Language and the rest of cognition: computational, algorithmic and possible neural commonalities

Shimon Edelman
Department of Psychology
232 Uris Hall, Cornell University
Ithaca, NY 14853-7601, USA
http://kybele.psych.cornell.edu/~edelman

June 25, 2003

Abstract

The computational program for theoretical neuroscience proposed by Marr and Poggio [1] calls for a study of biological information processing on several distinct levels of abstraction. At each of these levels — computational (defining the problems and considering possible solutions), algorithmic (specifying the sequence of operations leading to a solution) and implementational — significant progress has been made in the understanding of cognition. In the past three decades, computational principles have been discovered that are common to a wide range of functions in perception (vision, hearing, olfaction) and action (motor control). More recently, these principles have been applied to the analysis of cognitive tasks that require dealing with structured information, such as visual scene understanding and analogical reasoning. Insofar as language relies on cognition-general principles and mechanisms, it may be possible to capitalize on the recent advances in the computational study of cognition by extending its methods to linguistics.

Much of the discussion surrounding the integration of linguistics with the other cognitive sciences has traditionally been focusing on arguments for [2] and against [3, 4] the modular [5] status of language. Even if language is a module, however, it may still rely on the same computational principles (in the sense of Marr [1]), and may be supported by the same mechanisms, as the other cognitive functions. To explore this possibility, we need to bring together ideas from several fields. The first, a natural home for the integration project, is cognitive linguistics [6], which consistently produces valuable insights into the psychology of language, yet is little concerned with algorithmic or implementational issues. The second is computational linguistics [7] (including statistical natural language processing [8]), which examines the mathematical nature of language-related tasks and generates important applications, yet pays little attention to behavioral or neurobiological issues. Lastly, there is the Marr-Poggio computational framework [1], which is used across cognition and which spans all the relevant levels of analysis, but has not yet been extended to the study of language.

This paper discusses some of the general computational principles that emerge as useful for understanding cognition, focusing on those that are likely to be especially relevant in dealing with structured knowledge. It then brings these principles to bear on a theory of language that is rooted both in cognitive and in computational linguistics, and that views language as an incrementally learnable system of redundant, distributed representations akin to those found by neurobiologists in olfaction, audition and vision.
1 Common principles of cognitive representation and processing

The view that cognition hinges on internal representation of knowledge [9, 10] is widely accepted in linguistics, psychology and neuroscience. A representational state in a cognitive system is characterized by its covariation with certain aspects of the relevant state of affairs in the world, and, crucially, by having counterfactually supported effects on the rest of the system (in linguistics, this corresponds to the requirement of psychological reality [11, 12]). The reality of a representation can be indicated by double dissociation (in patients [13] or normal subjects [14]) or by priming (behavioral [15, 16] or neural [17]), and its causal effectiveness — by direct intervention (microampere-level current injection at the appropriate brain site [18]).

Theories that posit distributed, overlapping, graded representations [19] have enjoyed considerable explanatory success in contexts as diverse as olfaction, vision, reasoning and memory. In particular, researchers are now able to identify a few classes of computational processes, operating over distributed representations, that are common to a wide range of cognitive tasks. Some examples of such general-purpose computational building blocks are: (1) density estimation and hypothesis weighting in the service of probabilistic inference (e.g., in conceptual learning [20] and in vision [21]); (2) function approximation in the service of learning from examples (e.g., in vision [22] and in motor control [23]), and (3) dimensionality reduction in the service of feature detection and categorization (e.g., in language [24] and in vision [25]).

In some cases, these abstract principles have been mapped onto the function of the brain and its circuitry, resulting in explanatory models that span all three levels of Marr’s program. For example, in olfaction the anatomy and the physiology of the pathway leading from the sensory epithelium to the glomeruli in the olfactory bulb [26, 27] can be seen as filtering data through a bank of radial basis functions [23]. This operation implements what is known to be a universal approximation algorithm [28] that can be used in learning from examples [23]. The same algorithmic approach can support visual object recognition, as demonstrated by the Chorus of Prototypes model [29], in which the stimulus is represented by its similarities to (processed) memory traces of past stimuli. Recent single-cell studies in the monkey found neurons that are broadly and redundantly tuned to particular object categories [30] and that embody an ensemble representation of inter-object similarities that is veridical with respect to the distal stimuli [31]. Both these findings have been predicted by the Chorus of Prototypes model [32, 29].

2 Dealing with structure: a special challenge?

To be relevant to language, the computational principles behind these findings must be extended to situations that require highly structured representations. Recent work in various areas of cognition has been pursuing precisely such an extension. For example, in vision the Chorus of Fragments model (derived from the Chorus of Prototypes), which aims at dealing with structured objects and scenes [33], is based on the twin principles of distributed representation by similarities (mentioned above) and of the use of visual space to anchor the various shape fragments (cf. [34]; see Box 1: The Role of Space). Likewise, in complex analogy tasks a similarity-based model performs very well when the distributed representations it uses are made to reflect the structure of the input [35].

Such examples show that the goal of structure-sensitive processing by a distributed architecture is less forbidding than commonly thought, and that it is already within reach of cognition-general principles and mechanisms. This should not be surprising: functional (computation-level) analogies between language and vision suggest that the need to deal with structure is not unique to the former [36]. For instance, the treatment of a sentence with an embedded relative clause may be compared to the processing of a scene with occlusion.
A cognitive approach to language, which is based on these foundations, and which casts the relevant computational, algorithmic and implementational issues in cognition-general terms, is outlined in the next section.

3 Treatment of structure in computational cognitive linguistics

3.1 Computational approach: the Chorus of Phrases and Construction Grammar

When applied to language, the idea of a distributed representation of structure based on similarities to multiple structured exemplars (called the Chorus of Fragments in the setting of visual scene processing [33]) translates into a Chorus of Phrases: a redundant ensemble of potentially overlapping, mutually reinforcing phrase fragments that, as Langacker puts it, “motivate” the sentence they cover:

“... rather than seeing a composite structure as an edifice constructed out of smaller components, we can treat it as a coherent structure in its own right: component structures are not the building blocks out of which it is assembled, but function instead to motivate various aspects of it.” [6], p.453; italics in the original.

A simplified illustration of the Chorus of Phrases (CoPH) in action is shown in Figure 2, where a stimulus (which could be an entirely novel sentence) evokes a cloud of associations, pointing to snippets of previously encountered phrases, each of which approximately matches parts of the input, and which together cover all of it.

On the abstract, computational [1]) level, the construal of language exemplified by the CoPH approach differs radically from that of rule-based generative theories such as the Minimalist Program [37]. The basic theoretical challenge at this level is to specify what is it that needs to be done in comprehension and production. According to the CoPH framework, comprehension involves constructing a distributed representation of the stimulus in terms of its structure-dependent similarities to multiple stored exemplars, which convey information both about form (the exemplars are, generally, patterns with slots; see Figure 2) and about meaning. Production consists of letting a set of exemplars chosen for their semantics interact and constrain each other until a fully specified linear sequence of terminals is ready for output.

This distributed approach, which does not distinguish between syntactic and semantic representations and processes, is broadly compatible with the tenets of the Cognitive school in linguistics [6], and, more specifically, with Construction Grammar [38, 39, 40]. CoPH is, however, more than a metaphor for constructions, for several reasons. First, CoPH is deeply rooted in computational principles (multidimensional similarity spaces, distributed representations) and neural mechanisms (receptive fields and maps) that proved useful in analyzing other aspects of cognition. Second, by steering the goals of syntactic (and semantic) analysis towards those of cognition in general, CoPH brings to the fore a collection of mathematical tools hitherto not considered by linguists (see Box 2: Mathematical tools). Third, an implemented model of language acquisition and processing situated within the CoPH framework provides empirical support and constraints for the construction grammar theories, as described briefly below.

3.2 Algorithmic and implementational issues: the ADIOS model

The pattern acquisition algorithm behind this working model of language acquisition and processing [41] (ADIOS, or Automatic DIstillation Of Structure) learns, in an unsupervised fashion, a streamlined representation of linguistic structures from untagged, large-scale natural-language corpora. The algorithm represents
sentences as paths on a graph whose vertices are words. Significant patterns are defined as sets of paths in which a common prefix and suffix form a context surrounding a slot where distributionally equivalent [42] elements may appear. Such patterns, determined by recursive context-sensitive statistical inference, form new vertices. Linguistic constructions are encoded by trees composed of significant patterns and their associated equivalence classes. An entire utterance is typically represented by several such constructions (a Chorus of Phrases; cf. Figure 2), which may be activated to different degrees, depending on their fit to the input. Previously unseen inputs are processed by pursuing structural and lexical similarities to familiar patterns. When subjected to a standard test of English as a Second Language (ESL) proficiency, the model attained a level of performance considered to be “intermediate” for 9th-grade students, despite having been trained on a corpus (CHILDES [43]) containing transcribed speech of parents directed to small children [44].

The probabilistic principle that drives the context-sensitive, hierarchical pattern abstraction process in the ADIOS model is closely related both to the notion of “suspicious coincidences” long thought to be the key to unsupervised learning in neural systems [45, 46] and to the Minimum Description Length (MDL) criterion for representational efficiency [47]. Intuitively, two elements — such as two members of a potential linguistic construction or two fragments of a visual object — belong together to the extent that the probability of their joint appearance is higher than the product of the probabilities of their individual appearances; coding such elements as one results in a more concise representation. It has been conjectured that these principles, which can support structured learning in vision [48, 49, 50] and in language [47, 51, 44], may provide “common foundations for cortical computation” [52].

4 Conclusions

Computational cognitive science holds that a comprehensive theory of any information processing task must lead to its understanding on several levels of abstraction [1]. Although distinct, these levels cannot be studied independently, lest theorizing loses touch with psychological and neurobiological reality, or, conversely, the neurobiology becomes too myopic [29]. Accordingly, the framework for computational cognitive linguistics outlined here is informed by the top-down computational and algorithmic principles of context-dependent probabilistic learning and is based on a bottom-up implementational scheme that is ubiquitous in the brain: computing with connections, which carry dynamically unfolding neural activation patterns and which support low-dimensional, distributed, redundant, graded representations. The vision of language it offers should boost cognitively oriented theories such as Construction Grammar [40, 39] and help connect them to rich repositories of computational knowledge (from learning theory, probability and information, and natural language processing) and empirical data (from psychophysics and neuroscience) about the brain.

Acknowledgments

I thank Zach Solan, Eytan Ruppin, David Horn, Roni Katzir and Rick Dale for stimulating discussions.

References


Box 1: The Role of Space in Representing Structure

The idea that space should serve as a natural scaffolding for supporting structured representations, whose roots go back to the ancient mnemonic Method of Loci ([53], p.137), is stated forcefully in Wittgenstein’s *Tractatus* ([54], proposition 3.1431):

The essential nature of the propositional sign becomes very clean when we imagine it made up of spatial objects (such as tables, chairs, books) instead of written signs. The mutual spatial position of these things then expresses the sense of the proposition.

In recent years, space has been conjectured to play a central role both in linguistics and in cognition in general. Consider, for example, the notion of iconicity in syntax [55]: “... not only motor but also cognitive operations such as language, which do not appear to have any intrinsic spatial organization, are maintained in registration with spatial systems, and [...] this attention-requiring linkage confers a processing advantage” [56]. The iconicity hypothesis is intimately connected to Construction Grammar: linguistic freezes or prefabs [57] that are spatially (or, equivalently, temporally; cf. Figure 2) iconic become constructions when parameterized [58]. Indeed, the patterns and their associated equivalence classes in the ADIOS model [44] are just such parameterized constructions. Psycholinguistic and neuropsychological evidence in support of linguistic iconicity has been reviewed recently in this journal [59].

In vision, sorting shape cues by their location in the visual field goes a long way towards solving the binding problem in the representation of object and scene structure [60, 34, 33]. In particular, various components of a scene or an object need not be bound to each other in any special manner, as long as each of them is bound to its proper location in the visual space, merely by virtue of its appearance there.

The spatial scaffolding approach to the representation of visual structure is consistent with the omnipresence in the monkey inferotemporal and prefrontal cortex of what+where neurons, which are both shape-tuned (signaling what is the stimulus), and location-selective (signaling where it appears) [61, 62]. On a larger scale, the neural substrate of the perceptually defined external space may be the cortical surface itself, as indicated by the ubiquity of low-dimensional map-like representations [63] in vision [64], olfaction [65], and audition [66].

In those areas of the human brain that support language, the counterpart to the visual what+where neurons may be what+when neurons, tuned to particular structures appearing in a particular sequence (as illustrated in Figure 2). The possible role of temporal response properties of neuron assemblies in implementing sequence-sensitive processing has been discussed by Pulvermüller [67]; parallels between the brain representations of space and time in vision and in audition have been pointed out, among others, by Shamma [66].
Box 2: Mathematical Tools for Computational Cognitive Linguistics

From the standpoint of methodology, a computationally motivated approach to cognitive linguistics does not imply pitching “mathematics versus psychology” (an expression used by Tomasello as a section heading in his introduction to *The New Psychology of Language* [68], p.ix). Rather, the usual tools of mathematical linguistics (such as formal languages, \(\lambda\)-calculus and symbolic logic) should be traded in for new ones. Some of these, which proved well-suited for analyzing distributed representations in various areas of cognition, are listed below.

**Syntax: from constituent trees to string cover.** Computational learning theory offers various tools capable of dealing with distributed, potentially overcomplete [69] representations of sequence data. One of these is string kernels, a representation that tallies the occurrences of specific symbols in specific locations, and supports reasoning about global properties of the sequence probed in this manner and, in particular, about features that can help classify it [70]. Similar methods are increasingly in demand in computational biology, because of the sequential nature of the data in both domains, and, specifically, because of the close analogy between text analysis by the identification of multiple, overlapping local patterns on the one hand, and hybridization approaches to DNA sequencing on the other hand. Recent developments in this field include derivations of the algorithmic complexity of specifying a string by its substrings [71, 72, 73]. This approach is distinct from (and more relevant for our present purposes than) treebank-based parsing (combining multiple local or partial parse trees [74, 75]) in that the cover it seeks need be neither precise, nor exclusive.

**Semantics: from functions to constructions and relations.** According to the Chorus of Phrases metaphor (Figure 2), the representation of a sentence by an ensemble of active units can be approximately described as a *relation* (namely, as the subset of units whose activity exceeds some threshold). As in Construction Grammar [39], this representation captures both semantic and syntactic information about the input. Interestingly, recent work in computational semantics addressing various problematic aspects of compositionality suggests that systematicity of meaning is better served by defining meaning as a relation over sentence parts [76] (rather than as a function of the parts, as stipulated by the classical, Fregean approach).

**Acquisition: from parameter setting to structure discovery.** The ascendancy of the generative grammar and the accompanying innateness postulate over competing distributional and behaviorist ideas in the 1960s can be ascribed in a large part to the inadequacy of the contemporary statistical inference methods and the perceived inability of association-based learning to handle recursion. Statistics, however, need not be limited to counting word frequencies: in computer science, the integration of advanced statistical inference (including Bayesian methods, the Minimum Description Length principle and other related information-theoretic tools), progress in computational learning theory, efficient algorithms, and cheap hardware led to important conceptual progress, as well as to practical achievements [8]. Likewise, learning need not be limited to the establishment of pairwise associations: bounded-depth recursively structured patterns can be learned from examples, by efficient algorithms that rely on modern statistical inference [41, 44] (see [51] for an overview of the recent progress in this field).
Issues for Future Research

1. Recent psycholinguistic evidence indicates that listeners and readers routinely settle for “good enough” representations of the linguistic material they face, rather than seeking an exhaustive parse or even just a fully disambiguated semantic interpretation [77, 78, 79]; cf. Figure 1, right. How do these findings relate to the cluster of phenomena in visual psychophysics known as “change blindness” [80], which indicate that subjects do not fully parse visual scenes either?

2. More generally, how does the final product of language relate to that of vision? Goldberg’s thesis (“constructions all the way down” [39]) can be taken to imply that the Chorus of Phrases evoked by an utterance or a text is just about all there is to its interpretation. There is a clear parallel between this stance and the conjecture that in vision the Chorus of Fragments is an adequate, and in fact the only reasonable, bottom line [34].

3. On an abstract computational level, construction-based approaches, and in particular the Chorus of Phrases framework, readily integrate themselves into the rest of cognition. On the level of mechanism, however, the details have yet to be worked out. A crucial question is whether or not it is possible to avoid altogether the need to manipulate constructions dynamically, rather than through a pre-wired network. This is important because the act of binding a variable to a value (or inserting a constituent into a construction) dynamically is deeply problematic in the context of a neural implementation [33].

4. The ability of humans to do algebra or to program computers attests to the existence of some mechanism in the brain that at least creates the semblance of dynamic binding. People, however, must be trained for years before they become good at this kind of symbol manipulation, and it would be prudent to make it a means of last resort in a theory of any cognitive phenomenon that is more mundane than programming in Lisp. The same consideration applies to a related issue, recursion: although it has been recently reaffirmed by some linguists as the epitome of human uniqueness [81], humans are notoriously bad at deep recursion [82] (and shallow recursion, as well as the manipulation of complexity-controlled constructions, can be handled by finite means such as the ADIOS representation [44]). These considerations suggest that dynamic binding and deep recursion may both be supported in the brain by a virtual machine (cf. [83], p.209) that is difficult to build and expensive to maintain and operate, and that is at least once removed from the neural mechanisms that are so good at supporting everyday cognition.
Figure 1: Left: there is a task-level analogy between interpreting a composite scene with occlusion and the processing of a sentence that contains an embedded relative clause (adapted from M. Minsky [36], p.269). The existence of such analogies between vision and language on the abstract level of the computational tasks [1] faced by the brain, along with the uniformity of the underlying low-level cortical mechanisms [52], suggests that cross-cognition commonalities should be sought also on the algorithmic level. Right: the postcard shown here (Apolinère Enameled by M. Duchamp) can be used to make the same point about parallels between language and vision (cf. the occlusion of the girl’s legs by the bed-frame), and more. For instance, Duchamp’s painting could be represented (and understood) in terms of its local similarities to various familiar images (Chorus of Fragments [34, 33]), which need not match the scene perfectly [29]; likewise, an utterance could be represented (and understood) in terms of the cloud of constructions (Chorus of Phrases [44]) it evokes, as illustrated in Figure 2. Furthermore, just as many viewers fail to notice that the bed in this scene would be useless (look closely at the frame), subjects exposed to ungrammatical sentences of moderate complexity may rate them as more felicitous than similarly structured grammatical ones (e.g., the sentence “The apartment that the maid who the service had sent over was well decorated” tends to be rated as no worse [82] – and, in some settings, better [84] – than “The apartment that the maid who the service had sent over was cleaning every week was well decorated”; cf. [85, 86, 87]).
I read the news today, oh boy, about a lucky man who made the grade.

Figure 2: An illustration of the Chorus of Phrases in sentence processing (for actual examples produced by the ADIOS model, see [41, 44]). An input sentence is shown along with a subset of phrases it evokes from memory, each of which matches some word, sequence of words, or, generally, a parameterized pattern (in cartouche: I heard, read, saw, watched a story about a man) in the input. The unfolding of each pattern’s activation (which reflects its time-domain “receptive field”) may be important [67], but even without it the ensemble of active patterns is a highly informative representation (just as its counterparts in vision are [29, 33]). The members of the ensemble disambiguate each other by supplying multiple interacting constraints on the interpretation. Consequently, it should be possible to process various queries about the input, both syntactic (voice, aspect, etc.) and semantic (thematic, connotational, conceptual). Moreover, it may be possible to use for that purpose generic cortical mechanisms [52, 88] that would map the distributed phrase activation patterns onto the corresponding required outputs, as in the scenario of function approximation found across cognition [23, 25]. From the computational standpoint, it is interesting to observe that one can reconstruct the input sentence itself, should that be required for some reason, from a number of subsequence (phrase or pattern) queries that is on the order of \( n \log \alpha + \alpha \log n \), where \( n \) is the length of the sentence and \( \alpha \) is the size of the lexicon [71]. This computational complexity, which is quite benign in view of the \( \alpha \)-fold parallelism inherent in a distributed lexicon, can be further reduced by allowing approximate (in Valiant’s sense [89]) matching [72].