John Nerbonne

Vaulting Ambition

Abstract Grammar studies have had overly ambitious goals. Computational linguistics, grammatical theory and corpus linguistics increasingly avoid the claims and perhaps even the goals of comprehensiveness. While parse accuracy (really parse and disambiguation accuracy) was a hotly contested field in the 1990s in computational linguistics, progress has stagnated in grammar-based work, even with models that include hundreds of thousands of independent variables. In grammatical theory, transformational generative grammar now limits its interest to "core" processes, while alternatives such as construction grammar seem to foreswear comprehensive studies, at least implicitly. Corpus linguistics, the focus of this volume, has always been more modest, and while it draws on ever more impressive amounts of data (> 10^{10} words/tokens), it also includes a lot of work on grammar differences — a fascinating, but different subject — rather than what constitutes grammar. I'll argue here nonetheless that corpus linguistics has a very valuable additional task in verifying judgments of unacceptability.

1   Introduction: Where's grammar?

There are some confusing differences among the various research communities trying to understand grammar – both those focused on the purely scientific study of grammar, such as grammatical theory, cognitive linguistics, and corpus linguistics but also those whose interest is less direct, namely computational linguistics. The differences concern goals, but also methods and perspectives. My own perspective is likely to be dominated by computation, which is why I'll begin with computational linguistics, but I have also made modest contributions to grammar theory and to corpus linguistics, and I find work in all these traditions valuable and interesting. In spite of this general appreciation of a lot of the work I see, I'm also critical of several aspects of the subfields, especially about what's missing, and impatient about the fairly poor level of interaction among the communities.

Clearly, since computational linguists emphasize a processing or methodological point of view, and corpus linguists a data-oriented one, the pride of place in the discussion might seem to be due to grammatical theory. This paper will nonetheless first present the perspective from computational linguistics because I understand it best and because it provides a way to understand the other two a bit better, particularly the limitations on observation and accuracy.

It would be wonderful to close with a sketch of a perspective that might overcome some of these difficulties in and among the different research traditions. Wonderful, but unrealistic. Since this is a volume on grammar and corpora, I'll focus my – hopefully constructive – criticisms on corpus linguistics.

2 Computational linguistics

There is a good deal of work on syntax in computational linguistics, and some of it adopts the generative paradigm, so that it overlaps strongly with grammatical theory. I'll discuss that work in Sec. 3.2 below so that I can concentrate on some computational linguistics insights that might be better appreciated in the other fields.

There was a close connection between grammatical theory and computational linguistics for a long time, fueled by the goal of building a general-purpose language understanding system and informed by the view that inattention to grammatical distinctions would inevitably disrupt or blunt the process. Theoretical grammar was definitely seen as an authority on the sorts of distinctions that Flickinger et al. (1987), Nerbonne et al. (1993) and Oepen and Flickinger (1998) document in test suites for the purpose of evaluating grammar processing. Others had argued that one might best ignore grammar where speakers often also appeared to, for example the difference between explicitly including the complementizer *that* in sentences such as *Mary knew (that) Sue would leave*, but the late Ivan Sag was clever at showing how such seemingly inconsequential grammatical details could be crucial for interpretation (Flickinger et al. 1987: 4):

(1) Did Jones know the woman (that) was the project director?

Omitting the complementizer in the sentence above changes the meaning completely: with the complementizer in (1) we have a question about whether Jones is acquainted with someone who was the project director, and without it (1) is a question about whether Jones knows a certain fact, namely, that the woman was the project director. Ignoring grammatical details risked what Sag called "pernicious dysfunction".

It is important for our purposes to note that test suites were not taken from corpora, authentically occurring speech or text, but instead consisted of minimal

examples designed to determine whether grammatical processing systems were assigning correct analyses to sentences. It is therefore fair to say that they were inspired more by grammatical theory than by corpus linguistics.

Naturally, it was also understood that grammar constitutes only one module of a complicated system, which also needs to include a lexicon, a parser, semantic and pragmatic interpretation, and an interface to an application, but grammar and parsing were central in research, and the lines to non-computational grammar research were kept close.

2.1 Massive ambiguity

The statistical revolution in computational linguistics has changed this enormously, and not only for practical reasons, as is sometimes assumed, since the triggering insight chronologically followed the wish to explore practical applications. Computational linguists began to consider parsing naturally occurring text – originally, mostly newspaper text – intrigued by application possibilities but also by the scientific challenge of looking beyond what grammar theory had concentrated on. One crucial insight that emerged as naturally occurring data became a focus is that the degree of ambiguity one encounters rises sharply. In fact, the number of analyses assigned by a linguistically well-informed grammar rises exponentially in sentence length. Gertjan van Noord's ALPINO parser and grammar (essentially of a head-driven phrase structure grammar sort (Müller 2016)) for Dutch have been developed over a period of over twenty years, and he and his colleagues and students have been at pains to develop it in a linguistically responsible way, i.e., avoiding Sag's problem as much as possible (Van der Beek, Bouma and Van Noord 2002). ALPINO was originally developed to produce analysis trees (or labeled bracketings), but it was later re-engineered to produce dependency graphs à la dependency grammar (now the common basis of comparison in parsing). As Figure 1 shows, sentences up to about ten words long are not very ambiguous at all in ALPINO, but then things become confusing quickly. This is a common result in grammar-based processing.

A natural reaction of non-computational linguists to this massive ambiguity is often polite skepticism, much like the reaction of non-linguists to linguists' observations of ambiguity. While it would not be feasible to examine all of the analyses assigned by a linguistically informed parser, one can examine many of the options exemplified in the set of analyses and ask whether the options belong in a strict grammar. Abney (1996) does exactly this, examining a myriad of interpretations assigned to the following sentence:
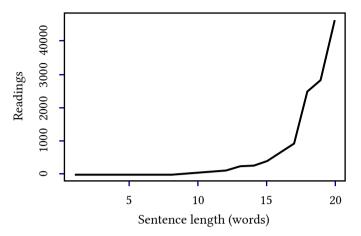
Figure 1: The average number of readings assigned by the linguistically well-informed grammar, ALPINO (Van Noord 2006). Note that 20-word long sentences (the length of an average newspaper sentence) have *on average* over 4,000 readings.

(2)   In a general way such speculation is epistemologically relevant as suggest-
       ing how organisms maturing and evolving in the physical environment we
       know might conceivably end up discoursing of abstract objects as we do
       (Quine, Word and Object)

Abney points out that it is syntactically possible (albeit semantically nonsensi-
cal) to read the sentence in (2) so that *might* is a noun and *objects* a verb. *As we do*
is naturally read as modifying the sentence or verb phrase headed by *discoursing*,
but it could also modify the sentence or verb phrase headed by *objects*. The (com-
putational) linguists that have tried to rule out ambiguities via stricter syntactic
rules or selectional restrictions have generally given up, conceding that language
is used flexibly enough to justify less strict rules.

The solution computational linguists turned to is STATISTICAL DISAMBIGUA-
TION. In these systems, driven by machine learning (ML), one first collects a set of
sentences and the analysis trees (or other analysis annotations) that correspond
best to how each sentence is normally understood – one analysis per sentence
with *no* indication of ambiguity. It would be unfeasible to ask human annotators
to note the entire range of the thousands of analyses that sentences normally
have, and the understood reading is most interesting in applications. The result
is ANNOTATED DATA, which is used to train ML classifiers to choose which anal-
ysis tree best describes sentences that were not used in training. The annotated
data is also used to evaluate how well sentences are parsed, and I will have more
to say on this below. In this case, the data used for evaluation is withheld from
the training data so as not to prejudice the evaluation (Black 1997). While the

annotations and therefore the testing were originally based directly on analysis trees (or, equivalently, on labeled brackets indicating analysis trees), the tree-like annotations in test material have largely given way to dependency labels (Briscoe et al. 2002). The points below do not hinge on the sort of annotation used.

From the point of view of grammatical theory, the move from categorical parsing to systems for parsing *cum* statistical disambiguation represented the loss, or at least the denigrating, of a valuable computational partner. The work of the 1980s involving categorical, non-statistical parsing was aimed at detecting errors in grammatical coverage in order to thereby improve grammars. The errors that were detected often led to discussion with theoretical grammarians about the sorts of distinctions and rules needed in the computational systems. However, the close collaboration was possible because test material consisted of minimal sentences designed to probe the discrimination of syntactic analysis systems. Once computational linguists began to work on real-world data (newspaper texts), the importance of length was brought home forcibly, as sentences in newspapers are about twenty words long on average. Pure grammatical analysis cannot be evaluated against all of the analysis trees produced by sentences working on newspaper text.

We note here that the original ambition of grammatically well-informed CL work had to be curbed – it turns out to be infeasible to check all the consequences of a grammar on all the data that is available.

2.2 Limited parsing accuracy

Computational linguists have agreed since the early 1990s that syntactic analysis systems needed to be evaluated strictly (Black, Lafferty and Roukos 1992). The discipline converged fairly quickly on a scheme borrowed from information retrieval in which both PRECISION and RECALL play a role.

In the precision-recall evaluation scheme, one parses a substantial amount of material – minimally a few hundred sentences, but often thousands – for which the correct analyses have been verified by humans. Let us focus for concreteness first on material that is annotated in labeled brackets. After the material has been parsed automatically by a system that is to be evaluated, one compares the results, constituent by constituent. A constituent the analysis assigns is regarded as correct in case the right label is assigned to the right sequence of words; anything else is incorrect. In particular we keep track of the size of the following sets:

1) the humanly annotated constituents the parser recognizes correctly (true positives, *tp*);

2) the humanly annotated constituents the parser failed to recognize (false negatives, *fn*);
3) the constituents postulated by the parser but not recognized by annotators (false positives, *fp*); and finally
4) the constituents not postulated by the parser and correctly not recognized by annotators (true negatives, *tn*).

Precision is then the fraction of analyses that are correct (recognized by human experts), *tp/(tp+fp)*, and recall is the fraction of the humanly recognized constituents that the parser detects, *tp/(tp+fn)*. The same sort of scheme may be applied to the currently more popular evaluation in terms of labeled dependencies (mentioned above), but I won't discuss this variant separately. We illustrate the analysis types with a tiny example (Figure 2, Table 1).
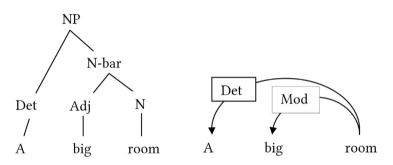
Figure 2: Syntactic analyses in test material against which parsers may be compared. Left, an analysis of constituency, and right, a (labeled) dependency graph. The tree on the left corresponds to the labeled bracketing [NP [Det a] [N-bar [Adj big] [N room]]], and the dependency graph shows a modifier dependency between the head 'room' and adjective 'big', and a determiner dependency between the same head and the determiner 'A'.

Table 1: The precision and recall rates of the analyses in the left column (parser output) based on the syntactic analysis above. I'm ignoring the non-branching nodes such as [Det a].

Parser Output	Prec.	Recall
([Det a, big]), ([NP a,big,room])	0.5	0.5
([NP a,big,room])	1.0	0.5
([N-bar big, room]), ([NP a,big,room])	1.0	1.0

Although I won't discuss separately the dependency-graphs as a basis for evaluation, I can mention that they're based on dependency triples of the sort "A-Det-room", and "big-Mod-room". One checks parser results against so-called "gold-standard" (human-annotated) structures, just as with the constituent-based evaluations.

There is a broad consensus in computational linguistics that these scores reflect parse accuracy faithfully, but those interested in grammar should keep concretely in mind what a (good) score of ninety percent (or *0.9*) means. A sentence *n* words long has *n-1* non-terminal (internal) nodes (if the nodes are binary branching, which is typical), so the average 20-word sentence from a newspaper corpus will have 17 constituents recognized correctly and two incorrectly. In other words, typical sentences will include some misanalysed nodes. (I'm ignoring the fact that errors don't appear uniformly, but instead tend to clump.) We return to this in Sec. 4 (below).

It is further worth mentioning that both precision and recall are measured because researchers may usually increase one at the expense of the other (Manning and Schütze 1999). Finally, there is an accepted way to combine the scores, F_1, a harmonic mean:

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall}$$

Although parsing remains a central topic in computational linguistics, still attracting lots of energy and producing many papers (see the ACL Anthology[1]), the improvements in the last twenty years have not been substantial. F_1 rates for newspaper texts range from 0.89 to 0.92 for a variety of good systems (Ravi, Knight, and Soricut 2008), and no one expects rates to improve a great deal anytime soon.[2] Steedman (2011) argues that we're bound to see a decreasing rate of progress because improvements have occurred (linearly) as data reserves increased in size exponentially, and that further increases of the required size (an order of magnitude) are infeasible. Of course this is frustrating to those who'd hoped and aimed for grammar and parsing systems that assign exactly the right analyses to all the phrases and sentences in a language.

It may be interesting to corpus linguists to know that several researchers have experimented with the grammars implicit in corpora annotated for other purposes. In this sort of experiment, one extracts all the sub-trees of depth one, e.g. the two subtrees in Figure 2, *NP→ Det N-bar* and *N-bar→ Adj N,* together with their frequencies, and then uses these in a statistical parser. Klein and Manning (2003: 424) report an F_1-score of 0.726 using this approach. ML-based approaches are clearly doing a lot more than simply extracting rules and using their frequencies as estimates.[3]

---

1   https://aclweb.org/anthology/
2   Choe and Charniak (2016) have just published a paper enabling F_0 = 0.938 on a standard test set; this would be the best result and one of the biggest improvements in the last twenty years.
3   But see Charniak (1996) for an appraisal of "tree-bank grammars" more optimistic than Klein and Manning's.

2.3 Current work in computational linguistics

For experts in corpus linguistics and grammatical theory, it is worth knowing that substantial improvements have been made in domain adaptation, i.e. adapting a grammar and parser originally developed for newspaper text to domains such as Twitter, technical manuals or email (McClosky, Charniak and Johnson 2010), and in exploiting existing parsers to develop multilingual technologies. The latter effort is known as the "Universal Dependencies" project which proceeds from a cross-linguistically consistent treebank annotation for 70 languages and seeks to stimulate multilingual parser development, cross-lingual learning, and parsing research from a language typology perspective (Nivre et al., 2016).

Finally, we cannot ignore in 2017 the ongoing shift in CL to the deep-learning methods inspired by neural networks (Socher et al. 2011, Schmidhuber 2015). These methods have already advanced to state-of-the-art practical use in some applications, e.g. machine translation (Wu et al. 2016), and they are contributing to modest improvements in parse accuracy, both in dependency parsing (Chen and Manning 2014) and in parsing categorial grammars (Ambati, Deoskar and Steedman 2016). However, improvements are difficult and modest (± 1 %). The developments are nonetheless worth following, since the results are being evaluated against the same sorts of annotated material used earlier. So improvements will accrue to tasks such as search for grammatical theory or corpus linguistics. A familiar complaint about neural-net-based processing would have it that the workings remain a black box, providing little insight as to why performance improves (when it does), but this criticism is being undercut by work showing how to analyze the inner workings of the networks a posteriori (Stoianov, Nerbonne, and Bouma 1998; Kuncoro et al. 2017).

This concludes our discussion of recent computational linguistics with regard to its relevance to the study of grammar. We return to generative models such as HPSG and LFG, which have a long history of collaboration with computational linguistics in Sec. 3.2 below.

3 Grammatical theory

In the first subsection below we consider work in the Chomskyan tradition, which I'll refer to as TRANSFORMATIONAL, in the second, work that strives toward more exactly formalized models (such as categorial grammar), and in the third work in cognitive linguistics.

3.1 The generative legacy

Grammar studies have inherited a well thought-out model from generative grammar including a statement of the task of grammar, namely to devise a completely explicit procedure capable of defining (or enumerating, or producing) all and only the grammatical structures in a language. Since sub-sentential structures occur within sentences, it is sufficient for the procedure to focus on producing – or GENERATING – sentences. And while the description of the task might sound at first pedestrian, even tedious, there is lots of room for theoretical discussion on what sorts of procedures, sub-procedures, modules, communication protocols, and even architectures are best suited for the task. If we date the beginning of Generative Grammar at the publication of Chomsky's *Syntactic Structures* (1957) then the model has inspired sixty years of research by thousands of researchers, and it has undoubtedly contributed enormously to the scientific understanding of grammar, its complexity, and to their implications for human cognition, learning and evolution.[4]

The operative goals of the enterprise have shifted over the decades, however, at least for most generative grammarians, both with respect to the range of phenomena considered and with respect to the level of detail of the description. Concerning the range of phenomena analyzed, attention focused increasingly on CORE GRAMMAR, essentially on constructions involving recursion. This was a conscious curtailment of the original descriptive ambitions of generative grammar (Pinker and Jackendoff 2005).

With respect to precision, early works emphasized the need for attention to concrete detail in generative theory building, in particular for specifying rules exactly and exhaustively. Most papers and even introductory texts from the last half of the generative period (since well before 1987, the mid-point between the publication of *Syntactic Structures* and now) are much less than explicit about the exact forms of rules, feature systems, and even grammar organization. The

---

4   See Roberta D'Alessandro's list "The achievements of generative syntax" for insights due to research in the transformational tradition. http://ling.auf.net/lingbuzz/003392 (viewed Apr. 11, 2017).

feeling of practitioners seems to be that more exact formulations can come later, that they may be abstracting usefully away from irrelevant details, and that the most pressing task currently is to understand the main lines of the overall system, e.g., the sorts of tree structures allowed (whether recursion is asymmetrically limited to the right edges of trees), whether tree transformations might be limited to a single rule or pair of rules ("merge"), or the nature of the information in the various modules and their interfaces, for which the concrete details in a broad coverage grammar are less than crucial. It also has turned out to be very difficult to obtain the abstract insights with any degree of certainty.

This wasn't always so. Early generative grammar was adamant about demanding exact and detailed formulations. For those who didn't witness this period personally, or who have forgotten it, the attitude was sharply different in the early days of this research line. When I worked on implementing grammar-processing systems in the 1980s and early 1990s, I kept a copy of Stockwell, Schachter and Partee (1973) on my desk for English, and Heidolph, Flämig and Motsch (1981) for German. The latter is not formalized, but the former is, and both present an enormous amount of very detailed material from the early generative tradition.[5]

Pullum (1989) complained about the decline in exact formulations, and documents the developments in the 1980s, stimulating an unusual response from Chomsky (1990), who basically defends the decreased level of formalization. Chomsky accepts the potential value of more formalization, but challenges that "the burden of proof is on those who consider the exercise worth undertaking." (p.146). It is interesting to note from our perspective that Chomsky (p.43) also speculates that the generative program as realized in his *The logical structure of linguistic theory* (1955) may have been "premature and far too ambitious".

Chomsky thus took issue with Pullum's complaint, but not with the observation that the methods have shifted in the generative grammar work that his ideas still dominated.[6] The main point for our argument is that detailed and explicit

---

5   *Mais l'honneur a ceux qui le meritent*! Broekhuis and Corver (2016) is the last of a
     seven-volume series on Dutch syntax from a transformational perspective and with
     an emphasis on description. See https://www.meertens.knaw.nl/cms/nl/medewerk-
     ers/143600-hansbr. It would be great to see more such works.

6   In the text I have refrained from stating that I agree with Pullum that there's a prob-
     lem here. In fact, I agree that work from theoretical grammar I discuss in this section
     also suffers in quality because the link to the empirical basis of the claims is often
     vague. For the record, it's every researcher's right to research in the direction he
     or she finds most promising, most interesting, or that she judges she's most likely
     to be able to contribute to. Given Chomsky's enormous contributions, I would not
     presume to criticize his choice in how he conducts his research. The problem Pullum
     sketches arises not because of how a single researcher works but rather when no one

formulations are no longer being produced. Whether that's good or bad, it hinders the cooperation among computational linguists and corpus linguists, many of whom would like to profit from theory.

The shift from a focus on explicit and concrete detail has made the work less interesting to computational linguists working on syntax, virtually all of whom have turned to other research lines when they sought information from linguistics.[7] We turn to some of that work in the next section. The decision to shun concrete formulations has also made the generative work less interesting to all those who'd like to know the details of grammars and not just the main lines, which, as noted above, have also been difficult to pin down. Those interested in the details are not only computational linguists but also second-language instructors, language documentation specialists, language pathologists, and students of language contact and language change, all of whom work with more concrete details. The study of grammar would benefit if the channels of communication were more open.

3.2 Other generative traditions

Several competing frameworks have continued to insist on exact and detailed formulations, among them categorial grammar (CG), head-driven phrase structure grammar (HPSG), lexical-functional grammar (LFG), and tree-adjoining grammar (TAG), and in fact, many researchers in these frameworks rely on computational implementations of their research in order to test its coverage concretely, something which is virtually unknown in contemporary transformational work.

Stefan's Müller work on German grammar in the HPSG framework may serve as an example of the continued energy in this research line, but I hasten to add that there is excellent research in the other frameworks as well (Kaplan et al. 2004; Hockenmaier and Steedman 2002; Abeillé 1988; Kallmeyer and Osswald 2012). To begin, there are several extensive works on German syntax (Müller 1999, 2002, 2010) as well as a large number of detailed studies (see https://hpsg.hu-berlin. de/~stefan/Pub/ for more specialized studies), and Müller is at pains to compare these to work from the transformational community where he can (Müller 2016).

in a research community is taking advantage of the various modern means of quality control – corpus investigation (see below), strict formalization or computational implementation.

7   Eric Wehrli's work constitutes an honorable exception (Wehrli 1988). An assiduous referee pointed me to Abney and Cole (1985), Nelson (1987) and Kuhns (1986), all of whom implemented some aspects of government-binding theory, but as far as I know only Wehrli pursued this research line to the point of broad coverage.

Müller (2015) presents the computational system he uses to test his analyses, undoubtedly one of the reasons for the success his theoretical work has enjoyed. Carl Pollard once argued convincingly that formalization was a good way to detect inconsistencies and carelessness in theorizing, arguing the "PRO-theorem" (Chomsky 1986) was an example of how unformalized ideas could become confused. The "theorem" may be formulated very simply:

1. Every governed anaphor must be bound.
2. Every governed pronominal must be free (non-bound).
3. PRO is an anaphor and a pronominal.

PRO must not be governed.

The logic appears impeccable, but Pollard (1993) argues that it fails because the notion 'governed' is defined differently for anaphors and for pronominals. This means that the formulation effectively hides an equivocation. Rigorous formalization would promote the exposure of the slip, Pollard argues. I think that Pollard was right, but also that computational implementation is yet another, better tool in theory testing. Müller's work, like that of colleagues in categorial grammar, lexical-functional grammar and tree-adjoining grammars, generally exploits this tool.

The question nonetheless arises as to how these frameworks deal with the massive ambiguity and limited accuracy of modern natural language processing, and, of course they are limited in the same way as anyone else. Their systems, too, suffer from finding too many analyses for long sentences and are not more accurate when measured on free text. However, the parsing systems, when developed to support grammar research rather than for applications such as information extraction from newspaper text, can be tested on limited material especially designed for the topic at hand. Following the tradition of Montague Grammar, researchers generally implement FRAGMENTS of the grammar under study, not pretending to completeness (Müller and Lipenkova 2013). It is clear that this strategy risks inconsistencies across fragments, but the advantage is a more rigorous test of the (limited) grammar.

Corpora are properly the focus in Sec. 4 below, but the researchers pursuing alternative, stricter generative models have vigorously exploited corpora (Bildhauer 2011; Abeillé, Clément, and Toussenel 2003). The lines of communication are open between these communities.

3.3 Cognitive linguistics

Cognitive linguistics emphasizes that grammars are largely learned through experience and pleas therefore for a "usage-based" perspective (Ungerer and Schmid 2013), and it is especially popular among researchers in second-language learning (Robinson and Ellis 2008). Among grammatical theories, practitioners see the greatest affinity with CONSTRUCTION GRAMMAR (Goldberg 2006), which rejects the old idea that fairly simple grammars rely on complex lexical information for treatments of compositional phrasal semantics and exceptionality (so-called "strict lexicalism").[8] Instead, each construction (or rule) may be associated with its own peculiar semantics and potentially idiosyncratic syntax. This perspective has stimulated hundreds of papers on various constructions, and has convinced most linguists that the older, strictly lexicalist view was flawed.

Construction grammar has also been stimulating theoretically. Stefanowitsch and Gries (2003) introduced COLLOSTRUCTIONS into the grammarian's tool box, and Boyd and Goldberg (2011) show the need to tease apart the effects of sheer frequency in use (ENTRENCHMENT) from the effects of encountering a form where another is expected (PREEMPTION). Preemption (also known as 'blocking') was well accepted in morphology, while the constructionists extended the applicability of the concept to syntax. There is also very interesting usage-based work on the effects of information density, but which doesn't identify Cognitive Linguistics or construction grammar as its inspiration. This is discussed in the following section.

In general, students of grammar appreciate not only detailed descriptions and analyses of individual constructions but also systematic treatments that attend to how constructions may interact and what sorts of constraints they might underlie. Berkeley Construction Grammar projected a systematic vision (Kay 2002), and Sag (2012) sketches a formalization within the HPSG feature formalism. While therefore more systematic developments of the Construction Grammar ideas exist, a great deal of the work focuses on the description of small numbers of constructions with no formal treatment.

If the usage-based perspective is maintained, one might easily imagine that "use" will turn out to break down into various sorts of uses, perhaps favoring different syntaxes, and that a closer cooperation with psycholinguists' work on sentence processing might develop (Pickering and van Gompel 2011), perhaps in particular in cooperation with models that involve memory crucially (Lewis and Vasishth 2005).

An encouraging development is Dunn's (2017) demonstration that construction grammars can be computationally learned, at least to some extent. While

8   See Müller and Wechsler (2014) for a dissenting view.

the result of the learning is not evaluated as parsers normally are (see above), Dunn does succeed in showing that instances of constructions are detected with measureable reliability. More work in this direction might accelerate collaboration among grammar theorists.

4 Corpus linguistics

Corpus linguistics has blossomed in the past thirty years, harvesting from ever larger corpora[9] and producing inter alia *the* standard grammar of English, built exclusively on corpus results (Huddleston and Pullum 2002). As noted above, there is an increasing amount of work in computational linguistics and in grammatical theory that tries to exploit corpus evidence extensively. The usage-based perspective of construction grammar lends itself immediately to corpus-based research, and this branch of theoretical linguistics has embraced corpora most enthusiastically (Gries and Stefanowitsch 2007), but, as we noted above, work in the alternative generative lines also exploits corpora where it can. Many researchers in grammar, particularly those who worked on languages they do not speak natively, welcome the opportunity to document that their non-native intuitions were not fantasies.[10]

Corpus linguists have also contributed theoretically, e.g. Biber and Conrad's (1999) ideas on lexical bundles, the affinities of words for other words, seen in recurrent sequences, even when they do not constitute idioms. Bresnan et al. (2007) demonstrate that a logistic regression model involving ten independent linguistic variables can predict the dative alternation with astounding accuracy (> 95%). The paper is a statistical tour de force, but we note that it is focused on explaining a variation in grammatical form rather than what constitutes grammaticality.

Another very interesting current research line has been enabled by work in corpus linguistics, even if it does not stem from it directly. Levy and Jaeger (2006), Jaeger (2010) and Linzen and Jaeger (2016) advance the thesis that texts tend to maintain a uniform information density, meaning that they distribute surprises (entropy peaks) fairly evenly. At least one large collection of research projects

---

9 As of Nov. 2016 DeReKo (IDS Mannheim) included over 30 billion word tokens.
10 An autobiographically inspired remark. My work on German impersonal constructions in the early 1980s met with skepticism about the acceptability of some data – even though it was taken in part from published works – because its author was not a native speaker. I was gratified when Hinrichs (2016) showed that the basic patterns are more widely attested. The general point is that corpora contribute in situations such as these as well.

is pursuing these ideas in concert ("Information Density and Linguistic Coding", http://www.sfb1102.uni-saarland.de/). Interestingly, Jaeger (2010: 24–25, 46–49) attributes some inspiration for his work to computational linguistics.

4.1 The status of corpus evidence

To avert misunderstandings, let's start with two points that might be platitudes, the first that corpus evidence is not completely reliable, just as evidence in general is not. Even well-edited newspapers and journals make mistakes, so that it always makes sense to examine crucial data and not simply assume that all that is published is well formed. Second, we acknowledge that the study of grammar cannot rely exclusively on corpus evidence. There are languages for which no corpora are available, and given the Zipfian distribution of words and constructions, rare structures, especially involving combinations of infrequent sub-structures, may simply not be instantiated, even in very large corpora. In addition, some of the methodology of grammatical theory goes well beyond that of checking whether or not a structure exists. I'm thinking of methods that ask whether one form is equivalent to another, whether one statement implies another, etc. We would impoverish grammatical theory if we made no use of that sort of data.

There are also numerous instances in which researchers have adduced corpus evidence that seems acceptable to the relevant judges and that contradicts putative generalizations. To someone who's worked with grammaticality intuitions, this does not seem surprising, as one often tries to construct data of a certain structure where it turns out that the judgments are less than robust. In particular, researchers working on languages other than their own are often leery lest they talk their respondents into preferred judgments or subliminally move them in a favored direction. So it's interesting to note that the more pressing problem is not obtaining an unbiased judgment of acceptability[11] but rather the failure to be imaginative enough to create well-formed examples of the sort under investigation. Intuition-based research is too quick to condemn. So Van der Beek et al. (2002) found examples of extraposition from comparatives in topic, and Meurers and Müller (2009) adduced violations of subjacency as well as doubly filled *Vorfeld* positions. The list may be extended easily, and I'd like to say more about this, given an attitude I've encountered among grammarians that

---

11 The experimental work showing that published grammaticality judgments tend to correlate with non-linguists' judgments (Schütze and Sprouse 2014, and references there) indeed ties intuitive data to a more general population, but note that it cannot address the problem that judges are too quick to condemn. That problem arises when researchers (or naïve judges) fail to consider a wide enough range of possibilities.

corpora cannot provide the "negative evidence" that plays such a central role in Generative Grammar (including both transformational and alternative generative theories of the sort discussed above). While it's true that corpora cannot *provide* negative evidence, they can contribute to *verifying* it. Corpus evidence shows that introspection is "too quick to condemn."

Bresnan (2007) reports on an especially interesting example involving the dative alternation, a construction which has been the subject of dozens, if not hundreds, of studies. The issue in focus was which verbs allow the "alternate" formulation. After searching for examples in corpora, she noted that she occasionally found *more* examples of the supposedly unacceptable sort than of the supposedly preferable one. One example involved the verb *drag*:

(3)   … while Sumomo dragged him a can of beer (in corpora)
      I dragged John the box ('' from published acceptability rating)

I tend to agree with the star in the example, albeit without great conviction, but I also find the dative-shifted example acceptable. The example is especially interesting because the construction is so well studied, but also because it illustrates succinctly one of the problems with negative judgments of acceptability, namely that they are based on concrete examples but are used to justify conclusions of general restrictions. A conclusion about the ungrammaticality of a construction (or rule) based on a single example is always a hasty generalization. Effectively, the unacceptability star in the example above was used to conclude that *drag* allowed only the PP indirect object, but not the double NP construction, a structural restriction. This conclusion is hasty without more extensive sampling, in this case with various combinations of arguments and adjuncts. Grammarians have been guilty of hasty generalizations when it comes to considering the import of unacceptability data.

Adli, García and Kaufmann (2015) document cases where critics of corpus linguistics methods have gone so far as to claim that intuitions and corpus evidence belong to different realms, so that corpus evidence could have no bearing on claims made in transformational research lines. Since the studies I surveyed adduced examples from corpora that pass the intuition test (they seem well formed), the move to dismiss all such evidence is hasty, and, given the "too-quick-to-condemn" problem, would render intuition-based theories methodologically ill-equipped. Finally, if it turned out to be necessary to isolate some theories from corpus evidence, that would *per se* make those theories less comprehensive and less interesting.

4.2 Shortcomings of corpus work today

Let's begin by reviewing known issues that ought to be solved. First, given the size of modern corpora, it seldom makes sense only to report statistical significance and not effect size. Despite the good advice of Baroni and Evert (2008), this still happens.

Second, search mechanisms are still not expressive (flexible) enough. Using XPATH and XQUERY, one can quantify over annotations, and this is useful if one wishes to study, e.g., impersonal constructions. Then one can search for nodes dominating finite verbs but which do not dominate subject complements (in the usual sorts of annotation). The following XPATH expression searches for the Dutch auxiliary verb *worden* that does not have a sister node with the dependency relation 'su' (subject)":[12]

```
//node[@lemma="worden" and not(../node[@rel="su"])]
```

It still finds some clauses with indirect objects that can look like subjects,[13] but it gets the job done. The key is in the interpretation of 'not' as 'there is no' (dependent subject). In TigerSearch one could use *atomic* negation (König and Lezius 2000: 6), which in the example above would amount to searching for nodes with the lemma *worden* and a dependent that is not a subject – which of course isn't the same thing. A search like the latter returns clauses with subjects as long as there are other dependents that are not subjects, e.g. direct or indirect objects. The negation was interpreted atomically, meaning 'with a dependent which is not a subject'. TigerSearch was an excellent tool in its time, but the query languages have improved. The improvements ought to be adopted more widely.

Third, and moving beyond the border of solved problems, we still need search interfaces for non-programmers. The query-by-example tool GrETEL (Augustinus et al. 2013, see too http://gretel.ccl.kuleuven.be) seems to be on the right track, but there may not be a perfect tool with respect to this issue.

However, there are much less tractable issues, too. Given the long tails in linguistic frequency distributions, large corpora are indispensable, and given the large size of contemporary corpora, only automatic annotation is feasible, i.e., annotation produced by taggers and parsers, not human judges. As we noted in the section on computational linguistics above, however, the accuracy of parses seems to have hit a ceiling still under 95% *per constituent*. This means that

12 With special thanks to Gosse Bouma, local XPATH guru.
13 Such as *u wordt verzocht* 'you are asked', but note *mij wordt verzocht* ('I(dat.) am asked'. These can be eliminated if one adds: and not(../node[@rel= "obj2"])).

sensitive work has to be checked manually for its dependence on potentially incorrect annotation. I don't suggest that everything needs to be checked, only samples, but I don't see any other way to immunize studies against annotation errors. It is worth adding here that parse errors and therefore annotations are not random noise, since some constructions, e.g., those involving coordination and ellipsis, are particularly error prone.

5   Conclusions and prospects

The direct communication between computational linguistics and grammatical theory is the focus of a relatively new journal, *Linguistic Theory and Language Technology*,[14] and this is surely a sign of serious interest. In particular, the reflective special issue on "The interaction of linguistics and computational linguistics" (Baldwin and Kordoni 2011) demonstrates that a number of prominent researchers in computational linguistics, including Ken Church, Eva Hajiová, Mark Johnson and Mark Steedman, advocate closer cooperation between the two fields, even if several authors see areas other than grammar as the most promising.

The transformationalists show less interest, and it is mostly critical. Everaert et al. (2015) criticize that computational linguists should focus more on hierarchical rather than sequential structure, but they select engineering-inspired work, which has practical constraints, to illustrate their points, and they ignore computational work on inducing phrase structure models, even in the only area that they discuss: machine translation (p.731).[15] The "deep-learning" models discussed in Sec. 2.3 above explicitly aim at modelling non-sequential phenomena.

My goal in writing this paper was to foster a bit more understanding about and among the various communities studying grammar. As I said above, virtually all students of grammar are taking advantage of corpora, and I also argued that corpora might be the only remedy to the "too-quick-to-condemn" problem. I don't think any special effort is needed to keep corpora in a central role in the study of grammar.

Of course, I won't try to argue that the field has been so ambitious that the allusion to Macbeth in the title is warranted. His admission to "vaulting ambition" closes a detailed, painfully honest reflection on why the actions he planned

---

14   *The Journal of Language Modeling* likewise aims "to help bridge the gap between theoretical linguistics and natural language processing (NLP)." (Przepiórkowski 2012).

15   Pereira (2000) suggests that a great many of the transformationalists' criticisms of work in computational linguistics implicitly assumes that computational linguistics never got beyond the stage of working on n-gram models of words. Everaert et al.'s paper is not reassuring on that score.

might be criticized. The colleagues in various sub-fields have also chosen their goals and methods judiciously.

Acknowledgments

References

Abeillé, Anne. 1988. Parsing French with Tree-adjoining Grammar: Some linguistic accounts. *Proc. 12th International Conf. on Computational Linguistics (COLING),* 7–12.

Abeillé, Anne, Lionel Clément and François Toussenel. 2003. Building a treebank for French. In Anne Abeillé (ed.), *Treebanks. Building and using parsed corpora.* 165–187. Netherlands: Kluwer.

Abney, Steven. 1996. Statistical methods and linguistics. In Judith Klavans and Philip Resnik (eds.), *The balancing act: Combining symbolic and statistical approaches to language*, 1–26. Cambridge: MIT Press.

Abney, Steven and Jennifer Cole. 1985. A government-binding parser. *Proceedings of the 16th Annual Meeting of the North-Eastern Linguistic Society.*

Adli, Aria, Marco García García and Göz Kaufmann. 2015. System and usage: (Never) mind the gap. In Aria Adli, Marco García García and Göz Kaufmann (eds.), *Variation in language: System-and usage-based approaches,* 1–25. Berlin: De Gruyter.

Ambati, Bharat Ram, Tejaswini Deoskar and Mark Steedman. 2016. Shift-Reduce CCG parsing using neural network models. *Proc. Conf. North American Chap., Association for Computational Linguistics: Human Language Technologies*, 447–453. Shroudsburg: ACL.

Augustinus, Liesbet, Vincent Vandeghinste, Ineke Schuurman and Frank Van Eynde. 2013. Example-based treebank querying with GrETEL–now also for spoken Dutch. *Proc. 19th Nordic Conference of Computational Linguistics* (NODALIDA 2013), 423–428. Linköping: Linköping University Electronic Press.

Baldwin, Timothy and Valia Kordoni. 2011. The interaction between Linguistics and Computational Linguistics. *Linguistic Issues in Language Technology* 6(1): 1–6.

Baroni, Marco and Stefan Evert. 2008. Statistical methods for corpus exploitation. In Anke Lüdeling and and Merja Kytö (eds.), *Corpus linguistics. An international handbook,* 777–803. Berlin: Mouton de Gruyter.

Biber, Douglas and Susan Conrad. 1999. Lexical bundles in conversation and academic prose. In Hilde Hasselgård and Signe Oksefjell (eds.), *Out of corpora: Studies in honour of Stig Johansson* (= Language and Computers 26), 181–190. Amsterdam: Rodopi.

Bildhauer, Felix. 2011. Mehrfache Vorfeldbesetzung und Informationsstruktur. Eine Bestandsaufnahme. *Deutsche Sprache* 4: 362–379.

Black, Ezra. 1997. Evaluation of broad-coverage natural-language parsers. In Giovanni Battista Varile et. al (eds.), *Survey of the state of the art in human language technology.* Cambridge: CUP.

Black, Ezra, John Lafferty and Salim Roukos. 1992. Development and evaluation of a broad-coverage probabilistic grammar of English-language computer manuals. *Proc. 30th Annual Meeting, Association for Computational Linguistics,* 185–192. Shroudsburg: ACL.

Boyd, Jeremy and Adele Goldberg. 2011. Learning what not to say: The role of statistical preemption and categorization in a-adjective production. *Language* 87(1): 55–83.

Bresnan, Joan. 2007. Is syntactic knowledge probabilistic? Experiments with the English dative alternation. In Sam Featherston and Wolfgang Sternefeld (eds.), *Roots: Linguistics in search of its evidential base,* 75–96. Berlin: Mouton de Gruyter.

Bresnan, Joan, Anna Cueni, Tatiana Nikitina and R. Harald Baayen. 2007. Predicting the dative alternation. In Gerlof Bouma, Irene Maria Krämer and Joost Zwarts (eds.), *Cognitive foundations of interpretation,* 69–94. Amsterdam: KNAW.

Briscoe, Ted, John Carroll, Jonathan Graham and Ann Copestake. 2002. Relational evaluation schemes. *Proc. Beyond PARSEVAL Workshop, 3rd International Conf. on Language Resources and Evaluation* (LREC), 4–8.

Broekhuis, Hans and Norbert Corver. 2016. *Syntax of Dutch: Verbs and verb phrases. Volume 3.* Amsterdam: Amsterdam University Press.

Charniak, Eugene. 1996. Tree-bank grammars. In *Proceedings of AAAI/IAAI,* 1031–1036.

Chen, Danqi and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. *Empirical Methods in Natural Language Processing,* 740–750. Shroudsburg: ACL.

Choe, Do Kook and Eugene Charniak. 2016. Parsing as language modeling. *Proc. 2016 Conference on Empirical Methods in Natural Language Processing*, 2331–2336.

Chomsky, Noam. 1986. *Knowledge of language: Its nature, origin, and use.* Portsmouth, NH: Greenwood Publishing Group.

Chomsky, Noam. 1990. Topic...comment. On formalization and formal linguistics *Natural Language & Linguistic Theory* 8(1): 143–147.

Correa, Nelson. 1987. An attribute-grammar implementation of Government-Binding theory. *Proc. 25th Ann. Meeting, Association for Computational Linguistics*, 45–51. Shroudsburg: ACL.

Dunn, Jonathan. 2017. Computational learning of construction grammars. *Language and Cognition* 9(2): 254–292. Online (3/2016), DOI: https://doi.org/10.1017/langcog.2016.7

Everaert, Martin, Marinus Huybregts, Noam Chomsky, Robert Berwick and Johann Bolhuis, 2015. Structures, not strings: Linguistics as part of the cognitive sciences. *Trends in cognitive sciences*, 19(12): 729–743.

Flickinger, Dan, John Nerbonne, Ivan Sag and Tom Wasow. 1987. Toward evaluation of NLP systems. Paper distributed at the 1987 ACL Meeting, Stanford. Avail. at www.let.rug.nl/nerbonne/papers/Old-Scans/Toward-Eval-NLP-1987.pdf

Goldberg, Adele E. 2006. *Constructions at work: The nature of generalization in language.* Oxford: Oxford University Press.

Gries, Stefan Thomas and Anatol Stefanowitsch. 2007. *Corpora in cognitive linguistics: Corpus-based approaches to syntax and lexis.* Berlin: Walter de Gruyter,

Heidolph, Karl Erich, Walter Flämig and Wolfgang Motsch. 1981. *Grundzüge einer deutschen Grammatik.* Berlin: Akademie-Verlag.

Hinrichs, Erhard. 2016. Impersonal passives in German: Some corpus evidence. In Martijn Wieling, Martin Kroon, Gertjan van Noord and Gosse Bouma (eds.), *From semantics to dialectometry: Festschrift in honor of John Nerbonne*, 149–158. Milton Keynes: College Publications.

Hockenmaier, Julia and Mark Steedman. 2002. Generative models for statistical parsing with combinatory categorial grammar. *Proc. 40th Ann. Meeting, Association for Computational Linguistics*, 335–342. Shroudsburg: ACL.

Huddleston, Rodney and Geoffrey K. Pullum. 2002. *The Cambridge grammar of the English language.* Cambridge: Cambridge University Press.

Jaeger, T. Florian. 2010. Redundancy and reduction: Speakers manage syntactic information density. *Cognitive Psychology* 61(1): 23–62.

Kallmeyer, Laura and Rainer Osswald. 2012. An analysis of directed motion expressions with lexicalized tree-adjoining grammars and frame

semantics. *International Workshop on Logic, Language, Information, and Computation*, 34–55. Berlin: Springer.

Kaplan, Ronald M., Stefan Riezler, Tracy King, John Maxwell III, Alexander Vasserman and Richard Crouch. 2004. Speed and accuracy in shallow and deep stochastic parsing. Xerox Palo Alto Research Center (PARC).

Kay, Paul. 2002. An informal sketch of a formal architecture for construction grammar. *Grammars* 5(1): 1–19.

Klein, Dan and Chris Manning. 2003. Accurate unlexicalized parsing. In *Proc. 41st Ann. Meeting, Association for Computational Linguistics,* 423–430. Shroudsburg: ACL.

König, Esther, and Wolfgang Lezius. 2000. The TIGER language--A description language for syntax graphs. (see CiteSeer^x, http://citeseerx.ist.psu.edu).

Kuhns, Robert J. 1986. A PROLOG implementation of Government-Binding theory. *Proc. 11th Intrnatn'l Conf. on Computational Linguistics.* (COLING), 546–550.

Kuncoro, Adhiguna, Miguel Ballesteros, Lingpeng Kong, Chris Dyer, Graham Neubig and Noah A. Smith. 2017. What do recurrent neural network grammars learn about syntax?. *Proc. 15th Meeting, European Chapter, Association for Computational Linguistics*, 1249–1258. Shroudsburg: ACL.

Linzen, Tal and T. Florian Jaeger. 2016. Uncertainty and expectation in sentence processing: Evidence from subcategorization distributions. *Cognitive Science* 40(6): 1382–1411.

Lewis, Richard L. and Shravan Vasishth. 2005. An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science* 29(3): 375–419.

Levy, Roger and T. Florian Jaeger. 2007. Speakers optimize information density through syntactic reduction. *Advances in neural information processing systems* 19, 849–856.

Manning, Christopher D. and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. Cambridge: MIT press.

McClosky, David, Eugene Charniak and Mark Johnson. 2010. Automatic domain adaptation for parsing. *Human Language Technologies: The 2010 Ann. Conf. North American Chap., Association for Computational Linguistics*, 28–36. Shroudsburg: ACL.

Meurers, Walt Detmar and Stefan Müller. 2009. Corpora and syntax. In Anke Lüdeling and and Merja Kytö (eds.), *Corpus linguistics. An international handbook,* 920–933. Berlin: Mouton de Gruyter.

Müller, Stefan. 1999. *Deutsche Syntax deklarativ: Head-driven phrase structure grammar* für das Deutsche. Vol. 394. Berlin: De Gruyter.

Müller, Stefan. 2002. *Complex predicates: Verbal complexes, resultative constructions, and particle verbs in German.* Vol. 13. Stanford: CSLI.

Müller, Stefan. 2010. *Grammatiktheorie.* Tübingen: Stauffenburg.

Müller, Stefan. 2015. The CoreGram project: Theoretical linguistics, theory development and verification. *Journal of Language Modelling* 3(1): 21–86.

Müller, Stefan. 2016. *Grammatical theory: From transformational grammar to constraint-based approaches.* Berlin: Language Science Press.

Müller, Stefan and Janna Lipenkova. 2013. Chingram: A TRALE implementation of an HPSG fragment of Mandarin Chinese. *Proceedings of the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 27)*, 240–249.

Müller, Stefan and Stephen Wechsler. 2014. Lexical approaches to argument structure. *Theoretical Linguistics* 40(1–2): 1–76.

Nerbonne, John, Klaus Netter, Kader Diagne, Judith Klein and Ludwig Dickmann. 1993. A diagnostic tool for German syntax. *Machine Translation* 8(1–2): 85–107.

Nivre, Joakim, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. *Proc. 3rd Intrnatn'l Conf. on Language Resources and Evaluation* (LREC), 1659–1666.

Oepen, Stephan and Daniel P. Flickinger. 1998. Towards systematic grammar profiling. Test suite technology 10 years after. *Computer Speech & Language* 12(4): 411–435.

Pereira, Fernando. 2000. Formal grammar and information theory: Together again? *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 358.1769: 1239–1253.

Pickering, Martin and Roger van Gompel. 2011. Sentence parsing. In Matthew Traxler, and Morton Ann Gernsbacher (eds.), *Handbook of psycholinguistics,* 375–409. London: Academic Press (Elsevier).

Pinker, Steven and Ray Jackendoff. 2005. The faculty of language: What's special about it? *Cognition* 95(2): 201–236.

Pollard, Carl. 1993. On formal grammars and empirical linguistics In Andreas Kathol and Michael Bernstein (eds.), *ESCOL '93: Proc. of the 10th Eastern States Conference on Linguistics.* Columbus: Ohio State University.

Przepiórkowski, Adam. 2012. Journal of Language Modelling. *Journal of Language Modelling* 1: 1–4.

Pullum, Geoffrey K. 1989. Topic... comment. Formal linguistics meets the boojum. *Natural Language & Linguistic Theory* 8(1): 137–143.

Ravi, Sujith, Kevin Knight and Radu Soricut. 2008. Automatic prediction of parser accuracy. *Proc. Conf. on Empirical Methods in Natural Language Processing* (EMNLP), 887–896. Shroudsburg: ACL.

Robinson, Peter and Nick C. Ellis (eds.). 2008. *Handbook of Cognitive Linguistics and second language acquisition.* Oxford: Routledge.

Sag, Ivan A. 2012. Sign-based construction grammar: An informal synopsis. In Hans Boas and Ivan Sag (eds.), *Sign-based construction grammar*, 69–202. Stanford: CSLI Publications.

Schmidhuber, Jürgen. 2015. Deep learning in neural networks: An overview. *Neural Networks* 61: 85–117.

Schütze, Carson and Jon Sprouse. 2014. Judgment data. In Podesva, Robert, and Devyani Sharma, (eds.) *Research methods in linguistics*, 27–50. Cambridge: CUP.

Socher, Richard, Cliff Chiung-Yu Lin, Andrew Ng and Chris Manning. 2011. Parsing natural scenes and natural language with recursive neural networks. *Proc. 28th International Conf. on Machine Learning (ICML-11)*, 129–136.

Steedman, Mark. 2011. Romantics and revolutionaries. *Linguistic Issues in Language Technology.* 6(11):1–20.

Stefanowitsch, Anatol and Stefan Gries. 2003. Collostructions: Investigating the interaction of words and constructions. *International Journal of Corpus Linguistics* 8(2):209–243.

Stoianov, Ivelin, John Nerbonne and Huub Bouma. 1998. Modelling the phonotactic structure of natural language words with simple recurrent networks. In Peter-Arno Coppen, Hans van Halteren and Lisanne Teunissen (eds.), *Computational Linguistics in the Netherlands 1997.* 77–95. Amsterdam: Rodopi. (= *Language and Computers: Studies in Practical Linguistics* 25).

Stockwell, Robert P., Paul Schachter and Barbara Hall Partee. 1973. *The major syntactic structures of English.* New York: Holt, Rinehart and Winston.

Ungerer, Friedrich and Hans-Jörg Schmid. 2013 (¹1996). *An introduction to cognitive linguistics.* London: Pearson.

Van der Beek, Leonoor, Gosse Bouma and Gertjan van Noord. 2002. Een brede computationele grammatica voor het Nederlands. *Nederlandse Taalkunde* 7(4): 353–374.

Van Noord, Gertjan. 2006. At last parsing is now operational. In Piet Mertens, Cedrick Fairon, Anne Dister, and Patrick Watrin (eds), *TALN06. Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles*, 20–42.

Wehrli, Eric. 1988. Parsing with a GB-grammar. In Uwe Reyle and Christian Rohrer (eds.), *Natural language parsing and linguistic theories*, 177–201. Netherlands: Springer.

Wu, Yonghui, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144.*