PROBABILISTIC MODELS OF WORD ORDER AND SYNTACTIC DISCONTINUITY

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF LINGUISTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Roger Levy

September 2005

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Christopher Manning
(Principal Adviser)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Daniel Jurafsky

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Ivan Sag

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Thomas Wasow

Approved for the University Committee on Graduate Studies:

Acknowledgements

Toward the later stages of work on this thesis, I have found myself thinking often of one of my favorite Chinese idioms: *pāo zhuān yǐn yù*, literally 'throw brick attract jade.' One of the so-called 36 Stratagems, the idea is that by throwing out something of perhaps limited value, one can induce others to produce something more valuable in return. (Although it originally referred to a deceptive wartime tactic, in modern Chinese the expression has come to be used in cooperative contexts as well.) This idiom captures what for me has been the singularly most rewarding feature of academic research: the collective process by which you share your ideas with other people, who help them grow, mature, and improve. I've been the beneficiary of an awful lot of jade over the past few years, though I hope that people have found some value in my bricks as well.

With respect to this thesis I've collected the largest share of treasure from my outstanding dissertation reading committee—Christopher Manning, Dan Jurafsky, Ivan Sag, and Tom Wasow. Special thanks go to Chris, my advisor, one of those admirable, rare individuals who makes difficult things look natural and easy wherever he ventures. Chris has taught me a tremendous amount about achieving mastery of a field, holding myself to the highest standards, balancing work and family life, and bringing out the best in people by being a supportive, genial advisor and friend. The rest of the committee have been no slouches, either: thanks to Dan for the pinpointed, laser-like suggestions that have transformed my understanding of my own work and its relationship to the field, Ivan for always demanding maximal clarity and accuracy of exposition, and Tom for his open-mindedness and attentive, punctual reading. The rest of the departmental faculty also deserve thanks: it is a testament to how remarkable the Stanford Linguistics department is that I have had substantial intellectual exchanges with every single member of the faculty during my time here.

Some individuals have played particularly important roles in the development of specific chapters. For Chapter 2, I thank John Hale for inspiration and numerous discussions, and Florian Jäger for encouragement and discussion of his own work. Chapter 3 was originally published as joint work with Christopher Manning in the Proceedings of the 2004 meeting of the Association for Computational Linguistics (ACL), and I am grateful to the ACL for permission to reproduce a revised version of that paper here. In addition, I am indebted to Kristina Toutanova for valuable discussion of the maximum-entropy models in Chapter 3, and Mark Johnson, Amit Dubey, and Peter Dienes for sharing code and results from their related work. Correspondence with Jens Michaelis was instrumental in clarifying some of the formal relationships in Chapter 4.

The journey that led me from my first day in graduate school at Stanford to completing a dissertation in computational linguistics was unexpected and filled with surprises. Along the way I have to thank fellow journeyers Luc Baronian, Julia de Caradeuc Bernd, Sarah Benor, Alison Bidwell, Kathryn Campbell-Kibler, Brady Clark, Ashwini Deo, Iván García Álvarez, Veronica Gerassimova, Florian Jäger, Andrew Koontz-Garboden, Jean-Philippe Marcotte, David Oshima, Richard Pocklington, Mary Rose, Devyani Sharma, Julie Sweetland, and pretty much the entire Linguistics department graduate student population. Special thanks go to long-time office-mates Dan Klein and Kristina Toutanova, both of whom I probably would have put on my dissertation committee if I could, to Jim Fox, Johanna Mountain, Arthur Wolf, and Bill Durham for encouraging me to explore my budding interest in linguistics, and to Joan Bresnan and John Rickford for teaching linguistics in such a compelling and interesting way that I couldn't say no.

During my time in graduate school I have received financial support through graduate fellowships from the National Science Foundation and the Stanford Linguistics department, and a research assistantship funded by the Advanced Research and Development Activity's Advanced Question Answering for Intelligence Program. I am grateful to these organizations for their support.

Most important of all: I count myself truly fortunate to have made and kept many dear friends, both before and during graduate school. I am especially grateful to have had several close friends in the Bay Area during my time at Stanford—thank you, Matt, Marcus, and Jesse for music, and Gabriel for tennis, and thank you all for your support and friendship. Finally, the biggest debts of gratitude remain to be acknowledged to my parents, my brothers Jonathan and Benjamin, and my dearest friend and companion of all, Natalia.

Abstract

This thesis takes up the problem of syntactic comprehension, or *parsing*—how an agent (human or machine) with knowledge of a specific language goes about inferring the hierarchical structural relationships underlying a surface string in the language. I take the position that probabilistic models of combining evidential information are cognitively plausible and practically useful for syntactic comprehension. In particular, the thesis applies probablistic methods in investigating the relationship between word order and psycholinguistic models of comprehension; and in the practical problems of accuracy and efficiency in parsing sentences with syntactic discontinuity.

On the psychological side, the thesis proposes a theory of *expectation-based* processing difficulty as a consequence of probabilistic syntactic disambiguation: the ease of processing a word during comprehension is determined primarily by the degree to which that word is expected. I identify a class of syntactic phenomena, associated primarily with verb-final clause order, where the predictions of expectation-based processing diverge most sharply from more established *locality-based* theories of processing difficulty. Using existing probabilistic parsing algorithms and syntactically annotated data sources, I show that the expectation-based theory matches a range of established experimental psycholinguistic results better than locality-based theories. The comparison of probabilistic- and locality-driven processing theories is a crucial area of psycholinguistic research due to its implications for the relationship between linguistic production and comprehension, and more generally for theories of modularity in cognitive science.

The thesis also takes up the problem of probabilistic models for *discontinuous constituency*, when phrases do not consist of continuous substrings of a sentence.

Discontinuity poses a computational challenge in parsing, because it expands the set of possible substructures in a sentence beyond the bound, quadratic in sentence length, on the set of possible continuous constituents. For discontinuous constituency, I investigate the problem of *accuracy* employing discriminative classifiers organized on principles of syntactic theory and used to introduce discontinuous relationships into otherwise strictly context-free phrase structure trees; and the problem of *efficiency* in joint inference over both continuous and discontinuous structures, using probabilistic instantiations of mildly context-sensitive grammatical formalisms and factorizing grammatical generalizations into probabilistic components of *dominance* and *linear order.*

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Rational models of syntactic comprehension

This thesis takes up the problem of syntactic comprehension—how an agent (human or machine) with knowledge of a specific language goes about inferring the hierarchical structural relationships underlying a surface string in the language. This problem has been of major concern from two different perspectives. First, the psychological: what are the major limitations constraining how the human parser goes about constructing a representation for spoken or read input, and how do these limitations shape our behavioral responses to different types of linguistic stimuli? Second, a more applied perspective: how can we devise algorithms and models enabling a computer to efficiently construct and disambiguate among the possible syntactic representations for a given linguistic input?

To understand the similarities and differences between the two perspectives, consider what a model of syntactic comprehension has to account for in the processing of the following sentence:

(1) These are the children that the witch wanted to eat.

This sentence is ambiguous in meaning, because the witch may be concerned about the children getting hungry, or she may be hungry herself. From both the psychological and applied perspectives, we are interested in the kinds of representations that can distinguish between these meanings, and how the appropriate (well-formed) representations can be automatically constructed from the input. We are also interested in knowing the factors involved in choosing the preferred representation in a given context. In the applied perspective, we need to know how to *estimate* the *parameters* of a model that encodes these factors explicitly and can make the disambiguation decision efficiently and accurately.

From the psychological perspective, we might not require ourselves to construct an explicit model that can make specific disambiguation decisions, so long as we have an understanding of the factors that native speakers themselves use in disambiguating. But we have additional concerns. One of these is *incrementality.* Spoken and written language is serial, and in general, people process it from beginning to end. There is extensive evidence that we start processing each sliver of input as soon as we can, and start making inferences about its relationship to earlier input and about what may come later in the sentence. Whatever partial representation a native speaker of English has after reading the word *that* in Example (1), it must not have foreclosed on the option of the children being either eater or eaten. A theory of human sentence processing must account for incremental comprehension.

There is also extensive evidence that some parts of some sentences are generally more difficult to process than some parts of other sentences. If we contrast Example (1) with the following sentence:

(2) These are the children that wanted to eat the witch.

we might well find that native English speakers tend to read the word *wanted* more quickly in (2) than in (1). So another requirement for a theory of human sentence processing is to account for *differential processing difficulty.*

This thesis concerns itself with incrementality, differential difficulty, and disambiguation in *rational* models of syntactic processing, where a "rational" model is informally taken to be something that tries to do as well as it can, as often as it can. Since language use is variable—some structures occur more often than others—it immediately follows that a rational syntactic processor should have *distributional* knowledge of patterns of language use. Distributional knowledge of this form can be transformed into a rational comprehension model using probability theory; the result is a probabilistic model of natural language syntax. Such models are already in widespread use in computational linguistics, precisely because they are good at disambiguation.

The consequences of rational syntactic processing for incrementality and differential difficulty are taken up in Chapter 2, where I argue that a rational, probabilistic disambiguation strategy leads directly to an *expectation-based* theory of incremental processing, where the differential difficulty of different words in different contexts is ascribed to the difference in how likely those words are to occur in those contexts. To determine the predictions of an expectation-based theory for specific psycholinguistic experiments, I use a probabilistic context-free grammar—the simplest probabilistic syntactic model that incorporates hierarchical structure—to estimate the conditional probability of a word in its context. But though probabilistic context-free grammars are adequate in this case as a means of testing a psychological theory, they are not a complete model of natural language syntax. In their simplest incarnation, they do not encode *nonlocal* or *discontinuous* syntactic dependencies, such as the relationship between the verb *eat* and its syntactically distant subject (or perhaps object), the relative pronoun *that*, in Example (1). The probabilistic syntactic models in widespread use in computational linguistics do not answer all the questions we need answered. The remainder of this thesis concerns itself with probabilistic models that incorporate discontinuous dependency, focusing on the problems of how to structure these models, how to estimate parameters for them, and how to disambiguate accurately and efficiently with them.

Section 1.2 of this introductory chapter is a brief reference defining probabilistic context-free grammars and a few of their crucial properties. Section 1.3 provides background information and argumentation for syntactic discontinuity from a theoretical linguistic perspective. Section 1.4 outlines the possible approaches to parsing discontinuous structures. Finally, Section 1.5 provides a more detailed overview of the remainder of the thesis.

1.2 Probabilistic Context-Free Grammars

By far the probabilistic grammatical model with widest currency is the *probabilistic context-free grammar* (PCFG). Since CFGs and PCFGs serve as the touchstone to work in every part of this thesis, I give a short definition of PCFGs here, along with several of their crucial properties.

A probabilistic context-free grammar consists of a tuple (N, V, S, R, P) such that:

- N is a finite set of non-terminal symbols;
- V is a finite set of terminal symbols;
- S is the start symbol;
- R is a finite set of rules of the form $X \to \alpha$ where $X \in N$ and α is a sequence of symbols drawn from $N \cup V$;
- P is a mapping from R into probabilities, such that for each $X \in N$,

$$\sum_{[X \to \alpha] \in R} P(X \to \alpha) = 1$$

A PCFG *derivation* is the recursive expansion of non-terminal symbols in a string by rules in R, starting with S, and a *derivation tree* T is the history of those rule applications. The probability $P(T)$ of a derivation tree is simply the product of the probabilities of each rule application.

PCFGs enjoy important properties with respect to estimation and parsing. Chi and Geman (1998) showed that maximum-likelihood estimation of a PCFG, both supervised and unsupervised, is guaranteed to produce a *proper* probability distribution.[1] This property is important in Chapter 4. Established bottom-up (Kasami,

[1] A proper probability distribution is one that does not "lose" probability mass; more formally, if T is the set of possible (finite) trees for a PCFG, then $\sum_T P(T) = 1$.

1965; Younger, 1967) and top-down (Earley, 1970) algorithms for CFG recognition can easily be extended to probabilistic parsing of PCFGs, permitting optimal (highest-probability) parse selection in time cubic and space quadratic in the length of the string. Furthermore, algorithms have been developed that allow us to efficiently calculate the probability of *strings* and *prefixes* with respect to a PCFG (Jelinek and Lafferty, 1991; Stolcke, 1995).[2] The computation of prefix probabilities with these algorithms plays a crucial role in Chapter 2.

1.3 Syntactic discontinuity from a categorical perspective

This section provides a motivation from the perspective of theoretical syntax for the investigation of discontinuous-constituency in parsing taken up in Chapters 3 and 4 of this thesis. Although exceedingly few constructions have been shown to exceed the capability of context-free grammars to provide weak structural descriptions (Culy, 1985; Shieber, 1985), resorting to discontinuous constituency can simplify the strong structural description of a much wider range of natural language syntax. Section 1.3.1 argues this case from the perspective of constituency and semantic composition. Section 1.3.2 ties together constituency- and dependency-based perspectives on discontinuity and nonlocality.

1.3.1 Discontinuous constituency

The discussion of discontinuity in the syntactic literature generally seems to take basic context-free grammatical description as a standard of comparison. The movement phenomena first analyzed by Chomsky as evidence for the trans-CF nature of natural language (Chomsky, 1956) come out as discontinuity phenomena of one type or another in the typology of discontinuity I introduce here. The pretheoretical diagnostic

[2] The probability of a string is the sum of the probabilities of all the trees that can dominate that string; the probability of a prefix is the sum of the probabilities of all the trees that dominate strings beginning with that prefix.

of discontinuity is based on the notion of the *phrase*; a phrase is discontinuous if it does not consist of a continuous string fragment. Somewhat more formally, we can situate the notion of discontinuity against the move in generative syntax to identify phrases with nodes of a context-free tree. This move has a semantic motivation as well: a large number of compositional semantic relationships in natural language can elegantly be characterized as head-sister relationships between nodes in a headed context-free tree. Discontinuity can then be defined as deviation from this standard: any compositional relationship between nodes in a context-free tree that are not in a head-sister relationship. There are two major means of formally accounting for discontinuity. First, an explicit connection can be made between a non-head/sister node pair that encodes their relationship. The slash-passing approaches of GPSG (Gazdar et al., 1985) and the functional annotations of LFG (Kaplan and Bresnan, 1982) are instances of this approach. The less widely adopted approach is to characterize the apparent discontinuity as a head-sister tree relation, but to relax the assumption that tree nodes dominate continuous string fragments. In the remainder of this section, I hope to show that this latter approach yields fruitful results.

To motivate this approach, I briefly discuss extraposition from NP, an example familiar from English of what can be viewed as syntactic discontinuity. Consider the (semantically equivalent readings of the) following pair:

(3) a. I *met* **a woman who was wearing a hat** *yesterday.*
b. I *met* **a woman** *yesterday* **who was wearing a hat**.

From a strict semantic point of view, elements belonging to the NP object and the VP have been shuffled together: *yesterday* is clearly semantically associated with *met*, and *who was wearing a hat* with *a man.* This can happen with NP subjects, too; note the contrast with the depictive construction, where there is a semantic connection of cotemporality between the verb and the depictive:

(4) a. A woman is coming who was once a track star.
b. A woman is coming alone. [cannot mean that she was alone in the past but not now]

Furthermore, the semantic contribution of the extraposed RC is clearly below the determiner, whereas with the depictive it is not. In (5) below, the extraposed relative clauses are intersective semantic modifications of the restrictor of the determiner ((5) and (5a)), whereas the depictive cannot modify inside the scope of the definite determiner ((5c)).[3]

(5) a. **No Democrat** will be elected president in 2004 **who is not strong on defense** (Senator Joseph Lieberman, May 2003, quoted in *The New York Times*)

b. **The Democrat** has the best chance of being elected president in 2004 **who is strongest on defense**.

c. **The Democrat** has the best chance of being elected president in 2004 *strong on defense.*

Prepositional phrases and appositives are also extraposable:

(6) . . . **no mechanism** exists **for finding a middle ground**. (WSJ)

(7) Imports of the types of watches that will now be eligible for duty-free treatment totaled **about $37 million** in 1998, **a relatively small share of the $1.5 billion in U.S. watch imports that year**. (WSJ)

A syntactically discontinuous representation of these extraposed NP-internal elements is attractive in that the semantics becomes simple and entirely compositional. The relative clause has the exact same semantic import in (3a) as in (3b), so if the underlying syntax (termed the *tectogrammatical structure* by Curry 1961) is the same, the analysis is simpler.

The simplicity of this approach is doubly attractive because this kind of semantic

[3] Example (5) is provided as a naturally-occurring example to motivate the plausibility of (5b). The depictive with negative universal determiner:

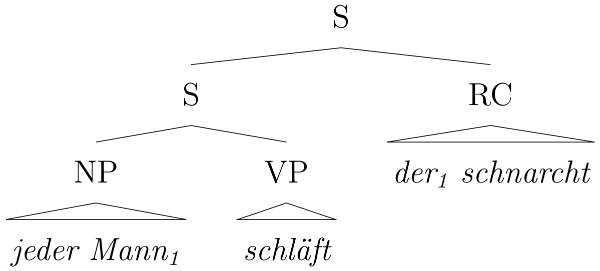
No Democrat will be elected president in 2004 *weak on defense*

does not offer a similarly clear contrast to the extraposed relative clause case, due to the interaction between negative quantification and the stage-level meaning of the depictive.

relationship often has other reflexes at the syntax-semantics interface. The result of this is that an analysis of discontinuity phenomena that maintains the string-continuity requirement of phrase-structure constituents can easily become highly baroque and unintuitive when refined against a detailed body of data. I illustrate this point drawing on a recent, detailed analysis of German relative clause extraposition by Kiss (2005). Kiss, in contrast with other recent accounts of German word order (Reape, 1994; Kathol and Pollard, 1995; Müller, 1999; Kathol, 2000), treats German extraposed relative clauses in HPSG using an exclusively context-free syntactic backbone, with the semantic relation between the extraposed element and the NP treated as a specialized form of anaphora. As an example of extraposition, Kiss gives the following syntactic structure:

(8) a. . . . weil **jeder Mann** schläft, **der schnarcht**
. . . because every man sleeps who snores

b.



Kiss's main argument against a discontinuous treatment of RC extraction is based on asymmetries in binding possibilities among dependents of a single verb. Following Büring and Hartmann (1996); Haider (1996, 1997), Kiss notes that a quantified argument in a VP may only bind a pronoun inside a relative clause of another argument of the same VP if the quantified argument linearly precedes the argument hosting the relative clause:

(9) a. Wir haben niemandem$_i$ die Frage gestellt, auf die er$_i$ sich
we have no-one.DAT the question put on which he REFL
vorbereitet hatte.
prepared had
We asked no one$_i$ the question that they$_i$ had prepared for.

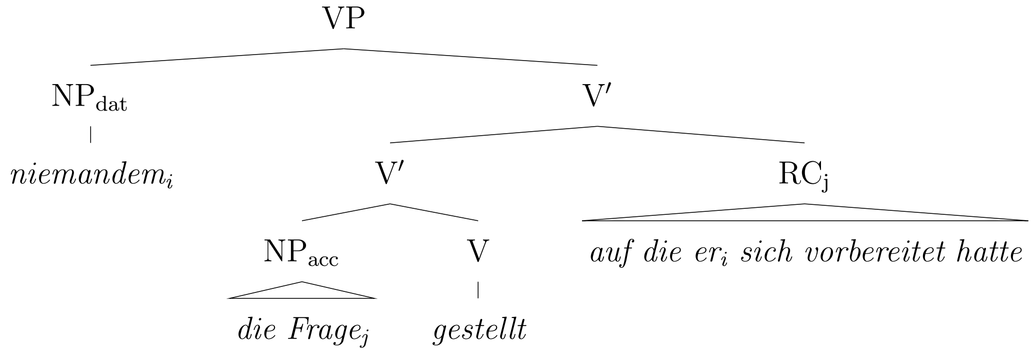
b.

*Wir	haben	die	Frage	niemandem$_\mathrm{i}$	gestellt,	auf	die	er$_\mathrm{i}$	sich
we	have	the	question	no-one.DAT	put	on	which	he	REFL

vorbereitet	hatte.
prepared	had

Kiss argues that this difference can be captured by assuming that verbal arguments combine with a final verb in binary branching trees, and that extraposed relative clauses right-adjoin to the verbal projection at arbitrary positions relative to the order of argument combination. In (9a), the syntactic tree looks as follows:

(10)



Kiss also stipulates that the anaphoric properties of extraposed RCs are such that they must configurationally command their nominal antecedents. He further assumes that ordinary quantifier-pronoun binding possibilities are also determined configurationally via c-command; the bindings in (10) are therefore licensed by the fact that *niemandem* c-commands the postposed relative clause, and the relative clause c-commands its antecedent NP. For (9b), in contrast, it is impossible for the RC in to be positioned such that it both commands its antecedent and is commanded by *niemandem*, explaining the grammaticality contrast in (9). If on the other hand, Kiss argues, German word order is purely a property of surface positions of strings, (as he takes discontinuity-based theories to require), it is impossible to explain the contrast in (9).

Kiss's analysis has both theoretical and empirical liabilities stemming from his string-continuous treatment of extraposition. Theoretically, by using totally different syntactic rules to license *in situ* and extraposed RCs, he complicates their semantic description. The anaphoric conditions required by Kiss for the RC's modificational force find no parallel in any other type of anaphora or construction. Trying to extend the behavior of conventional anaphora to that of extraposed elements leads to empirical problems, as we will see momentarily. Second, interleaving adjunction with complementation at different levels of a single syntactic projection, as Kiss is forced to do, is quite unorthodox.

Empirical shortcomings involve anaphora and postposed free relative clauses. If right-extraposed elements are associated with their NPs via anaphora, then they should follow the pattern of general anaphora of being able to bind an NP in a nominal or verbal conjunct, as holds for English:

(11) a. Regarding John$_i$, I don't like him$_i$ *or* Clyde.
b. Regarding John$_i$, I distrust him$_i$ and loathe Clyde.

But this is not possible with extraposed relative clauses in German, even if gender marking eliminates any possible ambiguity in antecedence:

(12) a. *Er hat mir einen Mann$_i$ und eine
He has me.DAT a.ACC.MASC man$_i$.ACC.MASC and a.ACC.FEM
Frau vorgestellt, der$_i$ bei IBM arbeitet.
woman.FEM introduced, who.MASC by IBM works
(He introduced me to a man who works for IBM and to a woman.)

b. *Er hat einen Mann$_i$ kritisiert und eine
He has a.ACC.MASC man$_i$.ACC.MASC criticized and a.ACC.FEM
Frau gelobt, der$_i$ bei IBM arbeitet.
woman.ACC.FEM praised, who.MASC by IBM works
(He criticized a man who works for IBM and praised a woman.)

More critically, it is possible to postpose free relative clause arguments in German, in which case no associated overt element remains to the left of the nonfinite verb:

(13) Er möchte nie essen, was ich ihm koche.
He wants never to_eat, what I him.DAT cook
'He never wants to eat what I cook for him.'

Under Kiss's analysis of ordinary extraposed RCs, there are three potential ways to account for extraposed free RCs (EFRCs): free RCs could be taken to have the distribution of NP arguments; they could have the distribution of ordinary RCs; or they could have their own unique distribution. Under the first alternative, the grammaticality of EFRCs should correlate with the grammaticality of postposed NP arguments. But judgements on postposing of full NP arguments range from highly unnatural to outright bad:

(14) */?? Er möchte nie essen, das alte, faule Gemüse das ich ihm
He wants never to_eat, the old, rotten vegetables that I him.DAT
koche.[4]
cook.
(He never wants to eat the old, rotten vegetables that I cook for him.)

The second alternative, generating EFRCs via the adjunction rule generating extraposed RCs in general, would require the EFRC to be identified with an NP argument via anaphora. This runs into the problem that Kiss's analysis requires the presence of an antecedent NP constituent with which to (anaphorically) associate extraposed RCs. Note that the antecedent cannot be taken to be some lexical feature of the verb; it must crucially be a constituent sister to a verbal projection in order for Kiss to get the quantifier binding facts right in (9). The only ways out for Kiss are to introduce into an analysis otherwise free of empty categories a null category heading an in-situ nominal argument projection, just for purposes of generating free relatives; or, as laid out previously, to create entirely new rule for free relative anaphora.

In a formalism allowing the explicit expression of discontinuous constituents, on the other hand, capturing the contrast presented by (9) is in fact quite simple as long as semantic constraints are allowed to be sensitive to *some* linear ordering effects. In particular, there is no difference in meaning, including quantificational possibilities, between two sentences differing only in whether an RC is *in situ* or extraposed:

[4] The speaker who rated (14) as highly awkward rather than outright bad noted the postposed NP's heaviness as being its only saving grace. Any rule permitting postposition of heavy NPs would therefore be different from that licensing EFRCs, since examples such as (13) clearly do not rely on heaviness for their acceptability. It is also worth noting that German does not seem to allow right extraposition of full NP arguments without leaving behind some kind of preverbal marker, such as an expletive pronoun.

(15) (c.f. (9))

a. Wir haben niemandem$_i$ die Frage, auf die er$_i$ sich vorbereitet
we have no-one.DAT the question on which he REFL prepared
hatte, gestellt.
had put
We asked no one$_i$ the question that they$_i$ had prepared for.

b. *Wir haben die Frage, auf die er$_i$ sich vorbereitet hatte,
we have the question on which he REFL prepared had,
niemandem$_i$ gestellt.
no-one.DAT put

The quantifier binding constraint can be simply expressed as a precedence relation (on either the tectogrammatical or phenogrammatical level) of the type made available in frameworks such as those proposed by Goetz and Penn (1997) or Kathol (2000). The data here underdetermine the exact nature of precedence required; it would be satisfactory to state that an argument A of a verb can bind material in another argument B of the verb if and only if:

(16) a. every terminal dominated by A linearly precedes every terminal dominated by B; or

b. every terminal dominated by the head daughter of A dominates every terminal dominated by the head daughter of B; or

c. the (recursively determined) *head* terminal of A linearly precedes the head of B; or

d. A commands B in the tectogrammatical structure, if arguments combine binarily with their governing verb.

Following typical linearization-based analyses, the position of the extraposed relative clause is accounted for by (i) positing that the VP is a single *linearization domain*; and (ii) that an NP → NP RC rewrite can allow either *isolation* (also called *compaction*) of the mother, requiring string continuity of material the mother dominates, or *liberation* of the RC from the NP, allowing the RC to resolve its domain-final positioning requirement at the level of the VP rather than its NP mother (see also Zwicky 1986). The former choice corresponds to an *in situ* relative clause; the latter to an extraposed RC. The following gives the rules required in the framework of Goetz and Penn (1997) to generate the proper word-order plus binding facts for Kiss's data.[5]

(17)

$$1. y_0^{\text{s}} \rightarrow y_1^{\text{np}} y_2^{\text{vp}}$$

$$2. y_0^{\text{vp}} \rightarrow y_1^{\text{vfin}} y_2^{\text{vp}} \wedge \rightarrow^2 y_1$$

$$3. y_0^{\text{vp}} \rightarrow y_1^{\text{vnfin}} y_2^{\text{np}} y_3^{\text{np}} \wedge ((\exists y^{\text{s'}} \in y_0 \wedge \leftarrow^2 y_1) \wedge \leftarrow^1)$$

$$4. y_0^{\text{np}} \rightarrow y_1^{\text{det}} y_2^{\text{n'}} \wedge < y_2 >: x \wedge < y_0 >: x \wedge y_1 < y_2$$

$$5. y_0^{\text{n'}} \rightarrow (y_1^{\text{n'}}) y_2^{\text{s'}} \wedge \leftarrow^1 y_2 \wedge (< y_0 > \vee < y_0 >: y_2)$$

$$6. sis(y_0, y_1) \wedge \exists y \in y_1.binds(y_0, y) \Rightarrow y_0 < y_1$$

The crucial work done in this set of rules is that by rule 5, an S′ daughter of NP must be final in its domain, but it may escape the isolation of the N′'s linearization domain; if it does, it likewise escapes the NP's domain (rule 4) and enters the linearization domain of VP and S; by rule 3, a non-finite verb must ordinarily be final in its domain but it can be second-to-last if an S′ is also present in the domain.[6] Rule 6 simply expresses the linear precedence relation required when a node binds a pronoun inside its sister; this single statement successfully captures the quantifier binding generalizations of both (9) and (15). The tectogrammatical structure of N′ modification by relative clauses is identical in the *in situ* and extraposed cases, so no special anaphoric rule is required to connect extraposed RCs to the NPs they modify. Finally, extraposed free relative clauses are generated simply via the optionality of the N′ daughter in Rule 5; in the case of a N′ →S′ rewrite, the S′ daughter may stay

[5] The rules given here differ somewhat from those given by Goetz and Penn for German extraposed RCs; I find the rules given here more intuitive. Symbols are to be interpreted as follows: y^{X} is a variable y with syntactic category X; $\rightarrow^{\text{n}} y$ ($\leftarrow^{\text{n}} y$) requires that y is the n^{th} child from the right (resp. left) in its domain; $< y >$ $(: x)$ states that y is an *isolation domain* – that is, material outside of y may not appear between elements of y at the level of surface order (and optionally that x *escapes* from the isolation domain of y); $x < y$ requires that x exhaustively precedes y; that is, all elements of x linearly precede all elements of y; the parenthesis around $y_1^{\text{n'}}$ in rule 4 are shorthand for optional expression of y_1 in the rewrite rule; and *sis* and *binds* are the familiar relations of sisterhood and binding.

[6] This aspect of the analysis follows Goetz and Penn closely; it could be made even more elegant by the incorporation of Optimality Theory-style violable, ranked edge constraints, under which both the non-finite verb and the S′ would be constrained to be final in their domain, only the S′'s constraint would be higher-ranked.

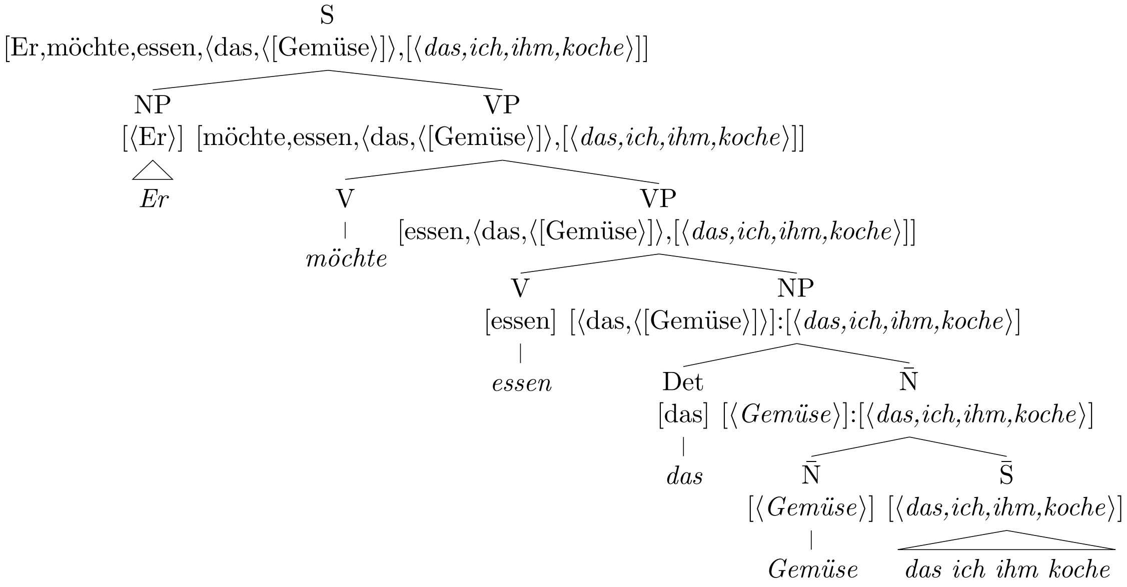

S
[Er,möchte,essen,⟨das,⟨[Gemüse⟩]⟩,[⟨das,ich,ihm,koche⟩]]
NP
[⟨Er⟩]
Er
VP
[möchte,essen,⟨das,⟨[Gemüse⟩]⟩,[⟨das,ich,ihm,koche⟩]]
V
möchte
VP
[essen,⟨das,⟨[Gemüse⟩]⟩,[⟨das,ich,ihm,koche⟩]]
V
[essen]
essen
NP
[⟨das,⟨[Gemüse⟩]⟩]:[⟨das,ich,ihm,koche⟩]
Det
[das]
das
N̄
[⟨Gemüse⟩]:[⟨das,ich,ihm,koche⟩]
N̄
[⟨Gemüse⟩]
Gemüse
S̄
[⟨das,ich,ihm,koche⟩]
das ich ihm koche

Figure 1.1: An extraposed free relative clause in a simple LSL grammar

inside the mother's linearization domain to yield an *in situ* free relative, or it may escape into the VP/S domain, getting postposed and leaving an empty N′ domain behind.

Note that this means that the surface positioning of verbal arguments is both a syntactic and semantic phenomenon; the surface position of the relative clause, on the other hand, is purely a syntactic phenomenon as it apparently has no semantic consequences. This simple distinction captures Kiss's data much more clearly than does his own analysis.

Finally, direct representations of discontinuous constituency also hold prospects for improved modeling of speech, because discontinuity seems to be more widespread in spoken language than in written language. I give three brief pieces of informal evidence to support this claim. The first piece of evidence involves the token frequency of relative clause extrapositions in the (spoken) Switchboard versus (written) Wall Street Journal sections of the Penn Treebank. A simple tree search turns up 535 extraposed relative and complement clauses from Switchboard and only 299 from Wall Street Journal, despite the fact that there are similar numbers of relative and complement clauses in the two sources (30,675 in the Switchboard and 29,129 in the WSJ). Although a stronger conclusion would require more detailed investigation, initial indications suggest that extraposition is more common in speech.

The second piece of evidence involves the types of relative clause extraposition found. Relative clause extraposition in English can occur from within possessors, but it only seems to occur in speech. A tentative search of the British National Corpus for such examples, using the template "⟨possessive marker⟩ ⟨noun⟩ who", turned up only three examples of RC extraposition from possessors, all from the spoken part of the corpus:

(18) a. . . . I remember when projects used to come into the group from on high, they used to filter through the organization, until they landed on **somebody**'s desk **who was actually supposed to carry out the work**.

b. I went over there and I stepped into this **guy**'s shoes **who had really a difficult fourth year class**.

c. . . . has, er, a process that we've got ⟨unclear⟩ has allayed **people**'s fears **who've been used for those residential home agreements**.

The BNC consists of only 10% spoken data, so it is quite unlikely that these extrapositions would all be from speech if they occurred equally often in writing.

The third piece of evidence comes from observations in the Switchboard parsed corpus. There are many cases of apparent rightward conjunct extraposition, which would be a violation of the Coordinate Structure Constraint's proscription on conjunct movement (Ross, 1967):

(19) a. so, you know, well a lot of the stuff you hear coming **from South Africa** now, **and from West Africa**, that's considered world music. . . (SWBD)

b. being an engineer in facilities I do read **a lot of. . . building magazines, and, and, and plant engineering magazines**, and read up on different ways to do things, **and energy management type of magazines** (SWBD)

c. or it feels **cool** compared to yesterday, **but very pleasant** (SWBD)

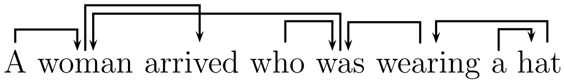


Figure 1.2: Syntactic discontinuity as crossing dependency

It may be that these cases are best analyzed as instances of ellipsis. Regardless, they seem only to occur in speech, never in writing; and from a computational and processing perspective it may be more useful in some cases to treat these simply as syntactic discontinuities, if a well-developed machinery for syntactic discontinuity is already in place.

1.3.2 Crossing dependencies and the constituency → dependency homomorphism

Another useful way of thinking about discontinuity and nonlocality is in terms of *dependency*. If we represent the syntactic structure of a sentence very simply with a word-word dependency tree, then nonlocality/discontinuity turns up slightly differently: as a crossing between dependency arrows. Figure 1.2 illustrates the crossing dependency induced by a relative clause extraposition. Although the criteria of discontinuous constituency and crossing dependency may at first glance seem to be rather different, we can actually translate rather freely between them due to a strong mapping between headed context-free trees and dependency trees. In formal terms, the local head-sister relation induces a *surjection*, or an *onto* function, from headed CF trees to word-word dependency trees without crossing dependencies. That is, every headed CF tree determines exactly one dependency tree (and the resulting dependency tree has no crossing dependencies), and for every dependency tree without crossing dependencies, there is a headed CF tree that maps to it. This result follows from the work of Miller (2000), but I provide a more direct proof in Appendix A.

This mapping allows us to translate between talking about discontinuity, nonlocal dependency, and crossing dependency rather freely. If we identify a sentence as having no discontinuous constituents by providing a headed CF tree analysis for it (meaning that all the dependencies are local), then we can be sure that the sentence has no crossing dependencies. Likewise, if we show that a sentence has a crossing dependency,

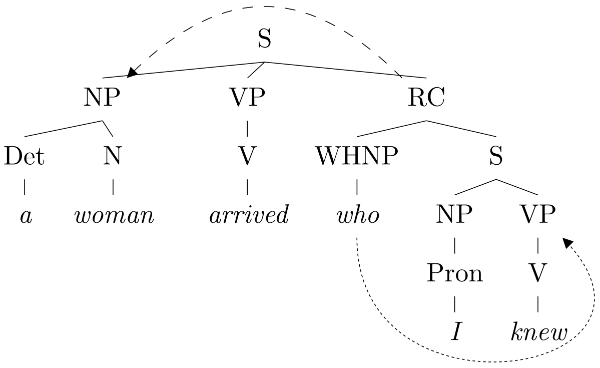


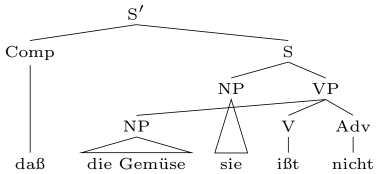
Figure 1.3: Post-hoc nonlocal dependency recovery on context-free trees

we can be sure that there is no headed CF tree analysis for it in which all the relevant node-node dependency relations are between local heads and sisters.[7]

1.4 Three approaches to discontinuous constituency and parsing

The latter part of this thesis concerns itself with the problem of identifying nonlocal dependencies in syntactic parse trees. We can usefully distinguish three different approaches to this problem. The first involves two distinct, serial components of dependency recovery—the first local, the second non-local—and the latter two involve a single unified process of recovering both local and nonlocal dependencies. In the former approach, the parsing task is first treated as *approximately* context-free: a CFG description of the language is used as the grammar for parsing, and only local

[7] What we *cannot* do is to reason that if we have a headed discontinuous-constituency tree analysis for a sentence, then it must have crossing dependencies. For example, a discontinuous-VP analysis could be proposed for OSV word order in German embedded clauses:



but, assuming that V is head of VP and VP is head of S, there is no crossing in the induced dependency tree.

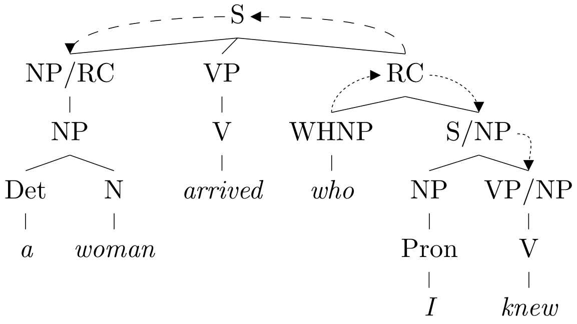


Figure 1.4: Category-structure enhancement of context-free trees

syntactic relationships are considered. Once inferences about the optimality of possible parses are made in the first, context-free phase, a second dependency correction phase is introduced, where the input is a context-free parse tree and the output is an augmentation of that context-free tree that includes annotation of its nonlocal dependencies. The outcome of such a two-phase process is illustrated in Figure 1.3. Note that nothing about the category structure of the tree itself illustrates that the extraposed relative clause is a modifier of the subject NP, or that the relative pronoun is the underlying object of *knew*. Rather, these facts are overlaid on top of the CF tree with special annotations (illustrated crudely with the dashed and dotted lines). This two-phase approach has the advantage of being able to directly leverage prior work, because the bulk of the last decade of broad-coverage Treebank parsing (most notably, Magerman 1994; Collins 1999; Charniak 1997, 2000) has concerned itself precisely with the first phase defined here. The potential disadvantage is that the context-free approximation in this first phase may encourage errors that cascade into the total process of dependency recovery. I take up construction of the second phase of this process in Chapter 3, and investigate the extent to which the context-free approximation is a safe move, and the difficulty of subsequent nonlocal dependency recovery.

The latter two approaches require the representation of discontinuous constituency *within* a parser; there are two major strategies for this. On the one hand, it is possible to track discontinuous relations in the *category structure* of the parse chart. This entails some variant of GPSG-style slashed categories to track the relationship

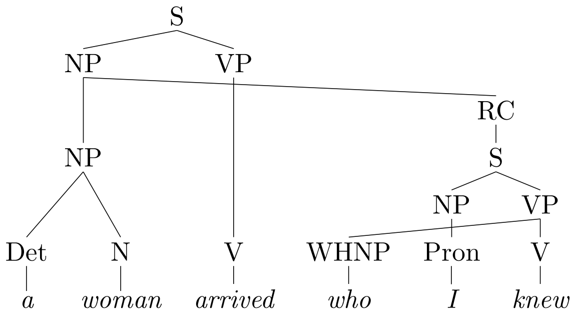


Figure 1.5: Discontinuous-constituency parse for extraposed relative clause

of a node inside a category X with some other node outside X. The *probability* of edge construction can then be estimated using a variety of techniques taking into account a variety of factors such as the major category, the slashed category, and a variety of potentially trackable lexical heads. The effect on computational complexity is well-understood: the number of categories C and right-hand rule sides R can inflate dramatically, but the exponent with respect to string length remains the same (although additional indexations, such as tracking the lexical head governing a slashed category, can further increase the polynomial order of complexity). Collins (1999) investigated a very limited subset of slashed-category grammars to recover relativization information in English Penn Treebank parse trees. Hockenmaier (2003) and Dienes and Dubey (2003a) also used this strategy, with mixed success. Because the category-enhancement approach to parsing with nonlocal dependency is fairly well-understood and has been previously investigated in a probabilistic setting, I will not pursue it further in this thesis.

The other major strategy is to track discontinuous relations in the *edge structure* of the parse chart. This approach has been much less extensively investigated. As I discussed above, context-free edges are indexed only by their left and right bounds; lexicalization effectively indexes one more internal position. A discontinuous category, such as an NP with an extraposed relative clause, can be directly represented with a more complicated edge structure. This situation is in fact not all that different from that of lexicalized grammars. The Head Grammars of Pollard (1984), for example, where a node in a derivation tree covers a *pair* of continuous substrings, can be parsed

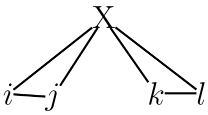


Figure 1.6: Chart entry combination in a Head Grammar parse

in $O(n^6)$ time and $O(n^4)$ space. This is because a category in a Head Grammar chart must have the structure in Figure 1.6. Four indices in the string track the complete structure of a Head Grammar edge; all possible edge combinations can therefore be expressed with six total indices. We can therefore retain tractability while allowing limited forms of discontinuous constituency directly into our grammar. Virtually no work has been done on probabilistic parsing with such grammars, however; Plaehn (2000), working in the Discontinuous Phrase Structure Grammar framework (Bunt, 1996), is a rare exception. In Chapter 4, I investigate generalizations of the Head Grammar approach to limited discontinuous constituency in a probabilistic parsing setting.

1.5 Thesis Overview

In Chapter 2 of this thesis, I present a case for an *expectation-based* theory of online syntactic processing. A realistic model of online syntactic processing must account for our abilies to perform rapid, incremental processing, and to disambiguate linguistic input robustly and accurately; it must also account for differential processing difficulty—some parts of some sentences are harder to comprehend than other parts of other sentences. The dominant theoretical paradigm of human sentence processing (Miller and Chomsky, 1963; Clifton and Frazier, 1989; Gibson, 1998) has taken a *resource-limitation* view of differential processing difficulty: different syntactic structures require different amounts of a given resource X (typically some form of *working memory*), and X is in short supply in the human parser. While these models have been successful in explaining a range of established processing results for English and syntactically similar languages, they have more recently been found to be more problematic for verb-final structures (Konieczny, 2000; Konieczny and Döring, 2003):

they predict increased reading time at the final verb for longer clauses, but in fact reading time seems to decrease as the number of preverbal dependents increases. I explore expectation-based syntactic processing as an alternative, *resource-allocation* view of differential processing difficulty, consistent with an alternative lineage of psycholinguistic research (Marslen-Wilson, 1975; Tanenhaus et al., 1995; Jurafsky, 1996) in which the human parser simultaneously explores a number of possible structural interpretations in parallel, allocating different amounts of resources to alternative interpretations in proportion to their plausibility at a given moment. Specifically, the allocated resource is probability mass, and I show that if we assume processing difficulty to arise from the redistribution of probability mass among alternative sentence interpretations upon encountering new evidence, the predicted difficulty of a given word in a sentence can be shown to be determined by the word's probability given what has come before it, a result equivalent to a previous proposal by Hale (2001). Under such a theory, all that is required to make predictions about incremental processing difficulty is a language model—that is, a probability distribution over strings. I use language models derived from probabilistic context-free grammars of German to show that this expectation-based model accurately predicts the results of Konieczny (2000); Konieczny and Döring (2003) for final-verb reading times in German, and also account quite well for results near the beginning of the German clause that had previously been argued to undermine frequency-based processing accounts (Schlesewsky et al., 2000).

In Chapters 3 and 4 I turn to the problem of incorporating non-local dependency into probabilistic grammatical models. Chapter 3 treats the identification of syntactic dependencies as a two-phase, serial process, an approach comparable to traditional LFG parsing (Kaplan and Maxwell, 1993), and more recently, the work of Johnson (2002) for Penn Treebank parsing. In the first phase, an input string is parsed with a probabilistic context-free grammar; in the second phase, the context-free parse serves as an input to a discriminative classification model that determines the non-local dependency relationships on the context-free tree. In each phase, the optimal structure is greedily selected; this corresponds to a modularity assumption between the stages of identifying context-free and trans-context-free structure. Among other things, the results of this serial algorithm can serve as a baseline to which any integrated algorithm of nonlocal dependency recovery—such as the discontinuous-constituency parsing algorithm of Chapter 4—can be compared. I find that when coupled with a state-of-the art lexicalized PCFG parser for English (Charniak, 2000), the serial approach does not lead to serious degradation of the ultimate recovered dependency trees. For German, however, where PCFG parsing is much less developed, nonlocal dependency recovery suffers from far more degradation.

Chapter 4 takes up the problem of probabilistic discontinuous-constituency parsing—that is, parsing with probabilistic grammars whose derivation tree nodes can correspond to discontinuous portions of the input string. In addition to a grammatical formalism with this capability, this undertaking requires a *parameter estimation procedure* and a *parsing algorithm*. I show that the *Linear Context-Free Rewrite System* (LCFRS) formalism (Vijay-Shanker et al., 1987) is well-suited to the task for two reasons. First, although it can generate mildly context-sensitive languages, the *derivations* in an LCFRS grammar are context-free, a property which makes the parameter estimation procedure for a probabilistic LCFRS particularly easy. Second, unlike some forms of discontinuous-constituency grammars, the complexity of LCFRS recognition is always polynomial, and the polynomial order of a particular LCFRS can be bounded based on its most complex rule. I take up the problem of estimating probabilistic LCFRSs (more succinctly, *probabilistic wrapping grammars* or PWGs) from corpora of discontinuous-constituency trees (such as NEGRA and the Penn Treebank) and parsing efficiently with the resulting grammars. I argue that a major bottleneck for efficient parsing with the resulting grammars is the incorporation of *distance sensitivity* into derivation tree probabilities: some kinds of discontinuous constituents in natural language tend strongly to favor small discontinuities, but this fact is not reflected in the grammars resulting from the simplest estimation procedures. I show that distance-sensitive PWGs can in fact be coherently formulated using a factorization of the grammar into *Immediate Dominance* (ID) and *Linear Precedence* (LP) components, a long-standing idea perhaps most prominent in Generalized Phrase Structure Grammar (Gazdar et al., 1985) but not previously given a probabilistic interpretation.

Chapter 5, the conclusion, summarizes the major results of the thesis, and points to future research directions opened up by this work, both in computational linguistics and psycholinguistics.

Chapter 2

Surprisal-based syntactic processing

In this chapter I investigate the application of probabilistic grammars in the formulation of *expectation-based* models of online syntactic processing. I provide a new information-theoretic derivation of a parallel, expectation-based model proposed by Hale (2001), and apply it to the results of four recent online processing experiments involving German word order. In every case, I show that the expectation-based model matches experimental results at least as well as, and in most cases considerably better than, competing memory-based models of online syntactic processing. Finally, I discuss several distinctive properties of the Hale (2001) model in comparison with other expectation-based models, and consider future propects for its further development.

2.1 Introduction

The advent of generative grammar in the late 1950s and early 1960s defined many of the central problems that still face the field of human sentence processing today. Perhaps two related problems remain foremost: *ambiguity resolution* and *differential difficulty*. The study of ambiguity resolution is perhaps most heavily motivated by the fact that local syntactic ambiguities can sometimes cause a comprehender to pursue the wrong analysis of a sentence, and may never be able to recover the correct analysis, as in (1) below.

(1) The horse raced past the barn fell.

Differential difficulty is the fact that not all sentences in the sets naturally defined by generative natural language grammars are equally easy to comprehend. Center-embedding is probably the best-known such case: syntactic analysis seems to point inevitably to the grammaticality of arbitrarily deep center embedding of relative clauses, such as in (2) below, but (2c) is generally considered incomprehensible, despite the fact that the sentence is at no point structurally ambiguous.

(2) a. The salmon fell off the grill.
b. The salmon that the man smoked fell off the grill.
c. The salmon that the man that the dog chased smoked fell off the grill.

Chomsky's competence/performance distinction immediately suggests an explanation: in competence terms, (2c) is perfectly grammatical, but performance limitations render it difficult to understand. This is in contrast to (3) below, which in competence terms is ungrammatical but nevertheless relatively easy to understand.

(3) *The salmon off the grill fell.

The original proposal by Miller and Chomsky (1963, pp. 470-472) was that the badness of nested-dependency constructions such as in (2c) arose from *working memory limitations*: at the deepest part of a multiply-embedded relative clause, the comprehender has to remember all the previous levels of embedding, and retrieve them as each relative clause is closed off. This idea synergized well with the wider understanding of working memory limitations in cognitive psychology (see, for example, Miller 1956), and has led to the dominant paradigm of online syntactic processing theories, which I will call *resource-requirement* or *resource-limitation* theories. In a nutshell, these propose that:

- some syntactic structures require more of a given resource than do others; and

- that resource is in short supply in the human parser; and
- this gives rise to greater processing difficulty for some structures than for others.

Typically this limited resource is some form of *memory*. The resource-limitation position has also come to inform a persistent view of ambiguity resolution: the resource-limited parser can only pursue one alternative at a time (i.e., the parser is serial), and in the face of local ambiguity, the processor chooses the alternative that minimizes the resources consumed. This viewpoint has inspired a variety of ambiguity resolution theories, including Late Closure (Frazier and Fodor, 1978), Minimal Attachment (Frazier, 1979), and the Active Filler Hypothesis (AFH; Clifton and Frazier 1989).

Both the notion of differential processing difficulty and the accompanying processing theories have evolved in the past forty years. More refined experimental techniques, notably self-paced reading and eye-tracking, have allowed researchers to identify loci of processing difficulty at specific points within individual sentences. Correspondingly, as measurements have improved, theories have become more refined. Perhaps the most salient modern incarnations of memory-centered resource-requirement theories are, for ambiguity resolution, the AFH; and, for locally unambiguous sentences, Gibson's Dependency Locality Theory (DLT; Gibson 1998, 2000).

At the same time, an alternative line of research has focused on the role of expectations in syntactic processing. This idea has historically been associated most closely with so-called *constraint-based* processing models such as that of MacDonald (1993); MacDonald et al. (1994); Tanenhaus et al. (1995); McRae et al. (1998), and can be traced back to early work by Marslen-Wilson (1975).[1] This line of work typically takes a strong integrationist and parallelist perspective: the comprehender draws on a variety of information sources (structural, lexical, pragmatic, discourse) to evaluate in parallel a number of possible alternatives for the input seen thus far. For the most part, the primary concern of constraint-based work has been ambiguity resolution, the argument being that possible structural analyses are ranked according to their plausibility on a number of dimensions, rather than according to the amount of resources

[1] See Jurafsky (2003) for a more comprehensive account of the history of expectation-based approaches in human sentence processing, including syntactic processing.

they consume. Empirically observed processing difficulty after local ambiguity resolution is informally ascribed to either a *reranking* of the favored analysis, or *competition* between closely-ranked analyses. In combination with a committment to parallelism, the constraint-based position can be thought of as a *resource-allocation* approach to syntactic processing: the parser allocates different amounts of resources to different interpretations of the partial input, and difficulty arises when those resources turn out to be inefficiently allocated.

As argued by Jurafsky (2003), probability theory fits naturally as an underlying infrastructure for constraint-based approaches to express the combination of multiple information sources. The use of probability theory for psycholinguistic modeling has in fact become more prevelant over the past decade, beginning with Jurafsky (1996) and continuing in Narayanan and Jurafsky (1998, 2002); Crocker and Brants (2000). Nevertheless, because ambiguity resolution has been the primary concern, none of this work has formulated a precise *linking theory* between ambiguity-resolution models and observable measures that would account for differential processing difficulty. The work of Hale (2001, 2003a) has begun to redress this state of affairs, by proposing models of processing difficulty based on incremental parsing with probabilistic grammars; I discuss this work in greater detail in Section 2.2. In this chapter I present a linking theory for constraint-based ambiguity resolution, by deriving a resource-allocation theory of processing difficulty: the possible structural analyses of a partial input are preferentially ranked in parallel, and the difficulty of a new word corresponds to the amount of reallocation necessary to reflect the word's effect on the preference ranking. In Section 2.2, I give the derivation of this theory, which allows us to make highly precise predictions about the difficulty of a given word in a given sentence while remaining agnostic as to exactly what the possible structural analyses of the sentence are, and show that it is equivalent to the *surprisal* theory stipulated by Hale (2001). In the remainder of the chapter I proceed to investigate the conditions under which resource-allocation and resource-requirement theories maximally diverge in their difficulty predictions, and argue that the surprisal theory explains a considerable portion of the variation in processing difficulty experimentally observed in online comprehension.

2.2 Deriving a resource-allocation theory of processing difficulty

In this section I present a new derivation of a theory of resource-allocation processing difficulty, based on a highly general conception of sentence comprehension, and accounting for principles that are necessary for any realistic model of human sentence processing.

A language contains a (normally infinite) set of *complete structures* such that a fully disambiguated utterance corresponds to exactly one structure. Each structure contains the complete string of the utterance, plus presumably at least some other information, since some well-formed strings are ambiguous. As an example, we might consider a complete structure to be the string plus its syntactic/semantic analysis, so that the sentence *the girl saw the boy with a telescope* might be compatible with two possible complete structures, one where *with a telescope* modifies *saw* and one where it modifies *boy*. However, we will remain agnostic as to precisely what these complete structures contain, so long as they contain the complete string.

We can reasonably define what it means to *comprehend* a sentence S as choosing the "best" or "most highly preferred" structure complete structure T (mnemonic for a (syntactic/semantic) *tree*, but not necessarily a structure with tree-like topology) that is consistent with S; more generally, sentence comprehension can be said to involve the (implicit or explicit) construction of a preference ranking over all the the possible structures T in the language, given S. We will use the language of probability theory to express preferences and rankings, so comprehension of S involves placing a probability distribution over all the possible T in the language once we have seen S.

There is ample evidence, however, that sentence comprehension is *incremental*: we do not wait until we have heard an entire sentence to start disambiguating and comprehending. Perhaps the most explicit demonstration of this fact comes from work in cross-modal eye-tracking (Altmann and Kamide, 1999), where listeners were demonstrated to start looking at the plausible objects in a picture for the main verb of a sentence as soon as they heard the verb. Comprehenders are able to make inferences about later parts of the sentence based on what they have heard earlier in the sentence.

To capture this fact, we define the comprehension of a *partial* input sequence $w_{1\cdots i}$ (the first i words of the sentence) to be placing a preference (i.e., probability) distribution D over the possible structures T based on $w_{1\cdots i}$, plus context external to the sentence itself. For listeners to be capable of incremental inference, they must be constantly updating D; for simplicity in the present context, we assume that they update D after every input word.

The probability distribution D consists of an allocation of resources among the possible interpretations of the sentence, and for the resource-allocation theory of processing difficulty our single stipulation will be that difficulty is incurred by updating D, and that difficulty is quantified by the degree that D has to be updated. To quantify the update of D we will use the *relative entropy* of the updated distribution with respect to the old distribution. The relative entropy of a probability distribution q with respect to another distribution p (also known as the *Kullback-Leibler (KL) divergence* of q from p) is defined as

$$D(q||p) = \sum_{T \in \mathcal{T}} q(T) \log \frac{q(T)}{p(T)} \tag{2.1}$$

Intuitively speaking, the relative entropy of q with respect to p can be thought of as the penalty incurred from encoding the distribution q with p. When $q = p$, $D(q||p) = 0$, and the greater the difference between the distributions, the greater the relative entropy.

Remarkably, it turns out that under this formulation of resource-allocation processing difficulty, regardless of the form of complete structures T or the preference distribution D, the predicted difficulty of the i^{th} word, w_i, is precisely equal to the *surprisal* of w_i, which is defined as the negative log-probability of w_i in its sentential context (which we denote by the already-seen input sequence $w_{1\cdots i-1}$) and extra-sentential context (which we denote simply by CONTEXT):

$$\textit{difficulty} \;\propto\; \log P(w_i | w_{1\cdots i-1}, \text{CONTEXT}) \tag{2.2}$$

Precisely this measure of difficulty was in fact proposed as a stipulation by Hale (2001), albeit specifically for the case of a probabilistic context-free grammar Earley parser with no extra-sentential context. Surprisal is minimized (goes to zero) when a word *must* appear in a given context (i.e., when $P(w_i|w_{1\cdots i-1}, \mathrm{CONTEXT}) = 1$), and approaches infinity as a word becomes less and less likely. I give the simple proof of this result in Section 2.2.1, and discuss its implications in Section 2.2.2.

2.2.1 Proof of equivalence to surprisal

Consider any stochastic generative process that generates complete structures that consist at least partly of surface strings to be identified with serial linguistic input. Examples of such processes include but are not limited to n-gram models, Hidden Markov Models (HMMs), and probabilistic context-free grammars. Call $\mathcal{T}$ the set of complete structures generated by this process, and P the probability distribution that the process induces over $\mathcal{T}$, conditioned on some (possibly null) external context. Furthermore, for any particular input prefix $w_{1\cdots i}$ define the probability distribution P_i as the conditional distribution over $\mathcal{T}$ induced by P, given the prefix $w_{1\cdots i}$:

$$P_i(T) \equiv P(T|w_{1\cdots i}), \forall T \in \mathcal{T} \tag{2.3}$$

and define the set $\mathcal{T}_i$ as the set of complete structures with prefix $w_{1\cdots i}$ (note that $\mathcal{T}_i$ is also the subset of $\mathcal{T}$ that has non-zero probability according to P_i). We will also give P and P_i a secondary meaning as signifying joint and conditional (respectively) probability distributions over words: $P(w_{1\cdots i}) \equiv \sum_{T \in T_i} P(T)$, and $P_i(w) \equiv P(w|w_{1\cdots i})$.

I will now show that

$$D(P_{k+1}||P_k) = -\log P_k(w_{k+1}) \tag{2.4}$$

i.e., the relative entropy of the distribution over hidden structures *after* having seen w_{k+1} from the distribution *before* having seen w_{k+1} is simply equal to the surprisal of w_{k+1}.

Proof. The proof requires only a simple application of the chain rule. First, note that for any integer j and any $T \in \mathcal{T}_j$,

$$\begin{aligned} P_j(T) &\equiv P(T|w_{1\cdots j}) && (2.5)\\ &= \frac{P(T, w_{1\cdots j})}{P(w_{1\cdots j})} && (2.6)\end{aligned}$$

And by virtue of the fact that T is in $\mathcal{T}_j$,

$$\begin{aligned} P_j(T) &= \frac{P(T, w_{1\cdots j})}{P(w_{1\cdots j})} && (2.7)\\ &= \frac{P(T)}{P(w_{1\cdots j})} && (2.8)\end{aligned}$$

Therefore, for all $T \in \mathcal{T}_{k+1}$,

$$\begin{aligned} \frac{P_{k+1}(T)}{P_k(T)} &= \frac{\frac{P(T)}{P(w_{1\cdots k+1})}}{\frac{P(T)}{P(w_{1\cdots k})}} && (2.9)\\ &= \frac{P(w_{1\cdots k})}{P(w_{1\cdots k+1})} && (2.10)\\ &\equiv \frac{1}{P_k(w_{k+1})} && (2.11)\end{aligned}$$

independent of T.

Therefore, the KL divergence from P_{k+1} to P_k is

$$\begin{aligned} D(P_{k+1}||P_k) &= \sum_{T\in\mathcal{T}_{k+1}} P_{k+1}(T) \log \frac{P_{k+1}(T)}{P_k(T)} && (2.12)\\ &= \log \frac{1}{P_k(w_{k+1})} \sum_{T\in\mathcal{T}_{k+1}} P_{k+1}(T) && (2.13)\\ &= -\log P_k(w_{k+1}) && (2.14)\end{aligned}$$

□

Intuitively, this proof results from the fact that the ratio of the probability of any complete structure T before versus after seeing a word w_{k+1} is constant, because the original process generating $\mathcal{T}$ is the same. This constant ratio has to be the amount of probability mass pruned away from P_k by the requirement of compatibility with w_{k+1}—in other words, the conditional probability of w_{k+1}, as seen in Equation 2.11. This is the probability ratio term in the KL divergence, as seen in Equation 2.13, and because it is constant, the probability over structures T can be independently summed out. Finally, note that this proof of equivalence only holds if the extrasentential context does not change at the same time as w_{k+1} is processed; if the extrasentential context is changed, it may change the structure of P, which may break the constancy of the probability ratio derived in 2.11.

2.2.2 Implications of relative-entropy derivation of surprisal

This equivalence has important implications for how we conceptualize the incremental parsing process. In a fully parallel, incremental probabilistic parser capable of online inference (that is, inference before input is complete), storing the complete set of ranked partial parses consistent with already-seen input is also equivalent to assigning a probability distribution over the complete structures to which the already-seen input may possibly extend. Upon termination of the input, this set of ranked partial parses determines a most-likely interpretation. On the way, after every new input token, such a parser must update its collection of ranked partial parses—and therefore its distribution over completed parses—to reflect the new information. Intuitively, the KL divergence from a distribution p to another distribution q measures the penalty incurred by encoding, or approximating, q with p. The surprisal can therefore be interpreted as the difficulty incurred in replacing the old distribution with the new.

In most cases, a partial input $w_{1 \cdots i}$ will be compatible with an infinite number of complete structures T—we can see this simply from the fact that the beginnings of most sentences can be completed in an infinite number of ways. Therefore, it is neither psychologically nor practically possible for the distribution D to be implemented as an enumeration over complete structures. Rather, D would be implicitly determined by some tractable incremental processing algorithm, such as a chart parser (Kay, 1980). The cognitive plausibility of such an architecture is demonstrated by the implementation and experiments I discuss in the subsequent sections of this chapter, using only a modest quantity of computational resources.[2]

Remarkably, however, we have found in Section 2.2.1 that in order to determine relative-entropy incremental update costs for a stochastic generative grammar G we do not necessarily even need to implement an incremental parsing architecture for it. Because the incremental update cost of a word w_i is simply its contextualized surprisal—$\log P(w_i|w_{1\cdots i-1}, \text{Context})$—as long as we can calculate the probabilistic word model that G determines, we can determine the word-by-word difficulty that G predicts. Deriving surprisal from this information-theoretic characterization of incremental update in the human parser provides an interpretation of the surprisal model of expectation that connects robust processing, disambiguation, inference, and processing difficulty. In addition, this derivation leads to several desirable properties of the surprisal-based processing model, which I now proceed to outline.

First, deriving rather than stipulating surprisal eliminates a potential conceptual vulnerability of expectation-based approaches: calculating expectations about upcoming structures in a sentence can be computationally expensive, so why would the human parser waste resources on constantly calculating and updating the likelihood of upcoming words and/or structures in a sentence? We now have a clear answer to this challenge: surprisal as the predicted difficulty of word w_i falls out of the incremental update process itself. Expectations about upcoming words in a sentence need not be explicitly calculated; rather, they are implicit in the partial parse of an incomplete input.

Second, the proof in Section 2.2.1 shows that a processing difficulty metric defined with respect to probability distributions over *arbitrary linguistic structures* can

[2] All of the models I describe in this chapter could be calculated using an untuned Java implementation of Stolcke (1995)'s incremental chart parsing algorithm using only a few hundred megabytes. The large majority of processing time required was devoted to logarithmic and exponential operations, nearly all of which could be done away with if desired, at the price of a small loss in arithmetic precision.

be re-expressed in terms of probability distributions over *strings.* This means that we have expressed a *representation agnosticism* for our theory of processing difficulty: in order to make predictions about word-by-word processing difficulties, we do not need to make a strong committment to a particular grammatical formalism and accompanying probability distribution. All we need to do is to estimate a probabilistic string model that captures the distributional linguistic regularities that we are interested in. The predictions made by such a model will apply for *any* probabilistic grammar that determines the same probability distributions over strings. In another manner of speaking, we have separated the problem of *estimation* from the problem of *representation.*

For example, McDonald and Shillcock (2003) show that a word's bigram probability is a significant predictor of reading times for a corpus of British newspaper articles. A bigram word model is a conditional probability model over strings. Given the representation agnosticism we have derived for surprisal-based processing, we do *not* need to conclude from McDonald and Shillcock's work that the human parser tracks bigrams (although the authors themselves conclude something close to this). We can instead conclude more agnostically that the probabilistic grammatical models the human parser uses for incremental processing and disambiguation determine probabilistic languages that, at a minimum, sensitize the probability of a word w_i to the word w_{i-1} that immediately precedes it. This encompasses a wide range of probabilistic structures, including not only n-grams but also, for example, lexicalized PCFGs (Charniak, 2001). The representation agnosticism is relevant to the modeling in this chapter, both in Section 2.4.1 for the treatment of German relative clauses, and in Section 2.6 in the treatment of German main-clause word-order variation.

A third point, closely related to representation agnosticism, has to do with *bias* in estimating word-by-word comprehension difficulty. Many stochastic string-generating processes (including HMMs and PCFGs) generate unobserved hidden structure "behind" the string—parts of speech in HMMs, syntactic trees in PCFGs—whose granularity is not known *a priori.* In order to determine a specific probabilistic model, a granularity level must be chosen and the relevant event probabilities must be estimated with respect to that granularity. Because a word's surprisal is totally dependent on the resulting probabilistic string language, however, a refinement in granularity level will not result in a change in surprisal predictions of a maximum-likelihood estimated model *unless* there are empirical differences in the relevant event probabilities at the finer granularity. As an example, in PCFG modeling we might wonder whether to grammatically distinguish animacy at the level of the noun phrase. Adding a binary animacy distinction to the grammar, for example, would split the following three rules into six:

(4) a. S → NP VP ⇝ S → NP[+anim] VP
S → NP[−anim] VP

b. NP → Det N[sg] ⇝ NP[+anim] → Det N[sg]
NP[−anim] → Det N[sg]

c. NP → N[pl] ⇝ NP[+anim] → N[pl]
NP[−anim] → N[pl]

Now suppose that animate and inanimate NPs turn out to have exactly the same relative frequency of realization as singular (as in (4b)) versus plural (as in (4c)). Under these circumstances, the resulting animacy-distinguished PCFG still determines exactly the same probabilistic string model, and so its surprisal predictions will be unchanged. (If animate and inanimate NPs were realized as singular/plural with different empirical differences, the resulting probablistic string model and surprisal values would of course reflect this, as would be desired.) This lack of granularity-induced bias contrasts sharply with other probabilistic syntactic processing models, as I proceed to describe in Section 2.2.3.

2.2.3 Comparison with other probabilistic syntactic processing models

As I note in Section 4.1, the constraint-based paradigm of ambiguity resolution generally ascribes empirically observed processing difficulty to the *reranking* of candidate analyses, or to *competition* between multiple high-probability analyses. We can also usefully distinguish *pruning* approaches (Jurafsky, 1996), which conceive of incremental parsing as a limited-parallel ranking process, and ascribe difficulty to the pruning away of temporarily low-probability candidates that turn out to be needed; and *attention-shifting* approaches (Narayanan and Jurafsky, 1998, 2002), a special form of reranking difficulty in which changes in the identity of the highest-ranked candidate cause difficulty. The surprisal theory can also be thought of as a special form of reranking, in which the cost of reranking is determined by the relative entropy between the old and new rankings.

All of these approaches (including reranking in its most general form), however, differ crucially from the surprisal theory in that they are neither representation-agnostic nor immune to granularity bias. To assess the degree of competition between analyses, it is necessary to make a prior committment as to what counts as "one" analysis. To determine whether a given analysis will be pruned away, it is necessary to specify that analysis precisely, either to determine the exact probability mass it determines, or to count how many analyses are ranked above it. Likewise, the attention-shifting approach requires a determination of the highest-ranked analysis, which again requires a precise specification. It might be thought that representation-dependence is a positive feature of a probabilistic processing theory, which might be of interest to theoretical linguists as a means of choosing between alternative syntactic analyses. I submit, however, that representation-dependence is, all else being equal, a *negative* feature of a probabilistic processing theory. The susceptibility of representation-dependent theories' predictions to shift as the structure and granularity of the model change makes the specific predictions of the theory (given a set of empirical syntactic distributional regularities) more flexible and more difficult to pin down, and therefore more difficult to falsify. Although the predictions of the surprisal theory are still dependent on the structure, empirical basis of estimation, and independence assumptions of the probabilistic model used, the route to determining empirical predictions is nevertheless clearer.

The issue of representation dependence also emerges in another recently-proposed syntactic processing theory, the *Entropy Reduction Hyphothesis* (ERH) of Hale (2003b,a, 2004). In this theory, the difficulty of a word w is the difference between the entropies of the probability distribution over compete structures T *before* and *after* seeing w.[3] Although the idea seems attractive and its implementation is technically interesting, the ERH is beset with several conceptual difficulties. First, entropy is a summary measure of an individual probability distribution, and taking the difference of the entropies of two different probability distributions in no way compares any important properties of the two distributions with each other (entropy can even *increase* after seeing a given word). The relative-entropy measure I have argued for *is* defined as a point-by-point comparison of two probability distributions in a conceptually meaningful way. Second, the ERH is highly susceptible to granularity bias; more fine-grained distributions have higher entropy.[4] Third, the ERH fails to capture certain important intuitions that we expect of an expectation-based processing theory. For example, although it predicts that processing an open-class word incurs more difficulty than a closed-class word (because the distribution of words in an open class is higher entropy), it does not predict that high-frequency words in an open class are generally easier to process than low-frequency words in an open class, because the entropy reduction is the same (in both cases the entropy associated with the unknown word goes to zero); the ERH therefore cannot smoothly incorporate well-known findings that word frequency and contextual plausibility are negatively correlated with measures of processing difficulty (see Rayner 1998, pp. 387–388 for discussion and references), as the surprisal model can.[5]

In summary, the surprisal model can be seen as an elegant instantiation of a reranking theory with especially elegant mathematical properties, capturing the core

[3] The *entropy* H of a probability distribution p is defined as

$$H(p) \equiv \sum_{T \in \mathcal{T}} p(T) \log p(T)$$

Note the close relationship between the form of the entropy of a distribution p and the relative entropy of p from another distribution q, given in 2.1.

[4] One way of remedying granularity bias would be to use complete-sentence entropy rather than complete-structure entropy as the measure of interest.

[5] One area where entropy *rate* (in contrast to entropy reduction) might plausibly play a useful role would be as a generalization of the *competition* model of processing difficulty, because “competing” candidate sets can be thought of as high-entropy. Keller (2004) has in fact conducted a preliminary investigation of entropy rate as a predictor of processing difficulty, although he used per-word rather than complete-sentence entropy.

intuitions underlying the idea that expectations play a role in determining processing difficulty. The intuitions behind pruning, competition, and attention-shift models of syntactic processing difficulty are, of course, valid and merit theoretical and experimental investigation; but the representation-independence and immunity to granularity bias make the surprisal model perhaps particularly robust and easy to use in a concrete analytical setting.

2.3 Divergence in predictions of expectation and locality theories

The surprisal-based theory of expectation-based processing outlined in the previous section provides a framework for predicting the degree of difficulty associated with comprehending each word in a sentence. It is therefore of interest for us to consider the general types of predictions this theory makes, and how they diverge from the predictions of locality-based theories of difficulty. I take locality-based processing theories to include two hallmark proposals as to the primary sources of difficulty in syntactic comprehension. The first is that difficulty is incurred when a syntactic relationship between distant entities needs to be constructed, and that the greater the distance, the greater the difficulty. Gibson's Dependency Locality Theory (DLT; Gibson 1998, 2000) is perhaps the best-known instance of this type of proposal. The second proposal is that preference for more local syntactic relationships directly guides disambiguation, and when maximally local structures turn out to be wrong, difficulty is incurred because the parser has been misled. This proposal has had a wider variety of incarnations, including perhaps most prominently Minimal Attachment/Late Closure (Frazier and Fodor, 1978; Frazier, 1987) and the Active Filler Hypothesis (AFH; Clifton and Frazier 1989).[6]

At a general level, how do the predictions of locality- and surprisal-based theories differ? Since locality-based theories only make predictions involving syntactic dependencies, we can conveniently divide our scope of discussion into the three major

[6] Gibson (1998) also gives suggestions as to how the DLT could guide attachment preference, but disambiguation has not been a major theme in DLT-guided research.

types of dependencies observed in natural language: head-initial, head-final, and long-distance dependencies.[7] Head-initial dependencies, under some circumstances, create an interesting but familiar contrast between late-closure and expectation-based processing theories. When a modifier attaches leftward into a recursive, right-branching phrase, it creates an attachment level ambiguity. In the phrase *the chauffeur of the actor who lost his wallet*, for example, the relative clause can modify either of the nouns to its left. Under the most common circumstances in English, the attachment preference is to the lower NP. Late Closure demands that the most recently processed node be the preferred site of attachment, and thus predicts this preference, since the lower NP is constructed more recently than the higher NP. However, Cuetos and Mitchell (1988) found that in Spanish, the higher attachment is preferred, calling into question Late Closure as a universal principle. It also turns out that the low attachment is more frequent than the high attachment in English, but in Spanish the reverse is the case. This led to the Tuning Hypothesis proposed by Mitchell and Cuetos (1991): parsing preferences are sensitive to construction frequencies. Language- and frequency-sensitive attachment preferences have been the subject of considerable research since then (Mitchell and Cuetos, 1991; Mitchell, 1994; Mitchell et al., 1995; Gibson et al., 1996a,b; Carreiras et al., 1996; Mitchell and Brysbaert, 1998; Gibson and Schütze, 1999; Desmet et al., 2002; Desmet and Gibson, 2003), which for reasons of space I will not go into. At present, however, all attachment-level preferences reflected in experimentally observed online reading time differentials seem to be consistent with fine-grained corpus frequencies.[8]

In the case of these head-initial dependencies, the surprisal theory simply reiterates Cuetos & Mitchell's Tuning Hypothesis. If a partially-completed right modifier is compatible with multiple leftward attachments, the rational comprehender will have

[7] A long-distance dependency may alternatively be considered a dependency between an extraposed element and its "origin" site, or a direct dependency with its governor. In practice, this distinction is usually irrelevant, as the primary point of interest is the (pretheoretical) ambiguity induced by a leftward-extraposed long-distance dependent.

[8] The two recent challenges to frequency-based accounts were Mitchell and Brysbaert (1998), who found a frequency/preference mismatch in Dutch; and Gibson and Schütze (1999), who found a frequency/preference mismatch for three-level complex NPs in English. Both findings have since been found to be consistent with finer-grained analysis of corpus frequency and follow-up experiments, by Desmet et al. (2002, 2005) and Desmet and Gibson (2003) respectively.

allocated more probability mass to the more frequent attachment. A continuing word consistent only with the less frequent attachment will be more surprising than one consistent only with the more frequent attachment. The surprisal model makes predictions potentially divergent from some locality-based models, but in exactly the same way as the Tuning Hypothesis.

Head-final dependencies, in contrast, turn out to be a rich source of divergence between the predictions of DLT-type and surprisal theories. There are a variety of syntactic circumstances in which a comprehender knows that a final governing category has to appear, but does not know exactly when it will appear, or what it will be. This happens when nouns in English are premodified: *the big fuzzy...* is the start of a noun phrase, and the next word *might* be the main noun, but it might be another adjective, and in any event we don't know *which word* the main noun eventually will be. The situation is more extreme in the typologically common configuration of obligatorally verb-final clauses, such as in German, Japanese, or Hindi, where the preverbal content of the clause is often quite long. As Konieczny (2000) points out for verb-final German clauses, the DLT predicts in these cases that a larger number of left dependents will cause greater processing difficulty at the final governor, because all the left dependents must be integrated with it at the same time (c.f. head-initial dependency, where each dependent is integrated as it is seen, one at a time). But the surprisal theory makes the *opposite* prediction in this case. The more dependents we have seen, the more information we have about their governor, and in general the more information we have, the more accurately we should be able to predict that governor's location and identity.[9] Experiments relevant to this divergence in prediction have been carried out by Konieczny (2000); Konieczny and Döring (2003); Vasishth (2002), and annotated data are available to construct surprisal models for German. In Sections 2.4 and 2.5 I construct models of surprisal in German verb-final clauses, showing that predictions of the surprisal theory closely

[9] Konieczny informally makes a similar point: extra dependents can help us narrow down the class of events that a final verb might denote, and therefore aid in lexical access. The surprisal theory encompasses this position, which involves prediction of the *identity* of the item ending the clause, but is more general, as it includes predictions about the *position* of the end of the clause. See discussion of Jäger et al. (2005) in Section 3.8 for evidence that humans make accurate, syntactically-driven positional predictions consistent with the surprisal model.

match qualitative reading-time patterns.

DLT-style and AFH-style theories make similar predictions regarding long-distance dependencies that violate minimal locality. The classic case of such a long-distance dependency is a leftward extraction that could potentially originate in one of a number of extraction sites, and turns out *not* to originate in the leftmost site, such as an object relativization or an embedded *wh-* relativization:

(5) a. The reporter *who* * the senator attacked admitted the error. (Gibson, 1998)

b. *Who* did the senator hope * the reporter would recognize?

In each of the examples, a local ambiguity arises from the fact that the italicized extract could originate from the starred location, which is the leftmost extraction site possible, but turns out not to. In the DLT and AFH, this causes greater difficulty (as compared to an extraction from the starred location) in between the leftmost possible and the true extraction sites—in the AFH due to the cost of backtracking, in the DLT due to the memory cost of maintaining the extracted element longer plus the final cost of a longer-distance integration. These predicted asymmetries have been confirmed (Gibson et al., 2005a).

In the case above, surprisal-based processing predicts the same general asymmetric difficulty, but for a different reason: in the above examples, extractions from the leftmost site are more common than more distant extractions.[10] Using a PCFG with empirically-estimated rule frequencies, Hale (2001) showed that the surprisal of object-extracted relative clauses such as (5a) exceeds that of subject-extracted relative clauses. The difference in surprisal follows directly from the fact that the

[10] Proponents of locality-based complexity in comprehension have argued that these frequency asymmetries should be viewed as derivative of, rather than causing, asymmetries in comprehension difficulty (see Gibson 1998, p. 59, Gibson and Pearlmutter 1994 as examples), implicitly contingent on the presupposition of altruistic speakers. Although this position may seeem attractive in the present context, carried out logically it also leads to problematic conclusions such as that the typologically common head-final word order should be rare. On a frequency-sensitive view of comprehension, locality preferences in grammars can still be reasonably ascribed to production pressures. Finally, the surprisal theory of comprehension *does* have non-trivial implications regarding comprehension-optimal grammars; I take up this issue in Section 2.8.3.

PCFG rule probability for the subject extraction (6a) below is much higher than that for the object extraction (6a).[11]

(6) a. S/NP → VP
    b. S/NP → NP VP/NP

A rigorous test of diverging predictions between locality and surprisal processing theories in the case of long-distance dependencies would be possible in a situation where the the most common origin site of a leftward-extracted element is *not* the leftmost site possible. In this situation, the divergence would emerge that in (appropriately controlled) contrasts between leftmost-site and most-common site extractions, locality predicts greater difficulty for most-common site extractions whereas surprisal predicts greater difficulty for leftmost-site extraction. Unfortunately, I am not aware of a clear-cut case where this mismatch between the locality and frequency of extraction site holds. Nevertheless, more fine-grained patterns within the canonical cases of long-distance extraction have been studied, varying the syntactic and semantic properties of extracted elements, their governors, and intervening material. One such study, that by Schlesewsky et al. (2000), proposes a syntactic treatment of ambiguity resolution in German finite-clause word order as involving non-local dependency, together with an AFH theory of processing patterns. In Section 2.6, I present a detailed analysis of experiments that Schlesewsky et al. argue undermine serial frequency-based processing accounts, and show that surprisal actually models these experiments more precisely than the AFH itself. This analysis highlights some of the differences between predictions in cases of ambiguity resolution of the parallel surprisal theory and serial theories based on both locality and frequency. In addition, a number of recent studies (Gordon et al., 2004; Grodner and Gibson, 2005; Jäger et al., 2005) present detailed experimental results on English relative clause reading times that are highly relevant to locality/surprisal contrasts. I discuss these results and their implications for locality and surprisal in Section 3.8.

[11] The "/NP" part of the rule is a Generalized Phrase Structure Grammar (Gazdar et al., 1985) style notation indicating the presence of a long-distance extracted element, in this case the relative pronoun. Although the form of the rules here is slightly different than that presented by Hale 2001, the generalization is exactly the same.

2.4 Konieczny's verb-final clause results

In German, nonfinite verbs (as well as finite verbs in embedded clauses) are clause-final. Unlike in English, therefore, when a verb is encountered, the number and distance of previous dependents can vary widely. Konieczny (2000) investigated the effect of this variation on processing difficulty in a self-paced reading time study, measuring reading time at clause-final verb in transitive German embedded clauses where the amount and type of material between the direct object and the final verb varied (7),(8).[12]

(7) a. Er hat die Rose HINGELEGT, und ...
he has the rose laid_down, and ...

b. Er hat die Rose **auf den Tisch** GELEGT, und ...
he has the rose on the table laid, and ...

c. Er hat die Rose **auf den kleinen runden Tisch** GELEGT, und ...
he has the rose on the small round table laid, and ...

(8) a. Er hat die Rose, *die wunderschön war,* HINGELEGT, und ...
he has the rose, that wonderful was, laid_down, and ...

b. Er hat die Rose, *die wunderschön war,* **auf den Tisch** GELEGT, und
he has the rose, that wonderful was, on the table laid, and
...
...

c. Er hat die Rose, *die wunderschön war,*
he has the rose, that wonderful was,
auf den kleinen runden Tisch GELEGT, und ...
on the small round table laid, and ...

In (7a) the verb directly follows the direct object; in (7b)-(7c) a prepositional phrase goal of varying size intervenes between the direct object and the verb; in (8a) the direct object is postmodified by an *in-situ* relative clause, after which the verb immediately appears; and in (8b)-(8c) both a relative clause and a varying-length

[12] Konieczny also varied the length of the relative clause, but the effect of this variation on reading time was not statistically sigificant and is irrelevant to the modeling in this section, so I ignore it for the moment.

PP intervene before the final verb. From a locality-based perspective the predictions are clear: the verb should be easiest to process in (7a), because it has the fewest and nearest dependents; and hardest to process in (8c), because it has the most and farthest dependents.

But Konieczny found the opposite pattern: the verb was processed the fastest in (8c) and slowest in (7a) (see Figure 2.1). From the perspective of locality-based processing theories where memory limitations are the main source of processing difficulty, this result is very surprising. The result is consistent, however, with the perspective of expectation-based processing. The more dependents we have seen, the more information we have about their governor, and in general the more information we have, the more accurately we should be able to predict that governor's location and identity. In order to quantify this effect in terms of surprisal we need only provide an appropriately formulated probabilistic language model. I now proceed to provide such a model and analyze its predictions with respect to Konieczny's experimental data.

2.4.1 An expectation-based analysis

In order to determine the predictions of surprisal-based sentence processing on Konieczny's data, it is necessary to choose a probabilistic language $p_i(w)$. The choice of model should be driven by our linking hyphothesis between incremental comprehension and difficulty: the model chosen as optimal for purposes of incremental processing and disambiguation should accurately predict per-word reading times. The principle of likelihood maximization seems appropriate here: models that give high likelihood to data are generally good for making inferences about that data. The state of the art in language models that give high likelihood to naturally-distributed data is, however, generally achieved through n-grams (although Charniak 2001 gives intriguing results for high-likelihood language models using a lexicalized PCFG parser). In our case, our data—the experimental stimuli used in reading-time experiments—do not follow natural distributions, but rather involve a single sharp contrast in one dimension of syntactic variation, while minimizing other lexical and structural variation. In particular, it will become clear that sensitivity to the *constituent history* of the sentence is crucial in determining expectations, so a grammar-sensitive conditional word model makes sense.

For this reason, I use a probabilistic context-free grammar (PCFG) of German to construct a language model. Given a PCFG, existing algorithms by Jelinek and Lafferty (1991) and Stolcke (1995) show us how to calculate the *prefix probability* of a string: the total probability of all trees (or strings) consistent with that prefix. As pointed out by Hale (2001), the conditional probability of w_i is then simply the ratio of the prefix probabilities of $w_{1 \cdots i-1}$ and $w_{1 \cdots i}$. I take advantage of the hand-parsed NEGRA corpus (Skut et al., 1997a), and use essentially the grammar read straight off the parsed corpus to construct a language model, making only minimal changes to the corpus necessary to reflect the crucial, coarse-grained dimensions of variation relevant in each experiment. In all cases I use relative-frequency (maximum-likelihood) estimation of rule probabilities.

Prepositional Phrases

Konieczny's prepositional-phrase results turn out quite nicely in the surprisal-based model. In addition to a small number of modifications of the NEGRA corpus,[13] we distinguish between *large* and *small* PP categories so that the difference in Konieczny's experimental stimuli are directly reflected in the PCFG. Calculating the final-verb surprisals of one of Konieczny's stimulus sets depicted in (9) below and comparing it to reported mean reading times, we get the graph in Figure 2.1.

(9)

a. Er hat den Abgeordneten begleitet, und …
He has the delegate escorted, and …
"He escorted the delegate, and …"

[13] In order to sharpen the PCFG's distributional knowledge of V2 versus verb-final contexts, I introduced syntactic distinctions in the VP and CVP (coordinated VP) NEGRA syntactic categories based on whether the clause was matrix or subordinate. Subordinate-clause VPs were defined as those under an S category and sister to a PRELS tag, which is the NEGRA syntactic category for relative pronouns.

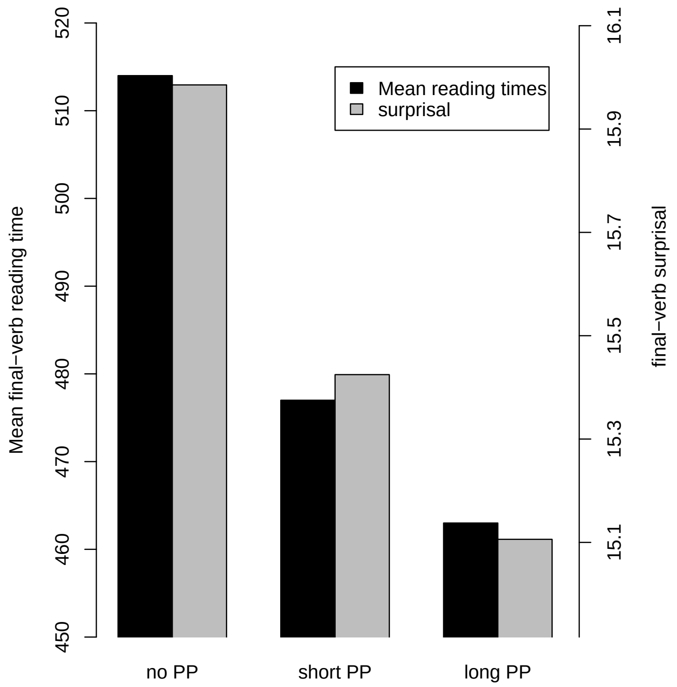


Figure 2.1: Empirical reading time versus log-probabilities at clause-final verb

b. Er hat den Abgeordneten ans Rednerpult begleitet, und ...
He has the delegate to_the lectern escorted, and ...
"He escorted the delegate to the lectern, and ..."

c. Er hat den Abgeordneten an das große Rednerpunt begleitet, und ...
He has the delegate to the big lectern escorted, and ...
"He escorted the delegate to the large lectern, and ..."

The log-probability pattern matches the reading-time probability quite closely. The DLT, in contrast, predicts the wrong monotonicity of reading difficulty.[14]

[14] Konieczny used a variety of experimental stimuli, of course, and the final-verb surprisal values differ for each stimulus. Although for every stimulus the monotonicity of surprisal is correct in the contrast between presence and absence of PPs, the small/large PP contrast does not always have the correct monotonicity. As can be seen in Figure 2.1, of course, the largest quantitative contrast is between presence and absence of PPs.

The reason the PCFG-derived surprisal values match the empirical results so well is that incremental parsing with a PCFG naturally captures the effect of a sentence's *constituent history* on the expectations regarding yet-to-be-seen input. As soon as the comprehender knows that the input is part of a verb-final clause, the incremental probabilistic parsing process implicitly determines set of expectations as to the next constituent. Each subsequent constituent affects these expectations. To a first approximation, seeing a constituent of a given type (a subject, a direct object, the final verb, a goal, a location, and so on) sharply decreases the expectation of seeing another constituent of the same type in the same clause, because multiple constituents of a single type rarely co-occur in a single clause; this is part of the comprehender's knowledge of linguistic argument structure, captured in the PCFG model by the structure of rewrite rules. When a PP goal is actually seen in the input, as in 2, the expectation allocated to seeing a PP goal is pruned away, and because expectation is actually a probability distribution that must sum to 1 at all times, it is reallocated among all the other types of constituents that have not yet been seen. The final verb, being one of those constituents, therefore has its expectation increased after every other constituent. In another manner of speaking, the comprehender's expectation as to the *location* of the final verb sharpens as the clause lengthens. The way this incremental expectation-narrowing process plays out in a PCFG-derived probabilistic string model is illustrated in Figure 2.2: as each constituent of a given category is seen and integrated into the incremental parse, it eliminates most of the expectation for seeing another constituent of the same type next, and as a result increases the expectation for seeing a constituent of one of the remaining types.[15]

What this model does *not* capture is the incremental sharpening of the comprehender's expectations as to the *identity* of the final verb. As Konieczny himself points out, seeing a goal PP restricts the syntactic/semantic classes from which the final verb

[15] Technically, the PCFG used to model this experiment does not distinguish goal PPs from other types of PPs, because the NEGRA corpus unfortunately does not make this distinction. The PP category in Figure 2.2 is therefore not subdivided. Nevertheless, the same intuitive argument holds for this cruder grammatical model, because PPs in general are in complementary distribution with each other in verb-final contexts.

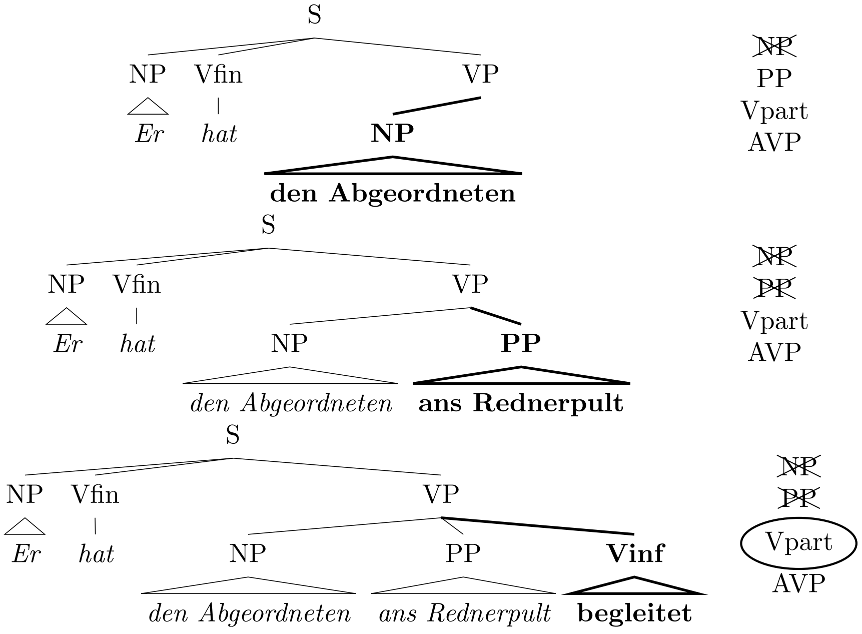


Figure 2.2: Incremental parse of (9), showing incremental narrowing of next-constituent syntactic expectations

can plausibly originate. The PCFG model I use here does not capture that information, because it is not *lexicalized*; the dependents of the verb and the constituency of its syntactic projection are not probabilistically conditioned on the verb's actual identity. This is not a limitation of the general surprisal approach, nor of PCFG grammars; it is well-understood in the computational linguistics literature how to conditionalize dependency and consituency on lexical govenors (see Collins 1999 for an influential treatment). However, the relatively small size of the NEGRA corpus and the morphological complexity of the language have put reliable lexicalization of German PCFG grammars out of reach for the moment (Dubey and Keller, 2003). In addition, the analysis here shows that expectations about verb identity are not strictly necessary to explain the observed pattern of final-verb reading time results; expectations about verb location are sufficient. Section 2.7.2 revisits the question of final-verb identity versus location in the context of English subject-modifying relative clauses.

	RC present	RC absent
Independence between NP and VP expansions	13.23	12.95
Presence of RC in NP affects VP expansion	13.18	14.32
mean empirical Reading Time (ms)	526	476

Table 2.1: Surprisal values at final verb when immediately preverbal dependent is NP object, with and without *in-situ* object-modifying relative clause.

Relative Clauses

Intuitively, a similar type of reasoning should apply to the effect of relative clauses as to prepositional phrases. Immediately after seeing an object noun, the human parser has a set of expectations about whether the object NP is complete or not, and correspondingly of whether the next word is part of the object NP or part of the VP. After seeing an in-situ RC, in contrast, the human parser should lower its expectation of seeing more object NP material, because empirically in-situ RCs almost always end their NP.[16] Since the expectation of seeing VP material is higher after seeing the RC, then the expectation of seeing a clause-final verb should in turn be higher, all else being equal.

The crucial question is whether, in fact, all else is equal. As it turns out, whether or not a probabilistic independence assumption is made between VP and NP rewrites turns out to affect incremental expectations because of relative clause extraposition in German. The first line of Table 2.1 shows final-verb suprisal values if such an independence assumption is made; the results are monotonically consistent with the empirical results, as predicted by the reasoning of the preceding paragraph. However, it is fairly well-known (Hawkins, 1994; Uszkoreit et al., 1998) that relative clause extraposition is sensitive to constituent size and position, with extraposition being more likely when the distance to the end of the clause is smaller. In particular, a German relative clauses nearly always extraposes if the NP it modifies is immediately preverbal. As a result, in all three corpora of German, having seen a relative clause modifying an object NP drastically *decreases* the conditional likelihood of seeing a verb next, as can be seen in the left half of Table 2.2. (As seen in the right half

[16] There will be some expectation that the RC itself is not ended and may be continued with, for example, an S coordination, but this expectation should be relatively small.

	after sequence NP RC				after sequence NP RC PP			
	with RC		without RC		with RC		without RC	
	N	%	*N*	%	*N*	%	*N*	%
NEGRA	39	2.6	3759	60.7	10	70.0	649	77.2
TIGER	89	0.0	8837	60.8	23	78.3	1493	78.6
TüBa-D/Z	60	1.7	14208	39.2	15	60.0	3059	62.6

Table 2.2: Frequency in German corpora of clause-final verb immediately after (post-finite verb) constituent sequences of (i) NP RC; and (ii) NP RC PP. The PP in context (ii) is required to be a verbal modifier. *N* is the number of existing contexts; % is the proportion of contexts immediately followed by the clause-final verb.

of Table 2.2, however, the effect disappears when a verbal PP follows the RC.) As a result, if comprehenders include weight effects in their judgements of syntactic felicity, as seems plausible, it has the effect of causing greater surprisal at the final verb after the sequence NP RC than after the sequence NP, and little to no effect on surprisal from *in-situ* RC presence if a verbal PP separates the object from the final verb.[17] The second line of Table 2.1 shows final-verb surprisal values when clausal constituency is made probabilistically dependent on the presence or absence of a relative clause modifiying an NP dependent. This time, the predictions of surprisal contravene the empirical reading-time results.[18]

These modeling results leave us in a somewhat uncertain position as to how to interpret the empirical finding that insertion of an *in-situ* relative clause modifying an immediately preverbal object decreases final-verb reading time. On the one hand, this finding unequivocally contravenes the DLT, which predicts that the discourse entities inside the RC will increase the integration cost of the preceding verbal dependents with the final verb. One way of interpreting the mixed predictions of surprisal is that the empirical RC result provides evidence that the human parser relies on coarse-grained rather than fine-grained statistics (Mitchell et al., 1995), and hence

[17] The technical means of incorporating the relevant weight effect simply involves distinguishing NPs with RC modifiers from those without modifiers in the underlying context-free grammar of German.

[18] The fact that the surprisal difference in the second line of Table 2.1 is more moderate than the frequency difference seen in Table 2.2 is likely due to the fact that the presence of the relative clause still signals the likely end of the NP, information that the comprehender does not have in the stimulus where the relative clause is absent.

incorporates a probabilistic independence assumption between the immediate constituent structure of the VP and the internal structure of its NP daughter. With such a coarse-grained probabilistic grammar, the final verb's surprisal value would be reduced by the object-modifying RC.

This interpretation is conceptually problematic, however. The felicity judgements of native speakers makes it clear that they are highly sensitive to weight effects in comprehension; the strong preference to interpret a right-extraposed element as modifying the rightmost NP eligible to be the extraposed element's origin also indicates that weight effects are involved in disambiguation. If weight effects *are* involved in probabilistic disambiguation, then they must also impact the surprisal values, which we originally derived in Section 2.2 as the result of incremental structural disambiguation.[19] Weight-sensitive surprisal predicts the same result as the DLT: RC insertion should hinder the reading of an immediately following verb. The empirical evidence therefore contravenes *both* theories.

This puzzling result creates a degree of uncertainty as to how to interpret the experimental results of Konieczny (2000). One possibility is that some or all of the final-verb results are confounded by sentence length, and that the only real result is that verbs preceded by more words are being read more quickly.[20] This interpretation would call into question the pretheoretical analysis underpinning the surprisal explanation of verb-final clause difficulty. Fortunately, Konieczny and Döring (2003) carried out another experiment free of potential sentence length confounds replicating the final-verb speedup effect in clauses with more dependents, which I analyze in the next section. Similar results have also been obtained for Hindi by Vasishth and Lewis (2003) and for Japanese by Gibson et al. (2005b). In contrast, no attempts have been made to replicate the post-relative clause speedup effect; since it contravenes the predictions of both locality- and expectation-based processing theories, it would certainly be worthwhile to attempt such a replication in future experimental research.

[19] See Chapter 4 for a more extensive investigation of how weight effects for extraposition can be embedded directly into stochastic generative grammatical formalisms.

[20] As Konieczny points out, he also recorded reading times of relative pronouns in extraposed relative clauses, which turn out to be read significantly more slowly than those of in-situ RCs, so not all reading time results in this experiment are entirely driven by string position.

2.5 Final verbs: effect of preverbal NP type

Konieczny and Döring (2003) report a variant of Konieczny (2000)'s original experiment, where the syntactic position of a preverbal NP, rather than the presence/absence of a preverbal PP, is varied:

(10) a. Die Einsicht, daß $[_{\text{NP}_{\text{NOM}}}$ der Freund] $[_{\text{NP}_{\text{DAT}}}$ dem Kunden] $[_{\text{NP}_{\text{ACC}}}$ das
the insight, that the friend the client the
Auto aus Plastik] verkaufte, ...
car from plastic bought, ...
"The insight that the friend bought the client the plastic car ..."

b. Die Einsicht, daß $[_{\text{NP}_{\text{NOM}}}$ der Freund $[_{\text{NP}_{\text{GEN}}}$ des Kunden]] $[_{\text{NP}_{\text{ACC}}}$ das
the insight, that the friend the client the
Auto aus Plastik] verkaufte, ...
car from plastic bought, ...
"The insight that the friend of the client bought the plastic car ..."

Konieczny and Döring found that reading time at the final verb *verkaufte* was significantly shorter for the dative condition, where *dem Kunden* is dependent on the final verb, than for the genitive condition, where *des Kundes* is dependent on the preceding noun *Freund*.[21] This study is a nice methodological confirmation of the original pattern observed in Konieczny (2000). The stimuli in (10) differ in only a single letter, thus controlling quite precisely for orthographic length, number of tokens, and also, as it turns out, word freqency (*dem* and *des* are quite close in overall word frequency, with *des* perhaps slightly more frequent in some contexts).

Intuitively, surprisal applies just as readily to this experiment as to Konieczny's original experiment. Just before seeing the final verb in (10a), the comprehender knows that nominative, accusative, and dative NP arguments have all appeared as preverbal dependents; in (10b), only nominative and accusative preverbal dependents have appeared. The comprehender's expectations are therefore more narrowly focused

[21] They also varied whether the immediately preverbal PP was a nominal dependent, as in *aus Plastik* in (10), or a verbal dependent such as *aus Freude*. Although they found slightly shorter average reading time for the nominal-dependent case, this difference was not statistically significant. If subsequent studies were to achieve a stastically significant result favoring faster reading times for the nominal-dependent condition, then it could be problematic for the expectation-based account presented here.

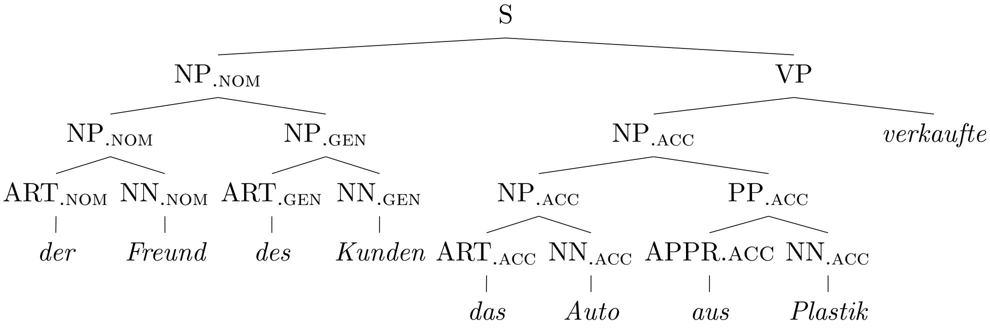


Figure 2.3: Case-percolated context-free grammar

in (10a), and so the surprisal at the final verb should be lower. In order to precisely model this effect of constituent history, we can use a PCFG grammar to determine a conditional word model as we did in Section 2.4. For this new experiment, however, we need to connect the grammatical function of each NP with its morphological form to obtain a better understanding of the effect of constituent history on expectations. In the PCFG grammars we used in Section 2.4, NP case/grammatical function was not incorporated into the syntactic component of the grammar; rewrites such as S →NP Vfin VP did not distinguish between accusative and nominative, or object and subject, NPs.

For a PCFG to induce a probabilistic language that reflects ordering tendencies for different grammatical functions or cases, however, we must include such information in the PCFG's grammatical backbone. The order of NP arguments is also connected with number marking, of course, since subject-verb agreement can constrain possible syntactic configurations. We can do this by deriving our grammar from the portion of NEGRA (about 100,000 words) that is morphologically annotated for case, and using a simple grammatical rule, similar to constraints used in unification-based formalisms such as Functional Unification Grammar, Head-Driven Phrase Structure Grammar, and Lexical-Functional Grammar, to distribute case marking. We recursively percolate case marking onto NPs and PPs from their head daughters (the case of a preposition is considered to be the case it governs). Doing this incorporates knowledge of the overall distribution of *argument realization frames* into the PCFG, capturing major distributional facts such as the rareness of multiple dative NPs in a

	Reading time (ms)	$P_{i-1}(w_i)$	DLT prediction
verbal dependent (dative)	555	8.38×10^{-8}	slower
nominal dependent (genitive)	793	6.35×10^{-8}	faster

Table 2.3: Reading time, conditional probability, and DLT predictions at final verb for (10)

single clause. An example tree resulting from case-percolation is shown in Figure 2.3. We then use these case-enriched symbols as atomic categories, and learn a PCFG via relative-frequency estimation from the enriched corpus.[22]

Table 2.3 shows empirical reading times, conditional word probabilities, and DLT-predicted reading times for the dative and genitive conditions of (10). Although the conditional probability of the final verb is quite low in both conditions, it is roughly 30% higher in the verbal-dependent condition than in the nominal-dependent condition, correctly predicting reading time monotonicity. DLT, on the other hand, predicts faster reading time for the nominal-dependent condition, since there are fewer preverbal dependents for the verb to integrate with.

Konieczny and Döring do in fact present a constraint-based computational model in the form of a simple recurrent network (SRN; Elman 1990) that captures probabilistic dependencies between specific verbs and their host of dependents. Their SRN, trained on an artificially generated corpus consisting of both transitive and ditransitive verbs, is able to model their experimental results by virtue of the fact that the presence of a preverbal dative argument excludes simple transitive final verbs from the space of possible final verbs, hence boosting the expectation for each individual ditransitive verb. That is, their model matches the empirical data by virtue of its knowledge of *probabilistic lexical selection preferences.* It must be emphasized that the

[22] Although the choice to percolate case marking onto phrasal categories might at first glance seem *ad hoc*, it is well motivated due to the fact that we have derived the surprisal theory as a consequence of incremental probabilistic disambiguation. If, for example, the probability of syntactic trees for OVS versus SVO word order in simple German matrix clauses (which we will see in Section 2.6) were not sensitive to the case-marking of the NPs, it would be impossible for comprehenders to globally preferentially disambiguate sentences with case-syncretized NPs of equal animacy as SVO. But native German speakers do exactly this. Within the PCFG framework, capturing this disambiguation preference requires percolating case marking onto NP categories.

model I present here encompasses but is more general than such an account: in principle, knowledge of probabilistic lexical selection preferences will affect the expectation for individual verbs, and this can be captured with a probabilistic parser. In practice, however, the reliable estimation of lexical selection preferences is in its own right a challenging and salient problem in natural language processing research (Roland and Jurafsky, 2002). I have not attempted to incorporate such information into the model here, because individual verbforms are simply too sparse and the resulting distributuional patterns of *monolexical dependency* (between words and categories) would be unreliable. I have, however, shown that lexical selection preferences—determining expectations about *which* verbs may appear—are not critical to modeling the results of Konieczny (2000) and Konieczny and Döring (2003); the results can be captured using solely expectations about *where* the final verb is likely to appear. Konieczny's lexical expectation model and the more general surprisal model do differ in the scope of their empirical predictions; I revisit this difference in Section 2.7.2, where I show that predictions made only by the surprisal theory are in fact borne out.

2.6 The subject preference

A second set of experimental results involving German word order and relevant to the contrast between locality- and surprisal- based theories is reported by Schlesewsky et al. (2000) involving the subject preference in German. German main-clause NPs can be freely reordered, and while case marking usually disambiguates their grammatical function, some NPs are completely syncretized between multiple cases. When the clause-initial NP is syncretized between nominative and accusative case, there is a temporary ambiguity between subject and object interpretations of that initial NP, as in (11). Hemforth (1993) demonstrated that the default interpretation of such an NP is as the subject. When case marking on the NP immediately following the finite verb disambiguates the clause, as in (11) below, the marking pattern consistent with a subject interpretation for the sentence-initial NP ((11a)) will be easier to read than the pattern consistent with an object interpretation for the sentence-initial NP ((11b)).

(11) a. die Henne sieht den Bussard
the hen$_{\text{NOM/ACC}}$ sees the$_{\text{ACC}}$ buzzard
"The hen sees the vulture."

b. die Henne sieht der Bussard
the hen$_{\text{NOM/ACC}}$ sees the$_{\text{NOM}}$ buzzard
"The vulture sees the hen."

In investigating the nature of this subject preference, Schlesewsky et al. point out that the unmarked word order for German is widely recognized to be sentence-initial, which leads to at least two straightforward theoretical accounts of the subject preference. On the one hand, in a movement-based syntactic account, the underlying syntactic structure of the German clause can be posited to be SOV, and declarative clause order may be derived by movement of the finite verb and an argument NP to the head and specifier positions of CP, respectively. Assume further the Active Filler Hypothesis (Clifton and Frazier, 1989), which states that once the human parser has identified a filler (such as an initial argument NP in German declarative clauses), it prefers to posit a gap and insert the filler as soon as possible. In (11), as soon as the parser has seen the finite verb *sieht*, it can posit an immediately following gap in the subject position and resolve it with the sentence-initial filler, as in Figure 2.4. Under this preferred parse, however, the next NP cannot be nominatively marked, so (11b) will cause processing difficulty.

Schlesewsky et al. also consider a serial construction-frequency account along the lines of the Tuning Hypothesis (Mitchell and Cuetos, 1991). They point out that most declarative clauses are subject-initial—consistent with the notion that the unmarked word order in German is subject-inital—even when the initial element is case-syncretized. If the parser immediately committed to the most frequent interpretation of the initial NP, then (11b) would require reanalysis (and therefore extra processing time) at the postverbal NP.

Schlesewsky et al. present two experiments which, however, they suggest undermine a serial Tuning Hypothesis account of subject preference, involving inanimate, case-syncretized NPs. They report that of 480 sentence-initial *was* 'what' items randomly selected from the Freiburg Corpus, about 55% were accusative, suggesting

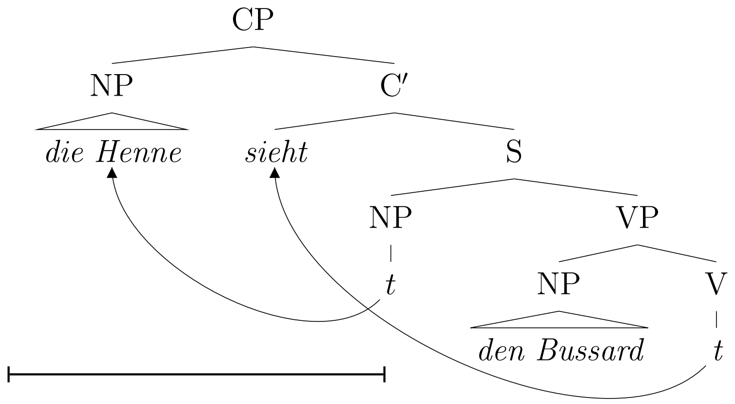


Figure 2.4: Schlesewsky et al. (2000)'s Subject Preference in German declarative clauses (Example (11a)), as derived by the AFH and a movement-based analysis of German clause order. For Example (11b), *Bussard* has the nominative article *der*, and the greedy assignment of *sieht* to the V gap creates a case marking conflict.

that frequency-based considerations should not favor a default subject interpretation for inanimate case-syncretized initial NPs. The AFH, in contrast, predicts the same default subject preference regardless of construction frequency. Both experiments involve singular neuter (and hence nom/acc syncretised) sentence-initial WH-NP's. In one experiment, disambiguation occurs via case marking on the postverbal NP (12); in the other, disambiguation involves agreement marking on the main verb (13).

(12) a. was || erfordete || **den Einbruch** || in die Nationalbank || ?
what required the.ACCbreakin into the national_bank ?
"What required the break-in into the national bank?"

b. was || erfordete || **der** Einbruch || in die
what required the$_{\text{.NOM}}$breakin into the national_bank
Nationalbank || ?
?
"What did the break-in into the national bank require?"

(13) a. welches System || **unterstützt** || die Programme || auf den Computer
which system supports the programs on the computer
|| ?
?
"Which system supports the programs on the computer?"

b. welches System || **unterstützen** || die Programme || auf den
which system support the programs on the
Computer || ?
computer ?
“Which system do the programs on the computer support?”

In the first experiment’s stimuli, shown in (12), verbal agreement is singular, meaning that the verb is compatible with either a subject or object reading for sentence-initial *was*. The case marking on the immediate postverbal NP is unambiguous, however, and disambiguates the grammatical function of *was*. In a serial parsing model, a default preference for subject interpretation of the sentence-initial NP would predict greater processing difficulty at the postverbal NP in the *der* condition. In the experiment, Schlesewsky et al. (2000) indeed found significantly higher reading time for the *der* condition, but at the postverbal NP it was small and statistically insignificant; it reached significance (as well as its largest numerical difference) at the postmodifying PP.

In the second experiment, shown in (13), verbal agreement in the plural *unterstützen* condition disambiguates the grammatical function of the sentence-initial singular neuter NP *welches System*; in the singular *unterstützt* condition, disambiguation occurs at the (nom/acc syncretized) postverbal NP, which being plural cannot be the subject of a singular verb. In a serial parsing model, a default subkect intepretation preference for the initial NP predicts greater processing difficulty at the main verb in the *unterstützen* condition. Schlesewsky et al. (2000) did find higher reading time for at the main verb for this condition, and increased reading time in this condition persisted throughout the rest of the sentence.

The case of Schlesewsky et al. (2000) against a serial construction-frequency account rests on their claim that inanimate case-syncretized initial NPs are not more frequently subjects than objects, the result they obtained from counts in the Freiburg corpus. Now that syntactically hand-annotated German corpora are readily available, we can try to corroborate this claim. Table 2.4 shows these frequency counts for NEGRA, TIGER, and TüBa-D/Z. Three immediate points stand out. First, the statistics for *was* do not unilaterally support the reported Freiburg counts. For

	was			*welches* + N		
	Subj	Obj	Other	Subj	Obj	Other
NEGRA	43	18	19	0	0	0
TIGER	84	47	23	0	1	0
TüBa-D/Z (Nom/Acc)	40	37	18	1	0	0

Table 2.4: Empirical frequencies of subject and object interpretations of sentence-initial *was* and *welches*

NEGRA and TIGER, which are corpora from the same newspaper, there is a clear trend toward greater frequency of subject for sentence-initial *was*. Second, corpus type seems to have a strong effect on these statistics. For TüBa-D/Z, which is a different (and more colloquially-written) newspaper than NEGRA and TIGER, subject and object are evenly split. Third, statistics for *welches* are extremely scarce: in 1.3 million words of text, we have only two instances of sentence-initial *welches* + N, so we are not in a position to confirm or disconfirm the generality of claims about behavior of sentence-initial *was* to *welches*.

The statistics for *was* suggest that Schlesewsky et al. (2000)'s case against serial construction-frequency accounts for this case of ambiguity resolution is not so cut and dried. We might also ask what the predictions of the surprisal theory are in this case. In the next two sections I proceed to demonstrate that for these experiments on German finite-clause word order, the surprisal account maintains the core ambiguity-resolution intuition of construction-frequency accounts that more probable structural analyses are favored, but the word-by-word difficulty predictions of surprisal are much more clear-cut. As it turns out, the same parallel, expectation-based approach that accounts so well for final verb reading times predicts a better fit to Schlesewsky et al. (2000)'s subject-preference data than either Active Filler or serial tuning-based accounts.

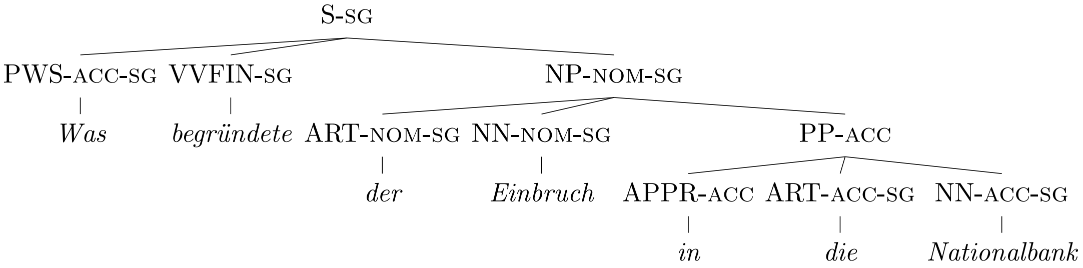


Figure 2.5: Case- and number-percolated CFG for (12)

2.6.1 *was* questions with disambiguating case marking

The order of NP arguments is clearly connected with case; it is also connected with number marking, since subject-verb agreement can constrain possible syntactic configurations. In order to apply a PCFG-derived by-word surprisal models to subject-preference data, we use case percolation as described in Section 2.5 for the morphologically annotated portion of NEGRA. We also percolate number marking onto NPs and Ss from their head daughters (the finite verb directly heads S in NEGRA), and assign plural number marking to all coordinate NPs. Figure 2.5 gives an example of such a percolated tree. As before, we use direct relative-frequency estimation to determine all rule probabilities.

Figure 2.6 plots the differences in the subject-*was* and object-*was* conditions for (a) surprisal, based on the case- and number-percolated PCFG read off the morphologically-annotated portion of NEGRA; and (b) actual mean reading time in (12).[23] The surprisal differential reaches its maximum at the onset of the PP, which is the only region that shows a significant difference in mean reading times.[24]

The reason *why* the surprisal-based account predicts processing difficulty at the PP, where it occurs, turns out to be straightforward. Within the case-marked PCFG

[23] In the simulation I have substituted *begründete* 'caused' for *erforderte* 'required', and *Bank* for *Nationalbank*, because the latter word of each pair does not appear in the morphologically annotated portion of NEGRA. Because phrase structure rewrite probabilities are not conditioned on lexical information, these substitutions have no effect on predictions at the regions of interest.

[24] The reason for the small surprisal differential at the onset of the postverbal NP despite the strong differential frequency of initial-NP grammatical function reported in Table 2.4 appears to be that in object-initial transitive clauses, there is a much stronger tendency for the subject to immediately follow the finite verb than for the object to immediately follow the finite verb in subject-initial clauses.

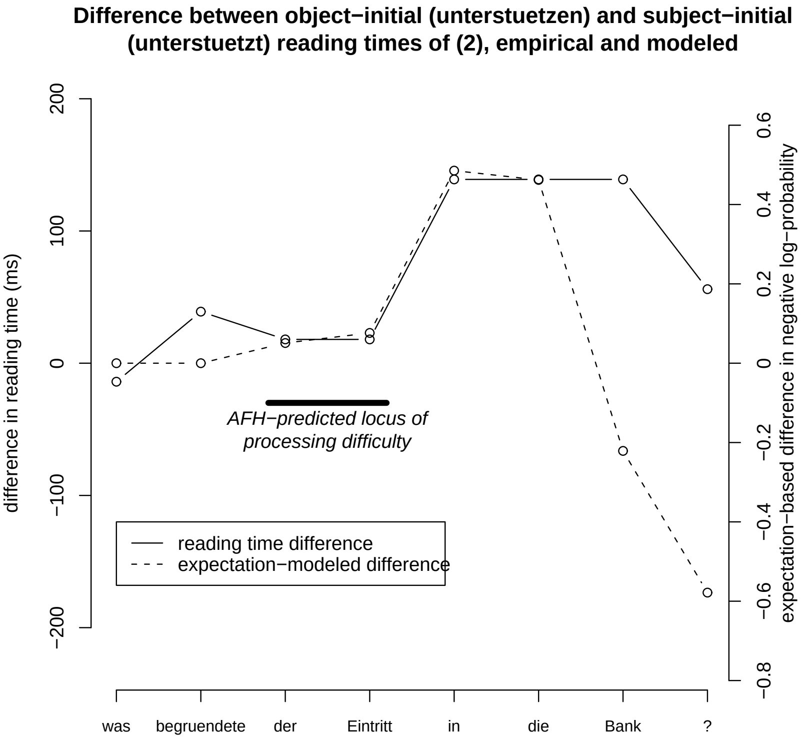


Figure 2.6: Predicted vs. actual reading time differentials for (12)

model, the only difference in the grammatical rules implicitly determining the expectation for the preposition *in* in the postverbal object versus subject case is the PP adjunction rule:

(14) $\mathrm{NP}_{\textsc{nom}} \rightarrow \mathrm{NP}_{\textsc{nom}}\ \mathrm{PP}$

versus

(15) $\mathrm{NP}_{\textsc{acc}} \rightarrow \mathrm{NP}_{\textsc{acc}}\ \mathrm{PP}$

In German, object NPs are more likely than subject NPs to be postmodified by prepositional phrases. As we can see in Table 2.5, this is true not only of subject versus object NPs overall, but also specifically of subject versus object NPs in the

	All				Postverbal			
	Subj		Obj		Subj		Obj	
	N	%	*N*	%	*N*	%	*N*	%
NEGRA	15220	15.3	7952	22.4	2393	12.2	1156	20.3
TIGER	30187	15.2	17490	23.7	4231	12.2	2461	24.4
TüBa-D/Z	20527	6.4	10212	11.3	9054	4.4	4505	9.4

Table 2.5: Frequency of PP modification for subject versus object NPs

immediate postverbal position. This means that the probability of the rule (14) is higher than the probability of (15). In the online comprehension of (12), immediately after hearing *Einbruch* the comprehender therefore has a greater expectation of seeing a PP (and hence a preposition) next in the *den* condition than in the *der* condition. Hence the surprisal at *in* is greater in the *der* condition.[25] Unlike the AFH, surprisal predicts processing difficulty in this experiment precisely where it occurs.

2.6.2 *welches* questions with disambiguating agreement

In the stimuli in 13, the contrasting word of interest is an open-class item whose surface forms, unterstützt and unterstützen, are sparse. In addition, the head noun of the the sentence-initial NP is an open-class word whose relative frequency of occurrence in nominative and accusative forms has a strong effect on the predictions of surprisal predicted by a PCFG. These words are sparse and hence the NEGRA PCFG surprisal estimates are unlikely to be reliable. Nevertheless, we can resort to alternative means of estimating the surprisal at the finite verb.

[25] In principle, the surprisal account should also predict the greater processing difficulty at *die* for the *der* condition. This is because, after seeing *in*, although the comprehender knows she is in the middle of a PP, she does not know whether the PP is verb-modifying or noun-modifying. But because PP modifiers are less likely for postverbal subject NPs than for postverbal object NPs, the comprehender has a greater expectation in the *der* case than in the *den* case that the PP is verb-modifying. The semantics of an accusative *in* PP are, however, compatible only with NP modification in this example, so upon perception of *die* the PP is disambiguated and the comprehender is more surprised in the *der* condition. This explanation, however, relies on much more refined cooccurrence statistics than are actually represented in the PCFG grammar (and there is too little data in NEGRA to achieve this level of fine granularity), so the close match at *die* in 2.6 must be considered a happy coincidence resulting from data sparsity. In addition, the negative surprisal differential at *Bank* is an artifact of data sparsity.

Most straightforwardly, because the region of interest is so close to the beginning of the sentence, we can use the n-gram frequency of the first three words of the sentence to estimate the surprisal at the finite verb directly, given a large corpus of German. A useful estimate of this sort turned out to require using the World Wide Web itself. An exact-match search using Google returned 15 valid matches of the trigram *welches System unterstützt*, many of which were sentence-initial; no instances of the trigram *welches System unterstützen* were found.[26] The direct counting estimate of surprisal at the finite verb therefore predicts the reading-time difference experimentally observed by Schlesewsky et al. (2000).

The Web, however, may not be the most reliable source of n-gram estimates, ended a rate of the direct estimation approach gives us little insight into the reason why the comprehender would have a stronger expectation for a singular finite verb than for a plural finite verb after *Welches System*. We can achieve a bit more insight by decomposing the plausible sources of expectation for the relevant finite verb forms, using grammatical theory and knowledge of frequentistic morphosyntactic distributions. The informal account is as follows. Because our model is *fully parallel*, after seeing the initial NP *welches System* it places a set of expectations on both the grammatical function of the initial NP and the number marking on the finite verb.[27] That is, continuations (a-c) in (16) below all receive some non-zero expectation. (Continuation (16d) receives no expectation, since it violates subject-verb agreement.)

(16) a. [Welches System]$_{\text{SUBJ}}$ singular_verb ...

b. [Welches System]$_{\text{OBJ}}$ singular_verb ...

[26] June 28, 2005, 12:07pm. I discarded one instance of the former trigram that appeared in a web page referencing Schlesewsky et al. (2000).

[27] For simplicity of exposition, we ignore two further recipients of allocated expectation: first, the possibility that the initial NP is not yet complete; second, the possibility that the finite verb is not third-person. Since we are empirically concerned only with contrastive reading times between third-person singular and plural finite verbs, the former recipient does not matter. The latter recipient could turn out to matter, in principle; but as I proceed to show in the rigorous probabilistic decomposition of this section, in practice it does not.

c. [Welches System]$_{\text{OBJ}}$ plural_verb ...

d. *[Welches System]$_{\text{SUBJ}}$ plural_verb ...

In our model, processing difficulty at a given word is associated with the word's *overall* predictability, not with its predictability for a specific partial parse. Crucially, then, the singular-verb *unterstützt* continuation of (13) receives the entire expectation the subject ((16a)) interpretation of the initial NP, and part of the expectation of the object((16b)) interpretations. The plural-verb *unterstützen* continuation, in contrast, receives only part of the expectation of the object interpretation. The parallel surprisal model thus correctly predicts the empirical reading time differences for (13), so long as the combined expectation of (16a) and (16b) together outweighs the expectation of (16c). In the remainder of this section I demonstrate this result with a more rigorous probabilistic decomposition of the relevant expectations.

Probabilistic decomposition of finite verb expectations

The probability of the singular and plural verb forms *unterstützt* and *unterstützen* in (13) can be decomposed as follows (WS standing for *Welches System*):[28]

$$\begin{aligned} P(\text{unterstützt}|\text{WS}) &= P(\text{V.3sg}|\text{WS})P(\text{unterstützt}|\text{V.3sg}, \text{WS}) \\ P(\text{unterstützen}|\text{WS}) &= P(\text{V.3pl}|\text{WS})P(\text{unterstützen}|\text{V.3pl}, \text{WS}) \end{aligned}$$

where V.3sg and V.3pl respectively denote singular and plural third-person finite verbs. This decomposition simply states that the probability of a particular verb-form v given the initial sequence *Welches System* is equal to the probability of a

[28] Although the finite verb forms *unterstützt* and *unterstützen* are compatible with first- and second-person agreement as well, I attend only to third-person agreement because available syntactically-annotated corpora have nearly exclusively third-person subjects. In a model whose parameters more closely reflected speech or another written genre, we might expect $P(\text{V.2}|WS)$ and $P(\text{V.1pl})$, which respectively contribute to the probabilities of *unterstützt* and *unterstützen*, to be substantial.

finite verb of the correct number and person marking given the initial sequence, times the probability that the finite verb is actually v. We can make a crude estimate of the right-hand side of the composition with the simplifying assumption that the conditioning on *Welches System* does not affect the verb's identity.[29] Under this assumption, the ratio of the right-hand half of the decomposition for *unterstützt* versus *unterstützen* turns out to be roughly 1 : 2.3—the forms themselves are roughly equal in frequency (5 versus 6 in NEGRA, 14 versus 11 in TIGER), and singular finite verbs are roughly 2.3 times as common as plural finite verbs (1844 to 789 in the morphologically-annotated part of NEGRA).

The left-hand side of the decomposition can be further subdivided:

$$\begin{aligned} P(\mathrm{V.3sg}|\mathrm{WS}) &= P(\mathrm{SUBJ}|\mathrm{WS})P(\mathrm{V.3sg}|\mathrm{WS},\mathrm{SUBJ}) \\ &\quad +P(\mathrm{OBJ}|\mathrm{WS})P(\mathrm{V.3sg}|\mathrm{WS},\mathrm{OBJ}) \\ P(\mathrm{V.3pl}|\mathrm{WS}) &= P(\mathrm{OBJ}|\mathrm{WS})P(\mathrm{V.3pl}|\mathrm{WS},\mathrm{OBJ}) \end{aligned}$$

where SUBJ and OBJ refer to the event of sentence-initial NP turning out to be the subject or respectively object of the matrix clause. Crucially, the probability of a singular finite verb has two terms summed together on the right-hand side, because the comprehender can derive expectation for finite verbs from both the subject and object interpretations of the initial NP. The probability of a plural finite verb at this point, on the other hand, has only one term, because a plural finite verb *requires* an object interpretation of the initial NP (so $P(\mathrm{V.3pl}|\mathrm{SUBJ}) = 0$). We will use the Freiburg corpus estimates reported by Schlesewsky et al. (2000) at face value for the probablities $P(\mathrm{SUBJ}|\mathrm{WS})$ and $P(\mathrm{OBJ}|\mathrm{WS})$, which are therefore 0.45 and 0.55. For the conditional probabilities of V.3sg and V.3pl, we will make the simplifying assumption of independence between the lexical content of the initial NP and the

[29] If we did not assume independence of the verb form from the lexical content of the initial noun phrase, the effect would most likely be to increase of the conditional probability of *unterstützt* relative to *unterstützen*, because the former is one of presumably a rather narrow range of semantically plausible verbs given *System* as the grammatical subject, whereas *System* as grammatical object is semantically compatible with a wide range of transitive verbs.

category of the subsequent constituent:

$$P(\mathrm{V.3\{sg/pl\}}|\mathrm{WS}, \{\mathrm{SUBJ/OBJ}\}) \approx P(\mathrm{V.3\{sg/pl\}}|\{\mathrm{SUBJ/OBJ}\})$$

These simplified probabilities can be estimated directly from structural counts in the morphologically annotated NEGRA corpus, giving the following estimated probabilities:[30]

$$\begin{aligned} P(\mathrm{V.3sg}|\mathrm{SUBJ}) &= 0.651 \\ P(\mathrm{V.3sg}|\mathrm{OBJ}) &= 0.606 \\ P(\mathrm{V.3pl}|\mathrm{OBJ}) &= 0.152 \end{aligned}$$

The crucial comparison is between the second and third lines: even when the initial NP is an object, the next word is far more often a singular finite verb than it is a plural finite verb.[31] With these probabilities we can now estimate the expectations for singular and plural finite verbs:

$$\begin{aligned} P(\mathrm{V.3sg}|\mathrm{WS}) &= 0.45 \times 0.651 + 0.55 \times 0.606 \\ &= 0.626 \\ P(\mathrm{V.3pl}|\mathrm{WS}) &= 0.55 \times 0.152 \\ &= 0.0836 \end{aligned}$$

The resulting probability ratio, 7.5 : 1 in favor of singular finite verbs, outweighs the 2.3 : 1 ratio we estimated for the probability of the verb form given the number-marked part of speech. Therefore, the surprisal at *unterstützt* is less than the surprisal at *unterstützen*, which is consistent with the empirical reading-time results—even if the relevant initial NPs are in fact more likely to be objects than subjects.

[30] Note that nearly all the verbs in the corpus are third-person.

[31] Neither set of conditional probabilities sums to 1 because the next word following the sentence-initial NP may not yet be the finite verb; evidently this happens more often with initial subjects than with initial objects.

2.7 Discussion and other relevant experiments

The analysis in Section 2.4 demonstrates that the informal explanation that additional preverbal dependents make clause-final verbs easier to predict actually plays out precisely in the surprisal-based model of processing difficulty when a probabilistic context-free grammar is used to estimate the surprisal of the clause-final verb. The surprisal theory encompasses the intuition of Konieczny (2000) explicitly built into the incremental parsing architecture of his dissertation (Konieczny, 1996), that preverbal dependents constrain the lexical type of the final verb and thus allow better prediction of that verb. But the surprisal analysis also shows that this explanation, based on predicting *what the verb is*, is not strictly necessary to explain the experimental results of Konieczny (2000) and Konieczny and Döring (2003). We need only adopt an unlexicalized PCFG—that is, assume that native German speakers are capable of discriminating good from bad constituency structures—for the surprisal model to make qualitatively correct predictions about final verb reading times, which derive from the implicit inferences about *where the verb is likely to be.* In the classic case of verb-final clauses, these two types of inference (the identity versus location of the final verb) are closely bound up with each other, but in Section 2.7.1 I discuss some evidence that they can be separated out, and independent evidence for inferences regarding the location of clause boundaries in syntactic comprehension.

It is also instructive at this point to discuss reading-time results for verb-final clauses in other languages that directly test locality-based processing. (Vasishth 2002 (Chapter 5), 2003, Vasishth and Lewis 2003) have conducted several experiments on processing difficulty within Hindi complement clauses, which are verb-final, varying the amount of material appearing before the final verb, as in (17) below:

(17) a. Siitaa-ne Hari-ko Ravi-ko [kitaab-ko khariid-neko] bol-neko kahaa
Sita-ERG Hari-DAT Ravi-DAT book-ACC buy-INF tell-INF told
'Sita told Hari to tell Ravi to buy the book.'

b. Siitaa-ne Hari-ko Ravi-ko [kitaab-ko **jitne-jaldi-ho-sake**
Sita-ERG Hari-DAT Ravi-DAT book-ACC as-soon-as-possible
khariid-neko] bol-neko kahaa
buy-INF tell-INF told

‘Sita told Hari to tell Ravi to buy the book as soon as possible.’

c. Siitaa-ne Hari-ko Ravi-ko [kitaab-ko **ek baḍhiya dukaan se**
Sita-ERG Hari-DAT Ravi-DAT book-ACC from-a-good-shop
khariid-neko] bol-neko kahaa
buy-INF tell-INF told
‘Sita told Hari to tell Ravi to buy the book from a good shop.’

d. Siitaa-ne Hari-ko Ravi-ko [kitaab-ko **jo-mez-par-thii**
Sita-ERG Hari-DAT Ravi-DAT book-ACC from-a-good-shop
khariid-neko] bol-neko kahaa
buy-INF tell-INF told
‘Sita told Hari to tell Ravi to buy the book that was on a/the table.’

Consistent with the surprisal account, reading time at the innermost verb *khariid-neko* ‘to tell’ is highest in (17a), when there is no adjunct intervening between the head noun of the direct object *kitaab-ko* ‘book’ and its governing verb.[32]

Two other relevant verb-final experiment have been carried out by Nakatani and Gibson (2003) and Gibson et al. (2005b) for Japanese, which is verb-final and has freely reorderable preverbal complements. In both cases, predicted asymmetries in integration cost at final verbs failed to emerge. Gibson et al. (2005b) found patterns similar to those we have already seen in German and Hindi: greater amounts of preverbal material decreased, rather than increased, final-verb reading times.[33] For Nakatani and Gibson (2003), the object of investigation was the degree of center-embeddedness in sentences with multiple sentential complements. They found that the increased complexity of multiply center-embedded sentences showed up at the onset of the most deeply embedded clause—signaled by a third consecutive animate

[32] Unlike the case with the effect of preverbal relative clause seen in Konieczny (2000), the decrease in reading time seen here with a preverbal relative clause is not anomalous, because in Hindi relative clauses cannot right-extrapose.

[33] Unlike the results of Konieczny and Döring (2003), Gibson et al. found no difference in the reading times in contrasts of adverbial versus adnominal positioning of a preverbal constituent. One plausible explanation of the difference in result would be that whereas the adverbial/adnominal alternation used by Konieczny and Döring had considerably different semantic content, and the adnominal element did not provide stage-level information that could be usefully correlated with information about the final verb, Gibson et al. alternated a locative constituent, which even in adnominal position could still facilititate evidential inferences about the final verb.

nominative NP at the beginning of the sentence—rather than at the final, least-embedded main verb, where the integration-cost component of DLT predicts it to occur. In a locality-based processing theory, this result can be explained by a storage cost component, as suggested by Nakatani and Gibson (2003). In a surprisal theory, the natural place to look for an explanation would be to estimate the probability of the conditional probability of a third consecutive animate, nominative NP given two such consecutive sentence-initial NPs. Unfortunately, large annotated corpora of Japanese are not nearly as available as for English or German, so I leave this investigation as a topic of future research.

2.7.1 Relativization revisited

At this point it is appropriate to return to the syntactic configuration most that has been extensively investigated in the context of syntactic processing difficulty: relativization. As I noted in Section 2.3, it is well-understood that, by a variety of measures, object-extracted RCs in English cause more processing difficulty than subject-extracted RCs. In locality-based theories, this is due to the fact that subject but not object relativizations minimize the distance between the extraposition and both the gap and the governing verb. As shown by Hale (2001), surprisal predicts the same general asymmetry due to the fact that object RCs are less common than subject RCs. However, different theories disagree on exactly *where* the increased difficulty of object RCs is predicted to occur, and more recent studies, however, have begun to address this issue by looking at word-by-word reading time patterns in greater detail.

To begin the analysis, note that the integration-cost component of the DLT predicts that it is the RC verb that will be harder to read in object extractions than in subject extractions, because the verb (and the immediately postverbal gap) is where the extra integration cost is paid. Within the surprisal theory, on the other hand, a relative pronoun triggers a syntactic environment much like a verb-final clause: the comprehender knows that the RC's verb must appear at some point, but is uncertain as to what it is and whether a subject will precede it. Surprisal therefore predicts that subject relativizations should be read more *slowly* than object relativizations at the

RC verb. The cost of low expectation for object RCs should be paid at the embedded subject, which is where the bulk of the expectation (devoted to seeing a subject-extracted RC) is pruned away.[34] But the empirical evidence in this case seems to side with locality over surprisal. Grodner et al. (2000) show that for stimuli of the form in (18) below, there is a marked increase in the reading time at the embedded verb *sent* for the object over the subject relativization. The embedded subject in (18b), *the photographer*, is read quickly (see Appendix B of Grodner and Gibson (2005) for word-by-word reading times):

(18) a. The reporter who sent the photographer to the editor hoped for a good story.
b. The reporter who the photographer sent to the editor hoped for a good story.

One possible interpretation of this result within the surprisal theory would be that the observed slowdown at the main verb is a *spillover* effect: the difficulty is actually incurred at the embedded subject NP, but it is not registered until the embedded verb. Two natural directions present themselves to test this interpretation. First, the distance between the embedded subject and the embedded verb could be increased: a spillover effect should occur on the material right after the embedded subject NP, whatever it happens to be. Alternatively, the surprisal theory could be tested for by modulating the the embedded subject NP so that it is more or less predictable. A more predictable embedded subject NP should be read more quickly than one that is less predictable.

Fortunately, recent research has begun to investigate both these directions. An experiment relevant to the spillover hypothesis was conducted by Grodner and Gibson (2005), who varied postmodification of the subject NP in matrix and embedded RC contexts:

[34] The DLT's storage component and the AFH both predict a degree of cost at the embedded subject in an object relativization, but these predictions have no baseline of comparison and at any rate turn out to be inferior in granularity to the predictions of surprisal, so I will not discuss them further.

(19) a. The nurse {∅/from the clinic/who was from the clinic} supervised the administrator...

b. The administrator who the nurse {∅/from the clinic/who was from the clinic} supervised...

The DLT predicts that the difficulty of the first verb will be lowest in the unmodified case, higher in the PP-modified case, and highest in the RC-modified case; first-verb difficulty will also be higher in the embedded contexts (19b) than in the matrix contexts (19a). Surprisal predicts exactly the reverse pattern; and furthermore, if the embedded-verb difficulty seen in (18b) is due to spillover from the embedded subject NP, we might expect to see a spillover spike in the postmodifiers (19b). The experimental results in this case generally support the DLT: there are no substantial differences among the matrix stimuli in (19a), but all the embedded first-verb reading times in (19b) are elevated, and in the double RC stimulus it is elevated significantly above all other stimuli. Furthermore, across all stimuli the PP *from the clinic* is consistently read quickly, which undermines a spillover account of verbal difficulty in (18b).

Gordon et al. (2004) provide another piece of the puzzle by varying the definiteness and quantification of embedded subject NPs in object-extracted RCs. The crucial contrasts involve the following stimulus types:

(20) a. The salesman that **{the/an} accountant** contacted spoke very quickly. (Definite/Indefinite)

b. The salesman that **(the) accountants** contacted spoke very quickly. (Definite/Bare Plural)

c. The salesman that **{the accountant/everyone}** contacted spoke very quickly. (Definite/Quantifier)

In a corpus study within the same article, the authors found definite NPs to outnumber their indefinite or bare counterparts for both singular and plural embedded subjects. To reason about the predictions of surprisal for these cases, it is necessary to recall that the theory links processing difficulty to the conditional probability of each *word* in its context. This encompasses lexical probabilities, so a rare word as a syntactically likely continuation may well be more surprising than a common word as a syntactically unlikely continuation. In the (20a) contrast, the discrepancy in definite/indefinite NP frequency is the only relevant statistic, so surprisal predicts that the definite NPs should be easier. Surprisal also predicts that the definite NPs should be easier in (20b), but the difference in difficulty should be more dramatic, because the comprehender receives more information at once—both the fact of an object RC and the main lexical content of the embedded subject—in the bare plural case than in the indefinite singular case. In the (20c) case, the relevant contrast is likely to be between open-class and closed-class (hence high-frequency) lexical NP head, so we predict lower difficulty for the *everyone* stimulus. These predictions are fairly consistent with the experimental results of Gordon et al. (2004): (20a) produced no significant differences in reading times, (20b) produced significantly faster reading times at the embedded subject NP for the definite stimulus (and at the matrix verb, though curiously not at the embedded verb), and (20c) produced significantly lower reading times at the embedded subject NP, the embedded verb, and the matrix verb.[35]

Finally, one experiment by Warren and Gibson (2002) simultaneously addresses both NP content and spillover, by varying the content of embedded NP subjects and separating the embedded NP from the RC verb with auxiliary and adverbial elements, as seen in (21) below:

(21) The woman who {the boy/you} had accidentally *pushed* off the sidewalk got upset...

Surprisal predicts that the *you* stimulus will be easier, because *you* is a high-frequency closed-class word, and is very common as the subject of an object-extraposed RC.

[35] In a fourth experiment, Gordon et al. (2004) looked at a specific/general (and hence, also more/less frequent) noun contrast for embedded subject, as in *the accountant* versus *the person*, and found no significant reading time differences anywhere. Although it is perhaps not necessary to attempt an explanation of a null result, one difference between *everyone* and *the person* that could be relevant to surprisal is that *the person* may set up a stronger expectation for postnominal modifying content than *everyone*. This expectation is violated in Gordon et al.'s stimuli, since the RC verb immediately follows the head noun of the subject.

If the differential difficulty shows up as spillover, it might be observed in between the RC subject and verb. The DLT's integration component predicts that difficulty should be elevated at the main verb *pushed*, because *you* does not introduce a new discourse referent. Results provide some confirmation for both theories: reading times are numerically but not significantly faster at *you* than at *the boy*; throughout the region *had accidentally pushed off*, reading times are significantly elevated for the lexical NP subject stimulus. Spillover from embedded-subject surprisal might be a source for the elevated reading times at the auxiliary and adverb, whereas the DLT's integration component gains support from reading time elevation at *pushed off*.

Taken together, recent results on verb-final clauses and English relative clauses pose a perplexing set of results. On the one hand, in verb-final environments, extra dependencies preceding the head seem to facilitate rather than hinder reading at the final verb. On the other hand, additional and more informative material before the verb of an object-extracted RC seems to hinder, not facilitate, reading time at that verb. Nevertheless, subregularities in the difficulty of embedded subject NPs observed in Warren and Gibson (2002); Gordon et al. (2004) are consistent with the predictions of surprisal.

One way of interpreting these mixed results is to hypothesize that surprisal has a major effect on word-by-word processing difficulty, but that truly non-local (i.e., long-distance) syntactic dependencies such as relativization and *wh*-question formation are handled fundamentally differently from local syntactic dependencies, and the storage, retrieval, and integration of long-distance dependents incurs a substantial processing cost comparable to the cost of a highly surprising word. On this theory, surprisal effects dominate the processing of verb-final clauses because none of the dependencies are long-distance, but processing a relative clause involves storing, retrieving, and integrating a long-distance dependent, so that relative clause reading times also exhibit substantial DLT-like effects that are not predicted by surprisal. Working out such a two-factor theory would be a non-trivial undertaking beyond the scope of this work, but the most recent available data suggests that formulating and testing such an approach could well be a promising direction for future research on syntactic processing difficulty.

2.7.2 Disentangling verb location from verb identity

English subject-*modifying* relative clauses trigger another syntactic context related to but crucially different from the verb-final clause: after the embedded verb has been observed, the comprehender knows that at some point the RC has to end, at which point return to the matrix clause is almost certain, but she does not know exactly *where* it will end, and does not find out until seeing the first input token that continues the matrix clause. This means that the surprisal theory makes a specific prediction about a recent experiment carried out by Jäger et al. (2005), involving the stimuli in (22) below.

(22) a. The player [that the coach met **at 8 o'clock**] bought the house. . .
b. The player [that the coach met *by the river* **at 8 o'clock**] bought the house. . .
c. The player [that the coach met NEAR THE GYM *by the river* **at 8 o'clock**] bought the house. . .

As in verb-final clauses, the more postverbal constituents within the RC that have been seen in (22), the fewer possible choices there are for subsequent constituents within the RC. This means that the comprehender's expectation for the end of the RC (and hence seeing the matrix verb next) increases as the number of already-seen postverbal constituents increases—once again, a prediction opposite to that of the DLT. Table 2.6 shows matrix-verb surprisal values estimated by a PCFG trained directly off the parsed Brown corpus, together with DLT predictions and empirical mean reading times.[36] The surprisal prediction matches the empirical result: surprisal and reading time at the matrix verb both decrease as the number of postverbal constituents in the preceding RC increases. Crucially, the observed effect does *not* plausibly follow from the account of Konieczny (1996), in which preverbal dependents help the comprehender guess the *identity* of the final verb, because there is no direct argument structure relation between the matrix verb and the verbal dependents in

[36] The paired comparisons between the 1 and 2 and 1 and 3 PP conditions are statistically significant; the paired comparison between the 2 and 3 conditions is not.

	Number of PPs intervening between embedded and matrix verb		
	1 PP	2 PPs	3 PPs
DLT prediction	Easier	Harder	Hardest
Surprisal	13.87	13.54	13.40
Mean Reading Time (ms)	510 ± 298	410 ± 184	394 ± 143

Table 2.6: Surprisal and average reading times at matrix verb for (22)

the RC. The broader surprisal theory encompasses the narrowing of expectations Konieczny stipulates for final-verb identities, but also predicts that comprehension patterns will reflect implicitly-formed expectations about upcoming constituency, a prediction that is borne out in this experiment. Finally, note that this data is entirely consistent with the mixed surprisal/long-distance DLT theory suggested at the end of Section 2.7.1, because the long-distance dependency is already resolved well before the matrix verb is encountered, so surprisal effects should dominate at the matrix verb.

2.8 Falsifiability and future directions

A common objection to frequentistic processing theories is they make it too easy to tell just-so stories, by choosing a grain-level of analysis in which the relevant frequencies happen to fit a set of experimental results. This objection calls the falsifiability of frequency-based processing theories into question. In this section, however, I argue that frequentistic processing, and in particular surprisal-based processing, is highly testable in comparison to other processing theories, and outline a future line of research which tests surprisal-based processing in a direct, highly-constrained way.

2.8.1 Falsifiability

First, it is not the case that non-frequentistic processing theories, such as the memory-based resource-requirement theories of syntactic locality and the Active Filler Hypothesis, are not subject to just-so stories. Syntactic locality-based theories such Gibson's or Hawkins's have a degree of flexibility in terms of what counts as a discourse entity or integration, or what counts as a constituent. In the DLT, for example, both head nouns and relative pronouns count as discourse entities, but complementizers do not (Gibson, 1998); copular verbs count as discourse entities (Grodner and Gibson, 2005), but auxiliary verbs do not (Warren and Gibson, 2002).

Second, the logical structure of frequency-based processing theories invites falsification. These theories state that for every experimental result E in condition C, there is some corpus C' grain size G such that the optimal frequency-driven resource allocation strategy in C, as estimated by G on C', will match E. Falsification requires only that we can find some E* for which no appropriate C' and G can be found.

Third, although the question of which grain size best reflects the upcoming-word expectations of an adult native speaker, there is good reason to believe that the appropriate grain size is quite fine. Adult native speakers disambiguate naturally-occuring utterances quite accurately, certainly far more accurately than any computer-implemented parsing system available today. In the theory outlined in this chapter, surprisal as a measure of processing difficulty is a direct consequence of incremental, stochastic disambiguation. By all indications available from the probabilistic parsing literature (Collins, 1999; Charniak, 2000; Collins and Duffy, 2001; Bod, 2003; Charniak and Johnson, 2005), a grain size considerably finer than gross syntactic category or even lexicalized syntactic category is required to achieve high-accuracy disambiguation. We can infer that with present-day technology and data sources, no robust conditional word probability model we build is likely to err on the side of being too fine-grained.[37]

This last consideration indicates a very concrete research strategy that can test surprisal-based processing directly. Since, as I have outlined in this chapter, surprisal-based processing can be derived directly as a consequence of a rational, incremental, and parallel parsing strategy, we expect that more rational parsing models should

[37] By *robust* I mean a model that does not fit too many parameters given the amount of data available—in statistical terminology, one that does not trade too much bias for variance. Given a finite amount of data, is possible to build a model so fine-grained that its conditional word probability predictions are unreliable because each parameter of the model is effectively estimated from only a few datapoints. We would not expect such an *overfitted* model to make reliable predictions about the probabilities of upcoming words, let alone about processing difficulty reading times.

make surprisal-based predictions on processing difficulty that more closely match experimental data. But the rationality of a parsing model can be operationalized by its degree of fit to empirical (corpus) data – in terms of classification accuracy, or perhaps more appropriately in this case, by its perplexity. As a research strategy, then, we can therefore work at improving the fit of parse models to corpora, and see whether the improved models model psycholinguistic data better.

The caveat to this is that the data against which the rationality of a parse model is assessed should be from the same population (in statistical terms) as the data from which online processing results are derived. Now, the vast majority of online sentence processing experiments bear little distributional relation to naturally occuring data: they pick out a highly specific syntactic construction and instantiate it many times to create a dataset. One strategy is to customize parsing models to individual psycholinguistic experiments. This is something along the lines of what I have done in this chapter, when I percolated case marking and morphological information to syntactic categories in the investigation of German clause order. To carry this strategy out too far, however, is dangerous, because the resulting models would begin to depart from what rational models of everyday linguistic performance must look like—and it is these latter models that we hypothesize are behind the psycholinguistic experiments themselves. Another strategy, perhaps more interesting and novel, would be to take advantage of newly emerging *eye-tracking corpora* (McDonald and Shillcock, 2003; Kennedy et al., 2003), whose stimuli are drawn from naturally-occuring sources. Virtually no such research has been done on such corpora; it has been shown that conditional word probabilities in a bigram language model are correlated with reading time (McDonald and Shillcock, 2003), but it is not even known how a syntax-based language model would perform. Since state-of-the-art parsing-based language models have recently become more competitive with good n-gram language models (Chelba and Jelinek, 1998; Charniak, 2001; Collins et al., 2004; Hall and Johnson, 2004), this would be an interesting subject for future research.

2.8.2 Deriving Surprisal

One piece of theoretical progress made in this chapter is that the association of *surprisal* with processing difficulty is given an information-theoretic basis as the KL divergence between probability distributions. Although Hale 2001 proposed surprisal as a model of processing difficulty, he provided no theoretical basis for surprisal as a plausible measure, as noted in Hale 2003a. The derivation of surprisal as the KL divergence between probability distributions allows us to understand surprisal as a measure emerging from the process of incremental, parallel probabilistic parsing, which simultaneously addresses the issues of ambiguity management and disambiguation. Surprisal as a measure of processing difficulty therefore becomes more closely connected with the basic functional problem posed by human online processing: how inherently ambiguous input is effortlessly and accurately processed.

A deeper question still remains, however: suppose we have accepted that human sentence processing is incremental, fully parallel, and probabilistic. Processing any word of input thus requires an update of the distribution over partial parses in the sentence. It still does not logically follow that the cost of this update should be the KL divergence between distributions. Imagine, for example, that the brain actually used Stolcke's algorithm for update. We might associate processing difficulty with time required for incremental update. But in Stolcke's algorithm, this time requirement is in general linearly dependent on the size of the category space and rule space in the grammar, and does not depend on the actual probability magnitudes involved. Unlike the KL divergence, then, processing time in Stolcke's algorithm is (a) representation-dependent; and (b) unrelated to actual probability masses.

Nevertheless, surprisal as an estimate of processing difficulty has powerful intuitive, theoretical, and empirical attraction. A major question, then, is whether we can derive surprisal or some related model of processing difficulty from deeper cognitive principles. I see two major prospects for such a derivation. The first is at the level of *mechanism*: we may be able to show that a neurally plausible distributed representation of incremental probabilistic parsing naturally leads to probability-sensitive measures of update difficulties. The second is at the level of *rational analysis* (Anderson, 1990): perhaps we can show that an optimal parallel and incremental parsing strategy entails probability-sensitive update difficulty. I can think of two plausible approaches in the latter tack. The first attributes costs to structure pruning, and perhaps to structure generation, along with some assumptions about typical distribution over partial parse probabilities; the second supposes that as a matter of course, a comprehender conducts and acts on inference from incomplete linguistic input, that the choice of which inference to act on is guided by relative beliefs about the incomplete input, and that there is a cost to aborting actions after their premises have been disconfirmed by further input.

2.8.3 Surprisal and linguistic typology

On a higher level, we can also ask what implications a surprisal theory of expectation-based processing may have for understanding the dimensions of linguistic variation. The connection between processing theories of language and linguistic universals has a strong tradition, perhaps best exemplified by the work of Hawkins (1994, 2004). Hawkins' theory of domain minimization is a strongly psychological theory of processing and grammar universals, and its formulation bears a close relationship to that of the DLT (Gibson, 1998, 2000). The simple concept of domain minimization allows Hawkins to derive several of Greenberg's most prominent linguistic universals, such as the correlation between verb-object and preposition-noun head positions, although it leaves a number of issues open. For example, Mandarin Chinese is a mysterious, unexplained language in Hawkins' typology because its combination of head-initial VPs and head-final NPs is clearly suboptimal in Hawkins' typology.

Expectation-based processing is a direct challenge to memory-based theories of syntactic processing such as Gibson's, however. It is therefore appropriate to ask what conclusions we can draw about language typology from an expectation-based theory. We will use surprisal-based processing for this investigation.

Let us first look at probabilistic languages and grammars from an information-theoretic perspective. Elementary information theory tells us the probability of a string in a given probabilistic language $\mathcal{PL}$ determines the actual amount of information that the string conveys to someone who knows $\mathcal{PL}$—the lower the string's probability, the more information it conveys. Now let us consider any string $s = w_1 \cdots w_n$ and let $P_{\mathcal{PL}}(s) = p$. Incremental processing tells us that the total processing difficulty for s should simply be the sum of the individual processing difficulties at each w_i.[38] Now we can ask how different probabilistic languages for which s conveys the same information p might be structured so as to change the distribution of incremental processing difficulties over s. We can do this because there may be many probabilistic languages $\mathcal{PL}'$ that assign the same total probability p to s, but whose array of conditional probabilities $P_{\mathcal{PL}'}(w_i|w_{1 \ldots i-1})$ is in general not the same as $\mathcal{PL}$'s. If we assume that the total processing difficulty $D_{\mathcal{PL}}(s)$ incurred in the comprehension of s is the sum of the processing difficulty at each w_i in s, then by the surprisal measure of processing difficulty we have:[39]

$$\begin{aligned}
D_{\mathcal{PL}}(s) &= -\sum_{i=1}^{n} \log P_{\mathcal{PL}}(w_i|w_{1 \cdots i-1}) && (2.15)\\
&= -\sum_{i=1}^{n} \log \frac{P_{\mathcal{PL}}(w_1 \cdots w_i)}{P_{\mathcal{PL}}(w_{1 \cdots i-1})} && (2.16)\\
&= -\sum_{i=1}^{n} \log P_{\mathcal{PL}}(w_{1 \cdots i}) - \log P_{\mathcal{PL}}(w_{1 \cdots i-1}) && (2.17)\\
&= \log P_{\mathcal{PL}}(\epsilon) - \big(\sum_{i=1}^{n-1} \log P_{\mathcal{PL}}(w_{1 \cdots i}) && (2.18)\\
&\quad - \log P_{\mathcal{PL}}(w_{1 \cdots i})\big) - \log P_{\mathcal{PL}}(w_{1 \cdots n}) && (2.19)\\
&= \log 1 - \log P_{\mathcal{PL}}(w_{1 \cdots i}) && (2.20)\\
&= -\log P_{\mathcal{PL}}(w_{1 \cdots i}) && (2.21)
\end{aligned}$$

[38] That is, the total amount of effort required to comprehend s. This is distinct from intuitive notions of complete-sentence complexity, as traditionally attributed in the pyscholinguistic literature to multiply center-embedded sentences and the like. I suggest that complete-sentence complexity would be more appropriately attributed to the maximal local difficulty within a given sentence.

[39] For notational convenience, for $i = 0$ I have taken $w_{1 \cdots i}$ to be ϵ, the empty string; and $P_{\mathcal{PL}}(\epsilon)$, the prefix probability of the empty string, is 1 because the empty string is a prefix for all strings.

This gives us a counter-intuitive result: no matter how information is distributed throughout a sentence, the resulting total processing difficulty is the same. To see why this is counter-intuitive, compare two probabilistic languages that assign p to s: $\mathcal{PL}_1$, for which $P_{\mathcal{PL}_1}(w_1) = p$ and $P_{\mathcal{PL}_1}(w_i|w_{1\cdots i-1}) = 1$ for all $i > 1$, and $\mathcal{PL}_2$, for which $P_{\mathcal{PL}_2}(w_i|w_{1\cdots i-1}) = \frac{p}{\text{length}}(s)$ for all i. That is, in $\mathcal{PL}_1$ every sentence s begin with a word w that begins no other sentence, so w conveys all the information in s; whereas in $\mathcal{PL}_2$, each word in each sentences has the same conditional probability, so information is distributed equally throughout s. The strict surprisal theory says that the total processing difficulty of s is the same in $\mathcal{PL}_1$ and $\mathcal{PL}_2$. But the strong intuition is that natural languages are much more like $\mathcal{PL}_2$: unambiguous grammatical sentences become incomprehensible when too much complexity is focused in a small region of the sentence (as with multiple center-embeddings).

We can escape this conundrum by departing a bit from the strict theory of surprisal-based processing we have investigated thus far and saying that for some k,

$$\textit{observed difficulty} \quad \propto \quad [\log P(w_i|w_1, \cdots, w_{i-1})]^k \tag{2.22}$$

the intuition being that the actual processing difficulty experienced at a word w may be a nonlinear function of w's strictly informational difficulty.

This family of surprisal-based models is still broadly consistent with all the theoretical and empirical concerns that we have investigated in this chapter so far: it is distribution-dependent and representation-independent, and processing difficulty is still a monotonic function of conditional word probability. The family can be broadly partitioned into three pieces:

$$k \begin{cases} < 1 & \mathcal{PL}_1\text{-optimal} \\ = 1 & \mathcal{PL}_0 \text{ (conventional surprisal)} \\ > 1 & \mathcal{PL}_2 \text{ -optimal} \end{cases} \tag{2.23}$$

I submit that natural languages are much more like $\mathcal{PL}_2$, so we would expect that we are in the region of $k > 1$. This claim captures the natural intuition that the more surprising an event, the more difficulty is associated with incremental increase in how surprising it is.

The $\mathcal{PL}_2$ hypothesis also makes empirical predictions with respect to sentence processing: that the best-fit regressions of surprisal against reading time should give exponents of $k > 1$. At present our limited understanding of how to devise fine-grained probablilistic string models appropriate to controlled experimental stimuli makes it premature to evaluate this claim against most of the sentence processing literature. However, I am aware of one intriguing recent study that can be interpreted as supporting the $\mathcal{PL}_2$ hypothesis. In an interesting self-paced reading time study whose stimuli involved a superposition of relative clauses and arithmetic problems, Fedorenko et al. (2004) found a *superadditive*—that is, non-linear—interaction between difficulty of the linguistic stimuli (subject vs. object relativization) and the arithmetic stimuli (large vs. small numbers) on reading time. This superadditive interaction indicates that the addition of a constant amount of difficulty has a greater incremental effect on reading time in the presence of difficulty than it does in the absence of difficulty.[40] The multiple-task nature of Fedorenko et al. 2004's task made it possible to quantify task difficulty in a way that allowed them to identify nonlinear effects. In single-task reading studies, surprisal can potentially serve as the yardstick against which to evaluate the linearity of processing difficulty effects.

Exploring the consequences of the $\mathcal{PL}_2$ hypothesis leads us, however, to another question. Natural languages are sets of form-meaning pairs, and it is possible that an alternative language $\mathcal{PL}'$ might assign the meaning that $\mathcal{PL}$ assigns to a given string s to another string s'. Let us assume that $\mathcal{PL}'$ can choose any string s' so long as $P_{\mathcal{PL}'}(s') = P_{\mathcal{PL}}(s)$, and can distribute the incremental conditional word probabilities in s' any way it wants. If $k > 1$, then for a given length of s', the lowest-difficulty assignment is to equalize the conditional probablity of each word; and if we vary the length of s', find that the longer it gets the lower the overall processing difficulty, so long as $\mathcal{PL}'$ assigns the same conditional probability to every word in s'. If language

[40] Although Fedorenko et al. 2004 use the term shared *working memory resources* in discussion of the interaction, nothing in their data crucially demands an account based on memory as opposed to expectation or other cognitive resources.

typology consisted of the exploration for probabilistic languages that convey messages with the lowest difficulty, then we would expect natural languages to prefer extremely long sentences, each word of which is only slightly informative.

The solution to this conundrum, I propose, comes again from psycholinguistics and also basic principles of linguistics. Independent of either memory-based or expectation-based processing effects, a large proportion of total variance in reading times can be explained simply by accounting for word length: people spend more average time reading a longer word than a shorter word (Just and Carpenter, 1980; Rayner et al., 1996). Similarly, sentences with more words are generally read more slowly than sentences with shorter words. This points to a *duality of patterning* in processing difficulty: there is a cost of processing phonetics and/or orthography, and a separate cost of processing syntax. We can re-incorporate this into the equation for incremental processing difficulty as follows:

$$\textit{difficulty} \quad \propto \quad [\log P(w_i|w_1, \cdots, w_{i-1})]^k + \mathrm{Cost}(\mathrm{phon}(w_i), \mathrm{orth}(w_i)) \qquad (2.24)$$

Assuming that the cost function has a positive, non-zero lower bound, arbitrarily long sentences with small, equally-distributed incremental probabilities will no longer have arbitrarily small total processing difficulties.

We are now in a position to entertain an adaptive theory of language typology derived from the surprisal theory of sentence comprehension. The typology of natural languages should consist of the solutions minimizing the overall difficulty given in 2.24 summed over the distribution of strings in the language. Given a probabilistic language $\mathcal{PL}$, we can compare a small perturbation $\mathcal{PL}'$ that assigns a longer string s' to the meaning of s in $\mathcal{PL}$. In some cases, the reduction of the context-sensitive syntactic expectation cost may outweigh the context-independent phonological and orthographic processing cost; in other cases, the context-independent phonological and orthographic cost may outweigh the context-sensitive syntactic expectation cost. We can then think of dimensions of language typology as a set of trade-offs between these costs. The overall *cost* of a natural language can be estimated using two things:

(a) an estimate of the probabilistic language associated with that natural language; and (b) a sample of strings in the language, whose cost may be estimated using the estimated probabilistic language.

A natural approach to elucidating this typological theory would be to take a language well-described in frequentistic terms and looking at the effect on the difficulty metric given in 2.24 of perturbing its (non-categorical) grammatical structure: altering, freeing, or stiffening syntactic word order; or looking at the effects of inserting or deleting function words, to name just two. Such investigations may already be possible with languages for which we have syntactically-annotated corpora and know how to estimate reasonably accurate probabilistic languages.

As a concrete example of how such a theory might play out in syntax, consider a recent study by Race and MacDonald (2003) on the effect of dropping relative pronouns on reading times for object relative clauses. From among a number of factors Race and MacDonald found two positively associated with *that* deletion: pronominality of the subject of the relative clause, and presence of a determiner in full-NP subjects of the relative clause. Race and MacDonald also conducted a self-paced reading time study whose results are generally consistent with expectation-based processing: when the embedded subject is a determinerless full NP, object relative clauses are read more slowly when *that* is deleted than when *that* is preserved; but when the embedded subject is pronominal, there is no significant reading time difference.

We can apply our embryonic theory of language typology to the question of why there might be a positive association between pronominal subjects (and full-NP subjects with determiners) and *that* deletion. Remember that, on our theory, languages should tend towards spreading information equally throughout a sentence. We have also seen that open-class and closed-class words vary tremendously in their predictability, simply due to the overwhelming number of open-class words. In a pseudo-English lacking complementizer deletion, then, pronominal subject NPs in relative clauses would have far less surprisal than determinerless full NPs in the same position, simply because pronouns are closed-class. If we think of the grammar as cooperating with the comprehender, *that* is doing useful work in the determinerless full-NP subject cases, because it signals to the comprehender that a relative clause has begun and to expect a subject NP. But in the pronominal subject case, *that* may be extranneous, because the somewhat high surprisal incurred by the onset of a relative clause would be offset by the high predictability of the pronominal subject. Since the other half of our theory is that a cost is also incurred in processing phonological material, dropping *that* before subject pronouns and determiners may be a beneficial tradeoff for the comprehender. This theory makes the highly specific and testable prediction that *that* deletion should correlate with the conditional probability of the first word of the relative clause given everything preceding the RC.

Such a theory also encompasses recent results in the quantitative phonology of listener modeling. As shown by Gregory (2001), speakers tend to choose as targets for phonological reduction those words that they consider highly predictable *to their audience*. Extending the notion of a two-level cost function with a tradeoff between a constant phonological processing cost and a predictability-dependent linguistic integration cost, some reduction of highly predictable words may pay off in terms of the decrease in phonetic processing cost, even if the associated integration cost may increase due to lowered predictability of the actual phonetic realization.

2.9 Conclusion

Recent experimental results in syntactic ambiguity resolution indicate that comprehenders incrementally integrate a variety of evidential knowledge in the process of discriminating the preferred interpretation of a sentence; probability theory serves as a coherent architecture for this *constraint-based*, *resource-allocation* paradigm of ambiguity resolution. We can extend the parallel, probabilistic disambiguation perspective of incremental sentence processing into a theory of syntactic complexity and processing difficulty by formalizing a linking hypothesis stating that the primary source of difficulty incurred in processing a given word is determined by the degree of update in the preference distribution over interpretations of the sentence that the word requires. Formalized appropriately using the information-theoretic measure of the *relative entropy* between probability distributions, we are able to derive a theory of processing difficulty previously stipulated by Hale (2001), that the difficulty of a word is the *surprisal* (negative log of the conditional probability) of that word given its context. This surprisal theory has several desirable theoretical and mathematical properties, including a coherent integration of rational disambiguation, incremental processing, and differential processing difficulty; representation-agnosticism; and freedom from granularity bias that plagues other probabilistic theories of syntactic comprehension. Empirically, it can smoothly incorporate major results in the ambiguity-resolututation literature; it also makes non-trivial predictions about processing difficulty in *unambiguous* sentences that compare favorably to locality-based resource-requirement theories of syntactic complexity, particularly in head-final and similar contexts where the comprehender knows that a certain type of constituent is upcoming, but is uncertain as to exactly *where* and *what* it is.

Chapter 3

Nonlocal dependencies from context-free trees

3.1 Introduction

This chapter takes up the problem of identifying the complete dependency structure of natural language sentences.[0] As I argued in Chapter 1, context-free phrase structure grammars (CFGs), which in some sense are at the the heart of the majority of both formal and computational syntactic research, in their simplest incarnations only provide adequate information to recover *local* dependency relations. In the context-free structural description Example (1) below, for instance, the *wh*-phrase *who* is semantically a dependent of the deeply embedded verb *attend*, but the verb with which it is in a local structural relationship is *do* (or perhaps *expect*):

(1) [$_{\text{SBAR}}$ [$_{\text{WHNP}}$ Who] [$_{\text{S}}$ do you [$_{\text{VP}}$ expect [$_{\text{VP}}$ to attend]]] ?]

The dependency between *who* and *attend* in this sentence is therefore syntactically *non-local.*

Although a variety of mechanisms exist for enhancing CF trees to express the classes of non-local dependency most common in natural language (see Section 1.4

[0] An earlier version of this chapter appeared previously as Levy and Manning (2004).

for an overview), relatively little work in broad-coverage statistical parsing has focused on these mechanisms. In this chapter I investigate the first of the three approaches outlined in Section 1.4. Complete dependency recovery is viewed as a two-phase process: first, a unique CF parse, encoding only the information necessary for surface dependencies, is chosen for a sentence; second, an algorithm for non-local dependency identification and resolution is applied to that unique CF parse. The resulting enhanced phrase-structure tree contains all the information necessary to uniquely determine the full dependency structure of the sentence.

The dependency-reconstruction algorithm is built from individual components involving *Maximum-Entropy* statistical classifiers. These classifiers must be trained on a hand-constructed syntactically annotated corpus. Fortunately, such corpora are now available for several languages. In this chapter, I compare results of the dependency-reconstruction algorithm on corpora of newspaper text in English and German. This is an interesting contrast because, despite their similarity in many grammatical respects, non-local dependency is considerably more prevalent in German than in English. Since this approach is dependent on a hand-annotated dataset, the types of non-local dependency covered here is dependent on the contents of the annotated corpora available. Nevertheless, I attempt to discuss these examples in the larger context of discontinuity as it relates to syntactic theory.

3.1.1 Previous and Related Work

Previous work on nonlocal dependency recovery in broad-coverage parsing has focused entirely on English, despite the disparity in type and frequency of various non-local dependency constructions for varying languages (Kruijff, 2002). Collins (1999)'s Model 3 investigated GPSG-style trace threading for resolving nonlocal relative pronoun dependencies. Johnson (2002) was the first post-processing approach to non-local dependency recovery, using a simple pattern-matching algorithm on context-free trees. More recently, Campbell (2004) and Jijkoun and de Rijke (2004) have followed a strategy similar to that presented here, taking context-free parse trees as a starting point and enhancing them with non-local dependency information. Dienes and

Dubey (2003a,b) and Dienes (2003) approached the problem by pre-identifying empty categories using an HMM on unparsed strings and threaded the identified empties into the category structure of a context-free parser, finding that this method compared favorably with both Collins' and Johnson's. The division of labor into context-free and trans-CF parsing phases can also be related to traditional non-stochastic Lexical-Functional Grammar (LFG) parsing (Kaplan and Maxwell, 1993), which first produces a context-free packed forest for an input tree and subsequently introduces constraints determining (sometimes trans-CF) functional relations between nodes in the CF phrase-structure forest. More recently, however, stochastic parsing of hand-crafted unification- or constraint-based grammars in both LFG (Riezler et al., 2002; Kaplan et al., 2004) and Head-Driven Phrase Structure Grammar (HPSG) (Toutanova et al., 2005) have tended towards single-phase inference using discriminative models over complete parses that include both local and non-local dependency information.

3.2 Datasets & types of non-local dependency covered

In this section I briefly discuss the annotation practices for discontinuity in Penn Treebanks and for the NEGRA treebank. This section is also useful for reference in Chapter 4.

3.2.1 Nonlocal Dependency in the Penn Treebanks

The Penn Treebanks use a variety of null-element annotations that can be viewed as syntactic discontinuities (Bies et al., 1995, see ch. 4–4). These in turn are a subset of a larger class of empty-category annotations in the Penn Treebanks. Figure 3.1 shows an example of these annotations in the Penn Treebank and their interpretation as discontinuous relationships. Of them, the following types involve coindexation:[1]

[1] There are also several kinds of null-element and "pseudo-attachment" annotations in the Penn Treebank that I ignore, because they are not plausible instances of syntactic discontinuity. These include `*U*`, which is used to canonicalize the implicit form of unit (mostly monetary) expressions; `*?*` and `*NOT*`, which are annotation patterns for ellipsis; and `*PPA*`, which indicates unresolvable

- traces (A′ movement in Government-Binding theory (Chomsky, 1981)): *wh*-movement, topicalization, relative clauses, parasitic gaps, and tough movement, annotated as `*T*`;
- vacant argument positions (GB's PRO and NP movement): passives, raising, controlled and arbitrary subjects of participials, gerunds, and infinitives, annotated as `*`;
- right node raising, annotated as `*RNR*`;
- right dislocations such as relative clause or prepositional phrase extraposition from NP, annotated as `*ICH*`;
- expletives, annotated as `*EXP*`;

Two major classes of discontinuous dependency are contained in these annotations. The first I term *dislocations*, cases where a constituent should *not* be given a transparent compositional (i.e., sister-head) semantic interpretation at its position in the parse tree, but rather has a compositional relationship with some more distant node. The second, *shared* nodes, are those that should be interpreted both at their site in the tree and in a compositional relationship with some other node. Of the above, traces, right dislocations, expletives, and some right node raisings are cases of dislocations (although unembedded instances of right node raising can be handled cleanly as a context-free local dependency by some accounts of coordination, such as that of Maxwell and Manning (1996)). Coindexed vacant argument positions such as those involved in raising and control are cases of shared nodes.[2] Passivization is also given the same annotation in the Penn Treebank as raising and control, with the empty dependent node appearing postverbally inside the VP.[3]

syntactic ambiguity (such as PP attachment ambiguities where the correct attachment cannot be determined from context).

[2] It is controversial in theoretical linguistics whether the shared dependency in all cases of raising and control is purely syntactic (see for example discussion in Chapter 7 of Pollard and Sag 1994), but addressing this controversy is beyond the scope of this chapter.

[3] The syntactic motivation for annotating passivization identically to raising is that they are both analyzed as A-movement in Government and Binding syntactic theory (Haegeman, 1991). Although control is *not* typically analyzed as A-movement, it shares with passivization and raising the common

There are also three cases in which a dislocated constituent is identified with more than one base position. The first is when there is an association with positions in coordinated conjuncts. When an argument sister of coordinated verbs is extracted, the trace appears as a sister of each conjunct. These examples can generally be handled by creating a coordinate mother node that the dislocation is in a unique relationship with. The second case is right node raising; unembedded instances can be subsumed under (non-constituent) coordination, but embedded instances require either multiple base positions or non-local feature passing through context-free categories. The third case is parasitic gaps, rare but arguably present in the Penn Treebank.[4]

One more type of Penn Treebank annotation needs to be explained in order to completely understand nonlocal dependency in the corpus. In Figure 3.1, the WHNP in the lower right corner dominates *0*, which is notation for a *null complementizer*—the silent beginning of a relative or complement clause that alternates with an overt instance of *that* or *which*. In this particular case, annotation on the maximal NP projection of the null complementizer mediates the (potentially long-distance) extraction of the RC object, as indicated by the -1 coindexation in Figure 3.1.

3.2.2 Nonlocal dependency in the NEGRA treebank

The NEGRA treebank (Skut et al., 1997b), which consists of roughly 350,000 words of German newspaper text, was constructed explicitly with the possibility of discontinuous constituency in mind. There are no string-position constraints on syntactic constituency in NEGRA, and there are no null elements. However, a transformed

property of entering into a syntactic relationship with the governing verb both at the overt and underlying position: at the overt position, agreement with the governing verb is determined, and semantic relationships are determined at the underlying position (as well as at the overt position, in the case of control). Since it is impossible to distinguish raising from control using only Treebank annotation, I have left the three phenomena together as "shared dependencies".

[4] Although parasitic gap annotation guidelines exist in the manual (Bies et al., 1995, p. 68), I have been unable to find a clear example of parasitic gapping in the Treebank. The closest (yet still dubious) example I have found is

> Of 1,224 companies surveyed, 31% __ expect to cut spending on plant equipment and machinery , while only 28% __ plan to spend more.

where the initial PP is annotated as a topic extracted from the two percentage NPs.

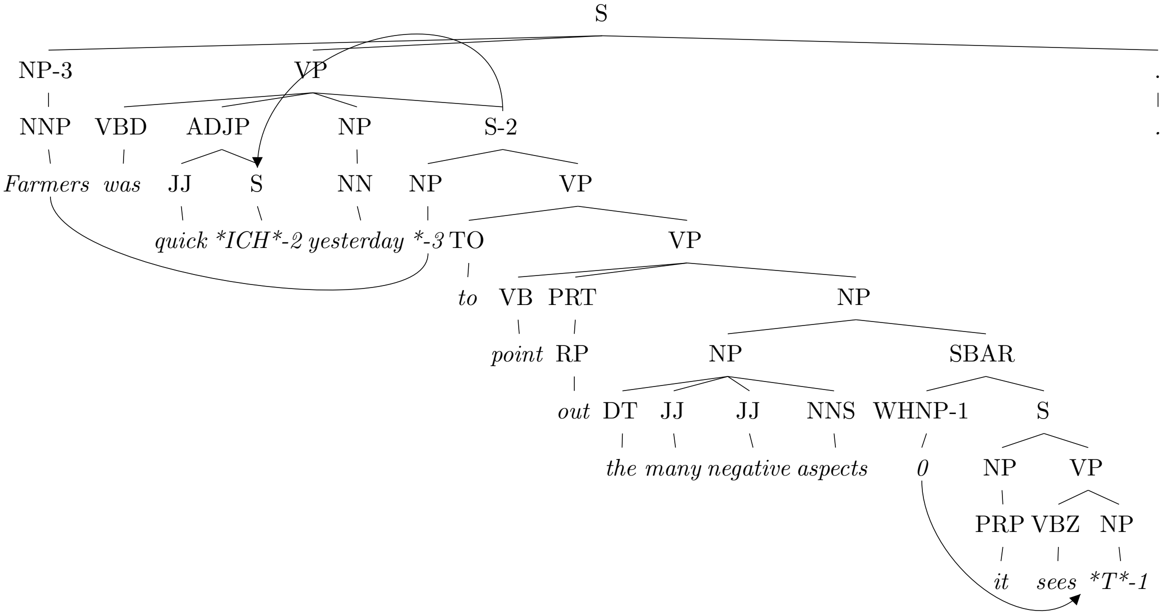


Figure 3.1: Example of empty and nonlocal annotations from the Penn Treebank of English, including null complementizers (*0*), relativization (**T*-1*), right-extraposition (**ICH*-2*), and syntactic control (**-3*).

version was also produced algorithmically (Brants, p.c.; Skut et al. 1997a) where all phrase structure trees are context-free, but discontinuous dependency relations are annotated using Penn Treebank-style dislocated-constituent coindexation. The types of discontinuity appearing as a result of the context-free transformation of NEGRA include roughly traces, rightward displacement, and expletives, as described in Section 3.2.1 for the Penn Treebanks.[5] They further include instances of *argument composition* or *clause union* (Hinrichs and Nakazawa, 1994), where one verb subcategorizes for another and the arguments of the two are positionally interleaved.[6] All these

[5] Interestingly, parasitic gaps appear not to exist in standard German (Kathol, 2001).

[6] Significantly, subjects in NEGRA are taken to be in the domain of the finite verb. This means that finite auxiliary + participial verb combinations are treated as having discontinuous clauses when a non-subject argument is fronted. For example, in the sentence

(i)	Das	Buch	habe	ich	gelesen
	the	book	have	I	read

the combination [*das Buch* + *gelesen*] is taken to be a discontinuous constituent. In dependency terms, the dependency of *Buch* on *gelesen* is a crossing dependency because it crosses the root of

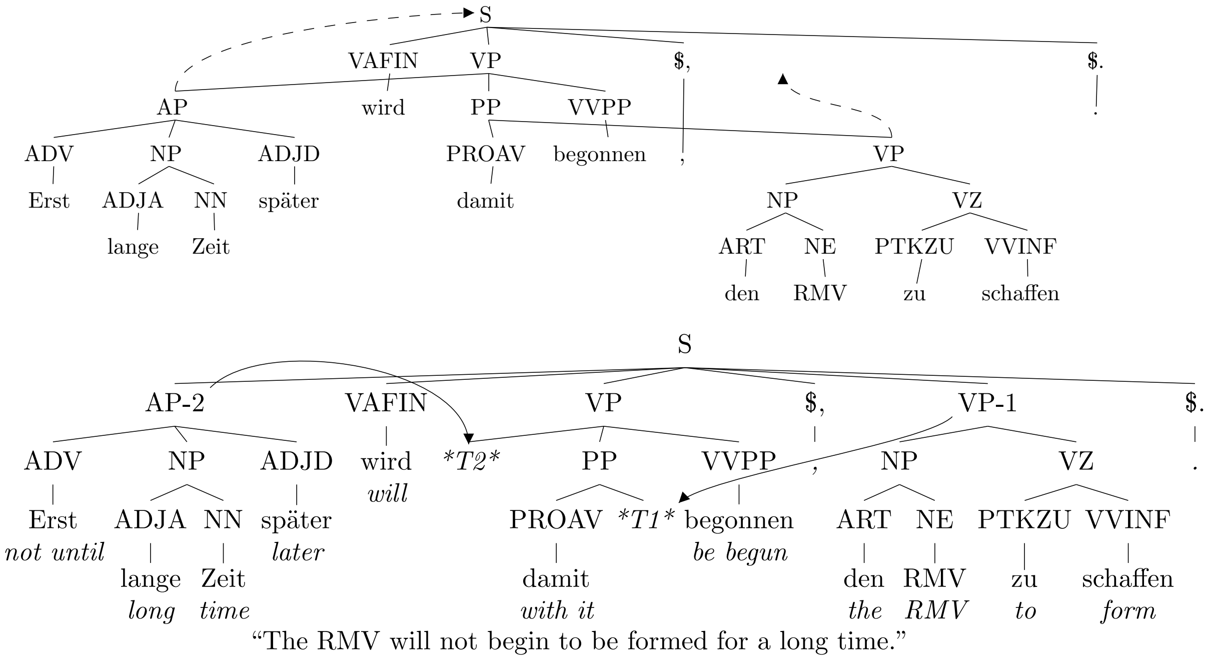


Figure 3.2: Nonlocal dependencies via right-extraposition (**T1**) and topicalization (**T2**) in the NEGRA corpus of German, before and after transformation to context-free form. Dashed lines show effect of remapping into context-free form.

cases can be considered as *dislocations* in the classification introduced above for Penn Treebank discontinuities. Figure 3.2 gives an example of an annotated NEGRA sentence containing discontinuous dependencies, and the corresponding automatically produced context-free version with annotation of nonlocalities.

3.3 Algorithm

Corresponding to the three types of empty-element annotation found in the Penn Treebank, we divide the process of CF tree enhancement into three phases. Each phase involves the identification of a certain subset of tree nodes to be operated on, followed by the application of the appropriate operation to the node. Operations may involve the insertion of a category at some position among a node's daughters; the marking of certain nodes as dislocated; or the relocation of dislocated nodes to

the sentence, *habe*.

other positions within the tree. The content and ordering of phases is consistent with the syntactic theory upon which treebank annotation is based. For example, WSJ annotates relative clauses lacking overt relative pronouns, such as the SBAR in Figure 3.1, with a trace in the relativization site whose antecedent is an empty relative pronoun. This requires that empty relative pronoun insertion precede dislocated element identification. Likewise, dislocated elements can serve as controllers of control loci, based on their *originating site*, so it is sensible to return dislocated nodes to their originating sites before identifying control loci and their controllers. For WSJ, the three phases are:

1. (a) Determine nodes at which to insert null COMPlementizers[7] (IdentNull)

   (b) For each COMP insertion node, determine position of each insertion and insert COMP (InsertNull)

2. (a) Classify each tree node as ± DISLOCATED (IdentMoved)

   (b) For each DISLOCATED node, choose an ORIGIN node (RelocMoved)

   (c) For each pair ⟨DISLOCATED,*origin*⟩, choose a position of insertion and insert *dislocated* (InsertReloc)

3. (a) Classify each node as ± control LOCUS (IdentLocus)

   (b) For each LOCUS, determine position of insertion and insert LOCUS (InsertLocus)

   (c) For each LOCUS, determine CONTROLLER (if any) (FindController)

Note in particular that phase 2 involves the classification of overt tree nodes as dislocated, followed by the identification of an origin site (annotated in the treebank as an empty node) for each dislocated element; whereas phase 3 involves the identification of (empty) control loci *first*, and of controllers later. This approach contrasts with that of Johnson (2002), who treats empty/antecedent identification as a joint task,

[7] The WSJ contains a number of SBARs headed by empty complementizers with trace S's, generally corresponding to fronted quotations such as the following:

> [$_{\text{S-1}}$ Those dividend increases may signal trouble ahead for stock prices] , some analysts [$_{\text{VP}}$ warn [$_{\text{SBAR}}$ $0 *T* -1$]].

These SBARs are introduced in the algorithm as projections of identified empty complementizers as daughters of non-SBAR categories.

and with Dienes and Dubey (2003a,b), who always identify empty nodes first and determine antecedents later. The motivation is that it should generally be easier to determine whether an overt element is dislocated than whether a given position is the origin of some as yet unknown dislocated element (particularly in the absence of a sophisticated model of argument expression); but control loci are highly predictable from local context, such as the subjectless non-finite S in Figure 3.1's S-2.[8] Indeed this difference seems to be implicit in the non-local feature templates used by Dienes and Dubey (2003a,b) in their empty element tagger, in particular lookback for *wh*-words preceding a candidate verb.

As described in Section 3.2, NEGRA's nonlocal annotation schema is much simpler, involving no uncoindexed empties or control loci. Correspondingly, for NEGRA the algorithm includes only phase 2 of the WSJ algorithm, step (c) of which is trivial for NEGRA due to the deterministic positioning of trace insertion in the treebank.

In each case I use a maximum-entropy (MaxEnt) model for node classification. The following section provides a brief introduction to Maximum Entropy models and particular aspects of them relevant to the work in this chapter. In the second and third parts of phases 2 and 3, when determining an originating site or controller for a given node N, or an insertion position for a node N′ in N, I use a competition-based setting, using a binary classification (yes/no for association with N) on each node in the tree, and during testing choosing the node with the highest score for positive association with N.[9] All other phases of classification involve independent decisions at each node. In phase 3, I include a special zero node to indicate a control locus with no antecedent.

[8] Additionally, whereas dislocated nodes are always overt, control loci may be controlled by other (null) control loci, meaning that identifying controllers before control loci would still entail looking for nulls.

[9] The choice of a unique origin site makes the algorithm unable to deal with right-node raising or parasitic gaps. Cases of right-node raising could be automatically transformed into single-origin dislocations by making use of a theory of coordination such as Maxwell and Manning (1996), while parasitic gaps would require the introduction of a secondary classifier. Both phenomena are low-frequency, and I ignore them here, but in principle we could allow multiple origins either (more straightforwardly) by selecting all origins above a certain threshold, or by treating multiple-origin assignments as distinct classification options that receive their own probability.

3.4 Maximum Entropy Models

The algorithm presented in this section for recovery of non-local dependency relies on the sequential application of classifiers to various parts of tree structure. These individual classifiers take the form of *Maximum Entropy* models—equivalently, in our case, *logistic regression* models. This section briefly overviews the key features of Maximum Entropy models utilized in this chapter. Further details can be found in many places in the literature, including Della Pietra et al. (1997) and Berger et al. (1996).

The general problem of classification is as follows: given an observed datum o, choose from among a set of predetermined hidden classes H the correct class h for o. The application of probabilistic models to classification involves the construction of a conditional probability distribution (c.p.d.)

$$P(h|o)$$

Discriminative probabilistic models, of which MaxEnt is an example, directly estimate such a c.p.d.

The further assumption is made that $P(h|o)$ is completely determined by a vector F of features that is deterministically specified by the combination h, o. That is, there is a function f specifying a feature vector of real-number values $f(h, o) = (f_1, f_2, \cdots)$ and $P(h|o) = P(h|f(h, o))$.[10] The method of maximum entropy involves a commitment to the *log-linear family* of functions for $P(h|o)$, namely:

$$\begin{aligned} P(h|o) &= \frac{\Pi_i^n e^{\lambda_i f_i}}{\Sigma_{h\in H} \Pi_i^n e^{\lambda_i f_i}} && (3.1) \\ &= \frac{e^{\Sigma_i^n \lambda_i f_i}}{\Sigma_{h\in H} e^{\Sigma_i^n \lambda_i f_i}} && (3.2) \end{aligned}$$

where f_i is the value of the i-th entry in $f(o)$. A particular member of this family of functions is specified by a *parameterization*—that is, a choice of specific values for

[10] Typically, F will be finite-dimension, but kernel methods (see Collins and Duffy 2001 and others) can in fact extend discriminative classification to certain cases where F is infinite-dimension.

the λ_i.

We can usefully distinguish two variants, *fixed-class* and *variable-class*, of maximum-entropy classification. In the fixed-class variant, the possible hidden classes of each observation o_i range over the same inventory $\{h_1 \cdots h_n\}$. In this situation, it is common to assume that each feature directly calculable exclusively from the observation corresponds to n different features in F, one for each hidden class, and that a feature in F derived from h_i must have value 0 for all pairings $\langle h_j, o\rangle, j \neq i$. If there are m different features arising from the possible o, then the length of the feature vector F must be $m \times n$. For example, if the task is to guess the part of speech of a word in isolation, there are 25 parts of speech in the language, and the two observation-based features to use are whether the word starts in *un-* ends in *-ing*, then F would be length 50 (25 features pair a hidden class with *un-* and 25 with *-ing*), but only two features could be non-zero for any particular $\langle h, o\rangle$. This fixed-class variant of maximum entropy is equivalent to the *logistic regression* models widely used in categorical data analysis (Agresti, 2002), and all the classifiers I use in nonlocal dependency recovery are fixed-class (the hidden classes always being “yes” and “no” for each node).

The variable-class variant of maximum entropy generalizes the fixed-class variant by allowing each observation o_i to have a different (and possibly infinite) range of possible hidden classes. As an example, an observation o could be a natural language string and the hidden class h could be a complete parse tree whose fringe is o, as in parse selection work such as Toutanova et al. (2005). The possible classes range over the possible parse trees for o given some grammar, and the feature function $f(h, o)$ would be sensitive to various properties of a complete parse tree.[11]

Given the choice of a family of probabilistic models, a criterion must be chosen by which a particular model parameterization is chosen as a *fit* to a set O of training observations—that is, a setting of the parameters $\{\lambda_i^h\}$ of Equation 3.1 to values that are somehow consistent with the dataset. For this purpose we use the method

[11] There might be a potential gain in treating origin-site and controller identification as a variable-class model, with the classes for each dislocated or shared dependent ranging over the nodes in its tree, because the resulting likelihood function is more closely related to the actual classification task, but I have resolved the prominent cases where this might be an issue through appropriate feature templates.

of *maximum likelihood*.[12] That is, if every example $\langle h, o\rangle$ in our training set has a likelihood determined by Equation 3.1, then we can calculate a total likelihood for our entire training set as follows:

$$
\begin{aligned}
L(O; \{\lambda_i^h\}) &= \Pi_{\langle h,o\rangle \in O} P(h|o) &(3.3)\\
&= \Pi_{\langle h,o\rangle \in O} \frac{e^{\Sigma_i^n \lambda_i^h f_i}}{\Sigma_{h\in H} e^{\Sigma_i^n \lambda_i^h f_i}} &(3.4)
\end{aligned}
$$

Since our problem is now framed as *choosing* a set of values for the parameters $\{\lambda_i^h\}$ of our model, we regard O as constant and seek to maximize $L(O)$ with respect to the parameters $\{\lambda_i^h\}$. It turns out that it is very easy (though possibly time-consuming) to maximize the MaxEnt family of models: the likelihood surface defined by 3.3 is convex, and thus has a unique maximum under ordinary circumstances.

Once a model has been fit to a set of training data, that model can then be used to classify a new observation o' by finding the hidden class h' for which the conditional probability $P(h'|o')$ is maximized.

3.4.1 Regularization

In a wide variety of machine learning problems, including this one, the *featurization* of the data—the dimension of the feature vector $(f_1, f_2, \cdots, f_n)$ corresponding to an observation o—is very high, often on the order of the number of training samples. This renders trained models prone to *overfitting*—that is, learning parameter settings too closely tuned to the particulars of the training set and unable to robustly generalize to new data. The most extreme example of this problem comes when a feature acts as a *perfect predictor*—when a particular feature f_{k^*} takes on a non-zero value only for a certain class h_{j^*} in the training data. In this case, the likelihood is monotonic in $\lambda_{k^*}^{j^*}$ (increasing or decreasing depending on whether f_{k^*} is positive or negative), and

[12] The maximum-likelihood values for the parameters λ_i of the loglinear-form c.p.d. given in Equation 3.1 also turn out to maximize the entropy of the c.p.d. $H(P(h|o)$ subject to the constraint that the model's expectation for each λ_i is equal to the observed expectation. This likelihood/entropy duality is the source of the term "maximum entropy".

thus has no maximum; if we approximate the best fit by giving a $\lambda_{k^*}^{j^*}$ an arbitrarily high value, then at classification time, an example with activation of f_{k^*} will always be assigned class h_{j^*}, *no matter what other features are active.* Crucially, this will occur even if there is only one example for which f_{k^*} is non-zero.[13]

A common solution to this problem, which I employ here, involves the application of a *penalty* or *regularizing* term $T(\{\lambda_i^h\})$ to the objective function. The regularizing term is structured to capture the intuition that we do not just want a model that maximizes the likelihood of our training data; we want a model that achieves high data likelihood without coming to strong conclusions on the basis of small amounts of evidence. This term finds its minimum at the origin and is multiplied with the likelihood to achieve the new objective function:

$$L(O;\{\lambda_i^h\}) \times T(\{\lambda_i^h\}) \tag{3.7}$$

I use the common *Gaussian prior* regularizing term (Berger and Miller, 1998; Chen and Rosenfeld, 2000), which is of the form

$$\exp(-\sum_{h,i} \frac{(\lambda_i^h)^2}{2\sigma^2}) \tag{3.8}$$

where σ^2, the variance of the Gaussian term being used, is chosen before training. Since a given feature weight λ_i^h helps the likelihood more if the feature is common, but is penalized only once regardless of feature frequency, the effect of regularization is that rarer features will tend to receive smaller weights than more common features,

[13] In the presence of a perfect predictor, the likelihood has the following form:

$$\begin{aligned} L(O;\{\lambda_i^h\}) = \; & (\Pi_{\langle h_j,o\rangle \in O, j\neq j^*} \frac{e^{\Sigma_i^n \lambda_i^j f_i}}{\Sigma_{h\in H} e^{\Sigma_i^n \lambda_i^j f_i}}) \\ & \times (\Pi_{\langle h_{j^*},o\rangle \in O} \frac{e^{\Sigma_{i\neq k^*}^n \lambda_i^{j^*} f_i} e^{\lambda_{k^*}^{j^*} f_{k^*}}}{\Sigma_{h\in H} e^{\Sigma_i^n \lambda_i^{j^*} f_i}}) \end{aligned} \tag{3.5}$$

With respect to $\lambda_{k^*}^{j^*}$, the first parenthesized term is constant, as j is never equal to j^*; the second parenthesized term can be put in the form

$$\frac{ae^{\lambda_{k^*}^{j^*} f_{k^*}}}{ae^{b\lambda_{k^*}^{j^*}} + C} \tag{3.6}$$

for positive constants a and C.

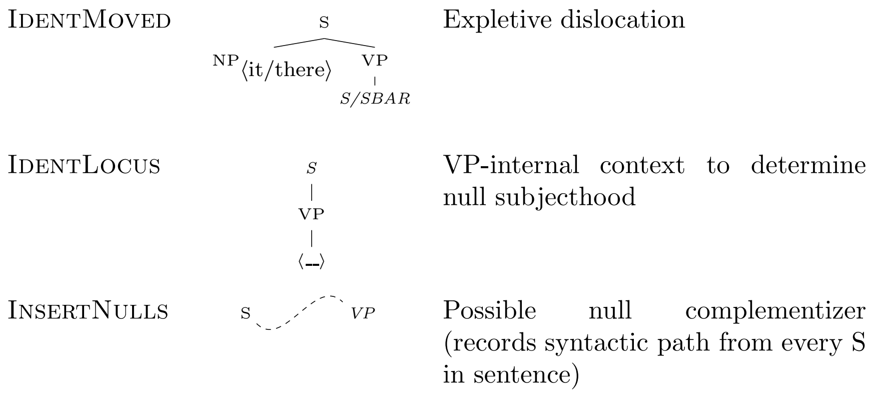

IDENTMOVED
S
NP ⟨it/there⟩
VP
S/SBAR
Expletive dislocation
IDENTLOCUS
S
VP
⟨–⟩
VP-internal context to determine null subjecthood
INSERTNULLS
S
VP
Possible null complementizer (records syntactic path from every S in sentence)

Figure 3.3: Different classifiers' specialized tree-matching fragments and their purposes

capturing the intuition that we should generalize more aggressively from more commonly occurring evidence.[14] Choosing smaller values for σ^2 increases the strength of penalty, in effect decreasing the predisposition to generalize from small amounts of evidence. In practice, however, I did not find that model performance was especially sensitive to the particular value of σ^2. The results reported in this chapter use a value of $\sigma^2 = 1.0$.

3.5 Features

Each subphase of the dependency reconstruction algorithm involves the training of a separate model and the development of a separate feature set. I found that it was important to include both a variety of general feature templates and a number of manually designed, specialized features to resolve specific problems observed for individual classifiers. I developed all feature templates exclusively on the training and development sets specified in Section 3.2.

Table 3.1 shows the general feature templates in each classifier. The features are coded as follows. The prefixes {∅,M,G,D,R} indicate that the feature value is calculated with respect to the node in question, its mother, grandmother, daughter

[14] In addition, perfect-predictor features have finite optimal weights under regularization.

Feature type	IdentNull	InsertNull	IdentMoved	RelocMoved	InsertReloc	IdentLocus	InsertLocus	FindController
Tag						✓		✓
Hd								✓
Cat×mCat	⊗							✓
Cat×mCat×gCat			✓			✓	✓	
Cat×Hd×mCat×mHd			⊗					
Cat×Tag×mCat×mTag			⊗					
Cat×Tag	✓					✓		
Cat×Hd						⊗		
(First/Last)Cat		✓			✓			
(l/rSis)Cat		✓			✓			
dPos×Cat		✓						
Path				✓				✓
Cat×rCat				✓				
Tag×rCat				✓				
Cat×Tag×rCat				✓				
Cat×rCat×dPos					✓			
Hd×rHd				⊗				
Cat×Hd×rHd				✓				
Cat×dCat	✓			✓		✓		✓
mHd×Hd			⊗					
# Special	9	0	11	0	0	12	0	3

Table 3.1: Shared feature templates. See text for template descriptions. # Special is the number of special templates used for the classifier. ⊗ denotes that all subsets of the template conjunction were included.

(applied once for each daughter in the local tree), or *relative* node respectively.[15] {Cat,Pos,Tag,Hd} stand for syntactic category, position (of daughter) in mother, head tag, and head word respectively. For example, when determining whether an infinitival VP is extraposed, such as S-2 in Figure 3.1, the plausibility of the VP head being a deep dependent of the head verb is captured with the mHd×Hd template. (First/Last)Cat and (l/rSis)Cat are templates used for choosing the position to insert relocated nodes, recording respectively whether a node of a given category is the first/last daughter, and the syntactic category of a node's left/right sisters. Path is the syntactic path between relative and base node, defined as the list of the syntactic categories on the (inclusive) node path linking the relative node to the node in question, paired with whether the step on the path was upward or downward.

[15] The relative node is Dislocated in RelocMoved and Locus in FindController.

For example, in Figure 3.2 the syntactic path from VP-1 to PP is [↑-VP,↑-S,↓-VP,↓-PP]. This is a crucial feature for the relativized classifiers RelocateMoved and FindController; in an abstract sense it mediates the gap-threading information incorporated into GPSG-style (Gazdar et al., 1985) parsers, and in concrete terms it closely matches the information derived from Johnson (2002)'s connected local tree set patterns. Gildea and Jurafsky (2002) is to my knowledge the first use of such a feature for classification tasks on syntactic trees; they found it important for the related task of semantic role identification.

I expressed specialized hand-coded feature templates as tree-matching patterns that capture a fragment of the content of the pattern in the feature value. Representative examples appear in Figure 3.3. The italicized node is the node for which a given feature is recorded; underscores _ indicate variables that can match any category; and the angle-bracketed parts of the tree fragment, together with an index for the pattern, determine the feature value. Details of specialized feature templates are given in Appendix B.

3.6 Results

3.6.1 Testbed

The datasets used for this study consist of the Wall Street Journal section of the Penn Treebank of English (WSJ) and the context-free version of the NEGRA (version 2) corpus of German (Skut et al., 1997b), as described in Section 3.2. Full-size experiments on WSJ described in this chapter used the standard sections 2–21 for training, 24 for development, and trees whose yield is under 100 words from section 23 for testing. In the comparative English/German experiments described in Section 3.6.4, I used for English the same development and test sets but files 200–059 of WSJ as a smaller training set; for German I followed Dubey and Keller (2003) in using the first 18,602 sentences of NEGRA for training, the final 1,000 sentences of NEGRA for development, and the remaining 1,000 for testing. In keeping with the output format of available Treebank-style parsers, I stripped all functional tags from the syntactic

	Gold trees		Parser output		
	Jn	Pres	Jn	DD	Pres
NP-*	62.4	75.3	55.6	(69.5)	61.1
WH-t	85.1	67.6	80.0	(82.0)	63.3
0	89.3	99.6	77.1	(48.8)	87.0
SBAR	74.8	74.7	71.0	73.8	71.0
S-t	90	93.3	87	84.5	83.6

Table 3.2: Comparison with previous work using Johnson's PARSEVAL metric. Jn is Johnson (2002); DD is Dienes and Dubey (2003b); Pres is the present work.

categories of input context-free trees prior to both training and testing (though in several cases this seems to have been a limiting move; see Section 3.8).

3.6.2 Comparison with previous work

If we think of a statistical parser as a function from strings to CF trees, and the nonlocal dependency recovery algorithm A presented in here as a function from trees to trees, we can naturally *compose* the algorithm with a parser P to form a function $A \circ P$ from strings to trees whose representation is, hopefully, an improvement over the trees from P.

Johnson (2002) introduced a PARSEVAL-based evaluation metric for evaluating empty-annotation recovery on context-free parse trees: a correct empty category inference requires the *string position* of the empty category, combined with the left and right boundaries plus syntactic category of the antecedent, if any. We can compare the algorithm's performance on this metric with the results of Johnson (2002) and Dienes and Dubey (2003a) on WSJ. Valid comparisons exist for the insertion of uncoindexed empty nodes (Comp and Arb-Subj), identification of control and raising loci (ControlLocus), and pairings of dislocated and controller/raised nodes with their origins (Disloc,Controller). Table 3.2 presents results using this metric in comparison with Johnson (2002) and Dienes and Dubey (2003a).[16,17] Note that this evaluation metric does not require correct attachment of the empty category into the parse tree. In Figure 3.1, for example, WHNP-1 could be erroneously remapped to the right edge of any S or VP node in the sentence without resulting in error according to this metric. I therefore abandon this metric in further evaluations as it is not clear whether it adequately approximates performance in predicate-argument structure recovery.[18]

3.6.3 Composition with a context-free parser

The evaluation metric used in the previous section is insufficiently stringent, and fails to capture the essential information that non-local dependency (or discontinuous-constituency) annotation is mean to provide. Furthermore, it gives us no way to address the major issue motivating non-local dependency algorithms in the first place: to what extent does the context-free simplification of constituency/dependency fail to capture the structural relationships present in the sentence of a natural language? In Sections 1.3.2 and 1.4, I showed that discontinuous constituency, head-sister phrase structure relationships that do not express government, and crossing dependency are different facets of the same phenomenon. This suggests that a natural way of comparing context-free trees with and without extra non-local dependency annotation is to (deterministically) induce dependency trees from the headed constituency trees resulting from a CF parser with and without a non-local dependency algorithm. Such a dependency tree can be compared against the gold-standard dependency tree induced

[16] For purposes of comparability with Johnson (2002) I used Charniak's 2000 parser in the "parser output" evaluation.

[17] This algorithm was evaluated on a more stringent standard for NP-* than in previous work: control loci-related mappings were done after dislocated nodes were actually relocated by the algorithm, so an incorrect dislocation remapping can render incorrect the indices of a correct NP-* labeled bracketing. Additionally, our algorithm does not distinguish the syntactic category of null insertions, whereas previous work has; as a result, the null complementizer class 0 and WH-*t* dislocation class are aggregates of classes used in previous work.

[18] Collins (1999) reports 93.8%/90.1% precision/recall in his Model 3 for accurate identification of relativization site in non-infinitival relative clauses. This figure is difficult to compare directly with other figures in this section; a tree search indicates that non-infinitival subjects make up at most 85.4% of the WHNP dislocations in WSJ.

	P_{CF}	P	$A \circ P$	$J \circ P$	D	G	$A \circ G$	$J \circ G$
Overall	91.2	87.6	**90.5**	90.0	88.3	95.7	**99.4**	98.5
NP	91.6	89.9	**91.4**	91.2	89.4	97.9	**99.8**	99.6
S	93.3	83.4	**91.2**	89.9	89.2	89.0	**98.0**	96.0
VP	91.2	87.3	**90.2**	89.6	88.0	95.2	**99.0**	97.7
ADJP	73.1	72.8	**72.9**	72.8	72.5	99.7	**99.6**	98.8
SBAR	94.4	66.7	**89.3**	84.9	85.0	72.6	**99.4**	94.1
ADVP	70.1	69.7	69.5	**69.7**	67.7	99.4	99.4	**99.7**

Table 3.3: Typed dependency F1 performance when composed with statistical parser. P_{CF} is parser output evaluated by context-free (shallow) dependencies; all others are evaluated on deep dependencies. P is parser, G is string-to-context-free-gold-tree mapping, A is present remapping algorithm, J is Johnson 2002, D is the Combined model of Dienes 2003.

from an enhanced CF tree as it appears in the annotated corpus.

To test this idea quantitatively, I evaluate performance with respect to recovery of *typed dependency relations* between words. A dependency relation is defined at a node N of a lexicalized parse tree as a pair $\langle w_i, w_j \rangle$ where w_i is the lexical head of N and w_j is the lexical head of some non-head daughter of N. Dependency relations may further be typed according to information at or near the relevant tree node; Collins (1999), for example, reports dependency scores typed on the syntactic categories of the mother, head daughter, and dependent daughter, plus on whether the dependent precedes or follows the head. I present here dependency evaluations where the gold-standard dependency set is defined by the *remapped tree*, typed by syntactic category of the mother node.[19] In Figure 3.1, for example, *to* would be an ADJP dependent of *quick* rather than a VP dependent of *was*; and *Farmers* would be an S dependent both of *to* in *to point out ...* and of *was*. Trees are lexicalized using the head-finding rules of Collins (1999), and assuming that null complementizers do not participate in dependency relations. To further compare these results with previous work, I obtained the output trees produced by Johnson (2002) and Dienes (2003) and evaluated them on typed dependency performance. Table 3.3 shows the results of this evaluation.

[19] Unfortunately, 46 WSJ dislocation annotations in this testset involve dislocated nodes dominating their origin sites. It is not entirely clear how to interpret the intended semantics of these examples, so they are ignored in evaluation.

	Performance on gold trees							Performance on parsed trees					
	ID			Rel	Combo			ID			Combo		
	P	R	F1	Acc	P	R	F1	P	R	F1	P	R	F1
WSJ(full)	92.0	82.9	87.2	95.0	89.6	80.1	84.6	34.5	47.6	40.0	17.8	24.3	20.5
WSJ(sm)	92.3	79.5	85.5	93.3	90.4	77.2	83.2	38.0	47.3	42.1	19.7	24.3	21.7
NEGRA	73.9	64.6	69.0	85.1	63.3	55.4	59.1	48.3	39.7	43.6	20.9	17.2	18.9

Table 3.4: Cross-linguistic comparison of dislocated node identification and remapping. ID is correct identification of nodes as ± dislocated; Rel is relocation of node to correct mother given gold-standard data on which nodes are dislocated (only applicable for gold trees); Combo is both correct identification and remapping.

For comparison, I include shallow dependency accuracy for Charniak's parser under P_{CF}.

3.6.4 Cross-linguistic comparison

In order to compare the results of nonlocal dependency reconstruction between languages, we must identify equivalence classes of nonlocal dependency annotation between treebanks. NEGRA's nonlocal dependency annotation is quite different from WSJ, as described in Section 3.2, ignoring controlled and arbitrary unexpressed subjects. The natural basis of comparison is therefore the set of all nonlocal NEGRA annotations against all WSJ dislocations, excluding relativizations (defined simply as dislocated *wh-* constituents under SBAR).[20]

Table 3.4 shows the performance comparison between WSJ and NEGRA of IdentDisloc and RelocMoved, on sentences of 40 tokens or less. For this evaluation metric, I use syntactic category and left & right edges of (1) dislocated nodes (ID); and (2) originating mother node to which dislocated node is mapped (Rel). Combo requires both (1) and (2) to be correct. NEGRA is smaller than WSJ (∼350,000 words vs. 1 million), so for fair comparison we tested WSJ using the smaller training set described in Section 3.6.1, comparable in size to NEGRA's. Since the positioning of traces within NEGRA nodes is trivial, we evaluate remapping and combination

[20] The interpretation of comparative results must be modulated by the fact that I spent more total time on feature engineering for WSJ than for NEGRA, and I am not a native speaker of German.

	P_{CF}	P	$A \circ P$	G	$A \circ G$
WSJ(full)	76.3	75.4	75.7	98.7	99.7
WSJ(sm)	76.3	75.4	75.7	98.7	99.6
NEGRA	62.0	59.3	61.0	90.9	93.6

Table 3.5: Typed dependency F1 performance when composed with statistical parser. Remapped dependencies involve only non-relativization dislocations and exclude control loci.

performances requiring only proper selection of the originating mother node; the algorithm is thus carried out on both treebanks through step (2b). This is adequate for purposes of typed dependency evaluation in Section 3.6.3, since typed dependencies do not depend on positional information. State-of-the-art statistical parsing is far better on WSJ (Charniak, 2000) than on NEGRA (Dubey and Keller, 2003), so for comparison of parser-composed dependency performance I used vanilla PCFG models for both WSJ and NEGRA trained on comparably-sized datasets; in addition to making similar types of independence assumptions, these models performed relatively comparably on labeled bracketing measures for development sets (73.2% performance for WSJ versus 70.9% for NEGRA).

Table 3.5 compares the testset performance of algorithms on the two treebanks on the typed dependency measure introduced in Section 3.6.3.[21]

3.7 Data analysis

3.7.1 General Trends

First, there is a considerable amount of annotation inconsistency for nonlocal dependencies in the Penn Treebank, in particular revolving around control loci and their antecedents. Inconsistencies include

- whether a passive verb governs a control locus, and if it does, whether that locus has an antecedent

[21] Many head-dependent relations in NEGRA are explicitly marked, but for those that are not I used a Collins (1999)-style head-finding algorithm independently developed for German PCFG parsing.

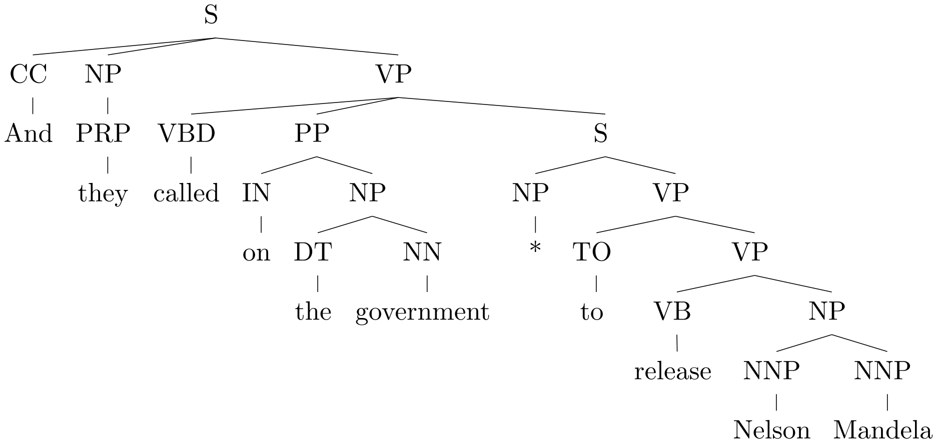


Figure 3.4: An ambiguous shared dependency

- whether control locus subjects of infinitival and gerund phrases are given antecedents

3.7.2 Genuine difficulties in nonlocal dependency reconstruction

In terms of control relations, one of the most genuinely difficult disambiguation tasks is to determine subject versus object control for infinitival VP complements in transitive (or otherwise two-argument) clauses. In Figure 3.4, the infinitival VP *to release Nelson Mandela* is truly ambiguous between being a purpose clause, in which case the infinitival VP complement should be coindexed with the subject *they*, and being a verbal complement, in which case it should be coindexed with the object *the government* (due to properties of the verb+preposition complex *call on*).[22] In this particular case, I judge the verbal complement reading to be more plausible, and in fact the annotator coindexed it with the object. But from the machine learning perspective, a number of facts have to be recognized to get this judgement correct:

[22] See Chapter 14 of Bies et al. (1995) for more details. Ironically (p. 246), *call on* is *not* one of the verbs listed as being able to select for, and thus specify object coindexation for, an infinitival complement. Although the annotator was clearly right in treating this example as an object-coindexed infinitival complement, the coindexation was actually annotated on the entire PP *on the government* rather than the correct target *the government*.

advise	8	detail	0	instruct	6	request	2
ask	78	direct	4	invite	16	recommend	0
beg	3	enjoin	0	order	19	teach	5
beseech	0	exhort	0	persuade	29	tell	12
challenge	1	forbid	2	pray	0	urge	53
command	0	implore	1	promise	2		
counsel	0	incite	0	remind	0		

Table 3.6: Sparsity of ditransitive verbs selecting infinitival VP complement: count of cooccurrences with infinitival VP

1. the VP should occur in an external environment that licenses a controlled subject;

2. the lexical governor of the infinitive is *call*

3. the PP sister, headed by *on*, is actually a determining factor in the relevant verbal lemma (c.f. *call on the phone*);

4. *call on* can select for an infinitive VP whose subject gets coindexed with the NP inside the *on* PP.

5. the infinitival VP actually is a selected complement in this particular case.

Even if we hard-coded the property of item 4, all other properties would have to be learned in the classifier, and items 3 and 5 could easily be dependent on the identity of the verb in question. But as we can see in Table 3.6, many such verbs are quite sparse in the training set (note that these are counts of stemmed forms, so inflected form counts are sparser still). Due to this sensitivity to the properties of a lexical governor, subject versus object control is an ambiguity where maximum entropy classification is able to do quite well in comparison to purely structure-based dependency recovery algorithms, but it is also highly susceptible to data sparsity and thus an inherently difficult ambiguity.

3.7.3 Parsed input and degradation of nonlocal dependency identification

Central to the question of whether a first phase of context-free parse disambiguation is a *safe* approximation is the degree to which nonlocal dependencies that are reliably identifiable from perfectly parsed context-free trees become incorrectly identified for state-of-the-art parsed input. An error in nonlocal dependency identification induced by an error in the context-free parse can be termed an *error cascade*. One simple measure of the frequency of error cascades lies in Table 3.3. If we compare the overall f-score improvement achieved by composing the non-local dependency algorithm to gold-standard trees versus automatically-parsed trees (4.7% versus 3.9%, judged by comparing the difference between $A \circ G$ and G with the difference between $A \circ P$ and P), we see that over 80% of the benefit on gold-standard trees carries over to automatically-parsed trees. For a state-of-the-art context-free parser of English, error cascades do not compromise the utility of our algorithm. The picture for composition with a vanilla PCFG parser is rather different: in Table 3.5, we can see that most of the dependency improvement achieved on gold-standard trees disappears on automatically-obtained parse trees.

It is also useful to look at the error cascades that do exist in the composition with the Charniak parser. One of the major cascades involves NP/VP attachment ambiguity and control. In Figure 3.5, the phrase *to take matters into their own hands* is correctly analyzed as the complement of *threats*. The general rule for ETB is that null subjects of infinitival VP complements inside NPs should not be assigned antecedents (see Bies et al. 1995, p. 241), whereas inside VPs they typically *are*, although ETB annotation is not entirely consistent in this regard.[23] The positioning of the infinitival complement clause induces an attachment ambiguity, however: in

[23] From a theoretical linguistic standpoint, the division of coindexation by governing category of infinitival complement is not necessarily well-founded. The ETB manual specifies that null subjects for infinitival complement clauses of VPs should be coindexed “with whatever lexical NP it is associated with” (presumably on intuitive grounds). But the same criterion could easily be applied to many null subjects in infinitival complement clauses of NPs, including that in Figure 3.5 (where the coindexed antecedent would presumably be the first appearance of the possessive pronoun *their*). A good deal of evidence, including cases of complement coercion such as the following:

addition to its correct association with only the right NP conjunct *threats*, higher attachments are possible, either at the level of coordinate NP mother *their agitation and threats* and as a verbal in association with the VP *step up their efforts*, in which case the infinitival complement would be interpreted as a purposive clause. The parser (quite reasonably, in this case) misattaches the infinitival complement to the VP; the nonlocal dependency recovery algorithm correctly identifies a control locus, but the underlying parse error incorrectly leads to the assignment of the subject *the large number of right-wing white males . . .* as its antecedent.

3.8 Discussion

The WSJ results shown in Tables 3.2 and 3.3 suggest that discriminative models incorporating both non-local and local lexical and syntactic information can achieve good results on the task of non-local dependency identification. On the PARSEVAL metric, this algorithm performed particularly well on null complementizer and control locus insertion, and on S node relocation. In particular, Johnson noted that the proper insertion of control loci was a difficult issue involving lexical as well as structural sensitivity. I found the loglinear paradigm a good one in which to model this feature combination; when run in isolation on gold-standard development trees, the model reached 96.4% F1 on control locus insertion, reducing error over the Johnson model's 89.3% by nearly two-thirds. The high performance is also evident in the substantial contribution to typed dependency accuracy seen in Table 3.3. For gold-standard input trees, the algorithm reduces error by over 80% from the surface-dependency baseline, and over 60% compared with Johnson's results. For parsed input trees, the algorithm reduces dependency error by 23% over the baseline, and by 5% compared

> The prison administration promised them$_i$ $*_i$ to be allowed being held together if they would stop their hungerstrike. (*http://www.blythe.org/nytransfer-subs/Covert_Actions/Palestine:_Updates_on_Internationalist_Prisoners*, May 19, 2005)
>
> Ed Heier flew by, and signalled $*_i$ to 'join up', but Magee$_i$ was too confused to respond. (*http://www.acepilots.com/usmc_magee.html*, May 19, 2005)

suggests that the underlying basis for control assignment is semantic rather than syntactic in any sense. See Pollard and Sag (1994), Chapter 7, for a detailed discussion.

with Johnson's results. Note that the dependency figures of Dienes lag behind even the parsed results for Johnson's model; this may well be due to the fact that Dienes built his model as an extension of Collins (1999), which lags behind Charniak (2000) by about 1.3–3.5%.

Manual investigation of errors on English gold-standard data revealed two major issues that suggest further potential for improvement in performance without further increase in algorithmic complexity or training set size. First, as suggested in Section 3.7.1, annotation inconsistency accounted for a large number of errors, particularly false positives. VPs from which an S has been extracted (*[$_S$Shut up,] he [$_{VP}$ said* t*]*) are inconsistently given an empty SBAR daughter, suggesting the cross-model low-70's performance on null SBAR insertion models (see Table 3.2) may be a ceiling. Control loci were often under-annotated; the first five development-set false positive control loci I checked were all due to annotation error. And *why*-WHADVPs under SBAR, which are always dislocations, were not so annotated 20% of the time. Second, both control locus insertion and dislocated NP remapping must be sensitive to the presence of argument NPs under classified nodes. But temporal NPs, indistinguishable by gross category, also appear under such nodes, creating a major confound. I used customized features to compensate to some extent, but temporal annotation already exists in WSJ and could be used. Note that Klein and Manning (2003b) independently found retention of temporal NP marking useful for PCFG parsing.

As can be seen in Table 3.3, the absolute improvement in dependency recovery is smaller for both this and Johnson's postprocessing algorithms when applied to parsed input trees than when applied to gold-standard input trees. It seems that this degradation is *not* primarily due to noise in parse tree outputs reducing recall of nonlocal dependency identification: precision/recall splits were largely the same between gold and parsed data, and manual inspection revealed that incorrect nonlocal dependency choices often arose from syntactically reasonable, yet incorrect, input from the parser. I gave examples of these error cascades in Section 3.7.3.

The English/German comparison shown in Tables 3.4 and 3.5 is suggestive, but caution is necessary in its interpretation due to the fact that differences in both language structure and treebank annotation may be involved. Results in the *G* column of Table 3.5, showing the accuracy of the context-free dependency approximation from gold-standard parse trees, quantitatively corroborates the intuition that nonlocal dependency is more prominent in German than in English.

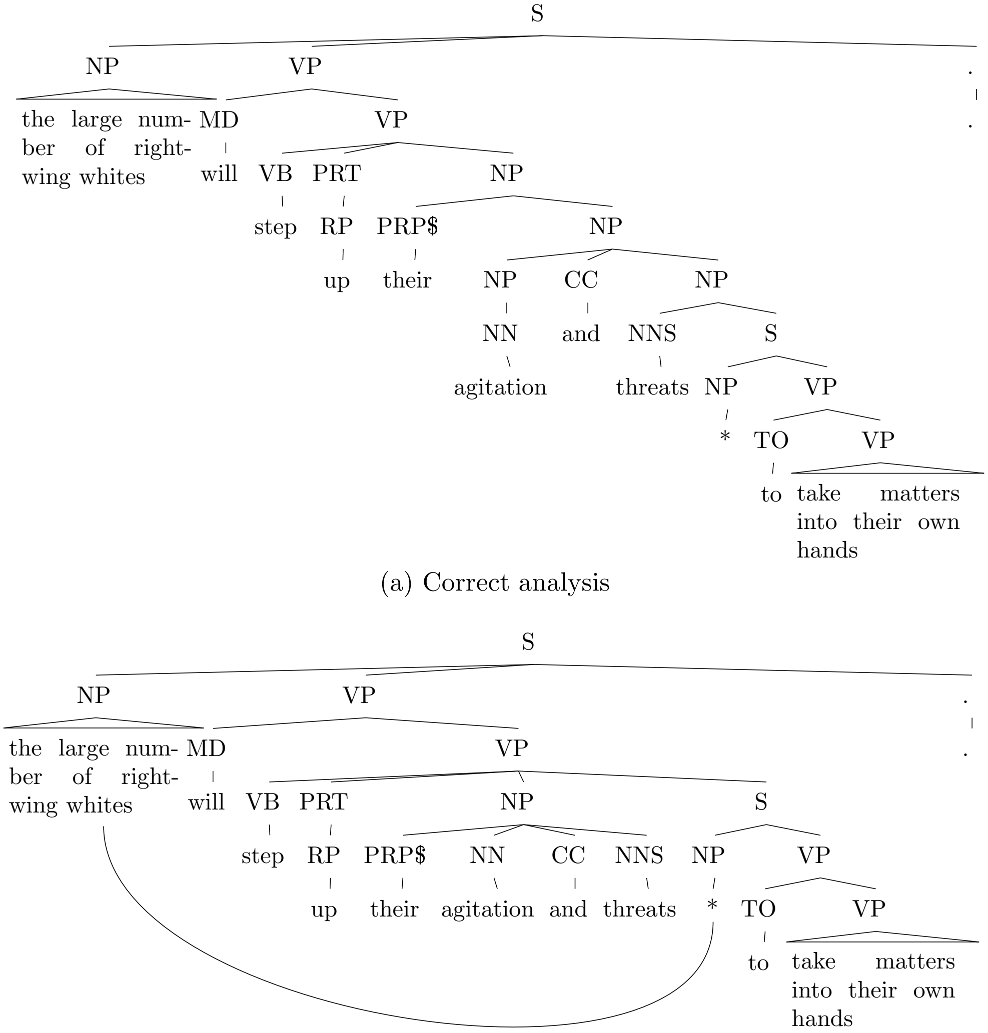
Manual investigation of errors made on German gold-standard data revealed two major sources of error beyond sparsity. The first was a widespread ambiguity of S and VP nodes within S and VP nodes; many true dislocations of all sorts are expressed at the S and VP levels in CFG parse trees, such as VP-1 of Figure 3.2, but many adverbial and subordinate phrases of S or VP category are genuine dependents of the main clausal verb. I was able to find a number of features to distinguish some cases, such as the presence of certain unambiguous relative-clause introducing complementizers beginning an S node, but much ambiguity remained. The second source of error was the ambiguity that some matrix S-initial NPs are actually dependents of the VP head (in these cases, NEGRA annotates the finite verb as the head of S and the non-finite verb as the head of VP). This is not necessarily a genuine discontinuity per se, but rather corresponds to identification of the subject NP in a clause. Having access to reliable case marking would improve performance in this area; such information is in fact included in NEGRA's morphological annotation, another argument for the utility of involving enhanced annotation in CF parsing.

As can be seen in the right half of Table 3.4, performance falls off considerably on vanilla PCFG-parsed data. This fall-off seems more dramatic than that seen in Sections 3.6.2 and 3.6.3, no doubt partly due to the poorer performance of the vanilla PCFG, but likely also because only non-relativization dislocations are considered in Section 3.6.4. These dislocations often require non-local information (such as identity of surface lexical governor) for identification and are thus especially susceptible to degradation in parsed data. Nevertheless, seemingly dismal performance here still provided a strong boost to typed dependency evaluation of parsed data, as seen in $A \circ P$ of Table 3.5. This may indicate that dislocated terminals are being usefully identified and mapped back to their proper governors, even if the syntactic projections of these terminals and governors are not being correctly identified by the parser.

3.9 Conclusion

In this chapter I have presented a high-performance algorithm for identifying nonlocal syntactic relationships—both *dislocated* and *shared* dependencies—in context-free parse trees. When applied to gold-standard context-free input tree, it correctly recovers 86% of nonlocal dependencies, a 57% relative error reduction over the best previous work. When applied to the input trees from a state-of-the-art context-free parser, it reduces the overall dependency error by 23%, a 5% improvement over the best previous work. Perhaps the outstanding feature of the algorithm presented here is its integration of structural and lexical information through log-linear discriminative models.

The basic motivation for enhancing the annotation of context-free parse trees of natural language sentences is that headed context-free trees (which constitute the output of most available statistical parsers) only explicitly represent *local* dependency relations. The non-local syntactic annotations already available in treebanks can, however, be interpreted as determining non-local dependency relations. Consistent with linguistic intuition, we found in this chapter that the *context-free* approximation of dependency structure implicitly embedded in most Treebank-based broad-coverage parsers is quite adequate for English, but less accurate for German. This difference in adequacy consists of two dimensions. First, a higher proportion of the dependencies found in English newspaper text are context-free than in German newspaper text. Second, for English greedy inference at the level of state-of-the-art context-free parsing does not seriously degrade the ability of a high-accuracy non-local dependency identifying algorithm, such as the one introduced here, to recover the complete dependency structure given a context-free input tree. For German, the state of the art in context-free parsing is rather less successful, and we see greater degradation in non-local dependency recovery. One conclusion that might be drawn from this is that greedy inference at the level of context-free parsing is not as safe for freer word order languages like German as it is for English. This leads us to consider parsing algorithms that directly integrate the continuous and discontinuous constituency, the topic of Chapter 4.



(a) Correct analysis

(b) Control error induced by misattachment of infinitive VP

Figure 3.5: Incorrect identification of a shared dependency due to error in parse input

Chapter 4

Direct discontinuous constituency parsing with probabilistic wrap grammars

In this chapter I conduct a preliminary investigation into the *direct* probabilistic parsing of discontinuous constituency—that is, parsing such that discontinuous constituents are directly represented as intermediate derivational items. I add a probabilistic component to the *multiple context-free grammar* (MCFG) formalism (Seki et al., 1991) and use a bottom-up agenda-based parsing algorithm, together with A* search heuristics (Klein and Manning, 2003a), to implement optimal discontinuous-constituent parsing. I also investigate how to incorporate *distance sensitivity* into a probabilistic MCFG, and argue that distance sensitivity is important to both efficiency and accuracy in probabilistic MCFG parsing.

4.1 Introduction and Background

In this section I briefly discuss the option of learning *generative* versus *discriminative* probabilistic models for parsing, and then discuss the mildly context-sensitive grammatical formalisms that permit tractable discontinuous constituency parsing.

4.2 Generative versus Discriminative Models

There are two principal approaches to the problem of classification in probabilistic machine learning. Formally, classification is the problem of choosing the hidden class H^* with the highest *conditional likelihood* given an observation O:

$$H^* = \arg\max_H P(H|O) \tag{4.1}$$

The requirement is therefore to come up with a conditional likelihood model for $P(H|O)$. This is typically done in one of two ways: *discriminatively* or *generatively*.

Discriminative models directly estimate $P(H|O)$. Logistic regression or maximum-entropy models are perhaps the most widely used probabilistic discriminative models in machine learning. Discriminative models are considered to have two primary advantages. First, training a probability model typically involves a search for parameter values of the model that maximize the model's likelihood. With discriminative models, the maximized likelihood is the likelihood that is actually used for classification. Second, it is possible to use arbitrary features of observed data in probability estimates of hidden structure, regardless of overall model complexity. This contrasts with generative models for reasons that will momentarily become clear.

Generative models directly estimate the *joint likelihood* of hidden plus observed datapoints, $P(H, O)$, and estimate the conditional likelihood via Bayes' rule:

$$P(H|O) = \frac{P(O, H)}{P(O)} \tag{4.2}$$

In classification, because $P(O)$ is constant, the candidate H^* with optimal joint likelihood is also the candidate with optimal conditional likelihood. Typical generative models used in natural language tasks, such as for document classification, tagging, or parsing, treat observations as conditionally dependent on hidden variables, rather than vice versa. In typical generative probabilistic parsing models, for example, each word w occurs in a terminal node of the tree, and the identity of w is conditionally dependent on the part-of-speech tag in the dominating preterminal node, and sometimes on more distant features of the tree such as the governing word, part of speech, and category. Because O is conditionally dependent on H, it is not possible to incorporate arbitrary features of observations into joint generative models over complex hidden classes, because these features must also be represented within the hidden classes to affect joint probabilities. Furthermore, training of generative models typically involves maximizing joint likelihood, but this likelihood is not directly used in classification, creating the possibility that the trained model is not optimal for the intended task (see Johnson 2001; Klein and Manning 2002 for discussion).

Despite these apparent disadvantages, in this chapter I investigate generative syntactic models for discontinuous constituency. There are several reasons for this. First, because generative syntactic models produce an optimized joint distribution over H, O pairs, they implicitly define a distribution over observations O. They therefore have theoretical interest as language models for speech recognition, or, as I have argued in Chapter 2, as plausible models of performance in human sentence processing. Discriminative models cannot readily be so applied, because they do not produce a reliable distribution over O.

Second, the purported advantage of incorporating arbitrary input-based features enjoyed by discriminative models has not been shown to be effective in broad-coverage, Treebank-style parse settings. It is notable that the best broad-coverage parsers of English newswire (Charniak, 2000; Collins, 2000; Bod, 2003; Charniak and Johnson, 2005) all involve a generative component, and all except Collins' and Charniak and Johnson's are exclusively generative.[1] When used for parse selection on the output of hand-crafted grammars, on the other hand, discriminative models enjoy a considerable advantage (Toutanova et al., 2005; Riezler et al., 2002). This contrast naturally arises from the fact that parses can be tractably enumerated and individually ranked in the hand-crafted grammar case, whereas in the Treebank grammar setting dynamic programming and hence strict locality assumptions regarding probabilistic dependence among components of hidden structure are essential to achieving tractability. This

[1] All state-of-the-art approaches to discriminative Treebank-style parsing, including Collins (2000), involve the application of discriminative models to reranking of an n-best parse list produced by a generative parser.

means that while complex, non-local interdependences between parts of hidden structure can be used in the hand-crafted grammar parse selection case, they cannot for Treebank-style parsing. Arbitrary *input* features could be used in broad-coverage parsing, but because the topology of parse trees is only dimly reflected in the linear ordering of observed strings, there is no guarantee that such input-exclusive features will consistently provide useful information.

While generative models have remained unsuperseded in Treebank-style broad-coverage parsing, however, the face of broad-coverage parsing itself may be open to future change. While broad-coverage and Treebank grammar have been nearly synonymous in the past, one particularly interesting result has been that of Kaplan et al. (2004), who achieved considerable success in broad-coverage parsing using a hand-crafted LFG grammar plus discriminative parse selection. While it is difficult to evaluate their results directly against those of broad-coverage statistical parsers on a completely level playing field, the possibility is enticing that high-quality broad-coverage parsing with hand-crafted grammars will become better and better in the future.

4.2.1 Mildly context-sensitive grammar formalisms

In Section 1.4 of Chapter 1, I outlined three major approaches to the recovery of non-local syntactic dependencies in statistical parsing. In the first approach, parsing is taken as a two-phase, serial process: the first phase approximates all dependencies as local, and the second phase corrects the resulting parse trees for nonlocal dependency. I investigated such an approach in Chapter 3. Both the second and third approaches identify nonlocal and local syntactic dependencies in a single, joint inferential step. The second approach simply enhances context-free parse trees by threading non-local dependencies through an enhanced *category* structure, in the fashion of Generalized Phrase Structure Grammar (GPSG; Gazdar et al. 1985). This is perhaps the most obvious approach to joint inference, and has already been investigated in both generative (Collins, 1999; Dienes, 2003; Dubey, 2004) and discriminative (Cahill et al., 2004) contexts, so I have not investigated it further. In the third approach, nonlocal dependency is captured through an enhancement of the *edge* structure of parse trees: tree nodes can be discontinuous constituents. This third approach is the topic of the present chapter.

The conceptual path leading from nonlocal dependency to discontinuous constituency trees turns out to be rather simple. As I point out in Section 1.3.2, for every dependency tree D over a given string S, there is a headed constituent parse tree over S that induces D through the head/sister relationships in each local tree. If we restrict ourselves to context-free headed parse trees—that is, trees where each node dominates a continuous substring—we can only induce dependency trees with no crossing branches. If we commit to generating crossing dependencies through head-sister relations on parse trees, we must therefore consider discontinuous constituency trees: trees where some nodes dominate discontinuous string spans. This chapter therefore concerns itself with parsing algorithms that can construct discontinuous constituency trees.

In full generality, parsing for discontinuous constituency trees cannot be tractable, because worst-case recognition involves considering and possibly storing every potential tree node for a string, and a string of length n there may be as many as 2^n such tree nodes (each subset of the words in the string may be the span of a tree node).[2] Fortunately, a great deal of work over the last three decades has gone into investigating precisely those aspects of natural language syntax that do not transparently lend themselves to context-free descriptions (either weak or strong). This work began in divergent frameworks, including Tree-Adjoining Grammars (Joshi et al., 1975; Joshi, 1985), Head Grammars (Pollard, 1984), Combinatory Categorial Grammars (Steedman, 2000), and Minimalist Grammars (Stabler, 1997), but has been strongly convergent: each formalization turns out to give rise to the same set of *mildly context-sensitive* languages, whose recognition remains tractable, although the worst-case recognition time is generally of higher-order polynomial complexity than the CFLs. Furthermore, each formalism is characterized by context-free *derivation* trees (Vijay-Shanker et al., 1987), and the grammars of one formalism can generally

[2] See Johnson (1985) for a parsing algorithm detailing just how to go about parsing with a grammar that allows arbitrary discontinuous constituency.

be converted to grammars of the other that lead to isomorphism in the derivation trees. When a given string is parsed in such a formalism, the nodes of the derivation tree can be identified with the nodes of the string's discontinuous constituency parse, from which a dependency tree can in turn be induced from head-sister derivational relationships. As I discuss in Section 4.3, the context-free property of the derivation trees is advantageous for probabilistic interpretation of these formalisms as well.

Equipped with a probabilistic instantiation of a mildly context-sensitive grammatical formalism—which I investigate under the moniker of *probabilistic wrap grammars*—I take up the problems of parameter estimation and efficient natural language parsing. In Section 4.5, I show that a simple form of "surface" parameter estimation is particularly easy to use given a corpus of discontinuous constituency trees, but that it fails to capture cross-cutting generalizations about constituency and linear order that are desirable from the perspective of NL syntax. In particular, "surface" parameter estimation fails to capture the *distance-sensitivity* that is quite prominent in many types of discontinuous constituency. The first results of Section 4.7 are consistent with this failure: the main barrier to efficient parsing is the construction of large numbers of discontinuous-constituent edges that should be disfavored on the grounds that the discontinuity is implausibly large (in terms of linear order). In the later part of Section 4.5, I show how to bring distance sensitivity into discontinuous constituency parsing by factorizing the probabilistic grammar into *immediate dominance* (ID) and *linear precedence* (LP) components, in the spirit of GPSG. The remainder of the chapter concerns itself with parameter estimation for an ID/LP-factorized probabilistic wrapping grammar.

4.3 Linear Context-Free Rewrite Systems and Generative Probabilistic Models

Given that we are interested in joint probabilistic models over syntactic structures specifying nonlocal dependency relations, we need a grammatical formalism $\mathcal{G}$ specifying a set of possible syntactic structures $T_{\mathcal{G}}$, and a probabilistic interpretation of the formalism to distribute probability mass over the structures. One crucial requirement is that the model be *proper* (or *non-deficient*): that is, that $\sum_{s \in T_{\mathcal{G}}} P(s) = 1$. This requirement immediately distinguishes between two classes of trans-CF grammatical formalism. Vijay-Shanker et al. (1987) point out that although unification-based grammatical formalisms (UGs; including Lexical-Functional Grammar (Kaplan and Bresnan, 1982) and Functional Unification Grammar (Kay, 1979)) and indexed grammars (IG; Gazdar 1985) share with Head Grammars (HGs) and Tree-Adjoining Grammars (TAGs) the ability to generate trans-CF languages, UGs and IGs are distinguished from HGs, TAGs, and CFGs in that the former can produce an unbounded number of dependent paths sharing an unbounded amount of information, whereas the latter cannot. Vijay-Shanker et al. (1987) generalize the latter class into Linear Context-Free Rewriting Systems (LCFRSs), which share the common property of *derivational independence*: every derivation structure is tree-shaped, and the set of derivational possibilities at any node of a partial derivational tree is completely independent of the context external to that node in the derivation tree. CFGs are a particularly transparent form of LCFRS in that the tree structure resulting from a derivation is completely isomorphic to the derivation tree itself. The formal definition of the LCFRSs is presented in Section 4.4.

Vijay-Shanker et al. (1987) show that LCFRSs always enjoy polynomial-time recognition. In a probabilistic context, their property of derivational independence gives rise to another important advantage: ease of parameter estimation. Suppose that we have a corpus of derivation trees in some LCFRS G, and we want to estimate the parameters of a probabilistic formulation of G, where each parameter denotes the probability of a particular rewrite of a given node in the derivation tree (so for each non-terminal symbol S in the LCFRS, $\sum_{\alpha} P(S \rightarrow \alpha) = 1$). Chi and Geman (1998) showed that the relative-frequency estimate of probabilistic parameters for a CFG always yields a proper, maximum-likelihood probability distribution. Nothing in their proof, however, hinged on CFGs as a grammar over trees or strings; so the proof can equally be applied to show that relative frequency is the maximum-likelihood proper estimate for parameters of an LCFRS. The parameter estimation problem for LCFRSs is therefore fairly easy.

UGs and IGs, in contrast, do not enjoy the property of derivational independence, hence relative frequency estimates do not in general produce a proper probability distribution. Abney (1997) discusses the reasons for deficiency, and shows one way to estimate proper probability distributions over unification-based grammars using random fields. This estimation process is, however, much more complicated than maximum-likelihood estimation for probabilistic LCFRSs.

4.4 Definition of probabilistic wrap grammars

Following the notation of Seki et al. (1991), a *multiple context-free grammar* (MCFG) is a tuple $G = (N, O, F, P, S)$ such that:

- N is a finite set of non-terminal symbols;
- V is a finite set of terminal symbols;
- O is a set of n-tuples ($n \geq 1$) of strings over a finite symbol set;
- F is a finite set of partial functions from $O \times \cdots \times O$ to O, and F_q is the set of partial functions from O^q to O that are in F;
- R is a set of rules, each of the form $A \to A_1 \cdots A_n; f$ (alternatively written as $A \to f[A_1 \cdots A_n]$) with $f \in F_n$; and
- S is the start symbol.

with the constraint that for every $f \in F_k$, if f^h is the component of f that determines the h^{th} tuple in the range of f, then is of the form

$$f^h(x_1, \cdots, x_n) = u_0 z_1 u_1 z_2 \cdots u_m z_m \tag{4.3}$$

where each u_i is a (possibly empty) string of terminals in $V*$, and each z_i is some component of one of the x_j tuples. Furthermore, each component of each x_j is used at least once by some f^h.[3]

[3] This is the component (f2) from Seki et al. (1991).

There are two other relevant conditions that can be placed on MCFGs. The first is the *information-lossless* condition of Seki et al. (1991):[4] that each component of each x_j is used at most (and hence exactly) once by f. Any MCFG that satisfies the information-lossless condition is also an LCFRS. The second condition, which I will call *strict concatenation*, is that every u_i in (4.3) is the empty string, unless f is a zero-arity function,[5] in which case u_0 must be a single terminal symbol. Seki et al. (1991) showed that imposing the information-lossless condition does not affect the weak generative capacity of MCFGs. It is also clear that imposing strict concatenation also has no effect, since an MCFG can simply be transformed into a strict-concatenation MCFG by interposing unary rewrites in between every non-terminal node and every terminal. I will call an MCFG that satisfies both the information-lossless and strict-concatenation conditions a *wrapping grammar* (WG).

A *probabilistic wrapping grammar*, or PWG, is then simply a WG combined with a probability function $P : R \rightarrow [0, 1]$ such that for each $X \in N$, $\sum_{X \rightarrow f[\alpha] \in R} P(X \rightarrow f[\alpha]) = 1$.

4.4.1 A simple probabilistic wrapping grammar

Although we can write out the component mappings f of rules $X \rightarrow f[X_1 \cdots X_n]$ explicitly as functions whose domains are lists of lists of symbols and whose ranges are lists of symbols, when we talk about specific rules it is generally more convenient to use a subscript notation on the categories to specify f implicitly. Consider for example a rule specifying that a two-component subject NP is wrapped around a one-component VP, producing a one-component S. A plausible WG rule for this would be S $\rightarrow f$[NP VP], where f_0 has one component:

$$f_0^1(x_{11}, x_{12}, x_{21}) = x_{11}x_{21}x_{12} \tag{4.4}$$

This rule can be much more succinctly expressed with subscript variables specifying the structure of f_0 (the $_\curlywedge$ indicates a split between components of a string tuple):

[4] This is the (f3) condition from Seki et al. (1991).

[5] The *arity* of $f : O^n \rightarrow O$ is simply n.

LHS		RHS	Probability
$\mathrm{S}_{\alpha\gamma\beta}$	→	$\mathrm{NP}_{\alpha\curlywedge\beta}$ VP_{γ}	1
$\mathrm{NP}_{\alpha\curlywedge\beta}$	→	NP_{α} RC_{β}	0.2
$\mathrm{NP}_{\alpha\beta\curlywedge\epsilon}$	→	Det_{α} N_{β}	0.5
$\mathrm{NP}_{\alpha\curlywedge\epsilon}$	→	PPro_{α}	0.3
VP_{α}	→	V_{α}	1.0
$\mathrm{RC}_{\beta\alpha\gamma}$	→	NP_{α} $\mathrm{VP/NP}_{\beta\curlywedge\gamma}$	1.0
$\mathrm{VP/NP}_{\beta\curlywedge\alpha}$	→	V_{α} RelPro_{β}	1.0
V_{α}	→	$\mathrm{arrived}_{\alpha}$	0.5
V_{α}	→	knew_{α}	0.5
Det_{α}	→	a_{α}	1.0
N_{α}	→	woman_{α}	1.0
PPro_{α}	→	I_{α}	1.0
RelPro_{α}	→	who_{α}	1.0

Figure 4.1: A simple probabilistic wrapping grammar

$$\mathrm{S}_{\alpha\gamma\beta} \rightarrow \mathrm{NP}_{\alpha\curlywedge\beta} \quad \mathrm{VP}_{\gamma} \tag{4.5}$$

Figure 4.1 shows a simple probabilistic wrapping grammar that generates both *in situ* and extraposed relative clauses for subjects, and wraps relative object pronouns around relative clauses.

4.5 PWG parameter estimation

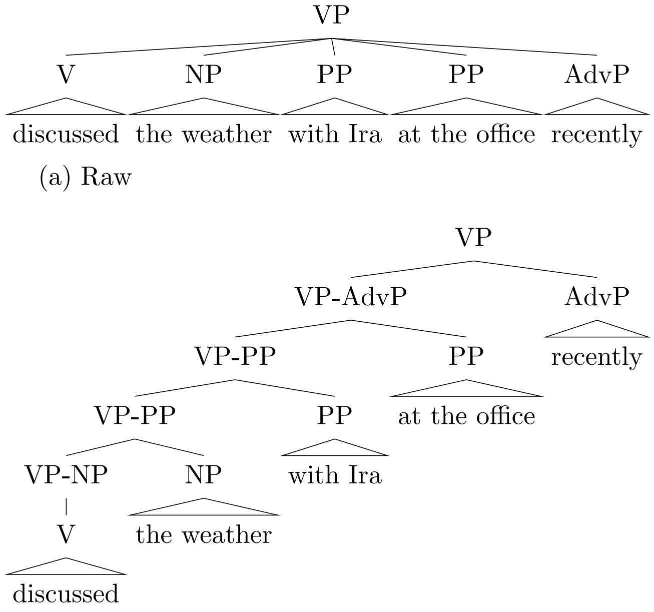
As noted in Section 4.3, the context-free derivation property of LCFRSs means that a simple means of parameter estimation for PWGs—maximum-likelihood estimation—can be employed, provided that we have a corpus of LCFRS derivation trees. No such corpus exists, however. What we *do* have access to is hand-parsed corpora of discontinuous constituency trees, including NEGRA and TIGER for German; the Prague Dependency Treebank; and, implicitly, all Penn Treebanks, including those for English, Chinese, and Arabic. In the rest of this chapter, I will address two closely related problems: estimating PWG models from discontinuous constituency treebanks; and efficient, accurate parsing with the resulting PWG models.

4.5.1 From treebank trees to wrap trees

The first guiding principle of PWG estimation we follow is to assume strict isomorphism between the tree structure of a discontinuous constituency parse and a LCFRS derivational history. That is, for each local tree $\mathrm{X} \rightarrow \mathrm{X}_1 \cdots \mathrm{X}_n$ in the discontinuous constituency parse we assume exactly one LCFRS rule of the form $\mathrm{X} \rightarrow \mathrm{X}_1 \cdots \mathrm{X}_n; f$. In a probabilistic setting, we might also be interested in *smoothing* the probabilities of these rules, by treating the observation of one LCFRS rule as evidence for the existence of related rules. One useful method of smoothing rule probabilities is to break down observed local trees into subcomponents with shorter right-hand sides. Thus we are faced with two important questions: first, how can we decompose a given local discontinuous constituency tree into a series of binary-branching trees (an important question in a probabilistic setting); and second, for a given binary-branching tree how do we determine the LCFRS component-matching function f?

Binarization

An important part of PCFG preprocessing has been the *binarization*, or equivalently, *markovization* process: transforming each n-ary headed local tree into a set of binary branching local trees, each of which encodes some subset of information from the original n-ary tree in the mothers and head daughter category structure of the new trees. This accomplishes two goals. First, no rule in the grammar read off of the resulting trees has more than two symbols on the right-hand side; this can facilitate parsing efficiency by minimizing the number of rule applications attempted. Second, depending on the amount of derivational history recorded in the binarized constituent labels, binarization can serve as a method of *smoothing* PCFG modelsby introducing new independence assumptions between the presence of what were jointly occurring siblings in the original trees. In this method of grammar smoothing, the number of original sister constituents encoded in each constituent label is known as the *Markov order* of smoothing. There are a variety of options available in context-free tree binarization (Roark and Johnson, 1999); in keeping with our interest in wrapping grammars as directly encoding head-dependent relations, we will restrict our scope to



P(VP → VP NP PP PP AdvP)	1.0

(a) Raw

P(VP → VP-AdvP AdvP)	1.0
P(VP-AdvP → VP-PP PP)	1.0
P(VP-PP → VP-PP PP)	0.5
P(VP-PP → VP-NP NP)	0.5
P(VP-NP → V)	1.0

(b) Smoothed

Figure 4.2: Smoothing a PCFG (Markov order 1)

head-outward binarization (Collins, 1999), where information from head daughter of the original local tree is retained through all binarized tree levels. Figure 4.2 shows the transformation and corresponding effect on MLE rewrite probabilities from head-outward binarization of a single local tree with Markov order 1.[6] Note that the presence or absence of a second or third PP modifier in the smoothed grammar is independent of the NP and AdvP sisters (whereas in the original grammar, a second PP modifier is obligatory and a third is impossible). In effect, the smoothed grammar distributes probability mass onto all rules of the form VP → NP PP^{+} AdvP (the $^{+}$ being the Kleene plus).

There is a limited set of options for head-outward binarization of context-free trees while keeping them context-free. Left- and right- headed local trees can only be binarized in one direction; trees whose head daughter has both left and right

[6] There are also minor degrees of freedom regarding exactly how to binarize in that the head daughter could be directly generated in the lowest binary tree, rather than as a separate unary as in Figure 4.2; and in that an additional unary could also be introduced above the highest binary tree, which is not done in Figure 4.2. These choices have no effect on the resulting probability models, and I ignore them in the remainder of the chapter.

sisters create some ambiguity, but they must still be binarized from the head out. Discontinuous constituency trees, on the other hand, can be binarized in any order. Furthermore, because some binarizations create intermediate constituents with more holes than others, the choice of binarization can affect computational complexity as well as smoothing effects. This leads to two natural criteria for choice of binarization: complexity minimization and likelihood maximization.

Since the maximum number of holes in a constituent determines the computational complexity of wrap grammar parsing, the complexity minimization criterion naturally leads to a *hole minimization* approach to binarization.

Adapting CFG head-finding algorithms

The binarization process outlined above, as well as the dependency-based evaluation metrics I use later in this chapter, presupposes a determination of the head daughter of each local tree. The prevalent head-finding algorithms used in the Treebank parsing literature have been defined for CF trees, and are generally definable as a nested iteration over a list of sets S of syntactic categories and a linear search among the daughters (in either left-to-right or right-to-left) order for the first daughter D whose syntactic category is in the first S (Magerman, 1994; Collins, 1999). In discontinuous constituency trees, however, complete-constituent precedence does not in general define a total linear ordering among sister nodes, so search order among the daughters is underdetermined.

One simple and intuitive way of adapting existing CF-tree head-finding algorithms to the discontinuous constituency context is to define precedence by the linear order of the head terminal of each node. Headship can then be unambiguously determined from the bottom up, and sister nodes are always totally ordered. This approach has the advantage of closely exploiting the isomorphism between headed phrase structure trees and word-dependency graphs. (In principle, of course, more complicated head-finding algorithms making reference to cases of incomplete precedence could be defined.) The adaptation of a CF head-finding algorithm is given in Figure 4.3.

```
function getHead(Tree $T$,HeadFinder $HF$)
    for each daughter $D_i$ of $T$ do
        if $D_i$ is a terminal node then
            $H_i \leftarrow$ string index position of $D_i$
        else
            $H_i \leftarrow$ string index position of the head terminal for getHead($D_i$,$HF$)
        end if
    end for
    $D' \leftarrow$ the list of $D_i$, ordered by their respective $H_i$
    return the daughter chosen when $HF$ is applied to $D'$
end function
```

Figure 4.3: Adaptation of context-free head-finding algorithm

4.5.2 "Surface tuple" estimation of PWGs

Adopting a principle of isomorphism between discontinuous constituency tree structure and derivational history still leaves the string tuple composition function f underspecified. The simplest approach to this component of the PWG estimation problem is to assume another strict isomorphism: between the "surface tuple" structure of every node in the discontinuous constituency tree and the underlying tuple structure of the corresponding node in the LCFRS derivation tree. (This is equivalent to saying that no LCFRS node has a string tuple component that is empty.) This additional isomorphism completely determines the identity f for each LCFRS node corresponding to a discontinuous constituency tree node, and thus determines a one-to-one relationship between trees in the corpus and LCFRS derivation trees. We can then use the maximum-likelihood estimates of all PWG parameters.

4.6 Parsing with PWGs

Harkema (2001) provides bottom-up, top-down, and Earley-style recognition algorithms for MCFGs. Any recognition algorithm can be easily converted to a parsing algorithm by tabulating partial derivations, and any parsing algorithm can be converted to a weighted parsing algorithm by including the weights (in this case probabilities) of partial derivations in the tabulation. In this chapter I focus on bottom-up

$$\begin{array}{rcc}
\textbf{Item form:} & [A, \vec{\alpha}] & \vec{\alpha} \text{ a list of integer pairs} \\
\textbf{Axioms:} & [A, ((i, i+1))] & A \rightarrow f[\epsilon], f[\epsilon] = w_{i+1} \\
\textbf{Goal:} & [ROOT, ((0, n))] & \\
\textbf{Inference Rule:} & \left.\begin{array}{c} I_A + O_A : [A, \vec{\alpha}] \\ I_B + O_B : [B, \vec{\beta}] \\ \hline I_A + I_B + Z + O_X : [X, \vec{\gamma}] \end{array}\right\} & \begin{array}{c} [X \rightarrow f[A\ B]] \in R \\ P(X \rightarrow f[A\ B]) = Z \\ f(\vec{\alpha} \bullet \vec{\beta}) = \vec{\gamma} \end{array}
\end{array}$$

Figure 4.4: Bottom-up weighted deduction scheme for binary PWGs. I_Y and O_Y are the inside and outside weights for item Y; Z is the weight of the invoked rule.

parsing, as it is most naturally compatible with the formulation of A* parsing as introduced in Klein and Manning (2003a). Figure 4.4 shows a deductive scheme for bottom-up parsing of binary-branching PWGs, in the style of Shieber et al. (1995). Each of the terms $\vec{\alpha}, \vec{\beta}, \vec{\gamma}$ is a list of index pairs that specify the beginning and ending of each substring component of the corresponding category.[7]

As noted by Nederhof (2003), such a deductive scheme can be adapted to finding lowest-weight (i.e., highest-probability) parsing by assigning weights to each item, and always processing the lowest-weight item, following Knuth's algorithm (Knuth, 1977). In Figure 4.4, these weights are subdivided into three components: inside weights I_Z, outside weights O_Z, and rule weights (the cost of invoking a rule) R. The inside weight of a chart item will be the exact weight of constructing the constituent from the terminals it dominates. In Figure 4.4, then, it must always be the case that $I_X = I_A + I_B + Z$. The outside weight is an *estimate* of the cost of reaching the goal from the item. Following Knuth, it is required that $I_A + I_B + Z + O_X \geq \max(I_A + O_A, I_B + O_B)$. In the simplest case, $O_Y = 0$ for all items Y; this corresponds to an LCFRS generalization of the bottom-up weighted deductive system given by Nederhof (2003) for CFGs. In the general case, O_Y must be a *monotonic, optimistic* estimate of Y's true outside weight; this leads to the A* parsing system introduced

[7] The equation $f(\vec{\alpha} \bullet \vec{\beta}) = \vec{\gamma}$ is shorthand for stating that α and β are disjoint, and that their indices align appropriately such that the result of the concatenation determined by f, applied to the multi-component substrings determined by α and β, is represented in the string by the new index pair list γ.

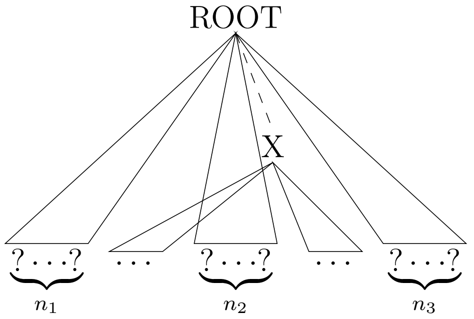


Figure 4.5: Outside probability estimates for an edge

for PCFG parsing by Klein and Manning (2003a).[8]

4.6.1 Outside probability estimates for A* parsing

In optimal probabilistic parsing, the value O in Figure 4.4 is interpreted as an estimate of the Viterbi outside probability of the situated category $[X, \vec{\gamma}]$. The trivial value is $O = 0$, which corresponds to prioritizing items by their *inside probability.* I investigate both the inside-probability priority and a simple outside estimate for O: the highest possible Viterbi outside probability for an edge of category X covering $\vec{\gamma}$ in a sentence of the correct length (this corresponds to the SX estimate of Klein and Manning 2003a). For a k-component category, determining this outside estimate requires filling in a $k+1$-dimensional chart (one dimension for the number of non-terminals to the left of each component, plus one dimension for the number of non-terminals to the right of the entire edge, as shown in Figure 4.5).

[8] As noted by Nederhof, Knuth's algorithm is not limited to the bottom-up parsing proposed by Klein and Manning; but there is good reason to use bottom-up parsing in an A* setting, because the weight of a partial derivation can be decomposed into its inside and outside weights, and bottom-up parsing always determines *exact* inside weights of partial derivations. In top-down parsing regimes, in contrast, both inside and outside weights must in part be estimated, potentially leading to looser bounds and thus less efficient parsing.

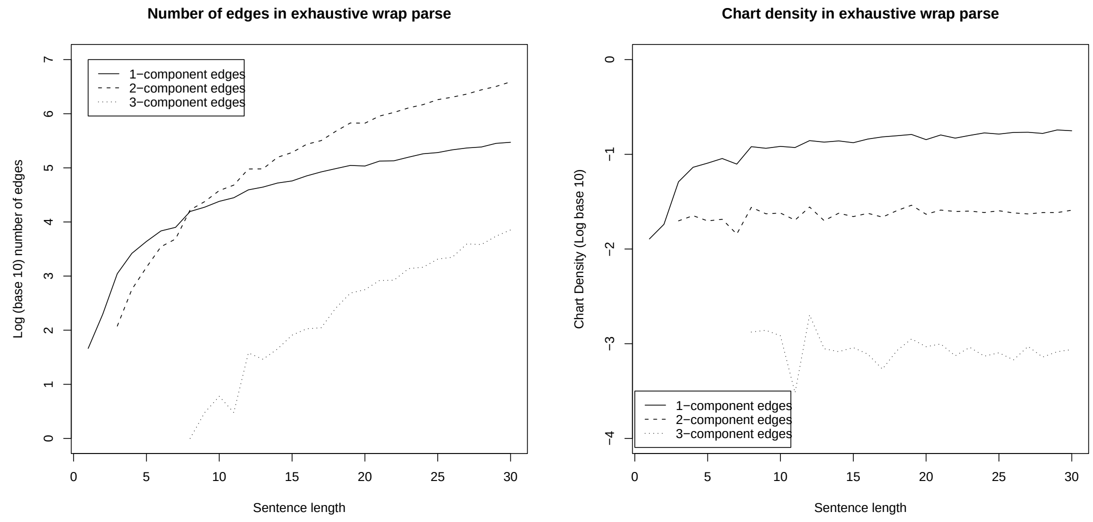


Figure 4.6: Chart density and number of edges by edge type in a complete chart parse

4.7 Parsing Efficiency

The most fundamental properties of chart parsing for PWGs involve the size and saturation of different components of the parse chart. I investigate these quantities first by using a very loosely fit surface-estimated PWG from sections 2-21 of the WSJ treebank (binarized with Markov factor 1, as explained in Section 4.5.1) and exhaustively parsing a sample of sentences from the training data. Figure 4.6 shows growth in the size of the complete parse chart as sentence length increases. We can see that chart density stabilizes around length 17 or so, and while the density of the chart (the number of observed N-component items divided by the number of theoretically possible N-component items, for a particular N) is high (> 10%) for continuous (1-component) items, it is lower for 2-component items and quite low for 3-component items. But because the number of k-component items grows as n^{2k} in the length n of the sentence, the stability of chart density means that the k-component part of the chart grows more quickly for higher k. In particular, 1-component items quickly come to dominate: by sentence length 20, they comprise over 90% of the chart. Our major concern, then, is how to limit their growth.

We might expect that simple A* item prioritization would contribute significantly to capping this growth. Figure 4.7 shows the effects of inside-probability and simple outside-estimate (based on syntactic category and the lengths of all the gaps between components) priorities of chart saturation. The results here are consistent with those of Caraballo and Charniak (1998); Klein and Manning (2003a) in that inside probability prioritization yields only a small effect on chart size; this result extends to discontinouous edges as well. The simple outside-estimate prioritization used here makes a larger contribution, but fails to change the qualitative picture that 2-component edges dominate the chart.

There are, of course, a variety of technical remedies available that can improve parsing efficiency at the cost of losing exactness, including beam search and inadmissible A* heuristics. It is theoretically interesting, however, to investigate all available means of improving efficiency while maintaining exact inference. In the remainder of this chapter, I introduce a more sophisticated PWG estimation procedure that, I argue, captures structure and ordering generalizations better than surface-tuple estimation and, in conjunction with A* prioritization, will lead to more efficient parsing.

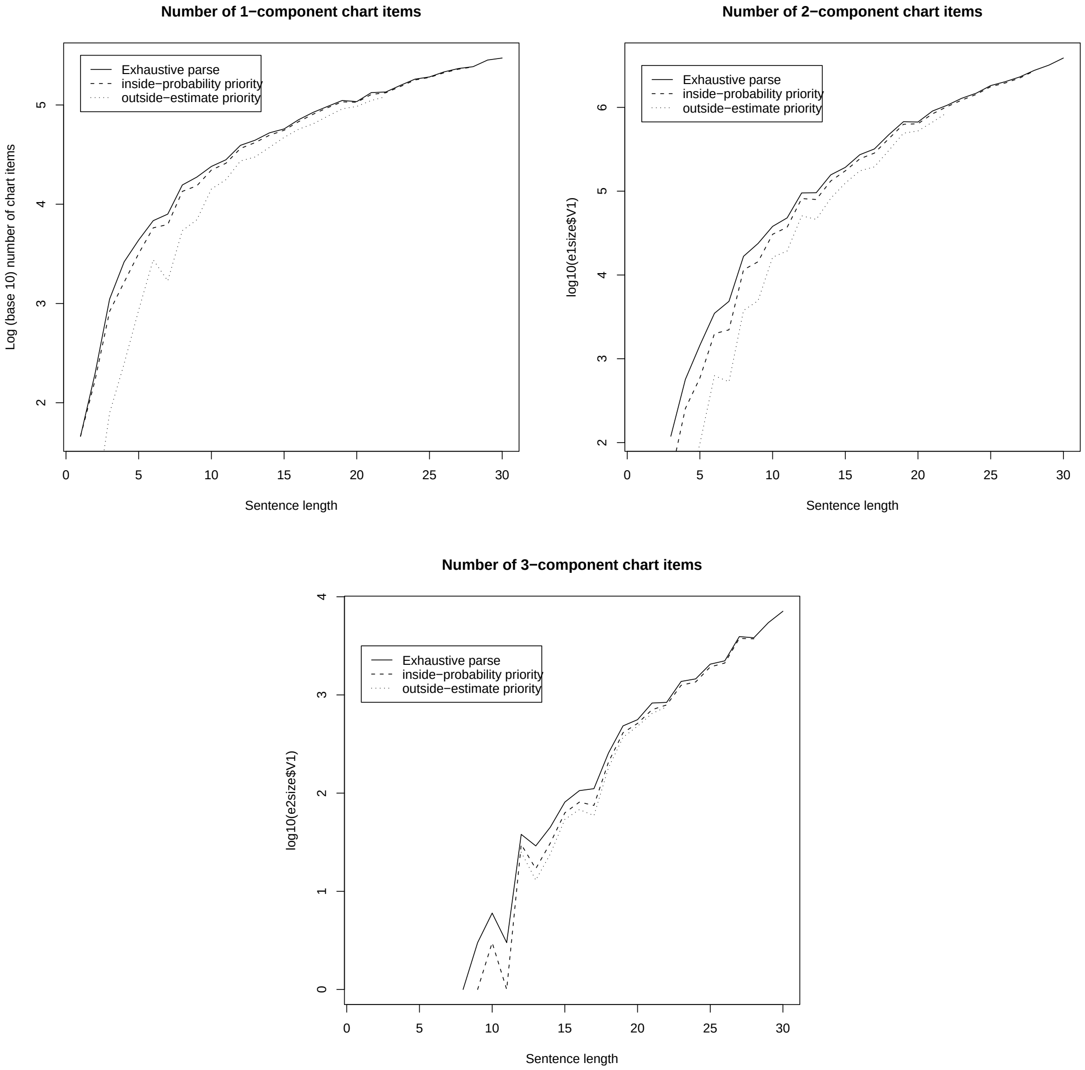


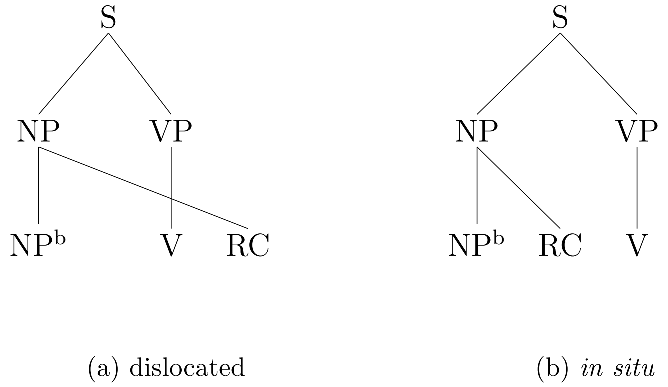
Figure 4.7: Growth in number of edges with varying A* priorities

4.8 ID/LP factorization of probabilistic wrapping grammars

The “surface tuple” approach to estimating PWGs from discontinuous constituency treebanks has the advantage that every discontinuous constituency tree defines a unique LCFRS derivation. But it has two drawbacks that become especially prominent in a probabilistic setting. First, there is no natural connection between similar constituencies in continuous vs. discontinuous rule variants. This can be seen clearly in Figure 4.8: in the extraposed case (Figure 4.8a), the top NP node is a two-component category, whereas in the *in situ* case (Figure 4.8b), the top NP node is a one-component category. This means that the *left-hand* sides of the rules are different in the two RC-producing NP rewrites, so the MLE rewrite probabilities of each derivation are determined independently.

The second drawback is that *distance* is intuitively an important consideration in discontinuous constituency models, because many kinds of discontinuity tend to happen only over short distances. Rightward extraposition of PPs and relative clauses is perhaps the clearest such case: rightward extraposition is more likely the longer the extraposable PP or RC, and less likely the more material intervenes between the modified NP and the end of its clause (see Uszkoreit et al. 1998 for a representative corpus study of German). But it is not clear where distance can figure into a surface-estimated PWG. In the surface-tuple model, the choice to extrapose or not to extrapose is made at the level of the S rewrite; but on the derivational generative view of context-free rewrite systems (Weir, 1988), the internal constituency of the NP and VP are not known at this point. Some modification to the model and parameter estimation process must be made to sensibly incorporate distance.

One intuitively attractive option for including distance sensitivity into PWGs would be to adopt a long-standing factorization of natural language grammars into constraints on *immediate dominance* (ID) and *linear precedence* (LP). The ID/LP approach to grammatical generalizations was perhaps most prominent in GPSG, where it was deployed primarily as a way of simplifying large numbers of context-free rules into a smaller number of basic generalizations; but it has also been an influential idea



(a) dislocated (b) *in situ*

Figure 4.8: Dislocated and *in situ* relative clauses

in the linearization approach to discontinuous constituency in more recent HPSG work (Reape 1994; Kathol 2000 among others). A simple probabilistic formulation of the ID/LP principle in PWGs would decompose an LCFRS rule into its *constituency* and its *shuffle*, and use the Markov rule to say that

$$P(X \to f[X_1 \cdots X_n]) = \overbrace{P(X \to X_1 \cdots X_n)}^{\text{dominance}}\overbrace{P(f|X \to X_1 \cdots X_n)}^{\text{precedence}} \tag{4.6}$$

The two halves of the Markov decomposition can then be individually estimated. The *dominance* half can be estimated by merging categories with the same label, regardless of the number of surface holes exhibited. The two NPs in Figure 4.8 would therefore be treated as the *same* two-component syntactic category. In 4.8a, the left component would consist of the string dominated by the NP$^{\mathrm{b}}$, the right of the string dominated by the relative clause. In 4.8b, the left component would consist of the concatenation of the NP$^{\mathrm{b}}$ and relative clause strings, and the right would consist of the empty string. This identification of syntactic category with surface node label (irregardless of number of surface holes) solves the first problem of generalizing constituency between rule variants with different degrees of surface discontinuity.

Distance can usefully be incorporated into the *precedence* half of the Markov decomposition by conditioning the likelihoods of different shuffles on the sizes of the individual pieces of Y and Z. At first glance, this may seem an ill-formed idea because on the derivational, generative approach the internal constituency of Y and

Z has not yet been determined. But there is a view of WG tree generation that saves distance in the Markov decomposition. Suppose that a tree is generated in two parts: beginning at the start symbol, the constituency—that is, the ID relations—are recursively generated top-down until every frontier node of the tree is a terminal. The probability of each ID relation is characterized by the dominance half of the Markov decomposition in Equation 4.6. After the constituency of the tree is complete, the LP relations are recursively generated bottom-up. At the point of determining the shuffle relation between each mother node X and the children of X, the internal constituencies and linear orderings of X's children are always completely known. Among other things, the size of each component of each of X's children is completely determined, and so it can be conditioned on. This derivational perspective also has a degree of plausibility as a model of natural language generation: the complete structural *contents* of a sentence are determined first, and only subsequently are these contents arranged in a *linear ordering.*

Although the ID/LP approach solves both the constituency-generalization and the distance-sensitivity problems for PWG formulation, it creates a new problem in parameter estimation: there is no longer a one-to-one correspondence between discontinuous constituency trees and wrapping grammar derivations.[9] This arises because the correspondence between the "surface tuple" structure of a node in a discontinuous constituency tree and its corresponding LCFRS category is no longer trivial. In Figure 4.8b, for example, the correspondence is underdetermined between the one surface component of the NP and the two components of the LCFRS NP category. (Contrast this with the "surface tuple" estimation procedure of Section 4.5.2, where this correspondence is guaranteed to be trivial at the cost of distinguishing between syntactic categories with the same label but different numbers of components.) The prospect remains, however, that useful ID/LP-factorized PWGs may still be learnable from discontinuous constituency treebanks through a combination of heuristics and unsupervised learning techniques.

[9] A similar problem is faced in the literature on probabilistic Tree-Adjoining Grammar estimation—see Chiang (2003) for an overview.

4.9 Conclusion

Although parsing with grammars allowing unrestricted discontinuous constituency is intractable, Linear Context-Free Rewrite Systems (LCFRSs) allow constituents with limited, lingustically plausible discontinuities while permitting parsing in polynomial time. In addition, the derivational independence of LCFRSs means that probabilistic LCFRS models (called probabilistic wrapping grammars, or PWGs for convenience here) learned in a completely supervised fashion using maximum-likelihood estimation are guaranteed to be proper. From this foundation, it is relatively easy to learn proper PWGs from available discontinuous constituency treebanks. The practical efficiency of these empirically-learned PWGs benefits from the fact that *most* observed natural language syntactic rules are actually continuous; and, in an A* exact parsing setting, from the fact that discontinuity-introducing syntactic rules are *likely* only in a limited set of circumstances.

Nevertheless, I have found that the time and memory required for bottom-up A* parsing of the simplest empirically-learned PWGs is dominated by the construction and processing of discontinuous constituents, simply because so many such constituents are possible. As a proposed remedy to this efficiency bottleneck, I have introduced a novel factorization of probability models over LCFRSs into *dominance* (ID) and *linear precedence* (LP) components, which in principle allows tighter probabilistic models over discontinuous constituency structures by scoring the *distance* between ID/LP components of a discontinuous constituent. However, completely supervised learning of such factorized PWGs is no longer possible with existing data sources. Future extensions of this work will involve empirical learning of ID/LP factorized grammars from treebanks using unsupervised learning techniques, and investigation of accuracy as well as efficiency in discontinuous constituency parsing.

Chapter 5

Conclusion

This dissertation has investigated the application of probabilistic disambiguation in models both of human syntactic comprehension and of naturally-occurring corpus data. With respect to human syntactic comprehension, I have argued for a simple theory of processing complexity in which the surprisal of an event determines its processing difficulty, which follows as a natural consequence of incremental, probabilistic disambiguation. With respect to refining models of corpus data, I have focused on problems of accuracy and efficiency in recovering nonlocal, or discontinuous, syntactic dependencies. From the work presented here we can draw several major conclusions that I enumerate below, and subsequently discuss.

1. Expectations from parallel, incremental disambiguation can explain syntactic "complexity"

2. Non-local dependencies are of particular interest in syntactic comprehension

3. Non-local dependencies can reliably be inferred serially, after context-free parsing, in English

4. Linguistically plausible discontinuous constituency trees can be parsed in polynomial time

5. Distance-sensitivity can be properly incorporated in generative probabilistic models over discontinuous constituency trees

5.1 Expectations and syntactic complexity

In the field of human syntactic comprehension, the effect of disambiguation on processing difficulty has primarily been thought of as relevant only in locally ambiguous sentences. There is ample evidence, however, that human syntactic processing involves constant, incremental disambiguation. In Chapter 2, I show that from a simple set of assumptions regarding incremental disambiguation it is possible to derive a highly intuitive model of processing complexity, where the difficulty of a word is inversely related to the word's expectation, or more formally its probability, in its context. Because difficulty in this model falls out of the redistribution of probability mass among possible completions of the sentence, it can be thought of as a *resource allocation* difficulty model. This resource-allocation model subsumes a prominent line of work in ambiguity resolution, which that argues that structural ambiguity is resolved in proportion to the frequency of the structures involved. In addition, the model makes a number of non-trivial predictions regarding processing difficulty in *unambiguous* sentences. In most of the cases where its predictions diverge from the predictions of more traditional resource-limitation models, available experimental evidence favors the expectation-based, resource-allocation model.

5.2 Nonlocal dependencies in syntactic comprehension

Nonlocal syntactic dependencies turn out to be particularly interesting both for the practical problem of syntactic disambiguation and for the theory of processing difficulty. As I discussed in Chapter 1, there is a close relationship between nonlocal dependency, context-sensitivity, crossing dependency, and discontinuous constituency. Most work in probabilistic parsing has focused on local, context-free syntactic relationships, rather than on identification and disambiguation of nonlocal relationships. Nevertheless, I showed in Chapter 3 that there is a considerable amount of nonlocal syntactic dependency (up to nearly 5%) even in written text of English, a language/genre pair with relatively little discontinuity; and considerably more in

German text. Major sources of nonlocal dependency, such as *wh*-question formation and relativization, are of course especially prominent in interactive utterances that are important in an interactive setting.

In the theory of processing difficulty, nonlocal depedendency is especially interesting because the one experimental result to my knowledge that clearly favors resource-limitation over resource-allocation models, that of Grodner and Gibson (2005), involves storage and retrieval of a long-distance dependency induced by relativization. As I suggested in Section 2.7.1, one possible way to reconcile the two processing theories might be to posit that resource limitation effects are substantial in syntactic processing *only* for true nonlocal syntactic dependencies. The precise formulation and testing of such a mixed theory awaits future research.

5.3 Parsing local and nonlocal dependencies serially

Chapter 3 presented an algorithm for determining the complete syntactic dependency structure of a sentence (i.e., both local and non-local dependencies) through a serial process, first identifying the most probable context-free parse of the sentence in a PCFG representing only local dependencies, and then identifying and disambiguating non-local dependencies between nodes of the context-free tree using cascaded maximum-entropy classifiers. Applying this algorithm to gold-standard English newswire input reduced error over purely context-free dependencies by over 80%. Furthermore, most of this improvement in dependency accuracy was maintained when the algorithm was composed with a state-of-the-art PCFG parser. These facts indicate that state-of-the-art context-free parsing is good enough that a serial approach to identifying the complete syntactic structure of English newswire sentences is viable. The situation with respect to German newswire text, in contrast, is much less clear, partly because German context-free parsing is much more error-prone than English, and partly because efforts to identify non-local dependencies in German were less successful overall.

5.4 Tractable discontinuous constituency parsing

Although this finding is not new, it is worth reiterating in the present context. Although grammars that allow completely unrestricted discontinuity do not admit polynomial-time recognition via traditional parsing algorithms, the Linear Context-Free Rewrite Systems (LCFRSs) allow limited degrees of discontinuity, for which polynomial-time recognition is possible. The expressivity (and correspondingly worst-case parsing complexity) of an LCFRSs can be tailored to fit the degree of discontinuity observed in a sample of natural language syntax, which facilitates the learning of LCFRSs from syntactically-annotated natural language data. Chapter 4 shows that most rules in LCFRS grammars learned from syntactically-annotated corpora involve no discontinuity, and very few rules involve more two discontinuities. Finally, the *local set* or *derivational independence* property of the LCFRSs means that proper probability distributions over LCFRS derivation trees can be learned relatively easily.

5.5 Distance sensitivity for discontinuous constituency

Empirically, many discontinuity-inducing constructions (perhaps most notably rightward extraposition of nominal postmodifiers) are optional and highly *distance-sensitive*: they are used only when the linear dependent-governor distance they create is not unsuitably large. It is clearly desirable to incorporate this type of distance sensitivity into probabilistic models over discontinuous constituency trees, but due to their derivational independence property it is not immediately obvious how probabilistic LCFRSs can condition on such information. In Chapter 4, I show how distance sensitivity can be built into a probabilistic LCFRS by introducing a probabilistic *immediate dominance/linear precedence* ID/LP factorization. In addition to introducing distance-sensitivity, this factorization has the added benefit of permitting natural cross-cutting probabilistic generalizations about structural constituency across continuous and discontinuous instantiations of a single syntactic category. Although ID/LP-factorized probabilistic LCFRSs can no longer be learned in a completely supervised manner from existing data sources, I outline an algorithm for learning factorized LCFRSs from corpora consisting of discontinuous constituency trees, and provide a supporting theoretical analysis.

Appendix A

Proof of equivalence of headed context-free trees and non-crossing dependency trees

Definition. A *headed context-free* (HCF)-tree is a context-free tree such that (i) terminal nodes have no sisters; and (ii) for each non-terminal node in the tree, exactly one daughter is distinguished as the *head daughter*. The *head* of a HCF-tree is the unique terminal node to which a path can be traced from the root of the tree through an uninterrupted sequence of head daughters.

Definition. A *dependency* (D)-tree on $1, \cdots, n$ is a set S of ordered pairs $\langle i, j\rangle$ such that for one number h between 1 and n, the *root*, there is no ordered pair in S beginning with h, and for all other numbers k between 1 and n there exactly one ordered pair in S beginning with k. An ordered pair $\langle i, j\rangle$ is called a *dependency* of j on i. A pair of dependencies $d_1 = \langle i, j\rangle$ and $d_2 = \langle i', j'\rangle$ is *crossing* iff, for x and x' the smaller numbers and y and y' the greater numbers in d_1 and d_2 respectively, $x < x' < y < y'$ or $x' < x < y' < y$. A dependency $\langle i, j\rangle$ such that h is between i and j is said to *cross the root*. A D-tree that has a pair of crossing dependencies or has a dependency that crosses the root is said to be a *crossing* D-tree.

A HCF-tree t *induces* a D-tree as follows: each non-unary node N in the HCF-tree has a head daughter H headed by h and non-head daughters $\mathrm{D}_{1,\cdots,n}$ headed by $d_{1,\cdots,n}$.

For each non-head daughter D$_i$ of N, the dependency $\langle h, d_i \rangle$ is in the D-tree induced by t.

Lemma 1. In a HCF-tree headed at h, no induced dependency $\langle i, j \rangle$ can exist such that $i < h < j$.

Proof: by induction on tree depth. Base case: in tree of depth 1, all induced dependencies are of the form $\langle i, h \rangle$. Therefore the lemma holds.

Induction: suppose that the lemma holds for trees of depth equal to or less than m. Now consider a tree T of depth $m + 1$. The top node has one head daughter H, and n non-head daughters $D_{1,\cdots,n}$. Now consider an arbitrary dependency $\langle i, j \rangle$ in the induced D-tree. i may be inside H or inside some D_k. If it is D_k, then it is either the head of D_k or it is not. If it is, then $j = h$ and thus $\langle i, j \rangle$ does not cross h. If it is not, then it is dependent on something else in D_k, and therefore $\langle i, j \rangle$ cannot cross h. If i is inside H, then it cannot be the head, since the head of H is the head of T and it is dependent on nothing. So $\langle i, j \rangle$ is induced by H. But since H is of depth equal to or less than m, we have that $\langle i, j \rangle$ does not cross h by the inductive assumption. Therefore the lemma holds for $m + 1$.

Theorem. No HCF-tree induces a crossing D-tree, and for every non-crossing D-tree, there is an HCF-tree that induces it.

Proof. Constructing an HCF-tree that induces a given D-tree is not difficult. We make use of an intermediate representation form involving ordered constituent pairs. The tree construction process goes as follows. First, for each $i \in 1, \cdots, n$ create a constituent C_i dominating only i. Then create a set of constituent pairs by replacing each i in every dependency pair with the corresponding C_i.

Next, for each constituent that nothing depends on (that is, each constituent that never appears in the second position of a constituent pair), find the constituent H that it depends on. Then, for each such constituent H, do the following:

1. If there are constituents $\{L_i\}$ to the left of H that depends on H, create a new constituent M headed by H with sister L_i nearest to H.

2. Otherwise there must be constituents $\{R_i\}$ to the right of H that depends on H. Create a new constituent M headed by H with sister R_i nearest to H.

Note that the constituent M resulting from the combination of H with its left or right dependent C must be continuous. If it were not, then there would be some word w in between H and C that depends neither on C nor on H. If w were dependent on nothing, it would be the root, and the dependency between the heads of H and C would cross the root. If w were not the root, then it would be dependent on something outside of H or C and the dependency between the heads of H and C would cross with the dependency of w.

To see that the resulting HCF-tree induces the original D-tree, note that there will be $n-1$ binary nodes in the HCF-tree, and no nodes with More than two sisters. Each binary node N induces a dependency between the head H and its sister C. But this means that the original D-tree must contain the dependency of the head c of C on the head h of H, or N would not have been constructed. Therefore, the HCF-tree induces all and only the dependencies in the original D-tree.

Now we must prove that no HCF-tree induces a D-tree with crossing dependencies. We prove this by contradiction. Suppose that an HCF-tree T induced a D-tree with crossing dependencies. This would mean either that there are two dependencies $\langle i, j\rangle$ and $\langle i', j'\rangle$ such that, if x and x' are the lesser and y and y' are the greater of $\langle i, j\rangle$ and $\langle i', j'\rangle$, then $x < x' < y < y'$; or that a dependency $\langle i, j\rangle$ crosses the head. The latter can be immediately ruled out as it is a violation of Lemma 1.

If the former is true, then there must be constituents X, X', Y, Y' with respective heads x, x', y, y' such that X and Y are in a head-sister relation and X' and Y' are as well. Call M the mother of X and Y and M' the mother of X' and Y'. Since the spans of M and M' overlap, either M is in M' or vice versa. Suppose that M' is in M. (The reverse case, if M is in M', also holds by symmetry.) The span of M' also overlaps with the span of Y, so either M' is in Y or Y is in M'. If M' is in Y, then we have a violation of Lemma 1, since Y is headed by y but there is a dependency across y induced by Y. If Y is in M', then M' cannot be in M as Y is a daughter of M. Therefore there can be no crossing dependencies induced by T.

Appendix B

Specialized tree-matching pattern features

This section contains a list of the specialized pattern templates used in the various classifiers of this chapter. Pattern syntax and semantics is that of `tregex`, a tree-matching software package written by myself and Galen Andrew. It can be downloaded at *http://nlp.stanford.edu/software/tregex.shtml*.

B.1 Empty Complementizer Insertion

B.1.1 Quoted S under SBAR

```
S > /^S/ $-- `` $++ ''
```

A quoted S node under an SBAR node is likely to be a direct in-situ (i.e., immediately following a speech verb) quotation. When direct in-situ quotes have no overt complementizer *that* introducing them, the Treebank annotates an empty complementizer instead.

> Slavery was prohibited in Massachusetts by the terms of the constitution of 1789, which declared ***0*** "all men are born free and equal".

B.1.2 Presence of SBAR daughter

```
__ < SBAR
```

Nothing with an SBAR daughter should have an empty complementizer.

B.1.3 VP right sister of S

```
VP > /^S/ $-- S
```

When a speech fragment (quoted or indirect) is topicalized, the VP is usually annotated as having an empty complementizer and a trace of topicalization:

> “ [Wage increases and overall compensation increases are beginning to curl upward a little bit,]$_i$ ” said ***0*** t_i Audrey Freedman, a labor economist at the Conference Board, a business research organization.

When deciding whether an apparently intransitive VP should be the origin site of quoted speech, it is useful to see whether there is a plausible topicalized speech fragment.

B.1.4 Matrix VP

```
VP > (S > ROOT)
```

Because topicalized speech fragments are a major source of empty complementizers, and topicalized speech is usually in the matrix clause, it is useful to distinguish between matrix and non-matrix VPs.

B.1.5 VP descendent of PRN

```
VP > PRN | > (__ > PRN)
```

VPs inside parenthetical expressions are more likely to have extraposed direct or indirect speech:

> [Eventually, he believes **0** t_i , investors will be willing to pay higher prices for companies with proven track records of earnings growth.]$_i$

B.1.6 Syntactic paths from S

A special feature template is activated for VP nodes, and is instantated three times for every S node in the tree. In each instantiation, the syntactic path (as defined in Section 3.5) from that S node is recorded. The head word and head tag of the VP node are also included in the second and third instantiation respectively.

B.1.7 VP with sister punctuation

```
VP $ /^(\.|\,|’’|‘‘)$/
```

Verb phrases next to punctuation are more likely to have topicalized speech fragments extracted from them, and hence more likely to have an empty complementizer. The template instantiation records the type of punctuation (period, comma, or quote).

B.2 Dislocated node identification – English Treebank

B.2.1 Quoted S

```
S > /^S/ $-- ‘‘  $++ ’’
```

This pattern picks out quoted S nodes, which are often dislocated from their canonical position:

> “[There is a large market out there hungry for hybrid seeds,]$_i$” he said t_i .

B.2.2 Parenthetical S

```
S < (PRN < (/^S/S=s))
```

Sometimes a single quotation is wrapped around the clause that introduces it. The Treebank annotates the introducing clause as a parenthetical expression (PRN), and uses a cyclical annotation structure for the dislocation:

> [Nevertheless, he said t_i , he is negotiating with Plant Genetic to acquire the technology to try breeding hybrid cotton.]$_i$

This template is instantiated twice: in one instantiation, the head word of the `=s` constituent is recorded; in the other, the head word, tag, and label of the `=s` constituent are all recorded.

B.2.3 Nodes in S or SINV preceding VP

```
__ > /^S/=s $++ VP=vp
```

Nodes of certain categories appearing in an S or SINV are likely to be dislocated from the VP of that clause. This includes locative inversion:

> [Even more at fault]$_i$ are t_i those leaders in and out of government who urged and supported their defections, thereby giving great help and comfort to the enemy propagandists.

and topicalized sentential complements (especially indirect speech):

> [The SEC will probably vote on the proposal early next year]$_i$, he said t_i.

This feature template is instantiated twice. Both instantiations include the syntactic categories of X and its parent. The second instantiation also includes the number of words dominated by the VP (the relevant VPs are typically small in cases of dislocation).

B.2.4 Expletive subjects

```
SBAR > (VP < VBN|VBD > (VP > (S < (NP < (PRP < it))) < (/^(VB|AUX)/
    < was|is|'s|WAS|be)))
/^S/=s > (VP >+/^S/ (/^S/ < /^NP/=np))
```

Right extrapositions that leave behind expletive subjects are annotated as dislocations:

> It t_i was enjoyable [to hear accomplished jazz without having to sit in a smoke-filled club]$_i$ . . .

The first pattern is more narrow, matching only those cases when the matrix verb is the copula, which is a strong indicator of expletive subjecthood. The second pattern was used only when the `=np` node was headed by *there* or *it* (ignoring capitalization), and was instantiated twice—once as a basic feature, the second time also recording the head word of the clause.

B.2.5 Final VP daughters separated from an NP

```
__ >- VP $-- (__=sep $-- NP)
```

Most right extrapositions from within VP are directly out of NPs:

> UNESCO is now holding its biennial meetings t_i in Paris [to devise its next projects]$_i$.

This means that jumping over an intervening constituent is a necessary condition for extraposition. The pattern above looks for VP-final phrases that are separated from an NP constituent; for every intervening constituent, the pattern is instantiated, recording the final constituent's syntactic category as well as the category of the intervening `=sep` constituent (PPs and ADVPs are intuitively more likely to be jumped over than NPs or Ss, for example). Note that finding these right-extraposed constituents at all is hard because they are rare.

B.2.6 Constituents following an initial coordinator

```
__ $- (CC=cc >, S)
```

Occasionally, there are adverbial-type constituents that occur immediately after an initial coordinator:

And [so]$_i$ it went t_i through the first half ...

The presence of an initial coordinator seemed to be a signal for this behavior. This feature template also recorded the specific head word of the coordinator.

B.2.7 Extra context for WH phrases

A WH-phrase (including the parent category of an empty complementizer) gets additional context feature templates instantiated, including:

- category × grandparent-category conjunction
- category × grandparent-head-word conjunction
- category × grandparent-head-tag conjunction

B.2.8 Daughter of SINV to the left of VP

Any category to the left of the main VP (defined as the leftmost VP whose head tag is VBZ, MD, VBD, or VBP) gets this feature.

B.2.9 ..., *he said* construction

```
S > S $++ (VP <, said|says|noted|notes)
```

This pattern is instantiated only if the VP in the pattern is the main verb (as defined in Section B.2.8 of its parent.

B.2.10 Paths from S to VP

Every S node has the syntactic paths to each VP recorded as a feature. Also, if a path contains an ↑-PRN element, then the head terminal of the VP is also recorded.

B.3 Dislocated node identification – NEGRA

B.3.1 Final S daughter of VP or S

```
S >- VP|S=parent <, __=dtr
S >- VP|S=parent <, (__=dtr << PRELS)
```

Final S daughters of VP or S are likely to be right extraposed relative clauses. The PRELS tag is a sign of this. Both of these templates are instantiated including the categories of the `=parent` and `=dtr` nodes.

> Mehr Sicherheit sollen überwachungsgeräte t_i bringen, [mit denen
> More safety should monitoring_equipment ↑ bring, [with which
> zu Hause Puls und Atemfrequenz kontrolliert werden können]$_i$.
> at home pulse and breathing_rate controlled be can].
>
> “Greater safety should arrive with monitoring equipment via which pulse and breathing rate can be controlled at home.”

B.3.2 Postposition from CS

```
S > S $-- (CS=cs)
```

Speech fragments wrapped around a speech-introducing verb are typically annotated in NEGRA as coordinated sentences with one extraposed element:[1]

[1] Contrast this annotation with the cyclical dislocation annotation in the WSJ, noted in Section B.2.2.

> “Wir identifizieren uns mit unserem Programm t_i ,” sagt Silvia
> “We identify ourselves with our program ,” said Silvia
> Stenger, “[arbeiten alle gemeinsam am gleichen Produkt]$_i$.”
> Stenger, “work all together on same product.”
>
> ‘“We identify ourselves with our program,” said Silvia Stenger, “everyone works together on the same product.”’

One sign of this is that if an S node has a CS daughter with only one S daughter. The number of S daughters of the `=cs` node is recorded with this template.

B.3.3 Path to pronominal PROAV

```
S|VP=tree >- (__=parent <+VP (PP < PROAV))
```

This pattern indicates a plausible path to a pronomial adverb PROAV:

> Und sicherlich ist es nicht falsch darauf t_i hinzuweisen, [daß sie
> And surely is it not incorrect at_it to_point_out, that they
> ja nun noch kein Wahlrecht haben]$_i$.
> well now still no suffrage have.
>
> “And surely it isn’t incorrect to point out that they are still disenfranchised.”

The syntactic category of `=tree` is recorded with this template.

B.3.4 Conjunct in a non-coordinated category

```
PP=t $- KON > S|VP=parent
```

The sequence KON PP should only occur in a coordinated PP element. But these sometimes appear elsewhere:[2]

[2] Note that examples such as this would probably be more conventionally analyzed as VP gapping, rather than conjunct extraposition.

Die Zivilisation eines Staates läßt sich $[_{\mathrm{CPP}}$ nicht allein an seinem
The civilization of_a state lets self not alone by self's
Bruttosozialprodukt t_i t_j] ablesen, [sondern]$_i$ [zum Beispiel auch
gross_social_product read_off, rather for example also
daran, welche Bedingungen er bietet, ein qualifiziertes,
thereby, which conditions he provides, a qualified,
verantwortbares Studium zu absolvieren]$_j$.
responsible study to complete.

"The civilization of a state should not be measured solely by its gross social production, but, for example, also by the conditions that it provides to complete a qualified, responsible course of study."

The parent category is also included in the instantiation.

B.3.5 Final S and VP constituents binned by size

```
S|VP >- S|VP=parent
```

Right extraposition is more likely for larger constituents. Therefore, two size features are recorded for S or VP constituents final in an S or VP: the size of the constituent in words, divided by four and rounded down; and the *proportion* of the size of its parent that the constituent makes up, multiplied by five and rounded down.[3]

B.3.6 Head, First, Last Daughter

```
__ ># __
__ >- __
__ >, __
```

A head daughter should definitely not be dislocated.[4] First and last daughters are the best positions to find dislocated constituents.

[3] Different bin sizes are treated as distinct indicator features, rather than treating size as a single real-valued feature (though the latter approach would also be plausible).

[4] Partial VP fronting (de Kuthy and Meurers 2001 and others) would in principle be a possible exception, but it is exceedingly rare.

B.3.7 Categories in between node and head sister

```
__ $++ (__=between $++ (__ ># __=head))
__ $-- (__=between $-- (__ ># __=head))
```

This is a general-purpose template designed to capture generalizations about what extraposed constituents like to extrapose across. The categories of the target constituent, the `=between` constituent, and the `=head` constituent are all recorded.

B.3.8 Expletive Antecedent

```
__ > S|VP $ (NP < (PPER < Es))
```

Since many right-extrapositions leave expletive antecedents, it is useful to look for a plausible expletive antecedent to judge whether a constituent is right-extraposed:

Es	t_i	spielt	eben	keine	Rolle,	[ob	die	Musik	gefällig	ist]$_i$	...
It		plays	precisely	no	role,	whether	the	music	pleasing	is	...

"Whether the music is pleasing plays no role ..."

B.3.9 Precedes last preverbal argument

```
__ $+ (__ $++ VP|CVP) !$-- VP|CVP
```

When a German clause is scrambled, the context-free NEGRA transformation puts the arguments of the main verb preceding the subject in S, and the subject typically becomes the last daughter of S preceding the main VP. Any argument before this likely subject has a good chance of being a dislocated (due to fronting) constituent.

[Im	April	nächsten	Jahres]$_i$	soll	die	Bevölkerung	t_i	über	ihre
In	April	next	year	shall	the	populace		over	their

Unabhängigkeit	von	Äthiopien	abstimmen...
independence	from	Ethiopia	vote...

"Next April the populace shall vote on independence from Ethiopia..."

B.3.10 Daughters of unembedded S nodes

```
__ > (S !<- VAFIN|VAIMP|VVFIN|VVIMP|VMFIN | < (__ $--
    VAFIN|VAIMP|VVFIN|VVIMP|VMFIN)
```

This pattern picks out nodes that are inside an unembedded (i.e., matrix or matrix-like) S node. It is instantiated once for each node that matches it. In addition, nodes matching this pattern that are the leftmost daughter of their parent (the most likely place for a dislocated constituent) have features with the following information recorded:

1. Their syntactic category, plus whether or not their parent has a VP;
2. Their syntactic category, plus the head word of their parent;
3. The conjunction of (1) and (2).

B.4 Control locus identification

B.4.1 Passive verbs

```
VP < VBN|VBD > (VP|SQ < (/^(VB|AUX)/ < be|was|is|are|were|been|
    being|’s|’re|’m|am|Been|Being|WAS|IS|get|got|getting|gets|
    Get|gotten|become|became|felt|feels|feel|seems|seem|seemed|
    remains|remained|remain))
VP < VBN|VBD > (VP < CC > (VP|SQ < (/^(VB|AUX)/ < be|was|is|are|
    were|been|being|’s|’re|’m|am|Been|Being|WAS|IS|get|got|getting|
    gets|Get|gotten|become|became|felt|feels|feel|seems|seem|seemed|
    remains|remained|remain)))
```

Passive verbs are annotated as having control loci, controlled by the subject, as sisters:

[A record date]$_i$ hasn't been set $*_i$.

The second pattern is for coordinated VPs with a single auxiliary.

B.4.2 Participials in selected small-clause context

```
VP < VBN|VBD > (S > (VP < (/^(VB|AUX)/ < be|was|is|are|were|been|being|
    ’s|’re|’m|am|Been|Being|WAS|IS|get|got|getting|gets|Get|gotten|
    become|became|felt|feels|feel|seems|seem|seemed|remains|remained|
    remain)))
VP < VBN|VBD > (S > (S > (VP < (/^(VB|AUX)/ < be|was|is|are|were|been|
    being|’s|’re|’m|am|Been|Being|WAS|IS|get|got|getting|gets|Get|gotten|
    become|became|felt|feels|feel|seems|seem|seemed|remains|remained|
    remain))))
```

Participials can receive a small-clause interpretation under some verbs. Under these circumstances they are annotated with a control locus in the VP, controlled by the small-clause subject:

> …if we can get [that Warsaw Pact superiority]$_i$ brought $*_i$ down to parity…

The second pattern is for coordinated VPs.

B.4.3 Participial verbs

```
VP < VBN|VBD > (NP < NP)
VP < VBN|VBD > (VP < CC > (NP < NP))
```

Participial verbs inside NPs have (uncontrolled) control loci:

> rising consumer prices reported * last week

The second pattern is for coordinated VPs.

B.4.4 Granddaughters via VP

```
S=s < VP=vp
```

When this pattern is matched, the feature is instantiated once for every daughter node of the `=vp` node, recording the syntactic category of that daughter. This is because information about the VP (notably whether it contains a TO, or a VBG gerund) is a strong indicator of whether there should be a controlled subject.

B.4.5 Unary-rewrite PP sister of VBN

```
PP <: __ $ VBN
```

When a verb-plus-preposition combination is passivized, a control locus annotation expresses the passivization:

> We have as much nostalgia as anyone for those leafy, breezy days in Washington when honorable men and women dickered over budgets and even log-rolled a bit to see that the bridges got build [*sic*], roads paved, soldiers paid or that [the desperately poor]$_i$ were cared for $*_i$.

The best indicator for this is a unary-rewrite PP next to a participial verb.

B.4.6 Node size

The length of the node’s yield (in number of words) is recorded.

B.4.7 Perfect auxiliary

```
  VP > (VP < (/^(VB|AUX)/ < has|have|had|’ve|having|’d|HAS))
```

The *have* auxiliary is a strong negative indicator that its VP complement is not passive.

B.4.8 Complement of verb that selects bare-verb VPs

```
VP < VB|NN > (S > (VP < (/^(VB|AUX)/ < help|helping|helped|helps|had|
    have|has|having|let|letting|lets|see|sees|saw|seeing|seen|make|
    makes|making|made|hear|hears|hearing|heard|watch|watches|watched|
    watching)))
```

The above verb forms select for bare-verb VPs, which have controlled subjects.

B.4.9 S or VP without non-temporal NP

```
S|SQ !< (NP=np $++ VP)
VP !< (NP=np $-- /^(V.*|MD|AUX.*)$/)
```

These are a more fine-grained attempt at filtering out temporal NPs, the presence of which should not be interpreted as the presence of an argument NP. These patterns are only considered matched when the `=np` constituent is *temporal*, defined as when its head matches the regular expression

```
Monday|Tuesday|Wednesday|Thursday|Friday|Saturday|Sunday|years?|
months?|weeks?|days?|mornings?|evenings?|January|February|March|
April|May|June|July|August|September|October|November|December|
[Tt]oday|[Yy]esterday|[Tt]omorrow|[Ss]pring|[Ss]ummer|[Ff]all|
[Aa]utumn|[Ww]inter
```

B.4.10 Subjectless TO or gerund VP

```
S !< NP < (VP <<# TO|VBG|AUX)
```

This pattern only works when the VP in the pattern is the main verb of the S, as defined in Section B.2.8. The motivation for this pattern is similar to that of the pattern in Section B.4.4.

B.5 Finding Controllers

B.5.1 Special features for uncontrolled loci

Since a control locus does not need a controller, a special classification choice, the `NULL` controller, is introduced. For this choice of controller, the following templates are introduced (the *governing node* is defined as the lowest node dominating the control locus that is not headed by the control locus):

1. The head tag of the governing node;
2. (1) conjoined with the category of the governing node's parent;
3. (2) conjoined with the governing node's parent's head tag;
4. (2) conjoined with the governing node's parent's head word.

B.5.2 Plausible Object Controller

```
__ > (NP > (S <  VP=vpSis >+/^PP/ (VP=vpAbove <+/^PP/ NP=controller)))
```

If this pattern matches on the control locus empty, then the `=controller` node gets two feature instantiations: one that includes the governing verb (the head of `=vpAbove`), and one that includes both the governing verb and its part of speech.

> …Sen. Strom Thurmond recently urged [fellow lawmakers]$_i$ $*_i$ to revive
> a broad federal death penalty. ⊔

This template is instantiated twice. Both times including the head tag of the `=vpSis` node; one of those times, the head word of the `=vpAbove` node is also included.

B.5.3 Subject controller in presence of object controller

```
__ > (NP > (S < VP=vpSis >+/^PP/ (VP=vpAbove <+/^PP/
    NP=objectControllerCandidate >+/^VP/ (S <
    NP=subjectControllerCandidate))))
```

If this pattern matches on the control locus empty, then the node matching the `=subjectControllerCandidate` pattern element gets two feature instantiations: one that includes the governing verb (the head of `=vpAbove`), and one that includes both the governing verb and its part of speech. This pattern is necessary in the independent binary training scheme, because the subject controller position is so highly favored overall. This template is instantiated twice, recording the exact same extra information as for the template in Section B.5.2.

Appendix C

NEGRA punctuation for discontinuous trees

As came up in Chapter 2, the placement of punctuation in the NEGRA corpus was not linguistically motivated. In the original crossing-dependency representation of the corpus, all punctuation attaches directly to the root node of the tree; in the context-free version of the corpus, it is attached as high as possible.[1] Both of these strategies, however, can lead to spurious discontinuities—defined as the presence of holes containing only punctuation—in the discontinuous-constituency representation. Such a hole can never be linguistically motivated, however, because punctuation marks prosodic bounds in the linear string. To minimize this discontinuity, I use the algorithm shown in Figure C.1 to reposition punctation in the discontinuous-constituency tree. Intuitively, this algorithm attaches every punctuation node P to the lowest node in the tree that covers both the first non-punctuation terminal to the left of P and the first non-punctuation terminal to the right of P. Punctuation nodes at the left and right edges of the sentence are both attached to the root node.

[1] Although maximally high attachment can obfuscate distributional generalizations, such as the fact that non-restrictive relative clauses in English occur with a following comma, it actually has a precedent in the statistical parsing work of Collins (1999). The discussion in Chapter 4 suggests a reason why this might be advantageous: bringing punctuation out of its distributionally motivated position effectively weakens probabilistic independence between the internal structure of the node where the punctuation originated and the external structure in which it is situated.

```
for each punctuation node $P$ in the tree do
    $i \leftarrow$ string position of $P$
    $l \leftarrow$ i
    while the terminal in the $l - 1^{\text{th}}$ string position is punctuation do
        $l := l - 1$
    end while
    $r \leftarrow$ i
    while the terminal in the $r + 1^{\text{th}}$ string position is punctuation do
        $r := r - 1$
    end while
    if $l = 1$ or $r = n$ then
        re-attach $P$ as child of the root node
    else
        let $N$ be the terminal in the $l - 1^{th}$ string position
        while $r + 1$ is not covered by $N$ do
            $N := PARENT(N)$
        end while
        re-attach $P$ as child of $N$
    end if
end for
```

Figure C.1: Punctuation-repositioning algorithm for discontinuous-constituency trees

Bibliography

Abney, S. (1997). Stochastic attribute-value grammars. *Computational Linguistics*, 23(4):597–618.

Agresti, A. (2002). *Categorical Data Analysis.* Wiley.

Altmann, G. T. and Kamide, Y. (1999). Incremental interpretation at verbs: restricting the domain of subsequent reference. *Cognition*, 73(3):247–264.

Anderson, J. R. (1990). *The Adaptive Character of Human Thought.* Lawrence Erlbaum.

Berger, A. and Miller, R. (1998). Just-in-time language modeling. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 23.

Berger, A. L., Pietra, S. A. D., and Pietra, V. J. D. (1996). A Maximum Entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Bies, A., Ferguson, M., Katz, K., and MacIntyre, R. (1995). *Bracketing Guidelines for Treebank II Style.* Penn Treebank Project.

Bod, R. (2003). An efficient implementation of a new DOP model. In *Proceedings of EACL.*

Bunt, H. (1996). Formal tools for describing and processing discontinuous constituency. In Bunt, H. and van Horck, A., editors, *Discontinuous Constituency*, number 6 in Natural Language Processing, pages 63–84. Mouton de Gruyer.

Büring, D. and Hartmann, K. (1996). All right! In Lutz, U. and Pafel, J., editors, *On Extraction and Extraposition in German*, pages 179–211. Amsterdam: John Benjamins.

Cahill, A., Burke, M., O'Donovan, R., Genabith, J. V., and Way, A. (2004). Long-distance dependency resolution in automatically acquired wide-coverage PCFG-based LFG approximations. In *Proceedings of ACL*.

Campbell, R. (2004). Using linguistic principles to recover empty categories. In *Proceedings of ACL*, pages 646–653.

Caraballo, S. A. and Charniak, E. (1998). New figures of merit for best-first probabilistic chart parsing. *Computational Linguistics*, 24(2):275–298.

Carreiras, M., Garcia-Albea, J. E., and Sabastian-Galles, N., editors (1996). *Language Processing in Spanish*. Hillsdale, NJ: Erlbaum.

Charniak, E. (1997). Statistical parsing with a context-free grammar and word statistics. In *Proceedings of AAAI*, pages 598–603.

Charniak, E. (2000). A Maximum-Entropy-inspired parser. In *Proceedings of NAACL*.

Charniak, E. (2001). Immediate-head parsing for language models. In *Proceedings of ACL*.

Charniak, E. and Johnson, M. (2005). Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of ACL*.

Chelba, C. and Jelinek, F. (1998). Exploiting syntactic structure for language modeling. In *Proceedings of ACL/COLING*.

Chen, S. and Rosenfeld, R. (2000). A survey of smoothing techniques for ME models. *IEEE Transactions on Speech and Audio Processing*, 8(1):37–150. Available online as A Gaussian Prior for Smoothing Maximum Entropy Models, Technical Report CMU-CS-99-108, School of Computer Science, Carnegie Mellon University, February 1999, *http://reports-archive.adm.cs.cmu.edu/anon/1999/CMU-CS-99-108.pdf*.

Chi, Z. and Geman, S. (1998). Estimation of probabilistic context-free grammars. *Computational Linguistics*, 24(2):299–305.

Chiang, D. (2003). Statistical parsing with an automatically-extracted tree adjoining grammar. In Bod, R., Scha, R., and Sima'an, K., editors, *Data-Oriented Parsing*. CSLI.

Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on Information Theory*, 2(3):113–124.

Chomsky, N. (1981). *Lecture on Government and Binding*. Mouton de Gruyer.

Clifton, C. and Frazier, L. (1989). Comprehending sentences with long distance dependencies. In Carlson, G. and Tanenhaus, M., editors, *Linguistic Structure in Language Processing*, pages 273–317. Dordrecht: Kluwer.

Collins, C., Carpenter, B., and Penn, G. (2004). Head-driven parsing for word lattices. In *Proceedings of ACL*.

Collins, M. (1999). *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania.

Collins, M. (2000). Discriminative reranking for natural language parsing. In *Proceedings of ICML*.

Collins, M. and Duffy, N. (2001). Convolution kernels for natural language. In *Proceedings of NIPS*.

Crocker, M. and Brants, T. (2000). Wide-coverage probabilistic sentence processing. *Journal of Psycholinguistic Research*, 29(6):647–669.

Cuetos, F. and Mitchell, D. C. (1988). Cross-linguistic differences in parsing: Restrictions on the use of the Late Closure strategy in Spanish. *Cognition*, 30(1):73–105.

Culy, C. (1985). The complexity of the vocabulary of Bambara. *Linguistics and Philosophy*, 8:345–351.

Curry, H. (1961). Some logical aspects of grammatical structure. In Jakobson, R., editor, *Structure of Language and its Mathematical Aspects*, volume 7 of *Proceedings of Symposia in Applied Mathematics*, pages 56–68. Providence: American Mathematical Society.

de Kuthy, K. and Meurers, W. D. (2001). On partial constituent fronting in German. *Journal of Comparative Germanic Linguistics*, 3:143–205.

Della Pietra, S., Della Pietra, V., and Lafferty, J. (1997). Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393.

Desmet, T., Brysbaert, M., and de Baecke, C. (2002). The correspondence between sentence production and corpus frequencies in modifier attachment. *The Quarterly Journal of Experimental Psychology*, 55A:879–896.

Desmet, T., De Baecke, C., and Drieghe, D. (2005). Relative clause attachment in Dutch: On-line comprehension corresponds to corpus frequencies when lexical variables are taken into account. *Language and Cognitive Processes*, 20. In Press.

Desmet, T. and Gibson, E. (2003). Disambiguation preferences and corpus frequencies in noun phrase conjunction. *Journal of Memory and Language*, 49(3):353–374.

Dienes, P. (2003). *Statistical Parsing with Non-local Dependencies.* PhD thesis, Saarland University.

Dienes, P. and Dubey, A. (2003a). Antecedent recovery: Experiments with a trace tagger. In *Proceedings of EMNLP*.

Dienes, P. and Dubey, A. (2003b). Deep processing by combining shallow methods. In *Proceedings of ACL*.

Dubey, A. (2004). *Statistical Parsing for German: Modeling Syntactic Properties and Annotation Differences.* PhD thesis, Saarland University.

Dubey, A. and Keller, F. (2003). Parsing German with sister-head dependencies. In *Proceedings of ACL*.

Earley, J. (1970). An efficient context-free parsing algorithm. *Communications of the ACM*, 13(2):94–102.

Elman, J. (1990). Finding structure in time. *Cognitive Science*, 14:179–211.

Fedorenko, E., Gibson, E., and Rodhe, D. (2004). Verbal working memory in sentence comprehension. In *Proceedings of the 26th annual meeting of the Cognitive Science Society*.

Frazier, L. (1979). *On Comprehending Sentences: Syntactic Parsing Strategies*. PhD thesis, University of Massachusetts.

Frazier, L. (1987). Sentence processing: A tutorial review. In Coltheart, M., editor, *Attention and Performance XII: The psychology of reading*. London: Erlbaum.

Frazier, L. and Fodor, J. D. (1978). The sausage machine: A new two-stage parsing model. *Cogntion*, 6(4):291–325.

Gazdar, G. (1985). Applicability of indexed grammars to natural languages. Technical Report CSLI-85-34, Center for the Study of Language and Information.

Gazdar, G., Klein, E., Pullum, G., and Sag, I. (1985). *Generalized Phrase Structure Grammar*. Harvard.

Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68:1–76.

Gibson, E. (2000). The dependency locality theory: A distance-based theory of linguistic complexity. In Marantz, A., Miyashita, Y., and O'Neil, W., editors, *Image, Language, Brain*, pages 95–126. MIT Press.

Gibson, E., Desmet, T., Grodner, D., Watson, D., and Ko, K. (2005a). Reading relative clauses in English. *Language and Cognitive Processes*, 16(2):313–353.

Gibson, E., Nakatani, K., and Chen, E. (2005b). Distinguishing theories of syntactic storage cost in sentence comprehension: Evidence from Japanese. To appear.

Gibson, E. and Pearlmutter, N. (1994). A corpus-based analysis of psycholinguistic constraints on PP attachment. In Clifton, C., Frazier, L., and Rayner, K., editors, *Perspectives on Sentence Processing*, pages 181–198. Erlbaum.

Gibson, E., Pearlmutter, N., Canseco-Gonzalez, E., and Hickok, G. (1996a). Recency preference in the human sentence processing mechanism. *Cognition*, 59(1):23–59.

Gibson, E., Schütze, C. T., and Salomon, A. (1996b). The relationship between the frequency and the complexity of linguistic structure. *Journal of Psycholinguistic Research*, 25:59–92.

Gibson, T. and Schütze, C. (1999). Disambiguation preferences in noun phrase conjunction do not mirror corpus frequency. *Journal of Memory and Language*, 40:263–279.

Gildea, D. and Jurafsky, D. (2002). Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Goetz, T. and Penn, G. (1997). A proposed linear specification language. Technical report, University of Tübingen.

Gordon, P. C., Hendrick, R., and Johnson, M. (2004). Effects of noun phrase type on sentence complexity. *Journal of Memory and Language*, 51(1):97–114.

Gregory, M. (2001). *Linguistic Informativeness and Speech Production: An Investigation of Contextual and Discourse-Pragmatic Effects on Phonological Variation*. PhD thesis, University of Colorado.

Grodner, D. and Gibson, E. (2005). Some consequences of the serial nature of linguistic input. *Cognitive Science*. In press.

Grodner, D., Watson, D., and Gibson, E. (2000). Locality effects on sentence processing. Presented at the 2003 annual CUNY Sentence Processing Conerence.

Haegeman, L. (1991). *Introduction to Government and Binding Theory.* Blackwell.

Haider, H. (1996). Downright down to the right. In Lutz, U. and Pafel, J., editors, *On Extraction and Extraposition in German*, pages 245–271. Amsterdam: John Benjamins.

Haider, H. (1997). Extraposition. In Beerman, D., LeBlanc, D., and v. Riemsdijk, H., editors, *Rightward Movement*, pages 115–151. Amsterdam: John Benjamins.

Hale, J. (2001). A probabilistic Earley parser as a psycholinguistic model. In *Proceedings of NAACL*, volume 2, pages 159–166.

Hale, J. (2003a). *Grammar, Uncertainty and Sentence Processing.* PhD thesis, John Hopkins University.

Hale, J. (2003b). The information conveyed by words in sentences. *Journal of Psycholinguistic Research*, 32(2):101–123.

Hale, J. (2004). The information-processing difficulty of incremental parsing. In Keller, F., Clark, S., Crocker, M., and Steedman, M., editors, *Proceedings of the ACL workshop on incremental parsing: bringing engineering and cognition together.*

Hall, K. and Johnson, M. (2004). Attention shifting for parsing speech. In *Proceedings of ACL.*

Harkema, H. (2001). *Parsing Minimalist Languages.* PhD thesis, UCLA.

Hawkins, J. (1994). *A Performance Theory of Order and Constituency.* Cambridge.

Hawkins, J. A. (2004). *Efficiency and Complexity in Grammars.* Oxford University Press.

Hemforth, B. (1993). *Kognitives Parsing: Repräsentation und Verarbeitung sprachlichen Wissens.* Sankt Augustin: Infix.

Hinrichs, E. and Nakazawa, T. (1994). Linearizing AUXs in German verbal complexes. In *German in Head-driven Phrase Structure Grammar*, number 46 in CSLI Lecture Notes, pages 11–38. CSLI.

Hockenmaier, J. (2003). *Data and models for Statistical Parsing with Combinatory Categorial Grammar*. PhD thesis, University of Edinburgh.

Jäger, F., Fedorenko, E., and Gibson, E. (2005). Dissociation between production and comprehension complexity. Poster Presentation at the 18th CUNY Sentence Processing Conference, University of Arizona.

Jelinek, F. and Lafferty, J. D. (1991). Computation of the probability of initial substring generation by stochastic context free grammars. *Computational Linguistics*, 17(3):315–323.

Jijkoun, V. and de Rijke, M. (2004). Enriching the output of a parser using memory-based learning. In *Proceedings of ACL*, pages 312–319.

Johnson, M. (1985). Parsing with discontinuous constituents. In *Proceedings of ACL*, volume 23.

Johnson, M. (2001). Joint and conditional estimation of tagging and parsing models. In *Proceedings of ACL*.

Johnson, M. (2002). A simple pattern-matching algorithm for recovering empty nodes and their antecedents. In *Proceedings of ACL*, volume 40.

Joshi, A. K. (1985). How much context-sensitivity is necessary for characterizing structural descriptions – Tree Adjoining Grammars. In Dowty, D., Karttunen, L., and Zwicky, A., editors, *Natural Language Processing – Theoretical, Computational, and Psychological Perspectives*. Cambridge.

Joshi, A. K., Levy, L. S., and Takahashi, M. (1975). Tree adjunct grammars. *Journal of Computer and System Sciences*, 10(1).

Jurafsky, D. (1996). A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive Science*, 20(2):137–194.

Jurafsky, D. (2003). Probabilistic modeling in psycholinguistics: Linguistic comprehension and production. In Bod, R., Hay, J., and Jannedy, S., editors, *Probabilistic Linguistics*. MIT Press.

Just, M. A. and Carpenter, P. A. (1980). A theory of reading: From eye fixations to comprehension. *Psychological Review*, 87:329–354.

Kaplan, R., Riezler, S., King, T. H., Maxwell, J. T., Vasserman, A., and Crouch, R. (2004). Speed and accuracy in shallow and deep stochastic parsing. In *Proceedings of NAACL*.

Kaplan, R. M. and Bresnan, J. (1982). Lexical-Functional Grammar: A formal system for grammatical representation. In Bresnan, J., editor, *The Mental Representation of Grammatical Relations*, pages 173–281. The MIT Press, Cambridge, MA. Reprinted in Mary Dalrymple, Ronald M. Kaplan, John Maxwell, and Annie Zaenen, eds., *Formal Issues in Lexical-Functional Grammar*, 29-130. Stanford: Center for the Study of Language and Information. 1995.

Kaplan, R. M. and Maxwell, J. T. (1993). The interface between phrasal and functional constraints. *Computational Linguistics*, 19(4):571–590.

Kasami, T. (1965). An efficient recognition and syntax algorithm for context-free languages. Technical Report AF-CRL-65-758, Air Force Cambridge Research Laboratory, Bedford, MA.

Kathol, A. (2000). *Linear Syntax*. Oxford: Oxford University Press.

Kathol, A. (2001). On the nonexistence of true parasitic gaps in standard German. In Culicover, P. and Postal, P., editors, *Parasitic Gaps*, number 35 in Current Studies in Linguistics, chapter 8. Cambridge, Mass: MIT Press.

Kathol, A. and Pollard, C. (1995). Extraposition via complex domain formation. In *Meeting of the Association for Computational Linguistics*, pages 174–180.

Kay, M. (1979). Functional grammar. In *Proceedings of the Berkeley Linguistics Society*.

Kay, M. (1980). Algorithm schemata and data structures in syntactic parsing. In *Proceedings of the Nobel Symposium on Text Processing*. Gothenburg.

Keller, F. (2004). The entropy rate principle as a predictor of processing effort: An evaluation against eye-tracking data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 317–324, Barcelona.

Kennedy, A., Hill, R., and Pynte, J. (2003). The Dundee corpus. Poster presented at ECEM 12, Dundee, Scotland.

Kiss, T. (2005). Semantic constraints on relative clause extraposition. *Natural Language and Linguistic Theory*, 23(2):281–334.

Klein, D. and Manning, C. D. (2002). Conditional structure versus conditional estimation in NLP models. In *Proceedings of ACL*.

Klein, D. and Manning, C. D. (2003a). A* parsing: Fast exact Viterbi parse selection. In *Proceedings of HLT-NAACL 03*.

Klein, D. and Manning, C. D. (2003b). Accurate unlexicalized parsing. In *Proceedings of ACL*.

Knuth, D. (1977). A generalization of Dijkstra's algorithm. *Information Processing Letters*, 6(1):1–5.

Konieczny, L. (1996). *Human sentence processing: a semantics-oriented parsing approach*. PhD thesis, Universität Freiburg.

Konieczny, L. (2000). Locality and parsing complexity. *Journal of Psycholinguistic Research*, 29(6):627–645.

Konieczny, L. and Döring, P. (2003). Anticipation of clause-final heads: Evidence from eye-tracking and SRNs. In *Proceedings of ICCS/ASCS*.

Kruijff, G.-J. (2002). Learning linearization rules from treebanks. Invited talk at the Formal Grammar'02/COLOGNET-ELSNET Symposium.

Levy, R. and Manning, C. (2004). Deep dependencies from context-free statistical parsers: correcting the surface dependency approximation. In *Proceedings of ACL*.

MacDonald, M. C. (1993). The interaction of lexical and syntactic ambiguity. *Journal of Memory and Language*, 32:692–715.

MacDonald, M. C., Pearlmuttter, N. J., and Seidenberg, M. S. (1994). Lexical nature of syntactic ambiguity resolution. *Psychological Review*, 101(4):676–703.

Magerman, D. M. (1994). *Natural Language Parsing as Statistical Pattern Recognition*. PhD thesis, Stanford.

Marslen-Wilson, W. (1975). Sentence perception as an interactive parallel process. *Science*, 189:382–386.

Maxwell, J. T. and Manning, C. D. (1996). A theory of non-constituent coordination based on finite-state rules. In Butt, M. and King, T. H., editors, *Proceedings of LFG*.

McDonald, S. A. and Shillcock, R. C. (2003). Low-level predictive inference in reading: The influence of transitional probabilities on eye movements. *Vision Research*, 43:1735–1751.

McRae, K., Spivey-Knowlton, M. J., and Tanenhaus, M. K. (1998). Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. *Journal of Memory and Language*, 38(3):283–312.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63:81–97.

Miller, G. A. and Chomsky, N. (1963). Finitary models of language users. In Luce, D. R., Bush, R. R., and Galanter, E., editors, *Handbook of Mathematical Psychology*, volume II. New York: John Wiley.

Miller, P. (2000). *Strong Generative Capacity: The Semantics of Linguistic Formalism*. Cambridge.

Mitchell, D. and Brysbaert, M. (1998). Challenges to recent theories of language differences in parsing: evidence from Dutch. In Hillert, D., editor, *Sentence Processing: A Crosslinguistic Perspective*, volume 31 of *Syntax and Semantics*, pages 313–336. Academic Press.

Mitchell, D. C. (1994). Sentence parsing. In Gernsbacher, M., editor, *Handbook of Psycholinguistics*. Academic Press.

Mitchell, D. C. and Cuetos, F. (1991). The origins of parsing strategies. In Smith, C., editor, *Current issues in natural language processing*. Austin: University of Texas.

Mitchell, D. C., Cuetos, F., Corley, M., and Brysbaert, M. (1995). Exposure-based models of human parsing: Evidence for the use of coarse-grained (nonlexical) statistical records. *Journal of Psycholinguistic Research*, 24:469–488.

Müller, S. (1999). *Deutsche Syntax deklarativ. Head-Driven Phrase Structure Grammar für das Deutsche*. Number 394 in Linguistische Arbeiten. Max Niemeyer Verlag, Tübingen.

Nakatani, K. and Gibson, E. (2003). An on-line study of Japanese nesting complexity. Presented at the 2003 annual CUNY Sentence Processing Conerence.

Narayanan, S. and Jurafsky, D. (1998). Bayesian models of human sentence processing. In *Proceedings of the Twelfth Annual Meeting of the Cognitive Science Society*.

Narayanan, S. and Jurafsky, D. (2002). A Bayesian model predicts human parse preference and reading time in sentence processing. In *Advances in Neural Information Processing Systems*, volume 14, pages 59–65.

Nederhof, M.-J. (2003). Weighted deductive parsing and Knuth's algorithm. *Computational Linguistics*, 29(1):135–143.

Plaehn, O. (2000). Computing the most probable parse for a discontinuous phrase structure grammar. In *Proceedings of IWPT*, Trento, Italy.

Pollard, C. (1984). *Generalized Phrase Structure Grammars, Head Grammars, and Natural Languages*. PhD thesis, Stanford.

Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. Chicago: University of Chicago Press and Stanford: CSLI Publications.

Race, D. S. and MacDonald, M. C. (2003). The use of "that" in the production and comprehension of object relative clauses. In *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, 124(3):372–422.

Rayner, K., Sereno, S. C., and Raney, G. E. (1996). Eye movement control in reading: A comparison of two types of models. *Journal of Experimental Psychology: Human Perception and Performance*, 22:1188–1200.

Reape, M. (1994). Domain union and word order variation in German. In Nerbonne, J., Netter, K., and Pollard, C., editors, *German in Head-driven Phrase Structure Grammar*, number 46 in CSLI Lecture Notes, pages 151–198. CSLI.

Riezler, S., King, T. H., Kaplan, R. M., Crouch, R. S., Maxwell, J. T., and Johnson, M. (2002). Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *Proceedings of ACL*, pages 271–278.

Roark, B. and Johnson, M. (1999). Efficient probabilistic top-down and left-corner parsing. In *Proceedings of ACL*.

Roland, D. and Jurafsky, D. (2002). Verb sense and verb subcategorization probabilities. In Merlo, P. and Stevenson, S., editors, *The Lexical Basis of Sentence Processing: Formal, Computational, and Experimental Issues*, pages 325–345. John Benjamins.

Ross, J. R. (1967). *Constraints on Variables in Syntax*. PhD thesis, MIT.

Schlesewsky, M., Fanselow, G., Kliegl, R., and Krems, J. (2000). The subject preference in the processing of locally ambiguous WH-questions in German. In Hemforth, B. and Konieczny, L., editors, *German Sentence Processing*. Kluwer.

Seki, H., Matsumura, T., Fujii, M., and Kasami, T. (1991). On multiple context-free grammars. *Theoretical Computer Science*, 88:191–229.

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

Shieber, S. M., Schabes, Y., and Pereira, F. C. N. (1995). Principles and implementation of deductive parsing. *Journal of Logic Programming*, 24(1–2):3–36.

Skut, W., Brants, T., Krenn, B., and Uszkoreit, H. (1997a). Annotating unrestricted German text. In *Fachtagung der Sektion Computerlinguistik der Deutschen Gesellschaft für Sprachwissenschaft*, Heidelberg, Germany.

Skut, W., Krenn, B., Brants, T., and Uszkoreit, H. (1997b). An annotation scheme for free word order languages. In *Proceedings of ANLP*, Washington, DC.

Stabler, E. P. (1997). Derivational minimalism. In Retoré, C., editor, *Logical Aspects of Computational Linguistics*, pages 68–95. Springer.

Steedman, M. (2000). *The Syntactic Process*. Cambridge, MA: MIT Press.

Stolcke, A. (1995). An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. *Computational Linguistics*, 21(2):165–201.

Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K., and Sedivy, J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, 268:1632–1634.

Toutanova, K., Manning, C. D., Flickinger, D., and Oepen, S. (2005). Stochastic HPSG parse disambiguation using the Redwoods corpus. *Research on Logic and Computation*. To appear.

Uszkoreit, H., Brants, T., Duchier, D., Krenn, B., Konieczny, L., Oepen, S., and Skut, W. (1998). Studien zur performzorientieren Linguistik: Aspekte der Relativsatzextraposition im Deutschen. *Kognitionswissenschaft*, 7:129–133.

Vasishth, S. (2002). *Working memory in sentence comprehension: Processing Hindi center embeddings*. PhD thesis, Ohio State University.

Vasishth, S. (2003). Quantifying processing difficulty in human sentence parsing: The role of decay, activation, and similarity-based interference. In *Proceedings of EuroCogSci*.

Vasishth, S. and Lewis, R. (2003). The role of decay and activation in human sentence processing. Presented at the 2003 AMLaP Conference, Glasgow, Scotland.

Vijay-Shanker, K., Weir, D. J., and Joshi, A. K. (1987). Characterizing structural descriptions produced by various grammatical formalisms. In *Proceedings of ACL*.

Warren, T. and Gibson, E. (2002). The influence of referential processing on sentence complexity. *Cognition*, 85(1):79–112.

Weir, D. J. (1988). *Characterizing mildly context-sensitive grammar formalisms*. PhD thesis, University of Pennsylvania.

Younger, D. H. (1967). Recognition and parsing of context-free languages in time n^3. *Information and Control*, 10(2):189–208.

Zwicky, A. (1986). Concatenation and liberation. In *Papers from the 22nd Regional Meeting of the Chicago Linguistic Society*, pages 65–74.