Computational Linguistics and Generative Linguistics: The Triumph of Hope over Experience

Geoffrey K. Pullum
School of Philosophy, Psychology, and Language Sciences
University of Edinburgh
gpullum@ling.ed.ac.ul

Abstract

It is remarkable if any relationship at all persists between computational linguists (CL) and that part of general linguistics comprising the mainstream of MIT transformational-generative (TG) theoretical syntax. If the lines are still open, it represents something of a tribute to CL practitioners' tolerance — a triumph of hope and goodwill over the experience of abuse — because the TG community has shown considerable hostility toward CL and everything it stands for over the past fifty years. I offer some brief historical notes, and hint at prospects for a better basis for collaboration in the future.

1 Introduction

The theme of this workshop is the interaction between computational linguistics (CL) and general linguistics. The organizers ask whether it has it been virtuous, vicious, or vacuous. They use only three of the rather extraordinary number of *v*-initial adjectives. Is the relationship vital, valuable, venturesome, visionary, versatile, and vibrant? Or vague, variable, verbose, and sometimes vexatious? Has it perhaps been merely vestigial and vicarious, with hardly any general linguists really participating? Or vain, venal, vaporous, virginal, volatile, and voguish, yet vulnerable, a relationship at risk? Or would the best description use adjectives like vengeful, venomous, vilificatory, villainous, vindictive, violent, vitriolic, vociferous, and vulpine?

I will argue that at least with respect to that part of general linguistics comprising the mainstream of American theoretical syntax, it would be quite remarkable if any relationship with computational linguistics (CL) had thrived. It would represent (as Samuel Johnson remarked cynically, and wrongly, about second marriages) a triumph of hope over experience. It seems to me that the relationship that could have been was at least somewhat blighted by the negative and defensive stance that MIT-influenced transformational-generative (TG) syntacticians have adopted on a diverse array of topics highly relevant to CL.

There was never any need for such attitudes. And at the conclusion of these brief remarks I will suggest a basis for thinking that relations could be much more satisfactory in the future. But I think it is worth taking a sober look at the half-century of history from 1959 to 2009, during which almost everything about the course of theoretical syntax, at least in the USA, where I worked during the latter half of the period, has been tacitly guided by a single line of thinking. 'Generative grammar' is commonly used to denote it, but that will not do. First, 'generative grammar' is often used to mean 'MIT-influenced transformational-generative grammar'. For that I will use the abbreviation TG. And second, it is sometimes (incorrectly) claimed that 'generative' means nothing more or less than 'explicit' (see Chomsky 1966, 12: 'a *generative grammar* (that is, an explicit grammar that makes no appeal to the reader's "faculé de langage" but rather attempts to incorporate the mechanisms of this faculty)').

We need more precise terminology in order to home in on what I am talking about. As Seuren (2004) has stressed, the relevant vision of what a grammar is like, built into most linguistic theorization today at a level so deep that most linguists are incapable of seeing past it or out of it, is not just that it is explicit, but that a grammar is and must be a syntax-centered random generator. I will therefore refer to **language specification by random generation** (LSRG).

The definitive technical paper defining grammars in LSRG terms is Chomsky (1959). This was a fine paper, which would have earned its

Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics, pages 12–21,
Athens, Greece, 30 March, 2009. ©2009 Association for Computational Linguistics

writer tenure in any department of linguistics, logic, computer science, or mathematics that knew what it was doing and could see the possibilities. But it brought into linguistics two things that were not going to go away for half a century. One was the notion that any formally precise linguistics had to be limited to LSRG. And the other was the combative and insular personality of the paper's author, which had such a great influence on the personality of his extraordinarily important department at MIT.

2 Obsession with random generation

The sense of 'generate' relevant to LSRG goes back to the work of the great mathematical logician Emil Post (as acknowledged by Chomsky 1959, 137n). Post's project was initially to formalize the notion of proof in logical systems — originally, the propositional logic that was informally used but not formally defined in Whitehead and Russell's *Principia Mathematica*. He represented well-formed formulae ('enunciations', including the 'primitive assertions', i.e. axioms, and the 'assertions' i.e. theorems) to be simply strings over a finite set of symbols, and rules of inference ('productions') as instructions for deriving a new string (the conclusion) from a set of strings already in hand (the premises). He then studied the question of what kinds of sets of strings could be generated if a set of initial strings were closed under the operation of applying inference rules. Post's rather ungainly general presentation of the general concept of rules of inference, or in his terms **productions**, looks like this:

$$\boxed{\begin{array}{c} g_{1_1}\ P_{i'_1}\ g_{1_2}\ P_{i'_2}\ \cdots\ g_{1_{m_1}}\ P_{i'_{m_1}}\ g_{1_{m_1+1}} \\ g_{2_1}\ P_{i''_1}\ g_{2_2}\ P_{i''_2}\ \cdots\ g_{2_{m_2}}\ P_{i''_{m_2}}\ g_{2_{m_2+1}} \\ \cdots\cdots\cdots\cdots \\ g_{k_1}\ P_{i'^k_1}\ g_{k_2}\ P_{i'^k_2}\ \cdots\ g_{k_{m_k}}\ P_{i'^k_{m_k}}\ g_{k_{m_k+1}} \\ \text{produce} \\ g_1\ P_{i_1}\ g_2\ P_{i_2}\ \cdots\ g_m\ P_{i_m}\ g_{m+1} \end{array}}$$

In specific instances of productions the g metavariables in this schema are replaced by actual strings over what is now known as the **terminal vocabulary**. The P metavariables function as cover symbols for arbitrary stretches of material — they are string variables, some of which may be repeated to copy material into the conclusion. A production provides a license, given a set of strings that match patterns of the form $g_0P_1g_1P_2\cdots P_kg_k$, to produce a certain other string composed in some way out of the various g_i and P_i.

Post defined a class of **canonical systems**, each consisting of a set of initial strings and a finite set of productions. Sets generated by canonical production systems he called **canonical sets**. Post had realized early on that the canonical sets were nothing more or less than the sets definable by recursive functions or Turing machines; that is, they were just the recursively enumerable (r. e.) sets.

He proceeded to prove that even if you restrict the number and distribution of the g_i and P_i extremely tightly, expressive power may not be reduced. Specifically, he proved that no reduction in the definable sets is obtained even if you set the number of P variables and the number of premises at 1, and require that every production has the form 'g_0P produces Pg_1'. Such very restricted systems were called **normal systems**. Normal systems can still derive every canonical set, provided you are allowed to use extra symbols that appear in derivations but not in the ultimately generated strings (these extra symbols are what would become known to formal language theorists within computer science as **variables** and to linguists as **nonterminals**).

In a notation more familiar to linguists, the result amounts to showing that every r. e. subset of Σ^+ can be generated by some generative grammar using a symbol vocabulary $V = \Sigma \cup N$ in which all rules have the form '$xW \rightarrow Wy$' for specified strings $x, y \in V^*$ and some fixed $W \in V^*$. This was the first weak generative capacity result: normal systems are equivalent in weak generative capacity to full canonical systems.

In a later paper, settling a conjecture of Thue, Post showed (1947) that you can derive every canonical set if your productions all have the form '$P_1g_iP_2$ produces $P_1g_jP_2$'. This amounts to showing that every canonical subset of Σ^+ can be generated by (what would later be called) a generative grammar using a symbol vocabulary $V = \Sigma \cup N$ in which all rules have the form '$WxZ \rightarrow WyZ$' for specified strings $x, y \in V^*$ and fixed $W, Z \in V^*$.

Hence the first demonstration that unrestricted rewriting systems (Chomsky's 'type-0' grammars) can derive any r. e. set was not original with Chomsky (1959). It had been published twelve years earlier by Post.

Post had in effect invented what could be called

top-down random generators. These randomly generate r. e. sets of symbols by expanding an initial axiomatic string, which can be just a single symbol. Their equivalence to Turing machines is obvious (Kozen 1997, 256–257).

Between the time of Post's doctoral work in 1920 and the 1943 paper in which he published his result on canonical systems (already present in compressed form in his thesis), Ajdukiewicz (1935) had proposed a different style of generative grammar, also motivated by the development of a better understanding of proof. Adjukiewicz's invention was categorial grammar, the first kind of **bottom-up random generators**. It composes expressions of the generated language by combining parts — initially primitive categorized symbols, and then previously composed subparts.

When Chomsky and Lasnik (1977) start talking about the 'computational system' of human language (a mode of speaking that rapidly caught on, and persists in current 'minimalist' work), the 'computation' of which they spoke was one that takes place nowhere: no such computations are ever done, except perhaps using pencil and paper as a syntactic theorist tries to figure out how or whether a certain string can be derived. This 'computational system' attempts randomly and nondeterministically to find some way to apply rules in order to build a particular structure, starting from an arbitrary syntactic starting point.

In the case of pre-1990 work the starting point was apparently a start symbol; in post-1990 'minimalist' work it is a **numeration**: a randomly chosen multiset of categorized items from the lexicon. The concept of a 'numeration' is a reflection of how firmly embedded the random-generation idea is. The numeration serves no real purpose. It would be possible to formalize a grammar as a set of combinatory principles for putting together words in a string as encountered, from first to last, so that it was in effect a parser. Categorial grammars seem ideally suited to that role (Steedman, 2000), and minimalist grammars are really just a variety of categorial grammar, stripped of some of the formal coherence and links to logic and semantics.

Chomsky has often written as if it were a necessary truth that a grammar must be a random generator. For example: 'Clearly, a grammar must contain … a 'syntactic component' that generates an infinite number of strings representing grammatical sentences … This is the classical model for grammar' (Chomsky 1962, 539). This says that a grammar *must* be a random generator. But this is not true. A grammar could in principle be formulated as, say a transducer mapping phonetic representation inputs to corresponding sets of logical forms. (Presumably this must be possible, given what human beings do.)

It is particularly strange to see Chomsky ignoring this possibility and yet asserting in *Knowledge of Language* (Chomsky, 1986b) that a person's internalized grammar 'assigns a status to every relevant physical event, say, every sound wave' (p. 26). The claim is false, simply because random generators are not transducers or functions: they do not take inputs. A random generator only 'assigns a status' to a string by generating it with a derivation that associates it with certain properties. And surely it is not a sensible hypothesis about human linguistic competence to posit that in the brain of every human being there is an internalized random generator generating every physically possible sequence of sounds, from a ship's foghorn to Mahler's ninth symphony.

3 Downplaying expressive power

Perhaps the most centrally important reason for linguists' concern with the possibility of excess expressive power in grammar formalisms was their sense that it should be guaranteed by the general theory of grammar that linguistic behaviors such as understanding a sentence should be represented as at least possible. This meant that grammars had to be defined in a way that at least made the general membership problem ('Given grammar G, is string w grammatical?') decidable.

It was in Chomsky's 1959 paper that progress was first made toward restricting the expressive power of production systems in ways that achieved this, and the early work on topics like pushdown automata and finite state machines shows that those topics were of interest.

As is well known, Chomsky showed that if productions of the general form $X\varphi Z \rightarrow X\psi Z$ (where X, ψ, Z are strings in V^* and $\varphi \in V^+$) are limited by the condition that ψ is no shorter than φ, we are no longer able to derive every r. e. set of strings over the alphabet; we get only the context-sensitive stringsets. If the further limitation that $\varphi \in N$ is imposed, we get only the context-free stringsets. And if on top of that the requirement that $\psi \in (\Sigma \cup \Sigma N)$ is imposed, we get only the regular sets.

Chomsky's 1959 position was that the set of all grammatical English word sequences was not a regular stringset over the set of English words, and that if any context-free grammar for English could be constructed, it would not be an elegant or revealing one. The search for intuitively adequate grammars therefore had to range over the class of grammars generating context-sensitive stringsets. This is a large class of grammars, but at least it is a proper subset of the class of grammars for which the membership recognition problem is decidable. Casting around outside that range was probably not sensible, since natural languages surely had to be decidable (it was taken to be quite obvious that native speakers could rapidly recognize whether or not a string of words was a sentence in their language).

As I have detailed elsewhere in somewhat tongue-in-cheek fashion (Pullum, 1989), Chomsky pulled back sharply from his initial interest in mathematical study of linguistic formalisms as it became clear that TG theories were being criticized for their Turing-equivalence, and began dismissing precise studies of the generative capacity of grammars as trivial and ridiculous. This, it seems to me, was one more clear sign of distancing from the concerns of CL. It was mainly computational linguists who showed interest in Gazdar's observation that a theory limited to generating context-free languages could guarantee not just recognition but recognition in polynomial (indeed, better than cubic) time, and in the related observation that none of the arguments for non-context-free characteristics in human languages seemed to be good ones (Pullum and Gazdar, 1982).

The MIT reaction to Gazdar's suggestion was to mount a major effort to find intractability in Gazdar-style (GPSG) grammars — to represent the recognition problem as NP-hard even for context-free-equivalent theories of grammar (Barton et al., 1987). This was something of a confidence trick. First, the results depended on switching attention from the fixed-grammar arbitrary-string recognition problem (the analog of what Vardi (1982) calls data complexity) to the variable-grammar arbitrary-string recognition problem (what Vardi calls combined complexity). Second, it seemed to be vaguely assumed that only GPSG had any charges to answer, and that the GB theory of that time (Chomsky, 1981) would not suffer from similar computational complexity problems, but GB eventually turned out to be, insofar as it was well defined, strongly equivalent to Gazdar's framework (Rogers, 1998).

For pre-GB varieties of TG, however, the problem had mainly been not that recognition was NP-hard but that it was not computable at all: transformational grammars from 1957 on kept proving to be Turing-equivalent. That was what seems to have driven the denigration of mathematical linguistics, and the downplaying of the relevance of decidability to such an extreme degree (see e.g. Chomsky 1980: 120ff, where the very idea that recognition is decidable is dismissed as an unimportant detail, and not necessarily even a true claim).

4 Hostility to machine testing

With many versions of TG offering no guarantee that there was any parser for the language even in principle, it was not clear that machine testing of grammatical theories by algorithmic checking of claims made about grammaticality of selected strings was a plausible idea. Perhaps machine theorem-proving algorithms could have been adapted to showing that a certain grammar could indeed derive a certain string, but in practice early transformational grammar was vastly too complex to permit the building of tools for grammar testing, and later transformational grammar far too vague.

I know of only one success story in grammar evaluation by implementing random generation, in fact. Ed Stabler (1992) coded up a Prolog grammaticality-proving system based on the *Barriers* theory of transformational grammar (Chomsky, 1986a), which (Pullum, 1989) had mocked for sloppiness of statement. The *Barriers* system had in particular abandoned the usual practice of defining trees in a way that had dominance as a reflexive relation. Chomsky casually asserted that he would take it to be irreflexive. Moreover, Stabler's careful and sympathetic reconstruction of Chomsky's intent defines the notion of 'exclusion' in such a way that every node excludes itself (Chomsky's definition said that 'α excludes β if [and only if] no segment of α dominates β', and of course a given α never dominates itself). And sure enough, the Stabler implementation revealed that this system of definitions had a problem: unbounded dependency constructions that Chomsky took to be allowed were in fact blocked by his theoretical machinery.

Stabler concluded from his discovery 'that the project of implementing GB theories transparently is both manageable and worthwhile'. But his paper has essentially never been referred to by any mainstream syntacticians. It was not exactly what they wanted to hear. Nor has anyone, to my knowledge, utilized Stabler's experience in doing syntactic research using the *Barriers* framework.

There has in any case traditionally been considerable resistance to machine testing of theories. I have heard a story told by MIT linguists of how one early graduate student devised a computer program to test the rule system of SPE, and told Morris Halle about some of the bugs he had thereby found, but Halle had already noticed all of them. The moral of the story is clearly supposed to be that machine testing is unneeded and of no value.

Mark Johnson as an undergraduate did some work showing that the Unix stream editor **sed** could serve as an excellent tool for implementing systems of ordered historical sound changes for the assistance of comparative-historical linguists; but this very sensible idea never led to widespread testing of synchronic phonological ordered-rule analyses.

In short, computational testbeds, however enthusiastically developed in some areas of science (chemistry, astrophysics, ecology, molecular biology), simply never (yet) took off in linguistic science.

5 Loathing of corpora

There has traditionally been hostility even to machine data-hunting or language study through computer-searchable corpora. This is fading away as a new generation of young linguists who do everything by searching the web do their data by web search too; but it held back collaboration for a long time. Early proposals for amassing computer corpora were treated with contempt by TG grammarians ('I'm a native speaker, I have intuitions; why do I need your arbitrary collection of computer-searchable text?').

And quite often evidence from attested sentences is simply dismissed. To take a random example, on page 48 of Postal (1971) the sentences *I am annoying to myself* is prefixed with an asterisk to show that it is ungrammatical. Searching for this exact strings using Google, as we can do today, reveals that it gets 229 hits. I take this multiple attestation to shift the burden overwhelmingly against the linguist who claims that it is barred by the grammar of the language. But anyone who has experience (as I do) with trying to talk TG linguists out of their beliefs by citing attested sentences will know that it is between the difficult and the impossible. From 'There are many errors in published works' to 'It may be OK for him, but it's not for me', there are many ways in which the linguist can escape from the conclusion that a machine has proved superior in assessing the data.

Hostility to corpus work has probably to some extent paved the way for the present situation, where the machine translation teams at Google's research labs has no linguists, the work depending entirely on heavily numerical tracking of statistical parallels seen in aligned bilingual texts.

And an unwholesome split is visible in the linguistics community between those who broadly want nothing to do with corpora and think personal intuitions are fine as a basis for data gathering, and the people that I have called corpus fetishists who treat all facts as unclean and unholy unless they come direct and unedited out of a corpus. At the extremes, we get a divide between dreamers and token-counters — on the one hand, people who think that speculations on how universal principles might account for subtle shades of their own inner reactions to particular sentences, and on the other, people who think that counting the different pronouns in ten million words of text and tabulating the results is a contribution to science.

6 Aversion to the stochastic

Mention of statistics reminds us that stochastic methods have revolutionized CL since the 1980s, but have made few inroads into general linguistics, and none into TG linguistics. This is despite the excellent introduction to probabilistic generative grammars provided in Levelt's excellent and far-sighted introduction to mathematical linguistics (Levelt, 1974), the first volume of which has now been republished separately (Levelt, 2008).

The reason for the extraordinarily low profile of probabilistic grammars within the ranks of TG linguists has to do with the very successful attack on the very possibility of their relevance in *Syntac-*

tic Structures (Chomsky, 1957). Insisting that any statistical model for grammaticality would have to treat *Colorless green ideas sleep furiously* and *Furiously sleep ideas green colorless* in exactly the same terms, as they are word strings with the same (pre-1957) frequency of zero, Chomsky argued that probability of a string had no conceivable relevance to its grammaticality.

Unfortunately he had made a mistake. He was tacitly assuming that the probability of an event type that has not yet occurred must be zero. Maximum likelihood estimation (MLE) does indeed yield that result; but Chomsky was not obliged to adopt MLE. The technique now known as smoothing had been developed during the Second World War by Alan Turing and I. J. Good, and although it took a while to become known, Good had published on it by 1953. Chomsky was simply not acquainted with the statistical literature and not interested in applying statistical methods to linguistic material. Most linguists for the next forty years followed him in his disdain for such work. But when Pereira (2000) finally applied Good-Turing estimation (smoothing) to the question of how different the probabilities of the two famous word sequences are from normal English text, he found that the first (the syntactically well-formed one) had a probability 200,000 times that of the second.

7 Contempt for applications

Theoretical linguists have tended to have an almost total lack of interest in anything that might offer a practical application for their theories. Most kinds of science tend eventually to support some sort of engineering or practical techniques: physics led to jet planes; geology gave us oil location methods; biology brought forth gene splicing; even logic and psychology have applications in factories and other workplaces. But not mainstream theoretical linguistics. Its theories do not seem to yield applications of any sort.

Very early on, Chomsky found that he had to distance himself from computers altogether: note the remark in Chomsky (1966, 9) that 'Quite a few commentators have assumed that recent work in generative grammar is somehow an outgrowth of an interest in the use of computers for one or another purpose, or that it has some engineering motivation', and note that he calls such views both 'incomprehensible' and 'entirely false'. Being taken to have ambitions relating to natural language processing was at that time clearly anathema for the leader of the TG community.

What takes the place of application of theories to practical domains today, since nothing has come of any computational TG linguistics, is an attempt to derive conclusions about human brain organization and mental anatomy. Linguists claim to be biologists rather than psychologists (psycholinguistics developed its own experimental paradigms and began its own steady progress away from interaction with TG linguistics). There is a journal called *Biolinguistics* now, and much talk about interfaces and evolution and perfection. Linguists somehow live with the fact that the real biologists and neurophysiologists are not getting involved.

It is probably this pretense at uncovering deep principles of structure in a putative mental organ (and pretense is what it is) that is responsible for the dramatic falling off of interest in precise description of languages. Getting the details right — what was described as 'observational adequacy' in *Aspects* (Chomsky, 1965) — is taken to be a low-prestige occupation when compared to one that is alleged to offer glimpses of universal principles that hold the key to language acquisition and the innate cognitive abilities of the species.

Yet these universal principles are never actually presented for examination in the way that genuine results in science are. It is as if what is important to the hunter after universal principles is the hunt itself, the call of the horn and the thrill of the chase, but not the grubby business of examining and weighing the kill. The fact is that no really robust and carefully formulated universals of language have been discovered, described, promulgated, confirmed, and widely accepted as correct in the fifty years that universals have been sought.

The notion that linguists have discovered innate principles that solve the mystery of first language acquisition (Scholz and Pullum, 2006) is particularly pernicious. The position generally advocated by TG linguists is widely known as **linguistic nativism**, and it says that some significant aspects of knowledge of language are not derived from any experience but are innately known. But when pressed on the question of what the evidence shows about linguistic nativism, about whether it can really be defended against its plausible rivals, nativists tend to react by drawing back very sharply into a trivial form of the thesis: of course linguistic nativism must be true, they insist, because when you raise a baby and a kitten in the same household under the same conditions it is only the baby ends up with knowledge of language. They therefore differ in some respect, innately. 'Universal grammar' is simply one name that linguists use for that which separates them: whatever it is that human infants have but kittens and monkeys and bricks don't.

But of course, that makes the thesis trivial: it is true in virtue of being merely a restatement of the observation that led to linguistic nativism being put forward. We know that it is only human neonates who accomplish the language acquisition task, and that is why we are seeking an explanatory theory of how humans accomplish the task. To say that there must be something special about them is certainly true, but that does not count as a scientific discovery. We need specifics. Serious scientists are like the private sector as characterized in the immortal line uttered by Ray Stantz (played by Dan Ackroyd) in *Ghostsbusters*: 'I've worked in the private sector. They expect results!'

8 Hope for the future

It is absolutely not the case that general and theoretical linguistics should continue to act as if the main object were to prevent any interaction with CL. Let me point to a few hopeful developments.

Over the period from about 1989 to 2001, a team of linguists worked on and completed a truly comprehensive informal grammar of the English language. It was published as Huddleston and Pullum et al. (2002), henceforth *CGEL*. It is an informal grammar, intended for serious academic users but not limited to those with a linguistics background. And it comes close to being fully exhaustive in its coverage of Standard English grammatical constructions and morphology.

It should not be forgotten that the era of TG, though it produced (in my view) no theories that are really worth having, an enormous number of interesting data discoveries about English were made. *CGEL* profited greatly from those, as the Further Reading section makes clear. But does not attempt to develop theoretical conclusions or participate in theoretical disputes. Wherever possible, *CGEL* takes a largely pretheoretic or at least basically neutral stance.

Where theoretical commitments have to be made explicit, they are, but they are then implemented in consistent terms across the entire book. Although more than a dozen linguists were involved, it is not an anthology; Huddleston and Pullum provide a unitary authorial voice for the book and rewrote every part of the book at least once. When disputes about analyses arose between the authors who drafted different chapters, they were settled one way or the other by recourse to evidence, and not permitted to create departures from consistency in the book as a whole.

CGEL was preceded by large-scale 3-volume grammars for Italian (Renzi et al., 2001) and for Spanish (Bosque and Demonte, 1999), and now a grammar of French on a similar scale, the *Grande Grammaire du français* is being written by a team of linguists in Paris under the leadership of Anne Abeillé (Paris 7), Annie Delaveau (Paris 10), and Danièle Godard (CNRS). In 2006 I visited Paris at the request of that team to give a workshop on the making of *CGEL*. Work continues, and the book is now planned for publication by Editions Bayard in 2010. If anything the scope of this work is broader than *CGEL*'s, since *CGEL* did not aim to cover uncontroversially non-standard dialects of English (for example, those that have negative concord), whereas the *Grande Grammaire* explicitly aims to cover regional and non-standard varieties of French. Additionally, an effort to produce a comparable grammar of Mandarin Chinese is now being mounted in Hong Kong under the directorship of Professor Chu-Ren Huang, the dean of the new Faculty of Humanities at Hong Kong Polytechnic University. I gave a workshop on *CGEL* there (in March 2009) too.

The importance of these projects is simply that they bear witness to the fact that, at least in some areas, there are linguists — and not just isolated individuals but teams of experienced linguists — who are prepared to get involved in detailed language description of the type that will be a prerequisite to any future computational linguistics that relies on details of syntax and semantics (rather than probabilistic number-crunching on n-grams and raw text, which has its own interest but does not involve input from linguistics or even a rudimentary knowledge of the language being processed). Among them are both traditional general linguists like Huddleston and people with serious CL experience like Abeillé and Huang.

But there is more. I have made a preliminary analysis of the inventory of syntactic categories used in the tagging for labelling trees in the

Penn Treebank (Marcus et al., 1993), comparing them to the categories used in *CGEL*. I would describe the fit as not perfect, but within negotiating range. In some ways the fit is remarkable, given the complete independence of the two projects (the Treebank under Mitch Marcus in Philadelphia was largely complete by 1992, when the *CGEL* project under the direction of Rodney Huddleston in Australia was only just getting up to speed, but Huddleston and Marcus did not know about each other's work).

The biggest discrepancy in categorization is in the problematic area of prepositions, adverbs, and subordinating conjunctions, where the Treebank has remained much too close to the confused older tradition (where many prepositions are claimed to have second lives as adverbs and quite a few are also included on the list of subordinating conjunctions, so that a word like *since* has one meaning but three grammatical categories). The heart of the problem is that the sage counsel of Jespersen (1924, 87–90) and the cogent arguments of Emonds (1972) were not taken under consideration by the devisers of the Treebank's tagging categories. But fixing that would involve nothing more than undoing some unmotivated partitioning of the preposition category.

Since there are few if any significant disagreements about bracketing, and the category systems could be brought into alignment, I believe it would not be a major project to convert the entire Penn Treebank into an alternate form where it was totally compatible with *CGEL* in the syntactic analyses it presupposed. There could be considerable value in a complex of reference tools that included a treebank of some 4.5 million words that is fully compatible in its syntactic assumptions with an 1,860-page reference grammar of high reliability and consistency.

And there is yet more. Here I will be brief, and things will get slightly technical. The question naturally arises of how one might formalize *CGEL* to get it in a form where it was explicit enough for use as a database that natural language processing systems could in principle make use of. James Rogers and I have recently considered that question (Pullum and Rogers, 2008) within the context of model-theoretic syntax, a line of work that first began to receive sophisticated formulations here at the EACL in various papers of the early 1990s (e.g. Blackburn et al. (1993), Kracht (1993), Blackburn & Gardent (1995); see Pullum (2007) for a brief historical survey, and Pullum & Scholz (2001) for a deeper treatment of relevant theoretical issues).

One thing that might appear to be a stumbling-block to formalizing *CGEL*, and an obstacle to the relationship with treebanks as well, is that strictly speaking *CGEL*'s assumed syntactic representations are not (or not all) trees. They are graphs that depart from being ordinary constituent-structure trees in at least two respects.

First, they are annotated not just with categories labelling the nodes, but also with syntactic functions (grammatical relations like Subject-of, Determiner-of, Head-of, Complement-of, etc.) that are perhaps best conceptualized as labelling the edges of the graph (the lines between the nodes in the diagrams).

Second, and perhaps more seriously, there is occasional downward convergence of branches: it is permitted for a given constituent, under certain conditions, to bear two different grammatical relations to two different superordinate nodes. (A determinative like *some*, for example, may be both the Determiner of an NP and the Head of the Nominal that is the phrasal head of that NP.) Often (as in HPSG work) the introduction of re-entrancy had dramatic consequences for key properties like decidability of satisfiability for descriptions, or even for model-checking. (I take it that the formal issues around HPSG are very well known to the EACL community. In this short paper I do not try to deal with HPSG at all. There is plenty to be said, but also plenty of excellent HPSG specialists in Europe who are more competent than I am to treat the topic.)

Pullum & Rogers (2008) shows, however, that given certain very weak conditions, which seem almost certainly to be satisfied by the kinds of grammatical analysis posited in grammars of the *CGEL* sort, there is a way of constructing a compatible directed ordered spanning tree for any *CGEL*-style syntactic structure in such a way that no information is lost and reachability via edge chains is preserved. Moreover, the mapping between *CGEL* structures and spanning trees is definable in weak monadic second-order logic (wMSO).

Put this together with the results of Rogers (1998) on definability of trees in wMSO, and there is a clear prospect of the *CGEL* analysis of English syntax being reconstructible in terms of the wMSO theory of trees. And what that means for parsing is clear from results of nearly 40 years ago (Doner, 1970): there is a strong equivalence via tree automata to context-free grammars, which means that all the technology of context-free parsing can potentially be brought to bear on processing them.

This does not mean it would be a crisis if some language of interest is found to be non-context-free, incidentally. By the results of Rogers (2003), wMSO theories interpreted on tree-like structures of higher dimensionality than 2 could be employed. For example, where the structures are 3-dimensional (so that individual nodes are allowed to bear the parent-of relation to all of the nodes in entire 2-dimensional trees), the string yield of the set of all structures satisfying a given wMSO sentence is always a tree-adjoining language, and for every tree-adjoining language there is such a characterizing wMSO sentence.

Notice, by the way, that the theoretical tools of use here are coming out of currently very active subdisciplines of computational logic and automata theory, such as finite model theory, descriptive complexity theory, and database theory. The very tools that linguistics needs in order to formalize syntactic theories in a revealing way are the ones that theoretical computer science is intensively working on because their investigation is intrinsically interesting.

To sum up, what this is all telling us is that there is no reason for anyone to continue being guided by the TG bias toward isolating theoretical linguistics from CL. There is not necessarily a major gulf between (i) cutting-edge current theoretical developments like model-theoretic syntax, (ii) large-scale descriptive grammars like *CGEL*, and (iii) feasible computational natural-language engineering. Given the excellent personal relations between general linguists and computational linguists in some European locations (Edinburgh being an excellent example), it seems to me that developments in interdisciplinary relations that would integrate the two disciplines quite thoroughly could probably happen quite fast. Perhaps it is happening already.

Acknowledgments

I am very grateful to Barbara Scholz for her detailed criticisms of a draft of this paper. I have taken account of many of her helpful suggestions, but since she still does not agree with what I say here, none of the failings or errors above should be blamed on her.

References

Kazimierz Ajdukiewicz. 1935. Die syntaktische konnexität. *Studia Philosophica*, 1:1–27. Reprinted in Storrs McCall, ed., *Polish Logic 1920–1939*, 207–231. Oxford: Oxford University Press.

G. Edward Barton, Robert C. Berwick, and Eric Sven Ristad. 1987. *Computational Complexity and Natural Language*. MIT Press, Cambridge, MA.

Patrick Blackburn and Claire Gardent. 1995. A specification language for lexical functional grammars. In *Seventh Conference of the European Chapter of the Association for Computational Linguistics: Proceedings of the Conference*, pages 39–44, Morristown, NJ. European Association for Computational Linguistics.

Patrick Blackburn, Claire Gardent, and Wilfried Meyer-Viol. 1993. Talking about trees. In *Sixth Conference of the European Chapter of the Association for Computational Linguistics: Proceedings of the Conference*, pages 21–29, Morristown, NJ. European Association for Computational Linguistics.

Ignacio Bosque and Violeta Demonte, editors. 1999. *Gramática Descriptiva de La Lengua Española*. Real Academia Española / Espasa Calpe, Madrid. 3 volumes.

Noam Chomsky and Howard Lasnik. 1977. Filters and control. *Linguistic Inquiry*, 8:425–504.

Noam Chomsky. 1957. *Syntactic Structures*. Mouton, The Hague.

Noam Chomsky. 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 (citation to the original on p. 125 of this reprinting is incorrect).

Noam Chomsky. 1962. Explanatory models in linguistics. In Ernest Nagel, Patrick Suppes, and Alfred Tarski, editors, *Logic, Methodology and Philosophy of Science: Proceedings of the 1960 International Congress*, pages 528–550, Stanford, CA. Stanford University Press.

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.

Noam Chomsky. 1966. *Topics in the Theory of Generative Grammar*. Mouton, The Hague.

Noam Chomsky. 1980. *Rules and Representations*. Basil Blackwell, Oxford.

Noam Chomsky. 1981. *Lectures on Government and Binding*. Foris, Dordrecht.

Noam Chomsky. 1986a. *Barriers*. MIT Press, Cambridge, MA.

Noam Chomsky. 1986b. *Knowledge of Language: Its Origins, Nature, and Use*. Praeger, New York.

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

Joseph E. Emonds. 1972. Evidence that indirect object movement is a structure-preserving rule. *Foundations of Language*, 8:546–561.

Rodney Huddleston, Geoffrey K. Pullum, et al. 2002. *The Cambridge Grammar of the English Language*. Cambridge University Press, Cambridge.

Otto Jespersen. 1924. *The Philosophy of Grammar*. Holt, New York.

Dexter Kozen. 1997. *Automata and Computability*. Springer, Berlin.

Marcus Kracht. 1993. Mathematical aspects of command relations. In *Sixth Conference of the European Chapter of the Association for Computational Linguistics: Proceedings of the Conference*, pages 240–249, Morristown, NJ. Association for Computational Linguistics.

W. J. M. Levelt. 1974. *Formal Grammars in Linguistics and Psycholinguistics. Volume I: An Introduction to the Theory of Formal Languages and Automata; Volume II: Applications in Linguistic Theory; Volume III: Psycholinguistic Applications*. Mouton, The Hague.

W. J. M. Levelt. 2008. *An Introduction to the Theory of Formal Languages and Automata*. John Benjamins, Amsterdam.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

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

Paul M. Postal. 1971. *Crossover Phenomena*. Holt, Rinehart and Winston, New York.

Geoffrey K. Pullum and Gerald Gazdar. 1982. Natural languages and context-free languages. *Linguistics and Philosophy*, 4:471–504.

Geoffrey K. Pullum and James Rogers. 2008. Expressive power of the syntactic theory implicit in *the cambridge grammar of the english language*. Paper presented at the annual meeting of the Linguistics Assocition of Great Britain, University of Essex, September 2008. Online at http://ling.ed.ac.uk/~gpullum/EssexLAGB.pdf.

Geoffrey K. Pullum and Barbara C. Scholz. 2001. On the distinction between model-theoretic and generative-enumerative syntactic frameworks. In Philippe de Groote, Glyn Morrill, and Christian Retoré, editors, *Logical Aspects of Computational Linguistics: 4th International Conference*, number 2099 in Lecture Notes in Artificial Intelligence, pages 17–43, Berlin and New York. Springer.

Geoffrey K. Pullum. 1989. Formal linguistics meets the Boojum. *Natural Language & Linguistic Theory*, 7:137–143.

Geoffrey K. Pullum. 2007. The evolution of model-theoretic frameworks in linguistics. In James Rogers and Stephan Kepser, editors, *Model-Theoretic Syntax at 10: ESSLLI 2007 Workshop*, pages 1–10, Trinity College Dublin, Ireland. Association for Logic, Language and Information.

Lorenzo Renzi, Giampaolo Salvi, and Anna Cardinaletti. 2001. *Grande grammatica italiana di consultazione*. Il Mulino, Bologna. 3 volumes.

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

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

Barbara C. Scholz and Geoffrey K. Pullum. 2006. Irrational nativist exuberance. In Robert Stainton, editor, *Contemporary Debates in Cognitive Science*, pages 59–80. Basil Blackwell, Oxford.

Pieter A. M. Seuren. 2004. *Chomsky's Minimalism*. Oxford University Press, Oxford.

Edward P. Stabler, Jr. 1992. Implementing government binding theories. In Robert Levine, editor, *Formal Grammar: Theory and Implementation*, pages 243–289. Oxford University Press, New York.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press, Cambridge, MA.

Moshe Y. Vardi. 1982. The complexity of relational query languages. In *Proceedings of the 14th ACM Symposium on Theory of Computing*, pages 137–146, New York. Association for Computing Machinery.