IV. Using this algorithm with  $t = 1$  is equivalent in every way to Algorithm III, so in fact there are only three algorithms under discussion. That said, we will retain this separation for the current discussion.

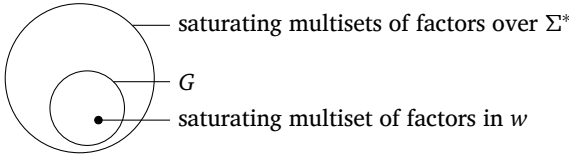


Figure 9:  
Grammars returned by Algorithm IV accept all and only those strings  $w$  whose saturating multiset of factors is an element of  $G$

### Learning with factors, revisited

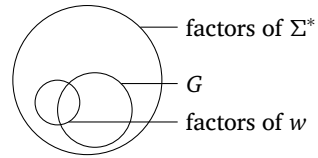
### 3.1.4

A variant of Algorithm II ignores all input words longer than  $k$  symbols. The only difference then is the information extraction function

$$f(m) := \begin{cases} m & \text{if } |m| \leq k \\ \emptyset & \text{otherwise.} \end{cases}$$

Upon convergence, a word  $w$  is accepted iff it contains a factor that also occur in  $G$  as shown in Figure 10. This is Algorithm I. Notably, using the anchored word models with this algorithm produces only finite languages. In contrast to the other algorithms, translating such models into unanchored ones provides an increase in expressive power.

Figure 10:  
Grammars returned by Algorithm I accept  
all and only those strings  $w$   
whose sets of factors are not disjoint with  $G$



3.1.5

Illustration

Considering a standard unanchored word model, with the algorithmic parameters  $k$  and  $t$  both set to 2, Table 1 represents the outputs of these four learning algorithms after seeing the single word ‘aaaab’. Notably, this word is not short enough to inform Algorithm I of anything. Also, despite the fact that ‘aa’ occurs as a substring three distinct times, Algorithm IV saturates at a count of 2 under these assumed parameters.

Table 1:  
Encountering the single word ‘aaaab’ with each learner

Algorithm	Resulting Grammar
I	$\emptyset$
II	{aa, ab}
III	{{aa, ab}}
IV	{{⟨aa, 2⟩, ⟨ab, 1⟩}}

3.2

The grid

These learning algorithms are model-agnostic. As long as there exists some way to extract windows or factors (*i.e.*, substructures) from a model, the algorithms will work with that. When allowed to range over selected model signatures, the classes learned by each algorithm are shown in Figure 11. Each of the cells of the grid represent a particular class of languages. For example, the strictly local class contains languages for which no word may contain any of a finite set of local factors.

For clarity, we restrict our discussion of the fifteen classes present in the subregular Grid to the Appendix. There, we provide a brief description of the class, as well as a sample of attested linguistic patterns that it accounts for, as well as an interpretable implementation of the grammar for each of those patterns.

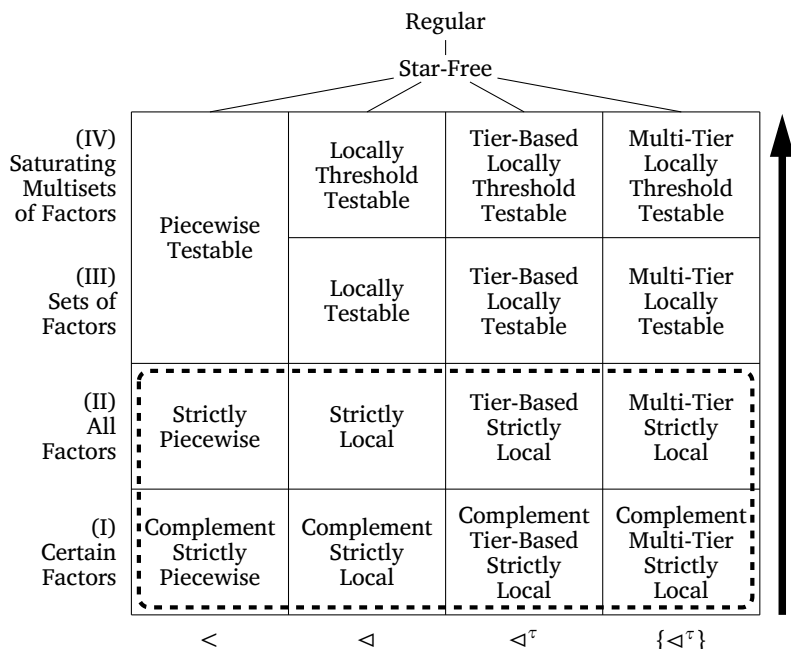


Figure 11: The subregular classes. Learning difficulty increases along the vertical axis. The horizontal axis is categorical, describing the type of substructure. The dotted line indicates the predicted region of phonological typology

Note that Algorithm I learns only strong classes (those in which domain boundaries, i.e. anchors, are unreferenced), while the others do not have this restriction. In the Piecewise case, where the model signature contains the general precedence relation ( $<$ ), the strong classes are equivalent to the general classes and this distinction is irrelevant.

We note that the amount of space required to store the grammar is fairly large for any of these algorithms. But Algorithms III and IV require exponentially more space than I and II. These space requirements are shown in Figure 12, where it is apparent that even on a binary alphabet, the smallest possible nontrivial alphabet, the Locally 5-Testable class of languages, for example, requires more storage space than there are synapses in an average human being (Azevedo *et al.* 2009; Herculano-Houzel and Lent 2005). With the larger alphabet sizes commonly encountered in natural language the restrictions become even tighter. The interested reader could as an exercise consider how this graph would change if the size of the alphabet were around

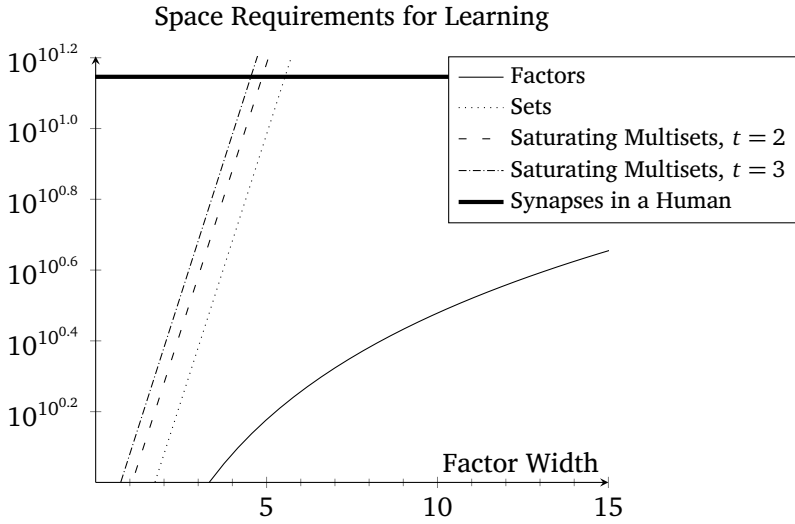


Figure 12: While gathering factors requires space exponential in terms of factor width, the requirements are doubly exponential for any of the larger structures we might employ. Here the space requirements are shown for just a binary alphabet

30 or 40, the average phoneme count in languages of the world. However, it should be noted that attempting to plot this graph for an alphabet of size 10 exceeded the numerical range of our plotting software.

Due to the enormous space requirements in terms of alphabet size of the higher-numbered algorithms, it seems in general unfeasible to learn patterns that lie strictly in their corresponding classes. That is, purely from learnability considerations alone, we would expect the typology of patterns in natural language to lie primarily within the region spanned by Algorithms I and II. This region is highlighted in Figure 11 by a dotted line. Further, we would expect any attested patterns to require relatively small values for the factor width parameter  $k$ , since that is the exponent of these singly and doubly exponential space complexities. This constraint on the learning algorithms is again agnostic to the representation, showing that the way the learner collects and stores the data matters.

The multiple-tier-based classes also require significant space, but in a different way. The classes of Algorithms III and IV admit exponentially more distinctions than Algorithms I and II, and thus require exponentially more space. The multiple-tier-based classes in contrast



require many grammars of the same type: one per subset of  $\Sigma$ . This scales the space requirements by a multiplicative factor of  $2^{|\Sigma|}$ . This difference, while less pronounced, is still significant and will be taken into account later.

To briefly sum up, this section presented a suite of online learning algorithms that extract structural information based on the particular representations it is given. The combination of a particular algorithm and a model-theoretic signature define a range of classes of languages that can be learned. One model signature may be used by any of the learning algorithms, and any of the algorithms may use any of the signatures. In this way, we have organized the space of possible generalizations the typology can inhabit, which ultimately amounts to possible restrictions on the capacity of the learner. This provides a unifying perspective on previously studied subregular classes. Another contribution of this section is the introduction of Algorithm I, which naturally leads to “Complement” classes of the “Strict” ones.

However, there is a strong divergence between the space requirements of the two algorithms that make distinctions based solely on the presence of individual factors and those two algorithms that make distinctions based on sets or multisets of factors. For a feasible learner, then, it is advantageous to disprefer learning strategies that rely on an ability to make as many distinctions as these two more complex algorithms allow. Drawing a boundary for the language classes learned by the two simpler algorithms, we significantly reduce the possible typology available to the learner. Can there be any other restrictions? This is the topic of the next section.

## COMPLEXITY IN TIME

4

The Strictly Local (SL) class (McNaughton and Papert 1971) is learned by gathering the factors of simple adjacency. Under such a model, there exists at most a single window of size  $k$  at any given point. Thus for each index in the word, we can simply insert the contents of this single window into the grammar. Including the time it takes to insert a factor into a set, the class is learnable in  $\mathcal{O}(nk \log |\Sigma|)$  time for input of size  $n$ , and since  $\Sigma$  and  $k$  are assumed constant this amounts to linear

time. As discussed, this Algorithm II learner also uses constant space that is but singly exponential in the width of the factors.

For the Tier-Based Strictly Local (TSL) class (Heinz *et al.* 2011, see also Lambert and Rogers 2020), if the tier alphabet  $\tau$  is known, then this approach applies directly to the projection of the word to  $\tau$ . But generally we assume that  $\tau$  is not known, and one might initially assume that a learner might need to construct grammars for all possibilities, which would result in increased resource requirements, be that in terms of time, space, or both. Per Jardine and McMullin (2017), maintaining the factors of width bounded above by  $k + 1$  is sufficient to determine the value of  $\tau$ . But their approach seems to require a batch approach, first deciding the value of the  $\tau$  parameter and then processing the (projections of) the input as for the Strictly Local class. But it turns out that, due to (inverse-)projection closure and the fact that in the Gold framework we assume a complete text, we can guarantee that any substring whose projection will appear on the tier will itself appear as a substring in some word. Since we still need to determine the value of  $\tau$ , we do still require the factors of width bounded above by  $k + 1$ , but nothing more. The exact learning algorithm used for  $SL_{k+1}$  will produce a grammar for  $TSL_k$ , and only the interpretation of the result is changed (Lambert 2021). These same properties hold true for the relativized variants of the Locally Testable (LT) (McNaughton and Papert 1971) and Locally Threshold Testable (LTT) (Beauquier and Pin 1989) classes as well, where the corresponding adjacency-based learners suffice to learn the relativized-adjacency classes (Lambert 2021).

The Strictly Piecewise (SP) class (Rogers *et al.* 2010, see also Haines 1969) is similar in that, one might expect a time complexity on the order of  $\mathcal{O}(n^k)$  to find all of the subsequences of each word. Heinz and Rogers (2013) show that in fact a factored approach can use simply  $\mathcal{O}(n|\Sigma|^k)$ , but we can reduce this even further by taking advantage of this same property. Given a complete text, every attested subsequence will eventually occur as a substring due to the SP stringsets' closure under deletion. Again then, the same learning algorithm used for  $SL_k$  will produce a grammar for  $SP_k$  as well, where the difference lies only in interpretation.

Given this ability to learn the SP, SL, and TSL classes in linear time and in space only singly exponential in factor width, we can mod-

ify Figure 11 to indicate the boundary between the classes that are learnable within these resource bounds and those that are not. This boundary is indicated by a thick line in Figure 13, which also uses dashed lines to indicate where one algorithm may be used for multiple distinct classes. As discussed in Section 3.2, the multiple-tier-based classes do not fit within this low-resource region because, in general, exponentially many grammars must be learned.

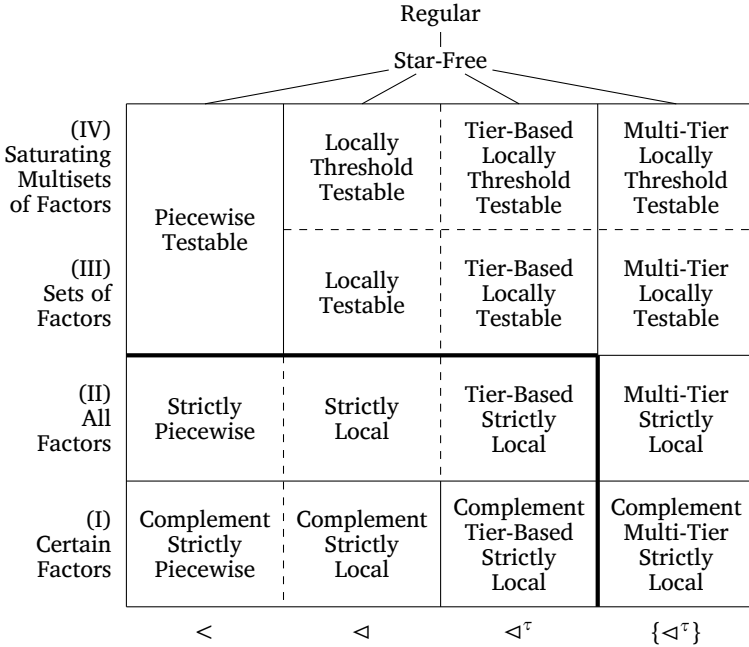


Figure 13: Dashed lines indicate that the classes on either side can be learned by exactly the same algorithm. The thick solid line denotes the barrier between linear-time  $\mathcal{O}(|\Sigma|^k)$ -space learning and more resource-intensive learning

Considering only the SP class of stringsets, there are at least three online learning algorithms of various complexities:

- Gather all factors under general precedence of each word.
  - $\mathcal{O}(n^k k \log |\Sigma|)$  time
  - $\mathcal{O}(|\Sigma|^k)$  space
  - Learns an SP least upper bound (lub) of the source text.

- Use factored learning as per Heinz and Rogers (2013).
  - $\mathcal{O}(n|\Sigma|^k)$  time
  - $\mathcal{O}(|\Sigma|^k)$  space
  - Learns an SP lub of the source text.
- Gather all adjacency factors of each word.
  - $\mathcal{O}(nk \log|\Sigma|)$  time
  - $\mathcal{O}(|\Sigma|^k)$  space
  - Only works if every permitted subsequence eventually occurs adjacently, which holds for SP targets.

One caveat is that these optimizations of the learning algorithms for the SP and TSL classes rely on certain properties of the input stringset. The nonoptimized variants are guaranteed to learn a smallest in-class superset of the input stringset, a property which is lost in this optimization. For example, a long distance sibilant harmony constraint (Heinz 2010a) will not be learned by the optimized SP learner if the text is drawn from a language that exemplifies both this constraint and a CV syllable structure, even though it would be learned by the nonoptimized variant. Other examples of this type may be found in the Appendix. This prompts a question regarding the learnability consequences of constraint interaction.

## 5

## LEARNING INTERACTIONS

Most natural languages are describable not by a single subregular class but by an interaction of constraints from multiple such classes. The interaction of constraints from different classes might influence the learnability of each constraint individually, in which case time or space tradeoffs might be necessary.

For example, we might consider the default-to-opposite stress pattern of Chuvash (Krueger 1961), where primary stress falls on the rightmost heavy syllable if there is one, or on the initial syllable otherwise. One way of describing this invokes the conjunction of two constraints from two different classes, namely an SP constraint detailing a lack of:

- a heavy syllable anywhere after a stressed syllable,
- a stressed light syllable after any other syllable, or
- two stressed syllables in the same word,

and a coSP constraint that states that every word contains some stressed syllable.

The requirement that some stress must occur does not affect which substrings may appear in a word, and so the SP constraint may be learned by any of the three algorithms that have been discussed for that class so far, including the optimized substring-based learner. Further, the precedence restrictions do not prevent seeing the two words (light and heavy stressed monosyllables) required to learn the stress requirement. In other words, these constraints interact in such a way that learning is not hindered. This is not always the case.

Consider now the sibilant harmony of Samala (Applegate 1972), in which as ‘s’ and an ‘j’ may not appear in the same word. Since this constraint acts on the segment level rather than the syllable level, we might assume that it is isolated from any kind of stress constraint. But other segment-level constraints will certainly have the possibility of interaction. For example, imposing a CV syllable pattern restricts the substrings that may occur, in such a way that using an SL learner to infer the SP constraints is not a possibility. This means that one has to decide among the other possible SP learning algorithms, where time or space tradeoffs must be made.

In contrast, a tone plateauing constraint like that which occurs in Luganda (Hyman and Katamba 1993) is  $SP_3$ , which means that it could be learned directly alongside this sibilant harmony constraint without fear of interaction effects. Note that the word ‘sàsàfá’ that has been our running example violates both the harmony constraint and the tone plateauing constraint.

Given our space-based learnability considerations, we would assume that Algorithms III or IV are not practically learnable and would likely be unattested. In other words, we would expect linguistic typology to inhabit only the lower regions of the hierarchy, or at least be biased heavily toward this region. Rogers and Lambert (2019b) provide strong evidence that this is in fact the case when it comes to stress patterns. Their exhaustive analysis of the more than one hundred stress patterns in the StressTyp2 database (Goedemans *et al.* 2015) showed

that each of these can be described as the interaction of constraints that can be learned by Algorithms I and II.

## 6 MODEL-THEORETIC REPRESENTATIONS OF NONLINEAR STRUCTURES

This model-theoretic formulation provides a distinct advantage when applied to various linguistic objects. It allows one to characterize the content of a particular linguistic representation, and in so doing, immediately guarantee that there are learning algorithms which can describe various constraints over those representations. This is important, because work describing nonlinear structures in syntax and phonology has proceeded in an ad-hoc way, by first defining constraints, and working backwards to the representations, often without any learning algorithms at all, or ones relativized to a particular structure.

The previous sections used various model signatures that characterized information based on a string data structure. This is because the subregular classes that were the central motivation for this paper are defined over strings, or model signatures based on strings, in the work of Büchi, Thomas, and others. The constructions considered to this point are not restricted to simple string models. Without modification, the algorithms may be applied to any relational model at all. They in fact apply to any structure that can be characterized as a graph. In this sense, strings are a special case, but the distinctions that each of the four learning algorithms pick out carry over onto these more general factors as well. In this section, we discuss some other linguistically-motivated models that one might consider.

### 6.1 *Autosegmental graphs*

An example of a nonlinear structure where the graph perspective is clearly relevant to linguistic research concerns autosegmental representations in phonology. Graphs were proposed to handle a variety of

prosodic phenomena for which the string-based perspective was inadequate. Phonological processes affecting domains larger than two adjacent segments, such as tonal alternations in tonal languages, have temporal properties that do not always map consistently onto discrete vowel segments in a one-to-one fashion (Goldsmith 1976; Williams 1976). Goldsmith introduced a model of the phonological word where tonal features formed an independent string from the segmental string, called a tier. Segments on the two strings are linked via many-to-one relations, turning the structure into a graph.

In practice, encoding these adjustments into a word model involves adding more relational structure. Jardine (2017a,b, 2019) uses a binary relation  $\alpha(x, y)$  to encode the association relation between autosegmental tiers. Augmenting the successor model signature used throughout this paper gives a signature as

$$\mathcal{M}^{\alpha, \triangleleft}(w) = \langle \mathcal{D}_w; \alpha_w, \triangleleft_w, s_w, \int_w, a_w, H_w, L_w \rangle.$$

Here, the domain is increased to accommodate the new autosegments, and the successor relation holds between elements on both tiers. The unary relations encoding vowels with tonal features have been split, into a relation ‘a’ for vowel information, and distinct ‘H’ and ‘L’ relations for tonal information. Under this signature, a word model for the example ‘sàsàfá’ is given in Figure 14.

Our notion of a factor is exactly a notion of a subgraph. The previous section showcased how this word violates a constraint on tone plateauing. The autosegmental model makes this information immediately accessible by encoding the ‘HLH’ structure as its own subgraph, shown on the bottom of Figure 14. Thus, the permissibility of tone sequences is liberated from the segmental elements that carry them.

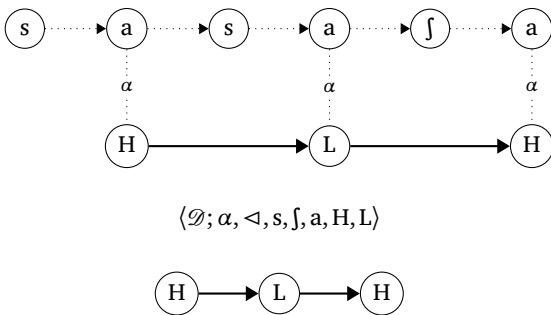


Figure 14:  
The autosegmental successor model of ‘sàsàfá’, along with its 3-factor ‘HLH’. The  $\alpha$  relation is shown without tips because it is symmetric

6.2

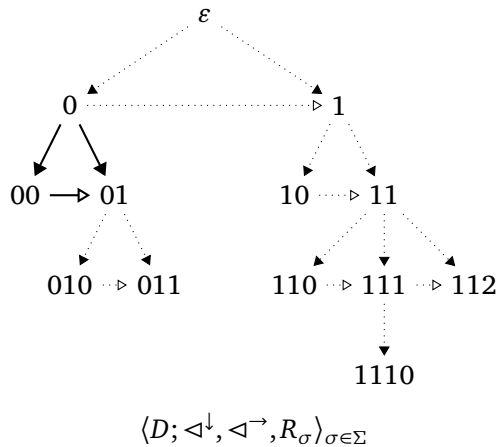
Tree models

The model-theoretic framework also allows describing tree structures (Rogers 1998), and opens the door to study parallels between phonological and syntactic constraints (Graf 2014). Rogers (2003) describes a model-theoretic characterization of trees of arbitrary dimensionality. In this framework, we specify the domain  $\mathcal{D}$  as a Gorn tree domain (Gorn 1967). This is a hereditarily prefix closed set  $D$  of node addresses, that is to say, for every  $d \in D$  with  $d = ai$ , it holds that  $a \in D$ , and if  $i \neq 0$  then  $a(i-1) \in D$  as well. In this view, a string may be called a one-dimensional or unary-branching tree, since it has one axis along which its nodes are ordered. In a standard tree, on the other hand, the set of nodes is ordered as above by two relations, “dominance” and “immediate right-of”. Suppose  $s$  is the mother of two nodes  $t$  and  $u$  in some standard tree, and also assume that  $t$  precedes  $u$ . Then we might say that  $s$  dominates the string  $tu$ .

While a Gorn tree domain as written encodes these dominance and adjacency relations implicitly, we may explicitly write them out model-theoretically so that a signature for a  $\Sigma$ -labelled two-dimensional tree  $T$  is  $\mathcal{M}^{\triangleleft^\downarrow, \triangleleft^\rightarrow} = \langle D; \triangleleft^\downarrow, \triangleleft^\rightarrow, R_\sigma \rangle_{\sigma \in \Sigma}$  where  $\triangleleft^\downarrow$  is the immediate dominance relation and  $\triangleleft^\rightarrow$  is the immediate right-of relation (see Figure 15). Model signatures that include the transitive closures of each of these relations have also been studied. Additionally, the anchored word models considered above for strings lift naturally

Figure 15:

A tree model. Nodes are organised by immediate dominance (black tip) and immediate right-of (white tip) relations. Labelling relations are omitted to show Gorn addresses. All edges are shown, with a particular factor noted with solid thick lines





to trees, where a root node is an anchor and each leaf is a separate anchor, or there is a single additional node which serves as the anchor for every leaf .

Recent work in syntax has synthesised the model-theoretic approach to trees with insights from the subregular approach to phonology. For instance, Graf and Shafiei (2019) hypothesise that the TSL class is sufficient to characterize syntactic constraints.

To sum up, this section has shown how the model-theoretic representations presented in Section 2 naturally apply to other linguistic representations.

## FURTHER READING

7

The subregular classes considered here have been widely studied for decades. McNaughton and Papert (1971) introduce the Local hierarchy, with Beauquier and Pin (1989) adding the Locally Threshold Testable class. The Piecewise branch of the hierarchy stems from Simon (1975), with the Strictly Piecewise class only being integrated into the hierarchy in 2010 by Rogers *et al.* (Languages closed under subsequence had been discussed by Haines 1969, though not in connection with other subregular classes.) The Tier-Based Strictly Local class was introduced by Heinz *et al.* (2011) and extended in various ways by De Santo and Graf (2019), Lambert and Rogers (2020), and Lambert (2021). Recent work in syntax has synthesized the model-theoretic perspective on trees with insights from the subregular program (Graf and Shafiei 2019; Graf 2020, 2014)

Provided a finite-state automaton, Caron (1998, see also Caron 2000) describe algorithms that decide whether the corresponding language is Locally or Piecewise Testable. An efficient algorithm for deciding SL is described by Edlefsen *et al.* (2008). Algorithms that extract SL and SP factors from a given language (and thus can also be used to decide class membership) are due to Rogers and Lambert (2019a), and these were extended to the TSL class by Lambert and Rogers (2020).

While this paper has so far focused on constraints, this work is easily extended to consider mappings between structures, expressed mathematically as Regular functions (Courcelle 1994; Courcelle and

Engelfriet 2012; Filiot 2015; Engelfriet and Hoogeboom 2001). The notion of strict locality has been generalized to functions and shown to be relevant for natural language phonology and morphology (Chandlee 2014, 2017). These local functions have been model-theoretically characterized and extended to consider nonlinear structures in phonology (Chandlee and Jardine 2019; Strother-Garcia 2019). Relativizing input representations to consider multi-arity functions allows a notion of strictly local transducers expressed using multi-tape automata (Rawski and Dolatian 2020; Dolatian and Rawski 2020). Expressed as functions, these subregular characterizations have been extended to consider continuous functions over vector spaces and learning algorithms operating over them (Rawski 2019; Nelson *et al.* 2020).

There exist other learning algorithms alongside the string extension learners of Heinz (2010b) and Heinz *et al.* (2012). Garcia *et al.* (1990) demonstrate the learnability of SL. Heinz and Rogers (2013) provide learning algorithms for the SL and SP classes as well as their Testable correlates. Other approaches have directly incorporated phonological features into the models (Vu *et al.* 2018; Chandlee *et al.* 2019). Learning of TSL classes has been discussed by Jardine and Heinz (2016) and Jardine and McMullin (2017), while online learners for this class and the remaining single-tier-based hierarchy were proposed by Lambert (2021).

## 8

## CONCLUSION

This paper showed how the nature of phonological typology emerges from simple representations and inference strategies. We discussed the nature of these representations in model-theoretic terms, forming a general notion of structural information (factors) that characterizes virtually any linguistic representation, from strings, to trees, to graphs. We also discussed a series of learning algorithms that work over any form of these factors, and are organised into a hierarchy of space complexity based on the distinctions they make with respect to structural information. We then derived the full hierarchical range of subregular formal language classes from the product of these different representations and inference strategies. Consideration of time complexity

further parameterises this hierarchy, drawing equivalences and distinctions amongst the classes with respect to learning. We find that the scope of phonological typology is strongly biased into the range defined by the simplest learning algorithms and representations.

The relevance of these results for linguistic theory is clear. A learner, faced with dramatically sparse data, favours grammar induction strategies that limit the amount of necessary distinctions between structural forms in order to ensure that learning is possible and feasible. The requirement for learners to structure and limit their hypothesis spaces plays off the distinctions learners make and the representations they make them over. The results here, as well as typological and experimental evidence, suggest that a learner may fix a learning algorithm and allow representational primitives to vary. From this perspective, the requirement of parsing from a linguistic input to a particular linguistic form is of the utmost importance. Linguistic learning can be relativized over various representations, be they strings or graphs for phonology, or trees for syntax. In this way, natural language typology, considered through an algorithmic lens, can be shown to emerge from the interaction of simple learning algorithms and simple but wide-ranging notions of representation.

## APPENDIX

9

This appendix offers a brief reference to the fifteen classes characterized by combinations of the learning algorithms and model signatures described in this text. Each class is accompanied by a sample of attested patterns that it can account for, with those accessible to a lower algorithm having backreferences. Each pattern is provided with a grammar given as a *plebby*-style expression<sup>3</sup> formatted in a way typical of the class.

---

<sup>3</sup>The Piecewise-Local Expression Builder Interpreter (*plebby*) is one component of the Language Toolkit, available from <https://github.com/vvulpes0/Language-Toolkit-2>, currently version 0.3.

## 9.1

*Expression syntax*

A complete formal description of the *plebby* expression language is available in the package documentation. An abridged summary follows here.

Expressions are built on factors, represented by sequences between angle brackets. For example  $\langle a\ b,\ c\ d \rangle$  asserts the occurrence of four positions, say 1, 2, 3, and 4, that are respectively labelled by symbols in sets a, b, c, and d, where positions 1 and 2 are connected by the successor relation, as are positions 3 and 4, while positions 2 and 3 are connected by the general precedence relation. If a left (right) boundary symbol is prefixed to this notation, that means the leftmost (rightmost) position aligns with the left (right) edge of the word. For instance,  $\bowtie\langle a \rangle$  asserts that all words consist of a single position labelled by an element of the symbolset a. Assignment of names to symbolsets is not discussed here.

More complex expressions are built from unary ( $\otimes e$ ) or  $n$ -ary ( $\otimes\{e_1, e_2, \dots, e_n\}$ ) operations, where  $\otimes$  is the operator and the various  $e$  are expressions. The Boolean ‘and’ ( $\cap$ ) and ‘or’ ( $\cup$ ) operations are  $n$ -ary and represent language intersection and union, respectively. The other  $n$ -ary operation is concatenation ( $\bullet$ ). Complement ( $\neg$ ) and projection ( $[s_1, s_2, \dots, s_n]$ ) are unary operations, where the projection operation asserts that the subexpression it operates over applies after a word has been projected to include only symbols in the union of symbolsets  $s_1$  through  $s_n$ .

## 9.2

*Algorithm I*

Words must contain at least one element of some finite set of factors.

## 9.2.1

## Complement Strictly Piecewise

Factors are subsequences.

- Minimum word length:  $\cup\{\langle \acute{\sigma}, \acute{\sigma} \rangle\}$ .  
Two syllables. More or fewer by adding or removing  $\acute{\sigma}$ .
- Stress obligatoriness (Hyman 2009):  $\cup\{\langle \acute{\sigma} \rangle\}$ .

Complement Strictly Local 9.2.2

Factors are substrings.

- Minimum word length:  $\cup\{\langle \bar{\sigma} \bar{\sigma} \rangle\}$ .  
Two syllables. More or fewer by adding or removing  $\bar{\sigma}$ .
- Stress obligatoriness:  $\cup\{\langle \acute{\sigma} \rangle\}$ .

Complement Tier-Based Strictly Local 9.2.3

Factors are substrings after projection to some subset of the alphabet.

- Anything complement strictly local.

Complement Multi-Tier Strictly Local 9.2.4

All words must satisfy at least one of a set of complement tier-based strictly local grammars.

- Anything complement tier-based strictly local.

*Algorithm II* 9.3

No word may contain any of a finite set of factors.

Strictly Piecewise 9.3.1

Factors are subsequences.

- Harmony, unblocked (Heinz 2010a):  $\neg\cup\{\langle s, \int \rangle, \langle \int, s \rangle\}$ .  
Symmetric. Asymmetric if only one factor were included.
- Stress culminativity (Hyman 2009):  $\neg\cup\{\langle \acute{\sigma}, \acute{\sigma} \rangle\}$ .
- Tone Plateauing (Hyman and Katamba 1993):  $\neg\cup\{\langle H, L, H \rangle\}$ .

Strictly Local 9.3.2

Factors are substrings.

- AB alternation:  $\neg\cup\{\langle A A \rangle, \langle B B \rangle\}$ .
- Cambodian stress (Lambert and Rogers 2019):  
 $\neg\cup\{\times\langle \bar{\sigma} \rangle, \times\langle \rangle, \langle \acute{\sigma} \bar{\sigma} \rangle, \langle H \rangle, \langle \bar{L} \bar{L} \rangle, \times\langle \bar{L} \rangle\}$ .
- No light monosyllables (Lambert and Rogers 2019):  $\neg\cup\{\times\langle \bar{L} \rangle\}$ .

9.3.3 Tier-Based Strictly Local

Factors are substrings after projection to some subset of the alphabet.

- Anything strictly local.
- Dissimilation (Heinz *et al.* 2011):  $[k, l, r] \neg \cup \{ \langle l l \rangle, \langle r r \rangle \}$ .
- Harmony (Heinz 2010a):  $[s, \int] \neg \cup \{ \langle s \int \rangle, \langle \int s \rangle \}$ .  
Unblocked. Blocked if other symbols project.
- Stress culminativity:  $[\acute{o}] \neg \cup \{ \langle \acute{o} \acute{o} \rangle \}$ .

9.3.4 Multi-Tier Strictly Local

Words must satisfy each member of a set of tier-based strictly local grammars.

- Anything tier-based strictly local.
- Bukusu harmony (Aksénova *et al.* 2020):  
 $\cap \{ [vowel] \neg \cup \{ \langle hi lo \rangle, \langle lo hi \rangle \}, [l, r] \neg \cup \{ \langle r l \rangle \} \}$ .

9.4 *Algorithm III*

The set of factors in a word must be a member of some finite set of factorsets.

9.4.1 Piecewise Testable

Factors are subsequences.

- Anything (complement) strictly piecewise.
- No light monosyllables:  $\cup \{ \neg \langle \acute{L} \rangle, \langle \acute{\sigma}^*, \acute{\sigma}^* \rangle \}$ .  
See also strictly local, Algorithm II.

9.4.2 Locally Testable

Factors are substrings.

- Anything (complement) strictly local.
- Harmony, unblocked, symmetric:  $\neg \cap \{ \langle s \rangle, \langle \int \rangle \}$ .  
See also strictly piecewise, Algorithm II.

Tier-Based Locally Testable 9.4.3

Factors are substrings after projection to some subset of the alphabet.

- Anything (complement) tier-based strictly local.
- Anything locally testable.

Multi-Tier Locally Testable 9.4.4

Words must satisfy a Boolean network of tier-based locally testable grammars.

- Anything (complement) multi-tier strictly local.
- Anything tier-based locally testable.

*Algorithm IV* 9.5

The multiset of factors in a word must be a member of some finite set of multisets of factors. Note that while *plebby* has no intrinsic notion of multisets, concatenation can be used as in the expression  $\bullet\{\langle a b \rangle, \langle a b \rangle\}$  which asserts that  $\langle a b \rangle$  occurs at least twice.

Locally Threshold Testable 9.5.1

Factors are substrings.

- Anything locally testable.
- Stress culminativity:  $\neg \cup \{\bullet\{\langle \acute{o} \rangle, \langle \acute{o} \rangle\}\}$ .  
See also strictly piecewise, Algorithm II.

Tier-Based Locally Threshold Testable 9.5.2

Factors are substrings after projection to some subset of the alphabet.

- Anything tier-based locally testable.
- Anything locally threshold testable.
- Tone plateauing:  $[H, L] \neg \cup \{\bullet\{\langle H L \rangle, \langle H L \rangle\}, \cap \{\langle H L \rangle, \times \langle H \rangle\}\}$ .  
See also strictly piecewise, Algorithm II.

Words must satisfy a Boolean network of tier-based locally threshold testable grammars.

- Anything multi-tier locally testable.
- Anything tier-based locally threshold testable.

## REFERENCES

Alëna AKSËNOVA, Jonathan RAWSKI, Thomas GRAF, and Jeffrey HEINZ (2020), The Computational Power of Harmony, in Harry VAN DER HULST, editor, *Oxford Handbook of Vowel Harmony*, Oxford University Press, under review.

Martin ANTHONY and Norman BIGGS (1992), *Computational Learning Theory*, Cambridge University Press.

Richard Brian APPLGATE (1972), *Ineseño Chumash Grammar*, Ph.D. thesis, University of California, Berkeley.

Frederico Augusto Casarsa AZEVEDO, Ludmila Ribeiro Bezerra CARVALHO, Lea Tenenholz GRINBERG, José Marcelo FARFEL, Renata Eloah de Lucena FERRETTI, Renata Elaine Paraizo LEITE, Wilson Jacob FILHO, Roberto LENT, and Suzana HERCULANO-HOUZEL (2009), Equal Numbers of Neuronal and Nonneuronal Cells Make the Human Brain an Isometrically Scaled-Up Primate Brain, *The Journal of Comparative Neurology*, 513:532–541, doi:10.1002/cne.21974.

Danièle BEAUQUIER and Jean-Éric PIN (1989), Factors of Words, in Giorgio AUSIELLO, Mariangiola DEZANI-CIANCAGLINI, and Simonetta RONCHI DELLA ROCCA, editors, *Automata, Languages and Programming: 16th International Colloquium*, volume 372 of *Lecture Notes in Computer Science*, pp. 63–79, Springer Berlin / Heidelberg, doi:10.1007/BFb0035752.

Julius Richard BÜCHI (1960), Weak Second-Order Arithmetic and Finite Automata, *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 6(1–6):66–92, doi:10.1002/malq.19600060105.

Pascal CARON (1998), LANGUAGE: A Maple Package for Automaton Characterization of Regular Languages, in Derick WOOD and Sheng YU, editors, *Automata Implementation*, volume 1436 of *Lecture Notes in Computer Science*, pp. 46–55, Springer Berlin / Heidelberg, doi:10.1007/BFb0031380.

Pascal CARON (2000), Families of Locally Testable Languages, *Theoretical Computer Science*, 242(1–2):361–376, doi:10.1016/S0304-3975(98)00332-6.



Jane CHANDLEE (2014), *Strictly Local Phonological Processes*, Ph.D. thesis, University of Delaware, [https://chandlee.sites.haverford.edu/wp-content/uploads/2015/05/Chandlee\\_dissertation\\_2014.pdf](https://chandlee.sites.haverford.edu/wp-content/uploads/2015/05/Chandlee_dissertation_2014.pdf).

Jane CHANDLEE (2017), Computational Locality in Morphological Maps, *Morphology*, 27(4):599–641, doi:10.1007/s11525-017-9316-9.

Jane CHANDLEE, Rémi EYRAUD, Jeffrey HEINZ, Adam JARDINE, and Jonathan RAWSKI (2019), Learning with Partially Ordered Representations, in *Proceedings of the 16th Meeting on the Mathematics of Language*, pp. 91–101, Association for Computational Linguistics, doi:10.18653/v1/W19-5708.

Jane CHANDLEE and Adam JARDINE (2019), Autosegmental Input Strictly Local Functions, *Transactions of the Association for Computational Linguistics*, 7:157–168, doi:10.1162/tacl\_a\_00260.

Alexander CLARK and Shalom LAPPIN (2011), *Linguistic Nativism and the Poverty of the Stimulus*, Wiley-Blackwell.

Bruno COURCELLE (1994), Monadic Second-Order Definable Graph Transductions: A Survey, *Theoretical Computer Science*, 126(1):53–75, doi:10.1016/0304-3975(94)90268-2.

Bruno COURCELLE and Joost ENGELFRIET (2012), *Graph Structure and Monadic Second-Order Logic: A Language-Theoretic Approach*, volume 138, Cambridge University Press.

Colin DE LA HIGUERA (2010), *Grammatical Inference: Learning Automata and Grammars*, Cambridge University Press, doi:10.1017/CBO9781139194655.

Aniello DE SANTO and Thomas GRAF (2019), Structure Sensitive Tier Projection: Applications and Formal Properties, in Raffaella BERNARDI, Greg KOBELE, and Sylvain POGODALLA, editors, *Formal Grammar 2019*, volume 11668 of *Lecture Notes in Computer Science*, pp. 35–50, Springer Verlag, doi:10.1007/978-3-662-59648-7\_3.

Hossep DOLATIAN and Jonathan RAWSKI (2020), Multi-Input Strictly Local Functions for Templatic Morphology, in *Proceedings of the Society for Computation in Linguistics*, volume 3, pp. 282–296, <https://scholarworks.umass.edu/scil/vol3/iss1/28>.

Matt EDLEFSEN, Dylan LEEMAN, Nathan MYERS, Nathaniel SMITH, Molly VISSCHER, and David WELLCOME (2008), Deciding Strictly Local (SL) Languages, in Jon BREITENBUCHER, editor, *Proceedings of the 2008 Midstates Conference for Undergraduate Research in Computer Science and Mathematics*, pp. 66–73.

Joost ENGELFRIET and Hendrik Jan HOOGEBOOM (2001), MSO Definable String Transductions and Two-Way Finite-State Transducers, *ACM Transactions on Computational Logic*, 2(2):216–254, doi:10.1145/371316.371512.

- Emmanuel FILIOT (2015), Logic-Automata Connections for Transformations, in *Logic and Its Applications*, volume 8923 of *Lecture Notes in Computer Science*, pp. 30–57, Springer Berlin / Heidelberg, doi:10.1007/978-3-662-45824-2\_3.
- Sara FINLEY (2008), *Formal and Cognitive Restrictions on Vowel Harmony*, Ph.D. thesis, Johns Hopkins University.
- Jerry Alan FODOR and Zenon Walter PYLYSHYN (1988), Connectionism and Cognitive Architecture: A Critical Analysis, *Cognition*, 28(1–2):3–71, doi:10.1016/0010-0277(88)90031-5.
- Pedro GARCIA, Enrique VIDAL, and José ONCINA (1990), Learning Locally Testable Languages in the Strict Sense, in *Proceedings of the 1st International Workshop on Algorithmic Learning Theory*, pp. 325–338, <https://grfia.dlsi.ua.es/repositori/grfia/pubs/111/alt1990.pdf>.
- R. W. N. GOEDEMANS, Jeffrey HEINZ, and Harry VAN DER HULST (2015), StressTyp2, <http://st2.ullet.net/>.
- Edward Mark GOLD (1967), Language Identification in the Limit, *Information and Control*, 10(5):447–474, doi:10.1016/S0019-9958(67)91165-5.
- John Anton GOLDSMITH (1976), *Autosegmental Phonology*, Ph.D. thesis, Swarthmore College, <https://dspace.mit.edu/bitstream/handle/1721.1/16388/03188555-MIT>.
- Saul GORN (1967), Explicit Definitions and Linguistic Dominoes, in John HART and Satoru TAKASU, editors, *Systems and Computer Science*, pp. 77–115, University of Toronto Press, doi:10.3138/9781487592769.
- Thomas GRAF (2011), Closure Properties of the Minimalist Derivation Tree Languages, in Sylvain POGODALLA and Jean-Philippe PROST, editors, *Logical Aspects of Computational Linguistics*, volume 6736 of *Lecture Notes in Computer Science*, pp. 96–111, Springer Berlin / Heidelberg, doi:10.1007/978-3-642-22221-4\_7.
- Thomas GRAF (2014), Beyond the Apparent: Cognitive Parallels Between Syntax and Phonology, in Carson T. SCHÜTZE and Linnaea STOCKALL, editors, *Connectedness: Papers by and for Sarah VanWagenen*, volume 18 of *UCLA Working Papers in Linguistics*, pp. 161–174, <http://phonetics.linguistics.ucla.edu/wp1/issues/wp118/papers/graf.pdf>.
- Thomas GRAF (2020), Curbing Feature Coding: Strictly Local Feature Assignment, in *Proceedings of the Society for Computation in Linguistics*, volume 3, pp. 362–371, doi:10.7275/f7y5-xz32.
- Thomas GRAF and Nazila SHAFIEI (2019), C-Command Dependencies as TSL String Constraints, in *Proceedings of the Society for Computation in Linguistics*, volume 2, pp. 205–215, doi:10.7275/4rrx-x488.
- Leonard H. HAINES (1969), On Free Monoids Partially Ordered by Embedding, *Journal of Combinatorial Theory*, 6(1):94–98, doi:10.1016/s0021-9800(69)80111-0.

Jeffrey HEINZ (2010a), Learning Long-Distance Phonotactics, *Linguistic Inquiry*, 41(4):623–661, doi:10.1162/ling\_a\_00015.

Jeffrey HEINZ (2010b), String Extension Learning, in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 897–906, Association for Computational Linguistics, <https://www.aclweb.org/anthology/P10-1092>.

Jeffrey HEINZ (2016), Computational Theories of Learning and Developmental Psycholinguistics, in Jeffrey LIDZ, William SYNDER, and Joe PATER, editors, *The Oxford Handbook of Developmental Linguistics*, chapter 27, pp. 633–663, Oxford University Press, doi:10.1093/oxfordhb/9780199601264.013.27.

Jeffrey HEINZ (2018), The Computational Nature of Phonological Generalizations, in Larry HYMAN and Frank PLANK, editors, *Phonological Typology*, volume 23 of *Phonetics and Phonology*, chapter 5, pp. 126–195, Mouton de Gruyter, doi:10.1515/9783110451931-005.

Jeffrey HEINZ, Anna KASPRZIK, and Timo KÖTZING (2012), Learning in the Limit with Lattice-Structured Hypothesis Spaces, *Theoretical Computer Science*, 457:111–127, doi:10.1016/j.tcs.2012.07.017.

Jeffrey HEINZ, Chetan RAWAL, and Herbert G. TANNER (2011), Tier-based Strictly Local Constraints for Phonology, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Short Papers*, volume 2, pp. 58–64, Association for Computational Linguistics, <https://aclweb.org/anthology/P11-2011>.

Jeffrey HEINZ and James ROGERS (2013), Learning Subregular Classes of Languages with Factored Deterministic Automata, in *Proceedings of the 13th Meeting on the Mathematics of Language*, pp. 64–71, Association for Computational Linguistics, <https://www.aclweb.org/anthology/W13-3007>.

Suzana HERCULANO-HOUZEL and Roberto LENT (2005), Isotropic Fractionator: A Simple, Rapid Method for the Quantification of Total Cell and Neuron Numbers in the Brain, *Journal of Neuroscience*, 25(10):2518–2521, doi:10.1523/JNEUROSCI.4526-04.2005.

Larry M. HYMAN (2009), How (Not) to Do Phonological Typology: The Case of Pitch-Accent, *Language Sciences*, 31(2–3):213–238, doi:10.1016/j.langsci.2008.12.007.

Larry M. HYMAN and Francis X. KATAMBA (1993), A New Approach to Tone in Luganda, *Language*, 69(1):34–67, doi:10.2307/416415.

Sanjay JAIN, Daniel OSHERSON, James S. ROYER, and Arun SHARMA (1999), *Systems That Learn: An Introduction to Learning Theory (Learning, Development and Conceptual Change)*, The MIT Press, 2nd edition.

Adam JARDINE (2017a), The Local Nature of Tone-Association Patterns, *Phonology*, 34(2):385–405, doi:10.1017/s0952675717000185.

- Adam JARDINE (2017b), On the Logical Complexity of Autosegmental Representations, in *Proceedings of the 15th Meeting on the Mathematics of Language*, pp. 22–35, Association for Computational Linguistics, doi:10.18653/v1/W17-3403.
- Adam JARDINE (2019), The Expressivity of Autosegmental Grammars, *Journal of Logic, Language and Information*, 28(1):9–54, ISSN 1572-9583, doi:10.1007/s10849-018-9270-x.
- Adam JARDINE, Nick DANIS, and Luca IACOPONI (2021), A Formal Investigation of Q-Theory in Comparison to Autosegmental Representations, *Linguistic Inquiry*, 52(2):333–358, doi:10.1162/ling\_a\_00376.
- Adam JARDINE and Jeffrey HEINZ (2016), Learning Tier-Based Strictly 2-Local Languages, *Transactions of the Association for Computation in Linguistics*, 4:87–98, doi:10.1162/tacl\_a\_00085.
- Adam JARDINE and Kevin MCMULLIN (2017), Efficient Learning of Tier-Based Strictly k-Local Languages, in Frank DREWES, Carlos MARTÍN-VIDE, and Bianca TRUTHE, editors, *Language and Automata Theory and Applications: 11th International Conference*, volume 10168 of *Lecture Notes in Computer Science*, pp. 64–76, Springer, Cham, doi:10.1007/978-3-319-53733-7\_4.
- Kent JOHNSON (2004), Gold’s Theorem and Cognitive Science, *Philosophy of Science*, 71:571–592.
- Mark JOHNSON (1988), *Attribute-Value Logic and the Theory of Grammar*, Center for the Study of Language and Information.
- Michael KEARNS and Umesh VAZIRANI (1994), *An Introduction to Computational Learning Theory*, MIT Press.
- Paul J. KING (1989), *A Logical Formalism for Head-Driven Phrase Structure Grammar*, Ph.D. thesis, University of Manchester, <https://www.proquest.com/docview/2201235827>.
- Gregory M. KOBELE (2011), Minimalist Tree Languages are Closed Under Intersection with Recognizable Tree Languages, in Sylvain POGODALLA and Jean-Philippe PROST, editors, *Logical Aspects of Computational Linguistics*, volume 6736 of *Lecture Notes in Computer Science*, pp. 129–144, Springer Berlin / Heidelberg, doi:10.1007/978-3-642-22221-4\_9.
- John Richard KRUEGER (1961), *Chuvash Manual: Introduction, Grammar, Reader, and Vocabulary*, volume 7 of *Uralic and Altaic Series*, Indiana University.
- Regine LAI (2015), Learnable vs. Unlearnable Harmony Patterns, *Linguistic Inquiry*, 46(3):425–451, doi:10.1162/LING\_a\_00188.
- Dakotah LAMBERT (2021), Grammar Interpretations and Learning TSL Online, in Adam JARDINE, Jane CHANDLEE, Jeffrey HEINZ, Menno VAN ZAAENEN, and Rémi EYRAUD, editors, *Proceedings of the Fifteenth International Conference on Grammatical Inference (ICGI 2020/21)*, PMLR: Proceedings of Machine Language Research, <http://proceedings.mlr.press/>, in press.

Dakotah LAMBERT and James ROGERS (2019), A Logical and Computational Methodology for Exploring Systems of Phonotactic Constraints, in *Proceedings of the Society for Computation in Linguistics*, volume 2, pp. 247–256, doi:10.7275/t0dv-9t05.

Dakotah LAMBERT and James ROGERS (2020), Tier-Based Strictly Local Stringsets: Perspectives from Model and Automata Theory, in *Proceedings of the Society for Computation in Linguistics*, volume 3, pp. 330–337, doi:10.7275/2n1j-pj39.

Shalom LAPPIN and Stuart Merrill SHIEBER (2007), Machine Learning Theory and Practice as a Source of Insight into Universal Grammar, *Journal of Linguistics*, 43(2):393–427, doi:10.1017/S0022226707004628.

Julie Anne LEGATE and Charles D. YANG (2002), Empirical Re-assessment of Stimulus Poverty Arguments, *The Linguistic Review*, 18(1–2):151–162, doi:10.1515/tlir.19.1-2.151.

Leonid LIBKIN (2004), *Elements of Finite Model Theory*, Texts in Theoretical Computer Science, Springer Berlin / Heidelberg, doi:10.1007/978-3-662-07003-1.

Kevin MCMULLIN and Gunnar Ólafur HANSSON (2019), Inductive Learning of Locality Relations in Segmental Phonology, *Laboratory Phonology: Journal of the Association for Laboratory Phonology*, 10(1):1–53, doi:10.5334/labphon.150, article 14.

Robert McNAUGHTON and Seymour A. PAPERT (1971), *Counter-Free Automata*, MIT Press.

Tom Michael MITCHELL (1982), Generalization as Search, *Artificial Intelligence*, 18(2):203–226, doi:10.1016/0004-3702(82)90040-6.

Tom Michael MITCHELL (2017), Key Ideas in Machine Learning, in *Machine Learning: Second Edition*, McGraw Hill, <http://www.cs.cmu.edu/~tom/mlbook/keyIdeas.pdf>, forthcoming.

Meyhar MOHRI, Afshin ROSTAMIZADEH, and Ameet TALWALKAR (2012), *Foundations of Machine Learning*, MIT Press.

Max NELSON, Hossep DOLATIAN, Jonathan RAWSKI, and Brandon PRICKETT (2020), Probing RNN Encoder-Decoder Generalizations of Subregular Functions Using Reduplication, in *Proceedings of the Society for Computation in Linguistics*, volume 3, pp. 31–42, doi:10.7275/xd0r-pg04.

Partha NIYOGI (2006), *The Computational Nature of Language Learning and Evolution*, Cambridge, MA: MIT Press.

Martin Andreas NOWAK, Natalia L. KOMAROVA, and Partha NIYOGI (2002), Computational and Evolutionary Aspects of Language, *Nature*, 417(6889):611–617, doi:10.1038/nature00771.

- Chris OAKDEN (2020), Notational Equivalence in Tonal Geometry, *Phonology*, 37(2):257–296, doi:10.1017/S0952675720000123.
- Christopher POTTS and Geoffrey PULLUM (2002), Model Theory and the Content of OT Constraints, *Phonology*, 19:361–393.
- Jonathan RAWSKI (2019), Tensor Product Representations of Subregular Formal Languages, in *Proceedings of the International Joint Conference on Artificial Intelligence workshop on Neural-Symbolic Learning and Reasoning*, pp. 36–42.
- Jonathan RAWSKI and Hossep DOLATIAN (2020), Multi-Input Strictly Local Functions for Tonal Phonology, in *Proceedings of the Society for Computation in Linguistics*, volume 3, pp. 245–260, <https://scholarworks.umass.edu/scil/vol3/iss1/25>.
- Jonathan RAWSKI and Jeffrey HEINZ (2019), No Free Lunch in Linguistics or Machine Learning: Response to Pater, *Language*, 93(1):e125–e135, doi:10.1353/lan.2019.0004.
- James ROGERS (1998), *A Descriptive Approach to Language-Theoretic Complexity*, (Monograph.) Studies in Logic, Language, and Information, CSLI Publications.
- James ROGERS (2003), Syntactic Structures as Multi-Dimensional Trees, *Research on Language and Computation*, 1(3–4):265–305, doi:10.1023/A:1024695608419.
- James ROGERS, Jeff HEINZ, Margaret FERO, Jeremy HURST, Dakotah LAMBERT, and Sean WIBEL (2012), Cognitive and Sub-Regular Complexity, in Glyn MORRILL and Mark-Jan NEDERHOF, editors, *Formal Grammar 2012*, volume 8036 of *Lecture Notes in Computer Science*, pp. 90–108, Springer-Verlag, doi:10.1007/978-3-642-39998-5\_6.
- James ROGERS, Jeffrey HEINZ, Gil BAILEY, Matt EDLEFSEN, Molly VISSCHER, David WELLCOME, and Sean WIBEL (2010), On Languages Piecewise Testable in the Strict Sense, in Christian EBERT, Gerhard JÄGER, and Jens MICHAELIS, editors, *The Mathematics of Language: Revised Selected Papers from the 10th and 11th Biennial Conference on the Mathematics of Language*, volume 6149 of *LNCS/LNAI*, pp. 255–265, FoLLI/Springer, doi:10.1007/978-3-642-14322-9\_19.
- James ROGERS and Dakotah LAMBERT (2019a), Extracting Subregular Constraints from Regular Stringsets, *Journal of Language Modelling*, 7(2):143–176, doi:10.15398/jlm.v7i2.209.
- James ROGERS and Dakotah LAMBERT (2019b), Some Classes of Sets of Structures Definable Without Quantifiers, in *Proceedings of the 16th Meeting on the Mathematics of Language*, pp. 63–77, Association for Computational Linguistics, doi:10.18653/v1/W19-5706.
- Marcel-Paul SCHÜTZENBERGER (1965), On Finite Monoids Having Only Trivial Subgroups, *Information and Control*, 8(2):190–194, doi:10.1016/s0019-9958(65)90108-7.

Imre SIMON (1975), Piecewise Testable Events, in Helmut BRAKHAGE, editor, *Automata Theory and Formal Languages*, volume 33 of *Lecture Notes in Computer Science*, pp. 214–222, Springer-Verlag, doi:10.1007/3-540-07407-4\_23.

Kristina STROTHER-GARCIA (2019), *Using Model Theory in Phonology: A Novel Characterization of Syllable Structure and Syllabification*, Ph.D. thesis, University of Delaware.

Wolfgang THOMAS (1982), Classifying Regular Events in Symbolic Logic, *Journal of Computer and Systems Sciences*, 25:360–376, doi:10.1016/0022-0000(82)90016-2.

Leslie Gabriel VALIANT (1984), A Theory of the Learnable, *Communications of the ACM*, 27(11):1134–1142, doi:10.1145/1968.1972.

Iris VAN ROOIJ and Giosuè BAGGIO (2021), Theory Before the Test: How to Build High-Verisimilitude Explanatory Theories in Psychological Science, *Perspectives on Psychological Science*, doi:10.1177/1745691620970604.

Vladimir VAPNIK (1995), *The Nature of Statistical Learning Theory*, Springer.

Mai Ha VU, Ashkan ZEHFROOSH, Kristina STROTHER-GARCIA, Michael SEBOK, Jeffrey HEINZ, and Herbert G. TANNER (2018), Statistical Relational Learning with Unconventional String Models, *Frontiers in Robotics and AI*, 5(76):1–26, doi:10.3389/frobt.2018.00076.

Edwin Samuel WILLIAMS (1976), Underlying Tone in Margi and Igbo, *Linguistic Inquiry*, 7(3):463–484.

Charles YANG (2013), Who's Afraid of George Kingsley Zipf? Or: Do Children and Chimps Have Language?, *Significance*, 10(6):29–34, doi:10.1111/j.1740-9713.2013.00708.x.

*Dakotah Lambert*

Ⓘ 0000-0002-7056-5950

dakotah.lambert@stonybrook.edu

*Jonathan Rawski*

Ⓘ 0000-0003-3996-9815

jon.rawski@sjsu.edu

*Jeffrey Heinz*

Ⓘ 0000-0002-5954-3195

jeffrey.heinz@stonybrook.edu

Department of Linguistics  
and Language Development,  
San José State University

Department of Linguistics  
and Institute for Advanced  
Computational Science,  
Stony Brook University

Dakotah Lambert, Jonathan Rawski, and Jeffrey Heinz (2021), *Typology emerges from simplicity in representations and learning*, *Journal of Language Modelling*, 9(1):151–194

Ⓓ <https://dx.doi.org/10.15398/jlm.v9i1.262>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

ⒸⒾ <http://creativecommons.org/licenses/by/4.0/>