Contrasting Applications of Logic in Natural Language Syntactic Description*

Geoffrey K. Pullum
University of California, Santa Cruz
and
Barbara C. Scholz
San José State University

Abstract. Formal syntax has hitherto worked mostly with theoretical frameworks that take grammars to be generative, in Emil Post's sense: they provide recursive enumerations of sets. This work has its origins in Post's formalization of proof theory. There is an alternative, with roots in the semantic side of logic: model-theoretic syntax (MTS). MTS takes grammars to be sets of statements of which (algebraically idealized) well-formed expressions are models. We clarify the difference between the two kinds of framework and review their separate histories, and then argue that the generative perspective has misled linguists concerning the properties of natural languages. We select two elementary facts about natural language phenomena for discussion: the gradient character of the property of being ungrammatical and the open nature of natural language lexicons. We claim that the MTS perspective on syntactic structure does much better on representing the facts in these two domains. We also examine the arguments linguists give for the infinitude of the class of all expressions in a natural language. These arguments turn out on examination to be either unsound or lacking in empirical content. We claim that infinitude is an unsupportable claim that is also unimportant. What is actually needed is a way of representing the structure of expressions in a natural language without assigning any importance to the notion of a unique set with definite cardinality that contains all and only the expressions in the language. MTS provides that.

Introduction

For the last half century a large community of linguists has been devoted to the goal of stating grammars for natural languages, and general linguistic theories, in the form of fully explicit theories. Attainment of this goal means making fully explicit all of the consequences of grammatical statements and theoretical principles. We argue here, specifically with respect to syntax, that linguists have been led astray as a result of taking too narrow a view of what it means to construct an explicit grammar or theory. Linguists have largely restricted themselves to a class

*This paper presents the main content of the invited lecture given by the first author at the 12th International Congress of Logic, Methodology and Philosophy of Science (Oviedo, Spain, 2003). It was based on joint research and is here written up jointly. We are grateful to many people for stimulating questions and comments on these ideas, not only at the International Congress but also at Stanford University; the University of California, Santa Cruz; Simon Fraser University; the Technical University of Vienna; the University of Pennsylvania; and Harvard University. Thanks to Peter Alrenga, John Colby, Robert Conde, Ascander Dost, Paul Postal, Chris Potts, James Rogers, and Dag Westerstahl for helpful comments; they enabled us to avoid some errors, and bear no blame for the ones that remain.

of theoretical frameworks for syntax, what we call **generative** frameworks, which are in fact ill suited to stating theories and grammars for natural languages (though ironically, they are ideal for stating explicit definitions of the denumerable infinite sets of formulae referenced in fields like logic). In consequence, highly distinctive features of natural languages, features crucially differentiating them from invented formal languages, have been overlooked.

There is an alternative type of framework for stating explicit grammars of natural languages. We use the term **model-theoretic syntax** to refer generically to frameworks of this type. Model-theoretic syntax has great heuristic value for linguistics, in the sense of providing appropriate guidance of theoreticians' thinking with respect to just those aspects of natural languages that generative frameworks have ignored. Among these are two that we discuss here: the gradience of ungrammaticality and the lexical openness of natural languages.

We also argue that the machinery and structure of generative frameworks have misled linguists, philosophers, and psycholinguists alike into thinking that the cardinality of the set of expressions generated by a generative grammar is important theoretically, in a way that is connected with settling the question of whether natural languages are infinite. We argue that this is a particularly clear case of scientists having confused their subject matter with their theoretical toolkit, and suggest that the model-theoretic perspective permits an escape from that confusion.

Origins of generative frameworks

In his invited address at the first International Congress of Logic, Methodology and Philosophy of Science in 1960, Noam Chomsky laid out a convincing case for the goal of stating explicit grammars. He then turned to the development of a framework for stating them, remarking:

> Clearly, a grammar must contain two basic elements: a 'syntactic component' that generates an infinite number of strings representing grammatical sentences and a 'morphophonemic component' that specifies the physical shape of each of these sentences. This is the classical model for grammar. (Chomsky 1962:539)

But such a model is not "classical" in any sense. The idea that grammars of natural languages generate infinite sets of strings was only a few years old in 1960, and the mathematics supporting it only a few decades older.

The most important source for the approach Chomsky advocated was the work of Emil Post in the period 1920–1950. Post's goal was a mathematicization of the notion of a proof in logic. In pursuit of this he developed a schema defining the general form of inference rules in terms of what he calls **productions** (see Post 1943:197). A production has a finite number of 'premises' and produces one 'conclusion'. Each of these is a finite string of symbols. A production thus maps a set of strings to a string, has the form seen in (1a), where '$\Rightarrow$' represents the relation of producing or licensing, and each σ_i is a string of the form in (1b).[2]

(1) a. $\{\sigma_1, \cdots, \sigma_{n-1}\} \quad \Rightarrow \quad \sigma_n$

b. $x_0\, X_1\, x_1\, X_2\, x_2\, \cdots\, X_m\, x_m$

[2]Kozen (1997:256-7) provides a brief but useful elementary overview of Post production systems, which we have drawn on here.

Each x_i in (1b) is a specified symbol string (possibly null), and each X_i is a free variable over (possibly null) symbol strings. There are two associated stipulations. First, every X_i in the conclusion σ_n must appear in at least one of the premises $\{\sigma_1, \cdots, \sigma_{n-1}\}$ (intuitively because inference rules never allow arbitrary extra material to be introduced into a conclusion). Second, no assignment of strings to variables is permitted if it would lead to σ_n being the empty string e (because conclusions have to say something).

Proofs are formally represented as **derivations**. A derivation is a sequence of symbol strings in which each string in the sequence either belongs to a fixed set of strings given at the start (i.e., it is an axiom) or is derivable by means of a production from some set of the strings that precede it in the derivation (i.e., it is licensed by a rule of inference). A **production system** is a set of axioms together with a set of productions, and it **generates** the set of all and only those strings that appear as the last line of some derivation.

Post showed that production systems could generate any recursively enumerable (r. e.) set of strings, thus connecting provability to the characterization of sets of strings by Turing machines. Chomsky (1962) refers to this work just half a dozen lines below the quotation given earlier, and comments:

> A rewriting rule is a special case of a production in the sense of Post; a rule of the form ZXW → ZYW, where Z or W (or both) may be null. (Chomsky 1962:539)

The idea of defining restricted special cases of production systems without losing expressive power was already present in Post's work. Post (1943) had defined a special case of production systems by imposing the following limitations:

- n in (1a) is limited to 2 (i.e., only one premise is allowed), so productions have the form $\sigma_1 \Rightarrow \sigma_2$;
- m in (1b) is limited to 1 (i.e., only one variable is allowed), so σ_1 and σ_2 have the form $x_0X_1x_1$;
- x_1 in σ_1 and x_0 in σ_2 are both null.

This means that each production has the form '$xX \Rightarrow Xy$' for some x and y. Post shows that this makes no difference to expressive power, provided only that the vocabulary can contain certain symbols that function in productions but do not appear in the strings derived; today these are called **nonterminals**, and the symbols appearing in generated strings are called **terminals**. The definition of 'generates' is altered slightly when nonterminals are introduced: the generated set is taken to be the set of all and only those *strings of terminals* that appear as the last line of some derivation. Post proves that any r. e. stringset over the terminal vocabulary can be generated by a production system in the restricted form. And he shows that this remains true if there is just a single axiom consisting of one symbol.

The Type 0 rules of Chomsky (1959)) are defined by a restriction very similar in spirit. Type 0 grammars are essentially Post's semi-Thue systems, deriving from earlier work by Axel Thue (see Post 1947). Again, n in (1a) is limited to 2 so that productions have the form $\sigma_1 \Rightarrow \sigma_2$, but:

– m in (1b) is limited to 2 (i.e., there are just two variables in each production), so σ_1 and σ_2 both have the form $x_0X_1x_1X_2x_2$; and
– in both the premise and the conclusion, x_0 and x_2 are both null.

In other words, each production has the form '$X_1xX_2 \Rightarrow X_1yX_2$' for some x and y: it rewrites a string x as a different string y in contexts where X_1 precedes and X_2 follows. Again there is no loss of expressive power from Post's unrestricted production systems over vocabularies of terminal and nonterminal symbols (this is proved in Chomsky 1959): all (and only) the r. e. sets over the terminal vocabulary can be generated.

The idea of cutting productions completely adrift from the formalization of inference calculi, and using them instead to enumerate strings of words corresponding to natural language sentences, did not come from any classical or traditional source; it was original with Chomsky, who had read Post and acknowledges the intellectual debt (Post 1944 was Chomsky's cited source for the term 'generate'; see Chomsky 1959:137n).

Today, virtually all theoretical linguistics that aims at explicitness is based on Chomsky's development of Post. Chomsky's early transformational grammars represented an elaboration rather than a further restriction of Post systems (though expressive power is not increased). The context-sensitive, context-free, and regular grammars defined in Chomsky (1959) are special cases that have reduced expressive power so that not all r. e. sets are generable.

Categorial grammars (the origins of which antedate not only Chomsky's work but also Post's) can be seen as a kind of bottom-up special case of Post systems where instead of starting with the single-symbol axiom S and iteratively rewriting it until a string over the terminal vocabulary Σ is obtained, we start with a multiset of pairs $\langle\sigma_i, \alpha\rangle$, where σ is a one-symbol string over Σ and α is a nonterminal, and we form larger strings by iterative combination under general principles (e.g., given $\langle\sigma_1, A/B\rangle$ and $\langle\sigma_2, B\rangle$ we are allowed to form $\langle\sigma_1\sigma_2, A\rangle$); the combination process proceeds until we obtain a pair $\langle\sigma_1 \ldots \sigma_k, S\rangle$ (S being the designated category of complete or saturated expressions), which corresponds in a standard Post system to a proof that the string $\sigma_1 \ldots \sigma_k$ can be derived from S.

Under some formalizations which allow function composition as well as function application, categorial grammars are equivalent to tree adjoining grammars (Weir and Vijayshanker 1994); some simpler forms are equivalent to context-free grammars. The so-called 'minimalist' grammars of Chomsky's recent transformationalist work appear to be very similar to categorial systems in this regard, and if Edward Stabler's formalization of them is accepted (Stabler 1997) they are equivalent to multi-component tree-adjoining grammars (Harkema 2001; Michaelis 2001).

Mathematical sources of model-theoretic syntax

There is an entirely distinct alternative way to state explicit grammars. With hindsight we can glimpse its mathematical beginnings in the work of Büchi (1960).[3] Büchi's results were motivated by questions of arithmetic. It took more than thirty years for their relevance to linguistics to be appreciated (e.g., by Rogers 1994,

[3]The recent literature contains some more elegant and accessible proofs of Büchi's theorem: see especially Thomas (1990), Engelfriet (1993), and Straubing (1994).

Rogers 1998, and Kracht 2001). The basic question concerned the expressive power of logic of a particular kind for talking about finite sequences. The logic in question was ordinary predicate logic augmented in such a way that in addition to variables ranging over individuals it has variables ranging over finite sets of individuals. It is known as **weak monadic second-order logic** (henceforth WMSOL). One very desirable property of WMSOL is that its satisfiability problem is decidable — there is an algorithm for determining whether the set of structures satisfying a WMSOL formula is empty.

When WMSOL is interpreted on finite linearly ordered structures, which we will call **string structures**, Büchi showed that the following holds:

(2) *Büchi's theorem*
Given any existential WMSOL formula, the set of all its string-structure models is a regular stringset, and for any regular stringset there is a WMSOL formula having that stringset as the set of all its string-structure models.

This gives us a purely model-theoretic perspective on the regular stringsets (generated by Chomsky's 'Type 3' grammars): a regular stringset is simply a set containing all and only those finite string structures that are models of a certain existential formula of WMSOL.

It took a number of years for Büchi's work (and the similar contemporaneous work of Calvin Elgot (1961)) to be appreciated, but in due course the result was generalized to structures having the form of labeled trees. Doner (1970) proved an analog of Büchi's theorem for finite, ordered, directed, acyclic, singly-rooted, non-tangled graphs with labeled nodes as used by linguists for representing syntactic structure that was of special interest in automata theory. In Rogers (1998) these results are finally applied to linguistics rather than computation. Rogers defines a WMSOL description language in which in addition to the binary relation symbol '$\prec$', interpreted by the 'left of' relation found in string structures, there is also a second binary relation symbol '$\triangleleft^*$', interpreted by the dominance relation. The usual exclusivity and exhaustiveness condition for dominance and precedence can be stated thus:

(3) $\forall x \forall y[(x \triangleleft^* y \vee y \triangleleft^* x) \leftrightarrow \neg(x \prec y \vee y \prec x)]$
A pair of nodes stands in the dominance relation or its inverse if and only if it does not stand in the precedence relation or its inverse.

The condition that says every tree has a node (the root) that is the minimum point in the weak partial dominance ordering can be stated thus:

(4) $\exists x \forall y[x \triangleleft^* y]$
There is a node that dominates every node.

The other basic axioms defining trees can be given in a similar way (for details see Rogers 1998:15-16).

Structural generalizations about trees with particular properties can also be defined (and most of the time we can do it with just the first-order fragment of our WMSOL language). For example, we can say of a tree that it has only binary branching. Let '$\lhd$' be a symbol interpreted by the immediate dominance or 'parent of' relation.[4] Then a tree is binary-branching iff it satisfies (5).

(5) $\forall x[\exists y \exists z[x \lhd y \wedge x \lhd z \wedge y \neq z] \vee \neg\exists y[x \lhd y]]$
Every node is the parent either of two distinct nodes or of none.

But WMSOL also allows quantification over finite sets of nodes, which enables further properties and relations on trees to be defined. Quantification over sets of nodes is the key to the full power of WMSOL on tree models that Doner (1970) exploited. It permits the expression of certain projections between label sets that enable the logic to define the property of being recognizable by a finite-state tree automaton, distinguishing it from the (more restrictive) property of being generable by a context-free grammar. The difference is, in effect, that a recognizable treeset may have certain dependencies between nodes in the trees that are not registered in the node labeling.[5] What Doner's theorem says is this:

(6) *Doner's theorem*
Given any existential WMSOL formula, the set of all its finite tree models is recognizable by a finite-state tree automaton, and thus the yield of that set of trees is a context-free stringset; and for any context-free stringset there is a WMSOL formula having that stringset as the yield of the (recognizable) set containing all and only its finite tree models.

The decidability result holds as before: WMSOL on binary trees corresponds to the WMSOL theory of arithmetic with two successor functions, which was shown by Rabin (1969) to be decidable.

We now have a characterization of both the context-free stringsets and the recognizable treesets in terms of a logic with a decidable satisfiability problem. What is important about this is that, as shown by the work of Gazdar et al. (1985), henceforth *GKPS*, a very large part of the central facts of syntax for English can be stated in terms of recognizable sets of trees. And Rogers (1997) shows that the kinds of theoretical statements made in GKPS can be restated much more simply using WMSOL to impose conditions on syntactic structure directly. For example, Rogers shows how to express the theory of feature specification defaults in a way that is vastly simpler than the cumbersome development of GKPS. A predicate P'_f of sets of nodes is defined for a feature specification f to characterize the property of (i) including all nodes that are free to take f without violating other statements of the theory, and (ii) being closed under propagation of f (Rogers 1997:739). Then the notion of being privileged with respect to feature f can be defined by

$$\text{Privileged}_f(x) \equiv \forall X[P'_f(X) \rightarrow X(x)]$$

[4]That is, let $x \lhd y$ mean by definition $x \lhd^* y \wedge x \neq y \wedge \neg\exists z[x \lhd^* z \wedge z \lhd^* y \wedge x \neq z \wedge z \neq y]$.

[5]Doner gives a useful example to distinguish the two: the set of all binary-branching trees in which every node is labeled A except for a unique node labeled B. Generating this set with a context-free grammar is impossible, yet it is easily recognizable by a finite-state tree automaton that keeps track in its state space of not just how many daughters the current node has and what the daughter labels are but also whether the unique B is contained within the subtree dominated by the current node.

(Rogers 1997:740). Feature specification defaults become easily statable as simple material conditionals about trees without any special non-classical default semantics; for example,

$$\forall x[\neg \text{Privileged}_{[-\text{INV}]} \rightarrow [-\text{INV}](x)]$$

says that if a node is not privileged with respect to [–INV], then it is [–INV].

Work on other natural languages shows that they too are mostly describable in terms of recognizable sets of trees, hence by WMSOL on trees. Where natural languages do have constructions that go beyond the sets of trees (henceforth, treesets) that are recognizable in the standard sense, the expression types involved are often grammatically rather marginal. For example, English has an idiomatic adjunct type illustrated by the bracketed part of *The US will go ahead,* [*UN support or no UN support*]; the adjunct must be of the form '... *W or no W*'. But the construction is not that common, the values of *W* are typically just individual nouns or noun-noun compounds, and the syntax of the construction does not interact with central properties of clause structure at all.

There are a few central clausal constructions in other languages that go beyond the recognition power of standard tree automata, but they have so far been found only in Germanic (Shieber 1985; Miller 1991), and they seem quite rare. Work in computational linguistics has shown that if we have the power to describe the recognizable treesets, we have enough descriptive power for doing most of the work necessary in natural language processing.

Insofar as describing non-context-free constructions model-theoretically is required, Rogers (2003) shows how to do it for a very interesting class of cases. He shows that the tree adjoining treesets (and derivatively, the tree-adjoining stringsets that are their yields) can be characterized in terms of WMSOL on three-dimensional tree manifolds — roughly, trees that have trees as their nodes; and Langholm (2001) has shown that the much larger class of indexed stringsets (Aho 1968) can also be characterized with a particular kind of bounded existential quantification in WMSOL on trees with added links between the nodes.

All the work just reviewed uses model theory to characterize sets of structures. The possibility this opens for linguistics is that if we idealize expressions of natural languages as structures such as trees or similar graphs, we can formulate grammars for natural languages as sets of interpreted statements in a logic, the models being expression structures. These grammars will be fully explicit, though not generative.

Linguistic foundations of model-theoretic frameworks

There is a sense in which McCawley (1968) might be said to have introduced the model-theoretic view to the linguistics literature, though in various ways his approach does not exactly coincide with the one we will adhere to below. McCawley pointed out that phrase structure rules can be interpreted as statements that are satisfied (or violated) by trees.[6] He noted that for context-sensitive rules there is an expressive power difference between what we could now call their generative and model-theoretic interpretations.

[6]This insight of McCawley's does not appear to have been influenced by the work of Büchi, and predates that of Doner. Note that McCawley (1968) does not explicitly make the link to using logic as a descriptive formalism for trees.

The generative interpretation of a rule of the form '$A_0 \rightarrow A_1 \ldots A_n / X\_\_Y$' is that it means "in a string containing the substring XA_0Y, the A_0 may be replaced by $A_1 \ldots A_n$." (A_0 is a nonterminal, $A_1, \ldots, A_n$ are terminals or nonterminals, and X, Y are strings of terminals or nonterminals.) Under this interpretation, grammars with context-sensitive rules generate all and only the context-sensitive stringsets over the terminal vocabulary (i.e., all and only the stringsets that are Turing-recognizable in linear space — a very large proper superset of the context-free stringsets).

McCawley's alternate interpretation for phrase structure rules, stated in the same formalism, was that they should be interpreted as stating sufficient conditions for well-formedness of local subtrees. The rule '$A \rightarrow A_1 \ldots A_n / W\_\_Y$' says that a subtree T with root A_0 and daughters $A_1 \ldots A_n$ is legitimate provided that the sequence W can be found immediately left-adjacent to T in the rest of the tree, and the sequence Y can be found immediately right-adjacent to T. A string is in the stringset defined by a grammar under this interpretation iff it is the yield of a tree in which every subtree is legitimate according to the rule set. This interpretation, surprisingly, is much more restrictive. It was shown by Peters and Ritchie (1969, 1973) that the set characterized by context-sensitive rules under the tree-admitting interpretation is the set of context-free stringsets.

A wider application of the same kind of thinking about how to state grammars can be seen in (Lakoff 1971). Lakoff proposes that the structure of a natural language expression should be idealized as a finite sequence of finite trees $\langle \Delta_0, \ldots, \Delta_k \rangle$ in which Δ_0 is by definition the deep structure of the expression (under Lakoff's generative semantics view, the part that determines the semantic interpretation), and has a form determined by conditions on those structures, and Δ_k is the surface structure, determining the phonological interpretation. For each $i \geq 1$, Δ_i results from the application of a transformation to Δ_{i-1}; the value of k will vary, since some expressions have more complex transformational 'derivations' than others.

Crucially, Lakoff takes grammars to be sets of assertions about the structural properties of tree sequences. The analog of a transformation in Lakoff's scheme is a condition specifying the permitted differences between two adjacent trees Δ_{i-1} and Δ_i ($0 < i < k$) in a sequence. Rule-ordering conditions and global derivational constraints are claimed to be formalizable as higher-level conditions on the form of tree sequences.[7]

Lakoff did not in fact present a well-defined theory of grammar. He left many crucial matters undefined. The exact character of the models was not clarified, and no specification of the logical description language or its interpretation was given (for example, the domain of quantification was never quite clear: sometimes he seemed to be quantifying over nodes and sometimes trees). The detailed critique offered by Soames (1974) established that Lakoff's proposals did not work as stated: Soames shows convincingly that Lakoff underestimated the difficulty of making his

[7] Postal (1972) regards it as a fundamental clarification that under this view there is no important distinction between transformations, which filter out illicit $\langle \Delta_{i-1}, \Delta_i \rangle$ pairs from sequences, and 'global' constraints that filter out illicit tree-pairs $\langle \Delta_i, \Delta_j \rangle$ for arbitrary i and j, since i and j will simply reference positions between 0 and k in a finite sequence, and whether $j - i = 1$ can hardly matter very much.

proposal explicit.[8] Nonetheless, the leading idea that syntactic descriptions can be stated as *sets of well-formedness conditions on syntactic structures*, rather than procedures for generating sets, is clearly present in Lakoff's paper.

Other antecedents of model-theoretic syntax were present in the work of a group of Russian mathematicians in Moscow at around the same time. Borščev and Xomjakov (1973) take 'languages' to be collections of what they call 'texts', and idealize texts as finite models in a suitable signature, formulating grammars as sets of axioms, a grammar being interpreted as a description of those texts that satisfy it. Borščev (personal communication, May 2002) informs us that the group did intend to apply this approach to natural language description, but the applied work that got done mostly related it to the description of chemical structures.

The model-theoretic approach foreshadowed in these early works was much more fully developed by Johnson and Postal (1980), who overtly adopted the idea of treating expression structures as models of statements in a logical description language. Their structures are complex graph-like objects (actually, graphs with additional relations holding between edges), some aspects of which e.g., the linear ordering of terminals that defines the yield) are not clearly worked out, and their metalanguage is not fully defined (it may be first-order, or it may be augmented with the power to define ancestrals of relations, one cannot tell from the exposition); but without question, the leading idea of model-theoretic syntax is there, along with a clear perception of some of its metatheoretical advantages.

Virtually no linguists followed Johnson and Postal's lead in the decade that followed. Gerald Gazdar proposed a model-theoretic reconstruction of the generalized phrase structure grammar of Gazdar et al. (1985) in various lectures in 1987, but otherwise the approach was forgotten until the 1990s, when papers independently developing the idea in two directions began to appear. First, Patrick Blackburn and his colleagues, and independently Marcus Kracht, began to develop an idea from Gazdar et al. (1988): using modal logic as a description language for syntax (Blackburn et al. 1993; Kracht 1993; Blackburn and Gardent 1995; Blackburn and Meyer-Viol 1997). And second, James Rogers began his work on applying WMSOL to linguistics (Rogers 1994, 1996, 1997, 1998, 1999).

These papers from the 1990s, however, were largely preoccupied with restating particular varieties of generative grammars in model-theoretic terms, the point being to attain new insights into the character of the syntactic facts or to compare the expressive power of different classes of grammars. If model-theoretic syntax were merely a matter of stating generative grammars in a different way, it would be of only minor importance to linguistics. Our thesis is that model-theoretic syntax offers the study of natural languages not just a restatement of generative grammar but a shift of framework type with profound heuristic consequences.

[8]The technical difficulties could probably be overcome. It would help to link the trees in the sequence via a correspondence relation holding between the nodes of one tree in the sequence and the nodes of the next. This would permit movement or erasure of a particular node to be reconstructed in terms of where or whether a node in one tree corresponded to a node in the next. See Potts and Pullum (2002) for an application of this idea (on pairs rather than arbitrary n-tuples of tree-like structures) in phonological theory.

Two key properties of natural languages

The invented languages of logic, mathematics, and computer science — henceforth, **formal languages** — are stipulated sets of strings (or other structures) defined over a finite vocabulary of symbols. Post production systems and the generative grammars that are based on them are ideally suited to stating the explicit grammars of formal languages, and were invented for exactly that purpose.

But natural languages have a number of properties that clearly differentiate them from formal languages. In this section we review two illustrative phenomena: first, the fact that being grammatically ill-formed is a matter of degree, and second, the fact that there is no fixed lexicon for a natural language. We point out that model-theoretic frameworks immediately suggest appropriate ways to describe the relevant phenomena. Generative frameworks do not.

We are not saying that it would be impossible to use a generative grammar in giving an explicit account of these phenomena. Augmentation with additional theoretical machinery is always possible. But it does appear that any such theoretical augmentations will be entirely ad hoc. The basic structure of generative grammars does not suggest them. Certain distinctive phenomena of natural language appear to have been ignored within generative grammar precisely because generative frameworks are ill suited to their description.

Ungrammaticality is gradient

Some utterances that do not correspond to fully well-formed expressions are vastly less deviant than others. And this feature of utterances is also a feature of expressions — or rather (since it may be better to limit the term 'expression' to what is fully grammatical), those objects that are like expressions except that they are only partially well-formed. Let us call these latter QUASI-EXPRESSIONS. For example, (7a) is a quasi-expression that is clearly ungrammatical; and (7b) is clearly more ungrammatical; but neither is ungrammatical to the same degree as the utterly incomprehensible (7c).

(7) a. * *The growth of of spam threatens to make email useless.*
b. * *The growth of of the spam threatens make email useless.*
c. * *The of email growth make threatens spam to useless of.*

The point is that *some quasi-expressions are closer to being grammatical than others.*

No unaugmented generative grammar describes degrees of ungrammaticality, nor does it suggest a way to do so. A generative grammar generates a single set L of expressions over a vocabulary V. L is a subset of V^* (the set of all strings over V). Given $w \in L - V^*$ and a generative grammar G such that $L(G) = L$, G will say absolutely nothing about w, since no derivation permitted by G will lead to w.

Chomsky (1955), Chomsky (1961), and Chomsky and Miller (1963) give several basically unsuccessful attempts to augment a generative grammar so that degrees of ungrammaticality can be described. All these proposals define degrees of ungrammaticality by matching ungrammatical word sequences with lexical category strings associated with grammatical word sequences. The idea is to define a series of lexical category inventories of graded coarseness, the finest being the set of lexical categories assigned to words in a full and accurate grammar for the language, and the coarsest consisting of just the single category 'Word'. A string w that is not generated by the grammar is assigned a degree of ungrammaticality according to which degree of lexical category coarseness must be used to get a match between w and some string that is generated. Consider these examples:

(8) a. *John plays golf.* ($\mathrm{N_{anim}}$ $\mathrm{V_t}$[+__anim. subj.] $\mathrm{N_{inan}}$)

b. *Golf plays John.* ($\mathrm{N_{inan}}$ $\mathrm{V_t}$[+__anim. subj.] $\mathrm{N_{anim}}$)

c. * *Golf fainted John.* ($\mathrm{N_{inan}}$ $\mathrm{V_i}$ $\mathrm{N_{anim}}$)

d. * *The of punctilious.* (D P Adj)

The example in (8a) is entirely normal; (8b) is not, but if we just ignore the requirement that *play* should have an animate subject we can say that it matches (8a) in its sequence of lexical categories;[9] (8c) is worse, because to find a match for it we have to descend to a greater level of coarseness of categorization where we ignore not only the selectional restriction that *faint* should have an animate subject but also that *faint* is syntactically required not to have a direct object (a strict subcategorization restriction); and finally (8d) is yet worse, because its sequence of lexical categories matches nothing in English unless we categorize its items at a level of coarseness where we treat it as simply 'Word Word Word'.

Proposals of this type are inadequate for describing the phenomena of ungrammaticality in natural languages. They provide too few degrees of ungrammaticality. The examples provided in the cited references yield only three levels, and it is not clear how to go beyond this. A serious problem is that the accounting system for grammatical errors is not cumulative; for example, all of the strings in (7) will be assigned exactly the same degree of ungrammaticality, clearly the wrong result.

But perhaps the most important inadequacies stem from deep theoretical failings in this kind of analysis. First, the proposal relies on an entirely nonconstructive definition of a transderivational relation between strings in various infinite sets. To determine whether the relevant relation holds between a a specific ungrammatical w and a certain string of lexical categories, we have to solve this problem:

(9) Input: an ungrammatical string $w_1 \dots w_n$ of words categorized at coarseness level i yielding a category string $K_1 \dots K_n$.

Output: a decision on whether there is a grammatical sentence that also has lexical category sequence $K_1 \dots K_n$ at coarseness level i.

But this is undecidable for Post production systems in general, and for the transformational grammars that Chomsky developed from them. All r. e. sets have transformational grammars; but r. e. sets do not necessarily have r. e. complements. This means that although there will always be some answer to a question of the form in (9), there can be no general effective procedure for finding it.

Even more importantly, the generative grammar that derives the well-formed strings is independent of the assignment of degrees of ungrammaticality. Degree of ungrammaticality depends entirely on lexical category sequence similarities, according to this proposal; it does not depend on any aspect of syntactic structure.

[9] Incidentally, we would follow McCawley 1968 in taking the deviance of selectional restriction violations like *Golf plays John* as semantic rather than syntactic.

Indeed, the independence of the grammar from the assignment of degrees of ungrammaticality is so extreme that it would be the same given *any* observationally adequate grammar with the same lexicon.

By contrast, with model-theoretic grammars the same resources employed to describe the fully grammatical expressions also yield a description of the quasi-expressions. Let Ω be a domain of relevant structures of some kind and Γ a set of constraints interpreted on structures in Ω. Let Ω_1 be the subset of Ω containing the structures that satisfy Γ. We note that Γ also structures the rest of the domain, assigning a status to each member of $\Omega - \Omega_1$, depending on which of the constraints in Γ that member fails to satisfy. And notice, the status assigned to a structure depends entirely on what structural properties it has and what the grammatical constraints in Γ say.

That is, model-theoretic grammars do not just partition a set of structures into two subsets (the fully well-formed set and the complement of that set). Rather, even under a crude view on which structures are evaluated as wholes,[10] a set of n constraints determines 2^n subsets, just as any set of n binary attributes determines a set of 2^n distinct attribute-value matrices.

Which quasi-expressions are assigned which degrees of ungrammaticality will not be invariant under reaxiomatization. What the right constraints are for the grammar of some language has to be decided on the basis of a wide range of empirical evidence and theoretical considerations — simply selecting a framework does not settle everything. For example, if we replace a set of two or more constraints by a single constraint which is their conjunction, we get only one degree of ungrammaticality, namely 100% ungrammaticality. So the model-theoretic account can always mimic the undesirable properties of the generative account.[11]

Under the model-theoretic view we are suggesting for consideration, therefore, a whole new range of empirical data, the class of facts about one quasi-expression being more ungrammatical than another, becomes relevant to decisions about what the constraints are, and with exactly what degree of delicacy they should be formulated, and in what description language.

The lexicon is open

Take the structure for some grammatical expression in a natural language and replace one of its lexical items by a piece of phonological material unassociated with any grammatical or semantic properties (henceforth we call such a nonsense form a *pseudo-word*). For example, take a well-formed structure for (10a) and replace *fox* by the random pseudo-word *meglip*, changing nothing else in the structure, yielding a similar structure for (10b).

(10) a. *The quick brown fox jumps over the lazy dog*

b. *The quick brown meglip jumps over the lazy dog*

[10] For a more refined view of ungrammaticality we could ask for each *node* in the structure which of the constraints is satisfied at that node; obviously, the normal interpretation strategy for modal logic would be ideal for this.

[11] Johnson and Postal (1980) actually do adopt the view that a grammar is to be stated as a single formula, the conjunction of the large set of material conditionals that state their constraints on syntactic structure, though without noticing the undesirable consequence we point out here.

What is the grammatical status of the new structure, the one with terminal string (10b)? Does it have grammatical structure?

An unaugmented generative grammar (either top-down or bottom-up), by definition, fixes a finite inventory of admissible terminal symbols. The lexicon for the grammar contains these symbols paired with certain grammatical and semantic features. In top-down generative grammars only a terminal symbol (lexical item) that appears on the right hand side of a rule can occur in a derived string, and all nonterminals must ultimately be eliminated through the application of rules if the derivation is to complete. Since non-terminating sequences are not derivations at all (by definition), no top-down generative grammar, in and of itself, allows any structure containing a pseudo-word to be in the set of grammatical sentences generated.

The failure of bottom-up generative grammars to derive the structure of expressions with pseudo-word terminals is particularly clear. Derivations start with a multiset of items (a 'numeration' in minimalist parlance) selected from some fixed, finite lexical stock, and operations of combination are performed on them. The generated language is the set of all structures that can be built using some multiset. Keenan and Stabler (1996), for example, define a bottom-up generative framework in which a grammar is a finite set Lex of lexical items together with a finite set $\mathcal{F}$ of combination operations defined on Lex. The language generated is the set of all expressions that can be derived by selecting some multiset M of items from Lex and iteratively applying operations in $\mathcal{F}$ until M is exhausted and a saturated expression (e.g., a sentence) has been derived. A structure containing an item not present in Lex cannot be derived.

Thus (10b) has no derivation in such grammars, and is not even classified by such grammars as a candidate for grammaticality in English, because it does not belong to the universe of strings over the appropriate set of symbols.

A distinctive and fundamental feature of natural languages is being missed by generative frameworks: the lexicons of natural languages are *not* fixed sets. Natural languages are strikingly different from formal languages in this respect. In a natural language the lexicon continuously changes, not just on an intergenerational time scale (in the sense that children do not learn exactly the same words as those their parents learned), but week by week and day by day. New brand and model names are introduced; novel personal names are given; technical terms are devised; new artifacts are dubbed; onomatopoeic terms are made up on the fly; words are borrowed from other languages; noises are imitated and used as words; and in dozens of other ways the word stock of a natural language is constantly under modification. The lexicon of a natural language is open and indefinitely extensible. In consequence, explicit grammars for natural languages, if they are to describe the phenomena, must not entail that the lexicon is some fixed, finite set.

We are not concerned here with whatever diachronic processes add new lexical items to languages during their history, nor with the psycholinguistic processes involved in the recognition of expressions containing pseudo-words or the coining or production of nonce words. Rather, we are concerned with a fundamental feature of natural languages and with how explicit grammars can be formulated in a way that is compatible with it.

A model-theoretic syntactic framework makes available a fully explicit description of lexical openness.[12] In model-theoretic terms, to say that a language has a lexical item with a certain phonological form and certain grammatical and semantic properties, is to say simply that there are constraints on that phonological form stating limits on the grammatical and semantic properties that are associated with it. That is, what it means for there to be a noun *fox* in English is that a condition in the grammar of English links the phonological representation /faks/ to the lexical category N and the property of having regular inflection and the meaning "member of the species *Vulpes vulpes crucigera*."

A pseudo-word, by definition, is phonologically well-formed but has no grammatical or semantic properties. A correct grammar will thus state no conditions on grammatical or semantic aspects of the structures in which it may occur. Thus there are no grammatical or semantic constraints in any model-theoretic grammar for a pseudo-word to violate. No expression structure will be ungrammatical solely in virtue of containing a pseudo-word at some terminal node. If the structure of (10b) is the same as that of (10a), then (10b) does not violate any constraint at all.

Introductory works on language and linguistics commonly observe that expressions containing pseudo-words are well formed. Many (see Pinker 1994:89 for a typical example) reprint the first stanza of Lewis Carroll's *Jabberwocky* ("'Twas brillig, and the slithy toves / Did gyre and gimble in the wabe; / All mimsy were the borogoves ...") to make exactly this point — the point that syntactic structure is not entirely dependent on lexical inventory. The authors correctly regard the clause *all mimsy were the borogoves* as grammatical English. What they appear to miss is that this insight is implicitly at odds with the machinery of generative frameworks, which define grammars in a way that precludes the explicit description of lexical openness.

We are not claiming that no generative framework could be modified to incorporate an open lexicon. There would doubtless be some way to modify generative grammar to get the effect of a lexicon containing all possible well-formed phonological representations, the default being that phonological representations are associated with the disjunction of all sets of grammatical and semantic properties. And we are not claiming that model-theoretic frameworks entail that the lexicon is open: it is perfectly possible to close the lexicon of a model-theoretic grammar (in fact Rogers (1998:119) does this, for good reason, since he is interested in demonstrating full equivalence to a context-free generative grammar). What must be done to close the lexicon is to give an exhaustive disjunction of all the phonological realizations and all the grammatical features they can be paired with, thus disallowing for each lexical category all other realizations than the ones appearing in a finite list.

The two framework types therefore can, to some extent, simulate each other. But the contrast between them is nonetheless stark. They point in opposite directions. In a model-theoretic framework, additional content (the stipulated impermissibility of other shapes for each lexical category) has to be built into a grammar to get the

[12] Here, as in Pullum and Scholz (2001), we are essentially just reiterating and elaborating an important point made by Johnson and Postal 1980:675–7. It has since been discussed more fully by Postal (2004).

effect of a closed lexicon. A generative framework as standardly defined, on the other hand, would have to be modified (in a way that has so far never been worked out) in order to get the (desirable) effect of lexical openness.

Languages, expressions, and infinity

The dominance of the generative conception of how language is to be described has led to two items of dogma: that the set of all expressions belonging to a natural language is an infinite set, and that each of those infinitely many expressions is a finite object. We now proceed to argue that both of these claims are artifactual.

The myth that natural languages are demonstrably infinite

Contrary to popular belief, it has never been shown that natural languages have infinitely many expressions. To say this is to reject familiar arguments that are frequently repeated in introductory linguistics texts, encyclopedia articles, and other presentations. Stabler (1999:321) expresses the standard wisdom tersely by saying: "there seems to be no longest sentence, and consequently no maximally complex linguistic structure, and we can conclude that human languages are infinite." Such arguments have been much repeated over the past thirty or forty years; in (11) we offer samples, one statement from each of the last four decades.

(11) a. If we admit that, given any English sentence, we can concoct some way to add at least one word to the sentence and come up with a longer English sentence, then we are driven to the conclusion that the set of English sentences is (countably) infinite. (Bach 1974:24)

b. Is there anyone who would seriously suggest that there is a number, n, such that *n is a number* is a sentence of English and *n+1 is a number* is not a sentence of English . . . ? On this basis we take it as conclusively demonstrated that the number of English sentences is infinite and, therefore, that English cannot be equated with any corpus no matter how large. (Atkinson et al. 1982:35-36)

c. By the same logic that shows that there are an infinite number of integers—if you ever think you have the largest integer, just add 1 to it and you will have another—there must be an infinite number of sentences. (Pinker 1994:86)

d. It is always possible to embed a sentence inside of a larger one. This means that Language is an infinite system. Carnie (2002:13–14)

Some of these are more carefully worded than others, but clearly the same basic argument is repeated over and over again. It assumes that natural languages have the property of **productivity** — that is, they have expression-lengthening operations that preserve well-formedness. We do not question this. A sentence as simple as (12a) illustrates that English has expressions with tautocategorial embedding — i.e., a constituent of type α having a proper subconstituent of type α, as shown in (12b).

(12) a. *See Spot run away.*

b. [$_{\mathrm{VP}}$ *see Spot* [$_{\mathrm{VP}}$ *run away*]]

This suggests a lengthening operation that will also permit longer expressions such as *Let Jane see Spot run away*, *Watch Dick let Jane see Spot run away*, etc., and indeed these are grammatical.

Our use of the term 'operation' here is an informal reflection of the algebraic use of the term, not the algorithmic one. In algebra, a unary operation on a set is just a function $f : A \mapsto A$, and A is said to be **closed under** an operation iff $a \in A$ implies $f(a) \in A$. The authors quoted above apparently think that given any productive expression-lengthening operation it follows immediately that the set of well-formed sentences is countably infinite. It does indeed follow that the set formed by *closing* a set of expressions under a lengthening operation will be infinite. But the argument is supposed to be about natural languages such as English. What needs to be supported is the claim that (for example) English actually contains all the members of the closure of some set of English expressions under certain lengthening operations.

The illustrative quotations in (11) are attempts at providing that support. We will refer to the underlying form they all share as the Master Argument for language infinity. Let us try to restate it in more precise form. Let $E(x)$ mean 'x is a well-formed English expression' and let μ be a measure of expression size in integer terms — for simplicity we can let μ measure the length of the yield in words, so that $\mu(w_1 \ldots w_k) = k$. Then the argument runs like this:

(13) The Master Argument for language infinity

a. There is at least one well-formed English expression that has size greater than zero:

$$\exists x[E(x) \wedge \mu(x) > 0]$$

b. For all n, if some well-formed English expression has size n, then some well-formed English expression has size greater than n:

$$\forall n[\exists x[E(x) \wedge \mu(x) = n] \rightarrow \exists x[E(x) \wedge \mu(x) > n]]$$

c. Therefore, for every positive integer n there are well-formed English expressions with size greater than n (i.e., the set of well-formed English expressions is at least countably infinite):

$$\therefore \forall n \exists x[E(x) \wedge \mu(x) > n]$$

The Master Argument fails, in one of two different ways, depending on what sort of universe we apply it to — more specifically, what set we assign as the interpretation for the predicate E. There are two cases.

(i) If we choose a finite set as the extension for E, then clearly (13b) is false, and the argument is unsound.

(ii) If we choose an infinite set as the extension for E, then (13b) is true, but we have assumed what we set out to show — we have begged the question.

That is, the argument fails because if English were finite one of the premises would be false, while if English were infinite the argument would be circular, and we are given no way to tell which is the case. (This point is not new; we are basically just paraphrasing Langendoen and Postal 1984:30-35.)

If some argument did show that English or some other natural language had infinitely many expressions, the fit with generative grammars would be a good one in this respect, for given nontrivial and unbounded productivity, Post production systems and all kinds of generative grammars generate countably infinite sets of expressions. But in the absence of such an argument, the cardinality of the set generated by a generative grammar is irrelevant to the choice of a framework for linguistic theory.

Under a model-theoretic framework, no claim is made about how many natural language expressions there are. Although each constraint in a model-theoretic grammar will entail claims about the structure of individual expressions, nothing in any collection of constraints need entail any claim about the size of the set of all expressions.[13] A set of constraints that accurately characterizes the syntactic properties that well-formed English expressions have in common will be satisfied by each well-formed expression no matter how many or how few there are, assuming only that no restriction stated in the grammar places a ceiling on productivity — a reasonable assumption.

The model-theoretic view enables us to distinguish two issues: (i) the existence of productive lengthening operations in natural languages, and (ii) the cardinality of the set of all expressions in a language. Our thesis is that the first of these is relevant and important to the formulation of grammars for natural languages, but the second is not.

Natural languages and expressions of infinite length

Schiffer (1972) proposes a definition of what he calls 'mutual belief' under which what it would mean for Jones and Smith to have mutual belief of the proposition that iron rusts might be expressed in an infinite string of which the following is a representative initial subpart:

(14) *Jones believes that iron rusts, and Smith believes that iron rusts, and Jones believes that Smith believes that iron rusts, and Smith believes that Jones believes that iron rusts, and Jones believes that Smith believes that Jones believes that iron rusts, and...*

Joshi (1982:182) actually states the mutual belief schema in terms of an infinitely long conjunctive formula of this sort. The truth conditions of the infinite conjunction are clear enough under the ordinary interpretive principles for English: each of Jones and Smith believes that iron rusts, and each has a set of beliefs about the other's beliefs that is closed under reapplying *Jones believes that* to sentences about Smith's beliefs and vice versa.

But is the infinite string of which a small finite initial subpart is seen in (14) a grammatical English sentence? If not, it is certainly not clear why, since the string

[13]Of course, if one added to the grammar a statement such as 'there are not more than 374 nodes', structures would be limited to a certain finite maximum size, and thus (assuming also a finite bound on the set of node labels and relations) would be finite in number. But the point is that one does not have to if one has no warrant for doing so.

is entirely Englishlike in terms of its grammatical properties. But if it is, then some English expressions have infinite length.

No generative grammar can derive the fully expanded infinite version of 14, because derivations, by definition, complete in a finite number of steps. A derivation-like sequence that goes on adding terminals to the string but in a way that always adds new nonterminals as well never generates a terminal string under the standard definitions.

Linguists seem to have been led astray by this property of Post production systems and their progeny, taking the finiteness of expressions to be a truth about natural languages rather than a stipulated fact about a class of formal systems.[14] We see no fact about natural languages here.

We therefore take it as a point in favor of model-theoretic syntax that it permits description of the structure of expressions without entailing claims any about their size. The infinite sequence suggested by (14) would appear to satisfy all the constraints for English: subjects precede predicates, clausal complements follow their licensing verbs, present-tense verbs agree with subjects, and so on. Those properties are the ones that a syntax for English should describe. The description need not say anything about whether infinite length is possible for English expressions.[15]

What seems to be wanted here is the right to remain silent concerning expression size. A model-theoretic syntax for English can provide that, defining the infinite version of (14) as not violating any grammatical constraints but not insisting that it is ipso facto a sentence.[16] It would also be possible to stipulate that the infinite version of (14) is ungrammatical by using WMSOL: assuming tree structures as above, we would state something like (15), where X is a variable over sets of nodes:

(15) a. $\forall X[\exists x[X(x)] \rightarrow \exists x[X(x) \wedge \forall y[(y \lhd^* x \wedge y \neq x) \rightarrow \neg X(y)]]$

b. $\forall X[\exists x[X(x)] \rightarrow \exists x[X(x) \wedge \forall y[(y \prec x) \rightarrow \neg X(y)]]$

These statements say that every sequence of nodes ordered by domination or precedence comes to an end (Rogers 1998:22), which entails that structures are finite in size (though with no specific finite upper bound). Our point, however, is that

[14] The relevant definitions can easily be modified, and have been for theoretical reasons within logic and computer science (see Thomas 1990); but the standard definitions absolutely exclude infinite expressions.

[15] The observation that model-theoretic ("nonconstructive") grammars are compatible with infinite-size grammatical expressions is stressed in Langendoen and Postal (1984). What distinguishes our position from theirs is that they hold that natural languages are proper classes, closed under an operation of infinite coordinate compounding that renders them too large for the laws of set theory to apply. We have no space to discuss this position here, but we note one point. Langendoen and Postal claim that for every set X of sentences in a natural language L there is a coordinate sentence of L having all the members of X as its coordinates. This claim is not statable as an MTS constraint, because it is not interpretable on individual expressions. So under a strict construal of our position, Langendoen and Postal's closure generalization is not just unmotivated but actually unstatable.

[16] Note, though, that if a first-order logic is used as the description language, the infinite version of (14) cannot be blocked. It is a simple corollary of the compactness theorem for first-order logic that if a theory places no upper bound on the size of its finite models, it must have an infinite model. Thus any model-theoretic grammar that permits finite structures of arbitrary size (that is, any grammar that is at all plausible) *must* admit infinite structures if it is stated in a first-order language.

model-theoretic frameworks do not require this. We can state a grammar without any such stipulation, and restrict attention to an appropriate class of models as necessary for a given task. To prove Doner's theorem, we need to restrict attention to finite models; for other theoretical purposes, allowing infinite models might be appropriate. Model-theoretic syntax does not require that a decision be made once and for all on this, because model-theoretic grammars can make claims about the structural regularities expressions have without saying anything about how big expressions can be.

Conclusion

For nearly fifty years explicit grammars for natural languages have been formulated within generative frameworks that heuristically suggest that

- quasi-expressions have no syntactic properties at all;
- all quasi-expressions are ungrammatical to the same degree;
- the lexicons of natural languages are fixed;
- natural language expressions with pseudo-words are ungrammatical; and
- a natural language contains a countable infinity of expressions.

Our thesis is that all of these are artifacts of a view that is imposed on linguistics by generative frameworks, and leading to neglect of the actual properties of natural language.

Model-theoretic frameworks guide explicit grammar development in a notably different direction, correctly suggesting that

- quasi-expressions have a full array of syntactic properties describable by means of the same grammar that defines the grammatical expressions;
- quasi-expressions are ungrammatical in a multidimensional variety of ways and to an indefinitely large number of different degrees;
- the lexicon of a natural language is open and continually changing, often without the changes having any syntactic implications;
- natural language expressions with pseudo-words are fully grammatical as well as meaningful; and
- there is no theoretically important notion of the cardinality of the set of expressions in a natural language.

A rethinking of the mathematical and logical foundations of 20th-century theoretical syntax is in order. In 1960, when the first International Congress of Logic, Methodology and Philosophy of Science took place, linguists were rightly taken with the results of Chomsky (1959) on formal language theory, and few if any knew of the almost contemporaneous work of Büchi.[17] But generative frameworks best fit the task for which Post initially developed them: describing the syntax of artificial languages in mathematical logic. We argue for a different theoretical basis for syntax, based not on Post's formalization of proof theory but on the more recent work that has begun to forge a link between the syntax of natural languages and the semantic side of logic.

[17]Though Büchi was in fact present as the first Congress in 1960, and a paper of his, Büchi (1962), opens the proceedings volume.

References

Aho, A. V.: 1968, Indexed grammars — an extension of context-free grammars, *Journal of the Association for Computing Machinery* **15**, 647–671.

Atkinson, M., Kilby, D. and Roca, I.: 1982, *Foundations of general linguistics*, George Allen and Unwin, London.

Bach, E.: 1974, *Syntactic theory*, Holt Rinehart and Winston, New York.

Blackburn, P. and Gardent, C.: 1995, A specification language for lexical functional grammars, *Proceedings of the 7th EACL*, European Association for Computational Linguistics, pp. 39–44.

Blackburn, P. and Meyer-Viol, W.: 1997, Modal logic and model-theoretic syntax, *in* M. de Rijke (ed.), *Advances in Intensional Logic*, Kluwer Academic, Dordrecht, pp. 29–60.

Blackburn, P., Gardent, C. and Meyer-Viol, W.: 1993, Talking about trees, *Proceedings of the 1993 Meeting of the European Chapter of the Association for Computational Linguistics*, pp. 21–29.

Borščev, V. B. and Xomjakov, M. V.: 1973, Axiomatic approach to a description of formalized languages and translation neighbourhood languages, *in* F. Kiefer (ed.), *Mathematical Models of Language*, Soviet Papers in Formal Linguistics, Skriptor, Stockholm, pp. 37–114.

Büchi, J. R.: 1960, Weak second-order arithmetic and finite automata, *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik* **6**, 66–92.

Büchi, J. R.: 1962, On a decision method in restricted second-order arithmetic, *Proceedings of the International Congress on Logic, Methodology, and Philosophy of Science*, Stanford University Press, Stanford, CA, pp. 1–11.

Carnie, A.: 2002, *Syntax: A General Introduction*, Blackwell, Oxford.

Chomsky, N.: 1955, The Logical Structure of Linguistic Theory, Unpublished dittograph, microfilmed.

Chomsky, N.: 1959, On certain formal properties of grammars, *Information and Control* **2**, 137–167. Reprinted in *Readings in Mathematical Psychology*, Volume II, ed. by R. Duncan Luce, Robert R. Bush, and Eugene Galanter, 125–155, New York: John Wiley & Sons, 1965. [Note: The citation to the original on p. 125 of this reprinting is incorrect.].

Chomsky, N.: 1961, Some methodological remarks on generative grammar, *Word* **17**, 219–239. Section 5 republished as 'Degrees of grammaticalness' in Jerry A. Fodor and Jerrold J. Katz (eds.), *The Structure of Language: Readings in the Philosophy of Language*, 384-389 (Englewood Cliffs, NJ: Prentice-Hall).

Chomsky, N.: 1962, Explanatory models in linguistics, *in* E. Nagel, P. Suppes and A. Tarski (eds), *Logic, Methodology and Philosophy of Science: Proceedings of the 1960 International Congress*, Stanford University Press, Stanford, CA, pp. 528–550.

Chomsky, N. and Miller, G. A.: 1963, Finitary models of language users, *in* R. D. Luce, R. R. Bush and E. Galanter (eds), *Handbook of Mathematical Psychology*, Vol. II, John Wiley and Sons, New York, pp. 419–491.

Doner, J.: 1970, Tree acceptors and some of their applications, *Journal of Computer and System Sciences* **4**, 406–451.

Elgot, C. C.: 1961, Decision problems of finite automata and related arithmetics, *Transactions of the American Mathematical Society* **98**, 21–51.

Engelfriet, J.: 1993, A regular characterization of graph languages definable in monadic second-order logic, *Theoretical Computer Science* **88**, 139–150.

Gazdar, G., Klein, E., Pullum, G. K. and Sag, I. A.: 1985, *Generalized Phrase Structure Grammar*, Basil Blackwell, Oxford.

Gazdar, G., Pullum, G. K., Carpenter, B., Klein, E., Hukari, T. E. and Levine, R. D.: 1988, Category structures, *Computational Linguistics* **14**, 1–19.

Harkema, H.: 2001, A characterization of minimalist languages, *in* P. de Groote, G. Morrill and C. Retoré (eds), *Logical Aspects of Computational Linguistics: 4th International Conference*, number 2099 in *Lecture Notes in Artificial Intelligence*, Springer Verlag, Berlin, pp. 193–211.

Johnson, D. E. and Postal, P. M.: 1980, *Arc Pair Grammar*, Princeton University Press, Princeton, NJ.

Joshi, A.: 1982, Mutual beliefs in question-answer systems, *in* N. Smith (ed.), *Mutual Knowledge*, Academic Press, London, pp. 181–197.

Keenan, E. L. and Stabler, E.: 1996, Abstract syntax, *in* A.-M. D. Sciullo (ed.), *Configurations: Essays on Structure and Interpretation*, Cascadilla Press, Somerville, MA, pp. 329–344.

Kozen, D. C.: 1997, *Automata and Computability*, Springer, Berlin.

Kracht, M.: 1993, Syntactic codes and grammar refinement, *Journal of Logic, Language and Information* **4**, 41–60.

Kracht, M.: 2001, Logic and syntax: a personal view, *in* M. Zhakharyaschev, K. Segerberg, M. de Rijke and H. Wansing (eds), *Advances in Modal Logic 2*, CSLI Publications, Stanford, pp. 355–384.

Lakoff, G.: 1971, On generative semantics, *in* D. D. Steinberg and L. A. Jakobovitz (eds), *Semantics: An Interdisciplinary Reader in Philosophy, Linguistics and Psychology*, Cambridge University Press, Cambridge, pp. 232–296.

Langendoen, T. and Postal, P. M.: 1984, *The Vastness of Natural Languages*, Basil Blackwell, Oxford.

Langholm, T.: 2001, A descriptive characterisation of indexed grammars, *Grammars* **4**, 205–262.

McCawley, J. D.: 1968, Concerning the base component of a transformational grammar, *Foundations of Language* **4**, 243–269. Reprinted in James D. McCawley, *Grammar and Meaning*, 35–58 (New York: Academic Press; Tokyo: Taishukan, 1973).

Michaelis, J.: 2001, Transforming linear context-free rewriting systems into minimalist grammars, *in* P. de Groote, G. Morrill and C. Retoré (eds), *Logical Aspects of Computational Linguistics: 4th International Conference*, number 2099 in *Lecture Notes in Artificial Intelligence*, Springer Verlag, Berlin, pp. 228–244.

Miller, P.: 1991, Scandinavian extraction phenomena revisited: Weak and strong generative capacity, *Linguistics and Philosophy* **14**, 101–113.

Peters, P. S. and Ritchie, R. W.: 1969, Context-sensitive immediate constituent analysis — context-free languages revisited, *Proceedings of the ACM Conference on the Theory of Computing*, pp. 1–8. Republished in *Mathematical Systems Theory* 6 (1973), 324-333.

Peters, P. S. and Ritchie, R. W.: 1973, Context-sensitive immediate constituent analysis — context-free languages revisited, *Mathematical Systems Theory*

6, 324–333.

Pinker, S.: 1994, *The Language Instinct*, William Morrow, New York.

Post, E.: 1943, Formal reductions of the general combinatory decision problem, *American Journal of Mathematics* **65**, 197–215.

Post, E.: 1944, Recursively enumerable sets of positive integers and their decision problems, *Bulletin of the American Mathematical Society* **50**, 284–316.

Post, E.: 1947, Recursive unsolvability of a problem of thue, *Journal of Symbolic Logic* **12**, 1–11.

Postal, P. M.: 1972, The best theory, *in* P. S. Peters (ed.), *Goals of Linguistic Theory*, Prentice-Hall, pp. 131–170.

Postal, P. M.: 2004, The openness of natural languages, *Skeptical Linguistic Essays*, Oxford University Press, pp. 173–201.

Potts, C. and Pullum, G. K.: 2002, Model theory and the content of OT constraints, *Phonology* **19**, 361–393.

Pullum, G. K. and Scholz, B. C.: 2001, On the distinction between model-theoretic and generative-enumerative syntactic frameworks, *in* P. de Groote, G. Morrill and C. Retoré (eds), *Logical Aspects of Computational Linguistics: 4th International Conference*, number 2099 in *Lecture Notes in Artificial Intelligence*, Springer Verlag, Berlin, pp. 17–43.

Rabin, M. O.: 1969, Decidability of second-order theories and automata on infinite trees, *Transactions of the American Mathematical Society* **141**, 1–35.

Rogers, J.: 1994, *Studies in the Logic of Trees with Applications to Grammar Formalisms*, PhD thesis, University of Delaware, Newark, DE.

Rogers, J.: 1996, A model-theoretic framework for theories of syntax, *34th Annual Meeting of the Assocation for Computational Linguistics: Proceedings of the Conference*, Morgan Kaufmann, San Francisco, CA, pp. 10–16.

Rogers, J.: 1997, "Grammarless" phrase structure grammar, *Linguistics and Philosophy* **20**, 721–746.

Rogers, J.: 1998, *A Descriptive Approach to Language-Theoretic Complexity*, CSLI Publications, Stanford, CA.

Rogers, J.: 1999, The descriptive complexity of generalized local sets, *in* H.-P. Kolb and U. Mönnich (eds), *The Mathematics of Syntactic Structure: Trees and their Logics*, number 44 in *Studies in Generative Grammar*, Mouton de Gruyter, Berlin, pp. 21–40.

Rogers, J.: 2003, wMSO theories as grammar formalisms, *Theoretical Computer Science* **293**, 291–320.

Schiffer, S. R.: 1972, *Meaning*, Clarendon Press, Oxford.

Shieber, S.: 1985, Evidence against the context-freeness of human language, *Linguistics and Philosophy* **8**, 333–343.

Soames, S.: 1974, Rule orderings, obligatory transformations, and derivational constraints, *Theoretical Linguistics* **1**, 116–138.

Stabler, E.: 1997, Derivational minimalism, *in* C. Retoré (ed.), *Logical Aspects of Computational Linguistics, LACL '96*, number 1328 in *Lecture Notes in Artificial Intelligence*, Springer Verlag, Berlin, pp. 68–95.

Stabler, E.: 1999, Formal grammars, *in* R. A. Wilson and F. C. Keil (eds), *The MIT Encyclopedia of the Cognitive Sciences*, MIT Press, Cambridge, MA, pp. 320–322.

Straubing, H.: 1994, *Finite Automata, Formal Logic, and Circuit Complexity*, Birkhäuser, Boston, MA.

Thomas, W.: 1990, Automata on infinite objects, *in* J. van Leeuwen (ed.) (ed.), *Handbook of Theoretical Computer Science*, Elsevier Science, New York, pp. 135–191.

Weir, D. J. and Vijayshanker, K.: 1994, The equivalence of four extensions of context-free grammars, *Mathematical Systems Theory* **27**, 511–545.