

Prevalence and recoverability of syntactic parameters in sparse distributed memories

Jeong Joon Park, Ronnel Boettcher, Andrew Zhao, Alex Mun, Kevin Yuh, Vibhor Kumar, and Matilde Marcolli

California Institute of Technology
1200 E. California Blvd, Pasadena, CA 91125, USA

Abstract. We propose a new method, based on sparse distributed memory, for studying dependence relations between syntactic parameters in the Principles and Parameters model of Syntax. By storing data of syntactic structures of world languages in a Kanerva network and checking recoverability of corrupted data from the network, we identify two different effects: an overall underlying relation between the prevalence of parameters across languages and their degree of recoverability, and a finer effect that makes some parameters more easily recoverable beyond what their prevalence would indicate. The latter can be seen as an indication of the existence of dependence relations, through which a given parameter can be determined using the remaining uncorrupted data.

Keywords: Syntactic structures, Principles and Parameters, Kanerva networks

1 Introduction

The general idea behind the Principles and Parameters approach to Syntax, [2], is the encoding of syntactic properties of natural languages as a vector of binary variables, referred to as *syntactic parameters*. (For an expository introduction, see [1].) While this model has controversial aspects, syntactic parameters are especially suitable from the point of view of a mathematical approach to understanding the geometry of the syntactic parameters space and the distribution of features across language families, with geometric methods of modern data analysis, see [16], [18], [20], [21], [22]. Among the shortcomings ascribed to the Principles and Parameters model (see for instance [7]) is the lack of a complete set of such variable, the unclear nature of the dependence relations between them, and the lack of a good set of independent coordinates.

In this paper we rely on data of syntactic structures collected in the ‘Syntactic Structures of the World’s Languages’ (SSWL) database [23]. We selected a list of 21 syntactic parameters (numbered 1 to 20 and A01 in [23]), which mostly describe word order relations¹, and a list of 166 languages, selected so that they

¹ A detailed description of the properties described by these syntactic features can be found at <http://sswl.railsplayground.net/browse/properties>.

cut across a broad range of different linguistic families, for which the values of these 21 parameters are fully recorded in the SSWL database. By storing these data of syntactic parameters in a Kanerva Network, we test for recoverability when one of the binary variables is corrupted.

We find an overall relation between recoverability and prevalence across languages, which depends on the functioning of the sparse distributed memory. Moreover, we also see a further effect, which deviates from a simple relation with the overall prevalence of a parameter. This shows that certain syntactic parameters have a higher degree of recoverability in a Kanerva Network. This property can be interpreted as a consequence of existing underlying dependence relations between different parameters. With this interpretation, one can envision a broader use of Kanerva Networks as a method to identify further, and less clearly visible, dependence relations between other groups of syntactic parameters. Another reason why it is interesting to analyze syntactic parameters using Kanerva Networks is the widespread use of the latter as models of human memory, [6], [10], [12]. In view of the problem of understanding mechanism of language acquisition, and how the syntactic structure of language may be stored in the human brain, sparse distributed memories appear to be a promising candidate for the construction of effective computational models.

2 Sparse Distributed Memory

Kanerva Networks were developed by Pentti Kanerva in 1988, [9], [10], as a mathematical model of human long term memory. The model allows for approximate accuracy storage and recall of data at any point in a high dimensional space, using fixed hard locations distributed randomly throughout the space. During storage of a datum, hard locations close to the datum encode information about the data point. Retrieval of information at a location in the space is performed by pooling nearby hard locations and aggregating their encoded data. The mechanism allows for memory addressability of a large memory space with reasonable accuracy in a sparse representation. Kanerva Networks model human memory in the following way: a human thought, perception, or experience is represented as an (input) feature vector – a point in a high dimensional space. Concepts stored by the brain are also represented as feature vectors, and are usually stored relatively far from each other in the high dimensional space (the mind). Thus, addressing the location represented by the input vector will yield, to a reasonable degree of accuracy, the concept stored near that location. Thus, Kanerva Networks model the fault tolerance of the human mind – the mind is capable of mapping imprecise input experiences to well defined concepts. For a short introduction to Kanerva Networks aimed at a general public, see §13 of [5].

The functioning of Kanerva Network models can be summarized as follows. Over the field $\mathbb{F}_2 = \{0, 1\}$, consider a vector space (Boolean space) \mathbb{F}_2^N of sufficiently large dimension N . Inside \mathbb{F}_2^N , choose a uniform random sample of 2^k hard locations, with $2^k \ll 2^N$ (a precise estimate is derived in §6 of [9]). Com-

pute the median Hamming distance between hard locations. The *access sphere* of a point in the space \mathbb{F}_2^N is a Hamming sphere of radius slightly larger than this median value (see §6 of [9] for some precise estimates). When writing to the network at some location X in the space \mathbb{F}_2^N , data is distributively stored by writing to all hard locations within the access sphere of that point X . Namely, each hard location stores N counters (initialized to 0), and all hard locations within the access sphere of X have their i -th counter incremented or decremented by 1, depending on the value of the i -th bit of X , see §3.3.1 of [10]. When the operation is performed for a set of locations, each hard location stores a datum whose i -th entry is determined by the majority rule of the corresponding i -th entries for all the stored data. One reads at a location Y in the network a new datum, whose i -th entry is determined by comparing 0 to the i -th counters of all the hard locations that fall within the access sphere of Y , that is, the i -th entry read at Y is itself given by the majority rule on the i -th entries of all the data stored at all the hard locations accessible from Y . The network is typically successful in reconstructing stored data, because intersections between access spheres are infrequent and small. Thus, copies of corrupted data in hard locations within the access sphere of a stored datum X are in the minority with respect to hard locations faithful to X 's data. When a datum is corrupted by noise (*i.e.* flipping bit values randomly), the network is sometimes capable of correctly reconstructing these corrupted bits. The ability to reconstruct certain bits hints that these bits are derived from the remaining, uncorrupted bits in the data. Thus, Kanerva networks are a valuable general tool for detecting dependencies in a high-dimensional data sets, see [8].

3 Recoverability of Syntactic Features

The 21 SSWL syntactic features and 166 languages considered provide 166 data points in a Kanerva Network with Boolean space \mathbb{F}_2^{21} , where each data point is a concatenated binary string of all the values, for that particular language, of the 21 syntactic parameters considered. The Kanerva network was initialized with an access sphere of $n/4$, with n the median Hamming distance between items. This was the optimal value we could work with, because larger values resulted in an excessive number of hard locations being in the sphere, which became computationally unfeasible with the Python SDM library.

Three different methods of corruption were tested. First, the correct data was written to the Kanerva network, then reads at corrupted locations were tested. A known language bit-string, with a single corrupted bit, was used as the read location, and the result of the read was compared to the original bit-string in order to test bit recovery. The average Hamming distance resulting from the corruption of a given bit, corresponding to a particular syntactic parameter, was calculated across all languages. In order to test for relationships independent of the prevalence of the features, another test was run that normalized for this. For each feature, a subset of languages of fixed size was chosen randomly such that half of the languages had that feature. Features that had too few languages with

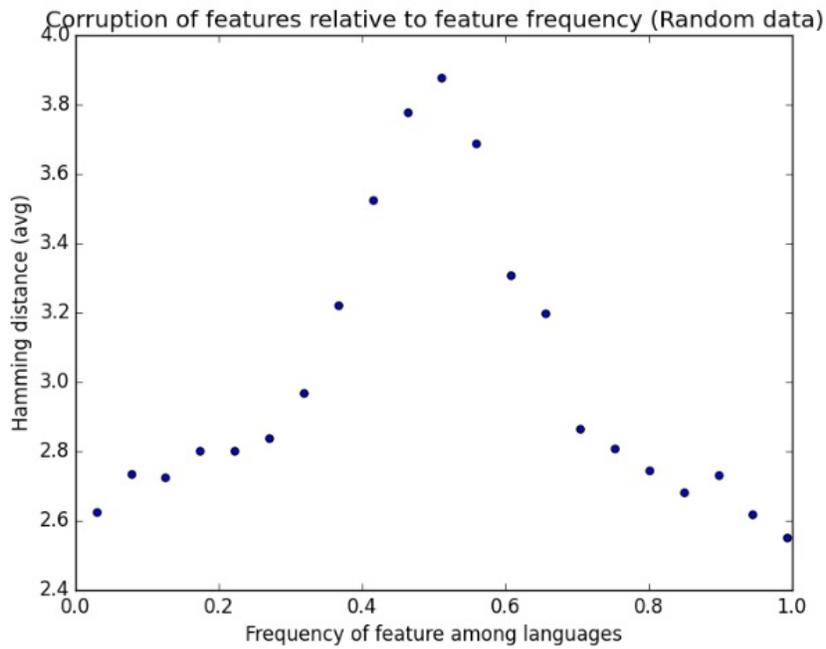
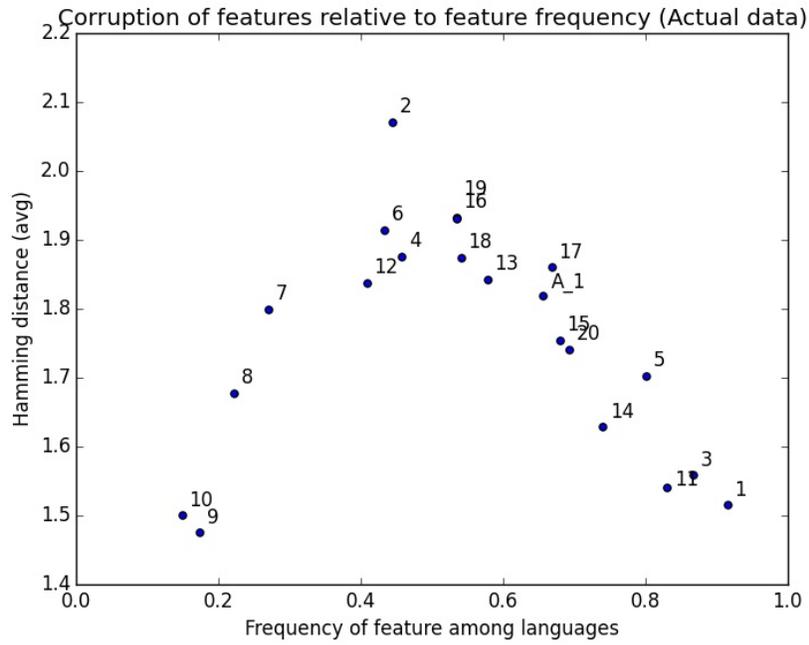


Fig. 1. Prevalence and recoverability for syntactic parameters in a Kanerva Network (actual data compared with random data).

or without the feature to reach the chosen fixed size were ignored for this purpose. For this test, a fixed size of 95 languages was chosen, as smaller sizes would yield less significant results, and larger sizes would result in too many languages being skipped. The languages were then written to the Kanerva network and the recoverability of that feature was measured. Finally, to check whether the different recovery rates we obtained for different syntactic parameters were really a property of the language data, rather than of the Kanerva network itself, the test was run again with random data generated with an approximately similar distribution of bits. The results for the actual data and for random data are reported in Figure 1.

The random data show an overall general shape of the curve that reflects a property of the Kanerva network relating frequency of occurrence and recoverability. This overall effect, relating frequencies and recoverability, seen in random data with the same frequencies as the chosen set of parameters, seems in itself interesting, given ongoing investigations on how prevalence rates of different syntactic parameters may correlate to neuroscience models, see for instance [13].

The magnitude of the values for the actual data, however, differs significantly from the random data curve. This indicates that the recoverability rates observed for the syntactic parameters are also being influenced by the existence of dependence relations between different syntactic parameters. The normalized test indicates a smaller but still significant variation in feature recoverability even when all features considered had the same prevalence among the dataset.

3.1 Recoverability scores

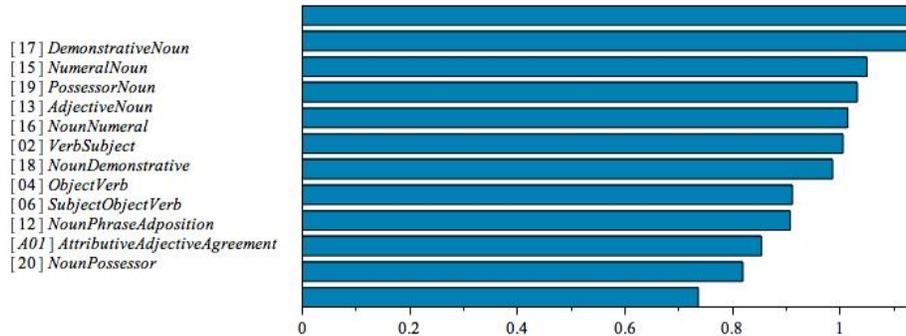


Fig. 2. Corruption (normalized test) of some syntactic parameters.

To each parameter we assign a score, obtained by computing the average Hamming distance between the resulting bit-vector in the corruption experiment and the original one. The lower the score, the more easily recoverable a parameter is from the uncorrupted data, hence from the other parameters.

The resulting levels of recoverability of the syntactic parameters are listed in the table below along with the frequency of expression among the given set of languages. The results of the normalized test are given, for a selection of parameters, in Figure 2.

Parameter	Frequency	Corruption (non-normalized)
[01] Subject-Verb	0.64957267	1.50385541439
[02] Verb-Subject	0.31623933	2.03638553143
[03] Verb-Object	0.61538464	1.56180722713
[04] Object-Verb	0.32478634	1.86186747789
[05] Subject-Verb-Object	0.56837606	1.6709036088
[06] Subject-Object-Verb	0.30769232	1.88596384645
[07] Verb-Subject-Object	0.1923077	1.7879518199
[08] Verb-Object-Subject	0.15811966	1.66993976116
[09] Object-Subject-Verb	0.12393162	1.46596385241
[10] Object-Verb-Subject	0.10683761	1.4907228899
[11] Adposition-Noun-Phrase	0.58974361	1.52427710056
[12] Noun-Phrase-Adposition	0.2905983	1.81512048125
[13] Adjective-Noun	0.41025642	1.82927711248
[14] Noun-Adjective	0.52564102	1.6037349391
[15] Numeral-Noun	0.48290598	1.74969880581
[16] Noun-Numeral	0.38034189	1.94036144018
[17] Demonstrative-Noun	0.47435898	1.87596385121
[18] Noun-Demonstrative	0.38461539	1.87463855147
[19] Possessor-Noun	0.38034189	1.91487951279
[20] Noun-Possessor	0.49145299	1.74102410674
[A 01] Attributive-Adjective-Agreement	0.46581197	1.79102409244

4 Further Questions and Directions

We outline here some possible directions in which we plan to expand the present work on an approach to the study of syntactic parameters using Kanerva Networks.

One limitation of our result is that this scalar score is simply computed as the average of the Hamming distance between the resultant bit-vector and the original bit-vector. The derivability of a certain parameter might vary depending on the family of languages that it belongs to. For example, when a certain language feature is not robust to corruption in certain regions of the Kanerva Network, which means the parameter is not dependent on other parameters, but robust to corruption in all the other regions, we will get a low scalar score. If a feature has a low scalar score in one family of languages, this means that feature is a sharing characteristic of the language group. Otherwise, it might indicate that the feature is a changeable one in the group. Thus, by conducting the same experiments grouped by language families, we may be able to get some information about which features are important in which language family.

It is reasonable to assume that languages belonging to the same historical-linguistic family are located near each other in the Kanerva Network. However,

a more detailed study where data are broken down by different language families will be needed to confirm whether syntactic proximity as detected by a Kanerva network corresponds to historical proximity. Under the assumption that closely related languages remain near in the Kanerva Network, the average of dependencies of a given parameter over the whole space might be less informative globally, because there is no guarantee that the dependencies would hold throughout all regions of the Kanerva Network. However, this technique may help identifying specific relations between syntactic parameters that hold within specific language families, rather than universally across all languages. The existence of such relations is consistent with the topological features identified in [18] which also vary across language families.

One of the main open frontiers in understanding human language is relating the structure of natural languages to the neuroscience of the human brain. In an idealized vision, one could imagine a Universal Grammar being hard wired in the human brain, with syntactic parameters being set during the process of language acquisition (see [1] for an expository account). This view is inspired by Chomsky's original proposals about Universal Grammar. A serious difficulty lies in the fact that there is, at present, no compelling evidence from the neuroscience perspective that would confirm this elegant idea. Some advances in the direction of linking a Universal Grammar model of human language to neurobiological data have been obtained in recent years: for example, some studies have suggested Broca's area as a biological substrate for Universal Grammar, [17]. Recent studies like [13] found indication of possible links between cross linguistic prevalence of syntactic parameters relating to word order structure and neuroscience models of how action is represented in Broca's area of the human brain. This type of results seems to cast a more positive light on the possibility of relating syntactic parameters to computational neuroscience models. Universal Grammar should be seen in the plasticity adaptive rules (storing algorithms) that shape the network structure and that are known to be universal across cortical areas and neural networks. Models of language acquisition based on neural networks have been previously developed, see for example the survey [19]. Various results, [3], [8], [11], [12], [14], have shown advantages of Kanerva's sparse distributed memories over other models of memory based on neural networks. To our knowledge, Kanerva Networks have not yet been systematically used in models of language acquisition, although the use of Kanerva Networks is considered in the work [15] on emergence of language. Thus, a possible way to extend the present model will be storing data of syntactic parameters in Kanerva Network, with locations representing (instead of different world languages) events in a language acquisition process that contain parameter-setting cues. In this way, one can try to create a model of parameter setting in language acquisition, based on sparse distributed memories as a model of human memory. We will return to this approach in future work.

Acknowledgment This work was performed in the last author's Mathematical and Computational Linguistics lab and CS101/Ma191 class at Caltech. The last author was partially supported by NSF grants DMS-1201512 and PHY-1205440.

References

1. M. Baker, *The Atoms of Language*, Basic Books, 2001.
2. N. Chomsky, H. Lasnik, *The theory of Principles and Parameters*, in “Syntax: An international handbook of contemporary research”, pp.506–569, de Gruyter, 1993.
3. Ph.A. Chou, *The capacity of the Kanerva associative memory*, IEEE Trans. Inform. Theory, Vol. 35 (1989) N. 2, 281–298.
4. S. Franklin, *Artificial Minds*, MIT Press, 2001.
5. S.B. Furber, G. Brown, J. Bose, J.M. Cumpstey, P. Marshall, J.L. Shapiro, *Sparse distributed memory using rank-order neural codes*, IEEE Trans. on Neural Networks, Vol. 18 (2007) N. 3, 648–659.
6. M. Haspelmath, *Parametric versus functional explanations of syntactic universals*, in “The limits of syntactic variation”, pp. 75–107, John Benjamins, 2008.
7. T.A. Hely, D.J. Willshaw, G.M. Hayes, *A New Approach to Kanerva’s Sparse Distributed Memory*, IEEE Trans. Neural Networks, Vol. 8 (1997) N. 3, 791–794.
8. P. Kanerva, *Sparse Distributed Memory*, MIT Press, 1988.
9. P. Kanerva, *Sparse Distributed Memory and Related Models*, in “Associative Neural Memories: Theory and Implementation”, M.H. Hassoun, Ed., pp. 50–76, Oxford University Press, 1993.
10. P. Kanerva, *Encoding structure in Boolean space*, in “ICANN 98”, L. Niklasson, M. Boden, and T. Ziemke (eds.), pp. 387–392, Springer 1998.
11. J.D. Keeler, *Capacity for patterns and sequences in Kanerva’s SDM as compared to other associative memory models*, in “Neural Information Processing Systems”, Ed. D.Z. Anderson, pp. 412–421, American Institute of Physics, 1988.
12. D. Kemmerer, *The cross-linguistic prevalence of SOV and SVO word orders reflects the sequential and hierarchical representation of action in Broca’s area*, Language and Linguistics Compass, Vol.6 (2012) N.1, 50–66.
13. A. Knoblauch, G. Palm, F.T. Sommer, *Memory capacities for synaptic and structural plasticity*, Neural Computation, Vol. 22 (2010) 289–341.
14. B. MacWhinney, *Models of the Emergence of Language*, Annual Review of Psychology, 49 (1998) 199–227.
15. M. Marcolli, *Syntactic parameters and a coding theory perspective on entropy and complexity of language families*, Entropy 18 (2016), no. 4, Paper No. 110, 17 pp.
16. G.F. Marcus, A. Vouloumanos, I.A. Sag, *Does Broca’s play by the rules?* Nature Neuroscience, Vol.6 (2003) N.7, 651–652.
17. A. Port, I. Gheorghita, D. Guth, J.M. Clark, C. Liang, S. Dasu, M. Marcolli, *Persistent Topology of Syntax*, arXiv:1507.05134 [cs.CL]
18. J. Poveda, A. Vellido, *Neural network models for language acquisition: a brief survey*, in “Intelligent Data Engineering and Automated Learning – IDEAL 2006”, Lecture Notes in Computer Science, Vol.4224, Springer, 2006, pp. 1346–1357.
19. K. Siva, J. Tao, M. Marcolli, *Spin Glass Models of Syntax and Language Evolution*, arXiv:1508.00504 [cs.CL]
20. K. Shu, M. Marcolli, *Syntactic structures and code parameters*, Math. Comp. Sci. 11 (2017) no. 1, 79–90.
21. K. Shu, S. Aziz, V.L. Huynh, D. Warrick, M. Marcolli, *Syntactic Phylogenetic Trees*, arXiv:1607.02791 [cs.CL]
22. Syntactic Structures of World Languages <http://sswl.railsplayground.net/>