Rich Syntax from a Raw Corpus: Unsupervised Does It

Shimon Edelman
Department of Psychology
Cornell University
Ithaca, NY 14853, USA
se37@cornell.edu

Zach Solan, David Horn, Eytan Ruppin
Sackler Faculty of Exact Sciences
Tel Aviv University
Tel Aviv, Israel 69978
{*rsolan,horn,ruppin*}*@post.tau.ac.il*

Abstract

We compare our model of unsupervised learning of linguistic structures, ADIOS [1], to some recent work in computational linguistics and in grammar theory. Our approach resembles the Construction Grammar in its general philosophy (e.g., in its reliance on structural generalizations rather than on syntax projected by the lexicon, as in the current generative theories), and the Tree Adjoining Grammar in its computational characteristics (e.g., in its apparent affinity with Mildly Context Sensitive Languages). The representations learned by our algorithm are truly emergent from the (unannotated) corpus data, whereas those found in published works on cognitive and construction grammars and on TAGs are hand-tailored. Thus, our results complement and extend both the computational and the more linguistically oriented research into language acquisition. We conclude by suggesting how empirical and formal study of language can be best integrated.

1 Unsupervised learning through redundancy reduction

Reduction of redundancy is a general (and arguably the only conceivable) approach to unsupervised learning [2, 3]. Written natural language (or transcribed speech) is trivially redundant to the extent it relies on a fixed lexicon. This property of language makes possible the unsupervised recovery of words from a text corpus with all the spaces omitted, through a straightforward minimization of per-letter entropy [4].

Pushing entropy minimization to the limit would lead to an absurd situation in which the agglomeration of words into successively longer "primitive" sequences renders the resulting representation useless for dealing with novel texts (that is, incapable of generalization; cf. [5], p.188). We observe, however, that a word-based representation is still redundant to the extent that different sentences share the same word sequences. Such sequences need not be contiguous; indeed, the detection of paradigmatic variation within a slot in a set of otherwise identical aligned sequences (syntagms) is the basis for the classical distributional theory of language [6], as well as for some modern NLP methods [7]. The *pattern* — the syntagm and the *equivalence class* of complementary-distribution symbols that may appear in its open slot — is the main representational building block of our system, ADIOS (for

Automatic DIstillation Of Structure) [1].[1]

Our goal here is to help bridge statistical and formal approaches to language [9] by placing our work on the unsupervised learning of structure in the context of current research in grammar acquisition in computational linguistics, and at the same time to link it to certain formal theories of grammar. Section 2 outlines the main computational principles behind the ADIOS model (for algorithmic details and empirical results, see [1, 10]). Sections 3 and 4 compare our model to select approaches from computational and formal linguistics, respectively. We conclude with a focus on the challenges ahead, discussed in section 5.

2 The principles behind the ADIOS algorithm

The representational power of ADIOS and its capacity for unsupervised learning rest on three principles: (1) probabilistic inference of pattern significance, (2) context-sensitive generalization, and (3) recursive construction of complex patterns. Each of these is described briefly below.

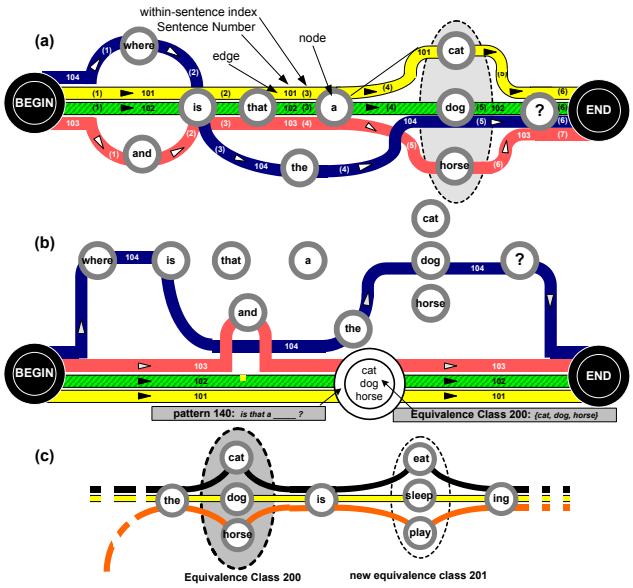

Figure 1: (a) The directed multi-graph for a simple corpus consisting of four partially "bundled" sentences, which form the pattern is that a {dog, cat, horse} ?. (b). The abstracted pattern and the equivalence class associated with it are highlighted (edges that belong to sequences not subsumed by this pattern, e.g., #104, are untouched). (c) The identification of new significant patterns is done using the acquired equivalence classes (e.g., #200). For details, see [1].

Probabilistic inference of pattern significance. ADIOS represents a corpus of sentences as an initially highly redundant directed graph, which can be informally visualized as a tangle of strands that are partially segregated into *bundles*. Each of these consists of some strands clumped together (Figure 1); a bundle is formed when two or more strands join together and run in parallel and is dissolved when more strands leave the bundle than stay

[1]The symbols may be letters or morphemes; in addition to text corpora and transcribed speech (CHILDES [8]), ADIOS has been tested on gene sequence data and on music.

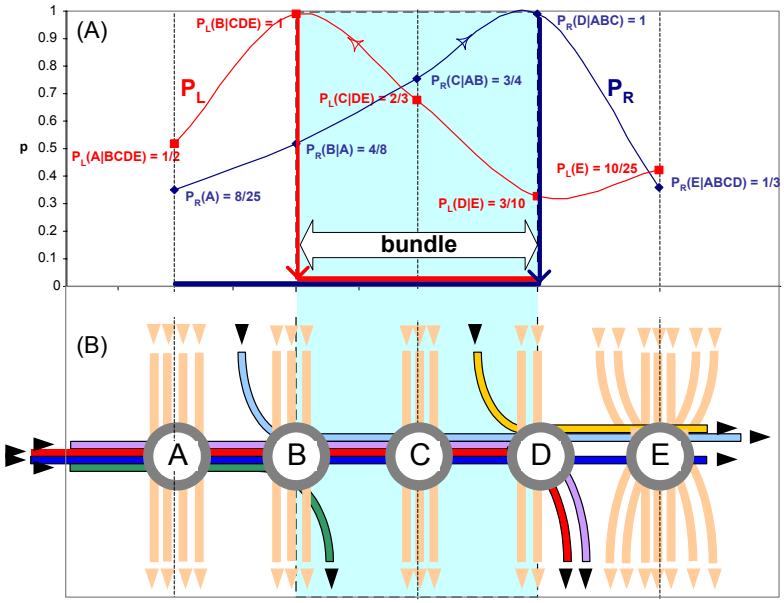

Figure 2: The bundle core is defined dynamically, with respect to graph traversals. The main idea is to predicate the membership of two elements e_i and e_j in the same bundle on increments in the probability of getting from e_i to e_j while following the paths in the forward and the backward directions: an element newly added to the candidate bundle is kept if its addition causes the corresponding probabilities P_R and P_L to increase (e.g., $P(A) < P(B|A) < P(C|AB) < P(D|ABC) > P(E|ABCD)$, so the bundle ends at D). The relevant probabilites are readily available in the graph: for example, $P(C|AB) = 3/4$ because there are four paths that travel through A and B, only three of which continue to C.

in. In a given corpus, there will be many bundles, with each strand (sentence) possibly participating in several. The computational challenge we face is how to identify significant bundles so as to balance high compression (small size of the bundle "lexicon") against good generalization (the ability to generate new grammatical sentences by splicing together various strand fragments each of which belongs to a different bundle). The intuition behind our algorithmic approach to this problem is sketched in Figure 2.

Context sensitivity of patterns. A pattern is an abstraction of a bundle of sentences that are identical up to variation in one place, where one of several symbols — the members of the equivalence class associated with the pattern — may appear (Figure 3). Because this variation is only allowed in the context specified by the pattern, the generalization afforded by a set of patterns is inherently safer than in approaches that posit globally valid categories ("parts of speech") and rules ("grammar"). The reliance of ADIOS on many context-sensitive patterns rather than on traditional rules can be compared both to the Construction Grammar (discussed later) and to the following observation made by Langacker ([11], p.46): "Out of this sea of particularity speakers extract whatever generalizations they can. Most of these are of limited scope, and some forms cannot be assimilated to any general patterns at all. Fully general rules are not the expected case in this perspective, but rather a special, limiting case along a continuum that also embraces totally idiosyncratic forms and patterns of all intermediate degrees of generality."

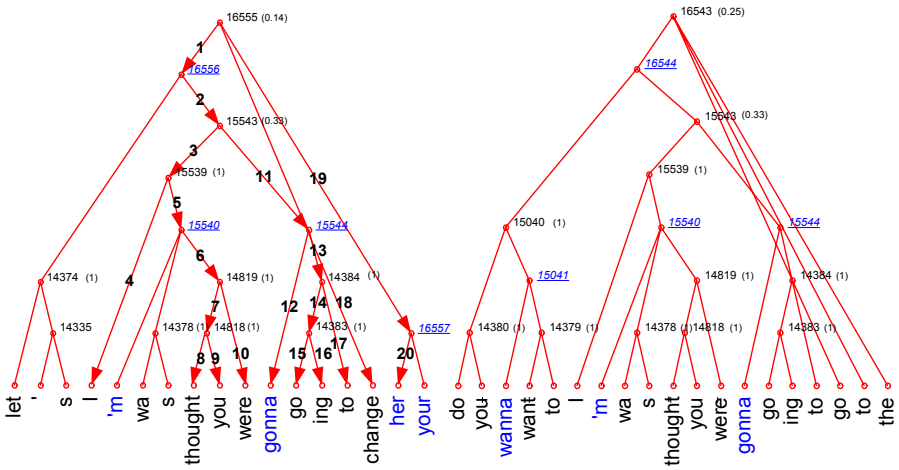

Figure 3: Two typical patterns extracted from a subset of the CHILDES collection [8]. Hundreds of such patterns and equivalence classes (underscored) together constitute a concise representation of the raw data. Some of the phrases that can be described/generated by these patterns are: let's change her...; I thought you gonna change her...; I was going to go to the...; none of these appear in the training data, illustrating the ability of ADIOS to generalize. The generation process, which operates as a depth-first search of the tree corresponding to a pattern, is illustrated on the left. For details see [1].

Hierarchical structure of patterns. The ADIOS representation meets the challenge of capturing long-range dependencies through two related mechanisms: hierarchical nesting of patterns and pattern recursion through self-reference. The graph that serves as the basic data structure for ADIOS is rewired every time a new significant pattern is detected, so that a bundle of strings subsumed by the pattern is represented by a single new arc corresponding to it; this rewiring is context-specific, just as the patterns themselves are [1]. Following the rewiring, potentially far-apart symbols that used to straddle the newly abstracted bundle become close neighbors. Patterns thus become hierarchically structured in that their constituent symbols may be either terminals (i.e., fully specified strings) or other patterns. Moreover, patterns are not precluded from referring to themselves, which in principle opens the door for true recursion (in practice, the depth of recursion is limited by the data that caused the successive rewiring, if not by implementational constraints).

3 Related computational approaches

In natural language processing (NLP), a distinction is usually made between unsupervised learning methods that attempt to find good structural primitives and those that merely seek good parameter settings for predefined primitives. ADIOS clearly belongs to the first category. Moreover, our algorithm is capable of learning from raw data, whereas most other systems start with corpora annotated by part of speech tags [12], or even rely on treebanks (collections of hand-parsed sentences [13]). Of the many such methods, we can mention here only a few.

Global grammar optimization using tagged data. Stolcke and Omohundro [14] learn structure (the topology of a Hidden Markov Model, or the productions of a Stochastic Context Free Grammar), by iteratively maximizing the probability of the current approximation to the target grammar, given the data. In contrast to this approach, which is global in that

all the data contribute to the figure of merit at each iteration, ADIOS is local in the sense that its inferences only apply to the current bundle candidate. Another important difference is that instead of general-scope rules stated in terms of parts of speech, we seek context-specific patterns. Perhaps because of its globality and unrestricted-scope rules, Stolcke and Omohundro's method has "difficulties with large-scale natural language applications" [14]. Similar conclusions are reached by Clark, who observes that POS tags are not enough to learn syntax from ("a lot of syntax depends on the idiosyncratic properties of particular words." [15], p.36). Clark's own algorithm [16] had attempted to learn a phrase-structure grammar from tagged text, by starting with local distributional cues, then filtering spurious non-terminals using a mutual information criterion (namely, requiring high MI between pattern prefix and suffix). In the final stage, his algorithm clustered the results to achieve a minimum description length (MDL) representation, by starting with maximum likelihood grammar, then greedily selecting the candidate for abstraction that would maximally reduce the description length. In its greedy approach to optimization (but not in its local search for good patterns or its ability to deal with untagged data), our approach resembles Clark's.

Probabilistic treebank-based learning. Bod, whose algorithm learns by gathering information about corpus probabilities of potentially complex trees, observes that "[...] the knowledge of a speaker-hearer cannot be understood as a grammar, but as a statistical ensemble of language experiences that changes slightly every time a new utterance is perceived or produced. The regularities we observe in language may be viewed as emergent phenomena, but they cannot be summarized into a consistent non-redundant system that unequivocally defines the structures of new utterances." ([13], p.145). Consequently, his memory- or analogy-based language model is not a typical example of unsupervised learning through redundancy reduction; we mention it here mainly because of the parallels between the data representation it employs (Stochastic Tree-Substitution Grammar [17]) and some of the formalisms discussed later, in section 4.

Split and merge pattern learning. The unsupervised structure learning algorithm developed by Wolff stands out in that it does not need the corpus to be tagged. In a 1988 book chapter describing his system [5], Wolff offers an excellent survey of earlier attempts at unsupervised learning of language, and of much relevant behavioral data. His representations consist of SYN (syntagmatic), PAR (paradigmatic) and M (terminal) elements. Although our patterns and equivalence classes can be seen as analogous to the first two of these, Wolff's learning criterion is much simpler than that of ADIOS: in each iteration, the most frequent pair of contiguous SYN elements are joined together. His system, however, has a unique provision for countering the usual propensity of unsupervised algorithms for over-generalization: PAR elements that do not admit free substitution among all their members in some context are rebuilt in a context-specific manner. Unfortunately, for implementational reasons Wolff's system has not been tested on unconstrained natural language.

4 Related linguistic approaches

Our work is data- rather than theory-driven in that we refrain from making *a priori* assumptions about the kind of "grammar" that we expect our algorithm to produce (cf. the quote from Langacker [11] in section 2). Clearly, however, the recursively structured, parameterized patterns learned by ADIOS, and their use in processing and generating novel sentences, do resemble certain features of some extensively studied syntactic formalisms. The similarities and differences between ADIOS and several such formalisms are discussed briefly in the remainder of this section. We distinguish between approaches that are motivated mainly by linguistic and psychological considerations (Cognitive and Construction grammars), and those motivated computationally (Local and Tree Adjoining grammars).

Cognitive Grammar. The main methodological tenets of ADIOS — populating the lexicon with "units" of varying complexity and degree of entrenchment, and using cognition-general mechanisms for learning and representation — are very much in the spirit of the foundations of Cognitive Grammar laid down by Langacker [11]. At the same time, whereas the cognitive grammarians typically attempt to hand-craft structures that would reflect the logic of language as they perceive it, ADIOS discovers the primitives of grammar empirically rather than accept them by fiat.

Construction Grammar. Similarities also exist between ADIOS and the various Construction Grammars [18, 19] (albeit the latter are all hand-crafted). A construction grammar consists of elements that differ in their complexity and in the degree to which they are specified: an idiom such as "big deal" is a fully specified, immutable construction, whereas the expression "the X, the Y" (as in "the more, the better"; cf. [20]) is a partially specified template. The patterns learned by ADIOS likewise vary along the dimensions of complexity and specificity (not every pattern has an equivalence class, for example). Moreover, we suspect that these patterns capture much of the semantics of the sentences from which they are abstracted, just as constructions are designed to serve as vehicles for expressing the conceptual/semantic content of intended messages in a form compatible with the structural constraints that apply to language. A proper evaluation of this claim must wait for the emergence of a semantic theory capable of dealing with all the complexities of natural language — something that current formal theories [21] cannot do. In the meanwhile, we concur with Jackendoff's position: "[...] we must explicitly deny that conceptual structures [...] *mean* anything. Rather, we want to say that they *are* meaning: they do exactly the things meaning is supposed to do, such as support inference and judgment." ([22], p.306).

Tree Adjoining Grammar. In capturing the regularities inherent in multiple criss-crossing paths through a corpus, ADIOS closely resembles the finite-state Local Grammar approach of Gross [23].[2] Note, however, that our pattern-based representations have counterparts for each of the two composition operations, substitution and adjoining, that characterize a Tree Adjoining Grammar, or TAG, developed by Joshi and others [25]. Specifically, both substitution and adjoining are subsumed in the relationships that hold among ADIOS patterns, such as the membership of one pattern in another (cf. section 2). Consider a pattern $\mathcal{P}_i$ and its equivalence class $\mathcal{E}(\mathcal{P}_i)$; any other pattern $\mathcal{P}_j \in \mathcal{E}(\mathcal{P}_i)$ can be seen as substitutable in $\mathcal{P}_i$. Likewise, if $\mathcal{P}_j \in \mathcal{E}(\mathcal{P}_i)$, $\mathcal{P}_k \in \mathcal{E}(\mathcal{P}_i)$ and $\mathcal{P}_k \in \mathcal{E}(\mathcal{P}_j)$, then the pattern $\mathcal{P}_j$ can be seen as adjoinable to $\mathcal{P}_i$. Because of this correspondence between the TAG operations and the ADIOS patterns, we believe that the latter represent regularities that are best described by Mildly Context-Sensitive Language formalism [25]. Moreover, because the ADIOS patterns are learned from data, they already incorporate the constraints on substitution and adjoining that in the original TAG framework must be specified manually.

5 Prospects and challenges

We have compared our approach to unsupervised learning of sequence structure (which is known to yield promising results when applied to raw corpora of language such as transcribed children-oriented speech [1]) to some recent work in computational linguistics and in grammar theory. The ADIOS approach to the representation of linguistic knowledge resembles the Construction Grammar in its general philosophy (e.g., in its reliance on structural generalizations rather than on syntax projected by the lexicon), and the Tree Adjoining Grammar in its computational capacity (e.g., in its apparent ability to accept Mildly Context Sensitive Languages). The representations learned by the ADIOS algorithm

[2] There are also interesting parallels here to the Variable Order Markov (VOM) models of symbolic sequence data [24].

are truly emergent from the (unannotated) corpus data, whereas those found in published works on cognitive and construction grammars and on TAGs are hand-tailored. Thus, our results complement and extend both the computational and the more linguistically oriented research into cognitive/construction grammar.

To further the cause of an integrated understanding of language, a crucial challenge must be met: a viable approach to the evaluation of performance of an unsupervised language learner must be developed, allowing testing both (1) neutral with respect to the linguistic dogma, and (2) cognizant of the plethora of phenomena documented by linguists over the course of the past half century.

Unsupervised grammar induction algorithms that work from raw data are in principle difficult to test, because any "gold standard" to which the acquired representation can be compared (such as the Penn Treebank [26]) invariably reflects its designers' preconceptions about language, which may not be valid, and which usually are controversial among linguists themselves [16]. As Wolff observes, a child "...must generalize from the sample to the language without overgeneralizing into the area of utterances which are not in the language. *What makes the problem tricky us that both kinds of generalization, by definition, have zero frequency in the child's experience.*" ([5], p.183, italics in the original). Instead of shifting the onus of explanation onto some unspecified evolutionary processes (which is what the innate grammar hypothesis amounts to), we suggest that a system such as ADIOS should be tested by monitoring its acceptance of massive amounts of human-generated data, and at the same time by getting human subjects to evaluate sentences generated by the system (note that this makes psycholinguistics a crucial component in the entire undertaking).

Such a purely empirical approach to the evaluation problem would waste the many valuable insights into the regularities of language accrued by the linguists over decades. Although some empiricists would consider this a fair price for quarantining what they perceive as a runaway theory that got out of touch with psychological and computational reality, we believe that searching for a middle way is a better idea, and that the middle way can be found, if the linguists can be persuaded to try and present their main findings in a theory-neutral manner. From recent reviews of syntax that do attempt to reach out to non-linguists (e.g., [27]), it appears that the core issues on which every designer of a language acquisition system should be focusing are dependencies (such as co-reference) and constraints on dependencies (such as island constraints), especially as seen in a typological (cross-linguistic) perspective [19].

Acknowledgment. Supported by the US-Israel Binational Science Foundation.

References

[1] Z. Solan, E. Ruppin, D. Horn, and S. Edelman. Automatic acquisition and efficient representation of syntactic structures. In S. Thrun, editor, *Advances in Neural Information Processing*, volume 15, Cambridge, MA, 2003. MIT Press.

[2] H. B. Barlow. Sensory mechanisms, the reduction of redundancy, and intelligence. In *The mechanisation of thought processes*, pages 535–539. H.M.S.O., London, 1959.

[3] H. B. Barlow. What is the computational goal of the neocortex? In C. Koch and J. L. Davis, editors, *Large-scale neuronal theories of the brain*, chapter 1, pages 1–22. MIT Press, Cambridge, MA, 1994.

[4] N. Redlich. Redundancy reduction as a strategy for unsupervised learning. *Neural Computation*, 5:289–304, 1993.

[5] J. G. Wolff. Learning syntax and meanings through optimization and distributional analysis. In Y. Levy, I. M. Schlesinger, and M. D. S. Braine, editors, *Categories and Processes in Language Acquisition*, pages 179–215. Lawrence Erlbaum, Hillsdale, NJ, 1988.

[6] Z. S. Harris. Distributional structure. *Word*, 10:140–162, 1954.

[7] M. van Zaanen. ABL: Alignment-Based Learning. In *COLING 2000 - Proceedings of the 18th International Conference on Computational Linguistics*, pages 961–967, 2000.

[8] B. MacWhinney and C. Snow. The Child Language Exchange System. *Journal of Computational Lingustics*, 12:271–296, 1985.

[9] F. Pereira. Formal grammar and information theory: Together again? *Philosophical Transactions of the Royal Society*, 358(1769):1239–1253, 2000.

[10] Z. Solan, E. Ruppin, D. Horn, and S. Edelman. Unsupervised efficient learning and representation of language structure. In R. Alterman and D. Kirsh, editors, *Proc. 25th Conference of the Cognitive Science Society*, Hillsdale, NJ, 2003. Erlbaum. in press.

[11] R. W. Langacker. *Foundations of cognitive grammar*, volume I: theoretical prerequisites. Stanford University Press, Stanford, CA, 1987.

[12] D. Klein and C. D. Manning. Natural language grammar induction using a constituent-context model. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[13] R. Bod. *Beyond grammar: an experience-based theory of language*. CSLI Publications, Stanford, US, 1998.

[14] A. Stolcke and S. Omohundro. Inducing probabilistic grammars by Bayesian model merging. In R. C. Carrasco and J. Oncina, editors, *Grammatical Inference and Applications*, pages 106–118. Springer, 1994.

[15] A. Clark. *Unsupervised Language Acquisition: Theory and Practice*. PhD thesis, COGS, University of Sussex, 2001.

[16] A. Clark. Unsupervised induction of Stochastic Context-Free Grammars using distributional clustering. In *Proceedings of CoNLL 2001*, Toulouse, 2001.

[17] R. Scha, R. Bod, and K. Sima'an. A memory-based model of syntactic analysis: data-oriented parsing. *J. of Experimental and Theoretical Artificial Intelligence*, 11:409–440, 1999.

[18] A. E. Goldberg. Constructions: a new theoretical approach to language. *Trends in Cognitive Sciences*, 7:219–224, 2003.

[19] W. Croft. *Radical Construction Grammar: syntactic theory in typological perspective*. Oxford University Press, Oxford, 2001.

[20] P. Kay and C. J. Fillmore. Grammatical constructions and linguistic generalizations: the What's X Doing Y? construction. *Language*, 75:1–33, 1999.

[21] P. M. Pietroski. The character of natural language semantics. In A. Barber, editor, *Epistemology of Language*. Oxford University Press, Oxford, UK, 2003. to appear.

[22] R. Jackendoff. *Foundations of language*. Oxford University Press, Oxford, 2002.

[23] M. Gross. The construction of local grammars. In E. Roche and Y. Schabès, editors, *Finite-State Language Processing*, pages 329–354. MIT Press, Cambridge, MA, 1997.

[24] M. Mächler and P. Bühlmann. Variable Length Markov Chains: Methodology, computing and software. Seminar for Statistics Report 104, ETH Zürich, 2002.

[25] A. Joshi and Y. Schabes. Tree-Adjoining Grammars. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3, pages 69 – 124. Springer, Berlin, 1997.

[26] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1994.

[27] C. Phillips. Syntax. In L. Nadel, editor, *Encyclopedia of Cognitive Science*, volume 4, pages 319–329. Macmillan, London, 2003.