

This is a repository copy of *Machine Learning Models of Universal Grammar Parameter Dependencies*.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/178496/>

Version: Accepted Version

Proceedings Paper:

Kazakov, Dimitar Lubomirov orcid.org/0000-0002-0637-8106, Cordoni, Guido, Ceolin, Andrea et al. (6 more authors) (2017) Machine Learning Models of Universal Grammar Parameter Dependencies. In: Proceedings of The Knowledge Resources for the Socio-Economic Sciences and Humanities Workshop. , pp. 31-37.

https://doi.org/10.26615/978-954-452-040-3_005

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

Machine Learning Models of Universal Grammar Parameter Dependencies

Dimitar Kazakov
Dept. of Comp. Science
University of York
dlk2@york.ac.uk

Guido Cordon
Lang. and Ling. Sci.
University of York
gc927@york.ac.uk

Andrea Ceolin
Dept. of Linguistics
U. of Pennsylvania
ceolin@sas.upenn.edu

Monica A. Irimia
Dept. of Comm. and Econ.
UniMORE
irimiamo@unimore.it

Shin-Sook Kim
Lang. and Ling. Sci.
University of York
sk899@york.ac.uk

Dimitris Michelioudakis
Lang. and Ling. Sci.
University of York
dm9540@york.ac.uk

Nina Radkevich
Lang. and Ling. Sci.
University of York
nr6920@york.ac.uk

Cristina Guardiano
Dip. Com. ed Econ.
UniMORE
cguardiano@unimore.it

Giuseppe Longobardi
Lang. and Ling. Sci.
University of York
gl6730@york.ac.uk

Abstract

The use of parameters in the description of natural language syntax has to balance between the need to discriminate among (sometimes subtly different) languages, which can be seen as a cross-linguistic version of Chomsky’s (1964) descriptive adequacy, and the complexity of the acquisition task that a large number of parameters would imply, which is a problem for explanatory adequacy. Here we present a novel approach in which a machine learning algorithm is used to find dependencies in a table of parameters. The result is a dependency graph in which some of the parameters can be fully predicted from others. These empirical findings can be then subjected to linguistic analysis, which may either refute them by providing typological counter-examples of languages not included in the original dataset, dismiss them on theoretical grounds, or uphold them as tentative empirical laws worth of further study.

1 Introduction

Parametric theories of generative grammar focus on the problem of a formal and principled theory of grammatical diversity (Chomsky, 1981; Baker, 2001; Roberts, 2012). The basic intuition of parametric approaches is that the

majority of observable syntactic differences among languages are derived, usually through complex deductive chains, from a smaller number of more abstract contrasts, drawn from a universal list of discrete, and normally binary, options, called parameters. The relation between observable patterns and the actual syntactic parameters which vary across languages is quite indirect: syntactic parameters are regarded as abstract differences often responsible for wider typological clusters of surface co-variation, often through an intricate deductive structure. In this sense, the concept of parametric data is not to be simplistically identified with that of syntactic pattern: co-varying syntactic properties/patterns are in fact the empirical manifestations of much more abstract cognitive structures.

Syntactic parameters are conceived as definable by UG (i.e. universally comparable) and set by each learner on the basis of her/his linguistic environment. Open parameters, or any set of more primitive concepts they can derive from (Longobardi, 2005; Lightfoot, 2017), define a variation space for biologically acquirable grammars, set (a.k.a. *closed*) parameters specify each of these grammars. Thus, the core grammar of every natural language can in principle be represented by a string of binary symbols (Clark and Roberts, 1993), each coding the value of a parameter of UG.

The Parametric Comparison Method (PCM, (Longobardi and Guardiano, 2009)) uses syntactic parameters to study historical

relationships among languages. An important aspect of parametric systems that is particularly relevant to the present research is that parameters form a pervasive network of partial implications (Guardiano and Longobardi, 2005; Longobardi and Guardiano, 2009; Longobardi et al., 2013): one particular value of some parameter A, but not the other, often entails the irrelevance of parameter B, whose consequences, i.e. the corresponding surface patterns, become predictable. Under such conditions, B becomes redundant and will not be set at all by the learner. The PCM system makes such interdependencies explicit: in our notation, the symbols $+$ and $-$ are used to represent the binary value of each parameter; the symbol ‘0’, instead, encodes the neutralising effect of implicational cross-parametric dependencies, i.e. cases in which the content of a parameter is either entirely predictable, or irrelevant altogether. The conditions which must hold for each parameter not to be neutralised are expressed in a Boolean form, i.e., either as simple states of another parameter (or negation thereof), or as conjunctions or disjunctions of values of other parameters.

The PCM has shown that an important effect of the pervasiveness of parameter interdependencies is a noticeable downsizing of the space of grammatical variation: according to some preliminary experiments (Bortolussi et al., 2011), the number of possible languages generated from a given set of independent binary parameters is reduced from 10^{18} to 10^{11} when their interdependencies are taken into account. This also crucially implies a noticeable reduction of the space of possible languages that a learner has to navigate when acquiring a language.

Here we adopt an empirical, data-driven approach to the task of identifying parameter dependencies, which has been implemented on a database of 71 languages described through the values of 91 syntactic parameters (see Appendix A) expressing the internal syntax of nominal structures. Our results show that applying machine learning techniques to the data reveals previously unknown dependencies between parameters, which could potentially lead to a further significant reduction of the

```

if  $P_1 = +$  and  $P_2 = -$  then
     $P_3 = +$ 
else
     $P_3 = -$ 

```

Figure 1: Parameter dependency model example

search space of possible languages.

This paper sets out to identify parameters whose entire range of values can be fully predicted from the values of other parameters. There is an important difference between previously published work on parameter dependencies and this paper’s contribution, which needs to be emphasised: rather than state that, for example, any language in which $P_1 = +$ will have a fully predictable value of P_2 (a fact which we encode as $P_2 = 0$), we seek parameters whose value can be deduced in *all* cases from the values of certain other parameters, e.g. as shown in the hypothetical example in Figure 1. Should such a rule prove to have universal validity, then parameter P_3 would never offer any advantage in separating any two languages, yet it could clearly still play a useful role in describing them.

2 Learning Dependencies

We process our table of dimensions ($\#lang \times \#param$) with the data mining package WEKA (v.3.6.13) (Hall et al., 2009). More specifically, we take the values of all parameters but one for all languages (i.e. a dataset of size $(\#lang \times \#param - 1)$, and learn a decision tree that predicts the value of the remaining parameter from the values of the other parameters. (Typically, only a few are necessary in each case.) This is repeated to produce a decision tree for each of the parameters. The machine learning algorithm used was ID3 (Quinlan, 1986). The algorithm produces a decision tree, in which each leaf corresponds to the value of the modelled parameter for the combination of parameter values listed on the way from the root to that leaf, e.g.: **if** $FGN = -$ **and** $FGP = +$ **then** $GCO = +$ (see Table 1). Unlike some of the more sophisticated decision tree learning algorithms, such as C4.5 (Quinlan, 1993), no postprocessing of the tree learnt

(such as *pruning* (Mitchell, 1997)) takes place, and the tree remains an accurate, exact reflection of the training data. If the combination of parameter values corresponding to one of the leaves of the tree is not observed in the data, the leaf contains the special label ‘null’ (see the tree predicting *GCO* in Table 1). In all other cases, that is, whenever the leaf label is ‘+’, ‘-’ or ‘0’, this is supported by one or more examples (languages) in the data.

Table 1: Examples of decision trees for parameters *FGN* and *GCO*

```

~~~~~
FGN:
if GCO = 0 then FGN = +
if GCO = + then FGN = -
if GCO = - then FGN = -
~~~~~
GCO:
if FGN = 0 then GCO = null ;never occurs
if FGN = + then GCO = 0
if FGN = - then
  if FGP = 0 then GCO = null;never occurs
  if FGP = + then GCO = +
  if FGP = - then GCO = -
~~~~~

```

3 Results

The decision trees for all parameters were used to produce a dependency graph in which each vertex represents a parameter, and directed edges link the parameters, whose values are needed to predict a given parameter, with the node representing that parameter. For instance, there are edges from both *FGN* and *FGP* to *GCO*, as the decision tree for *GCO* refers to the values of *FGN* and *FGP*. Some of the decision trees are more complex, making use of up to nine separate parameters. The resulting graph is very complex (see Fig. 2). Therefore, we also present a subset of the graph (see Fig. 3), which only visualises those trees predicting one parameter from the value of one (as in the case of *FGN*) or two other parameters (e.g. *GCO*). The fact that some of the rules are missing from this graph is not an issue: for each listed node, all of the incoming edges are present, so that if we know those parameters, we are guaranteed to know the parameter they point to.

The interpretation of the graph is straightforward. For instance, looking at its top right

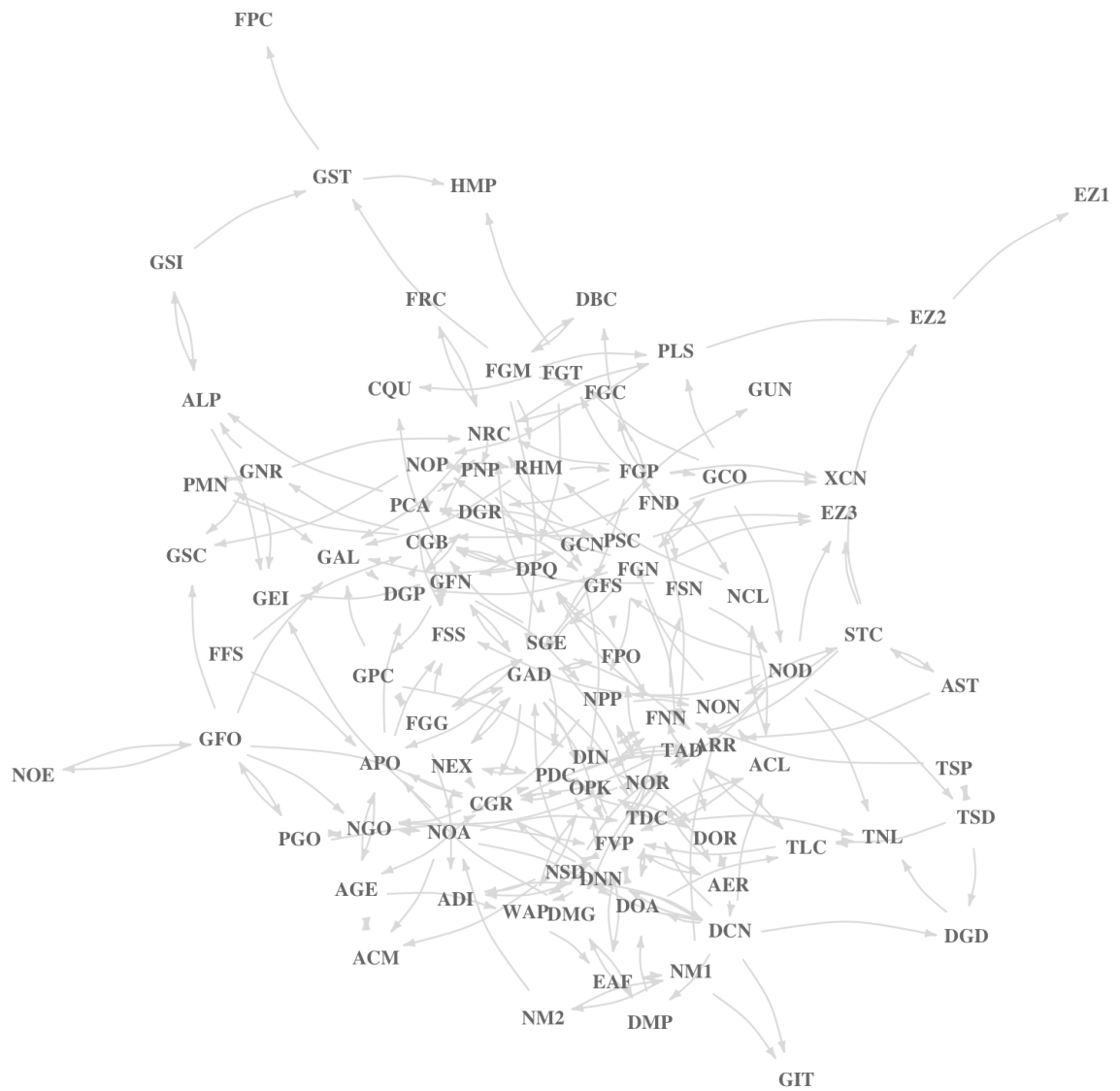
corner, one can deduce that for any language in the dataset, it is enough to know the values of parameters *EZ3* and *PLS* in order to know the value of *EZ2*, and therefore, of *EZ1*, too. Knowing (the value of) *FVP* means one also knows *DMG* and *NSD*; if one knows both *FVP* and *DNN*, the values of *DNG*, *NSD*, *DSN*, *DMP* and *DMG* are fully predictable for the given data set. In other words, 7 parameters (*FVP*, *DNN*, *DNG*, *NSD*, *DSN*, *DMP* and *DMG*) can be reduced to just 2 without any loss of information.

Some of the rules identified by the algorithm are not new, and are already contained in the dataset, as encoded by the implicational system described in Section 1. For instance, the parameter *RHM* is encoded as 0 when *FGP* = -, as the value of *RHM* is fully predictable in those cases. When a decision tree predicting *FGP* is learned, the result is as follows: **if *RHM* = 0 then *FGP* = - else *FGP* = +.**

Even the rest of the rules learned are still just empirical findings that may change with the addition of other examples of languages or their validity may be questioned by linguists on theoretical grounds.

Linguistic analysis of the results is ongoing, and while no part of the results has been accepted as sufficient evidence to dispose of a parameter, implication rules may be revised on the basis of the decision trees learned, as in the case of the parameter *PLS*. According to its definition, the parameter “*asks if in a language without grammaticalized Number, a plural marker can also appear outside a nominal phrase, marking a distributive relation between the plural subject and the constituent bearing it.*” (E.g. *PLS* = + for Korean, but *PLS* = - for Japanese.)

Prior to this research, there was an implication rule stating that *PLS* is neutralised (that is, its value is predictable) for all combinations of *CGO* and *FGN* values other than *CGO* = - and *FGN* = -. This rule has now been replaced with a new rule stating that *PLS* is neutralised for all combinations of values of *FGM* and *FGN*, except when *FGM* = + and *FGN* = -, and the evidence showing that the new rule is consistent with the data came from the tree learned for *PLS*.





4 Discussion

The results reported here show that applying machine learning techniques to the data can reveal previously unknown dependencies between parameters, leading to a potentially significant reduction in the search space of possible languages. The data contains more features than data points, which can make for the generation of spurious rules. The most obvious way to counteract this, adding more languages, comes at a very high cost, as it requires well-trained linguists. One can also use Occam's Razor and limit the search space of possible rules by limiting the number of antecedents in the rule, e.g. to two as we did here. Yet another approach is to collect data selectively for rules of interest, as only a small number of parameters, e.g. 2–3 per language, will be needed to test each rule.

This research could have important implications for the understanding of processes underlying the faculty of language (potentially strengthening the case for UG), with implications ranging from models of language acquisition to historical linguistics, where the syntactic relatedness between two languages may be more adequately measured. However, the approach requires a close collaboration between a machine learning expert, discovering empirical laws in the data, and a linguist who can test their plausibility and theoretical consequences. There is also an open theoretical computational learning challenge here presented by the need to estimate the significance of empirical findings from a given number of examples (languages) with respect to the available range of discriminative features in the dataset.

References

- M. Baker. 2001. *The Atoms of Language*. Basic Books, New York.
- L. Bortolussi, G. Longobardi, C. Guardiano, and A. Sgarro. 2011. How many possible languages are there? In G. Bel-Enguix, V. Dahl, and M.D. Jiménez-López, editors, *Biology, Computation and Linguistics*, IOS, Amsterdam, pages 168–179.
- N. Chomsky. 1964. *Current issues in linguistic theory*. Mouton, The Hague.
- N. Chomsky. 1981. *Lectures on Government and Binding*. Foris, Dordrecht.
- R. Clark and I. Roberts. 1993. A computational model of language learnability and language change. *Linguistic Inquiry* 24:299–345.
- C. Guardiano and G. Longobardi. 2005. Parametric comparison and language taxonomy. In M. Batllori, M. L. Hernanz, C. Picallo, and F. Roca, editors, *Grammaticalization and parametric variation*, OUP, Oxford, pages 149–174.
- M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. 2009. The WEKA data mining software. *ACM SIGKDD Explor. Newsl.* 11:149–174.
- D. W. Lightfoot. 2017. Discovering new variable properties without parameters. *Linguistic Analysis* 41. Special edition: Parameters: What are they? Where are they?
- G. Longobardi. 2005. A minimalist program for parametric linguistics? In H. Broekhuis, N. Corver, M. Huybregts, U. Kleinhenz, and J. Koster, editors, *Organizing Grammar: Linguistic Studies*, Mouton de Gruyter, Berlin/New York, pages 407–414.
- G. Longobardi and C. Guardiano. 2009. Evidence for syntax as a signal of historical relatedness. *Lingua* 119(11).
- G. Longobardi, C. Guardiano, G. Silvestri, A. Boattini, and A. Ceolin. 2013. Toward a syntactic phylogeny of modern Indo-European languages. *Journal of Historical Linguistics* 3(1):122–152.
- T. Mitchell. 1997. *Machine Learning*. MIT.
- R. Quinlan. 1986. Induction of decision trees. *Machine Learning* 1(1):81–106.
- R. Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publ., San Mateo, CA.
- I.G. Roberts. 2012. On the nature of syntactic parameters: a programme for research. In C. Galves, S. Cyrino, R. Lopes, F. Sandalo, and J. Avelar, editors, *Parameter Theory and Language Change*, Oxford University Press, Oxford, pages 319–334.

Appendix A: List of Parameters

FGP	gramm. person	AST	structured APs
FGM	gramm. Case	STC	structured cardinals
FPC	gramm. perception	GPC	gender polarity cardinals
FGT	gramm. temporality	PMN	personal marking on numerals
FGN	gramm. number	CQU	cardinal quantifiers
GCO	gramm. collective number	PCA	number spread through cardinal adjectives
PLS	plurality spreading	FFS	feature spread to structured APs
FND	number in D	ADI	D-controlled infl. on A
NOD	NP over D	PSC	number spread from cardinal quantifiers
FSN	feature spread to N	RHM	Head-marking on Rel
FNN	number on N	FRC	verbal relative clauses
SGE	semantic gender	NRC	nominalised relative clause
FGG	gramm. gender	NOR	NP over verbal relative clauses/ adpositional genitives
CGB	unbounded sg N	AER	relative extrap.
DGR	gramm. amount	ARR	free reduced rel
DGP	gramm. text anaphora	DOR	def on relatives
CGR	strong amount	NOP	NP over non-genitive arguments
NSD	strong person	PNP	P over complement
FVP	variable person	NPP	N-raising with obl. pied-piping
DGD	gramm. distality	NGO	N over GenO
DPQ	free null partitive Q	NOA	N over As
DCN	article-checking N	NM2	N over M2 As
DNN	null-N-licensing art	NM1	N over M1 As
DIN	D-controlled infl. on N	EAF	fronted high As
FGC	gramm. classifier	NON	N over numerals
DBC	strong classifier	FPO	feature spread to genitive postpositions
GSC	c-selection	ACM	class MOD
NOE	N over ext. arg.	DOA	def on all +N
DMP	def matching pronominal possessives	NEX	gramm. expletive article
DMG	def matching genitives	NCL	clitic poss.
GCN	Poss ^o -checking N	PDC	article-checking poss.
GFN	Gen-feature spread to Poss ^o	ACL	enclitic poss. on As
GAL	Dependent Case in NP	APO	adjectival poss.
GUN	uniform Gen	WAP	wackernagel adjectival poss.
EZ1	generalized linker	AGE	adjectival Gen
EZ2	non-clausal linker	OPK	obligatory possessive with kinship nouns
EZ3	non-genitive linker	TSP	split deictic demonstratives
GAD	adpositional Gen	TSD	split demonstratives
GFO	GenO	TAD	adjectival demonstratives
PGO	partial GenO	TDC	article-checking demonstratives
GFS	GenS	TLC	Loc-checking demonstratives
GIT	Genitive-licensing iterator	TNL	NP over Loc
GSI	grammaticalised inalienability	XCN	conjugated nouns
ALP	alienable possession		
GST	grammaticalised Genitive		
GEI	Genitive inversion		
GNR	non-referential head marking		
HMP	NP-heading modifier		