

Example-based Complexity--Syntax and Semantics as the Production of Ad-hoc Arrangements of Examples

Robert John FREEMAN
rjfreeman@email.com

Abstract

Computational linguists have traditionally sought to model language by finding underlying parameters which govern numerous examples. I describe a different approach which argues that numerous examples themselves, by virtue of their many possible arrangements, provide the only way to specify a sufficiently rich set of "parameters".

Essentially I argue for a different relationship between example and parameter. With examples primary, and parameterizations of them secondary, the real "productions". Rather than representing a redundant complexity, examples should actually be seen as a simplification, a basis for the numerous arrangements of their "parameterizations".

Another way of looking at it is to say I argue arrangements of examples, rather than simply revealing underlying parameters, represent in themselves an ignored resource for the modelling of syntactic, and semantic, complexity.

I have implemented a small, working, "shallow parser" based on these ideas.

Introduction--Machine Learning, Data, and Parameterizations

I contrast my work with Machine Learning. There are similarities in the emphasis on the analysis of relationships among data, but there are also differences in the assumptions about the nature of the system. I think there has been a tacit assumption in Machine Learning that language system consists of underlying parameters which generate a variety of

examples. My argument is that you can turn that relationship around and get a great deal more descriptive power in the form of varying parameterizations of the order in a set of examples.

Under the umbrella of Machine Learning I include a wide variety of data based analyses of language which have become popular in recent years. Both distributed and statistical data based models fit in that category: back-propagation networks, Hidden Markov Models, maximum entropy parameterizations. Apart from their emphasis on data, however, they have one thing in common, and in common with earlier symbolic attempts to codify language system. They all hypothesize parameters for distributions of data. I say it is worth considering that the essence of language is not in such underlying parameters but the collections of examples we seek them through. That there are no underlying parameters, only the chaos of example, much as is the case in a population of people (see also Kenneth Pike "analogies between linguistic structure and the structure of society", in de Beaugrande (1991)).

One way to describe this is to say that language might be "irreducibly distributed". A system where a collection of examples is the smallest set which describes all its structure. Although there might be different levels of this independence (along with differing abilities to parameterize: viz. phonology, morphology, syntax). We might contrast irreducibly distributed systems with those which are parametrically distributed, like a letter recognition system. Certainly, however, we could contrast them with statistical systems, where only the likelihood of the outcomes is variable.

R from N and the Descriptive Power of Sets

The best thing about such "irreducibly distributed" systems is their power.

The number of combinations of R objects taken from N is $C(N,R) = N!/(N-R)!R!$. This is the number of "word association classes" N word associations can model, for instance.

The idea that we can model syntactic classes as "word association classes" is not new. There are numerous studies dating from the early 1990's and before which take this approach e.g. Schuetze (1993), Finch (1993); and Powers (1996) lists references back to Pike's Tagmemics. What is different in my approach is the assumed relationship between these classes and the data which reveal them. If the variety of example can be generated by a small number of abstract parameters then we expect one set of relationships among that data to be more important than the others. If on the other hand we consider the full range of relationships possible among all the examples then we have an enormous range of structure at our disposal. Given the problems we have had describing language according to parameters, it is surprising that we have not more widely considered the attraction of this power.

Consider the evidence that we need this power:

a) Structure

Collocation, phraseology. The data based analysis of language has bought home more and more strongly that some structure is beyond any logic we can enumerate. Face to face with the reality of use this realization has been most widely accepted in areas of linguistics which deal with language acquisition and teaching. Examples of relevant discussions are Pawley and Syder (1983), Nattinger (1980), Weinert (1995). We are talking about explaining why you might say "strong tea" but not "powerful tea".

In practical terms a processor based fundamentally on distributions should be able to tell that "strong tea" is idiomatic and "powerful tea" less so because the "word association

distributions", say, of "strong" and "powerful" are different in detail, though not in generalities. A system based on labels, an assumption of underlying parameters, will not be able to do that (for a set of labels smaller than the set of all such distinct utterances).

An irreducibly distributed representation gives us the power to model collocation. We would need a different syntactic class for every collocational restriction otherwise.

b) Meaning

$N!/(N-R)!R!$ groupings give you an essentially infinite set of configurations. We have the power to associate a different configuration with everything we might ever want to say, if we like. In fact, by default we will do so. This means we have the power to represent not only syntactic idiosyncrasy, but the complexity of meaning, directly.

The idea of meaning implied by the association is interesting in itself. It is an organization of data. But this is reasonable. And if we accept it then we have a fundamental definition of meaning in terms we can quantify. Meaning is synonymous with an organization of data: events, observations. New organization equals new meaning.

There is an interesting topical analogy to be made here: a Web search engine. In a sense any collection of documents found "represent" the meaning of a set of search keys. There are many more subtleties of collection possible than can ever be labeled in an index.

In a way my argument is just that if we want to model the full complexity of syntactic restriction, or semantic subjectivity, we have no choice but to demote categories from being central, make them a product, and base them on the reorganization of content much the way they are treated in most Web search engines.

Such an irreducibly distributed definition explains many puzzling properties of thought. It provides a natural mechanism for how:

- new concepts can be created (novel reorganization of old examples--"Aha!")
- new meaning can be communicated (I force you to reorganize your examples in the way I've just reorganized mine)
- language (and conceptual) drift can occur (slow shift in balance of examples).

As well as the usual useful properties of distributed representations:

- flexibility (the group can vary)
- robustness (it does not matter if a few elements are missing)
- ambiguity (intersection sets)
- subjectivity (sub-sets etc.)

There is also an interesting tie in between this (meaning, and the primacy of data over parameter) and the vigorous "rebel" linguistic school of Systemic Functional Grammar. Most importantly in SFG the only irreducible definition of meaning, or structure, is a set of contrasts between events, or observations. Unfortunately in SFG an overemphasis on abstract parameters (function/meaning) means that in practice the full power of contrasts among sets to model complexity is not applied. Nevertheless, there are strong parallels between my model and the core tenets of Systemic Functional Grammar. I find that a natural analysis according to the principles I have outlined above results in structure along lines of functional category. In fact the association groupings on which I base my analysis lead me to propose an "inverse" relationship (in a sense that can be precisely defined) between functional category, about which SFG is described, and categories based on syntactic regularities of the type which have traditionally been seen as important.

A Simple "Association Parser"

I have implemented a small "association parser" based on these principles and the initial results have been interesting. I provide a list of typical "parses" in the appendix. Essentially it scores the grammaticality and provides a structural breakdown of each string of words it is

presented with. Among more interesting observations, as I mentioned above, is the fact that my parser seems to naturally identify structure along lines of functional equivalence. Rather like the kind of analysis a Systemic Functional Grammarian might favor.

Since processing is essentially a search over a database for similar examples the main bottleneck is the inefficiency of a serial processor for nearest neighbor search. There are two key complexities. The search over one I have managed to reduce to linear time. The other remains to be resolved.

References

- Beaugrande, Robert de (1991) *Linguistic Theory: The Discourse of Fundamental Works*, section 5.84, Harlow: Longman.
- Finch, Steven (1993) *Finding Structure in Language*. Ph.D. Thesis, University of Edinburgh.
- Nattinger, James R. (1980) *A lexical Phrase Grammar for ESL*, TESOL Quarterly Vol. XIV, No. 3, pp. 337-334.
- Pawley, A. & Syder F. (1983) *Two puzzles for linguistic theory: nativelike selection and nativelike fluency*, in J. Richards and R. Schmidt (eds.) 1983: *Language and Communication*, pp. 191-226. London: Longman.
- Powers, D. M. W. (1996) *Unsupervised learning of linguistic structure: An empirical evaluation*, *International Journal of Corpus Linguistics* 1#2.
- Schuetze, H. (1993) *Distributed Syntactic Representations with an Application to Part-of-Speech Tagging*, 1993 IEEE International Conference on Neural Networks, p1504-9 vol. 3.
- Weinert, Regina. (1995) *The Role of Formulaic Language in Second Language Acquisition: A Review*, *Applied Linguistics*, Vol. 16, No. 2, pp. 181-205.

Appendix--Examples of Parses Produced by my "Association Parser" Prototype

make some products

Parsed: (make (some products)), score: 1.329954

Parsed: ((make some) products), score: 0.023665

make some money

Parsed: (make (some money)), score: 1.555408

Parsed: ((make some) money), score: 0.042059

make a car

Parsed: (make (a car)), score: 5.689303

Parsed: ((make a) car), score: 2.120204

make another car

Parsed: (make (another car)), score: 1.642482

Parsed: ((make another) car), score: 0.189554

make another try

Parsed: ((make another) try), score: 0.051537

Parsed: (make (another try)), score: 0.039471

go with the president

Parsed: ((go with) (the president)), score:
7.983729

Parsed: (go (with (the president))), score:
4.620297

Parsed: (go ((with the) president)), score:
0.771305

Parsed: (((go with) the) president), score:
0.318181

Parsed: ((go (with the)) president), score:
0.065606

I try to go

Parsed: (i ((try to) go)), score: 4.343059

Parsed: (((i try) to) go), score: 1.297454

Parsed: ((i (try to)) go), score: 1.174891

Parsed: (i (try (to go))), score: 0.553270

Parsed: ((i try) (to go)), score: 0.474397

the election results

Parsed: (the (election results)), score: 89.247596

Parsed: ((the election) results), score: 15.212562

they held an election

Parsed: (they (held (an election))), score:
0.000238

Parsed: ((they held) (an election)), score:
0.000007

Parsed: (((they held) an) election), score:
0.000000

go with her

Parsed: ((go with) her), score: 9.073902

Parsed: (go (with her)), score: 0.107435