Linguistic **I**ssues in **L**anguage **T**echnology – **LiLT**
Submitted, October 2011

Romantics and Revolutionaries

What Theoretical and Computational Linguists Need to Know about Each Other*

Mark Steedman

Published by CSLI Publications

*But were Afraid to Ask

Romantics and Revolutionaries

What Theoretical and Computational Linguists Need to Know about Each Other*

Mark Steedman, *University of Edinburgh*

Round the decay
Of that colossal wreck, boundless and bare,
The lone and level sands stretch far away.

Shelley (1818) *Ozymandias*

The philosophers have merely interpreted the world in various ways; the point, however, is to change it.

Marx (1845) *Theses On Feuerbach, XI*

In every field in which progress beckons, romantics and revolutionaries find themselves in an uneasy alliance. The role of the romantics is to define the often unattainable goal. That of the revolutionaries is to advance towards it. Each needs the other, and constantly fears they are forsaken. Sometimes they are right.

Theoretical linguists are the romantics of our field: They seek to understand language for its own sake, intuitively, and on its own terms. Computational linguists are the revolutionaries: They want to make things work better. How are they getting along, these days?

*But were Afraid to Ask

LiLT Volume 6, Issue 11, October 2011.
Romantics and Revolutionaries.

1 That Colossal Wreck

From the 1960s until the mid 1970s, there was almost complete theoretical unanimity among linguists, psycholinguists, and computational linguists. This consensus was founded on some formal results due to Chomsky (1959a,b), showing that "competence" (or what Marr (1977) called the "Theory of the Computation" for natural language) could not be exactly captured using finite-state machines or even context-free grammars. The consensus model of competence was transformational generative grammar, which the linguists developed, the computational linguists found elegant ways of parsing (Woods 1973, Church (this issue); Kay (this issue)), and the psychologists used as a basis for the empirical study of human processing. The consensus model of performance was to pursue a single syntactic analysis, under the guidance of parsing "strategies" amounting to ordering on rules (Fodor et al. (1974)) to deal with the ambiguity in the competence grammar, supplemented by backtracking or "reanalysis" in cases (by assumption, rare) where such strategies led the processor into a blind alley. There was considerable shared interest in rare events like garden-path sentences, crossing dependencies, parasitic gaps, and inverse quantifier scope.

This consensus was immensely productive, leading to important insights into the nature of the processor and the interactions among modules including syntax, semantics, and context, and gave rise to a number of ingenious behavioral and physiological measures of transient processing load, some of which are still in use today (Garrett 2007).

The consensus soon fell apart, however, largely because of early disagreements about the role of semantics in the competence theory (Chomsky 1972), the recognition of the unconstrained power (and consequent weak explanatory force) of structure-dependent transformational rules (Peters and Ritchie 1973), and the realization of the huge amount of syntactic ambiguity inherent in human-scale grammars (and the consequent severity of the problem of search in parsers for those grammars—Martin et al. 1981, Church and Patil 1982). Many formal linguists in the transformationalist mainstream have reacted by disavowing any concern with limiting expressive power. Other, more semantically- or psycholinguistically- oriented linguistic theories, including Praguian Functional Generative Description (FGD, Sgall et al. 1986; Hajičová, this issue), Word/Dependency Grammar (Hudson 1984, 2007), Role and Reference Grammar (RRG, Van Valin 1993), Construction Grammar (Goldberg 1995, 2006, Croft 2001, Cognitive Grammar (Langacker 2008), and Optimality-Theoretic Grammar (Legendre et al. 2001), either allow arbitrarily powerful transformation-like rules

in derivational syntax, or talk in terms of global principles and constraints whose relation to specific formal or computational models is left unspecified. As a result, many contemporary linguistic accounts offer very little that psychologists and computational linguists can use. The psycholinguists themselves have split into two mutually antagonistic groups. One group is politely agnostic about the competence theory, talking either in terms of global constraints and heuristics that are largely independent of any specific theory of grammar (e.g. MacDonald 1994), or in terms of surface parsing strategies (e.g. Fodor 1998, Frazier and Clifton 1996), or a mixture of the two (e.g. Ferreira 2003, 2007) . The other camp vigorously denies the psychological relevance of linguistic theory and the competence-performance distinction itself, seeking explanations at connectionist or neurocomputational levels (e.g. Christiansen and Chater 2001).

The minority of linguists who have retained a concern with limiting expressive power and/or supporting computation have meanwhile been forced to invent their own grammar formalisms, such as Lexical-Functional Grammar (LFG, Bresnan 1982), Generalized Phrase Structure Grammar (GPSG, Gazdar et al. 1985), Lexicalized Tree-Adjoining Grammar (LTAG, Joshi and Schabes 1992), Head-Driven Phrase Structure Grammar (HPSG, Pollard and Sag 1994), and Combinatory Categorial Grammar (CCG, Steedman 2000).

A theoretical linguistics in this fragmented state might seem not to have much to offer in the way of models to computational linguistics. (We ask for bread. They give us empty categories.) And, in fact, with the exceptions mentioned above, computational linguists have mostly reverted to finite-state and context-free approximations to human language, often ignoring linguistically problematic phenomena like relativization and coordination entirely, and without exception depending upon parallel-searching algorithms and machine-learnable probabilistic parsing models to deal with the huge grammars and proliferating numbers of analyses that are needed for robust practical applications on a large scale.

Ask not what linguistics can do for computational linguistics. Ask first what computation can do for linguistics.

2 What Linguistic Theory has to Learn from Computation

The most important fact about language is, of course, that just about every phenomenon—from lexical items, and parts-of-speech, to word-order, constructions, and speech-act types—exhibits a power-law dis-

tribution, according to which a very small proportion of the relevant categories account for the vast majority of observed events, with the remainder constituting a "long tail" of double-exponentially rarer types. The linguists know it as Zipf's Law of word-frequencies (cf. Kay and Church, this issue), which says that if we group words by frequency, and assign each group a rank from most frequent to least, then the words at any rank are roughly twice as frequent as the words at the next rank. (For example, in the Brown corpus (Francis and Kučera 1964), the most frequent word "the" is almost exactly twice as frequent as the second, "of", accounting respectively for around 7% and 3.5% of word-tokens. At the other extreme, about half the words in the corpus occur exactly once.)

Linguists don't often talk about Zipf's law these days. It is the rare events that interest them, because those are the events that can be used to discriminate between alternative theories of the language system. However, this standpoint may encourage a distorted view of the system as a whole. Forgetting Zipf's Law may encourage one to ignore the problem of sparseness in the data one does have.

For example, in the '60s it was possible to doubt the existence of languages with OS basic word order for the transitive clause—that is, with the object preceding the subject (Greenberg 1963:76). However, language genera with all six possible orderings of the elements V(erb), S(ubject) and O(bject) turn out to fall on an almost perfectly power law-shaped frequency distribution (Haspelmath et al. 2005).[1] Linguists just hadn't looked hard enough at the long tail. Given the quite small number of known languages, there must be similar generalizations which the data will always be too sparse to test. (Cinque's 2005 generalization of Greenberg's Universal 20, concerning the possible orders of the elements Det, Num, Adj and N in the nounphrase, which is based on a survey of 350 languages, may well be such a case.)

It is therefore worth noting a couple of properties of computationally practical grammars that might cause us to question whether the grammars proposed by theoretical linguists are yet ready to help computational NLP.

2.1 Real Grammars are Large

Human-scale grammars of the size that is needed to read the newspaper or have a contentful conversation are very large indeed. For example,

[1] It is important to count language genera (e.g. Germanic, Celtic, etc.) rather than languages *per se*. Of course, with only six types to play with, the tail is truncated (although we should note that around 20% of language genera cannot be assigned a single dominant order).

the context-free phrase-structure (CF-PS) base grammar that Collins (2003) induced from the human-annotator-labeled Penn Wall Street Journal Treebank (WSJ, Marcus et al. 1993), using around 50 Part-of-Speech (POS) labels as lexical categories, has around 12,000 PS rule types. Other, more radically lexicalized, grammars for the same corpus typically have fewer (between roughly 1000 and 3000) rules, at the expense of a larger number (between around 500 and 1300) of more informative lexical categories (Hockenmaier and Steedman 2007; cf. Miyao and Tsujii 2005). A number of (usually somewhat smaller, but comparable) grammar-based parsers for linguistic formalisms such as LFG (Riezler et al. 2001) and HPSG (Baldwin et al. 2002) have been hand-built on a large scale.

Even grammars of this size are small by human standards. We know for certain that there are entire construction-types that are unrepresented in the million or so words of labeled WSJ training data, and the datasets associated with the hand-built parsers. Such numbers are to be contrasted with the even smaller rule sets that are typically adduced to cover the fragments addressed in formal linguistic grammars, such as the 26 rules listed in Chomsky 1957 or the 80 or so in Gazdar et al. 1985.[2]

Of course, this discrepancy might just mean that the computational linguists are simply being obtuse, willfully missing the generalizations concerning linguistic structures that the linguists have sought all along. It is certainly the case that the linguists' grammars include some of the most *interesting* rules. However, those who have tried to extend the linguists' general rules to support wide coverage have generally found themselves condemned to listing endless exceptions and lexical idiosyncrasies (see Gross 1978 for a case in point, or Friedman 1971). It seems equally possible natural grammars are structured more like traditional grammars such as Huddleston and Pullum (2002) or Kennedy (1882)—that is, large, lexically and morphologically anchored, and thereby licensed for idiosyncrasy and exception, more like the computationally-oriented lexicalized grammar-formalisms and parsers listed above.

None of this is particularly surprising, in view of the way the attested languages have been shaped historically. Nor, of course, does it call into question the truth at some level of abstraction of the theoretical linguists' generalizations. But it means that wide-coverage grammars induced by computational linguists from data are unlikely to embody those generalizations in other than a statistically approximate sense.

[2] These numbers should be taken as merely indicative of orders of magnitude. Both linguistic and computational grammars include schemas and metarules that make exact counts problematic.

It should also make us ask what the generalizations of theoretical linguistics are *about*. Notions like "subjacency" and "binding condition" seem have more to do with the notions "possible construction" or "possible lexical head" than with that of "possible language." As such, even such basic linguistic notions as constituency and dependency (for both of which the traditional criteria are very weak) may be better thought of as primarily semantic, rather than derivational-syntactic.

If such notions are semantic, and such generalizations concern the universal language of logical form (whose existence and accessibility to children seems to be the *sine qua non* for human language acquisition, and which therefore must be independent of any linguistic word-order), the present emphasis in non-computational linguistic theories of grammar on underlying structural description may be misplaced. What we need instead is a theory of *surface* derivational grammar, directly related to a universal inventory of constructions such as control, binding, relativization, and conjunction, as was originally proposed by Gazdar 1981.

Of course, derivations in such surface grammars must deliver logical forms compositionally. However, this observation is of limited utility, because we know next to nothing about the *natural* language of logical form, except a) that it can be derived easily from the surface grammars of all languages, and b) that it supports inference with an incredible facility, even when quantifiers and negation are involved, for the kinds of sentences that are actually found in corpora.

None of the standard linguistic semantic formalisms yet have either of these properties. The scopal ambiguity of quantifiers and other operators in these formalisms has instead led both linguists and computationalists to entertain proliferating structure-changing operations of (covert) movement (May 1985), type-changing (Hendriks 1993), and tree transformation (Hobbs and Shieber 1987), of exactly the same non-monotone kind that the computational linguists have been so eager to eliminate in syntactic parsing.

If the above is anywhere near the truth, then the theory of syntax itself needs radical overhaul. We need grammars that directly support low-complexity derivation of a considerable variety of surface constructions, and that monotonically determine compositional logical forms in a logical language whose form is to be "ontologically promiscuous" (Hobbs 1985) and determined by convenience for surface-compositional derivation.

This is the reverse of the methodology standard in non-computational linguistic semantics, which is to choose some familiar, ready-made, logical language such as first-order logic, case-frames, or whatever, and

tolerate whatever complexity in syntax it takes to derive appropriate formulæ from sentences. However this alternative approach offers the great promise of allowing easy inference of entailment relations on the basis of surface forms, of the kind implicit in proof-theoretic calculi like the Aristotelian Syllogistic, and recently revived in a different form for tasks like "textual entailment" by MacCartney and Manning (2007).

Such a move would also offer a way of dealing with a second lesson for linguistics from computational NLP:

2.2 Real Grammars are Very Ambiguous

By the mid '90s, when the machines got big enough and fast enough to actually try parsing with realistically-sized grammars, it became clear that the huge degree of lexical and derivational ambiguity found in all languages would swamp any known parsing technique (Charniak 1993:16), even using those low-polynomial time algorithms that had been discovered in the seventies for the CF case (Harrison 1978). This discovery directed attention away from the parsing algorithms themselves, and towards the problem of providing guidance to limit search in such algorithms, via "language models" based on frequency counts of events in labeled data sets like Penn WSJ. Among such parsing models, the most successful are those which use quasi-semantic "head dependencies", as between a verb and the head-noun of a given argument, for example (Magerman 1995, Collins 1997; see Klein and Manning 2003 and Petrov and Klein 2007 for a dissenting view).

These models work as well as they do because they incorporate a very helpful mixture of semantic information related to notions like "subcategorization" and "case-frame", and world knowledge, such as the frequent conjunction of "fish" and "chips". (The reason they don't work *better* than they do is that they arenecessarily built on the basis of laboriously human-labeled datasets like the Penn Treebank, which are known to be far too small for the purpose.)

Such models are at least as important as the grammar in assigning the correct analysis. While linguists (and psycholinguists) tend to think of sentences as usually having at most two analyses, computational linguists know there there are standardly thousands and in some cases millions of syntactically legal analyses of even moderately long sentences, and that some guidance in search is essential.

Experience with such models again suggests that theoretical linguists may need to question some of the assumptions they make concerning the structure of linguistic theory. For example, most treebank grammars for English omit number agreement from the grammar, and show little if any improvement if such a mechanism is added. They can

do so because the parsing model favors dependencies between subject and verb head-words that agree over those that do not. Of course, this strategem lets us down in cases such as *a series of pipes and a pressure-measuring chamber which record the rise and fall of the water surface*, where agreement is crucial to correct attachment of the relative clause (Rimell et al. 2009:815). However, it also has the amiable effect of making the treebank grammars "open-ended", to borrow a term from Mark Johnson elsewhere in this issue. This makes them robust in the face of agreement mismatches like the following, which are quite common in speech and in corpora, and are tolerated by experimental subjects (Quirk et al. 1972, Bock and Miller 1991, Franck et al. 2002):

The cost of the improvements have not yet been estimated.

Another anomaly for which responsibility might better be assigned to the performance parsing model than to grammar proper arises from certain "island conditions", including the Complex NP Constraint of Ross (1967), as Collins (2003:590) points out.

2.3 For a New Theoretical Linguistics

The above discussion suggests that the linguistic theory of grammar needs to be modified in several respects. In particular:

a. Syntactic operations must apply to local, rather than unbounded, domains;
b. Syntactic derivation must be monotonic and surface-compositional to semantic logical form;
c. Semantic logical form must support entailment directly.

Some candidates for such grammars, mostly developed in collaboration with computational linguists, and often realised more or less directly in large-scale parsers, have already been mentioned, including LFG, GPSG, LTAG, HPSG, and CCG. However, none yet exhibits all of these properties, least of all in semantics.

If the linguists can fix these problems, and deliver something a bit more usable in the way of syntactic and semantic theory than they offer right now, then computational linguists will have a lot to learn from them, for they too are in deep trouble, for reasons to be discussed next.

3 What Computational Linguistics has to Learn from Linguistics

Computational linguists also are painfully familiar with power law distributions and Zipf's Law. Such skewed distributions are what make

machine learning for natural language processing difficult, and different from standard machine learning based on Gaussian distributions. Zipf's Law also means that the key to the first 80-90% of performance on any evaluable task lies in capturing the few most frequent event types. Fortunately, known machine learning techniques are very good at learning tasks where the necessary information can be found in frequent events. In fact, in many cases, machine learning can be relied on to decide for itself what categories or event types optimally encode this information, gratifyingly reducing even further the ease and turnaround time per experiment and/or product enhancement.

3.1 Computational Linguistics Without Linguistics?

Because computers have grown exponentially bigger and faster according to Moore's Law, at a rate greater than that at which we can do experiments with machine learning techniques, natural language processing research has been through a period of explosive growth in what might be called "computational linguistics without linguistics", concentrating on the "short head" of most frequent events susceptible to machine learned models, and ignoring the long tail of individually exponentially less frequent events. For example, the dominant factor in improving commercial speech recognition in recent years has been Moore's Law, allowing training and accessing much larger hidden Markov models. Although Moore's law is widely recognized as no longer applying to single processors, the strongly parallelizable nature of training for HMMs and the alternative discriminative classifiers means that this improvement can be expected in principle to continue.

However, there are limits inherent to low-level language modeling which may begin to be felt quite soon. The amount of data that is needed to produce a just-noticeable improvement in performance measures such as word error rate also increases exponentially with the level of performance, even for such basic tasks as HMM speech recognition (Gauvain et al. 1994, Lamel et al. 2002). Moore (2003) shows that extrapolating this increase to estimate the amount of training material that would be required to attain human levels of performance, near zero-percent word error rate, leads to the prediction of datasets of around 1M hours of speech.

Such datasets are impossibly large. Even *collecting* 1M words of speech of adequate quality is a challenge. By comparison, commercial HMM speech recognition seems to be trained on datasets of from one to ten thousand hours. Even then, deriving the model is a huge computation, requiring massive parallelization. Scaling such a process by two or three orders of magnitude seems incredible.

The necessary datasets not only scale exponentially with performance, but also with the complexity of the models themselves. For example, for tasks that are typically addressed using higher-order Markov models, such as statistical machine translation (SMT), the datasets required grow exponentially with the size of the n-grams themselves. Callison-Burch (2007:83) shows that, while the quality of translation increased (not surprisingly) with n in an n-gram SMT model, the amount of training data to learn the first 30% of source-language n-grams present in a testset increased from order 10^4 words for unigrams to order 10^5 for bigrams, 10^6 for trigrams, and 10^7 for tetragrams. Brants et al. (2007:864) show that an n-gram set for n up to 5 continues to grow exponentially with exponentially increasing amounts of training material up to trillions of words. The resulting set included 56% of all pentagrams in a test set for a linear increase of BLEU score per section of the training data.[3]

Brants et al. also show that learning and inference on the basis of such models is a problem in its own right, calling for massive parallelism and sophisticated techniques for compressing models or pushing the search/inference problem off-line.

These numbers strongly suggest that there will never be large enough datasets and computational resources for the currently most successful engineering-based methods to scale to human levels of performance—especially in the case of SMT, where Brants et al. show that available resources of bilingual data have already been exhausted, even for the most populated language pairs.

Zipf's Law means that natural language corpora have a "self-similar" property.

This property of natural language data shows up in parser performance in relation to the size of the training set. For example, Hockenmaier and Steedman (2007:388) show that most of the most frequent category types and rule types in a lexicalized CCG treebank grammar have been encountered in the first 20% of the data—that is, in the first 200K words of a 1M word treebank (see figure 1).[4]

Fong and Berwick (2008:n.14) make a related point concerning the rate of increase in evaluation scores with amount of training data for Collins' parser. However, they are probably wrong to attribute the self-similarity property to the specific nature of the Penn Treebank. It is

[3] The slope of the linear increase was lower for web text than more controlled sources—see Brants et al. figure 5 and note 10.

[4] The different curves are for different low frequency cut-offs f of between 0 and 4 counts of each category/rule type, and are included to show that the growth in category types is not just due to noise.

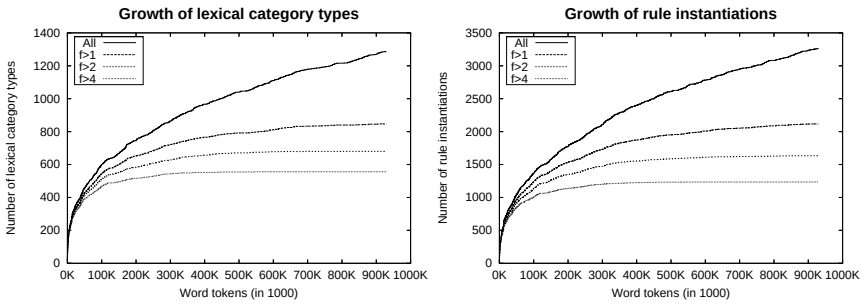

FIGURE 1 Growth of lexical category types and instantiated rule types in a lexicalized treebank grammar (from Hockenmaier and Steedman 2007)

text itself that is self-similar.[5]

This self-similarity property means that what treebank parser induction algorithms are learning in the initial steeply-rising phase (besides the most frequent lexical entries), is the probability distribution over frequent category types such as those of transitive and intransitive verbs, control verbs, and/or the associated rules-types. These are the most general facts about the language. They are the facts that dominate any global evaluation measure such as the widely-used Parseval/Eval-b measure, or dependency recovery rates. To the extent that they are represented in the data, the algorithm learns them very efficiently and quickly.

In the later phase, what the grammar induction algorithm is learning from the data (besides better counts of the most frequent events) is mainly new lexical items and rules of already-seen types, together with their head dependencies, and a few much rarer novel category- and rule-types. Since Zipf's Law tells us that half of them only occur once in the training data, they are unlikely to occur in the (typically, much smaller) test set. (Error analyses suggest that about half of all parser errors in lexicalized grammar parsers arise from missing lexical entries, and the other half arise from the weakness of the head dependency model. Again, one million words of labeled data is not enough to induce a reliable parser for text of this nature.) The global measures are therefore by their nature much less sensitive to these details, which affect the

[5] A small proportion of the text in the Penn WSJ corpus does in fact consist of literal repetition, due to the process of construction of newspaper text. (For example, the WSJ "What's News" section, which is included, repeats the first paragraph of each story that it indexes—see discussion by Webber (2009).) However, this is not the source of the effect noted by Fong and Berwick.

evaluation very little.[6]

To the extent that treebank comprising a greater diversity of genres might be expected to contain a somewhat wider variety of constructions, rather than merely more diverse, but necessarily even sparser, content words, it will give us a better grammar (although we must take care that it is not at the expense of the strength of the model). Questions are a good example of a completely general construction that is underrepresented in the Wall Street Journal—see Clark et al. 2004. However, such a treebank will still be self-similar, yielding learning curves like those in Figure 1, under the iron hand of Zipf's Law.

It follows, of course, that we would need an order of magnitude more labeled data—another ten million or so words of treebank—to make any significant impact on overall performance. Even that would not yield a sufficient approximation to human performance.

No-one is likely to give us an order of magnitude more expert-labeled data. Unless someone works out how to use "crowd-sourcing" for grammatical annotation (as Callison-Burch 2009 has for SMT training data), or finds a way to use user-generated correction data for the same purpose (as Google does to train its spell-checker and speech recognition), it is likely that computational NLP needs some help with theory from linguists.

4 The Way Forward

These numbers should remind us that the informativity of events and event-types concerning the nature of systems as a whole is unrelated to their frequency. (In many *other* spheres of active inquiry, such as physics, the necessity of looking at rare events in order to arrive at a true theory—that it is the exception that proves (or disproves) the rule—is taken for granted.)

This is bad news for the future of natural language processing without linguistics. Of course, it remains possible that purely engineering solutions, such as fifth or sixth-order Markov models learned over terabytes of data, may be able to solve the problem by brute force after all–say by the use of randomized algorithms investigated by Talbot and Osborne 2007. Computational linguists will certainly keep trying. But machine learning is not designed to learn from rare but information-laden events. If we can only get the linguists to step up to the plate, it is they who could tell us what class of computations the models should operate over, what types are involved, and what a natural semantics

[6] Of course, this is as much an indictment of the standard bracket- or dependency-recovery-based global evaluation measures as anything (Rimell et al. 2009).

looks like. Machine learning will then apply, to make such computations practicable (as in Headden et al. 2009) by building the parsing *model*. But it won't of itself give us a *theory*.

One way forward for computational linguistics might therefore be to use our linguists in some less mindless way than paying them not very much to label randomly selected data. Besides providing a basis for building large parsing models, the treeebank labeling exercise of Marcus et al. (1993) (which, as Mark Johnson reminds us elsewhere in this issue, was designed by linguists) was originally motivated in part to produce a more complete list of the most frequent constructions and most important distributional facts than linguists seemed able to provide unaided.[7]

However, now that we have established such basic facts, at least for a few languages, the linguists themselves ought to be able to look at them and tell us what the generalizations of these sets are, to enable us to predict unseen word-category pairs and even unseen lexical types, supporting better smoothing and more graceful degradation of performance in the long tail, in a process of semisupervised learning which has been called "linguist-in-the-loop". This version of "Active Learning" might be expected to be more successful than other versions that use humans merely to propose or correct analyses of arbitrary unlabeled data drawn from the same pool as the original training set. (Clark et al. 2004, Rimell and Clark 2008 present successful active learning of English question constructions in linguist-in-the-loop style. Blunsom and Baldwin 2006 apply a similar approach to lexical acquisition for a hand-built HPSG parser.)

Linguists might also tell us how to generalize our parsers to "low density" languages with little or no labeled data. However, for many languages, this programme would require a much more developed theory of grammatical categories—perhaps in the guise of a probabilistic version of "$\overline{X}$-theory"—than is currently available. It probably requires the development of a theory of the semantic categories that underlie a much larger set of morphosyntactic primitives than are explicit in European languages, including such elements as nominal classifiers, verbal evidentials, discourse particles, and other exotica. However, while insightful descriptions of such categories exist across substantial numbers of diverse languages (Dixon 1994, Aikhenvald 2000, 2004), such accounts remain determinedly unformalized. It seems possible that machine-learning techniques like those proposed by Snyder et al. (2009) might

[7] The treebank was also proposed as a standard test-set for evaluating hand-built parsers for various grammar formalisms.

be applied to this problem—cf. Levin and Bender in this issue.

One of the most useful and interesting applications for such a language of natural logical form would be to annotate a corpus of child-directed utterance with a deeper and more universal set of meaning representations than is currently available (cf. Sagae et al. 2007), as a step towards a more organic and robust form of semantically grounded grammar acquisition pioneered by Zettlemoyer and Collins (2005).

These are long-term projects, and it is not currently clear whether theoretical linguistics will take them on. If not, then computational linguists will just have to do the job unaided. That would be a pity, because they won't do it nearly as well.

5 Conclusion

A spectre is haunting linguistics. Probability is here to stay. All that is solid melts into air, all that is holy is profaned, all changed, changed utterly: a terrible beauty is born.

But computational linguistics still needs syntax and semantics to secure the revolution for the future. If, right now, the romantics won't deliver the vision, the revolutionists will have to change the world as best they can. *We have nothing to lose but our chains.*

Acknowledgments

This work was supported in part by EU IST Cognitive Systems IP grant FP6-2004-IST-4-27657 "Paco-Plus" and EU ERC Advanced Fellowship 249520 GRAMPLUS to the author. Thanks to Prachya Boonkwan, Steve Clark, James Curran, Julia Hockenmaier, Mark Johnson, Phillip Koehn, Tom Kwiatkowski, Kira Mourao, Miles Osborne, Emily Thomforde, Bonnie Webber, and the reviewers for LiLT.

References

Aikhenvald, Alexandra. 2000. *Classifiers: A Typology of Noun Categorization Devices*. Oxford: Oxford University Press.

Aikhenvald, Alexandra. 2004. *Evidentiality*. Oxford: Oxford University Press.

Baldwin, Tim, Emily Bender, Dan Flickinger, Ara Kim, and Stephan Oepen. 2002. Road-testing the English Resource Grammar over the British National Corpus. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 2047–2050.

Bender, Emily. 2011. On achieving and evaluating language independence in NLP. *Linguistic Issues in Language Technology* 6(3).

Blunsom, Phil and Timothy Baldwin. 2006. Multilingual deep lexical acquisition for HPSGs via supertagging. In *Proceedings of the 2006 Conference*

on Empirical Methods in Natural Language Processing, pages 164–171. Sydney, Australia: Association for Computational Linguistics.

Bock, Kathryn and Carol Miller. 1991. Broken agreement. *Cognitive Psychology* 23:45–93.

Brants, Thorsten, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. 2007. Large language models in machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 858–867.

Bresnan, Joan, ed. 1982. *The Mental Representation of Grammatical Relations*. Cambridge, MA: MIT Press.

Callison-Burch, Chris. 2007. *Paraphrasing and Translation*. Ph.D. thesis, University of Edinburgh, Edinburgh.

Callison-Burch, Chris. 2009. Fast, cheap, and creative: Evaluating translation quality using Amazon's Mechanical Turk. In *Proceedings of the International Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 286–295. ACL.

Charniak, Eugene. 1993. *Statistical Language Learning*. Cambridge, MA: MIT Press.

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

Chomsky, Noam. 1959a. On certain formal properties of grammars. *Information and Control* 2:137–167.

Chomsky, Noam. 1959b. Review of B.F. Skinner's *Verbal Behavior*. *Language* 35:26–58.

Chomsky, Noam. 1972. *Studies on semantics in generative grammar*. The Hague: Mouton.

Christiansen, Morten and Nick Chater, eds. 2001. *Connectionist Psycholinguistics*. Westport, CT: Ablex Publishing.

Church, Kenneth. 2011. A pendulum swung too far. *Linguistic Issues in Language Technology* 6(5).

Church, Kenneth and Ramesh Patil. 1982. Coping with syntactic ambiguity. *Computational Linguistics* 8:139–149.

Cinque, Guglielmo. 2005. Deriving Greenberg's universal 20 and its exceptions. *Linguistic Inquiry* 36:315–332.

Clark, Stephen, Mark Steedman, and James R. Curran. 2004. Object-extraction and question-parsing using CCG. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 111–118. Barcelona, Spain.

Collins, Michael. 1997. Three generative lexicalized models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, Madrid*, pages 16–23. San Francisco, CA: Morgan Kaufmann.

Collins, Michael. 2003. Head-driven statistical models for natural language parsing. *Computational Linguistics* 29:589–637.

Croft, William. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford: Oxford University Press.

Dixon, Robert M.W. 1994. *Ergativity*. Cambridge: Cambridge University Press.

Ferreira, Fernanda. 2003. The misinterpretation of noncanonical sentences. *Cognitive Psychology* 47:164–203.

Ferreira, Fernanda. 2007. The 'good enough' approach to language comprehension. *Language and Linguistics Compass* 1:71–83.

Fodor, Jerry, Thomas Bever, and Merrill Garrett. 1974. *The Psychology of Language*. New York: McGraw-Hill.

Fodor, Janet Dean, ed. 1998. *Reanalysis in Sentence Processing*. Dordrecht: Kluwer.

Fong, Sandiway and Robert Berwick. 2008. Treebank parsing and knowledge of language: A cognitive perspective. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pages 539–544. Cognitive Science Society.

Francis, W. Nelson and Henry Kučera. 1964. *Manual of Information to Accompany a Standard Corpus of Present-Day Edited American English for Use with Digital Computers*. Providence, RI: Brown University.

Franck, Julie, Gabriella Vigliocco, and Janet Nicol. 2002. Subject-verb agreement errors in French and English. *Language and Cognitive Processes* 17:371–404.

Frazier, Lyn and Charles Clifton. 1996. *Construal*. Cambridge, MA: MIT Press.

Friedman, Joyce. 1971. *A Computer Model of Transformational Grammar*. New York: Elsevier.

Garrett, Merrill. 2007. Thinking across the boundaries: Psycholinguuistic perspectives. In G. Gaskell, ed., *The Oxford Handbook of Psycholinguistics*, pages 805–820. Oxford: Oxford University Press.

Gauvain, J., Lori Lamel, G. Adda, and M. Adda-Decker. 1994. The LIMSI continuous speech dictation system. In *Proceedings of the Human Language Technology Workshop (HLT)*, pages 319–324. DARPA, ACL.

Gazdar, Gerald. 1981. Unbounded dependencies and coordinate structure. *Linguistic Inquiry* 12:155–184.

Gazdar, Gerald, Ewan Klein, Geoffrey K. Pullum, and Ivan Sag. 1985. *Generalized Phrase Structure Grammar*. Oxford: Blackwell.

Goldberg, Adèle. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago, IL: Chicago University Press.

Goldberg, Adèle. 2006. *Constructions at Work*. Oxford: Oxford University Press.

Greenberg, Joseph. 1963. Some universals of grammar with particular reference to the order of meaningful elements. In J. Greenberg, ed., *Universals of Language*, pages 73–113. Cambridge MA: MIT Press.

Gross, Maurice. 1978. On the failure of generative grammar. *Language* 55:859–885.

Hajičová, Eva. 2011. Computational linguistics without linguistics? View from Prague. *Linguistic Issues in Language Technology* 6(6).

Harrison, Michael. 1978. *Introduction to Formal Language Theory*. Reading MA: Addison-Wesley.

Haspelmath, Martin, Matthew Dryer, David Gil, and Bernard Comrie, eds. 2005. *The World Atlas of Language Structures*. Oxford: Oxford University Press.

Headden, William P. III, Mark Johnson, and David McClosky. 2009. Improving unsupervised dependency parsing with richer contexts and smoothing. In *Proceedings of Human Language Technology: the Annual Conference of the North American Chapter of ACL (HLT:NAACL-09)*, pages 101–109. ACL.

Hendriks, Herman. 1993. *Studied Flexibility: Categories and Types in Syntax and Semantics*. Ph.D. thesis, Universiteit van Amsterdam.

Hobbs, Jerry. 1985. Ontological promiscuity. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, pages 61–69. San Francisco, CA: Morgan Kaufmann.

Hobbs, Jerry and Stuart Shieber. 1987. An algorithm for generating quantifier scopings. *Computational Linguistics* 13:47–63.

Hockenmaier, Julia and Mark Steedman. 2007. CCGbank: a corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics* pages 355–396.

Huddleston, Rodney and Geoffrey K. Pullum. 2002. *Cambridge Grammar of English*. Cambridge: Cambridge University Press.

Hudson, Richard. 1984. *Word Grammar*. Oxford: Blackwell.

Hudson, Richard. 2007. *Language Networks: The New Word Grammar*. Oxford: Oxford University Press.

Johnson, Mark. 2011. How relevant is linguistics to computational linguistics? *Linguistic Issues in Language Technology* 6(7).

Joshi, Aravind and Yves Schabes. 1992. Tree-Adjoining Grammars and lexicalized grammars. In M. Nivat and A. Podelski, eds., *Definability and Recognizability of Sets of Trees*. Princeton, NJ: Elsevier.

Kay, Martin. 2011. Zipf's law and *l'arbitraire du signe*. *Linguistic Issues in Language Technology* 6(8).

Kennedy, Benjamin. 1882. *The Public School Latin Primer*. Longmans, Green and Co. revised ed. 1930.

Klein, Dan and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430.

Lamel, Lori, J-L. Gauvain, and G. Adda. 2002. Unsupervised acoustic model training. In *Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, pages 877–880. IEEE.

Langacker. 2008. *Cognitive Grammar: A Basic Introduction*. Oxford: Oxford University Press.

Legendre, Géraldine, Jane Grimshaw, and Sven Vikner, eds. 2001. *Optimality-Theoretic Syntax*. Cambridge MA: MIT Press.

Levin, Lori. 2011. Three linguistics lessons (for MT researchers). *Linguistic Issues in Language Technology* 6(10).

MacCartney, Bill and Christopher D. Manning. 2007. Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200. Prague: Association for Computational Linguistics.

MacDonald, Maryellen. 1994. The lexical nature of syntactic ambiguity resolution. *Psychological Review* 89:483–506.

Magerman, David. 1995. Statistical decision tree models for parsing. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, Cambridge MA*, pages 276–283. San Francisco, CA: Morgan Kaufmann.

Marcus, Mitch, Beatrice Santorini, and M. Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19:313–330.

Marr, David. 1977. Artificial Intelligence: A personal view. *Artificial Intelligence* 9:37–48.

Martin, William, Kenneth Church, and Ramesh Patil. 1981. Preliminary analysis of a breadth-first parsing algorithm: Theoretical and experimental results. Tech. Rep. 261, MIT, Cambridge, MA. Published as Martin et al. 1987.

Martin, William, Kenneth Church, and Ramesh Patil. 1987. Preliminary analysis of a breadth-first parsing algorithm: Theoretical and experimental results. In L. Bolc, ed., *Natural Language Parsing Systems*, pages 267–328. Berlin: Springer-Verlag. First published as Martin et al. 1981.

May, Robert. 1985. *Logical Form*. Cambridge, MA: MIT Press.

Miyao, Yusuke and Jun'ichi Tsujii. 2005. Probabilistic disambiguation models for wide-coverage HPSG parsing. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 83–90. Morristown, NJ, USA: ACL.

Moore, Roger. 2003. A comparison of the data requirements of automatic speech recognition systems and human listeners. In *Proceedings of Eurospeech Conference*, pages 2582–2585.

Peters, Stanley and Robert Ritchie. 1973. On the generative power of transformational grammars. *Information Science* 6:49–83.

Petrov, Slav and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411. Rochester, New York: Association for Computational Linguistics.

Pollard, Carl and Ivan Sag. 1994. *Head Driven Phrase Structure Grammar*. Stanford, CA: CSLI Publications.

Quirk, Randolph, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1972. *A Grammar of Contemporary English*. Longman.

Riezler, Stefan, Tracy H. King, Ronald M. Kaplan, Richard Crouch, III John T. Maxwell, and Mark Johnson. 2001. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 271–278. Morristown, NJ, USA: ACL.

Rimell, Laura and Stephen Clark. 2008. Adapting a lexicalized-grammar parser to contrasting domains. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 475–484. Association for Computational Linguistics.

Rimell, Laura, Stephen Clark, and Mark Steedman. 2009. Unbounded dependency recovery for parser evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 813–821. Singapore: Association for Computational Linguistics.

Ross, John Robert. 1967. *Constraints on Variables in Syntax*. Ph.D. thesis, MIT. Published as *Infinite Syntax!*, Ablex, Norton, NJ, 1986.

Sagae, Kenji, Eric Davis, Alon Lavie, Brian MacWhinney, and Shuly Wintner. 2007. High accuracy annotation and parsing of CHILDES transcripts. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 25–32. held in conjunction with ACL 2007 Prague, ACL.

Sgall, Petr, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence in its Semantic and Pragmatic Aspects*. Dordrecht: Reidel.

Snyder, Benjamin, Tahira Naseem, and Regina Barzilay. 2009. Unsupervised multilingual grammar induction. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 73–81. Suntec, Singapore: Association for Computational Linguistics.

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

Talbot, David and Miles Osborne. 2007. Smoothed Bloom Filter language models: Tera-scale LMs on the cheap. In *Preceedings of Empirical Methods in Natural Language Processing (EMNLP)*, pages 468–476. ACL.

Van Valin, Robert, ed. 1993. *Advances in Role and Reference Grammar*. Amsterdam: John Benjamins.

Webber, Bonnie. 2009. Discourse—early problems, current successes, future challenges. In *Invited Talk to the 47th Annual Meeting of the Association for Computational Linguistics, Singapore, August*. ACL.

Woods, William. 1973. An experimental parsing system for Transition Network Grammars. In R. Rustin, ed., *Natural Language Processing*, pages 111–154. New York: Algorithmics Press.

Zettlemoyer, Luke and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the 21st Conference on Uncertainty in AI (UAI)*, pages 658–666. ACL.